IBM SPSS Statistics Algorithms

Introduction to Algorithms

Throughout much of the documentation, we avoid detailed discussion of the inner workings of procedures in order to promote readability. This algorithms document is designed as a resource for those interested in the specific calculations performed by procedures.

Algorithms Used in Multiple Procedures

For some statistics, such as the significance of a *t* test, the same algorithms are used in more than one procedure. Another example is the group of post hoc tests that are used in ONEWAY and GLM. You can find algorithms for these tests in the appendixes.

Choice of Formulas

Starting with the first statistics course, students learn that there are often several equivalent ways to compute the same quantity. For example, the variance can be obtained using either of the following formulas:

$$s^2 = \sum_{i=1}^{N} (x_i - \overline{x})^2 / (N-1)$$

$$s^2 = \left(\sum_{i=1}^{N} x_i^2 - \left(\sum_{i=1}^{N} x_i \right)^2 / N \right) / (N-1)$$

Since the formulas are algebraically equal, the one preferred is often the one easier to use (or remember). For small data sets consisting of “nice” numbers, the arbitrary choice of several computational methods is usually of little consequence. However, for handling large data sets or “troublesome” numbers, the choice of computational algorithms can become very important, even when those algorithms are to be executed by a computer. Care must be taken to choose an algorithm that produces accurate results under a wide variety of conditions without requiring extensive computer time. Often, these two considerations must be balanced against each other.

You may notice that the same statistic is computed differently in various routines. Among the reasons for this are the precision of the calculations and the desirability of computing many statistics in the same procedure. For example, in the paired *t* test procedure (T-TEST), the need to compute both the correlation coefficient and the standard error of the difference led to the selection of a different algorithm than would have been chosen for computation of only the standard error. Throughout the years of development, the personal preferences of many designers and programmers have also influenced the choice of algorithms. Now, as new routines are added and old ones are updated, any unnecessary diversity is being replaced by a consistent core of algorithms.

Missing Values

Since similar options for treatment of missing values are available in many procedures, treatment of missing values has often not been specified in each chapter. Instead, the following rules should be used:

- If listwise deletion has been specified and a missing value is encountered, the case is not included in the analysis. This is equivalent, for purposes of following the algorithms, to setting the case weight to zero.
- If variable-wise deletion is in effect, a case has zero weight when the variable with missing values is involved in computations.
- If pairwise deletion has been specified, a case has zero weight when one or both of a pair of variables included in a computation is missing.
- If missing-values declarations are to be ignored, all cases are always included in the computation.

It should be noted again that the computer routines do not alter case weights for cases with missing data but, instead, actually exclude them from the computations. Missing-value specifications do not apply when a variable is used for weighting. All values are treated as weights.

2SLS Algorithms

2SLS produces the two-stage least-squares estimation for a structure of simultaneous linear equations.

Notation

The following notation is used throughout this chapter unless otherwise stated:

p	Number of predictors
p_1	Number of endogenous variables among p predictors
p_2	Number of non-endogenous variables among p predictors
k	Number of instrument variables
n	Number of cases
$\mathbf{y}$	$n\times 1$ vector which consists of a sample of the dependent variable
$\mathbf{Z}$	$n\times p$ matrix which represents observed predictors
$\boldsymbol{\beta}$	$p\times 1$ parameter vector
$\mathbf{X}$	$n\times 1$ matrix with element x_{ij}, which represents the observed value of the j^{th} instrumental variable for case i.
$\mathbf{Z}_1$	Submatrix of $\mathbf{Z}$ with dimension $n\times p_1$, which represents observed endogenous variables
$\mathbf{Z}_2$	Submatrix of $\mathbf{Z}$ with dimension $n\times p_2$, which represents observed non-endogenous variables
$\boldsymbol{\beta}_1$	Subvector of $\boldsymbol{\beta}$ with parameters associated with $\mathbf{Z}_1$
$\boldsymbol{\beta}_2$	Subvector of $\boldsymbol{\beta}$ with parameters associated with $\mathbf{Z}_2$

Model

The structure equations of interest are written in the form:

$$\mathbf{y} = \mathbf{Z}\beta = [\mathbf{Z}_1, \mathbf{Z}_2]\begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix} + \epsilon$$

$$\mathbf{Z}_1 = \mathbf{X}\gamma + \delta$$

where

$$\mathbf{Z} = [\mathbf{Z}_1, \mathbf{Z}_2],\ \beta = \begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix}$$

and ϵ and δ are the disturbances with zero means and covariance matrices $\sigma^2\mathbf{I}_n$ and $\zeta^2\mathbf{I}_n$, respectively.

Estimation

The estimation technique used was developed by Theil; (Theil, 1953), (Theil, 1953). First premultiply both sides of the model equation by $\mathbf{X}'$ to obtain

$$\mathbf{X}'\mathbf{y} = \mathbf{X}'\mathbf{Z}\beta + \mathbf{X}'\epsilon$$

Since the disturbance vector has zero mean and covariance matrix $\sigma^2\left(\mathbf{X}'\mathbf{X}\right)$, then $\left(\mathbf{X}'\mathbf{X}\right)^{-\frac{1}{2}}\mathbf{X}'\epsilon$ would have a covariance matrix $\sigma^2\mathbf{I}_n$. Thus, multiplying $\left(\mathbf{X}'\mathbf{X}\right)^{-\frac{1}{2}}$ to both sides of the above equation results in a multiple linear regression model

$$\left(\mathbf{X}'\mathbf{X}\right)^{-\frac{1}{2}}\mathbf{X}'\mathbf{y} = \left(\mathbf{X}'\mathbf{X}\right)^{-\frac{1}{2}}\mathbf{X}'\mathbf{Z}\beta + \left(\mathbf{X}'\mathbf{X}\right)^{-\frac{1}{2}}\mathbf{X}'\epsilon$$

The ordinary least-square estimator $\hat{\beta}$ for β is

$$\hat{\beta} = \left(\mathbf{Z}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{Z}\right)^{-1}\mathbf{Z}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{y}$$

Computational Details

- 2SLS constructs a matrix **R**,

$$\mathbf{R} = \begin{bmatrix} \mathbf{1} & \mathbf{V}' \\ \mathbf{V} & \mathbf{M} \end{bmatrix}$$

where

$$\mathbf{M} = \mathbf{C}_{zx}(\mathbf{C}_{xx})^{-1}\mathbf{C}'_{\boldsymbol{zx}}$$
$$\mathbf{V} = \mathbf{C}_{zx}(\mathbf{C}_{xx})^{-1}\mathbf{C}'_{\boldsymbol{xy}}$$

and $\mathbf{C}_{zx}$ is the correlation matrix between **Z** and **X**, and $\mathbf{C}_{xx}$ is the correlation matrix among instrumental variables.

- Sweep the matrix **R** to obtain regression coefficient estimate for β.

- Compute sum of the squares of residuals (SSE) by

$$\mathbf{y}'\mathbf{y} - \mathbf{u}\mathbf{Z}'\mathbf{y} - \mathbf{y}'\mathbf{Z}\mathbf{u}' + \mathbf{u}\mathbf{Z}'\mathbf{Z}\mathbf{u}'$$

where

$$\mathbf{u} = \mathbf{y}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{z}\left[\mathbf{z}'\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{z}\right]^{-1}$$

- Compute the statistics for the ANOVA table and for variables in the equation. For more information, see the topic “REGRESSION Algorithms”.

References

Theil, H. 1953. *Repeated least square applied to complete equation systems*. Netherlands: The Hague: Central Planning Bureau.

Theil, H. 1953. *Estimation and simultaneous correlation in complete equation systems.* Netherlands: The Hague: Central Planning Bureau.

ACF/PACF Algorithms

Procedures ACF and PACF print and plot the sample autocorrelation and partial autocorrelation functions of a series of data.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 3-1
Notation

Notation	Description
x_i	*i*th observation of input series, *i*=1,...,*n*
r_k	*k*th lag sample autocorrelation
$\hat{\phi}_{kk}$	*k*th lag sample partial autocorrelation

Basic Statistics

The following formulas are used if no missing values are encountered. If missing values are present, see “Series with Missing Values” for modification of some formulas.

Sample Autocorrelation

$$r_k = \frac{\sum_{i=1}^{n-k} (x_i - \overline{x})(x_{i+k} - \overline{x})}{\sum_{i=1}^{n} (x_i - \overline{x})^2}$$

where $\overline{x}$ is the average of the *n* observations.

Standard Error of Sample Autocorrelation

There are two formulas for the standard error of r_k based on different assumptions about the autocorrelation. Under the assumption that the true MA order of the process is *k*−1, the approximate variance of r_k(Bartlett, 1946) is:

$$var(r_k) \simeq \frac{1}{n}\left(1 + 2\sum_{l=1}^{k-1} r_l^2\right)$$

The standard error is the square root (Box and Jenkins, 1976), p. 35. Under the assumption that the process is white noise,

$$var(r_k) \simeq \frac{1}{n}\left(\frac{n-k}{n+2}\right)$$

Box-Ljung Statistic

At lag *k*, the Box-Ljung statistic is defined by

$$Q_k = n(n+2)\sum_{l=1}^{k} \frac{r_l^2}{n-l}$$

When *n* is large, Q_k has a chi-square distribution with degrees of freedom *k*−*p*−*q*, where *p* and *q* are autoregressive and moving average orders, respectively. The significance level of Q_k is calculated from the chi-square distribution with *k*−*p*−*q* degrees of freedom.

Sample Partial Autocorrelation

$$\hat{\phi}_{11} = r_1$$

$$\hat{\phi}_{22} = \left(r_2 - r_1^2\right)/\left(1 - r_1^2\right)$$

$$\hat{\phi}_{kj} = \hat{\phi}_{k-1,j} - \phi_{kk}\phi_{k-1,k-j} k = 2,\ldots,j = 1,2,\ldots,k-1$$

$$\hat{\phi}_{kk} = \left(r_k - \sum_{j=1}^{k-1} \phi_{k-1,j} r_{k-j}\right) / \left(1 - \sum_{j=1}^{k-1} \phi_{k-1,j} r_j\right), k = 3,\ldots$$

Standard Error of Sample Partial Autocorrelation

Under the assumption that the AR(*p*) model is correct and $p \le k-1$,

$\hat{\phi}_{kk} \cong N\left(0, \frac{1}{n}\right)$(Quenouville, 1949)

Thus

$$var\left(\hat{\phi}_{kk}\right) \cong \frac{1}{n}$$

Series with Missing Values

If there are missing values in *x*, the following statistics are computed differently (Cryer, 1986). First, define

$\overline{x} =$ average of nonmissing $x_1,\ldots,x_n$,

$$a_i = \begin{cases} x_i - \overline{x}, & \text{if } x_i \text{ is not missing} \\ \text{SYSMIS}, & \text{if } x_i \text{ is missing} \end{cases}$$

for *k*=0,1,2,..., and *j*=1,...,*n*

$$b_j^{(k)} = \begin{cases} a_j a_{j+k}, & \text{if both are not missing} \\ \text{SYSMIS}, & \text{otherwise} \end{cases}$$

$m_k =$ the number of nonmissing values in $b_1^{(k)},\ldots,b_{n-k}^{(k)}$

$m_0 =$ the number of nonmissing values in x

Sample Autocorrelation

$$r_k = \frac{\text{sum of nonmissing } b_1^{(k)},\ldots,b_{n-k}^{(k)}}{\text{sum of nonmissing } b_1^{(0)},\ldots,b_n^{(0)}}$$

Standard Error of Sample Autocorrelation

$$se(r_k) = \sqrt{\frac{1}{m_0}\left(1+\sum_{l=1}^{k-1} r_l^2\right)} \quad \text{(MA assumption)}$$

$$se(r_k) = \sqrt{\frac{m_k}{(m_0+2)m_0}} \quad \text{(white noise)}$$

Box-Ljung Statistic

$$Q = m_0(m_0+2)\sum_{l=1}^{k} \frac{r_l^2}{m_l}$$

Standard Error of Sample Partial Autocorrelation

$$se\left(\hat{\phi}_{kk}\right) \cong \sqrt{\frac{1}{m_0}}$$

References

Bartlett, M. S. 1946. On the theoretical specification of sampling properties of autocorrelated time series. *Journal of Royal Statistical Society, Series B*, 8, 27–27.

Box, G. E. P., and G. M. Jenkins. 1976. *Time series analysis: Forecasting and control*, Rev. ed. San Francisco: Holden-Day.

Cryer, J. D. 1986. *Time series analysis*. Boston, Mass.: Duxbury Press.

Quenouville, M. H. 1949. Approximate tests of correlation in time series. *Journal of the Royal Statistical Society, Series B*, 11, 68–68.

AIM Algorithms

The Attribute Importance (AIM) procedure performs tests to find out if the groups are homogeneous.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 4-1
Notation

Notation	Description
G	Number of groups.
C	Number of categories in the categorical variable.
n_{ij}	Number of cases in the *j*th category in the *i*th group, *i* = 1, …, *G* and *j* = 1, …, *C*. Assume that $n_{ij} \geq 0$.
n_i	Number of cases in the *i*th group. $n_i = \Sigma_{j=1}^{C} n_{ij}$
n	Overall number of cases. $n = \Sigma_{i=1}^{G} n_i$. Assume *n*>0.
p_j	Overall proportion of cases in the *j*th category. $p_j = \frac{1}{n} \Sigma_{i=1}^{G} n_{ij}$
$\overline{x}_i$	Mean of the continuous variable in the *i*th group.
s_i	Standard deviation of the continuous variable in the *i*th group. Assume that $s_i \geq 0$.
$\overline{x}$	Overall mean of the continuous variable. $\overline{x} = \frac{1}{n} \sum_{i=1}^{G} n_i x_i$

Test of Homogeneity of Proportions

This test is performed only for categorical variables. The null hypothesis is that the proportion of cases in the categories in the *i*th group is the same as the overall proportion. If *C* > 1, the Chi-square statistic for the *i*th group is computed as follows:

$$\chi^2 = \sum_{j=1}^{C} \frac{(n_{ij} - n_i p_j)^2}{n_i p_j}$$

The degrees of freedom is *C*−1. The significance is the probability that a Chi-square random variate with this degrees of freedom will have a value greater than the χ^2 statistic.

If *C*<1, the Chi-square statistic is always 0 with zero degrees of freedom, and the significance value is undefined.

Test of Equality of Means

This test is performed only for continuous variables. The null hypothesis is that the mean (of a continuous variable) in the *i*th group is the same as the overall mean. If $n_i > 1$ and $s_i > 0$, the Student's *t* statistic for the *i*th group is computed as follows:

$$t = \frac{(\overline{x}_i - \overline{x})}{s_i/\sqrt{n_i}}$$

The degrees of freedom is $n_i - 1$. The significance is the probability that a Student's *t* random variate with this degrees of freedom will have a value greater than the *t* statistic.

When $n_i > 1$ but $s_i = 0$, this implies that the continuous variable is constant in the *i*th group. In this case, the Student's *t* statistic is infinity with positive degrees of freedom $n_i - 1$, and the significance value is zero.

If $n_i \leq 1$, then s_i is undefined. In this case, the Student's *t* statistic is undefined, the degrees of freedom is 0, and the significance value is undefined.

Graphical Display of Significance

Since significance values are often very small numbers, the negative common logarithm $-\log_{10}$) of significance values are displayed instead in the bar charts.

ALSCAL Algorithms

ALSCAL attempts to find the structure in a set of distance measures between objects or cases.

Initial Configuration

The first step is to estimate an additive constant c_k, which is added to the observed proximity measures (for example, o_{ijk}). Thus,

$$o^*_{ijk} = o_{ijk} + c_k$$

such that for all triples the triangular inequality holds:

$$o^*_{ijk} + o^*_{jlk} \geq o^*_{ilk}$$

and positivity holds $o^*_{ijk} \geq 0$

where

Table 5-1
Notation

Notation	Description
o^*_{ijk}	is the adjusted proximity between stimulus *i* and stimulus *j* for subject *k*
o^*_{jlk}	is the adjusted proximity between stimulus *j* and stimulus *l* for subject *k*
o^*_{ilk}	is the adjusted proximity between stimulus *i* and stimulus *l* for subject *k*

The constant c_k, which is added, is as small as possible to estimate a zero point for the dissimilarity data, thus bringing the data more nearly in line with the ratio level of measurement. This step is necessary to make the $\mathbf{B}^*_k$ matrix, described below, positive semidefinite (that is, with no imaginary roots).

The next step is to compute a scalar product matrix $\mathbf{B}^{**}_k$ for each subject *k* by double centering $\mathbf{O}^*_k$, the adjusted proximity matrix for each subject. An element of the $\mathbf{B}^{**}_k$ matrix b^{**}_{ijk} is computed as follows:

$$b^{**}_{ijk} = -\frac{1}{2}\left(o^{*2}_{ijk} - o^{*2}_{i.k} - o^{*2}_{.jk} + o^{*2}_{..k}\right)$$

where

Table 5-2
Notation

Notation	Description
$o^*_{i.k}$	are the row means for the adjusted proximities for subject *k*
$o^*_{.jk}$	are the column means for the adjusted proximities for subject *k*
$o^*_{..k}$	is the grand mean for subject *k*

Double centering to convert distances to scalar products is necessary because a scalar products matrix is required to compute an initial configuration using the Young-Householder-Torgerson procedure.

Next the individual subject matrices are normalized so that they have the same variance. The normalized matrix $\mathbf{B}_k^*$ is found for each subject. The elements of the matrix are

$$b_{ijk}^* = \frac{b_{ijk}^{**}}{\left[\sum_i \sum_j \left(b_{ijk}^{**}\right)^2 / (n(n-1))\right]^{1/2}}$$

where *n* is the number of stimuli, and $n(n-1)$ is the number of off-diagonal elements in the $\mathbf{B}_k^{**}$ matrix. The denominator is both the root mean square and the standard deviation of the unnormalized scalar products matrix $\mathbf{B}^{**}$ (It is both because $b_{..k}^{**} = 0$, due to double centering.) $\mathbf{B}_k^*$ is thus a matrix with elements b_{ijk}^*, which are scalar products for individual subject *k*. Normalization of individual subjects' matrices equates the contribution of each individual to the formation of a mean scalar products matrix and thus the resulting initial configuration.

Next an average scalar products matrix $\mathbf{B}^*$ over the subjects is computed. The elements of this matrix are

$$b_{ij.}^* = \frac{\sum_k b_{ijk}^*}{m}$$

where *m* is the number of subjects.

The average $\mathbf{B}^*$ matrix used in the previous step is used to compute an initial stimulus configuration using the classical Young-Householder multidimensional scaling procedure

$$\mathbf{B}^* = \mathbf{X}\mathbf{X}^{'}$$

where **X** is an $n \times r$ matrix of *n* stimulus points on *r* dimensions, and $\mathbf{X}^{'}$ is the transpose of the **X** matrix; that is, the rows and columns are interchanged. The **X** matrix is the initial configuration.

For the weighted ALSCAL matrix model, initial weight configuration matrices $\mathbf{W}_k$ for each of the *m* subjects are computed. The initial weight matrices $\mathbf{W}_k$ are $r \times r$ matrices, where *r* is the number of dimensions. Later the diagonals of $\mathbf{W}_k$ will form rows of the **W** matrix, which is an $n \times r$ matrix. The matrices $\mathbf{W}_k$ are determined such that $\mathbf{B}_k^* = \mathbf{Y}\mathbf{W}_k\mathbf{Y}^{'}$, where $\mathbf{Y} = \mathbf{X}\mathbf{T}$ and $\mathbf{T}\mathbf{T}^{'} = \mathbf{I}$ and where **T** is an orthogonal rotation of the configuration **X** to a new orientation **Y**. **T** is computed by the Schönemann-de Leeuw procedure discussed by Young, Takane, and Lewyckyj (Young, Takane, and Lewyckyj, 1978). **T** rotates **X** so that $\mathbf{W}_k$ is as diagonal as possible (that is, off-diagonal elements are as close to zero as possible on the average over subjects). Off-diagonal elements represent a departure from the model (which assumes that subjects weight only the dimensions of the stimulus space).

Optimization Algorithm

The optimization algorithm is a series of steps which are repeated (iterated) until the final solution is achieved. The steps for the optimization algorithm are performed successively because disparities, weights, and coordinates cannot be solved for simultaneously.

Distance

Distances are computed according to the weighted Euclidean model

$$d_{ijk}^2 = \sum_{a=1}^{r} w_{ka}(x_{ia} - x_{ja})^2$$

where

Table 5-3
Notation

Notation	**Description**
w_{ka}	is the weight for subject *k* on a dimension *a*,
x_{ia}	is the coordinate of stimulus *i* on dimension *a*,
x_{ja}	is the coordinate of stimulus *j* on dimension *a*.

The first set of distances is computed from the coordinates and weights found in the previous steps. Subsequently, new distances are computed from new coordinates found in the iteration process (described below).

Optimal Scaling

Optimal scaling for ordinal data use Kruskal’s least-squares monotonic transformation. This yields disparities that are a monotonic transformation of the data and that are as much like the distances (in a least squares sense) as possible. Ideally, we want the distances to be in exactly the same rank order as the data, but usually they are not. So we locate a new set of numbers, called disparities, which are in the same rank order as the data and which fit the distances as well as possible. When we see an order violation we replace the numbers that are out of order with a block of values that are the mean of the out-of-order numbers. When there are ties in the data, the optimal scaling process is changed somewhat. Kruskal’s primary and secondary procedures are used in ALSCAL.

Normalization

The disparities computed previously are now normalized for technical reasons related to the alternating least squares algorithm (Takane, Young, and de Leeuw, 1977). During the course of the optimization process, we want to minimize a measure of error called SSTRESS. But the monotone regression procedure described above only minimizes the numerator of the SSTRESS formula. Thus, the formula below is applied to readjust the length of the disparities vector so that SSTRESS is minimized:

$$\mathbf{D}_k^{*N} = \mathbf{D}_k^{*}\left(\mathbf{D}_k{}'\mathbf{D}_k\right)\left(\mathbf{D}_k{}'\mathbf{D}_k^{*}\right)^{-1}$$

where

Table 5-4
Notation

Notation	Description
$\mathbf{D}_k^{*}$	is a column vector with $\frac{n(n-1)}{2}$ elements containing all the disparities for subject *k*,
$\mathbf{D}_k$	is a column vector with $\frac{n(n-1)}{2}$ elements containing all the distances for subject *k*,
$\mathbf{D}_k{}'\mathbf{D}_k$	is the sum of the squared distances,
$\mathbf{D}_k{}'\mathbf{D}_k^{*}$	is the sum of the cross products.

The normalized disparities vector $\mathbf{D}_k^{*N}$ is a conditional least squares estimate for the distances; that is, it is the least squares estimate for a given iteration. The previous $\mathbf{D}^{*}$ values are replaced by $\mathbf{D}^{*N}$ values, and subsequent steps utilize the normalized disparities.

SSTRESS

The Takane-Young-de Leeuw formula is used:

$$SSTRESS(1) = S = \left[\frac{1}{m}\sum_{k=1}^{m}\left[\frac{\sum_i\sum_j\left(d_{ijk}^2 - d_{ijk}^{*2}\right)^2}{\sum_i\sum_j d_{ijk}^{*4}}\right]\right]^{1/2}$$

where d_{ijk}^{*} values are the normalized disparity measures computed previously, and d_{ijk} are computed as shown above. Thus SSTRESS is computed from the normalized disparities and the previous set of coordinates and weights.

Termination

The current value of SSTRESS is compared to the value of SSTRESS from the previous iteration. If the improvement is less than a specified value (default equals 0.001), iteration stops and the output stages has been reached. If not, the program proceeds to the next step. (This step is skipped on the first iteration.)

Model Estimation

In ALSCAL the weights and coordinates cannot be solved for simultaneously, so we do it successively. Thus, the model estimation phase consists of two steps: (i) estimation of subject weights and (ii) estimation of stimulus coordinates.

(i) Estimation of subject weights. (This step is skipped for the simple, that is, unweighted, Euclidean model.)

A conditional least-squares estimate of the weights is computed at each iteration:

$$\mathbf{W} \simeq \mathbf{D}^{*}\mathbf{P}\left(\mathbf{P}^{'}\mathbf{P}\right)^{-1}$$

The derivation of the computational formula is as follows:

We have found disparities such that

$$d^{*}_{ijk} \simeq d^{2}_{ijk},$$

where

$$d^{2}_{ijk} = \sum_{a=1}^{r} w_{ka}(x_{ia} - x_{ja})^{2}$$

Let p_{ija} be the unweighted distance between stimuli *i* and *j* as projected onto dimension *a*, that is,

$$p_{ija} = (x_{ia} - x_{ja})^{2}.$$

Then

$$d^{*2}_{ijk} \simeq d^{2}_{ijk} = \sum_{a=1}^{r} w_{ka}p_{ija}.$$

In matrix notation, this is expressed as $\mathbf{D}^{*} = \mathbf{W}\mathbf{P}^{'}$, where $\mathbf{D}^{*}$ is now an $m \times \frac{n(n-1)}{2}$ matrix having one row for every subject and one column for each stimulus pair; **W** is an $m \times r$ matrix having one row for every subject and one column for each dimension; and $\mathbf{P}^{'}$ has one row for every dimension and one column for every stimulus pair.

We wish to solve for **W**, $\mathbf{W}\mathbf{P}^{'} \simeq \mathbf{D}^{*}$, which we do by noting that

$$\mathbf{W}\mathbf{P}^{'}\mathbf{P}\left(\mathbf{P}^{'}\mathbf{P}\right)^{-1} = \mathbf{D}^{*}\mathbf{P}\left(\mathbf{P}^{'}\mathbf{P}\right)^{-1}.$$

Therefore,

$$\mathbf{W} = \mathbf{D}^{*}\mathbf{P}\left(\mathbf{P}^{'}\mathbf{P}\right)^{-1}$$

and we have the conditional least squares estimate for **W**. We have in fact minimized SSTRESS at this point relative to the previously computed values for stimulus coordinates and optimal scaling. We replace the old subject weights with the newly estimated values.

(ii) Estimation of Stimulus Coordinates. The next step is to estimate coordinates, one at a time, using the previously computed values for $\mathbf{D}^*$ (disparities) and weights. Coordinates are determined one at a time by minimizing SSTRESS with regard to a given coordinate. Equation **(2)** allows us to solve for a given coordinate x_{le}:

$$\frac{\partial S}{\partial x_{le}} = \frac{1}{m}\Sigma c_k \frac{\partial S_k}{\partial x_{le}} \qquad (\mathbf{1})$$

$$\frac{\partial S_k}{\partial x_{le}} = 4w_{ke}^2 \sum_j \left(x_{le}^3 - 3x_{le}^2 x_{je} + 2x_{le}x_{je}^2 - b_{ljk}^2 x_{le} + b_{ljk}^2 x_{je}\right) \qquad (\mathbf{2})$$

Equation **(2)** can be substituted back into equation **(1)**. This equation with one unknown, x_{le}, is then set equal to zero and solved by standard techniques. All the other coordinates except x_{le} are assumed to be constant while we solve for x_{le}.

Immediately upon solving for x_{le}, we replace the value for x_{le} used on the previous iteration with the newly obtained value, and then proceed to estimate the value for another coordinate. We successively obtain values for each coordinate of point *l*, one at a time, replacing old values with new ones. This continues for point *l* until the estimates stabilize. We then move to a new point and proceed until new coordinates for all stimuli are estimated. We then return to the beginning of the optimization algorithm (the previous step above) and start another iteration.

References

Bloxom, B. 1978. Contrained multidimensional scaling in n spaces. *Psychometrika*, 43, 397–408.

Carroll, J. D., and J. J. Chang. 1970. Analysis of individual differences in multidimensional scaling via an n-way generalization of “Eckart-Young” decomposition. *Psychometrika*, 35, 238–319.

Carroll, J. D., and J. J. Chang. 1972. *IDIOSCAL (Individual differences in orientation scaling). Paper presented at the spring meeting of the Psychometric Society, Princeton, New Jersey*. : .

Carroll, J. D., S. Pruzansky, and J. B. Kruskal. 1980. CANDELINC: A general approach to multidimensional analysis with linear constraints on parameters. *Psychometrika*, 45, 3–24.

Harshman, R. A. 1970. *Foundations of the PARAFAC procedure: Models and conditions for an explanatory multi-model factor analysis*, Working Papers in Phonetics No. 16 ed. Los Angeles: University of California.

Jones, L. E., and F. W. Young. 1972. Structure of a social environment: longitudinal individual differences scaling of an intact group. *Journal of Personality and Social Psychology*, 24, 108–121.

MacCallum, R. C. 1977. Effects of conditionality on INDSCAL and ALSCAL weights. *Psychometrika*, 42, 297–305.

Mardia, K. V. 1972. *Statistics of Directional Data*. New York: Academic Press.

Mardia, K. V., J. T. Kent, and J. M. Bibby. 1979. *Multivariate Analysis*. New York: Academic Press.

Null, C. H., and W. S. Sarle. 1982. Multidimensional scaling by least squares. *Proceedings of the 7th Annual SAS User's Group International*, , – .

Schiffman, S. S., M. L. Reynolds, and F. W. Young. 1981. *Introduction to multidimensional scaling: theory, methods and applications*. New York: Academic Press.

Takane, Y., F. W. Young, and J. de Leeuw. 1977. Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features. *Psychometrika*, 42, 7–67.

Tucker, L. R. 1972. Relations between multidimensional scaling and three mode factor analysis. *Psychometrika*, 37, 3–28.

Young, F. W., and R. Lewyckyj. 1979. *ALSCAL-4 user's guide*. Carrboro: N.C.: Data Analysis and Theory Associates.

Young, F. W., D. V. Easterling, and B. N. Forsyth. 1984. *The general Euclidean model for scaling three mode dissimilarities: theory and application. In: Research Methods for Multi-mode Data Analysis in the Behavioral Sciences, H. G. Law, G. W. Snyder, Jr., J. Hattie, and R. P. McDonald, eds*. New York: Praeger.

Young, F. W., Y. Takane, and R. Lewyckyj. 1978. ALSCAL: A nonmetric multidimensional scaling program with several different options. *Behavioral Research Methods and Instrumentation*, 10, 451–453.

ANACOR Algorithms

The ANACOR algorithm consists of three major parts:

1. A singular value decomposition (SVD)
2. Centering and rescaling of the data and various rescalings of the results
3. Variance estimation by the delta method.

Other names for SVD are "Eckart-Young decomposition" after Eckart and Young (1936), who introduced the technique in psychometrics, and "basic structure" (Horst, 1963). The rescalings and centering, including their rationale, are well explained in Benzécri (1969), Nishisato (1980), Gifi (1981), and Greenacre (1984). Those who are interested in the general framework of matrix approximation and reduction of dimensionality with positive definite row and column metrics are referred to Rao (1980). The delta method is a method that can be used for the derivation of asymptotic distributions and is particularly useful for the approximation of the variance of complex statistics. There are many versions of the delta method, differing in the assumptions made and in the strength of the approximation (Rao, 1973, ch. 6; Bishop et al., 1975, ch. 14; Wolter, 1985, ch. 6).

Notation

The following notation is used throughout this chapter unless otherwise stated:

k_1	Number of rows (row objects)
k_2	Number of columns (column objects)
p	Number of dimensions

Data-Related Quantities

f_{ij}	Nonnegative data value for row *i* and column *j*: collected in table *F*
f_{i+}	Marginal total of row *i*, $i = 1, \ldots, k_1$
f_{+j}	Marginal total of column *j*, $j = 1, \ldots, k_2$
N	Grand total of *F*

Scores and Statistics

r_{is}	Score of row object *i* on dimension *s*
c_{js}	Score of column object *j* on dimension *s*
I	Total inertia

Basic Calculations

One way to phrase the ANACOR objective (cf. Heiser, 1981) is to say that we wish to find row scores $\{r_{is}\}$ and column scores $\{c_{js}\}$ so that the function

$$\sigma(\{r_{is}\};\{c_{js}\}) = \sum_i \sum_j f_{ij} \sum_s (r_{is} - c_{js})^2$$

is minimal, under the standardization restriction either that

$$\sum_i f_{i+} r_{is} r_{it} = \delta^{st}$$

or

$$\sum_j f_{+j} c_{js} c_{jt} = \delta^{st}$$

where δ^{st} is Kronecker's delta and t is an alternative index for dimensions. The trivial set of scores ({1},{1}) is excluded.

The ANACOR algorithm can be subdivided into five steps, as explained below.

Data scaling and centering

The first step is to form the auxiliary matrix **Z** with general element

$$z_{ij} = \frac{f_{ij}}{\sqrt{f_{i+} f_{+j}}} - \frac{\sqrt{f_{i+} f_{+j}}}{N}$$

Singular value decomposition

Let the singular value decomposition of **Z** be denoted by

$$\mathbf{Z} = \mathbf{K}\mathbf{\Lambda}\mathbf{L}'$$

with $\mathbf{K}'\mathbf{K} = \mathbf{I}$, $\mathbf{L}'\mathbf{L} = \mathbf{I}$, and L diagonal. This decomposition is calculated by a routine based on Golub and Reinsch (1971). It involves Householder reduction to bidiagonal form and diagonalization by a QR procedure with shifts. The routine requires an array with more rows than columns, so when $k_1 < k_2$ the original table is transposed and the parameter transfer is permuted accordingly.

Adjustment to the row and column metric

The arrays of both the left-hand singular vectors and the right-hand singular vectors are adjusted row-wise to form scores that are standardized in the row and in the column marginal proportions, respectively:

$$r_{is} = k_{is} / \sqrt{f_{i+}/N},$$

$$c_{js} = l_{js} / \sqrt{f_{+j}/N}.$$

This way, both sets of scores satisfy the standardization restrictions simultaneously.

Determination of variances and covariances

For the application of the delta method to the results of generalized eigenvalue methods under multinomial sampling, the reader is referred to Gifi (1981, ch. 12) and Israëls (1987, Appendix B). It is shown there that *N* time variance-covariance matrix of a function φ of the observed cell proportions $p = \{p_{ij} = f_{ij}/N\}$ asymptotically reaches the form

$$N \times cov(\phi(p)) \cong \sum_i \sum_j \pi_{ij} \left(\frac{\partial \phi}{\partial p_{ij}}\right) \left(\frac{\partial \phi}{\partial p_{ij}}\right)' - \left(\sum_i \sum_j \pi_{ij} \frac{\partial \phi}{\partial p_{ij}}\right) \left(\sum_i \sum_j \pi_{ij} \frac{\partial \phi}{\partial p_{ij}}\right)'$$

Here the quantities π_{ij} are the cell probabilities of the multinomial distribution, and $\partial\phi/\partial p_{ij}$ are the partial derivatives of φ (which is either a generalized eigenvalue or a generalized eigenvector) with respect to the observed cell proportion. Expressions for these partial derivatives can also be found in the above-mentioned references.

Normalization of row and column scores

Depending on the normalization option chosen, the scores are normalized, which implies a compensatory rescaling of the coordinate axes of the row scores and the column scores. The general formula for the weighted sum of squares that results from this rescaling is

row scores: $\sum_i f_{i+} r_{is}^2 = N\lambda_s(1+q)$

column scores: $\sum_j f_{+j} c_{js}^2 = N\lambda_s(1-q)$

The parameter *q* can be chosen freely or it can be specified according to the following designations:

$$q = \begin{cases} 0, & \text{canonical} \\ 1, & \text{row principal} \\ -1, & \text{column principal} \end{cases}$$

There is a fifth possibility, choosing the designation "principal," that does not fit into this scheme. It implies that the weighted sum of squares of both sets of scores becomes equal to $N\lambda_s^2$. The estimated variances and covariances are adjusted according to the type of normalization chosen.

Diagnostics

After printing the data, ANACOR optionally also prints a table of row profiles and column profiles, which are $\{f_{ij}/f_{i+}\}$ and $\{f_{ij}/f_{+j}\}$, respectively.

Singular Values, Maximum Rank and Inertia

All singular values λ_s defined in step 2 are printed up to a maximum of $\min\{(k_1-1),(k_2-1)\}$. Small singular values and corresponding dimensions are suppressed when they don't exceed the quantity $(k_1k_2)^{1/2}10^{-7}$; in this case a warning message is issued. Dimensionwise inertia and total inertia are given by the relationships

$$I=\sum_s \lambda_s^2=\sum_s\sum_i \frac{f_{i+}r_{is}^2}{N}$$

where the right-hand part of this equality is true only if the normalization is row principal (but for the other normalizations similar relationships are easily derived from "Normalization of row and column scores "). The quantities "proportion explained" are equal to inertia divided by total inertia: λ_s^2/I.

Scores and Contributions

This output is given first for rows, then for columns, and always preceded by a column of marginal proportions (f_{i+}/N and f_{+j}/N, respectively). The table of scores is printed in *p* dimensions. The contribution to the inertia of each dimension is given by

$$\tau_{is}=\frac{f_{i+}}{N}\frac{r_{is}^2}{\lambda_s^2}$$

$$\tau_{js}=\frac{f_{+j}}{N}c_{js}^2$$

The above formula is true only under the row principal normalization option. For the other normalizations, similar relationships are again easily derived from "Normalization of row and column scores ") The contribution of dimensions to the inertia of each point is given by, for $s,t=1,\ldots,p$,

$$\sigma_{is}=r_{is}^2/\sum_t r_{it}^2$$

$$\sigma_{js}=c_{js}^2/\sum_t c_{jt}^2$$

Variances and Correlation Matrix of Singular Values and Scores

The computation of variances and covariances is explained in "Determination of variances and covariances ". Since the row and column scores are linear functions of the singular vectors, an adjustment is necessary depending on the normalization option chosen. From these adjusted variances and covariances the correlations are derived in the standard way.

Permutations of the Input Table

For each dimension *s*, let $\rho(i|s)$ be the permutation of the first k_1 integers that would sort the *s*th column of $\{r_{is}\}$ in ascending order. Similarly, let $\rho(j|s)$ be the permutation of the first k_2 integers that would sort the *s*th column of $\{c_{js}\}$ in ascending order. Then the permuted data matrix is given by $\{f_{\rho(i|s),\rho(j|s)}\}$.

References

Benzécri, J. P. 1969. Statistical analysis as a tool to make patterns emerge from data. In: *Methodologies of Pattern Recognition,* S. Watanabe, ed. New York: Academic Press, 35–74.

Bishop, Y. M., S. E. Fienberg, and P. W. Holland. 1977. *Discrete Multivariate Analysis: Theory and Practice*. Cambridge, MA: MIT Press.

Eckart, C., and G. Young. 1936. The approximation of one matrix by another one of lower rank. *Psychometrika,* 1, 211–218.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Golub, G. H., and C. Reinsch. 1971. Linear Algebra. In: *Handbook for Automatic Computation, Volume II,* J. H. Wilkinson, and C. Reinsch, eds. New York: Springer-Verlag.

Greenacre, M. J. 1984. *Theory and applications of correspondence analysis*. London: Academic Press.

Heiser, W. J. 1981. *Unfolding analysis of proximity data*. Leiden: Department of Data Theory, University of Leiden.

Horst, P. 1963. *Matrix algebra for social scientists*. New York: Holt, Rinehart and

Winston. Israëls, A. 1987. *Eigenvalue techniques for qualitative data*. Leiden:

DSWO Press.

Nishisato, S. 1980. *Analysis of categorical data: Dual scaling and its applications*. Toronto: University of Toronto Press.

Rao, C. R. 1973. *Linear statistical inference and its applications*, 2nd ed. New York: John Wiley and Sons.

Rao, C. R. 1980. Matrix approximations and reduction of dimensionality in multivariate statistical analysis. In: *Multivariate Analysis, Vol. 5,* P. R. Krishnaiah, ed. Amsterdam: North-Holland, 3–22.

Wolter, K. M. 1985. *Introduction to variance estimation*. Berlin: Springer-Verlag.

ANOVA Algorithms

This chapter describes the algorithms used by the ANOVA procedure.

Model and Matrix Computations

Notation

The following notation is used throughout this section unless otherwise stated.

Table 7-1
Notation

Notation	Description
N	Number of cases
F	Number of factors
CN	Number of covariates
k_i	Number of levels of factor i
Y_k	Value of the dependent variable for case k
Z_{jk}	Value of the jth covariate for case k
w_k	Weight for case k
W	Sum of weights of all cases

The Model

A linear model with covariates can be written in matrix notation as

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{Z}\mathbf{C} + \mathbf{e} \qquad \textbf{(1)}$$

where

Table 7-2
Notation

Notation	Description
$\mathbf{Y}$	$N \times 1$ vector of values of the dependent variable
$\mathbf{X}$	Design matrix $(N \times p)$ of rank $q < p$
β	Vector of parameters $(p \times 1)$
$\mathbf{Z}$	Matrix of covariates $(N \times CN)$
$\mathbf{C}$	Vector of covariate coefficients $(CN \times 1)$
$\mathbf{e}$	Vector of error terms $(N \times 1)$

Constraints

To reparametrize equation **(1)** to a full rank model, a set of non-estimable conditions is needed. The constraint imposed on non-regression models is that all parameters involving level 1 of any factor are set to zero.

For regression model, the constraints are that the analysis of variance parameters estimates for each main effect and each order of interactions sum to zero. The interaction must also sum to zero over each level of subscripts.

For a standard two way ANOVA model with the main effects α_i and β_j, and interaction parameter γ_{ij}, the constraints can be expressed as

$$\begin{array}{ll} \alpha_1 = \beta_1 = \gamma_{1j} = \gamma_{i1} = 0 & \text{non} - \text{regression} \\ \alpha_\bullet = \beta_\bullet = \gamma_{i\bullet} = \gamma_{\bullet j} = 0 & \text{regression} \end{array}$$

where $\bullet$ indicates summation.

Computation of Matrices

$\mathbf{X}'\mathbf{X}$

Non-regression Model

The $\mathbf{X}'\mathbf{X}$ matrix contains the sum of weights of the cases that contain a particular combination of parameters. All parameters that involve level 1 of any of the factors are excluded from the matrix. For a two-way design with $k_1 = 2$ and $k_2 = 3$, the symmetric matrix would look like the following:

$$\begin{array}{c|ccccc} & \alpha_2 & \beta_2 & \beta_3 & \gamma_{22} & \gamma_{23} \\ \hline \alpha_2 & N_{2\bullet} & N_{22} & N_{23} & N_{22} & N_{23} \\ \beta_2 & & N_{\bullet 2} & 0 & N_{22} & 0 \\ \beta_3 & & & N_{\bullet 3} & 0 & N_{23} \\ \gamma_{22} & & & & N_{22} & 0 \\ \gamma_{23} & & & & & N_{23} \end{array}$$

The elements $N_{i\bullet}$ or $N_{\bullet j}$ on the diagonal are the sums of weights of cases that have level *i* of *a* or level *j* of β. Off-diagonal elements are sums of weights of cases cross-classified by parameter combinations. Thus, $N_{\bullet 3}$ is the sum of weights of cases in level 3 of main effect β_3, while N_{22} is the sum of weights of cases with α_2 and β_2.

Regression Model

A row of the design matrix **X** is formed for each case. The row is generated as follows:

If a case belongs to one of the 2 to k_i levels of factor *i*, a code of 1 is placed in the column corresponding to the level and 0 in all other $k_i - 1$ columns associated with factor *i*. If the case belongs in the first level of factor *i*, -1 is placed in *all* the $k_i - 1$ columns associated with factor *i*. This is repeated for each factor. The entries for the interaction terms are obtained as products of the entries in the corresponding main effect columns. This vector of dummy variables for a case will be denoted as $d(i), i = 1, \ldots, NC$, where *NC* is the number of columns in the reparametrized design matrix. After the vector **d** is generated for case *k*, the *ij*th cell of $\mathbf{X}'\mathbf{X}$ is incremented by $d(i)d(j)w_k$, where $i = 1, \ldots, NC$ and $j \geq i$.

Checking and Adjustment for the Mean

After all cases have been processed, the diagonal entries of $\mathbf{X}'\mathbf{X}$ are examined. Rows and columns corresponding to zero diagonals are deleted and the number of levels of a factor is reduced accordingly. If a factor has only one level, the analysis will be terminated with a message. If the first specified level of a factor is missing, the first non-empty level will be deleted from the matrix for non-regression model. For regression designs, the first level cannot be missing. All entries of $\mathbf{X}'\mathbf{X}$ are subsequently adjusted for means.

The highest order of interactions in the model can be selected. This will affect the generation of $\mathbf{X}'\mathbf{X}$ If none of these options is chosen, the program will generate the highest order of interactions allowed by the number of factors. If sub-matrices corresponding to main effects or interactions in the reparametrized model are not of full rank, a message is printed and the order of the model is reduced accordingly.

Cross-Product Matrices for Continuous Variables

Provisional means algorithm are used to compute the adjusted-for-the-means cross-product matrices.

Matrix of Covariates Z'Z

The covariance of covariates *m* and *l* after case *k* has been processed is

$$\mathbf{Z}'\mathbf{Z}_{ml}(k) = \mathbf{Z}'\mathbf{Z}_{ml}(k-1) + \frac{w_k\left(W_k Z_{lk} - \sum_{j=1}^{k} w_j Z_{lj}\right)\left(W_k Z_{mk} - \sum_{j=1}^{k} w_j Z_{mj}\right)}{W_k W_{k-1}}$$

where W_k is the sum of weights of the first *k* cases.

The Vector Z'Y

The covariance between the *m*th covariate and the dependent variable after case *k* has been processed is

$$\mathbf{Z}'\mathbf{Y}_m(k) = \mathbf{Z}'\mathbf{Y}_m(k-1) + \frac{w_k\left(W_kY_k - \sum_{j=1}^{k} w_jY_j\right)\left(W_kZ_{mk} - \sum_{j=1}^{k} w_jZ_{mj}\right)}{W_kW_{k-1}}$$

The Scalar Y'Y

The corrected sum of squares for the dependent variable after case *k* has been processed is

$$\mathbf{Y}'\mathbf{Y}(k) = \mathbf{Y}'\mathbf{Y}(k-1) + \frac{w_k\left(W_kY_k - \sum_{j=1}^{k} w_jY_j\right)^2}{W_kW_{k-1}}$$

The Vector X'Y

$\mathbf{X}'\mathbf{Y}$ is a vector with *NC* rows. The *i*th element is

$$\mathbf{X}'\mathbf{Y}_i = \sum_{k=1}^{N} Y_k w_k \delta_k,$$

where, for non-regression model, $\delta_k = 1$ if case *k* has the factor combination in column *i* of $\mathbf{X}'\mathbf{X}$; $\delta_k = 0$ otherwise. For regression model, $\delta_k = d(i)$ where *d(i)* is the dummy variable for column *i* of case *k*. The final entries are adjusted for the mean.

Matrix X'Z

The (*i*, *m*)th entry is

$$\mathbf{X}'\mathbf{Z}_{im} = \sum_{k=1}^{N} Z_{mk} w_k \delta_k$$

where δ_k has been defined previously. The final entries are adjusted for the mean.

Computation of ANOVA Sum of Squares

The full rank model with covariates

$$\mathbf{Y} = \mathbf{X}\beta + \mathbf{ZC} + \mathbf{e}$$

can also be expressed as

$$\mathbf{Y} = \mathbf{X}_k\mathbf{b}_k + \mathbf{X}_m\mathbf{b}_m + \mathbf{ZC} + \mathbf{e}$$

where **X** and **b** are partitioned as

$\mathbf{X} = [\mathbf{X}_k|\mathbf{X}_m]$ and $\beta = \begin{bmatrix} \mathbf{b}_k \\ \mathbf{b}_m \end{bmatrix}$.

The normal equations are then

$$\begin{bmatrix} \mathbf{Z}'\mathbf{Z} & \mathbf{Z}'\mathbf{X}_k & \mathbf{Z}'\mathbf{X}_m \\ \mathbf{X}'_k\mathbf{Z} & \mathbf{X}'_k\mathbf{X}_k & \mathbf{X}'_k\mathbf{X}_m \\ \mathbf{X}'_m\mathbf{Z} & \mathbf{X}'_m\mathbf{X}_k & \mathbf{X}'_m\mathbf{X}_m \end{bmatrix} \begin{bmatrix} \hat{\mathbf{C}} \\ \hat{\mathbf{b}}_k \\ \hat{\mathbf{b}}_m \end{bmatrix} = \begin{bmatrix} \mathbf{Z}'\mathbf{Y} \\ \mathbf{X}'_k\mathbf{Y} \\ \mathbf{X}'_m\mathbf{Y} \end{bmatrix} \quad \textbf{(2)}$$

The normal equations for any reduced model can be obtained by excluding those entries from equation **(2)** corresponding to terms that do not appear in the reduced model.

Thus, for the model excluding $\mathbf{b}_m$,

$$\mathbf{Y} = \mathbf{X}_k\mathbf{b}_k + \mathbf{ZC} + \mathbf{e}$$

the solution to the normal equation is:

$$\begin{bmatrix} \tilde{\mathbf{C}} \\ \tilde{\mathbf{b}}_k \end{bmatrix} = \begin{bmatrix} \mathbf{Z}'\mathbf{Z} & \mathbf{Z}'\mathbf{X}_k \\ \mathbf{X}'_k\mathbf{Z} & \mathbf{X}'_k\mathbf{X}_k \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{Z}'\mathbf{Y} \\ \mathbf{X}'_k\mathbf{Y} \end{bmatrix} \quad \textbf{(3)}$$

The sum of squares due to fitting the complete model (explained *SS*) is

$$R(\mathbf{C}, \mathbf{b}_k, \mathbf{b}_m) = \left[\hat{\mathbf{C}}', \hat{\mathbf{b}}'_k, \hat{\mathbf{b}}'_m\right] \begin{bmatrix} \mathbf{Z}'\mathbf{Y} \\ \mathbf{X}'_k\mathbf{Y} \\ \mathbf{X}'_m\mathbf{Y} \end{bmatrix} = \hat{\mathbf{C}}'\mathbf{Z}'\mathbf{Y} + \hat{\mathbf{b}}'_k\mathbf{X}'_k\mathbf{Y} + \hat{\mathbf{b}}'_m\mathbf{X}'_m\mathbf{Y}$$

For the reduced model, it is

$$R(\mathbf{C}, \mathbf{b}_k) = \left[\tilde{\mathbf{C}}', \tilde{\mathbf{b}}'_k\right] \begin{bmatrix} \mathbf{Z}'\mathbf{Y} \\ \mathbf{X}'_k\mathbf{Y} \end{bmatrix} = \tilde{\mathbf{C}}'\mathbf{Z}'\mathbf{Y} + \tilde{\mathbf{b}}'_k\mathbf{X}'_k\mathbf{Y}$$

The residual (unexplained) sum of squares for the complete model is $RSS = \mathbf{Y}'\mathbf{Y} - R(\mathbf{C}, \mathbf{b}_k, \mathbf{b}_m)$ and similarly for the reduced model. The total sum of squares is $\mathbf{Y}'\mathbf{Y}$. The reduction in the sum of squares due to including $\mathbf{b}_m$ in a model that already includes $\mathbf{b}_k$ and **C** will be denoted as $R(\mathbf{b}_m|\mathbf{C}, \mathbf{b}_k)$. This can also be expressed as

$$R(\mathbf{b}_m|\mathbf{C}, \mathbf{b}_k) = R(\mathbf{C}, \mathbf{b}_k, \mathbf{b}_m) - R(\mathbf{C}, \mathbf{b}_k)$$

There are several ways to compute $R(\mathbf{b}_m|\mathbf{C}, \mathbf{b}_k)$. The sum of squares due to the full model, as well as the sum of squares due to the reduced model, can each be calculated, and the difference obtained (Method 1).

$$R(\mathbf{b}_m|\mathbf{C}, \mathbf{b}_k) = \hat{\mathbf{C}}'\mathbf{Z}'\mathbf{Y} + \hat{\mathbf{b}}'_k\mathbf{X}'_k\mathbf{Y} + \hat{\mathbf{b}}'_m\mathbf{X}'_m\mathbf{Y} - \tilde{\mathbf{C}}'\mathbf{Z}'\mathbf{Y} - \tilde{\mathbf{b}}'_k\mathbf{X}'_k\mathbf{Y}$$

A sometimes computationally more efficient procedure is to calculate

$$R(\mathbf{b}_m|\mathbf{C},\mathbf{b}_k) = \hat{\mathbf{b}}_m^{'}\mathbf{T}_m^{-1}\hat{\mathbf{b}}_m$$

where $\hat{\mathbf{b}}_m$ are the estimates obtained from fitting the full model and $\mathbf{T}_m$ is the partition of the inverse matrix corresponding to $\mathbf{b}_m$ (Method 2).

$$\begin{bmatrix} \mathbf{Z}^{'}\mathbf{Z} & \mathbf{Z}^{'}\mathbf{X}_k & \mathbf{Z}^{'}\mathbf{X}_m \\ \mathbf{X}^{'}_k\mathbf{Z} & \mathbf{X}^{'}_k\mathbf{X}_k & \mathbf{X}^{'}_k\mathbf{X}_m \\ \mathbf{X}^{'}_m\mathbf{Z} & \mathbf{X}^{'}_m\mathbf{X}_k & \mathbf{X}^{'}_m\mathbf{X}_m \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{T}_c & \mathbf{T}_{ck} & \mathbf{T}_{cm} \\ \mathbf{T}_{kc} & \mathbf{T}_k & \mathbf{T}_{km} \\ \mathbf{T}_{mc} & \mathbf{T}_{mk} & \mathbf{T}_m \end{bmatrix}$$

Model and Options

Notation

Let **b** be partitioned as

$$\mathbf{b} = \begin{bmatrix} \mathrm{M} \\ \mathrm{D} \end{bmatrix} = \begin{bmatrix} \mathbf{m}_1 \\ . \\ \mathbf{m}_F \\ \hline \mathbf{d}_1 \\ . \\ \mathbf{d}_{F-1} \end{bmatrix}$$

where

Table 7-3
Notation

Notation	Description
M	Vector of main effect coefficients
$\mathbf{m}_i$	Vector of coefficients for main effect *i*
$\mathbf{m}^{(i)}$	**M** excluding $\mathbf{m}_i$
$\mathbf{M}^{i*}$	**M** including only $\mathbf{m}_1$ through $\mathbf{m}_{i-1}$
D	Vector of interaction coefficients
$\mathbf{d}_k$	Vector of *k*th order interaction coefficients
$\mathbf{d}_{k_i}$	Vector of coefficients for the *i*th of the *k*th order interactions
$\mathbf{D}^{(k)}$	**D** excluding $\mathbf{d}_k$
$\mathbf{D}^{k*}$	**D** including only $\mathbf{d}_1$ through $\mathbf{d}_{k-1}$
$\mathbf{d}_k^{(i)}$	$\mathbf{d}_k$ excluding $\mathbf{d}_{k_i}$
C	Vector of covariate coefficients
c_i	Covariate coefficient

Notation	Description
$\mathbf{C}^{(i)}$	**C** excluding c_i
$\mathbf{C}^{i*}$	**C** including only c_1 through c_{i-1}

Models

Different types of sums of squares can be calculated in ANOVA.

Sum of Squares for Type of Effects

Table 7-4
Sum of squares for type of effect

Type	Covariates	Main Effects	Interactions
Experimental and Hierarchical	$R(\mathbf{C})$	$R(\mathbf{M}\vert\mathbf{C})$	$R(\mathbf{d}_k\vert\mathbf{C},\mathbf{M},\mathbf{D}^{k*})$
Covariates with Main Effects	$R(\mathbf{C},\mathbf{M})$	$R(\mathbf{C},\mathbf{M})$	$R(\mathbf{d}_k\vert\mathbf{C},\mathbf{M},\mathbf{D}^{k*})$
Covariates after Main Effects	$R(\mathbf{C}\vert\mathbf{M})$	$R(\mathbf{M})$	$R(\mathbf{d}_k\vert\mathbf{C},\mathbf{M},\mathbf{D}^{k*})$
Regression	$R(\mathbf{C}\vert\mathbf{M},\mathbf{D})$	$R(\mathbf{M}\vert\mathbf{C},\mathbf{D})$	$R(\mathbf{d}_k\vert\mathbf{C},\mathbf{M},\mathbf{D}^{k*})$

All sums of squares are calculated as described in the introduction. Reductions in sums of squares $(R(\mathbf{A}|\mathbf{B}))$ are computed using Method 1. Since all cross-product matrices have been corrected for the mean, all sums of squares are adjusted for the mean.

Sum of Squares Within Effects

Table 7-5
Sum of squares within effects

Type	Covariates	Main Effects	Interactions
Default Experimental	$R\left(c_i\vert\mathbf{C}^{(i)}\right)$	$R\left(\mathbf{m}_i\vert\mathbf{C},\mathbf{M}^{(i)}\right)$	$R\left(\mathbf{d}_{k_i}\vert\mathbf{C},\mathbf{M},\mathbf{D}^{k*},\mathbf{d}_k^{(i)}\right)$
Covariates with Main Effects	$R\left(c_i\vert\mathbf{M},\mathbf{C}^{(i)}\right)$	$R\left(\mathbf{m}_i\vert\mathbf{C},\mathbf{M}^{(i)}\right)$	same as default
Covariates after Main Effects	$R\left(c_i\vert\mathbf{M},\mathbf{C}^{(i)}\right)$	$R\left(\mathbf{m}_i\vert\mathbf{M}^{(i)}\right)$	same as default
Regression	$R\left(c_i\vert\mathbf{M},\mathbf{C}^{(i)},\mathbf{D}\right)$	$R\left(m_i\vert\mathbf{M}^{(i)},\mathbf{C},\mathbf{D}\right)$	$R\left(\mathbf{d}_{k_i}\vert\mathbf{C},\mathbf{M},\mathbf{D}^{(k_i)}\right)$
Hierarchical	$R(c_i\vert\mathbf{C}^{i*})$	$R(\mathbf{m}_i\vert\mathbf{C},\mathbf{M}^{i*})$	same as default
Hierarchical and Covariates with Main Effects or Hierarchical and Covariates after Main Effects	$R(c_i\vert\mathbf{C}^{i*},\mathbf{M})$	$R(\mathbf{m}_i\vert\mathbf{M}^{i*})$	same as default

Reductions in sums of squares are calculated using Method 2, except for specifications involving the Hierarchical approach. For these, Method 1 is used. All sums of squares are adjusted for the mean.

Degrees of Freedom

Main Effects

$$df_M = \sum_{i=1}^{F} (k_i - 1)$$

Main Effect i

$$(k_i - 1)$$

Covariates

$$df_c = CN$$

Covariate i

1

Interactions

Interactions $\mathbf{d}_r$:

$df_r =$ number of linearly independent columns corresponding to interaction $\mathbf{d}_r$ in $\mathbf{X}'\mathbf{X}$

Interactions $\mathbf{d}_{r_i}$:

$df =$ number of independent columns corresponding to interaction $\mathbf{d}_{r_i}$ in $\mathbf{X}'\mathbf{X}$

Model

$$df_{Model} = df_M + df_c + \sum_{r=1}^{F-1} df_r$$

Residual

$$W - 1 - df_{Model}$$

Total

W − 1

Multiple Classification Analysis

Notation

Table 7-6
Notation

Notation	Description
Y_{ijk}	Value of the dependent variable for the *k*th case in level *j* of main effect *i*
n_{ij}	Sum of weights of observations in level *j* of main effect *i*
k_i	Number of nonempty levels in the *i*th main effect
W	Sum of weights of all observations

Basic Computations

Mean of Dependent Variable in Level j of Main Effect i

$$\overline{Y}_{ij} = \sum_{k=1}^{n_{ij}} Y_{ijk}/n_{ij}$$

Grand Mean

$$\overline{Y} = \sum_i \sum_j \sum_k Y_{ijk}/W$$

Coefficient Estimates

The computation of the coefficient for the main effects only model (b_{ij}) and coefficients for the main effects and covariates only model $\left(\tilde{b}_{ij}\right)$ are obtained as previously described.

Calculation of the MCA Statistics (Andrews, et al., 1973)

Deviations

For each level of each main effect, the following are computed:

Unadjusted Deviations

The unadjusted deviation from the grand mean for the *j*th level of the *i*th factor:

$$m_{ij} = \overline{Y}_{ij} - \overline{Y}$$

Deviations Adjusted for the Main Effects

$$m_{ij}^1 = b_{ij} - \sum_{j=2}^{k_i} b_{ij} n_{ij} / W, \text{ where } b_{i1} = 0 .$$

Deviations Adjusted for the Main Effects and Covariates (Only for Models with Covariates)

$$m_{ij}^2 = \tilde{b}_{ij} - \sum_{j=2}^{k_i} \tilde{b}_{ij} n_{ij} / W, \text{ where } \tilde{b}_{i1} = 0 .$$

ETA and Beta Coefficients

For each main effect *i*, the following are computed:

$$ETA_i = \sqrt{\sum_{j=2}^{k_i} n_{ij} \left(\overline{Y}_{ij} - \overline{Y} \right)^2 / \mathbf{Y}'\mathbf{Y}}$$

Beta Adjusted for Main Effects

$$Beta_i = \sqrt{\sum_{j=2}^{k_i} n_{ij} \left(m_{ij}^1 \right)^2 / \mathbf{Y}'\mathbf{Y}}$$

Beta Adjusted for Main Effects and Covariates

$$Beta_i = \sqrt{\sum_{j=2}^{k_i} n_{ij} \left(m_{ij}^2 \right)^2 / \mathbf{Y}'\mathbf{Y}}$$

Squared Multiple Correlation Coefficients

Main effects model

$$R_m^2 = \frac{R(\mathbf{M})}{\mathbf{Y}'\mathbf{Y}}.$$

Main effects and covariates model

$$R_{mc}^2 = \frac{R(\mathbf{M},\mathbf{C})}{\mathbf{Y}'\mathbf{Y}}.$$

The computations of *R*(**M**), *R*(**M**,**C**), and $\mathbf{Y}'\mathbf{Y}$ are outlined previously.

Unstandardized Regression Coefficients for Covariates

Estimates for the **C** vector, which are obtained the first time covariates are entered into the model, are printed.

Cell Means and Sample Sizes

Cell means and sample sizes for each combination of factor levels are obtained from the $\mathbf{X}'\mathbf{Y}$ and $\mathbf{X}'\mathbf{X}$ matrices prior to correction for the mean.

$$\overline{Y}_i = \frac{\left(\mathbf{X}'\mathbf{Y}\right)_i}{\left(\mathbf{X}'\mathbf{X}\right)_{ii}} \quad i = 1, \ldots, CN$$

Means for combinations involving the first level of a factor are obtained by subtraction from marginal totals.

Matrix Inversion

The Cholesky decomposition (Stewart, 1973) is used to triangularize the matrix. If the tolerance is less than 10^{-5}, the matrix is considered singular.

References

Andrews, F., J. Morgan, J. Sonquist, and L. Klein. 1973. *Multiple classification analysis*, 2nd ed. Ann Arbor: University of Michigan.

Searle, S. R. 1966. *Matrix algebra for the biological sciences*. New York: John Wiley& Sons, Inc.

Searle, S. R. 1971. *Linear Models*. New York: John Wiley & Sons, Inc.

Stewart, G. W. 1973. *Introduction to matrix computations*. New York: Academic Press.

AREG Algorithms

In the ordinary regression model the errors are assumed to be uncorrelated. The model considered here has the form

$$\begin{aligned} y_t &= a + \sum_{i=1}^{p} b_i x_{ti} + u_t \quad t = 1, \ldots, n \\ u_t &= \rho u_{t-1} + \epsilon_t \end{aligned} \tag{1}$$

where ϵ_t is an uncorrelated random error with variance σ^2 and zero mean. The error terms u_t follow a first-order autoregressive process. The constant term *a* can be included or excluded as specified. In the discussion below, if *a* is not included, it is set to be zero and not involved in the subsequent computation.

Two computational methods—Prais-Winsten and Cochrane-Orcutt—are described here.

Cochrane-Orcutt Method

Note that model (1) can be rewritten in two equivalent forms as:

$$y_t - \rho y_{t-1} = a(1-\rho) + \sum_{i=1}^{p} b_i \left(x_{ti} - \rho x_{(t-1)i}\right) + \epsilon_t \tag{2}$$

$$y_t - a - \sum_{i=1}^{p} b_i x_{ti} = \rho\left(y_{t-1} - a - \sum_{i=1}^{p} b_i x_{(t-1)i}\right) + \epsilon_t \tag{3}$$

Defining $y_t^* = y_t - \rho y_{t-1}$ and $x_{ti}^* = x_{ti} - \rho x_{(t-1)i}$ for $t = 2, \ldots, n$, equation (2) can be rewritten as

$$y_t^* = a(1-\rho) + \sum_{i=1}^{p} b_i x_{ti}^* + \epsilon_t \tag{2*}$$

Starting with an initial value for ρ, the difference y_t^* and x_{ti}^* in equation (2*) are computed and OLS then applied to equation (2*) to estimate *a* and b_i. These estimates in turn can be used in equation (3) to update $\hat{\rho}$ and the standard error of the estimate $\hat{\rho}$.

Initial Results

An initial value for ρ can be pre-set by the user or set to be zero by default. The OLS method is used to obtain an initial estimate for *a* (if constant term is include) and b_i.

ANOVA

Based on the OLS results, an analysis of variance table is constructed in which the degrees of freedom for regression are *p*, the number of *X* variables in equation (1), while the degrees of freedom for the residual are $n - p^* - 1$ if initial $\rho \neq 0$ and are $n - p^*$ otherwise. p^* is the number of coefficients in equation (1). The sums of squares, mean squares, and other statistics are computed as in the REGRESSION procedure.

Intermediate Results

At each iteration, the following statistics are calculated:

Rho

An updated value for ρ is computed as

$$\hat{\rho} = \frac{\sum_{t=2}^{n} \tilde{u}_t \tilde{u}_{t-1}}{\sum_{t=1}^{n} \tilde{u}_t^2}$$

where the residuals $\tilde{u}_t$ are obtained from equation (1).

Standard Error of rho

An estimate of the standard error of $\hat{\rho}$

$$se(\hat{\rho}) = \sqrt{\frac{1-\hat{\rho}^2}{n-1-p^*}}$$

where $p^* = p+1$ if there is a constant term; p otherwise.

Durbin-Watson Statistic

$$DW = \frac{\sum_{i=1}^{n-1} (\tilde{\epsilon}_{i+1} - \tilde{\epsilon}_i)^2}{\sum_{i=1}^{n} \tilde{\epsilon}_i^2}$$

where

$$\tilde{\epsilon}_1 = \sqrt{1-\hat{\rho}^2}\,\tilde{u}_1$$

$$\tilde{\epsilon}_i = \tilde{u}_i - \hat{\rho}\tilde{u}_{i-1} \quad i = 2, \ldots, n$$

Mean Square Error

An estimate of the variance of ϵ_t

$$MSE = \frac{\sum_{t=2}^{n} (\tilde{u}_t - \hat{\rho}\tilde{u}_{t-1})^2}{n - 2 - p^*}$$

Final Results

Iteration terminates if either all the parameters change by less than a specified value (default 0.001) or the number of iterations exceeds the cutoff value (default 10).

The following variables are computed for each case:

FIT

Fitted responses are computed as

$$\tilde{y}_1 = \hat{y}_1$$

and

$$\tilde{y}_t = \hat{y}_t + \hat{\rho}\hat{u}_{t-1} \quad t = 2, \ldots, n$$

in which $\hat{\rho}$ is the final estimate of ρ, and

$$\hat{y}_t = \hat{a} + \sum_{i=1}^{p} \hat{b}_i x_{ti}$$
$$\hat{u}_t = y_t - \hat{y}_t \quad t = 1, \ldots, n$$

ERR

Residuals are computed as

$$\tilde{\epsilon}_t = y_t - \tilde{y}_t \quad t = 2, \ldots, n$$
$$\tilde{\epsilon}_1 = \sqrt{1 - \hat{\rho}^2(y_1 - \tilde{y}_1)}$$

SEP

Standard error of predicted values at time *t*

$$SEP_1 = \sqrt{MSE}\sqrt{\left(\frac{1}{1-\hat{\rho}^2} + \tilde{h}_1\right)}$$

and

$$SEP_t = \sqrt{MSE}\sqrt{\left(1 + \tilde{h}_t\right)} \quad t = 2,\, \ldots, n$$

where

$$\tilde{h}_i = \mathbf{X}_i\left(\mathbf{X}^{*'}\mathbf{X}^*\right)^{-1}\mathbf{X}'_i$$

in which $\mathbf{X}_i$ is the predictor vector at time *i* with the first component 1 if a constant term is included in equation (2*). **X*** is a $(n-1) \times p^*$ design matrix for equation (2*). The first column has a value of $1-\hat{\rho}$ if a constant term is included in equation (2*).

LCL and UCL

95% prediction interval for the future y_k is

$$\tilde{y}_k \pm t_{n-1-p^*;0.025} SEP_k$$

Other Statistics

Other statistics such as Multiple *R*, *R*-Squared, Adjusted *R*-Squared, and so on, are computed. Consult the REGRESSION procedure for details.

Prais-Winsten Method

This method is a modification of the Cochrane-Orcutt method in that the first case gets explicit treatment. By adding an extra equation to (2*), the model has the form of

$$\begin{aligned} (1-\rho)y_1 &= a(1-\rho) + \sum_{i=1}^{p} b_i(1-\rho)x_{1i} + (1-\rho)u_1 \\ y_t^* &= a(1-\rho) + \sum_{i=1}^{p} b_i x_{ti}^* + \epsilon_t \quad t = 2,\, \ldots, n \end{aligned} \qquad \textbf{(4)}$$

Like the Cochrane-Orcutt method, an initial value of ρ can be set by the user or a default value of zero can be used. The iterative process of estimating the parameters is performed via weighted least squares (WLS). The weights used in WLS computation are $w_1 = \left(1-\hat{\rho}^2\right)/(1-\hat{\rho})^2$ and $w_i = 1$ for $i = 2,\, \ldots, n$. The computation of the variance of ϵ_t and the variance of $\hat{\rho}$ is the same as that of the WLS in the REGRESSION procedure.

Initial Results

The WLS method is used to obtain initial parameter estimates.

ANOVA

The degrees of freedom are p for regression and $n - p^*$ for residuals.

Intermediate Results

The formulas for RHO, SE Rho, DW, and MSE are exactly the same as those in the Cochrane-Orcutt method. The degrees of freedom for residuals, however, are $n - 1 - p^*$.

Final Results

The following variables are computed for each case.

SEP

Standard error of predicted value at time *t* is computed as

$$SEP_1 = \sqrt{MSE}\sqrt{\left(\frac{1}{1-\hat{\rho}^2} + \tilde{h}_1\right)}$$

$$SEP_t = \sqrt{MSE}\sqrt{\left(1 + \tilde{h}_t\right)} \quad t = 2, \ldots, n$$

where $\tilde{h}$ is computed as

$$\tilde{h}_i = \mathbf{X}_i\left(\mathbf{X}^{*'}\mathbf{X}^*\right)^{-1}\mathbf{X}'_i$$

in which $\mathbf{X}_i$ is the predictor vector at time *i* and $\mathbf{X}^*$ is a $n \times p^*$ design matrix for equation (4). If a constant term is included in the model, the first column of **X*** has a constant value of $1 - \hat{\rho}$, the first row of $\mathbf{X}^*$ is $\sqrt{w_1}(x_{11}, \ldots, x_{1p})$, and $p^* = p + 1$

LCL and UCL

95% prediction interval for y_k at time *k* is

$$\tilde{y}_k \pm t_{n-p^*;0.025} SEP_k$$

ARIMA Algorithms

The ARIMA procedure computes the parameter estimates for a given seasonal or non-seasonal univariate ARIMA model. It also computes the fitted values, forecasting values, and other related variables for the model.

Notation

The following notation is used throughout this chapter unless otherwise stated:

y_t (t=1, 2, ..., N)	Univariate time series under investigation.
N	Total number of observations.
a_t (t = 1, 2, ... , N)	White noise series normally distributed with mean zero and variance σ_a^2.
p	Order of the non-seasonal autoregressive part of the model
q	Order of the non-seasonal moving average part of the model
d	Order of the non-seasonal differencing
P	Order of the seasonal autoregressive part of the model
Q	Order of the seasonal moving-average part of the model
D	Order of the seasonal differencing
s	Seasonality or period of the model
$\phi_p(B)$	AR polynomial of B of order p, $\phi_p(B) = 1 - \varphi_1 B - \varphi_2 B^2 - ... - \varphi_p B^p$
$\theta_q(B)$	MA polynomial of B of order q, $\theta_q(B) = 1 - \vartheta_1 B - \vartheta_2 B^2 - ... - \vartheta_q B^q$
$\Phi_P(B^s)$	Seasonal AR polynomial of BS of order P, $\Phi_P(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{s2} - ... - \Phi_P B^{sP}$
$\Theta_Q(B^s)$	Seasonal MA polynomial of BS of order Q, $\Theta_Q(B^s) = 1 - \Theta_1 B^s - \Theta_2 B^{s2} - ... - \Theta_Q B^{sQ}$
Δ	Differencing operator $\Delta = (1-B)^d (1-B^s)^D$
B	Backward shift operator with $BY_t = Y_{t-1}$ and $Ba_t = a_{t-1}$

Models

A seasonal univariate ARIMA(p,d,q)(P,D,Q)$_S$ model is given by

$$\mathbf{\Phi}(B)[\Delta y_t - \mu] = \mathbf{\Theta}(B) a_t \qquad t = 1, \ldots, N$$

where

$$\mathbf{\Phi}(B) = \phi_p(B)\,\Phi_P(B)$$

$$\mathbf{\Theta}(B) = \theta_q(B)\,\Theta_Q(B)$$

and μ is an optional model constant. It is also called the stationary series mean, assuming that, after differencing, the series is stationary. When NOCONSTANT is specified, μ is assumed to be zero.

An optional log scale transformation can be applied to y_t before the model is fitted. In this chapter, the same symbol, y_t, is used to denote the series either before or after log scale transformation.

Independent variables $x_1, x_2, \ldots, x_m$ can also be included in the model. The model with independent variables is given by

$$\mathbf{\Phi}(B)\left[\Delta\left(y_t - \sum_{i=1}^{m} c_i x_{it}\right) - \mu\right] = \mathbf{\Theta}(B)a_t$$

where

$c_i, i = 1, 2, \ldots, m$, are the regression coefficients for the independent variables.

Estimation

Basically, two different estimation algorithms are used to compute maximum likelihood (ML) estimates for the parameters in an ARIMA model:

- **Melard's algorithm** is used for the estimation when there is no missing data in the time series. The algorithm computes the maximum likelihood estimates of the model parameters. The details of the algorithm are described in (Melard, 1984), (Pearlman, 1980), and (Morf, Sidhu, and Kailath, 1974).
- A **Kalman filtering algorithm** is used for the estimation when some observations in the time series are missing. The algorithm efficiently computes the marginal likelihood of an ARIMA model with missing observations. The details of the algorithm are described in the following literature: (Kohn and Ansley, 1986) and (Kohn and Ansley, 1985).

Initialization of ARMA parameters

The ARMA parameters are initialized as follows:

Assume that the series Y_t follows an ARMA(p,q)(P,Q) model with mean 0; that is:

$$Y_t - \varphi_1 Y_{t-1} - \cdots - \varphi_p Y_{t-p} = a_t - \theta_1 a_{t-1} - \cdots - \theta_q a_{t-q}$$

In the following c_l and ρ_l represent the *l*th lag autocovariance and autocorrelation of Y_t respectively, and $\hat{c}_l$ and $\hat{\rho}_l$ represent their estimates.

Non-seasonal AR parameters

For AR parameter initial values, the estimated method is the same as that in appendix A6.2 of (Box, Jenkins, and Reinsel, 1994). Denote the estimates as $\hat{\varphi}_1^{'}, \cdots, \hat{\varphi}_{p+q}^{'}$.

Non-seasonal MA parameters

Let

$$w_t = Y_t - \varphi_1 Y_{t-1} - \cdots - \varphi_p Y_{t-p} = a_t - \theta_1 a_{t-1} - \cdots - \theta_q a_{t-q}$$

The cross covariance

$$\lambda_l = E(w_{t+l}a_t) = E\big((a_{t+l} - \theta_1 a_{t+l-1} - \cdots - \theta_q a_{t+l-q})a_t\big) = \begin{cases} \sigma_a^2 & l = 0 \\ -\theta_1\sigma_a^2 & l = 1 \\ \cdots & \cdots \\ -\theta_q\sigma_a^2 & l = q \\ 0 & l > q \end{cases}$$

Assuming that an AR(p+q) can approximate Y_t, it follows that:

$$Y_t - \varphi'_1 Y_{t-1} - \cdots - \varphi'_p Y_{t-p} - \varphi'_{p+1} Y_{t-p-1} - \cdots - \varphi'_{p+q} Y_{t-p-q} = a_t$$

The AR parameters of this model are estimated as above and are denoted as $\hat{\varphi}'_1, \cdots, \hat{\varphi}'_{p+q}$.

Thus λ_l can be estimated by

$$\lambda_l \approx E\Big((Y_{t+l} - \varphi_1 Y_{t+l-1} - \cdots - \varphi_p Y_{t+l-p})\Big(Y_t - \varphi'_1 Y_{t-1} - \cdots - \varphi'_{p+q} Y_{t-p-q}\Big)\Big)$$
$$= \left(\rho_l - \sum_{j=1}^{p+q} \varphi_j \rho_{l+j} - \sum_{i=1}^{p} \varphi_i \rho_{l-i} + \sum_{i=1}^{p}\sum_{j=1}^{p+q} \varphi_i \varphi_j \rho_{l+j-i}\right) c_0$$

And the error variance σ_a^2 is approximated by

$$\hat{\sigma}_a^2 = Var\left(-\sum_{j=0}^{p+q} \varphi'_j Y_{t-j}\right) = \sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \varphi'_i \varphi'_j c_{i-j} = c_0 \sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \varphi'_i \varphi'_j \rho_{i-j}$$

with $\hat{\varphi}'_0 = -1$.

Then the initial MA parameters are approximated by $\theta_l = -\lambda_l/\sigma_a^2$ and estimated by

$$\hat{\theta}_l = -\hat{\lambda}_l/\hat{\sigma}_a^2 = \frac{\rho_l - \sum_{j=1}^{p+q} \hat{\varphi}_j \rho_{l+j} - \sum_{i=1}^{p} \hat{\varphi}_i \rho_{l-i} + \sum_{i=1}^{p}\sum_{j=1}^{p+q} \hat{\varphi}_i \hat{\varphi}_j \rho_{l+j-i}}{\sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \hat{\varphi}'_i \hat{\varphi}'_j \rho_{i-j}}$$

So $\hat{\theta}_l$ can be calculated by $\hat{\varphi}'_j$, $\hat{\varphi}_i$, and $\{\hat{\rho}_l\}_{l=1}^{p+2q}$. In this procedure, only $\{\hat{\rho}_l\}_{l=1}^{p+q}$ are used and all other parameters are set to 0.

Seasonal parameters

For seasonal AR and MA components, the autocorrelations at the seasonal lags in the above equations are used.

Diagnostic Statistics

The following definitions are used in the statistics below:

N_p	Number of parameters. $N_p = \begin{cases} p+q+P+Q+m & \text{without model constant} \\ p+q+P+Q+m+1 & \text{with model constant} \end{cases}$
SSQ	Residual sum of squares $SSQ = \mathbf{e}'\mathbf{e}$, where **e** is the residual vector
$\hat{\sigma}_a^2$	Estimated residual variance. $\hat{\sigma}_a^2 = \frac{SSQ}{df}$, where $df = N - N_p$
SSQ'	Adjusted residual sum of squares. $SSQ' = (SSQ)\lvert\mathbf{\Omega}\rvert^{1/N}$, where $\mathbf{\Omega}$ is the theoretical covariance matrix of the observation vector computed at MLE

Log-Likelihood

$$L = -N\ln(\hat{\sigma}_a) - \frac{SSQ'}{2\hat{\sigma}_a^2} - \frac{N\ln(2\pi)}{2}$$

Akaike Information Criterion (AIC)

$$AIC = -2L + 2N_p$$

Schwartz Bayesian Criterion (SBC)

$$SBC = -2L + \ln(N)N_p$$

Generated Variables

The following variables are generated for each case.

Predicted Values

Computation of predicted values depends upon the forecasting method.

Forecasting Method: Conditional Least Squares (CLS or AUTOINT)

In general, the model used for fitting and forecasting (after estimation, if involved) can be written as

$$y_t - D(B)y_t = \mathbf{\Phi}(B)\mu + \mathbf{\Theta}(B)a_t + \sum_{i=1}^{m} c_i\mathbf{\Phi}(B)\Delta x_{it}$$

where

$$D(B) = \mathbf{\Theta}(B)\mathbf{\Delta} - 1$$

$$\mathbf{\Phi}(B)\mu = \mathbf{\Phi}(1)\mu$$

Thus, the predicted values $(FIT)_t$ are computed as follows:

$$(FIT)_t = \hat{y}_t = D(B)\,\hat{y}_t + \Phi(B)\,\mu + \Theta(B)\,\hat{a}_t + \sum_{i=1}^{m} c_i \Phi(B)\,\mathbf{\Delta} x_{it}$$

where

$$\hat{a}_t = y_t - \hat{y}_t \quad 1 \le t \le n$$

Starting Values for Computing Fitted Series. To start the computation for fitted values, all unavailable beginning residuals are set to zero and unavailable beginning values of the fitted series are set according to the selected method:

CLS. The computation starts at the (*d*+*sD*)-th period. After a specified log scale transformation, if any, the original series is differenced and/or seasonally differenced according to the model
specification. Fitted values for the differenced series are computed first. All unavailable beginning fitted values in the computation are replaced by the stationary series mean, which is equal to the model constant in the model specification. The fitted values are then aggregated to the original series and properly transformed back to the original scale. The first *d*+*sD* fitted values are set to missing (SYSMIS).

AUTOINIT. The computation starts at the [*d*+*p*+*s*(*D*+*P*)]-th period. After any specified log scale transformation, the actual *d*+*p*+*s*(*D*+*P*) beginning observations in the series are used as beginning fitted values in the computation. The first *d*+*p*+*s*(*D*+*P*) fitted values are set to missing. The fitted values are then transformed back to the original scale, if a log transformation is specified.

Forecasting Method: Unconditional Least Squares (EXACT)

As with the CLS method, the computations start at the (*d*+*sD*)-th period. First, the original series (or the log-transformed series if a transformation is specified) is differenced and/or seasonally differenced according to the model specification. Then the fitted values for the differenced series are computed. The fitted values are one-step-ahead, least-squares predictors calculated using the theoretical autocorrelation function of the stationary autoregressive moving average (ARMA) process corresponding to the differenced series. The autocorrelation function is computed by treating the estimated parameters as the true parameters. The fitted values are then aggregated to the original series and properly transformed back to the original scale. The first *d*+*sD* fitted values are set to missing (SYSMIS). The details of the least-squares prediction algorithm for the ARMA models can be found in (Brockwell and Davis, 1991).

Residuals

Residual series are always computed in the transformed log scale, if a transformation is specified.

$$(ERR)_t = y_t - (FIT)_t \quad t = 1, 2, \ldots, N$$

Standard Errors of the Predicted Values

Standard errors of the predicted values are first computed in the transformed log scale, if a transformation is specified.

Forecasting Method: Conditional Least Squares (CLS or AUTOINIT)

$$(SEP)_t = \hat{\sigma}_a \quad t = 1, 2, \dots, N$$

Forecasting Method: Unconditional Least Squares (EXACT)

In the EXACT method, unlike the CLS method, there is no simple expression for the standard errors of the predicted values. The standard errors of the predicted values will, however, be given by the least-squares prediction algorithm as a byproduct.

Standard errors of the predicted values are then transformed back to the original scale for each predicted value, if a transformation is specified.

Confidence Limits of the Predicted Values

Confidence limits of the predicted values are first computed in the transformed log scale, if a transformation is specified:

$$(LCL)_t = (FIT)_t - t_{1-\alpha/2,df}(SEP)_t \quad t = 1, 2, \dots, N$$

$$(UCL)_t = (FIT)_t + t_{1-\alpha/2,df}(SEP)_t \quad t = 1, 2, \dots, N$$

where $t_{1-\alpha/2,df}$ is the $(1 - \alpha/2)$ -th percentile of a *t* distribution with *df* degrees of freedom and α is the specified confidence level (by default α=0.05).

Confidence limits of the predicted values are then transformed back to the original scale for each predicted value, if a transformation is specified.

Forecasting

The following values are computed for each forecast period.

Forecasting Values

Computation of forecasting values depends upon the forecasting method.

Forcasting Method: Conditional Least Squares (CLS or AUTOINIT)

$\hat{y}_t(l)$, the *l-step-ahead* forecast of y_{t+l} at the time *t*, can be represented as:

$$\hat{y}_t(l) = D(B)\hat{y}_{t+l} + \Phi(B)\mu + \Theta(B)\hat{a}_{t+l} + \sum_{i=1}^{m} c_i \Phi(B)\mathbf{\Delta} x_{i,t+l}$$

Note that

$$\hat{y}_{t+l-i} = \begin{cases} y_{t+l-i} & \text{if } l \le i \\ \hat{y}_t(l-i) & \text{if } l > i \end{cases}$$

$$\hat{a}_{t+l-j} = \begin{cases} y_{t+l-i} - \hat{y}_{t+l-i-1}(1) & \text{if } l \le i \\ 0 & \text{if } l > i \end{cases}$$

Forecasting Method: Unconditional Least Squares (EXACT)

The forecasts with this option are finite memory, least-squares forecasts computed using the theoretical autocorrelation function of the series. The details of the least-squares forecasting algorithm for the ARIMA models can be found in (Brockwell et al., 1991).

Standard Errors of the Forecasting Values

Computation of these standard errors depends upon the forecasting method.

Forcasting Method: Conditional Least Squares (CLS or AUTOINIT)

For the purpose of computing standard errors of the forecasting values, the model can be written in the format of weights (ignoring the model constant):

$$y_t = \frac{\vartheta_q(B)\Theta_Q(B)}{\phi_p(B)\Phi_P(B)} a_t = \psi(B)\, a_t = \sum_{i=0}^{\infty} \psi_i B^i a_{t-i}$$

where

$$\psi_0 = 1$$

Then

$$\text{se}\left[\hat{y}_t(l)\right] = \left\{1 + \psi_1^2 + \psi_2^2 + \ldots + \psi_{l-1}^2\right\}^{\frac{1}{2}} \hat{\sigma}_a$$

Note that, for the predicted value, $l = 1$. Hence, $(SEP)_t = \hat{\sigma}_a$ at any time t.

Computation of ΨWeights. Ψ weights can be computed by expanding both sides of the following equation and solving the linear equation system established by equating the corresponding coefficients on both sides of the expansion:

$$\phi_p(B)\,\Phi_P(B)\,\Delta\psi(B) = \theta_q(B)\,\Theta_Q(B)$$

An explicit expression of Ψ weights can be found in (Box et al., 1994).

Forecasting Method: Unconditional Least Squares (EXACT)

As with the standard errors of the predicted values, the standard errors of the forecasting values are a byproduct during the least-squares forecasting computation. The details can be found in (Brockwell et al., 1991).

References

Box, G. E. P., G. M. Jenkins, and G. C. Reinsel. 1994. *Time series analysis: Forecasting and control*, 3rd ed. Englewood Cliffs, N.J.: Prentice Hall.

Brockwell, P. J., and R. A. Davis. 1991. *Time Series: Theory and Methods*, 2 ed. : Springer-Verlag.

Kohn, R., and C. Ansley. 1985. Efficient estimation and prediction in time series regression models. *Biometrika*, 72:3, 694–697.

Kohn, R., and C. Ansley. 1986. Estimation, prediction, and interpolation for ARIMA models with missing data. *Journal of the American Statistical Association*, 81, 751–761.

Makridakis, S. G., S. C. Wheelwright, and R. J. Hyndman. 1997. *Forecasting: Methods and applications*, 3rd ed. ed. New York: John Wiley and Sons.

Melard, G. 1984. A fast algorithm for the exact likelihood of autoregressive-moving average models. *Applied Statistics*, 33:1, 104–119.

Morf, M., G. S. Sidhu, and T. Kailath. 1974. Some new algorithms for recursive estimation in constant, linear, discrete-time systems. *IEEE Transactions on Automatic Control*, AC-19:4, 315–323.

Pearlman, J. G. 1980. An algorithm for the exact likelihood of a high-order autoregressive-moving average process. *Biometrika*, 67:1, 232–233.

Automated Data Preparation Algorithms

The goal of automated data preparation is to prepare a dataset so as to generally improve the training speed, predictive power, and robustness of models fit to the prepared data.

These algorithms do not assume which models will be trained post-data preparation. At the end of automated data preparation, we output the predictive power of each recommended predictor, which is computed from a linear regression or naïve Bayes model, depending upon whether the target is continuous or categorical.

Notation

The following notation is used throughout this chapter unless otherwise stated:

X	A continuous or categorical variable
x_i	Value of the variable X for case i.
f_i	Frequency weight for case i. Non-integer positive values are rounded to the nearest integer. If there is no frequency weight variable, then all $f_i = 1$. If the frequency weight of a case is zero, negative or missing, then this case will be ignored.
w_i	Analysis weight for case i. If there is no analysis weight variable, then all $w_i = 1$. If the analysis weight of a case is zero, negative or missing, then this case will be ignored.
n	Number of cases in the dataset
N_X	$\sum_{i=1}^{n} f_i I\left(x_i\right.$ is not missing), where I (expression) is the indicator function taking value 1 when the expression is true, 0 otherwise.
W_X	$\sum_{i=1}^{n} f_i w_i I\left(x_i\right.$ is not missing$)$
N_{XY}	$\sum_{i=1}^{n} f_i I\left(x_i\right.$ and y_i are not missing$)$
W_{XY}	$\sum_{i=1}^{n} f_i w_i I\left(x_i\right.$ and y_i are not missing$)$
$\overline{x}$	The mean of variable X, $\frac{1}{W_X}\sum_{i=1}^{n} f_i w_i x_i I\left(x_i\right.$ is not missing$)$
M_X^r	$\sum_{i=1}^{n} f_i w_i (x_i - \overline{x})^r$
$\overline{x}_y$	$\frac{1}{W_{XY}}\sum_{i=1}^{n} f_i w_i x_i I\left(x_i\right.$ and y_i are not missing$)$
M_{XY}	$\sum_{i=1}^{n} f_i w_i \left(x_i - \overline{x}_y\right)\left(y_i - \overline{y}_x\right)$

A note on missing values

Listwise deletion is used in the following sections:

- "Univariate Statistics Collection "

- "Basic Variable Screening "
- "Measurement Level Recasting "
- "Missing Value Handling "
- "Outlier Identification and Handling "
- "Continuous Predictor Transformations "
- "Target Handling "
- "Reordering Categories "
- "Unsupervised Merge "

Pairwise deletion is used in the following sections:

- "Bivariate Statistics Collection "
- "Supervised Merge "
- "Supervised Binning "
- "Feature Selection and Construction "
- "Predictive Power "

A note on frequency weight and analysis weight

The frequency weight variable is treated as a case replication weight. For example if a case has a frequency weight of 2, then this case will count as 2 cases.

The analysis weight would adjust the variance of cases. For example if a case x_i of a variable *X* has an analysis weight w_i, then we assume that $x_i \sim N\left(\mu, \frac{\sigma^2}{w_i}\right)$.

Frequency weights and analysis weights are used in automated preparation of other variables, but are themselves left unchanged in the dataset.

Date/Time Handling

Date Handling

If there is a date variable, we extract the date elements (year, month and day) as ordinal variables. If requested, we also calculate the number of elapsed days/months/years since the user-specified reference date (default is the current date). Unless specified by the user, the "best" unit of duration is chosen as follows:

1. If the minimum number of elapsed days is less than 31, then we use days as the best unit.
2. If the minimum number of elapsed days is less than 366 but larger than or equal to 31, we use months as the best unit. The number of months between two dates is calculated based on average number of days in a month (30.4375): *months* = *days* / 30.4375.
3. If the minimum number of elapsed days is larger than or equal to 366, we use years as the best unit. The number of years between two dates is calculated based on average number of days in a year (365.25): *years* = *days* / 365.25.

Once the date elements are extracted and the duration is obtained, then the original date variable will be excluded from the rest of the analysis.

Time Handling

If there is a time variable, we extract the time elements (second, minute and hour) as ordinal variables. If requested, we also calculate the number of elapsed seconds/minutes/hours since the user-specified reference time (default is the current time). Unless specified by the user, the "best" unit of duration is chosen as follows:

1. If the minimum number of elapsed seconds is less than 60, then we use seconds as the best unit.
2. If the minimum number of elapsed seconds is larger than or equal to 60 but less than 3600, we use minutes as the best unit.
3. If the minimum number of elapsed seconds is larger than or equal to 3600, we use hours as the best unit.

Once the elements of time are extracted and time duration is obtained, then original time predictor will be excluded.

Univariate Statistics Collection

Continuous Variables

For each continuous variable, we calculate the following statistics:

- Number of missing values: $N_X^{missing} = \sum_{i=1}^{n} f_i I\left(x_i \text{ is missing}\right)$
- Number of valid values: N_X
- Minimum value: $\min_i x_i$
- Maximum value: $\max_i x_i$
- Mean, standard deviation, skewness. (see below)
- The number of distinct values *I*.
- The number of cases for each distinct value s_i: $c_i = \sum_{j=1}^{n} f_j I\left(x_j = s_i\right)$
- Median: If the distinct values of *X* are sorted in ascending order, $s_1 < s_2 < \cdots < s_I$, then the median can be computed by $Median\left(X\right) = \min\left\{s_i : \frac{cc_i}{N_X} \geq 0.5\right\}$, where $cc_i = \sum_{j=1}^{i} c_i$.

Note: If the number of distinct values is larger than a threshold (default is 5), we stop updating the number of distinct values and the number of cases for each distinct value. Also we do not calculate the median.

Categorical Numeric Variables

For each categorical numeric variable, we calculate the following statistics:

- Number of missing values: $N_X^{missing} = \sum_{i=1}^{n} f_i I\left(x_i \text{ is missing}\right)$

- Number of valid values: N_X
- Minimum value: $\min_i x_i$ (only for ordinal variables)
- Maximum value: $\max_i x_i$ (only for ordinal variables)
- The number of categories.
- The counts of each category.
- Mean, Standard deviation, Skewness (only for ordinal variables). (see below)
- Mode (only for nominal variables). If several values share the greatest frequency of occurrence, then the mode with the smallest value is used.
- Median (only for ordinal variables): If the distinct values of *X* are sorted in ascending order, $s_1 < s_2 < \cdots < s_I$, then the median can be computed by $Median\left(X\right) = \min\left\{s_i : \frac{cc_i}{N_X} \geq 0.5\right\}$, where $cc_i = \sum_{j=1}^{i} c_i$.

Notes:

1. If an ordinal predictor has more categories than a specified threshold (default 10), we stop updating the number of categories and the number of cases for each category. Also we do not calculate mode and median.
2. If a nominal predictor has more categories than a specified threshold (default 100), we stop collecting statistics and just store the information that the variable had more than threshold categories.

Categorical String Variables

For each string variable, we calculate the following statistics:

- Number of missing values: $N_X^{missing} = \sum_{i=1}^{n} f_i I\left(x_i\right.$ is missing$\left.\right)$
- Number of valid values: N_X
- The number of categories.
- Counts of each category.
- Mode: If several values share the greatest frequency of occurrence, then the mode with the smallest value is used.

Note: If a string predictor has more categories than a specified threshold (default 100), we stop collecting statistics and just store the information that the predictor had more than threshold categories.

Mean, Standard Deviation, Skewness

We calculate mean, standard deviation and skewness by updating moments.

1. Start with $N_X^{(0)} = W_X^{(0)} = \overline{x}^{(0)} = M_X^{2(0)} = M_X^{3(0)} = 0$.
2. For *j*=1,..,*n* compute:

   $N_X^{(j)} = N_X^{(j-1)} + f_j I\left(x_j\right.$ is not missing$\left.\right)$

$W_X^{(j)} = W_X^{(j-1)} + f_j w_i I\left(x_j \text{ is not missing}\right)$

$v_j = \frac{f_j w_j}{W_X^{(j)}}\left(x_j - \overline{x}^{(j-1)}\right)$

$\overline{x}^{(j)} = \overline{x}^{(j-1)} + v_j$

$M_X^{2(j)} = M_X^{2(j-1)} + \frac{W_X^{(j)} W_X^{(j-1)}}{f_j w_j} v_j^2$

$M_X^{3(j)} = M_X^{3(i-1)} - 3 v_j M_X^{2(j-1)} + \frac{W_X^{(j)} W_X^{(j-1)}}{(f_j w_j)^2}\left(W_X^{(j)} - 2 f_j w_j\right) v_j^3$

3. After the last case has been processed, compute:

   Mean: $\overline{x} = \overline{x}^{(n)}$

   Standard deviation: $sd = \sqrt{\frac{M_X^{2(n)}}{N_X \;\; 1}}$

   Skewness: $skew = \frac{\frac{N_X}{(N_X-2)}\frac{1}{(N_X-1)} M_X^{3(n)}}{sd^3}$

If $N_X \leq 2$ or $sd^2 < 10^{-20}$, then skewness is not calculated.

Basic Variable Screening

1. If the percent of missing values is greater than a threshold (default is 50%), then exclude the variable from subsequent analysis.

2. For continuous variables, if the maximum value is equal to minimum value, then exclude the variable from subsequent analysis.

3. For categorical variables, if the mode contains more cases than a specified percentage (default is 95%), then exclude the variable from subsequent analysis.

4. If a string variable has more categories than a specified threshold (default is 100), then exclude the variable from subsequent analysis.

Checkpoint 1: Exit?

This checkpoint determines whether the algorithm should be terminated. If, after the screening step:

1. The target (if specified) has been removed from subsequent analysis, or

2. All predictors have been removed from subsequent analysis,

then terminate the algorithm and generate an error.

Measurement Level Recasting

For each continuous variable, if the number of distinct values is less than a threshold (default is 5), then it is recast as an ordinal variable.

For each numeric ordinal variable, if the number of categories is greater than a threshold (default is 10), then it is recast as a continuous variable.

Note: The continuous-to-ordinal threshold must be less than the ordinal-to-continuous threshold.

Outlier Identification and Handling

In this section, we identify outliers in continuous variables and then set the outlying values to a cutoff or to a missing value. The identification is based on the robust mean and robust standard deviation which are estimated by supposing that the percentage of outliers is no more than 5%.

Identification

1. Compute the mean and standard deviation from the raw data. Split the continuous variable into non-intersecting intervals: $I_i = (\overline{x} + (i-1) \times sd_w, x + i \times sd_w]$, $i = -3, -2, \cdots, 2, 3, 4$, where $I_{-3} = (-\infty, \overline{x} - 3sd_w]$, $I_4 = (\overline{x} + 3sd_w, +\infty]$ and $sd_w = sd \times \sqrt{\frac{N_X-1}{W_X\ 1}}$.

2. Calculate univariate statistics in each interval:

   $N_{I_i} = \sum_{j=1}^{n} f_j I\left(x_j \in I_i\right)$, $W_{I_i} = \sum_{i=1}^{n} f_i w_i I\left(x_j \in I_i\right)$

   $\overline{x}_{I_i} = \frac{\sum_{j=1}^{n} f_j w_j x_j I(x_j \in I_i)}{W_{I_i}}$, $M_{I_i}^2 = \sum_{j=1}^{n} f_j w_j (x_j - \overline{x}_{I_i})^2 I\left(x_j \in I_i\right)$

3. Let $l = -3$, $r = 4$, and $p = 0$.

4. Between two tail intervals I_l and I_r, find one interval with the least number of cases.

5. If $N_{I_l} \le N_{I_r}$, then $p_{current} = \frac{N_{I_l}}{N_X}$. Check if $p + p_{current}$ is less than a threshold $p_{threshold}$ (default is 0.05). If it does, then $p = p + p_{current}$ and $l = l + 1$, go to step 4; otherwise, go to step 6.

   Else $p_{current} = \frac{N_{I_r}}{N_X}$. Check if $p + p_{current}$ is less than a threshold, $p_{threshold}$. If it is, then $p = p + p_{current}$ and $r = r - 1$, go to step 4; otherwise, go to step 6.

6. Compute the robust mean $\overline{x}_{robust}$ and robust standard deviation sd_{robust} within the range $(\overline{x} + (l-1) \times sd, \overline{x} + r \times sd]$. See below for details.

7. If x_i satisfies the conditions:

   $\sqrt{w_i}\left(x_i - \overline{x}_{robust}\right) < -cutoff \times sd_{robust}$ or $\sqrt{w_i}\left(x_i - \overline{x}_{robust}\right) > cutoff \times sd_{robust}$

   where *cutoff* is positive number (default is 3), then x_i is detected as an outlier.

Handling

Outliers will be handled using one of following methods:

- Trim outliers to cutoff values. If $\sqrt{w_i}\left(x_i - \overline{x}_{robust}\right) < -cutoff \times sd_{robust}$ then replace x_i by $\overline{x}_{robust} - cutoff \times sd_{robust}/\sqrt{w_i}$, and if $\sqrt{w_i}\left(x_i - \overline{x}_{robust}\right) > cutoff \times sd_{robust}$ then replace x_i by $\overline{x}_{robust} + cutoff \times sd_{robust}/\sqrt{w_i}$.
- Set outliers to missing values.

Update Univariate Statistics

After outlier handling, we perform a data pass to calculate univariate statistics for each continuous variable, including the number of missing values, minimum, maximum, mean, standard deviation, skewness, and number of outliers.

Robust Mean and Standard Deviation

Robust mean and standard deviation within the range $(\overline{x} + (l-1) \times sd, \overline{x} + r \times sd]$ are calculated as follows:

$$\overline{x}_{robust} = \frac{\sum_{i=l}^{r} W_{I_i} \overline{x}_{I_i}}{\sum_{i=l}^{r} W_{I_i}}$$

and

$$sd_{robust} = \sqrt{\frac{M^2_{robust}}{\sum_{i=l}^{r} N_{I_i} - 1}}$$

where $M^2_{robust} = \sum_{i=l}^{r} A_{I_i}$ and $A_{I_i} = M^2_{I_i} W_{I_i} (\overline{x}_{robust} - \overline{x}_{I_i})^2$

Missing Value Handling

Continuous variables. Missing values are replaced by the mean, and the following statistics are updated:

- Standard deviation: $sd \times \sqrt{\frac{N_X - 1}{N - 1}}$, where $N = N_X + N_X^{missing}$.
- Skewness: $skew \times \frac{L_1}{L_2}$, where $L_1 = \left(\frac{N}{N-2}\right)\left(\frac{N_X - 2}{N_X}\right)$ and $L_2 = \sqrt{\frac{N_X - 1}{N - 1}}$
- The number of missing values: $N_X^{missing} = 0$
- The number of valid values: $N_X = N$

Ordinal variables. Missing values are replaced by the median, and the following statistics are updated:

- The number of cases in the median category: $c_{median} + N_X^{missing}$, where c_{median} is the original number of cases in the median category.
- The number of missing values: $N_X^{missing} = 0$
- The number of valid values: $N_X = N$

Nominal variables. Missing values are replaced by the mode, and the following statistics are updated:

- The number of cases in the modal category: $c_{mode} + N_X^{missing}$, where c_{mode} is the original number of cases in the modal category.
- The number of missing values: $N_X^{missing} = 0$
- The number of valid values: $N_X = N$

Continuous Predictor Transformations

We transform a continuous predictor so that it has the user-specified mean $\overline{x}_{user}$ (default 0) and standard deviation sd_{user} (default 1) using the z-score transformation, or minimum $\min_{user}$ (default 0) and maximum $\max_{user}$ (default 100) value using the min-max transformation.

Z-score Transformation

Suppose a continuous variable has mean $\overline{x}$ and standard deviation *sd*. The *z*-score transformation is

$$x_i^{'} = \frac{sd_{user}}{sd} \times (x_i - \overline{x}) + \overline{x}_{user}$$

where $x_i^{'}$ is the transformed value of continuous variable *X* for case *i*.

Since we do not take into account the analysis weight in the rescaling formula, the rescaled values $x_i^{'}$ follow a normal distribution $N\left(\overline{x}_{user}, \frac{sd_{user}^2}{w_i}\right)$.

Update univariate statistics

After a z-score transformation, the following univariate statistics are updated:

- Number of missing values: $N_{X'}^{missing} = N_X^{missing}$
- Number of valid values: $N_{X'} = N_X$
- Minimum value: $\min\left(x_i^{'}\right) = \frac{sd_{user}}{sd} \times (\min x_i - \overline{x}) + \overline{x}_{user}$
- Maximum value: $\max\left(x_i^{'}\right) = \frac{sd_{user}}{sd} \times (\max x_i - \overline{x}) + \overline{x}_{user}$
- Mean: $\overline{x}^{'} = \overline{x}_{user}$
- Standard deviation: $sd\left(x^{'}\right) = sd_{user}$
- Skewness: $skew\left(x^{'}\right) = skew(x)$

Min-Max Transformation

Suppose a continuous variable has a minimum value $\min x_i$ and a minimum value $\max x_i$. The min-max transformation is

$$x_i^{'} = \frac{\max_{user} - \min_{user}}{\max x_i - \min x_i} \times (x_i - \min x_i) + \min_{user}$$

where $x_i^{'}$ is the transformed value of continuous variable *X* for case *i*.

Update univariate statistics

After a min-max transformation, the following univariate statistics are updated:

- The number of missing values: $N_{X'}^{missing} = N_X^{missing}$

- The number of valid values: $N_{X'} = N_X$
- Minimum value: $\min\left(x_i^{'}\right) = \min_{user}$
- Maximum value: $\max\left(x_i^{'}\right) = \max_{user}$
- Mean: $\overline{x}^{'} = \frac{\max_{user} - \min_{user}}{\max x_i - \min x_i} \times (\overline{x} - \min x_i) + \min_{user}$
- Standard deviation: $sd\left(x^{'}\right) = \frac{\max_{user} - \min_{user}}{\max x_i - \min x_i} \times sd$
- Skwness: $skew\left(x^{'}\right) = skew\,(x)$

Target Handling

Nominal Target

For a nominal target, we rearrange categories from lowest to highest counts. If there is a tie on counts, then ties will be broken by ascending sort or lexical order of the data values.

Continuous Target

The transformation proposed by Box and Cox (1964) transforms a continuous variable into one that is more normally distributed. We apply the Box-Cox transformation followed by the *z* score transformation so that the rescaled target has the user-specified mean and standard deviation.

Box-Cox transformation. This transforms a non-normal variable *Y* to a more normally distributed variable:

$$g_i(\lambda) = g(y_i, \lambda) = \begin{cases} \frac{\left((y_i - c)^{\lambda} - 1\right)}{\lambda} & \lambda \neq 0 \\ \ln(y_i - c) & \lambda = 0 \end{cases}$$

where $y_i, i = 1, 2, \cdots, n$ are observations of variable *Y*, and *c* is a constant such that all values $y_i - c$ are positive. Here, we choose $c = \min(Y) - 1$.

The parameter λ is selected to maximize the log-likelihood function:

$$L(\lambda) = -\frac{N_Y}{2} \ln\left[\frac{N_Y - 1}{N_Y}(sd(g(\lambda)))^2\right] + (\lambda - 1)\sum_{i=1}^{n} f_i \ln(y_i - c)$$

where $(sd(g(\lambda)))^2 = \frac{1}{N_Y - 1}\sum_{i=1}^{n} f_i w_i (g_i(\lambda_j) - \overline{g}(\lambda_j))^2$ and $\overline{g}(\lambda) = \frac{1}{W_Y}\sum_{i=1}^{n} f_i w_i g_i(\lambda)$.

We perform a grid search over a user-specified finite set [*a*,*b*] with increment *s*. By default *a*=−3, *b*=3, and *s*=0.5.

The algorithm can be described as follows:

1. Compute $\lambda_j = a + (j-1) * s$ where *j* is an integer such that $a \leq \lambda_j \leq b$.

2. For each λ_j, compute the following statistics:

   Mean: $\overline{g}\left(\lambda_j\right) = \frac{1}{W_Y}\sum_{i=1}^{n} f_i w_i g_i(\lambda_j)$

   Standard deviation: $sd\left(g\left(\lambda_j\right)\right) = \sqrt{\frac{1}{N_Y-1}\sum_{i=1}^{n} f_i w_i \left(g_i\left(\lambda_j\right) - \overline{g}\left(\lambda_j\right)\right)^2}$

   Skewness: $skew\left(g\left(\lambda_j\right)\right) = \frac{\frac{N_Y}{(N_Y-2)}\frac{1}{(N_Y-1)}\sum_{i=1}^{n} f_i w_i \left(g_i(\lambda_j) - \overline{g}(\lambda_j)\right)^3}{sd(g(\lambda_j))^3}$

   Sum of logarithm transformation: $\sum_{i=1}^{n} f_i \ln\left(y_i - c\right)$

3. For each λ_j, compute the log-likelihood function $L\left(\lambda_j\right)$. Find the value of *j* with the largest log-likelihood function, breaking ties by selecting the smallest value of λ_j. Also find the corresponding statistics $\overline{g}\left(\lambda^*\right)$, $sd\left(g\left(\lambda^*\right)\right)$ and $skew\left(g\left(\lambda^*\right)\right)$.

4. Transform target to reflect user's mean $\overline{y}_{user}$ (default is 0) and standard deviation sd_{user} (default is 1):

$$y_i' = \frac{sd_{user}}{sd\left(g\left(\lambda^*\right)\right)} \times \left(g_i\left(\lambda^*\right) - \overline{g}\left(\lambda^*\right)\right) + \overline{y}_{user}$$

   where $\overline{g}\left(\lambda^*\right) = \frac{1}{W_Y}\sum_{i=1}^{n} f_i w_i g_i(\lambda^*)$ and $sd\left(g\left(\lambda^*\right)\right) = \sqrt{\frac{1}{N_Y-1}\sum_{i=1}^{n} f_i w_i \left(g_i\left(\lambda^*\right) - \overline{g}\left(\lambda^*\right)\right)^2}$.

Update univariate statistics. After Box-Cox and Z-score transformations, the following univariate statistics are updated:

- Minimum value: $\frac{sd_{user}}{sd(g(\lambda^*))} \times \left(g\left(\min\left(y_i\right) - c, \lambda^*\right) - \overline{g}\left(\lambda^*\right)\right) + \overline{y}_{user}$
- Maximum value: $\frac{sd_{user}}{sd(g(\lambda^*))} \times \left(g\left(\max\left(y_i\right) - c, \lambda^*\right) - \overline{g}\left(\lambda^*\right)\right) + \overline{y}_{user}$
- Mean: $\overline{y}_{user}$
- Standard deviation: sd_{user}
- Skewness: $skew\left(g\left(\lambda^*\right)\right)$

Bivariate Statistics Collection

For each target/predictor pair, the following statistics are collected according to the measurement levels of the target and predictor.

Continuous target or no target and all continuous predictors

If there is a continuous target and some continuous predictors, then we need to calculate the covariance and correlations between all pairs of continuous variables. If there is no continuous target, then we only calculate the covariance and correlations between all pairs of continuous predictors. We suppose there are there are *m* continuous variables, and denote the covariance matrix as $C_{m\times m}$, with element c_{ij}, and the correlation matrix as $R_{m\times m}$, with element r_{ij}.

We define the covariance between two continuous variables *X* and *Y* as

$$c_{XY} = \frac{1}{N_{XY} - 1}\sum_{i=1}^{n} f_i w_i \left(x_i - \overline{x}_y\right)\left(y_i - \overline{y}_x\right)$$

where $\overline{x}_y = \frac{1}{W_{XY}} \sum_{i=1}^{n} x_i I\left(x_i \text{ and } y_i \text{ are not missing}\right)$ and $\overline{y}_x = \frac{1}{W_{XY}} \sum_{i=1}^{n} y_i I\left(x_j \text{ and } y_j \text{ are not missing}\right)$.

The covariance can be computed by a provisional means algorithm:

1. Start with $N_{XY}^{(0)} = W_{XY}^{(0)} = \overline{x}_y = \overline{y}_x = M_{XY}^{(0)} = 0$

2. For *j*=1,..,*n* compute:

   $N_{XY}^{(j)} = N_{XY}^{(j-1)} + f_j I\left(x_j \text{ and } y_j \text{ are not missing}\right)$

   $W_{XY}^{(j)} = W_{XY}^{(j-1)} + f_j w_j I\left(x_j \text{ and } y_j \text{ are not missing}\right)$

   $v_{xj} = \frac{f_j w_j}{W_{XY}^{(j)}}\left(x_j - \overline{x}_y\right)$

   $\overline{x}_y = \overline{x}_y + v_{xj}$

   $v_{yj} = \frac{f_j w_j}{W_{XY}^{(j)}}\left(y_j - \overline{y}_x\right)$

   $\overline{y}_x = \overline{y}_x + v_{yj}$

   $M_{XY}^{(j)} = M_{XY}^{(j-1)} + \left(x_j - \overline{x}_y\right)\left(y_j - \overline{y}_x\right)\left(f_j w_j - \frac{\left(f_j w_j\right)^2}{W_{XY}^{(j)}}\right)$

   After the last case has been processed, we obtain:

   $M_{XY} = M_{XY}^{(n)} = \sum_{i=1}^{n} f_i w_i \left(x_i - \overline{x}_y\right)\left(y_i - \overline{y}_x\right)$

3. Compute bivariate statistics between *X* and *Y*:

   Number of valid cases: N_{XY}

   Covariance: $c_{XY} = \frac{M_{XY}}{N_{XY} - 1}$

   Correlation: $r_{XY} = \frac{c_{XY}}{\sqrt{c_{XX}}\sqrt{c_{YY}}}$

   Note: If there are no valid cases when pairwise deletion is used, then we let $c_{XY} = 0$ and $r_{XY} = 0$.

Categorical target and all continuous predictors

For a categorical target *Y* with values $i = 1, 2, \cdots, J$ and a continuous predictor *X* with values $x_1, \cdots x_n$, the bivariate statistics are:

Mean of *X* for each *Y*=*i*, *i*=1,...,*J*:

$$\overline{x}_{\cdot i} = \frac{\sum_{j=1}^{n} f_j w_j x_j I\left(y_j = i\right)}{\sum_{j=1}^{n} f_j w_j I\left(y_j = i\right)}$$

Sum of squared errors of *X* for each *Y*=*i*, *i*=1,...,*J*:

$$M_{\cdot i}^2 = \sum_{j=1}^{n} f_j w_j (x_j - \overline{x}_{\cdot\, i})^2 I\left(y_j = i\right)$$

Sum of frequency weight for each *Y*=*i*, *i*=1,...,*J*:

$$N_{\cdot i} = \sum_{j=1}^{n} f_j I\left(y_j = i \wedge x_j \text{ is not missing}\right)$$

Number of invalid cases

$$N_{XY} = \sum_{i=1}^{J} N_{\cdot i}$$

Sum of weights (frequency weight times analysis weight) for each *Y*=*i*, *i*=1,...,*J*:

$$W_{\cdot i} = \sum_{j=1}^{n} f_j w_i I\left(y_j = i \wedge x_j \text{ is not missing}\right)$$

Continuous target and all categorical predictors

For a continuous target *Y* and a categorical predictor *X* with values *i*=1,...,*J*, the bivariate statistics include:

Mean of *Y* conditional upon *X*:

$$\overline{y}_x = \frac{\sum_{i=1}^{I} \sum_{j=1}^{n} f_j w_j y_j I\left(x_j = i\right)}{\sum_{i=1}^{I} \sum_{j=1}^{n} f_j w_j I\left(x_j = i\right)}$$

Sum of squared errors of *Y*:

$$M_{X\cdot}^2 = \sum_{j=1}^{n} f_j w_j (y_j - \overline{y}_x)^2$$

Mean of *Y* for each $X = i$, *i*=1,...,*J*:

$$\overline{y}_{i\cdot} = \frac{\sum_{j=1}^{n} f_j w_j y_j I\left(x_j = i\right)}{\sum_{j=1}^{n} f_j w_j I\left(x_j = i\right)}$$

Sum of squared errors of Y for each $X = i$, i=1,...,J:

$$M_{i\cdot}^2 = \sum_{j=1}^{n} f_j w_j (y_j - \overline{y}_{i\cdot})^2 I\left(x_j = i\right)$$

Sum of frequency weights for $X = i$, i=1,...,J:

$$N_{i\cdot} = \sum_{j=1}^{n} f_j I\left(x_j = i \wedge y_j \text{ is not missing}\right)$$

Sum of weights (frequency weight times analysis weight) for $X = i$, i=1,...,J:

$$W_{i\cdot} = \sum_{j=1}^{n} f_j w_j I\left(x_j = i \wedge y_j \text{ is not missing}\right)$$

Categorical target and all categorical predictors

For a categorical target Y with values j=1,...,J and a categorical predictor X with values i=1,...,I, then bivariate statistics are:

Sum of frequency weights for each combination of $x_k = i$ and $y_k = j$:

$$N_{ij} = \sum_{k=1}^{n} f_k I\left(x_k = i \wedge y_k = j\right)$$

Sum of weights (frequency weight times analysis weight) for each combination of $x_k = i$ and $y_k = j$:

$$W_{ij} = \sum_{k\in 1}^{n} f_k w_k I\left(x_k = i \wedge y_k = j\right)$$

Categorical Variable Handling

In this step, we use univariate or bivariate statistics to handle categorical predictors.

Reordering Categories

For a nominal predictor, we rearrange categories from lowest to highest counts. If there is a tie on counts, then ties will be broken by ascending sort or lexical order of the data values. The new field values start with 0 as the least frequent category. Note that the new field will be numeric even if the original field is a string. For example, if a nominal field's data values are “A”, “A”, “A”, “B”, “C”, “C”, then automated data preparation would recode “B” into 0, “C” into 1, and “A” into 2.

Identify Highly Associated Categorical Features

If there is a target in the data set, we select a ordinal/nominal predictor if its *p*-value is not larger than an alpha-level $\alpha_{selection}$ (default is 0.05). See “P-value Calculations” for details of computing these *p*-values.

Since we use pairwise deletion to handle missing values when we collect bivariate statistics, we may have some categories with zero cases; that is, $N_{i.} = 0$ for a category *i* of a categorical predictor. When we calculate *p*-values, these categories will be excluded.

If there is only one category or no category after excluding categories with zero cases, we set the *p*-value to be 1 and this predictor will not be selected.

Supervised Merge

We merge categories of an ordinal/nominal predictor using a supervised method that is similar to a Chaid Tree with one level of depth.

1. Exclude all categories with zero case count.

2. If *X* has 0 categories, merge all excluded categories into one category, then stop.

3. If *X* has 1 category, go to step 7.

4. Else, find the allowable pair of categories of *X* that is most similar. This is the pair whose test statistic gives the largest *p*-value with respect to the target. An allowable pair of categories for an ordinal predictor is two adjacent categories; for a nominal predictor it is any two categories. Note that for an ordinal predictor, if categories between the *i*th category and *j*th categories are excluded because of zero cases, then the *i*th category and *j*th categories are two adjacent categories. See “P-value Calculations” for details of computing these *p*-values.

5. For the pair having the largest *p*-value, check if its *p*-value is larger than a specified alpha-level $\alpha_{selection}$ (default is 0.05). If it does, this pair is merged into a single compound category and at the same time we calculate the bivariate statistics of this new category. Then a new set of categories of *X* is formed. If it does not, then go to step 6.

6. Go to step 3.

7. For an ordinal predictor, find the maximum value in each new category. Sort these maximum values in ascending order. Suppose we have *r* new categories, and the maximum values are: $i_1 < i_2 < \cdots < i_r$, then we get the merge rule as: the first new category will contain all original categories such that $X \le i_1$, the second new category will contain all original categories such that $i_1 < X \le i_2$,…, and the last new category will contain all original categories such that $X > i_{r-1}$.

   For a nominal predictor, all categories excluded at step 1 will be merged into the new category with the lowest count. If there are ties on categories with the lowest counts, then ties are broken by selecting the category with the smallest value by ascending sort or lexical order of the original category values which formed the new categories with the lowest counts.

Bivariate statistics calculation of new category

When two categories are merged into a new category, we need to calculate the bivariate statistics of this new category.

Scale target. If the categories *i* and i' can be merged based on *p*-value, then the bivariate statistics should be calculated as:

$$N_{i,i'}. = N_i. + N_{i'}.$$

$$W_{i,i'}. = W_i. + W_{i'}.$$

$$\overline{y}_{i,i'}. = \overline{y}_i. + \frac{W_{i'}.}{W_{i,i'}.}(\overline{y}_{i'}. - \overline{y}_i.)$$

$$M^2_{i,i'}. = M^2_i. + M^2_{i'}. + W_i.(\overline{y}_{i,i'}. - \overline{y}_i.)^2 + W_{i'}.(\overline{y}_{i,i'}. - \overline{y}_{i'}.)^2$$

Categorical target. If the categories *i* and i' can be merged based on *p*-value, then the bivariate statistics should be calculated as:

$$N_{i,i'j} = N_{ij} + N_{i'j}$$

$$W_{i,i'j} = W_{ij} + W_{i'j}$$

Update univariate and bivariate statistics

At the end of the supervised merge step, we calculate the bivariate statistics for each new category. For univariate statistics, the counts for each new category will be sum of the counts of each original categories which formed the new category. Then we update other statistics according to the formulas in “Univariate Statistics Collection”, though note that the statistics only need to be updated based on the new categories and the numbers of cases in these categories.

P-value Calculations

Each *p*-value calculation is based on the appropriate statistical test of association between the predictor and target.

Scale target

We calculate an *F* statistic:

$$F = \frac{\sum_{i=1}^{I} W_{i\cdot}(\overline{y}_{i\cdot} - \overline{y}_x)^2 / (I-1)}{\sum_{i=1}^{I} M^2_{i\cdot} / \left(\sum_{i=1}^{I} N_{i\cdot} - I\right)}$$

where $\overline{y}_x = \frac{\sum_{i=1}^{I} W_{i\cdot}\overline{y}_{i\cdot}}{\sum_{i=1}^{I} W_{i\cdot}}$.

Based on *F* statistics, the *p*-value can be derived as

$$p = \Pr\left(F\left(I-1, \sum_{i=1}^{I} N_{i\cdot} - I\right) > F\right)$$

where $F\left(I-1, \sum_{i=1}^{I} N_{i\cdot} - I\right)$ is a random variable following a *F* distribution with $I-1$ and $\sum_{i=1}^{I} N_{i\cdot} - I$ degrees of freedom.

At the merge step we calculate the *F* statistic and *p*-value between two categories *i* and i' of *X* as

$$F = \frac{W_{i\cdot}\left(\overline{y}_{i\cdot} - \overline{y}_{i,i'\cdot}\right)^2 + W_{i'\cdot}\left(\overline{y}_{i'\cdot} - \overline{y}_{i,i'\cdot}\right)^2}{\left(M_{i\cdot}^2 + M_{i'\cdot}^2\right) / \left(N_{i\cdot} + N_{i'\cdot} - 2\right)}$$

$$p = \Pr\left(F\left(1, N_{i\cdot} + N_{i'\cdot} - 2\right) > F\right)$$

where $\overline{y}_{i,i'\cdot}$ is the mean of *Y* for a new category i, i' merged by *i* and i':

$$\overline{y}_{i,i'\cdot} = \overline{y}_{i\cdot} + \frac{W_{i'\cdot}}{W_{i\cdot} + W_{i'\cdot}}\left(\overline{y}_{i'\cdot} - \overline{y}_{i\cdot}\right)$$

and $F\left(I-1, N_{i\cdot} + N_{i'\cdot} - 2\right)$ is a random variable following a *F* distribution with 1 and $N_{i\cdot} + N_{i'\cdot} - 2$ degrees of freedom.

Nominal target

The null hypothesis of independence of *X* and *Y* is tested. First a contingency (or count) table is formed using classes of *Y* as columns and categories of the predictor *X* as rows. Then the expected cell frequencies under the null hypothesis are estimated. The observed cell frequencies and the expected cell frequencies are used to calculate the Pearson chi-squared statistic and the *p*-value:

$$X^2 = \sum_{j=1}^{J}\sum_{i=1}^{I} \frac{\left(N_{ij} - \hat{m}_{ij}\right)^2}{\hat{m}_{ij}}$$

where $N_{ij} = \sum_{k \in D} f_k I\left(x_k = i \wedge y_k = j\right)$ is the observed cell frequency and $\hat{m}_{ij}$ is the estimated expected cell frequency for cell $\left(x_k = i, y_k = j\right)$ following the independence model. If $\hat{m}_{ij} = 0$, then $\frac{\left(N_{ij} - \hat{m}_{ij}\right)^2}{\hat{m}_{ij}} = 0$. How to estimate $\hat{m}_{ij}$ is described below.

The corresponding *p*-value is given by $p = \Pr\left(\chi_d^2 > X^2\right)$, where χ_d^2 follows a chi-squared distribution with $d = (J-1)(I-1)$ degrees of freedom.

When we investigate whether two categories *i* and i' of *X* can be merged, the Pearson chi-squared statistic is revised as

$$X^2 = \sum_{j=1}^{J} \left(\frac{(N_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}} + \frac{(N_{i'j} - \hat{m}_{i'j})^2}{\hat{m}_{i'j}} \right)$$

and the *p*-value is given by $p = \Pr\left(\chi^2_{J-1} > X^2\right)$.

Ordinal target

Suppose there are *I* categories of *X*, and *J* ordinal categories of *Y*. Then the null hypothesis of the independence of *X* and *Y* is tested against the row effects model (with the rows being the categories of *X* and columns the classes of *Y*) proposed by Goodman (1979). Two sets of expected cell frequencies, $\hat{m}_{ij}$ (under the hypothesis of independence) and $\hat{\hat{m}}_{ij}$ (under the hypothesis that the data follow a row effects model), are both estimated. The likelihood ratio statistic is

$$H^2 = 2 \sum_{i=1}^{I} \sum_{j=1}^{J} H_{ij}^2$$

where

$$H_{ij}^2 = \begin{cases} \hat{\hat{m}}_{ij} \ln\left(\hat{\hat{m}}_{ij}/\hat{m}_{ij}\right) & \hat{\hat{m}}_{ij}/\hat{m}_{ij} > 0 \\ 0 & else \end{cases}$$

The *p*-value is given by $p = \Pr\left(\chi^2_{I-1} > H^2\right)$.

Estimated expected cell frequencies (independence assumption)

If analysis weights are specified, the expected cell frequency under the null hypothesis of independence is of the form

$$m_{ij} = \overline{w}_{ij}^{-1} \alpha_i \beta_j$$

where α_i and β_j are parameters to be estimated, and $\overline{w}_{ij} = \frac{W_{ij}}{N_{ij}}$ if $N_{ij} > 0$, otherwise $\overline{w}_{ij} = 1$.

Parameter estimates $\hat{\alpha}_i$, $\hat{\beta}_j$, and hence $\hat{m}_{ij}$, are obtained from the following iterative procedure.

1. $k = 0$, $\alpha_i^{(0)} = \beta_j^{(0)} = 1$, $m_{ij}^{(0)} = \overline{w}_{ij}^{-1}$
2. $\alpha_i^{(k+1)} = \frac{N_{i.}}{\sum_j \overline{w}_{ij}^{-1} \beta_j^{(k)}} = \alpha_i^{(k)} \frac{N_{i.}}{\sum_j m_{ij}^{(k)}}$
3. $\beta_j^{(k+1)} = \frac{N_{.j}}{\sum_i \overline{w}_{ij}^{-1} \alpha_i^{(k+1)}}$
4. $m_{ij}^{(k+1)} = \overline{w}_{ij}^{-1} \alpha_i^{(k+1)} \beta_j^{(k+1)}$

5. If $\max_{i,j}\left|m_{ij}^{(k+1)}-m_{ij}^{(k)}\right|<\epsilon$ (default is 0.001) or the number of iterations is larger than a threshold (default is 100), stop and output $\alpha_i^{(k+1)},\beta_j^{(k+1)}$ and $m_{ij}^{(k+1)}$ as the final estimates $\hat{\alpha}_i,\hat{\beta}_j,\hat{m}_{ij}$. Otherwise, $k=k+1$ and go to step 2.

Estimated expected cell frequencies (row effects model)

In the row effects model, scores for classes of *Y* are needed. By default, s_j^* (the order of a class of *Y*) is used as the class score. These orders will be standardized via the following linear transformation such that the largest score is 100 and the lowest score is 0.

$$s_j=100\left(s_j^*-s_{\min}^*\right)/\left(s_{\max}^*-s_{\min}^*\right)$$

Where $s_{\min}^*$ and $s_{\max}^*$ are the smallest and largest order, respectively.

The expected cell frequency under the row effects model is given by

$$m_{ij}=\overline{w}_{ij}^{-1}\alpha_i\beta_j\gamma_i$$

where $\overline{s}=\sum_{j=1}^{J}W_{.j}s_j/\sum_{j=1}^{J}W_{.j}$, in which $W_{.j}=\Sigma_iW_{ij}$, and α_i, β_j, and γ_i are unknown parameters to be estimated.

Parameter estimates $\hat{\alpha}_i,\hat{\beta}_j,\hat{\gamma}_i$ and hence $\hat{m}_{ij}$ are obtained from the following iterative procedure.

1. $k=0$, $\alpha_i^{(0)}=\beta_j^{(0)}=\gamma_i^{(0)}=1$, $m_{ij}^{(0)}=\overline{w}_{ij}^{-1}$
2. $\alpha_i^{(k+1)}=\dfrac{N_{.j}}{\sum_j\overline{w}_{ij}^{-1}\beta_j^{(k)}\left(\gamma_i^{(k)}\right)^{(s_j-\overline{s})}}=\alpha_i^{(k)}\dfrac{N_{i.}}{\sum_j m_{ij}^{(k)}}$
3. $\beta_j^{(k+1)}=\dfrac{N_{.j}}{\sum_i\overline{w}_{ij}^{-1}\alpha_i^{(k+1)}\left(\gamma_i^{(k)}\right)^{(s_j-\overline{s})}}$
4. $m_{ij}^*=\overline{w}_{ij}^{-1}\alpha_i^{(k+1)}\beta_j^{(k+1)}\left(\gamma_i^{(k)}\right)^{(s_j-\overline{s})}$, $G_i=1+\dfrac{\sum_j(s_j-\overline{s})\left(N_{ij}-m_{ij}^*\right)}{\sum_j(s_j-\overline{s})^2m_{ij}^*}$
5. $\gamma_i^{(k+1)}=\begin{cases}\gamma_i^{(k)}G_i & G_i>0\\ \gamma_i^{(k)} & \text{otherwise}\end{cases}$
6. $m_{ij}^{(k+1)}=\overline{w}_{ij}^{-1}\alpha_i^{(k+1)}\beta_j^{(k+1)}\left(\gamma_i^{(k+1)}\right)^{(s_j-\overline{s})}$
7. If $\max_{i,j}\left|m_{ij}^{(k+1)}-m_{ij}^{(k)}\right|<\epsilon$ (default is 0.001) or the number of iterations is larger than a threshold (default is 100), stop and output $\alpha_i^{(k+1)},\beta_j^{(k+1)},\gamma_i^{(k+1)}$ and $m_{ij}^{(k+1)}$ as the final estimates $\hat{\alpha}_i,\hat{\beta}_j,\hat{\gamma}_i,\hat{m}_{ij}$. Otherwise, $k=k+1$ and go to step 2.

Unsupervised Merge

If there is no target, we merge categories based on counts. Suppose that *X* has *I* categories which are sorted in ascending order. For an ordinal predictor, we sort it according to its values, while for nominal predictor we rearrange categories from lowest to highest count, with ties broken by ascending sort or lexical order of the data values. Let c_i be the number of cases for the *i*th category, and N_X be the total number of cases for *X*. Then we use the equal frequency method to merge sparse categories.

1. Start with $j_1 = j_2 = 1$ and *g*=1.

2. If $j_1 > I$, go to step 5.

3. If $\sum_{i=j_1}^{j_2} c_i < [b\% \times N_X]$, then $j_2 = j_2 + 1$; otherwise the original categories $j_1, j_1+1, \cdots, j_2$ will be merged into the new category *g* and let $j_1 = j_2 + 1$, $j_2 = j_1$ and $g = g + 1$, then go to step 2.

4. If $j_2 \geq I$, then merge categories using one of the following rules:

   i) If $g = 1$, then categories $1, 2, \cdots, I-1$ will be merged into category *g* and *I* will be left unmerged.

   ii) If *g*=2, then $j_1, j_1+1, \cdots, I$ will be merged into category *g*=2.

   iii) If *g*>2, then $j_1, j_1+1, \cdots, I$ will be merged into category $g-1$.

   If $j_2 < I$, then go to step 3.

5. Output the merge rule and merged predictor.

   After merging, one of the following rules holds:

   - Neither the original category nor any category created during merging has fewer than $[b\% \times N_X]$ cases, where *b* is a user-specified parameter satisfying $1 < b < 100$ (default is 10) and [*x*] denotes the nearest integer of *x*.
   - The merged predictor has only two categories.

Update univariate statistics. When original categories $j_1, j_1+1, \cdots, j_2$ are merged into one new category, then the number of cases in this new category will be $\sum_{i=j_1}^{j_2} c_j$. At the end of the merge step, we get new categories and the number of cases in each category. Then we update other statistics according to the formulas in "Univariate Statistics Collection", though note that the statistics only need to be updated based on the new categories and the numbers of cases in these categories.

Continuous Predictor Handling

Continuous predictor handling includes supervised binning when the target is categorical, predictor selection when the target is continuous and predictor construction when the target is continuous or there is no target in the dataset.

After handling continuous predictors, we collect univariate statistics for derived or constructed predictors according to the formulas in "Univariate Statistics Collection". Any derived predictors that are constant, or have all missing values, are excluded from further analysis.

Supervised Binning

If there is a categorical target, then we will transform each continuous predictor to an ordinal predictor using supervised binning. Suppose that we have already collected the bivariate statistics between the categorical target and a continuous predictor. Using the notations introduced in "Bivariate Statistics Collection", the homogeneous subset will be identified by the Scheffe method as follows:

If $|\overline{x}_{\cdot i} - \overline{x}_{\cdot j}| \leq c_{critical}$ then $\overline{x}_{\cdot i}$ and $\overline{x}_{\cdot j}$ will be a homogeneous subset, where if $c_{critical} = \max(\overline{x}_{\cdot i}) - \min(\overline{x}_{\cdot i}) \quad N_{XY} = J$; otherwise $c_{critical} = R * C$ where $R = \sqrt{2(J-1)F_{1-\alpha}(J-1, N_{XY}-J)}$ and $C = MS \times \sqrt{\frac{\sum_{i=1}^{J} 1/W_{\cdot i}}{J}}$, $MS = \sqrt{\frac{\sum_{i=1}^{J} M_{\cdot i}^2}{N_{XY} \quad J}}$.

The supervised algorithm follows:

1. Sort the means $\overline{x}_{\cdot i}$ in ascending order, denote as $\overline{x}_{\cdot(1)} \leq \overline{x}_{\cdot(2)} \leq \cdots \leq \overline{x}_{\cdot(J)}$.
2. Start with *i*=1 and *q*=*J*.
3. If $\left|x_{\cdot(q)} - \overline{x}_{\cdot(i)}\right| \leq c_{critical}$, then $\left\{\overline{x}_{\cdot(i)}, \cdots, x_{\cdot(q)}\right\}$ can be considered a homogeneous subset. At the same time we compute the mean and standard deviation of this subset: $\overline{x}_{\cdot(i,q)} = \frac{\sum_{k=i}^{q} W_{\cdot(k)}\overline{x}_{\cdot(k)}}{\sum_{k=i}^{q} W_{\cdot(k)}}$ and $sd_{\cdot(i,q)} = \sqrt{\frac{M_{(i,q)}^2}{\sum_{k=i}^{q} N_{\cdot(k)} - 1}}$, where $M_{(i,q)}^2 = \sum_{k=i}^{q} A_{\cdot(k)}$ and $A_{\cdot(k)} = M_{\cdot(k)}^2 + W_{\cdot(k)}\left(\overline{x}_{\cdot(i,q)} - \overline{x}_{\cdot(k)}\right)^2$, then set *i* = *q* + *1* and *q* = *J*; Otherwise *q* = *q* – *1*.
4. If $i \leq J$, go to step 3.
5. Else compute the cut point of bins. Suppose we have $r \leq J$ homogeneous subsets and we assume that the means of these subsets are $\overline{x}_{\cdot(1)}^*, \overline{x}_{\cdot(2)}^*, \cdots, \overline{x}_{\cdot(r)}^*$, and standard deviations are $sd_{\cdot(1)}^*, sd_{\cdot(2)}^*, \cdots, sd_{\cdot(r)}^*$, then the cut points between the *i*th and (*i*+1)th homogeneous subsets are computed as $cut_i = \overline{x}_{(i)}^* + \frac{sd_{(i)}^* + \epsilon}{\left(sd_{(i)}^* + sd_{(i+1)}^* + 2\epsilon\right)}\left(\overline{x}_{(i+1)}^* - \overline{x}_{(i)}^*\right)$.
6. Output the binning rules. Category 1: $X \leq cut_1$; Category 2: $cut_1 < X \leq cut_2$;…; Category : $cut_{r-1} < X$.

Feature Selection and Construction

If there is a continuous target, we perform predictor selection using *p*-values derived from the correlation or partial correlation between the predictors and the target. The selected predictors are grouped if they are highly correlated. In each group, we will derive a new predictor using principal component analysis. However, if there is no target, we will do not implement predictor selection.

To identify highly correlated predictors, we compute the correlation between a scale and a group as follows: suppose that *X* is a continuous predictor and continuous predictors $X_1, X_2, \cdots, X_m$ form a group *G*. Then the correlation between *X* and group *G* is defined as:

$$r_{XG} = \min\{|r_{XX_i}|, X_i \in G\}$$

where r_{XX_i} is correlation between *X* and X_i.

Let α_{group} be the correlation level at which the predictors are identified as groups. The predictor selection and predictor construction algorithm is as follows:

1. (Target is continuous and predictor selection is in effect) If the *p*-value between a continuous predictor and target is larger than a threshold (default is 0.05), then we remove this predictor from the correlation matrix and covariance matrix. See "Correlation and Partial Correlation" for details on computing these *p*-values.
2. Start with $\alpha_{group} = 0.9$ and *i*=1.
3. If $\alpha_{group} \leq 0.1$, stop and output all the derived predictors, their source predictors and coefficient of each source predictor. In addition, output the remaining predictors in the correlation matrix.
4. Find the two most correlated predictors such that their correlation in absolute value is larger than α_{group}, and put them in group *i*. If there are no predictors to be chosen, then go to step 9.
5. Add one predictor to group *i* such that the predictor is most correlated with group *i* and the correlation is larger than α_{group}. Repeat this step until the number of predictors in group *i* is greater than a threshold (default is 5) or there is no predictor to be chosen.
6. Derive a new predictor from the group *i* using principal component analysis. For more information, see the topic "Principal Component Analysis".
7. (Both predictor selection and predictor construction are in effect) Compute partial correlations between the other continuous predictors and the target, controlling for values of the new predictor. Also compute the *p*-values based on partial correlation. See "Correlation and Partial Correlation" for details on computing these *p*-values. If the *p*-value based on partial correlation between a continuous predictor and continuous target is larger than a threshold (default is 0.05), then remove this predictor from the correlation and covariance matrices.
8. Remove predictors that are in the group from the correlation matrix. Then let *i*=*i*+1 and go to step 4.
9. $\alpha_{group} = \alpha_{group} - 0.1$, then go to step 3.

Notes:

- If only predictor selection is needed, then only step 1 is implemented. If only predictor construction is needed, then we implement all steps except step 1 and step 7. If both predictor selection and predictor construction are needed, then all steps are implemented.
- If there are ties on correlations when we identify highly correlated predictors, the ties will be broken by selecting the predictor with the smallest index in dataset.

Principal Component Analysis

Let $X_1, X_2, \cdots, X_m$ be *m* continuous predictors. Principal component analysis can be described as follows:

1. Input $C_{m \times m}$, the covariance matrix of $X_1, X_2, \cdots, X_m$.
2. Calculate the eigenvectors and eigenvalues of the covariance matrix. Sort the eigenvalues (and corresponding eigenvectors) in descending order, $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m$.

3. Derive new predictors. Suppose the elements of the first component v_1 are $v_{11}, v_{12}, \cdots, v_{1m}$, then the new derived predictor is $\frac{v_{11}}{\sqrt{\lambda_1}}X_1 + \frac{v_{12}}{\sqrt{\lambda_1}}X_2 + \cdots + \frac{v_{1m}}{\sqrt{\lambda_1}}X_m$.

Correlation and Partial Correlation

Correlation and P-value

Let r_{XY} be the correlation between continuous predictor *X* and continuous target *Y*, then the *p*-value is derived form the *t* test:

$$p = \Pr\left(|t\left(N_{XY} - 2\right)| > t\right)$$

where $t\left(N_{XY} - 2\right)$ is a random variable with a *t* distribution with $N_{XY} - 2$ degrees of freedom, and $t = r_{XY}\sqrt{\frac{N_{XY}-2}{1-r_{XY}^2}}$. If $r_{XY}^2 = 1$, then set *p*=0; If $N_{XY} \leq 2$, then set *p*=1.

Partial correlation and P-value

For two continuous variables, *X* and *Y*, we can calculate the partial correlation between them controlling for the values of a new continuous variable Z:

$$r_{XY|Z} = \frac{r_{XY} - r_{XZ}r_{YZ}}{\sqrt{1 - r_{XZ}^2}\sqrt{1 - r_{YZ}^2}}$$

Since the new variable *Z* is always a linear combination of several continuous variables, we compute the correlation of *Z* and a continuous variable using a property of the covariance rather than the original dataset. Suppose the new derived predictor *Z* is a linear combination of original predictors $X_1, X_2, \cdots, X_m$:

$$Z = a_1X_1 + a_2X_2 + \cdots + a_mX_m$$

Then for any a continuous variable *X* (continuous predictor or continuous target), the correlation between *X* and *Z* is

$$r_{ZX} = \frac{c_{ZX}}{\sqrt{c_{ZZ}c_{XX}}}$$

where $c_{ZX} = \sum_{i=1}^{m} a_i c_{X_i X}$, and $c_{ZZ} = \sum_{i=1}^{m} a_i^2 c_{X_i X_i} + 2\sum_{i \neq j} a_i a_j c_{X_i X_j}$.

If $1 - r_{XZ}^2$ or $1 - r_{YZ}^2$ is less than 10^{-10}, let $r_{XY|Z} = 0$. If $r_{XY|Z}$ is larger than 1, then set it to 1; If $r_{XY|Z}$ is less than −1, then set it to −1. (This may occur with pairwise deletion). Based on partial correlation, the *p*-value is derived from the *t* test

$$p = \Pr\left(|t\left(N_{XY} - 3\right)| > t\right)$$

where $t\left(N_{XY} - 3\right)$ is a random variable with a *t* distribution with $N_{XY} - 3$ degrees of freedom, and $t = r_{XY|Z}\sqrt{\frac{N_{XY}-3}{1-r_{XY|Z}^2}}$. If $r_{XY|Z}^2 = 1$, then set *p*=0; if $N_{XY} \leq 3$, then set *p*=1.

Discretization of Continuous Predictors

Discretization is used for calculating predictive power and creating histograms.

Discretization for calculating predictive power

If the transformed target is categorical, we use the equal width bins method to discretize a continuous predictor into a number of bins equal to the number of categories of the target. Variables considered for discretization include:

- Scale predictors which have been recommended.
- Original continuous variables of recommended predictors.

Discretization for creating histograms

We use the equal width bins method to discretize a continuous predictor into a maximum of 400 bins. Variables considered for discretization include:

- Recommended continuous variables.
- Excluded continuous variables which have not been used to derive a new variable.
- Original continuous variables of recommended variables.
- Original continuous variables of excluded variables which have not been used to derive a new variable.
- Scale variables used to construct new variables. If their original variables are also continuous, then the original variables will be discretized.
- Date/time variables.

After discretization, the number of cases and mean in each bin are collected to create histograms.

Note: If an original predictor has been recast, then this recast version will be regarded as the "original" predictor.

Predictive Power

Collect bivariate statistics for predictive power

We collect bivariate statistics between recommended predictors and the (transformed) target. If an original predictor of a recommended predictor exists, then we also collect bivariate statistics between this original predictor and the target; if an original predictor has a recast version, then we use the recast version.

If the target is categorical, but a recommended predictor or its original predictor/recast version is continuous, then we discretize the continuous predictor using the method in "Discretization of Continuous Predictors" and collect bivariate statistics between the categorical target and the categorical predictors.

Bivariate statistics between the predictors and target are same as those described in “Bivariate Statistics Collection”.

Computing predictive power

Predictive power is used to measure the usefulness of a predictor and is computed with respect to the (transformed) target. If an original predictor of a recommended predictor exists, then we also compute predictive power for this original predictor; if an original predictor has a recast version, then we use the recast version.

Scale target. When the target is continuous, we fit a linear regression model and predictive power is computed as follows.

- Scale predictor: $r^2_{XY} = \left(\frac{c_{XY}}{\sqrt{c_{XX}}\sqrt{c_{YY}}}\right)^2$
- Categorical predictor: $1 - \frac{S_e}{S_T}$, where $S_e = \sum_{i=1}^{I} M_{i\cdot}^2$ and $S_T = \sum_{i=1}^{n} f_i w_i (y_i - \overline{y}_x)^2$.

Categorical target. If the (transformed) target is categorical, then we fit a naïve Bayes model and the classification accuracy will serve as predictive power. We discretize continuous predictors as described in “Discretization of Continuous Predictors”, so we only consider the predictive power of categorical predictors.

If N_{ij} is the of number cases where $X = i$ and $Y = j$, $N_{i\cdot} = \sum_{j=1}^{J} N_{ij}$, and $N_{\cdot j} = \sum_{i=1}^{I} N_{ij}$,

$$\chi^2 = \sum_{i=1}^{I}\sum_{j=1}^{J} \frac{\left(N_{ij} - \hat{N}_{ij}\right)^2}{\hat{N}_{ij}}$$

where $\hat{N}_{ij} = \frac{N_{i\cdot}N_{\cdot j}}{N_{XY}}$

and Cramer’s V is defined as

$$V = \left(\frac{\chi^2}{N_{XY}\left(\min(I, J) - 1\right)}\right)^{1/2}$$

References

Box, G. E. P., and D. R. Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society, Series B*, 26, 211–246.

Goodman, L. A. 1979. Simple models for the analysis of association in cross-classifications having ordered categories. *Journal of the American Statistical Association*, 74, 537–552.

BAYES ANOVA Algorithms

Bayesian One-WAY ANOVA Models

The model 1 can be viewed as a special case of the general multiple linear regression model:

$$\mathcal{M}_1 : \boldsymbol{y} = \mathbf{1}_n \alpha + \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}\,, \tag{1}$$

where $\boldsymbol{y} = (y_{11}, ..., y_{1n_1}, \dots, y_{k1}, ..., y_{kn_k})^{\mathrm{T}}$; $n = n_1 + ... + n_k$; $\alpha = \mu_k$; $\boldsymbol{\beta} = (\mu_1 - \mu_k, \mu_2 - \mu_k, \dots, \mu_{k-1} - \mu_k, 0)^{\mathrm{T}}$; and

$$\boldsymbol{X} = \begin{pmatrix} \mathbf{1}_{n_1} & \mathbf{0}_{n_1} & \dots & \mathbf{0}_{n_1} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{n_k} & \mathbf{0}_{n_k} & \dots & \mathbf{1}_{n_k} \end{pmatrix}. \tag{2}$$

Note that $\boldsymbol{\epsilon} \sim \mathbf{Normal}(\mathbf{0}, \sigma^2 \boldsymbol{I})$.

Let f_i denote the frequency weight for the i-th case in the n observations. A non-integer f_i is rounded to the nearest integer. For those values less than 0.5 or missing, the corresponding case will not be used. The effective sample in the data set is thus $N = \sum_{i=1}^{n} f_i$. If no weights are present, $N = n$. Note that the sufficient sample size to estimate is $N > k$.

Using Bayes Factor

Considering the multiple linear regression model $\mathcal{M}_1$ in Equation 1, we would like to compare this full model with a null model:

$$\mathcal{M}_0 : \boldsymbol{y} = \mathbf{1}_n \alpha + \boldsymbol{\epsilon}, \tag{3}$$

and test the null hypothesis $H_0 : \boldsymbol{\beta} = \mathbf{0}$. Note that α is a common parameter in both $\mathcal{M}_0$ and $\mathcal{M}_1$. We are interested in making inference on $\boldsymbol{\beta}$, but need to place appropriate priors on all of the unknown parameters including α, $\boldsymbol{\beta}$, and σ^2. In the following discussions, we let $\phi = 1/\sigma^2$, where ϕ denotes a precision parameter.

Note: The following sections are the same as the sections in the "Bayesian Inference on Multiple Linear Regression Models" document. The only difference is substituting p with $k-1$. Note that if we define

$$\boldsymbol{W} = \begin{pmatrix} f_1 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & f_n \end{pmatrix}, \tag{4}$$

then under one-way ANOVA setting, we have:

$$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X} = \begin{pmatrix} n_{1,f} & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & n_{k,f} \end{pmatrix}, \tag{5}$$

where $n_{i,f} = \sum_{j=n_1+...+n_{i-1}+1}^{n_1+...+n_i} f_j$, for $i = 1, ..., k$.

Zellner's Method

Zellner once suggested a g prior broadly discussed under $\mathcal{M}_1$ [Zellner, 1986]:

- $p(\alpha, \phi | \mathcal{M}_1) = 1/\phi$.
- $\boldsymbol{\beta} | (\phi, g, \mathcal{M}_1) \sim \mathbf{Normal}\left(\mathbf{0}, \frac{g}{\phi}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X})^{-1}\right)$, where g is fixed.

Since g is fixed, Zellner's g prior has the computational efficiency. Under these settings, the Bayes factor suggested by Zellner between $\mathcal{M}_1$ and $\mathcal{M}_0$ has a closed form

$$\Delta_{10}^{z} = (1+g)^{(N-k)/2}\left[1 + g(1-R^2)\right]^{-(N-1)/2}, \tag{6}$$

where $g > 0$, which is fixed and preset, and R^2 is the unadjusted proportion of variance accounted for by the factor which can be computed by applying the REGRESSION procedure on model 1.

Zellner-Siow's Method

Zellner and Siow proposed a Cauchy prior [Zellner and Siow, 1980], and can be represented as a mixture of priors with an Inverse-Gamma(1/2, N/2) prior on g: Under these settings, the Bayes factor suggested by Zellner and Siow between $\mathcal{M}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{s} = \int_0^\infty (1+g)^{(N-k)/2} \left[1+g(1-R^2)\right]^{-(N-1)/2} \left(\frac{\sqrt{N/2}}{\Gamma(1/2)}\, g^{-3/2} e^{-N/(2g)}\right) dg\,, \tag{7}$$

where $\Gamma(1/2)=\sqrt{\pi}$, and R^2 is defined the same as in the "Zellner's Method" section.

Hyper-g Method

Liang et al introduced a family of priors on g by specifying

$$p(g) = \frac{a-2}{2}(1+g)^{-a/2}\,, \tag{8}$$

where $g>0$ and $a>2$ for a proper distribution [Liang et al., 2012]: Under these settings, the Bayes factor suggested by Liang et al between $\mathcal{M}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{h}(a) = \frac{a-2}{2}\int_0^\infty (1+g)^{(N-k-a)/2} \left[1+g(1-R^2)\right]^{-(N-1)/2} dg \tag{9}$$

$$\tag{10}$$

where a is preset, R^2 is defined the same as in the "Zellner's Method" section.

Rouder's Method

Under these settings, the Bayes factor suggested by Rouder and Morey between $\mathcal{M'}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{r}(s) = \int_0^\infty (1+g)^{(N-k)/2} \left[1+g(1-R^2)\right]^{-(N-1)/2} \left(\frac{s\sqrt{N/2}}{\Gamma(1/2)}\, g^{-3/2} e^{-Ns^2/(2g)}\right) dg\,, \tag{11}$$

where $s>0$, $\Gamma(1/2)=\sqrt{\pi}$, and R^2 is defined the same as in the "Zellner's Method" section.

Characterizing Posterior Distributions

The model 1 can also be viewed as another form of special case of the general multiple linear regression model:

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}\,, \tag{12}$$

where $\boldsymbol{y} = (y_{11}, ..., y_{1n_1}, \ldots, y_{k1}, ..., y_{kn_k})^{\mathrm{T}}$; $n = n_1 + ... + n_k$; $\boldsymbol{\beta} = (\mu_1, \mu_2, \ldots, \mu_k)^{\mathrm{T}}$; and

$$\boldsymbol{X} = \begin{pmatrix} \mathbf{1}_{n_1} & \mathbf{0}_{n_1} & \ldots & \mathbf{0}_{n_1} \\ \vdots & \vdots & \vdots & \vdots \\ \mathbf{0}_{n_k} & \mathbf{0}_{n_k} & \ldots & \mathbf{1}_{n_k} \end{pmatrix}. \tag{13}$$

Note that $\boldsymbol{\epsilon} \sim \mathbf{Normal}(\mathbf{0}, \sigma^2 \boldsymbol{I})$. In the following presentation, we would like to mainly discuss how to make statistical inference on $\boldsymbol{\beta}$, and σ^2 by using Bayesian approaches.

We define the frequency weight f_i and the matrix $\boldsymbol{W}$ the same way as in the "Using Bayes Factor" section. Thus, we still have:

$$\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X} = \begin{pmatrix} n_{1,f} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & n_{k,f} \end{pmatrix}, \tag{14}$$

where $n_{i,f} = \sum_{j=n_1+...+n_{i-1}+1}^{n_1+...+n_i} f_j$, for $i = 1, ..., k$.

Using Conjugate Prior

We place a conjugate prior by assuming that

- $\sigma^2 \sim \text{Inverse-Gamma}(a_0, b_0)$,
- $\boldsymbol{\beta}|\sigma^2 \sim \textbf{Normal}\left(\boldsymbol{\beta}_0, \sigma^2 \boldsymbol{V}_0\right)$.

Group means $\boldsymbol{\beta}$: Under the setting of model 12, $\boldsymbol{\beta} = (\mu_1, \mu_2, \ldots, \mu_k)^{\mathrm{T}}$ represents the means of the k groups corresponding to the k categories of the factor. With the conjugate prior, the resulting marginal posterior distribution $\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}$ follows a scaled multivariate t distribution with ν degrees of freedom, where $\nu = 2a_0 + N$.

Before finding the Bayes estimator of $\boldsymbol{\beta}$, we define the following quantities:

$$\boldsymbol{\beta}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right)^{-1}\left(\boldsymbol{V_0}^{-1}\boldsymbol{\beta}_0 + \boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right) \,, \tag{15}$$

$$\boldsymbol{V}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right)^{-1} \,, \tag{16}$$

$$a_1 = a_0 + \frac{N}{2} \,, \tag{17}$$

$$b_1 = b_0 + \frac{1}{2}\left(\boldsymbol{\beta}_0^{\mathrm{T}}\boldsymbol{V}_0^{-1}\boldsymbol{\beta}_0 + \boldsymbol{y}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} - \boldsymbol{\beta}_1^{\mathrm{T}}\boldsymbol{V}_1^{-1}\boldsymbol{\beta}_1\right) \,. \tag{18}$$

Hence, assuming $\nu > 4$, the mode and posterior mean of group means are both:

$$\hat{\boldsymbol{\beta}} = \mathbb{E}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\beta}_1, \tag{19}$$

and the variance-covariance matrix

$$\mathbb{C}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu - 2}\frac{b_1}{a_1}\boldsymbol{V}_1 \,, \tag{20}$$

where $\boldsymbol{V}_1$, a_1, and b_1, are defined by Equations (16)-(18), and the diagonal elements are the variances of the elements in $\boldsymbol{\beta} = (\mu_1, \mu_2, \ldots, \mu_k)^{\mathrm{T}}$. Define

$$\boldsymbol{B^*} \equiv \begin{pmatrix} B^*_{11} & B^*_{12} & \ldots & B^*_{1k} \\ B^*_{21} & B^*_{22} & \ldots & B^*_{2k} \\ \vdots & \vdots & \vdots & \vdots \\ B^*_{k1} & B^*_{k2} & \ldots & B^*_{kk} \end{pmatrix} = \frac{b_1}{a_1}\boldsymbol{V}_1 \,, \tag{21}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_i such that

$$\mu_i \in \left(\hat{\mu}_i - \text{IDF.T}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{ii}}, \, \hat{\mu}_i + \text{IDF.T}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{ii}}\right) \,, \tag{22}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, k$, $\hat{\mu}_i$ is the ith element in $\hat{\boldsymbol{\beta}} = \mathbb{E}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y})$, B^*_{ii} is the ith element on the diagonal of $\boldsymbol{B^*}$, and IDF.T$(\cdot)$ is the IBM® SPSS® Statistics function for the inverse cumulative t distribution.

Variance of error terms $\boldsymbol{\sigma^2}$: Under the setting of conjugate priors, the marginal posterior distribution of σ^2 is

$$\sigma^2|\boldsymbol{X}, \boldsymbol{y} \sim \text{Inverse-Gamma}(a_1, b_1) \,, \tag{23}$$

where a_1 and b_1 are defined by Equations (17) and (18), respectively.

We may find the mode and posterior estimators of σ^2 by computing the expected value

$$\hat{\sigma}^2 = \mathbb{E}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{b_1}{a_1 - 1} \,, \tag{24}$$

for $a_1 > 1$, and the variance of the marginal posterior distribution of $\sigma^2|\boldsymbol{X}, \boldsymbol{y}$

$$\mathbb{C}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{b_1^2}{(a_1 - 1)^2(a_1 - 2)} \,, \tag{25}$$

for $a_1 > 2$. We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering σ^2 such that

$$\sigma^2 \in \Big(\text{IDF.GAMMA}^{-1}(1-\frac{c}{2}, a_1, b_1), \text{IDF.GAMMA}^{-1}(\frac{c}{2}, a_1, b_1) \Big) , \tag{26}$$

with the probability of c, where $c = 0.05$ by default, and IDF.GAMMA$(\cdot)$ is the IBM® SPSS® Statistics function for the inverse cumulative Gamma distribution.

Using Standard Noninformative Prior

By setting $\boldsymbol{V}_0^{-1} \to 0$, $a_0 = -k/2$, and $b_0 = 0$, it turns out that we place a reference (non-informative) prior by assuming that

$$p(\boldsymbol{\beta}, \sigma^2) \propto 1/\sigma^2 . \tag{27}$$

Group means $\boldsymbol{\beta}$: Under the setting of Equation (27), the resulting marginal posterior distribution $\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}$ follows a scaled multivariate t distribution with $\nu = N - k$ degrees of freedom. We can also find the mode and posterior estimators of $\boldsymbol{\beta}$ by computing the expected value

$$\hat{\boldsymbol{\beta}} = \mathbb{E}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}) = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right)^{-1} \boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} = (\bar{y}_1, ..., \bar{y}_k)^{\mathrm{T}} , \tag{28}$$

where $\bar{y}_i = \frac{\sum_{j=n_1+\ldots+n_{i-1}+1}^{n_1+\ldots+n_i} f_j y_j}{\sum_{j=n_1+\ldots+n_{i-1}+1}^{n_1+\ldots+n_i} f_j}$ and the variance-covariance matrix

$$\mathbb{C}(\boldsymbol{\beta}|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu-2} s^2 \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right)^{-1} = \frac{\nu}{\nu-2} s^2 \begin{pmatrix} \frac{1}{n_{1,f}} & 0 & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \ldots & \frac{1}{n_{k,f}} \end{pmatrix} , \tag{29}$$

where

$$\begin{aligned} s^2 = \frac{1}{\nu} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{X}\hat{\boldsymbol{\beta}}) &= \frac{1}{\nu} \sum_{i=1}^{k} \sum_{j=n_1+\ldots+n_{i-1}+1}^{n_1+\ldots+n_i} f_j (y_j - \bar{y}_i)^2 \\ &= \frac{1}{\nu} [\sum_{i=1}^{k} \sum_{j=n_1+\ldots+n_{i-1}+1}^{n_1+\ldots+n_i} f_j y_j^2 - \sum_{i=1}^{k} \bar{y}_i^2 n_{i,f}], \end{aligned} \tag{30}$$

and the diagonal elements are the variances of the elements in $\boldsymbol{\beta} = (\mu_1, \mu_2, \ldots, \mu_k)^{\mathrm{T}}$.

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_i such that

$$\mu_i \in \left(\hat{\mu}_i - \text{IDF.T}(1-\frac{c}{2}, \nu)\sqrt{\frac{s^2}{n_{i,f}}} , \hat{\mu}_i + \text{IDF.T}(1-\frac{c}{2}, \nu)\sqrt{\frac{s^2}{n_{i,f}}} \right) , \tag{31}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, k$, $\hat{\mu}_i$ is the ith element in $\hat{\boldsymbol{\beta}}$.

Variance of error terms $\boldsymbol{\sigma^2}$: Under the prior setting of (27), the marginal posterior distribution of σ^2 is

$$\sigma^2|\boldsymbol{X}, \boldsymbol{y} \sim \text{Inverse-}\chi^2(\nu, s^2) , \tag{32}$$

where $\nu = N - k$, and s^2 is defined by Equation (30).

We may find the mode and posterior estimators of σ^2, when $\nu > 4$, by computing the expected value

$$\hat{\sigma}^2 = \mathbb{E}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu-2} s^2 , \tag{33}$$

and the variance of the marginal posterior distribution of $\sigma^2|\boldsymbol{X}, \boldsymbol{y}$

$$\mathbb{C}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{2\nu^2}{(\nu-2)^2(\nu-4)} s^4 . \tag{34}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering σ^2 such that

$$\sigma^2 \in \left(\text{IDF.GAMMA}^{-1}(1-\frac{c}{2}, \frac{\nu}{2}, \frac{\nu}{2}s^2\,)\,, \text{IDF.GAMMA}^{-1}(\frac{c}{2}, \frac{\nu}{2}, \frac{\nu}{2}s^2\,) \right)\,, \tag{35}$$

with the probability of c, where $c = 0.05$ by default.

References

[George and Foster, 2000] George, E. and Foster, D. P. (2000). Calibration and empirical bayes variable selection. *Biometrika*, 87(4):731–747.

[Liang et al., 2012] Liang, F., Paulo, R., Molina, G., Clyde, M. A., and Berger, J. O. (2012). Mixtures of g priors for bayesian variable selection. *Journal of the American Statistical Association.*

[Rouder and Morey, 2012] Rouder, J. N. and Morey, R. D. (2012). Default bayes factors for model selection in regression. *Multivariate Behavioral Research*, 47(6):877–903.

[Zellner, 1986] Zellner, A. (1986). On assessing prior distributions and bayesian regression analysis with g-prior distributions. *Bayesian inference and decision techniques: Essays in Honor of Bruno De Finetti*, 6:233–243.

[Zellner and Siow, 1980] Zellner, A. and Siow, A. (1980). Posterior odds ratios for selected regression hypotheses. *Trabajos de estadística y de investigación operativa*, 31(1):585–603.

BAYES CORRELATION Algorithms

Bayesian Inference for Pearson Correlation

Basic Statistics and Quantities in Estimating Sample Correlation Coefficient

Notations

The following notations defined in this section will be used for the subsequent sections.

N: Number of cases.

x_i: Observed value of the scale variable $X = (X_1, X_2, \dots, X_N)$ for the i-th case.

y_i: Observed value of the scale variable $Y = (Y_1, Y_2, \dots, Y_N)$ for the i-th case

w_i: Weight for the i-th case. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

W_x: Sum of weights of cases used in computation of statistics for variable X. $W_x = N$ if no weights is present.

W_y: Sum of weights of cases used in computation of statistics for variable Y. $W_y = N$ if no weights is present.

W_{xy}: Sum of weights of cases used in computation of statistics for variables X and Y. $W_{xy} = N$ if no weights is present.

Basic Statistics and Quantities

Suppose there are a set of N ordered pairs of observations. We assume that the pairs are independent of each other, while the observations of the same pair, x_i and y_i may be correlated. To estimate the sample correlation coefficient r, we may need to compute the following statistics.

Estimated sample mean: $\bar{x} = \frac{1}{W_x} \sum_{i=1}^{N} w_i x_i$. (1)

Estimated sample mean: $\bar{y} = \frac{1}{W_y} \sum_{i=1}^{N} w_i y_i$. (2)

Estimated sample variance: $s_x^2 = \frac{1}{W_x - 1} \left[\sum_{i=1}^{N} w_i x_i^2 - \left(\sum_{i=1}^{N} w_i x_i \right)^2 / W_x \right]$. (3)

Estimated sample variance: $s_y^2 = \frac{1}{W_y - 1} \left[\sum_{i=1}^{N} w_i y_i^2 - \left(\sum_{i=1}^{N} w_i y_i \right)^2 / W_y \right]$. (4)

The estimated cross-product deviation for variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ is

$$C_{xy} = \sum_{i=1}^{N} w_i x_i y_i - \left(\sum_{i=1}^{N} w_i x_i \right) \left(\sum_{i=1}^{N} w_i y_i \right) / W_{xy} \, . \tag{5}$$

The estimated covariance is thus

$$\mathbb{C}(X, Y) = \frac{C_{xy}}{W_{xy} - 1} \, , \tag{6}$$

and the estimated Pearson correlation efficient is

$$r_{xy} = \frac{C_{xy}}{\sqrt{C_{xx} C_{yy}}} \, , \tag{7}$$

It is also convenient to define

$$S_{XX} = \sum_{i=1}^{N} w_i (x_i - \bar{x})^2 \,, \qquad S_{YY} = \sum_{i=1}^{N} w_i (y_i - \bar{y})^2 \,, \qquad \text{and} \qquad S_{XY} = \sum_{i=1}^{N} w_i (x_i - \bar{x})(y_i - \bar{y}) \,. \tag{8}$$

Hence, the estimated sample correlation coefficient is

$$r_{xy} = \frac{S_{XY}}{\sqrt{S_{XX} S_{YY}}} \,. \tag{9}$$

Bayesian Inference on the Correlation Coefficient

Using Bayes Factor

Bayes Factor Based on the JZS Prior

The Bayes factor suggested by [Wetzels and Wagenmakers, 2012] under the JZS prior is

$$\Delta_{10} = \frac{(W_{xy}/2)^{1/2}}{\Gamma(1/2)} \int_0^\infty (1+g)^{(W_{xy}-2)/2} [1 + (1-r^2)g]^{-(W_{xy}-1)/2} g^{-3/2} e^{-W_{xy}/(2g)} \, dg \,, \tag{10}$$

where $\Gamma(1/2) = \sqrt{\pi}$, and r ($|r| \neq 1$) is the sample correlation coefficient which can be estimated by either Equation (7) or Equation (9). Therefore, the Bayes factor in favor of the null hypothesis is $\Delta_{01} = 1/\Delta_{10}$, with Δ_{10} defined by Equation (10). In case that the two variables have a perfect linear correlation, or $|r| = 1$, the integral in Equation (7) does not converge. In this scenario, we do not estimate the Bayes factor based on the JZS prior. Note that the sufficient sample size to estimate the Bayes factor is $W_{xy} \geq 2$.

Fractional Bayes Factor

The Bayes factor suggested by [Kang et al., 2001] is

$$\Delta_{01} = \frac{I_1(x,y)}{I_2(x,y)} \cdot \frac{I_2(x,y;b)}{I_1(x,y;b)} \,, \tag{11}$$

where

$$I_1(x,y) = \int_0^\infty (1-\rho_0^2)^{(W_{xy}-1)/2} V^{-1} \left[V^{-1/2} + V^{1/2} - 2r\rho_0 \right]^{-(W_{xy}-1)} dV \,, \tag{12}$$

$$I_2(x,y) = \int_{-1}^{1} \int_0^\infty (1-\rho^2)^{(W_{xy}-3)/2} V^{-1} \left[V^{-1/2} + V^{1/2} - 2r\rho \right]^{-(W_{xy}-1)} dV d\rho \,, \tag{13}$$

$$I_1(x,y;b) = \int_0^\infty (1-\rho_0^2)^{(bW_{xy}-1)/2} V^{-1} \left[V^{-1/2} + V^{1/2} - 2r\rho_0 \right]^{-(bW_{xy}-1)} dV \,, \tag{14}$$

$$I_2(x,y;b) = \int_{-1}^{1} \int_0^\infty (1-\rho^2)^{(bW_{xy}-3)/2} V^{-1} \left[V^{-1/2} + V^{1/2} - 2r\rho \right]^{-(bW_{xy}-1)} dV d\rho \,, \tag{15}$$

and r is defined by Equation (7). Note that the fraction $b \in (0,1)$, which is preset and specified by users. Similar to what aforementioned in the previous section, in case that the two variables have a perfect linear correlation, or $|r| = 1$, we do not estimate the Bayes factor the fractional Bayes factor.

Characterizing Posterior Distributions

The sufficient sample size to estimate the posterior distribution is $W_{xy} \geq 2$. Suppose $\mathbb{E}(X) = \lambda$, $\mathbb{E}(Y) = \mu$, $\mathbb{V}(X) = \phi$, and $\mathbb{V}(Y) = \psi$. We assume and place standard reference priors on λ, μ, ϕ, and ψ. To derive the posterior density of ρ, we use the following substitution and approximation discussed in [Fisher, 1915] by noting that

$$\Pr(\rho | X, Y) \propto p(\rho) \frac{(1-\rho^2)^{(W_{xy}-1)/2}}{(1-\rho r)^{W_{xy}-3/2}} \,, \tag{16}$$

where $p(\rho)$ is the prior density placed on ρ. The common choice of the prior has the form

$$p(\rho) \propto (1-\rho^2)^c \,, \tag{17}$$

where $c = 0$ and $c = -3/2$ are two popular choices. Theoretically, uses are allowed to specify any arbitrary $c \in (-\infty, +\infty)$.

After making the hyperbolic tangent transformation

$$\rho = \tanh(\xi) \quad \text{and} \quad r = \tanh(z)\,, \tag{18}$$

where $\tanh(z) = \sinh(z)/\cosh(z) = (e^z - e^{-z})/(e^z + e^{-z})$ and $|r| \neq 1$, we will finally have

$$\xi \sim \text{Normal}\,(z, 1/W_{xy}) \text{ for large } W_{xy}. \tag{19}$$

We also suggest

$$\xi \sim \text{Normal}\left(z - \frac{5r}{2W_{xy}}, \frac{1}{W_{xy} - 1.5 + 2.5(1-r^2)}\right)\,, \tag{20}$$

which is a slightly better approximation when a uniform prior is placed on ρ. In practice, we can stick with Equation (20).

To find the Bayes estimators, we can simulate ξ based on Equation (19) or (20), and then transform to ρ by using $\rho = \tanh(\xi)$. Define

$$\rho^* = \left(\rho^{(1)}, \rho^{(2)}, \ldots, \rho^{(I)}\right)\,, \tag{21}$$

where I ($I = 10^4$ by default) is a larger integer input from syntax, denoting the posterior samples that we finally collect. We may find the Bayes estimators of ρ by computing the mode

$$\hat{\rho} = \max_{\rho}\left\{\Pr(\rho|X,Y)\right\}\,, \tag{22}$$

the expected value

$$\mathbb{E}(\rho|X,Y) = \int_{\rho} \rho \Pr(\rho|X,Y)\,d\rho \approx \mathbb{E}(\rho^*) = \frac{1}{I}\sum_{i=1}^{I}\rho^{(i)}\,, \tag{23}$$

and the variance of the marginal posterior distribution

$$\mathbb{V}(\rho|X,Y) = \int_{\rho} \rho \Pr(\rho|X,Y)\,d\rho - \left[\mathbb{E}(\rho|X,Y)\right]^2 \approx \frac{1}{I}\sum_{i=1}^{I}(\rho^{(i)})^2 - \left[\mathbb{E}(\rho|X)\right]^2\,. \tag{24}$$

We can compare the estimated $\mathbb{E}(\rho|X,Y)$ and the null value to see whether there is a significant difference between them. We may also use $\mathbb{V}(\rho|X,Y)$ to evaluate the precision of the expected value we have computed from the posterior distribution.

To maintain a more comprehensive Bayes estimator, we can construct a $100(1-\alpha)\%$ highest density region (HDR), which is the smallest interval with a mass of $100(1-\alpha)\%$ of the distribution. By definition, that is to find a Bayesian confidence interval satisfying

$$\int_{L_{\alpha/2}}^{H_{\alpha/2}} \Pr(\rho|X,Y)\,d\rho = 1-\alpha\,, \tag{25}$$

where the length of $(L_{\alpha/2}, H_{\alpha/2})$ is the shortest among all the candidate pairs.

References

[Fisher, 1915] Fisher, R. A. (1915). Frequency distribution of the values of the correlation coeffients in samples from an indefinitely large popu;ation. *Biometrika*, 10(4):507–521.

[Kang et al., 2001] Kang, S. G., Kim, D. H., and Lee, W. D. (2001). Default bayesian testing for the bivariate normal correlation coefficient. *Journal of Cell Biology*, 153(2):367–380.

[Wetzels and Wagenmakers, 2012] Wetzels, R. and Wagenmakers, E.-J. (2012). A default bayesian hypothesis test for correlations and partial correlations. *Psychonomic bulletin & review*, 19(6):1057–1064.

BAYES INDEPENDENT Algorithms

Two-Sample Bayesian Inference on Normal Distribution

Bayes-Factor Two-Sample Inference

Notations

The following notations defined in this section will be used for the subsequent sections.

k: Group index, $k = 1, 2$.

x_{ki}: Observed value of variable X for the i-th case in group k.

w_{ki}: Weight for the i-th case in group k. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

N_k: Number of cases in the data set for group k.

W_k: Sum of weights of cases in group k, $W_k = \sum_{i=1}^{N_k} w_{ki}$. $W_k = N_k$ if no weights are present.

Basic Statistics for Two-Sample Unpaired *t*-Test

The Bayes factor for one-sample t-test can be extended to a two-sample unpaired design. Correspondingly, the following statistics are computed in a conventional way.

Sample mean $\bar{x}_k = \dfrac{1}{W_k} \sum_{i=1}^{N_k} w_{ki} x_{ki}\,.$ (1)

Group mean difference $d = \bar{x}_2 - \bar{x}_1\,.$ (2)

Sample variance $s_k^2 = \dfrac{1}{W_k - 1} \left[\sum_{i=1}^{N_k} w_{ki} x_{ki}^2 - \left(\sum_{i=1}^{N_k} w_{ki} x_{ki} \right)^2 / W_k \right].$ (3)

Sample standard deviation $s_k = \sqrt{s_k^2}\,.$ (4)

Pooled Test Statistics

Pooled variance of the mean difference $s_p^2 = \dfrac{(W_1 - 1)s_1^2 + (W_2 - 1)s_2^2}{W_1 + W_2 - 2}\,.$ (5)

Pooled standard deviation of the mean difference $s_p = \sqrt{s_p^2}\,.$ (6)

Pooled standard error of the difference $s_d = s_p \sqrt{\dfrac{1}{W_1} + \dfrac{1}{W_2}}\,.$ (7)

Observed *t*-statistic with pooled variance $t = \dfrac{d}{s_d}$, with $(W_1 + W_2 - 2)$ degrees of freedom. (8)

Pooled significance (2-tailed) Sig. (2-tailed) $= 2\left[1 - \mathrm{CdfT}(|t|, W_1 + W_2 - 2)\right].$ (9)

Unpooled Test Statistics

Unpooled standard error of the difference $s_d = \sqrt{\dfrac{s_1^2}{W_1} + \dfrac{s_2^2}{W_2}}\,.$ (10)

Observed *t*-statistic with unpooled variance $t = \dfrac{d}{s_d}$, with $1/(Z_1 + Z_2)$ degrees of freedom, where (11)

$$Z_k = \left(\frac{s_k^2/W_k}{s_1^2/W_1 + s_2^2/W_2} \right)^2 /(W_k - 1)\,. \tag{12}$$

Unpooled significance (2-tailed) Sig. (2-tailed) $= 2\left[1 - \text{CdfT}(|t|, 1/(Z_1 + Z_2))\right]$. (13)

Rouder's Method

The Bayes factor for two-sample unpaired t-test under the Rouder's method is

$$r\text{BF}_{01} = \frac{\left(1 + \dfrac{t^2}{\nu}\right)^{-(\nu+1)/2}}{\displaystyle\int_0^\infty (1+Ng)^{-1/2} \left(1 + \frac{t^2}{(1+Ng)\nu}\right)^{-(\nu+1)/2} (2\pi)^{-1/2} g^{-3/2} e^{-1/(2g)}\, dg}\,, \tag{14}$$

where t is the pooled-variance two-sample t-statistic defined by Equation (8); $N = W_1W_2/(W_1 + W_2)$; $\nu = W_1 + W_2 - 2$; and g is the variable to be integrated out.

Gönen's Method

The Bayes factor for two-sample unpaired t-test under the Gönen's method is

$$g\text{BF}_{01} = \frac{T_\nu(t|0,1)}{T_\nu(t|\sqrt{N}\lambda, 1 + N\sigma_\delta^2)} = \frac{\text{PDF.T}(t,\nu)\sqrt{1+N\sigma_\delta^2}}{\text{NPDF.T}\left(t/\sqrt{1+N\sigma_\delta^2}\,, \nu, \sqrt{N}\lambda/\sqrt{1+N\sigma_\delta^2}\right)}\,, \tag{15}$$

where t is the pooled-variance two-sample t-statistic defined by Equation (8); $\nu = W_1 + W_2 - 2$; $N = W_1W_2/(W_1+W_2)$; λ and σ_δ^2 denote the prior mean and variance of $(\mu_1 - \mu_2)/\sigma$; $T_\nu(\cdot)$ denotes the noncentral t probability density function; and PDF.T$(\cdot)$ and NPDF.T are the IBM® SPSS® Statistics probability density functions for the (noncentral) t distribution.

It is quite natural to assume that $\lambda = 0$. For the case where the prior mean of the effect size is assumed to be zero, Equation (15) can be reduced to

$$g\text{BF}_{01,\lambda=0} = \left(\frac{1 + t^2/\nu}{1 + t^2/\left[\nu(1+N\sigma_\delta^2)\right]} \right)^{-(\nu+1)/2} \sqrt{1 + N\sigma_\delta^2}\,. \tag{16}$$

The sufficient sample size to estimate is $W_1, W_2 > 1$.

Hyper-Prior Method

The Bayes factor for two-sample unpaired t-test under the hyper prior of σ_δ^2 is

$$p\text{BF}_{10,\lambda=0} = \int_0^\infty \left(\frac{1 + t^2/\nu}{1 + t^2/\left[\nu(1+N\sigma_\delta^2)\right]} \right)^{(\nu+1)/2} \left(1 + N\sigma_\delta^2\right)^{-1/2} \pi(\sigma_\delta^2)\, d\sigma_\delta^2\,. \tag{17}$$

Set $\kappa = N$ and $b = \dfrac{\nu+1}{2} - a - \dfrac{5}{2}$, Equation (17) can be reduced to a closed form

$$p\text{BF}_{10,\lambda=0} = \frac{\Gamma(\nu/2)\,\Gamma(a+3/2)}{\Gamma\big((\nu+1)/2\big)\,\Gamma(a+1)} \left(1 + \frac{t^2}{\nu}\right)^{(\nu-2a-2)/2}\,, \tag{18}$$

where $\Gamma(\cdot)$ denotes the Gamma function, t is the pooled-variance two-sample t-statistic defined by Equation (8); $\nu = W_1 + W_2 - 2$; a is input by users ($a = -0.75$ is the setting by default), and it is recommended that the choice of $a \in (-1, -1/2]$ [Wang and Liu, 2016]. Note that when we output the Bayes factor estimated by using the Hyper-prior method, the value has to be in favor of H_1 versus H_0.

The sufficient sample size to estimate is $W_1, W_2 > 1$.

Bayesian Two-Sample Inference By Estimating Posterior Distributions

Bayesian Two-Sample Inference Using Conjugate and Noninformative Priors

Notations

The following notations defined in this section will be used for the subsequent sections.

X: A random variable to be tested whose values are observed for Group 1. We assume $X \sim \text{Normal}(z_{\mu_x}, \sigma_x^2)$.

Y: A random variable to be tested whose values are observed for Group 2. We assume $Y \sim \text{Normal}(z_{\mu_y}, \sigma_y^2)$.

z_{μ_x}: Mean parameter of X.

z_{μ_y}: Mean parameter of Y.

d_μ: Mean parameter of $Y - X$, where $d_\mu = z_{\mu_y} - z_{\mu_x}$.

σ_x: Standard deviation parameter of X.

σ_y: Standard deviation parameter of Y.

N_x: Number of cases in the data set for group X.

N_y: Number of cases in the data set for group Y.

w_{xj}: Weight for the j-th case in X. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

w_{yi}: Weight for the i-th case in Y. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

W_x: Sum of weights of cases $W_x = \sum_{i=1}^{N_x} w_{xi}$. $W_x = N_x$ if no weights are present.

W_y: Sum of weights of cases $W_y = \sum_{i=1}^{N_y} w_{yi}$. $W_y = N_y$ if no weights are present.

Diffuse Priors with Known Variances

In this section, we assume that both σ_x^2 and σ_y^2 are known, and place the independent diffuse priors by noting that $p(z_{\mu_x}|\sigma_x^2) \propto 1$ and $p(z_{\mu_y}|\sigma_y^2) \propto 1$.

Under this setting, we are interested in drawing inference on d_μ. Thus, the marginal posterior distribution of d_μ is

$$d_\mu|(X, Y) \sim \text{Normal}(\mu_n, \sigma_n^2), \tag{19}$$

where

$$\mu_n = \bar{y} - \bar{x} = \frac{1}{W_y}\sum_{i=1}^{N_y} w_{yi}y_i - \frac{1}{W_x}\sum_{j=1}^{N_x} w_{xj}x_j, \quad \text{and} \quad \sigma_n^2 = \frac{\sigma_y^2}{W_y} + \frac{\sigma_x^2}{W_x}. \tag{20}$$

We may find the Bayes estimators of d_μ by computing the mode

$$\hat{d}_\mu = \bar{y} - \bar{x}, \tag{21}$$

the expected value

$$\mathbb{E}[\, d_\mu|(X, Y)\,] = \bar{y} - \bar{x}, \tag{22}$$

and the variance of the marginal posterior distribution of $d_\mu|(X,Y)$

$$\mathbb{V}\,[\,d_\mu|(X,Y)\,] = \sigma_n^2 \;. \tag{23}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering d_μ such that

$$d_\mu \in \Big(\,\text{IdfNorm}(\frac{c}{2},\, \mu_n,\, \sqrt{\sigma_n^2}\,)\,,\, \text{IdfNorm}(1-\frac{c}{2},\, \mu_n,\, \sqrt{\sigma_n^2}\,)\Big) \tag{24}$$

with the probability of c, where $c = 0.05$ by default.

The sufficient sample size to estimate is $W_x, W_y > 0$.

Normal Priors with Known Variances

In this section, we assume that both σ_x^2 and σ_y^2 are known, and place the independent normal priors $z_{\mu_x} \sim \text{Normal}(\mu_{x_0}, \sigma_{x_0}^2)$ and $z_{\mu_y} \sim \text{Normal}(\mu_{y_0}, \sigma_{y_0}^2)$.

Under this setting, we are interested in drawing inference on d_μ. Thus, the marginal posterior distribution of d_μ is

$$d_\mu|(X,Y) \sim \text{Normal}\left(\mu_{y_n} - \mu_{x_n}, \sigma_{y_n}^2 + \sigma_{x_n}^2\right) \;, \tag{25}$$

where

$$\sigma_{y_n}^2 = \left(\frac{1}{\sigma_{y_0}^2} + \frac{W_y}{\sigma_y^2}\right)^{-1} \;, \qquad \mu_{y_n} = \sigma_{y_n}^2\left(\frac{\mu_{y_0}}{\sigma_{y_0}^2} + \frac{\bar{y}W_y}{\sigma_y^2}\right) \;, \tag{26}$$

and

$$\sigma_{x_n}^2 = \left(\frac{1}{\sigma_{x_0}^2} + \frac{W_x}{\sigma_x^2}\right)^{-1} \;, \qquad \mu_{x_n} = \sigma_{x_n}^2\left(\frac{\mu_{x_0}}{\sigma_{x_0}^2} + \frac{\bar{x}W_x}{\sigma_x^2}\right) \;. \tag{27}$$

The computation of $\bar{x}$ and $\bar{y}$ is the same as in Equation (20). We may find the Bayes estimators of d_μ by computing the mode

$$\hat{d}_\mu = \mu_{y_n} - \mu_{x_n} \;, \tag{28}$$

the expected value

$$\mathbb{E}(d_\mu|X,Y) = \mu_{y_n} - \mu_{x_n} \;, \tag{29}$$

and the variance of the marginal posterior distribution of $d_\mu|(X,Y)$

$$\mathbb{V}(d_\mu|X,Y) = \sigma_{y_n}^2 + \sigma_{x_n}^2 \;. \tag{30}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering d_μ such that

$$d_\mu \in \Big(\,\text{IdfNorm}(\frac{c}{2},\, \mu_{y_n} - \mu_{x_n},\, \sqrt{\sigma_{y_n}^2 + \sigma_{x_n}^2}\,)\,,\, \text{IdfNorm}(1-\frac{c}{2},\, \mu_{y_n} - \mu_{x_n},\, \sqrt{\sigma_{y_n}^2 + \sigma_{x_n}^2}\,)\Big) \tag{31}$$

with the probability of c, where $c = 0.05$ by default.

The sufficient sample size to estimate is $W_x, W_y > 0$.

Diffuse-Jeffreys Priors with Equal Variances

In this section, we assume that $\sigma_x^2 = \sigma_y^2 = \sigma^2$, and $p(\sigma^2) \propto 1/\sigma^2$. We place the diffuse priors by noting that $p(z_{\mu_x}|\sigma^2) \propto 1$ and $p(z_{\mu_y}|\sigma^2) \propto 1$.

Under this setting, we are interested in drawing inference on d_μ. Thus, the marginal posterior distribution of d_μ is

$$d_\mu|(X,Y) \sim t_{\nu_n}(\mu_n, \sigma_n^2)\,, \tag{32}$$

where

$$\nu_n = W_x + W_y - 2\,, \tag{33}$$

$$\mu_n = \bar{y} - \bar{x} = \frac{1}{W_y}\sum_{i=1}^{N_y} w_{yi} y_i - \frac{1}{W_x}\sum_{j=1}^{N_x} w_{xj} x_j \, , \tag{34}$$

and

$$\sigma_n^2 = \frac{1}{\nu_n}\left(\sum_{i=1}^{N_y} w_{yi}(y_i - \bar{y})^2 + \sum_{j=1}^{N_x} w_{xj}(x_j - \bar{x})^2\right)\left(\frac{1}{W_y} + \frac{1}{W_x}\right) . \tag{35}$$

We may find the Bayes estimators of d_μ by computing the mode

$$\hat{d}_\mu = \bar{y} - \bar{x} \ , \tag{36}$$

the expected value

$$\mathbb{E}\left[\, d_\mu | (X, Y)\,\right] = \bar{y} - \bar{x} \ , \tag{37}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}\left[\, d_\mu | (X, Y)\,\right] = \frac{\nu_n}{\nu_n - 2}\,\sigma_n^2 \ . \tag{38}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \left(\,\mu_n - \mathrm{IdfT}(1 - \frac{c}{2},\, \nu_n)\sqrt{\sigma_n^2}\,,\, \mu_n + \mathrm{IdfT}(1 - \frac{c}{2},\, \nu_n)\sqrt{\sigma_n^2}\,\right) \tag{39}$$

with the probability of c, where $c = 0.05$ by default.

The sufficient sample size to estimate is $W_x, W_y > 1$.

Diffuse-Inverse Chi-Square Priors with Equal Variances

In this section, we assume that $\sigma_x^2 = \sigma_y^2 = \sigma^2$ and $\sigma^2 \sim$ Inverse-$\chi^2(\nu_0, \sigma_0^2)$, and rewrite $X \sim \mathrm{Normal}(z_{\mu_x}, \sigma^2)$ and $Y \sim \mathrm{Normal}(z_{\mu_x} + d_\mu, \sigma^2)$. We place the diffuse priors by noting that $p(z_{\mu_x}|\sigma^2) \propto 1$ and $p(z_{\mu_y}|\sigma^2) \propto 1$.

Under this setting, we are interested in drawing inference on d_μ. Thus, the marginal posterior distribution of d_μ is

$$d_\mu|(X, Y) \sim t_{\nu_n}(\mu_n, \sigma_n^2)\, , \tag{40}$$

where

$$\nu_n = \nu_0 + W_x + W_y - 2\, , \qquad \mu_n = \bar{y} - \bar{x} = \frac{1}{W_y}\sum_{i=1}^{N_y} w_{yi} y_i - \frac{1}{W_x}\sum_{j=1}^{N_x} w_{xj} x_j \, , \tag{41}$$

and

$$\sigma_n^2 = \frac{1}{\nu_n}\left(\nu_0\sigma_0^2 + \sum_{i=1}^{N_y} w_{yi}(y_i - \bar{y})^2 + \sum_{j=1}^{N_x} w_{xj}(x_j - \bar{x})^2\right)\left(\frac{1}{W_y} + \frac{1}{W_x}\right) . \tag{42}$$

We may find the Bayes estimators of d_μ by computing the mode

$$\hat{d}_\mu = \bar{y} - \bar{x} \ , \tag{43}$$

the expected value

$$\mathbb{E}\left[\, d_\mu | (X, Y)\,\right] = \bar{y} - \bar{x} \ , \tag{44}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}\left[\, d_\mu | (X, Y)\,\right] = \frac{\nu_n}{\nu_n - 2}\,\sigma_n^2 \ . \tag{45}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \left(\,\mu_n - \mathrm{IdfT}(1 - \frac{c}{2},\, \nu_n)\sqrt{\sigma_n^2}\,,\, \mu_n + \mathrm{IdfT}(1 - \frac{c}{2},\, \nu_n)\sqrt{\sigma_n^2}\,\right) \tag{46}$$

with the probability of c, where $c = 0.05$ by default.

The sufficient sample size to estimate is $W_x, W_y > 1$.

Diffuse Priors with Unequal Variances

In this section, we do not make any assumptions on the equality of σ_x^2 and σ_y^2. We place the diffuse priors on all of the parameters by noting that $p(z_{\mu_x}, z_{\mu_x}, \sigma_x^2, \sigma_y^2) \propto 1$.

Define

$$\psi = \arctan\left(\frac{s_y/\sqrt{W_y}}{s_x/\sqrt{W_x}}\right), \tag{47}$$

where

$$s_y = \left(\sum_{i=1}^{N_y} w_{yi}(y_i - \bar{y})^2/(W_y - 1)\right)^{1/2}, \quad \text{and} \quad s_x = \left(\sum_{j=1}^{N_x} w_{xj}(x_j - \bar{x})^2/(W_x - 1)\right)^{1/2}. \tag{48}$$

Note that

$$T = \frac{d_\mu - (\bar{y} - \bar{x})}{\sqrt{s_x^2/W_x + s_y^2/W_y}} = T_y \sin\psi - T_x \cos\psi\,, \tag{49}$$

where

$$T_x = \frac{z_{\mu_x} - \bar{x}}{s_x/\sqrt{W_x}} \sim t_{W_x - 1}\,, \qquad T_y = \frac{z_{\mu_y} - \bar{y}}{s_y/\sqrt{W_y}} \sim t_{W_y - 1}\,, \tag{50}$$

and ψ is defined by Equation (47). Hence,

$$T \sim \text{Behrens-Fisher}(W_y - 1, W_x - 1, \psi)\,. \tag{51}$$

In practice, we have to approximate Equation (51) according to the dicussions in [Patil, 1965] by finding

$$\begin{aligned}
\eta_1 &= \left(\frac{W_y - 1}{W_y - 3}\right)\sin^2\psi + \left(\frac{W_x - 1}{W_x - 3}\right)\cos^2\psi\,,\\
\eta_2 &= \frac{(W_y - 1)^2}{(W_y - 3)^2(W_y - 5)}\sin^4\psi + \frac{(W_x - 1)^2}{(W_x - 3)^2(W_x - 5)}\cos^4\psi\,,\\
\eta_3 &= 4 + \eta_1^2/\eta_2\,, \quad \text{and} \quad \eta_4 = \sqrt{\eta_1(\eta_3 - 2)/\eta_3}\,.
\end{aligned} \tag{52}$$

Under this setting, the marginal posterior distribution of d_μ is

$$d_\mu|(X,Y) \sim t_{\nu_n}(\mu_n, \sigma_n^2)\,, \tag{53}$$

where

$$\nu_n = \eta_3\,, \quad \mu_n = \bar{y} - \bar{x} = \frac{1}{W_y}\sum_{i=1}^{N_y} w_{yi} y_i - \frac{1}{W_x}\sum_{j=1}^{N_x} w_{xj} x_j\,, \quad \text{and} \quad \sigma_n^2 = \eta_4^2\left(s_x^2/W_x + s_y^2/W_y\right). \tag{54}$$

We may find the Bayes estimators of d_μ by computing the mode

$$\hat{d}_\mu = \bar{y} - \bar{x}\,, \tag{55}$$

the expected value

$$\mathbb{E}\,[\,d_\mu|(X,Y)\,] = \bar{y} - \bar{x}\,, \tag{56}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}\,[\,d_\mu|(X,Y)\,] = \frac{\nu_n}{\nu_n - 2}\,\sigma_n^2\,. \tag{57}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \left(\mu_n - \text{IdfT}(1 - \frac{c}{2}, \nu_n)\sqrt{\sigma_n^2}\,, \mu_n + \text{IdfT}(1 - \frac{c}{2}, \nu_n)\sqrt{\sigma_n^2}\right) \tag{58}$$

with the probability of c, where $c = 0.05$ by default.

The sufficient sample size to estimate is $W_x, W_y > 5$.

References

[Gönen et al., 2005] Gönen, M., Johnson, W. O., Lu, Y., and Westfall, P. H. (2005). The bayesian two-sample t test. *The American Statistician*, 59(3):252–257.

[Kruschke, 2013] Kruschke, J. K. (2013). Bayesian estimation supersedes the t test. *Journal of Experimental Psychology: General*, 142(2):573.

[Patil, 1965] Patil, V. (1965). Approximation to the behrens-fisher distributions. *Biometrika*, 52(1/2):267–271.

[Rouder et al., 2009] Rouder, J. N., Speckman, P. L., Sun, D., Morey, R. D., and Iverson, G. (2009). Bayesian t tests for accepting and rejecting the null hypothesis. *Psychonomic bulletin & review*, 16(2):225–237.

[Wang and Liu, 2016] Wang, M. and Liu, G. (2016). A simple two-sample bayesian t-test for hypothesis testing. *The American Statistician*, 70(2):195–201.

BAYES LOGLINEAR Algorithms

Bayesian Inference for the Independence of Two Factors

General Notations

We desire to test the null hypothesis H_0: No association between rows and columns versus H_1: They are associated. The following notations defined in this section will be used for the subsequent sections.

r: $r = 1, 2, \ldots, R$ denoting the non-empty row index, where $R \geq 2$, and R is an integer.

s: $s = 1, 2, \ldots, S$ denoting the non-empty column index, where $S \geq 2$, and S is an integer.

$\boldsymbol{y}_{**}$: A matrix containing all of the observed cell counts with

$$\boldsymbol{y}_{**} \equiv \begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1S} \\ y_{21} & y_{22} & \cdots & y_{2S} \\ \vdots & \vdots & \vdots & \vdots \\ y_{R1} & y_{R2} & \cdots & y_{RS} \end{pmatrix}, \tag{1}$$

where y_{rs} must be a nonnegative integer.

$\overrightarrow{\boldsymbol{y}}$: $\overrightarrow{\boldsymbol{y}} = (y_{11}, y_{12}, \ldots, y_{RS})^{\mathrm{T}}$, a vectorized $\boldsymbol{y}_{**}$ containing all of the observed cell counts.

y_{rs}: Observed count data in the cell on the r-th row and the s-th column of the contingency table. Note that $y_{rs} \geq 0$, and y_{rs} is an integer.

$y_{r.}$: $y_{r.} = \sum_{s=1}^{S} y_{rs}$, the marginal total of the r-th row.

$y_{.s}$: $y_{.s} = \sum_{r=1}^{R} y_{rs}$, the marginal total of the s-th column.

Y: $Y = \sum_{r=1}^{R} \sum_{s=1}^{S} y_{rs}$, the total count of the cells.

$\hat{y}_{rs}$: Expected count in the cell on the r-th row and the s-th column of the contingency table. $\hat{y}_{rs} = y_{r.}y_{.s}/Y$.

$\boldsymbol{y}_{.*}$: $\boldsymbol{y}_{.*} = (y_{.1}, y_{.2}, \ldots, y_{.S})^{\mathrm{T}}$, a vector containing marginal column sums, where $S \geq 2$.

$\boldsymbol{y}_{*.}$: $\boldsymbol{y}_{*.} = (y_{1.}, y_{2.}, \ldots, y_{R.})^{\mathrm{T}}$, a vector containing marginal row sums, where $R \geq 2$.

Bayesian Inference by Using Bayes Factors

To implement the following methods, we require the two factors both have the number of categories ≥ 2 to formulate a valid two-way contingency table. Otherwise, we give a warning message, and do not conduct any further Bayesian analyses.

Bayes Factors Based on Natural Conjugate Priors

[Gunel and Dickey, 1974] proposed a unified approach when considering the association between two factors in a contingency table under the different model settings. The general idea is to assume conjugate gamma priors for Poisson models, and then extend to the other further conditioned models.

We let $\boldsymbol{a}_{**}$ denote a matrix of prior shape parameters with the same dimension as $\boldsymbol{y}_{**}$

$$\boldsymbol{a}_{**} \equiv \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1S} \\ a_{21} & a_{22} & \cdots & a_{2S} \\ \vdots & \vdots & \vdots & \vdots \\ a_{R1} & a_{R2} & \cdots & a_{RS} \end{pmatrix}, \tag{2}$$

where $a_{rs} > 0$. $a_{rs} = 1$ is the setting by default. Users can overwrite this setting by specifying different values, the number of which must match that of $\boldsymbol{y}_{**}$. We further define the following notations:

$\overrightarrow{\boldsymbol{a}}$: $\overrightarrow{\boldsymbol{a}} = (a_{11}, a_{12}, \ldots, a_{RS})^{\mathrm{T}}$, a vectorized $\boldsymbol{a}_{**}$ containing all of the prior shape parameters.

A: $A = \sum_{r=1}^{R} \sum_{s=1}^{S} a_{rs}$, the total count of the cells.

X: $X = A - (R-1)(S-1)$.

$\boldsymbol{a}_{.*}$: $\boldsymbol{a}_{.*} = (a_{.1}, a_{.2}, \ldots, a_{.S})^{\mathrm{T}}$, a vector containing marginal column sums, where $S \geq 2$.

$\boldsymbol{a}_{*.}$: $\boldsymbol{a}_{*.} = (a_{1.}, a_{2.}, \ldots, a_{R.})^{\mathrm{T}}$, a vector containing marginal row sums, where $R \geq 2$.

$\boldsymbol{\xi}_{.*}$: $\boldsymbol{\xi}_{.*} = \boldsymbol{a}_{.*} - (R-1)$, or subtracts $(R-1)$ from each element of $\boldsymbol{a}_{.*}$.

$\boldsymbol{\xi}_{*.}$: $\boldsymbol{\xi}_{*.} = \boldsymbol{a}_{*.} - (S-1)$, or subtracts $(S-1)$ from each element of $\boldsymbol{a}_{*.}$.

We also define the multivariate Beta function

$$\mathrm{B}(\boldsymbol{\alpha}) = \frac{\prod_{r=1}^{R} \Gamma(\alpha_r)}{\Gamma\left(\sum_{r=1}^{R} \alpha_r\right)}, \tag{3}$$

where $\alpha_r > 0$.

Indepedent Poisson Sampling Models

The Bayes factor for independence under the Poisson sampling models is

$$\mathrm{BF}_{01} = \left(1 + \frac{1}{b}\right)^{(R-1)(S-1)} \frac{\Gamma(Y+X)}{\Gamma(X)} \prod_{r=1}^{R} \prod_{s=1}^{S} \frac{\Gamma(a_{rs})}{\Gamma(y_{rs}+a_{rs})} \frac{\mathrm{B}(\boldsymbol{y}_{*.}+\boldsymbol{\xi}_{*.})}{\mathrm{B}(\boldsymbol{\xi}_{*.})} \frac{\mathrm{B}(\boldsymbol{y}_{.*}+\boldsymbol{\xi}_{.*})}{\mathrm{B}(\boldsymbol{\xi}_{.*})}, \tag{4}$$

where R and S are determined by the numbers of categories found in the data sample; a_{rs} and b are specified by users. Note that $a_{rs} = 1$ and $b = R \times S \times \min(a_{rs})/Y$ are the settings by default.

Joint Multinomial Sampling Models

Under this sampling scheme, the total number of observations Y is fixed. The cell counts are jointly multinomially distributed, or $(y_{11}, y_{12}, \ldots, y_{RS}) \sim \text{Multinomial}(Y, \pi_{11}, \pi_{12}, \ldots, \pi_{RS})$, where $\sum_{r=1,s=1}^{R,S} \pi_{rs} = 1$. The prior distribution is the conjugate Dirichlet distribution $(\pi_{11}, \pi_{12}, \ldots, \pi_{RS}) \sim \text{Dirichlet}(a_{11}, a_{12}, \ldots, a_{RS})$. The Bayes factor for independence under the joint Multinomial sampling models is

$$\mathrm{BF}_{01} = \frac{\mathrm{B}(\boldsymbol{y}_{*.}+\boldsymbol{\xi}_{*.})}{\mathrm{B}(\boldsymbol{\xi}_{*.})} \frac{\mathrm{B}(\boldsymbol{y}_{.*}+\boldsymbol{\xi}_{.*})}{\mathrm{B}(\boldsymbol{\xi}_{.*})} \frac{\mathrm{B}(\overrightarrow{\boldsymbol{a}})}{\mathrm{B}(\overrightarrow{\boldsymbol{y}}+\overrightarrow{\boldsymbol{a}})}, \tag{5}$$

where a_{rs} is specified by users. Note that $a_{rs} = 1$ is the setting by default.

Independent Multinomial Sampling Models

The Bayes factor for independence under the independent Multinomial sampling models when the row margins are fixed is

$$\mathrm{BF}_{01} = \frac{\mathrm{B}(\boldsymbol{y}_{.*}+\boldsymbol{\xi}_{.*})}{\mathrm{B}(\boldsymbol{\xi}_{.*})} \frac{\mathrm{B}(\boldsymbol{y}_{*.}+\boldsymbol{a}_{*.})}{\mathrm{B}(\boldsymbol{a}_{*.})} \frac{\mathrm{B}(\overrightarrow{\boldsymbol{a}})}{\mathrm{B}(\overrightarrow{\boldsymbol{y}}+\overrightarrow{\boldsymbol{a}})}, \tag{6}$$

where a_{rs} is specified by users. Note that $a_{rs} = 1$ is the setting by default. Note that when the column margins are fixed, Equation (6) changes to

$$\mathrm{BF}_{01} = \frac{\mathrm{B}(\boldsymbol{y}_{*.}+\boldsymbol{\xi}_{*.})}{\mathrm{B}(\boldsymbol{\xi}_{*.})} \frac{\mathrm{B}(\boldsymbol{y}_{.*}+\boldsymbol{a}_{.*})}{\mathrm{B}(\boldsymbol{a}_{.*})} \frac{\mathrm{B}(\overrightarrow{\boldsymbol{a}})}{\mathrm{B}(\overrightarrow{\boldsymbol{y}}+\overrightarrow{\boldsymbol{a}})}. \tag{7}$$

Similarly, if $\boldsymbol{\xi}_{*.}$ contains any components ≤ 0, we set BF_{01} to be missing, and give a warning message indicating that "Bayes factor cannot appropriately be estimated because at least one component for the prior is too small."

Bayes Factors Based on a Mixture of Symmetric Dirichlet Distributions

The Bayes factors presented in this section are based on the methods discussed by [Good, 1976]. The work evaluated the independence in contingency tables by using a mixture of symmetric Dirichlet distributions.

In the following presentation, we define

$$\begin{aligned}\Phi(m_\nu, t, t') &\equiv \frac{Y!}{\prod_\nu m_\nu!} \int_0^\infty \frac{\Gamma(tk) \prod_\nu \Gamma(m_\nu + k)}{(\Gamma(k))^t\, \Gamma(Y + tk)} \phi\left(\frac{k}{t'}\right) \frac{dk}{t'} \\ &= \frac{Y!}{\prod_\nu m_\nu!}\, \Phi'(m_\nu, t, t')\,, \end{aligned} \tag{8}$$

where

$$\phi(k) = \frac{1}{k(\pi^2 + \log^2 k)}\,, \tag{9}$$

which is the specific log-Cauchy density function as a hyper prior suggested by [Good, 1976].

Under the null hypothesis H_0, the probabilities of the interior of a contingency table given the marginals, denoted by $\mathcal{P}_{F-Y}$, is

$$\mathcal{P}_{F-Y} \equiv P(\boldsymbol{y_{**}} | y_{r.}, y_{.s}, H_0) = \frac{\prod_{r=1}^R y_{r.}! \prod_{s=1}^S y_{.s}!}{Y! \prod_{r=1}^R \prod_{s=1}^S y_{rs}!}\,. \tag{10}$$

Joint Multinomial Sampling Models

Under H_0 when the total Y is fixed, the priors are Dirichlet$(R, \mathbf{1})$ and Dirichlet$(S, \mathbf{1})$ for $\pi_{r.}$ and $\pi_{.s}$, respectively. Note that Dirichlet$(R, \mathbf{1})$ and Dirichlet$(S, \mathbf{1})$ are assumed to be independent. Under H_1, the prior is Dirichlet$(RS, \mathbf{1})$ for π_{rs}. The proposed Bayes factor is

$$\mathrm{BF}_{10} = \frac{\Phi(y_{rs}, RS, 1)}{\Phi(y_{r.}, R, 1)\, \Phi(y_{.s}, S, 1)\, \mathcal{P}_{F-Y}} = \frac{\Phi'(y_{rs}, RS, 1)}{\Phi'(y_{r.}, R, 1)\, \Phi'(y_{.s}, S, 1)}\,, \tag{11}$$

where Φ' denotes the integral part within Equation (8); Φ and $\mathcal{P}_{F-Y}$ are defined by Equations (8) and (10), respectively. We compute $\mathrm{BF}_{01} = 1/\mathrm{BF}_{10}$ to output the Bayes factor in favor of the null hypothesis.

Independent Multinomial Sampling Models

Under H_0 when the column sums are fixed, the prior is Dirichlet$(S, \mathbf{1})$ for $\pi_{.s}$. The proposed Bayes factor is

$$\mathrm{BF}_{10} = \frac{\Phi(y_{rs}, RS, 1)}{\Phi(y_{r.}, R, 1)\, \Phi(y_{.s}, S, R)\, \mathcal{P}_{F-Y}} = \frac{\Phi'(y_{rs}, RS, 1)}{\Phi'(y_{r.}, R, 1)\, \Phi'(y_{.s}, S, R)}\,, \tag{12}$$

where Φ' denotes the integral part within Equation (8); Φ and $\mathcal{P}_{F-Y}$ are defined by Equations (8) and (10), respectively. We compute $\mathrm{BF}_{01} = 1/\mathrm{BF}_{10}$ to output the Bayes factor in favor of the null hypothesis. By symmetry, if the row sums are fixed, we can switch the columns and rows in the contingency table, and apply Equation (12).

Bayes Factors Based on Intrinsic Priors

[Casella and Moreno, 2009] proposed the Bayes factors based on intrinsic priors and posterior probabilities. Due to the computation hurdles, we only implement the methods discussed in this section for 2×2 contingency tables with $R = S = 2$.

In the following presentation, we let $\boldsymbol{z} = \{z_{rs}\}$ denote the possible design of a contingency table, and let the sign $\sum_{\boldsymbol{z}:\sum z_{rs}=Y}$ denote the summation over $\boldsymbol{z}$ with all possible designs of the contingency table of $\sum z_{rs} = Y$.

Joint Multinomial Sampling Models

The Bayes factor for independence based on the intrinsic prior under the joint Multinomial sampling models is

$$\mathrm{BF}_{10} = \frac{(Y+RS-1)!}{(2Y+RS-1)!} \sum_{\boldsymbol{z}:\sum z_{rs}=Y} \binom{Y}{\boldsymbol{z}} \frac{\left(\prod_{r=1}^{R} z_{r.}!\right)\left(\prod_{s=1}^{S} z_{.s}!\right)}{\left(\prod_{r=1}^{R} y_{r.}!\right)\left(\prod_{s=1}^{S} y_{.s}!\right)} \prod_{r=1}^{R}\prod_{s=1}^{S} \frac{(z_{rs}+y_{rs})!}{z_{rs}!}\,, \tag{13}$$

where

$$\binom{Y}{\boldsymbol{z}} = \binom{Y}{z_{11}, z_{12}, z_{21}, z_{22}} = \frac{Y!}{z_{11}!\, z_{12}! \,\ldots\, z_{RS}!}\,. \tag{14}$$

To conquer the computation hurdle, we introduce an additional parameter t to control the training sample size, and rewrite Equation (13) to

$$\mathrm{BF}_{10}(t) = \frac{(t+RS-1)!}{(t+Y+RS-1)!} \frac{\Gamma(Y+R)\Gamma(Y+S)}{\Gamma(t+R)\Gamma(t+S)} \sum_{\boldsymbol{z}:\sum z_{rs}=t} \binom{t}{\boldsymbol{z}} \frac{\left(\prod_{r=1}^{R} z_{r.}!\right)\left(\prod_{s=1}^{S} z_{.s}!\right)}{\left(\prod_{r=1}^{R} y_{r.}!\right)\left(\prod_{s=1}^{S} y_{.s}!\right)} \prod_{r=1}^{R}\prod_{s=1}^{S} \frac{(z_{rs}+y_{rs})!}{z_{rs}!}\,, \tag{15}$$

where we set $t = 500$ for 2 by 2 contingency tables. If the observed grand total $Y > t$, we compute $\mathrm{BF}_{10}(t)$, and BF_{10}, otherwise.

Finally, we compute $\mathrm{BF}_{01} = 1/\mathrm{BF}_{10}$, or $\mathrm{BF}_{01}(t) = 1/\mathrm{BF}_{10}(t)$, to output the Bayes factor in favor of the null hypothesis. In case the contingency table under analysis has the dimension exceeding 2 by 2, we will give a warning message, and compute the Bayes factor by using the method presented in the "Bayes Factors Based on a Mixture of Symmetric Dirichlet Distributions" section.

Independent Multinomial Sampling Models

Under the null hypothesis, the default marginal distribution is given by

$$m_0(\boldsymbol{y_{**}}) = \frac{\Gamma(S)}{\Gamma(Y+S)} \prod_{r=1}^{R} \binom{y_{r.}}{\boldsymbol{y_{r*}}} \times \prod_{s=1}^{S} y_{.s}!\,, \tag{16}$$

where

$$\binom{y_{r.}}{\boldsymbol{y_{r*}}} = \frac{y_{r.}!}{y_{r1}!\, y_{r2}! \,\ldots\, y_{rS}!}\,. \tag{17}$$

The intrinsic marginal distribution under the independent Multinomial sampling models when the row sums are fixed is

$$m_I(\boldsymbol{y_{**}}) = \Gamma(S) \prod_{r=1}^{R} \binom{y_{r.}}{\boldsymbol{y_{r*}}} \frac{\prod_{r=1}^{R}\Gamma(y_{r.}+S)}{\Gamma(Y+S)} \sum_{\substack{(\boldsymbol{z_{1*}},\boldsymbol{z_{2*}},\ldots,\boldsymbol{z_{R*}}):\\ \sum_s z_{rs}=y_{r.}}} \frac{\prod_{s=1}^{S} z_{.s}!}{\prod_{r=1}^{R}\prod_{s=1}^{S} z_{ij}!} \prod_{r=1}^{R} \binom{y_{r.}}{\boldsymbol{z_{r*}}} \frac{\prod_{s=1}^{S}(z_{rs}+y_{rs})!}{\Gamma(2y_{r.}+S)}\,, \tag{18}$$

where

$$\binom{y_{r.}}{\boldsymbol{z_{r*}}} = \frac{y_{r.}!}{z_{r1}!\, z_{r2}! \,\ldots\, z_{rS}!}\,. \tag{19}$$

To conquer the computation hurdle, we may consider

$$m_I(\boldsymbol{y_{**}};t) = \Gamma(S) \prod_{r=1}^{R} \binom{y_{r.}}{\boldsymbol{y_{r*}}} \frac{\prod_{r=1}^{R}\Gamma(t_{r.}+S)}{\Gamma(t+S)} \sum_{\substack{(\boldsymbol{z_{1*}},\boldsymbol{z_{2*}},\ldots,\boldsymbol{z_{R*}}):\\ \sum_s z_{rs}=t_{r.}}} \frac{\prod_{s=1}^{S} z_{.s}!}{\prod_{r=1}^{R}\prod_{s=1}^{S} z_{ij}!} \prod_{r=1}^{R} \binom{t_{r.}}{\boldsymbol{z_{r*}}} \frac{\prod_{s=1}^{S}(z_{rs}+y_{rs})!}{\Gamma(t_{r.}+y_{r.}+S)}\,, \tag{20}$$

where we set $t_{r.} = 5000$, and consider four different conditions as follows for a certain 2 by 2 contingency table design:

- When $y_{1.} > t_{1.}$ and $y_{2.} > t_{2.}$, use Equation (20) by setting $t = t_{1.} + t_{2.}$;
- When $y_{1.} > t_{1.}$ and $y_{2.} < t_{2.}$, use Equation (20) by setting $t = t_{1.} + y_{2.}$ and $t_{2.} = y_{2.}$;
- When $y_{1.} < t_{1.}$ and $y_{2.} > t_{2.}$, use Equation (20) by setting $t = y_{1.} + t_{2.}$ and $t_{1.} = y_{1.}$;
- When $y_{1.} < t_{1.}$ and $y_{2.} < t_{2.}$, use Equation (18).

Thus, the desired Bayes factor is

$$\mathrm{BF}_{01} = \frac{m_0(\boldsymbol{y_{**}})}{m_I(\boldsymbol{y_{**}})} \qquad \text{or} \qquad \mathrm{BF}_{01} = \frac{m_0(\boldsymbol{y_{**}})}{m_I(\boldsymbol{y_{**}}; t)}\,, \tag{21}$$

depending on the setting of $t_{r.}$.

Note that the results are symmetrical in terms of the columns and the rows. If the column sums are fixed, we can switch the rows and columns in the contingency table, and apply Equation (21). In case the contingency table under analysis has the dimension exceeding 2 by 2, we will give a warning message, and compute the Bayes factor by using the method presented in the "Independent Multinomial Sampling Models" section with corresponding fixed row or column margins.

Bayes Factors Based on Nonparametric Bayesian Models

[Quintana, 1998] proposed the Bayes factors based on nonparametric models and Dirichlet process priors. We only implement this method when $R = 2$ (or $S = 2$) with the row sums $y_{r.}$ (or column sums $y_{.s}$) fixed. Note that the priors are the Dirichlet processes. Under this particular situation, however, the prior probabilities cancel out, which frees users from specifying prior information on weight.

Let $\boldsymbol{\lambda} = (\lambda_1, \lambda_1, \ldots, \lambda_S)$, where $\lambda_s > 0$, and we define the Dirichlet prior

$$D(\boldsymbol{\lambda}) = \frac{\Gamma\left(\sum_{s=1}^{S} \lambda_s\right)}{\prod_{s=1}^{S} \Gamma(\lambda_s)}\,. \tag{22}$$

Thus, the Bayes factor is

$$\mathrm{BF}_{01} = \frac{\mathcal{L}_1(\boldsymbol{y_{**}})}{\mathcal{L}_2(\boldsymbol{y_{**}})}\,, \tag{23}$$

where

$$\mathcal{L}_1(\boldsymbol{y_{**}}) = \frac{D(\boldsymbol{\lambda})}{D(\boldsymbol{\lambda} + \boldsymbol{y_{1*}} + \boldsymbol{y_{2*}})}\,, \tag{24}$$

and

$$\mathcal{L}_2(\boldsymbol{y_{**}}) = \frac{D(\boldsymbol{\lambda})}{D(\boldsymbol{\lambda} + \boldsymbol{y_{1*}})} \times \frac{D(\boldsymbol{\lambda})}{D(\boldsymbol{\lambda} + \boldsymbol{y_{2*}})}\,. \tag{25}$$

Note that $\boldsymbol{\lambda}$ is specified by users. We set $\boldsymbol{\lambda} = \mathbf{1}$ by default. The results are symmetrical in terms of the columns and the rows. If the column sums are fixed, we can switch the columns and rows in the contingency table, and apply Equation (23).

Bayesian Inference by Constructing Credible Intervals

In this section, we consider the model

$$\pi_{rs} = A \exp\{\alpha_j + \beta_k + \gamma_{jk}\}\,, \tag{26}$$

where $j = 1, 2, \ldots, R$, $k = 1, 2, \ldots, S$, and $A^{-1} = \sum_{j=1}^{R} \sum_{k=1}^{S} \exp\{\alpha_j + \beta_k + \gamma_{jk}\}$ with the restrictions $\alpha_R = \beta_S = \gamma_{jR} = \gamma_{Rk} = 0$. To test the independence of two factors is equivalent to make inference on γ_{jk}, where $j = 1, 2, \ldots, R-1$ and $k = 1, 2, \ldots, S-1$.

Based on the model, [Nandram and Choi, 2007] proposed to draw a random sample from Dirichlet($\mathbf{1}$), and computed the posterior distribution of γ_{jk}. They finally applied the method discussed by [Besag et al., 1995] to construct the desired simultaneous credible interval region which is a hyper-rectangular credible region for $(R-1)(S-1)$ interaction effects in a two-way contingency table. Inference can be made by checking whether or not each interval contains 0.

References

[Besag et al., 1995] Besag, J., Green, P., Higdon, D., and Mengersen, K. (1995). Bayesian computation and stochastic systems. *Statistical science*, pages 3–41.

[Casella and Moreno, 2009] Casella, G. and Moreno, E. (2009). Assessing robustness of intrinsic tests of independence in two-way contingency tables. *Journal of the American Statistical Association*, 104(487):1261–1271.

[Good, 1976] Good, I. J. (1976). On the application of symmetric dirichlet distributions and their mixtures to contingency tables. *The Annals of Statistics*, pages 1159–1189.

[Gunel and Dickey, 1974] Gunel, E. and Dickey, J. (1974). Bayes factors for independence in contingency tables. *Biometrika*, pages 545–557.

[Nandram and Choi, 2007] Nandram, B. and Choi, J. W. (2007). Alternative tests of independence in two-way categorical tables. *Journal of Data Science*, 5(2):217–237.

[Quintana, 1998] Quintana, F. A. (1998). Nonparametric bayesian analysis for assessing homogeneity in k$\times$ l contingency tables with fixed right margin totals. *Journal of the American Statistical Association*, 93(443):1140–1149.

BAYES ONESAMPLE Algorithms

One-Sample Bayesian Inference on Normal Distribution

Notations

The following notations defined in this section will be used for the subsequent sections.

x_i: Observed value of variable X for the i-th case.

y_i: Observed value of variable Y for the i-th case

w_i: Frequency weight for the i-th case. A non-integer frequency weight is rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

N: Number of cases in the data set.

W: Effective sample size $W = \sum_{i=1}^{N} w_i$. $W = N$ if no weights are present.

μ_0: Test value specified by the null hypothesis.

Basic Statistics for One-Sample *t*-Test

The Bayes factor for one-sample t-test, proposed by [Rouder et al., 2009], actually relies on the conventional t-statistic, the computation of which is discussed in this section. The following statistics are computed.

Sample mean $\bar{x} = \frac{1}{W} \sum_{i=1}^{N} w_i x_i \,.$ (1)

Sample variance $s_x^2 = \frac{1}{W-1} \sum_{i=1}^{N} w_i \left(x_i - \bar{x}\right)^2 \,.$ (2)

Sample standard deviation $s_x = \sqrt{s_x^2} \,.$ (3)

Standard error of the mean $s_{\bar{x}} = \frac{s_x}{\sqrt{W}} \,.$ (4)

Mean difference $d = \bar{x} - \mu_0 \,.$ (5)

Observed *t*-statistic $t = \frac{d}{s_{\bar{x}}}$, with $(W-1)$ degrees of freedom. (6)

Significance (2-tailed) $\text{Sig. (2-tailed)} = 2\left[1 - \text{CdfT}(|t|, W-1)\right] \,.$ (7)

Basic Statistics for Two-Sample Paired t-Test

For Bayes factor two-sample paired t-test, the following statistics are computed.

Sample mean $\bar{x} = \dfrac{1}{W} \sum_{i=1}^{N} w_i x_i \,.$ (8)

Sample mean $\bar{y} = \dfrac{1}{W} \sum_{i=1}^{N} w_i y_i \,.$ (9)

Difference of the sample means $d = \bar{x} - \bar{y}$. (10)

Sample variance $s_x^2 = \dfrac{1}{W-1} \left[\sum_{i=1}^{N} w_i x_i^2 - \left(\sum_{i=1}^{N} w_i x_i \right)^2 / W \right] .$ (11)

Sample variance $s_y^2 = \dfrac{1}{W-1} \left[\sum_{i=1}^{N} w_i y_i^2 - \left(\sum_{i=1}^{N} w_i y_i \right)^2 / W \right] .$ (12)

Covariance between $\boldsymbol{X}$ and $\boldsymbol{Y}$ $s_{xy} = \dfrac{1}{W-1} \left(\sum_{i=1}^{N} w_i x_i y_i - \left(\sum_{i=1}^{N} w_i x_i \right) \left(\sum_{i=1}^{N} w_i y_i \right) / W \right) .$ (13)

Standard deviation of the mean difference $s_D = \sqrt{s_x^2 + s_y^2 - 2 s_{xy}}$. (14)

Standard error of the mean difference $s_d = \sqrt{\left(s_x^2 + s_y^2 - 2 s_{xy} \right) / W}$. (15)

Observed $\boldsymbol{t}$-statistic for equality of means $t = \dfrac{d}{s_d}$, with $(W-1)$ degrees of freedom. (16)

Significance (2-tailed) Sig. (2-tailed) $= 2 \left[1 - \mathrm{CdfT}(|t|, W-1) \right]$. (17)

Bayes Factor for One-Sample and Two-Sample Paired t-Test with Known Variance

We can use the sufficient statistic $\bar{X}$ to formulate the Bayes factor under this setting

$$\begin{aligned} B_{01} &= \frac{\Pr(\bar{x} | \mu = \mu_0)}{\Pr(\bar{x} | \mu \neq \mu_0)} \\ &= \frac{(2\pi \sigma_x^2 / W)^{-1/2} \exp\left[-\frac{1}{2} (\bar{x} - \mu_0)^2 / (\sigma_x^2 / W) \right]}{(2\pi (\psi^2 + \sigma_x^2 / W))^{-1/2} \exp\left[-\frac{1}{2} (\bar{x} - \mu_0)^2 / (\psi^2 + \sigma_x^2 / W) \right]} \\ &= \sqrt{1 + Wg} \exp\left[-\frac{1}{2} (\bar{x} - \mu_0)^2 (\sigma_x^2)^{-1} W (1 + 1/(Wg))^{-1} \right] , \end{aligned} \tag{18}$$

where μ_0, $\sigma_x^2 > 0$ and $g > 0$ are specified a priori by users.

For two-sample paired t-test, we can replace $\bar{x}$ with $d = \bar{y} - \bar{x}$ (see Equation (10)), and σ_x^2 with σ_d^2 (specified by users), respectively, in Equation (18) to estimate the desired Bayes factor.

Bayes Factor for One-Sample and Two-Sample Paired t-Test with Unknown Variance

Suppose $X_i \overset{\text{iid}}{\sim} \mathrm{Normal}(\mu, \sigma_x^2)$, $i = 1, 2, \ldots, N$, where σ_x^2 is unknown, and we are interested in testing the null hypothesis $H_0 : \mu = 0$ versus the alternative hypothesis $H_1 : \mu \neq 0$. We assume that $\mu \sim \mathrm{Normal}(\mu_0, \psi^2)$ and $p(\sigma^2) = 1/\sigma^2$. In addition, we further specify the relationship between ψ^2 and σ_x^2 by letting $\psi^2 = g \sigma_x^2$, where $g > 0$. Thus, the Bayes factor under this setting is

$$B_{01} = \frac{\left(1 + \frac{t^2}{\nu} \right)^{-(\nu+1)/2}}{(1 + Wg)^{-1/2} \left(1 + \frac{t^2}{(1 + Wg)\nu} \right)^{-(\nu+1)/2}} , \tag{19}$$

where t is defined by Equation (7); $\nu = W - 1$; and $g > 0$ is set a priori by users.

Note that Rouder et al proposed a more general approach we would like to consider here [Rouder et al., 2009]. To construct the Bayes factor, we have to choose and place priors on both μ and σ^2. Let $\delta = \mu/\sigma$, denoting the standardized effect size. It is then equivalent to test $H_0 : \delta = 0$. One way to set the alternative hypothesis is to assume that $H_1 : \delta \sim \text{Normal}(0, \sigma_\delta^2)$, where σ_δ^2 is specified a priori [Gönen et al., 2005]. A couple of reasonable setting of σ_δ^2 may include $\sigma_\delta^2 = 1$ or $\sigma_\delta^2 \sim \text{Inverse-}\chi^2(1)$. In this document, we assume that $\delta \sim$ Cauchy, which is a t distribution with a single degree of freedom. For variance σ^2, we apply a standard setting of the Jeffreys prior with $p(\sigma^2) = 1/\sigma^2$ [Jeffreys, 1998]. Such a combination of the Cauchy on effect size δ and the Jeffreys prior on variance σ^2 is coined JZS prior in [Rouder et al., 2009].

Thus, the Bayes factor for one-sample t-test with the JZS prior is

$$B_{01} = \frac{\left(1 + \dfrac{t^2}{\nu}\right)^{-(\nu+1)/2}}{\displaystyle\int_0^\infty (1 + Wg)^{-1/2} \left(1 + \frac{t^2}{(1 + Wg)\nu}\right)^{-(\nu+1)/2} (2\pi)^{-1/2} g^{-3/2} e^{-1/(2g)}\, dg} , \tag{20}$$

where t is defined by Equation (7); $\nu = W - 1$; and g is the variable to be integrated out.

For two-sample paired t-test, both Equation (19) and (20) apply with the substitution of t computed by Equation (16).

Bayesian One-Sample Inference on Mean By Characterizing Posterior Distributions

Bayesian One-Sample Inference on Mean Using Conjugate and Noninformative Priors

Notations

The following notations defined in this section will be used for the subsequent sections.

X: A random variable to be tested whose values are observed. We assume $X \sim \text{Normal}(\mu_x, \sigma_x^2)$, where both μ_x and σ_x^2 are unknown.

μ_x: Mean parameter of X, with its prior distribution assumed in later discussions.

σ_x^2: Variance parameter of X, with its prior distribution assumed in later discussions if unknown.

w_i: Frequency weight for the i-th case. A non-integer frequency weight is rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

N: Number of cases in the data set.

W: Effective sample size $W = \sum_{i=1}^N w_i$. $W = N$ if no weights are present.

Normal Prior with Known Variance

In this section, we assume that the variance parameter σ_x^2 is known. Although this situation is not common in practice, we consider it a nice example for a teaching perspective.

We place a normal prior on μ_x by assuming that $\mu_x \sim \text{Normal}(\mu_0, \sigma_0^2)$, where μ_0 and σ_0^2 are specified by users. Under this setting, the marginal posterior distribution of μ_x is

- $\mu_x | (X, \sigma_x^2) \sim \text{Normal}(\mu_n, \sigma_n^2)$,

where $\sigma_n^2 = \left(\dfrac{1}{\sigma_0^2} + \dfrac{W}{\sigma_x^2}\right)^{-1}$, and $\mu_n = \sigma_n^2 \left(\dfrac{\mu_0}{\sigma_0^2} + \dfrac{W\bar{x}}{\sigma_x^2}\right)$. We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \mu_n , \tag{21}$$

the expected value

$$\mathbb{E}(\mu_x | X) = \mu_n , \tag{22}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \sigma_n^2 \ . \tag{23}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \Big(\text{IdfNorm}(\frac{c}{2}, \, \mu_n, \, \sqrt{\sigma_n^2} \,) \, , \, \text{IdfNorm}(1-\frac{c}{2}, \, \mu_n, \, \sqrt{\sigma_n^2} \,) \Big) \tag{24}$$

with the probability of c, where $c = 0.05$ by default.

For two-sample paired t-test, we can repeat the procedure by replacing σ_x^2 with σ_d^2, and placing the prior on the mean difference.

Diffuse Prior with Known Variance

In this section, we assume that the variance parameter σ_x^2 is known, and place a flat prior on μ_x by assuming that $p(\mu_x) \propto 1$. Under this setting, the marginal posterior distribution of μ_x is

- $\mu_x|(X, \sigma_x^2) \sim \text{Normal}(\bar{x}, \sigma_x^2/W)$.

We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \bar{x} \ , \tag{25}$$

the expected value

$$\mathbb{E}(\mu_x|X) = \bar{x} \ , \tag{26}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \sigma_x^2/W \ . \tag{27}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \Big(\text{IdfNorm}(\frac{c}{2}, \, \bar{x}, \, \sqrt{\sigma_x^2/W} \,) \, , \, \text{IdfNorm}(1-\frac{c}{2}, \, \bar{x}, \, \sqrt{\sigma_x^2/W} \,) \Big) \tag{28}$$

with the probability of c, where $c = 0.05$ by default.

For two-sample paired t-test, we can repeat the procedure by replacing σ_x^2 with σ_d^2, and placing the prior on the mean difference.

Normal-Inverse Chi-Square Priors

In this section, we assume and place the following priors

- $\sigma_x^2 \sim \text{Inverse-}\chi^2(\nu_0, \sigma_0^2)$
- $\mu_x|\sigma_x^2 \sim \text{Normal}(\mu_0, \frac{1}{\kappa_0}\sigma_x^2)$,

where σ_x^2 is conditioned on, and scaled by κ_0 ($\kappa_0 > 0$, and $\kappa_0 = 1$ by default). Note that ν_0, σ_0^2, μ_0, and κ_0 are specified by users. Under this setting, the marginal posterior distributions are

- $\sigma_x^2|X \sim \text{Inverse-}\chi^2(\nu_n, \sigma_n^2)$
- $\mu_x|X \sim t_{\nu_n}(\mu_n, \frac{1}{\kappa_n}\sigma_n^2)$,

where $\nu_n = \nu_0 + W$, $\kappa_n = \kappa_0 + W$, $\mu_n = \mu_0 \frac{\kappa_0}{\kappa_n} + \bar{x}\frac{W}{\kappa_n}$, and $\sigma_n^2 = \frac{1}{\nu_n}\left(\nu_0\sigma_0^2 + \sum_{i=1}^{N} w_i(x_i - \bar{x})^2 + W\frac{\kappa_0}{\kappa_n}(\bar{x} - \mu_0)^2\right)$.
We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \mu_n \ , \tag{29}$$

the expected value

$$\mathbb{E}(\mu_x|X) = \mu_n \ , \tag{30}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \frac{\nu_n \, \sigma_n^2}{(\nu_n - 2)\, \kappa_n} \ . \tag{31}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \left(\mu_n - \text{IdfT}\left(1 - \frac{c}{2}, \, \nu_n\right) \sqrt{\frac{\sigma_n^2}{\kappa_n}} \, , \, \mu_n + \text{IdfT}\left(1 - \frac{c}{2}, \, \nu_n\right) \sqrt{\frac{\sigma_n^2}{\kappa_n}} \right) \tag{32}$$

with the probability of c, where $c = 0.05$ by default.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

Normal-Inverse Gamma Priors

In this section, we assume and place the following priors

- $\sigma_x^2 \sim \text{Inverse-Gamma}(\alpha_0, \beta_0)$
- $\mu_x|\sigma_x^2 \sim \text{Normal}(\mu_0, \frac{1}{\kappa_0}\sigma_x^2)$,

where σ_x^2 is conditioned on, and scaled by κ_0 ($\kappa_0 > 0$, and $\kappa_0 = 1$ by default). Note that α_0, β_0, μ_0, and κ_0 are specified by users. Under this setting, we can simply set $\nu_0 = 2\alpha_0$ and $\sigma_0^2 = 2\beta_0/\nu_0$. Thus, $\sigma_x^2 \sim \text{Inverse-}\chi^2(2\alpha_0, 2\beta_0/\nu_0)$. The same approach in the "Normal-Inverse Chi-Square Priors" section can be repeated to compute the posterior distributions.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

Normal-Gamma Priors

In this section, we reparameterize σ_x^2 by letting $\tau_x = 1/\sigma_x^2$, which denotes the precision parameter. We assume and place the following priors.

- $\tau_x \sim \text{Gamma}(\alpha_0, \beta_0)$
- $\mu_x|\tau_x \sim \text{Normal}(\mu_0, \frac{1}{\kappa_0 \tau_x})$,

where τ_x is conditioned, and scaled by κ_0 ($\kappa_0 > 0$, and $\kappa_0 = 1$ by default). Note that α_0, β_0, μ_0, and κ_0 are specified by users. Under this setting, the marginal posterior distributions are

- $\tau_x|X \sim \text{Gamma}(\alpha_n, \beta_n)$
- $\mu_x|X \sim t_{2\alpha_n}(\mu_n, \frac{\beta_n}{\alpha_n \kappa_n})$,

where $\alpha_n = \alpha_0 + \frac{W}{2}$, $\beta_n = \beta_0 + \frac{1}{2}\sum_{i=1}^{N} w_i(x_i - \bar{x})^2 + \frac{\kappa_0 W(\bar{x} - \mu_0)^2}{2(\kappa_0 + W)}$, $\mu_n = \mu_0 \frac{\kappa_0}{\kappa_n} + \bar{x}\frac{W}{\kappa_n}$, and $\kappa_n = \kappa_0 + W$. We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \mu_n \ , \tag{33}$$

the expected value

$$\mathbb{E}(\mu_x|X) = \mu_n \ , \tag{34}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \frac{\beta_n}{(\alpha_n - 1)\,\kappa_n}\ . \tag{35}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \left(\mu_n - \mathrm{IdfT}\left(1-\frac{c}{2},\,\nu_n\right)\sqrt{\frac{\beta_n}{\alpha_n\kappa_n}}\,,\, \mu_n + \mathrm{IdfT}\left(1-\frac{c}{2},\,\nu_n\right)\sqrt{\frac{\beta_n}{\alpha_n\kappa_n}}\right) \tag{36}$$

with the probability of c, where $c = 0.05$ by default, and $\nu_n = 2\alpha_n$.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

Normal-Chi-Square Priors

In this section, we reparameterize σ_x^2 by letting $\tau_x = 1/\sigma_x^2$, which denotes the precision parameter. We assume and place the following priors.

- $\tau_x \sim \chi^2(\lambda)$
- $\mu_x|\tau_x \sim \mathrm{Normal}(\mu_0, \dfrac{1}{\kappa_0\tau_x})$,

where τ_x is conditioned, and scaled by κ_0 ($\kappa_0 > 0$, and $\kappa_0 = 1$ by default). Note that λ, μ_0, and κ_0 are specified by users. Under this setting, we can simply set $\alpha_0 = \lambda/2$ and $\beta_0 = 1/2$. Thus, $\tau_x \sim \mathrm{Gamma}(\lambda/2, 1/2)$. The same approach in the "Normal-Gamma Priors" section can be repeated to compute the posterior distributions.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

Jeffreys Priors

In this section, we assume and place the Jeffreys priors

- $p(\sigma_x^2) \propto \dfrac{1}{\sigma_x^4}$ [Yang and Berger, 1996] or $p(\sigma_x^2) \propto \dfrac{1}{\sigma_x^2}$ [Kass and Wasserman, 1996]
- $p(\mu_x|\sigma_x^2) \propto 1$,

where there are two optional priors on σ_x^2, and μ_x has a flat prior. Under this setting, the marginal posterior distributions are

- $\sigma_x^2|X \sim \text{Inverse-Gamma}(\alpha_n, \beta_n)$
- $\mu_x|X \sim t_{\nu_n}(\bar{x}, \sigma_n^2)$,

where

for $p(\sigma_x^2) \propto \dfrac{1}{\sigma_x^4}$, $\alpha_n = \dfrac{W+1}{2}$, $\beta_n = \dfrac{2}{\sum_{i=1}^N w_i(x_i-\bar{x})^2}$, $\nu_n = W+1$, and $\sigma_n^2 = \dfrac{1}{W(W+1)}\sum_{i=1}^N w_i(x_i-\bar{x})^2$,

and

for $p(\sigma_x^2) \propto \dfrac{1}{\sigma_x^2}$, $\alpha_n = \dfrac{W-1}{2}$, $\beta_n = \dfrac{2}{\sum_{i=1}^N w_i(x_i-\bar{x})^2}$, $\nu_n = W-1$, and $\sigma_n^2 = \dfrac{1}{W(W-1)}\sum_{i=1}^N w_i(x_i-\bar{x})^2$.

We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \bar{x}\ , \tag{37}$$

the expected value

$$\mathbb{E}(\mu_x|X) = \bar{x}\ , \tag{38}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \frac{\nu_n}{\nu_n - 2}\,\sigma_n^2 \ . \tag{39}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \Big(\,\bar{x} - \text{IdfT}(1-\frac{c}{2},\,\nu_n)\sqrt{\sigma_n^2}\,,\,\bar{x} + \text{IdfT}(1-\frac{c}{2},\,\nu_n)\sqrt{\sigma_n^2}\,\Big) \tag{40}$$

with the probability of c, where $c = 0.05$ by default.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

Diffuse Priors

In this section, we assume and place the diffuse priors

- $p(\sigma_x^2) \propto 1$
- $p(\mu_x|\sigma_x^2) \propto 1$, where both μ_x and σ_x^2 have a flat prior.

Under this setting, the marginal posterior distributions are

- $\sigma_x^2|X \sim$ Inverse-Gamma(α_n, β_n)
- $\mu_x|X \sim t_{\nu_n}(\bar{x}, \sigma_n^2)$,

where $\alpha_n = \dfrac{W-3}{2}$, $\beta_n = \dfrac{2}{\sum_{i=1}^N w_i(x_i-\bar{x})^2}$, $\nu_n = W-3$, and $\sigma_n^2 = \dfrac{1}{W(W-3)}\sum\limits_{i=1}^N w_i(x_i-\bar{x})^2$.
We may find the Bayes estimators of μ_x by computing the mode

$$\hat{\mu}_x = \bar{x} \ , \tag{41}$$

the expected value

$$\mathbb{E}(\mu_x|X) = \bar{x} \ , \tag{42}$$

and the variance of the marginal posterior distribution of $\mu_x|X$

$$\mathbb{V}(\mu_x|X) = \frac{\nu_n}{\nu_n - 2}\,\sigma_n^2 \ . \tag{43}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering μ_x such that

$$\mu_x \in \Big(\,\bar{x} - \text{IdfT}(1-\frac{c}{2},\,\nu_n)\sqrt{\sigma_n^2}\,,\,\bar{x} + \text{IdfT}(1-\frac{c}{2},\,\nu_n)\sqrt{\sigma_n^2}\,\Big) \tag{44}$$

with the probability of c, where $c = 0.05$ by default.

For two-sample paired t-test, we can repeat the procedure by placing the priors on the mean and the variance of the difference between the two paired variables.

References

[Derflinger et al., 2010] Derflinger, G., Hörmann, W., and Leydold, J. (2010). Random variate generation by numerical inversion when only the density is known. *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 20(4):18.

[Gönen et al., 2005] Gönen, M., Johnson, W. O., Lu, Y., and Westfall, P. H. (2005). The bayesian two-sample t test. *The American Statistician*, 59(3):252–257.

[Jeffreys, 1998] Jeffreys, H. (1998). *The theory of probability.* OUP Oxford.

[Kass and Wasserman, 1996] Kass, R. E. and Wasserman, L. (1996). The selection of prior distributions by formal rules. *Journal of the American Statistical Association*, 91(435):1343–1370.

[Kruschke, 2013] Kruschke, J. K. (2013). Bayesian estimation supersedes the t test. *Journal of Experimental Psychology: General*, 142(2):573.

[Lee, 2012] Lee, P. M. (2012). *Bayesian statistics: an introduction.* John Wiley & Sons.

[Rouder et al., 2009] Rouder, J. N., Speckman, P. L., Sun, D., Morey, R. D., and Iverson, G. (2009). Bayesian t tests for accepting and rejecting the null hypothesis. *Psychonomic bulletin & review*, 16(2):225–237.

[Yang and Berger, 1996] Yang, R. and Berger, J. O. (1996). *A catalog of noninformative priors.* Institute of Statistics and Decision Sciences, Duke University.

One-Sample Bayesian Inference on Binomial Distribution

Using Bayes-Factor

Notations

The following notations defined in this section will be used for the subsequent sections.

- X: $X = (X_1, X_2, \ldots, X_N)$, a realization of Bernoulli trials with $p(X_i = 1) = \pi$ and $p(X_i = 0) = 1 - \pi$. X is observed by $x = (x_1, x_2, \ldots, x_N)$, where x_i is either 0 or 1. Note that we can handle any categorical variables with two different levels, either numeric (4 and 5) or string (Yes and No), which will be recoded to 0 or 1.
- N: A total fixed number of cases (trials) in the data set.
- f: $f = (f_1, f_2, \ldots, f_N)$, a frequency or replication weight for X. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.
- N_f: $N_f = \sum_{i=1}^{N} f_i$. If there is no frequencies present, $N_f = N$.
- Y: $Y = \sum_{i=1}^{N} f_i X_i \sim \text{Binomial}(N_f, \pi)$, where Y is observed by y.
- π_0: A population proportion parameter under the null hypothesis H_0. We assume that $\pi_0 \sim \text{Beta}(a_0, b_0)$.
- π_1: A population proportion parameter under the alternative hypothesis H_1. We assume that $\pi_1 \sim \text{Beta}(a_1, b_1)$.

Bayes-Factor Based on Beta-Binomial Distribution

The Bayes factor based on the Beta-Binomial distribution is

$$\Delta_{01} = \frac{\Pr(Y|H_0)}{\Pr(Y|H_1)} = \frac{B(a_0 + y, b_0 + N_f - y)B(a_1, b_1)}{B(a_1 + y, b_1 + N_f - y)B(a_0, b_0)}\,, \tag{1}$$

where B is the beta function defined by

$$B(a, b) = \frac{\Gamma(a)\Gamma(b)}{\Gamma(a + b)}\,. \tag{2}$$

Using Conjugate and Noninformative Priors

Notations

The following notations defined in this section will be used for the subsequent sections.

- X: $X = (X_1, X_2, \ldots, X_N)$, a realization of Bernoulli trials with $p(X_i = 1) = \pi$ and $p(X_i = 0) = 1 - \pi$. X is observed by $x = (x_1, x_2, \ldots, x_N)$, where x_i is either 0 or 1. Note that we can handle any categorical variables with two different levels, either numeric (4 and 5) or string (Yes and No), which will be recoded to 0 or 1.
- N: A total fixed number of cases (trials) in the data set.
- f: $f = (f_1, f_2, \ldots, f_N)$, a frequency or replication weight for X. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.
- N_f: $N_f = \sum_{i=1}^{N} f_i$. If there is no weights present, $N_f = N$.
- Y: $Y = \sum_{i=1}^{N} f_i X_i \sim \text{Binomial}(N_f, \pi)$, where Y is observed by y.
- π: A population proportion parameter, with its prior distribution assumed in later discussions.

Beta Prior

In this section, we place a conjugate prior placed on π by assuming that $\pi \sim \text{Beta}(a, b)$, where $a, b > 0$. The sufficient sample size to estimate is $N_f \geq 1$.

Under this setting, the marginal posterior distribution of π is

- $\pi|Y \sim \text{Beta}(a + y, b + N_f - y)$.

Note that this also applies to the following two special cases:

- Uniform prior, when $a = b = 1$,
- Jeffreys prior, when $a = b = 0.5$.

We may find the Bayes estimators of π by computing the expected value

$$\mathbb{E}(\pi|Y) = \frac{a+y}{a+b+N_f} \ , \tag{3}$$

and the variance of the marginal posterior distribution of $\pi|Y$

$$\mathbb{V}(\pi|Y) = \frac{(a+y)(b+N_f-y)}{(a+b+N_f)^2(a+b+N_f+1)} \ . \tag{4}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering π such that

$$\pi \in \Big(\text{IdfBeta}(\frac{c}{2},\, a+y,\, b+N_f-y\,)\, ,\, \text{IdfBeta}(1-\frac{c}{2},\, a+y,\, b+N_f-y\,) \Big) \tag{5}$$

with the probability of c, where $c = 0.05$ by default.

To find the mode of $\pi|Y$ needs a few more discussions on the parameter support.

- If $a + y > 1$ and $b + N_f - y > 1$,
$$\hat{\pi} = \frac{a+y-1}{a+b+N_f-2} \,. \tag{6}$$
- If $a + y < 1$ and $b + N_f - y < 1$, the left mode is 0, the right mode is 1, and we define the anti-mode
$$\tilde{\pi} = \frac{a+y-1}{a+b+N_f-2} \,. \tag{7}$$
In the output design, we may indicate that this is the "anti-mode".
- If $a + y < 1$ and $b + N_f - y \geq 1$, or if $a + y = 1$ and $b + N_f - y > 1$, $\hat{\pi} = 0$.
- If $a + y \geq 1$ and $b + N_f - y < 1$, or if $a + y > 1$ and $b + N_f - y = 1$, $\hat{\pi} = 1$.
- Note that if $a + y = b + N_f - y = 1$, the posterior distribution is actually a uniform distribution with the mode equal to any value in the range $[0, 1]$.

Haldane's Prior

The density function of the Haldane's prior is $p(\pi) = \pi^{-1}(1-\pi)^{-1}$, which is an improper prior distribution. It can be treated as a special Beta distribution with $a = b = 0$. The preceding statistics derived from the posterior distribution still apply. Hence, we can allow a conjugate prior placed on π by assuming that $\pi \sim \text{Beta}(a, b)$, where $a, b > 0$, together with a special case of $a = b = 0$ to handle the Haldane's prior.

Define "Success" for Variables

To include the scale variables and the categorical variables with more than two levels, we discuss several ways to define the "success" category, recode and dichotomize the variables.

Numerical Variables

To dichotomize a numerical variable with two or more than two values, we offer the following options to define “success”:

- Using the last value found in the category after sorted in an ascending order, which is the setting by default.
- Using the first value found in the category after sorted in an ascending order.
- Using the values $\geq$ the midpoint which is the average of the minimum and maximum sample data.
- Using the values $\geq$ a specified cutoff value.
- Using the specified values (can be more than 1) in the sample data.

String Variables

To recode a string variable with more than two levels, we offer the following options to define “success”:

- Using the last level found in the category after sorted in an ascending order, which is the setting by default.
- Using the first level found in the category after sorted in an ascending order.
- Using the specified levels (can be more than 1) in the sample data.

One-Sample Bayesian Inference On Poisson Distribution

Using Bayes-Factor

Notations

The following notations defined in this section will be used for the subsequent sections.

X: $X = (X_1, X_2, \dots, X_N)$, a random sample from Poisson distribution of mean λ, or $X_i \sim \text{Poisson}(\lambda)$. $X_i = 0, 1, 2, \dots,$ which takes a nonnegative integer.

N: A total number of cases (events) in the data set.

f: $f = (f_1, f_2, \dots, f_N)$, a frequency or replication weight for X. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

N_f: $N_f = \sum_{i=1}^{N} f_i$. If there is no frequencies present, $N_f = N$.

Y: $Y = \sum_{i=1}^{N} f_i X_i \sim \text{Poisson}(N_f \lambda)$, where Y is observed by y. Note that Y is a sufficient statistic.

λ_0: A population rate parameter under the null hypothesis H_0. We assume that $\lambda_0 \sim \text{Gamma}(a_0, b_0)$.

λ_1: A population rate parameter under the alternative hypothesis H_1. We assume that $\lambda_1 \sim \text{Gamma}(a_1, b_1)$.

Bayes-Factor Based on Gamma-Poisson Distribution

Consider the probability density function for Gamma prior defined by

$$p(\lambda | a, b) = \frac{b^a}{\Gamma(a)} \lambda^{a-1} e^{-b\lambda}\,, \tag{1}$$

where $a, b > 0$. If b_0 and b_1 are rate parameters, the Bayes factor based on the Gamma-Poisson distribution is

$$\Delta_{01} = \frac{\Pr(Y|H_0)}{\Pr(Y|H_1)} = \frac{b_0^{a_0}\,(b_1 + N_f)^{a_1+y}\,\Gamma(a_0+y)\,\Gamma(a_1)}{b_1^{a_1}\,(b_0+N_f)^{a_0+y}\,\Gamma(a_1+y)\,\Gamma(a_0)}\,, \tag{2}$$

where Γ is the gamma function defined by

$$\Gamma(k) = \int_0^\infty t^{k-1} e^{-t}\,dt\,. \tag{3}$$

Using Conjugate and Reference Priors

Notations

The following notations defined in this section will be used for the subsequent sections.

X: $X = (X_1, X_2, \dots, X_N)$, a random sample from Poisson distribution of mean λ, or $X_i \sim \text{Poisson}(\lambda)$. $X_i = 0, 1, 2, \dots,$ which takes a nonnegative integer.

f: $f = (f_1, f_2, \dots, f_N)$, a frequency or replication weight for X. Non-integer frequency weights are rounded to the nearest integer. For values less than 0.5 or missing, the corresponding case will not be used.

N_f: $N_f = \sum_{i=1}^{N} f_i$. If there is no frequencies present, $N_f = N$.

Y: $Y = \sum_{i=1}^{N} f_i X_i \sim \text{Poisson}(N_f \lambda)$, where Y is observed by y. Note that Y is a sufficient statistic.

λ: A population rate or intensity parameter, with its prior distribution assumed in later discussions.

Gamma Prior

In this section, we place a conjugate prior on λ by assuming that $\lambda \sim \text{Gamma}(a_0, b_0)$, where $a_0, b_0 > 0$, and b_0 is the rate parameter. The probability density function of the prior is thus

$$p(\lambda|a_0, b_0) = \frac{b_0^{a_0}}{\Gamma(a_0)}\lambda^{a_0-1}e^{-b_0\lambda}\,. \tag{4}$$

Under this setting, the marginal posterior distribution of λ is

- $\lambda|Y \sim \text{Gamma}(a_N, b_N)$,

where $a_N = \sum_{i=1}^N f_i x_i + a_0 = y + a_0$, and $b_N = N_f + b_0$. We may find the Bayes estimators of λ by computing the expected value

$$\mathbb{E}(\lambda|Y) = a_N/b_N\ , \tag{5}$$

and the variance of the marginal posterior distribution of $\lambda|Y$

$$\mathbb{V}(\lambda|Y) = a_N/b_N^2\ . \tag{6}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering λ such that

$$\lambda \in \Big(\text{IdfGam}(\frac{c}{2},\, a_N,\, b_N\,)\,,\, \text{IdfGam}(1-\frac{c}{2},\, a_N,\, b_N\,)\Big) \tag{7}$$

with the probability of c, where $c = 0.05$ by default.

To find the mode of $\lambda|Y$ needs a few more discussions on the parameter support.

- If $a_N \geq 1$,
$$\hat{\lambda} = (a_N - 1)/b_N\,. \tag{8}$$
- If $0 < a_N < 1$, mode does not exist. The density curve forms an asymptote near $y = 0$.

Uniform Prior

In this section, we place a reference prior on λ by assuming that $\lambda \sim \text{Uniform}(0, 1)$. Actually this prior follows a special case as discussed in the "Gamma Prior" section with $a_0 = 1$ and $b_0 = 0$. Under this setting, the marginal posterior distribution of λ is

- $\lambda|Y \sim \text{Gamma}(a_N, b_N)$,

where $a_N = \sum_{i=1}^N f_i x_i + 1 = y + 1$, and $b_N = N_f$. We may find the Bayes estimators of λ by computing the expected value

$$\mathbb{E}(\lambda|Y) = a_N/b_N = (y+1)/N_f\ , \tag{9}$$

and the variance of the marginal posterior distribution of $\lambda|Y$

$$\mathbb{V}(\lambda|Y) = a_N/b_N^2 = (y+1)/N_f^2\ . \tag{10}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering λ such that

$$\lambda \in \Big(\text{IdfGam}(\frac{c}{2},\, y+1,\, N_f\,)\,,\, \text{IdfGam}(1-\frac{c}{2},\, y+1,\, N_f\,)\Big) \tag{11}$$

with the probability of c, where $c = 0.05$ by default.

Similarly, the mode of $\lambda|Y$ depends on the parameter support.

- If $a_N \geq 1$,
$$\hat{\lambda} = (a_N - 1)/b_N = y/N_f\,. \tag{12}$$
- If $0 < a_N < 1$, mode does not exist. The density curve forms an asymptote near $y = 0$.

Jeffreys Prior

In this section, we place the Jeffreys prior on λ by assuming that $p(\lambda) \propto \lambda^{-1/2}$. Actually this prior follows a special case as discussed in the "Gamma Prior" section with $a_0 = 1/2$ and $b_0 = 0$. Under this setting, the marginal posterior distribution of λ is

- $\lambda|Y \sim \text{Gamma}(a_N, b_N)$,

where $a_N = \sum_{i=1}^{N} f_i x_i + 1/2 = y + 1/2$, and $b_N = N_f$. We may find the Bayes estimators of λ by computing the expected value

$$\mathbb{E}(\lambda|Y) = a_N/b_N = (y + 1/2)/N_f \ , \tag{13}$$

and the variance of the marginal posterior distribution of $\lambda|Y$

$$\mathbb{V}(\lambda|Y) = a_N/b_N^2 = (y + 1/2)/N_f^2 \ . \tag{14}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering λ such that

$$\lambda \in \Big(\text{IdfGam}(\frac{c}{2},\, y + 1/2,\, N_f\,)\,,\, \text{IdfGam}(1 - \frac{c}{2},\, y + 1/2,\, N_f\,) \Big) \tag{15}$$

with the probability of c, where $c = 0.05$ by default.

Similarly, the mode of $\lambda|Y$ depends on the parameter support.

- If $a_N \geq 1$,
$$\hat{\lambda} = (a_N - 1)/b_N = (y - 1/2)/N_f \,. \tag{16}$$

- If $0 < a_N < 1$, mode does not exist. The density curve forms an asymptote near $y = 0$.

BAYES REGRESSION Algorithms

Bayesian Inference on Multiple Linear Regression Models

Bayesian Inference on the Linear Regression Models

Let q_i denote the regression weight for the i-th case in the n observations. If there is no regression weight specified, $q_i = 1$. If $q_i \le 0$ or missing, the corresponding case is not used. Let f_i denote the frequency weight for the i-th case in the n observations. A non-integer f_i is rounded to the nearest integer. For $f_i \le 0.5$ or missing, the corresponding case will not be used. We further define $w_i \equiv q_i f_i$, and $\boldsymbol{W} \equiv \text{diag}(w_1, w_2, \ldots, w_n) = \text{diag}(q_1 f_1, q_2 f_2, \ldots, q_n f_n)$, or

$$\boldsymbol{W} = \begin{pmatrix} q_1 f_1 & 0 & \cdots & 0 \\ 0 & q_2 f_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & q_n f_n \end{pmatrix}_{n\times n} . \tag{1}$$

Note that the effective sample is $N = \sum_{i=1}^{n} f_i$. $N = n$ if no frequency weights are present.

Using Bayes Factor

Zellner's Method

Zellner once suggested a g prior broadly discussed under $\mathcal{M}_1$ [Zellner, 1986]:

- $p(\alpha, \phi | \mathcal{M}_1) = 1/\phi$.
- $\boldsymbol{\beta} | (\phi, g, \mathcal{M}_1) \sim \textbf{Normal}\left(\boldsymbol{0}, \frac{g}{\phi}(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X})^{-1}\right)$, where g is fixed.

Since g is fixed, Zellner's g prior has the computational efficiency. Under these settings, the Bayes factor suggested by Zellner between $\mathcal{M}_1$ and $\mathcal{M}_0$ has a closed form

$$\Delta_{10}^{z} = (1+g)^{(N-p-1)/2}\left[1 + g(1-R^2)\right]^{-(N-1)/2} , \tag{2}$$

where $g > 0$, which is fixed and preset, and R^2 is the unadjusted proportion of variance accounted for by the covariate which can be similarly computed by the REGRESSION algorithm.

Jeffreys-Zellner-Siow's (JZS) Method

The Bayes factor suggested by Zellner and Siow between $\mathcal{M}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{s} = \int_0^{\infty} (1+g)^{(N-1-p)/2}\left[1 + g(1-R^2)\right]^{-(N-1)/2} \left(\frac{\sqrt{N/2}}{\Gamma(1/2)}\, g^{-3/2} e^{-N/(2g)}\right) dg\, , \tag{3}$$

where $\Gamma(1/2) = \sqrt{\pi}$, and R^2 is the unadjusted proportion of variance accounted for by the covariate which can be similarly computed by the REGRESSION algorithm.

Hyper-g Method

The Bayes factor suggested by Liang et al between $\mathcal{M}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{h}(a) = \frac{a-2}{2}\int_0^{\infty} (1+g)^{(N-1-p-a)/2}\left[1 + g(1-R^2)\right]^{-(N-1)/2} dg \tag{4}$$

$$\tag{5}$$

where a is preset, R^2 is the unadjusted proportion of variance accounted for by the covariate which can be similarly computed by the REGRESSION algorithm.

Rouder's Method

The Bayes factor suggested by Rouder and Morey between $\mathcal{M}_1$ and $\mathcal{M}_0$ is

$$\Delta_{10}^{r}(s) = \int_0^{\infty} (1+g)^{(N-p-1)/2} \left[1+g(1-R^2)\right]^{-(N-1)/2} \left(\frac{\sqrt{Ns^2/2}}{\Gamma(1/2)} g^{-3/2} e^{-Ns^2/(2g)} \right) dg \,, \tag{6}$$

where s ($s > 0$) is specified by users, $\Gamma(1/2) = \sqrt{\pi}$, and R^2 is the unadjusted proportion of variance accounted for by the predictors which can be similarly computed by the REGRESSION algorithm.

Full model based Bayes factors

Although most of the approaches are based on the comparison of $\mathcal{M}_1$ with the null model $\mathcal{M}_0$, it is still necessary and tractable to derive a full model based Bayes factor. The full mode can be expressed by

$$\mathcal{M}_F : \boldsymbol{y} = \mathbf{1}_n\alpha + \boldsymbol{X\beta} + \boldsymbol{Z\gamma} + \boldsymbol{\epsilon} \,. \tag{7}$$

The null hypothesis we desire to test is $H_0 : \boldsymbol{\gamma} = \mathbf{0}$. To construct the Bayes factor based on the user-defined full model, we may follow the similar procedures aforementioned in the previous sections. Thus, the Bayes factors between $\mathcal{M}_1$ and $\mathcal{M}_F$ by different methods are

$$\text{Zellner:} \quad \Delta_{1F}^{z}(g) = (1+g)^{-(N-P-1)/2} \left[1 + g\left(\frac{1-R_F^2}{1-R_1^2}\right)\right]^{(N-p-1)/2} \,, \tag{8}$$

$$\text{JZS:} \quad \Delta_{1F}^{s}(g) = \int_0^{\infty} (1+g)^{-(N-P-1)/2} \left[1 + g\left(\frac{1-R_F^2}{1-R_1^2}\right)\right]^{(N-p-1)/2} \left(\frac{\sqrt{N/2}}{\Gamma(1/2)} g^{-3/2} e^{-N/(2g)} \right) dg \,, \tag{9}$$

$$\text{Hyper-}g\text{:} \quad \Delta_{1F}^{h}(a) = \frac{a-2}{2} \int_0^{\infty} (1+g)^{-(N-1-P+a)/2} \left[1 + g\left(\frac{1-R_F^2}{1-R_1^2}\right)\right]^{(N-p-1)/2} dg \,, \tag{10}$$

$$\text{Rouder:} \quad \Delta_{1F}^{r}(s) = \int_0^{\infty} (1+g)^{-(N-P-1)/2} \left[1 + g\left(\frac{1-R_F^2}{1-R_1^2}\right)\right]^{(N-p-1)/2} \left(\frac{s\sqrt{N/2}}{\Gamma(1/2)} g^{-3/2} e^{-Ns^2/(2g)} \right) dg \,, \tag{11}$$

where R_1^2 and R_F^2 are the unadjusted proportion of variance accounted for by the covariate of the models $\mathcal{M}_1$ and $\mathcal{M}_F$. The integrals in the Equations (8)-(11) can be numerically approximated by feeding in the correct input $f(g)$.

Characterizing Posterior Distributions

In this section, we still consider Model $\mathcal{M}_1$ represented by Equation (**??**). In the following discussions, we define $\boldsymbol{\theta}^{\mathrm{T}} = (\alpha, \boldsymbol{\beta}^{\mathrm{T}})$, and

$$\boldsymbol{A} = [\mathbf{1}, \boldsymbol{X}] = [\mathbf{1}, \boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_p] = \begin{pmatrix} 1 & x_{11} & x_{21} & \cdots & x_{p1} \\ 1 & x_{12} & x_{22} & \cdots & x_{p2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & x_{1n} & x_{2n} & \cdots & x_{pn} \end{pmatrix}_{n\times(p+1)} \,. \tag{12}$$

Note that the columns of $\boldsymbol{A}$ must be linearly independent, and rank$(\boldsymbol{A}) = p+1$. In practice, we can release this restriction. From the following presentation, we let $(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{WA})^{-1}$ denote the generalized inverse of $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$, and do not assume that $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{WA}$ is nonsingular.

Recall from the conventional statistical analysis on multiple linear regression models, the unbiased estimates of the regression parameters are

$$\tilde{\boldsymbol{\theta}} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{WA}\right)^{-1} \boldsymbol{A}^{\mathrm{T}}\boldsymbol{Wy} \,, \tag{13}$$

and the variance of the error terms

$$\tilde{\sigma}^2 = \frac{1}{N-(p+1)} \left(\boldsymbol{y} - \boldsymbol{A}\tilde{\boldsymbol{\theta}}\right)^{\mathrm{T}} \boldsymbol{W} \left(\boldsymbol{y} - \boldsymbol{A}\tilde{\boldsymbol{\theta}}\right) \,. \tag{14}$$

Using Conjugate Priors

We place a conjugate prior by assuming that

- $\sigma^2 \sim$ Inverse-Gamma(a_0, b_0),
- $\boldsymbol{\theta}|\sigma^2 \sim$ **Normal** $\left(\boldsymbol{\theta}_0, \sigma^2 \boldsymbol{V}_0\right)$.

Note that $\boldsymbol{V}_0$ must be positive definite, and specified with a correct size dimension. Otherwise, the output will give a warning message, and use the identity matrix $\boldsymbol{I}_{p+1}$ to continue the analysis. In case that redundant columns are identified in the design matrix, we will not do any estimating, but assign 0 to the estimated coefficients. For Bayesian prediction, we will zero out the corresponding elements in $\boldsymbol{\theta}_0$ and, columns and rows in $\boldsymbol{V}_0$.

Regression parameters $\boldsymbol{\theta}$: Under this setting, the resulting marginal posterior distribution $\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}$ follows a scaled multivariate t distribution with ν degrees of freedom, where $\nu = 2a_0 + N$.

Before finding the Bayes estimator of $\boldsymbol{\theta}$, we define the following quantities:

$$\boldsymbol{\theta}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1}\left(\boldsymbol{V_0}^{-1}\boldsymbol{\theta}_0 + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right), \tag{15}$$

$$\boldsymbol{V}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1}, \tag{16}$$

$$a_1 = a_0 + \frac{N}{2}, \tag{17}$$

$$b_1 = b_0 + \frac{1}{2}\left(\boldsymbol{\theta}_0^{\mathrm{T}}\boldsymbol{V}_0^{-1}\boldsymbol{\theta}_0 + \boldsymbol{y}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} - \boldsymbol{\theta}_1^{\mathrm{T}}(\boldsymbol{V_0}^{-1} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})\,\boldsymbol{\theta}_1\right). \tag{18}$$

Hence, assuming that $\nu > 4$, we compute the mode

$$\hat{\boldsymbol{\theta}} = \boldsymbol{\theta}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1}\left(\boldsymbol{V_0}^{-1}\boldsymbol{\theta}_0 + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right), \tag{19}$$

the expected value

$$\mathbb{E}(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}) = \boldsymbol{\theta}_1 = \left(\boldsymbol{V_0}^{-1} + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1}\left(\boldsymbol{V_0}^{-1}\boldsymbol{\theta}_0 + \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right), \tag{20}$$

and the variance-covariance matrix

$$\mathbb{C}(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu - 2}\frac{b_1}{a_1}\boldsymbol{V}_1, \tag{21}$$

where $\boldsymbol{V}_1$, a_1, and b_1, are defined by Equations (16)-(18), and the diagonal elements are the variances of the elements in $\boldsymbol{\theta} = (\alpha, \beta_1, \beta_2, \ldots, \beta_p)^{\mathrm{T}}$. Define

$$\boldsymbol{B^*} \equiv \begin{pmatrix} B^*_{11} & B^*_{12} & \cdots & B^*_{1p+1} \\ B^*_{21} & B^*_{22} & \cdots & B^*_{2p+1} \\ \vdots & \vdots & \vdots & \vdots \\ B^*_{p+11} & B^*_{p+12} & \cdots & B^*_{p+1p+1} \end{pmatrix} = \frac{b_1}{a_1}\boldsymbol{V}_1. \tag{22}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering α and β_i such that

$$\alpha \in \left(\hat{\theta}_1 - \mathrm{IdfT}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{11}}, \hat{\theta}_1 + \mathrm{IdfT}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{11}}\right), \text{ and} \tag{23}$$

$$\beta_i \in \left(\hat{\theta}_{i+1} - \mathrm{IdfT}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{i+1i+1}}, \hat{\theta}_{i+1} + \mathrm{IdfT}(1 - \frac{c}{2}, \nu)\sqrt{B^*_{i+1i+1}}\right), \tag{24}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, p$, $\hat{\theta}_i$ is the i-th element in $\hat{\boldsymbol{\theta}}$, and B^*_{i+1i+1} is the $(i+1)$-th element on the diagonal of $\boldsymbol{B^*}$. Note that we only need to evaluate the diagonal elements of $\boldsymbol{B^*}$.

Error variance $\boldsymbol{\sigma^2}$: Under the setting of conjugate priors, the marginal posterior distribution of σ^2 is

$$\sigma^2|\boldsymbol{X},\boldsymbol{y} \sim \text{Inverse-Gamma}(a_1, b_1)\,, \tag{25}$$

where a_1 and b_1 are defined by Equations (17) and (18), respectively.

We may find the Bayes estimators of σ^2 by computing the mode

$$\hat{\sigma}^2 = \frac{b_1}{a_1+1}\,, \tag{26}$$

the expected value

$$\mathbb{E}(\sigma^2|\boldsymbol{X},\boldsymbol{y}) = \frac{b_1}{a_1-1}\,, \tag{27}$$

for $a_1 > 1$, and the variance of the marginal posterior distribution of $\sigma^2|\boldsymbol{X},\boldsymbol{y}$

$$\mathbb{V}(\sigma^2|\boldsymbol{X},\boldsymbol{y}) = \frac{b_1^2}{(a_1-1)^2(a_1-2)}\,, \tag{28}$$

for $a_1 > 2$. We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering σ^2 such that

$$\sigma^2 \in \left(1/\text{IdfGam}(\,1-\frac{c}{2},\,a_1,\,b_1\,)\,,\,1/\text{IdfGam}(\,\frac{c}{2},\,a_1,\,b_1\,)\right)\,, \tag{29}$$

with the probability of c, where $c = 0.05$ by default.

Predicted Value $\tilde{\boldsymbol{y}}$: Suppose we have observed a new $m \times (p+1)$ matrix of regressors $\tilde{\boldsymbol{A}}$, and we are interested in predicting the corresponding outcome $\tilde{\boldsymbol{y}}$. Given $\boldsymbol{\theta}$ and σ^2, it follows that

$$\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}},\boldsymbol{\theta},\sigma^2 \sim \textbf{Normal}\left(\tilde{\boldsymbol{A}}\boldsymbol{\theta},\sigma^2\boldsymbol{I}_m\right)\,. \tag{30}$$

The marginal posterior distribution of $\tilde{\boldsymbol{y}}$ is

$$\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}},\boldsymbol{y} \sim \boldsymbol{t}_{2a_1}\left(\tilde{\boldsymbol{A}}\boldsymbol{\theta}_1,\frac{b_1}{a_1}(\boldsymbol{I}_m+\tilde{\boldsymbol{A}}\boldsymbol{V_1}\tilde{\boldsymbol{A}}^{\mathrm{T}}\right)\,, \tag{31}$$

where a_1, b_1, and $\boldsymbol{V_1}$ are defined in Equations (17), (18), and (16), respectively. We may find the Bayes estimators of $\tilde{\boldsymbol{y}}$ by computing the mode

$$\hat{\tilde{\boldsymbol{y}}} = \tilde{\boldsymbol{A}}\boldsymbol{\theta}_1\,, \tag{32}$$

the expected value

$$\mathbb{E}(\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}},\boldsymbol{y}) = \tilde{\boldsymbol{A}}\boldsymbol{\theta}_1\,, \tag{33}$$

and the variance-covariance matrix

$$\mathbb{C}(\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}},\boldsymbol{y}) = \frac{b_1}{a_1-1}\left(\boldsymbol{I}_m+\tilde{\boldsymbol{A}}\boldsymbol{V_1}\tilde{\boldsymbol{A}}^{\mathrm{T}}\right)\,. \tag{34}$$

Define

$$\boldsymbol{D^*} \equiv \begin{pmatrix} D^*_{11} & D^*_{12} & \cdots & D^*_{1m} \\ D^*_{21} & D^*_{22} & \cdots & D^*_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ D^*_{m1} & D^*_{m2} & \cdots & D^*_{mm} \end{pmatrix} = \frac{b_1}{a_1}\left(\boldsymbol{I}_m+\tilde{\boldsymbol{A}}\boldsymbol{V_1}\tilde{\boldsymbol{A}}^{\mathrm{T}}\right)\,. \tag{35}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_m)^{\mathrm{T}}$ such that

$$\tilde{y}_i \in \left(\hat{\tilde{y}}_i - \text{IdfT}(1-\frac{c}{2},\,2a_1)\sqrt{D^*_{ii}}\,,\,\hat{\tilde{y}}_i + \text{IdfT}(1-\frac{c}{2},\,2a_1)\sqrt{D^*_{ii}}\right)\,, \tag{36}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, m$, $\hat{\tilde{y}}_i$ is the i-th element in $\hat{\tilde{\boldsymbol{y}}}$, and D^*_{ii} is the i-th element on the diagonal of $\boldsymbol{D^*}$. Note that we only need to evaluate the diagonal elements of $\boldsymbol{D^*}$.

A subset of $\boldsymbol{\theta}$: If we desire to make statistical inference on $\boldsymbol{\theta} = \boldsymbol{c}$ by including all the regression parameters, we may construct a Bayesian F-statistic by computing

$$\mathfrak{F}(\boldsymbol{\theta}) = \frac{(\boldsymbol{\theta}_1 - \boldsymbol{c})^{\mathrm{T}} \boldsymbol{V}_1^{-1} (\boldsymbol{\theta}_1 - \boldsymbol{c})}{2b_1} * \frac{2a_1}{p+1} \sim F_{p+1, 2a_1}\,, \tag{37}$$

where $\boldsymbol{\theta}_1$, $\boldsymbol{V}_1$, a_1, and b_1 are defined by Equations (15)-(18), respectively, p denotes the number of non-redundant parameters excluding the intercept term, and $\boldsymbol{c}$ is a vector of testing values specified by users with it number of elements equal to the number of parameters under estimating. By default, $\boldsymbol{c} = \boldsymbol{0}$. The associated p-value is thus $1 - \mathrm{CDF.F}(\mathfrak{F}, p+1, 2a_1)$, where CDF.F is the IBM® SPSS® Statistics function for the cumulative F distribution.

Furthermore, it is not an uncommon scenario in which we are interested in a subset of k non-redundant parameter(s) in $\boldsymbol{\theta}$, where $1 \le k \le p+1$. Note that the redundant parameter(s) specified by users, if any, will be removed before the F-statistic is estimated. Let $\boldsymbol{\theta}'$ denote such k parameter(s) to be tested. To make inference on $\boldsymbol{\theta}'$, we rewrite the null hypothesis as $H_0 : \boldsymbol{L\theta} = \boldsymbol{c}$ by constructing an appropriate $\boldsymbol{L}_{k\times(p+1)}$ matrix such that its element on the i-th row and the i'-th column is equal to 1 with the rest elements equal to 0, where i $(1 \le i \le k)$ and i' $(1 \le i' \le p+1)$ are the position index of the parameter(s) in $\boldsymbol{\theta}'$ and $\boldsymbol{\theta}$, respectively. For instance, if $\boldsymbol{\theta}' = (\beta_1, \beta_3)^{\mathrm{T}}$, the $\boldsymbol{L}$ matrix would be

$$\boldsymbol{L} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 \end{pmatrix}_{2\times(p+1)} . \tag{38}$$

The F-statistic can be formulated by

$$\mathfrak{F}(\boldsymbol{\theta}') = \frac{(\boldsymbol{L\theta}_1 - \boldsymbol{c})^{\mathrm{T}} \left[\boldsymbol{L V}_1 \boldsymbol{L}^{\mathrm{T}}\right]^{-1} (\boldsymbol{L\theta}_1 - \boldsymbol{c})}{2b_1} * \frac{2a_1}{k} \sim F_{k, 2a_1}\,. \tag{39}$$

The associated p-value is thus $1 - \mathrm{CDF.F}(\mathfrak{F}, k, 2a_1)$, where CDF.F is the IBM® SPSS® Statistics function for the cumulative F distribution. Note that Equation (37) is a special case of Equation (39) when $\boldsymbol{L} = \boldsymbol{I}_{(p+1)\times(p+1)}$.

Using Standard Reference Priors

By setting $\boldsymbol{V}_0^{-1} \to 0$, $a_0 = -(p+1)/2$, and $b_0 = 0$, it turns out that we place a reference prior by assuming that

$$p(\boldsymbol{\theta}, \sigma^2) \propto 1/\sigma^2\,. \tag{40}$$

Regression parameters $\boldsymbol{\theta}$: Under the setting of Equation (40), the resulting marginal posterior distribution $\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}$ follows a scaled multivariate t distribution with $\nu = N - (p+1)$ degrees of freedom. We can also find the Bayes estimators of $\boldsymbol{\theta}$, assuming that $\boldsymbol{A}^{\mathrm{T}}\boldsymbol{A}$ is nonsingular, by computing the mode

$$\hat{\boldsymbol{\theta}} = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1} \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\,, \tag{41}$$

the expected value

$$\mathbb{E}(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}) = \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1} \boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\,, \tag{42}$$

and the variance-covariance matrix

$$\mathbb{C}(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu-2} s^2 \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1}\,, \tag{43}$$

where

$$s^2 = \frac{1}{\nu} (\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\theta}})^{\mathrm{T}} \boldsymbol{W} (\boldsymbol{y} - \boldsymbol{A}\hat{\boldsymbol{\theta}}) = \frac{1}{\nu} \left[\boldsymbol{y}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} - \boldsymbol{y}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}\right]\,, \tag{44}$$

and the diagonal elements are the variances of the elements in $\boldsymbol{\theta} = (\alpha, \beta_1, \beta_2, \ldots, \beta_p)^{\mathrm{T}}$. Define

$$\boldsymbol{A}^{*} \equiv \begin{pmatrix} A^{*}_{11} & A^{*}_{12} & \cdots & A^{*}_{1p+1} \\ A^{*}_{21} & A^{*}_{22} & \cdots & A^{*}_{2p+1} \\ \vdots & \vdots & \vdots & \vdots \\ A^{*}_{p+11} & A^{*}_{p+12} & \cdots & A^{*}_{p+1p+1} \end{pmatrix} = s^2 \left(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A}\right)^{-1} . \tag{45}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering α and β_i such that

$$\alpha \in \left(\hat{\theta}_1 - \mathrm{IdfT}(1-\frac{c}{2},\, \nu)\sqrt{A^{*}_{11}}\,,\, \hat{\theta}_1 + \mathrm{IdfT}(1-\frac{c}{2},\, \nu)\sqrt{A^{*}_{11}}\right), \text{ and} \tag{46}$$

$$\beta_i \in \left(\hat{\theta}_{i+1} - \mathrm{IdfT}(1-\frac{c}{2},\, \nu)\sqrt{A^{*}_{i+1i+1}}\,,\, \hat{\theta}_{i+1} + \mathrm{IdfT}(1-\frac{c}{2},\, \nu)\sqrt{A^{*}_{i+1i+1}}\right) , \tag{47}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, p$, $\hat{\theta}_i$ is the i-th element in $\hat{\boldsymbol{\theta}}$, and A^{*}_{i+1i+1} is the $(i+1)$-th element on the diagonal of $\boldsymbol{A}^{*}$. Note that we only need to evaluate the diagonal elements of $\boldsymbol{A}^{*}$.

Error Variance $\boldsymbol{\sigma^2}$: Under the prior setting of (40), the marginal posterior distribution of σ^2 is

$$\sigma^2|\boldsymbol{X}, \boldsymbol{y} \sim \text{Inverse-}\chi^2(\nu, s^2) = \text{Inverse-Gamma}(\nu/2, \nu s^2/2) , \tag{48}$$

where $\nu = N - (p+1)$, and s^2 is defined by Equation (44).

We may find the Bayes estimators of σ^2 by computing the mode

$$\hat{\sigma}^2 = \frac{\nu}{\nu+2} s^2 , \tag{49}$$

the expected value

$$\mathbb{E}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{\nu}{\nu-2} s^2 , \tag{50}$$

for $\nu > 2$, and the variance of the marginal posterior distribution of $\sigma^2|X, Y$

$$\mathbb{V}(\sigma^2|\boldsymbol{X}, \boldsymbol{y}) = \frac{2\nu^2}{(\nu-2)^2(\nu-4)} s^4 , \tag{51}$$

for $\nu > 4$. We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering σ^2 such that

$$\sigma^2 \in \left(\, 1/\mathrm{IdfGam}(1-\frac{c}{2}, \frac{\nu}{2}, \frac{\nu}{2}s^2\,)\,,\, 1/\mathrm{IdfGam}(\frac{c}{2}, \frac{\nu}{2}, \frac{\nu}{2}s^2\,)\right) , \tag{52}$$

with the probability of c, where $c = 0.05$ by default.

Predicted Value $\tilde{\boldsymbol{y}}$: Suppose we have observed a new $m \times (p+1)$ matrix of regressors $\tilde{\boldsymbol{A}}$, and we are interested in predicting the corresponding outcome $\tilde{\boldsymbol{y}}$. Given $\boldsymbol{\theta}$ and σ^2, it follows that

$$\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}}, \boldsymbol{\theta}, \sigma^2 \sim \textbf{Normal}\left(\tilde{\boldsymbol{A}}\boldsymbol{\theta}, \sigma^2 \boldsymbol{I}_m\right) . \tag{53}$$

The marginal posterior distribution of $\tilde{\boldsymbol{y}}$ is

$$\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}}, \boldsymbol{y} \sim \boldsymbol{t}_{N-(p+1)}\left(\tilde{\boldsymbol{A}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}, s^2(\boldsymbol{I}_m + \tilde{\boldsymbol{A}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})^{-1}\tilde{\boldsymbol{A}}^{\mathrm{T}})\right) , \tag{54}$$

where s^2 is defined by Equations (44). We may find the Bayes estimators of $\tilde{\boldsymbol{y}}$ by computing the mode

$$\hat{\tilde{\boldsymbol{y}}} = \tilde{\boldsymbol{A}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} = \tilde{\boldsymbol{A}}\hat{\boldsymbol{\theta}} , \tag{55}$$

the expected value

$$\mathbb{E}(\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}}, \boldsymbol{y}) = \tilde{\boldsymbol{A}}(\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{A})^{-1}\boldsymbol{A}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y} = \tilde{\boldsymbol{A}}\hat{\boldsymbol{\theta}} , \tag{56}$$

and the variance-covariance matrix

$$\mathbb{C}(\tilde{\boldsymbol{y}}|\tilde{\boldsymbol{A}}, \boldsymbol{y}) = \frac{\nu}{\nu - 2} s^2 \left(\boldsymbol{I}_m + \tilde{\boldsymbol{A}} (\boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A})^{-1} \tilde{\boldsymbol{A}}^{\mathrm{T}} \right) , \tag{57}$$

where $\nu = N - (p+1)$. Define

$$\boldsymbol{F}^* \equiv \begin{pmatrix} F^*_{11} & F^*_{12} & \cdots & F^*_{1m} \\ F^*_{21} & F^*_{22} & \cdots & F^*_{2m} \\ \vdots & \vdots & \vdots & \vdots \\ F^*_{m1} & F^*_{m2} & \cdots & F^*_{mm} \end{pmatrix} = s^2 \left(\boldsymbol{I}_m + \tilde{\boldsymbol{A}} (\boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A})^{-1} \tilde{\boldsymbol{A}}^{\mathrm{T}} \right) . \tag{58}$$

We may also find a $100(1-c)\%$ Bayesian credible interval with equal tail covering $\tilde{\boldsymbol{y}} = (\tilde{y}_1, \tilde{y}_2, \ldots, \tilde{y}_m)^{\mathrm{T}}$ such that

$$\tilde{y}_i \in \left(\hat{\tilde{y}}_i - \mathrm{IdfT}(1 - \frac{c}{2}, \nu) \sqrt{F^*_{ii}} \, , \, \hat{\tilde{y}}_i + \mathrm{IdfT}(1 - \frac{c}{2}, \nu) \sqrt{F^*_{ii}} \right) , \tag{59}$$

with the probability of c, where $c = 0.05$ by default, $i = 1, 2, \ldots, m$, $\hat{\tilde{y}}_i$ is the i-th element in $\hat{\tilde{\boldsymbol{y}}}$, and F^*_{ii} is the i-th element on the diagonal of $\boldsymbol{F}^*$. Note that we only need to evaluate the diagonal elements of $\boldsymbol{F}^*$.

A subset of $\boldsymbol{\theta}$: If we desire to make statistical inference on $\boldsymbol{\theta} = \boldsymbol{c}$ by including all the regression parameters, we may construct a Bayesian F-statistic by computing

$$\mathfrak{F}(\boldsymbol{\theta}) = \frac{(\tilde{\boldsymbol{\theta}} - \boldsymbol{c})^{\mathrm{T}} \boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A} (\tilde{\boldsymbol{\theta}} - \boldsymbol{c})}{\boldsymbol{y}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{y} - \tilde{\boldsymbol{\theta}}^{\mathrm{T}} \boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A} \tilde{\boldsymbol{\theta}}} * \frac{\nu}{p+1} \sim F_{p+1,\nu} , \tag{60}$$

where $\nu = N - (p+1)$, $\tilde{\boldsymbol{\theta}}$ is defined by Equation (13), p denotes the number of non-redundant parameters excluding the intercept term, and $\boldsymbol{c}$ is a vector of testing values specified by users with it number of elements equal to the number of parameters under estimating. By default, $\boldsymbol{c} = \boldsymbol{0}$. The associated p-value is thus $1 - \mathrm{CDF.F}(\mathfrak{F}, p+1, \nu)$, where CDF.F is the IBM® SPSS® Statistics function for the cumulative F distribution.

Furthermore, it is not an uncommon scenario in which we are interested in a subset of k non-redundant parameter(s) in $\boldsymbol{\theta}$, where $1 \le k \le p+1$. Note that the redundant parameter(s) specified by users, if any, will be removed before the F-statistic is estimated. Let $\boldsymbol{\theta}'$ denote such k parameter(s) to be tested. To make inference on $\boldsymbol{\theta}'$, we rewrite the null hypothesis as $H_0 : \boldsymbol{L\theta} = \boldsymbol{c}$ by constructing an appropriate $\boldsymbol{L}_{k \times (p+1)}$ matrix such that its element on the i-th row and the i'-th column is equal to 1 with the rest elements equal to 0, where i $(1 \le i \le k)$ and i' $(1 \le i' \le p+1)$ are the position index of the parameter(s) in $\boldsymbol{\theta}'$ and $\boldsymbol{\theta}$, respectively. For instance, if $\boldsymbol{\theta}' = (\beta_1, \beta_3)^{\mathrm{T}}$, the $\boldsymbol{L}$ matrix would be

$$\boldsymbol{L} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & \cdots & 0 \end{pmatrix}_{2 \times (p+1)} . \tag{61}$$

The F-statistic can be formulated by

$$\mathfrak{F}(\boldsymbol{\theta}') = \frac{(\boldsymbol{L}\tilde{\boldsymbol{\theta}} - \boldsymbol{c})^{\mathrm{T}} \left[\boldsymbol{L} (\boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A})^{-1} \boldsymbol{L}^{\mathrm{T}} \right]^{-1} (\boldsymbol{L}\tilde{\boldsymbol{\theta}} - \boldsymbol{c})}{\boldsymbol{y}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{y} - \tilde{\boldsymbol{\theta}}^{\mathrm{T}} \boldsymbol{A}^{\mathrm{T}} \boldsymbol{W} \boldsymbol{A} \tilde{\boldsymbol{\theta}}} * \frac{\nu}{k} \sim F_{k,\nu} . \tag{62}$$

The associated p-value is thus $1 - \mathrm{CDF.F}(\mathfrak{F}, k, \nu)$, where CDF.F is the IBM® SPSS® Statistics function for the cumulative F distribution. Note that Equation (60) is a special case of Equation (62) when $\boldsymbol{L} = \boldsymbol{I}_{(p+1) \times (p+1)}$.

References

[George and Foster, 2000] George, E. and Foster, D. P. (2000). Calibration and empirical bayes variable selection. *Biometrika*, 87(4):731–747.

[Liang et al., 2012] Liang, F., Paulo, R., Molina, G., Clyde, M. A., and Berger, J. O. (2012). Mixtures of g priors for bayesian variable selection. *Journal of the American Statistical Association.*

[Rouder and Morey, 2012] Rouder, J. N. and Morey, R. D. (2012). Default bayes factors for model selection in regression. *Multivariate Behavioral Research*, 47(6):877–903.

[Sartori, 2003] Sartori, N. (2003). A note on likelihood asymptotics in normal linear regression. *Annals of the Institute of Statistical Mathematics*, 55(1):187–195.

[Zellner, 1986] Zellner, A. (1986). On assessing prior distributions and bayesian regression analysis with g-prior distributions. *Bayesian inference and decision techniques: Essays in Honor of Bruno De Finetti*, 6:233–243.

[Zellner and Siow, 1980] Zellner, A. and Siow, A. (1980). Posterior odds ratios for selected regression hypotheses. *Trabajos de estadística y de investigación operativa*, 31(1):585–603.

Bootstrapping Algorithms

Bootstrapping is a method for deriving robust estimates of standard errors and confidence intervals for estimates such as the mean, median, proportion, odds ratio, correlation coefficient or regression coefficient. It may also be used for constructing hypothesis tests. Bootstrapping is most useful as an alternative to parametric estimates when the assumptions of those methods are in doubt (as in the case of regression models with heteroscedastic residuals fit to small samples), or where parametric inference is impossible or requires very complicated formulas for the calculation of standard errors (as in the case of computing confidence intervals for the median, quartiles, and other percentiles).

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 11-1
Notation

Notation	**Description**
K	Number of distinct records in the dataset.
X_k	The *k*th distinct record, *k*=1,..,*K*.
f_k	Frequency weight of the *k*th record.
N	Number of records, $N = \Sigma_{k=1}^{K} f_k$.
B	Number of bootstrap samples.
f_{bk}^*	Generated frequency weight for the *k*th record of the *b*th bootstrap sample.
T	Statistic to bootstrap.
T_b^*	The *b*th bootstrap copy of statistic *T*.
$T_{(1)}^* \leq \cdots \leq T_{(B)}^*$	Ordered bootstrap values.

Sampling

The following sampling methods are available.

Jackknife Sampling

Jackknife sampling is used in combination with bootstrap sampling to approximate influence functions that are used in computing BCa confidence intervals. The algorithm is performed by leaving out one record at a time, and outputs the following frequency weights:

$f_1 - 1$	f_1	...	f_1
f_2	f_2 1	...	f_2
...	...	...	...
f_K	f_K	...	f_K 1

Case Resampling

In the context of bootstrapping, case resampling means to randomly sample with replacement from the original dataset. This creates bootstrap samples of equal size to the original dataset. The algorithm is performed iteratively over *k*=1,..,*K* and *b*=1,...,*B* to generate frequency weights:

$$f_{bk}^{*} = \begin{cases} rv.binom\left(N, \frac{f_k}{N}\right) & k = 1 \\ rv.binom\left(N - \Sigma_{i=1}^{k-1} f_{bi}^{*}, \dfrac{f_k}{N - \Sigma_{i=1}^{k-1} f_i}\right) & otherwise \end{cases}$$

Stratified Sampling

When subpopulations vary considerably, it is advantageous to sample each subpopulation (stratum) independently. Stratification is the process of grouping members of the population into relatively homogeneous subgroups before sampling. The strata should be mutually exclusive: every element in the population must be assigned to only one stratum. The strata should also be collectively exhaustive: no population element can be excluded. Then simple case resampling is applied within each stratum to generate frequency weights $f_{bsk_s}^{*}$.

Residual Sampling

Residual sampling supports bootstrapping of regression models. In this case, the predicted variable for each record will be adjusted with a residual that is randomly sampled in the residual set with replacement. This adjusted variable will be used as the dependent variable in the new bootstrap sample. Residual sampling assumes homoscedastic residuals.

The following notation applies to residual sampling:

Table 11-2
Notation

Notation	Description
(x_k, y_k)	Data pairs used to build regression models.
$\hat{y}_k$	Predicted values under the fitted model.
$\hat{\epsilon}_k$	Residuals, $\hat{\epsilon}_k = y_k - \hat{y}_k$.
(x_i^{*}, y_{bi}^{*})	Data pairs for the *b*th bootstrap sample.

For *i*=1,..,*N*, the algorithm sets:

$$x_i^{*} = x_{k(i)}$$

where *k*(*i*) maps *i* to *k* based upon f_k; that is, if f_1=3 and f_2=5, then *k*(1)=*k*(3)=1, *k*(4)=*k*(8)=2, and so on.

For *i*=1,..,*N* and *b*=1,...,*B*, the algorithm sets:

$$y_{bi}^{*} = \hat{y}_{k(i)} + \hat{\epsilon} \times \text{rv.multinomial}\left(f_1, ..., f_K\right)$$

where $\hat{\epsilon}$ is the 1×*k* matrix of residuals and rv.multinomial $\left(f_1, ..., f_K\right)$ produces a *k*×1 matrix representing a single draw from a multinomial distribution with relative frequencies $f_1, ..., f_K$.

Wild Bootstrap Sampling

Wild bootstrap is similar to residual sampling, but the sign of the bootstrap residual for each record is randomly reversed. Wild bootstrap is useful in the presence of heteroscedastic residuals and small sample sizes.

For *i*=1,..,*N*, the algorithm sets:

$$x_i^{*} = x_{k(i)}$$

where *k*(*i*) maps *i* to *k* based upon f_k; that is, if f_1=3 and f_2=5, then *k*(1)=*k*(3)=1, *k*(4)=*k*(8)=2, and so on.

For *i*=1,..,*N* and *b*=1,...,*B*, the algorithm sets:

$$y_{bi}^{*} = \hat{y}_{k(i)} + \left(1 - 2\text{rv.bernoulli}\left(0.5\right)\right)\left(\hat{\epsilon} \times \text{rv.multinomial}\left(f_1, ..., f_K\right)\right)$$

where $\hat{\epsilon}$ is the 1×*k* matrix of residuals and rv.multinomial $\left(f_1, ..., f_K\right)$ produces a *k*×1 matrix representing a single draw from a multinomial distribution with relative frequencies $f_1, ..., f_K$.

Pooling

The following pooling methods are available: bootstrap estimates and percentile-t pivotal tests.

Bootstrap Estimates

Bias

The bias of statistic *T* can be estimated by the following equation

$$Bias\left(T\right) = B^{-1}\sum_{b=1}^{B} T_b^{*} - T$$

Standard error

The standard error of statistic *T* can be estimated by the standard deviation of the bootstrap values with the following equation

$$SE \approx \sqrt{\frac{1}{B-1}\sum_{b=1}^{B}\left(T_b^* - B^{-1}\sum_{b=1}^{B}T_b^*\right)^2}$$

Percentile confidence interval

Suppose that *T* estimates a scalar θ that we want an interval with left- and right-tail errors both equal to α, and that bootstrap values are ordered as $T_{(1)}^* \leq ... \leq T_{(B)}^*$. The basic percentile confidence interval is

$$\hat{\theta}_\alpha = T^*_{((B+1)\alpha)},\ \hat{\theta}_{1-\alpha} = T^*_{((B+1)(1-\alpha))}$$

If $(B+1)\,\alpha$ is not an integer, then interpolation can be used. A simple method that works well for approximately normal estimators is linear interpolation on the normal quantile scale. For example, suppose the integer part of $(B+1)\,\alpha$ is *k*, then we define

$$T^*_{((B+1)\alpha)} = T^*_{(k)} + \frac{\Phi^{-1}(\alpha) - \Phi^{-1}\left(\frac{k}{B+1}\right)}{\Phi^{-1}\left(\frac{k+1}{B+1}\right) - \Phi^{-1}\left(\frac{k}{B+1}\right)}\left(T^*_{(k+1)} - T^*_{(k)}\right)$$

where $\Phi^{-1}(\cdot)$ is the inverse normal(0,1) distribution. Similarly, if $(B+1)(1-\alpha)$ is not an integer, the same interpolation can be used by replacing α with $1-\alpha$ in the equation above. Clearly such interpolations fail if *k*=0, *B* or *B*+1. If this happens, we quote the extreme value and the implied level of error equal to $1/(B+1)$.

BCa confidence interval

The influence value of the k_sth record in the *s*th stratum is approximated by

$$l_{jack,sk_s} = (N_s - 1)(T - T_{-sk_s})$$

where T_{-sk_s} is the estimate calculated from the original data but with the frequency $f_{sk_s} - 1$ for the k_sth record in the *s*th stratum. It is reasonable to assume the empirical influence values $l_{sk_s} \doteq l_{jack,sk_s}$.

Defining $\tilde{l}_{sk_s} = l_{sk_s}N/N_s$, the BCa confidence interval is given as

$$\hat{\theta}_\alpha = T^*_{((B+1)\tilde{\alpha})},\ \hat{\theta}_{1-\alpha} = T^*_{((B+1)(1-\tilde{\alpha}))}$$

where

$$\tilde{\alpha} = \Phi\left(w + \frac{w+z_\alpha}{1-a(w+z_\alpha)}\right),$$
$$z_\alpha = \Phi^{-1}(\alpha),$$
$$w = \Phi^{-1}\left(\frac{|\{T_b^* \leq t\}|}{B+1}\right),$$
$$a = \frac{1}{6}\frac{\Sigma_{s,k_s} f_{sk_s}\tilde{l}^3_{sk_s}}{\left(\Sigma_{s,k_s} f_{sk_s}\tilde{l}^2_{sk_s}\right)^{3/2}}.$$

Interpolation will be used as in the Percentile confidence interval.

Percentile-t Pivotal Tests

Suppose the null hypothesis is $H_0 : T =$

T_0. Scalar T

Let $z_0 = (T - T_0)/SE$ and $z_b^* = (T_b^* - T)/SE_b^*$ where *SE* and SE_b^* are the standard errors of *T* and T_b^*, respectively. We estimate the standard error from the standard errors calculated within the procedure.

The alternative hypothesis can be $H_A : T > T_0$, $H_A : T < T_0$, or $H_A : T \neq T_0$, which correspond to right-sided, left-sided, and two-sided *p*-values, respectively. The bootstrap right-sided *p*-value is calculated as

$$p = \frac{\left|\left\{z_b^* \geq z_0\right\}\right| + 1}{B + 1}$$

The bootstrap left-sided *p*-value is calculated as

$$p = \frac{\left|\left\{z_b^* \leq z_0\right\}\right| + 1}{B + 1}$$

The bootstrap two-sided p-value is calculated as .

$$p = \frac{\left|\left\{z_b^{*2} \geq z_0^2\right\}\right| + 1}{B + 1}$$

Vector T

Let $z_0 = (T - T_0)^T Cov(T)^{-1} (T - T_0)$ and $z_b^* = (T_b^* - T)^T Cov(T_b^*)^{-1} (T_b^* - T)$, where $Cov(T)$ and $Cov(T_b^*)$ are the covariance matrices of *T* and T_b^*, respectively. We estimate the covariance matrix from the covariance matrix calculated within the procedure.

The alternative hypothesis is $H_A : T \neq T_0$, and the bootstrap *p*-value can be calculated as

$$p = \frac{\left|\left\{z_b^* \geq z_0\right\}\right| + 1}{B + 1}$$

The percentile-*t* pivotal tests can also support bootstrap testing for the null hypothesis of $H_0 : LT T_0$ where *L* is a matrix of linear combinations. In this case, let $z_0 = (LT - T_0)^T \left\{LCov(T) L^T\right\}^{-1} (LT - T_0)$ and $z_b^* = (LT_b^* - LT)^T \left\{LCov(T_b^*) L^T\right\}^{-1} (LT_b^* - LT)$. The alternative hypothesis is $H_A : LT \neq T_0$, and nd the bootstrap p-value can be calculated as

$$p = \frac{\left|\left\{z_b^* \geq z_0\right\}\right| + 1}{B + 1}$$

References

Davison, A. C., and D. V. Hinkley. 2006. *Bootstrap Methods and their Application*. : Cambridge University Press.

Shao, J., and D. Tu. 1995. *The Jackknife and Bootstrap*. New York: Springer.

CATPCA Algorithms

The CATPCA procedure quantifies categorical variables using optimal scaling, resulting in optimal principal components for the transformed variables. The variables can be given mixed optimal scaling levels and no distributional assumptions about the variables are made.

In CATPCA, dimensions correspond to components (that is, an analysis with two dimensions results in two components), and object scores correspond to component scores.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 12-1
Notation

Notation	Description
n	Number of analysis cases (objects)
n_w	Weighted number of analysis cases: $\sum_{i=1}^{n} w_i$
n_{tot}	Total number of cases (analysis + supplementary)
w_i	Weight of object *i*; $w_i = 1$ if cases are unweighted; $w_i = 0$ if object *i* is supplementary.
W	Diagonal $n_{tot} \times n_{tot}$ matrix, with w_i on the diagonal.
m	Number of analysis variables
m_w	Weighted number of analysis variables ($m_w = \sum_{j=1}^{m} v_j$)
m_{tot}	Total number of variables (analysis + supplementary)
m_1	Number of analysis variables with multiple nominal scaling level.
m_2	Number of analysis variables with non-multiple scaling level.
m_{w1}	Weighted number of analysis variables with multiple nominal scaling level.
m_{w2}	Weighted number of analysis variables with non-multiple scaling level.
J	Index set recording which variables have multiple nominal scaling level.
H	The data matrix (category indicators), of order $n_{tot} \times m_{tot}$, after discretization, imputation of missings , and listwise deletion, if applicable.
p	Number of dimensions

For variable *j*; $j = 1, \ldots, m_{tot}$

Table 12-2
Notation

Notation	Description
v_j	Variable weight; $v_j = 1$ if weight for variable *j* is not specified or if variable *j* is supplementary
k_j	Number of categories of variable *j* (number of distinct values in $\mathbf{h}_j$, thus, including supplementary objects)
$\mathbf{G}_j$	Indicator matrix for variable *j*, of order $n_{tot} \times k_j$

The elements of $\mathbf{G}_j$ are defined as $i = 1, \ldots, n_{tot}; r = 1, \ldots, k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i\text{th object is in the } r\text{th category of variable } j \\ 0 & \text{when the } i\text{th object is not in the } r\text{th category of variable } j \end{cases}$$

Table 12-3
Notation

Notation	Description
$\mathbf{D}_j$	Diagonal $k_j \times k_j$ matrix, containing the weighted univariate marginals; i.e., the weighted column sums of $\mathbf{G}_j$ ($\mathbf{D}_j = \mathbf{G}'_j\mathbf{W}\mathbf{G}_j$)
$\mathbf{M}_j$	Diagonal $n_{tot} \times n_{tot}$ matrix, with diagonal elements defined as

$$m_{(j)ii} = \begin{cases} 0 & \text{when the } i\text{th observation is missing and missing strategy variable } j \text{ is passive} \\ 0 & \text{when the } i\text{th object is in } r\text{th category of variable } j \text{ and } r\text{th category is only used by supplementary objects (i.e. when } d_{(j)rr} = 0) \\ v_j & \text{otherwise} \end{cases}$$

Table 12-4
Notation

Notation	Description
$\mathbf{M}_*$	$\Sigma_j \mathbf{M}_j$
$\mathbf{S}_j$	I-spline basis for variable *j*, of order $k_j \times (s_j + t_j)$(see Ramsay (1988) for details)
$\mathbf{b}_j$	Spline coefficient vector, of order $s_j + t_j$
d_j	Spline intercept.
s_j	Degree of polynomial
t_j	Number of interior knots

The quantification matrices and parameter vectors are:

Table 12-5
Notation

Notation	Description
$\mathbf{X}$	Object scores, of order $n_{tot} \times p$
$\mathbf{X}_w$	Weighted object scores ($\mathbf{X}_w = \mathbf{W}\mathbf{X}$)
$\mathbf{X}^{\text{n}}$	$\mathbf{X}$ normalized according to requested normalization option
$\mathbf{Y}_j$	Centroid coordinates, of order $k_j \times p$. For variables with optimal scaling level multiple nominal, this are the category quantifications
$\mathbf{y}_j$	Category quantifications for variables with non-multiple scaling level, of order k_j
$\mathbf{a}_j$	Component loadings for variables with non-multiple scaling level, of order *p*
$\mathbf{a}\text{n}_j$	$\mathbf{a}_j$ normalized according to requested normalization option
$\underline{\mathbf{Y}}$	Collection of category quantifications (centroid coordinates) for variables with multiple nominal scaling level $\mathbf{Y}_j$), and vector coordinates for non-multiple scaling level ($\mathbf{y}_j\mathbf{a}'_j$).

Note: The matrices $\mathbf{W}$, $\mathbf{G}_j$, $\mathbf{M}_j$, $\mathbf{M}_*$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Discretization

Discretization is done on the unweighted data.

Multiplying

First, the original variable is standardized. Then the standardized values are multiplied by 10 and rounded, and a value is added such that the lowest value is 1.

Ranking

The original variable is ranked in ascending order, according to the alphanumerical value.

Grouping into a specified number of categories with a normal distribution

First, the original variable is standardized. Then cases are assigned to categories using intervals as defined in Max (1960).

Grouping into a specified number of categories with a uniform distribution

First the target frequency is computed as divided by the number of specified categories, rounded. Then the original categories are assigned to grouped categories such that the frequencies of the grouped categories are as close to the target frequency as possible.

Grouping equal intervals of specified size

First the intervals are defined as lowest value + interval size, lowest value + 2*interval size, etc. Then cases with values in the *k*th interval are assigned to category *k*.

Imputation of Missing Values

When there are variables with missing values specified to be treated as active (impute mode or extra category), then first the k_j's for these variables are computed before listwise deletion. Next the category indicator with the highest weighted frequency (mode; the smallest if multiple modes exist), or $k_j + 1$ (extra category) is imputed. Then listwise deletion is applied if applicable. And then the k_j's are adjusted.

If an extra category is imputed for a variable with optimal scaling level Spline Nominal, Spline Ordinal, Ordinal or Numerical, the extra category is not included in the restriction according to the scaling level in the final phase.

For more information, see the topic “Objective Function Optimization”.

Configuration

CATPCA can read a configuration from a file, to be used as the initial configuration or as a fixed configuration in which to fit variables.

For an initial configuration see step 1 in "Objective Function Optimization "

A fixed configuration **X** is centered and orthonormalized as described in the optimization section in step 3 (with **X** instead of **Z**) and step 4 (except for the factor $n_w^{1/2}$), and the result is postmultiplied with $\mathbf{\Lambda}^{1/2}$ (this leaves the configuration unchanged if it is already centered and orthogonal). The analysis variables are set to supplementary and variable weights are set to one. Then CATPCA proceeds as described in "Supplementary Variables".

Objective Function

The CATPCA objective is to find object scores **X** and a set of $\underline{\mathbf{Y}}_j$ (for *j*=1,...,*m*) — the underlining indicates that they may be restricted in various ways — so that the function

$$\sigma(\mathbf{X};\underline{\mathbf{Y}}) = n_w^{-1}\sum_j c^{-1}\text{tr}\Big(\big(\mathbf{X}-\mathbf{G}_j\underline{\mathbf{Y}}_j\big)'\mathbf{M}_j\mathbf{W}\big(\mathbf{X}-\mathbf{G}_j\underline{\mathbf{Y}}_j\big)\Big)$$

where *c* is p if $j \in J$ and *c* is 1 if $j \notin J$.

$\notin$

is minimal, under the normalization restriction $\mathbf{X}'\mathbf{M}_*\mathbf{W}\mathbf{X} = n_w m_w \mathbf{I}$ (**I** is the *p*×*p* identity matrix). The inclusion of $\mathbf{M}_j$ in $\sigma(\mathbf{X};\underline{\mathbf{Y}})$ ensures that there is no influence of passive missing values (missing values in variables that have missing option passive, or missing option not specified). $\mathbf{M}_*$ contains the number of active data values for each object. The object scores are also centered; that is, they satisfy $\mathbf{u}'\mathbf{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ with **u** denoting an *n*-vector with ones.

Optimal Scaling Levels

The following optimal scaling levels are distinguished in CATPCA:

Multiple Nominal. $\underline{\mathbf{Y}}_j = \mathbf{Y}_j$ (equality restriction only).

Nominal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ (equality and rank – one restrictions).

Spline Nominal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j = d_j + \mathbf{S}_j\mathbf{b}_j$ (equality, rank – one, and spline restrictions).

Spline Ordinal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j = d_j + \mathbf{S}_j\mathbf{b}_j$ (equality, rank – one, and monotonic spline restrictions), with $\mathbf{b}_j$ restricted to contain nonnegative elements (to guarantee monotonic I-splines).

Ordinal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{C}_j$ (equality, rank – one, and monotonicity restrictions). The monotonicity restriction $\mathbf{y}_j \in \mathbf{C}_j$ means that $\mathbf{y}_j$ must be located in the convex cone of all k_j-vectors with nondecreasing elements.

Numerical. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{L}_j$ (equality, rank – one, and linearity restrictions). The linearity restriction $\mathbf{y}_j \in \mathbf{L}_j$ means that $\mathbf{y}_j$ must be located in the subspace of all k_j-vectors that are a linear transformation of the vector consisting of k_j successive integers.

For each variable, these levels can be chosen independently. The general requirement for all options is that equal category indicators receive equal quantifications. The general requirement for the non-multiple options is $\underline{\mathbf{Y}}_j = \mathbf{y}_j \mathbf{a}'_j$; that is, $\underline{\mathbf{Y}}_j$ is of rank one; for identification purposes, $\mathbf{y}_j$ is always normalized so that $\mathbf{y}'_j \mathbf{D}_j \mathbf{y}_j = n_w$.

Objective Function Optimization

Optimization is achieved by executing the following iteration scheme:

1. Initialization I or II
2. Update category quantifications
3. Update object scores
4. Orthonormalization
5. Convergence test: repeat (2) through (4) or continue
6. Rotation and reflection

The first time (for the initial configuration) initialization I is used and variables that do not have optimal scaling level Multiple Nominal or Numerical are temporarily treated as numerical, the second time (for the final configuration) initialization II is used. Steps (1) through (6) are explained below.

Initialization

I. If an initial configuration is not specified, the object scores **X** are initialized with random numbers. Then **X** is orthonormalized (see step 4) so that $\mathbf{u}'\mathrm{M}_*\mathbf{WX} = \mathbf{0}$ and $\mathbf{X}'\mathrm{M}_*\mathbf{WX} = n_w m_w \mathbf{I}$, yielding $\mathbf{X}_w^+$. The initial component loadings are computed as the cross products of $\mathbf{X}_w^+$ and the centered original variables $\left(\mathbf{I} - \mathrm{M}_j \mathbf{uu}'\mathbf{W}/\left(\mathbf{u}'\mathrm{M}_j\mathbf{Wu}\right)\right)\mathbf{h}_j$, rescaled to unit length.

II. All relevant quantities are copied from the results of the first cycle.

Update category quantifications; loop across analysis variables

With fixed current values $\mathbf{X}_w^+$ the unconstrained update of $\mathbf{Y}_j$ is

$$\tilde{\mathbf{Y}}_j = \mathbf{D}_j^{-1}\mathbf{G}'_j\mathbf{X}_w^+$$

Multiple nominal: $\mathbf{Y}_j^+ = \tilde{\mathbf{Y}}_j$.

For non-multiple scaling levels first an unconstrained update is computed in the same way:

$$\tilde{\mathbf{Y}}_j = \mathbf{D}_j^{-1}\mathbf{G}'_j\mathbf{X}_w^+$$

next one cycle of an ALS algorithm (De Leeuw et al., 1976) is executed for computing a rank-one decomposition of $\tilde{\mathbf{Y}}_j$, with restrictions on the left-hand vector, resulting in

$\tilde{\mathbf{y}}_j = \tilde{\mathbf{Y}}_j \mathbf{a}_j$

Nominal: $\mathbf{y}_j^* = \tilde{\mathbf{y}}_j$.

For the next four optimal scaling levels, if variable *j* was imputed with an extra category, $\mathbf{y}_j^*$ is inclusive category k_j in the initial phase, and is exclusive category k_j in the final phase.

Spline nominal and spline ordinal: $\mathbf{y}_j^* = d_j + \mathbf{S}_j \mathbf{b}_j$.

The spline transformation is computed as a weighted regression (with weights the diagonal elements of $\mathbf{D}_j$) of $\tilde{\mathbf{y}}_j$ on the I-spline basis $\mathbf{S}_j$. For the spline ordinal scaling level the elements of $\mathbf{b}_j$ are restricted to be nonnegative, which makes $\mathbf{y}_j^*$ monotonically increasing

Ordinal: $\mathbf{y}_j^* \leftarrow \tilde{\mathbf{y}}_j)$.

The notation WMON( ) is used to denote the weighted monotonic regression process, which makes $\mathbf{y}_j^*$ monotonically increasing. The weights used are the diagonal elements of $\mathbf{D}_j$ and the subalgorithm used is the up-and-down-blocks minimum violators algorithm (Kruskal, 1964; Barlow et al., 1972).

Numerical: $\mathbf{y}_j^* \leftarrow \tilde{\mathbf{y}}_j)$.

The notation WLIN( ) is used to denote the weighted linear regression process. The weights used are the diagonal elements of $\mathbf{D}_j$.

Next $\mathbf{y}_j^*$ is normalized (if variable *j* was imputed with an extra category, $\mathbf{y}_j^*$ is inclusive category k_j from here on):

$$\mathbf{y}_j^+ = n_w^{1/2} \mathbf{y}_j^* \left(\mathbf{y'}_j^* \mathbf{D}_j \mathbf{y}_j^* \right)^{-1/2}$$

Then we update the component loadings:

$$\mathbf{a}_j^+ = n_w^{-1} \tilde{\mathbf{Y}}' \mathbf{D}_j \mathbf{y}_j^+$$

Finally, we set $\underline{\mathbf{Y}}_j^+ = \mathbf{y}_j^+ \mathbf{a'}_j^+$.

Update object scores

First the auxiliary score matrix **Z** is computed as

$$\mathbf{Z} \leftarrow \Sigma_j \mathbf{M}_j \mathbf{G}_j \underline{\mathbf{Y}}_j^+$$

and centered with respect to **W** and $\mathbf{M}_*$:

$$\mathbf{X}^* = \left(\mathbf{I} - \mathbf{M}_* \mathbf{u}\mathbf{u}' \mathbf{W} / \left(\mathbf{u}' \mathbf{M}_* \mathbf{W}\mathbf{u} \right) \right) \mathbf{Z}$$

These two steps yield locally the best updates when there would be no orthogonality constraints.

Orthonormalization

To find an $\mathbf{M}_*$-orthonormal $\mathbf{X}^+$ that is closest to $\mathbf{X}^*$ in the least squares sense, we use for the Procrustes rotation (Cliff, 1966) the singular value decomposition $m_w^{1/2}\mathbf{M}_*^{-1/2}\mathbf{W}^{1/2}\mathbf{X}^* = \mathbf{K}\mathbf{\Lambda}^{1/2}\mathbf{L}'$, then yields $n_w^{1/2}m_w^{1/2}\mathbf{M}_*^{-1/2}\mathbf{W}^{1/2}\mathbf{K}\mathbf{L}'$-orthonormal weighted object scores: $\mathbf{X}_w^+ \leftarrow n_w^{1/2}m_w\mathbf{M}_*^{-1}\mathbf{W}\mathbf{X}^*\mathbf{L}\mathbf{\Lambda}^{-1/2}\mathbf{L}'$, and $\mathbf{X}^+ = \mathbf{W}^{-1}\mathbf{X}_w^+$. The calculation of $\mathbf{L}$ and $\mathbf{\Lambda}$ is based on tridiagonalization with Householder transformations followed by the implicit QL algorithm (Wilkinson, 1965).

Convergence test

The difference between consecutive values of the quantity

$$\text{TFIT} = (pn_w)^{-1}\sum_{j\in J} v_j \operatorname{tr}\left(\mathbf{Y}'_j\mathbf{D}_j\mathbf{Y}_j\right) + \sum_{j\notin J} v_j\mathbf{a}'_j\mathbf{a}_j$$

is compared with the user-specified convergence criterion ε - a small positive number. It can be shown that TFIT = $m_{w1} + pm_{w2} - \sigma(\mathbf{X};\underline{\mathbf{Y}})$. Steps (2) through (4) are repeated as long as the loss difference exceeds ε.

After convergence TFIT is also equal to tr $\left(\mathbf{\Lambda}^{1/2}\right)$, with $\mathbf{\Lambda}$ as computed in the Orthonormalization step during the last iteration. (See also "Model Summary " and variable correlations "Correlations and Eigenvalues " for interpretation of $\mathbf{\Lambda}^{1/2}$).

Rotation and reflection

To achieve principal axes orientation, $\mathbf{X}^+$ is rotated with the matrix $\mathbf{L}$. In addition the *s*th column of $\mathbf{X}^+$ is reflected if for dimension *s* the mean of squared loadings with a negative sign is higher than the mean of squared loadings with a positive sign. Then step (2) is executed, yielding the rotated and possibly reflected quantifications and loadings.

Supplementary Objects

To compute the object scores for supplementary objects, after convergence the category quantifications and object scores are again updated (following the steps in "Objective Function Optimization "), with the zero's in **W** temporarily set to ones in computing **Z** and $\mathbf{X}^+$. If a supplementary object has missing values, passive treatment is applied.

Supplementary Variables

The quantifications for supplementary variables are computed after convergence. For supplementary variables with multiple nominal scaling level, the Update Category Quantification step is executed once. For non-multiple supplementary variables, an initial a_j is computed as in the Initialization step. Then the rank-one and restriction substeps of the Update Category Quantification step are repeated as long as the difference between consecutive values of $\mathbf{a}'_j\mathbf{a}_j$ exceeds .00001, with a maximum of 100 iterations. For more information, see the topic "Objective Function Optimization " on p. 85.

Diagnostics

The procedure produces the following diagnostics.

Maximum Rank (may be issued as a warning when exceeded)

The maximum rank $p_{\max}$ indicates the maximum number of dimensions that can be computed for any dataset. In general

$$p_{\max} = \min\left(n-1, \left(\sum_{j\in J} k_j\right) - m_1 - m_2\right)$$

if there are variables with optimal scaling level multiple nominal without missing values to be treated as passive. If variables with optimal scaling level multiple nominal do have missing values to be treated as passive, the maximum rank is

$$p_{\max} = \min\left(n-1, \left(\sum_{j\in J} k_j\right) - \max(m_3, 1) - m_2\right)$$

with m_3 the number of variables with optimal scaling level multiple nominal without missing values to be treated as passive.

Here k_j is exclusive supplementary objects (that is, a category only used by supplementary objects is not counted in computing the maximum rank). Although the number of nontrivial dimensions may be less than $p_{\max}$ when m=2, CATPCA does allow dimensionalities all the way up to $p_{\max}$. When, due to empty categories in the actual data, the rank deteriorates below the specified dimensionality, the program stops.

Descriptives

The descriptives tables gives the weighted univariate marginals and the weighted number of missing values (system missing, user defined missing, and values less than or equal to 0) for each variable.

Fit and Loss Measures

When the HISTORY option is in effect, the following fit and loss measures are reported:

Total fit (VAF). This is the quantity TFIT as defined in the Convergence Test step.

Total loss. This is $\sigma(\mathbf{X}; \underline{\mathbf{Y}})$, computed as the sum of multiple loss and single loss defined below.

Multiple loss. This measure is computed as

$$\text{TMLOSS} = (m_{w1} + pm_{w2}) - \left((n_w p)^{-1} \sum_{j\in J} v_j \operatorname{tr}\left(\mathbf{Y}'_j \mathbf{D}_j \mathbf{Y}_j\right) + n_w^{-1} \sum_{j\notin J} v_j \operatorname{tr}\left(\mathbf{Y}'_j \mathbf{D}_j \mathbf{Y}_j\right)\right)$$

Single loss. This measure is computed only when some of the variables are

single: $\text{SLOSS} = n_w^{-1} \sum_{j\notin J} v_j \operatorname{tr}\left(\mathbf{Y}'_j \mathbf{D}_j \mathbf{Y}_j\right) - \sum_{j\notin J} v_j \mathbf{a}'_j \mathbf{a}_j$

Model Summary

Model summary information consists of Cronbach's alpha and the variance accounted for.

Cronbach's Alpha

Cronbach's Alpha per dimension (*s*=1,...,*p*):

$$\alpha_s = m_w \left(\lambda_s^{1/2} - 1\right) / \left(\lambda_s^{1/2} \left(m_w - 1\right)\right)$$

Total Cronbach's Alpha is

$$\alpha = m_w \left(\Sigma_s \lambda_s^{1/2} - 1\right) / \Sigma_s \lambda_s^{1/2} \left(m_w - 1\right)$$

with λ_s the *s*th diagonal element of $\mathbf{\Lambda}$ as computed in the Orthonormalization step during the last iteration.

Variance Accounted For

Variance Accounted For per dimension (*s*=1,...,*p*):

Multiple Nominal variables:

$\mathrm{VAF1}_s = n_w^{-1} \sum_{j \in J} v_j \mathrm{tr}\left(\mathbf{y}'_{(j)s} \mathbf{D}_j \mathbf{y}_{(j)s}\right)$, (% of variance is $\mathrm{VAF1}_s \times 100/m_{w1}$),

Non-Multiple variables:

$\mathrm{VAF2}_s = \sum_{j \notin J} v_j a_{js}^2$, (% of variance is $\mathrm{VAF2}_s \times 100/m_{w2}$).

Eigenvalue per dimension:

$\lambda_s^{1/2} = \mathrm{VAF1}_s + \mathrm{VAF2}_s$,

with λ_s the *s*th diagonal element of $\mathbf{\Lambda}$ as computed in the Orthonormalization step during the last iteration. (See also the Convergence Test step and variable correlations "Correlations and Eigenvalues " for interpretation of $\mathbf{\Lambda}^{1/2}$).

The Total Variance Accounted For for multiple nominal variables is the mean over dimensions, and for non-multiple variables the sum over dimensions. So, the total eigenvalue is

$\mathrm{tr}\left(\mathbf{\Lambda}^{1/2}\right) = p^{-1} \Sigma_s \mathrm{VAF1}_s + \Sigma s \mathrm{VAF2}_s$.

If there are no passive missing values, the eigenvalues $\mathbf{\Lambda}^{1/2}$ are those of the correlation matrix (see "Correlations and Eigenvalues ") weighted with variable weights:

$r_{jj}^{\mathbf{W}} = v_j r_{jj}$, and $r_{jl}^{\mathbf{W}} = r_{lj}^{\mathbf{W}} = v_j^{1/2} r_{jl}$

If there are passive missing values, then the eigenvalues are those of the matrix $m_w \mathbf{Q}'_{\mathrm{C}} \mathbf{M}_*^{-1} \mathbf{Q}_{\mathrm{C}}$, with $\mathbf{Q}_{\mathrm{C}} = n_w^{-1/2} \left(\mathbf{I} - \mathbf{M}_* \mathbf{u}\mathbf{u}' \mathbf{W} / \left(\mathbf{u}' \mathbf{M}_* \mathbf{W} \mathbf{u} \right) \right) \mathbf{Q}$, (see "Correlations and Eigenvalues ") which is not necessarily a correlation matrix, although it is positive semi-definite. This matrix is weighted with variable weights in the same way as $\mathbf{R}$.

Variance Accounted For

The Variance Accounted For table gives the VAF per dimension and per variable for centroid coordinates, and for non-multiple variables also for vector coordinates (see "Quantifications").

Centroid Coordinates

$$\mathrm{VAF}_{js} = v_j \mathrm{tr} \left(\mathbf{Y}'_{js} \mathbf{D}_j \mathbf{Y}_{js} \right)$$

Vector Coordinates

$\mathrm{VAF}_{js} = v_j a_{js}^2$, for $j \notin J$

Correlations and Eigenvalues

Before Transformation

$\mathbf{R} = n_w^{-1} \mathbf{H}'_{\mathrm{C}} \mathbf{W} \mathbf{H}_{\mathrm{C}}$, with $\mathbf{H}_{\mathrm{C}}$ weighted centered and normalized $\mathbf{H}$. For the eigenvalue decomposition of **R** (to compute the eigenvalues), first row *j* and column *j* are removed from **R** if *j* is a supplementary variable, and then r_{ij} is multiplied by $(v_i v_j)^{1/2}$.

If passive missing treatment is applicable for a variable, missing values are imputed with the variable mode, regardless of the passive imputation specification.

After Transformation

When all analysis variables are non-multiple, and there are no missing values, specified to be treated as passive, the correlation matrix is:

$\mathbf{R} = n_w^{-1} \mathbf{Q}' \mathbf{W} \mathbf{Q}$, with $\mathbf{q}_j = \mathbf{G}_j \mathbf{y}_j$.

The first *p* eigenvalues of **R** equal $\mathbf{\Lambda}^{1/2}$. (See also the Convergence Test step and "Model Summary" for interpretation of $\mathbf{\Lambda}^{1/2}$). When there are multiple nominal variables in the analysis, *p* correlation matrices are computed (*s*=1,...,*p*):

$\mathbf{R}_{(s)} = n_w^{-1} \mathbf{Q}'_{(s)} \mathbf{W} \mathbf{Q}_{(s)}$,

with $\mathbf{q}_{(s)j} = \mathbf{G}_j \mathbf{y}_j$ for non-multiple variables and $\mathbf{q}_{(s)j} = n_w^{1/2} \mathbf{G}_j \mathbf{Y}_{(j)s} \left(\mathbf{Y}'_{(j)s} \mathbf{D}_j \mathbf{Y}_{(j)s} \right)^{-1/2}$ for multiple nominal variables.

Usually, for the higher eigenvalues, the first eigenvalue of $\mathbf{R}_{(s)}$ is equal to $\lambda_s^{1/2}$(see “Model Summary ”). The lower values of $\mathbf{\Lambda}^{1/2}$ are in most cases the second or subsequent eigenvalues of $\mathbf{R}_{(s)}$.

If there are missing values, specified to be treated as passive, the mode of the quantified variable or the quantification of an extra category (as specified in syntax; if not specified, default (mode) is used) is imputed before computing correlations. Then the eigenvalues of the correlation matrix do not equal $\mathbf{\Lambda}^{1/2}$ (see Model Summary section). The quantification of an extra category for multiple nominal variables is computed as

$$\mathbf{Y}_{(j)_{(k_j+1)s}} = \left(\sum_{i\in I} w_i\right)^{-1} \sum_{i\in I} w_i x_{is},$$

with *I* an index set recording which objects have missing values.

For the quantification of an extra category for non-multiple variables first $\mathbf{Y}_{(j)_{(k_j+1)s}}$ is computed as above, and then

$$\mathbf{y}_{(k_j+1)j} = n_w^{1/2} \left(\sum_s a_{js}^2\right)^{-1} \sum_s a_{js} \mathbf{Y}_{(j)_{k_j+1)s}}.$$

For the eigenvalue decomposition of **R** (to compute the eigenvalues), first row *j* and column *j* are removed from **R** if *j* is a supplementary variable, and then r_{ij} is multiplied by $(v_i v_j)^{1/2}$.

Object Scores and Loadings

If all variables have non-multiple scaling level, normalization partitions the first *p* singular values of $n_w^{-1/2}\mathbf{W}^{1/2}\mathbf{Q}\mathbf{V}^{1/2}$ divided by m_w over the objects scores **X** and the loadings **A**, with **Q** the matrix of quantified variables (see “Correlations and Eigenvalues”), and **V** a diagonal matrix with elements v_j. The singular value decomposition of $n_w^{-1/2}\mathbf{W}^{1/2}\mathbf{Q}\mathbf{V}^{1/2}$ is

$$\text{SVD}\left(n_w^{-1/2}\mathbf{W}^{1/2}\mathbf{Q}\mathbf{V}^{1/2}\right) = \mathbf{K}\mathbf{\Phi}^{1/2}\mathbf{L}'.$$

With $\mathbf{X} = \mathbf{K}_p$ (the subscript *p* denoting the first *p* columns of **K**) and $\mathbf{A} = \left(\mathbf{L}\mathbf{\Phi}^{1/2}\right)_p$, $\mathbf{X}\mathbf{A}'$ gives the best *p*-dimensional approximation of $n_w^{-1/2}\mathbf{W}^{1/2}\mathbf{Q}\mathbf{V}^{1/2}$.

The first *p* singular values $\mathbf{\Phi}_p^{1/2}$ equal $\mathbf{\Lambda}^{1/4}$, with $\mathbf{\Lambda}$ as computed in the Orthonormalization step during the last iteration. (See also the Convergence Test step and “Model Summary ” for interpretation of $\mathbf{\Lambda}^{1/2}$).

For partitioning the first *p* singular values we write

$$\left(\mathbf{K}\mathbf{\Phi}^{1/2}\mathbf{L}'\right)_p = \mathbf{K}_p\mathbf{\Phi}_p^{a/2}\mathbf{\Phi}_p^{b/2}\mathbf{L}'_p = \mathbf{K}_p\mathbf{\Lambda}^{a/4}\mathbf{\Lambda}^{b/4}\mathbf{L}'_p, \text{ (}a+b\text{=1, see below).}$$

During the optimization phase, variable principal normalization is used. Then, after convergence $\mathbf{X} = n_w^{1/2}\mathbf{W}^{-1/2}\mathbf{K}_p$ and $\mathbf{A} = \mathbf{V}^{-1/2}\mathbf{L}_p\mathbf{\Lambda}^{▮/4}$.

If variable principal normalization is requested, $\mathbf{X}^{\mathrm{n}} = \mathbf{X}$ and $\mathbf{A}^{\mathrm{n}} = \mathbf{A}$, else $\mathbf{X}^{\mathrm{n}} = \mathbf{X}\mathbf{\Lambda}^{a/4}$ and $\mathbf{A}^{\mathrm{n}} = \mathbf{A}\mathbf{\Lambda}^{1/4(b-1)}$ with *a*=(1+*q*)/2, *b*=(1−*q*)/2, and *q* any real value in the closed interval [-1,1], except for independent normalization: then there is no *q* value and *a*=*b*=1. *q*=−1 is equal to variable principal normalization, *q*=1 is equal to object principal normalization, *q*=0 is equal to symmetrical normalization.

When there are multiple nominal variables in the analysis, there are *p* matrices $\mathbf{Q}_{(s)}$, *s*=1,...*p*, (see “Correlations and Eigenvalues ”). Then one of the singular values of $n_w^{-1/2}\mathbf{W}^{1/2}\mathbf{Q}_{(s)}\mathbf{V}^{1/2}$ equals $\mathbf{\Lambda}_s^{1/4}$.

If a variable has multiple nominal scaling level, the normalization factor is reflected in the centroids: $\mathbf{Y}_j^{\mathrm{n}} = \mathbf{Y}_j\mathbf{\Lambda}^{1/4(b-1)}$.

Quantifications

For variables with non-multiple scaling level the quantifications $\mathbf{y}_j$ are displayed, the vector coordinates $\mathbf{y}_j\left(\mathbf{a}_j^{\mathrm{n}}\right)'$, and the centroid coordinates: $\mathbf{Y}_j$ with variable principal normalization, $\mathbf{D}_j^{-1}\mathbf{G}'_j\mathbf{W}\mathbf{X}^{\mathrm{n}}$ with one of the other normalization options. For multiple nominal variables the quantifications are the centroid coordinates $\mathbf{Y}_j^{\mathrm{n}}$.

If a category is only used by supplementary objects (i.e. treated as a passive missing), only centroid coordinates are displayed for this category, computed as $\mathbf{y}_{(j)r} = n_w^{1/2}n_{jr}^{-1}\sum_{i\in I}\mathbf{x}_i^{\mathrm{n}}$ for variables with non-multiple scaling level and $\mathbf{y}_{(j)r} = n_w^{1/2}n_{jr}^{-1}\sum_{i\in I}\mathbf{x}_i\mathbf{\Lambda}^{1/4(b-1)}$ for variables with multiple nominal scaling level, where $\mathbf{y}_{(j)r}$ is the *r*th row of $\mathbf{Y}_j$, n_{jr} is the number of objects that have category *r*, and *I* is an index set recording which objects are in category *r*.

Residuals

For non-multiple variables, Residuals gives a plot of the quantified variable *j*($\mathbf{G}_j\mathbf{y}_j$)the approximation, $\mathbf{X}\mathbf{a}_j$. For multiple nominal variables plots per dimension are produced of $\mathbf{G}_j\mathbf{y}_{(j)s}^{\mathrm{n}}$ against the approximation $\mathbf{x}_s^{\mathrm{n}}$.

Projected Centroids

The projected centroids of variable *l* on variable *j*, $j \notin J$, are

$$\mathbf{Y}_l\mathbf{a}_j\left(\mathbf{a}'_j\mathbf{a}_j\right)^{-1/2}$$

Scaling factor Biplot, triplot, and loading plot

In plots including both the object scores or centroids and loadings (loading plot including centroids, biplot with objects and loadings, and triplot with objects, centroids and loadings), the object scores and centroids are rescaled using the following scaling factor:

$$\text{Scalefactor} = \frac{2\sum_{s=1}^{p} \max\left(a_{1s}^{\mathbf{n}}, ..., a_{ms}^{\mathbf{n}}\right)}{\sum_{s=1}^{p} \left|\min\left(x_{1s}^{\mathbf{n}}, ..., x_{ns}^{\mathbf{n}}\right)\right| + \left(\max\left(x_{1s}^{\mathbf{n}}, ..., x_{ns}^{\mathbf{n}}\right)\right.}$$

References

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions*. New York: John Wiley and Sons.

Cliff, N. 1966. Orthogonal rotation to congruence. *Psychometrika*, 31, 33–42.

De Leeuw, J., F. W. Young, and Y. Takane. 1976. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41, 471–503.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Max, J. 1960. Quantizing for minimum distortion. *Proceedings IEEE (Information Theory)*, 6, 7–12.

Pratt, J. W. 1987. Dividing the indivisible: Using simple symmetry to partition variance explained. In: *Proceedings of the Second International Conference in Statistics*, T. Pukkila, and S. Puntanen, eds. Tampere, Finland: Universityof Tampere, 245–260.

Ramsay, J. O. 1989. Monotone regression splines in action. *Statistical Science*, 4, 425–441.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem*. Oxford: Clarendon Press.

CATREG Algorithms

CATREG (Categorical regression with optimal scaling using alternating least squares) quantifies categorical variables using optimal scaling, resulting in an optimal linear regression equation for the transformed variables. The variables can be given mixed optimal scaling levels and no distributional assumptions about the variables are made.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of analysis cases (objects)
n_w	Weighted number of analysis cases: $\sum_{i=1}^{n} w_i$
n_{tot}	Total number of cases (analysis + supplementary)
w_i	Weight of object *i*; $w_i = 1$ if cases are unweighted; $w_i = 0$ if object *i* is supplementary.
W	Diagonal $n_{tot} \times n_{tot}$ matrix, with w_i on the diagonal.
p	Number of predictor variables
m	Total number of analysis variables
r	Index of response variable
J_p	Index set of predictor variables
H	The data matrix (category indicators), of order $n_{tot} \times m$, after discretization, imputation of missings , and listwise deletion, if applicable.
p	Number of dimensions
λ_1	Lasso penalty
λ_2	Ridge penalty

For variable *j*; $j = 1, \ldots, m$

k_j	Number of categories of variable *j* (number of distinct values in $\mathbf{h}_j$, thus, including supplementary objects)
$\mathbf{G}_j$	Indicator matrix for variable *j*, of order $n_{tot} \times k_j$

The elements of $\mathbf{G}_j$ are defined as $i = 1, \ldots, n_{tot}$; $r = 1, \ldots, k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i\text{th object is in the } r\text{th category of variable } j \\ 0 & \text{when the } i\text{th object is not in the } r\text{th category of variable } j \end{cases}$$

$\mathbf{D}_j$	Diagonal $k_j \times k_j$ matrix, containing the weighted univariate marginals; i.e., the weighted column sums of $\mathbf{G}_j$ ($\mathbf{D}_j = \mathbf{G}'_j\mathbf{W}\mathbf{G}_j$)
f	Vector of degrees of freedom for the predictor variables, of order *p*
$\mathbf{S}_j$	I-spline basis for variable *j*, of order $k_j \times (s_j + t_j)$ (see Ramsay (1988) for details)
$\mathbf{b}_j$	Spline coefficient vector, of order $s_j + t_j$
d_j	Spline intercept.

s_j	Degree of polynomial
t_j	Number of interior knots

The quantification matrices and parameter vectors are:

$\mathbf{y}_r$	Category quantifications for the response variable, of order k_r
$\mathbf{y}_j$	Category quantifications for predictor variable *j*, of order k_j
$\mathbf{b}$	Regression coefficients for the predictor variables, of order *p*
$\mathbf{v}$	Accumulated contributions of predictor variables: $\sum_{j\in J_p} b_j \mathbf{G}_j \mathbf{y}_j$

Note: The matrices $\mathbf{W}$, $\mathbf{G}_j$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Discretization

Discretization is done on the unweighted data.

Multiplying

First, the original variable is standardized. Then the standardized values are multiplied by 10 and rounded, and a value is added such that the lowest value is 1.

Ranking

The original variable is ranked in ascending order, according to the alphanumerical value.

Grouping into a specified number of categories with a normal distribution

First, the original variable is standardized. Then cases are assigned to categories using intervals as defined in Max (1960).

Grouping into a specified number of categories with a uniform distribution

First the target frequency is computed as divided by the number of specified categories, rounded. Then the original categories are assigned to grouped categories such that the frequencies of the grouped categories are as close to the target frequency as possible.

Grouping equal intervals of specified size

First the intervals are defined as lowest value + interval size, lowest value + 2*interval size, etc. Then cases with values in the *k*th interval are assigned to category *k*.

Imputation of Missing Values

When there are variables with missing values specified to be treated as active (impute mode or extra category), then first the k_j's for these variables are computed before listwise deletion. Next the category indicator with the highest weighted frequency (mode; the smallest if multiple modes exist), or $k_j + 1$ (extra category) is imputed. Then listwise deletion is applied if applicable. And then the k_j's are adjusted.

If an extra category is imputed for a variable with optimal scaling level Spline Nominal, Spline Ordinal, Ordinal or Numerical, the extra category is not included in the restriction according to the scaling level in the final phase.

For more information, see the topic “Objective Function Optimization”.

Objective Function

The CATREG objective is to find the set of $\mathbf{y}_r$, **b**, and $\mathbf{y}_j$, $j \in J_p$, so that the function

$$\sigma(\mathbf{y}_r; \mathbf{b}; \mathbf{y}_j) = \left(\mathbf{G}_r\mathbf{y}_r - \int_{j \in J_p} b_j \mathbf{G}_j \mathbf{y}_j\right)' \mathbf{W} \left(\mathbf{G}_r\mathbf{y}_r - \int_{j \in J_p} b_j \mathbf{G}_j \mathbf{y}_j\right)$$

is minimal, under the normalization restriction $\mathbf{y}'_r \mathbf{D}_r \mathbf{y}_r = n_w$. The quantifications of the response variable are also centered; that is, they satisfy $\mathbf{u}'\mathbf{W}\mathbf{G}_r\mathbf{y}_r = \mathbf{0}$ with **u** denoting an *n*-vector with ones.

With regularization, the loss function is subjected to:

$\int_{j \in J_p}^{P} \beta_j^2 \leq t_2$ for Ridge,

$\int_{j \in J_p}^{P} |\beta_j| \leq t_1$ for Lasso,

$\int_{j \in J_p}^{P} |\beta_j| \leq t_1$ and $\int_{j \in J_p}^{P} \beta_j^2 \leq t_2$ for Elastic Net.

The constrained loss functions can also be written as penalized loss functions:

$$L^{\text{ridge}} = L + \lambda_2 \int_{j \in J_p}^{P} \beta_j^2$$

$$L^{\text{lasso}} = L + \lambda_1 \int_{j \in J_p}^{P} \text{sign}(\beta_j)\beta_j$$

$$L^{\text{e-net}} = L + \lambda_1 \int_{j \in J_p}^{P} \text{sign}(\beta_j)\beta_j + \lambda_2 \int_{j \in J_p}^{P} \beta_j^2$$

Optimal Scaling Levels

The following optimal scaling levels are distinguished in CATREG:

Nominal. Equality restrictions only.

Spline Nominal. $\mathbf{y}_j = d_j + \mathbf{S}_j\mathbf{a}_j$ (equality and spline restrictions).

Spline Ordinal. $\mathbf{y}_j = d_j + \mathbf{S}_j\mathbf{a}_j$ (equality and monotonic spline restrictions), with $\mathbf{a}_j$ restricted to contain nonnegative elements (to guarantee monotonic I-splines).

Ordinal. $\mathbf{y}_j \in \mathbf{C}_j$ (equality and monotonicity restrictions). The monotonicity restriction $\mathbf{y}_j \in \mathbf{C}_j$ means that $\mathbf{y}_j$ must be located in the convex cone of all k_j-vectors with nondecreasing elements.

Numerical. $\mathbf{y}_j \in \mathbf{L}_j$ (equality and linearity restrictions). The linearity restriction $\mathbf{y}_j \in \mathbf{L}_j$ means that $\mathbf{y}_j$ must be located in the subspace of all k_j-vectors that are a linear transformation of the vector consisting of k_j successive integers.

For each variable, these levels can be chosen independently. The general requirement for all options is that equal category indicators receive equal quantifications. For identification purposes, $\mathbf{y}_j$ is always normalized so that $\mathbf{y}'_j\mathbf{D}_j\mathbf{y}_j = n_w$.

Objective Function Optimization

Optimization is achieved by executing the following iteration scheme:

1. Initialization I or II
2. Update category quantifications response variable
3. Update category quantifications and regression coefficients predictor variables
4. Convergence test: repeat (2) through (3) or continue

Steps (1) through (4) are explained below.

Initialization

I. Random

The initial category quantifications $\tilde{\mathbf{y}}_j$ (for *j*= 1, ..., *m*) are defined as the k_j category indicators of variable *j*, normalized such that $\mathbf{u}'\mathbf{W}\mathbf{G}_j\tilde{\mathbf{y}}_j = 0$ and $\tilde{\mathbf{y}}_j\mathbf{D}_j\tilde{\mathbf{y}}_j = n_w$, and the initial regression coefficients are the correlations with the response variable.

II. Numerical

In this case, the iteration scheme is executed twice. In the first cycle, (initialized with initialization I) all variables are treated as numerical. The second cycle, with the specified scaling levels, starts with the category quantifications and regression coefficients from the first cycle.

III. Multistart (ALL)

Choosing all multiple systematic starts guarantees obtaining the global optimal solution when the spline ordinal or ordinal scaling level is specified for one or more predictors (Van der Kooij, Meulman, and Heiser, 2006). When this option is chosen, the iteration scheme is executed 2^s times, where *s* is the number of predictor variables with (spline) ordinal scaling level and 2^s is the number of *all possible sign patterns* for the regression coefficients of the predictor variables with (spline) ordinal scaling level. Each execution of the iteration scheme starts with the same initial category quantifications and regression coefficients (initialized with initialization I), but with different sign patterns for the coefficients. In the iteration process, the signs are held fixed. Finally, the iteration scheme is executed one more time using the optimal sign pattern (the pattern resulting in the highest R^2, or RSQ^{regu} if regularization is applied).

IV. Multistart (value)

When a threshold value is specified with the multiple systematic starts option, the iteration scheme is executed twice for a selection of sign patterns for the regression coefficients of the predictor variables with (spline) ordinal scaling level. The sign patterns are selected by a combination of a percentage of loss of variance strategy and a hierarchical strategy (Van der Kooij, Meulman, and Heiser, 2006).

The maximum number of sign patterns with this option is $1 + \sum_{i=1}^{s} i$.

In the first cycle (initialized with initialization I) all variables are treated as nominal. The second cycle, with the specified scaling levels, starts with the category quantifications and regression coefficients from the first cycle. After one iteration in the second cycle, the decrease in variance going from the last iteration in the first cycle to the first iteration in the second cycle is determined for predictors with (spline) ordinal scaling level. If the percentage of decrease for a predictor is above the specified threshold value, the predictor is allowed to have a negative sign. Then the second cycle continues a number of times: one time with the regression coefficient for all (spline) ordinal predictor positive and *q* times with the regression coefficient for one (spline) ordinal predictor negative, where *q* is the number of predictors with (spline) ordinal scaling level that are allowed to have a negative sign. If the ‘all positive’ sign pattern gives a better result (higher R^2, or RSQ^{regu} if regularization is applied) then the ‘one negative’ signs patterns, the iteration scheme is executed one more time using the ‘all positive’ sign pattern. Else, if one of the ‘one negative’ signs patterns gives a better result than the ‘all positive’ sign pattern, the best ‘one negative’ signs pattern is selected and the second cycle is repeated for the ‘two negatives’ signs patterns: the patterns formed by adding one more negative sign to the best ‘one negative’ signs pattern. Then, the results of the ‘two negatives’ signs patterns are compared to the ‘one negative’ signs pattern and the ‘one negative’ signs pattern is selected if its result is better. Else, the second cycle is repeated for the ‘three negatives’ signs patterns, and so on.

V. Fixsigns

In this case, the iteration scheme is executed twice. In the first cycle, (initialized with initialization I) all variables are treated as nominal. The second cycle, with the specified scaling levels, starts with the category quantifications and regression coefficients from the first cycle and fixed signs (read from a user-specified file) for the regression coefficients of the predictor variables with (spline) ordinal scaling level.

Update category quantifications response variable

With fixed current values $\mathbf{y}_j$, $j \in J_p$ the unconstrained update of $\mathbf{y}_r$ is

$$\tilde{\mathbf{y}}_r = \mathbf{D}_r^{-1}\mathbf{G}'_r\mathbf{W}\mathbf{v}$$

Nominal: $\mathbf{y}_r^* = \tilde{\mathbf{y}}_r$

For the next four optimal scaling levels, if variable *j* was imputed with an extra category, $\mathbf{y}_r^*$ is inclusive category k_r in the initial phase, and is exclusive category k_r in the final phase.

Spline nominal and spline ordinal: $\mathbf{y}_r^* = d_r + \mathbf{S}_r\mathbf{a}_r$.

The spline transformation is computed as a weighted regression (with weights the diagonal elements of $\mathbf{D}_r$) of $\tilde{\mathbf{y}}_r$ on the I-spline basis $\mathbf{S}_r$. For the spline ordinal scaling level the elements of $\mathbf{a}_r$ are restricted to be nonnegative, which makes $\mathbf{y}_r^*$ monotonically increasing

Ordinal: $\mathbf{y}_r^* \leftarrow \tilde{\mathbf{y}}_r$) .

The notation WMON( ) is used to denote the weighted monotonic regression process, which makes $\mathbf{y}_r^*$ monotonically increasing. The weights used are the diagonal elements of $\mathbf{D}_r$ and the subalgorithm used is the up-and-down-blocks minimum violators algorithm (Kruskal, 1964; Barlow et al., 1972).

Numerical: $\mathbf{y}_r^* \leftarrow \tilde{\mathbf{y}}_r$).

The notation WLIN( ) is used to denote the weighted linear regression process. The weights used are the diagonal elements of $\mathbf{D}_r$.

Next $\mathbf{y}_r^*$ is normalized (if the response variable was imputed with an extra category, $\mathbf{y}_r^*$ is inclusive category k_r from here on):

$$\mathbf{y}_r^+ = n_w^{1/2}\mathbf{y}_r^*\left(\mathbf{y}'^*_r\mathbf{D}_r\mathbf{y}_r^*\right)^{-1/2}$$

Update category quantifications and regression weights predictor variables

For updating a predictor variable *j*, $j \in J_p$, first the contribution of variable *j* is removed from $\mathbf{v}$:$\mathbf{v}_j = \mathbf{v} - b_j\mathbf{G}_j\mathbf{y}_j$ Then the unconstrained update of $\mathbf{y}_j$ is

$$\tilde{\mathbf{y}}_j = \mathbf{D}_j^{-1}\mathbf{G}'_j\mathbf{W}(\mathbf{G}_r\mathbf{y}_r - \mathbf{v}_j)$$

Next $\tilde{\mathbf{y}}_j$ is restricted and normalized as in step (2) to obtain $\mathbf{y}_j^+$.

Finally, we update the regression coefficient

$$b_j^+ = n_{\mathbf{W}}^{-1}\tilde{\mathbf{y}}'_j\mathbf{D}_j\mathbf{y}_j^+$$

Regularized regression coefficients are obtained as

$\beta_j^+ = \frac{\beta_j^*}{1+\lambda_2}$ for Ridge,

$\beta_j^+ = \left(\beta_j^* - \frac{\lambda_1}{2} w_j\right)_+ = \beta_j^* - \frac{\lambda_1}{2} \text{if} \beta_j^* > 0 \text{and} \beta_j^* + \frac{\lambda_1}{2} \text{if} \beta_j^* < 0$ for Lasso, and

$\beta_j^+ = \frac{\left(\beta_j^* - \frac{\lambda_1}{2} w_j\right)_+}{1+\lambda_2} = \frac{\left(\beta_j^* - \frac{\lambda_1}{2}\right)_+}{1+\lambda_2} \text{if} \beta_j^* > 0 \text{and} \frac{\left(\beta_j^* + \frac{\lambda_1}{2}\right)_+}{1+\lambda_2} \beta_j^* \quad < 0$ for Elastic Net (van der Kooij, 2007).

Convergence test

The difference between consecutive values of the apparent Prediction Error is compared with the user-specified convergence criterion ε a small positive number.

The difference between consecutive values of the quantity

$$\text{APE} = n_w^{-1} \left(\mathbf{G}_r \mathbf{y}_r - \int_{j \in J_p} \beta_j \mathbf{G}_j \mathbf{y}_j \right)' \mathbf{W} \left(\mathbf{G}_r \mathbf{y}_r - \int_{j \in J_p} \beta_j \mathbf{G}_j \mathbf{y}_j \right)$$

Without regularization, APE is equal to 1 minus the squared multiple regression coefficient. Steps (2) and (3) are repeated as long as the APE difference exceeds ε.

Diagnostics

The procedure produces the following diagnostics.

Descriptive Statistics

The descriptives tables gives the weighted univariate marginals and the weighted number of missing values (system missing, user defined missing, and values less than or equal to 0) for each variable.

Fit and error measures

The squared multiple regression coefficient and the Apparent Prediction Error for each iteration are reported in the History table. Also, the decrease in APE for each iteration is reported.

Summary Statistics

The following model summary statistics are available.

Multiple R

$$R = (\mathbf{G}_r \mathbf{y}_r)' \mathbf{W} \mathbf{v} \left(n_w \mathbf{v}' \mathbf{W} \mathbf{v} \right)^{-1/2}$$

Multiple R Square

R^2

Adjusted Multiple R Square

$1 - \left(1 - R^2\right)(n_w - 1)\Big(n_w - 1 - \mathbf{u}'\mathbf{f}\Big)$

with **u** a *p*-vector of ones.

Regularization "R Square" (1-Error)

$\text{RSQ}^{\text{regu}} = 1\ \text{APE}$

Without regularization, RSQ^{regu} is equal to R^2.

Apparent Prediction Error

APE as computed in the convergence step in the last iteration of the optimization algorithm. For more information, see the topic "Objective Function Optimization".

Expected Prediction Error

The expected prediction error is computed for the standardized (quantified) data. Only when for all variables the numeric scaling level is specified, the EPE is computed for the raw data as well.

Supplementary objects (test cases)

The expected prediction error for the training data (active cases) is

$$\text{EPE}^{\text{train}} = \frac{1}{n_w}\sum_{i=1}^{n}\left((\mathbf{G}_r\mathbf{y}_r)_i - \left(\sum_{j\in J_p}\beta_j\mathbf{G}_j\mathbf{y}_j\right)_i\right)^2$$

and the standard error is

$$\text{SE}^{\text{train}} = \left(\frac{1}{n_w^2}\sum_{i=1}^{n} w_i\left(\text{EPE}_i^{\text{train}}\right)^2\right)^{1/2}$$

For the test data (supplementary objects), the expected prediction error is

$$\text{EPE}^{\text{test}} = \frac{1}{n_{tot} - n}\sum_{i\in S}\left((\mathbf{G}_r\mathbf{y}_r)_i - \left(\sum_{j\in J_p}\beta_j\mathbf{G}_j\mathbf{y}_j\right)_i\right)^2$$

where *S* is the index set of supplementary objects.

$$\text{SE}^{\text{test}} = \left(\frac{1}{(n_{tot} - n)^2} \sum_{i \in S} \left(\text{EPE}_i^{\text{test}} - \text{EPE}^{\text{test}} \right)^2 \right)^{1/2}$$

For the estimation of the quantification of a supplementary category (a category only occurring with supplementary cases), see the Quantification section below.

Multiplying $\text{EPE}^{\text{train}}$, SE^{train}, EPE^{test}, and SE^{test} with

$$\frac{1}{n_w} \sum_{i=1}^{n} \left(h_{r_i} - \frac{1}{n_w} \sum_{i=1}^{n} h_{r_i} \right)^2$$

(the variance of the response variable for the active cases) yields the EPE and SE for the raw data.

Resampling, .632 Bootstrap

Bootstrap datasets are created by randomly drawing (with replacement) *n* times from the active objects (training data), including the object (case) weights.

$$\text{EPE}^{\text{boot}} = \hat{\text{Err}}^{(.632)} = \overline{\text{err}} + \hat{\text{OP}}$$

where the optimism is estimated as

$$\hat{\text{OP}} = .632 \left(\overline{\text{Err}}^{(1)} - \overline{\text{err}} \right)$$

and $\overline{\text{Err}}^{(1)}$, the leave-one-out bootstrap estimate of prediction error, is

$$\overline{\text{Err}}^{(1)} = \frac{1}{n_w^{(1)}} \sum_{i=1}^{n} \frac{1}{|C^{-i}|} \sum_{b \in C^{-i}} w_i \left(\left(\mathbf{G}_r \mathbf{y}_r^b \right)_i - \left(\sum_{j \in J_p} \beta_j^b \mathbf{G}_j \mathbf{y}_j^b \right)_i \right)^2 \text{for} |C^{-i}| \neq 0$$

where C^{-i} is the set of indices of the bootstrap samples $b\,(b = 1, ..., B)$ that

(a) do not contain observation *i*,

(b) do contain the categories that apply to observation *i* for variables with nominal or ordinal transformations,

(c) do not require extrapolation for observation *I* for variables with spline transformations.

$n_w^{(1)}$ is the number of observations for which $|C^{-i}| \neq 0$. (The set $|C^{-i}|$ may become empty if, for example, observation *i* has one of the extreme categories on a variable with a spline transformation, and this category has a frequency of one. Then each bootstrap sample that does not contain this observation, also does not contain the extreme category; thus for observation *i* all bootstrap samples are excluded.)

The Standard Error is computed as

$$\mathrm{SE}^{\mathrm{boot}} = \left(\frac{1}{n_w^2}\sum_{i=1}^{n} w_i\left(\overline{\mathrm{Err}}_i^{(1)} - \overline{\mathrm{Err}}^{(1)}\right)^2\right)^{1/2}$$

Adding multiplication with the variance of the response variable for the cases in bootstrap sample *b* in the computation of $\overline{\mathrm{Err}}^{(1)}$(... $w_i var\left(\mathbf{h}_r^b\right)$(...)), yields the EPE and SE for the raw data.

Resampling, Cross-validation

The data are randomly divided into *K* disjoint subsets of the active objects (training data), including the object (case) weights.

$$\mathrm{EPE}^{\mathrm{CV}} = \frac{1}{n_w}\sum_{i=1}^{n}\sum_{i\in k} w_i\left(\left(\mathbf{G}_r\mathbf{y}_r^k\right)_i - \left(\sum_{j\in J_p}\beta_j^{-k}\mathbf{G}_j\mathbf{y}_j^{-k}\right)_i\right)^2$$

where *k* $(k = 1, ..., K)$ indexes the *k*th subset and $-k$ the remaining part of the data.

The Standard Error is computed as

$$\mathrm{SE}^{\mathrm{CV}} = \left(\frac{1}{n_w^2}\sum_{i=1}^{n} w_i\left(\mathrm{EPE}_i^{\mathrm{CV}}\right)^2\right)^{1/2}$$

Adding multiplication with the variance of the response variable for the cases with the *k*th part removed in the computation of $\mathrm{EPE}^{\mathrm{CV}}$(... $w_i var\left(\mathbf{h}_r^{-k}\right)$(...)), yields the EPE and SE for the raw data.

Quantifications of categories that do not occur in a bootstrap sample or in the data with the *k*th part removed, are estimated as for supplementary categories (see “Quantifications ”).

ANOVA Table

	Sum of Squares	df	Mean Sum of Squares
Regression	$n_W R^2$	$\mathbf{u}'\mathbf{f}$	$n_W R^2\left(\mathbf{u}'\mathbf{f}\right)^{-1}$
Residual	$n_W\left(1 - R^2\right)$	$n_W - 1 - \mathbf{u}'\mathbf{f}$	$n_W\left(1 - R^2\right)\left(n_W - 1 - \mathbf{u}'\mathbf{f}\right)^{-1}$

$F = \mathrm{MS}_{\mathrm{reg}}/\mathrm{MS}_{\mathrm{res}}$

Correlations and Eigenvalues

Before transformation

$\mathbf{R} = n_w^{-1}\mathbf{H}'_C\mathbf{W}\mathbf{H}_C$, with $\mathbf{H}_C$ weighted centered and normalized **H** excluding the response variable.

After transformation

$\mathbf{R} = n_w^{-1}\mathbf{Q}'\mathbf{W}\mathbf{Q}$, the columns of **Q** are $\mathbf{q}_j = \mathbf{G}_j\mathbf{y}_j,\ j \in J_p$.

Statistics for Predictor Variables

The following statistics are produced for each predicted variable.

Beta

The standardized regression coefficient is Beta$_j$= b_j.

Standard Error Beta

The standard error of Beta$_j$ is estimated from 1000 bootstrap samples.

Degrees of Freedom

The degrees of freedom for a variable depend on the optimal scaling level:

Numerical. $f_j = 1$.

Spline ordinal, spline nominal. $f_j = s_j + t_j$ minus the number of elements equal to zero in $\mathbf{a}_j$.

Ordinal, nominal. f_j = the number of distinct values in $\mathbf{y}_j$ minus 1.

F-Value

$$F_j = \left(\text{Beta}_j / \text{SE}\left(\text{Beta}_j\right)\right)^2$$

Zero-order correlation

Correlations between the transformed response variable $\mathbf{G}_r\mathbf{y}_r$ and the transformed predictor variables $\mathbf{G}_j\mathbf{y}_j$:

$$r_{rj} = n_w^{-1}(\mathbf{G}_r\mathbf{y}_r)'\mathbf{W}\mathbf{G}_j\mathbf{y}_j$$

Partial correlation

PartialCorr$_j$= $b_j\left((1/t_j)\left(1 - R^2\right) + b_j^2\right)^{-1/2}$

with t_j the tolerance for variable *j* (see “Tolerance”).

When a regularization method is applied, the OLS coefficients are computed as

$$\beta^* = (n_w \mathbf{R})^{-1} \mathbf{Q}^{'} \mathbf{W} (\mathbf{G}_r \mathbf{y}_r)$$

with **R** the correlation matrix after transformation and $\mathbf{R}^{-1}$ is computed using the eigenvectors and eigenvalues of $\mathbf{R}_p$, where $\mathbf{R}_p$ is the correlation matrix of the predictors that have regression coefficients > 0, and R^2 is computed as

$$\left((\mathbf{G}_r \mathbf{y}_r)^{'} \mathbf{W} \mathbf{Q} \beta^* \left(n_w (\mathbf{Q} \beta^*)^{'} \mathbf{W} \mathbf{Q} \mathbf{B}^* \right)^{-1/2} \right)^2$$

Part correlation

PartCorr$_j$= $b_j t_j^{1/2}$

with t_j the tolerance for variable *j* (see “Tolerance”).

For computation of the OLS coefficients if regularization is applied, see “Partial correlation”.

Importance

Pratt’s measure of relative importance (Pratt, 1987)

Imp$_j$= $b_j r_{rj} / R^2$

The relative importance is only displayed if no regularization is applied.

Tolerance

The tolerance for the optimally scaled predictor variables is given by

$$t_j = r_{p_{jj}}^{-1}$$

with $r_{p_{jj}}$ the *j*th diagonal element of $\mathbf{R}_p$, where $\mathbf{R}_p$ is the correlation matrix of predictors that have regression coefficients > 0.

The tolerance for the original predictor variables is also reported and is computed in the same way, using the correlation matrix for the original predictor variables, discretized, imputed, and listwise deleted, if applicable.

Quantifications

The quantifications are $\mathbf{y}_j$, *j*=1,...,*m*.

Supplementary objects

The category indicators of supplementary objects are replaced by the quantification of the categories if these categories also appear in the active data. If a category is only used by supplementary objects, the category quantification is estimated by interpolation for variables with numeric or spline scaling level if the supplementary category lies within the range of the categories in the active data. If the variable has numeric scaling level and the non-occurring category lies outside the range of categories in the active data, then extrapolation is applied. In all other cases, the category indicator is replaced by a system-missing value.

Predicted and residual values

There is an option to save the predicted values **v** and the residual values $\mathbf{G}_r\mathbf{y}_r - \mathbf{v}$.

Whether for a supplementary object the predicted and residual value can be computed, depends on whether all categories of the object are quantified (which is the case if all categories also appear with the active objects) or can be estimated by inter- or extrapolation (see “Quantifications”).

Residual Plots

The residual plot for predictor variable *j* displays two sets of points: unnormalized quantifications ($b_j\mathbf{y}_j$) against category indicators, and residuals when the response variable is predicted from all predictor variables except variable *j* $\mathbf{G}_r\mathbf{y}_r - (\mathbf{v} - b_j\mathbf{G}_j\mathbf{y}_j)$) against category indicators.

Regularization

If regularization is specified, all above diagnostics apply to the selected or specified regularized model. If more than one model is specified (more than one penalty value), diagnostics for each model can be requested.

Statistics

APE (see “Apparent Prediction Error”), EPE (see “Expected Prediction Error”), and the Standardized sum of coefficients for each model.

The standardized sum of coefficients are computed as

$$\frac{\int\limits_{j \in J_p}^{P} \beta_j^2}{\int\limits_{j \in J_p}^{P} (\beta_j^*)^2} \text{ for Ridge}$$

$$\frac{\underset{j \in J_p}{\overset{P}{\int}} \text{sign}(\beta_j)\beta_j}{\underset{j \in J_p}{\overset{P}{\int}} \text{sign}(\beta_j^*)\beta_j^*}$$ for Lasso and Elastic Net

Coefficients

The regularized standardized coefficients for each model.

Paths

The regularized standardized coefficients are plotted on the y-axis against the standardized sum of coefficients for each model on the x-axis. For the Elastic net, multiple plots are produced: a Lasso paths plot for each specified value of the ridge penalty.

References

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions*. New York: John Wiley and Sons.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Max, J. 1960. Quantizing for minimum distortion. *Proceedings IEEE (Information Theory)*, 6, 7–12.

Pratt, J. W. 1987. Dividing the indivisible: Using simple symmetry to partition variance explained. In: *Proceedings of the Second International Conference in Statistics*, T. Pukkila, and S. Puntanen, eds. Tampere, Finland: University of Tampere, 245–260.

Ramsay, J. O. 1989. Monotone regression splines in action. *Statistical Science*, 4, 425–441.

Van der Kooij, A. J., J. J. Meulman, and W. J. Heiser. 2006. Local Minima in Categorical Multiple Regression. *Computational Statistics and Data Analysis*, 50, 446–462.

Van der Kooij, A. J. 2007. *Prediction Accuracy and Stability of Regression with Optimal Scaling Transformations (Thesis)*. : Leiden University.

CCF Algorithms

CCF computes the cross-correlation functions of two or more time series.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 14-1
Notation

Notation	**Description**
X, *Y*	Any two series of length *n*
$r_{xy}(k)$	Sample cross correlation coefficient at lag *k*
S_x	Standard deviation of series *X*
S_y	Standard deviation of series *Y*
$C_{xy}(k)$	Sample cross covariance at lag *k*

Cross Correlation

The cross correlation coefficient at lag *k* is estimated by

$$r_{xy}(k) = \frac{C_{xy}(k)}{S_x S_y}$$

where

$$C_{xy}(k) = \begin{cases} \frac{1}{n}\sum_{t=1}^{n-k} (x_t - \overline{x})(y_{t+k} - \overline{y}), & k = 0, 1, 2, \ldots \\ \frac{1}{n}\sum_{t=1}^{n+k} (y_t - \overline{y})(x_{t-k} - \overline{x}), & k = -1, -2, \ldots \end{cases}$$

$$S_x = \sqrt{\frac{1}{n}\sum_{t=1}^{n} (x_t - \overline{x})^2}$$

$$S_y = \sqrt{\frac{1}{n}\sum_{t=1}^{n} (y_t - \overline{y})^2}$$

The cross correlation function is not symmetric about $k = 0$.

Approximate standard error of $r_{xy}(k)$ is

$$se(r_{xy}(k)) \cong \sqrt{\frac{1}{n - |k|}}, k = 0, \pm 1, \pm 2, \ldots$$

The standard error is based on the assumption that the series are not cross correlated and one of the series is white noise. (The general formula for the standard error can be found in (Box and Jenkins, 1976), p. 376, 11.1.7.)

References

Box, G. E. P., and G. M. Jenkins. 1976. *Time series analysis: Forecasting and control*, Rev. ed. San Francisco: Holden-Day.

CLUSTER Algorithms

CLUSTER produces hierarchical clusters of items based on distance measures of dissimilarity or similarity.

Cluster Measures

For more information, see the topic “Proximities Measures”.

Clustering Methods

The cluster method defines the rules for cluster formation. For example, when calculating the distance between two clusters, you can use the pair of nearest objects between clusters or the pair of furthest objects, or a compromise between these methods.

Notation

The following notation is used unless otherwise stated:

Table 15-1
Notation

Notation	Description
S	Matrix of similarity or dissimilarity measures
s_{ij}	Similarity or dissimilarity measure between cluster *i* and cluster *j*
N_i	Number of cases in cluster *i*

General Procedure

Begin with *N* clusters each containing one case. Denote the clusters 1 through *N*.

- Find the most similar pair of clusters *p* and $(p \quad > \quad q)$. Denote this similarity s_{pq}. If a dissimilarity measure is used, large values indicate dissimilarity. If a similarity measure is used, small values indicate dissimilarity.
- Reduce the number of clusters by one through merger of clusters *p* and *q*. Label the new cluster $(= q)$ and update similarity matrix (by the method specified) to reflect revised similarities or dissimilarities between cluster *t* and all other clusters. Delete the row and column of *S* corresponding to cluster *p*.
- Perform the previous two steps until all entities are in one cluster.
- For each of the following methods, the similarity or dissimilarity matrix *S* is updated to reflect revised similarities or dissimilarities (s_{tr}) between the new cluster *t* and all other clusters *r* as given below.

Average Linkage between Groups

Before the first merge, let $N_i = 1$ for$i = 1$ to *N*. Update s_{tr} by

$$s_{tr} = s_{pr} + s_{qr}$$

Update N_t by

$$N_t = N_p + N_q$$

and then choose the most similar pair based on the value

$$s_{ij}/(N_i N_j)$$

Average Linkage within Groups

Before the first merge, let $SUM_i = 0$ and $N_i = 1$ for $i = 1$ to *N*. Update s_{tr} by

$$s_{tr} = s_{pr} + s_{qr}$$

Update SUM_t and N_t by

$$SUM_t = SUM_p + SUM_q + s_{pq}$$

$$N_t = N_p + N_q$$

and choose the most similar pair based on

$$\frac{SUM_i + SUM_j + s_{ij}}{((N_i + N_j)(N_i + N_j - 1))/2}$$

Single Linkage

Update s_{tr} by

$$s_{tr} = \begin{cases} \min(s_{pr}, s_{qr}) & \text{if } S \text{ is a dissimilarity matrix} \\ \max(s_{pr}, s_{qr}) & \text{if } S \text{ is a similarity matrix} \end{cases}$$

Complete Linkage

Update s_{tr} by

$$s_{tr} = \begin{cases} \max(s_{pr}, s_{qr}) & \text{if } S \text{ is a dissimilarity matrix} \\ \min(s_{pr}, s_{qr}) & \text{if } S \text{ is a similarity matrix} \end{cases}$$

Centroid Method

Update s_{tr} by

$$s_{tr} = \frac{N_p}{N_p + N_q} s_{pr} + \frac{N_q}{N_p + N_q} s_{qr} - \frac{N_p N_q}{(N_p + N_q)^2} s_{pq}$$

Median Method

Update s_{tr} by

$$s_{tr} = (s_{pr} + s_{qr})/2 - s_{pq}/4$$

Ward's Method

Update s_{tr} by

$$s_{tr} = \frac{1}{(N_t + N_r)}[(N_r + N_p)s_{rp} + (N_r + N_q)s_{rq} - N_r s_{pq}]$$

Update the coefficient *W* by

$$W = W + .5s_{pq}$$

Note that for Ward's method, the coefficient given in the agglomeration schedule is really the within-cluster sum of squares at that step. For all other methods, this coefficient represents the distance at which the clusters *p* and *q* were joined.

References

Anderberg, M. R. 1973. *Cluster analysis for applications*. New York: Academic Press.

Cluster Evaluation Algorithms

This document describes measures used for evaluating clustering models.

- The **Silhouette coefficient** combines the concepts of cluster cohesion (favoring models which contain tightly cohesive clusters) and cluster separation (favoring models which contain highly separated clusters). It can be used to evaluate individual objects, clusters, and models.
- The **sum of squares error (SSE)** is a measure of prototype-based cohesion, while **sum of squares between (SSB)** is a measure of prototype-based separation.
- **Predictor importance** indicates how well the variable can differentiate different clusters. For both range (numeric) and discrete variables, the higher the importance measure, the less likely the variation for a variable between clusters is due to chance and more likely due to some underlying difference.

Notation

The following notation is used throughout this chapter unless otherwise stated:

x_{ik}	Continuous variable *k* in case *i* (standardized).
x_{iks}	The *s*th category of variable *k* in case *i* (one-of-c coding).
N	Total number of valid cases.
N_j	The number of cases in cluster *j*.
Y	Variable with *J* cluster labels.
μ_{jk}	The centroid of cluster *j* for variable *k*.
D_{ij}	The distance between case *i* and the centroid of cluster *j*.
D_j	The distance between the overall mean u and the centroid of cluster *j*.

Goodness Measures

The average Silhouette coefficient is simply the average over all cases of the following calculation for each individual case:

$$(B - A) \,/\, \max(A, B)$$

where *A* is the average distance from the case to every other case assigned to the same cluster and *B* is the minimal average distance from the case to cases of a different cluster across all clusters.

Unfortunately, this coefficient is computationally expensive. In order to ease this burden, we use the following definitions of *A* and *B*:

- *A* is the distance from the case to the centroid of the cluster which the case belongs to;
- *B* is the minimal distance from the case to the centroid of every other cluster.

Distances may be calculated using Euclidean distances. The Silhouette coefficient and its average range between −1, indicating a very poor model, and 1, indicating an excellent model. As found by Kaufman and Rousseeuw (1990), an average silhouette greater than 0.5 indicates reasonable partitioning of data; less than 0.2 means that the data do not exhibit cluster structure.

Data Preparation

Before calculating Silhouette coefficient, we need to transform cases as follows:

1. **Recode categorical variables using one-of-c coding.** If a variable has *c* categories, then it is stored as *c* vectors, with the first category denoted (1,0,...,0), the next category (0,1,0,...,0), ..., and the final category (0,0,...,0,1). The order of the categories is based on the ascending sort or lexical order of the data values.

2. **Rescale continuous variables.** Continuous variables are normalized to the interval [−1, 1] using the transformation [2*(x−min)/(max−min)]−1. This normalization tries to equalize the contributions of continuous and categorical features to the distance computations.

Basic Statistics

The following statistics are collected in order to compute the goodness measures: the centroid μ_{jk} of variable *k* for cluster *j*, the distance between a case and the centroid, and the overall mean *u*.

For μ_{jk} with an ordinal or continuous variable *k*, we average all standardized values of variable *k* within cluster *j*. For nominal variables, μ_{jk} is a vector $\{\varphi_{jks}\}$ of probabilities of occurrence for each state *s* of variable *k* for cluster *j*. Note that in counting , we do not consider cases with missing values in variable *k*. If the value of variable *k* is missing for all cases within cluster *j*, μ_{jk} is marked as missing.

The distance D_{ij}^2 between case *i* and the centroid of cluster *j* can be calculated in terms of the weighted sum of the distance components d_{ijk}^2 across all variables; that is

$$D_{ij}^2 = \frac{\Sigma_k w_{ijk} d_{ijk}^2}{\Sigma_k w_{ijk}}$$

where w_{ijk} denotes a weight. At this point, we do not consider differential weights, thus w_{ijk} equals 1 if the variable *k* in case *i* is valid, 0 if not. If all w_{ijk} equal 0, set $D_{ij}^2 = 0$.

The distance component d_{ijk}^2 is calculated as follows for ordinal and continuous variables

$$d_{ijk}^2 = \left(x_{ik} - \mu_{jk}\right)^2$$

For binary or nominal variables, it is

$$d_{ijk}^2 = \frac{1}{S_k}\sum_{s=1}^{S_k}\left(x_{iks} - \varphi_{jks}\right)^2$$

where variable *k* uses one-of-*c* coding, and S_k is the number of its states.

The calculation of D_j is the same as that of D_{ij}, but the overall mean *u* is used in place of μ_{jk} and μ_{jk} is used in place of x_{ik}.

Silhouette Coefficient

The Silhouette coefficient of case *i* is

$$\frac{\min\{D_{ij}, j \in C_{-i}\} - D_{ic_i}}{\max(\min\{D_{ij}, j \in C_{-i}\}, D_{ic_i})}$$

where C_{-i} denotes cluster labels which do not include case *i* as a member, while c_i is the cluster label which includes case *i*. If $\max(\min\{D_{ij}, j \in C_{-i}\}, D_{ic_i})$ equals 0, the Silhouette of case *i* is not used in the average operations.

Based on these individual data, the total average Silhouette coefficient is:

$$SC = \frac{1}{N}\sum_{i=1}^{N} \frac{\min\{D_{ij}, j \in C_{-i}\} - D_{ic_i}}{\max(\min\{D_{ij}, j \in C_{-i}\}, D_{ic_i})}$$

Sum of Squares Error (SSE)

SSE is a prototype-based cohesion measure where the squared Euclidean distance is used. In order to compare between models, we will use the averaged form, defined as:

$$\text{Average SSE} = \frac{1}{N}\sum_{j \in C}\sum_{i \in j} D_{ij}^2$$

Sum of Squares Between (SSB)

SSB is a prototype-based separation measure where the squared Euclidean distance is used. In order to compare between models, we will use the averaged form, defined as:

$$\text{Average SSB} = \frac{1}{N}\sum_{j \in C} N_j D_j^2$$

Predictor Importance

The importance of field *i* is defined as

$$VI_i = \frac{-\log_{10}(sig_i)}{\max_{j \in \Omega}(-\log_{10}(sig_j))}$$

where Ω denotes the set of predictor and evaluation fields, sig_i is the significance or *p*-value computed from applying a certain test, as described below. If sig_i equals zero, set $sig_i = MinDouble$, where *MinDouble* is the minimal double value.

Across Clusters

The *p*-value for **categorical** fields is based on Pearson's chi-square. It is calculated by

p-value =Prob($\chi_d^2 > X^2$),

where

$$X^2 = \sum_{i=1}^{I} \sum_{j=1}^{J} \left(N_{ij} - \hat{N}_{ij}\right)^2 / \hat{N}_{ij}$$

where $\hat{N}_{ij} = N_{i.}N_{.j}/N(X)$.

- If $N(X) = 0$, the importance is set to be undefined or unknown;
- If $N_{i.} = 0$, subtract one from *I* for each such category to obtain $I^{'}$;
- If $N_{.j} = 0$, subtract one from *J* for each such cluster to obtain $J^{'}$;
- If $J^{'} \leq 1$ or $I^{'} \leq 1$, the importance is set to be undefined or unknown.

The degrees of freedom are $\left(I^{'} - 1\right)\left(J^{'} - 1\right)$.

The *p*-value for **continuous** fields is based on an *F* test. It is calculated by

p-value = Prob{$F(J-1, N-J) > F$},

where

$$F = \frac{\sum_{j=1}^{J} N_j \left(\overline{x}_j - \overline{\overline{x}}\right)^2 / (J-1)}{\sum_{j=1}^{J} (N_j - 1)\, s_j / (N-J)}$$

- If *N*=0, the importance is set to be undefined or unknown;
- If $N_j = 0$, subtract one from *J* for each such cluster to obtain $J^{'}$;
- If $J^{'} \leq 1$ or $N \leq J^{'}$, the importance is set to be undefined or unknown;
- If the denominator in the formula for the *F* statistic is zero, the importance is set to be undefined or unknown;
- If the numerator in the formula for the *F* statistic is zero, set *p*-value = 1;

The degrees of freedom are $\left(J^{'} - 1, N - J^{'}\right)$.

Within Clusters

The null hypothesis for **categorical** fields is that the proportion of cases in the categories in cluster *j* is the same as the overall proportion.

The chi-square statistic for cluster *j* is computed as follows

$$X^2 = \sum_{i=1}^{I} \frac{(N_{ij} - N_j p_i)^2}{N_j p_i}$$

If $N_j = 0$, the importance is set to be undefined or unknown;

If $p_i = 0$, subtract one from *I* for each such category to obtain I';

If $I' \leq 1$, the importance is set to be undefined or unknown.

The degrees of freedom are $d = I' - 1$.

The null hypothesis for **continuous** fields is that the mean in cluster *j* is the same as the overall mean.

The Student's *t* statistic for cluster *j* is computed as follows

$$t = \frac{\left(\overline{x}_j - \overline{\overline{x}}\right)}{s_j / \sqrt{N_j}}$$

with $d = N_j - 1$ degrees of freedom.

If $N_j \leq 1$ or $s_j = 0$, the importance is set to be undefined or unknown;

If the numerator is zero, set *p*-value = 1;

Here, the *p*-value based on Student's *t* distribution is calculated as

p-value = 1 − Prob{$|T(d)| \leq |t|$}.

References

Kaufman, L., and P. J. Rousseeuw. 1990. *Finding groups in data: An introduction to cluster analysis*. New York: John Wiley and Sons.

Tan, P., M. Steinbach, and V. Kumar. 2006. *Introduction to Data Mining.* : Addison-Wesley.

CNLR Algorithms

CNLR is used to estimate the parameters of a function by minimizing a smooth nonlinear loss function (objective function) when the parameters are subject to a set of constraints.

Model

Consider the model

$$f = f\left(\underset{\sim}{\mathbf{x}}, \underset{\sim}{\theta}\right)$$

where $\underset{\sim}{\theta}$ is a p×1 parameter vector, $\underset{\sim}{\mathbf{x}}$ is an independent variable vector, and f is a function of $\underset{\sim}{\mathbf{x}}$ and $\underset{\sim}{\theta}$.

Goal

Find the estimate $\underset{\sim}{\theta}^*$ of $\underset{\sim}{\theta}$ such that $\underset{\sim}{\theta}^*$ minimizes

$$F = F(y, f)$$

subject to

$$\underset{\sim}{\mathbf{l}}' \le \begin{matrix} \underset{\sim}{\theta} \\ \mathbf{A}_L \mathbf{q}_{\sim} \\ \mathbf{C}\left(\underset{\sim}{\theta}\right) \end{matrix} \le \underset{\sim}{\mathbf{u}}'$$

where F is the smooth loss function (objective function), which can be specified by the user. $\mathbf{A}_L$ is an $m_L \times p$ matrix of linear constraints, and $\mathbf{C}\left(\underset{\sim}{\theta}\right)$ is an $m_N \times 1$ vector of nonlinear constraint functions. $\underset{\sim}{\mathbf{l}}' = \left(\mathbf{l}'_{\sim B}, \mathbf{l}'_{\sim L}, \mathbf{l}'_{\sim N}\right)$, where $\mathbf{l}'_{\sim B}$, $\mathbf{l}'_{\sim L}$, and $\mathbf{l}'_{\sim N}$ represent the lower bounds, linear constraints and nonlinear constraints, respectively. The upper bound $\underset{\sim}{\mathbf{u}}'$ is defined similarly.

Algorithm

CNLR uses the algorithms proposed and implemented in NPSOL by Gill, Murray, Saunders, and Wright. A description of the algorithms can be found in the User's Guide for NPSOL, Version 4.0 (Gill, Murray, Saunders, and Wright, 1986).

The method used in NPSOL is a sequential quadratic programming (SQP) method. For an overview of SQP methods, see (Gill, Murray, and Saunders, 1981), pp. 237–242.

The basic structure of NPSOL involves major and minor iterations. Based on the given initial value $\underset{\sim}{\theta}^{(0)}$ of $\underset{\sim}{\theta}$ the algorithm first selects an initial working set that includes bounds or general inequality constraints that lie within a crash tolerance (CRSHTOL). At the kth iteration, the algorithm starts with

Minor Iteration

This iteration searches for the direction $\mathbf{P}_k$, which is the solution of a quadratic subproblem; that is, $\mathbf{P}_k$ is found by minimizing

$$\mathbf{g}'_k\mathbf{P} + \frac{1}{2}\mathbf{P}'\mathbf{H}_k\mathbf{P}$$

subject to

$$\underset{\sim}{\bar{\mathbf{l}}}^{(k)} \le \left\{ \begin{matrix} \mathbf{P} \\ \mathbf{A}_L\mathbf{P} \\ \mathbf{A}_N\mathbf{P} \end{matrix} \right\} \le \underset{\sim}{\bar{\mathbf{u}}}^{(k)}$$

where $\mathbf{g}_k$ is the gradient of F at $\underset{\sim}{\theta}^{(k)}$, the matrix $\mathbf{H}_k$ is a positive-definite quasi-Newton approximation to the Hessian of the Lagrangian function, $\mathbf{A}_N$ is the Jacobian matrix of the nonlinear-constraint vector $\mathbf{C}$ evaluated at $\underset{\sim}{\theta}^{(k)}$, and

$$\underset{\sim}{\bar{\mathbf{l}}}^{(k)} = \left(\bar{\mathbf{l}}'_B, \bar{\mathbf{l}}'_L, \bar{\mathbf{l}}'_N\right)$$

$$\underset{\sim}{\bar{\mathbf{u}}}^{(k)} = \left(\bar{\mathbf{u}}'_B, \mathbf{u}'_L, \mathbf{u}'_N\right)$$

$$\bar{\mathbf{l}}_B = \mathbf{l}_B - \underset{\sim}{\mathbf{q}}^{(k)}$$

$$\bar{\mathbf{l}}_L = \mathbf{l}_L - \mathbf{A}_L\underset{\sim}{\mathbf{q}}^{(k)}$$

$$\bar{\mathbf{l}}_N = \mathbf{l}_N - \mathbf{C}\left(\underset{\sim}{\mathbf{q}}^{(k)}\right)$$

$$\bar{\mathbf{u}}_B = \mathbf{u}_B - \underset{\sim}{\mathbf{q}}^{(k)}$$

$$\bar{\mathbf{u}}_L = \mathbf{u}_L - \mathbf{A}_L\underset{\sim}{\mathbf{q}}^{(k)}$$

$$\bar{\mathbf{u}}_N = \mathbf{u}_N - \mathbf{C}\left(\underset{\sim}{\mathbf{q}}^{(k)}\right)$$

The linear feasibility tolerance, the nonlinear feasibility tolerance, and the feasibility tolerance are used to decide if a solution is feasible for linear and nonlinear constraints.

Once the search direction $\mathbf{P}_k$ is found, the algorithm goes to the major iteration.

Major Iteration

The purpose of the major iteration is to find a non-negative scalar α_k such that

$$\underset{\sim}{\theta}^{(k+1)} = \underset{\sim}{\theta}^{(k)} + \alpha_k\mathbf{P}_k$$

satisfies the following conditions:

- $\underset{\sim}{\theta}^{(k+1)}$ produces a "sufficient decrease" in the augmented Lagrangian merit function

  $$L\left(\underset{\sim}{\theta},\underset{\sim}{\lambda},\underset{\sim}{\mathbf{s}}\right)=F\left(\underset{\sim}{\theta}\right)-\sum_i \lambda_i\left(c_i\left(\underset{\sim}{\theta}\right)-s_i\right)+\frac{1}{2}\sum_i \rho_i\left(c_i\left(\underset{\sim}{\theta}\right)-s_i\right)^2$$

  The summation terms involve only the nonlinear constraints. The vector λ is an estimate of the Lagrange multipliers for the nonlinear constraints. The non-negative slack variables $\{s_i\}$ allow nonlinear inequality constraints to be treated without introducing discontinuities. The solution of the *QP* subproblem defined in "Minor Iteration " provides a vector triple that serves as a direction search for $\underset{\sim}{\theta}$, $\underset{\sim}{\lambda}$ and $\underset{\sim}{\mathbf{s}}$. The non-negative vector of penalty parameters (ρ) is initialized to zero at the beginning of the first major iteration. Function precision criteria are used as a measure of the accuracy with which the functions *F* and c_i can be evaluated.

- $\underset{\sim}{\theta}^{(k+1)}$ is close to a minimum of *F* along $\mathbf{P}_k$. The criterion is

  $$\left|\mathbf{g}'\left(\underset{\sim}{\theta}^{(k+1)}\right)\mathbf{P}_k\right|<-\eta\mathbf{g}'_k\mathbf{P}_k$$

  where η is the Line Search Tolerance and $0\le\eta<1$. The value of η determines the accuracy with which α_k approximates a stationary point of *F* along $\mathbf{P}_k$. A smaller value of η produces a more accurate line search.

- The step length is in a certain range; that is,

  $\|\underset{\sim}{\theta}^{(k+1)}-\underset{\sim}{\theta}^{(k)}\|=\|\alpha_k\mathbf{P}_k\|\le$ Step Limit

Convergence Tests

After α_k is determined from the major iteration, the following conditions are checked:

- $k+1\le$ Maximum number of major iterations
- The sequence $\left\{\underset{\sim}{\theta}^{(l)}\right\}$ converged at $\underset{\sim}{\theta}^{(k+1)}$; that is,

  $$\|\alpha_k\mathbf{P}_k\|\le\sqrt{r}\left(1+\|\underset{\sim}{\theta}^{(k+1)}\|\right)$$

- $\underset{\sim}{\theta}^{(k+1)}$ satisfies the Kuhn-Tucker conditions to the accuracy requested; that is,

  $$\|\mathbf{g}_z\left(\underset{\sim}{\theta}^{(k+1)}\right)\|\le\sqrt{r}\left(1+\max\left(1+\left|F\left(\underset{\sim}{\theta}^{(k+1)}\right)\right|,\|g\left(\underset{\sim}{\theta}^{(k+1)}\right)\|\right)\right)$$

  and

  $\|\text{res}_j\|\le$ FTOL for all *j*,

  where $\mathbf{g}_z$ is the projected gradient, **g** is the gradient of *F* with respect to the free parameters, res_j is the violation of the *j*th nonlinear constraint, FTOL is the Nonlinear Feasibility Tolerance, and *r* is the Optimality Tolerance.

If none of these three conditions are satisfied, the algorithm continues with the Minor Iteration to find a new search direction.

Termination

The following are termination conditions.

- Underflow. A single underflow will always occur if machine constants are computed automatically. Other floating-point underflows may occur occasionally, but can usually be ignored.
- Overflow. If the printed output before the overflow error contains a warning about serious ill-conditioning in the working set when adding the *j*th constraint, it may be possible to avoid the difficulty by increasing the magnitude of FTOL, LFTOL, or NFTOL and rerunning the program. If the message recurs after this change, the offending linearly dependent constrains (with index "*j*") must be removed from the problem.
- Optimal solution found.
- Optimal solution found, but the requested accuracy could not be achieved, NPSOL terminates because no further improvement can be made in the merit function. This is probably caused by requesting a more accurate solution than is attainable with the given precision of the problem (as specified by FPRECISION).
- No point has been found that satisfies the linear constraints. NPSOL terminates without finding a feasible point for the given value of LFTOL. The user should check that there are no constraint redundancies and ensure that the value of LFTOL is greater than the precision of parameter estimates.
- No point has been found which satisfies the nonlinear constraints. There is no feasible point found in *QP* subproblems. The user should check the validity of constraints. If the user is convinced that a feasible point does exist, NPSOL should be restarted at a different starting point.
- Too many iterations. If the algorithm appears to be making progress, increase the value of ITER and rerun NPSOL. If the algorithm seems to be "bogged down", the user should check for incorrect gradients.
- Cannot improve on current point. A sufficient decrease in the merit function could not be attained during the final line search. This sometimes occurs because an overly stringent accuracy has been requested; for example, Optimality Tolerance is too small or a too-small step limit is given when the parameters are measured on different scales.

Please note the following:

- Unlike the other procedures, the weight function is not treated as a case replicate in CNLR.
- When both weight and loss function are specified, the algorithm takes the product of these two functions are the loss function.
- If the loss function is not specified, the default loss function is a squared loss function and the default output in NLR will be printed. However, if the loss function is not a squared loss function, CNLR prints only the final parameter estimates, iteration history, and termination message. In order to obtain estimates of the standard errors of parameter estimates and correlations between parameter estimates, the bootstrapping method can be requested.

Bootstrapping Estimates

Bootstrapping is a nonparametric technique of estimating the standard error of a parameter estimate using repeated samples from the original data. This is done by sampling with replacement. CNLR computes and saves the parameter estimates for each sample generated. This results, for each parameter, in a sample of estimates from which the standard deviation is calculated. This is the estimated standard error.

Mathematically, the bootstrap covariance matrix **S** for the *p* parameter estimates is

$$\mathbf{S} = (s_{ij})_{p \times p}$$

where

$$s_{ij} = \sum_{k=1}^{m} \left(\hat{\theta}_{ik} - \overline{\theta}_i\right)\left(\hat{\theta}_{jk} - \overline{\theta}_j\right)$$
$$\overline{\theta}_i = \sum_{k=1}^{m} \hat{\theta}_{ik}/m$$

and $\hat{\theta}_{ik}$ is the CNLR parameter estimate of θ_i for the *k*th bootstrap sample and m is the number of samples generated by the bootstrap. By default, $m = \frac{10p(p+1)}{2}$. The standard error for the *j*th parameter estimate is estimated by

$$\sqrt{\frac{s_{jj}}{m-1}}$$

and the correlation between the *i*th and *j*th parameter estimates is estimated by

$$\frac{s_{ij}}{\sqrt{s_{ii}s_{jj}}}$$

The “95% Trimmed Range” values are the most extreme values that remain after trimming from the set of estimates for a parameter, the *g* largest, and the *g* smallest estimates, where *g* is the largest integer not exceeding 0.025*m*.

References

Gill, P. E., W. M. Murray, and M. A. Saunders. 1981. *Practical Optimization*. London: Academic Press.

Gill, P. E., W. M. Murray, M. A. Saunders, and M. H. Wright. 1984. Procedures for optimization problems with a mixture of bounds and general linear constraints. *ACM Transactions on Mathematical Software*, 10:3, 282–296.

Gill, P. E., W. M. Murray, M. A. Saunders, and M. H. Wright. 1986. *User’s guide for NPSOL (version 4.0): A FORTRAN package for nonlinear programming. Technical Report SOL 86-2.* Stanford University: Department of Operations Research.

CONJOINT Algorithms

This procedure performs conjoint analysis using ordinary least squares.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	The total number of regular cards in the PLAN file.
p	The total number of factors.
d	The number of discrete factors.
l	The number of linear factors.
q	The number of quadratic factors.
m_i	The number of levels of levels of the *i*th discrete factor.
a_{ij}	The *j*th level of the *i*th discrete factor *i*=1,...,*d*.
x_i	The *i*th linear factor, *i*=1,...,*l*.
z_i	The *i*th ideal or anti-ideal factor, *i*=1,...,*q*.
r_i	The response for the *i*th card, *i*=*i*,...,*n*.
t	The total number of subjects being analyzed at the same time. (When /SUBJECT is specified, *t* is usually 1.)

Model

The model for the response r_i for the *i*th card from a subject is

$$r_i = \beta_0 + \sum_{j=1}^{p} u_{jk_{ji}}$$

where $u_{jk_{ji}}$ is the utility (part worth) associated with the k_{ji}th level of the *j*th factor on the *i*th card.

Design Matrix

A design matrix **X** is formed from the values in the PLAN file. There is one row for each card in the PLAN file. The columns of the matrix are defined by each of the factor variables in the following manner:

- There is a column of 1s for the constant. This column is used for the estimate of β_0^*.
- For each discrete factor containing m_i levels, $m_i - 1$ columns are formed. Each column represents the deviation of one of the factor levels from the overall mean. There is a 1 in the column if that level of the factor was observed, *a*−1 if the last level of the factor was observed, or a 0 otherwise. These columns are used to estimate the $m_i - 1$ values of α_{ij}.

- For each linear factor, there is one column which is the centered value of that factor $(x_{ij} - \overline{x}_i)$ These columns are used to estimate the values for $\hat{\beta}_i$.
- For each quadratic factor there are two columns, one which contains the centered value of the factor $(z_{ij} - \overline{z}_i)$, the next which contains the square of the centered factor value $(z_{ij} - \overline{z}_i)^2$ These columns are used to estimate the values of $\hat{\gamma}^*$.

Converting Card Numbers to Ranks

If the observations are card numbers, they are converted to ranks. If card number *i* has a value of *k*, then $r_i = k$.

Estimation

The estimates

$$\left(\hat{\beta}_0^* \hat{\alpha} \hat{\beta} \hat{\gamma}^*\right)' = \left(X'X\right)^{-1} X'y$$

are computed by using a QR decomposition (see MANOVA) where

$$y_i = \begin{cases} r_i & \text{if responses are scores} \\ n+1-r_i & \text{if responses are ranks} \end{cases}$$

The variance-covariance matrix of these estimates is

$$\hat{\sigma}^2 \left(X'X\right)^{-1}$$

where

$$\hat{\sigma}^2 = \sum_{i=1}^{t} \sum_{j=1}^{n} (r_{ij} - \hat{r}_{ij})^2 / (nt - d - l - 2q - 1)$$

The values of $\hat{\gamma}$ are computed by

$$\hat{\gamma}_{i1} = \hat{\gamma}_{i1}^* - 2\hat{\gamma}_{i2}^* \overline{z}_i$$

and

$$\hat{\gamma}_{i2} = \hat{\gamma}_{i2}^*$$

with variances

$$var(\hat{\gamma}_{j1}) = var\left(\hat{\gamma}_{j1}^*\right) - 4\overline{z}_j cov\left(\hat{\gamma}_{j1}^*, \hat{\gamma}_{j2}\right) + 4\overline{z}_j^2 var(\hat{\gamma}_{j2})$$

and

$$var(\hat{\gamma}_{j2}) = var\left(\hat{\gamma}^*_{j2}\right)$$

where

$$cov(\hat{\gamma}_{j1}, \hat{\gamma}_{j2}) = cov\left(\hat{\gamma}^*_{j1}, \hat{\gamma}_{j2}\right) - 2\overline{z}^2 var(\hat{\gamma}_{j2})$$

The value for $\hat{\beta}_0$ is calculated by

$$\hat{\beta}_0 = \hat{\beta}^*_0 - \Sigma \hat{\beta}_i \overline{x}_i - \Sigma\left(\hat{\gamma}_{i1}\overline{z}_i + \hat{\gamma}_{i2}\overline{z}^2_i\right)$$

with variance

$$var\left(\hat{\beta}_0\right) = a\Sigma_a^{-1}a'$$

where

$$a = \left(1, -\overline{x}_1, \ldots, -\overline{x}_l, -\overline{z}_1, \overline{z}^2_1, \ldots, -\overline{z}_q, \overline{z}^2_q\right)$$

and

$$\Sigma_a = \begin{pmatrix} var\hat{\beta}^*_0 & cov\left(\hat{\beta}^*_0, \hat{\beta}_1\right) & cov\left(\hat{\beta}^*_0, \hat{\gamma}^*_{q1}\right) & cov\left(\hat{\beta}^*_0, \hat{\gamma}^*_{q2}\right) \\ & var\hat{\beta}_1 & & \\ & & var\hat{\gamma}^*_{q1} & \\ & & & var\hat{\gamma}^*_{q2} \end{pmatrix}$$

Utility (Part Worth) Values

Discrete Factors

$$\hat{u}_{jk} = \begin{cases} \hat{a}_{jk} & \text{for } k = 1, \ldots, m_j - 1 \\ -\sum_{j=1}^{m_j-1} \hat{a}_{jk} & \text{for } k = m_j \end{cases}$$

Linear Factors

$$\hat{u}_{jk} = \hat{\beta}_j x_k$$

Ideal or Anti-ideal Factors

$$\hat{u}_{jk} = \hat{\gamma}_{j1} z_{jk} + \hat{\gamma}_{j2} z^2_{jk}$$

Standard Errors of Part Worths

The standard error of part worth $u_{jk} = \sqrt{var(u_{jk})}$ where $var(u_{jk})$ is defined below:

Discrete Factors

$$var(u_{jk}) = \begin{cases} var(\hat{\alpha}_{jk}) & \text{for } k = 1, \ldots, m_j - 1 \\ \sum_{i=1}^{m_j-1} var(\hat{\alpha}_{jk}) - 2\sum_{i=1}^{m_j-1} \sum_{l<i} cov(\hat{\alpha}_{ji}, \hat{\alpha}_{jl}) & \text{for } k = m_j \end{cases}$$

Linear Factors

$$var(u_{jk}) = x_k^2 var\left(\hat{\beta}_j\right)$$

Ideal or Anti-ideal Factors

$$var(u_{jk}) = z_k^2 var(\hat{\gamma}_{j1}) + 2z_k^3 cov(\hat{\gamma}_{j1}, \hat{\gamma}_{j2}) + z_k^4 var(\hat{\gamma}_{j2})$$

Importance Scores

The importance score for factor *i* is

$$IMP_i = 100 \frac{RANGE_i}{\sum_{i=1}^{p} RANGE_i}$$

where $RANGE_i$ is the highest minus lowest utility for factor *i*. If there is a SUBJECT command, the importance for each factor is calculated separately for each subject, and these are then averaged.

Predicted Scores

$$\hat{r}_i = \hat{\beta}_0 + \sum_{j=1}^{p} \hat{u}_{jk_{ji}}$$

where $\hat{u}_{jk_{ji}}$ is the estimated utility (part worth) associated with the k_{ji}th level of the *j*th factor.

Correlations

Pearson and Kendall correlations are calculated between predicted $(\hat{r}_i)$ and the observed (r_i) responses. See the CORRELATIONS and NONPAR CORR chapters for algorithms. Pearson correlations for holdouts are not calculated.

Simulations

Each person is assigned a probability p_i for each simulation *i*. The probabilities are all computed based on the predicted score $(\hat{r}_i)$ for that product. The probabilities are computed as follows:

Max Utility

$$p_i = \begin{cases} 1 & \text{if } \hat{r}_i = \max(\hat{r}_i) \\ 0 & \text{otherwise} \end{cases}$$

BTL

$$p_i = \frac{\hat{r}_i}{\sum_j \hat{r}_j}$$

Logit

$$p_i = \frac{e^{\hat{r}_i}}{\sum_j e^{\hat{r}_j}}$$

Probabilities are averaged across respondents for the grouped simulation results. For the BTL and Logit methods, only subjects having all positive $\hat{r}_i$ values are used.

CORRELATIONS Algorithms

The user-specified treatment for missing values is used for computation of all statistics except, under certain conditions, the means and standard deviations.

Notation

The following notation is used throughout this section unless otherwise specified:

Table 19-1
Notation

Notation	Description
N	Number of cases
X_{kl}	Value of the variable *k* for case *l*
w_l	Weight for case *l*
W_k	Sum of weights of cases used in computation of statistics for variable *k*
W_{kj}	Sum of weights of cases used in computation of statistics for variables *k* and *j*

Statistics

The following statistics are available.

Means and Standard Deviations

$$\overline{X}_k = \frac{\sum_{l=1}^{N} w_l X_{kl}}{W_k}$$

$$S_k = \sqrt{\left(\sum_{l=1}^{N} w_l X_{kl}^2 - \overline{X}_k^2 W_k\right) / (W_k - 1)}$$

Note: If no treatment for missing values is specified (default is pairwise), means and standard deviations are computed based on all nonmissing values for each variable. If missing values are to be included or listwise is chosen, that option is used for means and standard deviations as well.

Cross-product Deviations and Covariances

The cross-product deviation for variables *i* and *j* is

$$C_{ij} = \sum_{l=1}^{N} w_l X_{il} X_{jl} - \left(\sum_{l=1}^{N} w_l X_{il}\right)\left(\sum_{l=1}^{N} w_l X_{jl}\right) / W_{ij}$$

The covariance is

$$S_{ij} = \frac{C_{ij}}{W_{ij} - 1}$$

Pearson Correlation

$$r_{ij} = \frac{C_{ij}}{\sqrt{C_{ii}C_{jj}}}$$

Significance Level of r

The significance level for r_{ij} is based on

$$t = r_{ij}\sqrt{\frac{W_{ij} - 2}{1 - r_{ij}^2}}$$

which, under the null hypothesis, is distributed as a *t* with $W_{ij} - 2$ degrees of freedom. By default, the significance level is two-tailed.

References

Blalock, H. M. 1972. *Social statistics*. New York: McGraw-Hill.

CORRESPONDENCE Algorithms

The CORRESPONDENCE algorithm consists of three major parts:

1. A singular value decomposition (SVD)
2. Centering and rescaling of the data and various rescalings of the results
3. Variance estimation by the delta method.

Other names for SVD are "Eckart-Young decomposition" after Eckart and Young (1936), who introduced the technique in psychometrics, and "basic structure" (Horst, 1963). The rescalings and centering, including their rationale, are well explained in Benzécri (1969), Nishisato (1980), Gifi (1981), and Greenacre (1984). Those who are interested in the general framework of matrix approximation and reduction of dimensionality with positive definite row and column metrics are referred to Rao (1980). The delta method is a method that can be used for the derivation of asymptotic distributions and is particularly useful for the approximation of the variance of complex statistics. There are many versions of the delta method, differing in the assumptions made and in the strength of the approximation (Rao, 1973, ch. 6; Bishop et al., 1975, ch. 14; Wolter, 1985, ch. 6).

Other characteristic features of CORRESPONDENCE are the ability to fit supplementary points into the space defined by the active points, the ability to constrain rows and/or columns to have equal scores, and the ability to make biplots using either chi-squared distances, as in standard correspondence analysis, or Euclidean distances.

Notation

The following notation is used throughout this chapter unless otherwise stated:

t_1	Total number of rows (row objects)
s_1	Number of supplementary rows
k_1	Number of rows in analysis ($t_1 - s_1$)
t_2	Total number of columns (column objects)
s_2	Number of supplementary columns
k_2	Number of columns in analysis ($t_2 - s_2$)
p	Number of dimensions

Data-related quantities:

f_{ij}	Nonnegative data value for row *i* and column *j*: collected in table *F*
f_{i+}	Marginal total of row *i*, $i = 1, \ldots, k_1$
f_{+j}	Marginal total of column *j*, $j = 1, \ldots, k_2$
N	Grand total of *F*

Scores and statistics:

r_{is}	Score of row object *i* on dimension *s*
c_{js}	Score of column object *j* on dimension *s*
I	Total inertia

Basic Calculations

One way to phrase the CORRESPONDENCE objective (cf. Heiser, 1981) is to say that we wish to find row scores $\{r_{is}\}$ and column scores $\{c_{js}\}$ so that the function

$$\sigma(\{r_{is}\};\{c_{js}\}) = \sum_i \sum_j f_{ij} \sum_s (r_{is} - c_{js})^2$$

is minimal, under the standardization restriction either that

$$\sum_i f_{i+} r_{is} r_{it} = \delta^{st}$$

or

$$\sum_j f_{+j} c_{js} c_{jt} = \delta^{st}$$

where δ^{st} is Kronecker's delta and *t* is an alternative index for dimensions. The trivial set of scores ({1},{1}) is excluded.

The CORRESPONDENCE algorithm can be subdivided into five steps, as explained below.

Data scaling and centering

When rows and/or columns are specified to be equal, first the frequencies of the rows/columns to be equal are summed. The sums are put into the row/column with the smallest row/column number and the other rows/columns are set to zero.

Measure is Chi Square

The first step is to form the auxiliary matrix **Z** with general element

$$z_{ij} = \frac{f_{ij}}{\sqrt{f_{i+} f_{+j}}} - \frac{\sqrt{f_{i+} f_{+j}}}{N}$$

The standardization with Chi Square measure is always rcmean (both row and column means removed.

Measure is Euclidean

When Euclidean measure is chosen, the auxiliary matrix **Z** is formed in two steps:

- $\tilde{z}_{ij} = \tilde{f}_{ij} - \frac{\tilde{f}_{i+} \tilde{f}_{+j}}{N}$

With $f_{ij}^{\sim}$, $f_{i+}^{\sim}$, and $f_{j+}^{\sim}$ depending on the standardization option.

rmean (remove row means). $f_{ij}^{\sim} = f_{ij}$; $f_{i+}^{\sim} = f_{i+}$; $f_{+j}^{\sim} = \frac{N}{k_2}$

cmean (remove column means). $f_{ij}^{\sim} = f_{ij}$; $f_{i+}^{\sim} = \frac{N}{k_1}$; $f_{j+}^{\sim} = f_{+j}$

rcmean (remove both row and column means). $f_{ij}^{\sim} = f_{ij}$; $f_{i+}^{\sim} = f_{i+}$; $f_{+j}^{\sim} = f_{+j}$

rsum (equalize row totals, then remove row means). $f_{ij}^{\sim} = \frac{f_{ij}f_{i+}^{\sim}}{f_{i+}}$; $f_{i+}^{\sim} = \frac{N}{k_1}$; $f_{+j}^{\sim} = \frac{N}{k_2}$

csum (equalize column totals, then remove column means). $f_{ij}^{\sim} = \frac{f_{ij}f_{+j}^{\sim}}{f_{+j}}$; $f_{i+}^{\sim} = \frac{N}{k_1}$; $f_{+j}^{\sim} = \frac{N}{k_2}$

- Then, if not computed yet in the previous step, $f_{i+}^{\sim}$, or/and $f_{+j}^{\sim}$ are computed:

$f_{i+}^{\sim} = \frac{N}{k_1}$, $f_{+j}^{\sim} = \frac{N}{k_2}$, and $z_{ij} = \frac{z_{ij}^{\sim}}{\sqrt{f_{i+}^{\sim} f_{+j}^{\sim}}}$

Singular value decomposition

When rows and/or columns are specified as supplementary, first these rows and/or colums of **Z** are set to zero, yielding $\underline{\mathbf{Z}}$

Let the singular value decomposition of $\underline{\mathbf{Z}}$ be denoted by

$$\underline{\mathbf{Z}} = \mathbf{K\Lambda L}^{'}$$

with $\mathbf{K}^{'}\mathbf{K} = \mathbf{I}$, $\mathbf{L}^{'}\mathbf{L} = \mathbf{I}$, and $\mathbf{\Lambda}$ diagonal. This decomposition is calculated by a routine based on Golub and Reinsch (1971). It involves Householder reduction to bidiagonal form and diagonalization by a QR procedure with shifts. The routine requires an array with more rows than columns, so when $k_1 < k_2$ the original table is transposed and the parameter transfer is permuted accordingly.

Adjustment to the row and column metric

The arrays of both the left-hand singular vectors and the right-hand singular vectors are adjusted row-wise to form scores that are standardized in the row and in the column marginal proportions, respectively:

$$\tilde{r}_{is} = k_{is}/\sqrt{f_{i+}^{\sim}/N},$$

$$\tilde{c}_{js} = l_{js}/\sqrt{f_{+j}^{\sim}/N}.$$

This way, both sets of scores satisfy the standardization restrictions simultaneously.

Determination of variances and covariances

For the application of the delta method to the results of generalized eigenvalue methods under multinomial sampling, the reader is referred to Gifi (1990, ch. 12) and Israëls (1987, Appendix B). It is shown there that *N* time variance-covariance matrix of a function φ of the observed cell proportions $p = \{p_{ij} = f_{ij}^{\sim}/N\}$ asymptotically reaches the form

$$N \times cov(\phi(p)) \simeq \sum_i \sum_j \pi_{ij} \left(\frac{\partial\phi}{\partial p_{ij}}\right)\left(\frac{\partial\phi}{\partial p_{ij}}\right)' - \left(\sum_i \sum_j \pi_{ij} \frac{\partial\phi}{\partial p_{ij}}\right)\left(\sum_i \sum_j \pi_{ij} \frac{\partial\phi}{\partial p_{ij}}\right)'$$

Here the quantities π_{ij} are the cell probabilities of the multinomial distribution, and $\partial\phi/\partial p_{ij}$ are the partial derivatives of φ (which is either a generalized eigenvalue or a generalized eigenvector) with respect to the observed cell proportion. Expressions for these partial derivatives can also be found in the above-mentioned references.

Normalization of row and column scores

Depending on the normalization option chosen, the scores are normalized. The normalization option *q* can be chosen to be any value in the interval [-1,1] or it can be specified according to the following designations:

$$q = \begin{cases} 0, & \text{symmetrical} \\ 1, & \text{row principal} \\ -1, & \text{column principal} \end{cases}$$

There is a fifth possibility, choosing the designation "principal", that does not correspond to a *q*-value.

When "principal" is chosen, normalization parameters α for the rows and β for the columns are both set to 1. When one of the other options is chosen, α and β are functions of *q*:

$\alpha = (1+q)/2$

$\beta = (1-q)/2$

The normalization implies a compensatory rescaling of the coordinate axes of the row scores and the column scores:

$r_{is} = \tilde{r}_{is}\lambda_s^{\alpha}$,

$c_{js} = \tilde{c}_{js}\lambda_s^{\beta}$.

The general formula for the weighted sum of squares that results from this rescaling is

row scores: $\sum_i f_{i+}^{\sim} r_{is}^2 = N\lambda_s^{2\alpha}$

column scores: $\sum_j f_{+j}^{\sim} c_{js}^2 = N\lambda_s^{2\beta}$

The estimated variances and covariances are adjusted according to the type of normalization chosen.

Diagnostics

After printing the data, CORRESPONDENCE optionally also prints a table of row profiles and column profiles, which are $\{f_{ij}/f_{i+}\}$ and $\{f_{ij}/f_{+j}\}$, respectively.

Singular Values, Maximum Rank and Inertia

All singular values λ_s defined in the second step are printed up to a maximum of $\min\{(k_1-1),(k_2-1)\}$. Small singular values and corresponding dimensions are suppressed when they don't exceed the quantity $(k_1k_2)^{1/2}10^{-7}$; in this case a warning message is issued. Dimensionwise inertia and total inertia are given by the relationships

$$I=\sum_s \lambda_s^2=\sum_s\sum_i \frac{f_{i+}^{\sim} r_{is}^2}{N}$$

where the right-hand part of this equality is true only if the normalization is row principal (but for the other normalizations similar relationships are easily derived from step 5). The quantities "proportion explained" are equal to inertia divided by total inertia: λ_s^2/I.

Supplementary Points

Supplementary row and column points are given by

$$r_{is}^{sup}=\sum_j \frac{f_{ij}^{\sim}}{f_{i+}^{\sim}} c_{js}\lambda_s^{2\alpha-2}$$

$$c_{js}^{sup}=\sum_i \frac{f_{ij}^{\sim}}{f_{+j}^{\sim}} r_{is}\lambda_s^{2\beta-2}$$

Mass, Scores, Inertia and Contributions

The mass, scores, inertia and contributions for the row and columns points (including supplementary points) are given in the Overview Row Points Table and the Overview Column Points Table. These tables are printed in *p* dimensions. The tables are given first for rows, then for columns. The masses are the marginal proportions ($f_{i+}^{\sim}/N$ and $f_{+j}^{\sim}/N$, respectively). The inertia of the rows/columns is given by:

$$I_i=\sum_j^{k_2} z_{ij}^2$$

$$I_j=\sum_i^{k_1} z_{ij}^2$$

For supplementary points, the contribution to the inertia of dimensions is zero. The contribution of the active points to the inertia of each dimension is given by

$$\tau_{is}=\frac{f_{i+}^{\sim}}{N}\frac{r_{is}^2}{\lambda_s^{2\alpha}}$$

$$\tau_{js}=\frac{f_{+j}^{\sim}}{N}\frac{c_{js}^2}{\lambda_s^{2\beta}}$$

The contribution of dimensions to the inertia of each point is given by

$$\sigma_{is} = \frac{f_{i+}^{\sim}}{N}\frac{r_{is}^2\lambda_s^{2-2\alpha}}{I_i}$$

$$\sigma_{js} = \frac{f_{+j}^{\sim}}{N}\frac{c_{js}^2\lambda_s^{2-2\beta}}{I_j}$$

Confidence Statistics of Singular Values and Scores

The computation of variances and covariances is explained in step 4. Since the row and column scores are linear functions of the singular vectors, an adjustment is necessary depending on the normalization option chosen. From these adjusted standard deviations and correlations are derived in the standard way.

Permutations of the Input Table

For each dimension *s*, let $\rho(i|s)$ be the permutation of the first t_1 integers that would sort the *s*th column of $\{r_{is}\}$ in ascending order. Similarly, let $\rho(j|s)$ be the permutation of the first t_2 integers that would sort the *s*th column of $\{c_{js}\}$ in ascending order. Then the permuted data matrix is given by $\{f_{\rho(i|s),\rho(j|s)}\}$.

References

Benzécri, J. P. 1969. Statistical analysis as a tool to make patterns emerge from data. In: *Methodologies of Pattern Recognition,* S. Watanabe, ed. New York: Academic Press, 35–74.

Bishop, Y. M., S. E. Fienberg, and P. W. Holland. 1977. *Discrete Multivariate Analysis: Theory and Practice*. Cambridge, MA: MIT Press.

Eckart, C., and G. Young. 1936. The approximation of one matrix by another one of lower rank. *Psychometrika*, 1, 211–218.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Golub, G. H., and C. Reinsch. 1971. Linear Algebra. In: *Handbook for Automatic Computation, Volume II,* J. H. Wilkinson, and C. Reinsch, eds. New York: Springer-Verlag.

Greenacre, M. J. 1984. *Theory and applications of correspondence analysis*. London: Academic Press.

Heiser, W. J. 1981. *Unfolding analysis of proximity data*. Leiden: Department of Data Theory, University of Leiden.

Horst, P. 1963. *Matrix algebra for social scientists*. New York: Holt, Rinehart and Winston.

Israëls, A. 1987. *Eigenvalue techniques for qualitative data*. Leiden: DSWO Press.

Nishisato, S. 1980. *Analysis of categorical data: Dual scaling and its applications*. Toronto: University of Toronto Press.

Rao, C. R. 1973. *Linear statistical inference and its applications*, 2nd ed. New York: John Wiley and Sons.

Rao, C. R. 1980. Matrix approximations and reduction of dimensionality in multivariate statistical analysis. In: *Multivariate Analysis, Vol. 5,* P. R. Krishnaiah, ed. Amsterdam: North-Holland, 3–22.

Wolter, K. M. 1985. *Introduction to variance estimation*. Berlin: Springer-Verlag.

COXREG Algorithms

Cox (1972) first suggested the models in which factors related to lifetime have a multiplicative effect on the hazard function. These models are called proportional hazards models. Under the proportional hazards assumption, the hazard function *h* of *t* given *X* is of the form

$$h(t|\mathbf{x}) = h_0(t)e^{\mathbf{x}'\beta}$$

where **x** is a known vector of regressor variables associated with the individual, β is a vector of unknown parameters, and $h_0(t)$ is the baseline hazard function for an individual with $\mathbf{x} = 0$. Hence, for any two covariates sets $\mathbf{x}_1$ and $\mathbf{x}_2$, the log hazard functions $h(t|\mathbf{x}_1)$ and $h(t|\mathbf{x}_2)$ should be parallel across time.

When a factor does not affect the hazard function multiplicatively, stratification may be useful in model building. Suppose that individuals can be assigned to one of *m* different strata, defined by the levels of one or more factors. The hazard function for an individual in the *j*th stratum is defined as

$$h_j(t|\mathbf{x}) = h_{0j}(t)e^{\mathbf{x}'\beta}$$

There are two unknown components in the model: the regression parameter β and the baseline hazard function $h_{0j}(t)$. The estimation for the parameters is described below.

Estimation

We begin by considering a nonnegative random variable *T* representing the lifetimes of individuals in some population. Let $f(t|\mathbf{x})$ denote the probability density function (pdf) of *T* given a regressor *x* and let $S(t|\mathbf{x})$ be the survivor function (the probability of an individual surviving until time *t*). Hence

$$S(t|\mathbf{x}) = \int_t^\infty f(u|\mathbf{x})du$$

The hazard $h(t|\mathbf{x})$ is then defined by

$$h(t|\mathbf{x}) = \frac{f(t|\mathbf{x})}{S(t|\mathbf{x})}$$

Another useful expression for $S(t|\mathbf{x})$ in terms of $h(t|\mathbf{x})$ is

$$S(t|\mathbf{x}) = \exp\left(-\int_0^t h(u|\mathbf{x})du\right)$$

Thus,

$$\ln S(t|\mathbf{x}) = -\int_0^t h(u|\mathbf{x})du$$

For some purposes, it is also useful to define the cumulative hazard function

$$H(t|\mathbf{x}) = \int_0^t h(u|\mathbf{x})du = -\ln S(t|\mathbf{x})$$

Under the proportional hazard assumption, the survivor function can be written as

$$S(t|\mathbf{x}) = [S_0(t)]^{\exp\left(\mathbf{x}'\beta\right)}$$

where $S_0(t)$ is the baseline survivor function defined by

$$S_0(t) = \exp\left(-H_0(t)\right)$$

and

$$H_0(t) = \int_0^t h_0(u)du$$

Some relationships between $S(t|\mathbf{x})$, $H(t|\mathbf{x})$ and $H_0(t)$, $S_0(t)$ and $h_0(t)$ which will be used later are

$$\ln S(t|\mathbf{x}) = -H(t|\mathbf{x}) = -\exp\left(\mathbf{x}'\beta\right)H_0(t)$$

$$\ln\left(-\ln S(t|\mathbf{x})\right) = \mathbf{x}'\beta + \ln H_0(t)$$

To estimate the survivor function $S(t|\mathbf{x})$, we can see from the equation for the survivor function that there are two components, β and $S_0(t)$, which need to be estimated. The approach we use here is to estimate β from the partial likelihood function and then to maximize the full likelihood for $S_0(t)$.

Estimation of Beta

Assume that

- There are *m* levels for the stratification variable.
- Individuals in the same stratum have proportional hazard functions.
- The relative effect of the regressor variables is the same in each stratum.

Let $t_{j1} < \cdots < t_{jk_j}$ be the observed uncensored failure time of the n_j individuals in the *j*th stratum and $x_{j1}, \ldots, x_{jk_j}$ be the corresponding covariates. Then the partial likelihood function is defined by

$$L(\beta) = \prod_{j=1}^{m}\prod_{i=1}^{k_j} \frac{e^{\mathbf{S}'_{ji}\beta}}{\left(\sum_{l\in R_{ji}} w_l e^{\mathbf{x}'_l\beta}\right)^{d_{ji}}}$$

where d_{ji} is the sum of case weights of individuals whose lifetime is equal to t_{ji} and $\mathbf{S}_{ji}$ is the weighted sum of the regression vector **x** for those d_{ji} individuals, w_l is the case weight of individual *l*, and R_{ji} is the set of individuals alive and uncensored just prior to t_{ji} in the *j*th stratum. Thus the log-likelihood arising from the partial likelihood function is

$$l = \ln L(\beta) = \sum_{j=1}^{m}\sum_{i=1}^{k_j} \mathbf{S}'_{ji}\beta - \sum_{j=1}^{m}\sum_{i=1}^{k_j} d_{ji}\ln\left(\sum_{l\in R_{ji}} w_l e^{\mathbf{x}'_l\beta}\right)$$

and the first derivatives of *l* are

$$D_{\beta_r} = \frac{\partial l}{\partial \beta_r} = \sum_{j=1}^{m}\sum_{i=1}^{k_j}\left(S_{ji}^{(r)} - d_{ji}\frac{\sum\limits_{l\in R_{ji}} w_l x_{lr} e^{\mathbf{x}'_l\beta}}{\sum\limits_{l\in R_{ji}} w_l e^{\mathbf{x}'_l\beta}} \right), \quad r = 1,\dots,p$$

$S_{ji}^{(r)}$ is the *r*th component of $\mathbf{S}_{ji} = \left(S_{ji}^{(1)},\dots,S_{ji}^{(p)}\right)'$. The maximum partial likelihood estimate (MPLE) of β is obtained by setting $\frac{\partial l}{\partial \beta_r}$ equal to zero for $r = 1,\dots,p$, where *p* is the number of independent variables in the model. The equations $\frac{\partial l}{\partial \beta_r} = 0 \quad (r = 1,\dots,p)$ can usually be solved by using the Newton-Raphson method.

Note that from its equation that the partial likelihood function $L(\beta)$ is invariant under translation. All the covariates are centered by their corresponding overall mean. The overall mean of a covariate is defined as the sum of the product of weight and covariate for all the censored and uncensored cases in each stratum. For notational simplicity, $\mathbf{x}_l$ used in the Estimation Section denotes centered covariates.

Three convergence criteria for the Newton-Raphson method are available:

- Absolute value of the largest difference in parameter estimates between iterations (δ) divided by the value of the parameter estimate for the previous iteration; that is,

  $$\text{BCON} = \left|\frac{\delta}{\text{parameter estimate for previous iteration}}\right|$$

- Absolute difference of the log-likelihood function between iterations divided by the log-likelihood function for previous iteration.
- Maximum number of iterations.

The asymptotic covariance matrix for the MPLE $\hat{\beta} = \left(\hat{\beta}_1,\dots,\hat{\beta}_p\right)'$ is estimated by $\mathbf{I}^{-1}$ where **I** is the information matrix containing minus the second partial derivatives of L. The (*r*, *s*)-th element of **I** is defined by

$$\begin{aligned}\mathbf{I}_{rs} &= -E\frac{\partial^2}{\partial\beta_r\partial\beta_s}\ln L \\ &= \sum_{j=1}^{m}\sum_{i=1}^{k_j} d_{ji}\left[\frac{\sum\limits_{l\in R_{ji}} w_l x_{ls} x_{lr} e^{\mathbf{x}'_l\beta}}{\sum\limits_{l\in R_{ji}} w_l e^{\mathbf{x}'_l\beta}} - \frac{\left(\sum\limits_{l\in R_{ji}} w_l x_{lr} e^{\mathbf{x}'_l\beta}\right)\left(\sum\limits_{l\in R_{ji}} w_l x_{ls} e^{\mathbf{x}'_l\beta}\right)}{\left(\sum\limits_{l\in R_{ji}} w_l e^{\mathbf{x}'_l\beta}\right)^2}\right]\end{aligned}$$

We can also write **I** in a matrix form as

$$I_{rs} = \sum_{j=1}^{m}\sum_{i=1}^{k_j} d_{ji}\left(x'(t_{ji})\right)V(t_{ji})(x(t_{ji}))$$

where $\mathbf{x}(t_{ji})$ is a $n_{ji}\times p$ matrix which represents the *p* covariate variables in the model evaluated at time t_{ji}, n_{ji} is the number of distinct individuals in R_{ji}, and $\mathbf{V}(t_{ji})$ is a $n_{ji}\times n_{ji}$ matrix with the *l*th diagonal element $v_{ll}(t_{ji})$ defined by

$$v_{ll}(t_{ji}) = p_l(t_{ji})w_l - (w_l p_l(t_{ji}))^2$$

$$p_l(t_{ji}) = \frac{\exp\left(\mathbf{x}'_l\hat{\beta}\right)}{\sum_{h \in R_{ji}} w_h \exp\left(\mathbf{x}'_h\hat{\beta}\right)}$$

and the (*l*, *k*) element $v_{lk}(t_{ji})$ defined by

$$v_{lk}(t_{ji}) = w_l p_l(t_{ji}) \times w_k p_k(t_{ji})$$

Estimation of the Baseline Function

After the MPLE $\hat{\beta}$ of β is found, the baseline survivor function $S_{0j}(t)$ is estimated separately for each stratum. Assume that, for a stratum, $t_1 < \cdots < t_k$ are observed lifetimes in the sample. There are n_i at risk and d_i deaths at t_i, and in the interval $[t_{i-1}, t_i)$ there are λ_i censored times. Since $S_0(t)$ is a survivor function, it is non-increasing and left continuous, and thus $\hat{S}_0(t)$ must be constant except for jumps at the observed lifetimes $t_1, \ldots, t_k$.

Further, it follows that

$$\hat{S}_0(t_1) = 1$$

and

$$\hat{S}_0(t_i+) = \hat{S}_0(t_{i+1})$$

Writing $\hat{S}_0(t_i+) = p_i (i = 1, \ldots, k)$, the observed likelihood function is of the form

$$L_1 = \prod_{i=1}^{k} \left\{ \prod_{l \in D_i} \left(p_{i-1}^{\exp\left(\mathbf{x}'_l\beta\right)} - p_i^{\exp\left(\mathbf{x}'_l\beta\right)} \right)^{w_l} \prod_{l \in C_i} \left(p_{i-1}^{\exp\left(\mathbf{x}'_l\beta\right)} \right)^{w_l} \right\} \prod_{l \in C_{k+1}} \left(p_k^{\exp\left(\mathbf{x}'_l\beta\right)} \right)^{w_l}$$

where D_i is the set of individuals dying at t_i and C_i is the set of individuals with censored times in $[t_{i-1}, t_i)$. (Note that if the last observation is uncensored, C_{k+1} is empty and $p_k = 0$)

If we let $\alpha_i = p_i / p_{i-1} (i = 1, \ldots, k)$, L_1 can be written as

$$L_1 = \prod_{i=1}^{k} \prod_{l \in D_i} \left(1 - \alpha_i^{\exp\left(\mathbf{x}'_l\beta\right)} \right)^{w_l} \prod_{l \in R_i - D_i} \alpha_i^{w_l \exp\left(\mathbf{x}'_l\beta\right)}$$

Differentiating $\ln L_1$ with respect to $\alpha_1, \ldots, \alpha_k$ and setting the equations equal to zero, we get

$$\sum_{l \in D_i} \frac{w_l \exp\left(\mathbf{x}'_l\beta\right)}{1 - \alpha_i^{\exp\left(\mathbf{x}'_l\beta\right)}} = \sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l\beta\right) \quad i = 1, \ldots, k$$

We then plug the MPLE $\hat{\beta}$ of β into this equation and solve these *k* equations separately.

There are two things worth noting:

- If any $|D_i| = 1$, $\hat{\alpha}_i$ can be solved explicitly.

$$\hat{\alpha}_i = \left[1 - \frac{w_i \exp\left(\mathbf{x}'_i\hat{\beta}\right)}{\sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l\hat{\beta}\right)}\right]^{\exp\left(-\mathbf{x}'_l\overline{\beta}\right)}$$

- If $|D_i| > 1$, the equation for the cumulative hazard function must be solved iteratively for $\hat{\alpha}_i$. A good initial value for $\hat{\alpha}_i$ is

  $$\hat{\alpha}_i = \exp\left(\frac{-d_i}{\sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l\beta\right)}\right)$$

  where $d_i = \sum_{l \in D_i} w_l$ is the weight sum for set D_i. (See Lawless, 1982, p. 361.)

Once the $\hat{\alpha}_i$, $i = 1, \ldots, k$ are found, $S_0(t)$ is estimated by

$$\hat{S}_0(t) = \prod_{i:(t_i<t)} \hat{\alpha}_i$$

Since the above estimate of $S_0(t)$ requires some iterative calculations when ties exist, Breslow (1974) suggests using the equation for α_i when $|D_i| > 1$ as an estimate; however, we will use this as an initial estimate.

The asymptotic variance for $-\ln \hat{S}_0(t)$ can be found in Chapter 4 of Kalbfleisch and Prentice (1980). At a specified time *t*, it is consistently estimated by

$$var\left(-\ln \hat{S}_0(t)\right) = \sum_{t_i<t} |D_i| \left(\sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l\hat{\beta}\right)\right)^{-2} + \mathbf{a}'\mathbf{I}^{-1}a$$

where a is a $p \times 1$ vector with the *j*th element defined by

$$\sum_{t_i<t} |D_i| \frac{\sum_{l \in R_i} w_l x_{lj} \exp\left(\mathbf{x}'_l\hat{\beta}\right)}{\left(\sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l\hat{\beta}\right)\right)^2}$$

and **I** is the information matrix. The asymptotic variance of $\hat{S}(t|x)$ is estimated by

$$e^{2\mathbf{x}'\hat{\beta}}\left(\hat{S}(t|\mathbf{x})\right)^2 var\left(-\ln \hat{S}_0(t)\right)$$

Selection Statistics for Stepwise Methods

The same methods for variable selection are offered as in binary logistic regression. For more information, see the topic "Stepwise Variable Selection". Here we will only define the three removal statistics—Wald, LR, and Conditional—and the Score entry statistic.

Score Statistic

The score statistic is calculated for every variable not in the model to decide which variable should be added to the model. First we compute the information matrix **I** for all eligible variables based on the parameter estimates for the variables in the model and zero parameter estimates for the variables not in the model. Then we partition the resulting **I** into four submatrices as follows:

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{bmatrix}$$

where $\mathbf{A}_{11}$ and $\mathbf{A}_{22}$ are square matrices for variables in the model and variables not in the model, respectively, and $\mathbf{A}_{12}$ is the cross-product matrix for variables in and out. The score statistic for variable $\mathbf{x}_i$ is defined by

$$\mathbf{D}'_{x_i}\mathbf{B}_{22,i}\mathbf{D}_{x_i}$$

where $\mathbf{D}_{x_i}$ is the first derivative of the log-likelihood with respect to all the parameters associated with $\mathbf{x}_i$ and $\mathbf{B}_{22,i}$ is equal to $\left(\mathbf{A}_{22,i} - \mathbf{A}_{21,i}\mathbf{A}_{11}^{-1}\mathbf{A}_{12,i}\right)^{-1}$, and $\mathbf{A}_{22,i}$ and $\mathbf{A}_{12,i}$ are the submatrices in $\mathbf{A}_{22}$ and $\mathbf{A}_{12}$ associated with variable $\mathbf{x}_i$.

Wald Statistic

The Wald statistic is calculated for the variables in the model to select variables for removal. The Wald statistic for variable $\mathbf{x}_j$ is defined by

$$\hat{\beta}'_j\mathbf{B}_{11,j}\hat{\beta}_j$$

where $\hat{\beta}_j$ is the parameter estimate associated with $\mathbf{x}_j$ and $\mathbf{B}_{11,j}$ is the submatrix of $\mathbf{A}_{11}^{-1}$ associated with $\mathbf{x}_j$.

LR (Likelihood Ratio) Statistic

The LR statistic is defined as twice the log of the ratio of the likelihood functions of two models evaluated at their own MPLES. Assume that *r* variables are in the current model and let us call the current model the full model. Based on the MPLES of parameters for the full model, *l(full)* is defined in “Estimation of Beta ”. For each of *r* variables deleted from the full model, MPLES are found and the reduced log-likelihood function, *l(reduced)*, is calculated. Then LR statistic is defined as

$$-2(l(reduced) - l(full))$$

Conditional Statistic

The conditional statistic is also computed for every variable in the model. The formula for conditional statistic is the same as LR statistic except that the parameter estimates for each reduced model are conditional estimates, not MPLES. The conditional estimates are defined as follows. Let $\hat{\beta} = \left(\hat{\beta}_1, \ldots, \hat{\beta}_r\right)'$ be the MPLES for the *r* variables (blocks) and *C* be the asymptotic covariance for the parameters left in the model given $\hat{\beta}_i$ is

$$\tilde{\beta}_{(i)} = \hat{\beta}_{(i)} - \mathbf{C}_{12}^{(i)}\left(\mathbf{C}_{22}^{(i)}\right)^{-1}\hat{\beta}_i$$

where $\hat{\beta}_i$ is the MPLE for the parameter(s) associated with $\mathbf{x}_i$ and $\hat{\beta}_{(i)}$ is $\hat{\beta}$ without $\hat{\beta}_i$, $\mathbf{C}_{12}^{(i)}$ is the covariance between the parameter estimates left in the model $\hat{\beta}_{(i)}$ and $\hat{\beta}_i$, and $\mathbf{C}_{22}^{(i)}$ is the covariance of $\hat{\beta}_i$. Then the conditional statistic for variable $\mathbf{x}_i$ is defined by

$$-2\left(l\left(\mathrm{b}_{(i)}\right) - l(full)\right)$$

where $l\left(\tilde{\beta}_{(i)}\right)$ is the log-likelihood function evaluated at $\tilde{\beta}_{(i)}$.

Note that all these four statistics have a chi-square distribution with degrees of freedom equal to the number of parameters the corresponding model has.

Statistics

The following output statistics are available.

Initial Model Information

The initial model for the first method is for a model that does not include covariates. The log-likelihood function *l* is equal to

$$l(0) = -\sum_{j=1}^{m}\sum_{i=1}^{k_j} d_{ji} \ln\left(n_{ji}^*\right)$$

where n_{ji}^* is the sum of weights of individuals in set R_{ji}.

Model Information

When a stepwise method is requested, at each step, the −2 log-likelihood function and three chi-square statistics (model chi-square, improvement chi-square, and overall chi-square) and their corresponding degrees of freedom and significance are printed.

–2 Log-Likelihood

$$-2\sum_{j=1}^{m}\sum_{i=1}^{k_j}\left(\mathbf{s}'_{ji}\hat{\beta} - d_{ji}\ln\left(\sum_{l\in R_{ji}} w_l \exp\left(\mathbf{x}'_l\hat{\beta}\right)\right)\right)$$

where $\hat{\beta}$ is the MPLE of β for the current model.

Improvement Chi-Square

(–2 log-likelihood function for previous model) – (–2 log-likelihood function for current model).

The previous model is the model from the last step. The degrees of freedom are equal to the absolute value of the difference between the number of parameters estimated in these two models.

Model Chi-Square

(–2 log-likelihood function for initial model) – (–2 log-likelihood function for current model).

The initial model is the final model from the previous method. The degrees of freedom are equal to the absolute value of the difference between the number of parameters estimated in these two model.

Note: The values of the model chi-square and improvement chi-square can be less than or equal to zero. If the degrees of freedom are equal to zero, the chi-square is not printed.

Overall Chi-Square

The overall chi-square statistic tests the hypothesis that all regression coefficients for the variables in the model are identically zero. This statistic is defined as

$$\mathbf{u}'(0)\mathbf{I}^{-1}\mathbf{u}(0)$$

where $\mathbf{u}(0)$ represents the vector of first derivatives of the partial log-likelihood function evaluated at $\beta = 0$. The elements of **u** and **I** are defined in “Estimation of Beta ”.

Information for Variables in the Equation

For each of the single variables in the equation, MPLE, SE for MPLE, Wald statistic, and its corresponding *df*, significance, and partial *R* are given. For a single variable, *R* is defined by

$$R = \left[\frac{\text{Wald}_{-2}}{-2\text{ log-likelihood for the intial model}}\right]^{1/2} \times \text{sign of MPLE}$$

if Wald > 2. Otherwise *R* is set to zero. For a multiple category variable, only the Wald statistic, *df*, significance, and partial *R* are printed, where *R* is defined by

$$R = \left[\frac{\text{Wald}_{-2*}\text{df}}{-2\text{ log-likelihood for the intial model}}\right]^{1/2}$$

if Wald > 2df. Otherwise *R* is set to zero.

Information for the Variables Not in the Equation

For each of the variables not in the equation, the Score statistic is calculated and its corresponding degrees of freedom, significance, and partial *R* are printed. The partial *R* for variables not in the equation is defined similarly to the *R* for the variables in the equation by changing the Wald statistic to the Score statistic.

There is one overall statistic called the residual chi-square. This statistic tests if all regression coefficients for the variables not in the equation are zero. It is defined by

$$\mathbf{u}'\left(\hat{\beta}\right)\mathbf{B}_{22}\mathbf{u}\left(\hat{\beta}\right)$$

where $\mathbf{u}\left(\hat{\beta}\right)$ is the vector of first derivatives of the partial log-likelihood function with respect to all the parameters not in the equation evaluated at MPLE $\hat{\beta}$ and $\mathbf{B}_{22}$ is equal to $\left(\mathbf{A}_{22}-\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}\right)^{-1}$ and **A** is defined in “Score Statistic ”.

Survival Table

For each stratum, the estimates of the baseline cumulative survival (S_0) and hazard (H_0) function and their standard errors are computed. $H_0(t)$ is estimated by

$$\hat{H}_0(t) = -\ln \hat{S}_0(t)$$

and the asymptotic variance of $\hat{H}_0(t)$ is defined in “Estimation of the Baseline Function ”. Finally, the cumulative hazard function $H(t|\mathbf{x})$ and survival function $S(t|\mathbf{x})$ are estimated by

$$\hat{H}(t|\mathbf{x}) = \exp\left(\mathbf{x}'\hat{\beta}\right)\hat{H}_0(t)$$

and, for a given **x**,

$$\hat{S}(t|\mathbf{x}) = \left[\hat{S}_0(t)\right]^{\exp\left(\mathbf{x}'\hat{\beta}\right)}$$

The asymptotic variances are

$$var\left(\hat{H}(t|\mathbf{x})\right) = \exp\left(2\mathbf{x}'\hat{\beta}\right)var\left(\hat{H}_0(t)\right)$$

and

$$var\left(\hat{S}(t|\mathbf{x})\right) = \exp\left(2\mathbf{x}'\hat{\beta}\right)\left(\hat{S}(t|\mathbf{x})\right)^2 var\left(\hat{H}_0(t)\right)$$

Diagnostic Statistics

Three casewise diagnostic statistics, Residual, Partial Residual, and DFBETAs, are produced. Both Residual and DFBETA are computed for all distinct individuals. Partial Residuals are calculated only for uncensored individuals.

Assume that there are n_j subjects in stratum *j* and k_j distinct observed events $t_1 < \cdots < t_{k_j}$. Define the selected probability for the *l*th individual at time t_i as

$$p_l(t_i) = \begin{cases} \dfrac{\exp\left(\mathbf{x}'_l(t_i)\hat{\beta}\right)}{\sum_{h \in R_i} w_h \exp\left(\mathbf{x}'_h(t_i)\hat{\beta}\right)} & \text{if } l\text{th individual is in } R_i \\ 0 & \text{otherwise} \end{cases}$$

and

$$u_l = \sum_{i=1}^{k_j} d_i\left[p_l(t_i) - p_l^2(t_i)\right]$$

$$y_l(t_i) = \begin{cases} 1 & \text{if } l\text{th individual is in } D_i \\ 0 & \text{otherwise} \end{cases}$$

$$r_l = \sum_{i=1}^{k_j} \left[y_l(t_i) - d_i p_l(t_i)\right]$$

DFBETA

The changes in the maximum partial likelihood estimate of beta due to the deletion of a single observation have been discussed in Cain and Lange (1984) and Storer and Crowley (1985). The estimate of DFBETA computed is derived from augmented regression models. The details can be found in Storer and Crowley (1985). When the *l*th individual in the *j*th stratum is deleted, the change $\mathbf{\Delta}\beta_l$ is estimated by

$$\mathbf{\Delta}\beta_l = -\frac{\mathbf{1}}{\mathbf{m}}\mathbf{I}^{-1} v_l r_l$$

where

$$w = diag\left(w_1, \ldots, w_{n_{ji}}\right)$$

$$v_l = \sum_{i=1}^{k_j} d_i p_l(t_i)\left(\mathbf{x}_l(t_i) - \mathbf{x}(t_i)\mathbf{w}\mathbf{p}(t_i)\right)$$

$$\mathbf{p}(t_i) = \left(p_1(t_i), \ldots, p_{n_{ji}}(t_i)\right)'$$

$$m_l = u_l - v_l \mathbf{I}^{-1} v_l$$

and $\mathbf{x}'(t_i)$ is an $n_{ji} \times p$ matrix which represents the *p* covariate variables in the model evaluated at t_i, and n_{ji} is the number of individuals in R_{ji}.

Partial Residuals

Partial residuals can only be computed for the covariates which are not time dependent. At time t_i in stratum *j*, x_g is the *p*×1 observed covariate vector for any *g*th individual in set D_i, where D_i is the set of individuals dying at t_i. The partial residual γ_g is defined by

$$\gamma_g = \begin{pmatrix} \gamma_{g1} \\ \cdots \\ \gamma_{gp} \end{pmatrix} = \mathbf{x}_g - p(t_i)\mathbf{x}$$

Rewriting the above formula in a univariate form, we get

$$\gamma_{gh} = x_{gh} - \frac{\sum_{l \in R_i} w_l x_{lh} \exp\left(\mathbf{x}'_l \hat{\beta}\right)}{\sum_{l \in R_i} w_l \exp\left(\mathbf{x}'_l \hat{\beta}\right)}, \quad h = 1, \ldots, p. g \in D_i$$

where x_{gh} is the *h*th component for x_g. For every variable, the residuals can be plotted against times to test the proportional hazards assumption.

Residuals

The residuals e_i are computed by

$$e_i = \hat{H}(t_i|\mathbf{x}_i) = \exp\left(\mathbf{x}'_i \hat{\beta}\right)\left(\hat{H}_0(t_i)\right)$$

which is the same as the estimate of the cumulative hazard function.

Plots

For a specified pattern, the covariate values $\mathbf{x}_c$ are determined and $\mathbf{x}_c$ is computed. There are three plots available for Cox regression.

Survival Plot

For stratum j, $\left(t_i, \hat{S}_0(t_i|\mathbf{x}_c)\right)$, $i = 1, \ldots, k_j$ are plotted where

$$\hat{S}(t_i|\mathbf{x}_c) = \left(\hat{S}_0(t_i)\right)^{\exp\left(\mathbf{x}'_c \hat{\beta}\right)}$$

When PATTERN(ALL) is requested, for every uncensored time t_i in stratum *j* the survival function is estimated by

$$\hat{S}(t_i) = \frac{\sum_{l=1}^{k_j} w_l \hat{S}(t_i|\mathbf{x}_c)}{\sum_{l=1}^{k_j} w_l} = \frac{\sum_{l=1}^{k_j} w_l \left(\hat{S}_0(t_i)\right)^{\exp\left(\mathbf{x}'_c \hat{\beta}\right)}}{\sum_{l=1}^{k_j} w_l}$$

Then $\left(t_i, \hat{S}(t_i)\right)$, $i = 1, \ldots, k_j$ are plotted for stratum *j*.

Hazard Plot

For stratum *j*, $\left(t_i, \hat{H}(t_i|\mathbf{x}_c)\right)$, $i = 1, \ldots, k_j$ are plotted where

$$\hat{H}(t_i|\mathbf{x}_c) = \exp\left(\mathbf{x}'_c \hat{\beta}\right)\hat{H}_0(t_i)$$

LML Plot

The log-minus-log plot is used to see whether the stratification variable should be included as a covariate. For stratum *j*, $\left(t_i, \mathbf{x}'_c\hat{\beta} + \ln \hat{H}_0(t_i)\right)$, $i = 1, \ldots, k_j$ are plotted. If the plot shows parallelism among strata, then the stratum variable should be a covariate.

References

Breslow, N. E. 1974. Covariance analysis of censored survival data. *Biometrics*, 30, 89–99.

Cain, K. C., and N. T. Lange. 1984. Approximate case influence for the proportional hazards regression model with censored data. *Biometrics*, 40, 493–499.

Cox, D. R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 34, 187–220.

Kalbfleisch, J. D., and R. L. Prentice. 2002. *The statistical analysis of failure time data*, 2 ed. New York: John Wiley & Sons, Inc.

Lawless, R. F. 1982. *Statistical models and methods for lifetime data*. New York: John Wiley & Sons, Inc..

Storer, B. E., and J. Crowley. 1985. A diagnostic for Cox regression and general conditional likelihoods. *Journal of the American Statistical Association*, 80, 139–147.

CREATE Algorithms

CREATE produces new series as a function of existing series.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 22-1
Notation

Notation	**Description**
Existing Series	$X_1, \ldots, X_n$
New Series	$Y_1, \ldots, Y_n$

Cumulative Sum (CSUM(X))

$$Y_j = \sum_{i=1}^{j} X_i \quad j = 1, \ldots, n$$

Differences of Order m (DIFF(X,m))

Define

$$Z_j(k) = Z_j(k-1) - Z_{j-1}(k-1) \quad k = 1, \ldots, m \quad j = k+1, \ldots, n$$

with

$$Z_j(0) = X_j \quad j = 1, \ldots, n$$

then

$$Y_j = \begin{cases} Z_j(m) & j = m+1, \ldots, n \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Lag of Order m (LAG(X,m))

$$Y_j = \begin{cases} X_{j-m} & j = m+1, \ldots, n \\ \text{SYSMIS} & j = 1, \ldots, m \end{cases}$$

Lead of Order m (LEAD(X,m))

$$Y_j = \begin{cases} X_{j+m} & j = 1, \ldots, n-m \\ \text{SYSMIS} & j = n-m+1, \ldots, n \end{cases}$$

Moving Average of Length m (MA(X,m))

If *m* is odd, define

$$q = \frac{m-1}{2}$$

then

$$Y_j = \begin{cases} \sum_{k=-q}^{q} X_{j+k}/m & j = q+1,\ldots,n-q \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

If *m* is even, define $q = m/2$ and

$$Z_j = \sum_{k=-q+1}^{q} X_{j+k}/m \quad j = q,\ldots,n-q$$

then

$$Y_j = \begin{cases} (Z_{j-1} + Z_j)/2 & j = q+1,\ldots,n-q \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Running Median of Length m (X,m)

If *m* is odd,

$$q = \tfrac{m-1}{2}$$

$$Y_j = \begin{cases} \text{median}(X_{j-q}, X_{j-q+1},\ldots,X_j, X_{j+1},\ldots,X_{j+q}) & j = q+1,\ldots,n-q \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

If *m* is even, define

$$Z_j = \text{median}(X_{j-q+1},\ldots,X_j, X_{j+1},\ldots,X_{j+q}) \quad j = q,\ldots,n-q$$

then

$$Y_j = \begin{cases} (Z_{j-1} + Z_j)/2 & j = q+1,\ldots,n-q \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where

$$median(a_1,\ldots,a_k) = \begin{cases} a_{(l)} & \text{if } k \text{ is odd} \\ \left(a_{(l)} + a_{(l+1)}\right)/2 & \text{if } k \text{ is even} \end{cases}$$

$$l = \begin{cases} (k+1)/2 & \text{if } k \text{ is odd} \\ k/2 & \text{if } k \text{ is even} \end{cases}$$

and $a_{(1)} < a_{(2)} < \ldots < a_{(k)}$ is the ordered sample of $a_1, \ldots, a_k$.

Seasonal Differencing of Order m and Period p (SDIFF(X,m,p))

Define

$$Z_j(k) = Z_j(k-1) - Z_{j-p}(k-1) \quad k = 1, \ldots, m \quad j = pk+1, \ldots, n$$

where

$$Z_j(0) = X_j \quad j = 1, \ldots, n$$

then

$$Y_j = Z_j(m) \quad j = mp+1, \ldots, n$$

The T4253H Smoothing Function (T4253H(X))

The original series is smoothed by a compound data smoother based on (Velleman, 1980). The smoother starts with:

- A running median of 4:

Let *Z* be the smoothed series, then

$$Z_{j+1/2} = median(X_{j-1}, X_j, X_{j+1}, X_{j+2}) \quad j = 2, \ldots, n-2$$

and

$$Z^{(1)}_{0.5} = X_1 \quad Z^{(1)}_{1.5} = median(X_1, X_2) = \tfrac{1}{2}(X_1 + X_2)$$

$$Z^{(1)}_{n-1/2} = median(X_{n-1}, X_n) = \tfrac{1}{2}(X_{n-1} + X_n) \quad Z^{(1)}_{n+1/2} = X_n$$

- A running median of *Z*:

$$Z^{(1)}_1 = Z_{0.5} \quad Z^{(1)}_n = Z_{n+1/2}$$

and

$$Z^{(1)}_j = \tfrac{1}{2}\left(Z_{j-1/2} + Z_{j+1/2}\right) \quad j = 2, \ldots, n-1$$

- A running median of 5 on $Z^{(1)}_1, \ldots, Z^{(1)}_n$ from the previous step:

Let $Z^{(2)}$ be the resulting series, then

$$Z_1^{(2)} = Z_1^{(1)} \quad Z_n^{(2)} = Z_n^{(1)}$$
$$Z_2^{(2)} = median\left(Z_1^{(1)}, Z_2^{(1)}, Z_3^{(1)}\right)$$
$$Z_{n-1}^{(2)} = median\left(Z_{n-2}^{(1)}, Z_{n-1}^{(1)}, Z_n^{(1)}\right)$$

and

$$Z_j^{(2)} = median\left(Z_{j-2}^{(1)}, Z_{j-1}^{(1)}, Z_j^{(1)}, Z_{j+1}^{(1)}, Z_{j+2}^{(1)}\right) \quad j = 3, \ldots, n-2$$

- A running median of 3 on $Z_1^{(1)}, \ldots, Z_n^{(1)}$ from the previous step:

  Let $Z^{(3)}$ be the resulting series, then

$$Z_j^{(3)} = median\left(Z_{j-1}^{(2)}, Z_j^{(2)}, Z_{j+1}^{(2)}\right) \quad j = 2, 3, \ldots, n-1$$
$$Z_1^{(3)} = median\left(3Z_2^{(3)} - 2Z_3^{(3)}, Z_1^{(2)}, Z_2^{(3)}\right)$$
$$Z_n^{(3)} = median\left(3Z_{n-1}^{(3)} - 2Z_{n-2}^{(3)}, Z_n^{(2)}, Z_{n-1}^{(3)}\right)$$

- Hanning (Running Weighted Averages):

$$Z_j^{(4)} = \tfrac{1}{4}Z_{j-1}^{(3)} + \tfrac{1}{2}Z_j^{(3)} + \tfrac{1}{4}Z_{j+1}^{(3)} \quad j = 2, \ldots, n-1$$
$$Z_1^{(4)} = Z_1^{(3)}, \quad Z_n^{(4)} = Z_n^{(3)}$$

- Residual:

$$D_i = X_i - Z_i^{(4)} \quad i = 1, \ldots, n$$

- Repeat the previous steps on the residuals $D_1, \ldots, D_n$:
- Let $D_1^{(4)}, \ldots, D_n^{(4)}$ be the final result.
- Final smooth:

$$Y_i = Z_i^{(4)} + D_i^{(4)} \quad i = 1, \ldots, n$$

Prior Moving Averages of Length m (PMA(X,m))

$$Y_i = \begin{cases} \displaystyle\sum_{j=i-m}^{i-1} X_j/m & i = m+1, \ldots, n \\ \text{SYSMIS} & i = 1, \ldots, m \end{cases}$$

Fast Fourier Transform (FFT(X))

The discrete Fourier transform of a sequence $X = \{X_1, \ldots, X_n\}$ is defined as

$$\begin{aligned}
Y_k &= \frac{1}{n}\sum_{t=1}^{n} X_t \exp\{-i2\pi f_k(t-1)\} \\
&= \frac{1}{n}\sum_{t=1}^{n} X_t[\cos(2\pi f_k(t-1)) - i\sin(2\pi f_k(t-1))] \\
&\frac{1}{n}\sum_{t=1}^{n} = X_t\cos(2\pi f_k(t-1)) + i\left[-\frac{1}{n}\sum_{t=1}^{n} X_t \sin(2\pi f_k(t-1))\right] \\
&= a_k + ib_k
\end{aligned}$$

Thus *a*, *b* are two sequences generated by FFT and they are called real and imaginary, respectively.

$$a_k = \frac{1}{n}\sum_{t=1}^{n} X_t \cos(2\pi f_k(t-1)) \quad k = 1, \ldots, r$$

$$b_k = -\frac{1}{n}\sum_{t=1}^{n} X_t \sin(2\pi f_k(t-1)) \quad k = 1, \ldots, r$$

where

$$r = \begin{cases} (n-1)/2 & \text{if } n \text{ is odd} \\ n/2 & \text{if } n \text{ is even} \end{cases}$$

and

$$a_0 = \overline{X}$$

$$b_0 = -\frac{1}{n}\sum_{t=1}^{n} X_t \cos(\pi(t-1))$$

Inverse Fast Fourier Transform of Two Series (IFFT(a,b))

The inverse Fourier Transform of two series $\{a, b\}$ is defined as

$$X_t = a_0 - b_0\cos(\pi(t-1)) + 2\left[\sum_{k=1}^{q} a_k \cos(2\pi f_k(t-1)) - \sum_{k=1}^{q} b_k \sin(2\pi f_k(t-1))\right]$$

References

Brigham, E. O. 1974. *The fast Fourier transform*. Englewood Cliffs, N.J.: Prentice-Hall.

Monro, D. M. 1975. Algorithm AS 83: Complex discrete fast Fourier transform. *Applied Statistics*, 24, 153–160.

Monro, D. M., and J. L. Branch. 1977. Algorithm AS 117: The Chirp discrete Fourier transform of general length. *Applied Statistics*, 26, 351–361.

Velleman, P. F. 1980. Definition and comparison of robust nonlinear data smoothing algorithms. *Journal of the American Statistical Association*, 75, 609–615.

Velleman, P. F., and D. C. Hoaglin. 1981. *Applications, basics, and computing of exploratory data analysis*. Boston, Mass.: Duxbury Press.

CROSSTABS Algorithms

The notation and statistics refer to bivariate subtables defined by a row variable *X* and a column variable *Y*, unless specified otherwise. By default, CROSSTABS deletes cases with missing values on a table-by-table basis.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 23-1
Notation

Notation	Description
X_i	Distinct values of row variable arranged in ascending order: $X_1 < X_2 < \cdots < X_R$
Y_j	Distinct values of column variable arranged in ascending order: $Y_1 < Y_2 < \cdots < Y_C$
f_{ij}	Sum of cell weights for cases in cell (i,j)
c_j	$\sum_{i=1}^{R} f_{ij}$, the *j*th column subtotal
r_i	$\sum_{j=1}^{C} f_{ij}$, the *i*th row subtotal
W	$\sum_{j=1}^{C} c_j = \sum_{i=1}^{R} r_i$, the grand total

Marginal and Cell Statistics

Count

$$\text{count} = f_{ij}$$

Expected Count

$$E_{ij} = \frac{r_i c_j}{W}$$

Row Percent

$$\text{row percent} = 100 \times (f_{ij}/r_i)$$

Column Percent

$$\text{column percent} = 100 \times (f_{ij}/c_j)$$

Total Percent

$$\text{total percent} = 100 \times (f_{ij}/W)$$

Residual

$$R_{ij} = f_{ij} - E_{ij}$$

Standardized Residual

$$SR_{ij} = \frac{R_{ij}}{\sqrt{E_{ij}}}$$

Adjusted Residual

$$AR_{ij} = \frac{R_{ij}}{\sqrt{E_{ij}\left(1 - \frac{r_i}{W}\right)\left(1 - \frac{c_j}{W}\right)}}$$

Chi-Square Statistics

Pearson’s Chi-Square

$$\chi_p^2 = \sum_{ij} \frac{(f_{ij} - E_{ij})^2}{E_{ij}}$$

The degrees of freedom are $(R-1)(C-1)$.

Likelihood Ratio

$$\chi_{LR}^2 = 2\sum_{ij} f_{ij} \ln\left(f_{ij}/E_{ij}\right)$$

The degrees of freedom are $(R-1)(C-1)$.

Note: when $f_{ij} = 0$, the entire term $f_{ij} \ln\left(f_{ij}/E_{ij}\right)$ is treated as 0, because $\underset{n \to 0}{lim}\, n\log(n) = 0$, and thus has no effect on the sum.

Fisher’s Exact Test

If the table is a 2×2 table, not resulting from a larger table with missing cells, with at least one expected cell count less than 5, then the Fisher exact test is calculated. For more information, see the topic “Significance Levels for Fisher’s Exact Test”.

Yates Continuity Corrected for 2 x 2 Tables

$$\chi_c^2 = \begin{cases} \frac{W(|f_{11}f_{22} - f_{12}f_{21}| - 0.5W)^2}{r_1 r_2 c_1 c_2} & \text{if} |f_{11}f_{22} - f_{12}f_{21}| > 0.5W \\ 0 & \text{otherwise} \end{cases}$$

The degrees of freedom are 1.

Mantel-Haenszel Test of Linear Association

$$\chi^2_{MH} = (W-1)r^2$$

where *r* is the Pearson correlation coefficient to be defined later. The degrees of freedom are 1.

Other Measures of Association

Phi Coefficient

For a table not 2×2

$$\varphi = \sqrt{\frac{\chi^2_p}{W}}$$

For a 2×2 table only, φ is equal to the Pearson correlation coefficient so that the sign of φ matches that of the correlation coefficients.

Coefficient of Contingency

$$CC = \left(\frac{\chi^2_p}{\chi^2_p + W} \right)^{1/2}$$

Cramér's V

$$V = \left(\frac{\chi^2_p}{W(q-1)} \right)^{1/2}$$

where $q = \min\{R, C\}$.

Measures of Proportional Reduction in Predictive Error

Lambda

Let f_{im} and f_{mj} be the largest cell count in row *i* and column *j*, respectively. Also, let r_m be the largest row subtotal and c_m the largest column subtotal. Define $\lambda_{Y|X}$ as the proportion of relative error in predicting an individual's *Y* category that can be eliminated by knowledge of the *X* category. $\lambda_{Y|X}$ is computed as

$$\lambda_{Y|X} = \frac{\sum_{i=1}^{R} f_{im} - c_m}{W - c_m}$$

The standard errors are

$$ASE_0 = \frac{\sqrt{\sum_{i=1}^{R}\sum_{j=1}^{C} f_{ij}(\delta_{ij}-\delta_j)^2 - \left(\sum_{i=1}^{R} f_{im} - c_m\right)^2 /W}}{W-c_m}$$

$$ASE_1 = \frac{\sqrt{\sum_{i=1}^{R}\sum_{j=1}^{C} f_{ij}(\delta_{ij}-\delta_j+\lambda_{Y|X}\delta_j)^2 - W\lambda_{(Y|X)}^2}}{W-c_m}$$

where

$$\delta_{ij} = \begin{cases} 1 & \text{if } j \text{ is column index for } f_{im} \\ 0 & \text{otherwise} \end{cases}$$

$$\delta_j = \begin{cases} 1 & \text{if } j \text{ is index for } c_m \\ 0 & \text{otherwise} \end{cases}$$

Lambda for predicting *X* from *Y*, $\lambda_{Y|X}$, is obtained by permuting the indices in the above formulae.

The two asymmetric lambdas are averaged to obtain the symmetric lambda.

$$\lambda = \frac{\sum_{i=1}^{R} f_{im} + \sum_{j=1}^{C} f_{mj} - c_m - r_m}{2W - r_m - c_m}$$

The standard errors are

$$ASE_0 = \frac{\sqrt{\sum_{i=1}^{R}\sum_{j=1}^{C} f_{ij}\left(\delta_{ij}^r + \delta_{ij}^c - \delta_i^r - \delta_j^c\right)^2 - \left[\left(\sum_{i=1}^{R} f_{im} + \sum_{j=1}^{C} f_{mj} - c_m - r_m\right)^2 /W\right]}}{2W-r_m-c_m}$$

$$ASE_1 = \frac{\sqrt{\sum_{i=1}^{R}\sum_{j=1}^{C} f_{ij}\left[\delta_{ij}^r + \delta_{ij}^c - \delta_i^r - \delta_j^c + \lambda\left(\delta_i^r + \delta_j^c\right)\right]^2 - 4W\lambda^2}}{2W-r_m-c_m}$$

where

$$\delta_{ij}^r = \begin{cases} 1 & \text{if } i \text{ is row index for } f_{mj} \\ 0 & \text{otherwise} \end{cases}$$

$$\delta_i^r = \begin{cases} 1 & \text{if } i \text{ is index for } r_m \\ 0 & \text{otherwise} \end{cases}$$

and where

$$\delta_{ij}^{c} = \begin{cases} 1 & \text{if } j \text{ is column index for } f_{im} \\ 0 & \text{otherwise} \end{cases}$$

$$\delta_{i}^{c} = \begin{cases} 1 & \text{if } j \text{ is index for } c_{m} \\ 0 & \text{otherwise} \end{cases}$$

Goodman and Kruskal’s Tau

Similarly defined is Goodman and Kruskal’s tau (τ)(Goodman and Kruskal, 1954):

$$\tau_{Y|X} = \frac{W\sum_{i,j}\left(f_{ij}^2/r_i\right) - \sum_{j=1}^{C} c_j^2}{W^2 - \sum_{j=1}^{C} c_j^2}$$

with standard error

$$ASE_1 = \sqrt{\frac{4}{\delta^4}\sum_{i,j} f_{ij}\left\{(v-\delta)\left(\frac{1}{r_i}\sum_{j=1}^{C} f_{ij}c_j - c_j\right) - W\delta\left(\frac{1}{r_i^2}\sum_{j=1}^{C} f_{ij}^2 - \frac{1}{r_i}f_{ij}\right)\right\}^2}$$

in which

$$\delta = W^2 - \sum_{j=1}^{C} c_j^2 \quad \text{and} \quad v = W\sum_{i,j} f_{ij}^2/r_i - \sum_{j=1}^{C} c_j^2$$

$\tau_{X|Y}$ and its standard error can be obtained by interchanging the roles of *X* and *Y*.

The significance level is based on the chi-square distribution, since

$$(W-1)(C-1)\tau_{Y|X} \sim \chi^2_{(R-1)(C-1)}$$

$$(W-1)(R-1)\tau_{X|Y} \sim \chi^2_{(R-1)(C-1)}$$

Uncertainty Coefficient

Let $U_{Y|X}$ be the proportional reduction in the uncertainty (entropy) of *Y* that can be eliminated by knowledge of *X*. It is computed as

$$U_{Y|X} = \frac{U(X) + U(Y) - U(XY)}{U(Y)}$$

where

$$U(X) = -\sum_{i=1}^{R} \frac{r_i}{W} \ln\left(\frac{r_i}{W}\right)$$

$$U(Y) = -\sum_{j=1}^{C} \frac{c_j}{W} \ln\left(\frac{c_j}{W}\right)$$

and

$$U(XY) = -\sum_{i,j} \frac{f_{ij}}{W} \ln\left(\frac{f_{ij}}{W}\right), \quad \text{for } f_{ij} > 0$$

The asymptotic standard errors are

$$ASE_1 = \frac{1}{WU(Y)^2}\sqrt{\sum_{i,j} f_{ij}\left\{U(Y)\ln\left(\frac{f_{ij}}{r_i}\right) + [U(X) - U(XY)]\ln\left(\frac{c_j}{W}\right)\right\}^2}$$

$$ASE_0 = \frac{\sqrt{P - W[U(X) + U(Y) - U(XY)]^2}}{[WU(Y)]}$$

where

$$P = \sum_{i,j} f_{ij}\left[\ln\left(\frac{c_j r_i}{W f_{ij}}\right)\right]^2$$

The formulas for $U_{X|Y}$ can be obtained by interchanging the roles of *X* and *Y*.

A symmetric version of the two asymmetric uncertainty coefficients is defined as follows:

$$U = 2\left[\frac{U(X) + U(Y) - U(XY)}{U(X) + U(Y)}\right]$$

with asymptotic standard errors

$$ASE_1 = \frac{2}{W[U(X) + U(Y)]^2}\sqrt{\sum_{i,j} f_{ij}\left\{U(XY)\ln\left(\frac{r_i c_j}{W^2}\right) - [U(X) + U(Y)]\ln\left(\frac{f_{ij}}{W}\right)\right\}^2}$$

or

$$ASE_0 = \frac{2}{W[U(X) + U(Y)]}\sqrt{P - W[U(X) + U(Y) - U(XY)]^2}$$

Cohen's Kappa

Cohen's kappa (κ), defined only for square table $(R = C)$, is computed as

$$\kappa = \frac{W\sum_{i=1}^{R} f_{ii} - \sum_{i=1}^{R} r_i c_i}{W^2 - \sum_{i=1}^{R} r_i c_i}$$

with variance

$$var_1 = W\left\{ \frac{(\Sigma f_{ii})(W - \Sigma f_{ii})}{(W^2 - \Sigma r_i c_i)^2} + \frac{2(W - \Sigma f_{ii})\left(2\Sigma f_{ii}\Sigma r_i c_i - W\Sigma f_{ii}(r_i + c_i)\right)}{(W^2 - \Sigma r_i c_i)^3} + \frac{(W - \Sigma f_{ii})^2\left[W\sum_{i,j} f_{ij}(r_j + c_i)^2 - 4(\Sigma r_i c_i)^2\right]}{(W^2 - \Sigma r_i c_i)^4} \right\}$$

$$var_0 = \frac{1}{W\left(W^2 - \sum_i r_i c_i\right)^2}\left[W^2\left(\sum_i r_i c_i\right) + \left(\sum_i r_i c_i\right)^2 - W\left(\sum_i r_i c_i(r_i + c_i)\right)\right]$$

Kendall’s Tau-b and Tau-c

Define

$$D_r = W^2 - \sum_{i=1}^{R} r_i^2$$

$$D_c = W^2 - \sum_{j=1}^{C} c_j^2$$

$$C_{ij} = \sum_{h<i}\sum_{k<j} f_{hk} + \sum_{h>i}\sum_{k>j} f_{hk}$$

$$D_{ij} = \sum_{h<i}\sum_{k>j} f_{hk} + \sum_{h>i}\sum_{k<j} f_{hk}$$

$$P = \sum_{i,j} f_{ij} C_{ij}$$

$$Q = \sum_{i,j} f_{ij} D_{ij}$$

Note: the *P* and *Q* listed above are double the “usual” *P* (number of concordant pairs) and *Q* (number of discordant pairs). Likewise, D_r is double the “usual” $P+Q+X_0$ (the number of concordant pairs, discordant pairs, and pairs on which the row variable is tied) and D_c is double the “usual” $P+Q+Y_0$ (the number of concordant pairs, discordant pairs, and pairs on which the column variable is tied).

Kendall’s Tau-b

$$\tau_b = \frac{P-Q}{\sqrt{D_r D_c}}$$

with standard error

$$ASE_1 = \tfrac{1}{(D_r D_c)} \sqrt{\sum_{i,j} f_{ij}\left(2\sqrt{D_r D_c}(C_{ij}-D_{ij}) + \tau_b v_{ij}\right)^2 - W^3 \tau_b^2 (D_r + D_c)^2}$$

where

$$v_{ij} = r_i D_c + c_j D_r$$

Under the independence assumption, the standard error is

$$ASE_0 = 2\sqrt{\frac{\sum_{i,j} f_{ij}(C_{ij}-D_{ij})^2 - \frac{1}{W}(P-Q)^2}{D_r D_c}}$$

Kendall’s Tau-c

$$\tau_c = \frac{q(P-Q)}{W^2(q-1)}$$

with standard error

$$ASE_1 = \frac{2q}{(q-1)W^2}\sqrt{\sum_{i,j} f_{ij}(C_{ij}-D_{ij})^2 - \frac{1}{W}(P-Q)^2}$$

or, under the independence assumption,

$$ASE_0 = \frac{2q}{(q-1)W^2}\sqrt{\sum_{i,j} f_{ij}(C_{ij}-D_{ij})^2 - \frac{1}{W}(P-Q)^2}$$

where

$$q = \min\{R, C\}$$

Gamma

Gamma (γ) is estimated by

$$\gamma = \frac{P-Q}{P+Q}$$

with standard error

$$ASE_1 = \frac{4}{(P+Q)^2}\sqrt{\sum_{i,j} f_{ij}(QC_{ij} - PD_{ij})^2}$$

or, under the hypothesis of independence,

$$ASE_0 = \frac{2}{(P+Q)}\sqrt{\sum_{i,j} f_{ij}(C_{ij} - D_{ij})^2 - \frac{1}{W}(P-Q)^2}$$

Somers' d

Somers' *d* with row variable *X* as the independent variable is calculated as

$$d_{Y|X} = \frac{P-Q}{D_r}$$

with standard error

$$ASE_1 = \frac{2}{D_r^2}\sqrt{\sum_{i,j} f_{ij}\{D_r(C_{ij} - D_{ij}) - (P-Q)(W-R_i)\}^2}$$

or, under the hypothesis of independence,

$$ASE_0 = \frac{2}{D_r}\sqrt{\sum_{i,j} f_{ij}(C_{ij} - D_{ij})^2 - \frac{1}{W}(P-Q)^2}$$

By interchanging the roles of *X* and *Y*, the formulas for Somers' *d* with *X* as the dependent variable can be obtained.

Symmetric version of Somers' *d* is

$$d = \frac{(P-Q)}{\frac{1}{2}(D_c + D_r)}$$

The standard error is

$$ASE_1 = \frac{2\sigma^2_{\tau_b}}{(D_r + D_c)}\sqrt{D_r D_c}$$

where $\sigma^2_{\tau_b}$ is the variance of Kendall's τ_b,

$$ASE_0 = \frac{4}{(D_c + D_r)}\sqrt{\sum_{i,j} f_{ij}(C_{ij} - D_{ij})^2 - \frac{1}{W}(P - Q)^2}$$

Pearson's r

The Pearson's product moment correlation *r* is computed as

$$r = \frac{cov(X,Y)}{\sqrt{S(X)S(Y)}} \equiv \frac{S}{T}$$

where

$$cov(X,Y) = \sum_{i,j} X_i Y_j f_{ij} - \left(\sum_{i=1}^{R} X_i r_i\right)\left(\sum_{j=1}^{C} Y_j c_j\right)/W$$

$$S(X) = \sum_{i=1}^{R} X_i^2 r_i - \left(\sum_{i=1}^{R} X_i r_i\right)^2/W$$

and

$$S(Y) = \sum_{j=1}^{C} Y_j^2 c_j - \left(\sum_{j=1}^{C} Y_j c_j\right)^2/W$$

The variance of *r* is

$$var_1 = \tfrac{1}{T^4}\sum_{i,j} f_{ij}\left\{T(X_i - \overline{X})(Y_j - \overline{Y}) - \frac{S}{2T}\left[(X_i - \overline{X})^2 S(Y) + (Y_j - \overline{Y})^2 S(X)\right]\right\}^2$$

If the null hypothesis is true,

$$var_0 = \frac{\sum_{i,j} f_{ij} X_i^2 Y_j^2 - \left(\sum_{i,j} f_{ij} X_i Y_j\right)^2/W}{\left(\sum_i r_i X_i^2\right)\left(\sum_j c_j Y_j^2\right)}$$

where

$$\overline{X} = \sum_{i=1}^{R} X_i r_i / W$$

and

$$\overline{Y} = \sum_{j=1}^{C} Y_j c_j / W$$

Under the hypothesis that $\rho = 0$,

$$t = \frac{r\sqrt{W-2}}{\sqrt{1-r^2}}$$

is distributed as a t with $W-2$ degrees of freedom.

Spearman Correlation

The Spearman's rank correlation coefficient r_s is computed by using rank scores R_i for X_i and C_i for Y_j. These rank scores are defined as follows:

$$R_i = \sum_{k<i} r_k + (r_i + 1)/2 \qquad \text{for } i = 1, 2, \ldots, R$$

$$C_j = \sum_{h<j} c_h + (c_j + 1)/2 \quad \text{for } \; j = 1, 2, \ldots, C$$

The formulas for r_s and its asymptotic variance can be obtained from the Pearson formulas by substituting R_i and C_j for X_i and Y_j, respectively.

Eta

Asymmetric η with the column variable Y as dependent is

$$\eta_Y = \sqrt{1 - \frac{S_{YW}}{S(Y)}}$$

where

$$S_{YW} = \sum_{i,j} Y_j^2 f_{ij} - \sum_{i=1}^{R} \frac{1}{r_i} \left(\sum_{j=1}^{C} Y_j f_{ij} \right)^2$$

Relative Risk

Consider a 2×2 table (that is, $R = C = 2$). In a case-control study, the relative risk is estimated as

$$R_0 = \frac{f_{11} f_{22}}{f_{12} f_{21}}$$

The $100(1-\alpha)$ percent *CI* for the relative risk is obtained as

$$\left[R_0 \exp\left(-z_{1-\alpha/2} v\right), \quad R_0 \exp\left(z_{1-\alpha/2} v\right) \right]$$

where

$$v = \left(\frac{1}{f_{11}} + \frac{1}{f_{12}} + \frac{1}{f_{21}} + \frac{1}{f_{22}} \right)^{1/2}$$

The relative risk ratios in a cohort study are computed for both columns. For column 1, the risk is

$$R_1 = \frac{f_{11}(f_{21} + f_{22})}{f_{21}(f_{11} + f_{12})}$$

and the corresponding $100(1-\alpha)$ percent *CI* is

$$\left[R_1 \exp\left(-z_{1-\alpha/2} v\right), \quad R_1 \exp\left(z_{1-\alpha/2} v\right) \right]$$

where

$$v = \left(\frac{f_{12}}{f_{11}(f_{11} + f_{12})} + \frac{f_{22}}{f_{21}(f_{21} + f_{22})} \right)^{1/2}$$

The relative risk for column 2 and the confidence interval are computed similarly.

McNemar-Bowker's Test

This statistic is used to test if a square table is symmetric.

Notations

Table 23-2
Notation

Notation	Description
n	Dimension of the table (both row and column)
p_{ij}	Unknown population cell probability of row *i* and column *j*
n_{ij}	Observed counts cell count of row *i* and column *j*

Algorithm

Given a $n \times n$ square table, the McNemar-Bowker's statistic is used to test the hypothesis $H_0 : p_{ij} = p_{ji}$ for all (*i*<*j*) v.s. $H_1 : p_{ij} \neq p_{ji}$ for at least one pair of (*i*,*j*). The statistic is defined by the formula

$$\chi^2 = \sum_{i<j} \frac{I\left(n_{ij} + n_{ji} > 0\right)\left(n_{ij} - n_{ji}\right)^2}{n_{ij} + n_{ji}}$$

Where *I()* is the indicator function. Under the null hypothesis, χ^2 has an asymptotic Chi-square distribution with $n\left(n-1\right)/2$ degrees of freedom. The null hypothesis will be rejected if χ^2 has a large value. The two-sided p-value is equal to $1 - F\left(n\left(n-1\right)/2, \chi^2\right)$, where $F\left(df, \quad\right)$ is the CDF of Chi-square distribution with *df* degrees of freedom.

A Special Case: 2x2 Tables

For 2x2 table, the statistic reduces to the classical McNemar statistic (McNemar, 1947) for which exact p-value can be computed. The two-tailed probability level is

$$2 \sum_{i=0}^{\min(n_{12},n_{21})} \binom{n_{12} + n_{21}}{i} (1/2)^{n_{12}+n_{21}}$$

Conditional Independence and Homogeneity

The Cochran's and Mantel-Haenzel statistics test the independence of two dichotomous variables, controlling for one or more other categorical variables. These "other" categorical variables define a number of strata, across which these statistics are computed.

The Breslow-Day statistic is used to test homogeneity of the common odds ratio, which is a weaker condition than the conditional independence (i.e., homogeneity with the common odds ratio of 1) tested by Cochran's and Mantel-Haenszel statistics. Tarone's statistic is the Breslow-Day statistic adjusted for the consistent but inefficient estimator such as the Mantel-Haenszel estimator of the common odds ratio.

Notation and Definitions

Table 23-3
Notation

Notation	**Description**
K	The number of strata.
f_{ijk}	Sum of cell weights for cases in the *i*th row of the *j*th column of the *k*th strata.
c_{jk}	$\sum_{i=1}^{R} f_{ijk}$, the *j*th column of the *k*th strata subtotal.

Notation	Description
r_{ik}	$\sum_{j=1}^{C} f_{ijk}$, the *i*th row of the *k*th strata subtotal.
n_k	$\sum_{j=1}^{C} c_{jk} = \sum_{i=1}^{R} r_{ik}$, the grand total of the *k*th strata.
E_{ijk}	$E(f_{ijk}) = \frac{r_{ik} c_{jk}}{n_k}$, the expected cell count of the *i*th row of the *j*th column of the *k*th strata.

A stratum such that $n_k = 0$ is omitted from the analysis. (*K* must be modified accordingly.) If $n_k = 0$ for all *k*, then no computation is done.

Preliminarily, define for each *k*

$\hat{p}_{ik} = \frac{f_{i1k}}{r_{ik}}$,

$d_k = \hat{p}_{1k} - \hat{p}_{2k}$,

$\hat{p}_k = \frac{c_{1k}}{n_k}$,

and

$w_k = \frac{r_{1k} r_{2k}}{n_k}$.

Cochran's Statistic

Cochran's statistic (Cochran, 1954) is

$$C = \frac{\sum_{k=1}^{K} w_k d_k / \sum_{k=1}^{K} w_k}{\sqrt{\sum_{k=1}^{K} w_k \hat{p}_k (1 - \hat{p}_k) / \sum_{k=1}^{K} w_k}} = \frac{\sum_{k=1}^{K} w_k d_k}{\sqrt{\sum_{k=1}^{K} w_k \hat{p}_k (1 - \hat{p}_k)}}.$$

All stratum such that $r_{1k} = 0$ or $r_{2k} = 0$ are excluded, because d_k is undefined. If every stratum is such, *C* is undefined. Note that a stratum such that $r_{1k} > 0$ and $r_{2k} > 0$ but that $c_{1k} = 0$ or $c_{2k} = 0$ is a valid stratum, although it contributes nothing to the denominator or numerator. However, if every stratum is such, *C* is again undefined. So, in order to compute a non-system missing value of *C*, at least one stratum must have all non-zero marginal totals.

Alternatively, Cochran's statistic can be written as

$$C = \frac{\sum_{k=1}^{K} (f_{11k} - E_{11k})}{\sqrt{\sum_{k=1}^{K} w_k \hat{p}_k (1 - \hat{p}_k)}}.$$

When the number of strata is fixed as the sample sizes within each stratum increase, Cochran's statistic is asymptotically standard normal, and thus its square is asymptotically distributed as a chi-squared distribution with 1 d.f.

Mantel and Haeszel's Statistic

Mantel and Haenszel's statistic (Mantel and Haenszel, 1959) is simply Cochran's statistic with small-sample corrections for continuity and variance "inflation." These corrections are desirable when r_{1k} and r_{2k} are small, but the corrections can make a noticeable difference even for relatively large r_{1k} and r_{2k}(Snedecor and Cochran, 1980) (p. 213). The statistic is defined as:

$$M = \frac{\left\{\left|\sum_{k=1}^{K}\left(f_{11k} - E_{11k}\right)\right| - 0.5\right\} sgn\left\{\sum_{k=1}^{K}\left(f_{11k} - E_{11k}\right)\right\}}{\sqrt{\sum_{k=1}^{K}\frac{r_{1k}r_{2k}}{n_k - 1}\hat{p}_k\left(1 - \hat{p}_k\right)}},$$

where sgn is the signum function

$$sgn\left(x\right) = \begin{cases} 1 \text{ if } x > 0 \\ 0 \text{ if } x = 0 \\ -1 \text{ if } x < 0 \end{cases}.$$

Any stratum in which $n_k = 1$ is excluded from the computation. If every stratum is such, then *M* is undefined. *M* is also undefined if every stratum is such that $r_{1k} = 0$, $r_{2k} = 0$, $c_{1k} = 0$, or $c_{2k} = 0$. In order to compute a non-system missing value of *M*, at least one stratum must have all non-zero marginal totals, just as for *C*.

When the number of strata is fixed as the sample sizes within each stratum increase, or when the sample sizes within each strata are fixed as the number of strata increases, this statistic is asymptotically standard normal, and thus its square is asymptotically distributed as a chi-squared distribution with 1 d.f.

The Breslow-Day Statistic

The Breslow-Day statistic for any estimator $\hat{\theta}$ is

$$\sum_{k=1}^{K}\frac{\left\{f_{11k} - \mathrm{E}\left(f_{11k}|c_{1k};\hat{\theta}\right)\right\}^2}{\mathrm{V}\left(f_{11k}|c_{1k};\hat{\theta}\right)}.$$

E and V are based on the exact moments, but it is customary to replace them with the asymptotic expectation and variance. Let $\hat{E}$ and $\hat{V}$ mean the estimated asymptotic expectation and the estimated asymptotic variance, respectively. Given the Mantel-Haenszel common odds ratio estimator $\hat{\theta}_{\text{MH}}$, we use the following statistic as the Breslow-Day statistic:

$$B = \sum_{k=1}^{K} \frac{\left\{ f_{11k} - \hat{E}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)\right\}^2}{\hat{V}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)},$$

where

$$\hat{E}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right) = \hat{f}_{11k}$$

satisfies the equations

$$\frac{\hat{f}_{11k}\left(n_k - r_{1k} - c_{1k} + \hat{f}_{11k}\right)}{\left(r_{1k} - \hat{f}_{11k}\right)\left(c_{1k} - \hat{f}_{11k}\right)} = \hat{\theta}_{\text{MH}},$$

with constraints such that

$$\begin{aligned} &\hat{f}_{11k} \geq 0, \\ &r_{1k} - \hat{f}_{11k} > 0, \\ &c_{1k} - \hat{f}_{11k} > 0, \\ &n_k - r_{1k} - c_{1k} + \hat{f}_{11k} \geq 0; \end{aligned}$$

and

$$\hat{V}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right) = \left(\frac{1}{\hat{f}_{11k}} + \frac{1}{\hat{f}_{12k}} + \frac{1}{\hat{f}_{21k}} + \frac{1}{\hat{f}_{22k}}\right)^{-1}$$

with constraints such that

$$\begin{aligned} &\hat{f}_{11k} > 0, \\ &\hat{f}_{12k} = r_{1k} - \hat{f}_{11k} > 0, \\ &\hat{f}_{21k} = c_{1k} - \hat{f}_{11k} > 0, \\ &\hat{f}_{22k} = n_k - r_{1k} - c_{1k} + \hat{f}_{11k} > 0; \end{aligned}$$

All stratum such that $r_{1k} = 0$ or $c_{1k} = 0$ are excluded. If every stratum is such, B is undefined. Stratum such that $\hat{f}_{11k} = 0$ are also excluded. If every stratum is such, then B is undefined.

Breslow-Day's statistic is asymptotically distributed as a chi-squared random variable with *K*-1 degrees of freedom under the null hypothesis of a constant odds ratio.

Tarone's Statistic

Tarone (Tarone, 1985) proposes an adjustment to the Breslow-Day statistic when the common odds ratio estimator is consistent but inefficient, specifically when we have the Mantel-Haenszel common odds ratio estimator. The adjusted statistic, Tarone's statistic, for $\hat{\theta}_{\text{MH}}$ is

$$
\begin{aligned}
T &= \sum_{k=1}^{K} \frac{\left\{f_{11k} - \hat{E}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)\right\}^2}{\hat{V}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)} - \frac{\left[\sum_{k=1}^{K}\left\{f_{11k} - \hat{E}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)\right\}\right]^2}{\sum_{k=1}^{K}\hat{V}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)} \\
&= B - \frac{\left[\sum_{k=1}^{K}\left\{f_{11k} - \hat{E}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)\right\}\right]^2}{\sum_{k=1}^{K}\hat{V}\left(f_{11k}|c_{1k};\hat{\theta}_{\text{MH}}\right)},
\end{aligned}
$$

where $\hat{E}$ and $\hat{V}$ are as before.

The required data conditions are the same as for the Breslow-Day statistic computation. T is, of course, undefined, when B is undefined.

T is also asymptotically distributed as a chi-squared random variable with K-1 degrees of freedom under the null hypothesis of a constant odds ratio.

Estimation of the Common Odds Ratio

For K strata of 2×2 tables, write the true odds ratios as

$$
\theta_k = \frac{p_{1k}\left(1 - p_{2k}\right)}{\left(1 - p_{1k}\right)p_{2k}}
$$

for $k = 1, ..., K$. And, assuming that the true common odds ratio exists, $\theta = \theta_1 = ... = \theta_K$, Mantel and Haenszel's estimator (Mantel et al., 1959) of this common odds ratio is

$$
\hat{\theta}_{\text{MH}} = \frac{\sum_{k=1}^{K} \frac{f_{11k}f_{22k}}{n_k}}{\sum_{k=1}^{K} \frac{f_{12k}f_{21k}}{n_k}}.
$$

If every stratum is such that $f_{12k} = 0$ or $f_{21k} = 0$, then $\hat{\theta}_{\text{MH}}$ is undefined. The (natural) log of the estimated common odds ratio is asymptotically normal. Note, however, that if $f_{11k} = 0$ or $f_{22k} = 0$ in every stratum, then $\hat{\theta}_{\text{MH}}$ is zero and $\log\left(\hat{\theta}_{\text{MH}}\right)$ is undefined.

The Asymptotic Confidence Interval

Robins et al. (Robins, Breslow, and Greenland, 1986) give an estimated asymptotic variance for $\log\left(\hat\theta_{MH}\right)$ that is appropriate in both asymptotic cases:

$$\hat\sigma^2\left[\log\left(\hat\theta_{MH}\right)\right] = \frac{\sum_{k=1}^{K}\frac{\left(f_{11k}+f_{22k}\right)f_{11k}f_{22k}}{n_k^2}}{2\left(\sum_{k=1}^{K}\frac{f_{11k}f_{22k}}{n_k}\right)^2} + \frac{\sum_{k=1}^{K}\frac{\left(f_{11k}+f_{22k}\right)f_{12k}f_{21k}+\left(f_{12k}+f_{21k}\right)f_{11k}f_{22k}}{n_k^2}}{2\left(\sum_{k=1}^{K}\frac{f_{11k}f_{22k}}{n_k}\right)\left(\sum_{k=1}^{K}\frac{f_{12k}f_{21k}}{n_k}\right)} + \frac{\sum_{k=1}^{K}\frac{\left(f_{12k}+f_{21k}\right)f_{12k}f_{21k}}{n_k^2}}{2\left(\sum_{k=1}^{K}\frac{f_{12k}f_{21k}}{n_k}\right)^2}.$$

An asymptotic $(100-\alpha)$% confidence interval for $\log(\theta)$ is

$$\log\left(\hat\theta_{\mathrm{MH}}\right) \pm \alpha/2) \quad \hat\sigma\left[\log\left(\hat\theta_{\mathrm{MH}}\right)\right],$$

where z $(\alpha/2)$ is the upper $\alpha/2$ critical value for the standard normal distribution. All these computations are valid only if $\hat\theta_{\mathrm{MH}}$ is defined and greater than 0.

The Asymptotic P-value

We compute an asymptotic *P*-value under the null hypothesis that $\theta\,(=\theta_k\forall k)=\theta_0\,(>0)$ against a 2-sided alternative hypothesis $(\theta\neq\theta_0)$, using the standard normal variate, as follows

$$\Pr\left(|Z| > \left|\frac{\log\left(\hat\theta_{\mathrm{MH}}\right)-\log(\theta_0)}{\hat\sigma\left[\log\left(\hat\theta_{\mathrm{MH}}\right)\right]}\right|\right) = 2\Pr\left(Z > \left|\frac{\log\left(\hat\theta_{\mathrm{MH}}\right)-\log(\theta_0)}{\hat\sigma\left[\log\left(\hat\theta_{\mathrm{MH}}\right)\right]}\right|\right),$$

given that $\log\left(\hat\theta_{\mathrm{MH}}\right)$ is defined.

Alternatively, we can consider using $\hat\theta_{\mathrm{MH}}$ and the estimated exact variance of $\hat\theta_{\mathrm{MH}}$, which is still consistent in both limiting cases:

$$\hat\sigma^2\left[\log\left(\hat\theta_{\mathrm{MH}}\right)\right]\hat\theta^2_{\mathrm{MH}}.$$

Then, the asymptotic *P*-value may be approximated by

$$\Pr\left(|Z| > \left|\frac{\hat\theta_{\mathrm{MH}}-\theta_0}{\hat\sigma\left[\log\left(\hat\theta_{\mathrm{MH}}\right)\right]\theta_0}\right|\right).$$

The caveat for this formula is that $\hat\theta_{\mathrm{MH}}$ may be quite skewed even in moderate sample sizes (Robins et al., 1986).

Column Proportions Test

This section describes the computation of the column proportions test.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 23-4
Notation

Notation	Description
R	Number of rows in the sub-table.
C	Number of columns in the sub-table.
A_i	*i*th category of the row variable.
B_j	*j*th category of the column variable.
f_{ij}	Total case weights in cell (*i*,*j*).
c_j	Marginal case weights total in *j*th column.
$\tilde{c}_j$	Rounded marginal case weights total in *j*th column.
z	z-statistic.
χ^2	Chi-Square statistic.
p_{ij}	Column proportion for cell (*i*,*j*).
$\hat{p}_{ij}$	Estimated column proportion for cell (*i*,*j*).
$\hat{p}_{ijk}$	Estimate of pooled column proportion of *j* and *k*th column in *i*th row.
p	*p*-value of a test.
p_B	Bonferroni corrected *p*-value.
α	The significance level supplied by the user.

Conditions and Assumptions

- Pairwise tests are performed on each row of all eligible innermost sub-tables within each layer.
- The number of rows and columns in the sub-table must each be greater than or equal to two.
- Tests are constructed by using all visible categories excluding totals and sub-totals. Hiding of categories and showing of user-missing categories are respected.
- If weighting is on, cell statistics must include weighted cell counts or weighted simple column percents; a weighted analysis will be performed. If weighting is off, cell statistics requested must include cell counts or simple column percents; an unweighted analysis will be performed.
- A proportion will be discarded if the proportion is equal to zero or one, or the sum of case weights in a category is less than 2; that is, if $c_j < 2$. If less than two proportions are left after discarding proportions, test will not be performed.

Statistics

The following statistics are available.

Table Layout

	B_1	B_2	...	B_C
A_1	p_{11}	p_{12}	...	p_{1C}
A_2	p_{21}	p_{22}	...	p_{2C}
...	...	...	...	...
A_R	p_{R1}	p_{R2}	...	p_{RC}

Hypothesis

Without lost of generality, we will only look at the *i*th row of the table. Let C^* be the number of categories in the *i*th row where the proportion is greater than zero and less than one, and where the sum of case weights in the corresponding column is at least 2. In the *i*th row, $C^*(C^*-1)/2$ comparisons will be made among $p_{i1}, p_{i2}, ..., p_{iC}$. The (*j*,*k*)th hypothesis will be

$H_{0jk} : p_{ij} = p_{ik}$ vs. $H_{1jk} : p_{ij} \neq p_{ik}$

Aggregated Statistics

Column proportions tests are based on the aggregated proportions ($\hat{p}_{ij}$) and cell counts for each column (c_j). Column proportions are computed using the un-rounded cell counts $\hat{p}_{ij} = \frac{f_{ij}}{c_j}$ which are equal to the proportions actually displayed.

Statistics for the (i,j)th Comparisons

Pooled proportion: $\hat{p}_{ijk} = \frac{\tilde{c}_j \hat{p}_{ij} + \tilde{c}_k \hat{p}_{ik}}{\tilde{c}_j + \tilde{c}_k}$

z statistic with a categorical variable in the columns: $z = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1-\hat{p}_{ijk})\left(\frac{1}{\tilde{c}_j} + \frac{1}{\tilde{c}_k}\right)}}$

When multiple response set defines columns there may exist cases that belong to both *j*th and *k*th columns. Let $\tilde{c}_{jk}$ be the rounded sum of weights for such cases.

z statistic with a multiple response set in the columns: $z = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1-\hat{p}_{ijk})\left(\frac{1}{\tilde{c}_j} + \frac{1}{\tilde{c}_k} - \frac{2\tilde{c}_{jk}}{\tilde{c}_j \tilde{c}_k}\right)}}$

p-value: $p = 2\left[1 - \Phi\left(|z|\right)\right]$

where $\Phi(z)$ is the CDF of standard normal distribution.

Alternatively, the statistic can be constructed as a chi-square statistic,

$\chi^2 = z^2$

the *p*-value will now be given by $p = 1 - F\left(\chi^2; 1\right)$, where $F(x; df)$ is the CDF of a chi-square distribution with *df* degrees of freedom.

A comparison is significant if $p<\alpha$ (or $p_B < \alpha$, if Bonferroni adjusted).

Bonferroni Adjustment

If Bonferroni adjustment for multiple comparisons is requested, the *p*-value will be adjusted by

$$p_B = \min\left(\frac{p * C * (C * -1)}{2}, 1\right)$$

Relationship to Pearson's Chi-Square Tests

With a categorical variable in the columns, the statistics used in column proportion tests is equivalent to the Pearson's chi-square test on a 2×2 table by taking *j* and *k*th column and collapsing all rows except the *i*th row. Therefore performing column proportion tests on a 2×2 table will give you the same result as Pearson's chi-square test.

Use of Case Weights

The case weights (or frequency weights) are supposed to be integers representing number of replications of each case. In column proportions tests, we will only check if the column marginal c_j's are integers. If not, they will be rounded to the nearest integer.

References

Agresti, A. 1990. *Categorical Data Analysis*. New York: John Wiley.

Agresti, A. 1996. *An Introduction to Categorical Data Analysis*. New York: John Wiley and Sons.

Bishop, Y. M., S. E. Feinberg, and P. W. Holland. 1975. *Discrete multivariate analysis: Theory and practice*. Cambridge, Mass.: MIT Press.

Bowker, A. H. 1948. A test for symmetry in contingency tables. *Journal of the American Statistical Association*, 43, 572–574.

Breslow, N. E., and N. E. Day. 1980. *Statistical Methods in Cancer Research, 1, The Analysis of Case-Control Studies*. : International Agency for Research on Cancer, Lyon..

Brown, M. B. 1975. The asymptotic standard errors of some estimates of uncertainty in the two-way contingency table. *Psychometrika*, 40(3), 291.

Brown, M. B., and J. K. Benedetti. 1977. Sampling behavior of tests for correlation in two-way contingency tables. *Journal of the American Statistical Association*, 72, 309–315.

Cochran, W. G. 1954. Some methods of strengthening the common chi-square tests. *Biometrics*, 10, 417–451.

Goodman, L. A., and W. H. Kruskal. 1954. Measures of association for cross-classification.. *Journal of the American Statistical Association*, 49, 732–764.

Goodman, L. A., and W. H. Kruskal. 1972. Measures of association for cross-classification, IV: simplification and asymptotic variances. *Journal of the American Statistical Association*, 67, 415–421.

Hauck, W. 1989. Odds ratio inference from stratified samples. *Commun. Statis.-Theory Meth.*, 18, 767–800.

Somes, G. W., and K. F. O'Brien. 1985. *Mantel-Haenszel statistic. In Encyclopedia of Statistical Sciences, Vol. 5 (S. Kotz and N. L. Johnson, eds.) 214–217*. New York: John Wiley.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. *J. Natl. Cancer Inst.*, 22, 719–748.

McNemar, Q. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12, 153–157.

Robins, J., N. Breslow, and S. Greenland. 1986. Estimators of the Mantel-Haenszel variance consistent in both sparse data and large-strata limiting models. *Biometrics*, 42, 311–323.

Snedecor, G. W., and W. G. Cochran. 1980. *Statistical Methods*, 7th ed. Ames, Iowa: Iowa University Press.

Tarone, R. E. 1985. On heterogeneity tests based on efficient scores. *Biometrika*, 72, 91–95.

CSCOXREG Algorithms

Survival analysis studies the failure time distribution. This algorithm considers the Cox proportional hazards regression model under the complex sampling setting. The failure time is assumed to be continuous here.

Notation

The following notation is used throughout this chapter unless otherwise stated:

t_i	For data with one time interval, the observed end time for record *i*.
t_{1i}, t_{2i}	For data with two time intervals, the observed enter and end time for record *i*, $t_{1i} < t_{2i}$.
δ_i	The zero-one status indicator with $\delta_i = 1$ indicating end time t_i or t_{2i} being failure time, and $\delta_i = 0$ indicating t_i or t_{2i} being right censoring time.
$0 = t_0^* < \cdots < t_{K+1}^* = \infty$	The ordered observed failure times where *K* is the number of distinct failure times in the data set.
$\mathbf{x}_i$	Predictor vector for record *i*, $\mathbf{x}_i = (x_{i1}, \cdots, x_{ip})'$. No intercept term.
x_0	Vector of reference values for transforming predictors. For more information, see the topic “Predictor Transformations”.
$\mathbf{X}$	Design matrix $\mathbf{X} = (\mathbf{x}_1, \cdots, \mathbf{x}_n)'$.
$D(t)$	The set of records failed at time *t*. $D(t) = \{i : t_i = t, \delta_i = 1\}$ for data with one time variable, and $D(t) = \{i : t_{2i} = t, \delta_i = 1\}$ for data with two time variables.
$R(t)$	The set of records at risk at time *t*. $R(t) = \{i : t_i \geq t\}$ for data with one time variable, and $R(t) = \{i : t_{1i} < t \leq t_{2i}\}$ for data with two time variables.
$Y_i(t)$	The at-risk indicator for record *i* such that $Y_i(t) = \begin{cases} 1 & \text{if } i \in R(t) \\ 0 & \text{otherwise} \end{cases}$
$d(t)$	The number of records failed at time *t*; that is, the number of records in $D(t)$
$S(t\|\mathbf{x})$	Survival function at time *t* for a given predictor vector $\mathbf{x}$, $S(t) \equiv \Pr(T > t\|\mathbf{x})$ where *T* is a random variable representing survival time.
$h(t\|\mathbf{x})$	Hazard function at time *t* for a given predictor vector $\mathbf{x}$, $h(t\|\mathbf{x}) \equiv \lim_{\Delta t \to 0^+} \frac{\Pr(t \leq T < t+\Delta t \| T \geq t\|\mathbf{x})}{\Delta t}$.
$H(t\|\mathbf{x}) = \int_0^t h(u\|\mathbf{x})\,du$	Cumulative hazard function at time *t* for a given predictor vector $\mathbf{x}$
$h_0(t)$	Baseline hazard function at time *t*, $h_0(t) = h(t\|\mathbf{x} = 0)$.
$H_0(t) = \int_0^t h_0(u)\,du$	Cumulative baseline hazard function at time *t*.
$S_0(t) = \exp(-H_0(t))$	Baseline survival function at time *t*.
N	The number of cases in the whole population.
n	The number of cases/records in the sample.
n_s	The number of subjects/individuals in the sample.

w_i	Sampling weight for record *i*, $w_i = 1/\pi_i$.
$\mathbf{B}$	The parameter of interest, the population or census parameter.
$\hat{\mathbf{B}}$	The estimate of census parameter $\mathbf{B}$ from the sample.

Input

Sampling plan. This plan is needed for sampling method, sampling weight, strata and cluster information.

Observed sample data. Two kinds of data structures are allowed.

- Data with one time intervals: $\{t_i, \delta_i, \mathbf{x}_i, w_i\}_{i=1}^n$.
- Data with two time intervals: $\{t_{1i}, t_{2i}, \delta_i, \mathbf{x}_i, w_i\}_{i=1}^n$, or $\{\mathrm{id}_i, t_{1i}, t_{2i}, \delta_i, \mathbf{x}_i, w_i\}_{i=1}^n$, where $(t_{1i}, t_{2i}]$ is the time interval during which the record is at risk, and id $_i$is the subject id for record *i*. Multiple records for the same subject have the same id and same sampling weight. Multiple records of the same subject should have disjoint time intervals. If id$_i$is not specified, each record is assumed from different subject.

Note: Data with one time interval is simply a special case of data with two time intervals where $t_{1i} = 0$ and $t_{2i} = t_i$. The rest of this document is written from the perspective of data with two time intervals.

Predictor Transformations

To decrease the chance of over- or underflow when calculating exp(.), first a transformation $z = x - x_0$ is performed on each predictor for a properly chosen x_0 (reference value). Then all the calculations described in other sections are performed on the transformed data. Except for baseline hazards and baseline survival functions, all other quantities based on transformed data are the same as those based on original data.

For a continuous predictor *x* in the original covariate list, the reference value x_0 is chosen to be

$$x_0 = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

Note that x_0 is not the mean of *x* when there are multiple cases per subject or *x* is a time dependent predictor.

For a categorical predictor, the last category is the reference value.

The reference values for model effects derived from original predictors, such as interactions, are derived from the reference values of the original predictors in the same way the effects are derived.

Proportional Hazards

Two phases of sampling are assumed. The first phase generates a finite population by a model or super population. The second phase selects a sample according to a sampling plan from the finite population generated in the first phase.

Model

For a given predictor vector $\mathbf{x}$, the hazard function at time *t* is

$$h(t|\mathbf{x}) = h_0(t)\exp\left\{\mathbf{x}'\beta\right\}$$

or

$$\ln\left(\frac{h(t|\mathbf{x})}{h_0(t)}\right) = \mathbf{x}'\beta$$

where $h_0(t)$ is the baseline hazard function. The regression parameter vector doesn't include an intercept term because the intercept can be absorbed by the baseline hazard.

Survival and cumulative hazard functions

From this model the cumulative hazard function is

$$H(t|\mathbf{x}) = \int_0^t h(u|\mathbf{x})\,du = \exp\left(\mathbf{x}'\beta\right)\int_0^t h_0(u)\,du = \exp\left(\mathbf{x}'\beta\right)H_0(t)$$

where $H_0(t) = \int_0^t h_0(u)\,du$ is the baseline cumulative hazard function. The survival function is

$$S(t|\mathbf{x}) = \exp\{-H(t|\mathbf{x})\} = \exp\left\{-\exp\left(\mathbf{x}'\beta\right)H_0(t)\right\} = \{S_0(t)\}^{\exp\left(\mathbf{x}'\beta\right)}$$

where $S_0(t) = \exp(-H_0(t))$ is the baseline survival function.

Pseudo Partial Likelihood and Derivatives

For a sample $S = \{t_{1i}, t_{2i}, \delta_i, \mathbf{x}_i, w_i\}_{i=1}^n$ drawn from the finite population according to a sample plan, we take the pseudo-likelihood approach. In this approach, pseudo-likelihood is a sample estimate of the population log-likelihood, and parameter estimates are derived by maximizing the pseudo-likelihood. Let $l_S(\beta)$, $U_S(\beta)$ and $J_S(\beta)$ denote the pseudo-likelihood, its first and second derivatives.

For the Breslow approximation:

$$l_S(\beta) = \sum_{i=1}^{n} w_i \delta_i \left(\mathbf{x}'_i \beta - \ln \sum_{l \in R(t_{2i})} w_l \exp\left(\mathbf{x}'_l \beta\right) \right)$$

For the Efron approximation:

$$l_S(\beta) = \sum_{i=1}^{n} w_i \delta_i \left(\mathbf{x}'_i \beta - \frac{1}{d(t_{2i})} \sum_{r=0}^{d(t_{2i})-1} \ln \left\{ \sum_{l \in R(t_{2i})} w_l \exp\left(\mathbf{x}'_l \beta\right) - \frac{r}{d(t_{2i})} \sum_{l \in D(t_{2i})} w_l \exp\left(\mathbf{x}'_l \beta\right) \right\} \right)$$

Let

$$E^{(0)}(\beta,t) = \sum_{l \in R(t)} w_l \exp\left(\mathbf{x}'_l \beta\right) = \sum_{l=1}^{n} w_l Y_l(t) \exp\left(\mathbf{x}'_l \beta\right)$$

$$E^{(1)}(\beta,t) = \frac{\partial E^{(0)}(\beta,t)}{\partial \beta} = \sum_{l \in R(t)} w_l \mathbf{x}_l \exp\left(\mathbf{x}'_l \beta\right) = \sum_{l=1}^{n} w_l Y_l(t) \mathbf{x}_l \exp\left(\mathbf{x}'_l \beta\right)$$

$$E^{(2)}(\beta,t) = \frac{\partial^2 E^{(0)}(\beta,t)}{\partial \beta \partial \beta} = \sum_{l \in R(t)} w_l \mathbf{x}_l \mathbf{x}'_l \exp\left(\mathbf{x}'_l \beta\right) = \sum_{l=1}^{n} w_l Y_l(t) \mathbf{x}_l \mathbf{x}'_l \exp\left(\mathbf{x}'_l \beta\right)$$

$$\begin{aligned} EE^{(0)}(\beta,t,r) &= \textstyle\sum_{l \in R(t)} w_l \exp\left(\mathbf{x}'_l \beta\right) - \frac{r}{d(t)} \sum_{l \in D(t)} w_l \exp\left(\mathbf{x}'_l \beta\right) \\ &= \textstyle\sum_{l=1}^{n} w_l \left(Y_l(t) - \frac{r \delta_l I(t_{2l}=t)}{d(t)} \right) \exp\left(\mathbf{x}'_l \beta\right) \end{aligned}$$

$$\begin{aligned} EE^{(1)}(\beta,t,r) &= \textstyle\sum_{l \in R(t)} w_l \mathbf{x}_l \exp\left(\mathbf{x}'_l \beta\right) - \frac{r}{d(t)} \sum_{l \in D(t)} w_l \mathbf{x}_l \exp\left(\mathbf{x}'_l \beta\right) \\ &= \textstyle\sum_{l=1}^{n} w_l \left(Y_l(t) - \frac{r \delta_l I(t_{2l}=t)}{d(t)} \right) \mathbf{x}_l \exp\left(\mathbf{x}'_l \beta\right) \end{aligned}$$

$$\begin{aligned} EE^{(2)}(\beta,t,r) &= \textstyle\sum_{l \in R(t)} w_l \mathbf{x}_l \mathbf{x}'_l \exp\left(\mathbf{x}'_l \beta\right) - \frac{r}{d(t)} \sum_{l \in D(t)} w_l \mathbf{x}_l \mathbf{x}'_l \exp\left(\mathbf{x}'_l \beta\right) \\ &= \textstyle\sum_{l=1}^{n} w_l \left(Y_l(t) - \frac{r \delta_l I(t_{2l}=t)}{d(t)} \right) \mathbf{x}_l \mathbf{x}'_l \exp\left(\mathbf{x}'_l \beta\right) \end{aligned}$$

$$\overline{\mathbf{x}}(\beta,t,r) = \frac{EE^{(1)}(\beta,t,r)}{EE^{(0)}(\beta,t,r)}$$

$$\overline{\mathbf{x}}(\beta,t) = \begin{cases} \frac{E^{(1)}(\beta,t)}{E^{(0)}(\beta,t)} & \text{Breslow} \\ \frac{1}{d(t)} \sum_{r=0}^{d(t)-1} \frac{EE^{(1)}(\beta,t,r)}{EE^{(0)}(\beta,t,r)} & \text{Efron} \end{cases}$$

$$\mathbf{u}_i(\beta,t,r) = \mathbf{x}_i - \overline{\mathbf{x}}(\beta,t,r)$$

$$\mathbf{u}_i(\beta,t) = \mathbf{x}_i - \overline{\mathbf{x}}(\beta,t)$$

$$I_i(\beta,t) = \begin{cases} \frac{E^{(1)}(\beta,t)\left(E^{(1)}(\beta,t)\right)'}{\left(E^{(0)}(\beta,t)\right)^2} - \frac{E^{(2)}(\beta,t)}{E^{(0)}(\beta,t)} & \text{Breslow} \\ \frac{1}{d(t)}\sum_{r=0}^{d(t)-1} & \text{Efron} \end{cases}$$

So

$$l_S(\beta) = \begin{cases} \sum_{i=1}^{n} w_i\delta_i\left(\mathbf{x}'_i\beta - \ln E^{(0)}(\beta, t_{2i})\right) & \text{Breslow} \\ \sum_{i=1}^{n} w_i\delta_i\left(\mathbf{x}'_i\beta - \frac{1}{d(t_{2i})}\sum_{r=0}^{d(t_{2i})-1}\ln EE^{(0)}(\beta, t_{2i}, r)\right) & \text{Efron} \end{cases}$$

$$U_S(\beta) = \frac{\partial l_S(\beta)}{\partial\beta} = \sum_{i=1}^{n} w_i\delta_i\mathbf{u}_i(\beta, t_{2i})$$

$$J_S(\beta) = \frac{\partial^2 l_S(\beta)}{\partial\beta\partial\beta} = \sum_{i=1}^{n} w_i\delta_i I_i(\beta, t_{2i})$$

These equations are used to calculate the needed quantities throughout the rest of the document. When predictors are time-dependent, these equations need to be modified accordingly. For more information, see the topic "Time-Dependent Predictors".

Parameter Estimation

To obtain the maximum pseudo-likelihood estimate of **B**, the Newton-Raphson iterative estimation method is used to solve the estimating equation. Redundant parameters are fixed at zero for all iterations. Let $\mathbf{B}^{(v)}$ be the parameter estimate at iteration step *v*, the parameter estimate $\mathbf{B}^{(v+1)}$ at iteration step *v* + 1 is updated as

$$\mathbf{B}^{(v+1)} = \mathbf{B}^{(v)} - \xi \left(J_S\left(\mathbf{B}^{(v)}\right)\right)^{-} U_S\left(\mathbf{B}^{(v)}\right)$$

where $(J_S(.))^{-}$ is a generalized inverse of $J_S(.)$. The stepping scalar ξ>0 is used to make $l_S\left(\mathbf{B}^{(v+1)}\right) \geq l_S\left(\mathbf{B}^{(v)}\right)$. Use the step-halving method if $l_S\left(\mathbf{B}^{(v+1)}\right) < l_S\left(\mathbf{B}^{(v)}\right)$. Let *s* be the maximum number of steps in step-halving; the set of values of ξ is then {1/2^r: *r* = 0, …, *s*−1}.

Starting with initial value $\mathbf{B}^{(0)}$, update $\mathbf{B}^{(v+1)}$ until one of the stopping criteria is satisfied. The final estimate is denoted as $\hat{\mathbf{B}}$.

Initial values

By default, $\mathbf{B}^{(0)} = 0$.

Stopping criteria

Given two small constants ϵ_l > 0 and ϵ_p > 0, the iteration stops if one of the following criteria is satisfied:

1. Pseudo-likelihood criterion

$$\begin{cases} \frac{\left|l_S\left(\mathbf{B}^{(v+1)}\right)-l_S\left(\mathbf{B}^{(v)}\right)\right|}{\left|l_S\left(\mathbf{B}^{(v)}\right)\right|+10^{-6}} < \epsilon_l & \text{if relative change} \\ \left|l_S\left(\mathbf{B}^{(v+1)}\right) - l_S\left(\mathbf{B}^{(v)}\right)\right| < \epsilon_l & \text{if absolute change} \end{cases}$$

2. Parameter criterion

$$\begin{cases} \max_j\left(\frac{\left|B_j^{(v+1)}-B_j^{(v)}\right|}{\left|B_j^{(v)}\right|+10^{-6}}\right) < \epsilon_p & \text{if relative change} \\ \max_j\left(\left|B_j^{(v+1)} - B_j^{(v)}\right|\right) \quad \epsilon_p & \text{if absolute change} \end{cases}$$

3. The maximum number of iteration is reached, or maximum number of steps in step-halving is reached.

Either relative or absolute change is considered in criteria 1 and 2.

Infinite valued parameters

There may be situations in which the maximum pseudo-likelihood estimates of some parameters are infinite. For example, if there is no failure at one level of a binary predictor, the estimated parameter would be infinity for this predictor. In this situation, the estimation procedure is performed as usual. At the end of the estimation, we will check for possible infinite parameters and issue warnings if there are any. Parameter B_j is possibly infinite if both of the followings are satisfied:

1. $\left|\hat{B}_j\right| \quad (x_{j,\max} - x_{j,\min}) \geq 10$
2. The Hessian is singular, or $se\left(\hat{B}_j\right)/\left|\hat{B}_j\right| \geq 3$.

When there are infinite valued parameters, the Wald statistic for hypothesis testing involving infinite valued parameters becomes worthless.

Properties of Estimates

Variance matrix

Let

$$\mathbf{U}_i(\beta) = \begin{cases} \delta_i\mathbf{u}_i(\beta, t_{2i}) - \sum_{\{m:t_{1i}<t_{2m}\leq t_{2i}\}} \frac{\delta_m w_m \mathbf{u}_i(\beta,t_{2m})\exp\left(\mathbf{x}'_i\beta\right)}{E^{(0)}(\beta,t_{2m})} & \text{Breslow} \\ \delta_i\mathbf{u}_i(\beta, t_{2i}) - \sum_{\{m:t_{1i}<t_{2m}\leq t_{2i}\}} \frac{\delta_m w_m}{d(t_{2m})} \sum_{r=0}^{d(t_{2m})-1} \frac{\mathbf{u}_i(\beta,t_{2m},r)\exp\left(\mathbf{x}'_i\beta\right)\left(1-\frac{rI(t_{2m}=t_{2i})\delta_i}{d(t_{2m})}\right)}{EE^{(0)}(\beta,t_{2m},r)} & \text{Efron} \end{cases}$$

We will use the following robust variance estimation (Binder 1992, Lin 2000),

$$\hat{V}\left(\hat{\mathbf{B}}\right) \approx \left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-} \hat{I}\left(\hat{\mathbf{B}}\right)\left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-}$$

where $\hat{I}(\beta)$ is the estimate of the design based variance of $U_S(\beta)$ with

$$U_S(\beta) = \sum_{j=1}^{n_s} w_j \mathbf{U}_j^{(a)}(\beta)$$

$$\mathbf{U}_j^{(a)}(\beta) = \sum_{i \in \{\mathrm{id}_i = j\}} \mathbf{U}_i(\beta)$$

Notice that the sum in $U_S(\beta)$ is over all n_s subjects, and the sum in $\mathbf{U}_j^{(a)}(\beta)$ is over all records for subject *j*. The $U_S(.)$ is an estimate for the population total of $\mathbf{U}_j^{(a)}(.)$ vectors. For more information, see the topic “Complex Samples: Covariance Matrix of Total”.

Confidence interval

The confidence interval for a single regression parameter B_j is approximately

$$\left[\hat{B}_j - t_{df,1-\frac{\alpha}{2}}\sqrt{\hat{V}\left(\hat{B}_j\right)}, \hat{B}_j + t_{df,1-\frac{\alpha}{2}}\sqrt{\hat{V}\left(\hat{B}_j\right)}\right]$$

where $t_{df,1-\frac{\alpha}{2}}$ is the $100(1-\alpha/2)$ percentile of a *t* distribution with *df* degrees of freedom.

The degrees of freedom *df* can be user specified; its default value is the difference between the number of primary sampling units and the number of strata in the first stage of sampling.

Design effect

For each parameter B_j, its design effect is the ratio of its variance under the design to its variance under the SRS design,

$$Deff\left(\hat{B}_j\right) = \frac{\hat{V}\left(\hat{B}_j\right)}{\hat{V}_{SRS}\left(\hat{B}_j\right)}$$

For SRS design, the variance matrix is

$$\hat{V}_{SRS}\left(\hat{\mathbf{B}}\right) \approx \left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-} \hat{I}_{SRS}\left(\hat{\mathbf{B}}\right)\left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-}$$

where

$$\hat{I}_{SRS}\left(\hat{\mathbf{B}}\right)=\hat{\mathbf{V}}_{SRS}\left(U_S\left(\hat{\mathbf{B}}\right)\right)=fpc\frac{\hat{N}}{n_s-1}\sum_{j=1}^{n_s}w_j\mathbf{U}_j^{(a)}\left(\hat{\mathbf{B}}\right)\left(\mathbf{U}_j^{(a)}\left(\hat{\mathbf{B}}\right)\right)'$$

$$\hat{N}=\sum_{j=1}^{n_s}w_j$$

$$fpc=\begin{cases}1-\frac{n_s}{\hat{N}} & \text{with finite population correction.}\\ 1 & \text{without finite population correction.}\end{cases}$$

t Tests

Testing hypothesis $H_0 : B_j = 0$ for each non-redundant model parameter B_j is performed using the *t* test statistic:

$$t\left(\hat{B}_j\right)=\frac{\hat{B}_j}{\sqrt{\hat{V}\left(\hat{B}_j\right)}}$$

The *p*-value for the two-sided test is given by the probability $P\left(|T|>\left|t\left(\hat{B}_j\right)\right|\right)$, where *T* is a random variable from the *t* distribution with *df* degrees of freedom.

Exponentiated parameter estimates

$\exp(B_j)$ can be interpreted as a hazard ratio for main effects model. Its $1-\alpha$ confidence interval is

$$\left[\exp\left(L\left(\hat{B}_j\right)\right),\exp\left(U\left(\hat{B}_j\right)\right)\right]$$

where $L\left(\hat{B}_j\right), U\left(\hat{B}_j\right)$ are the lower and upper confidence limits for census parameter B_j.

Survival and Cumulative Hazard Functions

In this section, $t_1^* < \cdots < t_K^*$ are the ordered observed failure times, and $t_0^*=0$, $t_{K+1}^*=\infty$ are used for convenience. The estimates are valid for $t\in\left[0,\max_i\left(t_{2i}\right)\right]$.

Estimation of Baseline Survival and Cumulative Hazard Functions

Only one of these needs to be estimated because $H_0(t)=-\ln S_0(t)$ and $S_0(t)=\exp\left(-H_0(t)\right)$ The baseline functions are estimated by right continuous step functions with jumps only at observed failure times; that is, $\hat{S}_0(t)=\hat{S}_0\left(t_j^*\right)$ and $\hat{H}_0(t)=\hat{H}_0\left(t_j^*\right)$ for $t\in\left[t_j^*,t_{j+1}^*\right)$.

Product-limit Estimate

The non-increasing right continuous baseline survival function $S_0(t)$ is estimated here. Let the ratio jump be $\alpha_j = S_0\left(t_j^*\right)/S_0\left(t_{j-1}^*\right)$ for *j* = 1 to *K*, and $\alpha_0 = 1$, so $S_0(t) = \prod_{\left\{l: t_l^* \le t\right\}} \alpha_l$
.

Assuming that the regression parameters are given, $\{\alpha_j\}_{j=1}^K$ will be the parameters to be estimated by maximum likelihood estimation.

Pseudo likelihood and its derivatives

Let $f(t|\mathbf{x})$ be the probability density function of failure time at *t* for a given predictor. The pseudo likelihood is

$$l_S(\alpha,\beta) = \sum_{j=1}^{K}\left\{\sum_{i\in D\left(t_j^*\right)} w_i \ln\left(1-\alpha_j^{\exp\left(\mathbf{x}'_i\beta\right)}\right) + \sum_{i\in R\left(t_j^*\right)-D\left(t_j^*\right)} w_i \exp\left(\mathbf{x}'_i\beta\right)\ln\alpha_j\right.$$

We will estimate α_j by maximizing $l_S\left(\alpha,\hat{\mathbf{B}}\right)$, which is equivalent to solving $\frac{\partial l_S\left(\alpha,\hat{\mathbf{B}}\right)}{\partial\alpha_j} = 0$ and hence the following equation.

$$\sum_{i\in D\left(t_j^*\right)} w_i \frac{\exp\left(\mathbf{x}'_i\hat{\mathbf{B}}\right)}{1-\alpha_j^{\exp\left(\mathbf{x}'_i\beta\right)}} = \sum_{i\in R\left(t_j^*\right)} w_i \exp\left(\mathbf{x}'_i\hat{\mathbf{B}}\right)$$

Failure times of single failure

If there is only a single failure i_j at failure time t_j^*, there exists a closed form solution,

$$\hat{\alpha}_j = \left(1-\frac{w_{i_j}\exp\left(\mathbf{x}'_{i_j}\hat{\mathbf{B}}\right)}{\sum_{i\in R\left(t_j^*\right)} w_i\exp\left(\mathbf{x}'_i\hat{\mathbf{B}}\right)}\right)^{\exp\left(-\mathbf{x}'_{i_j}\hat{\mathbf{B}}\right)} = \left(1-\frac{w_{i_j}\exp\left(\mathbf{x}'_{i_j}\hat{\mathbf{B}}\right)}{E^{(0)}\left(\hat{\mathbf{B}},t_j^*\right)}\right)^{\exp\left(-\mathbf{x}'_{i_j}\hat{\mathbf{B}}\right)}$$

Failure times with tied failures

If there are multiple failures at failure time t_j^*, Newton's iterative method is used to solve the equation with constraint $\alpha_j \in (0,1)$. A good initial value is

$$\alpha_{j0} = \exp\left(-\frac{\sum_{l\in D\left(t_j^*\right)} w_l}{\sum_{l\in R\left(t_j^*\right)} w_l\exp\left(\mathbf{x}'_l\hat{\mathbf{B}}\right)}\right) = \exp\left(-\frac{\sum_{l\in D\left(t_j^*\right)} w_l}{E^{(0)}\left(\hat{\mathbf{B}},t_j^*\right)}\right)$$

Kaplan-Meier estimator: a special situation of no predictors

When there are no predictors; that is, $\mathbf{x}=0$ always, the product-limit estimator becomes the Kaplan-Meier estimator,

$$\hat{\alpha}_j = 1 - \frac{\sum_{i\in D\left(t_j^*\right)} w_i}{\sum_{i\in R\left(t_j^*\right)} w_i}$$

Breslow, or Nelson-Aalan, or Empirical, Estimate

Here $H_0(t)$ is estimated by a non-decreasing step function with steps at observed failure times:

$$\hat{H}_0(t) = \sum_{\{k:t_k^*\le t\}} \frac{\sum_{i\in D(t_k^*)} w_i}{E^{(0)}\left(\hat{\mathbf{B}}, t_k^*\right)}$$

where $N_i(t)$ is the count of failures up to time *t* for record *i*.

Efron Estimate

When there are ties in failure times, the following estimation can also be used. This will reduce to Breslow when there are no ties.

$$\hat{H}_0(t) = \sum_{\{k:t_k^*\le t\}} \frac{\sum_{i\in D(t_k^*)} w_i}{d\left(t_k^*\right)} \sum_{r=0}^{d(t_k^*)-1} \frac{1}{EE^{(0)}\left(\hat{\mathbf{B}}, t_k^*, r\right)}$$

Prediction of Survival and Cumulative Hazard Functions

For a given $\mathbf{x}$, the cumulative hazard function and survival functions are predicted by

$$\hat{H}(t|\mathbf{x}) = \hat{H}_0(t)\exp\left(\mathbf{x}'\hat{\mathbf{B}}\right)$$

$$\hat{S}(t|\mathbf{x}) = \exp\left(-H(t|\mathbf{x})\right) = \left(\hat{S}_0(t)\right)^{\exp\left(\mathbf{x}'\hat{\mathbf{B}}\right)}$$

where $\hat{H}_0(t)$ and $\hat{S}_0(t)$ are the estimated baseline cumulative hazard function and baseline survival function.

For variance calculation, the same formula will be used regardless of different ways to estimate baseline functions. The variance for cumulative hazard is

$$\hat{V}\left(\hat{H}\left(t|\mathbf{x}\right)\right) \approx \left(Var\left(\sum_{j=1}^{n_s} w_j q_j^{(a)}\left(t|\mathbf{x}\right)\right)\right)_{q_j^{(a)}(t|\mathbf{x})=\hat{q}_j^{(a)}(t|\mathbf{x})}$$

where

$$\hat{q}_j^{(a)}\left(t|\mathbf{x}\right) = \sum_{i\in\{\mathrm{id}_i=j\}} \hat{q}_i\left(t|\mathbf{x}\right)$$

$$\hat{q}_i\left(t\,|\mathbf{x}\right) = v_i\left(t|\,\mathbf{x}\right) - \left(A\left(t|\mathbf{x}\right)\right)'\left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-1}\mathbf{U}_i\left(\hat{\mathbf{B}}\right)$$

$$v_i\left(t|\mathbf{x}\right) = \frac{\delta_i I\left(t_{2i} \le t\right)}{E^{(0)}\left(\hat{\mathbf{B}}, t_{2i}|\mathbf{x}\right)} - \sum_{l=1}^{n} w_l \frac{\delta_l I\left(t_{2l} \le t\right) Y_i\left(t_{2l}\right)\exp\left(\left(\mathbf{x}_i - \mathbf{x}\right)'\hat{\mathbf{B}}\right)}{\left(E^{(0)}\left(\hat{\mathbf{B}}, t_{2l}|\mathbf{x}\right)\right)^2}$$

$$A\left(t|\mathbf{x}\right) = -\sum_{l=1}^{n} w_l \frac{\delta_l I\left(t_{2l} \le t\right) E^{(1)}\left(\hat{\mathbf{B}}, t_{2l}|\mathbf{x}\right)}{\left(E^{(0)}\left(\hat{\mathbf{B}}, t_{2l}|\mathbf{x}\right)\right)^2}$$

$$E^{(0)}\left(\hat{\mathbf{B}}, t|\mathbf{x}\right) = \sum_{l=1}^{n} w_l Y_l\left(t\right)\exp\left(\left(\mathbf{x}_l - \mathbf{x}\right)'\hat{\mathbf{B}}\right)$$

$$E^{(1)}\left(\hat{\mathbf{B}}, t|\mathbf{x}\right) = \sum_{l=1}^{n} w_l Y_l\left(t\right)\left(\mathbf{x}_l - \mathbf{x}\right)\exp\left(\left(\mathbf{x}_l - \mathbf{x}\right)'\hat{\mathbf{B}}\right)$$

and $J_S\left(\beta\right)$ and $\mathbf{U}_i\left(\beta\right)$ are defined in "Pseudo Partial Likelihood and Derivatives " and "Properties of Estimates ", respectively. See Lin (2000) for more details. $Var\left(\sum_{j=1}^{n_s} w_j q_j^{(a)}\left(t\right)\right)$ is the design-based variance of $\sum_{j=1}^{n_s} w_j q_j^{(a)}\left(t\right)$ which is the estimated population total of $q_j^{(a)}\left(t\right)$. For more information, see the topic "Complex Samples: Covariance Matrix of Total".

The variance estimate for the survival function is

$$\hat{V}\left(\hat{S}\left(t|\mathbf{x}\right)\right) = \left(\hat{S}\left(t|\mathbf{x}\right)\right)^2\hat{V}\left(\hat{H}\left(t|\mathbf{x}\right)\right)$$

Confidence interval for survival function

A confidence interval for $\hat{S}\left(t|\mathbf{x}\right)$ can be calculated in the following ways. Let

$$Y(t) = \begin{matrix} \hat{S}(t|\mathbf{x}) & \text{original} \\ \ln \hat{S}(t|\mathbf{x}) & \log \\ \ln\left(-\ln \hat{S}(t|\mathbf{x})\right) & \log-\log \end{matrix}$$

the confidence interval for $\hat{S}(t|\mathbf{x})$ at $1-\alpha$ level is

$$\begin{cases} Y(t) \pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))} = \hat{S}(t|\mathbf{x}) \pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))} & \text{original} \\ \exp\left(Y(t) \pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))}\right) = \hat{S}(t|\mathbf{x}) \exp\left(\pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))}\right) & \log \\ \exp\left\{-\exp\left(Y(t) \pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))}\right)\right\} = \left(\hat{S}(t|\mathbf{x})\right)^{\exp\left(\pm z_{\alpha/2}\sqrt{\hat{V}(Y(t))}\right)} & \log-\log \end{cases}$$

where $z_{\alpha/2}$ is the $1-\frac{\alpha}{2}$ upper percentile point of the standard normal distribution and

$$\hat{V}(Y) = \begin{cases} \left(\hat{S}(t|\mathbf{x})\right)^2 \hat{V}\left(\hat{H}(t|\mathbf{x})\right) & \text{original} \\ \hat{V}\left(\hat{H}(t|\mathbf{x})\right) & \log \\ \left(\hat{H}(t|\mathbf{x})\right)^{-2} \hat{V}\left(\hat{H}(t|\mathbf{x})\right) & \log-\log \end{cases}$$

Please note that the first two confidence intervals may have values greater than 1 or less than zero (we can truncate them to 0 or 1 if they are out of range). The third one always between 0 and 1. However Link (1984 & 1986) suggested that the second one performed the best.

Residuals

Some residuals defined below depend on the baseline cumulative function. Three estimation methods for baseline cumulative function are available to user. If users don't request estimation of cumulative hazard or survival function, but request for residuals, then use Breslow estimate if Breslow approximation is chosen in estimating the parameters, and Efron estimate if Efron approximation is chosen in estimating the parameters.

Schoenfeld's partial residuals

This is calculated only for observations with $\delta_i = 1$.

$$\mathbf{r}_i^{(\text{Sch})} = w_i \mathbf{u}_i\left(\hat{\mathbf{B}}, t_{2i}\right)$$

where $\mathbf{u}_i(.)$ is defined in "Pseudo Partial Likelihood and Derivatives ".

Martingale residual

$$r_i^{(\mathrm{M})} = \delta_i - \left(H_0\left(t_{2i}\right) - H_0\left(t_{1i}\right)\right) \exp\left(\mathbf{x}'_i \hat{\mathbf{B}}\right)$$

Deviance residual

$$r_i^{(\mathrm{D})} = sign\left(r_i^{(\mathrm{M})}\right) \sqrt{2\left[-r_i^{(\mathrm{M})} - \delta_i \ln\left(\delta_i - r_i^{(\mathrm{M})}\right)\right]}$$

Cox-Snell residual

$$r_i^{(\mathrm{CS})} = \left(H_0\left(t_{2i}\right) - H_0\left(t_{1i}\right)\right) \exp\left(\mathbf{x}'_i \hat{\mathbf{B}}\right) = \delta_i - r_i^{(\mathrm{M})}$$

Score residual

$$\mathbf{r}_i^{(\mathrm{Sco})} = w_i \mathbf{U}_i\left(\hat{\mathbf{B}}\right)$$

where $\mathbf{U}_i\left(\beta\right)$ is defined in “Properties of Estimates”.

DFBETA

DFBETA that measures the influence of record *i* on parameter estimate is

$$-w_i J_S^{-1}\left(\hat{\mathbf{B}}\right) \mathbf{U}_i\left(\hat{\mathbf{B}}\right)$$

This is approximately the parameter change, $\hat{\mathbf{B}} - \hat{\mathbf{B}}_{(i)}$, where $\hat{\mathbf{B}}_{(i)}$ is the parameter estimate when the *i*th record is omitted.

Aggregated residual

When there are multiple records representing a single subject (as in data with two time variables), residuals can be given for each subject rather than for each record. Except for Schoenfeld’s and deviance residuals, the aggregated residual for a subject is simply the sum of the corresponding record residuals over all the records belonging to the same subject. Please notice that aggregation can only be done for data in the format $\{\mathrm{id}_i, t_{i1}, t_{i2}, \delta_i, \mathbf{x}_i, w_i\}_{i=1}^n$. For Schoenfeld’s residual, the aggregated version is the same as that of the non-aggregated version because Schoenfeld’s residual is only defined for records with $\delta_i = 1$. For deviance residual, the aggregated residual can be derived using the aggregated Martingale residual.

Baseline Hazard Strata

Cox regression can be extended to allow multiple baseline hazard strata (note that these are different from the sample design strata). The baseline hazard strata divide the subjects into disjoint groups, each of which has different baseline hazard function while the regression parameter β stays the same for all baseline hazard strata.

Suppose there are *G* baseline hazard strata. For baseline hazard stratum *g*, the model becomes

$$h_g(t|\mathbf{x}) = h_{0g}(t)\exp\left\{\mathbf{x}'\beta\right\}$$

Let V_g be the set of records belong to baseline hazard stratum *g*. Adding the subscript *g* to a quantity denotes that it is calculated only using data in V_g. For baseline stratum *g*, the previously defined quantities would be $\left\{E_g^{(j)}(\beta,t)\right\}_{j=0}^2$, $\left\{EE_g^{(j)}(\beta,t)\right\}_{j=0}^2$, $\overline{\mathbf{x}}_g(\beta,t)$, $\mathbf{u}_{gi}(\beta,t)$, $I_{gi}(\beta,t)$, $l_{Sg}(\beta)$, $U_{Sg}(\beta)$, $J_{Sg}(\beta)$, $\mathbf{U}_{gi}(\beta)$.

The overall pseudo partial likelihood, its first and second derivatives become $l_S(\beta) = \sum_{g=1}^{G} l_{Sg}(\beta)$, $U_S(\beta) = \sum_{g=1}^{G} U_{Sg}(\beta)$, $J_S(\beta) = \sum_{g=1}^{G} J_{Sg}(\beta)$.

The parameter $\mathbf{B}$ can be estimated by maximizing $l_S(\beta)$ as before. The variance of the parameter estimates and design effects are calculated by the same formulae with the following modifications:

$$\mathbf{U}_j^{(a)}(\beta) = \sum_{i\in\{\mathrm{id}_i=j\}} \mathbf{U}_{k_i i}(\beta)$$

where k_i is the baseline stratum that case *i* belongs to, and the sum is over all cases for subject *j*, no matter which baseline stratum the case is in.

After the regression parameters are estimated, the cumulative hazard and survival functions can be estimated for each baseline stratum separately using the same formula but on data only from that stratum. Let $\hat{H}_g(t|\mathbf{x})$ denote the estimate of stratum *g*'s cumulative hazard function at time *t* for a given predictor $\mathbf{x}$ Its variance calculation is similar as before but with the following changes.

$$\hat{V}\left(\hat{H}_g(t|\mathbf{x})\right) \approx \left(Var\left(\sum_{j=1}^{n_s} w_j q_{g,j}^{(a)}(t|\mathbf{x})\right)\right)_{q_{g,j}^{(a)}(t|\mathbf{x})=\hat{q}_{g,j}^{(a)}(t|\mathbf{x})}$$

where

$$\hat{q}_{g,j}^{(a)}(t|\mathbf{x}) = \sum_{i\in\{\mathrm{id}_i=j\}} \hat{q}_{g,k_i i}(t|\mathbf{x})$$

$$\hat{q}_{g,ki}(t|\mathbf{x}) = v_{gi}(t|\mathbf{x})\, I(i\in V_g) - (A_g(t|\mathbf{x}))'\left(J_S\left(\hat{\mathbf{B}}\right)\right)^{-1}\mathbf{U}_{k_i i}\left(\hat{\mathbf{B}}\right)$$

and $v_{gi}(t\,|\mathbf{x}), A_g(t|\,\mathbf{x})$ are calculated by the same equations as before but only using data from stratum *g*.

Given regression parameters at the estimated values, the residual for each record is calculated based on the data only from the stratum that the record belongs to. If record *i* belongs to stratum *g*, then in its residuals calculation, simply replace $\mathbf{u}_i, H_0, \mathbf{U}_i$ by $\mathbf{u}_{gi}, H_{0g}, \mathbf{U}_{gi}$.

Time-Dependent Predictors

Cox regression can also be extended to allow time dependent predictors, $\mathbf{x} = \mathbf{x}(t)$. The Cox regression model becomes

$$h(t|\mathbf{x}(t)) = h_0(t)\exp\left\{\mathbf{x}'(t)\,\beta\right\}$$

The previously defined equations still apply by simply replacing $\mathbf{x}$ with $\mathbf{x}(t)$ accordingly.

Note: If the values of a time-dependent predictor only depend on time and not the case number, then this predictor will be absorbed in the baseline hazard function. The regression parameter for this predictor is set as redundant.

Predictors

All predictor values for records in the risk set at each failure time are needed in the calculation. Two kinds of time dependent predictors are allowed: piecewise constant predictors, and predictor values that can be calculated at all the needed times.

Piecewise constant predictors

Often the predictors for a subject are measured many times during the study. Between measurements, the predictor value is assumed to be unchanged. Data with two time variables can handle this kind of piecewise constant predictors. For each subject, multiple records with two time variables (see “Input ”) are created, one record for each distinct pattern of the time-dependent measurements. The predictor values are constant for each record. This becomes the two failure time variables with time-independent covariate situation.

Note: it is the user’s responsibility to create the data set of two time variables.

Calculatable predictors

The predictor values can be calculated and hence known at any time point; for example, the age of a subject. The `TIME PROGRAM` command is used for this purpose.

Survival and Cumulative Hazard Functions

For product-limit estimate, solve for $\{\alpha_k\}_{k=1}^{K}$ from:

$$\sum_{i\in D(t_k^*)} w_i \frac{\exp\left(\mathbf{x}_i^{'}\left(t_k^*\right)\hat{\mathbf{B}}\right)}{1-\alpha_k} = \sum_{i\in R(t_k^*)} w_i \exp\left(\mathbf{x}_i^{'}\left(t_k^*\right)\hat{\mathbf{B}}\right)$$

For Breslow estimation:

$$\hat{H}_0\left(t\right) = \sum_{\{k:t_k^*\le t\}} \frac{\sum_{i\in D(t_k^*)} w_i}{\sum_{l\in R(t_k^*)} w_l \exp\left(\mathbf{x}_l^{'}\left(t_k^*\right)\hat{\mathbf{B}}\right)}$$

For Efron estimation:

$$\hat{H}_0\left(t\right) = \sum_{\{k:t_k^*<t\}} \frac{\sum_{i\in D\left(t_k^*\right)} w_i}{d(t_k^*)} \times \sum_{r=0} \frac{1}{\sum_{l\in R\left(t_k^*\right)} w_l \exp\left(\mathbf{x}_l^{'}(t_k^*)\hat{\mathbf{B}}\right) - \frac{r}{d\left(t_k^*\right)} \sum_{l\in D\left(t_k^*\right)} w_l \exp\left(\mathbf{x}_l^{'}(t_k^*)\hat{\mathbf{B}}\right)}$$

Using the fact that $\hat{H}_0\left(t\right)$ and $\hat{S}_0\left(t\right)$ are right continuous step functions with jumps only at observed failure times, then for a given predictor path up to time *T*: $\{\mathbf{x}\left(u\right) : u \le T\}$, the cumulative hazards and survival function are estimated by step functions. For $t \le \min\left(T, \max_i\left(t_{2i}\right)\right)$

$$H\left(t|\{\mathbf{x}\left(u\right) : u\le t\}\right) = \sum_{\{j:t_j^*\le t\}} \left(\hat{H}_0\left(t_j^*\right) - \hat{H}_0\left(t_{j-1}^*\right)\right) \exp\left(\mathbf{x}^{'}\left(t_j^*\right)\hat{\mathbf{B}}\right)$$

$$S\left(t|\{\mathbf{x}\left(u\right) : u\le t\}\right) = \prod_{\{j:t_j^*\le t\}} \left\{\frac{\hat{S}_0\left(t_j^*\right)}{\hat{S}_0\left(t_{j-1}^*\right)}\right\}^{\exp\left(\mathbf{x}^{'}\left(t_j^*\right)\hat{\mathbf{B}}\right)}$$

The variance of $H\left(t|\{\mathbf{x}\left(u\right) : u\le t\}\right)$ can be calculated as in the case without time-dependent predictors, but with the following changes:

$$v_i\left(t|\{\mathbf{x}\left(u\right) : u\le t\}\right) = \frac{\delta_i I\left(t_{2i}\le t\right)}{E^{(0)}\left(\hat{\mathbf{B}},t_{2i}|\mathbf{x}\left(t_{2i}\right)\right)} - \sum_{l=1}^{n} w_l \frac{\delta_l I\left(t_{2l}\le t\right) Y_i\left(t_{2l}\right)\exp\left(\left(\mathbf{x}_i\left(t_{2l}\right)-\mathbf{x}\left(t_{2l}\right)\right)^{'}\hat{\mathbf{B}}\right)}{\left(E^{(0)}\left(\hat{\mathbf{B}},t_{2l}|\mathbf{x}\left(t_{2l}\right)\right)\right)^2}$$

$$A\left(t|\{\mathbf{x}\left(u\right) : u\le t\}\right) = -\sum_{l=1}^{n} w_l \frac{\delta_l I\left(t_{2l}\le t\right) E^{(1)}\left(\hat{\mathbf{B}}, t_{2l}|\mathbf{x}\left(t_{2l}\right)\right)}{\left(E^{(0)}\left(\hat{\mathbf{B}}, t_{2l}|\mathbf{x}\left(t_{2l}\right)\right)\right)^2}$$

$$E^{(0)}\left(\hat{\mathbf{B}}, t|\mathbf{x}\left(t\right)\right) = \sum_{l=1}^{n} w_l Y_l\left(t\right)\exp\left(\left(\mathbf{x}_l\left(t\right) - \mathbf{x}\left(t\right)\right)^{'}\hat{\mathbf{B}}\right)$$

$$E^{(1)}\left(\hat{\mathbf{B}},t|\mathbf{x}\right)=\sum_{l=1}^{n}w_lY_l(t)\left(\mathbf{x}_l(t)-\mathbf{x}(t)\right)\exp\left(\left(\mathbf{x}_l(t)-\mathbf{x}(t)\right)'\hat{\mathbf{B}}\right)$$

There is no agreeable interpretation of the survival function when there are calculatable time-dependent predictors. Survival curves based on a time-dependent covariate must be used with extreme caution.

Residuals

When there are time dependent predictors, all residuals are calculated in the situation where data with two time variables are used to handle the time-dependent predictors. Only Schoenfeld's residual, score residual, and DFBETA are calculated in other situations.

Hypothesis Testing

Contrasts defined as a linear combination of regression parameters can be tested. Given matrix $\mathbf{L}$ with *r* rows and *p* columns, and vector $\mathbf{K}$ with *r* elements, we test the linear hypothesis H_0 : $\mathbf{LB}=\mathbf{K}$ if it is testable. For more information, see the topic "Complex Samples: Model Testing".

Testing Model Assumptions

Tests are performed by considering bigger alternative models involving additional parameters. When fitting alternative models, initial values are set to 0 for all additional parameters and $\beta=\hat{\mathbf{B}}$ for old parameters where $\hat{\mathbf{B}}$ is the previously estimated value of model $h(t|\mathbf{x})=h_0(t)\exp\left\{\mathbf{x}'\beta\right\}$.

If there are baseline hazard strata or time dependent covariates in the original model, then the alternative model should also include them. The only difference between the original and the alternative model is that there are more predictors in the alternative model.

Testing Proportional Hazards

A key assumption of Cox regression is proportional hazards. When predictors are constant, the hazard ratio $\frac{h(t|\mathbf{x}_2)}{h(t|\mathbf{x}_1)}=\exp\left\{(\mathbf{x}_2-\mathbf{x}_1)'\beta\right\}$ is independent of time, so the hazards at different predictor values are proportional. We test the adequacy of the proportional hazards assumption by considering an alternative model with time-dependent coefficients. Suppose that there are *p* predictors, and we are interested in testing the proportional hazard assumption for p^* predictors, assuming the first p^* predictors without loss of generality.

Specific alternative model

Consider the alternative model

$$h(t|\mathbf{x}) = h_0(t)\exp\left\{\mathbf{x}'\beta(t)\right\} = h_0(t)\exp\left\{\mathbf{x}'\beta + \mathbf{z}'(t)\theta\right\}$$

where $\mathbf{z}'(t) = (x_1 g_1(t), \cdots, x_{p^*} g_{p^*}(t))$ is a time dependent predictor vector, and $g_1(t), \cdots, g_{p^*}(t)$ are p^* user-specified functions of time, one for each of the predictors of interest. This is a proportional hazards model with time dependent covariates with parameter vector $\left(\beta', \theta'\right)'$. Fit this model and test $H_0 : \theta = 0$.

For the time functions, the available options are

$$g(t) = \begin{cases} t & \text{identity} \\ \ln t & \text{log} \\ rd(t) & \text{rank} \\ 1 - S_{KM}(t) & \text{KM} \end{cases}$$

where $S_{KM}(t)$ is the Kaplan-Meier estimate of the survival function, and *rd*(*t*) is

$$rd(t) = \begin{cases} 1 & t < t_1^* \\ j & t \in \left[t_{j-1}^*, t_j^*\right) \\ K+1 & t \geq t_K^* \end{cases}$$

For simplicity, we will only allow $g_1(t) = \cdots = g_{p^*}(t) = g(t)$. By default, $p^* = p$ and $g(t) = 1 - S_{KM}(t)$.

Note: When there are baseline strata, *rd*(*t*) and $S_{KM}(t)$ are calculated based on the whole data, not any individual strata.

Subpopulation Estimates

When analyses are requested for a given subpopulation, we perform calculations on the redefined data such that if the *i*th record is not in the subpopulation, then

$$t_{1i} = t_{2i} = 0, \delta_i = 0, \mathbf{x}_i = 0$$

In the estimations of regression parameters and the survival/cumulative hazard functions, this substitution is equivalent to including only the subpopulation elements in the calculations. In the calculation of variance $\hat{I}(\beta)$ and $\hat{I}_{SRS}(\beta)$, this means that $\mathbf{U}_i(\beta) = 0$ if the *i*th record is not in the subpopulation.

Missing Values

List-wise deletion is used to determine which records are used in the analysis. Negative failure times, t_i or t_{1i} or t_{2i}, are considered missing.

References

Binder, D. A. 1992. Fitting Cox's Proportional Hazards Models from Survey Data. *Biometrika*, 79, 139–147.

Collett, D. 2003. *Modelling survival data in medical research*, 2 ed. Boca Raton: Chapman & Hall/CRC.

Grambsch, P., and T. Therneau. 1994. Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81, 515–526.

Kalbfleisch, J. D., and R. L. Prentice. 2002. *The statistical analysis of failure time data*, 2 ed. New York: John Wiley & Sons, Inc.

Lin, D. Y. 2000. On fitting Cox's proportional hazards models to survey data. *Biometrika*, 87, 37–47.

Link, C. L. 1984. Confidence intervals for the survival function using Cox's proportional hazards model with covariates. *Biometrics*, 40, 601–610.

Link, C. L. 1986. Confidence intervals for the survival function in the presence of covariates. *Biometrics*, 42, 219–220.

Therneau, T., and P. Grambsch. 2000. *Modeling Survival Data: Extending the Cox Model*. New York: Springer.

Zhang, D. 2005. "Analysis of Survival Data, lecture notes, Chapter 10." Available at http://www4.stat.ncsu.edu/%7Edzhang2/st745/chap10.pdf.

CSDESCRIPTIVES Algorithms

This document describes the algorithms used in the complex sampling estimation procedure CSDESCRIPTIVES. The data do not have to be sorted.

Complex sample data must contain both the values of the variables to be analyzed and the information on the current sampling design. Sampling design includes the sampling method, strata and clustering information, and inclusion probabilities for all units at every sampling stage. The overall sampling weight must be specified for each observation.

The sampling design specification for CSDESCRIPTIVES may include up to three stages of sampling. Any of the following general sampling methods may be assumed in the first stage: random sampling with replacement, random sampling without replacement and equal probabilities and random sampling without replacement and unequal probabilities. The first two sampling methods can also be specified for the second and the third sampling stage.

Notation

The following notation is used throughout this chapter unless otherwise stated:

H	Number of strata.
n_h	Sampled number of primary sampling units (PSU) per stratum.
f_h	Sampling rate per stratum.
m_{hi}	Number of elements in the *i*th sampled unit in stratum *h*.
w_{hij}	Overall sampling weight for the *j*th element in the *i*th sampled unit in stratum *h*.
$\mathbf{y}_{hij}$	Value of variable *y* for the *j*th element in the *i*th sampled unit in stratum *h*.
Y	Population total sum for variable *y*.
n	Total number of elements in the sample.
N	Total number of elements in the population.

Weights

Overall weights specified for each ultimate element are processed as given. See “Weights ” in *Complex Samples: Covariance Matrix of Total* for more information on weights and variance estimation methods.

Z Expressions

$$z_{hij} = w_{hij} y_{hij}$$

$$z_{hi} = \sum_{j=1}^{m_{hi}} z_{hij}$$

$$\overline{z}_h = \frac{1}{n_h}\sum_{i=1}^{n_h} z_{hi}$$

$$S_h^2 = \frac{1}{n_h-1}\sum_{i=1}^{n_h}\left(z_{hi}-\overline{z}_h\right)^2$$

For multi-stage samples, the index *h* denotes a stratum in the given stage, and *i* stands for unit from *h* in the same stage. The index *j* runs over all final stage elements contained in unit *hi*.

Variable Total

An estimate for the population total of variable *y* in a single-stage sample is the weighted sum over all the strata and all the clusters:

$$\hat{Y} = \sum_{h=1}^{H}\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}} w_{hij} y_{hij}$$

Alternatively, compute the weighted sum over all the elements in the sample:

$$\hat{Y} = \sum_{i=1}^{n} w_i y_i$$

The latter expression is more general because it also applies to multi-stage samples.

Variable Total Variance

For a multi-stage sample containing a with replacement sampling stage, all specifications other than weights are ignored for the subsequent stages. They make no contribution to the variance estimates.

Single Stage Sample

The variance of the total for variable *y* in a single-stage sampling is estimated by the following:

$$\hat{V}\left(\hat{Y}\right) = \hat{V}_1\left(\hat{Y}\right) = \sum_{h=1}^{H} U_h$$

where U_h is an estimated contribution from stratum *h* and depends on the sampling method as follows:

- For sampling with replacement: $U_h = n_h S_h^2$
- For simple random sampling: $U_h = (1 - f_h)\, n_h S_h^2$
- For sampling without replacement and unequal probabilities:
  $$U_h = \sum_{i=1}^{n_h}\sum_{i>j}^{n_h}\left(\frac{\pi_{hi}\pi_{hj}}{\pi_{hij}} - 1\right)\left(z_{hi}-z_{hj}\right)^2$$

π_{hi} and π_{hj} are the inclusion probabilities for units *i* and *j* in stratum *h*, and π_{hij} is the joint inclusion probability for the same units. This estimator is due to Yates and Grundy (1953) and Sen (1953).

For each stratum *h* containing a single element, the variance contribution U_h is always set to zero.

Two-stage Sample

When the sample is obtained in two stages and sampling without replacement is applied in the first stage, use the following estimate for the variance of the total for variable *y*:

$$\hat{V}\left(\hat{Y}\right) = \hat{V}_2\left(\hat{Y}\right) = \hat{V}_1\left(\hat{Y}\right) + \sum_{h=1}^{H}\sum_{i=1}^{n_h}\pi_{hi}\sum_{k=1}^{K_{hi}}U_{hik}$$

where

- π_{hi} is the first stage inclusion probability for the primary sampling unit *i* in stratum *h*. In the case of simple random sampling, the inclusion probability is equal to the sampling rate f_h for stratum *h*.
- K_{hi} is the number of second stage strata in the primary sampling unit *i* within the first stage stratum *h*.
- U_{hik} is a variance contribution from the second stage stratum *k* from the primary sampling unit *hi*. Its value depends on the second stage sampling method; the corresponding formula from “Single Stage Sample ” applies.

Three-stage Sample

When the sample is obtained in three stages where sampling in the first stage is done without replacement and simple random sampling is applied in the second stage, we use the following estimate for the variance of the total for variable *y*:

$$\hat{V}\left(\hat{Y}\right) = \hat{V}_2\left(\hat{Y}\right) + \sum_{h=1}^{H}\sum_{i=1}^{n_h}\pi_{hi}\sum_{k=1}^{K_{hi}}f_{hik}\sum_{j=1}^{n_{hik}}\sum_{l=1}^{L_{hikj}}U_{hikjl}$$

where

- f_{hik} is the sampling rate for the secondary sampling units in the second stage stratum *hik*.
- L_{hikj} is the number of third stage strata in the secondary sampling unit *hikj*.
- U_{hikjl} is a variance contribution from the third stage stratum *l* contained in the secondary sampling unit *hikj*. Its value depends on the second stage sampling method; the corresponding formula from “Single Stage Sample ” applies.

Population Size Estimation

An estimate for the population size corresponds to the estimate for the variable total; it is sum of the sampling weights. We have the following estimate for the single-stage samples:

$$\hat{N} = \sum_{h=1}^{H} \sum_{i=1}^{n_h} \sum_{j=1}^{m_{hi}} w_{hij}$$

More generally,

$$\hat{N} = \sum_{i=1}^{n} w_i$$

The variance of $\hat{N}$ is obtained by replacing y_{hij} with 1; that is, by replacing z_{hij} with w_{hij} in the corresponding variance estimator formula for $\hat{V}\left(\hat{Y}\right)$.

Ratio Estimation

Let *R*=*Y*/*X* be the ratio of the totals for variables *y* and *x*. It is estimated by

$$\hat{R} = \hat{Y}/\hat{X}$$

where $\hat{Y}$ and $\hat{X}$ are the estimates for the corresponding variable totals.

The variance of $\hat{R}$ is approximated using the Taylor linearization formula following Woodruff (1971). The estimate for the approximate variance of the ratio estimate $\hat{V}\left(\hat{R}\right)$ is obtained by replacing z_{hij} with

$$z_{hij} = w_{hij}\left(y_{hij} - \hat{R}x_{hij}\right)/\hat{X}$$

in the corresponding variance estimator $\hat{V}\left(\hat{Y}\right)$.

Mean Estimation

The mean $\overline{Y}$ for the variable *y* is estimated by

$$\hat{\overline{Y}} = \hat{Y}/\hat{N}$$

where $\hat{Y}$ is the estimate for the total of *y* and $\hat{N}$ is the population size estimate.

The variance of the mean is estimated using the ratio formulas, as the mean is a ratio of $\hat{Y}$ and $\hat{N}$. Accordingly, $\hat{V}\left(\hat{\overline{Y}}\right)$ is obtained by substituting z_{hij} with

$$z_{hij} = w_{hij}\left(y_{hij} - \hat{\overline{Y}}\right)/\hat{N}$$

in the corresponding variance estimator $\hat{V}\left(\hat{Y}\right)$.

Domain Estimation

Let the population be divided into *D* domains. For each domain *d* define the following indicator variables:

$$\delta_{hij}(d) = \begin{cases} 1 & \text{if the sample unit } hij \text{ is in the domain } d \\ 0 & \text{otherwise} \end{cases}$$

To estimate a domain population total, domain variable total, ratios and means, substitute y_i with $\delta_i(d)\, y_i$ in the corresponding formula for the whole population as follows:

- Domain variable total: $\hat{Y}_d = \sum_{i=1}^{n} w_i \delta_i(d)\, y_i$
- Domain population total: $\hat{N}_d = \sum_{i=1}^{n} w_i \delta_i(d)$
- Domain variable ratio: $\hat{R}_d = \hat{Y}_d / \hat{X}_d$
- Domain variable mean: $\hat{\bar{Y}}_d = \hat{Y}_d / \hat{N}_d$

Similarly, in order to estimate the variances of the above estimators, substitute y_{hij} with $\delta_{hij}(d)\, y_{hij}$ in the corresponding formula for the whole population. The following substitution of z_{ij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating the variance of:

- Domain variable total: $z_{hij}(d) = \delta_{hij}(d)\, w_{hij} y_{hij}$
- Domain population total: $z_{hij}(d) = \delta_{hij}(d)\, w_{hij}$
- Domain variable ratio: $z_{hij} = \delta_{hij}(d)\, w_{hij}\left(y_{hij} - \hat{R}_d x_{hij}\right) / \hat{X}_d$
- Domain mean: $z_{hij} = \delta_{hij}(d)\, w_{hij}\left(y_{hij} - \hat{\bar{Y}}_d\right) / \hat{N}_d$

Standard Errors

Let *Z* denote any of the population or subpopulation quantities defined above: variable total, population size, ratio or mean. Then the standard error of an estimator $\hat{Z}$ is the square root of its estimated variance:

$$StdError\left(\hat{Z}\right) = \sqrt{\hat{V}\left(\hat{Z}\right)}$$

Coefficient of Variation

The coefficient of variation of the estimator $\hat{Z}$ is the ratio of its standard error and its value:

$$CV\left(\hat{Z}\right) = \frac{SE\left(\hat{Z}\right)}{\hat{Z}}$$

The coefficient of variation is undefined when $\hat{Z} = 0$.

T Tests

Testing the hypothesis that a population quantity *Z* equals θ_0; that is, $H_0 \; : Z = \theta_0$, is performed using the *t* test statistic:

$$t\left(\hat{Z}\right) = \frac{\hat{Z} - \theta_0}{StdError\left(\hat{Z}\right)}$$

The *p*-value for the two-sided test is given by the probability

$$P\left(|T| > \left|t\left(\hat{Z}\right)\right|\right)$$

where *T* is a random variable form the *t* distribution with *df* degrees of freedom.

The number of the degrees of freedom is calculated as the difference between the number of primary sampling units and the number of strata in the first stage of sampling.

Confidence Limits

A level 1−α confidence interval is constructed for a given $0 \le \alpha \le 1$. The confidence bounds are defined as

$$\hat{Z} \pm StdError\left(\hat{Z}\right) t_{df}\left(1 - \alpha/2\right)$$

where $StdError\left(\hat{Z}\right)$ is the estimated standard error of $\hat{Z}$, and $t_{df}\left(1 - \alpha/2\right)$ is the $100\left(1 - \alpha/2\right)$ percentile of the *t* distribution with *df* degrees of freedom.

Design Effects

The design effect *Deff* is estimated by

$$Deff = \frac{\hat{V}\left(\hat{Y}\right)}{\hat{V}_{srs}\left(\hat{Y}_{srs}\right)}$$

$\hat{V}\left(\hat{Y}\right)$ is the estimate of the variance of $\hat{Y}$ under the appropriate sampling design, while $\hat{V}_{srs}\left(\hat{Y}_{srs}\right)$ is the estimate of variance of $\hat{Y}_{srs}$ under the simple random sampling assumption as follows:

$$\hat{V}_{srs}\left(\hat{Y}_{srs}\right) = (fpc)\frac{\hat{N}}{n-1}\sum_{i=1}^{n} w_i\left(y_i - \frac{\hat{Y}}{\hat{N}}\right)^2$$

Assuming sampling without replacement we have $fpc = \left(1 - \frac{n}{\hat{N}}\right)$ given that $\frac{n}{\hat{N}} < 1$, while for sampling with replacement we set $fpc = 1$. This assumption is independent of the sampling specified for the complex sample design based variance $\hat{V}\left(\hat{Y}\right)$.

Whereas design effect is not relevant for estimates of the population size, we do compute the design effects for ratios and means in addition to the totals. The values of variable *y* in $\hat{V}_{srs}$ are then replaced by the linearized values as follows:

- Ratio estimation $\left(y_i - \hat{R}x_i\right)/\hat{X}$
- Mean estimation $\left(y_i - \hat{\overline{Y}}\right)/\hat{N}$

When estimating design effects for domains we use the familiar substitution $\delta_i\ (d)\, y_i$ for y_i in the $\hat{V}_{srs}$ formula in addition to any ratio or mean substitutions.

We also provide the square root of design effect $\sqrt{Deff}$.

Design effects and their applications have been discussed by Kish (1965) and Kish (1995).

References

Cochran, W. G. 1977. *Sampling Techniques*, 3rd ed. New York: John Wiley and Sons.

Hansen, M. H., W. N. Hurwitz, and W. G. Madow. 1953. *Sample Survey Methods and Theory, Volume II Theory*. New York: John Wiley & Sons.

Horwitz, D. G., and D. J. Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47, 663–685.

Kish, L. 1965. *Survey Sampling*. New York: John Wiley and Sons.

Kish, L. 1995. Methods for Design Effects. *Journal of Official Statistics*, 11, 119–127.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

Sen, A. R. 1953. On the estimate of the variance in sampling with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, 5, 55–77.

Wolter, K. M. 1985. *Introduction to variance estimation*. Berlin: Springer-Verlag.

Woodruff, R. S. 1971. A Simple Method for Approximating the Variance of a Complicated Estimate. *Journal of the American Statistical Association*, 66, 411–414.

Yates, F., and P. M. Grundy. 1953. Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society Series B*, 15, 253–261.

CSGLM Algorithms

CSGLM is a procedure for regression analysis as well as analysis of variance and covariance based on complex samples.

Complex sample data must contain both the values of the variables to be analyzed and the information on the current sampling design. Sampling design includes the sampling method, strata and clustering information, inclusion probabilities and the overall sampling weights.

Sampling design specification for CSGLM may include up to three stages of sampling. Any of the following general sampling methods may be assumed in the first stage: random sampling with replacement, random sampling without replacement and equal probabilities and random sampling without replacement and unequal probabilities. The first two sampling methods can also be specified for the second and the third sampling stage.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Total number of elements in the sample.
p	Number of regression parameters in the model.
Y	Dependent variable vector containing values $y_i, i = 1, \ldots, n$.
X	*n*×*p* design matrix. The rows correspond to the observations and the columns to the model parameters. The *i*th row is $\mathbf{x}'_i, i = 1, \ldots, n$.
W	Diagonal matrix with sampling weights $w_i, i = 1, \ldots, n$ on the diagonal.
B	Vector of *p* unknown population parameters.
N	Total number of elements in the population.

Weights

Overall weights specified for each ultimate element are processed as given. See "Weights " in *Complex Samples: Covariance Matrix of Total* for more information on weights and variance estimation methods.

Model Specification

Let the linear model be specified by the equation **Y=Xβ+E**, where **Y** is a vector of observed dependent variable values, **X** is the linear model design matrix, **β** is a vector of model parameters and **E** is a vector of random errors with zero mean. Each column of the design matrix corresponds to a parameter in the model equation. Each parameter corresponds to one of the intercept, factor main effects, factor interaction effects, factor nested effects, covariate effects and factors by covariates interaction effects. For every factor effect level occurring in data there is a separate parameter. This results in an over-parametrized model.

Estimation

Assuming that the entire finite population has been observed, we can obtain the least square parameter estimates for the linear model by solving the following normal equations

$$\mathbf{X}'_N \mathbf{X}_N \beta = \mathbf{X}'_N \mathbf{Y}_N$$

where $\mathbf{X}_N$ and $\mathbf{Y}_N$ denote the design matrix and dependent variable for all elements in the given population. A solution vector for this system, estimating the model parameters **β**, is denoted by **B**. In our analyses we take the established design-based approach concerned with estimating the finite population parameters **B** developed by Kish and Frankel (1974), Fuller (1975), Shah, Holt and Folsom (1977) and others. See Särndal et al. (1992) for an overview.

Estimates for the population matrices $\mathbf{X}'_N \mathbf{X}_N$ and $\mathbf{X}'_N \mathbf{Y}_N$ are given by $\mathbf{X}'\mathbf{W}\mathbf{X}$ and $\mathbf{X}'\mathbf{W}\mathbf{Y}$ respectively. We solve the following set of weighted normal equations

$$\mathbf{X}'\mathbf{W}\mathbf{X}\mathbf{B} = \mathbf{X}'\mathbf{W}\mathbf{Y}$$

where **W** is a diagonal matrix with sampling weights $w_i, i = 1 \ldots n$ on the diagonal. A solution for **B** is then given by the equation

$$\hat{\mathbf{B}} = \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{W}\mathbf{Y}$$

where $\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}$ is a generalized g2 inverse of $\mathbf{X}'\mathbf{W}\mathbf{X}$.

Predicted Values and Residuals

Predicted values for each observation are given by $\hat{y}_i = \mathbf{x}'_i \hat{\mathbf{B}}$.

The vector of residuals **r** is defined as $r_i = y_i - \hat{y}_i, i = 1, \ldots n$.

The residual sum of squares is: $\mathbf{r}'\mathbf{W}\mathbf{r} = \sum_{i=1}^{n} w_i \left(y_i - \mathbf{x}'_i \hat{\mathbf{B}}\right)^2$

Algorithm

Estimation begins with construction of the weighted sum-of-squares and crossed products (SSCP) matrix. Let $\mathbf{z}'_i = \left(\mathbf{x}'_i, y_i\right)$ be the *i*th row of the matrix **Z**. Then the SSCP matrix is computed by

$$\mathbf{Z}'\mathbf{W}\mathbf{Z} = \sum_{i=1}^{n} w_i \mathbf{z}_i \mathbf{z}'_i$$

where $\mathbf{z}_i \mathbf{z}'_i$ is the outer product for the vector $\mathbf{z}_i$. This matrix can be partitioned as follows

$$\mathbf{Z}'\mathbf{W}\mathbf{Z} = \begin{pmatrix} \mathbf{X}'\mathbf{W}\mathbf{X} & \mathbf{X}'\mathbf{W}\mathbf{Y} \\ \mathbf{Y}'\mathbf{W}\mathbf{X} & \mathbf{Y}'\mathbf{W}\mathbf{Y} \end{pmatrix}$$

After applying the sweep operator to the first *p* rows and columns of the matrix above, we obtain the following solution matrix

$$\begin{pmatrix} -\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-} & \hat{\mathbf{B}} \\ \hat{\mathbf{B}}' & \mathbf{r}'\mathbf{W}\mathbf{r} \end{pmatrix}$$

$\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}$ is a generalized g2 inverse of $\mathbf{X}'\mathbf{W}\mathbf{X}$, $\hat{\mathbf{B}}$ is a parameter solution, and $\mathbf{r}'\mathbf{W}\mathbf{r}$ is the residual sum of squares.

When a column of $\mathbf{X}'\mathbf{W}\mathbf{X}$ is found to be dependent on previous columns, the corresponding parameter is treated as redundant. The solution for redundant parameters is set to 0 as well as the corresponding rows and columns in $\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}$.

Variance Estimates

Variances of parameter estimates are computed according to the Taylor linearization method as presented by Binder (1983).

Define the vector $\mathbf{d}_i = \mathbf{x}_i\left(y_i - \mathbf{x}'_i\hat{\mathbf{B}}\right)$ for *i*=1,...,*n* and its total population estimate by

$$\hat{\mathbf{d}}_T = \sum_{i=1}^{n} w_i\mathbf{x}_i\left(y_i - \mathbf{x}'_i\hat{\mathbf{B}}\right)$$

Let $\hat{\mathbf{V}}\left(\hat{\mathbf{d}}_\mathbf{T}\right)$ be its sample design-based covariance matrix. See “Complex Samples: Covariance Matrix of Total” for more information on its computation. Then the covariance matrix of $\hat{\mathbf{B}}$ is estimated by

$$\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right) = \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}\hat{\mathbf{V}}\left(\hat{\mathbf{d}}_T\right)\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}$$

Note: If any diagonal element of $\hat{\mathbf{V}}\left(\hat{\mathbf{d}}_T\right)$ happens to be non-positive due to the use of the Yates-Grundy-Sen estimator, all elements in the corresponding row and column are set to zero.

Subpopulation Estimates

When analyses are requested for a given subpopulation *S*, we redefine $\left(\mathbf{x}'_i, y_i\right)'$ as follows:

$$\left(\mathbf{x}'_i, y_i\right) = \begin{cases} \left(\mathbf{x}'_i, y_i\right) & \text{if the } i\text{th element is in } S \\ (0,\ldots,0) & \text{otherwise} \end{cases}$$

When computing point estimates, this substitution is equivalent to including only the subpopulation elements in the calculations. This is in contrast to computing the variance estimates where all elements in the sample need to be included.

Standard Errors

Let $\hat{B}_i$ denote a non-redundant parameter estimate. Its standard error is the square root of its estimated variance:

$$SE\left(\hat{B}_i\right) = \sqrt{\hat{V}\left(\hat{B}_i\right)}$$

Standard error is undefined for redundant parameters.

Degrees of Freedom

The sample design degrees of freedom ν is used for computing confidence intervals and test statistics below and is calculated as the difference between the number of primary sampling units and the number of strata in the first stage of sampling. Alternatively, ν may be specified by the user.

Confidence Intervals

A level 1−α confidence interval is constructed for a given $0 \le \alpha \le 1$ for each non-redundant model parameter. Confidence bounds are given by

$$\hat{B}_i \pm SE\left(\hat{B}_i\right) t_\nu \left(1-\alpha/2\right)$$

where $t_\nu\left(1-\alpha/2\right)$ is the $100\left(1-\alpha/2\right)$ percentile of the *t* distribution with ν degrees of freedom.

t Tests

The hypothesis test $H_{0i} : \hat{B}_i = 0$ is performed for each non-redundant model parameter using the *t* test statistic:

$$t\left(\hat{B}_i\right) = \frac{\hat{B}_i}{SE\left(\hat{B}_i\right)}$$

The *p*-value for the two-sided test is given by the probability $P\left(|T| > \left|t\left(\hat{B}_i\right)\right|\right)$, where *T* is a random variable from the *t* distribution with ν degrees of freedom.

Design Effects

The design effect for each non-redundant parameter estimate is given by

$$Deff\left(\hat{B}_i\right) = \frac{\hat{V}\left(\hat{B}_i\right)}{\hat{V}_{srs}\left(\hat{B}_i\right)}$$

$\hat{V}\left(\hat{B}_i\right)$ is the estimate of variance of $\hat{B}_i$ under the complex sampling design, while $\hat{V}_{srs}\left(\hat{B}_i\right)$ is the estimate of variance of $\hat{B}_i$ under the simple random sampling assumption. The latter is computed as the *i*th diagonal element of the following matrix:

$$\hat{V}_{srs}\left(\hat{B}_i\right) = \left[\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{d}}_T\right)\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-}\right]_{ii}$$

where

$$\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{d}}_{\mathbf{T}}\right) = (fpc)\,\frac{\hat{N}}{n-1}\sum_{i=1}^{n} w_i\mathbf{d}_i\mathbf{d}'_i$$

with $\mathbf{d}_i$ as specified earlier.

Assuming sampling without replacement we have $fpc = \left(1 - \frac{n}{\hat{N}}\right)$ given that $\frac{n}{\hat{N}} < 1$, while for sampling with replacement we set $fpc = 1$. This assumption is independent of the sampling specified for the complex sample design based variance $\hat{\mathbf{V}}\left(\hat{\mathbf{d}}_{\mathbf{T}}\right)$.

For subpopulation analysis $\mathbf{d}_i = \mathbf{0}$ whenever observation *i* does not belong to a given subpopulation.

We also provide the square root of design effect $\sqrt{Deff}$.

Design effects and their application have been discussed by Kish (1965) and Kish (1995).

Multiple R-square

$$R^2 = 1 - \frac{\mathbf{r}'\mathbf{W}\mathbf{r}}{\left(\mathbf{Y}-\hat{\bar{Y}}_S\mathbf{1}\right)'\mathbf{W}\left(\mathbf{Y}-\hat{\bar{Y}}_S\mathbf{1}\right)}$$

where $\hat{\bar{Y}}_S = \hat{Y}_S/\hat{N}_S$ is the estimated subpopulation mean for variable *Y*.

If the specified model contains no intercept the following expression is used:

$$R^2 = 1 - \frac{\mathbf{r}'\mathbf{W}\mathbf{r}}{\mathbf{Y}'\mathbf{W}\mathbf{Y}}$$

Hypothesis Testing

Given an *r*×*p***L** matrix and *r*×1 **K** vector, CSGLM tests the linear hypothesis $H_0 : \mathbf{LB} = \mathbf{K}$ if **LB** is estimable. The Wald X^2 statistic is given by

$$X^2 = \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)'\left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^{-}\left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)$$

The statistic has an asymptotic chi-square distribution with $r_I = rank\left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)$ degrees of freedom. If $r_I < r$, $\left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^-$ is a generalized inverse such that Wald tests are effective for a restricted set of hypothesis $\mathbf{L}_I\mathbf{B} = \mathbf{K}_I$ containing a particular subset *I* of independent rows from H_0.

Each row l'_i of **L** is also tested separately. The estimate for the *i*th row is given by $l'_i\hat{\mathbf{B}}$ and its standard error by $\sqrt{l'_i\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)l_i}$.

See "Complex Samples: Model Testing" for additional tests and *p*-value adjustments.

Custom Tests

Custom hypothesis tests are conducted only when **L** is such that **LB** is estimable. This condition is verified using the following equality:

$$\mathbf{L} = \mathbf{L}\left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^- \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)$$

Default Tests of Model Effects

For each effect specified in the model, a Type III test **L** matrix is constructed such that **LB** is estimable. It involves parameters only for the given effect and the containing effects and it does not depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable. **K** is always set to **0** when computing the test statistics for model effects.

The hypothesis for the corrected model is that all the parameters except for the intercept are zero.

Estimated Marginal Means

Estimated marginal means (EMMEANS) are based on the estimated cell means. For a given fixed set of factors, or their interactions, we estimate marginal means as the mean value averaged over all cells generated by the rest of the factors in the model. Covariates may be fixed at any specified value. If not specified, the value for each covariate is set to its overall mean estimate.

When missing cells are present in the data, EMMEANS may not be estimable. In such circumstance, we provide a modified estimate proposed by Searle, Speed and Milliken (1980) that ignores the non-estimable cells.

Each marginal estimate is finally constructed in the form $l'\hat{\mathbf{B}}$ such that $l'\hat{\mathbf{B}}$ is estimable.

Comparing EMMEANS

For a given factor in the model, a vector of EMMEANS is created for all levels of the factor. This vector can be expressed in the form $\hat{\mu} = \mathbf{L}\hat{\mathbf{B}}$ where each row of **L** is generated as described above. The variance is then computed by the following formula:

$$\hat{\mathbf{V}}\left(\hat{\mu}\right) = \mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}^{'}$$

A set of contrasts for the factor is created according to the selected contrast type. Let this set of contrasts define the matrix **C** used for testing the hypothesis $H_0 : \mathbf{C}\mu = \mathbf{0}$

The Wald X^2 statistic is used for testing given set of contrasts for the factor as follows:

$$X^2 = \left(\mathbf{C}\hat{\mu}\right)^{'}\left(\mathbf{C}\hat{\mathbf{V}}\left(\hat{\mu}\right)\mathbf{C}^{'}\right)^{-}\left(\mathbf{C}\hat{\mu}\right)$$

The statistic has an asymptotic chi-square distribution with r_I degrees of freedom, where $r_I = rank\left(\mathbf{C}\hat{\mathbf{V}}\left(\hat{\mu}\right)\mathbf{C}^{'}\right)$.

Each row $c^{'}{}_i$ of **C** is also tested separately. The estimate for the *i*th row is given by $c^{'}{}_i\hat{\mu}$ and its standard error by $\sqrt{c^{'}{}_i\hat{\mathbf{V}}\left(\hat{\mu}\right)c_i}$.

See "Complex Samples: Model Testing" for additional tests and *p*-value adjustments. Substitute the following formula for the simple random sampling covariance: $\hat{\mathbf{V}}_{srs}\left(\hat{\mu}\right) = \mathbf{L}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)\mathbf{L}^{'}$.

References

Binder, D. A. 1983. On the variances of asymptotically normal estimators from complex surveys. *International Statistical Review*, 51, 279–292.

Fuller, W. A. 1975. Regression analysis for sample survey. *Sankhya, Series C*, 37, 117–132.

Kish, L. 1965. *Survey Sampling*. New York: John Wiley and Sons.

Kish, L. 1995. Methods for Design Effects. *Journal of Official Statistics*, 11, 119–127.

Kish, L., and M. R. Frankel. 1974. Inference from complex samples. *Journal of the Royal Statistical Society B*, 36, 1–37.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

Searle, S. R., F. M. Speed, and G. A. Milliken. 1980. Population marginal means in the linear model: an alternative to least squares means. *The American Statistician*, 34, 216–221.

Shah, B. V., M. M. Holt, and R. E. Folsom. 1977. Inference about regression models from sample survey data. *Bulletin of the International Statistical Institute*, 67:3, 43–57.

CSLOGISTIC Algorithms

Logistic regression is a commonly used analytical tool for categorical responses. LOGISTIC REGRESSION (for binary response) and NOMREG (for multi-category response) are procedures under the standard sampling setting. This document considers multinomial logistic regression model under the complex sampling setting extending the model in NOMREG to complex sampling.

There are different approaches for analytic inference in complex sampling (Chambers and Skinner 2003). We will take the two-phase sampling and pseudo-likelihood estimation approaches.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	Categorical dependent variable vector containing values y_i, *i*=1,...,*n*.
K	The total number of categories for dependent variable.
$y_i(k)$	Indicator variable for category *k*; $y_i(k) = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{otherwise} \end{cases}$.
X	*n*×*p* design matrix. The rows correspond to the observations and the columns to the model parameters.
π_i	Inclusion probability for case *i*.
w_i	Sampling weight for case *i*, $w_i = 1/\pi_i$.
$p_{\mathbf{x}}(k)$	The probability for response category *k* at **x**: $p_{\mathbf{x}}(k) = \Pr(y = k\|\mathbf{x})$, and denote $p_i(k) = p_{\mathbf{x}_i}(k)$ for case *i*.
N	The number of cases in the whole population.
n	The number of cases in the sample.
B	The parameter of interest, the population or census parameter.

Superpopulation Model

Two phases of sampling are assumed. The first phase generates a finite population by a model or super population. The second phase selects a sample according to a sampling plan from the finite population generated in the first phase.

Model Generating the Population

Assume that the response variable *y* at a given **x** follows a multinomial distribution with probability $p_{\mathbf{x}}(k)$ for *y*=*k*. Without loss of generality, let the last category *K* be the reference category. Then for *k* = 1, …, *K*−1,

$$\log \frac{p_{\mathbf{x}}(k)}{p_{\mathbf{x}}(K)} = \mathbf{x}'\beta_k$$

or

$$p_{\mathbf{x}}(k) = \begin{cases} \dfrac{\exp\left(\mathbf{x}'\beta_k\right)}{1+\sum_{k=1}^{K-1}\exp\left(\mathbf{x}'\beta_k\right)} & k = 1,\cdots,K-1 \\ \dfrac{1}{1+\sum_{k=1}^{K-1}\exp\left(\mathbf{x}'\beta_k\right)} & k = K \end{cases}$$

where $\beta_k = (\beta_{k1},\cdots,\beta_{kp})'$ is the regression parameter vector for response category k.

There are $p(K-1)$ regression parameters in total. This model is described in many books, for example Agresti (2002).

Let **B** denote the MLE of the model parameter **β** based on the whole population. This **B** is also called the census parameter. The parameter of interest is the census parameter **B**, rather than the model parameter **β**. The exact definition and formulation of **B** is described below in the estimating equation.

Parameter Estimation

For a sample drawn from the finite population according to a sample plan, we take the pseudo-likelihood approach. In this approach, the pseudo-likelihood is a sample estimate of the population log-likelihood, and parameter estimates are derived by maximizing the pseudo-likelihood.

From the sample, an unbiased estimate of population log-likelihood l_U is

$$l_S(\beta) = \sum_{i\in S}\sum_{k=1}^{K} w_i y_i(k)\log(p_i(k))$$

We will maximize $l_S(\beta)$ to get the estimates for census parameter **B**. The pseudo-score function is, for $k = 1, \ldots, K-1$,

$$S_S(\beta) = \sum_{i\in S} w_i(\mathbf{y}_i^* - \mathbf{p}_i^*)\otimes\mathbf{x}_i$$

The estimator obtained by solving $S_S(\beta) = 0$ is an estimator of **B**.

Redundant Parameters

In this procedure, the over-parameterization approach is similar to that in the NOMREG procedure. If a parameter is found to be redundant, it is set to zero and will not affect the estimation procedure.

Estimation Algorithm

The Newton-Raphson iterative estimation method is used to solve the estimating equation. Let $\mathbf{B}^{(v)}$ be the parameter estimate at iteration step *v*, the parameter estimate $\mathbf{B}^{(v+1)}$ at iteration step *v* + 1 is updated as

$$\mathbf{B}^{(v+1)} = \mathbf{B}^{(v)} - \xi \cdot J^{-}\left(\mathbf{B}^{(v)}\right) S_S\left(\mathbf{B}^{(v)}\right)$$

where

$$J(\beta) = \frac{\partial S_S(\beta)}{\partial \beta} = -\sum_{i \in S} w_i \left(diag\left(\mathbf{p}_i^*\right) - \mathbf{p}_i^* \left(\mathbf{p}_i^*\right)' \right) \otimes \mathbf{x}_i \mathbf{x}'_i$$

the (k, j)th block element of $J(\beta)$, for k, j = 1, …, K−1, is

$$J_{kj}(\beta) = \frac{\partial S_S(\beta_k)}{\partial \beta_j} = \begin{cases} \sum_{i \in S} w_i p_i(k) p_i(j) \mathbf{x}_i \mathbf{x}'_i & k \neq j \\ -\sum_{i \in S} w_i p_i(j)(1 - p_i(j)) \mathbf{x}_i \mathbf{x}'_i & k = j \end{cases}$$

$J^{-}(\beta)$ is a generalized inverse of $J(\beta)$. The stepping scalar ζ > 0 is used to make $l_S\left(\mathbf{B}^{(v+1)}\right) \geq l_S\left(\mathbf{B}^{(v)}\right)$. Use the step-halving method if $l_S\left(\mathbf{B}^{(v+1)}\right) < l_S\left(\mathbf{B}^{(v)}\right)$. Let t be the maximum number of steps in step-halving; the set of values of ζ is {1/2^r: r = 0, …, t−1}.

Starting with initial values $\mathbf{B}^{(0)}$, iteratively update $\mathbf{B}^{(v+1)}$ until one of the stopping criteria is satisfied. The final estimate is denoted as $\hat{\mathbf{B}}$.

Note: Sometimes, infinite parameters may be present in the model because of complete or quasi-complete separation of the data (Albert and Anderson, 1984) (Santner and Duffy, 1986). In CSLOGISTIC, a check for separation of the data can be performed. If either complete or quasi-complete separation is suggested by the test, a warning is issued and results based on the last iteration are given.

Initial Values

For all non-intercept regression parameters, set their initial values to be zero. For intercepts, if there are any, set for k = 1, …, K−1,

$$B_{k1}^{(0)} = \log\left(\frac{\hat{N}_k}{\hat{N}_K}\right)$$

where $\hat{N}_k = \sum_{i \in S} w_i y_i \; (k)$ is the estimated population number of responses in category k.

Stopping Criteria

Given two convergence criteria $\epsilon_l > 0$ and $\epsilon_p > 0$, the iteration is considered to be converged if one of the following criteria is satisfied:

1. $$\begin{cases} \frac{\left| l_S\left(\mathbf{B}^{(v+1)}\right) - l_S\left(\mathbf{B}^{(v)}\right) \right|}{\left| l_S\left(\mathbf{B}^{(v)}\right) \right| + 10^{-6}} < \epsilon_l & \text{if relative change} \\ \left| l_S\left(\mathbf{B}^{(v+1)}\right) - l_S\left(\mathbf{B}^{(v)}\right) \right| < \epsilon_l & \text{if absolute change} \end{cases}$$
2. $$\begin{cases} \max_{k,j}\left(\frac{\left| B_{kj}^{(v+1)} - B_{kj}^{(v)} \right|}{\left| B_{kj}^{(v)} \right| + 10^{-6}} \right) < \epsilon_p & \text{if relative change} \\ \max_{k,j}\left(\left| B_{kj}^{(v+1)} - B_{kj}^{(v)} \right| \right) < \epsilon_p & \text{if absolute change} \end{cases}$$

:

3. The maximum number of iterations is reached.

Parameter Covariance Matrix

The design-based variance of $\hat{\mathbf{B}}$ (Binder 1983) has estimate

$$\hat{V}\left(\hat{\mathbf{B}}\right) \approx J^{-}\left(\hat{\mathbf{B}}\right)\hat{I}\left(\hat{\mathbf{B}}\right)J^{-}\left(\hat{\mathbf{B}}\right)$$

where $\hat{I}(\beta)$ is the estimate of design based variance of $S_S(\beta)$. Let $\mathbf{d}_i = (\mathbf{y}_i^* - \mathbf{p}_i^*) \otimes \mathbf{x}_i$, then $S_S(\beta) = \sum_{i \in S} w_i(\mathbf{y}_i^* - \mathbf{p}_i^*) \otimes \mathbf{x}_i = \sum_{i \in S} w_i \mathbf{d}_i$ is an estimate for population total of $\mathbf{d}_i$ vectors. See "Complex Samples: Covariance Matrix of Total" for how to calculate the design-based variance matrix for the total.

Confidence Intervals

The confidence interval for a single regression parameter B_{kj} is approximately

$$\left[\hat{B}_{kj} - t_{df,1-\frac{\alpha}{2}}se\left(\hat{B}_{kj}\right), \hat{B}_{kj} + t_{df,1-\frac{\alpha}{2}}se\left(\hat{B}_{kj}\right)\right]$$

where $se\left(\hat{B}_{kj}\right) = \hat{V}\left(\hat{B}_{kj}\right)$ is the estimated standard error of $\hat{B}_{kj}$, and $t_{df,1-\frac{\alpha}{2}}$ is the $100(1-\alpha/2)$ percentile of a *t* distribution with *df* degrees of freedom. The degrees of freedom *df* can be user specified, and defaults to the difference between the number of primary sampling units and the number of strata in the first stage of sampling.

Design Effect

For each parameter B_{kj}, its design effect is the ratio of its variance under the design to its variance under the SRS design,

$$Deff\left(\hat{B}_{kj}\right) = \frac{\hat{V}\left(\hat{B}_{kj}\right)}{\hat{V}_{srs}\left(\hat{B}_{kj}\right)}$$

For SRS design, the variance matrix is

$$V_{SRS}\left(\hat{\mathbf{B}}\right) \approx J^{-}\left(\hat{\mathbf{B}}\right)I_{SRS}\left(\hat{\mathbf{B}}\right)J^{-}\left(\hat{\mathbf{B}}\right)$$

where

$$\hat{I}_{srs}\left(\hat{\mathbf{B}}\right) = \hat{\mathbf{V}}_{srs}\left(S_S\left(\hat{\mathbf{B}}\right)\right) = (fpc)\frac{\hat{N}}{n-1}\sum_{i \in S} w_i \mathbf{d}_i \mathbf{d}'_i$$

$$\hat{N} = \sum_{i \in S} w_i$$

Assuming sampling without replacement we have $fpc = \left(1 - \frac{n}{\hat{N}}\right)$ given that $\frac{n}{\hat{N}} < 1$, while for sampling with replacement we set $fpc = 1$. This assumption is independent of the sampling specified for the complex sample design based variance matrix $\hat{I}(\beta)$.

Pseudo -2 Log-Likelihood

For the model under consideration, the pseudo –2 Log Likelihood is

$$-2l_S\left(\hat{\mathbf{B}}\right)$$

Let the initial model be the intercept-only model if the intercept is in the considered model, or the empty model otherwise. For the initial model, the pseudo –2 Log Likelihood is

$$-2l_S\left(\mathbf{B}^{(0)}\right)$$

where $\mathbf{B}^{(0)}$ is the initial parameter vector used in the iterative estimating procedure.

Pseudo R Squares

Let $L_U(\mathbf{B})$ be the likelihood function for the whole population; that is, $L_U(\mathbf{B}) = \exp\left(l_U(\mathbf{B})\right)$ A sample estimate is $\hat{L}_U(\mathbf{B}) = \exp\left(l_S(\mathbf{B})\right)$.

Cox and Snell’s R Square

$$R^2_{CS} = 1 - \left(\frac{\hat{L}_U\left(\mathbf{B}^{(0)}\right)}{\hat{L}_U\left(\hat{\mathbf{B}}\right)}\right)^{\frac{2}{\hat{N}}} = 1 - \exp\left\{-\frac{-2l_S\left(\mathbf{B}^{(0)}\right) - \left(-2l_S\left(\hat{\mathbf{B}}\right)\right)}{\hat{N}}\right\}$$

Nagelkerke’s R Square

$$R^2_N = \frac{R^2_{\text{CS}}}{1 - \left\{\hat{L}_U\left(\mathbf{B}^{(0)}\right)\right\}^{2/\hat{N}}}$$

McFadden’s R Square

$$R^2_{\text{M}} = 1 - \frac{l_S\left(\hat{\mathbf{B}}\right)}{l_S\left(\mathbf{B}^{(0)}\right)}$$

Hypothesis Tests

Contrasts defined as linear combination of regression parameters can be tested. Given an $r \times p(K-1)$ **L** matrix and $r \times 1$ **K** vector, CSLogistic tests the linear hypothesis H_0 : **LB** = **K**. See “Complex Samples: Model Testing” for details.

Custom Tests

For a user specified **L** and **K**, $H_0 : \mathbf{LB} = \mathbf{K}$ is tested only when **LB** is estimable. Let $\mathbf{L} = (\mathbf{L}_1, \cdots, \mathbf{L}_{K-1})$, where each $\mathbf{L}_k$ is a *r*×*p* matrix. **LB** is estimable if for every

$$\mathbf{L}_k = \mathbf{L}_k\mathbf{H}$$

where $\mathbf{H} = \left(\mathbf{X}'\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{X}$ is a *p*×*p* matrix.

Note: In NOMREG, only block diagonal matrices such as $\mathbf{L} = diag\left(\mathbf{L}^*, \cdots, \mathbf{L}^*\right)$ are considered, where $\mathbf{L}^*$ is a *q*×*p* matrix. Also in NOMREG, testability is not checked.

Default Tests of Model Effects

For each effect specified in the model, a matrix $\mathbf{L} = diag\left(\mathbf{L}^*, \cdots, \mathbf{L}^*\right)$ is constructed and $H_0 : \mathbf{LB} = \mathbf{0}$ is tested. The matrix $\mathbf{L}^*$ is chosen to be the type III test matrix constructed based on matrix $\mathbf{H} = \left(\mathbf{X}'\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{X}$. This construction procedure makes sure that **LB** is estimable. It involves parameters only for the given effect and the effects containing the given effect. It does not depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable.

Predicted Values

For a predictor pattern **x**, the predicted probability of each response category is

$$\hat{p}_{\mathbf{x}}(k) = \begin{cases} \dfrac{\exp\left(\mathbf{x}'\hat{\mathbf{B}}_k\right)}{1+\sum_{k=1}^{K-1}\exp\left(\mathbf{x}'\hat{\mathbf{B}}_k\right)} & k = 1, \cdots, K-1 \\ \dfrac{1}{1+\sum_{k=1}^{K-1}\exp\left(\mathbf{x}'\hat{\mathbf{B}}_k\right)} & k = K \end{cases}$$

The predicted category $c(\mathbf{x})$ is the one with the highest predicted probability; that is

$$c(\mathbf{x}) = arg\max_k \hat{p}_{\mathbf{x}}(k)$$

Equivalently,

$$c(\mathbf{x}) = arg\max_k \left(\mathbf{x}'\hat{\mathbf{B}}_k\right)$$

where $\hat{\mathbf{B}}_K = 0$ is set for the last (reference) response category. This latter formula is less likely to have numerical problems and should be used.

Classification Table

A two-way table with (*i*,*j*)th element being the counts or the sum of weights for the observations whose actual response category is *i* (as row) and predicted response category is *j* (as column) respectively.

Odds Ratio

The ratio of odds at $\mathbf{x}_1$ to odds at $\mathbf{x}_2$ for response category k_1 versus k_2 is

$$or\left(\mathbf{x}_1,\mathbf{x}_2;k_1,k_2\right)=\frac{p_{\mathbf{x}_1}(k_1)/p_{\mathbf{x}_1}(k_2)}{p_{\mathbf{x}_2}(k_1)/p_{\mathbf{x}_2}(k_2)}=\exp\left(\left(\mathbf{x}_1-\mathbf{x}_2\right)'\left(\mathbf{B}_{k_1}-\mathbf{B}_{k_2}\right)\right)$$

For $k_1=k$ and $k_2=K$ (the reference response category), odds ratio is simplified as

$$or\left(\mathbf{x}_1,\mathbf{x}_2;k.K\right)=\exp\left(\left(\mathbf{x}_1-\mathbf{x}_2\right)'\mathbf{B}_k\right)$$

Equation for $or\left(\mathbf{x}_1,\mathbf{x}_2;k.K\right)$ will be the one we use to calculate odds ratios. The estimate and confidence interval for $or\left(\mathbf{x}_1,\mathbf{x}_2;k.K\right)$ are respectively

$$\exp\left(\left(\mathbf{x}_1-\mathbf{x}_2\right)'\hat{\mathbf{B}}_k\right)$$

and

$$\left[\exp\left(\hat{C}-t_{df,1-\frac{\alpha}{2}}se\left(\hat{C}\right)\right),\exp\left(\hat{C}+t_{df,1-\frac{\alpha}{2}}se\left(\hat{C}\right)\right)\right]$$

where

$$\hat{C}=\left(\mathbf{x}_1-\mathbf{x}_2\right)'\hat{\mathbf{B}}_k$$

$$se\left(\hat{C}\right)=\sqrt{\left(\mathbf{x}_1-\mathbf{x}_2\right)'Var\left(\hat{\mathbf{B}}_k\right)\left(\mathbf{x}_1-\mathbf{x}_2\right)}$$

exp(B)

$\exp\left(B_{kj}\right)$ can be interpreted as an odds ratio for main effects model. SUDAAN calls $\exp\left(B_{kj}\right)$ the odds ratio for parameter B_{kj} whether or not there is an interaction effect in the model. Even though they may not be odds ratios for models with interaction effects, they are still of interest. For each $\exp\left(B_{kj}\right)$, its 1−α confidence interval is

$$\left[\exp\left(L\left(\hat{B}_{kj}\right)\right),\exp\left(U\left(\hat{B}_{kj}\right)\right)\right]$$

where $L\left(\hat{B}_{kj}\right),U\left(\hat{B}_{kj}\right)$ are the lower and upper confidence limits for census parameter B_{kj}.

Subpopulation Estimates

When analyses are requested for a given subpopulation *D*, we perform calculations on the following redefined $\mathbf{x}_i$ and $y_i\left(k\right)$:

$$\mathbf{x}_i = \mathbf{x}_i \delta_i\left(D\right)$$
$$y_i\left(k\right) = y_i\left(k\right) \delta_i\left(D\right)$$

where

$$\delta_i\left(D\right) = \begin{cases} 1 & \text{if the sample unit } i \text{ is in the subpopulation D} \\ 0 & \text{otherwise} \end{cases}$$

When computing point estimates, this substitution is equivalent to including only the subpopulation elements in the calculations. This is in contrast to computing the variance estimates where all elements in the sample need to be included.

Missing Values

Missing values are handled using list-wise deletion; that is, any case without valid data on any design, dependent, or independent variable is excluded from the analysis.

References

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Albert, A., and J. A. Anderson. 1984. On the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, 71, 1–10.

Binder, D. A. 1983. On the variances of asymptotically normal estimators from complex surveys. *International Statistical Review*, 51, 279–292.

Chambers, R., and C. Skinner, eds. 2003. *Analysis of Survey Data*. New York: John Wiley& Sons.

Santner, T. J., and E. D. Duffy. 1986. A Note on A. Albert and J.A. Anderson's Conditions for the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, 73, 755–758.

CSORDINAL Algorithms

Complex Samples Ordinal Regression is a procedure for the analysis of ordinal responses using cumulative link models and allowing for both categorical and continuous predictors. Models specify threshold parameters associated with different response categories in addition to regression parameters associated with model predictors.

Complex sample data must contain both the values of the variables to be analyzed and the information on the current sampling design. Sampling design includes the sampling method, strata and clustering information, inclusion probabilities and the overall sampling weights.

Sampling design specification for Complex Samples Ordinal Regression may include up to three stages of sampling. Any of the following general sampling methods may be assumed in the first stage: random sampling with replacement, random sampling without replacement and equal probabilities and random sampling without replacement and unequal probabilities. The first two sampling methods can also be specified for the second and the third sampling stage.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Total number of complete records or cases in the dataset.
w_i	Overall sampling weight for each sample element in the *i*th record, $i=1,...,n$.
K	The number of values for the ordinal response variable, $K>1$.
Y	The ordinal response variable taking values coded into integers between 1 and K.
θ	Vector of $K-1$ population threshold parameters in the cumulative link model.
β	Vector of p population regression parameters associated with model predictors.
B	Vector of all model parameters $B=(\theta^T, \beta^T)^T$.
X	$n\times p$ design matrix. The rows correspond to the records and the columns to the model regression parameters. The *i*th row is $\mathbf{x}_i^T, i=1,\ldots,n$.
π_{ik}	Conditional response probability for category given observed independent variable vector $\mathbf{x}_i$; that is, $\pi_{ik}=P(Y=k\mid\mathbf{x}_i)$.
γ_{ik}	Conditional cumulative response probability for category given observed independent variable vector $\mathbf{x}_i$; that is, $\gamma_{ik}=P(Y\leq k\mid\mathbf{x}_i)$.
N	Total number of elements in the population: $\hat{N}=\sum_{i=1}^{n} w_i$.

Weights

Overall weights specified for each ultimate element are processed as given. See “Weights ” in *Complex Samples: Covariance Matrix of Total* for more information on weights and variance estimation methods.

Cumulative Link Model

Cumulative link models support regression of a single categorical dependent variable on a set of categorical or continuous independent variables. The dependent variable Y is assumed to be ordinal. Its values have an intrinsic linear ordering and correspond to consecutive integers from 1 to K. The cumulative link model links the conditional cumulative probabilities $P(Y \leq k|\mathbf{x}_i), k = 1, \ldots, K-1$ to a linear predictor. Threshold parameters $\theta_1 < \theta_2 < \cdots < \theta_{K-1}$are assumed different for each cumulative probability, but the vector of regression parameters $\beta = (\beta_1, \ldots, \beta_p)'$ remains the same. The cumulative link model is given by the following set of equations:

$$link(P(Y \leq k|\mathbf{x}_i)) = \theta_k - \beta'\mathbf{x}_i$$

Cumulative link function is specified as an inverse of a cumulative probability distribution function as follows:

$$\begin{cases} link(\gamma_{i,k}) = & \begin{array}{ll} \log(\gamma_{i,k}/(1-\gamma_{i,k})) & \text{Logit link} \\ \log(-\log(1-\gamma_{i,k})) & \text{Complementary log-log link} \\ -\log(-\log(\gamma_{i,k})) & \text{Negative log-log link} \\ \Phi^{-1}(\gamma_{i,k}) & \text{Probit link} \\ \tan(\pi(\gamma_{i,k}\text{-}0.5)) & \text{Cauchit link} \end{array} \\ \text{where } \gamma_{i,k} = P(Y \leq k|\mathbf{x}_i) \text{ for } k\text{=}1,\ldots,K\text{−}1. \end{cases}$$

Vector $\mathbf{x}_i$ denotes a linear model design matrix row matching the vector of regression parameters β. Each parameter corresponds to one of the factor main effects, factor interaction effects, factor nested effects, covariate effects and factors by covariates interaction effects. For every factor effect level occurring in data there is a separate parameter. This results in an over-parametrized model.

Cumulative link models gained popularity after the publication by McCullagh (1980). Further details and examples of these models are given in Agresti (2002).

Estimation

Assuming that the entire finite population $U = \{y_i, \mathbf{x}_i\}_{i=1}^N$ has been observed, we can obtain the maximum likelihood population parameter estimates for the cumulative model by maximizing the following multinomial log-likelihood function

$$l(\theta, \beta) = \log \prod_{i=1}^{N} \prod_{k=1}^{K} (\pi_{i,k})^{y_{i,k}} = \sum_{i=1}^{N} \sum_{k=1}^{K} y_{i,k} \log \pi_{i,k}$$

where we define indicator variables

$$y_{i,k} = \begin{cases} 1 & \text{if } y_i = k \\ 0 & \text{otherwise} \end{cases}$$

and model probabilities

$$\pi_{i,k} = P(Y = k|\mathbf{x}_i) = \gamma_{i,k} - \gamma_{i,k-1}, k = 1, \ldots, K \text{ with } \gamma_{i,0} = 0, \gamma_{i,K} = 1.$$

Taking the first derivatives of the log-likelihood function with respect to the model parameters (θ, β) and setting them equal to zero, we obtain a set of estimating equations for the population model. A solution vector for this set of equations is denoted by (θ_N, β_N). We follow the established design-based approach concerned with estimating the implicit finite population parameters as described by Binder (1983). Population totals in the estimating equations are replaced by their sample-based estimates. A solution for the sample-based estimating equations provides estimates for the population parameters $\left(\hat{\theta}_N, \hat{\beta}_N\right)$ and these are the estimates that we will consider in our analysis. For simplicity, we shall still denote them by $\left(\hat{\theta}, \hat{\beta}\right)$.

An equivalent approach for obtaining the estimates $\left(\hat{\theta}, \hat{\beta}\right)$ is the pseudo-maximum likelihood method where we maximize the sample-based estimate of the log-likelihood given as follows:

$$\hat{l}(\theta, \beta) = \sum_{i=1}^{n} \sum_{k=1}^{K} w_i y_{i,k} \log \pi_{i,k}$$

See Särndal et al. (1992) for an overview of designed-based approach in modeling survey data.

Predicted probabilities

Given a predictor design vector , the model-predicted probability for each response category is

$$\pi_{i,k} = \gamma_{i,k} - \gamma_{i,k-1}$$

Where

$$\gamma_{i,k} = \begin{cases} 0 & k = 0 \\ link^{-1}\left(\theta_k - \beta' \mathbf{x}_i\right) & k = 1, \ldots, K-1 \\ 1 & k = K \end{cases}$$

Let $\eta_{i,k} = \theta_k - \beta' \mathbf{x}_i$. The inverse of the link function; that is, the corresponding cumulative distribution function is given by the following formulas:

$$link^{-1}(\eta_{i,k}) = \begin{array}{ll} \exp(\eta_{i,k})/(1+\exp(\eta_{i,k})) & \text{for Logistic link} \\ 1-\exp(-\exp(\eta_{i,k})) & \text{for Complementary log-log link} \\ \exp(-\exp(-\eta_{i,k})) & \text{for Negative log-log link} \\ \Phi(\eta_{i,k}) & \text{for Probit link} \\ 0.5+\arctan(\eta_{i,k})/\pi & \text{for Cauchit link} \end{array}$$

Estimating equations

Sample-based estimating equations for the population parameters are given by

$$\hat{S}(\theta, \beta) = \frac{\partial \hat{l}(\theta, \beta)}{\partial(\theta, \beta)} = 0$$

$$\frac{\partial \hat{l}}{\partial \theta_k} = \sum_{i=1}^{n} w_i \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \left(\frac{y_{i,k}}{\pi_{i,k}} - \frac{y_{i,k+1}}{\pi_{i,k+1}} \right) = 0, k = 1, \ldots, K-1$$

and

$$\frac{\partial \hat{l}}{\partial \beta_t} = \sum_{i=1}^{n}\sum_{k=1}^{K} -w_i \left(\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} - \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}}\right)\frac{y_{i,k}}{\pi_{i,k}} x_{i,t} = 0, t = 1, \ldots, p$$

where

$$\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} = \begin{cases} \gamma_{i,k}(1-\gamma_{i,k}) & \text{Logit link} \\ -(1-\gamma_{i,k})\log(1-\gamma_{i,k}) & \text{Complementary log-log link} \\ -\gamma_{i,k}\log(\gamma_{i,k}) & \text{Negative log-log link} \\ \phi(\Phi^{-1}(\gamma_{i,k})) & \text{Probit link} \\ \cos^2(\pi(\gamma_{i,k}-0.5))/\pi & \text{Cauchit link} \end{cases}$$

for k=1,...,K−1, and by the definition $\frac{\partial \gamma_{i,0}}{\partial \eta_{i,0}} = \frac{\partial \gamma_{i,K}}{\partial \eta_{i,K}} = 0$. Note that if

$\gamma_{i,k} = 0$ or $\gamma_{i,k} = 1$ then $\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} = 0$ for all link functions.

Second derivatives

The matrix of the first derivatives of the estimated scores $\hat{\mathbf{S}}(\theta,\beta)$ is denoted by $\hat{\mathbf{J}}_0(\theta,\beta)$ and its elements are given by the following expressions:

$$\frac{\partial^2 \hat{l}}{\partial \theta_{k-1}\partial \theta_k} = \sum_{i=1}^{n} w_i \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}}\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}}\frac{y_{i,k}}{\pi_{i,k}^2}, k = 2, \ldots, K-1$$

$$\frac{\partial^2 \hat{l}}{\partial \theta_k^2} = \sum_{i=1}^{n} w_i \left[\frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2}\left(\frac{y_{i,k}}{\pi_{i,k}} - \frac{y_{i,k+1}}{\pi_{i,k+1}}\right) - \left(\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}}\right)^2\left(\frac{y_{i,k}}{\pi_{i,k}^2} + \frac{y_{i,k+1}}{\pi_{i,k+1}^2}\right)\right], k = 1, \ldots, K-1$$

$$\frac{\partial^2 \hat{l}}{\partial \theta_j \partial \theta_k} = 0, \text{ for } |j-k| > 1$$

$$\frac{\partial^2 \hat{l}}{\partial \theta_k \partial \beta_t} = -\sum_{i=1}^{n} w_i \left(\frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2}\pi_{i,k} - \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}}\left(\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} - \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}}\right)\right)\frac{y_{i,k}}{\pi_{i,k}^2}x_{i,t} +$$
$$\sum_{i=1}^{n} w_i \left(\frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2}\pi_{i,k+1} - \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}}\left(\frac{\partial \gamma_{i,k+1}}{\partial \eta_{i,k+1}} - \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}}\right)\right)\frac{y_{i,k+1}}{\pi_{i,k+1}^2}x_{i,t}, k = 1, \ldots K-1, t = 1, \ldots, p$$

$$\frac{\partial^2 \hat{l}}{\partial \beta_t \partial \beta_u} = \sum_{i=1}^{n}\sum_{k=1}^{K} w_i \left[\left(\frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2} - \frac{\partial^2 \gamma_{i,k-1}}{\partial \eta_{i,k-1}^2}\right)\pi_{i,k} - \left(\frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} - \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}}\right)^2\right]\frac{y_{i,k}}{\pi_{i,k}^2}x_{i,t}x_{i,u}$$

$t, u = 1, \ldots, p$

Second derivatives of the cumulative distribution functions are given by

$$\frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2} = \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \times \begin{cases} 1-2\gamma_{i,k} & \text{Logit link} \\ 1+\log(1-\gamma_{i,k}) & \text{Complementary log-log link} \\ -(1+\log(\gamma_{i,k})) & \text{Negative log-log link} \\ -\Phi^{-1}(\gamma_{i,k}) & \text{Probit link} \\ \sin(2\pi\gamma_{i,k}) & \text{Cauchit link} \end{cases}$$

for k=1,...,K−1, and by the definition $\frac{\partial^2 \gamma_{i,0}}{\partial \eta_{i,0}^2} = \frac{\partial^2 \gamma_{i,K}}{\partial \eta_{i,K}^2} = 0$.

Expected second derivatives

The matrix of the expected first derivatives of the estimated scores $\hat{\mathbf{S}}(\theta,\beta)$ is denoted by $\hat{\mathbf{J}}_1(\theta,\beta)$ and its elements are given by the following expressions:

$$\frac{\partial^2 \hat{l}}{\partial\theta_{k-1}\partial\theta_k} = \sum_{i=1}^{n} w_i \frac{\partial\gamma_{i,k-1}}{\partial\eta_{i,k-1}} \frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}} \frac{1}{\pi_{i,k}}, k = 2,\ldots,K-1$$

$$\frac{\partial^2 \hat{l}}{\partial\theta_k^2} = -\sum_{i=1}^{n} w_i \left(\frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}}\right)^2 \left(\frac{1}{\pi_{i,k}} + \frac{1}{\pi_{i,k+1}}\right), k = 1,\ldots,K-1$$

$$\frac{\partial^2 \hat{l}}{\partial\theta_j\partial\theta_k} = 0, \text{ for } |j-k| > 1$$

$$\frac{\partial^2 \hat{l}}{\partial\theta_k\partial\beta_t} = \sum_{i=1}^{n} w_i \left[\left(\frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}} - \frac{\partial\gamma_{i,k-1}}{\partial\eta_{i,k-1}}\right)\frac{1}{\pi_{i,k}} - \left(\frac{\partial\gamma_{i,k+1}}{\partial\eta_{i,k+1}} - \frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}}\right)\frac{1}{\pi_{i,k+1}}\right] \frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}} x_{i,t}$$

$$k = 1,\ldots K-1, t = 1,\ldots,p$$

$$\frac{\partial^2 \hat{l}}{\partial\beta_t\partial\beta_u} = \sum_{i=1}^{n}\sum_{k=1}^{K} -w_i \left(\frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}} - \frac{\partial\gamma_{i,k-1}}{\partial\eta_{i,k-1}}\right)^2 \frac{1}{\pi_{i,k}} x_{i,t} x_{i,u}, t,u = 1,\ldots,p$$

When conducting an analysis for a subpopulation *D*, only records that belong to the subpopulation enter the summation in all of the above derivatives formulas.

Redundant parameters

Due to our use of the over-parametrized model where there is a separate parameter for every factor effect level occurring in the data, the columns of the design matrix are often dependent. Collinearities among continuous variables in the data can also occur. To establish the dependencies in the design matrix we examine columns of $(1,-\mathbf{X})'(1,-\mathbf{X})$ using the sweep operator. When a column is found to be dependent on previous columns, the corresponding parameter is treated as redundant. The solution for redundant parameters is fixed at zero.

Parameter estimation

The vector of estimates of the population model parameters is obtained as a solution $\hat{\mathbf{B}} = (\hat{\theta},\hat{\beta})$ of the sample-based estimating equations. It is computed using the Newton-Raphson method, Fisher scoring or a hybrid method. The hybrid method consists of applying Fisher scoring steps for a specified number of iterations before switching to Newton-Raphson steps. The iteration step is described as follows. Given a vector of parameter estimates $\mathbf{B}^{(\nu)}$ at iteration step ν, the parameters $\mathbf{B}^{(\nu+1)}$ at iteration step $\nu+1$ are computed by solving the following equation:

$$\hat{\mathbf{J}}\left(\mathbf{B}^{(\nu)}\right)\mathbf{B}^{(\nu+1)} = \hat{\mathbf{J}}\left(\mathbf{B}^{(\nu)}\right)\mathbf{B}^{(\nu)} - \xi \cdot \hat{\mathbf{S}}\left(\mathbf{B}^{(\nu)}\right)$$

where

$$\hat{\mathbf{J}}(\mathbf{B}) = \begin{cases} \hat{\mathbf{J}}_0(\mathbf{B}) & \text{for Newton-Raphson step} \\ \hat{\mathbf{J}}_1(\mathbf{B}) & \text{for Fisher scoring step} \end{cases}$$

The stepping scalar $\xi > 0$ is used to ensure that $\hat{l}\left(\mathbf{B}^{(\nu+1)}\right) \geq \hat{l}\left(\mathbf{B}^{(\nu)}\right)$ and that $\pi_{ik} > 0$ if $y_{ik} = 1$ for every *i*. Use step-halving $\xi = 1/2^{\mu}, \mu = 0,\ldots,M-1$ until these conditions are satisfied or the maximum number of steps in step-halving *M* is reached.

Starting with initial values $\mathbf{B}^{(0)}$, iteratively update estimates $\mathbf{B}^{(v+1)}$ until one of the stopping criteria is satisfied. The final vector of estimates is denoted by $\hat{\mathbf{B}}$.

Initial values

Let $\hat{N}_k = \sum_{i=1}^{n} w_i y_{ik}$ be the estimated population number of responses in category $k = 1, \ldots, K$, and $\hat{N} = \sum_{i=1}^{n} w_i$ be the estimated population size. Initial thresholds are then computed according to the following formula:

$$\theta_k^{(0)} = link\left(\frac{\sum_{j=1}^{k} \hat{N}_j}{\hat{N}}\right) \text{ for } k=1,\ldots,K-1$$

Initial values for all regression parameters are set to zero, i.e. $\beta_t^{(0)} = 0$ for t=1,...,p.

Stopping Criteria

Given two convergence criteria $\epsilon_l > 0$ and $\epsilon_p > 0$, the iteration is considered to have converged if criterion 1 or 2 is satisfied, and it stops if any of the following criteria are satisfied:

1. Pseudo-likelihood criterion

$$\begin{cases} \frac{\left|\hat{l}\left(\mathbf{B}^{(v+1)}\right)-\hat{l}\left(\mathbf{B}^{(v)}\right)\right|}{\left|\hat{l}\left(\mathbf{B}^{(v)}\right)\right|+10^{-6}} < \epsilon_l & \text{if relative change} \\ \left|\hat{l}\left(\mathbf{B}^{(v+1)}\right) - \hat{l}\left(\mathbf{B}^{(v)}\right)\right| < \epsilon_l & \text{if absolute change} \end{cases}$$

2. Parameter criterion

$$\begin{cases} \max_i\left(\frac{\left|B_i^{(v+1)}-B_i^{(v)}\right|}{\left|B_i^{(v)}\right|+10^{-6}}\right) < \epsilon_p & \text{if relative change} \\ \max_i\left(\left|B_i^{(v+1)} - B_i^{(v)}\right|\right) < \epsilon_p & \text{if absolute change} \end{cases}$$

3. The maximum number of iteration, or steps in step-halving is reached.

4. Complete or quasi-complete separation of the data is established.

Depending on user's choice, either relative or absolute change (default) is considered in criterion 1 and 2.

If the hybrid algorithm converges with Fisher scoring step, the iterations continue with Newton-Raphson steps.

Variance estimates

Variances of parameter estimates are computed according to the Taylor linearization method as suggested by Binder (1983). Define vector $\mathbf{s}_i\left(\hat{\mathbf{B}}\right)$ of size $(K-1+t)$ to be the contribution of the *i*th element to the estimating equations as follows:

$$s_i^{(k)}\left(\hat{\mathbf{B}}\right)=\frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}}\left(\frac{y_{i,k}}{\pi_{i,k}}-\frac{y_{i,k+1}}{\pi_{i,k+1}}\right), k=1,\ldots,K-1$$

and

$$s_i^{(K-1+t)}\left(\hat{\mathbf{B}}\right)=\sum_{k=1}^{K}-\left(\frac{\partial\gamma_{i,k}}{\partial\eta_{i,k}}-\frac{\partial\gamma_{i,k-1}}{\partial\eta_{i,k-1}}\right)\frac{y_{i,k}}{\pi_{i,k}}x_{i,t}, t=1,\ldots,p$$

so that

$$\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)=\sum_{i=1}^{n}w_i\mathbf{s}_i\left(\hat{\mathbf{B}}\right).$$

The above sum is to be considered as an estimate for the population total of the vectors $\mathbf{s}_i\left(\hat{\mathbf{B}}\right)$. Its sample design-based covariance matrix is denoted by $\hat{\mathbf{V}}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)$. See “Complex Samples: Covariance Matrix of Total” for more information on its computation. Then the covariance matrix of $\hat{\mathbf{B}}$ is estimated by

$$\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)=\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)^{-}\hat{\mathbf{V}}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)^{-}$$

where $\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)^{-}$ is a generalized inverse of $\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)$.

Note: If any diagonal element of $\hat{\mathbf{V}}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)$ happens to be non-positive due to the use of Yates-Grundy-Sen estimator, all elements in the corresponding row and column are set to zero.

Subpopulation estimates

When analyses are requested for a given subpopulation *D*, we redefine $\left(y_i,\mathbf{x}'_i\right)'$ as follows:

$$\left(y_i,\mathbf{x}'_i\right)=\begin{cases}\left(y_i,\mathbf{x}'_i\right) & \text{if the } i \text{ the record is in } D\\ (0,\ldots,0) & \text{otherwise}\end{cases}$$

This is to ensure that the contribution to estimates of every element not in subpopulation *D* is zero. When computing point estimates, this substitution is equivalent to including only the subpopulation elements in the calculations. This is in contrast to computing the variance estimates where all elements in the sample need to be included.

Standard Errors

Let $\hat{B}_i$ denote a non-redundant parameter estimate. Its standard error is the square root of its estimated variance:

$$SE\left(\hat{B}_i\right) = \sqrt{\hat{V}\left(\hat{B}_i\right)}$$

Standard error is undefined for redundant parameters.

Degrees of Freedom

The number of the degrees of freedom *df* used for computing confidence intervals and test statistics below is calculated as the difference between the number of primary sampling units and the number of strata in the first stage of sampling. We shall also refer to this quantity as the sample design degrees of freedom. Alternatively, *df* may be specified by the user.

Confidence Intervals

A level 1−α confidence interval is constructed for a given $0 \le \alpha \le 1$ for each non-redundant model parameter $\hat{B}_i$. Confidence bounds are given by

$$\hat{B}_i \pm SE\left(\hat{B}_i\right) t_{df}\left(1 - \alpha/2\right)$$

where $SE\left(\hat{B}_i\right)$ is the estimated standard error of $\hat{B}_i$, and $t_{df}\left(1 - \alpha/2\right)$ is the $100\left(1 - \alpha/2\right)$ percentile of *t* distribution with *df* degrees of freedom.

t Tests

Testing hypothesis $H_{0i} : \hat{B}_i = 0$ for each non-redundant model parameter $\hat{B}_i$ is performed using the *t* test statistic:

$$t\left(\hat{B}_i\right) = \frac{\hat{B}_i}{SE\left(\hat{B}_i\right)}$$

The *p*-value for the two-sided test is given by the probability $P\left(|T| > \left|t\left(\hat{B}_i\right)\right|\right)$, where *T* is a random variable from the *t* distribution with *df* degrees of freedom.

Design Effects

The design effect $Deff\left(\hat{B}_i\right)$ for non-redundant parameter estimate $\hat{B}_i$ is given by

$$Deff\left(\hat{B}_i\right) = \frac{\hat{V}\left(\hat{B}_i\right)}{\hat{V}_{srs}\left(\hat{B}_i\right)}$$

Design effects are undefined for redundant parameters.

$\hat{V}\left(\hat{B}_i\right)$ is the estimate of variance of $\hat{B}_i$ under the appropriate sampling design, while $\hat{V}_{srs}\left(\hat{B}_i\right)$ is the estimate of variance of $\hat{B}_i$ under the simple random sampling assumption. The latter is computed as the *i*th diagonal element of the following matrix:

$$\hat{V}_{srs}\left(\hat{B}_i\right) = \left[\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)^{-}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)\hat{\mathbf{J}}\left(\hat{\mathbf{B}}\right)^{-}\right]_{ii}$$

$\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)$ can be computed by the following formula:

$$\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right) = (fpc)\,\frac{\hat{N}}{n-1}\sum_{i=1}^{n} w_i \mathbf{s}_i \mathbf{s}'_i$$

with $\mathbf{s}_i$ as specified earlier and $\hat{N}$ being an estimate of the population size.

Assuming sampling without replacement we have $fpc = \left(1 - \frac{n}{\hat{N}}\right)$ given that $\frac{n}{\hat{N}} < 1$, while for sampling with replacement we set $fpc = 1$. This assumption is independent of the sampling specified for the complex sample design based variance $\hat{\mathbf{V}}\left(\hat{\mathbf{S}}\left(\hat{\mathbf{B}}\right)\right)$.

For subpopulation analysis we have that $\mathbf{s}_i = \mathbf{0}$ whenever record does not belong to a given subpopulation.

We also provide the square root of design effects. Note that the square root of design effect *Deff*, computed without finite population correction, has been commonly denoted by *Deft* following paper by Kish (1995). Design effects and their application have been discussed since introduction by Kish (1965).

Linear combinations

Given a constant vector *l* of the same size as the vector of parameter estimates $\hat{\mathbf{B}}$, we compute variance estimates for the linear combination $l'\hat{\mathbf{B}}$ by the formulas:

$$\hat{V}\left(l'\hat{\mathbf{B}}\right) = l'\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)l$$

and

$$\hat{V}_{srs}\left(l'\hat{\mathbf{B}}\right) = l'\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)l$$

Design effect $Deff\left(l'\hat{\mathbf{B}}\right)$ for the linear combination $l'\hat{\mathbf{B}}$ is then given by

$$Deff\left(l'\hat{\mathbf{B}}\right) = \frac{\hat{V}\left(l'\hat{\mathbf{B}}\right)}{\hat{V}_{srs}\left(l'\hat{\mathbf{B}}\right)}$$

Pseudo -2 Log Likelihood

For the model under consideration, the sample-based estimate of the population –2 Log Likelihood is

$$-2\hat{l}\left(\hat{\mathbf{B}}\right)$$

For initial model, the estimate of the –2 Log Likelihood is

$$-2\hat{l}\left(\mathbf{B}^{(0)}\right)$$

where $\mathbf{B}^{(0)}$ is the initial parameter values used in iterative estimating procedure.

Pseudo R Squares

Let $L\left(\mathbf{B}\right)$ be the likelihood function for the whole population, that is, $L\left(\mathbf{B}\right) = \exp\left(l\left(\mathbf{B}\right)\right)$. A sample-based estimate of it is $\hat{L}\left(\mathbf{B}\right) = \exp\left(\hat{l}\left(\mathbf{B}\right)\right)$.

Cox and Snell's R Square

$$R^2_{CS} = 1 - \left(\frac{\hat{L}\left(\mathbf{B}^{(0)}\right)}{\hat{L}\left(\hat{\mathbf{B}}\right)}\right)^{\frac{2}{\hat{N}}} = 1 - \exp\left\{-\frac{-2\hat{l}\left(\mathbf{B}^{(0)}\right) - \left(-2\hat{l}\left(\hat{\mathbf{B}}\right)\right)}{\hat{N}}\right\}$$

Nagelkerke's R Square

$$R^2_N = \frac{R^2_{CS}}{1 - \left\{\hat{L}\left(\mathbf{B}^{(0)}\right)\right\}^{2/N}}$$

McFadden's R Square

$$R^2_{\mathrm{M}} = 1 - \frac{\hat{l}\left(\hat{\mathbf{B}}\right)}{\hat{l}\left(\mathbf{B}^{(0)}\right)}$$

Hypothesis Testing

Contrasts defined as linear combinations of threshold and regression parameters can be tested . Given matrix **L** with *r* rows and $K - 1 + p$ columns, and vector **K** with *r* elements, Complex Samples Ordinal Regression performs testing of linear hypothesis $H_0 : \mathbf{LB} = \mathbf{K}$. See "Complex Samples: Model Testing" for details.

Custom tests

For a user specified $\mathbf{L}$ and $\mathbf{K}$, $H_0 : \mathbf{LB} = \mathbf{K}$ is tested only when it is testable; that is, when vector $\mathbf{LB}$ is estimable. Consider partition $\mathbf{L} = \left(\mathbf{L}\left(\theta\right), \mathbf{L}\left(\beta\right)\right)$, where $\mathbf{L}\left(\theta\right) = \left(\mathbf{l}_1, \ldots, \mathbf{l}_{K-1}\right)$ consists of columns corresponding to threshold parameters and $\mathbf{L}\left(\beta\right)$ be the part of $\mathbf{L}$ corresponding to regression parameters. Consider matrix $\mathbf{L}_0 = \left(\mathbf{l}_0, \mathbf{L}\left(\beta\right)\right)$ where the column vectors corresponding to threshold parameters are replaced by their sum $\mathbf{l}_0 = \sum_{k=1}^{K-1} \mathbf{l}_k$. Then $\mathbf{LB}$ is estimable if and only if $\mathbf{L}_0 = \mathbf{L}_0\mathbf{H}$, where $\mathbf{H} = \left(\mathbf{X}'_1\mathbf{X}_1\right) \mathbf{X}'_1\mathbf{X}_1$ is a $(p+1) \times (p+1)$ matrix constructed using $\mathbf{X}_1 = (1, -\mathbf{X})$.

Default tests of Model effects

For each effect specified in the model excluding intercept, type III test matrix **L** is constructed and $H_0 : \mathbf{LB} = \mathbf{0}$ is tested. Construction of matrix **L** is based on matrix $\mathbf{H} = \left(\mathbf{X}'_1\mathbf{X}_1\right)^{-}\mathbf{X}'_1\mathbf{X}_1$ and such that **LB** is estimable. It involves parameters only for the given effect and the effects containing the given effect. It does not depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable.

See “Type III Sum of Squares and Hypothesis Matrix” in *Sums of Squares* for computational details on construction of type III test matrices.

Test of Parallel Lines Assumption

Consider an alternative model for the specified cumulative link model by allowing different regression parameters $\beta^{(k)} = \left(\beta_1^{(k)}, \ldots, \beta_p^{(k)}\right)'$ for the first *K*−1 response categories:

$$link\left(P\left(Y \leq k|\mathbf{x}\right)\right) = \theta_k - \beta^{(k)}\mathbf{x}$$

The alternative model then contains parameters with threshold parameters and regression parameters. Cumulative link model is a restriction of the alternative model based on the assumption of parallel lines corresponding to the following null hypothesis:

$$H_0 : \beta^{(1)} = \cdots = \beta^{(K-1)}$$

We conduct test of this hypothesis by estimating the parameters of the alternative model and applying a Wald type test for $\mathbf{LB}_A = 0$ with the contrast matrix **L** given by

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}^{(1)} \\ . \\ \mathbf{L}^{(p)} \end{bmatrix}$$

where each $\mathbf{L}^{(t)}, t = 1, \ldots p$ is a $(K-2) \times (p+1)(K-1)$ matrix containing pairwise contrasts for parameter t between the first and the rest of the regression equations for corresponding responses.

See “Complex Samples: Model Testing” for details of conducting an appropriate Wald test. There are several testing options available, but they all require previously computed alternative model parameter estimates $\hat{\mathbf{B}}_A$ as well as their covariance matrix $\hat{\mathbf{V}}\left(\hat{\mathbf{B}}_\mathbf{A}\right)$. For some of the options, covariance matrix $\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}_\mathbf{A}\right)$ must be computed as well.

See Peterson and Harrell (1990) for a discussion of the alternative model.

Estimation of the Alternative Model

Algorithm applied for computing solution of the alternative model $\hat{\mathbf{B}}_A$ is similar to the algorithm for the restricted cumulative link model $\hat{\mathbf{B}}$. The main difference is in computation of estimating equations and second derivatives appropriate for the alternative model. They are outlined below.

Expressions $\frac{\partial \hat{l}_A}{\partial \theta_k}$, $\frac{\partial^2 \hat{l}_A}{\partial \theta_j \partial \theta_k}$ and expected $\frac{\partial^2 \hat{l}_A}{\partial \theta_j \partial \theta_k}$ for $j,k=1,...,K-1$ are identical to their restricted model counterparts.

Estimating equations for alternative model regression parameters are given by

$$\frac{\partial \hat{l}_A}{\partial \beta_t^{(k)}} = \sum_{i=1}^{n} -w_i \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \left(\frac{y_{i,k}}{\pi_{i,k}} - \frac{y_{i,k+1}}{\pi_{i,k+1}} \right) x_{i,t} = 0, k = 1, \ldots, K-1, t = 1, \ldots, p$$

Derivatives of the estimated scores for the alternative model are given by:

$$\frac{\partial^2 \hat{l}_A}{\partial \theta_j \partial \beta_t^{(k)}} = \sum_{i=1}^{n} f_{i,j,k,t} \text{ and } \frac{\partial^2 \hat{l}_A}{\partial \beta_u^{(j)} \partial \beta_t^{(k)}} = -\sum_{i=1}^{m} f_{i,j,k,t} x_{i,u}, j,k = 1, \ldots K-1, t,u = 1, \ldots, p$$

where

$$\begin{aligned} &f_{i,j,k,t} = \\ &-w_i \left[\delta_{jk} \frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2} \pi_{i,k} - \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \left(\delta_{jk} \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} - \delta_{jk-1} \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}} \right) \right] \frac{y_{i,k}}{\pi_{i,k}^2} x_{i,t} + \\ &w_i \left[\delta_{jk} \frac{\partial^2 \gamma_{i,k}}{\partial \eta_{i,k}^2} \pi_{i,k+1} - \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \left(\delta_{jk+1} \frac{\partial \gamma_{i,k+1}}{\partial \eta_{i,k+1}} - \delta_{jk} \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \right) \right] \frac{y_{i,k+1}}{\pi_{i,k+1}^2} x_{i,t}, \\ &i = 1, \ldots m, j,k = 1, \ldots K-1, t = 1, \ldots, p \end{aligned}$$

Expected derivatives of the estimated scores for alternative model are given by the following expressions:

$$\frac{\partial^2 \hat{l}_A}{\partial \theta_j \partial \beta_t^{(k)}} = \sum_{i=1}^{n} e_{i,j,k,t} \text{ and } \frac{\partial^2 \hat{l}_A}{\partial \beta_u^{(j)} \partial \beta_t^{(k)}} = -\sum_{i=1}^{m} e_{i,j,k,t} x_{i,u}, j,k = 1, \ldots K-1, t,u = 1, \ldots, p$$

where

$$\begin{aligned} &e_{i,j,k,t} = \\ &w_i \left[\left(\delta_{jk} \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} - \delta_{jk-1} \frac{\partial \gamma_{i,k-1}}{\partial \eta_{i,k-1}} \right) \frac{1}{\pi_{i,k}} - \left(\delta_{jk+1} \frac{\partial \gamma_{i,k+1}}{\partial \eta_{i,k+1}} - \delta_{jk} \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} \right) \frac{1}{\pi_{i,k+1}} \right] \frac{\partial \gamma_{i,k}}{\partial \eta_{i,k}} x_{i,t} \\ &i = 1, \ldots m, j,k = 1, \ldots K-1, t = 1, \ldots, p \end{aligned}$$

Initial values for threshold and regression parameters in the alternative model are taken as the final estimated parameters in the restricted model.

Solution of the alternative model $\hat{\mathbf{B}}_A$ is provided as an optional output.

Predicted Values

For a predictor design vector $\mathbf{x}_i$ and estimated parameters $\left(\hat{\theta}, \hat{\beta}\right)$, the predicted probability for each response category is denoted by $\hat{\pi}_{ik},\ i,k,\ k = 1, \ldots K$. The predicted category $c\left(\mathbf{x}_i\right)$ is the one with the highest predicted probability; that is,

$$c\left(\mathbf{x}_i\right) = arg\max_k \hat{\pi}_{i,k}$$

If there is a tie in determining $c\left(\mathbf{x}_i\right)$, choose the category with

1. higher $\hat{N}_k = \sum_{i=1}^{n} w_i y_{ik}$

2. If there is still a tie, choose the one with higher $n_k = \sum_{i=1}^{n} y_{ik}$
3. If there is still a tie, choose the one with lower category number.

Classification table

A two-way classification table is constructed with (*k*,*l*)th element, *k*,*l*=1,...,*K*, being the sum of weights w_i for the sample elements *i* whose actual response category is *k* and predicted response category is *l* respectively.

Predictions for new or incomplete records

Predicted probabilities and category are also computed for the records not used in the analysis, but having non-missing values for all the model predictors and subpopulation variable if any. An additional requirement is that given predictor values could be properly parametrized by using only the existing model parameters.

Cumulative odds ratios

Given user specified design vectors $\mathbf{x}_1$ and $\mathbf{x}_2$, the ratio of cumulative odds at $\mathbf{x}_1$ to cumulative odds at $\mathbf{x}_2$ is computed for cumulative logistic link. For response category *k*=1,...,*K*−1

$$or\left(\mathbf{x}_1, \mathbf{x}_2\right) = \frac{P(Y \le k|\mathbf{x}_1)/P(Y > k|\mathbf{x}_1)}{P(Y \le k|\mathbf{x}_2)/P(Y > k|\mathbf{x}_2)} = \exp\left(-\beta'\left(\mathbf{x}_1 - \mathbf{x}_2\right)\right)$$

Notice that cumulative odds for this particular link do not depend on the response category *k*. Because of this property, ordinal response model with cumulative logistic link is also called a proportional odds model.

A level 1−α confidence interval for $or\left(\mathbf{x}_1, \mathbf{x}_2\right)$ is given by

$$\exp\left[\hat{C} \pm SE\left(\hat{C}\right) t_{df}\left(1 - \alpha/2\right)\right]$$

where

$$\hat{C} = -\hat{\beta}'\left(\mathbf{x}_1 - \mathbf{x}_2\right)$$

and

$$SE\left(\hat{C}\right) = \sqrt{\left(\mathbf{x}_1 - \mathbf{x}_2\right)' Var\left(\hat{\beta}\right)\left(\mathbf{x}_1 - \mathbf{x}_2\right)}$$

Given a factor we compute odds ratios for all its categories relative to the reference category. If a covariate is specified, we compute odds ratios for its unit change. Other factors are held fixed at their respective reference categories, while other covariates are held fixed at their mean values, unless requested differently by the user.

References

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Binder, D. A. 1983. On the variances of asymptotically normal estimators from complex surveys. *International Statistical Review*, 51, 279–292.

Kish, L. 1965. *Survey Sampling*. New York: John Wiley and Sons.

Kish, L. 1995. Methods for Design Effects. *Journal of Official Statistics*, 11, 119–127.

McCullagh, P. 1980. Regression Models for Ordinal Data. *Journal of the Royal Statistical Society B*, 42:2, 109–142.

Peterson, B., and F. Harrell. 1990. Partial proportional odds models for ordinal response variables. *Applied Statistics*, 39, 205–217.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

CSSELECT Algorithms

This document describes the algorithm used by CSSELECT to draw samples according to complex designs. The data file does not have to be sorted. Population units can appear more than once in the data file and they do not have to be in a consecutive block of cases.

Notation

The following notation is used throughout this chapter unless otherwise stated:

N	Population size
n	Sample size
f	Sampling fraction
h_i	Hit counts of *i*th population unit (*i*=1,...,*N*).
M_i	Size measure of *i*th population unit (*i*=1,...,*N*).
M	Total size. $M = \sum_{i=1}^{N} M_i$
p_i	$p_i = \frac{M_i}{M}$ is the relative size of *i*th population unit (*i*=1,...,*N*)

Stratification

Stratification partitions the sampling frame into disjoint sets. Sampling is carried out independently within each stratum. Therefore, without loss of generality, the algorithm described in this document only considers sampling from one population.

In the first stage of selection, the sampling frame is partitioned by the stratification variables specified in stage 1. In the second stage, the sampling frame is stratified by first-stage strata and cluster variables as well as strata variables specified in stage 2. If sampling with replacement is used in the first stage, the first-stage duplication index is also one of the stratification variables. Stratification of the third stage continues in a like manner.

Population Size

Sampling units in a population are identified by all unique level combinations of cluster variables within a stratum. Therefore, the population size *N* of a stratum is equal to the number of unique level combinations of the cluster variables within a stratum. When a sampling unit is selected, all cases having the same sampling unit identifier are included in the sample. If no cluster variable is defined, each case is a sampling unit.

Sample Size

CSSELECT uses a fixed sample size approach in selecting samples. If the sample size is supplied by the user, it should satisfy $0 \leq n \leq N$ for any without replacement design and $n \geq 0$ for any with replacement design.

If a sampling fraction *f* is specified, it should satisfy $0 < f \leq 1$ for any without replacement design and $f > 0$ for any with replacement design. The actual sample size is determined by the formula $n = round\,(f * N)$. When the option RATEMINSIZE is specified, a sample size less than RATEMINSIZE is raised to RATEMINSIZE. Likewise, a sample size exceeding RATEMAXSIZE is lowered to RATEMAXSIZE.

Simple Random Sampling

This algorithm selects *n* distinct units out of *N* population units with equal probability; see Fan, Muller & Rezucha (1962) for more information.

- Inclusion probability of *i*th unit = n/N
- Sampling weight of *i*th = N/n

Algorithm

1. If *f* is supplied, compute n=round(f*N).
2. Set k=0, i=0 and start data scan.
3. Get a population unit and set k=k+1. If no more population units are left, terminate.
4. Test if *k*th unit should go into the sample.

   Generate a uniform (0,1) random number U.

   If $(n - i)\,/\,(N - k + 1) > U$, *k*th population unit is selected and set i=i+1.

   If i=n, terminate. Otherwise, go to step 3.

Unrestricted Random Sampling

This algorithm selects *n* units out of *N* population units with equal probability and with replacement.

- Inclusion probability of *i*th unit = $1-(1-1/N)n$
- Sampling weight of *i*th = N/n. (For use with Hansen-Hurwitz(1943) estimator)
- Expected number of hits of *i*th = n/N

Algorithm

1. Set i=0 and initialize all hit counts to zero.
2. Generate an integer *k* between 1 and *N* uniformly.
3. Increase hit count of *k*th population unit by 1.
4. Set i=i+1.
5. If i=n, then terminate. Otherwise go to step 2.

At the end of the procedure, population units with hit count greater than zero are selected.

Systematic Sampling

This algorithm selects *n* distinct units out of *N* population units. If the selection interval (*N*/*n*) is not an integer, an exact fractional selection interval is used.

- Inclusion probability of a unit = *n*/*N*
- Sampling weight = *N*/*n*

Algorithm

1. Draw a uniform (0,1) random number *U*.
2. Population units with indices {*i*: *i*=trunc((*U*+*k*)**N*/*n*)+1, *k*=0,...,*n*−1} are included in the sample.

Sequential Sampling (Chromy)

See the section on PPS sequential sampling. This algorithm is a special case of PPS Chromy with all size measures M_i equal.

PPS Sampling without Replacement (Hanurav & Vijayan)

This algorithm selects *n* distinct units out of *N* population units with probability proportional to size without replacement. This method is first proposed by Hanurav (1967) and extended by Vijayan (1968) to the case of *n*>2.

- Inclusion probability of *i*th unit = np_i
- Sampling weight of *i*th unit = $\frac{1}{np_i}$
- Special requirement: $\max M_i \le \frac{M}{n}$

Algorithm (Case 1)

This algorithm assumes that the population units are sorted by M_i; that is, $M_1 \le M_2 \le \ldots \le M_N$ with the additional assumption that $M_{N-n+1} < M_N$.

1. Compute the probabilities $\theta_j = \frac{n(p_{N-n+j+1} - p_{N-n+j})(S + jp_{N-n+1})}{S}$, *j*=1,...,*n*, where $S = \sum_{k=1}^{N-n} p_k$.
2. Select one integer from 1,...,*n* with probability proportional to θ_j.
3. If the integer selected is *i*, then the last (*n*−*i*) population units are selected.
4. Define a new set of probabilities for the first (*N*−*n*+*i*) population units.
   $$p_j^* = \frac{p_j}{S + ip_{N-n+1}}, 1 \le j \le N - n + 1$$
   $$= \frac{p_{N-n+1}}{S + ip_{N-n+1}}, N - n + 1 < j \le N - n + i$$
5. Define $P_j = \frac{M_j}{(M_{j+1} + \ldots + M_{N-n+i})}, j = 1, \ldots, N - n + i - 1$

6. Set *m*=1 and select one unit from the first (*N*−*n*+1) population units with probability proportional to

$$a_1 = ip_1^*$$
$$a_j = np_j^* \prod_{k=1}^{j-1} [1-(i-1)P_k], j = 2, ..., N-n+1$$

7. Denote the index of the selected unit by j_m.

8. Set *m*=*m*+1 and select one unit from the $(j_{m-1}+1)$th to (*N*−*n*+*m*)th population units with the following revised probabilities

$$a_{j_{m-1}+1} = (i-m+1)\,p^*_{j_{m-1}+1}$$
$$a_j = (i-m+1)\,p_j^* \prod_{k=j_{m-1}+1}^{j-1} [1-(i-m)\,P_k]\,, j = j_{m-1}+2, ..., N-n+m$$

9. Denote the selected unit in step 8 by j_m.

10. If *m*=*i*, terminate. Otherwise, go to step 8. At the end of the algorithm, the last (*n*−*i*) units and units with indices $j_1, ..., j_i$ are selected.

Joint Inclusion Probabilities (Case 1)

The joint inclusion probabilities of unit *i* and unit *j* in the population ($1 \le i < j \le N$) is given by

$$\pi_{ij} = \sum_{r=1}^{n} \theta_r K_{ij}^{(r)}$$

where

$$K_{ij}^{(r)} = \begin{cases} 1 & \text{if } N-1 \ge i > N-n+r, \\ \frac{rp_{N-n+1}}{S+rp_{N-n+1}} & \text{if } N-n+r \ge i > N-n \text{ and } j > N-n+r, \\ \frac{rp_i}{S+rp_{N-n+1}} & if N-n \ge i > 0 \text{ and } j > N-n+r, \\ \pi_{ij}^{(r)} & if j \le N-n+r. \end{cases}$$

$\pi_{ij}^{(r)}$’s are the conditional joint inclusion probabilities given that the last (*n*−*r*) units are selected at step 3. They can be computed by the following formula

$$\pi_{ij}^{(r)} = r\,(r-1)\left(1-P_1^{(r)}\right)...\left(1-P_{i-1}^{(r)}\right)P_i^{(r)}p_j^{(r)}$$

where

$$p_k^{(r)} = \begin{cases} \frac{p_k}{S+rp_{N-n+1}} & \text{if } k \le N-n+1 \\ \frac{p_{N-n+1}}{S+rp_{N-n+1}} & \text{if } N-n+1 < k \le N-n+r \end{cases}$$

and

$$P_k^{(r)} = \frac{p_k^{(r)}}{\left(p_{k+1}^{(r)}+...+p_{N-n+r}^{(r)}\right)}$$

Note: There is a typo in (3.5) of Vijayan(1967) and (3.3) of Fox(1989). The factor (1/2) should not be there. See also Golmant (1990) and Watts (1991) for other corrections.

Algorithm (Case 2)

This algorithm assumes that the population units are sorted by M_i with the order $M_1 \le M_2 \le ... \le M_N$ and the additional assumption $M_{N-n+1} = M_N$.

1. Define the probabilities

   $P_j = \frac{M_j}{(M_{j+1}+...+M_N)}, j = 1, ..., N-1$
2. Select one unit from the first ($N-n+1$) population units with probability proportional to

   $a_1 = np_1$

   $a_j = np_j \prod_{k=1}^{j-1} [1-(n-1)P_k], j = 2, ..., N-n+1$
3. Set m=1 and denote the index of the selected unit by j_m.
4. Set m=m+1.
5. Select one unit from the ($j_{m-1}+1$)th to the ($N-n+m$)th population unit with probability proportional to

   $a_{j_{m-1}+1} = (n-m+1)p_{j_{m-1}+1}$

   $a_j = (n-m+1)p_j \prod_{k=j_{m-1}+1}^{j-1} [1-(n-m)P_k], j = j_{m-1}+2, ..., N-n+m$
6. Denote the index of the unit selected in step 5 by j_m.
7. If m=n, terminate. Otherwise, go to step 4.

At the end of the algorithm, population units with indices $j_1, ..., j_n$ are selected.

Joint Inclusion Probabilities (Case 2)

Joint inclusion probabilities π_{ij} of unit i and unit j in the population ($1 \le i < j \le N$) are given by $\pi_{ij} = n(n-1)(1-P_1)...(1-P_{i-1})P_i p_j$.

PPS Sampling with Replacement

This algorithm selects n units out of N population units with probability proportional to size and with replacement. Any units may be sampled more than once.

- Inclusion probability of ith unit = $1-(1-p_i)^n$
- Sampling weight of ith unit = $\frac{1}{np_i}$. (For use with Hansen-Hurwitz(1943) estimator)
- Expected number of hits of ith unit = np_i

Algorithm

1. Compute total size $M = \sum_{i=1}^{N} M_i$.
2. Generate n uniform (0,M) random numbers $U_1, ..., U_n$.

3. Compute hit counts of *i*th population unit h_i = card$\{U_j : M^*_{i-1} < U_j \leq M^*_i, j = 1, ..., n\}$, where card{} is the number of elements in the set, $M_0 = 0$, and $M^*_i = \sum_{k=1}^{i} M_k$

   At the end of the algorithm, population units with hit count $m_i > 0$ are selected.

PPS Systematic Sampling

This algorithm selects *n* units out of *N* population units with probability proportional to size. If the size of the *i*th unit M_i is greater than the selection interval, the *i*th unit is sampled more than once.

- Inclusion probability of *i*th unit = np_i
- Sampling weight of *i*th unit = $\frac{1}{np_i}$
- Expected number of hits of *i*th unit = np_i. In order to ensure no duplicates in the sample, the condition $\max M_i \leq \frac{M}{n}$ is required.

Algorithm

1. Compute cumulated sizes $M^*_i = \sum_{k=1}^{i} M_k$.

2. Compute the selection interval *I*=*M*/*n*.

3. Generate a random number *S* from uniform(0,*I*).

4. Generate the sequence $\{S_j : S_j = S + (j - 1) I, j = 1, ..., n\}$.

5. Compute hit counts of *i*th population unit h_i = card$\{M^*_{i-1} < S_j \leq M^*_i, j = 1, ..., n\}$, *k*=1,...,*N*, where card{} is the number of elements in the set.

   At the end of the algorithm, population with hit counts $h_i > 0$ are selected.

PPS Sequential Sampling (Chromy)

This algorithm selects *n* units from *N* population units sequentially proportional to size with minimum replacement. This method is proposed by Chromy (1979).

- Inclusion probability of *i*th unit = np_i
- Sampling weight of *i*th unit = $\frac{1}{np_i}$
- Maximum number of hits of *i*th unit = $trunc(np_i) + 1$
- Applying the restriction $\max M_i \leq \frac{M}{n}$ ensures maximum number of hits is equal to 1.

Algorithm

1. Select one unit from the population proportional to its size M_i. The selected unit receives a label 1. Then assign labels sequentially to the remaining units. If the end of the list is encountered, loop back to the beginning of the list until all *N* units are labeled. These labels are the index *i* in the subsequent steps.

2. Compute the integer part of expected hit counts $I_i = trunc(M_i^*)$, where $M_i^* = \sum_{k=1}^{i} M_k$, *i*=1,...,*N*.
3. Compute the fractional part of expected hit counts $F_i = M_i^* - I_i$, *i*=1,...,*N*.
4. Define $I_0 = 0$, $F_0 = 0$ and $T_0 = 0$.
5. Set *i*=1.
6. If $T_{i-1} = I_{i-1}$, go to step 8.
7. If $T_{i-1} = I_{i-1} + 1$, go to step 9.
8. Determine accumulated hits at *i*th step (case 1).

   Set $T_i = I_i$.

   If $F_i > F_{i-1}$, set $T_i = T_i + 1$ with probability $(F_i - F_{i-1}) / (1 - F_{i-1})$

   Set *i*=*i*+1.

   If *i* > *N*, terminate. Otherwise go to step 6.
9. Determine accumulated hits at *i*th step (case 2).

   Set $T_i = I_i$.

   If $F_i > F_{i-1}$, set $T_i = T_i + 1$.

   If $F_{i-1} \geq F_i$, set $T_i = T_i + 1$ with probability F_i / F_{i-1}.

   Set *i*=*i*+1.

   If *i* > *N*, terminate. Otherwise go to step 6.

   At the end of the algorithm, number of hits of each unit can be computed by the formula $h_i = T_i - T_{i-1}$ *i*=1,...,*N*. Units with $m_i > 0$ are selected.

PPS Sampford's Method

Sampford's (1967) method selects *n* units out of *N* population units without replacement and probabilities proportional to size.

- Inclusion probability of *i*th unit = np_i
- Sampling weight of *i*th unit = $\frac{1}{np_i}$
- Special requirement: $\max M_i < \frac{M}{n}$

Algorithm

1. If $\max M_i < \frac{M}{n}$, then go to step 2, otherwise go to step 5.
2. Select one unit with probability proportional to p_i, *i*=1,...,*N*.
3. Select the remaining (*n*−1) units with probabilities proportional to $\frac{p_i}{1-np_i}$, *i*=1,...,*N*.

4. If there are duplicates, reject the sample and go to step 2. Otherwise accept the selected units and stop.

5. If $N = n$ and the M_i's are constant, then select all units in the population and set all sampling weights, 1st and 2nd order inclusion probabilities to 1.

Joint Inclusion Probabilities

First define the following quantities:

$\lambda_i = \frac{p_i}{(1-np_i)}$, *i*=1,...,*N*

$R_r = \sum_{k=1}^{N} \lambda_k^r$, *r*=1,...,*n*

$L_0 = L_{0,ij} = 1$, *i*,*j*=1,...,*N*

$L_m = \frac{1}{m}\sum_{k=1}^{m} (-1)^{k-1} R_k L_{m-k}$, *m*=1,...,*n*

$L_{m,ij} = L_m - (\lambda_i + \lambda_j) L_{m-1,ij} - \lambda_i \lambda_j L_{m-2,ij}$, *m*=1,...,*n*, *i*,*j*=1,...,*N*

$$K_n = \left(\sum_{k=1}^{n} \frac{kL_{n-k}}{n^k}\right)^{-1}$$

Given the above quantities, the joint inclusion probability of the *i*th and *j*th population units is

$$\pi_{ij} = K_n \lambda_i \lambda_j \sum_{k=2}^{n} \frac{[k - n(p_i + p_j)]L_{n-k,ij}}{n^{k-2}}$$

PPS Brewer's Method (n=2)

Brewer's (1963) method is a special case of Sampford's method when *n*=2.

PPS Murthy's Method (n=2)

Murthy's (1957) method selects two units out of *N* population units with probabilities proportional to size without replacement.

- Inclusion probability of *i*th unit = $p_i\left(1 - \frac{p_i}{1-p_i} + \sum_{k=1}^{N} \frac{p_k}{1-p_k}\right)$
- Sampling weight of *i*th unit = inverse of inclusion probability

Algorithm

1. Select first unit from the population with probabilities p_k, *k*=1,...,*N*.

2. If the first selected unit has index *i*, then select second unit with probabilities $p_k / (1 - p_i)$, $k \neq i$.

Joint Inclusion Probabilities

The joint inclusion probability of population units *i* and *j* is given by

$$\pi_{ij} = p_i p_j (2 - p_i - p_j) / (1 - p_i)(1 - p_j)$$

Saved Variables

STAGEPOPSIZE saves the population sizes of each stratum in a given stage.

STAGESAMPSIZE saves the actual sample sizes of each stratum in a given stage. See the "Sample Size" section for details on sample size calculations.

STAGESAMPRATE saves the actual sampling rate of each stratum in a given stage. It is computed by dividing the actual sample size by the population size. Due to the use of rounding and application of RATEMINSIZE and RATEMAXSIZE on sample size, the resulting STAGESAMPRATE may be different from sampling rate specified by the user.

STAGEINCLPROB saves stage inclusion probabilities. These depend on the selection method. The formulae are given in the individual sections of each selection method.

STAGEWEIGHT saves the inverse of stage inclusion probabilities.

SAMPLEWEIGHT saves the product of previous weight (if specified) and all the stage weights.

STAGEHITS saves the number of times a unit is selected in a given stage. When a WOR method is used the value is always 0 or 1. When a WR method is used it can be any nonnegative integer.

SAMPLEHITS saves the number of times an ultimate sampling unit is selected. It is equal to STAGEHITS of the last specified stage.

STAGEINDEX saves an index variable used to differentiate duplicated sampling units resulted from sampling with replacement. STAGEINDEX ranges from one to number of hits of a selected unit.

References

Brewer, K. W. R. 1963. A Model of Systematic Sampling with Unequal Probabilities. *Australian Journal of Statistics*, 5, 93–105.

Chromy, J. R. 1979. . *Sequential Sample Selection MethodsProceedings of the American Statistical Association, Survey Research Methods Section*, , 401–406.

Fan, C. T., M. E. Muller, and I. Rezucha. 1962. Development of Sampling Plans by Using Sequential (Item by Item) Selection Techniques and Digital Computers. *Journal of the American Statistical Association*, 57, 387–402.

Fox, D. R. 1989. Computer Selection of Size-Biased Samples. *The American Statistician*, 43:3, 168–171.

Golmant, J. 1990. Correction: Computer Selection of Size-Biased Samples. *The American Statistician*, , 194–194.

Hanurav, T. V. 1967. Optimum Utilization of Auxiliary Information: PPS Sampling of Two Units from a Stratum. *Journal of the Royal Statistical Society, Series B*, 29, 374–391.

Hansen, M. H., and W. N. Hurwitz. 1943. On the theory of sampling from finite populations. *Annals of Mathematical Statistics*, 14, 333–362.

Murthy, M. N. 1957. Ordered and Unordered Estimators in Sampling Without Replacement. *Sankhya*, 18, 379–390.

Sampford, M. R. 1967. On Sampling without Replacement with Unequal Probabilities of Selection. *Biometrika*, 54, 499–513.

Vijayan, K. 1968. An Exact PPS Sampling Scheme: Generalization of a Method of Hanurav. *Journal of the Royal Statistical Society, Series B*, 54, 556–566.

Watts, D. L. 1991. Correction: Computer Selection of Size-Biased Samples. *The American Statistician*, 45:2, 172–172.

CSTABULATE Algorithms

This document describes the algorithms used in the complex sampling estimation procedure CSTABULATE.

Complex sample data must contain both the values of the variables to be analyzed and the information on the current sampling design. The sampling design includes the sampling method, strata and clustering information, inclusion probabilities and the overall sampling weights.

The sampling design specification for CSTABULATE may include up to three stages of sampling. Any of the following general sampling methods may be assumed in the first stage: random sampling with replacement, random sampling without replacement and equal probabilities and random sampling without replacement and unequal probabilities. The first two sampling methods can also be specified for the second and the third sampling stage.

Notation

The following notation is used throughout this chapter unless otherwise stated:

H	Number of strata.
n_h	Sampled number of primary sampling units (PSU) per stratum.
f_h	Sampling rate per stratum.
m_{hi}	Number of elements in the *i*th sampled unit in stratum *h*.
w_{hij}	Overall sampling weight for the *j*th element in the *i*th sampled unit in stratum *h*.
$\mathbf{y}_{hij}$	Value of variable *y* for the *j*th element in the *i*th sampled unit in stratum *h*.
Y	Population total sum for variable *y*.
n	Total number of elements in the sample.
N	Total number of elements in the population.

Weights

Overall weights specified for each ultimate element are processed as given. See “Weights” in *Complex Samples: Covariance Matrix of Total* for more information on weights and variance estimation methods.

Z Expressions

For variables *y* and $y^{'}$:

$$z_{hij} = w_{hij}y_{hij},\ z^{'}{}_{hij} = w_{hij}y^{'}{}_{hij}$$

$$z_{hi} = \sum_{j=1}^{m_{hi}} z_{hij},\ z^{'}{}_{hi} = \sum_{j=1}^{m_{hi}} z^{'}{}_{hij}$$

$$\overline{z}_h = \frac{1}{n_h}\sum_{i=1}^{n_h} z_{hi}, \overline{z}'_h = \frac{1}{n_h}\sum_{i=1}^{n_h} z'_{hi}$$

$$S_h^2\left(y, y'\right) = \frac{1}{n_h - 1}\sum_{i=1}^{n_h}\left(z_{hi} - \overline{z}_h\right)\left(z'_{hi} - \overline{z}'_h\right)$$

For multi-stage samples, the index *h* denotes a stratum in the given stage, and *i* stands for unit from *h* in the same stage. The index *j* runs over all final stage elements contained in unit *hi*.

Variable Total

An estimate for the population total of variable *y* in a single-stage sample is the weighted sum over all the strata and all the clusters:

$$\hat{Y} = \sum_{h=1}^{H}\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}} w_{hij} y_{hij}$$

Alternatively, we compute the weighted sum over all the elements in the sample:

$$\hat{Y} = \sum_{i=1}^{n} w_i y_i$$

The latter expression is more general as it also applies to multi-stage samples.

Variables Total Covariance

For a multi-stage sample containing a with replacement sampling stage, all specifications other than weights are ignored for the subsequent stages. They make no contribution to the variance estimates.

Single Stage Sample

The covariance of the total for variables *y* and y' in a single-stage sample is estimated by the following:

$$\hat{C}\left(\hat{Y}, \hat{Y}'\right) = \hat{C}_1\left(\hat{Y}, \hat{Y}'\right) = \sum_{h=1}^{H} U_h\left(\hat{Y}, \hat{Y}'\right)$$

where $U_h\left(\hat{Y}, \hat{Y}'\right)$ is an estimate contribution from stratum *h* and depends on the sampling method as follows:

- For sampling with replacement: $U_h\left(\hat{Y}, \hat{Y}'\right) = n_h S_h^2\left(y, y'\right)$
- For simple random sampling: $U_h\left(\hat{Y}, \hat{Y}'\right) = \left(1 - f_h\right) n_h S_h^2\left(y, y'\right)$
- For sampling without replacement and unequal probabilities:
  $$U_h\left(\hat{Y}, \hat{Y}'\right) = \sum_{i=1}^{n_h}\sum_{i>j}^{n_h}\left(\frac{\pi_{hi}\pi_{hj}}{\pi_{hij}} - 1\right)\left(z_{hi} - z_{hj}\right)\left(z'_{hi} - z'_{hj}\right)$$

π_{hi} and π_{hj} are the inclusion probabilities for units *i* and *j* in stratum *h*, and π_{hij} is the joint inclusion probability for the same units. This estimator is due to Yates and Grundy (1953) and Sen (1953).

For each stratum *h* containing a single element, the covariance contribution $U_h\left(\hat{Y},\hat{Y}'\right)$ is always set to zero.

Two-stage Sample

When the sample is obtained in two stages and sampling without replacement is applied in the first stage, we use the following estimate for the covariance of the total for variables *y* and y':

$$\hat{C}\left(\hat{Y},\hat{Y}'\right)=\hat{C}_2\left(\hat{Y},\hat{Y}'\right)=\hat{C}_1\left(\hat{Y},\hat{Y}'\right)+\sum_{h=1}^{H}\sum_{i=1}^{n_h}\pi_{hi}\sum_{k=1}^{K_{hi}}U_{hik}\left(\hat{Y},\hat{Y}'\right)$$

where

π_{hi} is the first stage inclusion probability for the primary sampling unit *i* in stratum *h*. In the case of simple random sampling, the inclusion probability is equal to the sampling rate f_h for stratum *h*.

K_{hi} is the number of second stage strata in the primary sampling unit *i* within the first stage stratum *h*.

$U_{hik}\left(\hat{Y},\hat{Y}'\right)$ is a covariance contribution from the second stage stratum *k* from the primary sampling unit *hi*. It depends on the second stage sampling method. The corresponding formula given in the "Single Stage Sample" section applies.

Three-stage Sample

When the sample is obtained in three stages where sampling in the first stage is done without replacement and simple random sampling is applied in the second stage, we use the following estimate for the covariance of the total for variables *y* and y':

$$\hat{C}\left(\hat{Y},\hat{Y}'\right)=\hat{C}_2\left(\hat{Y},\hat{Y}'\right)+\sum_{h=1}^{H}\sum_{i=1}^{n_h}\pi_{hi}\sum_{k=1}^{K_{hi}}f_{hik}\sum_{j=1}^{n_{hik}}\sum_{l=1}^{L_{hikj}}U_{hikjl}\left(\hat{Y},\hat{Y}'\right)$$

where

- f_{hik} is the sampling rate for the secondary sampling units in the second stage stratum *hik*.
- L_{hikj} is the number of third stage strata in the secondary sampling unit *hikj*.
- $U_{hikjl}\left(\hat{Y},\hat{Y}'\right)$ is a variance contribution from the third stage stratum *l* contained in the secondary sampling unit *hikj*. It depends on the third stage sampling method. The corresponding formula given in the "Single Stage Sample" section applies.

Variable Total Variance

The variance of the total for variable *y* in a complex sample is estimated by

$$\hat{V}\left(\hat{Y}\right)=\hat{C}\left(\hat{Y},\hat{Y}\right)$$

with $\hat{C}\left(\hat{Y},\hat{Y}\right)$ defined above.

Population Size Estimation

An estimate for the population size corresponds to the estimate for the variable total; it is sum of the sampling weights. We have the following estimate for the single-stage samples:

$$\hat{N}=\sum_{h=1}^{H}\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}}w_{hij}$$

More generally,

$$\hat{N}=\sum_{i=1}^{n}w_i$$

The variance of $\hat{N}$ is obtained by replacing y_{hij} with 1; that is, by replacing z_{hij} with w_{hij} in the corresponding variance estimator formula for $\hat{V}\left(\hat{Y}\right)$.

Cell Estimates: One-Way Tables

Let the population be classified according to the values of a single categorical row variable and possibly one or more categorical variables in the layer. Categories for the row variable are enumerated by *r*=1,...,*R* and categories for the layer variables are given by *l*=1,...,*L*. Each combination of the values (*r*,*l*) defines a domain and a cell in the one-way table (*r*,*l*), *r*=1,...,*R*. For each cell we define a corresponding indicator variable:

$$\delta_{hij}\left(r,l\right)=\begin{cases}1 & \text{if the sample unit } hij \text{ is in the cell } \left(r,l\right)\\ 0 & \text{otherwise}\end{cases}$$

Sizes

To estimate a cell population size or a table population size, we replace y_i with $\delta_i\left(r,l\right)$ in the formula for the population total and obtain the following expressions:

- Cell population size: $\hat{N}\left(r,l\right)=\sum_{i=1}^{n}w_i\delta_i\left(r,l\right)$
- Table population size: $\hat{N}\left(+,l\right)=\sum_{i=1}^{n}\sum_{r=1}^{R}w_i\delta_i\left(r,l\right)$

Similarly, in order to estimate variances of the above estimators, we substitute y_{hij} with $\delta_{hij}(r,l)$ in the corresponding formula for the whole population. The following substitutions of z_{hij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating the variances of these estimators:

- Cell population size: $z_{hij}(r,l) = w_{hij}\delta_{hij}(r,l)$
- Table population size: $z_{hij}(+,l) = \sum_{r=1}^{R} w_{hij}\delta_{hij}(r,l)$

Proportions

A table proportion estimate is computed at each layer category as follows:

$$\hat{P}_{tab}(r,l) = \hat{N}(r,l)/\hat{N}(+,l)$$

This estimator is a ratio and we apply Taylor linearization formulas as suggested by Woodruff (1971). The following substitution of z_{hij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating the variance of the table proportion at a given layer:

$$z_{hij}(r,l) = w_{hij}\frac{\delta_{hij}(r,l)-\delta_{hij}(+,l)\hat{P}_{tab}(r,l)}{\hat{N}(+,l)}$$

Cell Estimates: Two-Way Tables

Let the population be cross-classified according to the values of a categorical row variable, a categorical column variable and possibly one or more categorical variables in the layer. Categories for the row variable are enumerated by *r*=1,...,*R* while categories for the column variable are denoted by *c*=1,...,*C* and categories for the layer variables are given by *l*=1,...,*L*. Each combination of values (*r*,*c*,*l*) defines a domain and a cell in the two-way table (*r*,*c*,*l*) . For each cell we define a corresponding indicator variable:

$$\delta_{hij}(r,c,l) = \begin{cases} 1 & \text{if the sample unit } hij \text{ is in the cell } (r,c,l) \\ 0 & \text{otherwise} \end{cases}$$

We will also use the following indicator notation:

- Row indicator: $\delta_i(r,+,l) = \sum_{c=1}^{C} \delta_i(r,c,l)$
- Column indicator: $\delta_i(+,c,l) = \sum_{r=1}^{R} \delta_i(r,c,l)$
- Table indicator: $\delta_i(+,+,l) = \sum_{r=1}^{R}\sum_{c=1}^{C} \delta_i(r,c,l)$

Sizes

To estimate various domain sizes, we substitute y_i with δ_i in the corresponding formula for the whole population as follows:

- Cell population size: $\hat{N}(r,c,l) = \sum_{i=1}^{n} w_i \delta_i (r,c,l)$
- Row population size: $\hat{N}(r,+,l) = \sum_{i=1}^{n} w_i \delta_i (r,+,l)$
- Column population size: $\hat{N}(+,c,l) = \sum_{i=1}^{n} w_i \delta_i (+,c,l)$
- Table population size: $\hat{N}(+,+,l) = \sum_{i=1}^{n} w_i \delta_i (+,+,l)$

Similarly, in order to estimate variance of the above estimators, we substitute y_{hij} with δ_{hij} in the corresponding formula for the whole population. The following substitutions of z_{hij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating variances of:

- Cell population size: $z_{hij}(r,c,l) = w_{hij}\delta_{hij}(r,c,l)$
- Row population size: $z_{hij}(r,+,l) = w_{hij}\delta_{hij}(r,+,l)$
- Column population size: $z_{hij}(+,c,l) = w_{hij}\delta_{hij}(+,c,l)$
- Table population size: $z_{hij}(+,+,l) = w_{hij}\delta_{hij}(+,+,l)$

Proportions

We define various proportion estimates to be computed as follows:

- Row population proportion: $\hat{P}_{row}(r,c,l) = \hat{N}(r,c,l)/\hat{N}(r,+,l)$
- Column population proportion: $\hat{P}_{col}(r,c,l) = \hat{N}(r,c,l)/\hat{N}(+,c,l)$
- Table population proportion: $\hat{P}_{tab}(r,c,l) = \hat{N}(r,c,l)/\hat{N}(+,+,l)$
- Marginal column population proportion: $\hat{P}_{mcol}(+,c,l) = \hat{N}(+,c,l)/\hat{N}(+,+,l)$
- Marginal row population proportion: $\hat{P}_{mrow}(r,+,l) = \hat{N}(r,+,l)/\hat{N}(+,+,l)$

In order to estimate variances of the above estimators, again apply the Taylor linearization formulas as for the one-way tables. The following substitutions of z_{ij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating variances of:

- Row population proportion: $z_{hij}(r,c,l) = w_{hij}\frac{\delta_{hij}(r,c,l) - \delta_{hij}(r,+,l)\hat{P}_{row}(r,c,l)}{\hat{N}(r,+,l)}$
- Column population proportion: $z_{hij}(r,c,l) = w_{hij}\frac{\delta_{hij}(r,c,l) - \delta_{hij}(+,c,l)\hat{P}_{col}(r,c,l)}{\hat{N}(+,c,l)}$
- Table population proportion: $z_{hij}(r,c,l) = w_{hij}\frac{\delta_{hij}(r,c,l) - \delta_{hij}(+,+,l)\hat{P}_{tab}(r,c,l)}{\hat{N}(+,+,l)}$
- Marginal column population proportion: $z_{hij}(+,c,l) = w_{hij}\frac{\delta_{hij}(+,c,l) - \delta_{hij}(+,+,l)\hat{P}_{mcol}(+,c,l)}{\hat{N}(+,+,l)}$
- Marginal row population proportion: $z_{hij}(r,+,l) = w_{hij}\frac{\delta_{hij}(r,+,l) - \delta_{hij}(+,+,l)\hat{P}_{mrow}(r,+,l)}{\hat{N}(+,+,l)}$

Standard Errors

Let *Z* denote any of the population or subpopulation quantities defined above: variable total, population size, ratio or mean. Then the standard error of an estimator $\hat{Z}$ is the square root of its estimated variance:

$$StdError\left(\hat{Z}\right) = \sqrt{\hat{V}\left(\hat{Z}\right)}$$

Coefficient of Variation

The coefficient of variation of the estimator $\hat{Z}$ is the ratio of its standard error and its value:

$$CV\left(\hat{Z}\right) = \frac{SE\left(\hat{Z}\right)}{\hat{Z}}$$

The coefficient of variation is undefined when $\hat{Z} = 0$.

Confidence Limits

A level 1−α confidence interval is constructed for a given $0 \leq \alpha \leq 1$ for any domain size N_d defined earlier. The confidence bounds are defined as

$$\hat{N}_d \pm SE\left(\hat{N}_d\right) t_\nu \left(1 - \alpha/2\right)$$

where $SE\left(\hat{N}_d\right)$ is the estimated standard error of $\hat{N}_d$, and $t_\nu \left(1 - \alpha/2\right)$ is the $100\left(1 - \alpha/2\right)$ percentile of the *t* distribution with ν degrees of freedom.

Proportions

For any domain proportion P_d, we use the logistic transformation $f\left(p\right) = \ln\left(p/\left(1 - p\right)\right)$ and obtain the following $1 - \alpha$ level confidence bounds for the transformed estimate:

$$\ln\left(\frac{\hat{P}_d}{1-\hat{P}_d}\right) \pm \frac{SE\left(\hat{P}_d\right)}{\hat{P}_d\left(1-\hat{P}_d\right)} t_\nu \left(1 - \alpha/2\right)$$

These bounds are transformed back to the original metric using the logistic inverse $f^{-1}\left(y\right) = \exp\left(y\right)/\left(1 + \exp\left(y\right)\right)$

Degrees of Freedom

The degrees of freedom for the *t* distributions above is calculated as the difference between the number of primary sampling units and the number of strata in the first stage of sampling. This quantity is also referred to as the sample design degrees of freedom.

Design Effects

Size

The design effect *Deff* for a two-way table cell population size is estimated by

$$Deff = \frac{\hat{V}\left(\hat{N}(r,c,l)\right)}{\hat{V}_{srs}\left(\hat{N}(r,c,l)\right)}$$

$\hat{V}\left(\hat{N}(r,c,l)\right)$ is an estimate of the variance of $\hat{N}(r,c,l)$ under the complex sample design, while $\hat{V}_{srs}\left(\hat{N}(r,c,l)\right)$ is its estimate of variance under the simple random sampling assumption:

$$\hat{V}_{srs}\left(\hat{N}(r,c,l)\right) = (fpc)\frac{1}{n-1}\hat{N}(r,c,l)\left(\hat{N} - \hat{N}(r,c,l)\right)$$

Assuming sampling without replacement we have $fpc = \left(1 - \frac{n}{\hat{N}}\right)$ given that $\frac{n}{\hat{N}} < 1$, while for sampling with replacement we set $fpc = 1$. This assumption is independent of the sampling specified for the complex sample design based variance $\hat{V}\left(\hat{N}(r,c,l)\right)$.

Computations of the design effects for the one-way table cells, as well as for the row, column and table population sizes are analogous to the one above.

Proportions

Deff for a two-way table population proportion is estimated by

$$Deff = \frac{\hat{V}\left(\hat{P}_{tab}(r,c,l)\right)}{\hat{V}_{srs}\left(\hat{P}_{tab}(r,c,l)\right)}$$

$\hat{V}\left(\hat{P}_{tab}(r,c,l)\right)$ is an estimate of the variance of $\hat{P}_{tab}(r,c,l)$ under the complex sample design, while $\hat{V}_{srs}\left(\hat{P}_{tab}(r,c,l)\right)$ is its estimate of variance under the simple random sampling assumption:

$$\hat{V}_{srs}\left(\hat{P}_{tab}(r,c,l)\right) = (fpc)\frac{\hat{N}}{n-1}\frac{\hat{P}_{tab}(r,c,l)\left(1 - \hat{P}_{tab}(r,c,l)\right)}{\hat{N}(+,+,l)}$$

with *fpc* as specified earlier.

Computations of the design effects for one-way table proportions, as well as for the row, column, marginal row and marginal column population proportions are analogous to the one above.

Design effects for various estimates are computed only when the condition $\frac{n}{\hat{N}} < 1$ is satisfied.

Design effect square root

We also compute the square root of a design effect $\sqrt{Deff}$.

Design effects and their applications have been discussed by Kish (1965) and Kish (1995).

Tests of Independence for Two-Way Tables

Let the population be cross-classified according to the values of a categorical row variable, a categorical column variable and possibly one a more categorical variables in the layer. Categories for the row variable are enumerated by *r*=1,...,*R*, while categories for the column variable are denoted by *c*=1,...,*C*. When the layer variables are given we assume that their categories coincide with the strata in the first sampling stage. In the following we omit reference to the layers as the formulas apply for each stratum separately when needed.

We use a contrast matrix **C** defined as follows. Let $\mathbf{A}_R$ be the contrast matrix given by

$$\mathbf{A}_R = \left[\mathbf{I}_{R-1} | -\mathbf{1}_{R-1}\right]'$$

$\mathbf{I}_{R-1}$ is an identity matrix of size *R*−1 and $\mathbf{1}_{R-1}$ is a vector with *R*−1 elements equal to 1. Define **C** to be a $RC \times (R-1)(C-1)$ matrix defined by the following Kronecker product:

$$\mathbf{C} = \mathbf{A}_R \otimes \mathbf{A}_C$$

Adjusted Pearson Statistic

$$X^2 = n \sum_{r=1}^{R} \sum_{c=1}^{C} \frac{\left(\hat{P}(r,c) - \hat{P}(r,+)\hat{P}(+,c)\right)^2}{\hat{P}(r,+)\hat{P}(+,c)}$$

Under the null hypothesis, the asymptotic distribution of X^2 is generally not a chi-square distribution, so we perform an adjustment using the following $\hat{\mathbf{\Delta}}$ matrix:

$$\hat{\mathbf{\Delta}} = n\left(\mathbf{C}'\mathbf{D}_{\hat{\mathbf{P}}}^{-1}\hat{\mathbf{M}}\mathbf{D}_{\hat{\mathbf{P}}}^{-1}\mathbf{C}\right)^{-1}\left(\mathbf{C}'\mathbf{D}_{\hat{\mathbf{P}}}^{-1}\hat{\mathbf{V}}\left(\hat{\mathbf{P}}\right)\mathbf{D}_{\hat{\mathbf{P}}}^{-1}\mathbf{C}\right)$$

$\hat{\mathbf{P}}$ is a vector and $\mathbf{D}_{\hat{\mathbf{P}}}$ is a diagonal matrix of size *RC* containing elements $\hat{P}(r,c)$. $\hat{\mathbf{M}} = \left[\hat{\mathbf{D}}_{\hat{\mathbf{P}}} - \hat{\mathbf{P}}\hat{\mathbf{P}}'\right]$ is a multinomial covariance matrix estimating the asymptotic covariance of $\hat{\mathbf{P}}$ under the simple random sampling design, while $\hat{\mathbf{V}}\left(\hat{\mathbf{P}}\right)$ estimates covariance matrix of $\hat{\mathbf{P}}$ under the complex sampling design.

We use the F-based variant of the Rao and Scott’s (1984) second-order adjustment

$$FX^2 = \frac{X^2}{tr\hat{\mathbf{\Delta}}}$$

where

$$d = \frac{\left(tr\hat{\mathbf{\Delta}}\right)^2}{tr\hat{\mathbf{\Delta}}^2}$$

This statistic has an approximate $F(d, d\nu)$ distribution. Properties of this test are given in a review of simulation studies by Rao and Thomas (2003).

Adjusted Likelihood Ratio Statistic

$$G^2 = 2n\sum_{r=1}^{R}\sum_{c=1}^{C}\hat{P}(r,c)\ln\left(\frac{\hat{P}(r,c)}{\hat{P}(r,+)\hat{P}(+,c)}\right)$$

The adjusted likelihood ratio statistic is computed in an analogous manner to the Pearson adjustment where $\hat{\boldsymbol{\Delta}}$ is the same as before and

$$FG^2 = \frac{G^2}{tr\hat{\boldsymbol{\Delta}}}$$

where

$$d = \frac{\left(tr\hat{\boldsymbol{\Delta}}\right)^2}{tr\hat{\boldsymbol{\Delta}}^2}$$

This statistic has an approximate $F(d, d\nu)$ distribution.

Residuals

Under the independence hypothesis, the expected table proportion estimates are given by $\hat{E}(r,c) = \hat{P}(r,+)\hat{P}(+,c)$ and residual are defined as $\hat{R}(r,c) = \hat{P}(r,c) - \hat{E}(r,c)$.

Standardized residuals are computed by

$$\frac{\hat{R}(r,c)}{\sqrt{\hat{V}\left(\hat{R}(r,c)\right)}}$$

where $\hat{V}\left(\hat{R}(r,c)\right)$ denotes the estimated residual variance.

Let $\hat{\mathbf{M}} = \left[\hat{\mathbf{D}}_{\hat{\mathbf{P}}} - \hat{\mathbf{P}}\hat{\mathbf{P}}'\right]$ estimate the asymptotic covariance matrix under simple random sampling where $\hat{\mathbf{P}}$ and $\mathbf{D}_{\hat{\mathbf{P}}}$ are defined as above. $\mathbf{X}$ is another contrast matrix specified by

$$\mathbf{X} = [\mathbf{A_R} \otimes \mathbf{1_C} | \mathbf{1_R} \otimes \mathbf{A_C}]$$

Contrast matrices $\mathbf{A}_R$ and $\mathbf{A}_C$, as well as the unit vectors $\mathbf{1}_R$ and $\mathbf{1}_C$, are defined as earlier. Variance estimates for residuals are obtained from the diagonal of the following matrix:

$$\hat{\mathbf{V}}\left(\hat{\mathbf{R}}\right) = \left[\mathbf{I} - \hat{\mathbf{M}}\mathbf{X}\left(\mathbf{X}'\hat{\mathbf{M}}\mathbf{X}\right)^{-1}\mathbf{X}'\right]\hat{\mathbf{V}}\left(\hat{\mathbf{P}}\right)\left[\mathbf{I} - \mathbf{X}\left(\mathbf{X}'\hat{\mathbf{M}}\mathbf{X}\right)^{-1}\mathbf{X}'\hat{\mathbf{M}}\right]$$

Odds Ratios and Risks

These statistics are computed only for 2×2 tables. If any layers are specified, they must correspond to the first stage strata.

Let $\hat{N}_{11}$, $\hat{N}_{12}$, $\hat{N}_{21}$ and $\hat{N}_{22}$ be the cell population size estimates, $\hat{N}_{1+}$, $\hat{N}_{2+}$, $\hat{N}_{+1}$, and $\hat{N}_{+2}$ be marginal estimates and $\hat{N}_{++}$ the population size estimate.

Estimates and Variances

The odds ratio is defined by the following expression:

$$OR = \frac{\hat{N}_{11}\hat{N}_{22}}{\hat{N}_{12}\hat{N}_{21}}$$

Relative risks are defined by

$RR_1 = \frac{\hat{N}_{11}/\hat{N}_{1+}}{\hat{N}_{21}/\hat{N}_{2+}}$ and $RR_2 = \frac{\hat{N}_{12}/\hat{N}_{1+}}{\hat{N}_{22}/\hat{N}_{2+}}$

Risk differences are given by

$D_1 = \frac{\hat{N}_{11}}{\hat{N}_{1+}} - \frac{\hat{N}_{21}}{\hat{N}_{2+}}$ and $D_2 = \frac{\hat{N}_{12}}{\hat{N}_{1+}} - \frac{\hat{N}_{22}}{\hat{N}_{2+}}$

The following substitutions of z_{ij} in the formulas for $\hat{V}\left(\hat{Y}\right)$ are used for estimating variances:

- Odds ratio: $z_{hij}\left(r,c\right) = w_{hij}\left(\frac{\delta_{hij}(1,1)}{\hat{N}_{11}} - \frac{\delta_{hij}(1,2)}{\hat{N}_{12}} - \frac{\delta_{hij}(2,1)}{\hat{N}_{21}} + \frac{\delta_{hij}(2,2)}{\hat{N}_{22}}\right) \times OR$
- Risk ratio RR_1: $z_{hij}\left(r,c\right) = w_{hij}\left(\frac{\delta_{hij}(1,1)\hat{N}_{12}}{\hat{N}_{11}\hat{N}_{1+}} - \frac{\delta_{hij}(1,2)}{\hat{N}_{1+}} - \frac{\delta_{hij}(2,1)\hat{N}_{22}}{\hat{N}_{21}\hat{N}_{2+}} + \frac{\delta_{hij}(2,2)}{\hat{N}_{2+}}\right) \times RR_1$
- Risk difference D_1: $z_{hij}\left(r,c\right) = w_{hij}\left(\frac{\delta_{hij}(1,1)\hat{N}_{12} - \delta_{hij}(1,2)\hat{N}_{11}}{\hat{N}_{1+}^2} - \frac{\delta_{hij}(2,1)\hat{N}_{22} - \delta_{hij}(2,2)\hat{N}_{21}}{\hat{N}_{2+}^2}\right)$

The estimations of variance for RR_2 and D_2 are performed using similar substitutions.

Confidence Limits

A level 1−α confidence interval is constructed for a given $0 \leq \alpha \leq 1$ for odds ratio, risk ratio and risk difference in every table.

For the odds ratio or risk ratio *R* we use the logarithm transformation and obtain the confidence bounds

$$\ln\left(\hat{R}\right) \pm \frac{SE\left(\hat{R}\right)}{\hat{R}} t_{\nu}\left(1-\alpha/2\right)$$

These bounds are transformed back to the original metric using the exponential function. No transformations are used when estimating confidence bounds for a risk difference *D*:

$$\hat{D} \pm SE\left(\hat{D}\right) t_{\nu}\left(1-\alpha/2\right)$$

Tests of Homogeneity for One-Way Tables

Let the population be classified according to the values of a categorical row variable and possibly one a more categorical variables in the layer. Categories for the row variable are enumerated by *r*=1,...,*R*. When the layer variables are given we assume that their categories coincide with the strata in the first sampling stage. In the following we omit references to the layers as the formulas apply for each stratum separately when needed.

We study proportions $P(r) = N(r)/N(+)$. The test of homogeneity consists of testing the null hypotheses $\mathbf{H}_0 : P(r) = 1/R$ for r=1,...,R.

Adjusted Pearson Statistic

We perform an adjusted Pearson statistic test for testing the homogeneity. The Pearson test statistic is computed according to the following standard formula:

$$X^2 = n\sum_{r=1}^{R} R\left(\hat{P}(r) - 1/R\right)^2$$

Under the null hypothesis, the asymptotic distribution of X^2 is generally not a chi-square distribution, so we perform an adjustment using the following $\hat{\mathbf{\Delta}}$ matrix:

$$\hat{\mathbf{\Delta}} = n\left(\hat{\mathbf{M}}\left(\hat{\mathbf{P}}_0\right)\right)^{-1}\hat{\mathbf{V}}\left(\hat{\mathbf{P}}_0\right)$$

$\hat{\mathbf{V}}\left(\hat{\mathbf{P}}_0\right)$ is the estimated covariance matrix under the complex sample design, while $\hat{\mathbf{M}}\left(\hat{\mathbf{P}}_0\right)$ is an estimated asymptotic covariance matrix under the simple random sampling given by

$$\hat{\mathbf{M}}\left(\hat{\mathbf{P}}_0\right) = \left[diag\left(\hat{\mathbf{P}}_0\right) - \hat{\mathbf{P}}_0\hat{\mathbf{P}}_0'\right]$$

where $\hat{\mathbf{P}}_0$ is a vector and $diag\left(\hat{\mathbf{P}}_0\right)$ is a diagonal matrix of size R−1 containing elements $\hat{P}(r)$, r=1,...,R−1.

We use the *F*-based variant of the Rao and Scott's (1984) second-order adjustment

$$FX^2 = \frac{X^2}{tr\hat{\mathbf{\Delta}}}$$

where

$$d = \frac{\left(tr\hat{\mathbf{\Delta}}\right)^2}{tr\hat{\mathbf{\Delta}}^2}$$

This statistic has an asymptotic approximate $F(d, d\nu)$ distribution.

Adjusted Likelihood Ratio Statistic

The likelihood ratio test statistic is given by

$$G^2 = 2n\sum_{r=1}^{R} \hat{P}(r)\ln\left(R\hat{P}(r)\right)$$

The adjusted likelihood ratio statistic is computed in an identical way as the adjustment for the Pearson statistic:

$$FG^2 = \frac{G^2}{tr\hat{\mathbf{\Delta}}}$$

d and $\hat{\mathbf{\Delta}}$ are the same as specified before. This statistic has an asymptotic approximate distribution.

$F(d, d\nu)$

References

Cochran, W. G. 1977. *Sampling Techniques*, 3rd ed. New York: John Wiley and Sons.

Hansen, M. H., and W. N. Hurwitz. 1943. On the theory of sampling from finite populations. *Annals of Mathematical Statistics*, 14, 333–362.

Horwitz, D. G., and D. J. Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47, 663–685.

Kish, L. 1965. *Survey Sampling*. New York: John Wiley and Sons.

Kish, L. 1995. Methods for Design Effects. *Journal of Official Statistics*, 11, 119–127.

Rao, J. N. K., and A. J. Scott. 1981. The analysis of categorical data from complex sample surveys: chi-squared tests for goodness of fit and independence in two-way tables. *Journal of the American Statistical Association*, 76, 221–230.

Rao, J. N. K., and A. J. Scott. 1984. On chi-squared tests for multiway contingency tables with cell proportions estimated from survey data. *Annals of Statistics*, 12, 46–60.

Rao, J. N. K., and D. R. Thomas. 2003. Analysis of categorical response data from complex surveys: an Appraisal and update. In: *Analysis of Survey Data*, R. Chambers, and C. Skinner, eds. New York: John Wiley & Sons.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

Sen, A. R. 1953. On the estimate of the variance in sampling with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, 5, 55–77.

Thomas, D. R., and J. N. K. Rao. 1987. Small-sample comparisons of level and power for simple goodness-of-fit statistics under cluster sampling. *Journal of the American Statistical Association*, 82, 630–636.

Woodruff, R. S. 1971. A Simple Method for Approximating the Variance of a Complicated Estimate. *Journal of the American Statistical Association*, 66, 411–414.

Yates, F., and P. M. Grundy. 1953. Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society Series B*, 15, 253–261.

Complex Samples: Covariance Matrix of Total

This document describes the algorithms used in the complex sampling module procedures for estimation of covariance matrix of population total estimates. It contains a more general formulation of the algorithms given in CSDESCRIPTIVES and CSTABULATE.

Complex sample data must contain both the values of the variables to be analyzed and the information on the current sampling design. Sampling design includes the sampling method, strata and clustering information, inclusion probabilities and the overall sampling weights.

Sampling design specification may include up to three stages of sampling. Any of the following general sampling methods may be assumed in the first stage: random sampling with replacement, random sampling without replacement and equal probabilities and random sampling without replacement and unequal probabilities. The first two sampling methods can also be specified for the second and the third sampling stage.

Notation

The following notation is used throughout this chapter unless otherwise stated:

H	Number of strata.
n_h	Sampled number of primary sampling units (PSU) per stratum.
f_h	Sampling rate per stratum.
m_{hi}	Number of elements in the *i*th sampled unit in stratum *h*.
w_{hij}	Overall sampling weight for the *j*th element in the *i*th sampled unit in stratum *h*.
$\mathbf{y}_{hij}$	Values of vector **y** for the *j*th element in the *i*th sampled unit in stratum *h*.
$\mathbf{y}_T$	Population total sum for vector of variables **y**.
n	Total number of elements in the sample.
N	Total number of elements in the population.

Weights

Overall weights specified for each ultimate element are processed as given. They can be obtained as a product of weights for corresponding units computed in each sampling stage.

When sampling without replacement in a given stage, the substitution $w_{hi} = 1/\pi_{hi}$ for unit *i* in stratum *h* will result in application of the estimator for the population totals due to Horvitz and Thompson (1952). The corresponding variance estimator will also be unbiased. π_{hi} is the probability of unit *i* from stratum *h* being selected in the given stage.

If sampling with replacement in a given stage, the substitution $w_{hi} = 1/(n_h p_{hi})$ yields the estimator for the population totals due to Hansen and Hurwitz (1943). Repeatedly selected units should be replicated in the data. The corresponding variance estimator will be unbiased. p_{hi} is the probability of selecting unit *i* in a single draw from stratum *h* in the given stage.

Weights obtained in each sampling stage need to be multiplied when processing multi-stage samples. The resulting overall weights for the elements in the final stage are used in all expressions and formulas below.

Z Expressions

$$\mathbf{z}_{hij} = w_{hij}\mathbf{y}_{hij}$$

$$\mathbf{z}_{hi} = \sum_{j=1}^{m_{hi}} \mathbf{z}_{hij}$$

$$\overline{\mathbf{z}}_h = \frac{1}{n_h}\sum_{i=1}^{n_h} \mathbf{z}_{hi}$$

$$\mathbf{S}_h^2(\mathbf{y}) = \frac{1}{n_h-1}\sum_{i=1}^{n_h} (\mathbf{z}_{hi} - \overline{\mathbf{z}}_h)(\mathbf{z}_{hi} - \overline{\mathbf{z}}_h)'$$

For multi-stage samples, index *h* denotes a stratum in the given stage, and *i* stands for unit from *h* in the same stage. Index *j* runs over all final stage elements contained in unit *hi*.

Total Estimation

An estimate for the population total of vector of variables **y** in a single-stage sample is the weighted sum over all the strata and all the clusters:

$$\hat{\mathbf{y}}_T = \sum_{h=1}^{H}\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}} w_{hij}\mathbf{y}_{hij}$$

Alternatively, we compute the weighted sum over all the elements in the sample:

$$\hat{\mathbf{y}}_T = \sum_{i=1}^{n} w_i\mathbf{y}_i$$

The latter expression is more general as it also applies to multi-stages samples.

Total covariances

For a multi-stage sample containing a with replacement sampling stage, all specifications other than weights are ignored for the subsequent stages. They make no contribution to the variance estimates.

Single stage sample

Covariance of the total for vector **y** in a single-stage sample is estimated by the following:

$$\hat{\mathbf{V}}\left(\hat{\mathbf{y}}_T\right) = \hat{\mathbf{V}}_1\left(\hat{\mathbf{y}}_T\right) = \sum_{h=1}^{H} \mathbf{U}_h\left(\hat{\mathbf{y}}_T\right)$$

where $\mathbf{U}_h\left(\hat{\mathbf{y}}_T\right)$ is an estimate contribution from stratum *h* and depends on the sampling method as follows:

For sampling with replacement

$$\mathbf{U}_h\left(\hat{\mathbf{y}}_T\right) = n_h \mathbf{S}_h^2\left(\mathbf{y}\right)$$

For simple random sampling

$$\mathbf{U}_h\left(\hat{\mathbf{y}}_T\right) = \left(1 - f_h\right) n_h \mathbf{S}_h^2\left(\mathbf{y}\right)$$

For sampling without replacement and unequal probabilities

$$\mathbf{U}_h\left(\hat{\mathbf{y}}_T\right) = \sum_{i=1}^{n_h} \sum_{i>j}^{n_h} \left(\frac{\pi_{hi}\pi_{hj}}{\pi_{hij}} - 1\right)\left(\mathbf{z}_{hi} - \mathbf{z}_{hj}\right)\left(\mathbf{z}_{hi} - \mathbf{z}_{hj}\right)'$$

π_{hi} and π_{hj} are the inclusion probability for units *i* and *j* in stratum *h*, and π_{hij} is the joint inclusion probability for the same units. This estimator is due to Yates and Grundy (1953) and Sen (1953). In some situations it may yield a negative estimate and is treated as undefined. For each stratum *h* containing a single element, the covariance contribution $\mathbf{U}_h\left(\hat{\mathbf{y}}_T\right)$ is always set to zero.

Two-stage sample

When the sample is obtained in two stages and sampling without replacement is applied in the first stage, we use the following estimate for the covariance of the total for vector **y**:

$$\hat{\mathbf{V}}\left(\hat{\mathbf{y}}_T\right) = \hat{\mathbf{V}}_2\left(\hat{\mathbf{y}}_T\right) = \hat{\mathbf{V}}_1\left(\hat{\mathbf{y}}_T\right) + \sum_{h=1}^{H} \sum_{i=1}^{n_h} \pi_{hi} \sum_{k=1}^{K_{hi}} \mathbf{U}_{hik}\left(\hat{\mathbf{y}}_T\right)$$

π_{hi} is the first stage inclusion probability for the primary sampling unit *i* in stratum *h*. In case of simple random sampling, the inclusion probability is equal to the sampling rate f_h for stratum *h*.

K_{hi} is the number of second stage strata in the primary sampling unit *i* within the first stage stratum *h*.

$\mathbf{U}_{hik}\left(\hat{\mathbf{y}}_T\right)$ is a covariance contribution from the second stage stratum *k* from the primary sampling unit *hi*. Its value depends on the second stage sampling method; the corresponding formula from “Single stage sample ” applies.

Three-stage sample

When the sample is obtained in three stages where sampling in the first stage is done without replacement and simple random sampling is applied in the second stage, we use the following estimate for the covariance of the total for vector **y**:

$$\hat{\mathbf{V}}\left(\hat{\mathbf{y}}_T\right) = \hat{\mathbf{V}}_2\left(\hat{\mathbf{y}}_T\right) + \sum_{h=1}^{H}\sum_{i=1}^{n_h}\pi_{hi}\sum_{k=1}^{K_{hi}} f_{hik}\sum_{j=1}^{n_{hik}}\sum_{l=1}^{L_{hikj}} \mathbf{U}_{hikjl}\left(\hat{\mathbf{y}}_T\right)$$

f_{hik} is the sampling rate for the secondary sampling units in the second stage stratum *hik*.

L_{hikj} is the number of the third stage strata in the secondary sampling unit *hikj*.

$\mathbf{U}_{hikjl}\left(\hat{\mathbf{y}}_T\right)$ is a covariance contribution from the third stage stratum *l* contained in the secondary sampling unit *hikj*. Its value depends on the second stage sampling method; the corresponding formula from “Single stage sample ” applies.

Total variance

Variance of the total estimate for the *r*th element of the vector $\hat{\mathbf{y}}_T$, is estimated by the *r*th diagonal element of the covariance matrix for $\hat{\mathbf{y}}_T$

$$\hat{V}\left(\left(\hat{\mathbf{y}}_T\right)_r\right) = \hat{\mathbf{V}}\left(\hat{\mathbf{y}}_T\right)_{rr}$$

Population Size Estimation

An estimate for the population size corresponds to the estimate for the variable total; it is sum of the sampling weights. We have the following estimate for the single-stage samples:

$$\hat{N} = \sum_{h=1}^{H}\sum_{i=1}^{n_h}\sum_{j=1}^{m_{hi}} w_{hij}$$

More generally,

$$\hat{N} = \sum_{i=1}^{n} w_i$$

Variance of $\hat{N}$ is obtained by replacing y_{hij} with 1, i.e. by replacing z_{hij} with w_{hij} in the corresponding variance estimator formula for $\hat{V}\left(\hat{y}_T\right)$.

References

Hansen, M. H., and W. N. Hurwitz. 1943. On the theory of sampling from finite populations. *Annals of Mathematical Statistics*, 14, 333–362.

Horwitz, D. G., and D. J. Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47, 663–685.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

Sen, A. R. 1953. On the estimate of the variance in sampling with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, 5, 55–77.

Yates, F., and P. M. Grundy. 1953. Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society Series B*, 15, 253–261.

Complex Samples: Model Testing

This document describes the methods used for conducting linear hypothesis tests based on the estimated parameters in Complex Samples models.

Required input is a set of the linear hypothesis, parameter estimates and their covariance matrix estimated for the complex sample design. Some methods require an estimate of the parameter covariance matrix under the simple random sampling assumption as well. Also needed is the number of degrees of freedom for the complex sample design; typically this will be the difference between the number of primary sampling units and the number of strata in the first stage of sampling.

Given consistent estimates of the above constructs, no additional restrictions are imposed on the complex sample design.

Notation

The following notation is used throughout this chapter unless otherwise stated:

p	Number of regression parameters in the model.
r	The number of linear hypotheses considered.
$\mathbf{L}$	$r \times p$ generalized linear hypothesis matrix.
$\mathbf{K}$	$r \times 1$ vector of hypothesis values.
$\mathbf{B}$	$p \times 1$ vector of population parameters.
$\hat{\mathbf{B}}$	$p \times 1$ vector of estimated population parameters (solution).
$\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)$	$p \times p$ estimated covariance matrix for $\hat{\mathbf{B}}$ given the complex sample design.
ν	Sampling design degrees of freedom.

Hypothesis Testing

Given $\mathbf{L}$ and $\mathbf{K}$, the following generalized linear hypothesis test is performed:

$$H_0 : \mathbf{LB} = \mathbf{K}$$

It is assumed that $\mathbf{LB}$ is estimable.

Wald Chi-Square Test

$$X^2 = \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)' \left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^{-}\left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)$$ Koch et al. (1975)

The statistic has an asymptotic chi-square distribution with $r_I = rank\left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)$ degrees of freedom. If $r_I < r$, $\left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^{-}$ is a generalized inverse such that Wald tests are effective for a restricted set of hypothesis $\mathbf{L}_I\mathbf{B} = \mathbf{K}_I$ containing a particular subset I of independent rows from H_0.

Wald F Test

$F = \frac{\nu - r_I + 1}{r_I \nu} X^2$ Fellegi (1980)

This statistic has an approximate asymptotic F-distribution $F\left(r_I, \nu - r_I + 1\right)$. The statistic is undefined if $\nu < r_I$. See Korn and Graubard (1990) for the properties of this statistic.

Adjusted Wald Chi-Square Test

The Wald chi-square statistic under the simple random sampling assumption is given by the following expression:

$$X_{srs}^2 = \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)' \left(\mathbf{L}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^{-} \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{K}\right)$$

where $\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)$ is an asymptotic covariance matrix estimated under the simple random sampling assumption. If $rank\left(\mathbf{L}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right) < r$, adjusted Wald tests are effective for a restricted set of hypotheses $\mathbf{L}_I\mathbf{B} = \mathbf{K}_I$ containing a particular subset *I* of independent rows from H_0.

Since the asymptotic distribution of X_{srs}^2 is generally not a chi-square distribution, it is adjusted using the following matrix:

$$\hat{\mathbf{\Delta}} = \left(\mathbf{L}\hat{\mathbf{V}}_{srs}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)^{-} \left(\mathbf{L}\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)\mathbf{L}'\right)$$

where $\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)$ is an estimated asymptotic covariance matrix under the complex sample design. We use second-order adjustment as in Rao and Scott's (1984) given by

$$X_{adj}^2 = \frac{X_{srs}^2}{tr\hat{\mathbf{\Delta}}/d}$$

where

$$d = \frac{\left(tr\hat{\mathbf{\Delta}}\right)^2}{tr\hat{\mathbf{\Delta}}^2}$$

This statistic has an approximate asymptotic chi-square distribution with *d* degrees of freedom. See Graubard and Korn (1993) for properties of this statistic in reference to regression problems.

Adjusted Wald F Test

$F_{adj} = \frac{X_{srs}^2}{tr\hat{\mathbf{\Delta}}}$ Rao and Scott's (1984)

This statistic has an approximate asymptotic *F* distribution $F\left(d, d\nu\right)$ where *d* is defined as above. See Thomas and Rao (1987) for the heuristic derivation of this test, and Rao and Thomas (2003) for a review of the related simulation studies.

Individual Tests

Each row l' of the **L** matrix may also be tested separately. For such tests, or when the **L** matrix contains a single row, the statistics above simplify as follows:

$$X^2 = \frac{\left(l'\hat{\mathbf{B}}-k\right)^2}{l'\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)l}$$

and

$$X^2 = F = ?^2_{adj} = F_{adj}$$

The test statistics X^2 and X^2_{adj} have asymptotic chi-square distributions with 1 degree of freedom. The test statistics *F* and F_{adj} have approximate asymptotic *F* distributions $F(1,\nu)$. The tests are undefined if $l'\hat{\mathbf{V}}\left(\hat{\mathbf{B}}\right)l$ is not positive.

Significance Values

Given a value of test statistic *T* and a corresponding cumulative distribution function *G* as specified above, the *p*-value of the given test is computed as *p*=1−*G*(*T*).

Multiple Comparisons

In addition to the testing methods mentioned in the previous section, the hypothesis $H_0 : \mathbf{LB} = \mathbf{K}$ can also be tested using the multiple row hypotheses testing technique. Let l'_i be the *i*th row vector of the **L** matrix, and k_i be the *i*th element of the **K** vector. The *i*th row hypothesis is $H_{0i} : l'_i\mathbf{B} = k_i$. Testing H_0 is the same as testing multiple hypotheses $\{H_{0i}\}_{i=1}^R$ simultaneously, where *R* is the number of non-redundant row hypotheses. A hypothesis H_{0i} is redundant if there exists another hypothesis $H_{0j}, j \neq i$ such that $l_i = cl_j, k_i = ck_j, c \neq 0$.

For each individual hypothesis H_{0i}, tests described in the previous section can be performed. Let p_i denote the *p*-value for testing H_{0i}, and p_i^* denote the adjusted *p*-value. The conclusion from multiple testing is, at level α (the family-wise type I error),

reject $H_{0i} : l'_i\mathbf{B} = k_i$ if $p_i^* < \alpha$

reject $H_0 : \mathbf{LB} = \mathbf{K}$ if $\min_i(p_i^*) < \alpha$

There are different methods for adjusting *p*-values. If the adjusted *p*-value is bigger than 1, it is set to 1 in all the methods.

Sequential Tests. In sequential testing, the *p*-values are first ordered from the smallest to the biggest, and then adjusted depending on the order. Let the ordered *p*-values be
$p_{(1)} \leq p_{(2)} \leq \cdots \leq p_{(R)}$.

LSD (Least Significant Difference)

The adjusted *p*-values are the same as the original *p*-values:

$$p_i^* = p_i$$

Bonferroni

The adjusted *p*-values are:

$$p_i^* = Rp_i$$

Sidak

The adjusted *p*-values are:

$$p_i^* = 1 - (1 - p_i)^R$$

Sequential Bonferroni

The adjusted *p*-values are:

$$p_{(i)}^* = \begin{cases} Rp_{(1)} & i = 1 \\ \max\left((R - i + 1)\, p_{(i)}, p_{(i-1)}^*\right) & i \geq 2 \end{cases}$$

Sequential Sidak

The adjusted *p*-values are:

$$p_{(i)}^* = \begin{cases} 1 - \left(1 - p_{(1)}\right)^R & i = 1 \\ \max\left(1 - \left(1 - p_{(i)}\right)^{R-i+1}, p_{(i-1)}^*\right) & i \geq 2 \end{cases}$$

Comparison of Adjustment Methods

A multiple testing procedure tells not only if H_0 is rejected, but also if each individual H_{0i} is rejected. All the methods, except LSD, control the family-wise type I error for testing H_0; that is, the probability of rejecting at least one individual hypothesis under H_0. In addition, sequential methods also control the family-wise type I error for testing any subset of H_0.

LSD is the one without any adjustment, it rejects H_0 too often. It does not control the family-wise type I error and should never be used to test H_0. It is provided here mainly for reference.

Bonferroni is conservative in the sense that it rejects H_0 less often than it should. In some situations, it becomes extremely conservative when test statistics are highly correlated.

Sidak is also conservative in most cases, but is less conservative than Bonferroni. It gives the exact type I error when test statistics are independent.

Sequential Bonferroni is as conservative as the Bonferroni in terms of testing H_0 because the smallest adjusted *p*-value used in making decision is the same in both methods. But in term of testing individual H_{0i}, it is less conservative than the Bonferroni. Sequential Bonferroni rejects at least as many individual hypotheses as Bonferroni.

Sequential Sidak is as conservative as the Sidak in terms of testing H_0, but less conservative than the Sidak in terms of testing individual H_{0i}. Sequential Sidak is less conservative than sequential Bonferroni.

References

Fellegi, I. P. 1980. Approximate tests of independence and goodness of fit based on stratified multistage samples. *Journal of the American Statistical Association*, 75, 261–268.

Graubard, B. I., and E. L. Korn. Graubard. Hypothesis testing with complex survey data: The use of classical quadratic test statistics with particular reference to regression problems. *Journal of the American Statistical Association*, 88, 629–641.

Holm, S. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6, 65–70.

Koch, G. G., D. H. Freeman, and J. L. Freeman. 1975. Strategies in the multivariate analysis of data from complex surveys. *International Statistical Review*, 43, 59–78.

Korn, E. L., and B. L. Graubard. 1990. Simultaneous testing of regression coefficients with complex survey data: Use of Bonferroni t statistics. *American Statistician*, 44, 270–276.

Rao, J. N. K., and A. J. Scott. 1984. On chi-squared tests for multiway contingency tables with cell proportions estimated from survey data. *Annals of Statistics*, 12, 46–60.

Rao, J. N. K., and D. R. Thomas. 2003. Analysis of categorical response data from complex surveys: an Appraisal and update. In: *Analysis of Survey Data*, R. Chambers, and C. Skinner, eds. New York: John Wiley & Sons.

Thomas, D. R., and J. N. K. Rao. 1987. Small-sample comparisons of level and power for simple goodness-of-fit statistics under cluster sampling. *Journal of the American Statistical Association*, 82, 630–636.

Wright, S. P. 1992. Adjusted P-values for simultaneous inference. *Biometrics*, 48, 1005–1013.

CTABLES Algorithms

This document describes the algorithms used in the Custom Tables procedure.

Weights

There are two ways in which weighting can be applied in CTABLES:

1. Frequency or case weighting is specified via a WEIGHT command. Weights specified in this manner represent frequency replication (i.e., cases with the same values for all variables) and should be positive integers. Non-integer values are accepted and are used as specified for descriptive statistics, but sums are generally rounded in computing inferential statistics (standard errors, confidence intervals and test statistics).

2. Effective sample size or effective base adjustment weighting is specified via a WEIGHT subcommand. Weights need only be positive and no rounding is applied at any point in computations when using this form of adjustment weighting.

If the WEIGHT subcommand has been specified, formulas for effective base weighting are used and if a WEIGHT command is also in effect, it is silently ignored. If no WEIGHT subcommand is specified, formulas for weighted analyses using the WEIGHT command are used. If no weighting is in effect, these formulas are used with all weights equal to 1.

A note on weights and multiple response sets

Case weights are always based on Counts, not Responses, even when one of the variables is a multiple response variable.

Means and Sums

This section describes the algorithm used in computing variances, standard errors and confidence intervals for the means and sums of scale variables.

Notation

c_i	Unweighted case count in the i-th category, i=1,...,k.
$\propto_i$	Population mean of the i-th category, i=1,...,k.
x_{il}	l-th observation in i-th category, i=1,...,k.
w_{il}	Weight of the l-th observation in i-th category, i=1,...,k.
w_i	Sum of weights in category i, i=1,...,k.
w'_i	Rounded sum of weights in category i, i=1,...,k.

q_i	Sum of squared weights in category i, i=1,...,k.
e_i	Effective base in category i, i=1,...,k.
$\bar{x}_i$	Weighted mean of category i, i=1,...,k.
x_{i+}	Weighted sum of category i, i=1,...,k
s_i^2	Weighted variance of category i, i=1,...,k..
$\hat{s}_i^2$	Adjusted variance of category i, i=1,...,k incorporating effective base.
$\hat{s}_{\bar{x}_i}$	Estimated standard error of the mean of category i, i=1,...,k.
$\hat{s}_{x_{i+}}$	Estimated standard error of the sum of category i, i=1,...,k.
$(1-\alpha)\%$	Confidence interval coverage level supplied by the user.

Conditions and assumptions

- User and system missing values of scale variables are excluded.

Algorithm

Means and Sums

Basic weighted statistics

Weighted sum of i-th category: $x_{i+} = \sum_{l=1}^{c_i} w_{il} x_{il}$.

Weighted mean of i-th category: $\bar{x}_i = \dfrac{\sum_{l=1}^{c_i} w_{il} x_{il}}{w_i}$

Weighted variance of i-th category: $s_i^2 = \dfrac{\sum_{l=1}^{c_i} w_{il} (x_{il} - \bar{x}_i)^2}{w_i - 1}$.

Weighted standard deviation of the i-th category: $s_i = \sqrt{s_i^2}$.

Standard errors with WEIGHT command in effect

Estimated standard error of weighted mean of i-th category:

$\hat{s}_{\bar{x}_i} = \dfrac{s_i}{\sqrt{w'_i}}$.

Estimated standard error of weighted sum of i-th category:

$\hat{s}_{x_{i+}} = \sqrt{w'_i} s_i$.

Adjusted weighted statistics and standard errors with effective base weighting

Effective base of i-th category: $e_i = \dfrac{w_i^2}{q_i}$

Adjusted variance estimate of i-th category incorporating effective base:

$$\hat{s}_i^2 = \frac{\dfrac{e_i}{w_i}\sum_{l=1}^{c_i} w_{il}(x_{il}-\bar{x}_i)^2}{e_i - 1} = \frac{\sum_{l=1}^{c_i} w_{il}(x_{il}-\bar{x}_i)^2}{w_i - \dfrac{w_i}{e_i}} = \frac{(w_i - 1)s_i^2}{w_i - \dfrac{w_i}{e_i}}$$

Estimated standard error of weighted mean of i-th category: $\hat{s}_{\bar{x}_i} = \dfrac{\hat{s}_i}{\sqrt{e_i}}$

Estimated standard error of weighted sum of i-th category:

$\hat{s}_{x_{i+}} = w_i \dfrac{\hat{s}_i}{\sqrt{e_i}}$

Confidence interval for weighted mean with WEIGHT command

(1- α)% confidence interval for population mean μ_i of i-th category:

$$\bar{x}_i \pm t_{1-\alpha/2, w'_i - 1} \, \hat{s}_{\bar{x}_i} \, ,$$

where $t_{\alpha,df}$ is the value for a Student's *t* distribution with df degrees of freedom that exceeds α % of the distribution.

Confidence interval for weighted mean with effective base weighting

(1- α)% confidence interval for population mean μ_i of i-th category incorporating effective base:

$$\bar{x}_i \pm t_{1-\alpha/2, e_i - 1} \, \hat{s}_{\bar{x}_i}$$

where $t_{\alpha,df}$ is the value for a Student's *t* distribution with df degrees of freedom that exceeds α % of the distribution.

Confidence interval for weighted sum with WEIGHT command

(1- α)% confidence interval for population sum of i-th category:

$$x_{i+} \pm t_{1-\alpha/2,w'_i-1} \hat{s}_{x_{i+}} ,$$

where $t_{\alpha,df}$ is the value for a Student's *t* distribution with df degrees of freedom that exceeds α % of the distribution.

Confidence interval for weighted sum with effective base weighting

(1- α)% confidence interval for population sum of i-th category:

$$x_{i+} \pm t_{1-\alpha/2,e_i-1} \hat{s}_{x_{i+}} ,$$

where $t_{\alpha,df}$ is the value for a Student's *t* distribution with df degrees of freedom that exceeds α % of the distribution.

Counts and Percentages

This section describes the algorithms used in computing adjusted standard errors and confidence intervals for counts and percentages for categorical variables.

Notation

c	Unweighted case count.
δ_{il}	Indicator of whether the l-th case is in the i-th category.
w_l	Weight of the l-th observation.
w_i	Sum of weights in category i.
W	Sum of weights over all categories used in forming the proportion/percentage denominator.
q	Sum of squared weights over all categories.
E	Effective sample size or effective base over all categories used in forming the proportion/percentage denominator.
w'_i	Rounded sum of weights in category i.
W'	Rounded sum of weights over all categories used in forming the proportion/percentage denominator.
p_i	Weighted observed proportion of category i.

$\hat{p}_i$	Estimated population proportion of category i.
$\hat{s}_{p_i}$	Estimated standard error of the proportion in category i.
$\hat{P}_i\%$	Estimated population percentage in category i.
$\hat{\Sigma}_i$	Estimated population count in category i.
$(1-\alpha)\%$	Confidence interval coverage level supplied by the user.

Conditions and assumptions

- Cases with user and system missing values of scale variables are excluded.

Algorithm

Counts and Percentages

Basic weighted statistics

Weighted observed count in i-th category: $w_i = \sum_{l=1}^{c} w_l \delta_{il}$

Weighted observed proportion in i-th category:

$$p_i = \frac{\sum_{l=1}^{c} w_l \delta_{il}}{W}.$$

Adjusted weighted statistics

Estimated population proportion in i-th category: $\hat{p}_i = p_i$.

Estimated standard error of population proportion in i-th category with WEIGHT command in effect:

$$\hat{s}_{p_i} = \sqrt{\frac{\hat{p}_i\left(1-\hat{p}_i\right)}{W'}}.$$

Estimated standard error of population proportion in i-th category with WEIGHT subcommand in effect:

$$\hat{s}_{p_i} = \sqrt{\frac{\hat{p}_i\left(1-\hat{p}_i\right)}{E}}$$

where

$$E = \frac{W^2}{q}.$$

Estimated standard error of weighted percentage in i-th category:

$$SE(\hat{P}_i\,\%) = 100\,\hat{s}_{p_i}.$$

Estimated standard error of population count in i-th category:

$$SE(\hat{\Sigma}_i) = W'\,\hat{s}_{p_i}.$$

Confidence interval for weighted population proportion

$(1-\alpha)\%$ confidence interval for estimated population proportion of i-th category:

If WEIGHT subcommand is not in effect:

Lower bound ($\hat{p}_i$) = *IDF.BETA*($\alpha/2$, $w'_i+.5$, $W'-w'_i+.5$),

Upper bound ($\hat{p}_i$) = *IDF.BETA*($1-\alpha/2$, $w'_i+.5$, $W'-w'_i+.5$)

where *IDF.BETA* is the inverse *Beta* distribution function.

If WEIGHT subcommand is in effect:

Lower bound ($\hat{p}_i$) = *IDF.BETA*($\alpha/2$, $E*p_i+.5$, $E*(1-p_i)+.5$),

Upper bound ($\hat{p}_i$) = *IDF.BETA*($1-\alpha/2$, $E*p_i+.5$, $E*(1-p_i)+.5$)

where *IDF.BETA* is the inverse *Beta* distribution function.

Confidence interval for weighted population percentage

Lower bound ($\hat{p}_i$ %) = 100 * Lower bound ($\hat{p}_i$)

Upper bound ($\hat{p}_i$ %) = 100 * Upper bound ($\hat{p}_i$)

Confidence interval for weighted population count

Lower bound ($\hat{\sum}_i$) = W' * Lower bound ($\hat{p}_i$)

Upper bound ($\hat{\sum}_i$) = W' * Upper bound ($\hat{p}_i$)

Percentiles

This section describes the algorithms used in computing percentiles and confidence intervals for percentiles for scale variables. The following applies to any cell or marginal of the sub-table aside from the sub-table total. Therefore no subscripts for categories or cells are used here. Note that the median is the 50th percentile.

Notation

p	Percentile specified by the user divided by 100.
P	Proportion of data less than or equal to the $100 * p$ th percentile.
W	Sum of weights.
W'	Rounded sum of weights.
q	Sum of squared weights.
w	Cumulative sum of weights for cases less than or equal to the $100 * p$ th percentile.
w'	Rounded cumulative sum of weights for cases less than or equal to the $100 * p$ th percentile
(1- α)%	Confidence interval coverage level supplied by the user.

Conditions and assumptions

- Cases with user and system missing values of scale variables are excluded.

Algorithm

Percentiles

Percentiles

Percentiles are computed using the averaged empirical (AEMPIRICAL) method documented in the statistical algorithms for the EXAMINE procedure.

Confidence Intervals for Percentiles

Confidence intervals for percentiles are computed in a three-step manner, adapted from Shah & Vaish (2012) and Woodruff (1952):

Step 1) Compute the desired percentile.

Step 2) Fit a binomial confidence interval for the proportion of the data less or equal to the estimated percentile ($\hat{p}$):

If there is no WEIGHT subcommand, compute:

$$P_{lower} = \mathit{IDF.BETA}(\alpha / 2 , w'+.5 , W'-w'+.5)$$

$$P_{upper} = \mathit{IDF.BETA}(1 - \alpha / 2 , w'+.5 , W'-w'+.5)$$

where *IDF.BETA* is the inverse *Beta* distribution function.

If there is a WEIGHT subcommand, compute:

$$P = \frac{w}{W}$$

$$P_{lower} = \mathit{IDF.BETA}(\alpha / 2 , E * P + .5 , E * (1 - P) + .5)$$

$$P_{upper} = \mathit{IDF.BETA}(1 - \alpha / 2 , E * P + .5 , E * (1 - P) + .5)$$

where E is the effective base or effective sample size, computed as the sum of weights squared divided by sum of squared weights:

$$E = \frac{W^2}{q}$$

and *IDF.BETA* is the inverse *Beta* distribution function.

Step 3) Apply the percentile-finding algorithm in step 1 to P_{lower} and P_{upper} to obtain lower and upper interval bounds for the percentile.

References:

Shah, B. V., & Vaish, A. K. (2012). Confidence intervals for quantile estimation from complex survey data. *Proceedings of the Joint Statistical Meetings, Section on Survey Methods*, 3720-3728.

Woodruff, R. S. (1952). Confidence intervals for medians and other position measures. *Journal of the American Statistical Association, 47*, 635-646.

Pearson's Chi-square

This section describes computation of Pearson's chi-square statistics.

Notation

R	Number of rows in the sub-table.
C	Number of columns in the sub-table.
f_{ij}	Case weights total in cell (i,j).
r_i	Marginal case weights total in i-th row.
c_j	Marginal case weights total in j-th column.
W	Marginal case weights total in the sub-table.
q	Marginal sum of squared weights in the sub-table.
E_{SS}	Effective sample size or effective base= $\frac{W^2}{q}$
E_{ij}	Expected cell counts.
χ_p^2	Pearson's Chi-Square statistic.
χ_a^2	Pearson's Chi-Square statistic adjusted for effective base weighting.
p_{ij}	Population proportion for cell (i,j).
$p_{i.}$	Marginal population proportion for i-th row.
$p_{.j}$	Marginal population proportion for j-th column.
df	Degrees of Freedom.
p	p-value of the chi-square test.
α	Significance level supplied by the user.

Conditions and assumptions

- Tests will not be performed on Comperimeter tables.
- Chi-square tests are performed on each innermost sub-table of each layer.
- If a scale variable is in the layer, that layer will not be used in analysis.

- The row variable and column variable must be two different categorical variables or multiple response sets.
- The contingency table must have at least two non-empty rows and two non-empty columns.
- Non-empty rows and columns do not include subtotals and totals.
- Empty rows and columns are assumed to be structural zeros. Therefore, R and C are the numbers of non-empty rows and columns in the table.
- If weighting is in effect, cell statistics must include weighted cell counts or weighted simple row/column percents; the analysis will be performed using these weighted cell statistics. If weighting is not in effect, cell statistics must include cell counts or simple row/column percents; the analysis will be unweighted.
- Tests are constructed by using all visible categories. Hiding of categories and showing of user-missing categories are respected.

Algorithm

Pearson's Chi-square

Hypothesis: $H_0 : p_{ij} = p_{i.}p_{.j} \quad i = 1,...,R$ and $j = 1,...,C$ vs. not H_0

Let $E_{ij} = \frac{r_i c_j}{W}$.

Statistic

Categorical variables in rows and columns

$$\chi_p^2 = \sum_{i=1}^{R}\sum_{j=1}^{C}\frac{(f_{ij} - E_{ij})^2}{E_{ij}}.$$

Under the null hypothesis, the statistic has a Chi-square distribution with $df = (R-1)(C-1)$ degrees of freedom.

Categorical variable in rows and multiple response set in columns

$$\chi_p^2 = \sum_{i=1}^{R}\sum_{j=1}^{C}\frac{(f_{ij} - E_{ij})^2}{E_{ij}(1 - \frac{c_j}{W})}.$$

Under the null hypothesis, the statistic has an approximate Chi-square distribution with $df = (R-1)C$ degrees of freedom.

Multiple response set in rows and categorical variable in columns

$$\chi_p^2 = \sum_{i=1}^{R}\sum_{j=1}^{C}\frac{(f_{ij} - E_{ij})^2}{E_{ij}(1 - \frac{r_i}{W})}.$$

Under the null hypothesis, the statistic has an approximate Chi-square distribution with $df = R(C-1)$ degrees of freedom.

Multiple response sets in rows and columns

$$\chi_p^2 = \sum_{i=1}^{R}\sum_{j=1}^{C} \frac{(f_{ij} - E_{ij})^2}{E_{ij}(1-\frac{r_i}{W})(1-\frac{c_j}{W})} .$$

Under the null hypothesis, the statistic has an approximate Chi-square distribution with $df = RC$ degrees of freedom.

P-value

$$p = 1 - F(\chi_p^2, df),$$

where $F(x; df)$ is the cumulative distribution function of Chi-square distribution with df degrees of freedom.

The chi-square test is significant if $p < \alpha$.

Use of weights

If the WEIGHT command is used, the case weights (or frequency weights) are supposed to be integers representing the number of replications of each case. In the chi-square test, we will only check if the aggregated cell counts f_{ij} are integers. If not, they will be rounded to the nearest integers before computations.

If the WEIGHT subcommand is used, the f_{ij} are treated as effective sample size or effective base adjustment weights and need not be integers. The Pearson chi-square statistic is computed as indicated above without rounding aggregated cell counts to integers, then is adjusted by

$$\chi_a^2 = E_{SS}\chi_p^2 = \frac{W}{q}\chi_p^2 .$$

Degrees of freedom and p for χ_a^2 are calculated as with χ_p^2. The chi-square test is significant if $p < \alpha$.

Test statistics for multiple response sets

In the formulas above we use a variation of the Pearson chi-square test statistics developed for a combination of categorical variable and a multiple response set as initially suggested by Agresti and Liu (1999). Formulas and properties of this test can be found in a comparative study by Bilder et al. (2000).

An extension of this approach when both variables are multiple response sets is given in the paper by Thomas and Decady (2004). It contains a study of the test properties as well as additional references.

References

Agresti, A. and Liu, I.-M. (1999), "Modeling responses to a categorical variable allowing arbitrarily many category choices", *Biometrics*, 55, 936-943.

Bilder, C.R., Loughin, T.M. and Nettleton, D. (2000), "Multiple marginal independence testing for pick any/c variables", *Communications in Statistics: Simulation*, 29, 1285-1316.

Thomas, D.R. and Decady, Y.J. (2004), "Testing for association using multiple response survey data: approximate procedures based on Rao-Scott Approach", *International Journal of Testing*, 4, 43-59.

Column Means Tests (No Effective Base Weighting)

This section describes the algorithm used in pairwise comparisons of scale variables over levels of a categorical variable or a multiple response set when effective base weighting is not used.

Notation

k	Number of categories in the sub-table.
k^*	Number of categories with case weights greater than or equal to 2.
μ_i	Population mean of the i-th category, i=1,...,k.
x_{ij}	j-th observation in i-th category, i=1,...,k.
w_{ij}	Case weight of the j-th observation in i-th category, i=1,...,k.
w_i	Sum of case weights in category i, i=1,...,k.
$\tilde{w}_i$	Rounded sum of case weights in category i, i=1,...,k.
$\bar{x}_i$	Mean of category i, i=1,...,k.
s_i	Standard devation of category i, i=1,...,k.
s_{ij}	Pooled standard deviation from i-th and j-th categories.
s_w	Pooled standard deviation from all categories.

W	Total case weights. Sum of rounded w_i's.
α	Significance level supplied by the user.

Conditions and assumptions

- Tests will not be performed for Comperimeter tables.
- Tests are performed on each innermost sub-table for each layer.
- Row variable must be a scale variable, possibly nested under or over some categorical variables or multiple response sets. Column variable must be categorical or a multiple response set.
- If weighting is on, cell statistics must include weighted means; a weighted analysis will be performed using the weighted statistics. If weighting is off, cell statistics must include means; an unweighted analysis will be performed.
- Tests are constructed by using all visible, non-empty categories excluding totals and sub-totals. Hiding of categories and showing of user-missing categories are respected.
- Total case weights in each category must be at least two. Categories not satisfying this assumption are not used. If number of categories satisfying this condition is less than two, no comparisons will be made.
- Population variances of all categories are assumed to be equal.
- User and system missing values of scale variables are excluded.

Algorithm

All Pairwise Comparisons

Hypothesis: $H_{0ij} : \mu_i = \mu_j$, vs. $H_{1ij} : \mu_i \neq \mu_j$, for all $i > j$.

Total number of hypotheses: $\frac{k^*(k^*-1)}{2}$ ***** (where $k^* = \sum_{i=1}^{k} I\,(w_i\,\varepsilon\,2)$).

Note that this assumes that a positive variance estimate can be computed using the specified method (pooling over all categories or over the two categories compared). If the pooled variance estimate using all categories is 0, no comparisons will be made. If the pooled variance estimate using only two categories is 0, this comparison will not be made and the number of hypotheses tested is reduced.

Aggregated statistics

The statistics in pairwise comparisons are computed from aggregated category means ($\bar{x}_i$), sample variances (s_i^2) and sample sizes (w_i), i=1,...,k. Various quantities used in the comparisons are shown below.

Total case weight (sample size): $W = \sum_{i=1}^{k} \tilde{w}_i I(w_i \geq 2)$

Mean of i-th category: $\bar{x}_i = \frac{\sum_{j=1}^{n_i} w_{ij} x_{ij}}{w_i}$

Sample variance of i-th category: $s_i^2 = \frac{\sum_{j=1}^{n_i} w_{ij} (x_{ij} - \bar{x}_i)^2}{w_i - 1}$

Statisitics for (i,j)th comparisons with variance pooled from the two compared categories

Assume $w_i \geq 2$ and $w_j \geq 2$.

Variance pooled from the two compared categories:

$$s_{ij}^2 = \frac{(\tilde{w}_i - 1)s_i^2 + (\tilde{w}_j - 1)s_j^2}{\tilde{w}_i + \tilde{w}_j - 2}.$$

t-statistic for comparing levels of a categorical variable:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{s_{ij}\sqrt{\left(\frac{1}{\tilde{w}_i} + \frac{1}{\tilde{w}_j}\right)}}.$$

P-value $p = 2[1 - F(|t_{ij}|, \tilde{w}_i + \tilde{w}_j - 2)]$, where $F(t, n)$ is the distribution function of t-distribution with n degrees of freedom.

When multiple response set determines categories there may exist cases that belong to both i-th and j-th category. Let $\tilde{w}_{ij}$ be the rounded sum of weights for such cases.

t-statistic for comparing levels of a multiple response set:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{s_{ij}\sqrt{\left(\frac{1}{\tilde{w}_i} + \frac{1}{\tilde{w}_j} - \frac{2\tilde{w}_{ij}}{\tilde{w}_i \tilde{w}_j}\right)}}.$$

P-value $p = 2[1 - F(|t_{ij}|, \tilde{w}_i + \tilde{w}_j - \tilde{w}_{ij} - 2)]$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

Statistics for (i,j)th comparisons with variance pooled from all categories

Within groups variance pooled from all the categories:

$$s_w^2 = \frac{\sum_{i=1}^{k} I(w_i \geq 2)(\tilde{w}_i - 1)s_i^2}{W - k^*}.$$

t-statistic for levels of a categorical variable:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{s_w\sqrt{\left(\frac{1}{\tilde{w}_i} + \frac{1}{\tilde{w}_j}\right)}}.$$

Note: This pooled-variance version of the test is available only for categories defined by a categorical variable (it is not available when categories are defined by a multiple response variable).

P-value $p = 2[1 - F(|t_{ij}|, W - k^*)]$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

Use of case weights

The case weights (or frequency weights) are supposed to be integers representing number of replications of each case. If the sum of case weights in any group (w_i ,i=1,...,k) is not an integer, it will be rounded to the nearest integer before calculations. Consequently, the total weight W will become the sum of the rounded w_i's.

Column Means Tests (With Effective Base Weighting)

This section describes the algorithm used in pairwise comparisons of scale variables over levels of a categorical variable or a multiple response set when effective base weighting is used.

Notation

k	Number of categories in the sub-table.
k^*	Number of categories with at least two unweighted cases $(c_i \geq 2)$.
c_i	Unweighted case count in the i-th category, i=1,...,k.

μ_i	Population mean of the i-th category, i=1,...,k.
x_{il}	l-th observation in i-th category, i=1,...,k.
w_{il}	Weight of the l-th observation in i-th category, i=1,...,k.
w_i	Sum of weights in category i, i=1,...,k.
q_i	Sum of squared weights in category i, i=1,...,k.
e_i	Effective base in category i, i=1,...,k.
$\bar{x}_i$	Weighted mean of category i, i=1,...,k.
s_i	Weighted standard deviation of category i, i=1,...,k.
$\hat{s}_i$	Adjusted weighted standard deviation of category i, i=1,...,k incorporating effective base.
p	p-value of a test.

Conditions and assumptions

- Tests will not be performed for Comperimeter tables.
- Tests are performed on each innermost sub-table for each layer.
- The row variable must be a scale variable, possibly nested under or over some categorical variables or multiple response sets. The column variable must be categorical or a multiple response set.
- Cell statistics must include weighted means.
- Tests are constructed by using all visible, non-empty categories excluding totals and sub-totals. Hiding of categories and showing of user-missing categories are respected.
- In order for two categories to be compared, each must have at least one valid case and at least one of the two categories must have at least two valid cases. If no categories have more than one valid case, no comparisons will be made.
- Population variances of all categories are assumed to be equal.
- User and system missing values of scale variables are excluded.

Algorithm

All Pairwise Comparisons

Hypothesis: $H_{0ij}: \mu_i =\neq \mu_j$ vs. $H_{1ij}: \mu_i \neq \mu_j$ for all $j > i$.

Total number of hypotheses tested:

$\frac{k^*(k^*-1)}{2} + k^*(k-k^*)$ **,** where $k^* = \sum_{i=1}^{k} I\,(c_i \geq 2)$ if the default pooled population variance estimate is used and assuming that this pooled variance estimate is positive. If the population variance estimate is based on only the two categories used in the comparison, $k^* = \sum_{i=1}^{k} I(c_i \geq 2)$.

Note that if all categories have at least two valid cases and a positive variance estimate can be computed, this reduces to the more familiar $\frac{k(k-1)}{2}$.

Aggregated statistics

The statistics in pairwise comparisons are computed from aggregated category means ($\bar{x}_i$), sample variances (s_i^2) and sample sizes (w_i), i=1,...,k. Various quantities used in the comparisons are shown below.

Weighted mean of i-th category: $\bar{x}_i = \frac{\sum_{l=1}^{c_i} w_{il} x_{il}}{w_i}$

Weighted variance of i-th category: $s_i^2 = \frac{\sum_{l=1}^{c_i} w_{il}(x_{il} - \bar{x}_i)^2}{w_i - 1}$

Effective base of i-th category: $e_i = \frac{w_i^2}{q_i}$

Adjusted variance estimate of i-th category incorporating effective base:

$$\hat{s}_i^2 = \frac{\frac{e_i}{w_i}\sum_{l=1}^{c_i} w_{il}(x_{il} - \bar{x}_i)^2}{e_i - 1} = \frac{\sum_{l=1}^{c_i} w_{il}(x_{il} - \bar{x}_i)^2}{w_i - \frac{w_i}{e_i}} = \frac{(w_i - 1)s_i^2}{w_i - \frac{w_i}{e_i}}$$

Statistics for (i,j)th comparisons with variance pooled from the two compared categories

Pooled variance estimate from the two compared categories:

$$\hat{s}_{ij}^2 = \frac{(w_i - 1)s_i^2 + (w_j - 1)s_j^2}{\left(w_i - \frac{w_i}{e_i}\right) + \left(w_j - \frac{w_j}{e_j}\right)}.$$

T-statistic for comparing levels of a categorical variable:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{\hat{s}_{ij}\sqrt{\left(\frac{1}{e_i} + \frac{1}{e_j}\right)}}.$$

p-value: $p = 2[1 - F(|t_{ij}|, e_i + e_j - 2)]$, where $F(t, n)$ is the distribution function of a *t*-distribution with *n* degrees of freedom.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

When a multiple response set determines categories there may exist cases that belong to both the i-th and j-th categories. Let w_{ij} be the sum of weights for such cases. The effective base of such cases is then $e_{ij} = .5 * w_{ij}\left(\frac{e_i}{w_i} + \frac{e_j}{w_j}\right)$.

t-statistic for comparing levels of a multiple response set:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{\hat{s}_{ij}\sqrt{\left(\frac{1}{e_i} + \frac{1}{e_j} - \frac{2e_{ij}}{e_i e_j}\right)}}.$$

p-value $p = 2[1 - F(|t_{ij}|, e_i + e_j - e_{ij} - 2)]$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

Statistics for (i,j)th comparisons with variance pooled from all categories

Within groups variance estimate pooled from all the categories:

$$\hat{s}_w^2 = \frac{\sum_{i=1}^{k} I(c_i \geq 2)(w_i - 1)s_i^2}{\sum_{i=1}^{k} I(c_i \geq 2)\left(w_i - \frac{w_i}{e_i}\right)}.$$

t-statistic for levels of a categorical variable:

$$t_{ij} = \frac{(\bar{x}_i - \bar{x}_j)}{\hat{s}_w \sqrt{\left(\frac{1}{e_i} + \frac{1}{e_j}\right)}} .$$

P-value: $p = 2[1 - F(|t_{ij}|, \sum_{i=1}^{k^*} e_i - k^*)]$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

Note: This pooled-variance version of the test is available only for categories defined by a categorical variable (it is not available when categories are defined by a multiple response variable).

Column Proportions Tests (No Effective Base Weighting)

This document describes the algorithm used in computation of column proportions test when effective base weighting is not in use.

Notation

R	Number of rows in the sub-table.
C	Number of columns in the sub-table.
A_i	i-th category of the row variable.
B_j	j-th category of the column variable.
f_{ij}	Case weights total in cell (i,j).
c_j	Marginal case weights total in j-th column.
$\tilde{c}_j$	Rounded marginal case weights total in j-th column.
z	z-statistic.
χ^2	Chi-Square statistic.
p_{ij}	Column proportion for cell (i,j).
$\hat{p}_{ij}$	Estimated column proportion for cell (i,j).
$\hat{p}_{ijk}$	Estimate of pooled column proportion of j-th and k-th column in i-th row.
p	p-value of a test.
α	The significance level supplied by the user.

Conditions and assumptions

- Tests will not be performed on Comperimeter tables and tables with scale variables in the layer.
- Pairwise tests are performed on each row of all eligible innermost sub-tables within each layer.
- Sub-tables must have categorical variables or multiple response sets in both rows and columns.
- Number of rows and columns must be larger than or equal to two, i.e. $R \geq 2$ and $C \geq 2$.
- Tests are constructed by using all visible categories excluding totals and sub-totals. Hiding of categories and showing of user-missing categories are respected.
- If weighting is on, cell statistics must include weighted cell counts or weighted simple column percents; a weighted analysis will be performed. If weighting is off, cell statistics requested must include cell counts or simple column percents; an unweighted analysis will be performed.
- A proportion will be discarded if the proportion is equal to zero or one, or the sum of case weights in a column is less than 2, (i.e. $c_j < 2$). If less than two proportions are left after discarding proportions, test will not be performed.

Algorithm

All Pairwise Comparisons

Table layout:

	B$_1$	**B$_2$**	**...**	**B$_C$**
A$_1$	p$_{11}$	p$_{12}$		p$_{1C}$
A$_2$	p$_{21}$	p$_{22}$		p$_{2C}$
...	**...**	**...**	**...**	**...**
A$_R$	p$_{R1}$	p$_{R2}$	**...**	p$_{RC}$

Hypothesis:

Without loss of generality, we will only look at the i-th row of the table. Let C^* be the number of categories in the i-th row where the proportion is greater than zero and less than one, and where the sum of case weights in the corresponding column is at least 2. In the i-th row, $C^*(C^*-1)/2$ comparisons will be made among $p_{i1}, p_{i2}, \ldots, p_{iC}$. The (j,k)th hypothesis will be

$H_{0jk}: p_{ij} = p_{ik}$ vs. $H_{1jk}: p_{ij} \neq p_{ik}$.

Aggregated statistics

Column proportions tests are based on the aggregated proportions ($\hat{p}_{ij}$) and cell counts for each column (c_j). Column proportions are computed using the un-rounded cell counts $\hat{p}_{ij} = \frac{f_{ij}}{c_j}$ which are equal to the proportions actually displayed in the table output.

Statistics for the (j,k)th comparisons

Let $\tilde{c}_j = round(c_j)$ and $\tilde{c}_k = round(c_k)$.

Pooled proportion:

$$\hat{p}_{ijk} = \frac{\tilde{c}_j \hat{p}_{ij} + \tilde{c}_k \hat{p}_{ik}}{\tilde{c}_j + \tilde{c}_k} .$$

z statistic with a categorical variable in the columns:

$$z = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1-\hat{p}_{ijk})(\frac{1}{\tilde{c}_j} + \frac{1}{\tilde{c}_k})}} .$$

When multiple response set defines columns there may exist cases that belong to both j-th and k-th columns. Let $\tilde{c}_{jk}$ be the rounded marginal weights total for such cases.

z statistic with a multiple response set in the columns:

$$z = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1-\hat{p}_{ijk})(\frac{1}{\tilde{c}_j} + \frac{1}{\tilde{c}_k} - \frac{2\tilde{c}_{jk}}{\tilde{c}_j \tilde{c}_k})}} .$$

p -value: $p = 2[1 - \phi(|z|)]$, where $\phi(z)$ is the CDF of standard normal distribution.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

Relationship to Pearson's chi-square tests

With a categorical variable in the columns, the statistics used in column proportion tests are equivalent to the Pearson's chi-square test on a 2 x 2 table by taking the j and k-th columns and collapsing all rows except the i-th row. Therefore performing column proportion tests on a 2 x 2 table will give the same result as Pearson's chi-square test.

Use of case weights

The case weights (or frequency weights) are supposed to be integers representing the number of replications of each case. In column proportions tests, we will only check if the column marginal c_j's are integers. If not, they will be rounded to the nearest integers.

Column Proportions Tests (With Effective Base Weighting)

This section describes the algorithms used in computation of column proportions tests when effective sample size or effective base weighting is used.

Notation

R	Number of rows in the sub-table.
C	Number of columns in the sub-table.
n_j	Number of valid unweighted cases in the j-th column of the sub-table.
C^*	Number of categories in the i-th row where the number of valid cases in the corresponding column is at least 2 $(n_j \varepsilon 2)$
A_i	i-th category of the row variable.
B_j	j-th category of the column variable.
w_{ij}	Sum of weights in cell (i,j).
w_j	Marginal sum of weights in j-th column.
q_{ij}	Sum of squared weights in cell (i,j).
e_{ij}	Effective base in cell (i,j).
e_j	Effective base in j-th column.
t	t-statistic.
p_{ij}	Column proportion for cell (i,j).
$\hat{p}_{ij}$	Estimated column proportion for cell (i,j).
$\hat{p}_{ijk}$	Estimate of pooled proportion of j-th and k-th columns in the i-th row.
p	p-value of a test.

Conditions and assumptions

- Tests will not be performed on Comperimeter tables and tables with scale variables in the layer.

- Pairwise tests are performed on each row of all eligible innermost sub-tables within each layer.
- Sub-tables must have categorical variables or multiple response sets in both rows and columns.
- The number of rows and columns must be larger than or equal to two, i.e. $R \geq 2$ and $C \geq 2$.
- Tests are constructed by using all visible categories excluding totals and sub-totals. Hiding of categories and showing of user-missing categories are respected.
- Cell statistics must include weighted cell counts or weighted simple column percents.
- In order for two categories to be compared, each must have at least one valid case and at least one of the two categories must have at least two valid cases. If no categories have more than one valid case, no comparisons will be made.

Algorithm

All Pairwise Comparisons

Table layout:

	C_1	**C_2**	**...**	**C_C**
R_1	p_{11}	p_{12}		p_{1C}
R_2	p_{21}	p_{22}		p_{2C}
...	**...**	**...**	**...**	**...**
R_R	p_{R1}	p_{R2}	**...**	p_{RC}

Hypothesis:

Without loss of generality, we will only look at the i-th row of the table. The (j,k)th hypothesis will be

$H_{0jk}: p_{ij} = p_{ik}$ vs. $H_{1jk}: p_{ij} \neq p_{ik}$ for all $k > j$.

Let C^* be the number of categories in the i-th row where the number of valid unweighted cases is at least two.

Total number of hypotheses tested:

$\frac{C^*(C^*-1)}{2} + C^*(C - C^*)$, where $C^* = \sum_{j=1}^{C} I(n_j \geq 2)$.

Note that if all categories have at least two valid cases this reduces to $\frac{C(C-1)}{2}$.

Aggregated statistics

Column proportions tests are based on the aggregated proportions ($\hat{p}_{ij}$) and cell counts for each column (w_j). Column proportions are computed as $\hat{p}_{ij} = \frac{w_{ij}}{w_j}$.

Statistics for the (j,k)th comparisons

Pooled proportion estimate:

$$\hat{p}_{ijk} = \frac{w_j \hat{p}_{ij} + w_k \hat{p}_{ik}}{w_j + w_k}.$$

Let the effective base for cell (i,j) be $e_{ij} = \frac{(w_{ij})^2}{q_{ij}}$ and the effective base for column j be $e_j = \sum_{i=1}^{R} e_{ij}$. Similarly, let the effective base for cell (i,k) be $e_{ik} = \frac{(w_{ik})^2}{q_{ik}}$ and the effective base for column k be $e_k = \sum_{i=1}^{R} e_{ik}$. (When a multiple response set defines the rows, effective base computations over rows within a column are not additive. In this case the effective base for column k is computed as $e_k = \frac{(w_k)^2}{q_k}$).

t statistic with a categorical variable in the columns:

$$t = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1 - \hat{p}_{ijk})(\frac{1}{e_j} + \frac{1}{e_k})}}.$$

p-value: $p = 2[1 - F(|t|, df)]$, $F(t, df)$ is the CDF of Student's *t* distribution with *df* degrees of freedom. Here $df = e_j + e_k - 2$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

When a multiple response set defines the columns there may exist cases that belong to both the j-th and k-th columns (these cases are said to overlap). Let w_{ijk} be the sum of the weights for the cases in row i that belong to both columns j and k. Then the effective base of these overlapping cases for columns j and k in row i is $e_{ijk} = .5 * w_{ijk}\left(\frac{e_{ij}}{w_{ij}} + \frac{e_{ik}}{w_{ik}}\right)$, and the effective base of these overlapping cases for columns j and k is $e_{jk} = \sum_{i=1}^{R} e_{ijk}$.

t statistic with a multiple response set in the columns:

$$t = \frac{(\hat{p}_{ij} - \hat{p}_{ik})}{\sqrt{\hat{p}_{ijk}(1-\hat{p}_{ijk})(\frac{1}{e_j}+\frac{1}{e_k}-\frac{2e_{jk}}{e_je_k})}} .$$

p -value: $p = 2[1-F(|t|,df)]$, where $F(t,df)$ is the CDF of Student's *t* distribution with df degrees of freedom. Here $df = e_j + e_k - e_{jk} - 2$.

A comparison is significant if $p < \alpha$ if no multiple comparison adjustments are made. For multiple comparison adjustment formulas see the final section.

If rows are also defined by a multiple response set, then overlapping cases cannot be identified individually by rows, so the effective base of the overlapping cases for columns j and k is $e_{jk} = .5 * w_{jk}\left(\frac{e_j}{w_j} + \frac{e_k}{w_k}\right)$, where w_{jk} is the sum of the weights of overlapping cases in columns j and k.

Multiple Comparison Adjustments for Column Means and Column Proportions Tests

This section describes the algorithms used in adjusting p -values or significance levels for pairwise comparisons among column means or proportions.

Notation

m	Number of distinct comparisons performed.
p	Unadjusted p-value of a test.
p_B	Bonferroni corrected p-value.
$p_{BH,i}$	Benjamini-Hochberg adjusted p-value for the i^{th} comparison.
α	The significance level supplied by the user.

Algorithm

Multiple Comparison Adjustments

Unadjusted comparisons

A comparison is significant if $p < \alpha$.

Bonferroni adjustment

If the Bonferroni adjustment for multiple comparisons is requested, the p-value p will be adjusted by

$$p_B = \min(mp, 1) .$$

A comparison is significant if $p_B < \alpha$.

Benjamini-Hochberg False Discovery Rate Procedure

If the Benjamini-Hochberg adjustment for multiple comparisons is requested, the method from Benjamini & Hochberg (1995) for controlling the false discovery rate (FDR) is used.

Statistically significant comparisons

Sort the unadjusted p-values from $i = 1,..., m$ in ascending order.

Find the largest unadjusted p-value p_k for which

$$p_i \le \frac{i}{m} \alpha.$$

Then all comparisons associated with $p_i = p_1,..., p_k$ are declared significant.

Adjusted p-values

The adjusted p-value $p_{BH,i}$ for the i^{th} comparison is computed as:

$$p_{BH,i} = p \text{ if } i = m$$

$$p_{BH,i} = \min(p_{BH,i+1}, \frac{m}{i} p_i) \text{ if } i < m .$$

Reference:

Benjamini, Y., & Hochberg, Y. (1995). Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society B, 57*(1), 289-300.

CURVEFIT Algorithms

Eleven models can be selected to fit times series and produce forecasts, forecast errors, and confidence limits. In all of the models, the observed series is some function of time.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 34-1
Notation

Notation	**Description**
Y_t	Observed series; $t = 1, \ldots, n$
$E(Y_t)$	Expected value of Y_t
$\hat{Y}_t$	Predicted value for Y_t

Models

CURVEFIT allows the user to specify a model with or without a constant term designated by β_0. If this constant term is excluded, simply set it zero or one depending upon whether it appears in an additive or multiplicative manner in the models listed below.

Model	**Description**
(1) Linear	$E(Y_t) = \beta_0 + \beta_1 t$
(2) Logarithmic	$E(Y_t) = \beta_0 + \beta_1 \ln(t)$
(3) Inverse	$E(Y_t) = \beta_0 + \beta_1 / t$
(4) Quadratic	$E(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2$
(5) Cubic	$E(Y_t) = \beta_0 + \beta_1 t + \beta_2 t^2 + \beta_3 t^3$
(6) Compound	$E(Y_t) = \beta_0 \beta_1^t$
(7) Power	$E(Y_t) = \beta_0 t^{\beta_1}$
(8) S	$E(Y_t) = \exp(\beta_0 + \beta_1 / t)$
(9) Growth	$E(Y_t) = \exp(\beta_0 + \beta_1 t)$
(10) Exponential	$E(Y_t) = \beta_0 e^{\beta_1 t}$
(11) Logistic	$E(Y_t) = \left(\frac{1}{u} + \beta_0 \beta_1^t\right)^{-1}$

Assumption

We assume that nonlinear models (6) to (11) can be expressed in linear model form by logarithmic transformation. So, for models (6) to (10),

$$\ln(Y_t) = \ln(E(Y_t)) + \epsilon_t$$

and for model (11),

$$\ln\left(\frac{1}{Y_t} - \frac{1}{u}\right) = \ln\left(\frac{1}{E(Y_t)} - \frac{1}{u}\right) + \epsilon_t$$

with $\epsilon_t, t = 1, \ldots, n$ being independently identically distributed $N\left(0, \sigma^2\right)$.

Application of Regression

Each of the models is expressed in linear form and computational techniques described in the REGRESSION procedure are applied. The dependent variable and independent variables for each model are listed as follows:

Model	**Dependent Variable**	**Independent Variables**	**Coefficients**
(1)	Y	t	β_0, β_1
(2)	Y	$\ln(t)$	β_0, β_1
(3)	Y	$1/t$	β_0, β_1
(4)	Y	t, t^2	$\beta_0, \beta_1, \beta_2$
(5)	Y	t, t^2, t^3	$\beta_0, \beta_1, \beta_2, \beta_3$
(6)	$\ln(Y)$	t	β_0^*, β_1^*
(7)	$\ln(Y)$	$\ln(t)$	β_0^*, β_1
(8)	$\ln(Y)$	$1/t$	β_0, β_1
(9)	$\ln(Y)$	t	β_0, β_1
(10)	$\ln(Y)$	t	β_0^*, β_1
(11)	$\ln\left(\frac{1}{Y} - \frac{1}{u}\right)$	t	β_0^*, β_1^*

where $\beta_0^* = \ln(\beta_0)$ and $\beta_1^* = \ln(\beta_1)$.

The ANOVA table, coefficient estimates and their standard errors, *t*-values, and significance levels are computed as in the REGRESSION procedure. Note that for the nonlinear models (6) to (11), we have

$$se\left(\hat{\beta}_0\right) \approx \exp\left(\hat{\beta}_0^*\right) \times se\left(\hat{\beta}_0^*\right)$$

and

$$se\left(\hat{\beta}_1\right) \approx \exp\left(\hat{\beta}_1^*\right) \times se\left(\hat{\beta}_1^*\right)$$

Predicted Values and Confidence Intervals

The regression coefficients for models (1) to (5) are used to obtain the predicted values. For the transformed models, more computations are required to obtain the predicted values for the original models. The formulas are listed below:

Model Description

(1) $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 t$

(2) $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 \ln(t)$

(3) $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 / t$

(4) $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 t + \hat{\beta}_2 t^2$

(5) $\hat{Y}_t = \hat{\beta}_0 + \hat{\beta}_1 t + \hat{\beta}_2 t^2 + \hat{\beta}_3 t^3$

(6) $\hat{Y}_t^* = \hat{\beta}_0^* + \hat{\beta}_1^* t$

(7) $\hat{Y}_t^* = \hat{\beta}_0^* + \hat{\beta}_1 \ln(t)$

(8) $\hat{Y}_t^* = \hat{\beta}_0 + \hat{\beta}_1 / t$

(9) $\hat{Y}_t^* = \hat{\beta}_0 + \hat{\beta}_1 t$

(10) $\hat{Y}_t^* = \hat{\beta}_0^* + \hat{\beta}_1 t$

(11) $\hat{Y}_t^* = \hat{\beta}_0^* + \hat{\beta}_1^* t$

where $\hat{Y}_t^* = \ln\left(\hat{Y}_t\right)$ in models (5) to (10), and $\hat{Y}_t^* = \ln\left(\frac{1}{\hat{Y}_t} - \frac{1}{u}\right)$ in model (11).

The 95% prediction interval for an observation at time *t* is constructed as follows:

For models (1) to (5):

$\hat{Y}_t \pm t_{0.025}\sqrt{MSE\left(1 + h_t + \frac{1}{n}\right)}$ if constant term is included

$\hat{Y}_t \pm t_{0.025}\sqrt{MSE(1 + h_t)}$ otherwise

For models (6) to (10):

$$\exp\left(\hat{Y}_t^* \pm t_{0.025}\sqrt{MSE\left(1 + h_t + \frac{1}{n}\right)}\right)$$

and for model (11):

$$\frac{1}{\exp\left(\hat{Y}_t^* \pm t_{0.025}\sqrt{MSE\left(1 + h_t + \frac{1}{n}\right)}\right) + \frac{1}{u}}$$

where MSE is the mean square error obtained by fitting the linear model, $t_{0.025}$ is the 97.5 percentage point from Student *t*-distribution with MSE degrees of freedom, and h_t is the leverage (computational detail in the REGRESSION procedure).

DESCRIPTIVES Algorithms

DESCRIPTIVES computes univariate statistics—including the mean, standard deviation, minimum, and maximum—for numeric variables.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 35-1
Notation

Notation	Description
X_i	Value of the variable for case i
w_i	Weight for case i
N	Number of cases
W_i	Sum of the weights for the first i cases
$\overline{X}_i$	Mean for the first i cases

Moments

Moments about the mean are calculated recursively using a provisional means algorithm (Spicer, 1972):

$$W_j = \sum_{i=1}^{j} w_i$$

$$v_j = \frac{w_j}{W_j}\left(X_j - \overline{X}_{j-1}\right)$$

$$M_j^4 = M_{j-1}^4 - 4v_j M_{j-1}^3 + 6v_j^2 M_{j-1}^2 + \left(\frac{W_j^2 - 3w_j W_{j-1}}{w_j^3}\right) v_j^4 W_{j-1} W_j$$

$$M_j^3 = M_{j-1}^3 - 3v_j M_{j-1}^2 + \frac{W_j W_{j-1}}{w_j^2}(W_j - 2w_j) v_j^3$$

$$M_j^2 = M_{j-1}^2 + \frac{W_j W_{j-1}}{w_j} v_j^2$$

$$\overline{X}_j = \overline{X}_{j-1} + v_j$$

$$W_0 = \overline{X}_0 = M_0^2 = M_0^3 = M_0^4 = 0$$

After the last observation has been processed,

$W_N =$ sum of weights for all cases

$\overline{X}_N$ = mean

$$M_N^r = \sum_{i=1}^{N} w_i \left(X_i - \overline{X}\right)^r$$

Basic Statistics

Mean

$$\overline{X}_N$$

Variance

$$S^2 = M_N^2/(W_N - 1)$$

Standard Deviation

$$S = \sqrt{S^2}$$

Standard Error

$$S_{\overline{X}} = \frac{S}{\sqrt{W_N}}$$

Minimum

$$\min_j X_j$$

Maximum

$$\max_j X_j$$

Sum

$$\overline{X}_N W_N$$

Skewness and Standard Error of Skewness

$$g_1 = \frac{W_N M_N^3}{(W_N - 1)(W_N - 2)S^3} \qquad se(g_1) = \sqrt{\frac{6W_N(W_N - 1)}{(W_N - 2)(W_N + 1)(W_N + 3)}}$$

If $W_N \le 2$ or $S^2 < 10^{-20}$ g_1 and its standard error are not calculated.

Kurtosis (Bliss, 1967, p. 144) and Standard Error of Kurtosis

$$g_2 = \frac{W_N(W_N+1)M_N^4 - 3M_N^2 M_N^2 (W_N-1)}{(W_N-1)(W_N-2)(W_N-3)S^4} \quad se(g_2) = \sqrt{\frac{4(W_N^2-1)(SE(g_1))^2}{(W_N-3)(W_N+5)}}$$

If $W_N \leq 3$ or $S^2 < 10^{-20} g_2$ and its standard error are not calculated.

Z-Scores

$$Z_i = \frac{X_i - \overline{X}_N}{S}$$

If X_i is missing or $S \leq 0$, Z_i is set to the system missing value.

References

Bliss, C. I. 1967. *Statistics in biology, Volume 1*. New York: McGraw-Hill.

Spicer, C. C. 1972. Algorithm AS 52: Calculation of power sums of deviations about the mean. *Applied Statistics*, 21, 226–227.

DETECTANOMALY Algorithms

The Anomaly Detection procedure searches for unusual cases based on deviations from the norms of their cluster groups. The procedure is designed to quickly detect unusual cases for data-auditing purposes in the exploratory data analysis step, prior to any inferential data analysis. This algorithm is designed for generic anomaly detection; that is, the definition of an anomalous case is not specific to any particular application, such as detection of unusual payment patterns in the healthcare industry or detection of money laundering in the finance industry, in which the definition of an anomaly can be well-defined.

Data Assumptions

Data. This procedure works with both continuous and categorical variables. Each row represents a distinct observation, and each column represents a distinct variable upon which the peer groups are based. A case identification variable can be available in the data file for marking output, but it will not be used in the analysis. Missing values are allowed. The weight variable, if specified, is ignored.

The detection model can be applied to a new test data file. The elements of the test data must be the same as the elements of the training data. And, depending on the algorithm settings, the missing value handling that is used to create the model may be applied to the test data file prior to scoring.

Case order. Note that the solution may depend on the order of cases. To minimize order effects, randomly order the cases. To verify the stability of a given solution, you may want to obtain several different solutions with cases sorted in different random orders. In situations with extremely large file sizes, multiple runs can be performed with a sample of cases sorted in different random orders.

Assumptions. The algorithm assumes that all variables are nonconstant and independent and that no case has missing values for any of the input variables. Each continuous variable is assumed to have a normal (Gaussian) distribution, and each categorical variable is assumed to have a multinomial distribution. Empirical internal testing indicates that the procedure is fairly robust to violations of both the assumption of independence and the distributional assumptions, but be aware of how well these assumptions are met.

Notation

The following notation is used throughout this chapter unless otherwise stated:

ID	The identity variable of each case in the data file.
n	The number of cases in the training data X_{train} .
X_{ok}, k = 1, …, K	The set of input variables in the training data.
M_k, $k \in \{1, …, K\}$	If X_{ok} is a continuous variable, M_k represents the grand mean, or average of the variable across the entire training data.
SD_k, $k \in \{1, …, K\}$	If X_{ok} is a continuous variable, SD_k represents the grand standard deviation, or standard deviation of the variable across the entire training data.
X_{K+1}	A continuous variable created in the analysis. It represents the percentage of variables (k = 1, …, K) that have missing values in each case.
X_k, k = 1, …, K	The set of processed input variables after the missing value handling is applied. For more information, see the topic “Modeling Stage”.

H, or the boundaries of H: $[H_{min}, H_{max}]$	H is the pre-specified number of cluster groups to create. Alternatively, the bounds $[H_{min}, H_{max}]$ can be used to specify the minimum and maximum numbers of cluster groups.
n_h, h = 1, …, H	The number of cases in cluster h, h = 1, …, H, based on the training data.
p_h, h = 1, …, H	The proportion of cases in cluster h, h = 1, …, H, based on the training data. For each h, $p_h = n_h/n$.
M_{hk}, k = 1, …, K+1, h = 1, …, H	If X_k is a continuous variable, M_{hk} represents the cluster mean, or average of the variable in cluster h based on the training data. If X_k is a categorical variable, it represents the cluster mode, or most popular categorical value of the variable in cluster h based on the training data.
SD_{hk}, $k \in \{1, …, K+1\}$, h = 1, …, H	If X_k is a continuous variable, SD_{hk} represents the cluster standard deviation, or standard deviation of the variable in cluster h based on the training data.
$\{n_{hkj}\}$, $k \in \{1, …, K\}$, h = 1, …, H, j = 1, …, J_k	The frequency set $\{n_{hkj}\}$ is defined only when X_k is a categorical variable. If X_k has J_k categories, then n_{hkj} is the number of cases in cluster h that fall into category j.
m	An adjustment weight used to balance the influence between continuous and categorical variables. It is a positive value with a default of 6.
VDI_k, k = 1, …, K+1	The variable deviation index of a case is a measure of the deviation of variable value X_k from its cluster norm.
GDI	The group deviation index GDI of a case is the log-likelihood distance d(h, s), which is the sum of all of the variable deviation indices $\{VDI_k$, k = 1, …, K+1$\}$.
anomaly index	The anomaly index of a case is the ratio of the GDI to that of the average GDI for the cluster group to which the case belongs.
variable contribution measure	The variable contribution measure of variable X_k for a case is the ratio of the VDI_k to the case's corresponding GDI.
$pct_{anomaly}$ or $n_{anomaly}$	A pre-specified value $pct_{anomaly}$ determines the percentage of cases to be considered as anomalies. Alternatively, a pre-specified positive integer value $n_{anomaly}$ determines the number of cases to be considered as anomalies.
$cutpoint_{anomaly}$	A pre-specified cutpoint; cases with anomaly index values greater than $cutpoint_{anomaly}$ are considered anomalous.
$k_{anomaly}$	A pre-specified integer threshold $1 \le k_{anomaly} \le K+1$ determines the number of variables considered as the reasons that the case is identified as an anomaly.

Algorithm Steps

This algorithm is divided into three stages:

Modeling. Cases are placed into cluster groups based on their similarities on a set of input variables. The clustering model used to determine the cluster group of a case and the sufficient statistics used to calculate the norms of the cluster groups are stored.

Scoring. The model is applied to each case to identify its cluster group and some indices are created for each case to measure the unusualness of the case with respect to its cluster group. All cases are sorted by the values of the anomaly indices. The top portion of the case list is identified as the set of anomalies.

Reasoning. For each anomalous case, the variables are sorted by its corresponding variable deviation indices. The top variables, their values, and the corresponding norm values are presented as the reasons why a case is identified as an anomaly.

Modeling Stage

This stage performs the following tasks:

1. **Training Set Formation.** Starting with the specified variables and cases, remove any case with extremely large values (greater than 1.0E+150) on any continuous variable. If missing value handling is not in effect, also remove cases with a missing value on any variable. Remove variables with all constant nonmissing values or all missing values. The remaining cases and variables are used to create the anomaly detection model. Statistics output to pivot table by the procedure are based on this training set, but variables saved to the dataset are computed for all cases.

2. **Missing Value Handling (Optional).** For each input variable X_{ok}, k = 1, …, K, if X_{ok} is a continuous variable, use all valid values of that variable to compute the grand mean M_k and grand standard deviation SD_k. Replace the missing values of the variable by its grand mean. If X_{ok} is a categorical variable, combine all missing values into a "missing value" category. This category is treated as a valid category. Denote the processed form of $\{X_{ok}\}$ by $\{X_k\}$.

3. **Creation of Missing Value Pct Variable (Optional).** A new continuous variable, X_{K+1}, is created that represents the percentage of variables (both continuous and categorical) with missing values in each case.

4. **Cluster Group Identification.** The processed input variables $\{X_k, k = 1, …, K+1\}$ are used to create a clustering model. The two-step clustering algorithm is used with noise handling turned on (see the TwoStep Cluster algorithm document for more information).

5. **Sufficient Statistics Storage.** The cluster model and the sufficient statistics for the variables by cluster are stored for the Scoring stage:
   - The grand mean M_k and standard deviation SD_k of each continuous variable are stored, $k \in \{1, …, K+1\}$.
   - For each cluster h = 1, …, H, store the size n_h. If X_k is a continuous variable, store the cluster mean M_{hk} and standard deviation SD_{hk} of the variable based on the cases in cluster h. If X_k is a categorical variable, store the frequency n_{hkj} of each category j of the variable based on the cases in cluster h. Also store the modal category M_{hk}. These sufficient statistics will be used in calculating the log-likelihood distance d(h, s) between a cluster h and a given case s.

Scoring Stage

This stage performs the following tasks on scoring (testing or training) data:

1. **New Valid Category Screening.** The scoring data should contain the input variables $\{X_{ok}, k= 1, …, K\}$ in the training data. Moreover, the format of the variables in the scoring data should be the same as those in the training data file during the Modeling Stage.

   Cases in the scoring data are screened out if they contain a categorical variable with a valid category that does not appear in the training data. For example, if *Region* is a categorical variable with categories IL, MA and CA in the training data, a case in the scoring data that has a valid category FL for *Region* will be excluded from the analysis.

2. **Missing Value Handling (Optional).** For each input variable X_{ok}, if X_{ok} is a continuous variable, use all valid values of that variable to compute the grand mean M_k and grand standard deviation SD_k. Replace the missing values of the variable by its grand mean. If X_{ok} is a categorical variable, combine all missing values and put together a missing value category. This category is treated as a valid category.

3. **Creation of Missing Value Pct Variable (Optional depending on Modeling Stage).** If X_{K+1} is created in the Modeling Stage, it is also computed for the scoring data.

4. **Assign Each Case to its Closest Non-Noise Cluster.** The clustering model from the Modeling Stage is applied to the processed variables of the scoring data file to create a cluster ID for each case. Cases belonging to the noise cluster are reassigned to their closest non-noise cluster. See the TwoStep Cluster algorithm document for more information on the noise cluster.

5. **Calculate Variable Deviation Indices.** Given a case s, the closest cluster h is found. The variable deviation index VDI_k of variable X_k is defined as the contribution $d_k(h, s)$ of the variable to its log-likelihood distance d(h, s). The corresponding norm value is M_{hk}, which is the cluster sample mean of X_k if X_k is continuous, or the cluster mode of X_k if X_k is categorical.

6. **Calculate Group Deviation Index.** The group deviation index GDI of a case is the log-likelihood distance d(h, s), which is the sum of all the variable deviation indices $\{VDI_k, k = 1, \ldots, K+1\}$.

7. **Calculate Anomaly Index and Variable Contribution Measures.** Two additional indices are calculated that are easier to interpret than the group deviation index and the variable deviation index.

   The anomaly index of a case is an alternative to the GDI, which is computed as the ratio of the case's GDI to the average GDI of the cluster to which the case belongs. Increasing values of this index correspond to greater deviations from the average and indicate better anomaly candidates.

   A variable's variable contribution measure of a case is an alternative to the VDI, which is computed as the ratio of the variable's VDI to the case's GDI. This is the proportional contribution of the variable to the deviation of the case. The larger the value of this measure, the greater the variable's contribution to the deviation.

Odd Situations

Zero Divided by Zero

The situation in which the GDI of a case is zero and the average GDI of the cluster that the case belongs to is also zero is possible if the cluster is a singleton or is made up of identical cases and the case in question is the same as the identical cases. Whether this case is considered as an anomaly or not depends on whether the number of identical cases that make up the cluster is large or small. For example, suppose that there is a total of 10 cases in the training and two clusters are resulted in which one cluster is a singleton; that is, made up of one case, and the other has nine cases. In this situation, the case in the singleton cluster should be considered as an anomaly as it does not belong to the larger cluster. One way to calculate the anomaly index in this situation is to set it as the ratio of average cluster size to the size of the cluster *h*, which is:

$$\frac{n/H}{n_h}$$

Following the 10 cases example, the anomaly index for the case belonging to the singleton cluster would be (10/2)/1 = 5, which should be large enough for the algorithm to catch it as an anomaly. In this situation, the variable contribution measure is set to 1/(K+1), where (K+1) is the number of processed variables in the analysis.

Nonzero Divided by Zero

The situation in which the GDI of a case is nonzero but the average GDI of the cluster that the case belongs to is 0 is possible if the corresponding cluster is a singleton or is made up of identical cases and the case in question is not the same as the identical cases. Suppose that case *i* belongs to cluster *h*, which has a zero average GDI; that is, average$(GDI)_h = 0$, but the GDI between case *i* and cluster *h* is nonzero; that is, GDI(i, h) ≠ 0. One choice for the anomaly index calculation of case *i* could be to set the denominator as the weighted average GDI over all other clusters if this value is not 0; else set the calculation as the ratio of average cluster size to the size of cluster *h*. That is,

$$\begin{cases} \dfrac{GDI(i,h)}{\frac{1}{(n-n_h)}\Sigma_{s=1,\neq h}^{H} n_s \cdot average(GDI)_s} & \text{if } \frac{1}{(n-n_h)}\sum_{s=1,\neq h}^{H} n_s \cdot average(GDI)_s \neq 0 \\ \dfrac{n/H}{n_h} & \text{otherwise} \end{cases}$$

This situation triggers a warning that the case is assigned to a cluster that is made up of identical cases.

Reasoning Stage

Every case now has a group deviation index and anomaly index and a set of variable deviation indices and variable contribution measures. The purpose of this stage is to rank the likely anomalous cases and provide the reasons to suspect them of being anomalous.

1. **Identify the Most Anomalous Cases.** Sort the cases in descending order on the values of the anomaly index. The top $pct_{anomaly}$ % (or alternatively, the top $n_{anomaly}$) gives the anomaly list, subject
to the restriction that cases with an anomaly index less than or equal to $cutpoint_{anomaly}$ are not considered anomalous.

2. **Provide Reasons for Considering a Case Anomalous.** For each anomalous case, sort the variables by their corresponding VDI_k values in descending order. The top $k_{anomaly}$ variable names, its value (of the corresponding original variable X_{ok}), and the norm values are displayed as reasoning.

Key Formulas from Two-Step Clustering

The two-step clustering algorithm consists of: (a) a pre-cluster step that pre-clusters cases into many sub-clusters and (b) a cluster step that clusters the sub-clusters resulting from pre-cluster step into the desired number of clusters. It can also select the number of clusters automatically.

The formula for the log-likelihood distance d(j, s) between 2 clusters j and s is as follows:

$$d(j,s) = \xi_j + \xi_s - \xi_{<j,s>}$$

where

$$\xi_v = -N_v\left(\Sigma_{k=1}^{K^A}\log\left(\Delta_k + \hat{\sigma}_{vk}^2\right)/2 + \Sigma_{k=1}^{K^B}\hat{E}_{vk}\right)$$

and

$$\hat{E}_{vk} = -\Sigma_{l=1}^{L_k} N_{vkl}/N_v \log\left(N_{vkl}/N_v\right)$$

in which $\Delta_k > 0$ is a positive adjustment included in the formula to avoid the logarithm of zero in the calculation. Its value is set as:

$$\Delta_k = \frac{\hat{\sigma}_k^2}{m}$$

where m is user-specified and set to m = 6 by default, and $\hat{\sigma}_k^2$ is the sample variance of variable X_k over the entire training sample.

The log-likelihood distance can be computed as follows:

$$d\left(j,s\right) = \Sigma_{k=1}^{K^A+K^B} d_k\left(j,s\right)$$

where

$$d_k\left(j,s\right) = \begin{cases} \left\{-N_j\log\left(\Delta_k + \hat{\sigma}_{jk}^2\right) - N_s\log\left(\Delta_k + \hat{\sigma}_{sk}^2\right) + N_{<j,s>}\log\left(\Delta_k + \hat{\sigma}_{<j,s>k}^2\right)\right\}/2 \\ \left\{-N_j\hat{E}_{jk} - N_s\hat{E}_{sk} + N_{<j,s>}\hat{E}_{<j,s>k}\right\} \end{cases}$$

depending on whether the corresponding variable X_k is continuous or categorical.

For more information, see the topic “TWOSTEP CLUSTER Algorithms”.

DISCRIMINANT Algorithms

No analysis is done for any subfile group for which the number of non-empty groups is less than two or the number of cases or sum of weights fails to exceed the number of non-empty groups. An analysis may be stopped if no variables are selected during variable selection or the eigenanalysis fails.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 37-1
Notation

Notation	Description
g	Number of groups
p	Number of variables
q	Number of variables selected
X_{ijk}	Value of variable *i* for case *k* in group *j*
f_{jk}	Case weights for case *k* in group *j*
m_j	Number of cases in group *j*
n_j	Sum of case weights in group *j*
n	Total sum of weights

Basic Statistics

The procedure calculates the following basic statistics.

Mean

$$\overline{X}_{ij} = \left(\sum_{k=1}^{m_j} f_{jk} X_{ijk} \right) / n_j \qquad \text{(variable } i \text{ in group } j\text{)}$$

$$\overline{X}_{i\bullet} = \left(\sum_{j=1}^{g} \sum_{k=1}^{m_j} f_{jk} X_{ijk} \right) / n \qquad \text{(variable } i\text{)}$$

Variances

$$S_{ij}^2 = \frac{\left(\sum_{k=1}^{m_j} f_{jk} X_{ijk}^2 - n_j \overline{X}_{ij}^2 \right)}{(n_j - 1)} \qquad \text{(variable } i \text{ in group } j\text{)}$$

$$S_{i\bullet}^2 = \frac{\left(\sum_{j=1}^{g} \sum_{k=1}^{m_j} f_{jk} X_{ijk}^2 - n \overline{X}_{i}^2 \right)}{(n - 1)} \qquad \text{(variable } i\text{)}$$

Within-Groups Sums of Squares and Cross-Product Matrix (W)

$$w_{il} = \sum_{j=1}^{g}\sum_{k=1}^{m_j} f_{jk}X_{ijk}X_{ljk} - \sum_{j=1}^{g}\left(\sum_{k=1}^{m_j} f_{jk}X_{ijk}\right)\left(\sum_{k=1}^{m_j} f_{jk}X_{ljk}\right)/n_j \quad i,l = 1,\ldots,p$$

Total Sums of Squares and Cross-Product Matrix (T)

$$t_{il} = \sum_{j=1}^{g}\sum_{k=1}^{m_j} f_{jk}X_{ijk}X_{ljk} - \left(\sum_{j=1}^{g}\sum_{k=1}^{m_j} f_{jk}X_{ijk}\right)\left(\sum_{j=1}^{g}\sum_{k=1}^{m_j} f_{jk}X_{ljk}\right)/n$$

Within-Groups Covariance Matrix

$$\mathbf{C} = \frac{\mathbf{W}}{(n-g)} \quad n > g$$

Individual Group Covariance Matrices

$$c_{il}^{(j)} = \frac{\left(\sum_{k=1}^{m_j} f_{jk}X_{ijk}X_{ljk} - \overline{X}_{ij}\overline{X}_{lj}n_j\right)}{(n_j-1)}$$

Within-Groups Correlation Matrix (R)

$$r_{il} = \begin{cases} \frac{w_{il}}{\sqrt{w_{ii}w_{ll}}} & \text{if } w_{ii}w_{ll} > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Total Covariance Matrix

$$\mathbf{T}' = \frac{\mathbf{T}}{n-1}$$

Univariate F and $\Lambda$for Variable I

$$F_i = \frac{(t_{ii}-w_{ii})(n-g)}{w_{ii}(g-1)}$$

with g−1 and n−g degrees of freedom

$$\Lambda_i = \frac{w_{ii}}{t_{ii}}$$

with 1, g−1 and n−g degrees of freedom

Rules of Variable Selection

Both direct and stepwise variable entry are possible. Multiple inclusion levels may also be specified.

Method = Direct

For direct variable selection, variables are considered for inclusion in the order in which they are written on the ANALYSIS = list. A variable is included in the analysis if, when it is included, no variable in the analysis will have a tolerance less than the specified tolerance limit (default = 0.001).

Stepwise Variable Selection

At each step, the following rules control variable selection:

- Eligible variables with higher inclusion levels are entered before eligible variables with lower inclusion levels.
- The order of entry of eligible variables with the same even inclusion level is determined by their order on the ANALYSIS = specification.
- The order of entry of eligible variables with the same odd level of inclusion is determined by their value on the entry criterion. The variable with the “best” value for the criterion statistic is entered first.
- When level-one processing is reached, prior to inclusion of any eligible variables, all already-entered variables which have level one inclusion numbers are examined for removal. A variable is considered eligible for removal if its *F*-to-remove is less than the *F* value for variable removal, or, if probability criteria are used, the significance of its *F*-to-remove exceeds the specified probability level. If more than one variable is eligible for removal, that variable is removed that leaves the “best” value for the criterion statistic for the remaining variables. Variable removal continues until no more variables are eligible for removal. Sequential entry of variables then proceeds as described previously, except that after each step, variables with inclusion numbers of one are also considered for exclusion as described before.
- A variable with a zero inclusion level is never entered, although some statistics for it are printed.

Ineligibility for Inclusion

A variable with an odd inclusion number is considered ineligible for inclusion if:

- The tolerance of any variable in the analysis (including its own) drops below the specified tolerance limit if it is entered, or
- Its *F*-to-enter is less than the *F*-value for a variable to enter value, or
- If probability criteria are used, the significance level associated with its *F*-to-enter exceeds the probability to enter.

A variable with an even inclusion number is ineligible for inclusion if the first condition above is met.

Computations During Variable Selection

During variable selection, the matrix **W** is replaced at each step by a new matrix $\mathbf{W}^*$ using the symmetric sweep operator described by Dempster (1969). If the first *q* variables have been included in the analysis, **W** may be partitioned as:

$$\begin{bmatrix} \mathbf{W}_{11} & \mathbf{W}_{12} \\ \mathbf{W}_{21} & \mathbf{W}_{22} \end{bmatrix}$$

where $\mathbf{W}_{11}$ is $q \times q$. At this stage, the matrix $\mathbf{W}^*$ is defined by

$$\mathbf{W}^* = \begin{bmatrix} -\mathbf{W}_{11}^{-1} & \mathbf{W}_{11}^{-1}\mathbf{W}_{12} \\ \mathbf{W}_{21}\mathbf{W}_{11}^{-1} & \mathbf{W}_{22} - \mathbf{W}_{21}\mathbf{W}_{11}^{-1}\mathbf{W}_{12} \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{11}^* & \mathbf{W}_{12}^* \\ \mathbf{W}_{21}^* & \mathbf{W}_{22}^* \end{bmatrix}$$

In addition, when stepwise variable selection is used, *T* is replaced by the matrix T^*, defined similarly.

The following statistics are computed.

Tolerance

$$\text{TOL}_i = \begin{cases} 0 & \text{if } w_{ii} = 0 \\ w_{ii}^*/w_{ii} & \text{if variable } i \text{ is not in the analysis and } w_{ii} \neq 0 \\ -1/(w_{ii}^* w_{ii}) & \text{if variable } i \text{ is in the analysis and } w_{ii} \neq 0. \end{cases}$$

If a variable's tolerance is less than or equal to the specified tolerance limit, or its inclusion in the analysis would reduce the tolerance of another variable in the equation to or below the limit, the following statistics are not computed for it or any set including it.

F-to-Remove

$$F_i = \frac{(w_{ii}^* - t_{ii}^*)(n-q-g+1)}{t_{ii}^*(g-1)}$$

with degrees of freedom *g*−1 and *n*−*q*−*g*+1.

F-to-Enter

$$F_i = \frac{(t_{ii}^* - w_{ii}^*)(n-q-g)}{w_{ii}^*(g-1)}$$

with degrees of freedom *g*−1 and *n*−*q*−*g*.

Wilks' Lambda for Testing the Equality of Group Means

$$\Lambda = |\mathbf{W}_{11}|/|\mathbf{T}_{11}|$$

with degrees of freedom *q*, *g*−1 and *n*−*g*.

The Approximate F Test for Lambda (the “overall F”), also known as Rao’s R (Tatsuoka, 1971)

$$F = \frac{(1-\Lambda^s)(r/s+1-qh/2)}{\Lambda^s qh}$$

where

$$s = \begin{cases} \sqrt{\frac{q^2+h^2-5}{q^2h^2 \quad 4}} & \text{if } q^2+h^2 \quad 5 \\ 1 & \neq \text{otherwise} \end{cases}$$
$$r = n-1-(q+g)/2$$
$$h = g-1$$

with degrees of freedom qh and $r/s+1-qh/2$. The approximation is exact if q or h is 1 or 2.

Rao’s V (Lawley-Hotelling Trace) (Rao, 1952; Morrison, 1976)

$$V = -(n-g)\sum_{i=1}^{q}\sum_{l=1}^{q} w_{il}^*(t_{il}-w_{il})$$

When $n-g$ is large, V, under the null hypothesis, is approximately distributed as χ^2 with $q(g-1)$ degrees of freedom. When an additional variable is entered, the change in V, if positive, has approximately a χ^2 distribution with $g-1$ degrees of freedom.

The Squared Mahalanobis Distance (Morrison, 1976) between groups a and b

$$D_{ab}^2 = -(n-g)\sum_{i=1}^{q}\sum_{l=1}^{q} w_{il}^*\left(\overline{X}_{ia}-\overline{X}_{ib}\right)\left(\overline{X}_{la}-\overline{X}_{lb}\right)$$

The F Value for Testing the Equality of Means of Groups a and b

$$F_{ab} = \frac{(n-q-g+1)n_a n_b}{q(n-g)(n_a+n_b)} D_{ab}^2$$

The Sum of Unexplained Variations (Dixon, 1973)

$$R = \sum_{a=1}^{g-1}\sum_{b=a+1}^{g} 4/\left(4+D_{ab}^2\right)$$

Classification Functions

Once a set of q variables has been selected, the classification functions (also known as Fisher’s linear discriminant functions) can be computed using

$$b_{ij} = (n-g)\sum_{l=1}^{q} w_{il}^*\overline{X}_{lj} \quad i=1,2,\ldots,q; j=1,2,\ldots,g$$

for the coefficients, and

$$a_j = \log p_j - \tfrac{1}{2}\sum_{i=1}^{q} b_{ij}\overline{X}_{ij} \quad j = 1, 2, \ldots, q$$

for the constant, where p_j is the prior probability of group *j*.

Canonical Discriminant Functions

The canonical discriminant function coefficients are determined by solving the general eigenvalue problem

$$(\mathbf{T} - \mathbf{W})\mathbf{V} = \lambda\mathbf{WV}$$

where **V** is the unscaled matrix of discriminant function coefficients and **λ** is a diagonal matrix of eigenvalues. The eigensystem is solved as follows:

The Cholesky decomposition

$$\mathbf{W} = \mathbf{LU}$$

is formed, where **L** is a lower triangular matrix, and $\mathbf{U} = \mathbf{L}'$.

The symmetric matrix $\mathbf{L}^{-1}\mathbf{B}\mathbf{U}^{-1}$ is formed and the system

$$\big(\mathbf{L}^{-1}(\mathbf{T} - \mathbf{W})\mathbf{U}^{-1} - \lambda\mathbf{I}\big)(\mathbf{UV}) = 0$$

is solved using tridiagonalization and the QL method. The result is *m* eigenvalues, where $m = \min(q, g-1)$ and corresponding orthonormal eigenvectors, **UV**. The eigenvectors of the original system are obtained as

$$\mathbf{V} = \mathbf{U}^{-1}(\mathbf{UV})$$

For each of the eigenvalues, which are ordered in descending magnitude, the following statistics are calculated.

Percentage of Between-Groups Variance Accounted for

$$\frac{100\lambda_k}{\sum_{k=1}^{m}\lambda_k}$$

Canonical Correlation

$$\sqrt{\lambda_k/(1+\lambda_k)}$$

Wilks' Lambda

Testing the significance of all the discriminating functions after the first *k*:

$$\Lambda_k = \prod_{i=k+1}^{m} 1/(1+\lambda_i) \quad k = 0, 1, \ldots, m-1$$

The significance level is based on

$$\chi^2 = -(n-(q+g)/2-1)\ln\Lambda_k$$

which is distributed as a χ^2 with $(q-k)(g-k-1)$ degrees of freedom.

The Standardized Canonical Discriminant Coefficient Matrix D

The standard canonical discriminant coefficient matrix **D** is computed as

$$\mathbf{D} = \mathbf{S}_{11}\mathbf{V}$$

where

$\mathbf{S}$=diag$\left(\sqrt{w_{11}}, \sqrt{w_{22}}, \ldots, \sqrt{w_{pp}}\right)$

$\mathbf{S}_{11}$= partition containing the first *q* rows and columns of **S**

V is a matrix of eigenvectors such that $\mathbf{V}'\mathbf{W}_{11}\mathbf{V}=I$

The Correlations Between the Canonical Discriminant Functions and the Discriminating Variables

The correlations between the canonical discriminant functions and the discriminating variables are given by

$$\mathbf{R} = \mathbf{S}_{11}^{-1}\mathbf{W}_{11}\mathbf{V}$$

If some variables were not selected for inclusion in the analysis ($q<p$), the eigenvectors are implicitly extended with zeroes to include the nonselected variables in the correlation matrix. Variables for which $W_{ii} = 0$ are excluded from **S** and **W** for this calculation; *p* then represents the number of variables with non-zero within-groups variance.

The Unstandardized Coefficients

The unstandardized coefficients are calculated from the standardized ones using

$$\mathbf{B} = \sqrt{(n-g)}\mathbf{S}_{11}^{-1}\mathbf{D}$$

The associated constants are:

$$a_k = -\sum_{i=1}^{q} b_{ik} \overline{X}_{i\bullet}$$

The group centroids are the canonical discriminant functions evaluated at the group means:

$$\overline{f}_{kj} = a_k + \sum_{i=1}^{q} b_{ik} \overline{X}_{ij}$$

Tests For Equality Of Variance

Box's *M* is used to test for equality of the group covariance matrices.

$$M = (n-g)\log\left|\mathbf{C}'\right| - \sum_{j=1}^{g} (n_j - 1)\log\left|\mathbf{C}^{(j)}\right|$$

where

$\mathbf{C}'$ = pooled within-groups covariance matrix excluding groups with singular covariance matrices

$\mathbf{C}^{(j)}$ = covariance matrix for group *j*.

Determinants of $\mathbf{C}'$ and $\mathbf{C}^{(j)}$ are obtained from the Cholesky decomposition. If any diagonal element of the decomposition is less than 10^{-11}, the matrix is considered singular and excluded from the analysis.

$$\log\left|\mathbf{C}^{(j)}\right| = 2\sum_{i=1}^{p} \log l_{ii} - p\log(n_j - 1)$$

where l_{ii} is the *i*th diagonal entry of **L** such that $(n_j - 1)\mathbf{C}^{(j)} = \mathbf{L}'\mathbf{L}$. Similarly,

$$\log\left|\mathbf{C}'\right| = 2\sum_{i=1}^{p} \log l_{ii} - p\log\left(n' - g\right)$$

where

$$\left(n' - g\right)\mathbf{C}' = \mathbf{L}'\mathbf{L}$$

n' = sum of weights of cases in all groups with nonsingular covariance matrices

The significance level is obtained from the *F* distribution with t_1 and t_2 degrees of freedom using (Cooley and Lohnes, 1971):

$$F = \begin{cases} M/b & \text{if } e_2 > e_1^2 \\ \frac{t_2 M}{t_1(b-M)} & \text{if } e_2 < e_1^2 \end{cases}$$

where

$$e_1 = \left(\sum_{j=1}^{g} \frac{1}{n_j - 1} - \frac{1}{n - g} \right) \frac{2p^2 + 3p - 1}{6(g-1)(p+1)}$$

$$e_2 = \left(\sum_{j=1}^{g} \frac{1}{(n_j - 1)^2} - \frac{1}{(n - g)^2} \right) \frac{(p-1)(p+2)}{6(g-1)}$$

$$t_1 = (g-1)p(p+1)/2$$

$$t_2 = (t_1 + 2)/\left|e_2 - e_1^2\right|$$

$$b = \begin{cases} \frac{t_1}{1 - e_1 - t_1/t_2} & \text{if } e_2 > e_1^2 \\ \frac{t_2}{1 - e_1 - 2/t_2} & \text{if } e_2 < e_1^2 \end{cases}$$

If $e_1^2 - e_2$ is zero, or much smaller than e_2, t_2 cannot be computed or cannot be computed accurately. If

$$e_2 = e_2 + 0.0001\left(e_2 - e_1^2\right)$$

the program uses Bartlett's χ^2 statistic rather than the *F* statistic:

$$\chi^2 = M(1 - e_1)$$

with t_1 degrees of freedom.

For testing the group covariance matrix of the canonical discriminant functions, the procedure is similar. The covariance matrices $\mathbf{C}'$ and $\mathbf{C}^{(j)}$ are replaced by $\mathbf{D}_j$ and $\mathbf{D}'$, where

$$\mathbf{D}_j = \mathbf{B}'\mathbf{C}^{(j)}\mathbf{B}$$

is the group covariance matrix of the discriminant functions.

The pooled covariance matrix in this case is an identity, so that

$$\mathbf{D}' = (n-g)\mathbf{I}_m - \sum_j (n_j - 1)\mathbf{D}_j$$

where the summation is only over groups with singular $\mathbf{D}_j$.

Classification

The basic procedure for classifying a case is as follows:

- If $\mathbf{X}$ is the $1 \times q$ vector of discriminating variables for the case, the $1 \times m$ vector of canonical discriminant function values is

  $$\mathbf{f} = \mathbf{XB} + \mathbf{a}$$

- A chi-square distance from each centroid is computed

  $$\chi_j^2 = \left(\mathbf{f} - \bar{\mathbf{f}}_j\right)\mathbf{D}_j^{-1}\left(\mathbf{f} - \bar{\mathbf{f}}_j\right)'$$

where $\mathbf{D}_j$ is the covariance matrix of canonical discriminant functions for group *j* and $\bar{\mathbf{f}}_j$ is the group centroid vector. If the case is a member of group *j*, χ_j^2 has a χ^2 distribution with *m* degrees of freedom. *P*(**X**|**G**), labeled as *P*(**D**>*d*|**G**=*g*) in the output, is the significance level of such a χ_j^2 .

- The classification, or posterior probability, is

$$P\left(\mathbf{G}_j|\mathbf{X}\right) = \frac{P_j|\mathbf{D}_j|^{-1/2}e^{-\chi_j^2/2}}{\sum_{j=1}^{g} P_j|\mathbf{D}_j|^{-1/2}e^{-\chi_j^2/2}}$$

where p_j is the prior probability for group *j*. A case is classified into the group for which $P\left(\mathbf{G}_j|\mathbf{X}\right)$ is highest.

The actual calculation of $P\left(\mathbf{G}_j|\mathbf{X}\right)$ is

$$g_j = \log P_j - \tfrac{1}{2}\left(\log|\mathbf{D}_j| + \chi_j^2\right)$$

$$P\left(\mathbf{G}_j|\mathbf{X}\right) = \begin{cases} \dfrac{\exp\left(g_j - \max_j g_j\right)}{\sum_{j=1}^{g} \exp\left(g_j - \max\limits_j g_j\right)} & \text{if } g_j - \max_j g_j > -46 \\ 0 & \text{otherwise} \end{cases}$$

If individual group covariances are not used in classification, the pooled within-groups covariance matrix of the discriminant functions (an identity matrix) is substituted for $\mathbf{D}_j$ in the above calculation, resulting in considerable simplification.

If any $\mathbf{D}_j$ is singular, a pseudo-inverse of the form

$$\begin{bmatrix} \mathbf{D}_{j11}^{-1} & 0 \\ 0 & 0 \end{bmatrix}$$

replaces $\mathbf{D}_j^{-1}$ and $|\mathbf{D}_{j11}|$ replaces $|\mathbf{D}_j|$. $\mathbf{D}_{j11}$ is a submatrix of $\mathbf{D}_j$ whose rows and columns correspond to functions not dependent on preceding functions. That is, function 1 will be excluded only if the rank of $\mathbf{D}_j = 0$, function 2 will be excluded only if it is dependent on function 1, and so on. This choice of the pseudo-inverse is not optimal for the numerical stability of $\mathbf{D}_{j11}^{-1}$, but maximizes the discrimination power of the remaining functions.

Cross-Validation

The following notation is used in this section:

Table 37-2
Notation

Notation	**Description**
$\underset{\sim}{X}_{jk}$	$(X_{1jk}, \ldots, X_{qjk})^T$
$\underset{\sim}{M}_j$	Sample mean of *j*th group $\underset{\sim}{M}_j = \frac{1}{n_j}\sum_{k=1}^{m_j} f_{jk}\underset{\sim}{X}_{jk}$

Notation	**Description**
$\underset{\sim}{M}_{jk}$	Sample mean of *j*th group excluding point $\underset{\sim}{X}_{jk}$ $$\underset{\sim}{M}_{jk} = \frac{1}{n_j - f_{jk}} \sum_{\substack{l=1 \\ l \neq k}}^{m_j} f_{jl} \underset{\sim}{X}_{jl}$$
Σ	Polled sample covariance matrix
Σ_j	Sample covariance matrix of *j*th group
Σ_{jk}	Polled sample covariance matrix without point $\underset{\sim}{X}_{jk}$
Σ_{jk}^{-1}	$$= \frac{n-g-f_{jk}}{n-g}\left(\Sigma^{-1} + \frac{n_j \Sigma_j^{-1}\left(\underset{\sim}{X}_{jk} - \underset{\sim}{M}_j\right)\left(\underset{\sim}{X}_{jk} - \underset{\sim}{M}_j\right)^T \Sigma_j^{-1}}{(n_j - f_{jk})(n_j - g) - n_j \left(\underset{\sim}{X}_{jk} - \underset{\sim}{M}_j\right)^T \Sigma_j^{-1} \left(\underset{\sim}{X}_{jk} - \underset{\sim}{M}_j\right)}\right)$$
$d_0^2\left(\underset{\sim}{a} \cdot \underset{\sim}{b}\right)$	$\left(\underset{\sim}{a} - \underset{\sim}{b}\right)^T \Sigma_{jk}^{-1} \left(\underset{\sim}{a} - \underset{\sim}{b}\right)^T$

Cross-validation applies only to linear discriminant analysis (not quadratic). During cross-validation, all cases in the dataset are looped over. Each case, say $\underset{\sim}{X}_{jk}$, is extracted once and treated as test data. The remaining cases are treated as a new dataset.

Here we compute $d_0^2\left(\underset{\sim}{X}_{jk}, \underset{\sim}{M}_{jk}\right)$ and $d_0^2\left(\underset{\sim}{X}_{jk}, \underset{\sim}{M}_i\right)(i = 1, ..., g.i \neq j)$. If there is an $i(i \neq j)$ that satisfies ($\log(P_i) - d_0^2\left(\underset{\sim}{X}_{jk}, \underset{\sim}{M}_i\right)/2 > \log(P_j) - d_0^2\left(\underset{\sim}{X}_{jk}, \underset{\sim}{M}_{jk}\right)/2$), then the extracted point $\underset{\sim}{X}_{jk}$ is misclassified. The estimate of prediction error rate is the ratio of the sum of misclassified case weights and the sum of all case weights.

To reduce computation time, the linear discriminant method is used instead of the canonical discriminant method. The theoretical solution is exactly the same for both methods.

Rotations

Varimax rotations may be performed on either the matrix of canonical discriminant function coefficients or on that of the correlation between the canonical discriminant functions and the discrimination variables (the structure matrix). The actual algorithm for the rotation is described in FACTOR. For the Kaiser normalization

$$h_i^2 = \begin{cases} 1 + 1/w_{ii}w_{ii}^{*} \\ \text{(squared multiple correlation)} & \text{if coefficients rotated} \\ \sum_{k=1}^{m} r_{ik}^2 & \text{if correlations rotated} \end{cases}$$

The unrotated structure matrix is

$$\mathbf{R} = \mathbf{S}_{11}^{-1}\mathbf{W}_{11}\mathbf{V}$$

If the rotation transformation matrix is represented by $\mathbf{K}$, the rotated standardized coefficient matrix $\mathbf{D}_R$ is given by

$$\mathbf{D}_R = \mathbf{DK}$$

The rotated matrix of pooled within-groups correlations between the canonical discriminant functions and the discriminating variables $\mathbf{R}_R$ is

$$\mathbf{R}_R = \mathbf{RK}$$

The eigenvector matrix $\mathbf{V}$ satisfies

$$\mathbf{V}^{'}(\mathbf{T}-\mathbf{W})\mathbf{V} = \Lambda = \text{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m)$$

where the λ_k are the eigenvalues. The equivalent matrix for the rotated coefficient $\mathbf{V}_R$

$$(\mathbf{V}_R)^{'}(\mathbf{T}-\mathbf{W})\mathbf{V}_R$$

is not diagonal, meaning the rotated functions, unlike the unrotated ones, are correlated for the original sample, although their within-groups covariance matrix is an identity. The diagonals of the above matrix may still be interpreted as the between-groups variances of the functions. They are the numerators for the proportions of variance printed with the transformation matrix. The denominator is their sum. After rotation, the columns of the transformation are exchanged, if necessary, so that the diagonals of the matrix above are in descending order.

References

Anderson, T. W. 1958. *Introduction to multivariate statistical analysis*. New York: John Wiley & Sons, Inc..

Cooley, W. W., and P. R. Lohnes. 1971. *Multivariate data analysis*. New York: John Wiley & Sons, Inc..

Dempster, A. P. 1969. *Elements of Continuous Multivariate Analysis*. Reading, MA: Addison-Wesley.

Dixon, W. J. 1973. *BMD Biomedical computer programs*. Los Angeles: University of California Press.

Tatsuoka, M. M. 1971. *Multivariate analysis*. New York: John Wiley & Sons, Inc. .

Ensembles Algorithms

Ensembles are used to enhance model accuracy (boosting), enhance model stability (bagging), and build models for very large datasets (pass, stream, merge).

- For more information, see the topic "Very large datasets (pass, stream, merge) algorithms".
- For more information, see the topic "Bagging and Boosting Algorithms".

Bagging and Boosting Algorithms

Bootstrap aggregating (Bagging) and boosting are algorithms used to improve model stability and accuracy. Bagging works well for unstable base models and can reduce variance in predictions. Boosting can be used with any type of model and can reduce variance and bias in predictions.

Notation

The following notation is used for bagging and boosting unless otherwise stated:

K	The number of distinct records in the training set.
X_k	Predictor values for the *k*th record.
y_k	Target value for the *k*th record.
f_k	Frequency weight for the *k*th record.
w_k	Analysis weight for the *k*th record.
N	The total number of records; $N = \Sigma_{k=1}^{K} f_k$.
M	The number of base models to build; for bagging, this is the number of bootstrap samples.
$T^m(\cdot)$	The model built on the *m*th bootstrap sample.
f_k^m	Simulated frequency weight for the *k*th record of the *m*th bootstrap sample.
w_k^m	Updated analysis weight for the *k*th record of the *m*th bootstrap sample.
$\hat{y}_k^m = T^m(X_k)$	Predicted target value of the *k*th record by the *m*th model.
$P_{l_i}^m(X_k)$	For a categorical target, the probability that the kth record belongs to category l_i, *i*=1, ..., *C*, in model *m*.
$II(\pi)$	For any condition π, $II(\pi)$ is 1 if π holds and 0 otherwise.

Bootstrap Aggregation

Bootstrap aggregation (bagging) produces replicates of the training dataset by sampling with replacement from the original dataset. This creates bootstrap samples of equal size to the original dataset. The algorithm is performed iteratively over *k*=1,..,*K* and *m*=1,...,*M* to generate frequency weights:

$$f^*_{mk} = \begin{cases} rv.binom\left(N, \frac{f_k}{N}\right) & k = 1 \\ rv.binom\left(N - \Sigma_{i=1}^{k-1} f^*_{mi}, \frac{f_k}{N - \Sigma_{i=1}^{k-1} f_i}\right) & \text{otherwise} \end{cases}$$

Then a model is built on each replicate. Together these models form an ensemble model. The ensemble model scores new records using one of the following methods; the available methods depend upon the measurement level of the target.

Scoring a Continuous Target

- Mean

  $$\hat{y}_k = \tfrac{1}{M}\sum_{m=1}^{M} \hat{y}_k^m$$

- Median

  Sort $\hat{y}_k^m$ and relabel them $\hat{y}_{(1)} \leq ... \leq \hat{y}_{(M)}$

  $$\hat{y}_k = \left\{ \begin{array}{cc} \hat{y}_{\left(\frac{M+1}{2}\right)} & \text{if } M \text{ is odd} \\ \frac{1}{2}\left(\hat{y}_{\left(\frac{M}{2}\right)} + \hat{y}_{\left(\frac{M}{2}+1\right)}\right) & \text{if } M \text{ is even} \end{array} \right\}$$

Scoring a Categorical Target

- Voting

  $$\hat{y}_k = arg\max\nolimits_{l_i \in \Omega} \tfrac{1}{|M_{l_i}|} \sum_{m \in M_{l_i}} P_{l_i}^m (X_k)$$

  $$\hat{p}_{\hat{y}_k} = \tfrac{1}{|M_{\hat{y}_k}|} \sum_{m \in M_{\hat{y}_k}} P_{\hat{y}_k}^m (X_k)$$

  where $\Omega = \{arg\max_{l_i} |M_{l_i}|\}$

- Highest probability

  $$\hat{y}_k = arg\max\nolimits_{l_i}\left(\max\nolimits_m \left(P_{l_i}^m (X_k)\right)\right)$$
  $$\hat{p}_{\hat{y}_k} = \max\nolimits_m \left(P_{\hat{y}_k}^m (X_k)\right)$$

- Highest mean probability

  $$\hat{y}_k = arg\max\nolimits_{l_i} \tfrac{1}{M}\sum_{m=1}^{M} P_{l_i}^m (X_k)$$

  $$\hat{p}_{\hat{y}_k} = \tfrac{1}{M}\sum_{m=1}^{M} P_{\hat{y}_k}^m (X_k)$$

Bagging Model Measures

Accuracy

Accuracy is computed for the naive model, reference (simple) model, ensemble model (associated with each ensemble method), and base models.

For categorical targets, the classification accuracy is

$$\frac{1}{N}\sum_{k=1}^{K} f_k II\left(y_k == \hat{y}_k\right)$$

For continuous targets, it is

$$R^2 = 1 - \frac{\Sigma_{k=1}^{K} f_k (y_k - \hat{y}_k)^2}{\Sigma_{k=1}^{K} f_k (y_k - \overline{y})^2}$$

where $\overline{y} = \frac{1}{N}\Sigma_{k=1}^{K} f_k y_k$

Note that R^2 can never be greater than one, but can be less than zero.

For the naïve model, $\hat{y}_k$ is the modal category for categorical targets and the mean for continuous targets.

Diversity

Diversity is a range measure between 0 and 1 in the larger-is-more-diverse form. It shows how much predictions vary across base models.

For categorical targets, diversity is

$$\frac{1}{N \cdot M^2}\sum_{k=1}^{K} f_k L\left(y_k\right)\left[M - L\left(y_k\right)\right]$$

where $L\left(y_k\right) = \sum_{m=1}^{M} II\left(y_k = \hat{y}_k^m\right)$.

For continuous targets, diversity is

$$\text{D} = \frac{\sum_{k=1}^{K} f_k \left[\frac{1}{M\left(M-1\right)}\sum_{m=1}^{M}\sum_{n=1, n\neq m}^{M}\left(y_k - \hat{y}_k^m\right)\left(\hat{y}_k^n - y_k\right)\right]}{\Sigma_{k=1}^{K} f_k (y_k - \overline{y}_k)^2}$$

Adaptive Boosting

Adaptive boosting (AdaBoost) is an algorithm used to boost models with continuous targets (Freund and Schapire 1996, Drucker 1997).

1. Initialize values.

   Set $w_k = \begin{cases} \frac{w_k}{\Sigma_{i=1}^K w_i f_i} & \text{if analysis weights specified} \\ 1/N & \text{otherwise} \end{cases}$

   Set *m*=1, $w_k^m = w_k$, and $f_k^m = f_k$. Note that analysis weights are initialized even if the method used to build base models does not support analysis weights.

2. Build base model *m*, $T^m(\cdot)$, using the training set and score the training set.

   Set the model weight for base model *m*, $\omega^m = \log\left(\frac{1-\sum_{k=1}^K L_k w_k^m f_k}{\sum_{k=1}^K L_k w_k^m f_k}\right)$

   where $L_k = \frac{abs(\hat{y}_k^m - y_k)}{\max_k\left(abs(\hat{y}_k^m - y_k)\right)}$.

3. Set weights for the next base model.

   $w_k^{m+1} = \frac{a_k^{m+1}}{\Sigma_{i=1}^K a_i^{m+1} f_i}$

   where $a_k^{m+1} = w_k^m \left(\frac{\sum_{k=1}^K L_k w_k^m f_k}{1-\sum_{k=1}^K L_k w_k^m f_k}\right)^{1-L_k}$. Note that analysis weights are always updated. If the method used to build base models does not support analysis weights, the frequency weights are updated for the next base model as follows:

   $$f_k^{m+1} = \begin{cases} rv.binom\left(N, w_k^{m+1} f_k\right) & k = 1 \\ rv.binom\left(N - \Sigma_{i=1}^{k-1} f_k^{m+1}, \frac{w_k^{m+1} f_k}{1-\Sigma_{i=1}^{k-1} w_k^{m+1} f_i}\right) & \text{otherwise} \end{cases}$$

   If *m*<*M*, set *m*=*m*+1 and go to step 2. Otherwise, the ensemble model is complete.

   Note: base models where $\sum_{k=1}^K L_k w_k^m f_k \geq 0.5$ or $\max_k\left(abs(\hat{y}_k^m - y_k)\right)$ are removed from the ensemble.

Scoring

AdaBoost uses the weighted median method to score the ensemble model.

Sort $\hat{y}_k^m$ and relabel them $\hat{y}_{(1)} \leq ... \leq \hat{y}_{(M)}$, retaining the association of the model weights, ω^m, and relabeling them $\omega_{(1)}, ..., \omega_{(M)}$

The ensemble predicted value is then $\hat{y}_k = \hat{y}_{(i)}$, where *i* is the value such that $\sum_{m=1}^{i-1} \omega^m < \frac{1}{2}\sum_{m=1}^{M} \omega^m \leq \sum_{m=1}^{i} \omega^m$

Stagewise Additive Modeling using Multiclass Exponential loss

Stagewise Additive Modeling using a Multiclass Exponential loss function (SAMME) is an algorithm that extends the original AdaBoost algorithm to categorical targets.

1. Initialize values.

   Set $w_k = \begin{cases} \frac{w_k}{\Sigma_{i=1}^{K} w_i f_i} & \text{if analysis weights specified} \\ 1/N & \text{otherwise} \end{cases}$

   Set *m*=1, $w_k^m = w_k$, and $f_k^m = f_k$. Note that analysis weights are initialized even if the method used to build base models does not support analysis weights.

2. Build base model *m*, $T^m(\cdot)$, using the training set and score the training set.

   Set the model weight for base model *m*, $\omega^m = \log \frac{1-err_m}{err^m} + \log(C-1)$

   where $err^m = \sum_{k=1}^{K} w_k^m f_k II(y_k \neq \hat{y}_k^m)$.

3. Set weights for the next base model.

   $$w_k^{m+1} = \frac{a_k^{m+1}}{\Sigma_{i=1}^{K} a_i^{m+1} f_i}$$

   where $a_k^{m+1} = w_k^m \exp(\omega^m II(y_k \neq \hat{y}_k^m))$. Note that analysis weights are always updated. If the method used to build base models does not support analysis weights, the frequency weights are updated for the next base model as follows:

   $$f_k^{m+1} = \begin{cases} rv.binom\left(N, w_k^{m+1} f_k\right) & k = 1 \\ rv.binom\left(N - \Sigma_{i=1}^{k-1} f_k^{m+1}, \frac{w_k^{m+1} f_k}{1 - \Sigma_{i=1}^{k-1} w_k^{m+1} f_i}\right) & \text{otherwise} \end{cases}$$

   If *m*<*M*, set *m*=*m*+1 and go to step 2. Otherwise, the ensemble model is complete.

   Note: base models where $err_m = 0$ or $\omega^m <= 0$ are removed from the ensemble.

Scoring

SAMME uses the weighted majority vote method to score the ensemble model.

The predicted value of the *k*th record for the *m*th base model is $\hat{y}_k^m = arg\max_{l_i} P_{l_i}^m(X_k)$.

The ensemble predicted value is then $\hat{y}_k = arg\max_{l_i} \sum_{m=1}^{M} \omega^m II(\hat{y}_k^m == l_i)$. Ties are resolved at random.

The ensemble predicted probability is $\hat{p}_{\hat{y}_k} = \sum_{m \in M_{\hat{y}_k}} \frac{\omega^m}{\sum_{i \in M_{\hat{y}_k}} \omega^i} P^m_{\hat{y}_k}(X_k)$

Boosting Model Measures

Accuracy

Accuracy is computed for the naive model, reference (simple) model, ensemble model (associated with each ensemble method), and base models.

For categorical targets, the classification accuracy is

$$\frac{1}{N}\sum_{k=1}^{K} f_k II\left(y_k == \hat{y}_k\right)$$

For continuous targets, it is

$$R^2 = 1 - \frac{\Sigma_{k=1}^{K} f_k(y_k - \hat{y}_k)^2}{\Sigma_{k=1}^{K} f_k(y_k - \overline{y})^2}$$

where $\overline{y} = \frac{1}{N}\Sigma_{k=1}^{K} f_k y_k$

Note that R^2 can never be greater than one, but can be less than zero.

For the naïve model, $\hat{y}_k$ is the modal category for categorical targets and the mean for continuous targets.

References

Drucker, H. 1997. Improving regressor using boosting techniques. In: *Proceedings of the 14th International Conferences on Machine Learning* , D. H. Fisher,Jr., ed. San Mateo, CA: Morgan Kaufmann, 107–115.

Freund, Y., and R. E. Schapire. 1995. A decision theoretic generalization of on-line learning and an application to boosting. In: *Computational Learning Theory: 7 Second European Conference, EuroCOLT '95,* , 23–37.

Very large datasets (pass, stream, merge) algorithms

We implement the PSM features PASS, STREAM, and MERGE through ensemble modeling. PASS builds models on very large data sets with only one data pass; STREAM updates the existing model with new cases without the need to store or recall the old training data; MERGE builds models in a distributed environment and merges the built models into one model.

In an ensemble model, the training set will be divided into subsets called blocks, and a model will be built on each block. Because the blocks may be dispatched to different threads (here one process contains one thread) and even different machines, models in different processes can be built at the same time. As new data blocks arrive, the algorithm simply repeats this procedure. Therefore it can easily handle the data stream and perform incremental learning for ensemble modeling.

Pass

The PASS operation includes following steps:

1. Split the data into training blocks, a testing set and a holdout set. Note that the frequency weight, if specified, is ignored when splitting the training set into blocks (to prevent blocks from being entirely represented by a single case) but is accounted for when creating the testing and holdout sets.

2. Build base models on training blocks and build a reference model on the testing set. A single model is built on the testing set and each training block.

3. Evaluate each base model by computing the accuracy based on the testing set. Select a subset of base models as ensemble elements according to accuracy.

4. Evaluate the ensemble model and the reference model by computing the accuracy based on the holdout set. If the ensemble model's performance is not better than the reference model's performance on the holdout set, we use the reference model to score the new cases.

Computing Model Accuracy

The accuracy of a base model is assessed on the testing set. For each vector of predictors x_i and the corresponding label c_i observed in the testing set *T*, let $\hat{c}(x_i)$ be the label predicted by the given model. Then the testing error is estimated as:

Categorical target.

$$E = \frac{1}{\sum_{i=1}^{|T|} f_i} \sum_{i=1}^{|T|} \left(f_i \cdot I\left(c_i \neq \hat{c}(x_i)\right)\right)$$

Continuous target.

$$E = \frac{1}{\sum_{i=1}^{|T|} f_i} \sum_{i=1}^{|T|} \left(f_i \cdot |y_i - \hat{y}_i| \right)$$

Where $I\left(c_i \neq \hat{c}(x_i)\right)$ is 1 if $c_i \neq \hat{c}(x_i)$ and 0 otherwise.

The accuracy for the given model is computed by $A=1-E$. The accuracy for the whole ensemble model and the reference model is assessed on the holdout set.

Stream

When new cases arrive and the user wants to update the existing ensemble model with these cases, the algorithm will:

1. Start a PASS operation to build an ensemble model on the new data, then
2. MERGE the newly created ensemble model and the existing ensemble model.

Merge

The MERGE operation has the following steps:

1. Merge the holdout sets into a single holdout set and, if necessary, reduce this set to a reasonable size.
2. Merge the testing sets into a single testing set and, if necessary, reduce this set to a reasonable size.
3. Build a merged reference model on the merged testing set.
4. Evaluate every base model by computing the accuracy based on the merged testing set. Select a subset of base models as elements of the merged ensemble model according to accuracy.
5. Evaluate the merged ensemble model and the merged reference model by computing the accuracy based on the merged holdout set.

Adaptive Predictor Selection

There are two methods, depending upon whether the method used to build base models has an internal predictor selection algorithm.

Method has predictor selection algorithm

The first base model is built with all predictors available to the method's predictor selection algorithm. Base model *j* (*j* > 1) makes the *i*th predictor available with probability

$$p_i = \max\left(\frac{n'_i + C}{n_i + C}, \beta\right)$$

where n'_i is the number of times the *i*th predictor was selected by the method's predictor selection algorithm in the previous *j*−1 base models, n_i is the number of times the *i*th predictor was made available to the method's predictor selection algorithm in the previous *j*−1 base models, *C* is a constant to smooth the value of p_i, and β is a lower limit on p_i.

Method does not have predictor selection algorithm

Each base model makes the *i*th predictor available with probability

$$p_i = \begin{cases} (1-\rho_i)^2 & \text{if } \rho_i < 0.05 \\ \beta & \text{otherwise} \end{cases}$$

where ρ_i is the *p*-value of a test for the *i*th predictor, as defined below.

- For a categorical target and categorical predictor, ρ is a chi-square test of $G^2 = 2\sum_{i=1}^{I}\sum_{j=1}^{J} G_{ij}^2$ where $G_{ij}^2 = \begin{cases} N_{ij}\ln\left(N_{ij}/\hat{N}_{ij}\right) & N_{ij} > 0 \\ 0 & \text{else} \end{cases}$ and with degrees freedom $(I-1)(J-1)$. N_{ij} is the number of cases with *X*=*i* and *Y*=*j*, $N_{i\cdot} = \sum_{j=1}^{J} N_{ij}$,$N_{\cdot j} = \sum_{i=1}^{I} N_{ij}$, and $\hat{N}_{ij} = N_{i\cdot}N_{\cdot j}/N$.
- For a categorical target and continuous predictor, ρ is an *F* test of $F = \frac{\sum_{j=1}^{J} N_j\left(\overline{x}_j - \overline{\overline{x}}\right)^2/(J-1)}{\sum_{j=1}^{J}(N_j-1)s_j/(N-J)}$ with degrees of freedom $J-1, N-J$. N_j is the number of cases with *Y*=*j*, $\overline{x}_j$ and s_j^2 are the sample mean and sample variance of *X* given *Y*=*j*, and $\overline{\overline{x}} = \sum_{j=1}^{J} N_j\overline{x}_j/N$
- For a continuous target and categorical predictor, ρ is an *F* test of $F = \frac{\sum_{i=1}^{I} N_i\left(\overline{y}_i - \overline{\overline{y}}\right)^2/(I-1)}{\sum_{i=1}^{I}(N_i-1)s(y)_i/(N-I)}$ with degrees of freedom $I-1, N-I$. N_i is the number of cases with *X*=*i*, $\overline{y}_i$ and $s(y)_i^2$ are the sample mean and sample variance of *Y* given *X*=*i*, and $\overline{\overline{y}} = \Sigma_{i=1}^{I} N_i\overline{y}_i/N$.
- For a continuous target and continuous predictor, ρ is a two-sided *t* test of $t = r\sqrt{\frac{N-2}{1-r^2}}$ where $r = \frac{\Sigma_{i=1}^{N}(x_i-\overline{x})(y_i-\overline{y})/(N-1)}{\sqrt{s(x)^2 s(y)^2}}$ and with degrees of freedom $N-2$. $s(x)^2$ is the sample variance of *X* and $s(y)^2$ is the sample variance of *Y*.

Automatic Category Balancing

When a target category occurs relatively infrequently, many models do a poor job of predicting members of that rarely occurring category, even if the overall prediction rate of the model is fairly good. Automatic category balancing should improves the model’s accuracy when predicting infrequently occurring values.

As records arrive, they are added to a training block until it is full. Then the proportion of records in each category is computed: $C_i = \frac{w_i}{w}$ where w_i is the weighted number of records taking category *i* and *w* is the total weighted number of records.

- If there is any category such that $C_i < \alpha/(10\cdot|C|)$, where $|C|$ is the number of target categories and α= 0.3, then randomly remove each record from the training block with probability

  $$Min\left\{(1-Min(C)/C_i), \left(1-\frac{\alpha}{|C|}\right)\right\}$$

  This operation will tend to remove records from frequently-occurring categories. Add new records to the training block until it is full again, and repeat this step until the condition is not satisfied.
- If there is any category such that $C_i < \alpha/|C|$, then recompute the frequency weight for record *k* as $f_k = f_k \max\left(10, \alpha\max(C)/C_{i(k)}\right)$, where $i(k)$ is the category of the *k*th record. This operation gives greater weight to infrequently occurring categories.

Model Measures

The following notation applies.

N	Total number of records
M	Total number of base models
f_k	The frequency weight of record *k*
y_k	The observed target value of record *k*
$\hat{y}_k$	The predicted target value of record *k* by the ensemble model
$\hat{y}_k^m$	The predicted target value of record *k* by base model *m*

Accuracy

Accuracy is computed for the naive model, reference (simple) model, ensemble model (associated with each ensemble method), and base models.

For categorical targets, the classification accuracy is

$$\frac{1}{N}\sum_{k=1}^{K} f_k II\left(y_k == \hat{y}_k\right)$$

where

$$II\left(y_k = \hat{y}_k\right) = \begin{cases} 1, \text{if } \left(y_k = \hat{y}_k\right) \\ 0, \text{otherwise} \end{cases}$$

For continuous targets, it is

$$R^2 = 1 - \frac{\Sigma_{k=1}^{K} f_k\left(y_k - \hat{y}_k\right)^2}{\Sigma_{k=1}^{K} f_k\left(y_k - \overline{y}\right)^2}$$

where $\overline{y} = \frac{1}{N}\Sigma_{k=1}^{K} f_k y_k$

Note that R^2 can never be greater than one, but can be less than zero.

For the naïve model, $\hat{y}_k$ is the modal category for categorical targets and the mean for continuous targets.

Diversity

Diversity is a range measure between 0 and 1 in the larger-is-more-diverse form. It shows how much predictions vary across base models.

For categorical targets, diversity is

$$\frac{1}{N \cdot M^2} \sum_{k=1}^{K} f_k L(y_k)\left[M - L(y_k)\right]$$

where $L(y_k) = \sum_{m=1}^{M} II(y_k = \hat{y}_k^m)$ and $II(y_k = \hat{y}_k^m)$ is defined as above.

Diversity is not available for continuous targets.

Scoring

There are several strategies for scoring using the ensemble models.

Continuous Target

Mean. $\hat{y}_{i,PSM} = \frac{1}{M} \sum_{m=1}^{M} \hat{y}_{i,m}$

Median. $\hat{y}_{i,PSM} = Median_1^M(\hat{y}_{i,m})$

where $\hat{y}_{i,PSM}$ is the final predicted value of case *i*, and $\hat{y}_{i,m}$ is the *m*th base model's predicted value of case *i*.

Categorical Target

Voting. Assume that $d_{m,k}$ represents the label output of the *m*th base model for a given vector of predictor values. $d_{m,k}$ = 1if the label assigned by the *m*th base model is the *k*th target category and 0 otherwise. There are total of *M* base models and *K* target categories. The majority vote method selects the *j*th category if it is assigned by the plurality of base models. It satisfies the ollowing equation:

$$\sum_{m=1}^{M} d_{m,j} = \max_{k=1}^{K}\left(\sum_{m=1}^{M} d_{m,k}\right)$$

Let E_m be the testing error estimated for the *m*th base model. Weights for the weighted majority vote are then computed according to the following expression:

$$w_m = \max\left(\log\frac{1-E_m}{E_m}, 0\right) / \sum_{i=1}^{M} \max\left(\log\frac{1-E_i}{E_i}, 0\right)$$

Probability voting. Assume that $p_{m,k}$ is the posterior probability estimated for the kth target category by the *m*th base model for a given vector of predictor values. The following rules combine the probabilities computed by the base models. The jth category is selected such that it satisfies the corresponding equation.

- Highest probability. $\max_{m=1}^{M}(p_{m,j}) = \max_{k=1}^{K}(\max_{m=1}^{M}(p_{m,k}))$
- Highest mean probability. $\frac{1}{M}\sum_{m=1}^{M} p_{m,j} = \max_{k=1}^{K}\left(\frac{1}{M}\sum_{m=1}^{M} p_{m,k}\right)$

Ties are resolved at random.

Softmax smoothing. The softmax function can be used for smoothing the probabilities:

$$p_i^S = \frac{Exp\left(p_i\right)}{\sum_{i=1}^{K} Exp\left(p_i\right)}$$

where p_i is the rule-based confidence for category *i* and p_i^S is the smoothed value.

ERROR BARS Algorithms

This section describes the algorithms for error bar computation of the mean, median and their confidence intervals for a simple random sample.

Notation

The following notation is used throughout this section unless otherwise noted:

Let $y_1 \leq ... \leq y_m$ be *m* ordered observations for the sample and $w_1, ..., w_m$ be the corresponding case weights. Then

$ww_i = \sum_{k=1}^{i} w_k =$ cumulative sum of weights up to and including y_i

and

$W = ww_m = \sum_{k=1}^{m} w_k =$ total sum of weights

$p = \frac{CI}{100}$,*CI* is the confidence interval level $0 \leq CI < 100$.

Descriptive Statistics

The following statistics are available.

Mean

$$\overline{y} = \frac{\sum_{i=1}^{m} w_i y_i}{W}$$

Confidence Interval for the Mean

Lower bound = $\overline{y} - IDF.T\left(\frac{p+1}{2}, W-1\right) \quad SE$

Upper bound = $\overline{y} + IDF.T\left(\frac{p+1}{2}, W-1\right) \quad SE$

where SE is the standard error, and IDF.T is the inverse student t function documented in the COMPUTE command.

Variance

$$s^2 = \frac{1}{W-1}\sum_{i=1}^{m} w_i(y_i - \overline{y})^2$$

Standard Deviation

$$s = \sqrt{s^2}$$

Standard Error

$$SE = \frac{s}{\sqrt{W}}$$

Median

The Aempirical method in the EXAMINE procedure is used for computation of the median.

Let

$$v = \frac{W}{2}$$

and *k* satifies

$$ww_k \leq v < ww_{k+1}$$

Then,

$$g = v - ww_k$$

Let *m* be the estimated median, then it is defined as

$$m = \begin{cases} \frac{(y_k + y_{k+1})}{2}, & g = 0 \\ y_{k+1}, & g > 0 \end{cases}$$

Confidence Interval for the Median

Note: the case weights $w_1, \cdots, w_m$ must be integers for the following computation. If at least one weight is not integer, an error message is issued.

Let

$$\begin{aligned} b_i &= \Pr\left[Binomial\left(W, 0.5\right) \geq i\right] \\ &= \sum_{j=i}^{W} \binom{W}{i} 0.5^W \\ &= IB\left(0.5; i, W - i\right) \end{aligned}$$

where IB is the incomplete Beta function.

Define

$$\begin{aligned} \gamma_i &= \Pr\left[i \leq Binomial\left(W, 0.5\right) \leq W - i\right] \\ &= b_i - b_{W-i+1} \end{aligned}, \ i = 0, 1, ..., floor\left(W/2\right)$$

and define

$\gamma_{w/2+1} = 0,$ if *W* is even;

$\gamma_{(w+1)/2} = 0$, if *W* is odd.

Algorithm: Hettmansperger-Sheather Interpolation (1986)

1. Re-index all the cases to be $x_1 \leq x_2, ..., \leq x_W$ in which

$$\begin{array}{llll} x_1 & = x_2, \ldots & = x_{ww_1} & = y1 \\ x_{ww_1+1} & = x_{ww_1+2} \ldots & = x_{ww_2} & = y_2 \\ . \\ . \\ x_{ww_{m-1}+1} & = x_{ww_{m-1}+2} \ldots & = x_{ww_m} & = y_m \end{array}$$

2. If *W* is even, compute $\gamma_0, \ldots, \gamma_{W/2}$.
   If *W* is odd, compute $\gamma_0, \ldots, \gamma_{(W+1)/2}$.

3. Choose the smallest index *k* such that $\gamma_{k+1} \leq p$. If *k* is found, go to Step 4; otherwise, stop and issue a message.

4. Compute

   $l = \frac{\gamma_k - p}{\gamma_k - \gamma_{k+1}}$,

   and

   $\lambda = \frac{(W-k)l}{k+(W-2k)l}$.

   The *p* confidence interval is

   Lower bound = $\lambda \cdot x_{k+1} + (1-\lambda) \cdot x_k$

   Upper bound = $\lambda \cdot x_{W-k} + (1-\lambda) \cdot x_{W-k+1}$

References

Hettmansperger, T. P., and S. J. Sheather. 1986. Confidence Interval Based on Interpolated Order Statistics. *Statistical Probability Letters*, 4, 75–79.

EXAMINE Algorithms

EXAMINE provides stem-and-leaf plots, histograms, boxplots, normal plots, robust estimates of location, tests of normality, and other descriptive statistics. Separate analyses can be obtained for subgroups of cases.

Univariate Statistics

This section discusses the computation of statistics for a variable considered on its own.

Notation

The following notation is used throughout this chapter unless otherwise noted:

Let $y_1 < \ldots y_m$ be *m* distinct ordered observations for the sample and $c_1, \ldots, c_m$ be the corresponding caseweights. Then

$$cc_i = \sum_{k=1}^{i} c_k = \text{cumulative frequency up to and including } y_i$$

and

$$W = cc_m = \sum_{k=1}^{m} c_k = \text{total sum of weights.}$$

Descriptive Statistics

The following statistics are available.

Minimum and Maximum

$$\min = y_1, \quad \max = y_m$$

Range

$$\text{range} = y_m - y_1$$

Mean

$$\bar{y} = \frac{\sum_{i=1}^{m} c_i y_i}{W}$$

Confidence Interval for the Mean

$$\text{lower bound} = \overline{y} - t_{\alpha/2,W-1}\text{SE}$$
$$\text{upper bound} = \overline{y} + t_{\alpha/2,W-1}\text{SE}$$

where SE is the standard error.

Median

The median is the 50th percentile, which is calculated by the method requested. The default method is HAVERAGE.

Interquartile Range

(IQR) IQR = 75th percentile − 25th percentile, where the 75th and 25th percentiles are calculated by the method requested for percentiles.

Variance

$$s^2 = \frac{1}{W-1}\sum_{i=1}^{m} c_i(y_i - \overline{y})^2$$

Standard Deviation

$$s = \sqrt{s^2}$$

Standard Error

$$SE = \frac{s}{\sqrt{W}}$$

Skewness and SE of Skewness

$$g_1 = \frac{WM_3}{(W-1)(W-2)s^3}$$

$$SE(g_1) = \sqrt{\frac{6W(W-1)}{(W-2)(W+1)(W+3)}}$$

$$M_3 = \sum_{i=1}^{m} c_i(y_i - \overline{y})^3$$

Kurtosis and SE of Kurtosis

$$g_2 = \frac{W(W+1)M_4 - 3M_2^2(W-1)}{(W-1)(W-2)(W-3)s^4}$$

$$M_2 = \sum_{i=1}^{m} c_i (y_i - \overline{y})^2$$

$$M_4 = \sum_{i=1}^{m} c_i (y_i - \overline{y})^4$$

$$SE(g_2) = \sqrt{\frac{4(W^2-1)SE^2(g_1)}{(W-3)(W+5)}}$$

5% Trimmed Mean

$$T_{0.9} = \frac{1}{0.9W}\left\{ (cc_{k_1+1} - tc)y_{k_1+1} + (W - cc_{k_2-1} - tc)y_{k_2} + \sum_{i=k_1+2}^{k_2-1} c_i y_i \right\}$$

where k_1 and k_2 satisfy the following conditions

$$cc_{k_1} < tc \le cc_{k_1+1}, \quad W - cc_{k_2} < tc \le W - cc_{k_2-1}$$

and

$$tc = 0.05W$$

Note: If $k_1 + 1 = k_2$, then $T_{0.9} = Yk_2$

Percentiles

There are five methods for computation of percentiles. Let

$$tc_1 = Wp, \quad tc_2 = (W+1)p$$

where *p* is the requested percentile divided by 100, and k_1 and k_2 satisfy

$$cc_{k_1} \le tc_1 < cc_{k_1+1}$$
$$cc_{k_2} \le tc_2 < cc_{k_2+1}$$

Then,

$$g_1 = \frac{(tc_1 - cc_{k_1})}{c_{k_1+1}}, \quad g_1^* = tc_1 - cc_{k_1}$$

$$g_2 = \frac{(tc_2 - cc_{k_2})}{c_{k_2+1}}, \quad g_2^* = tc_2 - cc_{k_2}$$

Let *x* be the *p*th percentile; the five definitions are as follows:

Waverage (Weighted Average)

Round (Closest Observation)

Empirical (Empirical Distribution Function)

Haverage (Weighted Average)

Aempirical (Empirical Distribution Function with Averaging)

Waverage (Weighted Average)

This is a weighted average at y_{tc_1}.

$$x = \begin{cases} y_{k_1+1} & \text{if } g_1^* \geq 1 \\ (1 - g_1^*)y_{k_1} + g_1^* y_{k_1+1} & \text{if } g_1^* < 1 \text{ and } c_{k_1+1} \geq 1 \\ (1 - g_1)y_{k_1} + g_1 y_{k_1+1} & \text{if } g_1^* < 1 \text{ and } c_{k_1+1} < 1 \end{cases}$$

Round (Closest Observation)

This is the observation closest to y_{tc_1}.

If $c_{k_1+1} \geq 1$, then

$$x = \begin{cases} y_{k_1} & \text{if } g_1^* < \frac{1}{2} \\ y_{k_1+1} & \text{if } g_1^* \geq \frac{1}{2} \end{cases}$$

If $c_{k_1+1} < 1$, then

$$x = \begin{cases} y_{k_1} & \text{if } g_1 < \frac{1}{2} \\ y_{k_1+1} & \text{if } g_1 \geq \frac{1}{2} \end{cases}$$

Empirical (Empirical Distribution Function)

$$x = \begin{cases} y_{k_1} & \text{if } g_1^* = 0 \\ y_{k_1+1} & \text{if } g_1^* > 0 \end{cases}$$

Haverage (Weighted Average)

This is a weighted average at y_{tc_2}.

$$x = \begin{cases} y_{k_2+1} & \text{if } g_2^* \geq 1 \\ (1-g_2^*)y_{k_2} + g_2^* y_{k_2+1} & \text{if } g_2^* < 1 \text{ and } c_{k_2+1} \geq 1 \\ (1-g_2)y_{k_2} + g_2 y_{k_2+1} & \text{if } g_2^* < 1 \text{ and } c_{k_2+1} < 1 \end{cases}$$

Aempirical (Empirical Distribution Function with Averaging)

$$x = \begin{cases} (y_{k_1} + y_{k_1+1})/2 & \text{if } g_1^* = 0 \\ y_{k_1+1} & \text{if } g_1^* > 0 \end{cases}$$

Note: If either the 25th, 50th, or 75th percentiles is request, Tukey Hinges will also be printed.

Tukey Hinges

Let Q_1, Q_2, and Q_3 be the 25th, 50th, and 75th percentiles. If $c^* \geq 1$, where $c^* = \min(c_1, \ldots, c_m)$, define

$$d = \frac{\text{greatest integer} < ((W+3)/2)}{2}$$
$$L_1 = d$$
$$L_2 = W/2 + 1/2$$
$$L_3 = W + 1 - d$$

Otherwise

$$d = \frac{\text{greatest integer} \leq (W/c^*+3)/2}{2}$$

and

$$L_1 = dc^*$$
$$L_2 = W/2 + c^*/2$$
$$L_3 = W + c^* - dc^*$$

Then for every *i*, $i = 1, 2, 3$, find h_i such that

$$cc_{h_i} \leq L_i < cc_{h_i+1}$$

and

$$Q_i = \begin{cases} (1-a_i^*)y_{h_i} + a_i^* y_{h_i+1} & \text{if } a_i^* < 1 \text{ and } c_{h_i+1} \geq 1 \\ (1-a_i)y_{h_i} + a_i y_{h_i+1} & \text{if } a_i^* < 1 \text{ and } c_{h_i+1} < 1 \\ y_{h_i+1} & \text{if } a_i^* \geq 1 \end{cases}$$

where

$$a_i^* = L_i - cc_{h_i}$$
$$a_i = \frac{a_i^*}{c_{h_i+1}}$$

M-Estimation (Robust Location Estimation)

The M-estimator *T* of location is the solution of

$$\sum_{i=1}^{m} c_i \Psi\left(\frac{y_i - T}{s}\right) = 0$$

where Ψ is an odd function and *s* is a measure of the spread.

An alternative form of M-estimation is

$$\sum_{i=1}^{m} c_i \left(\frac{y_i - T}{s}\right) \omega \left(\frac{y_i - T}{s}\right) = 0$$

where

$$\omega(u) = \frac{\Psi(u)}{u}$$

After rearranging the above equation, we get

$$T = \frac{\sum_{i=1}^{m} c_i y_i \omega\left(\frac{y_i - T}{s}\right)}{\sum_{i=1}^{m} c_i \omega\left(\frac{y_i - T}{s}\right)}$$

Therefore, the algorithm to find M-estimators is defined iteratively by

$$T_{k+1} = \frac{\sum_{i=1}^{m} c_i y_i \omega\left(\frac{y_i - T_k}{s}\right)}{\sum_{i=1}^{m} c_i \omega\left(\frac{y_i - T_k}{s}\right)}$$

The algorithm stops when either

$|T_{k+1} - T_k| \leq \epsilon[(T_{k+1} + T_k)/2]$, where $\epsilon = 0.005$

or the number of iterations exceeds 30.

M-Estimators

Four M-estimators (Huber, Hampel, Andrew, and Tukey) are available. Let

$$u_i = \frac{y_i - T}{s}$$

where

$s =$ median of $\tilde{y}_1, \ldots, \tilde{y}_m$ with caseweights $c_1, \ldots, c_m$

and

$\tilde{y}_i = |y_i - \tilde{y}|$, where $\tilde{y}$ is the median.

Huber (k), k > 0

$$\omega(u_i) = \begin{cases} 1 & \text{if } |u_i| \le k \\ \frac{k}{u_i} sgn(u_i) & \text{if } |u_i| > k \end{cases}$$

The default value of $k = 1.339$

Hampel (a, b, c), 0 < a ≤ b ≤ c

$$\omega(u_i) = \begin{cases} 1 & \text{if } |u_i| \le a \\ \frac{a}{u_i} sgn(u_i) & \text{if } a < |u_i| \le b \\ \frac{a}{u_i} \frac{c - |u_i|}{c - b} sgn(u_i) & \text{if } b < |u_i| \le c \\ 0 & \text{if } |u_i| > c \end{cases}$$

By default, *a* = 1.7, *b* = 3.4 and *c* = 8.5.

Andrew's Wave (c), c > 0

$$\omega(u_i) = \begin{cases} \frac{c}{\pi u_i} \sin\left(\frac{\pi u_i}{c}\right) & \text{if } |u_i| \le c \\ 0 & \text{if } |u_i| > c \end{cases}$$

By default, $c = 1.34\pi$

Tukey's Biweight (c)

$$\omega(u_i) = \begin{cases} \left(1 - \frac{u_i^2}{c^2}\right)^2 & \text{if } |u_i| \le c \\ 0 & \text{if } |u_i| > c \end{cases}$$

By default, c = 4.685.

Tests of Normality

The following tests are available.

Shapiro-Wilk Statistic (W)

Since the *W* statistic is based on the order statistics of the sample, the caseweights have to be restricted to integers. Hence, before *W* is calculated, all the caseweights are rounded to the closest integer and the series is expanded. Let c_i^* be the closest integer to c_i; then

$$cc_i^* = \sum_{k=1}^{i} c_k^*, \quad W_s = cc_m^* = \sum_{k=1}^{m} c_k^*$$

The original series $y = \{y_1, \ldots, y_m\}$ is expanded to

$$x = \{x_1, \ldots, x_{W_s}\}$$

where

$$x_{cc_{i-1}^*+1} = \ldots = x_{cc_i^*} = y_i, \quad i = 1, \ldots, m$$

Then the *W* statistic is defined as

$$W = \frac{\left(\sum_{i=1}^{W_s} a_i x_i\right)^2}{\sum_{i=1}^{W_s} (x_i - \overline{x})^2}$$

where

$$\overline{x} = \frac{\sum_{i=1}^{W_s} x_i}{W_s}$$

$$a_1^2 = a_{W_s}^2 = \begin{cases} \frac{\Gamma(W_s/2)}{\sqrt{2}\Gamma((W_s+1)/2)} & \text{if } 5 \leq W_s \leq 20 \\ \frac{\Gamma((W_s+1)/2)}{\sqrt{2}\Gamma(W_s/2+1)} & \text{if } W_s > 20 \end{cases}$$

$$a_1 = -\sqrt{a_1^2}, a_{W_s} = \sqrt{a_{W_s}^2}$$

$$a_i = (2/c)m_i, i = 2, \ldots, W_s - 1$$

$$m_i = \Psi^{-1}\left(\frac{i-\alpha}{W_s - 2\alpha + 1}\right)$$, where Ψ is the c.d.f. of a standard normal distribution

$$\alpha = 0.314195 + 0.063336\beta - 0.010895\beta^2$$

$$\beta = \log_{10} W_s$$

$$c^2 = 4\sum_{i=1}^{W_s-1} \frac{m_i^2}{(1-2a_i^2)}$$

Based on the computed *W* statistic, the significance is calculated by linearly interpolating within the range of simulated critical values given in Shapiro and Wilk (1965).

If non-integer weights are specified, the Shapiro-Wilk's statistic is calculated when the weighted sample size lies between 3 and 50. For no weights or integer weights, the statistic is calculated when the weighted sample size lies between 3 and 5000.

If $W > w_{0.99}$, the critical value of 99th percentile, the significance is reported as>0.99. Similarly, if $W < w_{0.01}$, the critical value of first percentile, the significance is reported as <0.01.

Kolmogorov-Smirnov Statistic with Lilliefors' Significance

Lilliefors (Lilliefors, 1967) presented a table for testing normality using the Kolmogorov-Smirnov statistic when the mean and variance of the population are unknown. This statistic is

$$D_a = \max\{D_+, D_-\}$$

where

$$D_+ = \max_i\left\{\hat{F}(y_i) - F(y_i)\right\}$$
$$D_- = \max_i\left\{F(y_i) - \hat{F}(y_{i-1})\right\}$$

where $\hat{F}(x)$ is the sample cumulative distribution and $F(x)$ is the cumulative normal distribution whose mean and variance are estimated from the sample.

Dallal and Wilkinson (Dallal and Wilkinson, 1986) corrected the critical values for testing normality reported by Lilliefors. With the corrected table they derived an analytic approximation to the upper tail probabilities of D_a for probabilities less than 0.1. The following formula is used to estimate the critical value D_c for probability 0.1.

$$D_c = \frac{\left(-b - \sqrt{b^2 - 4ac}\right)}{2a}$$

where, if $W \leq 100$,

$$a = -7.01256(W + 2.78019)$$
$$b = 2.99587\sqrt{W + 2.78019}$$
$$c = 2.1804661 + \frac{0.974598}{\sqrt{W}} + \frac{1.67997}{W}$$

If W > 100

$$a = -7.90289126054 * W^{0.98}$$
$$b = 3.180370175721 * W^{0.49}$$
$$c = 2.2947256$$

The Lilliefors significance *p* is calculated as follows: If $D_a = D_c, p = .0.1$

If $D_a > D_c, p = \exp\left\{aD_a^2 + bD_a + c - 2.3025851\right\}$.

If $D_{0.2} \leq D_a < D_c$, linear interpolation between $D_{0.2}$ and D_c where $D_{0.2}$ is the critical value for probability 0.2 is done.

If $D_a > D_{0.2}, p$ is reported as > 0.2.

Group Statistics

Assume that there are $k(k \geq 2)$ combinations of grouping factors. For every combination *i*, $i = 1, 2, \ldots, k$, let $\{y_{i1}, \ldots, y_{im_i}\}$ be the sample observations with the corresponding caseweights $\{c_{i1}, \ldots, c_{im_i}\}$.

Spread versus Level

If a transformation value, *a*, is given, the spread(*s*) and level(*l*) are defined based on the transformed data. Let *x* be the transformed value of *y*; for every $i = 1, \ldots, k, j = 1, \ldots, m_i$

$$x_{ij} = \begin{cases} \ln y_{ij} & \text{if } a = 0 \\ y_{ij}^a & \text{otherwise} \end{cases}$$

Then the spread (s_i) and the level (l_i) are respectively defined as the Interquartile Range and the median of $\{x_{i1}, \ldots, x_{im_i}\}$ with corresponding caseweights $\{c_{i1}, \ldots, c_{im_i}\}$. However, if *a* is not specified, the spread and the level are natural logarithms of the Interquartile Range and of the median of the original data. Finally, the slope is the regression coefficient of *s* on *l*, which is defined as

$$\frac{\sum_{i=1}^{k}\left(l_i-\bar{l}\right)\left(s_i-\bar{s}\right)}{\sum_{i=1}^{k}\left(l_i-\bar{l}\right)^2}$$

In some situations, the transformations cannot be done. The spread-versus-level plot and Levene statistic will not be produced if:

- *a* is a negative integer and at least one of the data is 0
- *a* is a negative non-integer and at least one of the data is less than or equal to 0
- *a* is a positive non-integer and at least one of the data is less than 0
- *a* is not specified and the median or the spread is less than or equal to 0

Levene Test of Homogeneity of Variances

The Levene test statistic is based on the transformed data and is defined by

$$L_a=\left(\frac{W-k}{k-1}\right)\frac{\sum_{i=1}^{k}w_i(\bar{z}_i-\bar{z})^2}{\sum_{i=1}^{k}\sum_{l=1}^{m_i}c_{il}(z_{il}-\bar{z}_i)^2}$$

where

$$w_i=\sum_{l=1}^{m_i}c_{il}$$

$$\bar{x}_i=\frac{\sum_{l=1}^{m_i}c_{il}x_{il}}{w_i}$$

$$z_{il}=\left|x_{il}-\bar{x}_i\right|$$

$$\bar{z}_i=\sum_{k=1}^{m_i}\frac{c_{il}z_{il}}{w_i}$$

$$\bar{z}=\sum_{i=1}\frac{w_i\bar{z}_i}{W}$$

The significance of L_a is calculated from the *F* distribution with degrees of freedom $k-1$ and $W-k$.

Groups with zero variance are included in the test.

Robust Levene's Test of Homogeneity of Variances

With the current version of Levene's L_a the followings can be considered as options in order to obtain robust Levene's tests:

- Levene's test L_b based on $z_{li}^{(b)}$ = $|X_{il} - \tilde{x}_i|$ where $\tilde{x}_i$ is the median of X_{il}'s for group *i*.
  Median calculation is done by the method requested. The default method is HAVERAGE.
  Once the $\tilde{x}_i$'s and hence $z_{li}^{(b)}$'s are calculated, apply the formula for L_a, shown in the section above, to obtain L_b by replacing Z_{il}, $\overline{z}_i$ and $\overline{z}$ with $z_{li}^{(b)}$, $\overline{z}_i^{(b)}$ and $\overline{z}^{(b)}$ respectively.
  Two significances of L_b are given. One is calculated from a *F*-distribution with degrees of freedom $k-1$ and $W-$ k. Another is calculated from a *F*-distribution with degrees of freedom $k-1$ and *v*. The value of *v* is given by:

$$v = \frac{\left(\sum_{i=1}^{k} u_i\right)^2}{\left(\sum_{i=1}^{k} \frac{u_i^2}{v_i}\right)}$$

where

$$u_i = \sum_{l=1}^{m_i} c_{il}\left(z_{il}^{(b)} - \overline{z}_i^{(b)}\right)^2$$

in which

$$\overline{z}_i^{(b)} = \sum_{l=1}^{m_i} \frac{c_{il} z_{il}^{(b)}}{w_i}$$

and

$$v_i = w_i - 1$$

- Levene's test L_c based on $z_{il}^{(c)} = |x_{il} - T_{i,0.9}|$ where $T_{i,0.9}$ is the 5% trimmed mean of x_{il}'s for group *i*.
  Once the $T_{i,0.9}$'s and hence $z_{il}^{(c)}$'s are calculated, apply the formula of L_a to obtain L_c by replacing zil, $\overline{z}_i$ and $\overline{z}$ with $z_{li}^{(c)}$, $\overline{z}_i^{(c)}$ and $\overline{z}^{(c)}$ respectively.
  The significance of L_c is calculated from a *F*-distribution with degrees of freedom $k-1$ and $W-k$.

Plots

The following plots are available.

Normal Probability Plot (NPPLOT)

For every distinct observation y_i, R_i is the rank (the mean of ranks is assigned to ties). The normal score NS_i is calculated by

$$NS_i = \Psi^{-1}\left(\frac{R_i}{W+1}\right)$$

where Ψ^{-1} is the inverse of the standard normal cumulative distribution function. The NPPLOT is the plot of $(y_1, NS_1), \ldots, (y_m, NS_m)$.

Detrended Normal Plot

The detrended normal plot is the scatterplot of $(y_1, D_1), \ldots, (y_m, D_m)$, where D_i is the difference between the *Z*-score and normal score, which is defined by

$$D_i = Z_i - NS_i$$

and

$$Z_i = \frac{y_i - \overline{y}}{s}$$

where $\overline{y}$ is the average and *s* is the standard deviation.

Boxplot

The boundaries of the box are Tukey's hinges. The length of the box is the interquartile range based on Tukey's hinges. That is,

$$IQR = Q_3 - Q_1$$

Define

STEP = 1.5 IQR

A case is an outlier if

$$Q_3 + STEP \le y_i < Q_3 + 2STEP$$

or

$$Q_1 - 2STEP < y_i \le Q_1 - STEP$$

A case is an extreme if

$y_i \geq Q_3 + 2STEP$

or

$y_i \leq Q_1 - 2STEP$

References

Brown, M. B., and A. B. Forsythe. 1974b. Robust tests for the equality of variances. *Journal of the American Statistical Association*, 69, 364–367.

Dallal, G. E., and L. Wilkinson. 1986. An analytic approximation to the distribution of Lilliefor’s test statistic for normality. *The American Statistician*, 40(4): 294–296 (Correction: 41: 248), – .

Frigge, M., D. C. Hoaglin, and B. Iglewicz. 1987. Some implementations for the boxplot. In: *Computer Science and Statistics Proceedings of the 19th Symposium on the Interface,* R. M. Heiberger, and M. Martin, eds. Alexandria, Virginia: AmericanStatistical Association.

Glaser, R. E. 1983. Levene’s Robust Test of Homogeneity of Variances. In: *Encyclopedia of Statistical Sciences 4*, New York: Wiley, p608–610.

Hoaglin, D. C., F. Mosteller, and J. W. Tukey. 1983. *Understanding robust and exploratory data analysis*. New York: John Wiley and Sons.

Hoaglin, D. C., F. Mosteller, and J. W. Tukey. 1985. *Exploring data tables, trends, and shapes*. New York: John Wiley and Sons.

Levene, H. 1960. Robust tests for equality of variances. In: *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, I. Olkin, ed. Palo Alto, Calif.: Stanford University Press, 278–292.

Lilliefors, H. W. 1967. On the Kolmogorov-Smirnov tests for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62, 399–402.

Loh, W. Y. 1987. Some Modifications of Levene’s Test of Variance Homogeneity. *Journal of the Statistical Computation and Simulation*, 28, 213–226.

Shapiro, S. S., and M. B. Wilk. 1965. An analysis of variance test for normality. *Biometrika*, 52:3, 591–599.

Tukey, J. W. 1977. *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Velleman, P. F., and D. C. Hoaglin. 1981. *Applications, basics, and computing of exploratory data analysis*. Boston, Mass.: Duxbury Press.

EXSMOOTH Algorithms

EXSMOOTH produces one period ahead forecasts for different models.

Notation

The following notation is used throughout this section unless otherwise stated:

X_t	Observed series, $t = 1, \ldots, n$
$\hat{X}_t$	Forecast of one period ahead from time t
P	Number of periods
K	Number of complete cycles $([n/p])$
e_t	tth residual $\left(X_t - \hat{X}_{t-1}\right)$
S_0	Initial value for series
T_0	Initial value for trend
$I_{1-p}, \ldots, I_0$	Initial values for seasonal factors
m_l	Mean for the lth cycle, $\sum_{i=(l-1)p+1}^{lp} X_i/p$

Note the following points:

- $I_{1-p}, \ldots, I_0$ are obtained from the SEASON procedure with MA = EQUAL if p is even; otherwise MA = CENTERED is used for both multiplicative and additive models.
- The index for the fitted series starts with zero.
- The value saved in the FIT variable for the tth case is $\hat{X}_{t-1}$.

Models

The following models are available.

No Trend, No Seasonality Model

$$X_t = b + \epsilon_t$$

Initial value

$$S_0 = \overline{X}$$

then

$$\hat{X}_0 = S_0, \quad e_1 = X_1 - \hat{X}_0$$

$$S_t = S_{t-1} + \alpha e_t$$

$$\hat{X}_t = S_t$$

No Trend, Additive Seasonality Model

$$X_t = b + I_t + \epsilon_t$$

Initial value

$$S_0 = \frac{\sum_{i=1}^{k} m_i}{k}$$

then

$$\hat{X}_0 = S_0 + I_{1-p}$$
$$e_1 = X_1 - \hat{X}_0$$

$$S_t = S_{t-1} + \alpha e_t$$

$$I_t = I_{t-p} + \delta(1-\alpha)e_t$$

$$\hat{X}_t = S_t + I_{t-p+1}$$

No Trend, Multiplicative Seasonality Model

$$X_t = bI_t + \epsilon_t$$

Initial value

$$S_0 = \frac{\sum_{i=1}^{k} m_i}{k}$$

then

$$\hat{X}_0 = S_0 I_{1-p}$$
$$e_1 = X_1 - \hat{X}_0$$
$$S_t = S_{t-1} + \alpha e_t / I_{t-p}$$
$$I_t = I_{t-p} + \delta(1-\alpha) e_t / S_t$$
$$\hat{X}_t = S_t I_{t-p+1}$$

Linear Trend, No Seasonality Model

$$X_t = b_0 + b_1 t + \epsilon_t$$

Initial values

$$T_0 = \frac{X_n - X_1}{n-1}$$
$$S_0 = X_1 - \frac{1}{2} T_0$$

then

$$\hat{X}_0 = S_0 + T_0$$
$$e_1 = X_1 - \hat{X}_0$$
$$S_t = S_{t-1} + T_{t-1} + \alpha e_t$$
$$T_t = T_{t-1} + \alpha\gamma e_t$$
$$\hat{X}_t = S_t + T_t$$

Linear Trend, Additive Seasonality Model

$$X_t = b_0 + b_1 t + I_t + \epsilon_t$$

Initial values

$$T_0 = \frac{m_k - m_1}{(k-1)p}$$
$$S_0 = X_1 - \frac{p}{2} T_0$$

then

$$\hat{X}_0 = S_0 + T_0 + I_{1-p}$$
$$S_t = S_{t-1} + T_{t-1} + \alpha e_t$$

$$T_t = T_{t-1} + \alpha\gamma e_t$$

$$I_t = I_{t-p} + \delta(1-\alpha)e_t$$

$$\hat{X}_t = S_t + T_t + I_{t-p+1}$$

Linear Trend, Multiplicative Seasonality Model

$$X_t = (b_0 + b_1 t)\, I_t + \epsilon_t$$

Initial values

$$T_0 = \frac{m_k - m_1}{(k-1)p}$$

$$S_0 = m_1 - \frac{p}{2}T_0$$

then

$$\hat{X}_0 = (S_0 + T_0)\, I_{1-p}$$

$$S_t = S_{t-1} + T_{t-1} + \alpha(e_t / I_{t-p})$$
$$T_t = T_{t-1} + \alpha\gamma(e_t / I_{t-p})$$
$$I_t = I_{t-p} + \delta(1-\alpha)(e_t / S_t)$$
$$\hat{X}_t = (S_t + T_t)\, I_{t-p+1}$$

Exponential Trend, No Season Model

$$X_t = b_0 b_1^t + \epsilon_t$$

Initial values

$$T_0 = \exp\{\ln X_2 - \ln X_1\} = \frac{X_2}{X_1}$$

$$S_0 = \exp\left\{\ln X_1 - \frac{1}{2}\ln T_0\right\} = \frac{X_1}{\sqrt{T_0}}$$

then

$$\hat{X}_0 = S_0 T_0$$

$$S_t = S_{t-1} T_{t-1} + \alpha e_t$$

$$T_t = T_{t-1} + \alpha\gamma e_t / S_{t-1}$$

$$\hat{X}_t = S_t T_t$$

Exponential Trend, Additive Seasonal Model

$$X_t = b_0 b_1^t + I_t + \epsilon_t$$

Initial values

$$T_0 = \exp\left\{(\ln m_2 - \ln m_1)/p\right\}$$

$$S_0 = \exp\left\{\ln m_1 - \frac{p}{2}\ln T_0\right\}$$

then

$$\hat{X}_0 = S_0 T_0 + I_{1-p}$$
$$S_t = S_{t-1} T_{t-1} + \alpha e_t$$

$$T_t = T_{t-1} + \alpha\gamma e_t / S_{t-1}$$

$$I_t = I_{t-p} + \delta(1-\alpha) e_t$$

$$\hat{X}_t = S_t T_t + I_{t-p+1}$$

Exponential Trend, Multiplicative Seasonality Model

$$X_t = \left(b_0 b_1^t\right) I_t + \epsilon_t$$

Initial values

$$T_0 = \exp\left\{(\ln m_2 - \ln m_1)/(k-1)\right\}$$

$$S_0 = \exp\left\{\ln m_1 - \frac{p}{2}\ln T_0\right\}$$

then

$$\hat{X}_0 = (S_0 T_0)\, I_{1-p}$$

$$S_t = S_{t-1} T_{t-1} + \alpha e_t / I_{t-p}$$

$$T_t = T_{t-1} + \alpha\gamma e_t / (I_{t-p} S_{t-1})$$

$$I_t = I_{t-p} + \delta(1-\alpha) e_t / S_t$$

$$\hat{X}_t = (S_t T_t)\, I_{t-p+1}$$

Damped Trend, No Seasonality Model

$$X_t = b_0 + \phi b_1 t + \epsilon_t$$

Initial values

$$T_0 = \frac{X_n - X_1}{(n-1)\phi}$$

$$S_0 = X_1 - \frac{1}{2} T_0$$

then

$$\hat{X}_0 = S_0 + \phi T_0$$
$$S_t = S_{t-1} + \phi T_{t-1} + \alpha e_t$$

$$T_t = \phi T_{t-1} + \alpha\gamma e_t$$

$$\hat{X}_t = S_t + \phi T_t$$

Damped Trend, Additive Seasonality Model

$$X_t = b_0 + \phi b_1 t + I_t + \epsilon_t$$

Initial values

$$T_0 = \frac{m_k - m_1}{(k-1)p\phi}$$

$$S_0 = m_1 - \frac{p}{2} T_0$$

then

$$\hat{X}_0 = S_0 + \phi T_0 + I_{1-p}$$
$$S_t = S_{t-1} + \phi T_{t-1} + \alpha(2-\alpha)e_t$$

$$T_t = \phi T_{t-1} + \alpha(\alpha - \phi + 1)e_t$$

$$I_t = I_{t-p} + \delta[1 - \alpha(2-\alpha)]e_t$$

$$\hat{X}_t = S_t + \phi T_t + I_{t-p+1}$$

Damped Trend, Multiplicative Seasonality Model

$$X_t = (b_0 + b_1 \phi t)I_t + \epsilon_t$$

Initial values

$$T_0 = \frac{m_k - m_1}{(k-1)p\phi}$$

$$S_0 = m_1 - \frac{p}{2}T_0\phi$$

then

$$\hat{X}_0 = (S_0 + \phi T_0)I_{1-p}$$
$$S_t = S_{t-1} + \phi T_{t-1} + \alpha(2-\alpha)e_t/I_{t-p}$$

$$T_t = \phi T_{t-1} + \alpha(\alpha - \phi + 1)e_t/I_{t-p}$$

$$I_t = I_{t-p} + \delta[1 - \alpha(2-\alpha)]e_t/S_t$$

$$\hat{X}_t = (S_t + \phi T_t)I_{t-p+1}$$

References

Abraham, B., and J. Ledolter. 1983. *Statistical methods of forecasting*. New York: John Wiley and Sons.

Gardner, E. S. 1985. Exponential smoothing: The state of the art. *Journal of Forecasting*, 4, 1–28.

Ledolter, J., and B. Abraham. 1984. Some comments on the initialization of exponential smoothing. *Journal of Forecasting*, 3, 79–84.

Makridakis, S., S. C. Wheelwright, and V. E. McGee. 1983. *Forecasting: Methods and applications*. New York: John Wiley and Sons.

FACTOR Algorithms

FACTOR performs factor analysis based either on correlations or covariances and using one of the seven extraction methods.

Extraction of Initial Factors

The following extraction methods are available.

Principal Components Extraction (PC)

The matrix of factor loadings based on factor *m* is

$$\Lambda_m = \Omega_m \Gamma_m^{1/2}$$

where

$$\Omega_m = (\omega_1, \omega_2, \ldots, \omega_m)$$
$$\Gamma_m = diag(|\gamma_1|, |\gamma_2|, \ldots, |\gamma_m|)$$

The communality of variable *i* is given by

$$h_i = \sum_{j=1}^{m} |\gamma_j| \omega_{ij}^2$$

Analyzing a Correlation Matrix

$\gamma_1 \geq \gamma_2 \geq \ldots \geq \gamma_m$ are the eigenvalues and ω_i are the corresponding eigenvectors of **R**, where **R** is the correlation matrix.

Analyzing a Covariance Matrix

$\gamma_1 \geq \gamma_2 \geq \ldots \geq \gamma_m$ are the eigenvalues and ω_i are the corresponding eigenvectors of Σ, where $\Sigma = (\sigma_{ij})_{n \times n}$ is the covariance matrix.

The rescaled loadings matrix is $\Lambda_{mR} = [diag\Sigma]^{-1/2} \Lambda_m$.

The rescaled communality of variable *i* is $h_{iR} = \sigma_{ii}^{-1} h_i$.

Principal Axis Factoring

Analyzing a Correlation Matrix

An iterative solution for communalities and factor loadings is sought. At iteration *i*, the communalities from the preceding iteration are placed on the diagonal of **R**, and the resulting **R** is denoted by $\mathbf{R}_i$. The eigenanalysis is performed on $\mathbf{R}_i$ and the new communality of variable *j* is estimated by

$$h_{j(i)} = \sum_{j=1}^{m} \left|\gamma_{k(i)}\right| \omega_{jk(i)}^2$$

The factor loadings are obtained by

$$\Lambda_{m(i)} = \Omega_{m(i)} \Gamma_{m(i)}^{1/2}$$

Iterations continue until the maximum number (default 25) is reached or until the maximum change in the communality estimates is less than the convergence criterion (default 0.001).

Analyzing a Covariance Matrix

This analysis is the same as analyzing a correlation matrix, except Σ is used instead of the correlation matrix **R**. Convergence is dependent on the maximum change of *rescaled* communality estimates.

At iteration i, the rescaled loadings matrix is $\Lambda_{m(i)R} = [diag\Sigma]^{-1/2}\Lambda_{m(i)}$. The rescaled communality of variable *i* is $h_{j(i)R} = \sigma_{ii}^{-1} h_{j(i)}$.

Maximum Likelihood (ML)

The maximum likelihood solutions of Λ and ψ^2 are obtained by minimizing

$$F = tr\left[\left(\Lambda\Lambda^{'} + \psi^2\right)^{-1}\mathbf{R}\right] - \log\left|\left(\Lambda\Lambda^{'} + \psi^2\right)^{-1}\mathbf{R}\right| - p$$

with respect to Λ and ψ, where *p* is the number of variables, Λ is the factor loading matrix, and ψ^2 is the diagonal matrix of unique variances.

The minimization of *F* is performed by way of a two-step algorithm. First, the conditional minimum of *F* for a given y is found. This gives the function $f(\psi)$, which is minimized numerically using the Newton-Raphson procedure. Let $\mathbf{x}^{(s)}$ be the column vector containing the logarithm of the diagonal elements of y at the *s*th iteration; then

$$\mathbf{x}^{(s+1)} = \mathbf{x}^{(s)} - \mathbf{d}^{(s)}$$

where $\mathbf{d}^{(s)}$ is the solution to the system of linear equations

$$\mathbf{H}^{(s)}\mathbf{d}^{(s)} = \mathbf{h}^{(s)}$$

and where

$$\mathbf{H}^{(s)} = \left(\partial^2 f(\psi) / \partial x_i \partial x_j\right)$$

and $\mathbf{h}^{(s)}$ is the column vector containing $\partial f(\psi)/\partial x_i$. The starting point $\mathbf{x}^{(1)}$ is

$$\mathbf{x}_i^{(1)} = \begin{cases} \log\left[(1-m/2p)/r^{ii}\right] & \text{for ML and GLS} \\ \left[(1-m/2p)/r^{ii}\right]^{1/2} & \text{for ULS} \end{cases}$$

where *m* is the number of factors and r^{ii} is the *i*th diagonal element of $\mathbf{R}^{-1}$.

The values of $f(\psi)$, $\partial f/\partial x_i$, and $\partial^2 f/\partial x_i \partial x_j$ can be expressed in terms of the eigenvalues

$$\gamma_1 \le \gamma_2 \le \ldots \le \gamma_p$$

and corresponding eigenvectors

$$\omega_1, \omega_2, \ldots, \omega_p$$

of matrix $\psi \mathbf{R}^{-1} \psi$. That is,

$$f(\psi) = \sum_{k=m+1}^{p} \left(\log \gamma_k + \gamma_k^{-1} - 1\right)$$

$$\frac{\partial f}{\partial x_i} = \sum_{k=m+1}^{p} \left(1 - \gamma_k^{-1}\right)\omega_{ik}^2$$

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = -\delta_{ij}\frac{\partial f}{\partial x_i} + \sum_{k=m+1}^{p} \omega_{ik}\omega_{jk}\left(\sum_{n=1}^{m} \frac{\gamma_k + \gamma_n - 2}{\gamma_k - \gamma_n}\omega_{in}\omega_{jn} + \delta_{ij}\right)$$

where

$$\delta_{ij} = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \ne j \end{cases}$$

The approximate second-order derivatives

$$\frac{\partial^2 f}{\partial x_i \partial x_j} \cong \left(\sum_{k=m+1}^{p} \omega_{ik}\omega_{jk}\right)^2$$

are used in the initial step and when the matrix of the exact second-order derivatives is not positive definite or when all elements of the vector $\mathbf{d}$ are greater than 0.1. If $\partial^2 f/\partial x_i^2 < 0.05$ (Heywood variables), the diagonal element is replaced by 1 and the rest of the elements of that column and row are set to 0. If the value of $f(\psi)$ is not decreased by step $\mathbf{d}$, the step is halved and halved again until the value of $f(\psi)$ decreases or 25 halvings fail to produce a decrease. (In this case, the computations are terminated.) Stepping continues until the largest absolute value of the elements of $\mathbf{d}$ is less than the criterion value (default 0.001) or until the maximum number of iterations (default 25) is reached. Using the converged value of ψ (denoted by $\hat{\psi}$), the eigenanalysis is performed on the matrix $\hat{\psi}\mathbf{R}^{-1}\hat{\psi}$. The factor loadings are computed as

$$\hat{\Lambda}_m = \hat{\psi}\Omega_m\left(\Gamma_m^{-1} - \mathbf{I}_m\right)^{1/2}$$

where

$$\Gamma_m = \text{diag}(\gamma_1, \gamma_2, \ldots, \gamma_m)$$
$$\Omega_m = (\omega_1, \omega_2, \cdots, \omega_m)$$

Unweighted and Generalized Least Squares (ULS, GLS)

The same basic algorithm is used in ULS and GLS methods as in maximum likelihood, except that

$$f(\psi) = \begin{cases} \sum_{k=m+1}^{p} \frac{\gamma_k^2}{2} & \text{for ULS} \\ \sum_{k=m+1}^{p} \frac{(\gamma_k - 1)^2}{2} & \text{for GLS} \end{cases}$$

for the ULS method, the eigenanalysis is performed on the matrix $\mathbf{R} - \psi^2$, where $\gamma_1 \geq \gamma_2 \geq \ldots \geq \gamma_p$ are the eigenvalues. In terms of the derivatives, for ULS

$$\frac{\partial f}{\partial x_i} = 2x_i \sum_{k=m+1}^{p} \gamma_k \omega_{ik}^2$$

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = 4\left[x_i x_j \sum_{k=m+1}^{p} \omega_{ik}\omega_{jk} \sum_{n=1}^{m} \frac{\gamma_k + \gamma_n}{\gamma_k - \gamma_n}\omega_{ik}\omega_{jk} + \delta_{ij} \sum_{k=m+1}^{p} \left(x_i^2 - \frac{\gamma_k}{2}\right)\omega_{ik}^2\right]$$

and

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = 4x_i x_j \left(\sum_{k=m+1}^{p} \omega_{ik}\omega_{jk}\right)^2$$

For GLS

$$\frac{\partial f}{\partial x_i} = \sum_{k=m+1}^{p} \left(\gamma_k^2 - \gamma_k\right)\omega_{ik}^2$$

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \delta_{ij}\frac{\partial f}{\partial x_i} + \sum_{k=m+1}^{p} \gamma_k \omega_{ik}\omega_{jk}\left(\sum_{n=1}^{m} \gamma_n \frac{\gamma_k + \gamma_n - 2}{\gamma_k - \gamma_n}\omega_{in}\omega_{jn} + r^{ii}\exp\left[(x_i + x_j)/2\right]\right)$$

and

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \left(\sum_{k=m+1}^{p} \omega_{ik}\omega_{jk} \right)^2$$

Also, the factor loadings of the ULS method are obtained by

$$\hat{\Lambda}_m = \Omega_m \Gamma_m^{1/2}$$

The chi-square statistics for *m* factors for the ML and GLS methods is given by

$$\chi_m^2 = \left(W - 1 - \frac{2p+5}{6} - \frac{2m}{3} \right) f\left(\hat{\psi}\right)$$

with $\left((p-m)^2 - p - m\right)/2$

Alpha (Harman, 1976)

Iteration for Communalities

At each iteration *i*:

- The eigenvalues $\left(r_{(i)}\right)$ and eigenvectors $\left(\Omega_{(i)}\right)$ of $\mathbf{H}_{(i-1)}^{1/2}(\mathbf{R}-\mathbf{I})\mathbf{H}_{(i-1)}^{1/2} + \mathbf{I}$ are computed.
- The new communalities are

$$h_{k(i)} \left(\sum_{j=1}^{m} \left|\gamma_{j(i)}\right| \omega_{kj(i)}^2 \right) h_{k(i-1)}$$

The initial values of the communalities, $\mathbf{H}_0$, are

$$h_{io} = \begin{cases} 1 - 1/r^{ii} & |\mathbf{R}| \geq 10^{-8} \text{ and all } 0 \leq h_{io} \leq 1 \\ \max_j |r_{ij}| & \text{otherwise} \end{cases}$$

where r^{ii} is the *i*th diagonal entry of $\mathbf{R}^{-1}$.

If $|\mathbf{R}| \geq 10^{-8}$ and all r^{ii} are equal to one, the procedure is terminated. If for some *i*, $\max_j |r_{ij}| > 1$, the procedure is terminated.

- Iteration stops if any of the following are true:

$$\max_k \left|h_{k(i)} - h_{k(i-1)}\right| < \text{EPS}$$

$$i = \text{MAX}$$

$$h_{k(i)} = 0 \text{for any} k$$

Final Communalities and Factor Pattern Matrix

The communalities are the values when iteration stops, unless the last termination criterion is true, in which case the procedure terminates. The factor pattern matrix is

$$\mathbf{F}_m = \mathbf{H}_{(f)}^{1/2} \Omega_{m(f)} \Gamma_{m(f)}^{1/2}$$

where *f* is the final iteration.

Image (Kaiser, 1963)

Factor Analysis of a Correlation Matrix

- Eigenvalues and eigenvectors of $\mathbf{S}^{-1}\mathbf{R}\mathbf{S}^{-1}$ are found.
  $\mathbf{S}^2 = \text{diag}(1/r^{11}, \ldots, 1/r^{nn})$
  $r^{ii} = i$th diagonal element of $\mathbf{R}^{-1}$
- The factor pattern matrix is
  $\mathbf{F}_m = \mathbf{S}\Omega_m(\Lambda_m - I_m)\Lambda_m^{-1/2}$

  where Ω_m and Λ_m correspond to the *m* eigenvalues greater than 1.

  If $m = 0$, the procedure is terminated.
- The communalities are

$$h_i = \sum_{j=1}^{m} (\gamma_j - 1)^2 \omega_{ij}^2 / (\gamma_j r^{ii})$$

- The image covariance matrix is

$$\mathbf{R} + \mathbf{S}^2\mathbf{R}^{-1}\mathbf{S}^2 - 2\mathbf{S}^2$$

- The anti-image covariance matrix is

$$\mathbf{S}^2\mathbf{R}^{-1}\mathbf{S}^2$$

Factor Analysis of a Covariance Matrix

We are using the covariance matrix Σ instead of the correlation matrix $\mathbf{R}$. The calculation is similar to the correlation matrix case.

The rescaled factor pattern matrix is $\mathbf{F}_{mR} = [diag\Sigma]^{-1/2}\mathbf{F}_m$. The rescaled communality of variable *i* is $h_{iR} = \sigma_{ii}^{-1} h_i$.

Factor Rotations

The following rotation methods are available.

Orthogonal Rotations (Harman, 1976)

Rotations are done cyclically on pairs of factors until the maximum number of iterations is reached or the convergence criterion is met. The algorithm is the same for all orthogonal rotations, differing only in computations of the tangent values of the rotation angles.

- The factor pattern matrix is normalized by the square root of communalities:

$$\Lambda_m^* = \mathbf{H}^{-1/2}\Lambda_m$$

where

$\Lambda_m = (\underline{\lambda}_1, \ldots, \underline{\lambda}_m)$ is the factor pattern matrix

$\mathbf{H} = \text{diag}(h_1, \ldots, h_n)$ is the diagonal matrix of communalities

- The transformation matrix **T** is initialized to $\mathbf{I}_m$
- At each iteration *i*

(1) The convergence criterion is

$$SV_{(i)} = \sum_{j=1}^{m} \left(n \sum_{k=1}^{n} \lambda_{kj(i)}^{*4} - \left(\sum_{k=1}^{n} \lambda_{kj(i)}^{*2} \right)^2 \right) / n^2$$

where the initial value of $\Lambda_{m(1)}^*$ is the original factor pattern matrix. For subsequent iterations, the initial value is the final value of $\Lambda_{m(i-1)}^*$ when all factor pairs have been rotated.

(2) For all pairs of factors (λ_j, λ_k) where $k > j$, the following are computed:

(a) Angle of rotation

$$P = 1/4\tan^{-1}(X/Y)$$

where

$$X = \begin{cases} D - 2AB/n & \text{Varimax} \\ D - mAB/n & \text{Equimax} \\ D & \text{Quartimax} \end{cases}$$

$$Y = \begin{cases} C - \left(A^2 - B^2\right)/n & \text{Varimax} \\ C - m\left(A^2 - B^2\right)/2n & \text{Equimax} \\ C & \text{Quartimax} \end{cases}$$

$$u_{p(i)} = f^{*2}_{pj(i)} - f^{*2}_{pk(i)} \qquad v_{p(i)} = 2f^*_{pj(i)} f^*_{pk(i)} \qquad p = 1, \ldots, n$$

$$A = \sum_{p=1}^{n} u_{p(i)} \qquad B = \sum_{p=1}^{n} v_{p(i)}$$

$$C = \sum_{p=1}^{n} \left[u^2_{p(i)} - v^2_{p(i)}\right] \qquad D = \sum_{p=1}^{n} 2u_{p(i)} v_{p(i)}$$

If $|\sin(P)| \le 10^{-15}$, no rotation is done on the pair of factors.

(b) New rotated factors

$$\left(\tilde{\underline{\lambda}}_{j(i)}, \tilde{\underline{\lambda}}_{k(i)}\right) = \left(\underline{\lambda}^*_{j(i)}, \underline{\lambda}^*_{k(i)}\right) \quad \begin{vmatrix} \cos(P) & -\sin(P) \\ \sin(P) & \cos(P) \end{vmatrix}$$

where $\underline{\lambda}^*_{j(i)}$ are the last values for factor *j* calculated in this iteration.

(c) Accrued rotation transformation matrix

$$\left(\tilde{t}_j, \tilde{t}_k\right) = \left(t_j, t_k\right) \quad \begin{vmatrix} \cos(P) & -\sin(P) \\ \sin(P) & \cos(P) \end{vmatrix}$$

where t_j and t_k are the last calculated values of the *j*th and *k*th columns of *T*.

(d) Iteration is terminated when

$$\left|SV_{(i)} - SV_{(i-1)}\right| \le 10^{-5}$$

or the maximum number of iterations is reached.

(e) Final rotated factor pattern matrix

$$\tilde{\Lambda}_m = \mathbf{H}^{1/2} \Lambda^*_{m(f)}$$

where

$$\Lambda^*_{m(f)}$$

is the value of the last iteration.

(f) Reflect factors with negative sums

If

$$\sum_{i=1}^{n} \tilde{\lambda}_{ij(f)} < 0$$

then

$$\underline{\tilde{\lambda}}_j = -\underline{\tilde{\lambda}}_{j(f)}$$

(g) Rearrange the rotated factors such that

$$\sum_{j=1}^{n} \tilde{\lambda}^2_{j1} \geq \cdots \geq \sum_{j=1}^{n} \tilde{\lambda}^2_{jm}$$

(h) The communalities are

$$h_j = \sum_{i=1}^{m} \tilde{\lambda}^2_{ji}$$

Oblique Rotations

The direct oblimin method (Jennrich and Sampson, 1966) is used for oblique rotation. The user can choose the parameter δ. The default value is $\delta = 0$.

(a) The factor pattern matrix is normalized by the square root of the communalities

$$\Omega^*_m = \mathbf{H}^{-1/2}\Lambda_m$$

where

$$h_j = \sum_{k=1}^{m} \lambda^2_{jk}$$

If no Kaiser is specified, this normalization is not done.

(b) Initializations

The factor correlation matrix **C** is initialized to $\mathbf{I}_m$. The following are also computed:

$$s_k = \begin{cases} 1 & \text{if Kaiser} \\ h_k & \text{if no Kaiser} \end{cases} \qquad k = 1, \ldots, n$$

$$u_i = \sum_{j=1}^{n} \lambda_{ji}^{*2} \qquad i = 1, \ldots, m$$

$$v_i = \sum_{j=1}^{n} \lambda_{ji}^{*4}$$

$$x_i = v_i - (\delta/n) u_i^2$$

$$D = \sum_{i=1}^{m} u_i$$

$$G = \sum_{i=1}^{m} x_i$$

$$H = \sum_{k=1}^{n} s_i^2 - (\delta/n) D^2$$

$$FO = H - G$$

(c) At each iteration, all possible factor pairs are rotated. For a pair of factors $\underline{\lambda}_p^*$ and $\underline{\lambda}_q^* (p \neq q)$, the following are computed:

$$D_{pq} = D - u_p - u_q$$

$$G_{pq} = G - x_p - x_q$$

$$s_{pq,i} = s_i - \lambda_{ip}^{*2} - \lambda_{iq}^{*2}$$

$$y_{pq} = \sum_{i=1}^{n} \lambda_{ip}^* \lambda_{iq}^*$$

$$z_{pq} = \sum_{i=1}^{n} \lambda_{ip}^{*2} \lambda_{iq}^{*2}$$

$$T = \sum_{i=1}^{n} s_{pq,i} \lambda_{ip}^{*2} - (\delta/n) u_p D_{pq}$$

$$Z = \sum_{i=1}^{n} s_{pq,i} \lambda_{ip}^* \lambda_{iq}^* - (\delta/n) y_{pq} D_{pq}$$

$$P = \sum_{i=1}^{n} \lambda_{ip}^{*3} \lambda_{iq}^* - (\delta/n) u_p y_{pq}$$

$$R = z_{pq} - (\delta/n) u_p u_q$$

$$P' = \frac{3}{2}(c_{pq} - P/x_p)$$

$$Q' = \frac{1}{2}(x_p - 4c_{pq}P + R + 2T)/x_p$$

$$R' = \frac{1}{2}(c_{pq}(T + R) - P - Z)/x_p$$

- A root, *a*, of the equation

$b^3 + P'b^2 + Q'b + R' = 0$ is computed, as well as:

$$A = 1 + 2c_{pq}a + a^2$$
$$t_1 = |A|^{1/2}$$
$$t_2 = a/t_1$$

- The rotated pair of factors is

$$\left(\tilde{\underline{\lambda}}_p^*, \tilde{\underline{\lambda}}_q^*\right) = \left(\underline{\lambda}_p^*, \underline{\lambda}_q^*\right) \begin{vmatrix} t_1 & -a \\ 0 & 1 \end{vmatrix}$$

These replace the previous factor values.

- New values are computed for

$$\tilde{u}_p = |A| u_p$$

$$\tilde{x}_p = A^2 x_p$$

$$\tilde{v}_q = \sum_{i=1}^{n} \tilde{\lambda}_{iq}^{*4}$$

$$\tilde{u}_q = \sum_{i=1}^{n} \tilde{\lambda}_{iq}^{*2}$$

$$\tilde{x}_q = \tilde{v}_q - (\delta/n)\tilde{u}_q^2$$

$$\tilde{S}_k = S_{pq,k} + \tilde{\lambda}_{kp}^{*2} + \tilde{\lambda}_{kq}^{*2}$$

$$\tilde{D} = D_{pq} + \tilde{u}_p + \tilde{u}_q$$

$$\tilde{G} = G_{pq} + \tilde{x}_p + \tilde{x}_q$$

All values designated as $\tilde{V}$ replaces *V* and are used in subsequent calculations.

► The new factor correlations with factor *p* are

$$\tilde{c}_{ip} = t_1^{-1} c_{ip} + t_2 c_{iq} \quad (i \neq p)$$

$$\tilde{c}_{pi} = \tilde{c}_{ip}$$

$$\tilde{c}_{pp} = 1$$

► After all factor pairs have been rotated, iteration is terminated if

MAX iterations have been done or

$$\left| F1_{(i)} - F1_{(i-1)} \right| < (FO)(EPS)$$

where

$$F1_{(i)} = \tilde{H} - \tilde{G}$$

$$\tilde{H} = \sum_{k=1}^{n} \tilde{s}_k^2 - (\delta/n)\tilde{D}^2$$

$$F1_{(0)} = FO$$

Otherwise, the factor pairs are rotated again.

► The final rotated factor pattern matrix is

$$\tilde{\lambda}_m = \mathbf{H}^{1/2} \tilde{\lambda}_m^*$$

where $\tilde{\lambda}_m$ is the value in the final iteration.

- The factor structure matrix is

$$\mathbf{S} = \tilde{\Lambda}_m \tilde{\mathbf{C}}_m$$

where $\tilde{\mathbf{C}}_m$ is the factor correlation matrix in the final iteration.

Promax Rotation

(Hendrickson and White, 1964) proposed a computationally fast rotation. The speed is achieved by first rotating to an orthogonal varimax solution and then relaxing the orthogonality of the factors to better fit simple structure.

- Varimax rotation is used to get an orthogonal rotated matrix $\Lambda_R = \{\lambda_{ij}\}$.
- The matrix $\mathbf{P} = (p_{ij})_{p\times m}$ is calculated, where

$$p_{ij} = \left| \frac{\lambda_{ij}}{\left(\sum_{j=1}^{m} \lambda_{ij}^2\right)^{1/2}} \right|^{k+1} \left(\sum_{j=1}^{m} \lambda_{ij}^2\right)^{1/2} / \lambda_{ij}$$

Here, *k* (*k* > 1) is the power of promax rotation.

- The matrix **L** is calculated.

$$\mathbf{L} = \left(\Lambda'_R \Lambda_R\right)^{-1} \Lambda'_R \mathbf{P}$$

- The matrix **L** is normalized by column to a transformation matrix

$$\mathbf{Q} = \mathbf{L}\mathbf{D}$$

where $\mathbf{D} = \left(diag\left(\mathbf{L}'\mathbf{L}\right)\right)^{-1/2}$ is the diagonal matrix that normalizes the columns of **L**.

At this stage, the rotated factors are

$$f_{pro\max\_temp} = \mathbf{Q}^{-1} f_{vari\max}.$$

Because

$$\text{var}\left(f_{pro\max\_temp}\right) = \left(\mathbf{Q}'\mathbf{Q}\right)^{-1},$$

and the diagonal elements do not equal 1, we must modify the rotated factor to

$$f_{pro\max} = \mathbf{C} f_{pro\max\_temp}$$

where

$$\mathbf{C} = \left\{ diag\left(\left(\mathbf{Q}^{'}\mathbf{Q} \right)^{-1} \right) \right\}^{-1/2}$$

The rotated factor pattern is

$$\Lambda_{pro\max} = \Lambda_{vari\max}\mathbf{Q}\mathbf{C}^{-1}$$

The correlation matrix of the factors is

$$\mathbf{R}_{ff} = \mathbf{C}\left(\mathbf{Q}^{'}\mathbf{Q}\right)^{-1}\mathbf{C}^{'}$$

The factor structure matrix is

$$\Lambda_S = \Lambda_{pro\max}\mathbf{R}_{ff}$$

Factor Score Coefficients (Harman, 1976)

Creates one new variable for each factor in the final solution. The following alternative methods for calculating the factor scores are available.

Regression

$$\mathbf{W} = \begin{cases} \Lambda_m \Gamma_m^{-1} & \text{PC without rotation} \\ \Lambda_m \left(\Lambda^{'}{}_m \Lambda_m\right)^{-1} & \text{PC with rotation} \\ \mathbf{R}^{-1}\mathbf{S}_m & \text{otherwise} \end{cases}$$

where

$$\begin{aligned} \mathbf{S}_m &= \text{factor structure matrix} \\ \mathbf{S}_m &= \Lambda_m \text{for orthogonal rotations} \end{aligned}$$

For PC without rotation if any $|\gamma_i| \le 10^{-8}$, factor score coefficients are not computed. For PC with rotation, if the determinant of $\Lambda^{'}{}_m \Lambda_m$ is less than 10^{-8}, the coefficients are not computed. Otherwise, if **R** is singular, factor score coefficients are not computed.

Bartlett

$$\mathbf{W} = \mathbf{J}^{-1}\Lambda^{'}\mathbf{U}^{-2}$$

where

$$\mathbf{J} = \Lambda^{'}\mathbf{U}^{-2}\Lambda$$
$$\mathbf{U}^2 = \mathbf{R} - \hat{\mathbf{R}}$$

Anderson Rubin

$$\mathbf{W} = \left(\Lambda^{'}\mathbf{U}^{-2}\mathbf{R}\mathbf{U}^{-2}\Lambda\right)^{-1/2}\Lambda^{'}\mathbf{U}^{-2}$$

where the symmetric square root of the parenthetical term is taken.

Optional Statistics (Dziubin and Shirkey, 1974)

- The anti-image covariance matrix $\mathbf{A} = (a_{ij})$ is given by

$$a_{ij} = \frac{r^{ij}}{r^{ii}r^{jj}}$$

- The chi-square value for Bartlett's test of sphericity is

$$\chi^2 = -\left(W - 1 - \frac{2p+5}{6}\right)\log|\mathbf{R}|$$

with $p(p-1)/2$ degrees of freedom.

- The Kaiser-Mayer-Olkin measure of sample adequacy is

$$KMO_j = \frac{\sum_{i\neq j} r_{ij}^2}{\sum_{i\neq j} r_{ij}^2 + \sum_{i\neq j} a_{ij}^{2*}} \quad KMO = \frac{\sum_{i\neq j} \Sigma r_{ij}^2}{\sum_{i\neq j} \Sigma r_{ij}^2 + \sum_{i\neq j} \Sigma a_{ij}^{2*}}$$

where a_{ij}^* is the anti-image correlation coefficient.

References

Dziuban, C. D., and E. C. Shirkey. 1974. When is a correlation matrix appropriate for factor analysis?. *Psychological Bulletin*, 81, 358–361.

Harman, H. H. 1976. *Modern Factor Analysis*, 3rd ed. Chicago: University of Chicago Press.

Hendrickson, A. E., and P. O. White. 1964. Promax: a quick method for rotation to oblique simple structure. *British Journal of Statistical Psychology*, 17, 65–70.

Jennrich, R. I., and P. F. Sampson. 1966. Rotation for simple loadings. *Psychometrika*, 31, 313–323.

Jöreskog, K. G. 1977. Factor analysis by least-square and maximum-likelihood method. In: *Statistical Methods for Digital Computers, volume 3,* K. Enslein, A. Ralston, and R. S. Wilf, eds. New York: John Wiley andSons.

Kaiser, H. F. 1963. Image analysis. In: *Problems in Measuring Change,* C. W. Harris, ed. Madison: Universityof Wisconsin Press.

Rummel, R. J. 1970. *Applied factor analysis*. Evanston: Ill.: Northwestern University Press.

FIT Algorithms

FIT displays a variety of descriptive statistics computed from the residual series as an aid in evaluating the goodness of fit of one or more models.

Notation

The following notation is used throughout this chapter unless otherwise stated:

DFH	Hypothesis degrees of freedom
DFE	Error degrees of freedom
$e_1, \ldots, e_n$	Residual (error) series
$X_1, \ldots, X_n$	Observed series
n	Number of cases

Statistics Computed in FIT

Mean Error (ME)

$$ME = \sum_{i=1}^{n} e_i / n$$

Mean Percent Error (MPE)

$$MPE = \frac{100}{n} \sum_{i=1}^{n} e_i / X_i$$

Mean Absolute Error (MAE)

$$MAE = \sum_{i=1}^{n} |e_i| / n$$

Mean Absolute Percent Error (MAPE)

$$MAPE = \frac{100}{n} \sum_{i=1}^{n} |e_i / X_i|$$

Sum of Square Error (SSE)

$$SSE = \sum_{i=1}^{n} e_i^2$$

Mean Square Error (MSE)

$$MSE = \begin{cases} SSE/n, & \text{if none of } DFE \text{ and } DFH \text{ is specified} \\ SSE/DFE, & \text{if } DFE \text{ is specified or } DFH \text{ is specified;} \\ & \text{then } DFE{=}n{-}DFH. \end{cases}$$

Root Mean Square Error (RMS)

$$RMS = \sqrt{MSE}$$

Durbin-Watson Statistics (DW)

$$DW = \frac{\sum_{i=1}^{n-1} (e_i - e_{i+1})^2}{\sum_{i=1}^{n} e_i^2}$$

FREQUENCIES Algorithms

If the absolute value of any observation is greater than 10^{13}, no calculations are done. For sorting of the observations, see *Sorting and Searching*. For information on percentiles for grouped data, see *Grouped Percentiles*.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 44-1
Notation

Notation	**Description**
X_k	Value of the variable for case *k*
w_k	Weight for case *k*
NV	Number of distinct values the variable assumes
N	Number of cases
W	Sum of weights of the cases

Basic Statistics

The values are sorted into ascending order and the following statistics are calculated.

Sum of Weights of Cases Having Each Value of X

$$f_j = \sum_{i=1}^{N} w_i k_i \quad j = 1, 2, \ldots, NV$$

where

$$k_i = \begin{cases} 1 & \text{if } X_i = X_j \\ 0 & \text{otherwise} \end{cases}$$

where X_j is the *j*th largest distinct value of *X*.

Relative Frequency (Percentage) for each Value of X

$$Rf_j = \left(\frac{f_j}{W'}\right) \times 100$$

where

$$W' = \sum_{i=1}^{NV} f_i \text{ (sum over all categories including those declared as missing values)}$$

Adjusted Frequency (Percentage)

$$Af_j = \left(\frac{f_j}{W}\right) \times 100$$

where

$$W = \sum_{i=1}^{NV} f_i k_i \text{ (sum over nonmissing categories)}$$

and

$$k_i = \begin{cases} 0 & \text{if } X_i \text{ has been declared missing} \\ 1 & \text{otherwise} \end{cases}$$

For all X_j declared missing, an adjusted frequency is not printed.

Cumulative Frequency (Percentage)

$$Cf_j = \sum_{i=1}^{j} f_i$$

Minimum

$$\min_k X_k$$

Maximum

$$\max_k X_k$$

Mode

Value of X_j which has the largest observed frequency. If several are tied, the smallest value is selected.

Range

Maximum – Minimum

The pth percentile

Find the first score interval (*x2*) containing more than *tp* cases.

$$p\text{th percentile} = \begin{cases} x_2 & \text{if } tp - cp_1 \geq 100/W \\ \{1 - [(W+1)p/10\underline{0} - cc_1]\}x_1 \\ \quad + [(W+1)p/100 \quad cc_1]x_2 & \text{if } tp - cp_1 < 100/W \end{cases}$$

where

$tp = (W+1)p/100$
$cp_1 < tp < cp_2$
x_1 and x_2 are the values corresponding to cp_1 and cp_2, respectively
cc_1 is the cumulative frequency up to x_1
$cp_1$is the cumulative percent up tox_1

Note: when *p*=50, this is the median.

Mean

$$\overline{X} = \frac{\sum_{j=1}^{NV} f_j X_j}{W}$$

Moments about the mean are calculated as:

$$M_j = \sum_{i=1}^{NV} f_i \left(X_i - \overline{X}\right)^j \quad j = 2,3,4$$

Variance

$$S^2 = \frac{M_2}{(W-1)}$$

Standard Deviation

$$S = \sqrt{S^2}$$

Standard Error of the Mean

$$SEM = \frac{S}{\sqrt{W}}$$

Skewness (Bliss, 1967, p. 144)

$$g_1 = \frac{WM_3}{(W-1)(W-2)S^3} \quad se(g_1) = \sqrt{\frac{6W(W-1)}{(W-2)(W+1)(W+3)}}$$

The skewness if computed only if W≥3 and Variance>0.

Kurtosis

$$g_2 = \frac{W(W+1)M_4 - 3(W-1)M_2^2}{(W-1)(W-2)(W-3)S^4} \quad se(g_2) = \sqrt{\frac{4(W^2-1)se(g_1)^2}{(W-3)(W+5)}}$$

The kurtosis is computed only if W≥4 and Variance>0.

References

Blalock, H. M. 1972. *Social statistics*. New York: McGraw-Hill.

Bliss, C. I. 1967. *Statistics in biology, Volume 1*. New York: McGraw-Hill.

Generalized linear mixed models algorithms

Generalized linear mixed models extend the linear model so that:

- The target is linearly related to the factors and covariates via a specified link function.
- The target can have a non-normal distribution.
- The observations can be correlated.

Generalized linear mixed models cover a wide variety of models, from simple linear regression to complex multilevel models for non-normal longitudinal data.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of complete cases in the dataset. It is an integer and $n \geq 1$.
p	Number of parameters (including the constant, if it exists) in the model. It is an integer and $p \geq 1$.
p_X	Number of non-redundant columns in the design matrix of fixed effects. It is an integer and $p_X \geq 1$.
K	Number of random effects.
$\mathbf{y}$	$n\times 1$ target vector. The rows are records.
$\mathbf{r}$	$n\times 1$ events vector for the binomial distribution representing the number of “successes” within a number of trials. All elements are non-negative integers.
$\mathbf{m}$	$n\times 1$ trials vector for the binomial distribution. All elements are positive integers and $m_i \geq r_i$, i=1,...,n.
$\boldsymbol{\mu}$	$n\times 1$ expected target value vector.
$\boldsymbol{\eta}$	$n\times 1$ linear predictor vector.
X	$n\times p$ design matrix. The rows represent the records and the columns represent the parameters. The ith row is $\mathrm{x}_i^T = (x_{i1}, \ldots, x_{ip})$, where the superscript T means transpose of a matrix or vector, $i = 1, \ldots, n$ with $x_{i1} = 1$ if the model has an intercept.
$\mathbf{Z}$	$n\times r$ design matrix of random effects.
$\mathbf{O}$	$n\times 1$ offset vector. This can’t be the target or one of the predictors. Also this can’t be a categorical field.
$\boldsymbol{\beta}$	$p\times 1$ parameter vector. The first element is the intercept, if there is one.
$\boldsymbol{\gamma}$	$r\times 1$ random effect vector.
$\boldsymbol{\omega}$	$n\times 1$ scale weight vector. If an element is less than or equal to 0 or missing, the corresponding record is not used.
$\mathbf{f}$	$n\times 1$ frequency weight vector. Non-integer elements are treated by rounding the value to the nearest integer. For values less than 0.5 or missing, the corresponding records are not used.
N	Effective sample size, $N = \sum_{i=1}^{n} f_i$.. If frequency weights are not used, $N = n$.
θ_k	covariance parameters of the kth random effect
θ_G	covariance parameters of the random effects, $\theta_G = \left[\theta_1^T, \ldots, \theta_K^T\right]^T$

θ_R covariance parameters of the residuals

$\boldsymbol{\theta}$ $\theta = \left[\theta_G^T, \theta_R^T\right]^T = \left[\theta_1^T, \ldots, \theta_K^T, \theta_R^T\right]^T$

$V_{Y|\gamma}$ Covariance matrix of **y**, conditional on the random effects

Model

The form of a generalized linear mixed model for the target **y** with the random effects **γ** is

$$\eta = g(E(y|\gamma)) = \mathbf{X}\beta + \mathbf{Z}\gamma + O, y|\gamma \sim F$$

where η is the linear predictor; *g*(.) is the monotonic differentiable link function; **γ** is a $(r \times 1)$ vector of random effects which are assumed to be normally distributed with mean 0 and variance matrix **G**, **X** is a $(n \times p)$ design matrix for the fixed effects; **Z** is a $(n \times r)$ design matrix for the random effects; **O** is an offset with a constant coefficient of 1 for each observation; *F* is the conditional target probability distribution. Note that if there are no random effects, the model reduces to a generalized linear model (GZLM).

The probability distributions without random effects offered (except multinomial) are listed in Table 45-1. The link functions offered are listed in Table 45-3. Different combinations of probability distribution and link function can result in different models.

See "Nominal multinomial distribution" for more information on the nominal multinomial distribution.

See "Ordinal multinomial distribution" for more information on the ordinal multinomial distribution.

Note that the available distributions depend on the measurement level of the target:

- A continuous target can have any distribution except multinomial. The binomial distribution is allowed because the target could be an "events" field. The default distribution for a continuous target is the normal distribution.
- A nominal target can have the multinomial or binomial distribution. The default is multinomial.
- An ordinal target can have the multinomial or binomial distribution. The default is multinomial.

Table 45-1
Distribution, range and variance of the response, variance function, and its first derivative

Distribution	**Range of *y***	***V*(μ)**	**Var(*y*)**	***V'*(μ)**
Normal	$(-\infty,\infty)$	1	ϕ	0
Inverse Gaussian	$(0,\infty)$	μ^3	$\phi\mu^3$	$3\mu^2$
Gamma	$(0,\infty)$	μ^2	$\phi\mu^2$	2μ
Negative binomial	$0(1)\infty$	$\mu+k\mu^2$	$\mu+k\mu^2$	$1+2k\mu$
Poisson	$0(1)\infty$	μ	μ	1
Binomial(*m*)	$0(1)m/m$	$\mu(1-\mu)$	$\mu(1-\mu)/m$	$1-2\mu$

Notes

- 0(1)z means the range is from 0 to z with increments of 1; that is, 0, 1, 2, …, z.
- For the binomial distribution, the binomial trial variable m is considered as a part of the weight variable ω.
- If a scale weight variable ω is presented, ϕ is replaced by ϕ/ω.
- For the negative binomial distribution, the ancillary parameter (k) is estimated by the maximum likelihood (ML) method. When k = 0, the negative binomial distribution reduces to the Poisson distribution. When k = 1, the negative binomial is the geometric distribution.

The full log-likelihood function (ℓ), which will be used as the objective function for parameter estimation, is listed for each distribution in the following table.

Table 45-2
The log-likelihood function for probability distribution

Distribution	ℓ
Normal	$\ell = \ell_k + \sum_{i=1}^{n} -\frac{f_i}{2}\{\ln(2\pi)\}$
Inverse Gaussian	$\ell = \ell_k + \sum_{i=1}^{n} -\frac{f_i}{2}\{\ln(2\pi)\}$
Gamma	$\ell = \ell_k + \sum_{i=1}^{n} f_i\{-\ln(y_i)\}$
Negative binomial	$\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{-\ln(\Gamma(y_i+1))\}$
Poisson	$\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{-\ln(y_i!)\}$
Binomial(m)	$\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\left\{\ln\begin{pmatrix} m_i \\ r_i \end{pmatrix}\right\}$,where $\begin{pmatrix} m_i \\ r_i \end{pmatrix} = \frac{m_i!}{r_i!(m_i-r_i)!}$

The following tables list the form, inverse form, range of $\hat{\mu}$, and first and second derivatives for each link function.

Table 45-3
Link function name, form, inverse of link function, and range of the predicted mean

Link function	**η=g(μ)**	**Inverse μ=g^{-1}(η)**	**Range of $\hat{\mu}$**
Identity	μ	η	$\hat{\mu} \in R$
Log	ln(μ)	exp(η)	$\hat{\mu} \geq 0$
Logit	$\ln\left(\frac{\mu}{1-\mu}\right)$	$\frac{\exp(\eta)}{1+\exp(\eta)}$	$\hat{\mu} \in [0,1]$
Probit	$\Phi^{-1}(\mu)$, where $\Phi(\xi) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\xi} e^{-z^2/2}dz$	Φ(η)	$\hat{\mu} \in [0,1]$
Complementary log-log	ln(−(ln(1−μ))	1−exp(−exp(η))	$\hat{\mu} \in [0,1]$

Link function	**η=*g*(μ)**	**Inverse μ=*g*⁻¹(η)**	**Range of $\hat{\mu}$**
Power(α)$\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\mu^{\alpha}\\ \ln(\mu)\end{cases}$	$\begin{cases}\eta^{1/\alpha}\\ \exp(\eta)\end{cases}$	$\begin{cases}\hat{\mu} \in R \text{ if } \alpha 1/\alpha \text{ is odd integer}\\ \hat{\mu} \geq 0 \text{ otherwise}\end{cases}$
Log-complement	$\ln(1-\mu)$	$1-\exp(\eta)$	$\hat{\mu} \leq 1$
Negative log-log	$-\ln(-\ln(\mu))$	$\exp(-\exp(-\eta))$	$\hat{\mu} \in [0,1]$

Note: In the power link function, if |α| < 2.2e-16, α is treated as 0.

Table 45-4
The first and second derivatives of link function

Link function	**First derivative** $g'(\mu) = \frac{\partial\eta}{\partial\mu} = \Delta$	**Second derivative** $g''(\mu) = \frac{\partial^2\eta}{\partial\mu^2}$
Identity	1	0
Log	$\frac{1}{\mu}$	$-\Delta^2$
Logit	$\frac{1}{\mu(1-\mu)}$	$\Delta^2(2\mu-1)$
Probit	$\frac{1}{\phi\left(\Phi^{-1}(\mu)\right)}$, where $\phi(z) = \frac{1}{\sqrt{2\pi}}e^{-z^2/2}$	$\Delta^2\Phi^{-1}(\mu)$
Complementary log-log	$\frac{1}{(\mu-1)\ln(1-\mu)}$	$-\Delta^2(1+\ln(1-\mu))$
Power(α)$\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\alpha\mu^{\alpha-1}\\ \frac{1}{\mu}\end{cases}$	$\begin{cases}\Delta\frac{\alpha-1}{\mu}\\ -\Delta^2\end{cases}$
Log-complement	$\frac{-1}{1-\mu}$	$-\Delta^2$
Negative log-log	$\frac{-1}{\mu\ln(\mu)}$	$\Delta^2(1+\ln(\mu))$

When the canonical parameter is equal to the linear predictor, $\theta = \eta$, then the link function is called the **canonical link function**. Although the canonical links lead to desirable statistical properties of the model, particularly in small samples, there is in general no a priori reason why the systematic effects in a model should be additive on the scale given by that link. The canonical link functions for probability distributions are given in the following table.

Table 45-5
Canonical and default link functions for probability distributions

Distribution	**Canonical link function**
Normal	Identity
Inverse Gaussian	Power(−2)
Gamma	Power(−1)
Negative binomial	Negative binomial
Poisson	Log
Binomial	Logit

The variance of **y**, conditional on the random effects, is

$$var(\mathrm{y}|\gamma) = \mathrm{A}^{1/2}\mathrm{R}\mathrm{A}^{1/2}$$

The matrix **A** is a diagonal matrix and contains the variance function of the model, which is the function of the mean **μ**, divided by the corresponding scale weight variable; that is, $A = \text{diag}(V(\mu_i)/\omega_i), i = 1, \ldots, n$. The variance functions, $V(\mu)$, are different for different distributions. The matrix **R** is the variance matrix for repeated measures.

Generalized linear mixed models allow correlation and/or heterogeneity from random effects (**G**-side) and/or heterogeneity from residual effects (**R**-side). resulting in 4 types of models:

1. If a GLMM has no **G**-side or **R**-side effects, then it reduces to a GZLM; **G**=**0** and $R = \phi I$, where **I** is the identity matrix and ϕ is the scale parameter. For continuous distributions (normal, inverse Gauss and gamma), ϕ is an unknown parameter and is estimated jointly with the regression parameters by the maximum likelihood (ML) method. For discrete distributions (negative binomial, Poisson, binomial and multinomial), ϕ is estimated by Pearson chi-square as follows:

$$\hat{\phi} = \frac{1}{N^*}\sum_{i=1}^{n} f_i\omega_i \frac{(y_i - \mu_i)^2}{V(\mu_i)},$$

where $N^* = N - p_x$ for the restricted maximum pseudo-likelihood (REPL) method.

2. If a model only has **G**-side random effects, then the **G** matrix is user-specified and $R = \phi I$. ϕ is estimated jointly with the covariance parameters in **G** for continuous distributions and $\phi = 1$ for discrete distributions..

3. If a model only has **R**-side residual effects, then **G** = **0** and the **R** matrix is user-specified. All covariance parameters in **R** are estimated using the REPL method, defined in "Estimation".

4. If a model has both **G**-side and **R**-side effects, all covariance parameters in **G** and **R** are jointly estimated using the REPL method.

For the negative binomial distribution, there is the ancillary parameter *k*, which is first estimated by the ML method, ignoring random and residual effects, then fixed to that estimate while other regression and covariance parameters are estimated.

Fixed effects transformation

To improve numerical stability, the **X** matrix is transformed according to the following rules.

The *i*th row of **X** is $x_i = (x_{i1}, \ldots, x_{ip})^T$, *i*=1,...,*n* with $x_{i1} = 1$ if the model has an intercept. Suppose x_i^* is the transformation of x_i then the *j*th entry of x_i^* is defined as

$$x_{ij}^* = \frac{x_{ij} - c_j}{s_j}$$

where c_j and s_j are centering and scaling values for x_{ij}, respectively, for j=1,...,p and choices of c_j and s_j , are listed as follows:

a. For a non-constant continuous predictor or a derived predictor which includes a continuous predictor, if the model has an intercept, $c_1 = 0$ and $c_j = \overline{x}_j, j \neq 1$, where $\overline{x}_j$ is the sample mean of the jth predictor, $\overline{x}_j = \frac{1}{N}\sum_{i=1}^{n} f_i x_{ij}$ and $s_1 = 1$ and $s_j = \sqrt{s_{x_j}^2}, j \neq 1$, where $\sqrt{s_{x_j}^2}$ is the sample standard deviation of the jth predictor and $s_{x_j}^2 = \frac{1}{N-1}\sum_{i=1}^{n} f_i (x_{ij} - \overline{x}_j)^2$Note that the intercept column is not transformed. If the model has no intercept, $c_j = 0$ and $s_j = \sqrt{s_{x_j}^2 + \overline{x}_j^2}$.

b. For a constant predictor $x_{ij} = a \neq 0, \forall i$, $c_j = 0$ and $s_j = a$, that is, scale it to 1.

c. For a dummy predictor that is derived from a factor or a factor interaction, $c_j = 0$and$s_j = 1$; that is, leave it unchanged.

Estimation

We estimate GLMMs using linearization-based methods, also called the pseudo likelihood approach (PL; Wolfinger and O'Connell (1994)), penalized quasi-likelihood (PQL; Breslow and Clayton (1993)), marginal quasi-likelihood (MQL; Goldstein (1991)). They are based on the similar principle that the GLMMs are approximated by an LMM so that well-established estimation methods for LMMs can be applied. More specifically, the mean target function; that is, the inverse link function is approximated by a linear Taylor series expansion around the current estimates of the fixed-effect regression coefficients and different solutions of random effects (0 is used for MQL and the empirical Bayes estimates are used for PQL). Applying this linear approximation of the mean target leads to a linear mixed model for a transformation of the original target. The parameters of this LMM can be estimated by Newton-Raphson or Fisher scoring technique and the estimates then are used to update the linear approximation. The algorithm iterates between two steps until convergence. In general, the method is a doubly iterative process. The outer iterations are to update the transformed target for an LMM and the inner iterations are to estimate parameters of the LMM.

It is well known that parameter estimation for an LMM can be based on maximum likelihood (ML) or restricted (or residual) maximum likelihood (REML). Similarly, parameter estimation for a GLMM in the inner iterations can based on maximum pseudo-likelihood (PL) or restricted maximum pseudo-likelihood (REPL).

Linear mixed pseudo model

Following Wolfinger and O'Connell (1993), a first-order Taylor series of **μ** in (1) about $\tilde{\beta}$ and $\tilde{\gamma}$yields

$$\mu \approx \tilde{\mu} + \left(g^{-1}\right)'\left(\mathrm{X}\tilde{\beta} + \mathrm{Z}\tilde{\gamma} + \mathrm{O}\right)\left[\mathrm{X}\left(\beta - \tilde{\beta}\right) + \mathrm{Z}(\gamma - \tilde{\gamma})\right]$$

where $(g^{-1})'\left(\mathbf{X}\tilde{\beta}+\mathrm{Z}\tilde{\gamma}+\mathrm{O}\right)$ is a diagonal matrix with elements consisting of evaluations of the 1st derivative of g^{-1}. Since $(g^{-1})'\left(\mathbf{X}\tilde{\beta}+\mathrm{Z}\tilde{\gamma}+\mathrm{O}\right)=\left(g'(\tilde{\mu})\right)^{-1}$, this equation can be rearranged as

$$g'(\tilde{\mu})(\mu-\tilde{\mu})+\mathbf{X}\tilde{\beta}+\mathrm{Z}\tilde{\gamma}\approx\mathbf{X}\beta+\mathrm{Z}\gamma$$

If we define a pseudo target variable as

$$\mathrm{v}\equiv g'(\tilde{\mu})(\mathrm{y}-\tilde{\mu})+\mathbf{X}\tilde{\beta}+\mathrm{Z}\tilde{\gamma}=g'(\tilde{\mu})(\mathrm{y}-\tilde{\mu})+g(\tilde{\mu})-\mathrm{O},$$

then the conditional expectation and variance of v, based on $\mathrm{E}(\mathrm{y}|\gamma)$ and $var(\mathrm{y}|\gamma)=\mathrm{A}^{1/2}\mathrm{R}\mathrm{A}^{1/2}$, are

$$\mathrm{E}(\mathrm{v}|\gamma)=g'(\tilde{\mu})(\mu-\tilde{\mu})+\mathbf{X}\tilde{\beta}+\mathrm{Z}\tilde{\gamma}$$

$$var(\mathrm{v}|\gamma)=g'(\tilde{\mu})\mathrm{A}_{\tilde{\mu}}^{1/2}\mathrm{R}\mathrm{A}_{\tilde{\mu}}^{1/2}g'(\tilde{\mu})$$

where $\mathrm{A}_{\tilde{\mu}}^{1/2}=\mathrm{diag}\left[\left(V(\tilde{\mu}_i)/\omega_i\right)^{1/2}\right], i=1,\ldots,n.$

Furthermore, we also assume $\mathrm{v}|\gamma$ is normally distributed. Then we consider the model of v

$$\mathrm{v}=\mathbf{X}\beta+\mathrm{Z}\gamma+\varepsilon$$

as a weighted linear mixed model with fixed effects β, random effects $\gamma\sim N(0,\mathrm{G})$, error terms $\varepsilon\sim N\left(0,g'(\tilde{\mu})\mathrm{A}_{\tilde{\mu}}^{1/2}\mathrm{R}\mathrm{A}_{\tilde{\mu}}^{1/2}g'(\tilde{\mu})\right)$, because $var(\varepsilon)=var(\mathrm{v}|\gamma)$, and diagonal weight matrix $\tilde{W}=\mathrm{A}_{\tilde{\mu}}\left[g'(\tilde{\mu})\right]^{-2}$. Note that the new target v (with **O** if an offset variable exists) is a Taylor series approximation of the linked target $g(\mathrm{y})$. The estimation method of unknown parameters of β and θ, which contains all unknowns in **G** and **R**, for traditional linear mixed models can be applied to this linear mixed pseudo model.

The Gaussian log pseudo-likelihood (PL) and restricted log pseudo-likelihood (REPL), which are expressed as the functions of covariance parameters in θ, corresponding to the linear mixed model for v are the following:

$$\ell(\theta;\mathrm{v})=-\frac{1}{2}\ln|\mathrm{V}(\theta)|-\frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta)-\frac{N}{2}\ln(2\pi)$$

$$\ell_R(\theta;\mathrm{v})=-\frac{1}{2}\ln|\mathrm{V}(\theta)|-\frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta)-\frac{1}{2}\ln\left|\mathrm{X}^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{X}\right|-\frac{N-p_x}{2}\ln(2\pi)$$

where
$\mathrm{V}(\theta)=\mathrm{Z}\mathrm{G}(\theta)\mathrm{Z}+\tilde{\mathbf{W}}^{-1/2}\mathrm{R}(\theta)\tilde{\mathbf{W}}^{-1/2}$, $\mathrm{r}(\theta)=\mathrm{v}-\mathrm{X}\left(\mathrm{X}^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{X}\right)^{-}\mathrm{X}^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{v}=\mathrm{v}-\hat{\beta}$, N denotes the effective sample size, and p_{X} denotes the rank of the design matrix of **X** or the number of non-redundant parameters in **X**. Note that the regression parameters in β are profiled from the above equations because the estimation of β can be obtained analytically. The covariance

parameters in θ are estimated by Newton-Raphson or Fisher scoring algorithm. Following the tradition in linear mixed models, the objection functions of minimization for estimating θ would be $-2\ell(\theta;\mathrm{v})$ or $-2\ell_R(\theta;\mathrm{v})$. Upon obtaining $\hat{\theta}$, estimates for β and γ are computed as

$$\hat{\beta} = \left(\mathrm{X}^{\mathrm{T}}\mathrm{V}\left(\hat{\theta}\right)^{-1}\mathrm{X}\right)^{-}\mathrm{X}^{\mathrm{T}}\mathrm{V}\left(\hat{\theta}\right)^{-1}\mathrm{v}$$

$$\hat{\gamma} = \hat{G}\mathrm{Z}^{\mathrm{T}}\mathrm{V}\left(\hat{\theta}\right)^{-1}\hat{r}$$

where $\hat{\beta}$ is the best linear unbiased estimator (BLUE) of β and $\hat{\gamma}$ is the estimated best linear unbiased predictor (BLUP) of γ in the linear mixed pseudo model. With these statistics, v and $\tilde{W}$ are recomputed based on $\tilde{\mu}$ and the objective function is minimized again to obtain updated $\hat{\theta}$. Iteration between $-2\ell(\theta;\mathrm{v})$ and the above equation yields the PL estimation procedure and between $-2\ell_R(\theta;\mathrm{v})$ and the above equation the REPL procedure.

There are two choices for $\tilde{\gamma}$ (the current estimates of γ):

1. $\hat{\gamma}$ for PQL; and
2. 0 for MQL.

On the other hand, $\hat{\beta}$ is always used as the current estimate of the fixed effects. Based on the two objective functions (PL or REPL) and two choices of random effect estimates (PQL or MQL), 4 estimation methods can be implemented for GLMMs:

1. PL-PQL: pseudo-likelihood with $\tilde{\gamma}=\hat{\gamma}$;
2. PL-MQL: pseudo-likelihood with $\tilde{\gamma}=0$;
3. REPL-PQL: residual pseudo-likelihood with $\tilde{\gamma}=\hat{\gamma}$;
4. REPL-MQL: residual pseudo-likelihood with $\tilde{\gamma}=0$.

We use method 3, REPL-PQL.

Iterative process

The doubly iterative process for the estimation of θ is as follows:

1. Obtain an initial estimate of $\mu, \mu^{(0)}$. Specifically, $\mu_i^0 = (y_i m_i + 0.5)/(m_i + 1)$ for a binomial distribution (y_i can be a proportion or 0/1 value) and $\mu_i^0 = y_i$ for a non-binomial distribution. Also set the outer iteration index $j = 0$.

2. Based on $\tilde{\mu}$, compute

   $\mathrm{v} = g(\tilde{\mu}) - \mathrm{O} + g'(\tilde{\mu})(\mathrm{y} - \tilde{\mu})$ and $\tilde{W} = \mathrm{A}_{\tilde{\mu}}^{-1}\left[g'(\tilde{\mu})\right]^{-2}$.

   Fit a weighted linear mixed model with pseudo target v, fixed effects design matrix **X**, random effects design matrix **Z**, and diagonal weight matrix $\tilde{W}$. The fitting procedure, which is called the inner iteration, yields the estimates of θ, and is denoted as $\theta^{(j)}$. The procedure uses the specified settings for parameter, log-likelihood, and Hessian convergence criteria for determining convergence of the linear mixed model. If *j* = 0, go to step 4; otherwise go to the next step. See “MIXED Algorithms” for more information on fitting the linear mixed model.

3. Check if the following criterion with tolerance level ξ is satisfied:

$$\max_i \left(2 \times \frac{\left|\theta_i^{(j)} - \theta_i^{(j-1)}\right|}{\left|\theta_i^{(j-1)}\right| + \left|\theta_i^{(j-1)}\right|} \right) < \xi.$$

   If it is met or maximum number of outer iterations is reached, stop. Otherwise, go to the next step.

4. Compute $\hat{\beta}$ by setting $\hat{\theta} = \theta^{(j)}$ then set $\tilde{\beta} = \hat{\beta}$. Depending on the choice of random effect estimates, set $\tilde{\gamma}=\hat{\gamma}$.

5. Compute the new estimate of μ by

$$\tilde{\mu} = g^{-1}\left(\mathbf{X}\tilde{\beta} + \mathrm{Z}\tilde{\gamma} + \mathrm{O}\right),$$

   set *j* = *j* + 1 and go to step 2.

Wald confidence intervals for covariance parameter estimates

Here we assume that the estimated parameters of **G** and **R** are obtained through the above doubly iterative process. Then their asymptotic covariance matrix can be approximated by $2\mathrm{H}^{-1}$, where **H** is the Hessian matrix of the objective function $-2\ell(\theta; \mathrm{y})$ or $-2\ell_R(\theta; \mathrm{v})$) evaluated at $\hat{\theta}$. The standard error for the *i*th covariance parameter estimate in the $\hat{\theta}$ vector, say $\hat{\theta}_i$, is the square root of the *i*th diagonal element of $2\mathrm{H}^{-1}$.

Thus, a simple Wald’s type confidence interval or test statistic for any covariance parameter can be obtained by using the asymptotic normality. However, these can be unreliable in small samples, especially for variance and correlation parameters that have a range of $[0, \infty)$ and $[-1, 1]$ respectively. Therefore, following the same method used in linear mixed models, these parameters are transformed to parameters that have range $(-\infty, \infty)$. Using the delta method, these transformed estimates still have asymptotic normal distributions.

For variance type parameters in **G** and **R**, such as σ^2 in the autoregressive, autoregressive moving average, compound symmetry, diagonal, Toeplitz, and variance components, and θ_{ii} in the unstructured type, the 100(1 – α)% Wald confidence interval is given, assuming the variance parameter estimate is $\hat{\sigma}^2$ and its standard error is $\mathrm{se}\left(\hat{\sigma}^2\right)$ from the corresponding diagonal element of $2\mathrm{H}^{-1}$, by

$$\exp\left(\ln\left(\hat{\sigma}^2\right) \pm z_{1-\alpha/2} \cdot \hat{\sigma}^{-2} \cdot \mathrm{se}\left(\hat{\sigma}^2\right)\right)$$

For correlation type parameters in **G** and **R**, such as ρ in the autoregressive, autoregressive moving average, and Toeplitz types and φ in the autoregressive moving average type, which usually come with the constraint of $|\rho| \le 1$, the 100(1 – α)% Wald confidence interval is given, assuming the correlation parameter estimate is $\hat{\rho}$ and its standard error is $(\hat{\rho})$ from the corresponding diagonal element of $2\mathrm{H}^{-1}$, by

$$\tanh\left(\tanh^{-1}(\hat{\rho}) \pm z_{1-\alpha/2} \cdot \left(1 - \hat{\rho}^2\right)^{-1} \cdot \mathrm{se}(\hat{\rho})\right)$$

where $\tanh x = \frac{\exp(x)-\exp(-x)}{\exp(x)+\exp(-x)}$ and $\tanh^{-1} x = \frac{1}{2}\ln\left[\frac{1+x}{1-x}\right]$ are hyperbolic tangent and inverse hyperbolic tangent, respectively.

For general type parameters, other than variance and correlation types, in **G** and **R**, such as σ_1 in the compound symmetry type and $\theta_{ij}, i \neq j$, (off-diagonal elements) in the unstructured type, no transformation is done. Then the 100(1 – α)% Wald confidence interval is simply, assuming the parameter estimate is $\hat{\sigma}_1$ and its standard error is $\text{se}(\hat{\sigma}_1)$ from the corresponding diagonal element of 2H^{-1},

$$\left(\hat{\sigma}_1 - z_{1-\alpha/2} \cdot \text{se}(\hat{\sigma}_1)\right), \hat{\sigma}_1 + z_{1-\alpha/2} \cdot \text{se}(\hat{\sigma}_1)\big)$$

The 100(1 – α)% Wald confidence interval for ϕ is

$$\left(\exp\left(\hat{\tau} - z_{1-\alpha/2}\hat{\sigma}_\tau\right), \exp\left(\hat{\tau} + z_{1-\alpha/2}\hat{\sigma}_\tau\right)\right)$$

where $\tau = \ln(\phi)$.

Note that the *z*-statistics for the hypothesis $H_{0i} : \theta_i = 0$, where θ_i is a covariance parameter in θ vector, are calculated; however, the Wald tests should be considered as an approximation and used with caution because the test statistics might not have a standardized normal distribution.

Statistics for estimates of fixed and random effects

The approximate covariance matrix of $(\hat{\beta} - \beta, \hat{\gamma} - \gamma)$ is

$$\hat{C} = \begin{bmatrix} \text{X}^\text{T}\text{R}^{*-1}\text{X} & \text{X}^\text{T}\text{R}^{*-1}\text{Z} \\ \text{Z}^\text{T}\text{R}^{*-1}\text{X} & \text{Z}^\text{T}\text{R}^{*-1}\text{Z} + \text{G}\left(\hat{\theta}\right)^{-1} \end{bmatrix}^{-} = \begin{bmatrix} \text{C}_{11} & \text{C}_{21}^\text{T} \\ \text{C}_{21} & \text{C}_{22} \end{bmatrix}$$

where $\text{R}^* \equiv \hat{var}(\text{v}|\gamma) = g'(\hat{\mu})\text{A}_{\hat{\mu}}^{1/2}\text{R}\text{A}_{\hat{\mu}}^{1/2}g'(\hat{\mu})$ is evaluated at the converged estimates and

$$\hat{C}_{11} = \left(\mathbf{X}^\text{T}\hat{V}^{-1}\mathbf{X}\right)^{-}$$

$$\hat{C}_{21} = -\hat{G}\text{Z}^\text{T}\hat{V}^{-}\mathbf{X}\hat{C}_{11}$$

$$\hat{C}_{22} = \left(\text{Z}^\text{T}\hat{R}^{-1}\text{Z} + \hat{G}^{-1}\right)^{-1} - \hat{C}_{21}\mathbf{X}^\text{T}\hat{V}^{-1}\text{Z}\hat{G}$$

Statistics for estimates of fixed effects on original scale

If the **X** matrix is transformed, the restricted log pseudo-likelihood (REPL) would be different based on transformed and original scale, so the REPL on the transformed scale should be transformed back on the final iteration so that any post-estimation statistics based on REPL can be calculated correctly. Suppose the final objective function value based on the transformed and original scales are $-2\ell_R^*(\theta; v)$ and $-2\ell_R(\theta; v)$, respectively, then $-2\ell_R(\theta; v)$ can be obtained from $-2\ell_R^*(\theta; v)$ as follows:

$$-2\ell_R(\theta; v) = -2\ell_R^*(\theta; v) - 2\ln|A|$$

Because REPL has the following extra term involved the X matrix

$$\begin{aligned}
-\tfrac{1}{2}\ln\left|X^{*T}V(\theta)^{-1}X^*\right| &= -\tfrac{1}{2}\ln\left|(XA)^{T}V(\theta)^{-1}XA\right| \\
&= -\tfrac{1}{2}\ln\left(\left|A^{T}\right| \times \left|XV(\theta)^{-1}X\right| \times |A|\right) \\
&= -\tfrac{1}{2}\left(\ln\left|XV(\theta)^{-1}X\right| + \ln|A| + \ln\left|A^{T}\right|\right) \\
&= -\tfrac{1}{2}\ln\left|XV(\theta)^{-1}X\right| - \ln|A|
\end{aligned}$$

then $-\frac{1}{2}\ln\left|XV(\theta)^{-1}X\right| = -\frac{1}{2}\ln\left|X^{*T}V(\theta)^{-1}X^*\right| + \ln|A|$ and $\ell_R(\theta; v) = \ell_R^*(\theta; v) + \ln|A|$. Please note that PL values are the same whether the X matrix is transformed or not.

In addition, the final estimates of **β**, $\mathbf{C}_{11}$, $\mathbf{C}_{21}$ and $\mathbf{C}_{22}$ are based on the transformed scale, denoted as $\hat{\beta}^*$,$\hat{C}_{11}^*$,$\hat{C}_{21}^*$ and $\hat{C}_{22}^*$, respectively. They are transformed back to the original scale, denoted as $\hat{\beta}$,$\hat{C}_{11}$,$\hat{C}_{21}$ and $\hat{C}_{22}$, respectively, as follows:

$$\hat{\beta} = A\hat{\beta}^*,$$

$$\hat{C}_{11} = A\hat{C}_{11}^*A^{T},$$

$$\hat{C}_{21} = \hat{C}_{21}^*A^{T},$$

$$\hat{C}_{22} = \hat{C}_{22}^*.$$

Note that **A** could reduce to S^{-1}; hereafter, the superscript * denotes a quantity on the transformed scale.

Estimated covariance matrix of the fixed effects parameters

Two estimated covariance matrices of the fixed effects parameters can be calculated: model-based and robust.

The model-based estimated covariance matrix of the fixed effects parameters is given by

$$\Sigma_m = \hat{C}_{11}$$

The robust estimated covariance matrix of the fixed effects parameters for a GLMM is defined as the classical sandwich estimator. It is similar to that for a generalized linear model or a generalized estimating equation (GEE). If the model is a generalized linear mixed model and it is processed by subjects, then the robust estimator is defined as follows

$$\Sigma_r = \Sigma_m \left(\sum_{j=1}^{S} X_j^T \hat{V}_j^{-1} \hat{r}_j \hat{r}_j^T \hat{V}_j^{-1} X_j \right) \Sigma_m$$

where $\hat{r}_j = v_j - X_j \hat{\beta}$.

Standard errors for estimates in fixed effects and predictions in random effects

Let $\hat{\beta}_i$ denote a non-redundant parameter estimate in fixed effects. Its standard error is the square root of the *i*th diagonal element of Σ_m or Σ_r,

$$\hat{\sigma}_{\beta_i} = \sqrt{\sigma_{ii}}$$

The standard error for redundant parameter estimates is set to a system missing value.

Let $\hat{\gamma}_i$ denote a prediction in random effects. Its standard error is the square root of the *i*th diagonal element of $\hat{C}_{22}$:

$$\hat{\sigma}_{\gamma_i} = \sqrt{\hat{C}_{22,ii}}$$

Test statistics for estimates in fixed effects and predictions in random effects

The hypothesis $H_{0i} : \beta_i = 0$ is tested for each non-redundant parameter in fixed effects using the *t* statistic:

$$t_i = \frac{\hat{\beta}_i}{\hat{\sigma}_{\beta_i}}$$

which has an asymptotic *t* distribution with υ degrees of freedom. See “Method for computing degrees of freedom” for details on computing the degrees of freedom.

Wald confidence intervals for estimates in fixed effects and predictions in random effects

The 100(1 – α)% Wald confidence interval for β_i is given by

$$\left(\hat{\beta}_i - t_{\upsilon,\alpha/2} \hat{\sigma}_{\beta_i}, \hat{\beta}_i + t_{\upsilon,\alpha/2} \hat{\sigma}_{\beta_i} \right)$$

where $t_{\upsilon,\alpha/2}$ is the $(1 - \alpha/2)$ 100th percentile of the t_υ distribution.

For some models (see the list below), the exponentiated parameter estimates, their standard errors, and confidence intervals are computed. Using the delta method, the estimate of $\exp(\beta_i)$ is $\exp\left(\hat{\beta}_i\right)$, the standard error estimate is $\left(\exp\left(\hat{\beta}_i\right) \cdot \hat{\sigma}_{\beta_i}\right)$ and the corresponding $100(1 - \alpha)$% Wald confidence interval for exp (β_i) is

$$\left(\exp\left(\hat{\beta}_i - t_{\upsilon,\alpha/2} \hat{\sigma}_{\beta_i}\right), \exp\left(\hat{\beta}_i + t_{\upsilon,\alpha/2} \hat{\sigma}_{\beta_i}\right) \right).$$

The list of models is as follows:

1. Logistic regression (binomial distribution + logit link).
2. Nominal logistic regression (nominal multinomial distribution + generalized logit link).
3. Ordinal logistic regression (ordinal multinomial distribution + cumulative logit link).
4. Log-linear model (Poisson distribution + log link).
5. Negative binomial regression (negative binomial distribution + log link).

Testing

After estimating parameters and calculating relevant statistics, several tests for the given model are performed.

Goodness of fit

Information criteria

Information criteria are used when comparing different models for the same data. The formulas for various criteria are as follows.

Finite sample corrected (AICC)	$-2\ell+\frac{2d\cdot N}{(N-d-1)}$
Bayesian information criteria (BIC)	$-2\ell+d\ln(N)$

where ℓ is the restricted log-pseudo-likelihood evaluated at the parameter estimates. For REPL, *N* is the effective sample size minus the number of non-redundant parameters in fixed effects ($\sum_{i=1}^{n} f_i - p_x$) and *d* is the number of covariance parameters.

Note that the restricted log-pseudo-likelihood values are of the linearized model, not on the original scale. Thus the information criteria should not be compared across models with different distribution and link function and they should be interpreted with caution.

Tests of fixed effects

For each effect specified in the model, a type III test matrix **L** is constructed and H_0: $\mathbf{L}_i\boldsymbol{\beta} = \mathbf{0}$ is tested. Construction of **L** and the generating estimable function (GEF) is based on the generating matrix $\mathrm{H}_\omega=\left(\mathrm{X}^\mathrm{T}\Psi\mathrm{X}\right)^{-}\mathrm{X}^\mathrm{T}\Psi\mathrm{X}$, where $\Psi=\mathrm{diag}(f_1\omega_1,\ldots f_n\omega_n)$, such that $\mathbf{L}_i\boldsymbol{\beta}$ is estimable; that is, $\mathrm{L}_i=\mathrm{L}_i\mathrm{H}_\omega$. It involves parameters only for the given effect and the effects containing the given effect. For type III analysis, **L** does not depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable.

Then the **L** matrix is then used to construct the test statistic

$$\mathbf{F} = \frac{\hat{\beta}^{\mathrm{T}}\mathbf{L}^{\mathrm{T}}\left(\mathbf{L}\sum\mathbf{L}^{\mathrm{T}}\right)^{-1}\mathbf{L}\hat{\beta}}{r_c}$$

where $r_c = rank\left(\mathbf{L}\sum\mathbf{L}^{\mathrm{T}}\right)$. The statistic has an approximate *F* distribution. The numerator degrees of freedom is r_c and the denominator degrees of freedom is υ. See "Method for computing degrees of freedom" for details on computing the denominator degrees of freedom.

In addition, we test a null hypothesis that all regression parameters (except intercept if there is one) equal zero. The test statistic would be the same as the above F statistic except the L matrix is from GEF. If there is no intercept, the L matrix is the whole GEF. If there is an intercept, the L matrix is GEF without the first row which corresponds to the intercept. This test is similar to the "corrected model" in linear models.

Estimated marginal means

There are two types of estimated marginal means calculated here. One corresponds to the specified factors for the linear predictor of the model and the other corresponds to those for the original scale of the target.

Estimated marginal means are based on the estimated cell means. For a given fixed set of factors, or their interactions, we estimate marginal means as the mean value averaged over all cells generated by the rest of the factors in the model. Covariates may be fixed at any specified value. If not specified, the value for each covariate is set to its overall mean estimate.

Estimated marginal means are not available for the multinomial distribution.

Estimated marginal means for the linear predictor

Calculating estimated marginal means for the linear predictor

Estimated marginal means for the linear predictor are based on the link function transformation, and constructed such that **LB** is estimable.

Suppose there are *r* combined levels of the specified categorical effect. This *r*×1 vector can be expressed in the form $\hat{\mathbf{u}} = \mathbf{L}\hat{\beta}$. The variance matrix of $\hat{\mathbf{u}}$ is then computed by

$$\mathrm{V}(\hat{\mathbf{u}}) = \mathrm{L}\Sigma\mathrm{L}^{\mathrm{T}}$$

The standard error for the *j*th element of $\hat{\mathbf{u}}$ is the square root of the *j*th diagonal element of $(\hat{\mathbf{u}})$. Let the *j*th element of $\hat{\mathbf{u}}$ and its standard error be $\hat{u}_j$ and $\hat{\sigma}_{u_j}$, respectively, then the corresponding 100(1 – α)% confidence interval for $u_j, j = 1, \ldots, r$; is given by

$$\hat{u}_j \pm t_{\upsilon^j,\alpha/2}\hat{\sigma}_{u_j}$$

where $t_{v^j,\alpha/2}$ is the $(1-\alpha/2)100th$ percentile of the *t* distribution with v^j degrees of freedom. See "Method for computing degrees of freedom" for details on computing the degrees of freedom.

Comparing estimated marginal means for the linear predictor

We can compare estimated marginal means for the linear predictor based on a selected contrast type, for which a set of contrasts for the factor is created. Let this set of contrasts define matrix **C** used for testing the hypothesis $H_0 : \mathrm{C}u = 0$. An *F* statistic is used for testing given set of contrasts for the factor as follows:

$$F = \frac{(\mathrm{C}\hat{u})^{\mathrm{T}}\left(\mathrm{C}\mathrm{V}(\hat{u})\mathrm{C}^{\mathrm{T}}\right)^{-}(\mathrm{C}\hat{u})}{r_I}$$

which has an asymptotic *F* distribution with r_I degrees of freedom, where $r_I = \mathrm{rank}\left(\mathrm{C}\mathrm{V}(\hat{\mathbf{u}})\mathrm{C}^{\mathrm{T}}\right)$. See "Method for computing degrees of freedom" for details on computing the denominator degrees of freedom. The *p*-values can be calculated accordingly. Note that adjusted *p*-values based on multiple comparisons adjustments won't be computed for the overall test.

Each row $\mathrm{c}_i^{\mathrm{T}}$ of matrix **C** is also tested separately. The estimate for the *i*th row is given by $\mathrm{c}_i^{\mathrm{T}}\hat{\mathbf{u}}$ and its standard error by $\sqrt{\mathrm{c}_i^{\mathrm{T}}\mathrm{V}(\hat{\mathbf{u}})\mathrm{c}_i}$. The corresponding 100(1 – α)% confidence interval is given by

$$\mathrm{c}_i^{\mathrm{T}}\hat{u} \pm t_{v^i,\alpha/2}\hat{\sigma}_{cu_i}$$

The test statistic for $H_0 : \mathrm{c}_i^{\mathrm{T}}\mathbf{u} = 0$ is

$$t_i = \frac{\mathrm{c}_i^{\mathrm{T}}\hat{u}}{\hat{\sigma}_{cu_i}}$$

It has an asymptotic *t* distribution. See "Method for computing degrees of freedom" for details on computing the degrees of freedom. The *p*-values can be calculated accordingly. In addition, adjusted *p*-values for multiple comparisons can also computed.

Estimated marginal means in the original scale

Estimated marginal means for the target are based on the original scale. As a conditional predictor defined by Lane and Nelder (1982), estimated marginal means for the target are derived from those for the linear predictor.

Calculating estimated marginal means for the target

The estimated marginal means for the target are defined as

$$\hat{\mathbf{M}} = g^{-1}\left(\mathrm{L}\hat{\beta}\right) = g^{-1}(\hat{\mathbf{u}})$$

The variance of estimated marginal means for the target is

$$V\left(\hat{\mathbf{M}}\right) = diag\left(\frac{\partial g^{-1}(\hat{v}_j)}{\partial \hat{u}_j}\right) L\Sigma L^T diag\left(\frac{\partial g^{-1}(\hat{u}_j)}{\partial \hat{v}_j}\right)$$

where $diag(\partial g^{-1}(\hat{u}_j)/\partial \hat{u}_j)$ is a *r*×*r* matrix and $\partial g^{-1}(\hat{u}_j)/\partial \hat{u}_j$ is the derivative of the inverse of the link with respect to the *j*th value in $\hat{\mathbf{u}}$ and $\partial g^{-1}(\hat{u}_j)/\partial \hat{u}_j = 1/g'\left(\hat{M}_j\right)$ where $g'\left(\hat{M}_j\right)$ is from Table 45-4.

The 100(1 – α)% confidence interval for $M_i, i = 1, \ldots, r$ is given by

$$g^{-1}\left(\hat{u}_i \pm t_{v^i,\alpha/2}\hat{\sigma}_{u_i}\right).$$

Note: $\hat{\mathbf{M}}$ is estimated marginal means for the proportion, not for the number of events when events and trials variables are used for the binomial distribution.

Comparing estimated marginal means for the target

This is similar to comparing estimated marginal means for the linear predictor; just replace $\hat{\mathbf{u}}$ with $\hat{\mathbf{M}}$ and $V(\hat{\mathbf{u}})$ with $V\left(\hat{\mathbf{M}}\right)$. For more information, see the topic “Estimated marginal means for the linear predictor”.

Multiple comparisons

The hypothesis $H_0 : \mathbf{Cu} = \mathbf{0}$ can be tested using the multiple row hypotheses testing technique. Let $\mathbf{c}_i^T$ be the *i*th row vector of matrix **C**. The *i*th row hypothesis is $H_{0i} : \mathbf{c}_i^T\mathbf{u} = 0$. Testing H_0 is the same as testing multiple non-redundant row hypotheses $\{H_{0i}^*\}_{i=1}^R$ simultaneously, where *R* is the number of non-redundant row hypotheses, and H_{0i}^* represents the *i*th non-redundant hypothesis. A hypothesis H_{0i} is redundant if there exists another hypothesis $H_{0j},\ j \neq i$ such that $c_i = ac_j,\ a \neq 0$.

Adjusted p-values. For each individual hypothesis H_{0i}, test statistics can be calculated. Let p_i denote the *p*-value for testing H_{0i} and p_i^* denote and adjusted *p*-value. The conclusion from multiple testing is, at level α (the family-wise type I error),

reject $H_{0i} : \mathbf{c}_i^T\mathbf{u} = 0$ if $p_i^* < \alpha$;

reject $H_0 : \mathbf{Cu} = \mathbf{0}$ if $\min_i(p_i^*) < \alpha$.

Several different methods to adjust *p*-values are provided here. Please note that if the adjusted *p*-value is bigger than 1, it is set to 1 in all the methods.

Adjusted confidence intervals. Note that if confidence intervals are also calculated for the above hypothesis, then adjusting confidence intervals is required to correspond to adjusted *p*-values. The only item needed to be adjusted in the confidence intervals is the critical value from the standard normal distribution. Assume that the original critical value is $z_{1-\alpha/2}$ and the adjusted critical value is z^*.

LSD (Least Significant Difference)

The adjusted *p*-values are the same as the original *p*-values:

$$p_i^* = p_i$$

The adjusted critical value is:

$$t^* = t_{v^i,\alpha/2}$$

Sequential Bonferroni

The adjusted *p*-values are:

$$p^*_{(i)} = \begin{cases} Rp_{(1)} & i = 1 \\ \max\left((R-i+1)\,p_{(i)}, p^*_{(i-1)}\right) & i \geq 2 \end{cases}$$

The adjusted critical values will correspond to the ordered adjusted *p*-values as follows:

$$t^*_{v^{(i)}} = \begin{cases} t_{v^{(i)},\frac{\alpha}{2R}} & \text{if } i = 1 \\ t_{v^{(i)},\frac{\alpha}{2(R-i+1)}} & \text{if } p^*_{(i)}{=}(R-i+1)p_{(i)} \text{ for } i \geq 2 \\ t_{v^{(i)},\frac{\alpha}{2\left(p^*_{(i-1)}/p_{(i)}\right)}} & \text{if } p^*_{(i)}p^*_{(i-1)} \text{ for } \ i \geq 2 \end{cases}$$

Sequential Sidak

The adjusted *p*-values are:

$$p^*_{(i)} = \begin{cases} 1-\left(1-p_{(1)}\right)^R & i = 1 \\ \max\left(1-\left(1-p_{(i)}\right)^{R-i+1}, p^*_{(i-1)}\right) & i \geq 2 \end{cases}$$

The adjusted critical values will correspond to the ordered adjusted *p*-values as follows:

$$t^*_{v^{(i)}} = \begin{cases} t_{v^{(i)},\frac{1-(1-\alpha)^{1/R}}{2}} & \text{if } i = 1, \\ t_{v^{(i)},\frac{1-(1-\alpha)^{1/(R-i+1)}}{2}} & \text{if } p^*_{(i)} = (R-i+1)p_{(i)} \text{ for } i \geq 2 \\ t_{v^{(i)},\frac{1-(1-\alpha)^{1/x}}{2}} & \text{if } p^*_{(i)}{=}p^*_{(i-1)} \text{ for } i \geq 2 \end{cases};$$

where $x = \frac{\ln\left(1-p^*_{(i-1)}\right)}{\ln\left(1-p_{(i)}\right)}$.

Method for computing degrees of freedom

Residual method

The value of degrees of freedom is given by $N - rank(\mathbf{X})$, where *N* is the effective sample size and **X** is the design matrix of fixed effects.

Satterthwaite's approximation

First perform the spectral decomposition $\mathbf{L}\hat{\mathbf{C}}\mathbf{L}^T = \mathbf{\Gamma}^T\mathbf{D}\mathbf{\Gamma}$ where $\mathbf{\Gamma}$ is an orthogonal matrix of eigenvectors and $\mathbf{D}$ is a diagonal matrix of eigenvalues. If l_m is the *m*th row of $\mathbf{\Gamma}\mathbf{L}$, d_m is the *m*th eigenvalues and

$$\nu_m = \frac{2d_m}{\mathbf{g_m}\Sigma(\hat{\theta})^{-1}\mathbf{g_m}}$$

where $\mathbf{g}_m = \frac{\partial l_m \mathbf{C} l_m^T}{\partial \theta}|_{\theta=\hat{\theta}}$ and $\Sigma_{\hat{\theta}}$ is the asymptotic covariance matrix of $\hat{\theta}$ obtained from the Hessian matrix of the objective function; that is, $\Sigma_{\hat{\theta}} = 2\ \mathrm{H}^{-1}$. If

$$E = \sum_{m=1}^{q} \frac{\nu_m}{\nu_m - 2} I\left(\nu_m > 2\right)$$

then the denominator degree of freedom is given by

$$\nu = \frac{2E}{E-q}$$

Note that the degrees of freedom can only be computed when *E*>*q*.

Scoring

For GLMMs, predicted values and relevant statistics can be computed based on solutions of random effects. PQL-type predictions use $\hat{\gamma}$ as the solution for the random effects to compute predicted values and relevant statistics.

PQL-type predicted values and relevant statistics

Predicted value of the linear predictor

$$\mathrm{x}_i^T\hat{\beta} + \mathrm{z}_i^T\hat{\gamma} + o_i$$

Standard error of the linear predictor

$$\hat{\sigma}_\eta = \sqrt{\mathrm{x}_i^T\Sigma\mathrm{x}_i + \mathrm{z}_i^T\hat{C}_{22}\mathrm{z}_i + 2\mathrm{z}_i^T\hat{C}_{21}\mathrm{x}_i},$$

Predicted value of the mean

$$g^{-1}\left(\mathrm{x}_i^T\hat{\beta} + \mathrm{z}_i^T\hat{\gamma} + o_i\right)$$

For the binomial distribution with 0/1 binary target variable, the predicted category $c(\mathrm{x}_i)$ is

$$c(\mathrm{x}_i) = \begin{cases} 1 \text{ (or sucess)} & \text{if } \hat{\mu}_i \geq 0.5 \\ 0 \text{ (or failure)} & \text{otherwise} \end{cases}$$

Approximate 100(1−α)% confidence intervals for the mean

$$g^{-1}\left(\mathbf{x}_i^T\hat{\beta} + \mathbf{z}_i^T\hat{\gamma} + o_i \pm t_{\upsilon,\alpha/2}\hat{\sigma}_\eta\right)$$

Raw residual on the link function transformation

$$r_{\eta,i}^R = v_i - \hat{\eta}_i$$

Raw residual on the original scale of the target

$$r_i^R = y_i - \hat{\mu}_i$$

Pearson-type residual on the link function transformation

$$r_{\eta,i}^P = \frac{r_{\eta,i}^R}{\sqrt{v\hat{a}r(v_i|\gamma)}},$$

where $v\hat{a}r(v_i|\gamma)$ is the *i*th diagonal element of $v\hat{a}r(\mathbf{v}|\gamma)$ and $v\hat{a}r(\mathbf{v}|\gamma) = g'(\hat{\mu})\mathbf{A}_{\hat{\mu}}^{1/2}\hat{R}\mathbf{A}_{\hat{\mu}}^{1/2}g'(\hat{\mu})$ where $\hat{\mu}$ is an *n*× 1 vector of PQL-type predicted values of the mean.

Pearson-type residual on the original scale of the target

$$r_i^P = \frac{r_i^R}{\sqrt{v\hat{a}r(y_i|\gamma)}},$$

where $v\hat{a}r(y_i|\gamma)$ is the *i*th diagonal element of $v\hat{a}r(\mathbf{y}) = \mathbf{A}_{\hat{\mu}_m}^{1/2}\hat{R}\mathbf{A}_{\hat{\mu}_m}^{1/2}$ and $\hat{\mu}_m = \hat{\mu}$.

Classification Table

Suppose that $c\left(j, j'\right)$ is the sum of the frequencies for the observations whose actual target category is *j* (as row) and predicted target category is j' (as column), $j, j' = 1, \cdots, J$ (note that *J* = 2 for binomial), then

$$c\left(j, j'\right) = \sum_{i=1}^{n} f_i I\left(y_i = j, c\left(x_i\right) = j'\right)$$

where $I\left(\cdot\right)$ is indicator function.

Suppose that $p\left(j, j'\right)$ is the $\left(j, j'\right)^{\text{th}}$ element of the classification table, which is the row percentage, then

$$p_{j,j'} = \left(\frac{c\left(j,j'\right)}{\sum_{j'=1}^{J} c\left(j,j'\right)} \right) \times 100\%$$

The percentage of total correct predictions of the model (or "overall percent correct") is

$$p_{total} = \left(\frac{\sum_{j=1}^{J} c\left(j,j\right)}{\sum_{j=1}^{J}\sum_{j'=1}^{J} c\left(j,j'\right)} \right) \times 100\%$$

Nominal multinomial distribution

The nominal multinomial distribution requires some extra notation and explanation.

Notation

The following notation is used throughout this section unless otherwise stated:

S	Number of super subjects.
T_s	Number of cases in the *s*th super subject.
y_{st}	Nominal categorical target for the *t*th case in the *s*th super subject. Its category values are denoted as 1, 2, and so on.
J	The total number of categories for target.
$\mathbf{y}_{st}$	Dummy vector of y_{st}, $\mathbf{y}_{st} = \left(y_{st,1}, \cdots, y_{st,J-1}\right)^{\mathrm{T}}$, where $y_{st,j} = 1$ if $y_{st} = j$, otherwise $y_{st,j} = 0$. The superscript *T* means the transpose of a matrix or vector.
$\mathbf{y}_s$	$\mathbf{y}_s = \left(\mathbf{y}_{s1}^{\mathrm{T}}, \ldots, \mathbf{y}_{sT_s}^{\mathrm{T}}\right)^{\mathrm{T}}, s = 1, \cdots, S.$
$\mathbf{y}$	$\mathbf{y} = \left(\mathbf{y}_1^{\mathrm{T}}, \cdots, \mathbf{y}_S^{\mathrm{T}}\right)^{\mathrm{T}}$
$\pi_{st,j}$	Probability of category *j* for the *t*th case in the *s*th super subject; that is, $\pi_{st,j} = P\left(y_{st} = j\right)$.
$\boldsymbol{\pi}_{st}$	$\boldsymbol{\pi}_{st} = \left(\pi_{st,1}, \cdots, \pi_{st,J-1}\right)^{\mathrm{T}}$
$\boldsymbol{\pi}_s$	$\boldsymbol{\pi}_s = \left(\boldsymbol{\pi}_{s1}^{\mathrm{T}}, \cdots, \boldsymbol{\pi}_{sT_s}^{\mathrm{T}}\right)^{\mathrm{T}}, s = 1, \cdots, S$
$\boldsymbol{\pi}$	$\boldsymbol{\pi} = \left(\boldsymbol{\pi}_1^{\mathrm{T}}, \cdots, \boldsymbol{\pi}_S^{\mathrm{T}}\right)^{\mathrm{T}}$

$\eta_{st,j}$	Linear predictor value for category *j* of the *t*th case in the *s*th super subject.
η_{st}	$\eta_{st} = (\eta_{st,1}, \cdots, \eta_{st,J-1})^{\mathrm{T}}$
η_s	$\eta_s = \left(\eta_{s1}^{\mathrm{T}}, \ldots, \eta_{sT_s}^{\mathrm{T}}\right)^{\mathrm{T}}, s = 1, \cdots, S$
η	(*n* (*J*–1)) × 1 vector of linear predictor. $\eta = \left(\eta_1^{\mathrm{T}}, \cdots, \eta_S^{\mathrm{T}}\right)^{\mathrm{T}}$
x_{st}	*p*× 1 vector of predictor variables for the *t*th case in the *s*th super subject. The first element is 1 if there is an intercept.
X_s	$X_s = (I_{J-1} \otimes x_{s1}, \cdots, I_{J-1} \otimes x_{sT_s})^T, s = 1, \cdots, S$
X	(*n* (*J*–1)) × (*J*–1)*p* design matrix of fixed effects, $X = \left(X_1^T, \cdots, X_S^T\right)^T$
z_{st}	*r*× 1 vector of coefficients for the random effect corresponding to the *t*th case in the *s*th super subject.
Z_s	$Z_s = (I_{J-1} \otimes z_{s1}, \cdots, I_{J-1} \otimes z_{sT_s})^T, s = 1, \cdots, S$
Z	Design matrix of random effects, $Z = \bigoplus_{s=1}^{S} Z_s$, where ⊕ is the direct sum of matrices.
O	*n*× 1 vector of offsets, $O = (o_{11}, \cdots, o_{1T_1}, \cdots, o_{ST_S})^T$, where o_{st} is the offset value of the *t*th case in the *s*th super subject. This can't be the target (y) or one of the predictors (X). The offset must be continuous.
O^*	$O^* = O \otimes 1_{J-1}$, where 1_q is a length *q* vector of 1.
β_j	*p*× 1 vector of unknown parameters for category *j*, $\beta_j = (\beta_{j1}, \cdots, \beta_{jp})^T, j = 1, \cdots, J$. The first element in β_j is the intercept for the category *j*, if there is one.
β	$\beta = \left(\beta_1^T, \cdots, \beta_{J-1}^T\right)^T$
γ_{sj}	*r* × 1 vector of random effects for category *j* in the *s*th super subject, $j = 1, \cdots, J-1$.
γ_s	Random effects for the *s*th super subject, $\gamma_s = \left(\gamma_{s,1}^{\mathrm{T}}, \cdots, \gamma_{s,J-1}^{\mathrm{T}}\right)^{\mathrm{T}}$.
γ	$\gamma = \left(\gamma_1^T, \cdots, \gamma_S^T\right)^T$
ω_{st}	Scale weight of the *t*th case in the *s*th super subject. It does not have to be integers. If it is less than or equal to 0 or missing, the corresponding case is not used.
ω	*n*× 1 vector of scale weight variable, $\omega = (\omega_{11}, \cdots, \omega_{1T_1}, \cdots, \omega_{S1}, \cdots, \omega_{ST_S})^{\mathrm{T}}$.
f_{st}	Frequency weight of the *t*th case in the *s*th super subject. If it is a non-integer value, it is treated by rounding the value to the nearest integer. If it is less than 0.5 or missing, the corresponding cases are not used.
f	*n*× 1 vector of frequency count variable, $f = (f_{11}, \cdots, f_{1T_1}, \cdots, f_{S1,}, \cdots, f_{ST_S})^{\mathrm{T}}$
N	Effective sample size, $N = \sum_{i=1}^{n} f_i$. If frequency count variable *f* is not used, *N* = *n*.

Model

The form of a generalized linear mixed model for nominal target with the random effects is

$$\eta = g\left(E\left(y\right)\left|\gamma\right.\right) = X\beta + Z\gamma + O^*$$

where η is the linear predictor; **X** is the design matrix for fixed effects; **Z** is the design matrix for random effects; γ is a vector of random effects which are assumed to be normally distributed with mean **0** and variance matrix **G**; $g(.)$ is the logit link function such that

$$\eta_{st,j} = g(\pi_{st,j}) = \log\left(\frac{\pi_{st,j}}{\pi_{st,J}}\right)$$

And its inverse function is

$$\pi_{st,j} = g^{-1}(\eta_{st,j}) = \begin{cases} \dfrac{\exp(\eta_{st,j})}{1+\sum_{k=1}^{J-1}\exp(\eta_{st,k})}, j = 1, \cdots, J-1, \\ \dfrac{1}{1+\sum_{k=1}^{J-1}\exp(\eta_{st,k})}, j = J. \end{cases}$$

The variance of **y**, conditional on the random effects is

$$Var(y|\gamma) = A_\mu^{1/2} R A_\mu^{1/2}$$

where $A_\mu = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \left(diag(\pi_{st}) - \pi_{st}\pi_{st}^{\mathrm{T}}\right)/\omega_{st}$ and R = ϕI which means that R-side effects are not supported for the multinomial distribution. ϕ is set to 1.

Estimation

Linear mixed pseudo model

Similarly to “Linear mixed pseudo model”, we can obtain a weighted linear mixed model

$$v = X\beta + Z\gamma + \epsilon$$

where v ≡ $D^{-1}(y - \tilde{\pi}) + g(\tilde{\pi}) - O^*$ and error terms ε ~ $N\left(0, D^{-1}A_{\tilde{\pi}}^{1/2} R A_{\tilde{\pi}}^{1/2} D^{-1}\right)$ with

$$D = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} D_{st} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \frac{dg^{-1}(\tilde{\eta}_{st})}{d\tilde{\eta}_{st}} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \left(diag(\tilde{\pi}_{st}) - \tilde{\pi}_{st}\tilde{\pi}_{st}^{\mathrm{T}}\right)$$

and

$$A_{\tilde{\mu}} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \left(diag(\tilde{\pi}_{st}) - \tilde{\pi}_{st}\tilde{\pi}_{st}^{\mathrm{T}}\right)/\omega_{st}.$$

And block diagonal weight matrix is

$$\tilde{W} = \mathrm{D}A_{\tilde{\mu}}^{-1}\mathrm{D} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \omega_{st}\mathrm{D}_{st}.$$

The Gaussian log pseudo-likelihood (PL) and restricted log pseudo-likelihood (REPL), which are expressed as the functions of covariance parameters in θ, corresponding to the linear mixed model for *v* are the following:

$$\ell(\theta;\mathrm{v}) = -\frac{1}{2}\ln|\mathrm{V}(\theta)| - \frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta) - \frac{N}{2}\ln(2\pi)$$

$$\ell_R(\theta;\mathrm{v}) = -\frac{1}{2}\ln|\mathrm{V}(\theta)| - \frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta) - \frac{1}{2}\ln\left|\mathrm{X}^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{X}\right| - \frac{N-p_x}{2}\ln(2\pi)$$

where $\mathrm{V}(\theta) = Z\mathrm{G}(\theta)Z^T + \tilde{W}^{-1/2}\mathrm{R}(\theta)\tilde{W}^{-1/2}$,$r(\theta) = v - X\hat{\beta}$,*N* denotes the effective sample size, and p_x denotes the total number of non-redundant parameters for β.

The parameter θ can be estimated by linear mixed model using the objection function $-2\ell(\theta;\mathrm{v})$or $-2\ell_R(\theta;\mathrm{v})$, β and γ are computed as

$$\hat{\beta} = \left(X^{\mathrm{T}}V\left(\hat{\theta}\right)^{-1}X\right)^{-1}X^{\mathrm{T}}V\left(\hat{\theta}\right)^{-1}v$$

$$\hat{\gamma} = \hat{G}Z^{\mathrm{T}}V\left(\hat{\theta}\right)^{-1}\hat{r}$$

Iterative process

The doubly iterative process for the estimation of θ is the same as that for other distributions, if we replace $\tilde{\mu}$ and $X\tilde{\beta} + Z\tilde{\gamma} + O$ with $\tilde{\pi}$ and $X\tilde{\beta} + Z\tilde{\gamma} + \mathrm{O}^*$ respectively, and set initial estimation of π as

$$\pi^{(0)} = \frac{y + 1/J}{2}$$

For more information, see the topic "Iterative process".

Post-estimation statistics

Wald confidence intervals

The Wald confidence intervals for covariance parameter estimates are described in "Wald confidence intervals for covariance parameter estimates".

Statistics for estimates of fixed and random effects

Similarly to "Statistics for estimates of fixed and random effects", the approximate covariance matrix of $(\beta - \beta, \hat{\gamma} - \gamma)$ is

$$\hat{C} = \begin{bmatrix} X^T R^{*-1} X & X^T R^{*-1} Z \\ Z^T R^{*-1} X & Z^T R^{*-1} Z + G\left(\hat{\theta}\right)^{-1} \end{bmatrix}^{-} = \begin{bmatrix} C_{11} & C_{21}^T \\ C_{21} & C_{22} \end{bmatrix}$$

Where $R^* = v\hat{a}r\left(v|\gamma\right) = \hat{D}^{-1} A_{\hat{\pi}}^{1/2} R A_{\hat{\pi}}^{1/2} \hat{D}^{-1}$ with $\hat{D} = \overset{S}{\underset{s=1}{\oplus}} \overset{T_s}{\underset{t=1}{\oplus}} \left(diag\left(\hat{\pi}_{st}\right) - \hat{\pi}_{st}\hat{\pi}_{st}^{\mathrm{T}}\right)$, and

$$\hat{C}_{11} = \left(X^T \hat{V}^{-1} X\right)^{-}$$

$$\hat{C}_{21} = -\hat{G} Z^T \hat{V}^{-1} X \hat{C}_{11}$$

$$\hat{C}_{22} = \left(Z^T \hat{R}^{-1} Z + \hat{G}^{-1}\right)^{-1} - \hat{C}_{21} X^T \hat{V}^{-1} Z G$$

Statistics for estimates of fixed and random effects on original scale

If the fixed effects are transformed when constructing matrix **X**, then the final estimates of β, C_{11}, C_{21}, and C_{22} above are based on transformed scale, denoted as $\hat{\beta}^*$, $\hat{C}_{11}^*$, $\hat{C}_{21}^*$ and $\hat{C}_{22}^*$, respectively. They would be transformed back on the original scale, denoted as $\hat{\beta}$, ,$\hat{C}_{11}$ $\hat{C}_{21}$ and $\hat{C}_{22}$, respectively, as follows:

$$\hat{\beta} = T\hat{\beta}^*$$

$$\hat{C}_{11} = T\hat{C}_{11}^* T^{\mathrm{T}}$$

$$\hat{C}_{21} = \hat{C}_{21}^* T^{\mathrm{T}}$$

$$\hat{C}_{22} = \hat{C}_{22}^*$$

where $T = \overset{J-1}{\underset{j=1}{\oplus}} \mathrm{A}_j$.

Estimated covariance matrix of the fixed effects parameters

Model-based estimated covariance

$$\Sigma_m = \hat{C}_{11}$$

Robust estimated covariance of the fixed effects parameters

$$\Sigma_r = \Sigma_m \left(\sum_{s=1}^{S} X_s^T \hat{V}_s^{-1} \hat{r}_s \hat{r}_s^T \hat{V}_s^{-1} X_s \right) \Sigma_m$$

where $\hat{r}_s = v_s - X_s\hat{\beta}$, and v_s is a part of v corresponding to the *s*th super subject.

Standard error for estimates in fixed effects and predictions in random effects

Let $\hat{\beta}_{jc}$ denote a non-redundant fixed effects parameter estimate. Its standard error is the square root of the $((j-1)p+c)th$ diagonal element of Σ

$$\hat{\sigma}_{\beta_{jc}} = \sqrt{\sigma_{((j-1)p+c),((j-1)p+c)}}$$

The standard error for redundant parameter estimates is set to system missing value.

Similarly, let $\hat{\gamma}_i$ denote the *i*th random effects prediction. Its standard error is the square root of the *i*th diagonal element of $\hat{C}_{22}$:

$$\hat{\sigma}_{\gamma_i} = \sqrt{\hat{C}_{22,ii}}$$

Test statistics for estimates in fixed effects and predictions in random effects

Test statistics for estimates in fixed effects and predictions in random effects are as those described in “Statistics for estimates of fixed and random effects”.

Wald confidence intervals for estimates in fixed effects and random effects predictions

Wald confidence intervals are as those described in “Statistics for estimates of fixed and random effects”.

Testing

Information criteria

These are as described in “Goodness of fit”.

Tests of fixed effects

For each effect specified in the model, a type III test matrix **L** is constructed from the generating matrix $H_\omega = \left(x^T\Omega x\right)^- x^T\Omega x$ where, $x = \left(x_{11}^T, \cdots, x_{st}^T, \cdots x_{ST_S}^T\right)^T$ and $\Omega = diag\left(\omega_{11}, \cdots, \omega_{1T_1,}, \cdots, \omega_{S1}, \cdots, \omega_{ST_S}\right)$. Then the test statistic is

$$F = \frac{\hat{\beta}^T L^{*T}\left(L^*\Sigma L^{*T}\right)^{-1} L^*\hat{\beta}}{r_c}$$

where $r_c = rank\left(L^*\Sigma L^{*T}\right)$ and $L^* = I_{J-1} \otimes$ L. The statistic has an approximate *F* distribution. The numerator degrees of freedom is r_c and the denominator degree of freedom is υ. For more information, see the topic “Method for computing degrees of freedom”.

Scoring

PQL-type predicted values and relevant statistics

$(J-1)\times 1$ predicted vector of the linear predictor

$$\hat{\eta}_{st} = (I_{J-1}\otimes x_{st})^{\mathrm{T}}\hat{\beta} + (I_{J-1}\otimes \mathrm{z}_{st})^{\mathrm{T}}\hat{\gamma}_s + 1_{J-1}\otimes o_{st}$$

Estimated covariance matrix of the linear predictor

$$\begin{aligned}\Sigma_{\hat{\eta}_{st}} = \quad & (I_{J-1}\otimes x_{st})^T\Sigma(I_{J-1}\otimes x_{st}) + (I_{J-1}\otimes \mathrm{z}_{st})^T\hat{C}^s_{22}(I_{J-1}\otimes \mathrm{z}_{st})\\ & +(I_{J-1}\otimes \mathrm{z}_{st})^T\hat{C}^s_{21}(I_{J-1}\otimes x_{st}) + (I_{J-1}\otimes x_{st})^T\left(\hat{C}^s_{21}\right)^T(I_{J-1}\otimes \mathrm{z}_{st})\end{aligned}$$

where $\hat{C}^s_{22}$ is a diagonal block corresponding to the *s*th super subject, the approximate covariance matrix of $\hat{\gamma}_s-\gamma_s$; $\hat{C}^s_{21}$ is a part of $\hat{C}_{21}$ corresponding to the *s*th super subject.

The estimated standard error of the *j*th element in $\hat{\eta}_{st}$, $\hat{\eta}_{st,j}$, is the square root of the *j*th diagonal element of $\Sigma_{\hat{\eta}_{st}}$,

$$\sigma_{\hat{\eta}_{st,j}} = \sqrt{\sigma_{\hat{\eta}_{st},jj}}$$

Predicted value of the probability for category *j*

$$\hat{\pi}_{st,j} = g^{-1}(\hat{\eta}_{st,j}) = \begin{cases}\dfrac{\exp\left(\hat{\eta}_{st,j}\right)}{1+\sum\limits_{k=1}^{J-1}\exp\left(\hat{\eta}_{st,k}\right)}, j=1,\cdots,J-1,\\ \dfrac{1}{1+\sum\limits_{k=1}^{J-1}\exp\left(\hat{\eta}_{st,k}\right)}, j=J.\end{cases}$$

Predicted category

$$c(\mathrm{x}_{st}) = arg\max_j \hat{\pi}_{st,j},$$

If there is a tie in determining the predicted category, the tie will be broken by choosing the category with the highest $N_j=\sum_{s=1}^{S}\sum_{t=1}^{T_s} f_{st}y_{st,j}$. If there is still a tie, the one with the lowest category number is chosen.

Approximate 100(1−α)% confidence intervals for the predicted probabilities

The covariance matrix of $\hat{\pi}_{st}$ can be computed as

$$Cov\left(\hat{\pi}_{st}\right) = \nabla g^{-1}(\hat{\eta}_{st})^T\Sigma_{\hat{\eta}_{st}}\nabla g^{-1}\left(\hat{\eta}_{st}\right)$$

where

$$\nabla a^{-1}(\hat{\eta}_{st}) = \begin{bmatrix} \frac{\partial \hat{\pi}_{st,1}}{\partial \hat{\eta}_{st,1}} & \cdots & \frac{\partial \hat{\pi}_{st,J-1}}{\partial \hat{\eta}_{st,1}} & \frac{\partial \hat{\pi}_{st,J}}{\partial \hat{\eta}_{st,1}} \\ \vdots & & \vdots & \vdots \\ \frac{\partial \hat{\pi}_{st,1}}{\partial \hat{\eta}_{st,J-1}} & \cdots & \frac{\partial \hat{\pi}_{st,J-1}}{\partial \hat{\eta}_{st,J-1}} & \frac{\partial \hat{\pi}_{st,J}}{\partial \hat{\eta}_{st,J-1}} \end{bmatrix}$$

with

$$\frac{\partial \hat{\pi}_{st,j}}{\partial \hat{\eta}_{st,k}} = \begin{cases} \hat{\pi}_{st,j}(1-\hat{\pi}_{st,j}), j = k \\ -\hat{\pi}_{st,j}\hat{\pi}_{st,k}, j \neq k \end{cases}$$

then the confidence interval is

$$\hat{\pi}_{st,j} \pm t_{\upsilon,\alpha/2}\hat{\sigma}_{\pi_{st,j}}, j = 1, \cdots, J$$

where $\hat{\sigma}^2_{\pi_{st,j}}$ is the *j*th diagonal element of $Cov(\hat{\pi}_{st})$ and the estimated variance of $\hat{\pi}_{st,j}, j = 1, \cdots, J$.

Ordinal multinomial distribution

The ordinal multinomial distribution requires some extra notation and explanation.

Notation

The following notation is used throughout this section unless otherwise stated:

S	Number of super subjects.
T_s	Number of cases in the *s*th super subject.
y_{st}	Ordinal categorical target for the *t*th case in the *s*th super subject. Its category values are denoted as consecutive integers from 1 to *J*.
J	The total number of categories for target.
$\mathbf{y}_{st}$	Indicator vector of y_{st}, $\mathbf{y}_{st} = (y_{st,1}, \cdots, y_{st,J-1})^T$, where $y_{st,j} = 1$ if $y_{st} = j$, otherwise $y_{st,j} = 0$. The superscript *T* means the transpose of a matrix or vector.
$\mathbf{y}_s$	$\mathbf{y}_s = \left(\mathbf{y}_{s1}^T, \ldots, \mathbf{y}_{sT_s}^T\right)^T, s = 1, \cdots, S.$
$\mathbf{y}$	$\mathbf{y} = \left(\mathbf{y}_1^T, \cdots, \mathbf{y}_S^T\right)^T$
$\lambda_{st,j}$	Cumulative target probability for category *j* for the *t*th case in the *s*th super subject; $\lambda_{st,j} = P(y_{st} \leq j)$.
$\boldsymbol{\lambda}$	$\boldsymbol{\lambda} = \left(\boldsymbol{\lambda}_1^T, \cdots, \boldsymbol{\lambda}_S^T\right)^T$, where $\boldsymbol{\lambda}_s = \left(\boldsymbol{\lambda}_{s1}^T, \cdots, \boldsymbol{\lambda}_{sT_s}^T\right)^T$ and $\boldsymbol{\lambda}_{st}^T = (\lambda_{st,1}, \cdots, \lambda_{st,J-1})$, $s = 1, \ldots, S$ and $t = 1, \ldots, T_s$.
$\pi_{st,j}$	Probability of category *j* for the *t*th case in the *s*th super subject; that is, $\pi_{st,j} = P(y_{st} = j)$ and $\pi_{st,j} = \lambda_{st,j} - \lambda_{st,j-1}$.

π_{st}	$\pi_{st} = (\pi_{st,1}, \cdots, \pi_{st,J-1})^{\mathrm{T}}$
π_s	$\pi_s = \left(\pi_{s1}^{\mathrm{T}}, \cdots, \pi_{sT_s}^{\mathrm{T}}\right)^{\mathrm{T}}, s = 1, \cdots, S$
π	$\pi = \left(\pi_1^{\mathrm{T}}, \cdots, \pi_S^{\mathrm{T}}\right)^{\mathrm{T}}$
$\eta_{st,j}$	Linear predictor value for category *j* of the *t*th case in the *s*th super subject.
η_{st}	$\eta_{st} = (\eta_{st,1}, \cdots, \eta_{st,J-1})^{\mathrm{T}}$
η_s	$\eta_s = \left(\eta_{s1}^{\mathrm{T}}, \ldots, \eta_{sT_s}^{\mathrm{T}}\right)^{\mathrm{T}}, s = 1, \cdots, S$
η	(*n* (*J*−1)) × 1 vector of linear predictor. $\eta = \left(\eta_1^{\mathrm{T}}, \cdots, \eta_S^{\mathrm{T}}\right)^{\mathrm{T}}$
x_{st}	*p*× 1 vector of predictors for the *t*th case in the *s*th super subject.
z_{st}	*r*× 1 vector of coefficients for the random effect corresponding to the *t*th case in the *s*th super subject.
O	*n*× 1 vector of offsets, $O = (o_{11}, \cdots, o_{1T_1}, \cdots, o_{ST_S})^T$, where o_{si} is the offset value of the *t*th case in the *s*th super subject. This can't be the target (y) or one of the predictors (X). The offset must be continuous.
O^*	$O^* = O \otimes 1_{J-1}$, where 1_q is a length *q* vector of 1's.
ψ	*J*−1 × 1 vector of threshold parameters, $\psi = (\psi_1, \psi_2, \ldots, \psi_{J-1})^{\mathrm{T}}$ and $\psi_1 < \psi_2 < \cdots < \psi_{J-1}$
β	*p*× 1 vector of unknown parameters.
B	(*J*−1+*p*) × 1 vector of all parameters B=$\left(\psi^{\mathrm{T}}, \beta^{\mathrm{T}}\right)^{\mathrm{T}}$
ω_{st}	Scale weight of the *t*th case in the *s*th super subject. It does not have to be integers. If it is less than or equal to 0 or missing, the corresponding case is not used.
ω	*n*× 1 vector of scale weight variable, $\omega = (\omega_{11}, \cdots, \omega_{1T_1}, \cdots, \omega_{S1}, \cdots, \omega_{ST_S})^{\mathrm{T}}$.
f_{st}	Frequency weight of the *i*th case in the *s*th super subject. If it is a non-integer value, it is treated by rounding the value to the nearest integer. If it is less than 0.5 or missing, the corresponding cases are not used.
f	*n*× 1 vector of frequency count variable, $f = (f_{11}, \cdots, f_{1T_1}, \cdots, f_{S1,}, \cdots, f_{ST_S})^{\mathrm{T}}$
N	Effective sample size, $N = \sum_{i=1}^{n} f_i$. If frequency count variable *f* is not used, *N* = *n*.
A ⊗ B	direct (or Kronecker) product of **A** and **B**, which is equal to $\begin{bmatrix} a_{11}\mathrm{B} & a_{12}\mathrm{B} & a_{13}\mathrm{B} \\ a_{21}\mathrm{B} & a_{22}\mathrm{B} & a_{23}\mathrm{B} \\ a_{31}\mathrm{B} & a_{32}\mathrm{B} & a_{33}\mathrm{B} \end{bmatrix}$
1_m	*m*× 1 vector of 1's; $1_m = (1, \ldots, 1)^{\mathrm{T}}$

Model

The form of a generalized linear mixed model for an ordinal target with random effects is

$$\eta = g(\lambda) = XB + Z\gamma + O^*$$

where η is the expanded linear predictor vector; **λ** is the expanded cumulative target probability vector; $g\left(.\right)$ is a cumulative link function; **X** is the expanded design matrix for fixed effects arranged as follows

$$\mathrm{X} = \begin{pmatrix} \mathrm{X}_1 \\ \vdots \\ \mathrm{X}_S \end{pmatrix},$$

$$\mathrm{X}_s = \begin{pmatrix} \mathrm{X}_{s1} \\ \vdots \\ \mathrm{X}_{sT_s} \end{pmatrix}_{T_s(J-1)\times(J-1+p)},$$

$$\mathrm{X}_{st} = \begin{pmatrix} \mathrm{I}_{J-1} & 1_{J-1} \otimes -\mathrm{x}_{st}^{\mathrm{T}} \end{pmatrix}_{(J-1)\times(J-1+p)}$$
$$= \begin{pmatrix} 1 & \cdots & 0 & -\mathrm{x}_{st}^{\mathrm{T}} \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 1 & -\mathrm{x}_{st}^{\mathrm{T}} \end{pmatrix}$$
$$= \begin{pmatrix} 1 & \cdots & 0 & -x_{st,1} & \cdots & -x_{st,p} \\ \vdots & \ddots & \vdots & \vdots & \cdots & \vdots \\ 0 & \cdots & 1 & -x_{st,1} & \cdots & -x_{st,p} \end{pmatrix}$$

$\mathrm{B}=\left(\psi^{\mathrm{T}},\beta^{\mathrm{T}}\right)^{\mathrm{T}} = \left(\left(\psi_1,\ldots,\psi_{J-1}\right),\beta^{\mathrm{T}}\right)^{\mathrm{T}}$; **Z** is the expanded design matrix for random effects arranged as follows

$$\mathrm{Z} = \begin{pmatrix} \mathrm{Z}_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \mathrm{Z}_S \end{pmatrix},$$

$$\mathrm{Z}_s = \begin{pmatrix} \mathrm{Z}_{s1} \\ \vdots \\ \mathrm{Z}_{sT_s} \end{pmatrix}_{T_s(J-1)\times r},$$

$$\mathrm{Z}_{st} = \begin{pmatrix} 1_{J-1} \otimes -\mathrm{z}_{st}^{\mathrm{T}} \end{pmatrix}_{(J-1)\times r},$$

γ is a vector of random effects which are assumed to be normally distributed with mean **0** and variance matrix **G**.

The variance of **y**, conditional on the random effects is

$$Var\left(y|\gamma\right) = A_{\mu}^{1/2} R A_{\mu}^{1/2}$$

where $A_{\mu} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \left(diag\left(\pi_{st}\right) - \pi_{st}\pi_{st}^{\mathrm{T}}\right)/\omega_{st}$ and $\mathrm{R} = \phi\mathrm{I}$ which means that R-side effects are not supported for the multinomial distribution. ϕ is set to 1.

Estimation

Linear mixed pseudo model

Similarly to “Linear mixed pseudo model”, we can obtain a weighted linear mixed model

$$v = X\beta + Z\gamma + \epsilon$$

where $\mathrm{v} \equiv \mathrm{D}^{-1}(\mathrm{y} - \tilde{\pi}) + g\left(\tilde{\lambda}\right) - \mathrm{O}^{*}$ and error terms $\varepsilon \sim N\left(0, \mathrm{D}^{-1} A_{\tilde{\pi}}^{1/2} R A_{\tilde{\pi}}^{1/2} (\mathrm{D}^{-1})^{\mathrm{T}}\right)$ with

$$\mathrm{D} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \mathrm{D}_{st} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \frac{\mathrm{d}g^{-1}(\tilde{\eta}_{st})}{\mathrm{d}\tilde{\eta}_{st}} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \frac{\mathrm{d}\tilde{\lambda}_{st}}{\mathrm{d}\tilde{\eta}_{st}}$$

$$\mathrm{D}_{st} = \begin{bmatrix} \frac{\partial\tilde{\lambda}_{st,1}}{\partial\tilde{\eta}_{st,1}} & 0 & \cdots & 0 & 0 \\ -\frac{\partial\tilde{\lambda}_{st,1}}{\partial\tilde{\eta}_{st,1}} & \frac{\partial\tilde{\lambda}_{st,2}}{\partial\tilde{\eta}_{st,2}} & \cdots & 0 & 0 \\ \vdots & \ddots & \ddots & \vdots & \vdots \\ 0 & 0 & \ddots & \frac{\partial\tilde{\lambda}_{st,J-2}}{\partial\tilde{\eta}_{st,J-2}} & 0 \\ 0 & 0 & \cdots & -\frac{\partial\tilde{\lambda}_{st,J-2}}{\partial\tilde{\eta}_{st,J-2}} & \frac{\partial\tilde{\lambda}_{st,J-1}}{\partial\tilde{\eta}_{st,J-1}} \end{bmatrix}$$

and

$$A_{\tilde{\mu}} = \bigoplus_{s=1}^{S} \bigoplus_{t=1}^{T_s} \left(diag\left(\tilde{\pi}_{st}\right) - \tilde{\pi}_{st}\tilde{\pi}_{st}^{\mathrm{T}}\right) / \omega_{st}.$$

And block diagonal weight matrix is

$$\tilde{W} = \mathrm{D}^{\mathrm{T}} A_{\tilde{\mu}}^{-1} \mathrm{D}$$

The Gaussian log pseudo-likelihood (PL) and restricted log pseudo-likelihood (REPL), which are expressed as the functions of covariance parameters in θcorresponding to the linear mixed model for v are the following:

$$\ell(\theta; \mathrm{v}) = -\frac{1}{2}\ln|\mathrm{V}(\theta)| - \frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta) - \frac{N}{2}\ln(2\pi)$$

$$\ell_R(\theta; \mathrm{v}) = -\frac{1}{2}\ln|\mathrm{V}(\theta)| - \frac{1}{2}\mathrm{r}(\theta)^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{r}(\theta) - \frac{1}{2}\ln\left|\mathrm{X}^{\mathrm{T}}\mathrm{V}(\theta)^{-1}\mathrm{X}\right| - \frac{N - p_x}{2}\ln(2\pi)$$

where $\mathrm{V}(\theta) = Z\mathrm{G}(\theta)Z^{T} + \tilde{W}^{-1/2}\mathrm{R}(\theta)\tilde{W}^{-1/2}$,$r(\theta) = v - X\hat{\mathbf{B}}$,$N$ denotes the effective sample size, and p_x denotes the total number of non-redundant parameters for $\mathbf{B}$.

The parameter θ can be estimated by linear mixed model using the objection function $-2\ell(\theta; \mathrm{v})$or $-2\ell_R(\theta; \mathrm{v})$, $\mathbf{B}$ and γ are computed as

$$\hat{\mathbf{B}} = \left(X^{\mathrm{T}} V\left(\hat{\theta}\right)^{-1} X\right)^{-1} X^{\mathrm{T}} V\left(\hat{\theta}\right)^{-1} v$$

$$\hat{\gamma} = \hat{G} Z^{\mathrm{T}} V\left(\hat{\theta}\right)^{-1} \hat{r}$$

Iterative process

The doubly iterative process for the estimation of θ is the same as that for other distributions, if we replace $\tilde{\mu}$ and $X\tilde{\boldsymbol{B}} + Z\tilde{\gamma} + O$ with $\tilde{\pi}$ and $X\tilde{\boldsymbol{B}} + Z\tilde{\gamma} + O^*$ respectively, and set initial estimation of π as

$$\pi^{(0)} = \frac{y + 1/J}{2}$$

For more information, see the topic “Iterative process”.

Post-estimation statistics

Wald confidence intervals

The Wald confidence intervals for covariance parameter estimates are described in “Wald confidence intervals for covariance parameter estimates”.

Statistics for estimates of fixed and random effects

$\hat{C}$ is the approximate covariance matrix of $\left(\hat{\mathbf{B}} - \mathbf{B}, \hat{\gamma} - \gamma\right)$ and R^* in $\hat{C}$ should be

$$R^* = v\hat{a}r\left(v|\gamma\right) = \mathrm{D}^{-1} A_{\tilde{\pi}}^{1/2} R A_{\tilde{\pi}}^{1/2} \left(\mathrm{D}^{-1}\right)^{\mathrm{T}}.$$

Statistics for estimates of fixed and random effects on original scale

If the fixed effects are transformed when constructing matrix **X**, then the final estimates of **B**, denoted as $\hat{B}^*$. They would be transformed back on the original scale, denoted as $\hat{B}$, as follows:

$$\mathrm{B} = \begin{pmatrix} \psi \\ \beta \end{pmatrix} = \begin{pmatrix} \psi_1 \\ \vdots \\ \psi_{J-1} \\ \beta \end{pmatrix} = \mathrm{A} \begin{pmatrix} \psi^* \\ \beta^* \end{pmatrix} = \mathrm{AB}^*$$

where

$$\mathrm{A} = \begin{pmatrix} \mathrm{I}_{J-1} & 1_{J-1} \otimes \left(\mathbf{c}^{\mathrm{T}} \mathrm{S}^{-1}\right) \\ 0 & \mathrm{S}^{-1} \end{pmatrix}$$

Estimated covariance matrix of the fixed effects parameters

The estimated covariance matrix of the fixed effects parameters are described in "Statistics for estimates of fixed and random effects".

Standard error for estimates in fixed effects and predictions in random effects

Let $\hat{\psi}_j, j = 1, \ldots, J-1$, be threshold parameter estimates and $\hat{\beta}_i, i = 1, \ldots, p$, denote non-redundant regression parameter estimates. Their standard errors are the square root of the diagonal elements of Σm or Σr: $\hat{\sigma}_{\psi_j} = \sqrt{\sigma_{jj}}$ and $\hat{\sigma}_{\beta_i} = \sqrt{\sigma_{(J-1+i),(J-1+i)}}$, respectively, where σ_{ii} is the *i*th diagonal element of Σm or Σr.

Standard errors for predictions in random effects are as those described in "Statistics for estimates of fixed and random effects".

Test statistics for estimates in fixed effects and predictions in random effects

The hypotheses $H_{0j} : \psi_j = 0, j = 1, \ldots, J-1$, are tested for threshold parameters using the *t* statistic:

$$t_{\psi_j} = \frac{\hat{\psi}_j}{\hat{\sigma}_{\psi_j}}$$

Test statistics for estimates in fixed effects and predictions in random effects are otherwise as those described in "Statistics for estimates of fixed and random effects".

Wald confidence intervals for estimates in fixed effects and random effects predictions

The 100(1 – α)% Wald confidence interval for threshold parameter is given by

$$\left(\hat{\psi}_j - t_{\upsilon,\alpha/2}\hat{\sigma}_{\psi_j}, \hat{\psi}_j + t_{\upsilon,\alpha/2}\hat{\sigma}_{\psi_j}\right)$$

Wald confidence intervals are otherwise as those described in "Statistics for estimates of fixed and random effects".

The degrees of freedom can be computed by the residual method or Satterthwaite method. For the residual method, $\upsilon = N - (J - 1 + p_x)$. For the Satterthwaite method, it should be similar to that described in "Method for computing degrees of freedom".

Testing

Information criteria

These are as described in "Goodness of fit", with the following modifications.

For REPL, the value of *N* is chosen to be effective sample size minus number of non-redundant parameters in fixed effects, $\sum_{i=1}^{n} f_i - (J - 1 + p_x)$, where p_x is the number of non-redundant parameters in fixed effects, and *d* is the number of covariance parameters.

For PL, the value of *N* is effective sample size, $\sum_{i=1}^{n} f_i$, and *d* is the number of number of non-redundant parameters in fixed effects, $J - 1 + p_x$, plus the number of covariance parameters.

Tests of fixed effects

For each effect specified in the model excluding threshold parameters, a type I or III test matrix $\mathbf{L}_i$ is constructed and H_0: $\mathbf{L}_i\mathbf{B} = \mathbf{0}$ is tested. Construction of matrix $\mathbf{L}_i$ is based on matrix $H_\omega = \left(X_1^T\Omega X_1\right)\ X_1^T\Omega X_1$, where $X_1 = (1, -X)$ and such that $\mathbf{L}_i\mathbf{B}$ is estimable. Note that $\mathbf{L}_i\mathbf{B}$ is estimable if and only if $L_0 = L_0 H_\omega$, where $L_0 = (l_0, L(\beta))$. Construction of L_0 considers a partition of the more general test matrix $L_i = (L_i(\psi), L_i(\beta))$ first, where $L_i(\psi) = (l_1, \ldots, l_{J\;1})$ consists of columns corresponding to the threshold parameters and $L_i(\beta)$ is the part of $\mathbf{L}_i$ corresponding to regression parameters, then replace $L_i(\psi)$ with their sum $l_0 = \sum_{j=1}^{J-1} l_j$ to get L_0.

Note that the threshold-parameter effect is not tested for both type I and III analyses and construction of $\mathbf{L}_i$ is the same as in GENLIN. For more information, see the topic "Default Tests of Model Effects". Similarly, if the fixed effects are transformed when constructing matrix $\mathbf{X}$, then H_ω should be constructed based on transformed values.

Scoring

PQL-type predicted values and relevant statistics

$(J-1) \times 1$ predicted vector of the linear predictor

$$\hat{\eta}_{st} = X_{st}\hat{\mathbf{B}} + Z_{st}\hat{\gamma}_s + 1_{J-1} \otimes o_{st}$$

Estimated covariance matrix of the linear predictor

$$\Sigma_{\hat{\eta}_{st}} = X_{st}\Sigma X_{st}^T + Z_{st}\hat{C}_{22}^s Z_{st}^T + Z_{st}\hat{C}_{21}^s X_{st}^T + X_{st}\left(\hat{C}_{21}^s\right)^T Z_{st}^T$$

where $\hat{C}_{22}^s$ is a diagonal block corresponding to the *s*th super subject, the approximate covariance matrix of $\hat{\gamma}_s - \gamma_s$; $\hat{C}_{21}^s$ is a part of $\hat{C}_{21}$ corresponding to the *s*th super subject.

The estimated standard error of the *j*th element in $\hat{\eta}_{st}$, $\hat{\eta}_{st,j}$, is the square root of the *j*th diagonal element of $\Sigma_{\hat{\eta}_{st}}$,

$$\sigma_{\hat{\eta}_{st,j}} = \sqrt{\sigma_{\hat{\eta}_{st},jj}}$$

Predicted value of the cumulative probability for category *j*

$$\hat{\gamma}_{st,j}=g^{-1}(\hat{\eta}_{st,j}), j=1,\ldots,J-1$$

with $\hat{\gamma}_{i,J}=1$.

Predicted category

$$c(\mathrm{x}_{st})=arg\max_{j}\hat{\pi}_{st,j},$$

where $\hat{\pi}_{st,j}=\hat{\gamma}_{st,j}-\hat{\gamma}_{st,j-1}$.

If there is a tie in determining the predicted category, the tie will be broken by choosing the category with the highest $N_j=\sum_{s=1}^{S}\sum_{t=1}^{T_s}f_{st}y_{st,j}$. If there is still a tie, the one with the lowest category number is chosen.

Approximate 100(1−α)% confidence intervals for the cumulative predicted probabilities

$$g^{-1}\left(\hat{\eta}_{st,j}\pm t_{\upsilon,\alpha/2}\hat{\sigma}_{\hat{\eta}_{st,j}}\right), j=1,\ldots,J-1,$$

If either endpoint in the argument is outside the valid range for the inverse link function, the corresponding confidence interval endpoint is set to a system missing value.

The degrees of freedom can be computed by the residual method or Satterthwaite method. For the residual method, $\upsilon=N-(J-1+p_x)$. For Satterthwaite's approximation, the **L** matrix is constructed by $(\mathrm{X}_{st,j},\mathrm{Z}_{st,j})$, where $\mathrm{X}_{st,j}$ and $\mathrm{Z}_{st,j}$ are the *j*th rows of X_{st} and Z_{st}, respectively, corresponding to the *j*th category. For example, the **L** matrix is $\left(1,0,\ldots,0,\ -\mathrm{x}_{st}^{\mathrm{T}},\quad \mathrm{z}_{st}^{\mathrm{T}}\right)_{1\times(J-1+p+r)}$ for the 1st category. The computation should then be similar to that described in "Method for computing degrees of freedom".

References

Agresti, A., J. G. Booth, and B. Caffo. 2000. Random-effects Modeling of Categorical Response Data. *Sociological Methodology*, 30, 27–80.

Diggle, P. J., P. Heagerty, K. Y. Liang, and S. L. Zeger. 2002. *The analysis of Longitudinal Data*, 2 ed. Oxford: Oxford University Press.

Fahrmeir, L., and G. Tutz. 2001. *Multivariate Statistical Modelling Based on Generalized Linear Models*, 2nd ed. New York: Springer-Verlag.

Hartzel, J., A. Agresti, and B. Caffo. 2001. Multinomial Logit Random Effects Models. *Statistical Modelling*, 1, 81–102.

Hedeker, D. 1999. Generalized Linear Mixed Models. In: *Encyclopedia of Statistics in Behavioral Science*, B. Everitt, and D. Howell, eds. London: Wiley, 729–738.

McCulloch, C. E., and S. R. Searle. 2001. *Generalized, Linear, and Mixed Models*. New York: John Wiley and Sons.

Skrondal, A., and S. Rabe-Hesketh. 2004. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Boca Raton, FL: Chapman & Hall/CRC.

Tuerlinckx, F., F. Rijmen, G. Molenberghs, G. Verbeke, D. Briggs, W. Van den Noortgate, M. Meulders, and P. De Boeck. 2004. Estimation and Software. In: *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach*, P. De Boeck, and M. Wilson, eds. New York: Springer-Verlag, 343–373.

Wolfinger, R., and M. O'Connell. 1993. Generalized Linear Mixed Models: A Pseudo-Likelihood Approach. *Journal of Statistical Computation and Simulation*, 4, 233–243.

Wolfinger, R., R. Tobias, and J. Sall. 1994. Computing Gaussian likelihoods and their derivatives for general linear mixed models. *SIAM Journal on Scientific Computing*, 15:6, 1294–1310.

GENLIN Algorithms

Generalized linear models (GZLM) and generalized estimating equations (GEE) are commonly used analytical tools for different types of data. Generalized linear models cover not only widely used statistical models, such as linear regression for normally distributed responses, logistic models for binary data, and log linear model for count data, but also many useful statistical models via its very general model formulation. However, the independence assumption prohibits application of generalized linear models to correlated data. Generalized estimating equations were developed to extend generalized linear models to accommodate correlated longitudinal data and clustered data.

Generalized Linear Models

Generalized linear models were first introduced by Nelder and Wedderburn (1972) and later expanded by McCullagh and Nelder (1989). The following discussion is based on their works.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 46-1
Notation

Notation	Description
n	Number of complete cases in the dataset. It is an integer and $n \geq 1$.
p	Number of parameters (including the intercept, if exists) in the model. It is an integer and $p \geq 1$.
p_X	Number of non-redundant columns in the design matrix. It is an integer and $p_X \geq 1$.
$\mathbf{y}$	$n \times 1$ dependent variable vector. The rows are the cases.
$\mathbf{r}$	$n \times 1$ vector of events for the binomial distribution; it usually represents the number of "successes." All elements are non-negative integers.
$\mathbf{m}$	$n \times 1$ vector of trials for the binomial distribution. All elements are positive integers and $m_i \geq r_i$, i=1,...,n.
$\boldsymbol{\mu}$	$n \times 1$ vector of expectations of the dependent variable.
$\boldsymbol{\eta}$	$n \times 1$ vector of linear predictors.
$\mathbf{X}$	$n \times p$ design matrix. The rows represent the cases and the columns represent the parameters. The ith row is $(x_{i1}, \ldots, x_{ip})T$, i=1,…, n with $x_{i1} = 1$ if the model has an intercept.
$\mathbf{O}$	$n \times 1$ vector of scale offsets. This variable can't be the dependent variable (**y**) or one of the predictor variables (**X**).
β	$p \times 1$ vector of unknown parameters. The first element in β is the intercept, if there is one.
$\boldsymbol{\omega}$	$n \times 1$ vector of scale weights. If an element is less than or equal to 0 or missing, the corresponding case is not used.
$\mathbf{f}$	$n \times 1$ vector of frequency counts. Non-integer elements are treated by rounding the value to the nearest integer. For values less than 0.5 or missing, the corresponding cases are not used.
N	Effective sample size. $N = \sum_{i=1}^{n} f_i$. If frequency count variable **f** is not used, $N = n$.

Model

A GZLM of **y** with predictor variables **X** has the form

$$\eta = g\left(\mathrm{E}\left(\mathbf{y}\right)\right) = \mathbf{X}\beta + \mathbf{O}, \quad \mathbf{y} \sim F$$

where **η** is the linear predictor; **O** is an offset variable with a constant coefficient of 1 for each observation; *g*(.) is the monotonic differentiable link function which states how the mean of **y**, $E\left(\mathrm{y}\right) = \mu$, is related to the linear predictor **η** ; *F* is the response probability distribution. Choosing different combinations of a proper probability distribution and a link function can result in different models.

Some combinations are well known models and have been provided in different IBM® SPSS® Statistics procedures. The following table lists these combinations and corresponding procedures.

Table 46-2
Distribution, link function, and corresponding procedure

Distribution	**Link function**	**Model**	**Procedure**
Normal	Identity	Linear regression	GLM, REGRESSION
Binomial	Logit	Logistic regression	LOGISTIC REGRESSION
Poisson	Log	Loglinear	GENLOG

In addition, GZLM also assumes y_i are independent for *i*=1,….,*n*. This is the main assumption which separates GZLM and GEE. Then for each observation, the model becomes

$$\eta_i = g\left(\mu_i\right) = x_i^{\mathrm{T}}\beta + o_i, \quad y_i \sim F$$

Notes

- **X** can be any combination of scale variables (covariates), categorical variables (factors), and interactions. The parameterization of **X** is the same as in the GLM procedure. Due to use of the over-parameterized model where there is a separate parameter for every factor effect level occurring in the data, the columns of the design matrix **X** are often dependent. Collinearity between scale variables in the data can also occur. To establish the dependencies in the design matrix, columns of $\mathbf{X}^{\mathrm{T}}\mathbf{\Psi}\mathbf{X}$, where $\mathbf{\Psi} = \mathrm{diag}(f_1\omega_1, \ldots f_n\omega_n)$,, are examined by using the sweep operator. When a column is found to be dependent on previous columns, the corresponding parameter is treated as redundant. The solution for redundant parameters is fixed at zero.
- When **y** is a binary dependent variable which can be character or numeric, such as "male"/"female" or 1/2, its values will be transformed to 0 and 1 with 1 typically representing a success or some other positive result. In this document, we assume to be modeling the probability of success. In this document, we assume that **y** has been transformed to 0/1 values and we always model the probability of success; that is, Prob(**y** = 1). Which original value should be transformed to 0 or 1 depends on what the reference category is. If the reference category is the last value (`REFERENCE=LAST` in the syntax), then the first category represents a success and we are modeling the probability of it. For example, if `REFERENCE=LAST` is used in the syntax, "male" in "male"/"female" and 2 in 1/2 are the last values (since "male" comes later in the dictionary than "female") and would be transformed to 0, and "female" and 1 would be transformed to 1 as we model the probability of them,

respectively. However, one way to change to model the probability of “male” and 2 instead is to specify `REFERENCE=FIRST` in the syntax. Note if original binary format is 0/1 and `REFERENCE=LAST` is specified, then 0 would be transformed to 1 and 1 to 0.

- When **r**, representing the number of successes (or number of 1s) and **m**, representing the number of trials, are used for the binomial distribution, the response is the binomial proportion **y** = **r**/**m**.

Multinomial Distribution

The response variable *y* is assumed to be ordinal; its values have an intrinsic ordering and correspond to consecutive integers from 1 to *J*. The design matrix *X* includes model predictors, but not an intercept. The following new notations are needed to define the model form:

Table 46-3
Notation

Notation	**Description**
J	The number of values for the ordinal response variable, $J \geq 1$.
ψ	$J - 1 \times 1$ vector of threshold parameters $\psi = (\psi_1, \psi_2, \ldots, \psi_{J-1})'$ and $\psi_1 < \psi_2 < \ldots \psi_{J-1}$.
β	$p \times 1$ vector of regression parameters associated with model predictors, $\beta = (\beta_1, \beta_2, \ldots, \beta_p)'$.
$\mathbf{B}$	$(J - 1 + p) \times 1$ vector of all parameters, $\mathbf{B} = \left(\psi', \beta'\right)'$
$\gamma_{i,j}$	Conditional cumulative response probability for category *j* given observed independent variable vector, $\gamma_{i,j} = P\left(y_i \leq j \mid \mathrm{x_i}\right)$
$\pi_{i,j}$	Conditional response probability for category *j* given observed independent variable vector, $\pi_{i,j} = P\left(y_i = j \mid \mathrm{x_i}\right)$ and $\pi_{i,j} = \gamma_{i,j} - \gamma_{i,j-1}$ for $j = 1, \ldots, J$.
$\eta_{i,j}$	Linear predictor value of case *i* for category *j*. It is related to $\gamma_{i,j}$ through a cumulative link function.

$$\eta_{i,j} = g\left(\gamma_{i,j}\right) = \psi_j - \mathrm{x}_i^T\beta + o_i, y_i \sim F.$$

Probability Distribution

GZLMs are usually formulated within the framework of the exponential family of distributions. The probability density function of the response *Y* for the exponential family can be presented as

$$f\left(y\right) = \exp\left\{\frac{y\theta - b\left(\theta\right)}{\phi/\omega} + c\left(y, \phi/\omega\right)\right\}$$

where θ is the canonical (natural) parameter, ϕ is the scale parameter related to the variance of *y* and ω is a known prior weight which varies from case to case. Different forms of *b*(θ) and *c*(*y*, ϕ/ω) will give specific distributions. In fact, the exponential family provides a notation that allows us to model both continuous and discrete (count, binary, and proportional) outcomes. Several are available including continuous ones: normal, inverse Gaussian, gamma; discrete ones: negative binomial, Poisson, binomial, ordinal multinomial; and a mixed distribution: Tweedie.

The mean and variance of *y* can be expressed as follows

$$E(y) = b'(\theta) = \mu$$

$$Var(y) = b''(\theta)\frac{\phi}{\omega} = V(\mu)\frac{\phi}{\omega}$$

where $b'(\theta)$ and $b''(\theta)$ denote the first and second derivatives of *b* with respect to θ, respectively; $V(\mu)$ is the variance function which is a function of μ.

In GZLM, the distribution of *y* is parameterized in terms of the mean (μ) and a scale parameter (ϕ) instead of the canonical parameter (θ). The following table lists the distribution of *y*, corresponding range of *y*, variance function (*V*(μ)), the variance of *y* (Var(*y*)), and the first derivative of the variance function $V'(\mu)$), which will be used later.

Table 46-4
Distribution, range and variance of the response, variance function, and its first derivative

Distribution	**Range of *y***	***V*(μ)**	**Var(*y*)**	***V'*(μ)**
Normal	$(-\infty,\infty)$	1	ϕ	0
Inverse Gaussian	$(0,\infty)$	μ^3	$\phi\mu^3$	$3\mu^2$
Gamma	$(0,\infty)$	μ^2	$\phi\mu^2$	2μ
Negative binomial	$0(1)\infty$	$\mu+k\mu^2$	$\mu+k\mu^2$	$1+2k\mu$
Poisson	$0(1)\infty$	μ	μ	1
Binomial(*m*)	$0(1)m/m$	$\mu(1-\mu)$	$\mu(1-\mu)/m$	$1-2\mu$
Tweedie(*q*)	$[0,\infty)$	μ^q	$\phi\mu^q$	$q\mu^{q-1}$
Multinomial	$1(1)J$	There are not simple forms for ordinal multinomial, but they are not needed for parameter estimation.		

Notes

- $0(1)z$ means the range is from 0 to *z* with increments of 1; that is, 0, 1, 2, …, *z*.
- For the binomial distribution, the binomial trial variable *m* is considered as a part of the weight variable ω.
- If a weight variable ω is presented, ϕ is replaced by ϕ/ω.
- For the negative binomial distribution, the ancillary parameter (*k*) can be user-specified or estimated by the maximum likelihood (ML) method. When $k = 0$, the negative binomial distribution reduces to the Poisson distribution. When $k = 1$, the negative binomial is the geometric distribution.
- The Tweedie class of distributions includes discrete, continuous and mixed densities as long as $q \le 0$ or $q \ge 1$, where *q* is the exponent in the variance function. Special cases include the normal ($q = 0$), Poisson ($q = 1$), gamma ($q = 2$) and inverse Gaussian ($q = 3$). Except for these special cases, the Tweedie distributions cannot be written in closed form. Here, we only consider the Tweedie distributions for $1 < q < 2$, which can be represented as Poisson mixtures of gamma distributions and are mixed distributions with mass at zero and with support on the non-negative real values. These distributions are sometimes called “compound Poisson”, “compound gamma” and “Poisson-gamma” distributions. *q* must be user-specified.

Scale parameter handling. The expressions for $V(\mu)$ and Var(y) for continuous distributions and Tweedie distributions include the scale parameter ϕ which can be used to scale the relationship of the variance and mean (Var(y) and μ). Since it is usually unknown, there are three ways to fit the scale parameter:

1. It can be estimated with β jointly by maximum likelihood method.
2. It can be set to a fixed positive value.
3. It can be specified by the deviance or Pearson chi-square. For more information, see the topic "Goodness-of-Fit Statistics".

On the other hand, discrete distributions do not have this extra parameter (it is theoretically equal to one). Because of it, the variance of y might not be equal to the nominal variance in practice (especially for Poisson and binomial because the negative binomial has an ancillary parameter k). A simple way to adjust this situation is to allow the variance of y for discrete distributions to have the scale parameter as well, but unlike continuous distributions, it can't be estimated by the ML method. So for discrete distributions, there are two ways to obtain the value of ϕ:

1. It can be specified by the deviance or Pearson chi-square.
2. It can be set to a fixed positive value.

To ensure the data fit the range of response for the specified distribution, we follow the rules:

- For the gamma or inverse Gaussian distributions, values of **y** must be real and greater than zero. If a value of **y** is less than or equal to 0 or missing, the corresponding case is not used.
- For the negative binomial and Poisson distributions, values of **y** must be integer and non-negative. If a value of **y** is non-integer, less than 0 or missing, the corresponding case is not used.
- For the binomial distribution and if the response is in the form of a single variable, **y** must have only two distinct values. If **y** has more than two distinct values, the algorithm terminates in an error.
- For the binomial distribution and the response is in the form of ratio of two variables denoted events/trials, values of **r** (the number of events) must be nonnegative integers, values of **m** (the number of trials) must be positive integers and $m_i \geq r_i$, $\forall\ i$. If a value of **r** is not integer, less than 0, or missing, the corresponding case is not used. If a value of **m** is not integer, less than or equal to 0, less than the corresponding value of **r**, or missing, the corresponding case is not used.
- For the Tweedie distributions, values of **y** must be zero or positive real. If a value of **y** is less than 0 or missing, the corresponding case is not used.

The ML method will be used to estimate β and possibly ϕ for continuous distributions and the Tweedie distribution, or k for the negative binomial. The kernels of the log-likelihood function (ℓ_k) and the full log-likelihood function (ℓ), which will be used as the objective function for parameter estimation, are listed for each distribution in the following table. Using ℓ or ℓ_k won't affect the parameter estimation, but the selection will affect the calculation of information criteria. For more information, see the topic "Goodness-of-Fit Statistics".

Table 46-5
The log-likelihood function for probability distribution

Distribution	**ℓ_k and ℓ**
Normal	$\ell_k = \sum_{i=1}^{n} -\frac{f_i}{2}\left\{\frac{\omega_i(y_i-\mu_i)^2}{\phi} + \ln\left(\frac{\phi}{\omega_i}\right)\right\}$ $\ell = \ell_k + \sum_{i=1}^{n} -\frac{f_i}{2}\{\ln(2\pi)\}$
Inverse Gaussian	$\ell_k = \sum_{i=1}^{n} -\frac{f_i}{2}\left\{\frac{\omega_i(y_i-\mu_i)^2}{\phi y_i \mu_i^2} + \ln\left(\frac{\phi y_i^3}{\omega_i}\right)\right\}$ $\ell = \ell_k + \sum_{i=1}^{n} -\frac{f_i}{2}\{\ln(2\pi)\}$
Gamma	$\ell_k = \sum_{i=1}^{n} f_i\left\{\frac{\omega_i}{\phi}\ln\left(\frac{\omega_i y_i}{\phi\mu_i}\right) - \frac{\omega_i y_i}{\phi\mu_i} - \ln\left(\Gamma\left(\frac{\omega_i}{\phi}\right)\right)\right\}$ $\ell = \ell_k + \sum_{i=1}^{n} f_i\{-\ln(y_i)\}$
Negative binomial	$\ell_k = \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{y_i\ln(k\mu_i) - (y_i+1/k)\ln(1+k\mu_i) + \ln(\Gamma(y_i+1/k)) - \ln(\Gamma(1/k))\}$ $\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{-\ln(\Gamma(y_i+1))\}$
Poisson	$\ell_k = \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{y_i\ln(\mu_i) - \mu_i\}$ $\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\{-\ln(y_i!)\}$
Binomial(*m*)	$\ell_k = \sum_{i=1}^{n} f_i\frac{\omega_i^*}{\phi}\{y_i\ln(\mu_i) + (1-y_i)\ln(1-\mu_i)\}$ $\ell = \ell_k + \sum_{i=1}^{n} f_i\frac{\omega_i}{\phi}\left\{\ln\begin{pmatrix} m_i \\ r_i \end{pmatrix}\right\}$,where $\begin{pmatrix} m_i \\ r_i \end{pmatrix} = \frac{m_i!}{r_i!(m_i-r_i)!}$
Tweedie	$\ell_k = \sum_{i=1}^{n} f_i\left\{\ln(V_i) + \frac{\omega_i}{\phi}\left(\frac{y_i\mu_i^{1-q}}{(1-q)} - \frac{\mu_i^{2-q}}{(2-q)}\right)\right\}$ $\ell = \ell_k + \sum_{i=1}^{n} f_i\{-\ln(y_i)\}$
Multinomial	$\ell = \ell_k = \sum_{i=1}^{n}\frac{f_i\omega_i}{\phi}\sum_{j=1}^{J} y_{i,j}\ln(\pi_{i,j})$,where $y_{i,j} = \begin{cases} 1 \text{ if } y_i = j \\ 0 \text{ otherwise} \end{cases}$.

When an individual $y = 0$ for the negative binomial, Poisson or Tweedie distributions and $y = 0$ or 1 for the binomial distribution, a separate value of the log-likelihood is given. Let $\ell_{k,i}$ be the log-likelihood value for individual case *i* when $y_i = 0$ for the negative binomial, Poisson and Tweedie and 0/1 for the binomial. The full log-likelihood for *i* is equal to the kernel of the log-likelihood for *i*; that is, $\ell_i = \ell_{k,i}$.

Table 46-6
Log-likelihood

Distribution	$\boldsymbol{\ell_{k,i}}$
Negative binomial	$\ell_{k,i} = -f_i \frac{\omega_i}{\phi} \frac{\ln(1+k\mu_i)}{k}$ if $y_i = 0$
Poisson	$\ell_{k,i} = -f_i \frac{\omega_i}{\phi} \mu_i$ if $y_i = 0$
Binomial(*m*)	$\ell_{k,i} = \begin{cases} f_i \frac{\omega_i}{\phi} \ln(1-\mu_i) & \text{if } y_i = 0 \\ f_i \frac{\omega_i}{\phi} \ln(\mu_i) & \text{if } y_i = 1 \end{cases}$
Tweedie	$-f_i \frac{\omega_i}{\phi} \frac{\mu_i^{2-q}}{(2-q)}$ if $y_i = 0$

- $\Gamma(z)$ is the gamma function and $\ln(\Gamma(z))$ is the log-gamma function (the logarithm of the gamma function), evaluated at z.
- For the negative binomial distribution, the scale parameter is still included in ℓ_k for flexibility, although it is usually set to 1.
- For the negative binomial distribution, y_i must be a non-negative integer, which means $\Gamma(y_i + 1) = y_i!$ and $\ell = \ell_k + \sum_{i=1}^{n} f_i \frac{\omega_i}{\phi}\{-\ln(y_i!)\}$. In addition, ℓ_k can be written as $\sum_{i=1}^{n} f_i \frac{\omega_i}{\phi}\left\{ y_i \ln(k\mu_i) - \left(y_i + \frac{1}{k}\right)\ln(1+k\mu_i) + \sum_{j=1}^{y_i-1} \ln(1+kj) \right\}$ because $\frac{\Gamma(y_i+1/k)}{\Gamma(1/k)} = \prod_{j=0}^{y_i-1}(j+1/k)$. Some potential computational problems can be avoided by using this form. See Cameron and Trivedi (1998, P. 72).
- For the binomial distribution (**r/m**), the scale weight variable becomes $\omega_i^* = \omega_i m_i$ in ℓ_k; that is, the binomial trials variable **m** is regarded as a part of the weight. However, the scale weight in the extra term of ℓ is still ω_i.
- V_i in the Tweedie distribution is an infinite series as follows:

  $$V_i = \sum_{j=1}^{\infty} V_{ij}$$

  and

  $$V_{ij} = \frac{\omega_i^{j(1-\alpha)} y_i^{-j\alpha} (q-1)^{j\alpha}}{\phi^{j(1-\alpha)} (2-q)^j \Gamma(-j\alpha) j!}$$

  where $\alpha = \frac{2-q}{1-q}$ and. To evaluate the infinite summation for V_i, the value of *j* is determined for which V_{ij} reaches a maximum and sum the necessary terms of the series in that region. The method proposed by Dunn and Smyth (2005) is adopted here.

Link Function

The following tables list the form, inverse form, range of $\hat{\mu}$, and first and second derivatives for each link function.

Table 46-7
Link function name, form, inverse of link function, and range of the predicted mean

Link function	**η=g(μ)**	**Inverse μ=g⁻¹(η)**	**Range of $\hat{\mu}$**
Identity	μ	η	$\hat{\mu} \in R$
Log	$\ln(\mu)$	$\exp(\eta)$	$\hat{\mu} \geq 0$
Logit	$\ln\left(\frac{\mu}{1-\mu}\right)$	$\frac{\exp(\eta)}{1+\exp(\eta)}$	$\hat{\mu} \in [0,1]$
Probit	$\Phi^{-1}(\mu)$, where $\Phi(\xi) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\xi} e^{-z^2/2}dz$	$\Phi(\eta)$	$\hat{\mu} \in [0,1]$
Complementary log-log	$\ln(-(\ln(1-\mu))$	$1-\exp(-\exp(\eta))$	$\hat{\mu} \in [0,1]$
Power(α) $\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\mu^{\alpha}\\ \ln(\mu)\end{cases}$	$\begin{cases}\eta^{1/\alpha}\\ \exp(\eta)\end{cases}$	$\begin{cases}\hat{\mu} \in R \text{ if } \alpha \text{ or} 1/\alpha \text{ is odd integer}\\ \hat{\mu} \geq 0 \text{ otherwise}\end{cases}$
Log-complement	$\ln(1-\mu)$	$1-\exp(\eta)$	$\hat{\mu} \leq 1$
Negative log-log	$-\ln(-\ln(\mu))$	$\exp(-\exp(-\eta))$	$\hat{\mu} \in [0,1]$
Negative binomial	$\ln\left(\frac{\mu}{\mu+\frac{1}{k}}\right)$	$\frac{\exp(\eta)}{k(1-\exp(\eta))}$	$\hat{\mu} \geq 0$
Odds power(α) $\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\frac{(\mu/(1-\mu))^{\alpha}-1}{\alpha}\\ \ln\left(\frac{\mu}{1-\mu}\right)\end{cases}$	$\begin{cases}\frac{(1+\alpha\eta)^{1/\alpha}}{1+(1+\alpha\eta)^{1/\alpha}}\\ \frac{\exp(\eta)}{1+\exp(\eta)}\end{cases}$	$\hat{\mu} \in [0,1]$

Note: In the power link function, if |α| < 2.2e-16, α is treated as 0.

Table 46-8
The first and second derivatives of link function

Link function	**First derivative $g'(\mu) = \frac{\partial\eta}{\partial\mu} = \Delta$**	**Second derivative $g''(\mu) = \frac{\partial^2\eta}{\partial\mu^2}$**
Identity	1	0
Log	$\frac{1}{\mu}$	$-\Delta^2$
Logit	$\frac{1}{\mu(1-\mu)}$	$\Delta^2(2\mu-1)$
Probit	$\frac{1}{\phi\left(\Phi^{-1}(\mu)\right)}$, where $\phi(z) = \frac{1}{\sqrt{2\pi}}e^{-z^2/2}$	$\Delta^2\Phi^{-1}(\mu)$
Complementary log-log	$\frac{1}{(\mu-1)\ln(1-\mu)}$	$-\Delta^2(1+\ln(1-\mu))$
Power(α) $\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\alpha\mu^{\alpha-1}\\ \frac{1}{\mu}\end{cases}$	$\begin{cases}\Delta\frac{\alpha-1}{\mu}\\ -\Delta^2\end{cases}$
Log-complement	$\frac{-1}{1-\mu}$	$-\Delta^2$
Negative log-log	$\frac{-1}{\mu\ln(\mu)}$	$\Delta^2(1+\ln(\mu))$
Negative binomial	$\frac{1}{\mu+k\mu^2}$	$-\Delta^2(1+2k\mu)$
Odds power(α) $\begin{cases}\alpha \neq 0\\ \alpha = 0\end{cases}$	$\begin{cases}\frac{\mu^{\alpha-1}}{(1-\mu)^{\alpha+1}}\\ \frac{1}{\mu(1-\mu)}\end{cases}$	$\begin{cases}\Delta\left(\frac{\alpha-1}{\mu}+\frac{\alpha+1}{1-\mu}\right)\\ \Delta^2(2\mu-1)\end{cases}$

Table 46-9
Cumulative Link Function Name, Form, Inverse Form and Range of the Predicted Cumulative Probability

Link function	**η=*g*(γ)**	**Inverse γ=*g*$^{-1}$(η)**	**Range of $\hat{\gamma}$**
Cumulative logit	$\ln\left(\frac{\gamma}{1-\gamma}\right)$	$\frac{\exp(\eta)}{1+\exp(\eta)}$	$\hat{\gamma} \in [0,1]$
Cumulative probit	$\Phi^{-1}(\gamma)$, where $\Phi(\xi) = \frac{1}{\sqrt{2\pi}}{}^{/2}\int_{-\infty}^{\xi} dz\ e^{-z^2}$	$\Phi(\eta)$	$\hat{\gamma} \in [0,1]$
Cumulative complementary log-log	$\ln(-\ln(1-\gamma))$	$1-\exp(-\exp(\eta))$	$\hat{\gamma} \in [0,1]$
Cumulative negative log-log	$-\ln(-\ln(\gamma))$	$\exp(-\exp(-\eta))$	$\hat{\gamma} \in [0,1]$
Cumulative Cauchit	$\tan(\pi(\gamma-0.5))$	$0.5+\arctan(\eta)/\pi$	$\hat{\gamma} \in [0,1]$

Note: π in the formulae is the number, not the response probability.

Table 46-10
The Inverse First and Second Derivatives of Cumulative Link Function

Link function	**Inverse first derivative** $\frac{\partial\gamma}{\partial\eta} = \Delta$	**Inverse second derivative** $\frac{\partial^2\gamma}{\partial\eta^2}$
Cumulative logit	$\gamma(1-\gamma)$	$\Delta(1-2\gamma)$
Cumulative probit	$\phi(\Phi^{-1}(\gamma))$, where $\phi(z) = \frac{1}{\sqrt{2\pi}}e^{-z^2/2}$	$-\Delta \times \Phi^{-1}(\gamma)$
Cumulative complementary log-log	$(\gamma-1)\ln(1-\gamma)$	$\Delta(1+\ln(1-\gamma))$
Cumulative negative log-log	$-\gamma\ln(\gamma)$	$-\Delta(1+\ln(\gamma))$
Cumulative Cauchit	$\cos^2(\pi(\gamma-0.5))/\pi$	$\Delta \times \sin(2\pi\gamma)$

When the canonical parameter is equal to the linear predictor, $\theta = \eta$, then the link function is called the **canonical link function**. Although the canonical links lead to desirable statistical properties of the model, particularly in small samples, there is in general no a priori reason why the systematic effects in a model should be additive on the scale given by that link. The canonical link functions for probability distributions are given in the following table.

Table 46-11
Canonical and default link functions for probability distributions

Distribution	**Canonical link function**
Normal	Identity
Inverse Gaussian	Power(−2)
Gamma	Power(−1)
Negative binomial	Negative binomial
Poisson	Log
Binomial	Logit
Tweedie	Power(1−*q*)
Multinomial	Cumulative logit

Estimation

Having selected a particular model, it is required to estimate the parameters and to assess the precision of the estimates.

Parameter estimation

The parameters are estimated by maximizing the log-likelihood function (or the kernel of the log-likelihood function) from the observed data. Let **s** be the first derivative (gradient) vector of the log-likelihood with respect to each parameter, then we wish to solve

$$\mathbf{s} = \left[\frac{\partial \ell}{\partial \beta}\right]_{p\times 1} = 0$$

or, for the multinomial distribution,

$$\mathrm{s} = \left[\frac{\partial \ell}{\partial \mathrm{B}}\right]_{(J-1+p)\times 1} = \begin{bmatrix} \frac{\partial \ell}{\partial \Psi} \\ \frac{\partial \ell}{\partial \beta} \end{bmatrix} = 0$$

In general, there is no closed form solution except for a normal distribution with identity link function, so estimates are obtained numerically via an iterative process. A Newton-Raphson and/or Fisher scoring algorithm is used and it is based on a linear Taylor series approximation of the first derivative of the log-likelihood.

First Derivatives

If the scale parameter ϕ is not estimated by the ML method, **s** is a p×1 vector with the form:

$$\mathbf{s} = \sum_{i=1}^{n} \frac{f_i \omega_i (y_i - \mu_i)}{\phi V(\mu_i)\, g'(\mu_i)} \cdot x_i = \frac{1}{\phi} \sum_{i=1}^{n} \frac{f_i \omega_i (y_i - \mu_i)}{V(\mu_i)\, g'(\mu_i)} \cdot x_i$$

where μ_i, $V(\mu_i)$ and $g'(\mu_i)$ are defined in Table 46-7 “Link function name, form, inverse of link function, and range of the predicted mean”, Table 46-4 “Distribution, range and variance of the response, variance function, and its first derivative”, and Table 46-8 “The first and second derivatives of link function”, respectively.

If the scale parameter ϕ is estimated by the ML method, it is handled by searching for ϕ) since ϕ is required to be greater than zero. Similarly, if the ancillary parameter *k* for negative binomial is estimated by the ML method, it is still handled by searching for ln(*k*) since *k* is also required to be greater than zero.

Let $\tau = \phi$) so $\phi = \exp(\tau)$ (or $\tau = \ln(k)$ and $k = \exp(\tau)$ for negative binomial), then **s** is a (p+1)×1 vector with the following form

$$\mathbf{s} = \begin{bmatrix} \frac{\partial \ell}{\partial \beta} \\ \frac{\partial \ell}{\partial \tau} \end{bmatrix}_{(p+1)\times 1} = \begin{bmatrix} \frac{1}{\exp(\tau)} \sum_{i=1}^{n} \frac{f_i \omega_i (y_i - \mu_i)}{V(\mu_i)\, g'(\mu_i)} \quad x_i \\ \frac{\partial \ell}{\partial \tau} \end{bmatrix}$$

where $\partial\ell/\partial\beta$ is the same as the above with ϕ is replaced with exp(τ) (though for negative binomial, φ is not replaced), $\partial\ell/\partial\tau$ has a different form depending on the distribution as follows:

Table 46-12
The 1st derivative functions w.r.t. the scale parameter for probability distributions

Distribution	$\frac{\partial \ell}{\partial \tau}$
Normal	$\sum_{i=1}^{n} \frac{f_i}{2} \left\{ \frac{\omega_i (y_i - \mu_i)^2}{\exp(\tau)} - 1 \right\}$
Inverse Gaussian	$\sum_{i=1}^{n} \frac{f_i}{2} \left\{ \frac{\omega_i (y_i - \mu_i)^2}{\exp(\tau)\, y_i \mu_i^2} - 1 \right\}$
Gamma	$\sum_{i=1}^{n} - \frac{f_i \omega_i}{\exp(\tau)} \left\{ \ln\left(\frac{\omega_i y_i}{\exp(\tau)\,\mu_i} \right) + \left(1 - \frac{y_i}{\mu_i} \right) - \psi\left(\frac{\omega_i}{\exp(\tau)} \right) \right\}$
Negative Binomial	$\sum_{i=1}^{n} \frac{f_i \omega_i}{\phi \exp(\tau)} \left\{ a_i + \ln(1 + \exp(\tau)\,\mu_i) - \psi\left(y_i + \frac{1}{\exp(\tau)} \right) + \psi\left(\frac{1}{\exp(\tau)} \right) \right\}$ where for all appropriate link functions other than negative binomial link function, $a_i = \frac{\exp(\tau)(y_i - \mu_i)}{(1 + \exp(\tau)\mu_i)}$ and for the negative binomial link function, $a_i = 0$
Tweedie	$\sum_{i=1}^{n} f_i \frac{\partial \ell_i}{\partial \tau}$, where $\frac{\partial \ell_i}{\partial \tau} = \begin{cases} \frac{\omega_i \mu_i^{2-q}}{\exp(\tau)(2-q)} \text{for } y_i = 0 \\ \frac{\frac{\partial V_i}{\partial \tau}}{V_i} - \frac{\omega_i y_i \mu_i^{1-q}}{\exp(\tau)(1-q)} + \frac{\omega_i \mu_i^{2-q}}{\exp(\tau)(2-q)} \text{for } y_i > 0 \end{cases}$

Note: $\psi(z)$ is a digamma function, which is the derivative of logarithm of a gamma function, evaluated at *z*; that is, $\psi(z) = \frac{\partial \ln(\Gamma(z))}{\partial z} = \frac{\Gamma'(z)}{\Gamma(z)}$.

As mentioned above, for normal distribution with identity link function which is a classical linear regression model, there is a closed form solution for both β and τ, so no iterative process is needed. The solution for β, after applying the SWEEP operation in GLM procedure, is

$$\hat{\beta} = \left(\sum_{i=1}^{n} f_i \omega_i \mathrm{x}_i^{\mathrm{T}} \mathrm{x}_i \right)^{-} \left(\sum_{i=1}^{n} f_i \omega_i \mathrm{x}_i^{\mathrm{T}} (y_i - o_i) \right) = \left(\mathrm{X}^{\mathrm{T}} \Psi \mathrm{X} \right)^{-} \left(\mathrm{X}^{\mathrm{T}} \Psi (\mathbf{y} - \mathbf{o}) \right),$$

where $\Psi = \text{diag}(f_1\omega_1, \dots f_n\omega_n)$ and $(Z)^-$ is the generalized inverse of a matrix **Z**. If the scale parameter ϕ is also estimated by the ML method, the estimate of τ is

$$\hat{\tau} = \ln\left(\hat{\phi}\right) = \ln\left(\frac{1}{N}\sum_{i=1}^{n} f_i\omega_i\left(y_i - x_i^{T}\hat{\beta} - o_i\right)^2\right).$$

For the ordinal multinomial model:

$$s = \left[\frac{\partial \ell}{\partial \psi_1}, \cdots, \frac{\partial \ell}{\partial \psi_{J-1}}, \frac{\partial \ell}{\partial \beta_1}, \cdots, \frac{\partial \ell}{\partial \beta_p}\right]^T,$$

where

$$\frac{\partial \ell}{\partial \psi_j} = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\left(\frac{y_{i,j}}{\pi_{i,j}} - \frac{y_{i,j+1}}{\pi_{i,j+1}}\right), j = 1, \dots, J-1$$

$$\frac{\partial \ell}{\partial \beta_t} = -\sum_{i=1}^{n}\sum_{j=1}^{J} \frac{f_i\omega_i}{\phi}\left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right)\frac{y_{i,j}}{\pi_{i,j}} x_{it}, t = 1, \dots, p,$$

and

$$\pi_{i,j} = \gamma_{i,j} - \gamma_{i,j-1} \text{ for } j = 1, \dots, J$$

$$\gamma_{i,j} = \begin{cases} 0 & j = 0 \\ g^{-1}\left(\psi_j - x_i^{T}\beta + o_i\right) & j = 1, \dots, J-1 \\ 1 & j = J \end{cases},$$

Note: if $\partial\gamma_{i,j} = 0$ or $\partial\gamma_{i,j} = 1$ then $\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} = 0$ for all cumulative link functions.

Second Derivatives

Let **H** be the second derivative (Hessian) matrix. If the scale parameter is not estimated by the ML method, **H** is a *p*×*p* matrix with the following form

$$\mathbf{H} = \left[\frac{\partial^2 \ell}{\partial \beta \partial \beta^{T}}\right]_{p\times p} = -\mathbf{X}^{T}\mathbf{W}\mathbf{X}$$

where **W** is an *n*×*n* diagonal matrix. There are two definitions for **W** depending on which algorithm is used: $\mathbf{W}_e$ for Fisher scoring and $\mathbf{W}_o$ for Newton-Raphson. The *i*th diagonal element for $\mathbf{W}_e$ is

$$w_{e,i} = \frac{f_i\omega_i}{\phi} \cdot \frac{1}{V(\mu_i)\left(g'(\mu_i)\right)^2},$$

and the *i*th diagonal element for $\mathbf{W}_o$ is

$$w_{o,i} = w_{e,i} + \frac{f_i \omega_i}{\phi}(y_i - \mu_i) \cdot \frac{V(\mu_i) g''(\mu_i) + V'(\mu_i) g'(\mu_i)}{(V(\mu_i))^2 (g'(\mu_i))^3},$$

where $V'(\mu_i)$ and $g''(\mu_i)$ are defined in Table 46-4 “Distribution, range and variance of the response, variance function, and its first derivative” and Table 46-8 “The first and second derivatives of link function”, respectively. Note the expected value of $\mathbf{W}_o$ is $\mathbf{W}_e$ and when the canonical link is used for the specified distribution, then $\mathbf{W}_o = \mathbf{W}_e$.

If the scale parameter is estimated by the ML method, **H** becomes a $(p+1)\times(p+1)$ matrix with the form

$$\mathbf{H} = \begin{bmatrix} \frac{\partial^2 \ell}{\partial \beta \partial \beta^T} & \frac{\partial^2 \ell}{\partial \beta \partial \tau} \\ \frac{\partial^2 \ell}{\partial \tau \partial \beta^T} & \frac{\partial^2 \ell}{\partial \tau^2} \end{bmatrix}_{(p+1)\times(p+1)}$$

where $\partial^2 \ell / \partial \beta \partial \tau$ is a $p\times 1$ vector and $\partial^2 \ell / \partial \tau \partial \beta^T$ is a $1\times p$ vector and the transpose of $\partial^2 \ell / \partial \beta \partial \tau$. For all three continuous distributions:

$$\frac{\partial^2 \ell}{\partial \beta \partial \tau} = \sum_{i=1}^{n} -\frac{f_i \omega_i (y_i - \mu_i)}{\exp(\tau) V(\mu_i) g'(\mu_i)} \quad \mathrm{x}_i = -\frac{\partial \ell}{\partial \beta}.$$

The forms of $\partial^2 \ell / \partial \beta \partial \tau$ for negative binomial are as follows depending on the link functions:

For all appropriate link functions other than negative binomial link function,

$$\frac{\partial^2 \ell}{\partial \beta \partial \tau} = \sum_{i=1}^{n} -\frac{f_i \omega_i \exp(\tau)(y_i - \mu_i)}{\phi (1 + \exp(\tau) \mu_i)^2 g'(\mu_i)} \quad \mathrm{x}_i;$$

for the negative binomial link function,

$$\frac{\partial^2 \ell}{\partial \beta \partial \tau} = \sum_{i=1}^{n} \frac{f_i \omega_i \mu_i}{\phi} \quad \mathrm{x}_i.$$

The forms of $\partial^2 \ell / \partial \tau^2$ are listed in the following table.

Table 46-13
The second derivative functions w.r.t. the scale parameter for probability distributions

Distribution	$\frac{\partial^2 \ell}{\partial \tau^2}$
Normal	$\sum_{i=1}^{n} -\frac{f_i \omega_i}{2 \exp(\tau)} (y_i - \mu_i)^2$
Inverse Gaussian	$\sum_{i=1}^{n} -\frac{f_i \omega_i}{2 \exp(\tau) y_i \mu_i^2} (y_i - \mu_i)^2$

Distribution	$\frac{\partial^2 \ell}{\partial \tau^2}$
Gamma	$\sum_{i=1}^{n} \frac{f_i \omega_i}{\exp(\tau)} \left\{ \ln\left(\frac{\omega_i y_i}{\exp(\tau)\mu_i}\right) + \left(2 - \frac{y_i}{\mu_i}\right) - \psi\left(\frac{\omega_i}{\exp(\tau)}\right) - \frac{\omega_i}{\exp(\tau)} \psi'\left(\frac{\omega_i}{\exp(\tau)}\right) \right\}$
Negative Binomial	$\sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left\{ \begin{array}{l} a_i - \frac{1}{\exp(\tau)} \ln(1+\exp(\tau)\mu_i) + \\ \frac{1}{\exp(\tau)} \left[\psi\left(y_i + \frac{1}{\exp(\tau)}\right) - \psi\left(\frac{1}{\exp(\tau)}\right)\right] + \\ \frac{1}{\exp(2\tau)} \left[\psi'\left(y_i + \frac{1}{\exp(\tau)}\right) - \psi'\left(\frac{1}{\exp(\tau)}\right)\right] \end{array} \right\};$ where for all appropriate link functions other than negative binomial link function, $a_i = \frac{-y_i \exp(\tau)\mu_i + \mu_i + 2\exp(\tau)\mu_i^2}{(1+\exp(\tau)\mu_i)^2}$ and for the negative binomial link function, $a_i = 0$
Tweedie	$\sum_{i=1}^{n} f_i \frac{\partial^2 \ell_i}{\partial \tau^2},$ where $\frac{\partial^2 \ell_i}{\partial \tau^2} = \begin{cases} -\frac{\omega_i \mu_i^{2-q}}{\exp(\tau)(2-q)} & \text{for } y_i = 0 \\ \frac{\frac{\partial^2 V_i}{\partial \tau^2}}{V_i} - \left(\frac{\frac{\partial V_i}{\partial \tau}}{V_i}\right)^2 + \frac{\omega_i y_i \mu_i^{1-q}}{\exp(\tau)(1-q)} - \frac{\omega_i \mu_i^{2-q}}{\exp(\tau)(2-q)} & \text{for } y_i > 0 \end{cases}$

Note: $\psi'(z)$ is a trigamma function, which is the derivative of $\psi(z)$, evaluated at *z*.

$\frac{\partial V_i}{\partial \tau} = (\alpha - 1)\sum_{j=1}^{\infty} j V_{ij}$ and the evaluation of it is similar to that of the series $V_i = \sum_{j=1}^{\infty} V_{ij}$.

For the ordinal multinomial model:

$$\mathbf{H} = \left[\frac{\partial^2 \ell}{\partial \mathrm{B} \partial \mathrm{B}^T}\right]_{(J-1+p)\times(J-1+p)} = \begin{bmatrix} \frac{\partial^2 \ell}{\partial \psi \partial \psi^T} & \frac{\partial^2 \ell}{\partial \psi \partial \beta^T} \\ \frac{\partial^2 \ell}{\partial \beta \partial \psi^T} & \frac{\partial^2 \ell}{\partial \beta \partial \beta^T} \end{bmatrix}.$$

The elements of **H** have two forms: (1) the expected first derivatives of the estimating equation **s** which is applied to Fisher scoring and (2) the first derivatives of the estimating equation s which is applied to Newton Raphson.

Expected second derivatives have the following expressions:

$$\frac{\partial^2 \ell}{\partial \psi_{j-1} \partial \psi_j} = \sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}} \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} \frac{1}{\pi_{i,j}}, j = 2, \ldots, J-1,$$

$$\frac{\partial^2 \ell}{\partial \psi_j^2} = -\sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\right)^2 \left(\frac{1}{\pi_{i,j}} + \frac{1}{\pi_{i,j+1}}\right), j = 1, \ldots, J-1,$$

$$\frac{\partial^2 \ell}{\partial \psi_l \partial \psi_j} = 0, \text{ for } |l-j| > 1,$$

$$\frac{\partial^2 \ell}{\partial \psi_j \partial \beta_t} = \sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left[\left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right) \frac{1}{\pi_{i,j}} - \left(\frac{\partial \gamma_{i,j+1}}{\partial \eta_{i,j+1}} - \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\right) \frac{1}{\pi_{i,j+1}}\right] \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} x_{it},$$
$$j = 1, \ldots, J-1, t = 1, \ldots, p.$$

$$\frac{\partial^2 \ell}{\partial \beta_t \partial \beta_u} = -\sum_{i=1}^{n} \sum_{j=1}^{J} \frac{f_i \omega_i}{\phi} \left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right)^2 \frac{1}{\pi_{i,j}} x_{it} x_{iu}, t, u = 1, \ldots, p.$$

Second derivatives have the following expressions:

$$\frac{\partial^2 \ell}{\partial \psi_{j-1} \partial \psi_j} = \sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}} \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} \frac{y_{i,j}}{\pi_{i,j}^2}, j = 2, \ldots, J-1,$$

$$\frac{\partial^2 \ell}{\partial \psi_j^2} = \sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left[\frac{\partial^2 \gamma_{i,j}}{\partial \eta_{i,j}^2} \left(\frac{y_{i,j}}{\pi_{i,j}} - \frac{y_{i,j+1}}{\pi_{i,j+1}}\right) - \left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\right)^2 \left(\frac{y_{i,j}}{\pi_{i,j}^2} + \frac{y_{i,j+1}}{\pi_{i,j+1}^2}\right)\right], j = 1, \ldots, J-1,$$

$$\frac{\partial^2 \ell}{\partial \psi_l \partial \psi_j} = 0, \text{ for } |l-j| > 1,$$

$$\frac{\partial \ell}{\partial \psi_j \partial \beta_t} = -\sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left[\frac{\partial^2 \gamma_{i,j}}{\partial \eta_{i,j}^2} \pi_{i,j} - \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} \left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right)\right] \frac{y_{i,j}}{\pi_{i,j}^2} x_{it} +$$
$$\sum_{i=1}^{n} \frac{f_i \omega_i}{\phi} \left[\frac{\partial^2 \gamma_{i,j}}{\partial \eta_{i,j}^2} \pi_{i,j+1} - \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} \left(\frac{\partial \gamma_{i,j+1}}{\partial \eta_{i,j+1}} - \frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\right)\right] \frac{y_{i,j+1}}{\pi_{i,j+1}^2} x_{it},$$
$$j = 1, \ldots, J-1, t = 1, \ldots, p,$$

$$\frac{\partial \ell}{\partial \beta_t \partial \beta_u} = \sum_{i=1}^{n} \sum_{j=1}^{J} \frac{f_i \omega_i}{\phi} \left[\left(\frac{\partial^2 \gamma_{i,j}}{\partial \eta_{i,j}^2} - \frac{\partial^2 \gamma_{i,j-1}}{\partial \eta_{i,j-1}^2}\right) \pi_{i,j} - \left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right)^2\right] \frac{y_{i,j}}{\pi_{i,j}^2} x_{it} x_{iu},$$
$$t, u = 1, \ldots, p,$$

Iterations

An iterative process to find the solution for β (which might include ϕ, *k* for negative binomial or Ψ for multinomial) is based on Newton-Raphson (for all iterations), Fisher scoring (for all iterations) or a hybrid method. The hybrid method consists of applying Fisher scoring steps for a specified number of iterations before switching to Newton-Raphson steps. Newton-Raphson performs well if the initial values are close to the solution, but the hybrid method can be used to improve the algorithm's robustness from bad initial values. Apart from improved robustness, Fisher scoring is faster due to the simpler form of the Hessian matrix.

The following notation applies to the iterative process:

Table 46-14
Notation

Notation	Description
I	Starting iteration for checking complete separation and quasi-complete separation. It must be 0 or a positive integer. This criterion is not used if the value is 0.
J	The maximum number of steps in step-halving method. It must be a positive integer.
K	The first number of iterations using Fisher scoring, then switching to Newton-Raphson. It must be 0 or a positive integer. A value of 0 means using Newton-Raphson for all iterations and a value greater or equal to M means using Fisher scoring for all iterations.
M	The maximum number of iterations. It must be a non-negative integer. If the value is 0, then initial parameter values become final estimates.
$\epsilon_\ell, \epsilon_p, \epsilon_{\mathbf{H}}$	Tolerance levels for three types of convergence criteria.
Abs	A 0/1 binary variable; *Abs* = 1 if absolute change is used for convergence criteria and *Abs* = 0 if relative change is used.

And the iterative process is outlined as follows:

1. Input values for *I*, *J*, *K*, *M*, $\epsilon_\ell, \epsilon_p, \epsilon_{\mathbf{H}}$ and *Abs* for each type of three convergence criteria.

2. Input initial values $\beta^{(0)}$ or if no initial values are given,

3. Let ξ=1.

4. Compute estimates of *i*th iteration:

   $\beta^{(i)} = \beta^{(i-1)} - \xi\left(\mathbf{H}^{(i-1)}\right)^{-}\mathbf{s}^{(i-1)}$, where $(\mathbf{H})^{-}$ is a generalized inverse of **H**. Then compute the log-likelihood based on $\beta^{(i)}$.

5. Use step-halving method if $\ell^{(i)} < \ell^{(i-1)}$: reduce ξ by half and repeat step (4). The set of values of ξ is $\{0.5^j : j = 0, \ldots, J-1\}$. If *J* is reached but the log-likelihood is not improved, issue a warning message, then stop.

6. Compute gradient vector $\mathbf{s}^{(i)}$ and Hessian matrix $\mathbf{H}^{(i)}$ based on $\beta^{(i)}$. Note that $\mathbf{W}_e$ is used to calculate $\mathbf{H}^{(i)}$ if $i \le K$; $\mathbf{W}_o$ is used to calculate $\mathbf{H}^{(i)}$ if $i>K$.

7. Check if complete or quasi-complete separation of the data is established (see below) if distribution is binomial or ordinal multinomial and the current iteration $i \ge I$. If either complete or quasi-complete separation is detected, issue a warning message, then stop.

8. Check if all three convergence criteria (see below) are met. If they are not but *M* is reached, issue a warning message, then stop.

9. If all three convergence criteria are met, check if complete or quasi-complete separation of the data is established if distribution is binomial or ordinal multinomial and $i<I$ (because checking for complete or quasi-complete separation has not started yet). If complete or quasi-complete separation is detected, issue a warning message, then stop, otherwise, stop (the process converges for binomial or ordinal multinomial successfully). If all three convergence criteria are met for the distributions other than binomial or ordinal multinomial, stop (the process converges for other

distributions successfully). The final vector of estimates is denoted by $\hat{\beta}$ (and $\hat{\tau}$ and $\hat{\Psi}$ for ordinal multinomial). Otherwise, go back to step (3).

Initial Values

If initial values are not specified by the user, they are calculated as follows:

1. Set the initial fitted values $\tilde{\mu}_i = (y_i m_i + 0.5)/(m_i + 1)$ for a binomial distribution (y_i can be a proportion or 0/1 value) and $\tilde{\mu}_i = y_i$ for a non-binomial distribution. From these derive $\tilde{\eta}_i = g(\tilde{\mu}_i)$, $g'(\tilde{\mu}_i)$ and $V(\tilde{\mu}_i)$. If $\tilde{\eta}_i$ becomes undefined, set $\tilde{\eta}_i = 1$.
2. Calculate the weight matrix $\tilde{W}_e$ with the diagonal element $\tilde{w}_{ei} = \frac{f_i \omega_i}{\phi} \frac{1}{V(\tilde{\mu}_i)\left(g'(\tilde{\mu}_i)\right)^2}$, where ϕ is set to 1 or a fixed positive value. If the denominator of $\tilde{w}_{ei}$ becomes 0, set $\tilde{w}_{ei} = 0$.
3. Assign the adjusted dependent variable *z* with the *i*th observation $z_i = (\tilde{\eta}_i - o_i) + (y_i - \tilde{\mu}_i) g'(\tilde{\mu}_i)$ for a binomial distribution and $z_i = (\tilde{\eta}_i - o_i)$ for a non-binomial distribution.
4. Calculate the initial parameter values

$$\beta^{(0)} = \left(\mathrm{X}^{\mathrm{T}} \tilde{W}_e \mathrm{X}\right)^{-1} \mathrm{X}^{\mathrm{T}} \tilde{W}_e \mathrm{z}$$

and

$$\phi^{(0)} = \left(\mathrm{z} - \mathrm{X}\beta^{(0)}\right)^{\mathrm{T}} \tilde{W}_e \left(\mathrm{z} - \mathrm{X}\beta^{(0)}\right).$$

if the scale parameter is estimated by the ML method.

For the ancillary parameter *k* of the negative binomial model, the initial *k* = 1, so the initial τ = 0.

For the ordinal multinomial model, let $N_j = \sum_{i=1}^{n} f_i y_{i,j}$ be the number of responses in category *j*, and $N = \sum_{i=1}^{n} f_i$ be the effective sample size. Initial values for the threshold parameters, with and without the offset variable, are then computed according to the following formulae:

$$\psi_j^{(0)} = g\left(\frac{\sum_{l=1}^{j} N_l}{N}\right)$$

and

$$\psi_j^{(0)} = g\left(\frac{\sum_{l=1}^{j} N_l}{N}\right) - \bar{o}_j$$

for j=1,...,J−1, where $\bar{o}_j = \sum_{l=1}^{j}\sum_{i=1}^{n} f_i y_{i,j} o_i / \sum_{l=1}^{j}\sum_{i=1}^{n} f_i y_{i,j}$.

Initial values for all regression parameters are set to zero.

Scale Parameter Handling

1. For normal, inverse Gaussian, gamma and Tweedie response, if the scale parameter is estimated by the ML method, then it will be estimated jointly with the regression parameters; that is, the last element of the gradient vector **s** is with respect to τ.
2. If the scale parameter is set to be a fixed positive value, then it will be held fixed at that value for in each iteration of the above process.
3. If the scale parameter is specified by the deviance or Pearson chi-square divided by degrees of freedom, then it will be fixed at 1 to obtain the regression estimates through the whole iterative process. Based on the regression estimates, calculate the deviance and Pearson chi-square values and obtain the scale parameter estimate.

Checking for Separation

For each iteration after the user-specified number of iterations; that is, if $i > I$, calculate (note here v refers to cases in the dataset)

$$p_{\min} = \min_v p_v$$

$$p_{\max} = \max_v p_v,$$

$$p^*_{\min} = \min_v \left(\min\left(\mu_v, 1-\mu_v\right)\right),$$

where

$$p_v = \begin{cases} \mu_v & \text{if } y_v = \text{success} (= 1) \\ 1-\mu_v & \text{if } y_v = \text{failure} (= 0) \end{cases}$$

(p_v is the probability of the observed response for case v) and $\mu_v = g^{-1}\left(\mathbf{x}_v^T \beta + o_v\right)$

For the ordinal multinomial model, the definitions are modified as follows:

$$p_{\min} = \min_v \pi_{v,y_v}$$

$$p_{\max} = \max_v \pi_{v,y_v},$$

$$p^*_{\min} = \min_v \left(\min_j \pi_{v,j}\right).$$

The rules for checking complete separation or quasi-complete separation for binomial or multinomial models are otherwise the same.

If $\min(p_{\min}, p_{\max}) = p_{\min} > 0.99$ we consider there to be complete separation. Otherwise, if $p_{\max} > 0.99$ or $p^*_{\min} < 0.001$ and if there are very small diagonal elements (absolute value $< \sqrt{10^{-7}} \approx 3.16 \times 10^{-4}$) in the non-redundant parameter locations in the lower triangular matrix in Cholesky decomposition of –**H**, where **H** is the Hessian matrix, then there is a quasi-complete separation.

Convergence Criteria

The following convergence criteria are considered:

$$\text{Log-likelihood convergence: } \begin{cases} \dfrac{\left|\ell^{(i)} - \ell^{(i-1)}\right|}{\left|\ell^{(i-1)}\right| + 10^{-6}} < \epsilon_\ell \text{ if relative change} \\ \left|\ell^{(i)} \quad \ell^{(i-1)}\right| < \epsilon_\ell \text{ if absolute change} \end{cases}$$

$$\text{Parameter convergence: } \begin{cases} \max_j \left(\dfrac{\left|\beta_j^{(i)} - \beta_j^{(i-1)}\right|}{\left|\beta_j^{(i-1)}\right| + 10^{-6}}\right) < \epsilon\text{p if relative change} \\ \max_j \left(\left|\beta_j^{(i)} - \beta_j^{(i-1)}\right|\right) < \epsilon\text{p if absolute change} \end{cases}$$

$$\text{Hessian convergence: } \begin{cases} \dfrac{\left(\mathbf{s}^{(i)}\right)^{\mathrm{T}} \left(\mathbf{H}^{(i)}\right)^{-} \left(\mathbf{s}^{(i)}\right)}{\left|\ell^{(v)}\right| + 10^{-6}} < \epsilon_{\mathbf{H}} \text{ if relative change} \\ \left(\mathbf{s}^{(i)}\right)^{\mathrm{T}} \left(\mathbf{H}^{(i)}\right)^{-} \left(\mathbf{s}^{(i)}\right) < \epsilon_{\mathbf{H}} \text{ if absolute change} \end{cases}$$

where ϵ_ℓ, ϵp and $\epsilon_{\mathbf{H}}$ are the given tolerance levels for each type.

If the Hessian convergence criterion is not user-specified, it is checked based on absolute change with $_{\mathbf{H}}$ = 1E-4 after the log-likelihood or parameter convergence criterion has been satisfied. If Hessian convergence is not met, a warning is displayed.

Parameter Estimate Covariance Matrix, Correlation Matrix and Standard Errors

The parameter estimate covariance matrix, correlation matrix and standard errors can be obtained easily with parameter estimates. Whether or not the scale parameter is estimated by ML, parameter estimate covariance and correlation matrices are listed for $\hat{\beta}$ only because the covariance between $\hat{\beta}$ and $\hat{\tau}$ should be zero.

If the ancillary parameter *k* (τ) of negative binomial is estimated by ML method, the parameter estimate covariance and correlation matrices are still listed for $\hat{\beta}$ only even though the covariance between $\hat{\beta}$ and $\hat{\tau}$ is generally not zero.

For the ordinal multinomial model, parameter estimate covariance and correlation matrices are listed for $\hat{\beta}$ and $\hat{\Psi}$.

Model-Based Parameter Estimate Covariance

The model-based parameter estimate covariance matrix is given by

$$\Sigma_m = -H^{-} = -(-XWX)^{-}$$

where $\mathbf{H}^{-}$ is the generalized inverse of the Hessian matrix evaluated at the parameter estimates. The corresponding rows and columns for redundant parameter estimates should be set to zero.

Robust Parameter Estimate Covariance

The validity of the parameter estimate covariance matrix based on the Hessian depends on the correct specification of the variance function of the response in addition to the correct specification of the mean regression function of the response. The robust parameter estimate covariance provides a consistent estimate even when the specification of the variance function of the response is incorrect. The robust estimator is also called Huber's estimator because Huber (1967) was the first to describe this variance estimate; White's estimator or HCCM (heteroskedasticity consistent covariance matrix) estimator because White (1980) independently showed that this variance estimate is consistent under a linear regression model including heteroskedasticity; or the sandwich estimator because it includes three terms. The robust (or Huber/White/sandwich) estimator is defined as follows

$$\Sigma_r = \Sigma_m\left(\sum_{i=1}^{n}\left[\frac{\partial \ell_i}{\partial \beta}\right]\left[\frac{\partial \ell_i}{\partial \beta}\right]^{T}\right)\Sigma_m = \Sigma_m\left(\sum_{i=1}^{n} f_i\left(\frac{\omega_i(y_i-\mu_i)}{\phi V(\mu_i)\, g'(\mu_i)}\right)^{2}\cdot x_i\cdot x_i^{T}\right)\Sigma_m$$

For the ordinal multinomial model,

$$\Sigma_r=\Sigma_m\left(\sum_{i=1}^{n} f_i\left[\frac{\partial \ell_i}{\partial B}\right]\left[\frac{\partial \ell_i}{\partial B}\right]^{T}\right)\Sigma_m$$

where

$$\frac{\partial \ell_i}{\partial B}=\left[\frac{\partial \ell_i}{\partial \psi}^{T}, \frac{\partial \ell_i}{\partial \beta}^{T}\right]^{T} = \left[\frac{\partial \ell_i}{\partial \psi_1}, \cdots, \frac{\partial \ell_i}{\partial \psi_{J-1}}, \frac{\partial \ell_i}{\partial \beta_1}, \cdots, \frac{\partial \ell_i}{\partial \beta_p}\right]^{T}$$

$$\frac{\partial \ell_i}{\partial \psi_j} = \frac{\omega_i}{\phi}\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}}\left(\frac{y_{i,j}}{\pi_{i,j}} - \frac{y_{i,j+1}}{\pi_{i,j+1}}\right), j = 1,\ldots,J-1$$

$$\frac{\partial \ell_i}{\partial \beta_t} = -\sum_{j=1}^{J}\frac{\omega_i}{\phi}\left(\frac{\partial \gamma_{i,j}}{\partial \eta_{i,j}} - \frac{\partial \gamma_{i,j-1}}{\partial \eta_{i,j-1}}\right)\frac{y_{i,j}}{\pi_{i,j}} x_{it}, t = 1,\ldots,p.$$

Parameter Estimate Correlation

The correlation matrix is calculated from the covariance matrix as usual. Let σ_{ij} be an element of Σ_m or Σ_r, then the corresponding element of the correlation matrix is $\frac{\sigma_{ij}}{\sqrt{\sigma_{ii}}\sqrt{\sigma_{jj}}}$. The corresponding rows and columns for redundant parameter estimates should be set to system missing values.

Parameter Estimate Standard Error

Let $\hat{\beta}_i$ denote a non-redundant parameter estimate for all distributions except multinomial. Its standard error is the square root of the *i*th diagonal element of Σ_m or Σ_r:

$$\hat{\sigma}_{\beta_i} = \sqrt{\sigma_{ii}}$$

The standard error for redundant parameter estimates is set to a system missing value. If the scale parameter is estimated by the ML method, we obtain $\hat{\tau}$ and its standard error estimate $\hat{\sigma}_\tau = \sqrt{-\frac{1}{\left(\frac{\partial^2 \ell}{\partial \tau^2}\right)}}$, where $\frac{\partial^2 \ell}{\partial \tau^2}$ can be found in Table 46-13 "The second derivative functions w.r.t. the scale parameter for probability distributions". Then the estimate of the scale parameter is exp $(\hat{\tau})$ and the standard error estimate is (exp $(\hat{\tau}) \cdot \hat{\sigma}_\tau$)

For the ordinal multinomial model, let $\hat{\psi}_j, j = 1, \ldots, J-1$, be threshold parameter estimates and $\hat{\beta}_i, i = 1, \ldots, p$, denote non-redundant regression parameter estimates. Their standard errors are the square root of the *i*th diagonal element of Σ_m or Σ_r: $\hat{\sigma}_{\psi_j} = \sqrt{\sigma_{jj}}$ and $\hat{\sigma}_{\beta_i} = \sqrt{\sigma_{(J-1+i),(J-1+i)}}$ respectively.

Confidence Intervals

There are two methods of computing confidence intervals for the non-redundant parameters. One is based on the asymptotic normality of the parameter estimators, and the other is based on the profile likelihood function. The latter is time consuming because it needs to run iterative processes many times.

Wald Confidence Intervals

Wald confidence intervals are based on the asymptotic normal distribution of the parameter estimates. The 100(1 – α)% Wald confidence interval for β_j is given by

$$\left(\hat{\beta}_j - z_{1-\alpha/2}\hat{\sigma}_{\beta_j}, \hat{\beta}_j + z_{1-\alpha/2}\hat{\sigma}_{\beta_j}\right),$$

where z_p is the 100*p*th percentile of the standard normal distribution.

If exponentiated parameter estimates are requested for logistic regression or log-linear models, then using the delta method, the estimate of $\exp(\beta_j)$ is $\exp\left(\hat{\beta}_j\right)$, the standard error estimate of $\exp\left(\hat{\beta}_j\right)$ is $\left(\exp\left(\hat{\beta}_j\right)\cdot\hat{\sigma}_{\beta_j}\right)$ and the corresponding 100(1 – α)% Wald confidence interval $\exp(\beta_j)$ for is

$$\left(\exp\left(\hat{\beta}_j - z_{1-\alpha/2}\hat{\sigma}_{\beta_j}\right), \exp\left(\hat{\beta}_j + z_{1-\alpha/2}\hat{\sigma}_{\beta_j}\right)\right).$$

Wald confidence intervals for redundant parameter estimates are set to system missing values.

Similarly, the 100(1 – α)% Wald confidence interval for ϕ or *k* of the negative binomial model is

$$\left(\exp\left(\hat{\tau} - z_{1-\alpha/2}\hat{\sigma}_\tau\right), \exp\left(\hat{\tau} + z_{1-\alpha/2}\hat{\sigma}_\tau\right)\right)$$

Additionally, for the ordinal multinomial model, the 100(1 – α)% Wald confidence interval for

ψ_j is given by

$$\left(\hat{\psi}_j - z_{1-\alpha/2}\hat{\sigma}_{\psi_j}, \hat{\psi}_j + z_{1-\alpha/2}\hat{\sigma}_{\psi_j}\right)$$

the estimate of $\exp(\psi_j)$ is $\exp\left(\hat{\psi}_j\right)$, the standard error estimate of $\exp\left(\hat{\psi}_j\right)$ is $\left(\exp\left(\hat{\psi}_j\right) \cdot \hat{\sigma}_{\psi_j}\right)$ and the corresponding 100(1 – α)% Wald confidence interval for $\exp(\psi_j)$ is

$$\left(\exp\left(\hat{\psi}_j - z_{1-\alpha/2}\hat{\sigma}_{\psi_j}\right), \exp\left(\hat{\psi}_j + z_{1-\alpha/2}\hat{\sigma}_{\psi_j}\right)\right)$$

Profile Likelihood Confidence Intervals

The construction of profile likelihood confidence interval (PLCI) is first derived from the asymptotic Chi-square distribution of the generalized likelihood ratio test by Venzon and Moolgavkav (1988). We use the modified algorithm, which is equivalent to theirs, by Heinze and Ploner (2002). The computation is iterative and very time consuming, especially if the number of predictors is large because the number of iterative processes needed is $2p_X$; for the ordinal multinomial model, it is $2(J - 1 + p_X)$. PLCIs for redundant parameter estimates are set to system missing values and won't involve iterative processes.

The iterative process is as follows:

1. Let initial values $\beta^{(0)}$ (note it might include τ; Ψ for multinomial) be the maximum likelihood estimates and initial log-likelihood $\ell^{(0)}$, gradient vector $\mathbf{s}^{(0)}$ and Hessian matrix $\mathbf{H}^{(0)}$ are obtained based on $\beta^{(0)}$.

2. Calculate $\ell_0 = \ell^{(0)} - 0.5\chi^2_{1,(1-\alpha)}$, where $\chi^2_{1,(1-\alpha)}$ is the 100(1 – α)% percentile of the Chi-square distribution with one degree of freedom.

3. Set the parameter number $j = 1$.

4. Set the iteration number $i = 1$.

5. Compute the incremental value λ at the $(i - 1)$th iteration:

$$\lambda^{(i-1)} = \pm\left\{\frac{2\left[\ell_0 - \ell^{(i-1)}\right] + \left(\mathrm{s}^{(i-1)}\right)\left(\mathrm{H}^{(i-1)}\right)^{-}\left(\mathrm{s}^{(i-1)}\right)}{\mathrm{e}_j\left(\mathrm{H}^{(i-1)}\right)^{-}\mathrm{e}_j}\right\}^{1/2},$$

   where e_j is the jth unit vector. Take the positive values of λ first.

   In rare cases, the value in the above braces is negative or $\ell^{(i-1)}$ is missing or undefined. In that case, $\lambda^{(i-1)}$ is undefined (note that $\lambda^{(0)}$ is highly unlikely to be undefined) and the parameters can't be updated. To solve this problem, in general, we just take a simple average of parameters from the two previous iterations $\beta^{(i)} = \frac{1}{2}\left(\beta^{(i-1)} + \beta^{(i-2)}\right)$. If $\lambda^{(i)}$ based on $\beta^{(i)}$ is still undefined, we continue the process up to 5 times by taking the average of the current $\beta^{(i)}$ value and $\beta^{(i-2)}$ till $\lambda^{(i)}$ becomes defined, otherwise, we issue a warning and stop.

6. Compute the step size $\mathrm{d}^{(i-1)} = -\left(\mathbf{H}^{(i-1)}\right)^{-}\left(\mathbf{s}^{(i-1)} + \lambda^{(i-1)}\mathrm{e}_j\right)$.

7. Update parameter estimates $\beta^{(i)} = \beta^{(i-1)} + \mathrm{d}^{(i-1)}$

8. Compute log-likelihood $\ell^{(i)}$, gradient vector $\boldsymbol{s}^{(i)}$ and Hessian matrix $\mathbf{H}^{(i)}$ based on $\beta^{(i)}$. Note that whether $\mathbf{W}_e$ or $\mathbf{W}_o$ is used to calculate $\mathbf{H}^{(i)}$ should be based on what had been used in the maximum likelihood estimates of β.

9. Check if the following two criteria with tolerance levels ϵ_ℓ and ϵ_H are satisfied:

   (a) $\left|\ell^{(i)} - \ell_0\right| < \epsilon_\ell$

   (b) $\left(\mathbf{s}^{(i)} + \lambda^{(i}j\right)\left(\mathbf{H}^{\ (i)}\right)^{-}\left(\mathbf{s}^{(i)} + \lambda^{(i}j\right) < \epsilon_{\mathbf{H}}$.

   If both criteria are met or the maximum number of iterations is reached, stop. Otherwise, set *i* = *i* + 1 and go back to step (5).

10. The final vector of estimates is denoted by $\hat{\beta}^u$, then $\hat{\beta}^u_j$ is the upper confidence limit for β_j.

11. Repeat steps (4) – (9) with negative values of λ in step (5) to find the lower confidence limit $\hat{\beta}^l_j$.

12. Repeat steps (4) – (11) by setting the parameter number *j* = 2, …, p_X.

    Note

    - If the scale parameter or ancillary parameter *k* of the negative binomial model is estimated by ML method, then it will be estimated jointly with regression parameters for the iterative processes of each regression parameter β_j, *j* = 1, …, p_X. Then the PLCI for ϕ will be obtained by the iterative processes as well, and is equal to $\left(\exp\left(\hat{\tau}^l\right), \exp\left(\hat{\tau}^u\right)\right)$ Similarly, the profile likelihood confidence interval for $\exp\left(\beta_j\right)$ is calculated as $\left(\exp\left(\hat{\beta}^l_j\right), \exp\left(\hat{\beta}^u_j\right)\right)$.
    - If the scale parameter or ancillary parameter *k* of the negative binomial model is set to be a fixed positive value, then it will be held fixed at that value for each iterative process.
    - If the scale parameter is specified for all distributions by the deviance or Pearson chi-square divided by degrees of freedom, then ϕ will be held fixed at the value estimated from the deviance or Pearson statistic during the full model fit for each iterative process. For more information, see the topic "Goodness-of-Fit Statistics".

Chi-Square Statistics

The hypothesis $H_{0i} : \beta_i = 0$ is tested for each non-redundant parameter using the chi-square statistic:

$$c_i = \left(\frac{\hat{\beta}_i}{\hat{\sigma}_{\beta_J}}\right)^2$$

which has an asymptotic chi-square distribution with 1 degree of freedom.

Chi-square statistics and their corresponding *p*-values are set to system missing values for redundant parameter estimates.

The chi-square statistic is not calculated for the scale parameter, even if it is estimated by ML method.

For the ordinal multinomial model, the hypotheses $H_{0j} : \psi_j = 0, j = 1, \ldots, J-1$, and $H_{0i} : \beta_i = 0, i = 1, \ldots, p_x$, are tested for threshold parameters and non-redundant regression parameters using the chi-square statistics

$$c_{\psi_j} = \left(\frac{\hat{\psi}_j}{\hat{\sigma}_{\psi_j}} \right)^2$$

and

$$c_{\beta_i} = \left(\frac{\hat{\beta}_i}{\hat{\sigma}_{\beta_i}} \right)^2$$

P Values

Given a test statistic *T* and a corresponding cumulative distribution function *G* as specified above, the *p*-value is defined as $p = 1 - G(T)$. For example, the *p*-value for the chi-square test of $H_{0i} : \beta_i = 0$ is $p_i = 1 - prob\left(\chi_1^2 \le c_i\right)$.

Model Testing

After estimating parameters and calculating relevant statistics, several tests for the given model are performed.

Lagrange Multiplier Test

If the scale parameter for normal, inverse Gaussian, gamma, and Tweedie distributions is set to a fixed value or specified by the deviance or Pearson chi-square divided by the degrees of freedom (when the scale parameter is specified by the deviance or Pearson chi-square divided by the degrees of freedom, it can be considered as a fixed value), or an ancillary parameter *k* for the negative binomial is set to a fixed value other than 0, the Lagrange Multiplier (LM) test assesses the validity of the value. For a fixed ϕ or *k*, the test statistic is defined as

$$T_{LM} = \frac{s^2}{A}$$

where $s = \partial\ell/\partial\tau$ and $A = -\left(\frac{\partial^2\ell}{\partial\tau^2}\right) - \left(-\frac{\partial^2\ell}{\partial\tau\partial\beta}^{\mathrm{T}}\right)\left(-\frac{\partial^2\ell}{\partial\beta\partial\beta}^{\mathrm{T}}\right)^{-}\left(-\frac{\partial^2\ell}{\partial\beta\partial\tau}\right)$ evaluated at the parameter estimates and fixed ϕ or *k* value. T_{LM} has an asymptotic chi-square distribution with 1 degree of freedom, and the *p*-values are calculated accordingly.

For testing ϕ, see Table 46-12 "The 1st derivative functions w.r.t. the scale parameter for probability distributions" and see Table 46-13 "The second derivative functions w.r.t. the scale parameter for probability distributions" for the elements of *s* and *A*, respectively.

If *k* is set to 0, then the above statistic can't be applied. According to Cameron and Trivedi (1998), the LM test statistic should now be based on the following auxiliary OLS regression (without constant)

$$\frac{(y_i-\hat{\mu}_i)^2-y_i}{\hat{\mu}_i}=\alpha\hat{\mu}_i+\epsilon_i$$

where $\hat{\mu}_i = g^{-1}\left(x_i^T\hat{\beta}\right)$ and ϵ_i is an error term. Let the response of the above OLS regression $\left[(y_i-\hat{\mu}_i)^2-y_i\right]/\hat{\mu}_i$ be z_i and the explanatory variable $\hat{\mu}_i$ be w_i. The estimate of the above regression parameter α and the standard error of the estimate of α are

$$\hat{\alpha}=\frac{\sum_{i=1}^{n} f_i w_i z_i}{\sum_{i=1}^{n} f_i w_i^2} \text{ and } \hat{\sigma}_\alpha = \sqrt{\frac{s_e^2}{\sum_{i=1}^{n} f_i w_i^2}},$$

where $s_e^2 = \frac{1}{N-1}\sum_{i=1}^{n} f_i e_i^2$ and $e_i = z_i - \hat{\alpha} w_i$. Then the LM test statistic is a z statistic

$$z=\frac{\hat{\alpha}}{\hat{\sigma}_\alpha},$$

and it has an asymptotic standard normal distribution under the null hypothesis of equidispersion in a Poisson model ($H_0 : k=0$). Three *p*-values are provided. The alternative hypothesis can be one-sided overdispersion ($H_a : k>0$), underdispersion ($H_a : k<0$) or two-sided non-directional ($H_a : k\neq 0$) with the variance function of $V(\mu) = \mu + k\mu^2$. The calculation of *p*-values depends on the alternative. For $H_a : k > 0$, *p*-value $= 1-\Phi(z)$, where $\Phi(\cdot)$ is the cumulative probability of a standard normal distribution; for $H_a : k < 0$, *p*-value $= \Phi(z)$; and for $H_a : k \neq 0$, *p*-value $= 2(1-\Phi(|z|))$.

Goodness-of-Fit Statistics

Several statistics are calculated to assess goodness of fit of a given generalized linear model.

Deviance

The theoretical definition of deviance is:

$$D = 2\phi(\ell(\mathrm{y};\mathrm{y}) - \ell(\hat{\mu};\mathrm{y})),$$

where $\ell(\hat{\mu}; y)$ is the log-likelihood function expressed as the function of the predicted mean values $\hat{\mu}$ (calculated based on the parameter estimates) given the response variable, and $\ell(y; y)$ is the log-likelihood function computed by replacing $\hat{\mu}$ with **y**. The formula used for the deviance is $\sum_{i=1}^{n} f_i d_i$, where the form of di for the distributions are given in the following table:

Table 46-15
Deviance for individual case

Distribution	***di***
Normal	$\omega_i (y_i - \mu_i)^2$
Inverse Gaussian	$\frac{\omega_i}{y_i \mu_i^2}(y_i - \mu_i)^2$
Gamma	$2\omega_i \left\{ -\ln\left(\frac{y_i}{\mu_i}\right) + \frac{y_i - \mu_i}{\mu_i} \right\}$
Negative Binomial	$2\omega_i \left\{ y_i \ln\left(\frac{y_i}{\mu_i}\right) - (y_i + 1/k)\ln\left(\frac{y_i + 1/k}{\mu_i + 1/k}\right) \right\}$
Poisson	$2\omega_i \left\{ y_i \ln\left(\frac{y_i}{\mu_i}\right) - (y_i - \mu_i) \right\}$
Binomial(*m*)	$2\omega_i^* \left\{ y_i \ln\left(\frac{y_i}{\mu_i}\right) + (1 - y_i)\ln\left(\frac{1 - y_i}{1 - \mu_i}\right) \right\}$
Tweedie	$2\omega_i \left\{ \frac{y_i^{2-q} - (2-q) y_i \mu_i^{1-q} + (1-q)\mu_i^{2-q}}{(1-q)(2-q)} \right\}$

Note

- When **y** is a binary dependent variable with 0/1 values (binomial distribution), the deviance and Pearson chi-square are calculated based on the subpopulations; see below.
- When *y* = 0 for negative binomial and Poisson distributions and *y* = 0 (for *r* = 0) or 1 (for *r* = *m*) for binomial distribution with **r**/**m** format, separate values are given for the deviance. Let d_i be the deviance value for individual case *i* when $y_i = 0$ for negative binomial and Poisson and 0/1 for binomial.

Table 46-16
Deviance for individual case

Distribution	d_i
Negative Binomial	$2\omega_i \frac{\ln(1 + k\mu_i)}{k}$ if $y_i = 0$
Poisson	$2\omega_i \mu_i$ if $y_i = 0$
Binomial(*m*)	$\begin{cases} -2\omega_i^* \ln(1 - \mu_i) \text{ if } y_i = 0 \text{ or } r_i = 0 \\ -2\omega_i^* \ln(\mu_i) \text{ if } y_i = 1 \text{ or } r_i = m_i \end{cases}$

Pearson Chi-Square

$$\chi^2 = \sum_{i=1}^{n} f_i \gamma_i$$

where $\gamma_i = \frac{\omega_i^* (y_i - \mu_i)^2}{V(\mu_i)}$ for the binomial distribution and $\gamma_i = \frac{\omega_i (y_i - \mu_i)^2}{V(\mu_i)}$ for other distributions.

Scaled Deviance and Scaled Pearson Chi-Square

The scaled deviance is $D^* = D/\phi$ and the scaled Pearson chi-square is $\chi^{2*} = \chi^2/\phi$.

Since the scaled deviance and Pearson chi-square statistics have a limiting chi-square distribution with $N - p_X$ degrees of freedom, the deviance or Pearson chi-square divided by its degrees of freedom can be used as an estimate of the scale parameter for both continuous and discrete distributions.

$\hat{\phi} = \frac{D}{N-p_x}$ or $\hat{\phi} = \frac{\chi^2}{N-p_x}$.

If the ancillary parameter *k* of the negative binomial model is estimated by the ML method, the scale parameter is measured by the deviance or Pearson chi-square divided by its degrees of freedom, then the degrees of freedom is $N - p_x - 1$ because *k* is the extra parameter estimated by ML method.

If the scale parameter is measured by the deviance or Pearson chi-square, first we assume $\phi = 1$, then estimate the regression parameters, calculate the deviance and Pearson chi-square values and obtain the scale parameter estimate from the above formula. Then the scaled version of both statistics is obtained by dividing the deviance and Pearson chi-square by $\hat{\phi}$. In the meantime, some statistics need to be revised. The gradient vector and the Hessian matrix are divided by $\hat{\phi}$ and the covariance matrix is multiplied by $\hat{\phi}$. Accordingly the estimated standard errors are also adjusted, the Wald confidence intervals and significance tests will be affected even the parameter estimates are not affected by $\hat{\phi}$.

Note that two log likelihood values are displayed: the original one (based on $\phi = 1$) and the revised one (based on $\phi = \hat{\phi}$ which is plugged into the log likelihood function of the corresponding distribution). Prior to version 16, only the original one is displayed. The original log likelihood is used in computing the information criteria but the revised log likelihood is used in the model fitting omnibus test.

Overdispersion

For the Poisson and binomial distributions, if the estimated scale parameter is not near the assumed value of one, then the data may be overdispersed if the value is greater than one or underdispersed if the value is less than one. Overdispersion is more common in practice. The problem with overdispersion is that it may cause standard errors of the estimated parameters to be underestimated. A variable may appear to be a significant predictor, when in fact it is not.

Deviance and Pearson Chi-Square for Binomial with 0/1 Binary Response and Ordinal Multinomial

When **r** and **m** (event/trial) variables are used for the binomial distribution, each case represents m Bernoulli trials. When **y** is a binary dependent variable with 0/1 values, each case represents a single trial. The trial can be repeated for several times with the same setting (i.e. the same values for all predictor variables). For example, suppose the first 10 *y* values are 2 1s and 8 0s and **x** values are the same (if recorded in events/trials format, these 10 cases is recorded as 1 case with r = 2 and m = 10), then these 10 cases should be considered from the same subpopulation. Cases with common values in the variable list that includes all predictor variables are regarded as coming from the same subpopulation. When the binomial distribution with binary response is used, we should calculate the deviance and Pearson chi-square based on the subpopulations. If we calculate them based on the cases, the results might not be useful.

If subpopulations are specified for the binomial distribution with 0/1 binary response variable, the data should be reconstructed from the single trial format to the events/trials format. Assume the following notation for formatted data:

Table 46-17
Notation

Notation	Description
n_s	Number of subpopulations.
r_{j1}	Sum of the product of the frequencies and the scale weights associated with $y = 1$ in the *j*th subpopulation. So r_{j0} is that with $y = 0$ in the *j*th subpopulation.
m_j	Total weighted observations; $m_j = r_{j1} + r_{j0}$.
y_{j1}	The proportion of 1s in the *j*th subpopulation; $y_{j1} = r_{j1}/ m_j$.
μ_j	The fitted probability in the *j*th subpopulation $\hat{\mu}_j$ would be the same for each case in the *j*th subpopulation because values for all predictor variables are the same for each case.)

The deviance and Pearson chi-square are defined as follows:

$$D = 2\sum_{j=1}^{n_s} m_j \left\{ y_{j1} \ln\left(\frac{y_{j1}}{\mu_j}\right) + (1 - y_{j1}) \ln\left(\frac{1 - y_{j1}}{1 - \mu_j}\right) \right\} \text{ and } \chi^2 = \sum_{j=1}^{n_s} \frac{m_j (y_{j1} - \mu_j)^2}{\mu_j (1 - \mu_j)},$$

then the corresponding estimate of the scale parameter will be

$\hat{\phi} = \frac{D}{n_s - p_x}$ and $\hat{\phi} = \frac{\chi^2}{n_s - p_x}$.

The full log likelihood, based on subpopulations, is defined as follows:

$$\ell = \ell_k + \sum_{j=1}^{n_s} \frac{1}{\phi} \left\{ \ln \binom{m_j}{r_{j1}} \right\} = \ell_k + \sum_{j=1}^{n_s} \frac{1}{\phi} \left\{ \ln \frac{m_j!}{r_{j1}! r_{j0}!} \right\},$$

where ℓ_k is the kernel log likelihood; it should be the same as the kernel log-likelihood computed based on cases before, there is no need to compute again.

For the ordinal multinomial model, similarly, the data will be reconstructed based on subpopulations. Assume the following notation for reconstructed ordinal multinomial data:

Table 46-18
Notation

Notation	Description
n_s	Number of subpopulations.
r_{ij}	Sum of the product of the frequencies and the scale weights associated with the *j*th category in the *i*th subpopulation.
m_i	Total weighted observations for the *i*th subpopulation; $m_i = \sum_{j=1}^{J} r_{i,j}$
$\hat{\pi}_{i,j}$	The fitted probability for the *j*th category in the *i*th subpopulation.

The deviance and Pearson chi-square are defined as follows.

$$D = 2\sum_{i=1}^{n_s}\sum_{j=1}^{J} r_{i,j} \ln\left(\frac{r_{i,j}}{m_i \hat{\pi}_{i,j}}\right)$$

and

$$\chi^2 = \sum_{i=1}^{n_s}\sum_{j=1}^{J} \frac{(r_{i,j} - m_i\hat{\pi}_{i,j})^2}{m_i\hat{\pi}_{i,j}}.$$

with $n_s(J-1)-d$ degrees of freedom, where $d = J-1+p_x$. The corresponding estimates of the scale parameter will be

$$\hat{\phi} = \frac{D}{n_s(J-1)-d}$$

and

$$\hat{\phi} = \frac{\chi^2}{n_s(J-1)-d}$$

The full log likelihood, based on subpopulations, is defined as follows:

$$\ell = \ell_k + c = \ell_k + \sum_{i=1}^{n_s}\frac{1}{\phi}\left\{\ln\frac{m_j!}{r_{j1}!\cdots r_{jJ}!}\right\},$$

where again ℓ_k is the same as before.

Information Criteria

Information criteria are used when comparing different models for the same data. The formulas for various criteria are as follows.

Akaike information criteria (AIC). $-2\ell + 2d$

Finite sample corrected (AICC). $-2\ell + \frac{2d \cdot N}{(N-d-1)}$

Bayesian information criteria (BIC). $-2\ell + d\ln(N)$

Consistent AIC (CAIC). $-2\ell + d(\ln(N)+1)$.

where ℓ is the log-likelihood evaluated at the parameter estimates. Notice that $d = p_X$ if only β is included; $d = p_X + 1$ if the scale parameter is included for normal, inverse Gaussian, gamma, and Tweedie, or k for the negative binomial distribution; for multinomial, $d = J - 1 + p_X$.

Notes

- ℓ (the full log-likelihood) can be replaced with ℓ_k (the kernel of the log-likelihood) depending on the user’s choice.

- If the scale parameter is specified by the deviance or Pearson chi-square, then the log likelihood is based on $\phi = 1$, for fair comparison among different models.
- When **r** and **m** (event/trial) variables are used for the binomial distribution, then the *N* used here would be the sum of the trials frequencies; $N = \sum_{i=1}^{n} f_i m_i$. In this way, the same value results whether the data are in raw, binary form (using single-trial syntax) or in summarized, binomial form (events/trials syntax).

Test of Model Fit

The model fitting omnibus test is based on –2 log-likelihood values for the model under consideration and the initial model. For the model under consideration, the value of the –2 log-likelihood is

$$-2\ell\left(\hat{\beta}\right)$$

Let the initial model be the intercept-only model if intercept is in the considered model or the empty model otherwise. For the intercept-only model, the value of the –2 log-likelihood is

$$-2\ell\left(\hat{\beta}_0\right)$$

For the empty model, the value of the –2 log-likelihood is

$$-2\ell(0)$$

Then the omnibus (or global) test statistic is

$S = 2\left(\ell\left(\hat{\beta}\right) - \ell(\beta_0)\right)$ for the intercept-only model or

$S = 2\left(\ell\left(\hat{\beta}\right) - \ell(0)\right)$ for the empty model.

S has an asymptotic chi-square distribution with *r* degrees of freedom, equal to the difference in the number of valid parameters between the model under consideration and the initial model. *r* = $p_x - 1$ for the intercept-only model,; *r* = p_x for the empty model. The *p*-values then can be calculated accordingly.

Note if the scale parameter or the ancillary parameter is estimated by the ML method in the model under consideration, then it will also be estimated by the ML method in the initial model.

For the ordinal multinomial model, the value of –2 log-likelihood for the model under consideration is

$$-2\ell\left(\hat{\mathrm{B}}\right)$$

the value of –2 log-likelihood for the thresholds-only model is

$$-2\ell\left(\mathrm{B}^{(0)}\right),$$

where $\mathrm{B}^{(0)} = \left(\psi^{(0)}, \beta^{(0)}\right)$ is the initial parameter values used in the iterative process. Then the omnibus test statistic is

$$S = 2\left(\ell\left(\hat{\mathrm{B}}\right) - \ell\left(\mathrm{B}^{(0)}\right)\right)$$

and it has an asymptotic chi-square distribution with p_x degrees of freedom.

When calculating the value of –2 log-likelihood of initial model, the following rules are used to handle the scale parameter or the ancillary parameter *k* in the initial model.

If the scale parameter or the ancillary parameter is estimated by the ML method in the model under consideration, then it will also be estimated by the ML method in the initial model.

If the scale parameter or the ancillary parameter is held fixed in the model under consideration, then the same value is fixed in the initial model.

If the scale parameter is specified by the deviance or Pearson chi-square divided by degrees of freedom in the model under consideration, then that value will be held fixed in the initial model. Note that the log likelihood for the model under consideration would be revised; that is, based on $\phi = \hat{\phi}$, so the log likelihoods for both models (the model under consideration and initial model) are calculated based on the same scale parameter value. This is to be consistent with the way chi-squares statistics in type I and III analyses are computed. Prior to version 16, the log likelihoods for both models are calculated based on $\phi = 1$; thus the omnibus test statistic will be different between 15 and later versions.

Default Tests of Model Effects

For each regression effect specified in the model, type I and III analyses can be conducted.

Type I Analysis

Type I analysis consists of fitting a sequence of models, starting with the null model as the baseline model (for all distributions except ordinal multinomial), adding one additional effect, which can be an intercept term (if there is one), covariates, factors and interactions, of the model on each step. For the ordinal multinomial model, the baseline model is a thresholds-only model. Thus, the test depends on the order of effects specified in the model. On the other hand, type III analysis won't depend on the order of effects. The reason for using the null model as the baseline model is to obtain the chi-square statistic for the first parameter which might be for an intercept or the first predictor variable.

There are two kinds of test statistics for type I analysis: likelihood ratio statistics and Wald statistics.

Likelihood ratio statistics. Different formulae are used to calculate likelihood ratio statistics depending on how the scale parameter or ancillary parameter is handled.

- **Estimated by ML method.** The likelihood ratio statistics are twice the difference of the log likelihoods between two successive models. Unlike type III analysis, we don't obtain the log likelihood of the constrained model based on the type I test matrix.

  Start by considering the first pair of models $\eta=o$ (the null model with the log likelihood $\ell(0)$ and φ or *k* might be estimated) and $\eta=x_1^T\beta_1+o$ (with the log likelihood $\ell\left(\hat{\beta}_1\right)$ and β_1 and φ or *k* are estimated jointly) and the test statistic for the null hypothesis $H_0:\beta_1=0$ is

  $$S=2\left(\ell\left(\hat{\beta}_1\right)-\ell(0)\right)$$

  The log-likelihood convergence criterion is used estimating the above two models. The tolerance level is the same as that used for the parameter estimation iterative process. A similar rule applies to usage of relative or absolute change.

  Note the optimal estimated scale parameter would be different for the above two models. If either log-likelihood is not available due to numerical problems in parameter estimation, then the test statistic, degrees of freedom and *p*-value are all set to system missing values. Similar rules will apply to other pairs of models below.

  Then consider the second pair of two models $\eta=x_1^T\beta_1+o$ and $\eta=x_1^T\beta_1+x_2^T\beta_2+o$, the test statistic for the null hypothesis $H_0:\beta_2=0$ based on β_1 is

  $$S=2\left(\ell\left(\hat{\beta}_1,\hat{\beta}_2\right)-\ell\left(\hat{\beta}_1\right)\right).$$

  Then consider the third pair of models $\eta=x_1^T\beta_1+x_2^T\beta_2+$ offset, and $\eta=x_1^T\beta_1+x_2^T\beta_2+x_3^T\beta_3+$ offset. The likelihood ratio statistic for the null hypothesis $H_0:\beta_3=0$

  $$S=2\left(\ell\left(\hat{\beta}_1,\hat{\beta}_2,\hat{\beta}_3\right)-\ell\left(\hat{\beta}_1,\hat{\beta}_2\right)\right)$$

  Continue this way until all effects in the model are included. Similar convergence criterion applies to all reduced models except the full model. Each likelihood ratio statistic *S* has an asymptotic chi-square distribution with degrees of freedom equal to the difference in the number of parameters estimated in the successive models. The *p*-values can be calculated accordingly.

- **Set to a fixed positive value.** The likelihood ratio statistics are calculated as above except ϕ or *k* is held fixed at that value.

  For the ordinal multinomial model, the scale parameter can be set to a fixed value or be specified by the deviance or Pearson chi-square divided by degrees of freedom. We briefly describe how the statistics can be constructed when it is a fixed value here.

  First, consider the first pair of two models $\eta=\psi+o$ and $\eta=\psi-x_1^T\beta_1+o$, the likelihood ratio statistic for the null hypothesis $H_0:\beta_1=0$ based on Ψ is

  $$S=2\left(\ell\left(\hat{\psi},\hat{\beta}_1\right)-\ell(\hat{\psi})\right)$$

  Then consider the second pair of two models $\eta=\psi+o$ and $\eta=\psi-X_1^T\beta_1+o$, the likelihood ratio statistic for the null hypothesis $H_0:\beta_2=0$ based on ψ and β_1 is

  $$S=2\left(\ell\left(\hat{\psi},\hat{\beta}_1,\hat{\beta}_2\right)-\ell\left(\hat{\psi},\hat{\beta}_1\right)\right)$$

Again, *S* has an asymptotic chi-square distribution with degrees of freedom equal to the difference in the number of parameters in the successive models.

- **Specified from the full model by the deviance or Pearson chi-square.** In this case, the likelihood ratio chi-square and *F* statistics can be computed to assess the significance of each additional effect.

  Suppose that $\ell f, 1$ is the log-likelihood from fitting a generalized model (model *f*) and that $\ell s, 1$ is the log-likelihood from fitting a sub-model (model *s*). Both models are fit assuming the scale parameter equals 1. Then the test statistic is defined by

  $S = \frac{2(\ell_{f,1} - \ell_{s,1})}{\hat{\phi}}$.

  It has an asymptotic chi-square distribution with *r* degrees of freedom, where *r* is the difference in the number of parameters between the two models and $\hat{\phi}$ is estimated from the full model.

  In some references the test statistic is defined by

  $S = \frac{D_s - D_f}{\hat{\phi}}$

  where D_f is the deviance from fitting model *f* and D_s is the deviance from fitting a sub-model *s*. However, this formulation can result in negative chi-square statistics for negative binomial responses where the ancillary parameter is estimated by maximum likelihood.

  Since ϕ is unknown and the estimator $\hat{\phi}$ is the deviance or Pearson chi-square statistic divided by its degrees of freedom, then $(N - p_x)\hat{\phi}/\phi$ has an asymptotic chi-square distribution with $N-p_X$ degrees of freedom. Thus, the *F* statistic can be defined as

  $\mathrm{F} = \frac{S}{r}$

  Under the assumption that $2(\ell_{f,1} - \ell_{s,1})/\phi$ and $(N - p_x)\hat{\phi}/\phi$ are approximately independent, the *F* statistic has an asymptotic F distribution with *r* and $N-p_X$ degrees of freedom, and the *p*-values can be calculated accordingly. Note for the negative binomial with the ancillary parameter *k* estimated by the ML method and with the scale parameter measured by the deviance or Pearson chi-square divided by its degrees of freedom, the degrees of freedom in the denominator for the *F* statistic are $N - p_X - 1$; for the binomial distribution with 0/1 binary response, the degrees of freedom for the denominator should be $n_S - p_X$; for the ordinal multinomial model, the degrees of freedom for the denominator should be $n_s(J-1) - (J - 1 + p_x)$.

  For type I analysis, model *f* is the higher order model obtained by including one additional effect in model *s*. For example, for the second pair of two models, model *f* is $\eta = x_1^T\beta_1 + x_2^T\beta_2 + o$ and model *s* is $\eta = x_1^T\beta_1 + o$.

Wald Statistics. For each effect specified in the model, type I test matrix $\mathbf{L}_i$ is constructed and H_0: $\mathbf{L}_i\beta = \mathbf{0}$ is tested. Construction of matrix $\mathbf{L}_i$ is based on the generating matrix $\mathbf{H}_\omega = \left(\mathbf{X}^T\Omega\mathbf{X}\right)^- \mathbf{X}^T\Omega\mathbf{X}$, where $\mathbf{\Omega}$ is the scale weight matrix with *i*th diagonal element ω_i and such that $\mathbf{L}_i\beta$ is estimable. It involves parameters only for the given effect and the effects containing the given effect. If such a matrix cannot be constructed, the effect is not testable.

Since Wald statistics can be applied to type I and III analysis and custom tests, we express Wald statistics in a more general form. The Wald statistic for testing $\mathbf{L}_i\beta = \mathbf{K}$, where $\mathbf{L}_i$ is a $r \times p$ full row rank hypothesis matrix and $\mathbf{K}$ is a $r \times 1$ resulting vector, is defined by

$$S=\left(\mathbf{L}_i\hat{\beta}-\mathbf{K}\right)^{\mathrm{T}}\left(\mathbf{L}_i\boldsymbol{\Sigma}\mathbf{L}_i^{\mathrm{T}}\right)^{-}\left(\mathbf{L}_i\hat{\beta}-\mathbf{K}\right)$$

where β is the maximum likelihood estimate and **Σ** is the parameter estimates covariance matrix. *S* has an asymptotic chi-square distribution with r_C degrees of freedom, where $r_C=rank\left(\mathrm{L}\Sigma\mathrm{L}^{\mathrm{T}}\right)$. If $r_C<r$, then $\left(\mathrm{L}\Sigma\mathrm{L}^{\mathrm{T}}\right)^{-}$ is a generalized inverse such that Wald tests are effective for a restricted set of hypotheses $\mathbf{L}_{iC}\beta-\mathbf{K}_C$ containing a particular subset *C* of independent rows from H_0.

For type I and III analysis, calculate the Wald statistic for each effect *i* according to the corresponding hypothesis matrix $\mathbf{L}_i$ and **K**=**0**.

For the ordinal multinomial model, first consider partitions of the more general test matrix $\mathrm{L}=(\mathrm{L}(\psi),\mathrm{L}(\beta))$, where $\mathrm{L}(\psi)=(\mathrm{l}_1,\ldots,\mathrm{l}_{J-1})$ consists of columns corresponding to threshold parameters and $\mathrm{L}(\beta)$ be the part of **L** corresponding to regression parameters. Consider matrix $\mathrm{L}_0=(\mathrm{l}_0,\mathrm{L}(\beta))$ where the column vectors corresponding to threshold parameters are replaced by their sum $\mathrm{l}_0=\sum_{j=1}^{J-1}\mathrm{l}_j$. Then **LB** is estimable if and only if $\mathrm{L}_0=\mathrm{L}_0\mathrm{H}_\omega$, where $\mathrm{H}_\omega=\left(\mathrm{X}_1^{\mathrm{T}}\Omega\mathrm{X}_1\right)^{-}\mathrm{X}_1^{\mathrm{T}}\Omega\mathrm{X}_1$, is a $(1+p)\times(1+p)$ matrix constructed using $\mathrm{X}_1=(1,-\mathrm{X})$. The Wald statistic for testing $\mathrm{LB}=\mathrm{K}$, where **L** is a $r\times(J-1+p)$ full row rank hypothesis matrix and **K** is a $r\times 1$ resulting vector, is defined by

$$S=\left(\mathbf{L}\hat{\mathbf{B}}-\mathbf{K}\right)^{\mathrm{T}}\left(\mathbf{L}\boldsymbol{\Sigma}\mathbf{L}^{\mathrm{T}}\right)^{-}\left(\mathbf{L}\hat{\mathbf{B}}-\mathbf{K}\right)$$

where $\widehat{\mathrm{B}}=(\widehat{\psi},\widehat{\beta})$ is the maximum likelihood estimate and $\sum$ is the estimated covariance matrix (Σ could be the model based or robust estimator). The asymptotic distribution of *S* is $\chi^2_{r_C}$, where $r_C=rank\left(\mathrm{L}\Sigma\mathrm{L}^{\mathrm{T}}\right)$.

For each effect specified in the model excluding intercept, a type I test matrix $\mathbf{L}_i$ is constructed and H_0: $\mathbf{L}_i\mathbf{B}=0$ is tested. Construction of matrix $\mathbf{L}_i$ is based on matrix $\mathrm{H}_\omega=\left(\mathrm{X}_1^{\mathrm{T}}\Omega\mathrm{X}_1\right)^{-}\mathrm{X}_1^{\mathrm{T}}\Omega\mathrm{X}_1$ and such that $\mathbf{L}_i\boldsymbol{\beta}$ is estimable. Thus the way to construct $\mathbf{L}_i$ (type I and III) for ordinal multinomial is the same as that for other distributions.

Type III Analysis

Similar to type I analysis, two kinds of test statistics are available for type I analysis: chi-square statistics and Wald statistics.

Likelihood ratio statistics. The likelihood ratio statistics can be obtained as follows:

Calculate the log-likelihood evaluated at the constrained maximum likelihood estimate under the constraint $\mathrm{L}_i\beta=0$ for each effect:

$$\ell\left(\tilde{\beta}_{-i}\right)=\max_{\beta}\ell\left(\mu(\beta);\mathrm{y}\right)\ \mathrm{s.t.}\mathrm{L}_i\beta=0,$$

where L_i is the type III test matrix for the *i*th effect. $\ell\left(\tilde{\beta}_{-i}\right)$ will be obtained by sequential quadratic programming. For more information, see the topic "Sequential Quadratic Programming".

The calculation of $\ell\left(\hat{\beta}\right)$ and $\ell\left(\tilde{\beta}_{-i}\right)$ are based on how the scale parameter φ or the ancillary parameter *k* is handled:

1. If φ or *k* is estimated jointly with **β** by ML method, then $\ell\left(\hat{\beta}\right)$ is the log likelihood evaluated at $\hat{\beta}$ and $\hat{\phi}$ or $\hat{k}$ and $\ell\left(\tilde{\beta}_{-i}\right)$ is the log likelihood evaluated at $\tilde{\beta}_{-i}$ and $\tilde{\phi}_{-i}$ or $\tilde{k}_{-i}$ under the constraint $L_i\beta = 0$ for each effect *i*. Note that the constraint should be expanded by including φ or *k* so that the last element in expanded **β** is φ or k and the last element in expanded L_i is 0.

2. If φ or *k* is set to a fixed value, then $\ell\left(\hat{\beta}\right)$ and $\ell\left(\tilde{\beta}_{-i}\right)$ are calculated with φ or *k* held fixed at that value for both unconstrained and constrained estimation processes.

3. If φ is specified from the full model by the deviance or Pearson chi-square divided by degrees of freedom, then $\ell\left(\hat{\beta}\right)$ and $\ell\left(\tilde{\beta}_{-i}\right)$ are calculated with φ assumed to be 1. In addition, the deviance values for both unconstrained and constrained models are also calculated.

Then calculate the likelihood ratio statistic for each effect *i*.

1. If φ or *k* is estimated jointly with **β** by ML method or set to a fixed value,

   $$S_i = 2\left(\ell\left(\hat{\beta}\right) - \ell\left(\tilde{\beta}_{-i}\right)\right)$$

   Then S_i has an asymptotic chi-square distribution with degrees of freedom *r*, where *r* is equal to the rank of the $\mathbf{L}_i$ matrix.

2. If φ is specified from the full model by the deviance or Pearson chi-square divided by degrees of freedom,

   $S_i = \frac{2\left(\ell(\hat{\beta})-\ell(\tilde{\beta}_{-i})\right)}{\hat{\phi}}$ and $F_i = \frac{S_i}{r}$,

   respectively. Then S_i has an asymptotic chi-square distribution with degrees of freedom *r*. *F* has an asymptotic F distribution with *r* and $N-p_X$ degrees of freedom. Note for the negative binomial with the ancillary parameter *k* estimated by the ML method and with the scale parameter measured by the deviance or Pearson chi-square divided by its degrees of freedom, the degrees of freedom in the denominator for the *F* statistic are $N - p_X - 1$; for the binomial distribution with 0/1 binary response, the degrees of freedom for the denominator should be $n_S - p_X$; for the ordinal multinomial model, the degrees of freedom for the denominator should be $n_s(J-1)-(J-1+p_x)$.

Wald statistics. See the discussion of Wald statistics for Type I analysis above. L_i is the type III test matrix for the *i*th effect.

Sequential Quadratic Programming

Sequential quadratic programming is a method of linear constrained optimization that can be applied to type III analysis and custom tests. It has the general form:

$$\max_\beta \ell(\mu(\beta); y)$$
$$\text{s.t.} L\beta = K,$$

where **L** is a *r*×*p* full row rank hypothesis matrix and **K** is a *r*×1 resulting vector. Note for the ordinal multinomial model, **L** is a $r \times (J-1+p)$ full row rank hypothesis matrix for ordinal multinomial. To simplify the notation, we write the log-likelihood as $\ell(\beta)$ here. The Lagrange function with an *r*×1 vector of Lagrange multipliers is

$$L(\beta,\lambda) = \ell(\beta) + \lambda^T(L\beta - K) = \ell(\beta) + \sum_{i=1}^{r} \lambda_i(L_i\beta - K_i)$$

The first order conditions with respect to β and **λ** are

$$\begin{cases} \frac{\partial L(\beta,\lambda)}{\partial\beta} = \frac{\partial\ell(\beta)}{\partial\beta} + L^T\lambda = [0]_{p\times 1}\text{(dual feasibility equations)} \\ \frac{\partial L(\beta,\lambda)}{\partial\lambda} = L\beta - K = [0]_{r\times 1}\text{(primal feasibility equations)} \end{cases}$$

We would like to find a solution (β^*andλ^*) for the above KKT (Karush-Kuhn-Tucker) equations, which is a set of *p* + *r* equations. The method usually used is extensions of Newton Raphson's method. First we replace the log-likelihood with its second-order Taylor approximation near to reform the problem

$$\max_\delta \ell(\beta+\delta) = \ell(\beta) + \mathbf{s}(\beta)^T\delta + \tfrac{1}{2}\delta^T\mathbf{H}(\beta)\delta$$
$$\text{s.t. } L(\beta+\delta) = K$$

This is a quadratic optimization problem with variable **δ**. We use the feasible start method solve the KKT equations.

Feasible Start Method

The feasible β values satisfy β=**K** and belong to the domain of the log-likelihood. If the initial values of β are feasible, then **L**δ=**0** and the constrained problem is almost the same as the Newton-Raphson method without constraints. The iterative process can be outlined briefly as follows:

1. Find initial values $\beta^{(0)}$ with $L\beta^{(0)} = K$ (see below), then compute $\ell(\beta^{(0)})$, **s** $(\beta^{(0)})$ and **H** $(\beta^{(0)})$.
2. Let $\xi = 1$.
3. Find a solution of $\delta^{(i-1)}$and$\lambda^{(i)}$ for the following KKT equations:

$$\begin{bmatrix} \mathbf{H}(\beta^{(i-1)}) & L^T \\ L & 0 \end{bmatrix} \begin{bmatrix} \delta^{(i-1)} \\ \lambda^{(i)} \end{bmatrix} = \begin{bmatrix} -\mathbf{s}(\beta^{(i-1)}) \\ 0 \end{bmatrix}_{(p+r)\times 1}.$$

4. Compute estimates of ith iteration:

   $\beta^{(i)} = \beta^{(i-1)} + \xi\delta^{(i-1)}$, then compute $\ell\left(\beta^{(i)}\right)$.

5. Use step-halving method if $\ell\left(\beta^{(i)}\right) < \ell\left(\beta^{(i-1)}\right)$: reduce ξ by half and repeat step (4). If the maximum number of steps in step-halving is reached but the log-likelihood is not improved, stop.

6. Check if convergence criteria (see below) are met. If they are or the maximum number of iterations is reached, stop. The final vector of estimates is denoted by $\tilde{\beta}^{(\tilde{T})}$. Otherwise, go back to step (2).

Initial Values

The initial values for constrained optimization under the constraint $L_i\beta = 0$ for each effect i in type III analysis can be obtained by applying the method the initial values obtained for unconstrained parameter estimation with a constraint that type III contrast equals zero. Specifically, follow steps (1) to (3) of the method for computing initial values for parameter estimation (see the appropriate section under "Parameter estimation"), then solve the following KKT equations

$$\begin{bmatrix} X^T\tilde{W}_e X & L_i^T \\ L_i & 0 \end{bmatrix} \begin{bmatrix} \beta^{(0)} \\ \lambda^{(0)} \end{bmatrix} = \begin{bmatrix} X^T\tilde{W}_e z \\ 0 \end{bmatrix}_{(p+r)\times 1}.$$

The solution will be a feasible point. Then the initial value for φ or k can be obtained as before. For the ordinal multinomial model, initial values for unconstrained parameter estimation can be applied here because they are feasible values.

Convergence Criteria

We only consider the log-likelihood convergence criterion for the constrained optimization problem to speed the iterative processes here. If ϵ_ℓ and relative or absolute change is user-specified for the unconstrained optimization problem, then they will be also apply here; otherwise, the internal default values will be used.

Estimated Marginal Means

There are two types of estimated marginal means (EMMEANS) calculated here. One corresponds to the specified factors for the linear predictor of the model and the other corresponds to those for the response of the model. EMMEANS for the predictor are equivalent to LSMEANS (least-squares means) used by SAS. EMMEANS for the response are equivalent to conditional marginals used by SUDAAN or conditional prediction used by Lane and Nelder (1982).

EMMEANS are based on the estimated cell means. For a given fixed set of factors, or their interactions, we estimate marginal means as the mean value averaged over all cells generated by the rest of the factors in the model. Covariates may be fixed at any specified value. If not specified, the value for each covariate is set to its overall mean estimate.

For the ordinal multinomial model, EMMEANS are not available.

EMMEANS for the Linear Predictor

Calculating EMMEANS for the Linear Predictor

EMMEANS for the linear predictor are based on the link function transformation. They are computed for the linear predictor. Since the given model with respect to the linear predictor is a linear model, the way to construct **L** is the same as that for the GLM procedure. Each EMMEAN for the linear predictor is constructed such that **LB** is estimable.

Briefly, for a given set of factors in the model, a vector of EMMEANS for the linear predictor is created for all combined levels of the factors. Assume there are *r* levels. This *r*×1 vector can be expressed in the form $\hat{\mathbf{v}} = \mathbf{L}\hat{\beta}$. The variance matrix of $\hat{\mathbf{v}}$ is then computed by

$$V(\hat{\mathbf{v}}) = \mathrm{L}\Sigma\mathrm{L}^{\mathrm{T}}$$

The standard error for the *j*th element of $\hat{\mathbf{v}}$ is the square root of the *j*th diagonal element of $V(\hat{\mathbf{v}})$. Let the *j*th element of $\hat{\mathbf{v}}$ and its standard error be $\hat{v}_j$ and $\hat{\sigma}_{v_j}$, respectively, then the corresponding $100(1-\alpha)\%$ Wald confidence interval for $v_j, j = 1, \dots, r$, is given by

$$\left(\hat{v}_j - z_{1-\alpha/2}\hat{\sigma}_{v_j}, \hat{v}_j + z_{1-\alpha/2}\hat{\sigma}_{v_j}\right)$$

Comparing EMMEANS for the Linear Predictor

We can compare EMMEANS for the linear predictor based on a selected contrast type, for which a set of contrasts for the factor is created. Let this set of contrasts define matrix **C** used for testing the hypothesis $H_0 : \mathrm{Cv} = 0$. A Wald statistic is used for testing given set of contrasts for the factor as follows:

$$S = (\mathrm{C}\hat{\mathbf{v}})^{\mathrm{T}}\left(\mathrm{C}V(\hat{\mathbf{v}})\mathrm{C}^{\mathrm{T}}\right)^{-}(\mathrm{C}\hat{\mathbf{v}})$$

S has an asymptotic chi-square distribution with r_I degrees of freedom, where $r_I = \mathrm{rank}\left(\mathrm{C}V(\hat{\mathbf{v}})\mathrm{C}^{\mathrm{T}}\right)$. The *p*-values can be calculated accordingly. Note that adjusted *p*-values based on multiple comparisons adjustments won't be computed for the overall test.

Each row $\mathrm{c}_i^{\mathrm{T}}$ of matrix **C** is also tested separately. The estimate for the *i*th row is given by $\mathrm{c}_i^{\mathrm{T}}\hat{\mathbf{v}}$ and its standard error by $\sqrt{\mathrm{c}_i^{\mathrm{T}}V(\hat{\mathbf{v}})\mathrm{c}_i}$. The corresponding 100(1 – α)% Wald confidence interval for is given by

$$\left(\mathrm{c}_i^{\mathrm{T}}\hat{\mathbf{v}} \pm z_{1-\alpha/2}\sqrt{\mathrm{c}_i^{\mathrm{T}}V(\hat{\mathbf{v}})\mathrm{c}_i}\right)$$

The Wald statistic for $H_0 : \mathrm{c}_i^{\mathrm{T}}\mathbf{v} = 0$ is

$$S_i = \left(\frac{\mathrm{c}_i^{\mathrm{T}}\hat{\mathbf{v}}}{\sqrt{\mathrm{c}_i^{\mathrm{T}}V(\hat{\mathbf{v}})\mathrm{c}_i}}\right)^2$$

It has an asymptotic chi-square distribution with 1 degree of freedom. The *p*-values can be calculated accordingly. In addition, adjusted *p*-values for multiple comparisons can also computed.

EMMEANS for the Response

EMMEANS for the response are based on the original scale of the dependent variable except for the binomial response with events/trials format (see note below). They can be defined as the estimator of the expected response for a subject conditional on his/her belonging to a specified effect and having the averages of covariates. Note that as for the so called predicted marginals used by SUDAAN or marginal prediction used by Lane and Nelder (1982), we will not offer them because they require some assumptions about the distribution of the predictor variables.

Calculating EMMEANS for the Response

The way to construct EMMEANS for the response is based on EMMEANS for the linear predictor. Let $\hat{\mathbf{M}}_c$ be EMMEANS for the response and it is defined as

$$\hat{\mathbf{M}}_c = g^{-1}\left(\mathrm{L}\hat{\beta}\right)$$

The variance of EMMEANS for the response is

$$V\left(\hat{M}_c\right) = diag\left(\frac{\partial g^{-1}(\hat{v}_j)}{\partial \hat{v}_j}\right) L\Sigma L^T diag\left(\frac{\partial g^{-1}(\hat{v}_j)}{\partial \hat{v}_j}\right)$$

where $diag\left(\partial g^{-1}(\hat{v}_j)/\partial \hat{v}_j\right)$ is a *r*×*r* matrix and $\partial g^{-1}(\hat{v}_j)/\partial \hat{v}_j$ is the derivative of the inverse of the link with respect to the *j*th value in $\hat{\mathbf{v}}$ and $\partial_{g^{-1}}(\hat{v}_j)/\partial\hat{v}_j = 1/\acute{g}(\widehat{M}_{cj})$ where $\acute{g}(\widehat{M}_{cj})$ is from Table 46-8. The standard error for the *j*th element of $\widehat{\mathrm{M}}_c$ and the corresponding confidence interval are calculated similarly to those of $\hat{\mathbf{v}}$. For more information, see the topic “EMMEANS for the Linear Predictor”.

Note: $\hat{\mathbf{M}}_c$ is EMMEANS for the proportion, not for the number of events when events and trials variables are used for the binomial distribution.

Comparing EMMEANS for the Response

This is similar to comparing EMMEANS for the linear predictor; just replace $\hat{\mathbf{v}}$ with $\hat{\mathbf{M}}_c$ and $V(\hat{\mathbf{v}})$ with $V\left(\hat{M}_c\right)$. For more information, see the topic “EMMEANS for the Linear Predictor”.

Multiple Comparisons

The hypothesis $H_0 : \mathbf{Cv} = \mathbf{0}$ can be tested using the multiple row hypotheses testing technique. Let $\mathbf{c}_i^T$ be the *i*th row vector of matrix **C**. The *i*th row hypothesis is $H_{0i} : \mathbf{c}_i^T\mathbf{v} = 0$. Testing H_0 is the same as testing multiple non-redundant row hypotheses $\{H_{0i}^*\}_{i=1}^R$ simultaneously, where *R* is the number of non-redundant row hypotheses, and H_{0i}^* represents the *i*th non-redundant hypothesis. A hypothesis H_{0i} is redundant if there exists another hypothesis $H_{0j}, j \neq i$ such that $c_i = ac_j, a \neq 0$.

Adjusted p-values. For each individual hypothesis H_{0i}, test statistics can be calculated. Let p_i denote the p-value for testing H_{0i} and p_i^* denote the adjusted p-value. The conclusion from multiple testing is, at level α (the family-wise type I error),

reject $H_{0i} : \mathbf{c}_i^T \mathbf{v} = 0$ if $p_i^* < \alpha$;

reject $H_0 : \mathbf{C}\mathbf{v} = \mathbf{0}$ if $\min_i (p_i^*) < \alpha$.

Several different methods to adjust p-values are provided here. Please note that if the adjusted p-value is bigger than 1, it is set to 1 in all the methods.

Adjusted confidence intervals. Note that if confidence intervals are also calculated for the above hypothesis, then adjusting confidence intervals is required to correspond to adjusted p-values. The only item needed to be adjusted in the confidence intervals is the critical value from the standard normal distribution. Assume that the original critical value is $z_{1-\alpha/2}$ and the adjusted critical value is z^*.

LSD (Least Significant Difference)

The adjusted p-values are the same as the original p-values:

$$p_i^* = p_i$$

The adjusted critical value is:

$$z^* = z_{1-\frac{\alpha}{2}}.$$

Bonferroni

The adjusted p-values are:

$$p_i^* = Rp_i$$

The adjusted critical value is:

$$z^* = z_{1-\frac{\alpha}{2R}}.$$

Sidak

The adjusted p-values are:

$$p_i^* = 1 - (1 - p_i)^R$$

The adjusted critical value is:

$$z^* = z_{1-\frac{1-(1-\alpha)^{1/R}}{2}}.$$

Sequential Bonferroni

The adjusted *p*-values are:

$$p^*_{(i)} = \begin{cases} Rp_{(1)} & i = 1 \\ \max\left((R-i+1)\,p_{(i)}, p^*_{(i-1)}\right) & i \geq 2 \end{cases}$$

The adjusted critical values will correspond to the ordered adjusted *p*-values as follows:

$$z^*_{(i)} = \begin{cases} z_{1-\frac{\alpha}{2R}} & \text{if } i = 1 \\ z_{1-\frac{\alpha}{2(R-i+1)}} & \text{if } p^*_{(i)} = (R-i+1)p_{(i)}, \text{ for } i \geq 2 \\ z^*_{(i-1)} & \text{if } p^*_{(i)} = p^*_{(i-1)}, \text{ for } i \geq 2 \end{cases}.$$

Sequential Sidak

The adjusted *p*-values are:

$$p^*_{(i)} = \begin{cases} 1-\left(1-p_{(1)}\right)^R & i = 1 \\ \max\left(1-\left(1-p_{(i)}\right)^{R-i+1}, p^*_{(i-1)}\right) & i \geq 2 \end{cases}$$

The adjusted critical values will correspond to the ordered adjusted *p*-values as follows:

$$z^*_{(i)} = \begin{cases} z_{1-\frac{1-(1-\alpha)^{1/R}}{2}} & \\ z_{1-\frac{1-(1-\alpha)^{1/(R-i+1)}}{2}} & \text{if } p^*_{(i)}=1-\left(1-p_{(i)}\right)^{R-i+1}, \text{ for } i \geq 2 \\ z^*_{(i-1)} & \text{if } p^*_{(i)}=p^*_{(i-1)}, \text{ for } i \geq 2 \end{cases}.$$

Comparison of Adjustment Methods

A multiple testing procedure tells not only if H_0 is rejected, but also if each individual H_{0i} is rejected. All the methods, except LSD, control the family-wise type I error for testing H_0; that is, the probability of rejecting at least one individual hypothesis under H_0. In addition, sequential methods also control the family-wise type I error for testing any subset of H_0.

LSD is the one without any adjustment, it rejects H_0 too often. It does not control the family-wise type I error and should never be used to test H_0. It is provided here mainly for reference.

Bonferroni is conservative in the sense that it rejects H_0 less often than it should. In some situations, it becomes extremely conservative when test statistics are highly correlated.

Sidak is also conservative in most cases, but is less conservative than Bonferroni. It gives the exact type I error when test statistics are independent.

Sequential Bonferroni is as conservative as the Bonferroni in terms of testing H_0 because the smallest adjusted *p*-value used in making decision is the same in both methods. But in term of testing individual H_{0i}, it is less conservative than the Bonferroni. Sequential Bonferroni rejects at least as many individual hypotheses as Bonferroni.

Sequential Sidak is as conservative as the Sidak in terms of testing H_0, but less conservative than the Sidak in terms of testing individual H_{0i}. Sequential Sidak is less conservative than sequential Bonferroni.

Scoring

Scoring is defined as assigning one or more values to a case in a data set. Two types are considered here: predicted values and model diagnostics.

Predicted Values

Due to the non-linear link functions, the predicted values will be computed for the linear predictor and the mean of the response separately. Also, since estimated standard errors of predicted values of linear predictor are calculated, the confidence intervals for the mean are obtained easily.

Predicted values are still computed as long all the predictor variables have non-missing values in the given model.

Predicted Values of the Linear Predictors

$$\hat{\eta}_i = x_i^T \hat{\beta} + o_i$$

For the ordinal multinomial model, a predicted value of the linear predictor for category *j* is given by

$$\hat{\eta}_{i,j} = \hat{\psi}_j - \mathrm{x}_i^T \hat{\beta} + o_i, j = 1, \ldots, J-1.$$

Estimated Standard Errors of Predicted Values of the Linear Predictors

$$\hat{\sigma}_\eta = \sqrt{x_i^T \Sigma x_i}$$

For the ordinal multinomial model, the estimated standard error of $\hat{\eta}_{i,j}$ is given by

$$\hat{\sigma}_{\eta_{i,j}} = \sqrt{\left(1, -\mathrm{x}_i^T\right) \Sigma_j \begin{pmatrix} 1 \\ -\mathrm{x}_i \end{pmatrix}}, j = 1, \ldots, J-1,$$

where Σ_j is a reduced parameter estimates covariance $(1 + p)\times(1 + p)$ matrix from Σ. Suppose Σ for ordinal multinomial models has the following form:

$$
\begin{aligned}
\Sigma &= \begin{bmatrix} \Sigma_{\psi,\psi} & \Sigma_{\psi,\beta} \\ \Sigma_{\beta,\psi} & \Sigma_{\beta,\beta} \end{bmatrix} \\
&= \begin{bmatrix}
\begin{bmatrix} \sigma_{1,1} & \cdots & \sigma_{1,(J-1)} \\ \vdots & \ddots & \vdots \\ \sigma_{(J-1),1} & \cdots & \sigma_{(J-1),(J-1)} \end{bmatrix} & \begin{bmatrix} \sigma_{1,J} & \cdots & \sigma_{1,(J-1+p)} \\ \vdots & \ddots & \vdots \\ \sigma_{(J-1),J} & \cdots & \sigma_{(J-1),(J-1+p)} \end{bmatrix} \\
\begin{bmatrix} \sigma_{J,1} & \cdots & \sigma_{J,(J-1)} \\ \vdots & \ddots & \vdots \\ \sigma_{(J-1+p),1} & \cdots & \sigma_{(J-1+p),(J-1)} \end{bmatrix} & \begin{bmatrix} \sigma_{J,J} & \cdots & \sigma_{J,(J-1+p)} \\ \vdots & \ddots & \vdots \\ \sigma_{(J-1+p),J} & \cdots & \sigma_{(J-1+p),(J-1+p)} \end{bmatrix}
\end{bmatrix}
\end{aligned}
$$

then Σ_j will have the following form as it takes the corresponding elements in the *j*th row and column of Σ and $\Sigma_{\beta,\beta}$:

$$
\begin{aligned}
\Sigma_j &= \begin{bmatrix}
\sigma_{j,j} & \begin{bmatrix} \sigma_{j,J} & \cdots & \sigma_{j,(J-1+p)} \end{bmatrix} \\
\begin{bmatrix} \sigma_{J,j} \\ \vdots \\ \sigma_{(J-1+p),j} \end{bmatrix} & \Sigma_{\beta,\beta}
\end{bmatrix} \\
&= \begin{bmatrix}
\sigma_{j,j} & \sigma_{j,J} & \cdots & \sigma_{j,(J-1+p)} \\
\sigma_{J,j} & \sigma_{J,J} & \cdots & \sigma_{J,(J-1+p)} \\
\vdots & \vdots & \ddots & \vdots \\
\sigma_{(J-1+p),j} & \sigma_{(J-1+p),J} & \cdots & \sigma_{(J-1+p),(J-1+p)}
\end{bmatrix}
\end{aligned}
$$

Predicted Values of the Means

$$\hat{\mu}_i = g^{-1}\left(x_i^{\mathrm{T}}\hat{\beta} + o_i\right)$$

where g^{-1} is the inverse of the link function. For binomial response with 0/1 binary response variable, this the predicted probability of category 1.

For the ordinal multinomial model, a predicted value of the cumulative response probability for category *j* is given by

$$\hat{\gamma}_{i,j} = g^{-1}\left(\hat{\psi}_j - \mathrm{x}_i^{\mathrm{T}}\beta + o_i\right), \quad j = 1, \ldots, J-1$$

Confidence Intervals for the Means

Approximate 100(1−α)% confidence intervals for the mean can be computed as follows

$$g^{-1}\left(x_i^{\mathrm{T}}\hat{\beta} + \mathrm{o}_i \pm z_{1-\alpha/2}\hat{\sigma}_\eta\right)$$

Approximate 100(1−α)% confidence intervals for the cumulative response probability can be computed as follows

$$g^{-1}\left(\hat{\psi}_j - \mathrm{x}_i^{\mathrm{T}}\beta + o_i \pm z_{1-\alpha/2}\hat{\sigma}_{\eta_{i,j}}\right),\ \ j = 1,\ldots,J-1.$$

If either endpoint in the argument is outside the valid range for the inverse link function, the corresponding confidence interval endpoint is set to a system missing value.

Predicted category for binomial and multinomial distributions

For the binomial distribution with 0/1 binary response variable, the predicted category is

$$c(\mathrm{x}_i) = \begin{cases} 1 \text{ (or success) if } \hat{\mu}_i \geq 0.5 \\ 0 \text{ (or failure) otherwise} \end{cases}.$$

For the ordinal multinomial model, the predicted category is the one with the highest predicted probability; that is

$$c(\mathrm{x}_i) = arg\max_j \hat{\pi}_{i,j}.$$

- If there are ties in determining $c(\mathrm{x}_i)$ choose the category with highest $N_j = \sum_{i=1}^{n} f_i y_{i,j}$.
- If there are still ties, choose the one with lowest category number.

Diagnostics

In addition to predicted values, we can calculate some values which would be good for model diagnostics for all distributions except the ordinal multinomial.

Leverage

The leverage value h_i is defined as the ith diagonal element of the hat matrix

$$\mathrm{H} = \mathrm{W}_\epsilon^{1/2}\mathrm{X}\left(\mathrm{X}^{\mathrm{T}}\mathrm{W}_\epsilon\mathrm{X}\right)^{-}\mathrm{X}^{\mathrm{T}}\mathrm{W}_\epsilon^{1/2},$$

where the ith diagonal element for W_ϵ is

$$w_{e,i} = \frac{\omega_i}{\phi}\cdot\frac{1}{V(\mu_i)\left(g'(\mu_i)\right)^2}.$$

Raw Residuals

$$r_i^R = y_i - \hat{\mu}_i$$

where y_i is the ith response and $\hat{\mu}_i$ is the corresponding predicted mean. Note for binomial response with a binary format, y values are 0 for the reference category and 1 for the category we are modeling.

Pearson Residuals

The Pearson residual is the square root of the *i*th contribution to the Pearson chi-square.

$$r_i^P = (y_i - \hat{\mu}_i)\sqrt{\frac{\omega_i}{V(\hat{\mu}_i)}}$$

Deviance Residuals

The deviance residual is the square root of the contribution of the *i*th observation to the deviance, with the sign of the raw residual.

$$r_i^D = \text{sign}\,(y_i - \hat{\mu}_i)\sqrt{d_i}$$

where d_i is the contribution of the *i*th case to the deviance and sign() is 1 if its argument is positive and −1 if it is negative.

Standardized (and Studentized) Pearson Residuals

$$r_i^{SP} = (y_i - \hat{\mu}_i)\sqrt{\frac{\omega_i}{\phi V(\hat{\mu}_i)(1-h_i)}} = r_i^P\sqrt{\frac{1}{\phi(1-h_i)}}$$

Standardized (and Studentized) Deviance Residuals

$$r_i^{SD} = \text{sign}(y_i - \hat{\mu}_i)\sqrt{d_i}\sqrt{\frac{1}{\phi(1-h_i)}} = r_i^D\sqrt{\frac{1}{\phi(1-h_i)}}$$

Likelihood Residuals

$$r_i^L = \text{sign}(y_i - \hat{\mu}_i)\sqrt{h_i\left(r_i^{SP}\right)^2 + (1-h_i)\left(r_i^{SD}\right)^2}$$

Cook’s Distance

$$C_i = \frac{1}{p_x}\frac{h_i}{1-h_i}\left(r_i^{SP}\right)^2$$

Generalized Estimating Equations

Generalized estimating equations (GEE) extend the GZLM algorithm to accommodate correlated data. The algorithms of generalized estimating equations are based on Liang and Zeger (1986) and Diggle, Heagerty, Liang and Zeger (2002).

Data Format

The data formation in GEE is very different from that in GZLM, so the data used in GEE need to be formatted appropriately. The structure of the correlated data has two dimensions: there are some independent subjects (the subject effect) where each subject has correlated measurements (the within-subject effect).

The subject effect can be a single variable or a list of variables connected by asterisks (*). In general, the number of subjects equals to the number of distinct combinations of values of the variables except under some circumstances (see example below).

The within-subject effect defines the ordering of measurements within subjects. If specified, it can be a single variable or a list of variables connected by asterisks (*). The start and end of the within-subject effect could be different for each subject, so the whole data file is checked to find the complete set of measurements which include all distinct combinations of values of within-subject effect from all subjects. The dimension of the complete set of measurement will be the dimension of the working correlation matrix (see "Model " for more information). If some measurements do not appear in the data for some subjects, then the existing measurements are ordered and the omitted measurements are treated as missing values.

Note that the within-subject effect might not be equally spaced. This is relevant for the time dependent working correlation structures. We will assume that the lags based on the data ordered by the within-subject effect are appropriate and fit the model.

The data have to be properly grouped by the subject effect and sorted by the within-subject effect if it exists. If you specify not to sort the data file (`SORT=NO`), we assume that the cases in the data file are pre-sorted. If you specify to sort the data file (`SORT=YES`), the data will be sorted internally by the subject effect and the within-subject effect, if the within-subject effect is specified.

Consider the following artificial data:

Table 46-19
Example data

center	**id**	**year**	**y**	**x1**
A	11	91	4	0
A	11	93	5	1
A	12	93	5	1
A	11	94	6	1
A	12	94	6	0
A	12	95	7	1
B	1	91	6	0
B	1	94	3	0
B	2	93	5	1
B	2	95	7	0
B	2	94	8	1

Suppose the subject effect is specified as *center***id*. The number of subjects or clusters depends on whether the within-subject effect is specified or not and whether the data are indicated to be sorted or not. Thus we consider the following cases:

Within-subject effect is specified, data will be sorted by procedure (SORT=YES)

There are four distinct combinations for the subject effect: (center*id) = (A*11), (A*12), (B*1), (B*2). The data will be grouped internally based on them, so the number of clusters or groups = 4. The complete set of measurements = (91, 93, 94, 95) with the dimension = 4, the maximum and minimum sizes of the within-subject effect are 3 and 2, respectively. Note the measurements for the within-subject effect are not equally spaced, we assume the measurements are spaced appropriately when calculating the time dependent working correlation structures.

Figure 46-1
GEE model information about the data

Number of Levels	Subject Effect	center	2
		id	4
	Within-Subject Effect	year	4
Number of Subjects			4
Number of Measurements per Subject	Minimum		2
	Maximum		3
Correlation Matrix Dimension			4

The data file is then organized internally as follows (subject and withinsubject are internal variables):

Table 46-20
Data file structure

center	**id**	**year**	**y**	**x1**	**subject**	**withinsubject**
A	11	91	4	0	1	1
A	11	93	5	1	1	2
A	11	94	6	1	1	3
A	11	95	.	.	1	4
A	12	91	.	.	2	1
A	12	93	5	1	2	2
A	12	94	6	0	2	3
A	12	95	7	1	2	4
B	1	91	6	0	3	1
B	1	93	.	.	3	2
B	1	94	3	0	3	3
B	1	95	.	.	3	4
B	2	91	.	.	4	1
B	2	93	5	1	4	2
B	2	94	8	1	4	3
B	2	95	7	0	4	4

Within-subject effect is not specified, data will be sorted by procedure (SORT=YES)

There are still 4 distinct combinations for the subject effect and the number of clusters or groups = 4. The dimension of the working correlation matrix is3 which is determined by the maximum size of measurements from all subjects, the maximum and minimum sizes of repeated measurements are 3 and 2, respectively. A summary is as follows:

Figure 46-2
GEE model information about the data

Number of Levels	Subject Effect	center	2
		id	4
Number of Subjects			4
Number of Measurements per Subject	Minimum		2
	Maximum		3
Correlation Matrix Dimension			3

The data file is then organized internally as follows (subject and withinsubject are internal variables):

Table 46-21
Data file structure

center	**id**	**year**	**y**	**x1**	**subject**	**withinsubject**
A	11	91	4	0	1	1
A	11	93	5	1	1	2
A	11	94	6	1	1	3
A	12	93	5	1	2	1
A	12	94	6	0	2	2
A	12	95	7	1	2	3
B	1	91	6	0	3	1
B	1	94	3	0	3	2
B	1	.	.	.	3	3
B	2	93	5	1	4	1
B	2	95	7	0	4	2
B	2	94	8	1	4	3

Data will not be sorted by procedure (SORT=NO)

When data are not to be sorted, the within-subject effect will be ignored whether specified or not.

From the original data file, we notice that the same combinations of values for the subject effect are in different blocks, so they will be considered as different clusters. For example:

The 1st cluster (certer*id = A *11) includes the 1st and 2nd observations.

The 2nd cluster (center*id = A*12) includes the 3rd observation.

The 3rd cluster (center*id = A*11) includes the 4th observation.

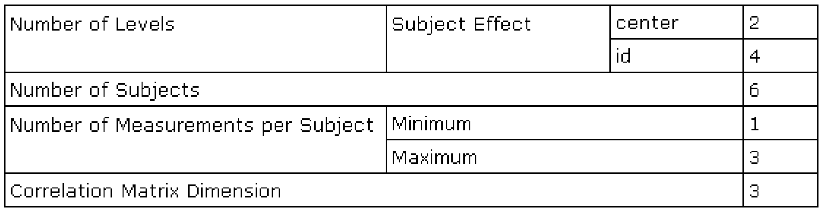
The 4th cluster (center*id = A*12) includes the 5th and 6th observations.

The 5th cluster (center*id = B*1) includes the 7th and 8th observations.

The 6th cluster (center*id = B*2) includes the 9th, 10th and 11th observations.

So the number of clusters =6. The dimension of the working correlation matrix is 3, the maximum and minimum sizes of repeated measurements are 3 and 1, respectively. A summary is as follows:

Figure 46-3
GEE model information about the data



Number of Levels	Subject Effect	center	2
		id	4
Number of Subjects			6
Number of Measurements per Subject	Minimum		1
	Maximum		3
Correlation Matrix Dimension			3

The data file is then organized internally as follows (subject and withinsubject are internal variables):

Table 46-22
Data file structure

center	id	year	y	x1	subject	withinsubject
A	11	91	4	0	1	1
A	11	93	5	1	1	2
A	11	.	.	.	1	3
A	12	93	5	1	2	1
A	12	.	.	.	2	2
A	12	.	.	.	2	3
A	11	94	6	1	3	1
A	11	.	.	.	3	2
A	11	.	.	.	3	3
A	12	94	6	0	4	1
A	12	95	7	1	4	2
A	12	.	.	.	4	3
B	1	91	6	0	5	1
B	1	94	3	0	5	2
B	1	.	.	.	5	3
B	2	93	5	1	6	1
B	2	95	7	0	6	2
B	2	94	8	1	6	3

After reformatting the data, we assume there are $i = 1, \ldots, K$ subjects or clusters where each subject or cluster has $t = 1, \ldots, n_i$ correlated measurements. Note now that $n_1 = n_2 = \ldots = n_K$. The following notations should be applied to the reformatted data, not the original data.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 46-23
Notation

Notation	Description
K	Number of subjects (clusters or groups) in the data set. It is an integer and $K \geq 1$.
n_i	Number of complete measurements on the ith subject. It is an integer and $n_i \geq 1$.
n	Total number of measurement; $n = \Sigma_{i=1}^{K} n_i$. It is an integer and $n \geq 1$.
p	Number of parameters (including the intercept, if exists) in the model. It is an integer and $p \geq 1$.
p_x	Number of non-redundant columns in the design matrix. It is an integer and $p_x \geq 1$.
y	$n \times 1$ dependent variable vector. $\mathrm{y} = \left[\mathrm{y}_1^T, \ldots, \mathrm{y}_K^T\right]^T$ with $\mathrm{y}_i = [y_{i1}, \ldots, y_{in_i}]^T$ for each i.
r	$n \times 1$ vector of events for the binomial distribution; it usually represents the number of "successes". All elements are non-negative integers.

Notation	Description
m	$n \times 1$ vector of trials for the binomial distribution. All elements are positive integers and $m_i \geq r_i$, i=1,...,n.
μ	$n \times 1$ vector of expectations of the dependent variable.
η	$n \times 1$ vector of linear predictors.
X	$n \times p$ design matrix. The vector for the tth measurement on the ith subject is $x_{it} = [x_{it1}, \ldots, x_{itp}]^T$, i = 1, …, K and t = 1, …, n_i with $x_{it1} = 1$ if the model has an intercept.
β	$p \times 1$ vector of unknown parameters. The first element in β is the intercept, if there is one.
ω	$n \times 1$ vector of scale weights. If an element is less than or equal to 0 or missing, the corresponding case is not used.
f	$n \times 1$ vector of frequency counts. Non-integer elements are treated by rounding the value to the nearest integer. For values less than 0.5 or missing, the corresponding cases are not used.
N	Effective sample size. $N = \sum_{i=1}^{n} f_i n_i$.

Model

GEE offers the same link functions and distributions as those for GZLM. The generalized estimating equation is given by

$$\mathbf{s}(\beta) = \sum_{i=1}^{K} f_i \left(\frac{\partial \boldsymbol{\mu}_i}{\partial \beta}\right)^T \mathbf{V}_i^{-}(\mathbf{y}_i - \boldsymbol{\mu}_i) = [0]_{p\times 1},$$

where

$$\begin{aligned}\left(\frac{\partial \boldsymbol{\mu}_i}{\partial \beta}\right) &= \left(\frac{\partial \boldsymbol{\mu}_i}{\partial \boldsymbol{\eta}_i}\right)\left(\frac{\partial \boldsymbol{\eta}_i}{\partial \beta}\right) \\ &= \text{diag}\left(\frac{1}{g'(\mu_{ij})}\right)\begin{pmatrix} x_{i11} & \cdots & x_{i1p} \\ \vdots & \ddots & \vdots \\ x_{in_i1} & \cdots & x_{in_ip}\end{pmatrix} \\ &= \begin{pmatrix} \frac{x_{i11}}{g'(\mu_{i1})} & \cdots & \frac{x_{i1p}}{g'(\mu_{i1})} \\ \vdots & \ddots & \vdots \\ \frac{x_{in_i1}}{g'(\mu_{in_i})} & \cdots & \frac{x_{in_ip}}{g'(\mu_{in_i})}\end{pmatrix}_{n_i\times p},\end{aligned}$$

$\text{diag}\left(1/g'(\mu_{ij})\right), j = 1, \ldots, n_i$, is an $n_i \times n_i$ matrix, $\mathbf{V}_i$ is the assumed covariance matrix of y_i and $\mathbf{V}_i^-$ is a generalized inverse of $\mathbf{V}_i$.

If the measurements within the ith subject are independent like in GZLM, then $\mathbf{V}_i$ is a diagonal matrix which can be decomposed into

$$\mathbf{V}_i = \left(\phi \mathbf{A}_i^{1/2} \mathbf{I}_{n_i} \mathbf{A}_i^{1/2}\right)_{n_i\times n_i},$$

where $\mathbf{A}_i = \text{diag}(V(\mu_{ij})/\omega_{ij}), j = 1, \ldots, n_i$, is an $n_i \times n_i$ matrix and I_{n_i} is an $n_i \times n_i$ identity matrix. However, if the measurements within the *i*th subject are correlated, we simply replace the identity matrix I_{n_i} with a more general correlation R(α)

$$V_i = \phi \mathbf{A}_i^{1/2} \mathbf{R}(\alpha) \mathbf{A}_i^{1/2},$$

where $\mathbf{R}(\alpha)$ is an $n_i \times n_i$ “working” correlation matrix which can be estimated through the parameter vector **α**. Since $\mathbf{R}(\alpha)$ usually doesn’t have a diagonal form, neither does V_i. If $\mathbf{R}(\alpha)$ is indeed the true correlation matrix of the y_i’s, then V_i is the true covariance matrix of y_i.

Ordinal Multinomial Model

For ordinal multinomial GEE models, we need to transform original response variable and define some notation as follows:

Table 46-24
Notation

Notation	Description
J	Number of values for the ordinal response. It is an integer and $J \geq 2$.
y	$n \times 1$ dependent variable vector. $y = \left[y_1^T, \ldots, y_K^T\right]^T$ with $y_i = [y_{i1}, \ldots, y_{in_i}]^T$ for each *i*.
z	$K \times n_i \times (J-1)) \times 1$ transformed dependent variable vector. $z = \left[z_1^T, \ldots, z_K^T\right]^T, z_i = \left[z_{i1}^T, \ldots, z_{in_i}^T\right]^T, z_{it} = [y_{it,1}, \ldots, y_{it,J-1}]^T$ and $y_{it,j} = \begin{cases} 1 \text{if} y_{it} = j \\ 0 \text{otherwise.} \end{cases}$
π	$K \times n_i \times (J-1)) \times 1$ conditional response probability vector. $\pi = \left[\pi_1^T, \ldots, \pi_K^T\right]^T, \pi_i = \left[\pi_{i1}^T, \ldots, \pi_{in_i}^T\right]^T$ and $\pi_{it} = [\pi_{it,1}, \ldots, \pi_{it,J-1}]^T$, where $\pi_{it,j}$ is the conditional response probability of measurement *t* on subject *i* for category *j* given the observed independent variable vector; that is, $\pi_{it,j} = P(y_i = j \mid x_{it})$ and $\pi_{it,j} = \gamma_{it,j} - \gamma_{it,j-1}$ for $j = 1, \ldots, J$.
$\gamma_{it,j}$	Conditional cumulative response probability of measurement *t* on subject *i* for category *j* given observed independent variable vector; that is, $\gamma_{it,j} = P(y_i \leq j \mid x_{it})$.
$\eta_{it,j}$	Linear predictor value of measurement *t* on subject *i* for category *j*. It is related to $\gamma_{it,j}$ through a cumulative link function.
Ψ	$(J-1) \times 1$ vector of threshold parameters; $\psi = (\psi_1, \psi_2, \ldots, \psi_{J-1})'$ and $\psi_1 < \psi_2 < \cdots < \psi_{J-1}$.
β	$p \times 1$ vector of regression parameters associated with model predictors; $\beta = (\beta_1, \beta_2, \ldots, \beta_p)'$.
B	$(J-1+p) \times 1$ vector of all parameters; $B = \left(\psi^T, \beta^T\right)^T$

The generalized estimating equation for estimating parameters **B** is given by

$$s(B) = \sum_{i=1}^{K} f_i \left(\frac{\partial \pi_i}{\partial B}\right)^T V_i^- (z_i - \pi_i) = [0]_{(J-1+p)\times 1},$$

where

$$\frac{\partial \pi_i}{\partial B} = \begin{bmatrix} \frac{\partial \pi_{i1,1}}{\partial \psi_1} & \cdots & \frac{\partial \pi_{i1,1}}{\partial \psi_{J-1}} & \frac{\partial \pi_{i1,1}}{\partial \beta_1} & \cdots & \frac{\partial \pi_{i1,1}}{\partial \beta_p} \\ & \ddots & & & \ddots & \\ \frac{\partial \pi_{i1,J-1}}{\partial \psi_1} & \cdots & \frac{\partial \pi_{i1,J-1}}{\partial \psi_{J-1}} & \frac{\partial \pi_{i1,J-1}}{\partial \beta_1} & \cdots & \frac{\partial \pi_{i,J-1}}{\partial \beta_p} \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial \pi_{in_i,1}}{\partial \psi_1} & \cdots & \frac{\partial \pi_{in_i,1}}{\partial \psi_{J-1}} & \frac{\partial \pi_{in_i,1}}{\partial \beta_1} & \cdots & \frac{\partial \pi_{in_i,1}}{\partial \beta_p} \\ & \ddots & & & \ddots & \\ \frac{\partial \pi_{in_i,J-1}}{\partial \psi_1} & \cdots & \frac{\partial \pi_{in_i,J-1}}{\partial \psi_{J-1}} & \frac{\partial \pi_{in_i,J-1}}{\partial \beta_1} & \cdots & \frac{\partial \pi_{in_i,J-1}}{\partial \beta_p} \end{bmatrix}_{(n_i \times (J-1)) \times (J-1+p)}$$

and for all $t = 1, \ldots, n_i$,

$$\frac{\partial \pi_{it,j}}{\partial \psi_j} = \frac{\partial \gamma_{it,j}}{\partial \eta_{it,j}}, j = 1, \ldots, J-1,$$

$$\frac{\partial \pi_{it,j}}{\partial \psi_{j-1}} = -\frac{\partial \gamma_{it,j-1}}{\partial \eta_{it,j-1}}, j = 2, \ldots, J-1,$$

$$\frac{\partial \pi_{it,j}}{\partial \psi_l} = 0, \text{ for } j < l \text{ or } j - l > 1, \ j = 1, \ldots, J-1 \text{ and } l = 1, \ldots, J-1,$$

$$\frac{\partial \pi_{it,1}}{\partial \beta_\ell} = -\frac{\partial \gamma_{it,1}}{\partial \eta_{it,1}} x_{it\ell}, \ell = 1, \ldots, p,$$

$$\frac{\partial \pi_{it,j}}{\partial \beta_\ell} = -\left(\frac{\partial \gamma_{it,j}}{\partial \eta_{it,j}} - \frac{\partial \gamma_{it,j-1}}{\partial \eta_{it,j-1}}\right) x_{it\ell}, j = 2, \ldots, J-1 \text{ and } \ell = 1, \ldots, p,$$

and V_i^- is a is a generalized inverse of V_i. Here

$$V_i = \phi \mathbf{A}_i^{1/2} \mathbf{R}_i \mathbf{A}_i^{1/2},$$

where

$$\mathbf{A}_i = \text{diag}(\mathbf{A}_{i1}, \ldots, \mathbf{A}_{in_i}),$$

$$\mathbf{A}_{it} = \frac{1}{\omega_{it}} \text{diag}\left(\pi_{it,1}(1 - \pi_{it,1}), \ldots, \pi_{it,J-1}(1 - \pi_{it,J-1})\right),$$

and

$$\mathbf{R}_i(\alpha) = \frac{1}{\phi} \begin{bmatrix} \mathbf{A}_{i1}^{-1/2} V_{i1} \mathbf{A}_{i1}^{-1/2} & \rho_{i12} & \cdots & \rho_{i1n_i} \\ \rho_{i21} & \mathbf{A}_{i2}^{-1/2} V_{i2} \mathbf{A}_{i2}^{-1/2} & \cdots & \rho_{i2n_i} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{in_i1} & \rho_{in_i2} & \cdots & \mathbf{A}_{in_i}^{-1/2} V_{in_i} \mathbf{A}_{in_i}^{-1/2} \end{bmatrix}$$

and note that there is a subscript *i* in $\mathbf{R}_i(\alpha)$ which means each subject has different working correlation matrix. In fact, only the diagonal blocks are different for different subjects, the off-diagonal blocks will be the same for all subjects. The diagonal blocks of $\mathbf{R}_i(\alpha)$, $\mathbf{A}_{it}^{-1/2}\mathrm{V}_{it}\mathbf{A}_{it}^{-1/2}$, with

$$\mathbf{A}_{it}^{-1/2} = \omega_{it}^{1/2} \times \mathrm{diag}\left(\{\pi_{it,1}(1-\pi_{it,1})\}^{-1/2}, \ldots, \{\pi_{it,J-1}(1-\pi_{it,J-1})\}^{-1/2}\right)$$

and

$$\mathrm{V}_{it} = \frac{\phi}{\omega_{it}} \times \left[\mathrm{diag}\left(\pi_{it,1}, \ldots, \pi_{it,J-1}\right) - \boldsymbol{\pi}_{it}\boldsymbol{\pi}_{it}^{\mathrm{T}}\right]$$
$$= \frac{\phi}{\omega_{it}} \times \begin{bmatrix} \pi_{it,1}(1-\pi_{it,1}) & -\pi_{it,1}\pi_{it,2} & \cdots & -\pi_{it,1}\pi_{it,J-1} \\ -\pi_{it,2}\pi_{it,1} & \pi_{it,2}(1-\pi_{it,2}) & \cdots & -\pi_{it,2}\pi_{it,J-1} \\ . & \ddots & \ddots & \ddots \\ -\pi_{it,J-1}\pi_{it,1} & -\pi_{it,J-1}\pi_{it,2} & \cdots & \pi_{it,J-1}(1-\pi_{it,J-1}) \end{bmatrix},$$

are specified entirely by $\boldsymbol{\pi}_i$. In particular, the diagonal elements of $\frac{1}{\phi}\mathbf{A}_{it}^{-1/2}\mathrm{V}_{it}\mathbf{A}_{it}^{-1/2}$ are 1 and off-diagonal (j,l) elements are

$$\frac{-\pi_{it,j}\pi_{it,l}}{\left\{\pi_{it,j}(1-\pi_{it,j})\pi_{it,l}(1-\pi_{it,l})\right\}^{1/2}}$$

which are not constant and depend on the categories *j* and *l* at measurement *t*. The unknown off-diagonal blocks of $\mathbf{R}_i(\alpha)$ are the $(J-1)\times(J-1)$ matrix $\rho_{iuv}(\alpha), u, v = 1, \ldots, n_i$, which we need to parameterize and estimate them.

Working correlation matrix

The working correlation matrix is usually unknown and should be estimated. We use the estimated Pearson residuals

$$r_{it} = (y_{it} - \mu_{it})\sqrt{\frac{\omega_{it}}{V(\mu_{it})}}$$

from the current fit of the model to estimate $\boldsymbol{\alpha}$.

For the ordinal multinomial model, we define estimated Pearson-like residuals as follows

$$r_{it,j} = (y_{it,j} - \pi_{it,j})\sqrt{\frac{\omega_{it}}{\pi_{it,j}(1-\pi_{it,j})}}$$

and the vector

$$\mathrm{r}_{it} = \left[r_{it,1}, \ldots, r_{it,J-1}\right]^{\mathrm{T}}$$

The following structures are available.

Independent

The independent correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 & \text{if } u = v \\ 0 & \text{otherwise} \end{cases}$$

For the ordinal multinomial model:

$$\rho_{iuv} = 0, u, v = 1, \ldots, n_i.$$

No parameters need to be estimated for this structure.

Exchangeable

The exchangeable correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 & \text{if } u = v \\ \alpha & \text{otherwise} \end{cases}$$

1 parameter is estimated as follows:

$$\alpha = \frac{\sum_{i=1}^{K} \sum_{t<t'} f_i r_{it} r_{it'}}{\left(\frac{1}{2}\sum_{i=1}^{K} f_i n'_i (n'_i - 1)\right) - p_x} \Big/ \left(\frac{1}{N' - p_x} \sum_{i=1}^{K} \sum_{t=1}^{n_i} f_i r_{it}^2\right),$$

where $N' = \Sigma_{i=1}^{K} f_i n'_i$ and n'_i is the number of non-missing measurements on the *i*th subject.

For the ordinal multinomial model:

$$\alpha = \frac{\sum_{i=1}^{K} f_i \sum_{t<t'} \frac{1}{2}\left(r_{it} r_{it'}^T + r_{it'} r_{it}^T\right)}{\left(\frac{1}{2}\sum_{i=1}^{K} f_i n'_i (n'_i - 1)\right) - (J - 1 + p_x)}$$

and $\rho_{iuv} = \alpha, u, v = 1, \ldots, n_i$ and $u \neq v$.

AR(1)

The first-order autoregressive correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 & \text{if } u = v \\ \alpha^{|u-v|} & \text{otherwise} \end{cases}$$

1 parameter is estimated as follows:

$$\alpha = \frac{\sum_{i=1}^{K}\sum_{t=1}^{n_i-1} f_i r_{it} r_{i,t+1}}{\left(\sum_{i=1}^{K} f_i n''_i\right) - p_x} \Big/ \left(\frac{1}{N' - p_x}\sum_{i=1}^{K}\sum_{t=1}^{n_i} f_i r_{it}^2\right),$$

where n''_i is the number of non-missing pairs used in the numerator part for the *i*th subject. If there is no non-missing measurement for the *i*th subject, $n''_i = n_i - 1$.

For the ordinal multinomial model:

$$\alpha = \frac{\sum_{i=1}^{K} f_i \sum_{t=1}^{n_i-1} \frac{1}{2}\left(r_{it} r_{it+1}^T + r_{it+1} r_{it}^T\right)}{\left(\sum_{i=1}^{K} f_i n''_i\right) - (J - 1 + p_x)}$$

and $\rho_{iuv} = \alpha^{|u-v|}, u, v = 1, \ldots, n_i$ and $u \neq v$.

M-dependent

The *m*-dependent correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 \text{ if } u = v \\ \alpha_{|u-v|} \text{ if } |u - v| \leq m \\ 0 \text{ otherwise} \end{cases}$$

m parameters are estimated as follows:

$$\alpha_j = \frac{\sum_{i=1}^{K}\sum_{t=1}^{n_i-j} f_i r_{it} r_{i,t+j}}{\left(\sum_{i=1}^{K} f_i n'''_{ij}\right) - p_x} \Big/ \left(\frac{1}{N' - p_x}\sum_{i=1}^{K}\sum_{t=1}^{n_i} f_i r_{it}^2\right),$$

where n'''_{ij} is the number of non-missing pairs for the *i*th subject in calculating α_j. If there is no non-missing measurement for the *i*th subject, $n'''_{ij} = n_i - j$.

For the ordinal multinomial model:

$$\alpha_j = \frac{\sum_{i=1}^{K} f_i \sum_{t=1}^{n_i-j} \frac{1}{2}\left(r_{it} r_{it+j}^T + r_{it+j} r_{it}^T\right)}{\left(\sum_{i=1}^{K} f_i n'''_{ij}\right) - (J - 1 + p_x)}$$

and $\rho_{iuv} = \begin{cases} \alpha_{|u-v|} \text{ if } |u - v| \leq m \\ 0 \text{ otherwise} \end{cases}$

Unstructured

The unstructured correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 & \text{if } u = v \\ \alpha_{uv} & \text{otherwise} \end{cases}$$

$\frac{1}{2}n_i(n_i-1)$ parameters are estimated as follows:

$$\alpha_{uv} = \frac{\sum_{i=1}^{K} f_i r_{iu} r_{iv}}{\left(\sum_{i=1}^{K} f_i I_{i,uv}\right) - p_x} \Big/ \left(\frac{1}{N' - p_x}\sum_{i=1}^{K}\sum_{t=1}^{n_i} f_i r_{it}^2\right),$$

where $I_{i,uv} = 1$ if the *i*th subject has non-missing measurements at times *u* and *v*; 0 otherwise

For the ordinal multinomial model:

$$\alpha_{uv} = \frac{\sum_{i=1}^{K} f_i \mathbf{r}_{iu} \mathbf{r}_{iv}^T}{\left(\sum_{i=1}^{K} f_i I_{i,uv}\right) - (J - 1 + p_x)}$$

and $\rho_{iuv} = \alpha_{uv}, u,v = 1,\ldots,n_i$ and $u \neq v$

Fixed

The fixed correlation structure is defined as:

$$R_{uv} = \begin{cases} 1 & \text{if } u = v \\ \gamma_{uv} & \text{otherwise} \end{cases}$$

where γ_{uv} is user-specified

Fixed correlation structures are not allowed for ordinal multinomial models.

No parameters need to be estimated for this structure.

Notes

- When the scale parameter is updated by the current Pearson residuals, the denominator for the $\boldsymbol{\alpha}$ parameter vector is an estimator of the scale parameter.
- The denominators in the above equations and in the estimator of the scale parameter are all adjusted by the number of non-redundant parameters (not subtracted by p_x). The user can specify that these adjustments not be used so that the numerator and denominator parts are

invariant to subject-level replication changes of the data. If the denominators are non-positive; that is, if the summation part is smaller than or equal to p_X, then only the summation part is used.

Estimation

Having selected a particular model, it is required to estimate the parameters and to assess the precision of the estimates.

Parameter Estimation

The algorithm for estimating model parameters using GEEs is outlined below. Note the scale parameter or the ancillary parameter *k* is not a part of parameter estimation and see below on how to deal with them.

Some definitions are needed for an iterative process:

Table 46-25
Notation

Notation	Description
M	The maximum number of iterations. It must be a non-negative integer. If the value is 0, then initial parameter values become final estimates.
N_u	The number of iterations between updates of the working correlation matrix. It must be a positive integer.
CORRTYPE	The specified working correlation structure.
$\epsilon_p, \epsilon_{\mathbf{H}}$	Tolerance levels for different types of convergence criteria.
Abs	A 0/1 binary variable; *Abs* = 1 if absolute change is used for convergence criteria and *Abs* = 0 if relative change is used.

1. Input initial values $\beta^{(0)}$ and/or $\phi^{(0)}$ or if no initial values are given, compute initial estimates with an independent generalized linear model.

2. Compute the working correlation R(α) based on $\beta^{(0)}$, Pearson residuals and a specified working correlation structure (*CORRTYPE*). Check if R(α) is positive definite for exchangeable, *m*-dependent and unstructured structures. If it is not, revise it to be equal to $\frac{1}{1+\varsigma}(\mathrm{R}(\alpha)+\varsigma\mathrm{I})$, where **I** is an identity matrix and ς is a ridge value such that the adjusted matrix is positive definite. If a fixed correlation matrix is specified by the users and it is not positive definite, issue a warning and stop. Then compute the initial estimate of the covariance matrix of y_i ($\mathrm{V}_i^{(0)}$), the generalized estimating equation $\mathbf{s}^{(0)}$, and generalized Hessian matrix $\mathbf{H}^{(0)}$ (see formulae below) based on $\beta^{(0)}$ and $\mathrm{V}_i^{(0)}$.

3. Initialize *v*=0.

4. Set *v*=*v*+1.

5. Compute estimates of *v*th iteration

$$\beta^{(v)} = \beta^{(v-1)} - \left(\mathbf{H}^{(v-1)}\right)^{-}\mathbf{s}^{(v-1)},$$

6. If v/N_u is a positive integer, update the working correlation, checking for positive definiteness as above.

7. Compute an estimate of the covariance matrix of y_i and its generalized inverse

$$V_i^{(v)} = \phi A_i^{1/2} R(\alpha) A_i^{1/2} \text{ and } \left(V_i^{(v)}\right)^- = \tfrac{1}{\phi} A_i^{-1/2} R(\alpha)^- A_i^{-1/2}.$$

For the ordinal multinomial model, replace $R(\alpha)$ with $R_i(\alpha)$ in the above equations.

8. Revise $\mathbf{s}^{(v)}$ and $\mathbf{H}^{(v)}$ based on $\beta^{(v)}$ and $V_i^{(v)}$.

$$\mathbf{s}^{(v)} = \sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^T \left(V_i^{(v)}\right)^- (y_i - \mu_i),$$

$$\mathbf{H}^{(v)} = -\sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^T \left(V_i^{(v)}\right)^- \left(\tfrac{\partial \mu_i}{\partial \beta}\right).$$

For the ordinal multinomial model,

$$s^{(v)} = \sum_{i=1}^{K} f_i \left(\frac{\partial \pi_i}{\partial B}\right)^T \left(V_i^{(v)}\right)^- (z_i - \pi_i),$$

$$H^{(v)} = -\sum_{i=1}^{K} f_i \left(\frac{\partial \pi_i}{\partial B}\right)^T \left(V_i^{(v)}\right)^- \left(\tfrac{\partial \pi_i}{\partial B}\right).$$

9. Check the convergence criteria. If they are met or the maximum number of iterations is reached, stop. The final vector of estimates is denoted by $\hat{\beta}$ (and $\hat{\Psi}$ for the ordinal multinomial). Otherwise, go back to step (4).

Scale Parameter Handling

If no initial values are given,, the initial values are computed with an independent GZLM. The ways to deal with the scale parameter in the GZLM step (1) and the GEE step (7) are as follows:

- For normal, inverse Gaussian, gamma, and Tweedie response, if the scale parameter is estimated by the ML method in step (1), then in step (7) ϕ would be updated as

$$\hat{\phi} = \tfrac{1}{N' - p_x} \sum_{i=1}^{K} \sum_{t=1}^{n_i} f_i r_{it}^2,$$

  where r_{it} is the Pearson residual, and n_i is the number of non-missing measurements on the *i*th subject.
- If the scale parameter is set to a fixed value in step (1), then ϕ would be held fixed at that value in step (7) as well.

Convergence Criteria

We consider parameter convergence and Hessian convergence. For parameter convergence, we consider both absolute and relative change, but for Hessian convergence, we only consider absolute change because the log-likelihood values used as the denominator for relative change are not valid for GEE. Let ϵ_p and ϵ_H be given tolerance levels for each type, then the criteria can be written as follows:

$$\text{Parameter convergence: } \begin{cases} \max_j \left(\frac{\left|\beta_j^{(v)} - \beta_j^{(v-1)}\right|}{\left|\beta_j^{(v-1)}\right| + 10^{-6}} \right) < \epsilon_p \text{ if relative change} \\ \max_j \left(\left|\beta_j^{(v)} - \beta_j^{(v-1)}\right| \right) < \epsilon_p \text{ if absolute change} \end{cases}$$

$$\text{Hessian convergence: } \left(\mathbf{s}^{(v)}\right)^T \left(\mathbf{H}^{(v)}\right)^- \left(\mathbf{s}^{(v)}\right) < \epsilon_{\mathbf{H}} \text{ if absolute change}$$

If the Hessian convergence criterion is not user-specified, it is checked based on absolute change with ϵ_H = 1E-4 after the log-likelihood or parameter convergence criterion has been satisfied. If Hessian convergence is not met, a warning is displayed.

Parameter Estimate Covariance Matrix, Correlation Matrix and Standard Errors

Parameter Estimate Covariance

Two parameter estimate covariance matrices can be calculated. One is the model-based estimator and the other one is the robust estimator. As in the generalized linear model, the consistency of the model-based parameter estimate covariance depends on the correct specification of the mean and variance of the response (including correct choice of the working correlation matrix). However, the robust parameter estimate covariance is still consistent even when the specification of the working correlation matrix is incorrect as we often expect.

The model-based parameter estimate covariance is

$$\Sigma_m = -\mathbf{H}_1^-$$

where $\mathbf{H}_1^-$ is the generalized inverse of $\mathbf{H}_1 = -\sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^T V_i^- \left(\frac{\partial \mu_i}{\partial \beta}\right)$.

For the ordinal multinomial model, $H_1 = -\sum_{i=1}^{K} f_i \left(\frac{\partial \pi_i}{\partial B}\right) V_i^- \left(\frac{\partial \pi_i}{\partial B}\right)$.

The robust parameter estimate covariance is

$$\Sigma_r = \mathbf{H}_1^- \mathbf{H}_2 \mathbf{H}_1^-$$

where $\mathbf{H}_2 = \sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^T V_i^- \operatorname{cov}(y_i) V_i^- \left(\frac{\partial \mu_i}{\partial \beta}\right)$ and $cov\,(y_i)$ can be estimated by $(y_i - \mu_i)(y_i - \mu_i)^T$.

For the ordinal multinomial model, $H_2 = \sum_{i=1}^{K} f_i \left(\frac{\partial \pi_i}{\partial B}\right) V_i^- cov\,(z_i)\, V_i^- \left(\frac{\partial \pi_i}{\partial B}\right)$ and $cov\,(z_i)$ can be estimated by $(z_i - \pi_i)(z_i - \pi_i)$.

Note that model-based parameter estimate covariance will be affected by how the scale parameter is handled, but the robust parameter estimate covariance will not be affected by the estimate of the scale parameter because ϕ is cancelled in different terms.

Parameter Estimate Correlation

Parameter estimate correlation is calculated as described in GZLM. For more information, see the topic “Parameter Estimate Covariance Matrix, Correlation Matrix and Standard Errors”.

Parameter Estimate Standard Error

Parameter estimate standard errors are calculated as described in GZLM. There is no standard error for the scale parameter in GEE. For more information, see the topic “Parameter Estimate Covariance Matrix, Correlation Matrix and Standard Errors”.

Wald Confidence Intervals

Wald confidence intervals are calculated as described in GZLM. For more information, see the topic “Confidence Intervals”.

Chi-Square Statistics

The chi-square statistics and corresponding *p*-values are calculated as described in GZLM. For more information, see the topic “Chi-Square Statistics”.

Model Testing

Since GEE is not a likelihood-based method of estimation, the inferences based on likelihoods are not possible for GEEs. Most notably, the Lagrange multiplier test, goodness-of-fit tests, and omnibus tests are invalid and will not be offered.

Default tests of model effects are as in GZLM. For more information, see the topic “Default Tests of Model Effects”.

Estimated marginal means are as in GZLM. For more information, see the topic “Estimated Marginal Means”.

Goodness of Fit

None of the goodness-of-fit statistics which are available for GZLM are valid for GEE. However, Pan (2001b) introduced two useful extensions of AIC as goodness-of-fit statistics for model selection based on the quasi-likelihood function. One is for working correlation structure selection and the other is for variable selection. Both of them are based on the quasi-likelihood function under the independence model and the known scale parameter assumptions.

For the ordinal multinomial model, these goodness of fit statistics are not available because the log quasi-likelihood function cannot be derived.

Based on the model specification $E(y) = \mu$ and $\mathrm{Var}(y) = V(\mu)\phi/\omega$, the (log) quasi-likelihood function for each case is

$$Q_k(\mu; \phi, \omega, y) = \int^{\mu} \frac{y - t}{\frac{\phi}{\omega} V(t)} dt = F(\mu)$$

or

$$Q(\mu; \phi, \omega, y) = \int_y^{\mu} \frac{y - t}{\frac{\phi}{\omega} V(t)} dt = F(\mu) - F(y)$$

which we shall call the kernel quasi-likelihood and full quasi-likelihood, respectively.

Since the components of **Y** are independent by assumption, the kernel and full quasi-likelihood for the complete data is the sum of the individual contributions

$$Q_k(\mu; \phi, \omega, y) = \sum_{i=1}^{n} Q_{k,i}(\mu_i; \phi, \omega_i, y_i)$$

and

$$Q(\mu; \phi, \omega, y) = \sum_{i=1}^{n} Q_i(\mu_i; \phi, \omega_i, y_i).$$

Since **μ** would depend on **β**, we change notation from $Q_k(\mu; \phi, \omega, y)$ to $Q_k(\beta; I)$ and $Q(\mu; \phi, \omega, y)$ to $Q(\beta; I)$ where I implies independence assumption. The quasi-likelihood functions for the probability distributions are listed in the following table:

Table 46-26

Quasi-likelihood functions for probability distributions

Distribution	$Q_k(\beta; I)$ **and** $Q(\beta; I)$
Normal	$Q_k(\beta; I) = Q(\beta; I) = \sum_{i=1}^{n} -\frac{f_i\omega_i}{2\phi}\{(y_i - \mu_i)^2\}$
Inverse Gaussian	$Q_k(\beta; I) = \sum_{i=1}^{n} -\frac{f_i\omega_i}{\phi}\left\{\frac{y_i}{2\mu_i^2} - \frac{1}{\mu_i}\right\}$ $Q(\beta; I) = \sum_{i=1}^{n} -\frac{f_i\omega_i}{2\phi y_i \mu_i^2}(y_i - \mu_i)^2$
Gamma	$Q_k(\beta; I) = \sum_{i=1}^{n} -\frac{f_i\omega_i}{\phi}\left\{\frac{y_i}{\mu_i} + \ln(\mu_i)\right\}$ $Q(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\left\{\ln\left(\frac{y_i}{\mu_i}\right) + \frac{\mu_i - y_i}{\mu_i}\right\}$

Distribution	**Q_k (β; I) and$Q$ (β; I)**
Negative binomial	$Q_k(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\{y_i \ln(k\mu_i) - (y_i + 1/k)\ln(1 + k\mu_i)\}$ $Q(\beta; I) = \sum_{i=1}^{n} -\frac{f_i\omega_i}{\phi}\left\{y_i \ln\left(\frac{y_i}{\mu_i}\right) - (y_i + 1/k)\ln\left(\frac{y_i + 1/k}{\mu_i + 1/k}\right)\right\}$
Poisson	$Q_k(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\{y_i \ln(\mu_i) - \mu_i\}$ $Q(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\left\{-y_i \ln\left(\frac{y_i}{\mu_i}\right) + (y_i - \mu_i)\right\}$
Binomial(m)	$Q_k(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i^*}{\phi}\{y_i \ln(\mu_i) + (1 - y_i)\ln(1 - \mu_i)\}$ $Q(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i^*}{\phi}\left\{y_i \ln\left(\frac{\mu_i}{y_i}\right) + (1 - y_i)\ln\left(\frac{1 - \mu_i}{1 - y_i}\right)\right\}$
Tweedie	$Q_k(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\left\{\frac{(2 - q)y_i\mu_i^{1-q} - (1 - q)\mu_i^{2-q}}{(1 - q)(2 - q)}\right\}$ $Q(\beta; I) = \sum_{i=1}^{n} \frac{f_i\omega_i}{\phi}\left\{\frac{y_i^{2-q} + (2 - q)y_i\mu_i^{1-q} - (1 - q)\mu_i^{2-q}}{(1 - q)(2 - q)}\right\}$

Then the quasi-likelihood under the independence model criterion (QIC) for choosing the best correlation structure is defined as

QIC(**R**) = $-2Q(\beta_{\mathbf{R}}; \mathrm{I}) + 2\mathrm{trace}\left(-\mathbf{H}_{1,\mathrm{I}} \ \boldsymbol{\Sigma}_{r,\mathbf{R}}\right)$

There are three terms in the above formula:

1. $Q(\beta_{\mathbf{R}}; \mathrm{I})$ is the value of the quasi-likelihood computed using the parameter estimates from the model with hypothesized correlation structure **R**; that is, the estimates of $\beta_{\mathbf{R}}$. In evaluating the quasi-likelihood, we use $\hat{\mu}$ in place of μThe scale parameter is unknown in practice, so we have to assign a value. If it is set to a fixed value by the user, then that value is used; otherwise 1 is used. Note that $Q(\beta_{\mathbf{R}}; \mathrm{I})$ could be replaced by $Q_k(\beta_{\mathbf{R}}; \mathrm{I})$.

2. $\mathbf{H}_{1,\mathrm{I}}$ is the generalized Hessian matrix obtained by fitting an independent working correlation model.

3. $\boldsymbol{\Sigma}_{r,\mathbf{R}}$ is the robust estimator for parameter estimate covariance from the model with hypothesized correlation structure **R**.

Under the assumption that all modeling specifications in GEE are correct, trace$\left(-\mathbf{H}_{1,\mathrm{I}} \ \boldsymbol{\Sigma}_{r,\mathbf{R}}\right) \approx p_x$, then the above QIC reduces to

QIC$_u$(**R**) = $-2Q(\beta_{\mathbf{R}}; \mathrm{I}) + 2p_x$

So $\mathrm{QIC_u}(\mathbf{R})$ can be useful for choosing the best subset of covariates for a particular model. For the use of QIC and $\mathrm{QIC_u}(\mathbf{R})$, the model with the smallest value is preferred. Note again that $Q(\beta_{\mathbf{R}};\mathrm{I})$ could be replaced by $Q_k(\beta_{\mathbf{R}};\mathrm{I})$.

Default Tests of Model Effects

For type I and III analyses, Wald statistics are still valid.

Generalized Score Statistics

For type I and III analyses, the method of constructing a generalized score statistic is the same, the only difference is the method of constructing **L** matrices. A generalized score statistic can be computed as follows (when the process is applied to the ordinal multinomial model, all formulae should be modified accordingly):

Calculate $\tilde{\beta}_{-i}$ under the constraint $\mathrm{L}_i\beta = 0$ for each effect *i*,

$$\tilde{\beta}_{-i} = \underset{\beta}{\mathrm{arg}}(\beta) = 0 \text{ s.t. } \mathrm{L}_i\beta = 0,$$

where L_i is a type III test matrix for the *i*th effect.

The iterative process to calculate the above optimal $\tilde{\beta}_{-i}$ is a combination of sequential quadratic programming and GEE parameter estimation. This kind of iterative process will be used here and for custom tests, so we will describe the iterative process in a more general form:

$$\tilde{\beta} = \underset{\beta}{\mathrm{arg}}\,\mathbf{s}(\beta) = 0 \text{ s.t. } \mathrm{L}\beta = \mathrm{K}.$$

The iterative process is outlined briefly as follows:

1. Find initial values $\beta^{(0)}$ with $\mathrm{L}\beta^{(0)} = \mathrm{K}$ as described in Section 2.3.4.2-(a).
2. Compute the working correlation R(α) based on the last iteration's estimate $\beta^{(v-1)}$, Pearson residuals and a specified working correlation structure if $(v-1+N_u)/N_u$ is an integer, otherwise working correlation is not updated.
3. Compute an estimate of the covariance matrix of y_i and its generalized inverse

$\mathrm{V}_i^{(v-1)} = \phi \mathrm{A}_i^{1/2}\mathrm{R}(\alpha)\mathrm{A}_i^{1/2}$ and $\left(\mathrm{V}_i^{(v-1)}\right)^{-} = \frac{1}{\phi}\mathrm{A}_i^{-1/2}\mathrm{R}(\alpha)^{-}\mathrm{A}_i^{-1/2}$.

Also compute $\mathbf{s}^{(v-1)}$ and $\mathbf{H}^{(v-1)}$ based on $\beta^{(v-1)}$ and $\mathrm{V}_i^{(v-1)}$ as follows:

$$\mathbf{s}^{(v-1)} = \sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^{\mathrm{T}} \left(\mathrm{V}_i^{(v-1)}\right)^{-}(y_i - \mu_i),$$

$$\mathbf{H}^{(v-1)} = -\sum_{i=1}^{K} f_i \left(\frac{\partial \mu_i}{\partial \beta}\right)^{\mathrm{T}} \left(\mathrm{V}_i^{(v-1)}\right)^{-}\left(\frac{\partial \mu_i}{\partial \beta}\right).$$

4. Find a solution of $\delta^{(v-1)}$ and $\lambda^{(v)}$ for the following KKT equations

$$\begin{bmatrix} \mathbf{H}^{(v-1)} & L^{\mathrm{T}} \\ L & 0 \end{bmatrix} \begin{bmatrix} \delta^{(v-1)} \\ \lambda^{(v)} \end{bmatrix} = \begin{bmatrix} -\mathbf{s}^{(v-1)} \\ 0 \end{bmatrix};$$

5. Compute estimates of the *v*th iteration:

$$\beta^{(v)} = \beta^{(v-1)} + \delta^{(v-1)}$$

6. Check if convergence criteria are met. If they are or the maximum number of iterations is reached, stop. The final vector of estimates is denoted by $\tilde{\beta}$. Otherwise, go back to step (2).

   Note: the convergence criteria here are similar to those for parameter estimation, except that the Hessian convergence criterion is modified as follows:

$$\left(\mathrm{s}^{(v)} + \mathrm{L}^{\mathrm{T}}\lambda^{(v+1)}\right)^{\mathrm{T}} \left(\mathrm{H}^{(v)}\right)^{-} \left(\mathrm{s}^{(v)} + \mathrm{L}^{\mathrm{T}}\lambda^{(v+1)}\right) < \epsilon_{\mathrm{H}}.$$

Compute the generalized estimating equation based on the optimal $\tilde{\beta}_{-i}$.

Calculate the generalized score statistic for each effect *i*,

$$T_{\mathrm{GS},i} = \mathbf{s}\left(\tilde{\beta}_{-i}\right)^{\mathrm{T}} \Sigma_{\mathrm{m}} \mathrm{L}_i^{\mathrm{T}} \left(\mathrm{L}_i \Sigma_{\mathrm{r}} \mathrm{L}_i^{\mathrm{T}}\right)^{-1} \mathrm{L}_i \Sigma_{\mathrm{m}} \mathbf{s}\left(\tilde{\beta}_{-i}\right),$$

where Σ_{m} is the model-based parameter estimate covariance and Σ_{r} is the robust parameter estimate covariance, each evaluated at $\tilde{\beta}_{-i}$. Then the asymptotic distribution of $T_{\mathrm{GS},i}$ is χ^2_r, where *r* is the rank of L_i and the *p*-values can be calculated accordingly.

Wald Statistics

For more information, see the topic "Default Tests of Model Effects". Note Σ_{r} (or Σ_{rm}) should be used as the estimated covariance matrix.

Scoring

Predicted values of the linear predictor, estimated standard errors of predicted values of linear predictor, predicted values of the means and confidence intervals for the means are calculated. For more information, see the topic "Predicted Values".

Only two types of residuals are offered as model diagnostics in GEE: raw residuals and Pearson residuals. For more information, see the topic "Diagnostics".

References

Aitkin, M., D. Anderson, B. Francis, and J. Hinde. 1989. *Statistical Modelling in GLIM*. Oxford: Oxford Science Publications.

Albert, A., and J. A. Anderson. 1984. On the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, 71, 1–10.

Cameron, A. C., and P. K. Trivedi. 1998. *Regression Analysis of Count Data*. Cambridge: Cambridge University Press.

Diggle, P. J., P. Heagerty, K. Y. Liang, and S. L. Zeger. 2002. *The analysis of Longitudinal Data*, 2 ed. Oxford: Oxford University Press.

Dobson, A. J. 2002. *An Introduction to Generalized Linear Models*, 2 ed. Boca Raton, FL: Chapman & Hall/CRC.

Dunn, P. K., and G. K. Smyth. 2005. Series Evaluation of Tweedie Exponential Dispersion Model Densities. *Statistics and Computing*, 15, 267–280.

Dunn, P. K., and G. K. Smyth. 2001. Tweedie Family Densities: Methods of Evaluation. In: *Proceedings of the 16th International Workshop on Statistical Modelling*, Odense, Denmark: .

Gill, J. 2000. *Generalized Linear Models: A Unified Approach*. Thousand Oaks, CA: Sage Publications.

Hardin, J. W., and J. M. Hilbe. 2001. *Generalized Estimating Equations*. Boca Raton, FL: Chapman & Hall/CRC.

Hardin, J. W., and J. M. Hilbe. 2003. *Generalized Linear Models and Extension*. Station, TX: Stata Press.

Horton, N. J., and S. R. Lipsitz. 1999. Review of Software to Fit Generalized Estimating Equation Regression Models. *The American Statistician*, 53, 160–169.

Huber, P. J. 1967. The Behavior of Maximum Likelihood Estimates under Nonstandard Conditions. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, CA: University of California Press, 221–233.

Lane, P. W., and J. A. Nelder. 1982. Analysis of Covariance and Standardization as Instances of Prediction. *Biometrics*, 38, 613–621.

Lawless, J. E. 1984. Negative Binomial and Mixed Poisson Regression. *The Canadian Journal of Statistics*, 15, 209–225.

Liang, K. Y., and S. L. Zeger. 1986. Longitudinal Data Analysis Using Generalized Linear Models. *Biometrika*, 73, 13–22.

Lipsitz, S. H., K. Kim, and L. Zhao. 1994. Analysis of Repeated Categorical Data Using Generalized Estimating Equations. *Statistics in Medicine*, 13, 1149–1163.

McCullagh, P. 1983. Quasi-Likelihood Functions. *Annals of Statistics*, 11, 59–67.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

Miller, M. E., C. S. Davis, and J. R. Landis. 1993. The Analysis of Longitudinal Polytomous Data: Generalized Estimating Equations and Connections with Weighted Least Squares. *Biometrics*, 49, 1033–1044.

Nelder, J. A., and R. W. M. Wedderburn. 1972. Generalized Linear Models. *Journal of the Royal Statistical Society Series A*, 135, 370–384.

Pan, W. 2001. Akaike's Information Criterion in Generalized Estimating Equations. *Biometrics*, 57, 120–125.

Pregibon, D. 1981. Logistic Regression Diagnostics. *Annals of Statistics*, 9, 705–724.

Smyth, G. K., and B. Jorgensen. 2002. Fitting Tweedie's Compound Poisson Model to Insurance Claims Data: Dispersion Modelling. *ASTIN Bulletin*, 32, 143–157.

White, H. 1980. A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity. *Econometrica*, 48, 817–836.

Williams, D. A. 1987. Generalized Linear Models Diagnostics Using the Deviance and Single Case Deletions. *Applied Statistics*, 36, 181–191.

Zeger, S. L., and K. Y. Liang. 1986. Longitudinal Data Analysis for Discrete and Continuous Outcomes. *Biometrics*, 42, 121–130.

GENLOG Multinomial Loglinear and Logit Models Algorithms

This chapter describes the algorithms used to calculate maximum-likelihood estimates for the multinomial loglinear model and the multinomial logit model. This algorithm is applicable only to aggregated data.

Notation

The following notation is used throughout this chapter unless otherwise stated:

A	Generic categorical independent (explanatory) variable. Its categories are indexed by an array of integers.
B	Generic categorical dependent (response) variable. Its categories are indexed by an array of integers.
r	Number of categories of *B*. $r \geq 1$
c	Number of categories of *A*. $c \geq 1$
p	Number of nonredundant (nonaliased) parameters.
i	Generic index for the categories of *B*. $i=1,...,r$
j	Generic index for the categories of *A*. $j=1,...,c$
k	Generic index for the parameters. $k=1,...,p$
n_{ij}	Observed count in the *i*th response of *B* and the *j*th setting of *A*. $n_{ij} \geq 0$
N_j	Marginal total count at the jth setting of A. $N_j = \Sigma_{i=1}^{r} n_{ij}$
N	Total observed count. $N = \Sigma_{j=1}^{c} \Sigma_{i=1}^{r} n_{ij}$
m_{ij}	Expected count. $m_{ij} \geq 0$
π_{ij}	Probability of having an observation in the *i*th response of *B* and the *j*th setting of *A*. $0 \leq \pi_{ij} \leq 1$ and $\Sigma_{j=1}^{c} \Sigma_{i=1}^{r} \pi_{ij} = 1$
z_{ij}	Cell structure value.
α_j	*j*th normalizing constant.
β_k	*k*th nonredundant parameter.
β	A vector of $(\beta_1, \ldots, \beta_p)'$.
x_{ijk}	An element in the *i*th row and the *k*th column of the design matrix for the *j* setting.

The same notation is used for both loglinear and logit models so that the methods are presented in a unified way. Conceptually, one can consider a loglinear model as a special case of a logit model where the explanatory variable has only one level (that is, *c*=1).

Model

There are two components in a loglinear model: the random component and the systematic component.

Random Component

The random component describes the joint distribution of the counts.

- The counts $\{n_{1j},\ldots,n_{rj}\}$ at the *j*th setting of *A* have the multinomial $(N_j,\pi_{1j},\ldots,\pi_{rj})$ distribution.
- The counts n_{ij} and $n_{i'j'}$ are independent if $j\neq j'$.
- The joint probability distribution of $\{n_{ij}\}$ is the product of these *c* independent multinomial distributions. The probability density function is

$$\prod_{j=1}^{c}\left(\frac{N_j!}{\prod_{i=1}^{r}n_{ij}!}\prod_{i=1}^{r}\pi_{ij}\right)$$

- The expected count is $\mathrm{E}(n_{ij})=m_{ij}=N_j\pi_{ij}$.
- The covariance is

$$\mathrm{cov}\left(n_{ij},n_{i'j'}\right)=\begin{cases}N_j\pi_{ij}\left(\delta_{ii'}-\pi_{i'j}\right) & \text{if } j=j'\\ 0 & \text{if } j\neq j'\end{cases}$$

where $\delta_{ab}=1$ if $a=b$ and $\delta_{ab}=0$ if $a\neq b$.

Systematic Component

The systematic component describes the linkage function between the expected counts and the parameters. The expected counts are themselves functions of other parameters. Explicitly, for *i*=1...,*r* and *j*=1,...,*c*,

$$m_{ij}=\begin{cases}z_{ij}e^{\alpha_j+v_{ij}} & \text{if } z_{ij}>0\\ 0 & \text{if } z_{ij}\leq 0\end{cases}$$

where

$$v_{ij}=\sum_{k=1}^{p}x_{ijk}\beta_k$$

Normalizing Constants

α_j are not considered as parameters, but as normalizing constants.

$$\alpha_j=\log\left(\frac{N_j}{\sum_{i=1}^{r}z_{ij}e^{v_{ij}}}\right)j=1,\ldots,c$$

Cell Structure Values

The cell structure values play two roles in loglinear procedures, depending on their signs. If $z_{ij}>0$, it is a usual weight for the corresponding cell and log (z_{ij}) is sometimes called the **offset**. If $z_{ij}\leq 0$, a **structural zero** is imposed on the cell $(B=i,A=j)$. Contingency tables containing at least one structural zero are called incomplete tables. If $n_{ij}=0$ but $z_{ij}>0$, the cell

$(B=i, A=j)$ contains a **sampling zero**. Although a structural zero is still considered part of the contingency table, it is not used in fitting the model. Cellwise statistics are not computed for structural zeros.

Maximum-Likelihood Estimation

The multinomial log-likelihood is

$$L(\beta)=L(\beta_1,\ldots,\beta_p)=\text{constant}+\sum_{j=1}^{c}\sum_{i=1}^{r}n_{ij}\log(m_{ij})$$

Likelihood Equations

It can be shown that:

$$\frac{\partial L}{\partial\beta_k}=\sum_{j=1}^{c}\sum_{i=1}^{r}(n_{ij}-m_{ij})x_{ijk} \text{ for } k=1,\ldots,p$$

Let $\mathbf{g}(\beta)=(g_1(\beta),\ldots,g_p(\beta))'$ be the (p+1) gradient vector with

$$g_k(\beta)=\frac{\partial L}{\partial\beta_k}$$

The maximum-likelihood estimates $\hat\beta=\left(\hat\beta_1,\ldots,\hat\beta_p\right)^t$ are regarded as a solution to the vector of likelihood equations:

$$\mathbf{g}(\beta)=0$$

Hessian Matrix

The likelihood equations are nonlinear functions of β. Solving them for $\hat\beta$ requires an iterative method. The Newton-Raphson method is used. It can be shown that

$$\frac{\partial^2 L}{\partial\beta_k\partial\beta_t}=-\sum_{j=1}^{c}\sum_{i=1}^{r}m_{ij}(x_{ijk}-\theta_{jk})(x_{ijl}-\theta_{jl})$$

where

$$\theta_{jk}=\frac{1}{N_j}\sum_{i=1}^{r}m_{ij}x_{ijk} \quad j=1,\ldots,c \text{ and } k=1,\ldots,p$$

Let $\mathbf{H}(\beta)$ be the p×p information matrix, where $-\mathbf{H}(\beta)$ is the Hessian matrix of the log-likelihood. The elements of $\mathbf{H}(\beta)$ are

$$h_{kl}(\beta)=-\frac{\partial^2 L}{\partial\beta_k\partial\beta_l} \quad k=1,\ldots,p \text{ and } l=1,\ldots,p$$

Note:$\mathbf{H}(\beta)$ is a symmetric positive-definite matrix. The asymptotic covariance matrix of β is estimated by $\mathbf{H}^{-1}(\beta)$.

Newton-Raphson Method

Let $\beta^{(s)}$ denote the *s*th approximation for the solution. By the Newton-Raphson method,

$$\beta^{(s+1)} = \beta^{(s)} + \mathbf{H}^{-1}\big(\beta^{(s)}\big)\mathbf{g}\big(\beta^{(s)}\big)$$

Define $\mathbf{q}(\beta) = \mathbf{H}(\beta)\beta + \mathbf{g}(\beta)$. The *k*th element of $\mathbf{q}(\beta)$ is

$$q_k(\beta) = \sum_{j=1}^{c}\sum_{i=1}^{r}\eta_{ij}(x_{ijk} - \theta_{jk})$$

where

$$\eta_{ij} = \begin{cases} m_{ij}v_{ij} + \underset{\eta_{ij}}{(}n_{ij} - m_{ij}) & \text{if } z_{ij} > 0 \text{ and } m_{ij} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Then

$$\mathbf{H}\big(\beta^{(s)}\big)\beta^{(s+1)} = \mathbf{q}\big(\beta^{(s)}\big)$$

Thus, given $\beta^{(s)}$, the (*s*+1)th approximation $\beta^{(s+1)}$ is found by solving this systemof equations.

Initial Values

$\beta^{(0)}$, which corresponds to a saturated mode, is used as the initial value for β. Then the initial estimates for the expected cell counts are

$$m_{ij}^{(0)} = \begin{cases} n_{ij} + \Delta & \text{if } z_{ij} > 0 \\ 0 & \text{if } z_{ij} \le 0 \end{cases}$$

where $\Delta \ge 0$ is a constant.

Note: For saturated models, Δ is added to n_{ij} if $z_{ij} > 0$. This is done to avoid numerical problems in case some observed counts are 0. We advise users to set Δ to 0 whenever all observed counts (other than structural zeros) are positive.

The initial values for other quantities are

$$\theta_{jk}^{(0)} = \tfrac{1}{N_j}\sum_{i=1}^{r} m_{ij}^{(0)} x_{ijk}$$

and

$$\eta_{ij}^{(0)} = \begin{cases} m_{ij}^{(0)} \log\left(m_{ij}^{(0)}/z_{ij}\right) + \left(n_{ij} - m_{ij}^{(0)}\right) & \text{if } z_{ij} > 0 \text{ and } m_{ij}^{(0)} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Stopping Criteria

The following conditions are checked for convergence:

1. $\max_{i,j}\left(\left|m_{ij}^{(s+1)} - m_{ij}^{(s)}\right|/m_{ij}^{(s)}\right) < \epsilon$ provided that $m_{ij} > 0$
2. $\max_{i,j}\left(\left|m_{ij}^{(s+1)} - m_{ij}^{(s)}\right|\right) < \epsilon$
3. $\sqrt{\left(\Sigma_{k=1}^{p} g_k^2\left(\hat{\beta}\right)\right)/p} < \epsilon$

The iteration is said to be converged if either conditions 1 and 3 or conditions 2 and 3 are satisfied. If *p*=0, then condition 3 will be automatically satisfied. The iteration is said to be not converged if neither pair of conditions is satisfied within the maximum number of iterations.

Algorithm

The iteration process uses the following steps:

1. Calculate $m_{ij}^{(0)}$, $\theta_{jk}^{(0)}$, and $n_{ij}^{(0)}$.
2. Set *s*=0.
3. Calculate $\mathbf{H}\left(\beta^{(s)}\right)$ evaluated at $m_{ij} = m_{ij}^{(s)}$; calculate $\mathbf{q}\left(\beta^{(s)}\right)$ evaluated at $n_{ij} = n_{ij}^{(s)}$.
4. Solve for $\beta^{(s+1)}$.
5. Calculate $v_{ij}^{(s+1)} = \Sigma_{k=1}^{p} x_{ijk}\beta_k^{(s+1)}$ and

$$m_{ij}^{(s+1)} = \begin{cases} N_j\left(z_{ij}e^{v_{ij}^{(s+1)}}/\left(\Sigma_{t=1}^{r} z_{tj}e^{v_{ij}^{(s+1)}}\right)\right) & \text{if } z_{ij} > 0 \\ 0 & \text{if } z_{ij} \le 0 \end{cases}$$

6. Check whether the stopping criteria are satisfied. If yes, stop iteration and declare convergence. Otherwise continue.
7. Increase s by 1 and check whether the maximum iteration has been reached. If yes, stop iteration and declare the process not converged. Otherwise repeat steps 3-7.

Estimated Normalizing Constants

The maximum-likelihood estimate for α_j is

$$\hat{\alpha}_j = \log\left(\frac{N_j}{\Sigma_{i=1}^{r} z_{ij}e^{\hat{v}_{ij}}}\right) j = 1, \ldots, c$$

where

$$\hat{v}_{ij} = \sum_{k=1}^{p} x_{ijk}\hat{\beta}_k$$

Estimated Cell Counts

The estimated expected count is

$$\hat{m}_{ij} = \begin{cases} N_j\left(z_{ij}e^{\hat{v}_{ij}}/\left(\Sigma_{t=1}^{r} z_{tj}e^{\hat{v}_{tj}}\right)\right) & \text{if } z_{ij} > 0 \\ 0 & \text{if } z_{ij} \le 0 \end{cases}$$

Goodness-of-Fit Statistics

The Pearson chi-square statistic is

$$X^2 = \sum_{j=1}^{c}\sum_{i=1}^{r} X_{ij}^2$$

where

$$X_{ij}^2 = \begin{cases} (n_{ij} - \hat{m}_{ij})^2/\hat{m}_{ij} & \text{if } z_{ij} > 0, n_{ij} > 0, \text{ and } \hat{m}_{ij} > 0 \\ \text{SYSMIS} & \text{if } z_{ij} > 0, n_{ij} > 0, \text{ and } \hat{m}_{ij} = 0 \\ 0 & \text{if } z_{ij} \le 0 \text{ or } n_{ij} = \hat{m}_{ij} \end{cases}$$

If any X_{ij}^2 is system missing, then X^2 is also system missing.

The likelihood-ratio chi-square statistic is

$$G^2 = 2\sum_{j=1}^{c}\sum_{i=1}^{r} X_{ij}^2$$

where

$$G_{ij}^2 = \begin{cases} n_{ij}(\log(n_{ij}/\hat{m}_{ij})) & \text{if } z_{ij} > 0, n_{ij} > 0 \text{ and } \hat{m}_{ij} > 0 \\ \text{SYSMIS} & \text{if } z_{ij} > 0, n_{ij} > 0 \text{ and } \hat{m}_{ij} = 0 \\ 0 & \text{if } z_{ij} > 0, n_{ij} = 0, \text{ and } \hat{m}_{ij} \ge 0; \\ & z_{ij} \le 0 \text{ or } n_{ij} = \hat{m}_{ij} \end{cases}$$

If any G_{ij}^2 is system missing, then G^2 is also system missing.

Degrees of Freedom

The degrees of freedom for each statistic is defined as $a = c(r-1) - p - E$, where E is the number of cells with $z_{ij} \le 0$ or $\hat{m}_{ij} = 0$.

Significance Level

The significance level (or the p value) for the Pearson chi-square statistic is Prob$(x_a^2 > X^2)$ and that for the likelihood-ratio chi-square statistic is Prob$(x_a^2 > G^2)$. In both cases, x_a^2 is the central chi-square distribution with a degrees of freedom.

Analysis of Dispersion (Logit Models Only)

The analysis of dispersion is based on two types of dispersion: entropy and concentration. The following definitions are used:

S(A)	Dispersion due to the model
S(B|A)	Dispersion due to residuals
S(B)	Total dispersion
R=S(A)/S(B)	Measure of association

where $S(A) + S(B|A) = S(B)$. Also define

$$\hat{\pi}_i = \frac{\Sigma_{j=1}^{c}\hat{m}_{ij}}{\Sigma_{j=1}^{c}N_j}$$

$$\hat{\pi}_{i|j} = \frac{\hat{m}_{ij}}{N_j}$$

The bounds are $0 \le \hat{\pi}_i \le 1$and $0 \le \hat{\pi}_{ij} \le 1$.

Entropy

$$S(B) = -N\sum_{i=1}^{r} S_i(B)$$

where

$$S_i(B) = \begin{cases} \hat{\pi}_i \log(\pi_i) & \text{if } 0 < \hat{\pi}_i \le 1 \\ 0 & \text{if } \hat{\pi}_i = 0 \end{cases}$$

and

$$S(B|A) = -\sum_{j=1}^{c} N_j \sum_{i=1}^{r} S_{ij}(B|A)$$

where

$$S_{ij}(B|A) = \begin{cases} \hat{\pi}_{i|j} \log\left(\hat{\pi}_{i|j}\right) & \text{if } 0 < \hat{\pi}_{i|j} \le 1 \\ 0 & \text{if } \hat{\pi}_{i|j} = 0 \end{cases}$$

Concentration

$$S(B) = N\left(1 - \sum_{i=1}^{r} \hat{\pi}_i^2\right)$$

$$S(B|A) = \sum_{j=1}^{c} N_j \left(1 - \sum_{i=1}^{r} \hat{\pi}_{i|j}^2\right)$$

Degrees of Freedom

Source	**Measure**	**Degrees of Freedom**
Model	*S(A)*	$f(r-1)$
Residual	*S(B|A)*	$(N-f-1)(r-1)$
Total	*S(B)*	$(N-1)(r-1)$

where *f* equals *p* minus the number of nonredundant columns (in the design matrix) associated with the main effects of the dependent factors.

Residuals

Goodness-of-fit statistics provide only broad summaries of how models fit data. The pattern of lack of fit is revealed in cell-by-cell comparisons of observed and fitted cell counts.

Simple Residuals

The simple residual of the (*i*,*j*)th cell is

$$r_{ij} = \begin{cases} n_{ij} - \hat{m}_{ij} & \text{if } z_{ij} > 0 \\ \text{SYSMIS} & \text{if } z_{ij} \le 0 \end{cases}$$

Standardized Residuals

The standardized residual for the (*i*,*j*)th cell is

$$r_{ij}^S = \begin{cases} (n_{ij} - \hat{m}_{ij})/\sqrt{\hat{m}_{ij}(1-\hat{m}_{ij}/N_j)} & \text{if } z_{ij} > 0 \text{ and } 0 < \hat{m}_{ij} < N_j \\ 0 & \text{if } z_{ij} > 0 \text{ and } n_{ij} = \hat{m}_{ij} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

The standardized residuals are also known as Pearson residuals even though $\Sigma_{j=1}^{c}\Sigma_{i=1}^{r}\left(r_{ij}^S\right)^2 \ne X^2$. Although the standardized residuals are asymptotically normal, their asymptotic variances are less than 1.

Adjusted Residuals

The adjusted residual is the simple residual divided by its estimated standard error. Its definition and applications first appeared in Haberman (1973) and re-appeared on page 454 of Haberman (1979). This statistic for the (*i*,*j*)th cell is

$$r_{ij}^A = \begin{cases} (n_{ij} - \hat{m}_{ij})/\sqrt{s_{ij}} & \text{if } z_{ij} > 0 \text{ and } \hat{m}_{ij} > 0 \\ 0 & \text{if } z_{ij} > 0 \text{ and } n_{ij} = \hat{m}_{ij} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where

$$s_{ij} = \hat{m}_{ij}\left(1 - \frac{\hat{m}_{ij}}{N_j} - \hat{m}_{ij}\sum_{k=1}^{p}\sum_{l=1}^{p}\left(x_{ijk} - \hat{\theta}_{jk}\right)\left(x_{ijl} - \hat{\theta}_{jl}\right)h^{kl}\right)$$

h^{kl} is the (*k*,*l*)th element of $\mathbf{H}^{-1}\left(\hat{\beta}\right)$. The adjusted residuals are asymptotically standard normal.

Deviance Residuals

Pierce and Schafer (1986) and McCullagh and Nelder (1989) define the signed square root of the individual contribution to the G^2 statistic as the deviance residual. This statistic for the (*i*,*j*)th cell is

$$r_{ij}^{D} = \text{sign}(n_{ij} - \hat{m}_{ij})\sqrt{d_{ij}}$$

where

$$d_{ij} = \begin{cases} 2(n_{ij}(\log(n_{ij}/\hat{m}_{ij})) - (n_{ij} - \hat{m}_{ij})) & \text{if } z_{ij} > 0, \hat{m}_{ij} > 0, \text{ and } n_{ij} > 0 \\ 2\hat{m}_{ij} & \text{if } z_{ij} > 0, \hat{m}_{ij} \geq 0, \text{ and } n_{ij} = 0 \\ 0 & \text{if } z_{ij} > 0 \text{ and } n_{ij} = \hat{m}_{ij} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

For multinomial sampling, the individual contribution to the G^2 statistic is only $2n_{ij}\log(n_{ij}/\hat{m}_{ij})$, but this is negative when $n_{ij} < \hat{m}_{ij}$. Thus, an extra term $2(n_{ij} - \hat{m}_{ij})$ is added to it so that $d_{ij} > 0$ for all *i* and *j*. However, we still have $\Sigma_{j=1}^{c}\Sigma_{i=1}^{r}\left(r_{ij}^{D}\right)^2 = G^2$.

Generalized Residual

Consider a linear combination of the cell counts $\Sigma_{j=1}^{c}\Sigma_{i=1}^{r}d_{ij}n_{ij}$, where d_{ij} are real numbers.

The estimated expected value is

$$\sum_{j=1}^{c}\sum_{i=1}^{r}d_{ij}\hat{m}_{ij}$$

The simple residual for this linear combination is

$$\sum_{j=1}^{c}\sum_{i=1}^{r}d_{ij}(n_{ij} - \hat{m}_{ij})$$

The standardized residual for this linear combination is

$$\frac{\Sigma_{j=1}^{c}\Sigma_{i=1}^{r}d_{ij}(n_{ij} - \hat{m}_{ij})}{\sqrt{\Sigma_{j=1}^{c}\left(\Sigma_{i=1}^{r}d_{ij}^{2}\hat{m}_{ij} - \left(\Sigma_{i=1}^{r}d_{ij}m_{ij}\right)^2/N_j\right)}}$$

The adjusted residual for this linear combination is, as given on page 420 of Haberman (1979),

$$\frac{\Sigma_{j=1}^{c}\Sigma_{i=1}^{r}d_{ij}(n_{ij} - \hat{m}_{ij})}{\sqrt{V}}$$

where

$$V = \sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}^2 \hat{m}_{ij} - \sum_{j=1}^{c} \frac{1}{N_j}\left(\sum_{i=1}^{r} d_{ij}\hat{m}_{ij}\right)^2 - \sum_{k=1}^{p}\sum_{l=1}^{p} f_k f_l h^{kl}$$

$$f_k = \sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\hat{m}_{ij}(x_{ijk} - \theta_{ik})$$

Generalized Log-Odds Ratio

Consider a linear combination of the natural logarithm of cell counts

$$\sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(m_{ij})$$

where d_{ij} are real numbers with the restriction

$$\sum_{i=1}^{r} d_{ij} = 0 \quad j = 1,\ldots,c$$

The linear combination is estimated by

$$\sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(\hat{m}_{ij}) = \sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(z_{ij}) + \sum_{j=1}^{c}\sum_{i=1}^{r}\sum_{k=1}^{p} d_{ij}x_{ijk}\hat{\beta}_k$$

The variance of the estimate is

$$\text{var}\left(\sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(\hat{m}_{ij})\right) = \sum_{k=1}^{p}\sum_{l=1}^{p} w_k w_l h^{kl}$$

where

$$w_k = \sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}x_{ijk} \quad k = 1,\ldots,p$$

Wald Statistic

The null hypothesis is

$$H_0 : \sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(m_{ij}) = 0$$

The Wald statistic is

$$W = \frac{\left(\Sigma_{j=1}^{c}\Sigma_{i=1}^{r} d_{ij}\log(\hat{m}_{ij})\right)^2}{\Sigma_{k=1}^{p}\Sigma_{l=1}^{p} w_k w_l h^{kl}}$$

Under H_0, *W* asymptotically distributes as a chi-square distribution with 1 degree of freedom. The significance level is Prob$\left(\chi_1^2 \geq W\right)$. *Note: W* will be system missing if the variance of the estimate is 0.

Asymptotic Confidence Interval

The asymptotic $(1-\alpha)\times 100\%$confidence interval is

$$\sum_{j=1}^{c}\sum_{i=1}^{r} d_{ij}\log(\hat{m}_{ij}) \pm z_{\alpha/2}\sqrt{\sum_{k=1}^{p}\sum_{l=1}^{p} w_k w_l h^{kl}}$$

where $z_{\alpha/2}$ is the upper $\alpha/2$ point of the standard normal distribution. The default value of α is 0.05.

Aggregated Data

This section shows how data are aggregated for a multinomial distribution. The following notation is used in this section:

v_{ij}	Number of cases for $B=i$ $(i=1,\ldots,r)$ and $A=j$ $(j=1,\ldots,c)$
n_{ijs}	*s*th caseweight for $B=i$ and $A=j(s=1,\ldots,v_i)$
x_{ijs}	Covariate
z_{ijs}	Cell weight
c_{ijs}	GRESID coefficient
e_{ijs}	GLOR coefficient
v_{ij}^{+}	Number of positive z_{ijs} (cell weights) for $1\le s\le v_{ij}$

The cell count is

$$n_{ij}=\begin{cases}\Sigma^{*}_{1\le s\le v_{ij}} n^{+}_{ijs} & \text{if } v^{+}_{ij}>0\\ 0 & \text{if } v_{ij}=0 \text{ or } v^{+}_{ij}=0\end{cases}$$

where

$$n^{+}_{ijs}=\begin{cases}n_{ijs} & \text{if } n_{ijs}>0 \text{ and } z_{ijs}>0\\ 0 & \text{if } n_{ijs}\le 0 \text{ and } z_{ijs}>0\end{cases}$$

and $\Sigma^{*}_{1\le s\le v_{ij}}$ means summation over the range of *s* with the terms $z_{ijs}>0$.

The cell weight value is

$$z_{ij}=\begin{cases}\Sigma^{*}_{1\le s\le v_{ij}} n^{+}_{ijs}z_{ijs}/n_{ij} & \text{if } n_{ij}>0 \text{ and } v^{+}_{ij}>0\\ \Sigma^{*}_{1\le s\le v_{ij}} z_{ijs}/v^{+}_{ij} & \text{if } n_{ij}=0 \text{ and } v^{+}_{ij}>0\\ 0 & \text{if } v^{+}_{ij}=0\\ 1 & \text{if} v_{ij}=0\end{cases}$$

If no variable is specified as the cell weight variable, then all cases have unit cell weights by default.

The cell covariate value is

$$x_{ij} = \begin{cases} \Sigma^*_{1\le s\le v_{ij}} n^+_{ijs} x_{ijs}/n_{ij} & \text{if } n_{ij} > 0 \text{ and } v_{ij} > 0 \\ \Sigma^*_{1\le s\le v_{ij}} x_{ijs}/v^+_{ij} & \text{if } n_{ij} = 0 \text{ and } v^+_{ij} > 0 \\ 0 & \text{if } v^+_{ij} = 0 \text{ or } v_{ij} = 0 \end{cases}$$

The cell GRESID coefficient is

$$c_{ij} = \begin{cases} \Sigma^*_{1\le s\le v_{ij}} n^+_{ijs} c_{ijs}/n_{ij} & \text{if } n_{ij} > 0 \text{ and } v_{ij} > 0 \\ \Sigma^*_{1\le s\le v_{ij}} c_{ijs}/v^+_{ij} & \text{if } n_{ij} = 0 \text{ and } v^+_{ij} > 0 \\ 0 & \text{if } v^+_{ij} \text{ or } v_{ij} = 0 \end{cases}$$

There are no defaults for the GRESID coefficients.

The cell GLOR coefficient is

$$e_{ij} = \begin{cases} \Sigma^*_{1\le s\le v_{ij}} n^+_{ijs} e_{ijs}/n_{ij} & \text{if } n_{ij} > 0 \text{ and } v_{ij} > 0 \\ \Sigma^*_{1\le s\le v_{ij}} e_{ijs}/v^+_{ij} & \text{if } n_{ij} = 0 \text{ and } v^+_{ij} > 0 \\ 0 & \text{if } v^+_{ij} = 0 \text{ or } v_{ij} = 0 \end{cases}$$

There are no defaults for the GLOR coefficients.

References

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Christensen, R. 1990. *Log-linear models*. New York: Springer-Verlag.

Haberman, S. J. 1973. The analysis of residuals in cross-classified tables. *Biometrics*, 29, 205–220.

Haberman, S. J. 1978. *Analysis of qualitative data*. London: Academic Press.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

Pierce, D. A., and D. W. Schafer. 1986. Residuals in generalized linear models. *Journal of the American Statistical Association*, 81, 977–986.

GENLOG Poisson Loglinear Model Algorithms

This chapter describes the algorithm to calculate maximum-likelihood estimates for the Poisson loglinear model. This algorithm is applicable only to aggregated data. See "Aggregated Data" for producing aggregated data.

Notation

The following notation is used throughout this chapter unless otherwise stated:

B	Generic categorical dependent (response) variable. Its categories are indexed by an array of integers.
r	Number of categories of *B*. $r \geq 1$
p	Number of nonredundant (nonaliased) parameters.
i	Generic index for the category of *B*. $i=1,...,r$
k	Generic index for the parameters. $k=0,...,p$
n_i	Observed count in the ith response of *B*. $n_{ij} \geq 0$
N	Total observed count, equal to $\sum_{i=1}^{r} n_i$. $N>0$
m_i	Expected count. $m_i > 0$
z_i	Cell structure value.
β_k	The *k*th nonredundant parameter.
β	Vector of $(\beta_0, \beta_{1,}, \ldots, \beta_p)'$
x_{ik}	An element in the *i*th row and the *k*th column of the design matrix.

- Because of the Poisson distribution assumptions, the logit model is not applicable for a Poisson distribution.
- The Poisson distribution is available in GENLOG only.

Model

There are two components in a loglinear model: the random component and the systematic component.

Random Component

The random component describes the joint distribution of the counts.

- The count $\{n_i\}$ has a Poisson distribution with parameter m_i.
- The counts n_i and n'_i are independent if $i \neq i'$.
- The joint probability distribution of $\{n_i\}$ is the product of these *r* independent Poisson distributions. The probability density function is

$$\prod_{i=1}^{r} \frac{m_i^{n_i} e^{-m_i}}{n_i!}$$

- The expected count is $\mathrm{E}(n_i) = m_i$.
- The covariance is $\mathrm{cov}\left(n_i, n'_i\right) = \begin{cases} m_i & \text{if } i = i' \\ 0 & \text{if } i \neq i' \end{cases}$

Systematic Component

The systematic component describes the linkage function between the expected counts and the parameters. The expected counts are themselves functions of parameters. For $i = 1, \ldots, r$,

$$m_i = \begin{cases} z_i e^{\beta_0 + v_i} & \text{if } z_i > 0 \\ 0 & \text{if } z_i \leq 0 \end{cases}$$

where

$$v_i = \sum_{k=1}^{p} x_{ik} \beta_k$$

Since there are no constraints on the observed counts, β_0 is a free parameter in a Poisson loglinear model.

Cell Structure Values

Cell structure values play two roles in loglinear procedures, depending on their signs. If $z_i > 0$, it is a usual weight for the corresponding cell and $\log(z_i)$ is sometimes called the **offset**. If $z_i \leq 0$, a **structural zero** is imposed on the cell (*B*=*i*). Contingency tables containing at least one structural zero are called **incomplete tables**. If $n_i = 0$ but $z_i > 0$, the cell (*B*=*i*) contains a **sampling zero**. Although a structural zero is still considered part of the contingency table, it is not used in fitting the model. Cellwise statistics are not computed for structural zeros.

Maximum-Likelihood Estimation

The multinomial log-likelihood is

$$L(\beta) = L(\beta_0, \ldots, \beta_p) = \text{constant} + \sum_{i=1}^{r} \left(n_i \log(m_i) - m_i\right)$$

Likelihood Equations

It can be shown that

$$\frac{\partial L}{\partial \beta_0} = \sum_{i=1}^{r} (n_i - m_i)$$

$$\frac{\partial L}{\partial \beta_k} = \sum_{i=1}^{r} (n_i - m_i) x_{ik} \qquad k = 1, \ldots, p$$

Let $\mathbf{g}(\beta) = (g_0(\beta), \ldots, g_p(\beta))'$ be the $(p+1) \times 1$ gradient vector with

$$g_k(\beta) = \frac{\partial L}{\partial \beta_k}$$

The maximum-likelihood estimates $\hat{\beta} = \left(\hat{\beta}_0, \ldots, \hat{\beta}_p\right)'$ are regarded as a solution to the vector of likelihood equations:

$$\mathbf{g}(\beta) = 0$$

Hessian Matrix

The likelihood equations are nonlinear functions of β. Solving them for $\hat{\beta}$ requires an iterative method. The Newton-Raphson method is used. It can be shown that

$$\frac{\partial^2 L}{\partial^2 \beta_0} = -\sum_{i=1}^{r} m_i$$
$$\frac{\partial^2 L}{\partial \beta_0 \partial \beta_1} = -\sum_{i=1}^{r} m_i x_{il}$$
$$\frac{\partial^2 L}{\partial \beta_k \partial \beta_0} = -\sum_{i=1}^{r} m_i x_{ik}$$
$$\frac{\partial^2 L}{\partial \beta_k \partial \beta_l} = -\sum_{i=1}^{r} m_i x_{ik} x_{il}$$

Let $\mathbf{H}(\beta)$ be the $(p+1) \times (p+1)$ information matrix, where $-\mathbf{H}(\beta)$ is the Hessian matrix of the log-likelihood. The elements of $\mathbf{H}(\beta)$ are

$$h_{kl}(\beta) = \frac{\partial^2 L}{\partial \beta_k \partial \beta_l} \qquad k = 0, \ldots, p \text{ and } l = 1, \ldots, p$$

Note: $\mathbf{H}(\beta)$ is a symmetric positive definite matrix. The asymptotic covariance matrix of $\hat{\beta}$ is estimated by $-\mathbf{H}(\beta)$.

Newton-Raphson Method

Let $\beta^{(s)}$ denote the *s*th approximation for the solution to the vector of likelihood equations. By the Newton-Raphson method,

$$\beta^{(s+1)} = \beta^{(s)} + \mathbf{H}^{-1}\left(\beta^{(s)}\right)\mathbf{g}\left(\beta^{(s)}\right)$$

Define $\mathbf{q}(\beta) = \mathbf{H}(\beta)\beta + \mathbf{g}(\beta)$. The *k*th element of $\mathbf{q}(\beta)$ is

$$q_k(\beta) = \sum_{i=1}^{r} \eta_i x_{ik}$$

where

$$\eta_i = \begin{cases} m_i v_i + (n_i - m_i) & \text{if } z_i > 0 \text{ and } m_i > 0 \\ 0 & \text{otherwise} \end{cases}$$

Then

$$\mathbf{H}(\beta^{(s)})\beta^{(s+1)} = \mathbf{q}(\beta^{(s)})$$

Thus, given $\beta^{(s)}$, the (*s*+1)th approximation $\beta^{(s+1)}$ is found by solving this system of equations.

Initial Values

$\beta^{(0)}$, which corresponds to a saturated model, is used as the initial value for β. Then the initial estimates for the expected cell counts are

$$m_i^{(0)} = \begin{cases} n_i + \Delta & \text{if } z_i > 0 \\ 0 & \text{if } z_i \le 0 \end{cases}$$

where $\Delta \ge 0$ is a constant.

Note: For saturated models, Δ is added to n_i if $z_i > 0$. This is done to avoid numerical problems in case some observed counts are 0. We advise users to set Δ to 0 whenever all observed counts (other than structural zeros) are positive.

The initial values for η_i are

$$\eta_i^{(0)} = \begin{cases} m_i^{(0)} \log\left(m_i^{(0)}/z_i\right) + \left(n_i - m_i^{(0)}\right) & \text{if } z_i > 0 \text{ and } m_i^{(0)} > 0 \\ 0 & \text{otherwise} \end{cases}$$

Stopping Criteria

The following conditions are checked for convergence:

1. $\max_i\left(\left|m_i^{(s+1)} - m_i^{(s)}\right|/m_i^{(s)}\right) < \epsilon$ provided that $m_i > 0$
2. $\max_i\left(\left|m_i^{(s+1)} - m_i^{(s)}\right|\right) < \epsilon$
3. $\sqrt{\left(\Sigma_{k=1}^{p} g_k^2\left(\hat{\beta}\right)\right)/p} < \epsilon$

The iteration is said to be converged if either conditions 1 and 3 or conditions 2 and 3 are satisfied. If *p*=0, then condition 3 will be automatically satisfied. The iteration is said to be not converged if neither pair of conditions is satisfied within the maximum number of iterations.

Algorithm

The iteration process uses the following steps:

1. Calculate $m_i^{(0)}$ and $n_i^{(0)}$.

2. Set *s*=0.
3. Calculate $\mathbf{H}\left(\beta^{(s)}\right)$ evaluated at $m_i = m_i^{(s)}$; calculate $\mathbf{q}\left(\beta^{(s)}\right)$ evaluated at $\eta_i = \eta_i^{(s)}$.
4. Solve for $\beta^{(s+1)}$.
5. Calculate $v_i^{(s+1)} = \Sigma_{k=1}^{p} x_{ik}\beta_k^{(s+1)}$ and
6. $m_i^{(s+1)} = \begin{cases} z_i e^{\beta_0^{(s+1)} + v_i^{(s+1)}} & \text{if } z_i > 0 \\ 0 & \text{if } z_i \le 0 \end{cases}$
7. Check whether the stopping criteria are satisfied. If yes, stop iteration and declare convergence. Otherwise continue.
8. Increase *s* by 1 and check whether the maximum iteration has been reached. If yes, stop iteration and declare the process not converged. Otherwise repeat steps 3-7.

Estimated Cell Counts

The estimated expected count is

$$\hat{m}_i = \begin{cases} z_i e^{\hat{\beta}_0 + \hat{v}_i} & \text{if } z_i > 0 \\ 0 & \text{if } z_i \le 0 \end{cases}$$

where

$$\hat{v}_i = \sum_{k=1}^{p} x_{ik}\hat{\beta}_k$$

Goodness-of-Fit Statistics

The Pearson chi-square statistic is

$$X^2 = \sum_{i=1}^{r} X_i^2$$

where

$$X_i^2 = \begin{cases} (n_i - \hat{m}_i)^2 / \hat{m}_i & \text{if } z_i > 0, n_i > 0, \text{ and } \hat{m}_i > 0 \\ \text{SYSMIS} & \text{if } z_i > 0, n_i > 0, \text{ and } \hat{m}_i = 0 \\ 0 & \text{if } z_i \le 0 \text{ or } n_i = \hat{m}_i \end{cases}$$

If any X_i^2 is system missing, then X^2 is also system missing.

The likelihood-ratio chi-square statistic is

$$G^2 = 2\sum_{i=1}^{r} G_i^2$$

where

$$G_i^2 = \begin{cases} n_i(\log(n_i/\hat{m}_i)) - (n_i - \hat{m}_i) & \text{if } z_i > 0, n_i > 0, \text{ and } \hat{m}_i > 0 \\ \text{SYSMIS} & \text{if } z_i > 0, n_i > 0, \text{ and } \hat{m}_i = 0 \\ \hat{m}_i & \text{if } z_i > 0, n_i = 0, \text{ and } \hat{m}_i > 0 \\ 0 & \text{if } z_i \le 0 \text{ or } n_i = \hat{m}_i \end{cases}$$

If any G_i^2 is system missing, then G^2 is also system missing.

Degrees of Freedom

The degrees of freedom for each statistic is defined as $a = r - 1 - p - E$, where *E* is the number of cells with $z_i \le 0 \hat{m}_i$ $=.0$

Significance Level

The significance level (or the *p* value) for the Pearson chi-square statistic is Prob$(x_a^2 > X^2)$ and that for the likelihood-ratio chi-square statistic is Prob$(x_a^2 > G^2)$. In both cases, x_a^2 is the central chi-square distribution with a degrees of freedom.

Residuals

Goodness-of-fit statistics provide only broad summaries of how models fit data. The pattern of lack of fit is revealed in cell-by-cell comparisons of observed and fitted cell counts.

Simple Residuals

The simple residual of the *i*th cell is

$$r_i = \begin{cases} n_i - \hat{m}_i & \text{if } z_i > 0 \\ \text{SYSMIS} & \text{if } z_i \le 0 \end{cases}$$

Standardized Residuals

The standardized residual for the *i*th cell is

$$r_i^S = \begin{cases} (n_i - \hat{m}_i)/\sqrt{\hat{m}_i} & \text{if } z_i > 0 \text{ and } 0 < \hat{m}_i \\ 0 & \text{if } z_i > 0 \text{ and } n_i = \hat{m}_i \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

The standardized residuals are also known as Pearson residuals because $\Sigma_{i=1}^r \left(r_i^S\right)^2 = X^2$ when all $z_i > 0$. Although the standardized residuals are asymptotically normal, their asymptotic variances are less than 1.

Adjusted Residuals

The adjusted residual is the simple residual divided by its estimated standard error. This statistic for the *i*th cell is

$$r_i^A = \begin{cases} (n_i - \hat{m}_i)/\sqrt{\hat{m}_i(1-a_{ii})} & \text{if } z_i > 0, n_i \neq \hat{m}_i, \text{ and } \hat{m}_i > 0 \\ 0 & \text{if } z_i > 0 \text{ and } n_i = \hat{m}_i \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where

$$a_{ii} = \hat{m}_i \left(h^{00} + 2\sum_{k=1}^{p} x_{ik}h^{k0} + \sum_{k=1}^{p}\sum_{l=1}^{p} x_{ik}x_{il}h^{kl} \right)$$

h^{kl} is the (*k*,*l*)th element of $\mathbf{H}^{-1}\left(\hat{\beta}\right)$. The adjusted residuals are asymptotically standard normal.

Deviance Residuals

Pierce and Schafer (1986) and McCullagh and Nelder (1989) define the signed square root of the individual contribution to the G^2 statistic as the deviance residual. This statistic for the *i*th cell is

$$r_i^D = \text{sign}(n_i - \hat{m}_i)\sqrt{d_i}$$

where

$$d_i = \begin{cases} 2(n_i(\log(n_i/\hat{m}_i)) - (n_i - \hat{m}_i)) & \text{if } z_i > 0, \hat{m}_i > 0, \text{ and } n_i > 0 \\ 2\hat{m}_i & \text{if } z_i > 0, \hat{m}_i \geq 0, \text{ and } n_i = 0 \\ 0 & \text{if } z_i > 0 \text{ and } n_i = \hat{m}_i \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

When all $z_i > 0$, $\Sigma_{i=1}^{r}\left(r_i^D\right)^2 = G^2$

Generalized Residual

Consider a linear combination of the cell counts $\Sigma_{i=1}^{r} d_i n_i$, where d_i are real numbers.

The estimated expected value is

$$\sum_{i=1}^{r} d_i \hat{m}_i$$

The simple residual for this linear combination is

$$\sum_{i=1}^{r} d_i (n_i - \hat{m}_i)$$

The standardized residual for this linear combination is

$$\frac{\Sigma_{i=1}^{r} d_i(n_i - \hat{m}_i)}{\sqrt{\Sigma_{i=1}^{r} d_i^2 \hat{m}_i}}$$

Using the results in Christensen (1990, p. 227), the adjusted residual for this linear combination is

$$\frac{\Sigma_{i=1}^{r} d_i(n_i - \hat{m}_i)}{\sqrt{V}}$$

where

$$V = \sum_{i=1}^{r} \sum_{j=1}^{r} d_i d_j \hat{m}_i (\delta_{ij} - a_{ij})$$
$$= \sum_{i=1}^{r} d_i^2 \hat{m}_i - \sum_{i=1}^{r} \sum_{j=1}^{r} d_i d_j a_{ij} \hat{m}_i$$

where

$$a_{ij} = \hat{m}_i \left(h^{00} + \sum_{k=1}^{p} (x_{ik} + x_{jk}) h^{k0} + \sum_{k=1}^{p} \sum_{l=1}^{p} x_{ik} x_{jl} h^{kl} \right)$$

Generalized Log-Odds Ratio

Consider a linear combination of the natural logarithm of cell counts

$$\sum_{i=1}^{r} d_i \log(m_i)$$

where d_i are real numbers with the restriction

$$\sum_{i=1}^{r} d_i = 0$$

The linear combination is estimated by

$$\sum_{i=1}^{r} d_i \log(\hat{m}_i) = \sum_{i=1}^{r} d_j \log(z_i) + \sum_{i=1}^{r} \sum_{k=1}^{p} d_i x_{ik} \hat{\beta}_k$$

The variance is

$$\text{var}\left(\sum_{i=1}^{r} d_i \log(\hat{m}_i) \right) = \sum_{k=1}^{p} \sum_{l=1}^{p} w_k w_l h^{kl}$$

where

$$w_k = \sum_{i=1}^{r} d_i x_{ik} \quad k = 1, \ldots, p$$

Wald Statistic

The null hypothesis is

$$H_0 : \sum_{i=1}^{r} d_i \log(m_i) = 0$$

The Wald statistic is

$$W = \frac{\left(\Sigma_{i=1}^{r} d_i \log(\hat{m}_i) \right)^2}{\Sigma_{k=1}^{p} \Sigma_{l=1}^{p} w_k w_l h^{kl}}$$

Under the null hypothesis, the statistic has asymptotic chi-square distribution with 1 degree of freedom. The significance level is Prob$\left(\chi_1^2 \geq W\right)$. *Note:W* will be system missing if the variance is 0.

Asymptotic Confidence Interval

The asymptotic $(1-\alpha)\times 100\%$ confidence interval is

$$\sum_{i=1}^{r} d_i \log(\hat{m}_i) \pm z_{\alpha/2}\sqrt{\sum_{k=1}^{p}\sum_{l=1}^{p} w_k w_l h^{kl}}$$

where $z_{\alpha/2}$ is the upper $\alpha/2$ point of the standard normal distribution. The default value of α is 0.05.

Aggregated Data (Poisson)

This section shows how data are aggregated for a Poisson distribution. The following notation is used in this section:

v_i	Number of cases for $B=i(i=1,\ldots,r)$
n_{is}	*s*th caseweight for $B=i(s=1,\ldots,v_i)$
x_{is}	Covariate
z_{is}	Cell weight
c_{is}	GRESID coefficient
e_{is}	GLOR coefficient
v_i^+	Number of positive z_{is} (cell weights) for $1 \leq s \leq v_i$

The cell count is

$$n_i = \begin{cases} \Sigma^*_{1\leq s\leq v_i} n_{is}^+ & \text{if } v_i^+ > 0 \\ 0 & \text{if } v_i = 0 \text{ or } v_i^+ = 0 \end{cases}$$

where

$$n_{is}^+ = \begin{cases} n_{is} & \text{if } n_{is} > 0 \text{ and } z_{is} > 0 \\ 0 & \text{if } n_{is} \leq 0 \text{ and } z_{is} > 0 \end{cases}$$

and $\Sigma^*_{1\leq s\leq v_i}$ means summation over the range of *s* with the terms $z_{is} > 0$.

The cell weight value is

$$z_i = \begin{cases} \Sigma^*_{1\leq s\leq v_i} n_{is}^+ z_{is}/n_{ij} & \text{if } n_i > 0 \text{ and } v_i^+ > 0 \\ \Sigma^*_{1\leq s\leq v_i} z_{is}/v_i^+ & \text{if } n_i = 0 \text{ and } v_i^+ > 0 \\ 0 & \text{if } v_i^+ = 0 \\ 1 & \text{if } v_i = 0 \end{cases}$$

If no variable is specified as the cell weight variable, then all cases have unit cell weights by default.

The cell covariate value is

$$x_{ij} = \begin{cases} \Sigma^*_{1\le s\le v_i} n^+_{is} x_{is}/n_i & \text{if } n_i > 0 \text{ and } v_i > 0 \\ \Sigma^*_{1\le s\le v_i} x_{is}/v^+_i & \text{if } n_i = 0 \text{ and } v^+_i > 0 \\ 0 & \text{if } v^+_i = 0 \text{ or } v_i = 0 \end{cases}$$

The cell GRESID coefficient is

$$c_i = \begin{cases} \Sigma^*_{1\le s\le v_i} n^+_{is} c_{is}/n_i & \text{if } n_i > 0 \text{ and } v_i > 0 \\ \Sigma^*_{1\le s\le v_i} c_{is}/v^+_i & \text{if } n_i = 0 \text{ and } v^+_i > 0 \\ 0 & \text{if } v^+_i = 0 \text{ or } v_i = 0 \end{cases}$$

There are no defaults for the GRESID coefficients.

The cell GLOR coefficient is

$$e_i = \begin{cases} \Sigma^*_{1\le s\le v_i} n^+_{is} e_{is}/n_i & \text{if } n_i > 0 \text{ and } v_i > 0 \\ \Sigma^*_{1\le s\le v_i} e_{is}/v^+_i & \text{if } n_i = 0 \text{ and } v^+_i > 0 \\ 0 & \text{if } v^+_i = 0 \text{ or } v_i = 0 \end{cases}$$

There are no defaults for the GLOR coefficients.

References

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Christensen, R. 1990. *Log-linear models*. New York: Springer-Verlag.

Haberman, S. J. 1973. The analysis of residuals in cross-classified tables. *Biometrics*, 29, 205–220.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

Pierce, D. A., and D. W. Schafer. 1986. Residuals in generalized linear models. *Journal of the American Statistical Association*, 81, 977–986.

GLM Algorithms

GLM (general linear model) is a general procedure for analysis of variance and covariance, as well as regression. It can be used for both univariate, multivariate, and repeated measures designs. Algorithms that apply only to repeated measures are in “Repeated Measures ”.

For information on post hoc tests, see *Post Hoc Tests*. For sums of squares, see *Sums of Squares*. For distribution functions, see *Distribution and Special Functions*.For Box’s M test, see *Box’s M Test*.

Notation

The following notation is used throughout this chapter. Unless otherwise stated, all vectors are column vectors and all quantities are known.

n	Number of cases.
N	Effective sample size.
p	Number of parameters (including the constant, if it exists) in the model.
r	Number of dependent variables in the model.
$\mathbf{Y}$	n×r matrix of dependent variables. The rows are the cases and the columns are the dependent variables. The *i*th row is $\mathbf{y}'_i$, i=1,...,n.
$\mathbf{X}$	n×p design matrix. The rows are the cases and the columns are the parameters. The *i*th row is $\mathbf{x}'_i$, i=1,...,n.
r_X	Number of nonredundant columns in the design matrix. Also the rank of the design matrix.
w_i	Regression weight of the *i*th case.
f_i	Frequency weight of the *i*th case.
$\mathbf{B}$	unknown parameter matrix. The columns are the dependent variables. The *j*th column is $\mathbf{b}_j$, j=1,...,r.
$\mathbf{\Sigma}$	r×r unknown common multiplier of the covariance matrix of any row of $\mathbf{Y}$. The (i, j)th element is σ_{ij}, i=1,...,r, j=1,...,r.

Model

The model is $\mathbf{Y} = \mathbf{XB}$ and $\mathbf{y}'_i$ is independently distributed as a *p*-dimensional normal distribution with mean $\mathbf{x}'_i\mathbf{B}$ and covariance matrix $w_i^{-1}\Sigma$. The *i*th case is ignored if $w_i \le 0$.

Frequency Weight and Total Sample Size

The frequency weight f_i is the number of replications represented by a case in IBM® SPSS® Statistics; therefore, the weight must be a non-negative integer. It is computed by rounding the value in the weight variable to the nearest integer. The total sample size is $N = \Sigma_{i=1}^{n} f_i I(w_i > 0)$, where $(w_i \ \ > 0) = 1$ if $w_i > 0$ and is equal to 0 otherwise.

The Cross-Product and Sums-of-Squares Matrices

To prepare for the SWEEP operation, an augmented row vector of length $(p+r)$ is formed:

$$\mathbf{z}'_i = \left(\mathbf{x}'_i, \mathbf{y}'_i\right)$$

Then the $(p+r)\times(p+r)$ matrix is computed:

$$\mathbf{Z}'\mathbf{W}\mathbf{Z} = \Sigma_{i=1}^{n} f_i w_i \mathbf{z}_i \mathbf{z}'_i$$

This matrix is partitioned as

$$\mathbf{Z}'\mathbf{W}\mathbf{Z} = \begin{pmatrix} \mathbf{X}'\mathbf{W}\mathbf{X} & \mathbf{X}'\mathbf{W}\mathbf{Y} \\ \mathbf{Y}'\mathbf{W}\mathbf{X} & \mathbf{Y}'\mathbf{W}\mathbf{Y} \end{pmatrix}$$

The upper left *p*×*p* submatrix is **X’WX** and the lower right *r*×*r* submatrix is **Y’WY**.

Sweep Operation

Three important matrices, **G**, $\hat{\mathbf{B}}$, and **S**, are obtained by sweeping the **Z’WZ** matrix as follows:

1. Sweep sequentially the first *p* rows and the first *p* columns of **Z’WZ**, starting from the first row and the first column.
2. After the *p*th row and the *p*th column are swept, the resulting matrix is

$$\begin{pmatrix} -\mathbf{G} & \hat{\mathbf{B}} \\ \hat{\mathbf{B}}' & \mathbf{S} \end{pmatrix}$$

where **G** is a *p*×*p* symmetric g_2 generalized inverse of **X’WX**, $\hat{\mathbf{B}}$ is the *p*×*r* matrix of parameter estimates and **S** is the *r*×*r* symmetric matrix of sums of squares and cross products of residuals.

The SWEEP routine is adapted from Algorithm AS 178 by Clarke (1982) and Remarks R78 by Ridout and Cobby (1989).

Residual Covariance Matrix

The estimated *r*×*r* covariance matrix is $\hat{\Sigma} = \mathbf{S}/(N - r_{\mathbf{X}})$ provided $r_X < N$. If $r_X = N$, then $\hat{\Sigma} = 0$. If $r_X > N$, then all elements of $\hat{\Sigma}$ are system missing. The residual degrees of freedom is $N - r_X$. If $r_X > N$, then the degrees of freedom is system missing.

Lack of Fit

Source of Variation	Sum of Squares	df
Lack of fit	$\sum_{i=1}^{n_u} n_i(\overline{y}_i - \hat{y}_i)^2$	$n_u - p$
Pure error	$\sum_{i=1}^{n_u}\sum_{j=1}^{n_i}(y_{ij} - \overline{y}_i)^2$	$N - n_u$

Where n_u is the number of unique combinations of observed predictor values and n_i is the number of cases with the *i*th combination.

Mean squares are calculated by dividing each sum of squares by its degrees of freedom.

The *F* ratio for testing lack of fit is the ratio of the Lack of fit mean squares to the Pure error mean squares.

The significance level is obtained from the *F* distribution with $n_u - p$ and $N - n_u$ degrees of freedom.

Parameter Estimates

Let the elements of $\hat{\Sigma}$ be $\hat{\sigma}_{ij}$, the elements of **G**, g_{ij}, and the elements of $\hat{\mathbf{B}}$, $\hat{b}_{ij}$. Then $var\left(\hat{b}_{ij}\right)$ is estimated by $\hat{\sigma}_{jj}g_{ii}$ for *i*=1,...,*p*, *j*=1,...,*r* and $cov\left(\hat{b}_{ij}, \hat{b}_{rs}\right)$ is estimated by $\hat{\sigma}_{js}g_{ir}$ for *i*, *r*=1,...,*p*, *j*, *s*=1,...,*r* .

Standard Error of the Estimate

$$se\left(\hat{b}_{ij}\right) = \sqrt{\hat{\sigma}_{jj}g_{ii}}$$

When the *i*th parameter is redundant, the standard error is system missing.

The t Statistic

For testing $\mathrm{H}_0 : b_{ij} = 0$ versus $\mathrm{H}_1 : b_{ij} \neq 0$, the *t* statistic is

$$t = \begin{cases} \hat{b}_{ij}/\mathrm{se}\left(\hat{b}_{ij}\right) & \text{if the standard error is positive} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

The significance value for this statistic is $2(1 - \text{CDF.T}(|t|, N - r_x))$ where CDF.T is the IBM® SPSS® Statistics function for the cumulative t distribution.

Partial Eta Squared Statistic

$$\eta^2 = \begin{cases} \hat{b}_{ij}^2/\left(\hat{b}_{ij}^2 + (N - r_X)\, var\left(\hat{b}_{ij}\right)\right) & \text{if } r_X < N \text{ and the denominator is positive} \\ 1 & \text{if } r_X = N \text{ but } b_{ij} \neq 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

The value should be within $0 \leq \eta^2 \leq 1$.

Noncentrality Parameter

$$c = |t|$$

Observed Power

$$p = \begin{cases} 1 - \text{NCDF.T}\left(t_c, N - r_X, c\right) + \text{NCDF.T}\left(-t_c, N - r_X, c\right) & r_X < N \\ \text{SYSMIS} & r_X \geq N, \text{ or any arguments to NCDF.T or IDF.T are SYSMIS} \end{cases}$$

where $t_c = \text{IDF.T}(1 - \frac{\alpha}{2}, N - r_x)$ and α is the user-specified chance of Type I error $(0 < \alpha < 1)$. NCDF.T and IDF.T are the IBM® SPSS® Statistics functions for the cumulative noncentral t distribution and for the inverse cumulative t distribution, respectively.

The default value is $\alpha = 0.05$. The observed power should be within $0 \leq p \leq 1$.

Confidence Interval

For the *p*% level, the individual univariate confidence interval for the parameter is

$$\hat{b}_{ij} \pm t_\alpha se\left(\hat{b}_{ij}\right)$$

where $t_\alpha = \text{IDF.T}(0.5(1 + p/100), N - r_x)$ for *i*=1,…,*n*,*j*=1,…,*r*. The default value of *p* is 95 (0<*p*<100).

Correlation

$$\text{corr}\left(\hat{b}_{ij}, \hat{b}_{rs}\right) = \begin{cases} \hat{\sigma}_{js} g_{ir} / \left(se\left(\hat{b}_{ij}\right) \times se\left(\hat{b}_{rs}\right)\right) & \text{if the standard errors are positive} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

for *i*, *r*=1,...,*n*, *j*, *s*=1,...,*r*.

Estimated Marginal Means

Estimated marginal means (EMMEANS) are computed as the generic $\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}$ expression with appropriate **l** and **m** vectors. **l** is a column vector of length *p* and **m** is a column vector of length *r*. Since the **l** vector is chosen to be always estimable, the quantity $\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}$ is in fact the estimated modified marginal means (Searle, Speed, and Milliken, 1980). When covariates (or products of covariates) are present in the effects, the overall means of the covariates (or products of covariates) are used in the **l** matrix. Suppose X and Y are covariates and they appear as X*Y in an effect; then the mean of X*Y is used instead of the product of the mean of X and the mean of Y.

L Matrix

For each level combination of the between subjects factors in TABLES, identify the nonmissing cases with positive caseweights and positive regression weights which are associated with the current level combination. Suppose the cases are classified by three between-subjects factors: A, B and C. Now A and B are specified in TABLES and the current level combination is A=1 and B=2. A case in the cell A=1, B=2, and C=3 is associated with the current level combination, whereas a case in the cell A=1, B=3 and C=3 is not. Compute the average of the design matrix rows corresponding to these cases.

If an effect contains a covariate, then its parameters which belong to the current level combination are equal to the mean of the covariate, and are equal to 0 otherwise. Using the above example, for effect A*X where X is a covariate, the parameter [A=1]*X belongs to the current level combination where the parameter [A=2]*X does not. If the effect contains a product of covariates, then the mean of the product is applied.

The result is the **l** vector for the current between-subjects factor level combination. When none of the between-subjects effects contain covariates, the vector always forms an estimable function. Otherwise, a non-estimable function may occur, depending on the data.

M Matrix

The M matrix is formed as a series of Kronecker products

$$\mathbf{M} = \mathbf{I}_c \otimes \mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_t$$

where

$$\mathbf{A}_k = \begin{cases} \mathbf{I}_{r_k} & \text{if the } k\text{th within subjects factor is specified in TABLES} \\ (1/r_k)\mathbf{1}_{r_k} & \text{otherwise} \end{cases}$$

with $\mathbf{1}_{r_k}$ a column vector of length r_k and all of its elements equal to 1.

If OVERALL or only between-subjects factors are specified in TABLES, then $\mathbf{A}_k = (1/r_k)\mathbf{1}_{r_k}$ for *k*=1,...,*t*.

The column for a particular within-subjects factor level combination, denoted by **m**, is extracted accordingly from this **M** matrix.

Standard Error

$$se\left(\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}\right) = \begin{cases} \sqrt{(\mathbf{l}'\mathbf{G}\mathbf{l})\left(\mathbf{m}'\hat{\Sigma}\mathbf{m}\right)} & \text{if } N - r_{\mathbf{X}} > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Since **l** are coefficients of an estimable function, the standard error is the same for any generalized inverse **G**.

Significance

The *t* statistic is

$$t = \begin{cases} \mathbf{l}'\hat{\mathbf{B}}\mathbf{m}/se\left(\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}\right) & \text{if } se\left(\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}\right) > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

If the *t* statistic is not system missing, then the significance is computed based on a *t* distribution with $N - r_{\mathbf{X}}$ degrees of freedom.

Pairwise Comparison

The levels of a between-subjects or within-subjects factor can be compared pair-by-pair. For example, a factor with 3 levels produces 3 pairwise comparisons: 1 vs. 2, 1 vs. 3, and 2 vs. 3.

Between-Subjects Factor

Suppose the **l** vectors are indexed by the level of the between-subjects factor as $\mathbf{l}_{i_1,\ldots,i_b}$, $i_s = 1, \ldots, n_s$ and $s = 1, \ldots, b$ where n_s is the number of levels of between-subjects factor *s* and *b* is the number of between-subjects factors specified inside TABLES. The difference in estimated marginal means of level i_s and level i'_s of between-subjects factor *s* at fixed levels of other between-subjects factors is

$$\left(\mathbf{l}_{i_1,\ldots,i_{s-1},i_s,i_{s+1},\ldots,i_b} - \mathbf{l}_{i_1,\ldots,i_{s-1},i'_s,i_{s+1},\ldots,i_b}\right)'\hat{\mathbf{B}}\mathbf{m} \text{ for } i_s, i'_s = 1, \ldots, n_s; i_s \neq i'_s$$

The standard error of the difference is computed by substituting for **l** in (1): $\mathbf{l}_{i_1,\ldots,i_{s-1},i_s,i_{s+1},\ldots,i_b} - \mathbf{l}_{i_1,\ldots,i_{s-1},i'_s,i_{s+1},\ldots,i_b}$.

Within-Subjects Factor

Suppose the **m** vectors are indexed by level of the within-subjects factor as $\mathbf{m}_{j_1,\ldots,j_w}$, $j_s = 1, \ldots, n_s$ and $s = 1, \ldots, w$, where n_s is the number of levels of within-subjects factor *s* and *w* is the number of within-subjects factors specified inside TABLES. The difference in estimated marginal means of level j_s and level j'_s of within-subjects factor *s* at fixed levels of other within-subjects factors is

$$\mathbf{l}'\mathbf{B}\left(\mathbf{m}_{j_1,\ldots,j_{s-1},j_s,j_{s+1},\ldots,j_b} - \mathbf{m}_{j_1,\ldots,j_{s-1},j'_s,j_{s+1},\ldots,j_b}\right) \text{ for } j_s, j'_s = 1, \ldots, n_s; j_s \neq j'_s$$

The standard error of the difference is computed by substituting for **m** in (1) $\mathbf{m}_{i_1,\ldots,i_{s-1},i_s,i_{s+1},\ldots,i_b} - \mathbf{m}_{i_1,\ldots,i_{s-1},i'_s,i_{s+1},\ldots,i_b}$

Confidence Interval

The $(1-\alpha)\times 100\%$ confidence interval is:

$$\mathbf{l}'\hat{\mathbf{B}}\mathbf{m} \pm t_{1-\alpha/2;N-r_{\mathbf{X}}} \times se\left(\mathbf{l}'\hat{\mathbf{B}}\mathbf{m}\right)$$

and $t_{1-\alpha/2;N-r_{\mathbf{X}}}$ is the $(1-\alpha/2)\times 100\%$ percentile of a *t* distribution with $N - r_{\mathbf{X}}$ degrees of freedom. No confidence interval is computed if $N - r_{\mathbf{X}} \leq 0$.

Saved Values

Temporary variables can be added to the working data file. These include predicted values, residuals, and diagnostics.

Predicted Values

The $n \times r$ matrix of predicted values is $\hat{\mathbf{Y}} = \mathbf{X}\hat{\mathbf{B}}$. The *i*th row of $\hat{\mathbf{Y}}$ is $\hat{\mathbf{y}}_i' = \mathbf{x}_i'\hat{\mathbf{B}}$, i=1,...,n. Let the elements of $\hat{\mathbf{Y}}$ be $\hat{y}_{ij}$ and the elements of **XGX'** be π_{ij}.

The standard error of $\hat{y}_{ij}$ is

$se(\hat{y}_{ij}) = \sqrt{\hat{\sigma}_{jj}\pi_{ii}}$ for i=1,...,n, j=1,...,r

The weighted predicted value of the *i*th case is $\sqrt{w_i}\hat{\mathbf{y}}_i'$.

Residuals

The $n \times r$ matrix of residuals is $\hat{\mathbf{E}} = \mathbf{Y} - \hat{\mathbf{Y}}$

The *i*th row of $\hat{\mathbf{E}}$ is $\hat{\mathbf{e}}_i' = \mathbf{y}_i' - \hat{\mathbf{y}}_i'$, i=1,...,n.

Let the elements of $\hat{\mathbf{E}}$ be $\hat{e}_{ij}$; then

$\hat{e}_{ij} = y_{ij} - \hat{y}_{ij}$, for i=1,...,n, j=1,...,r

The weighted residual is $\sqrt{w_i}\hat{\mathbf{e}}_i'$.

Deleted Residuals (PRESS Residuals)

The deleted residual is the predicted residual for the *i*th case that results from omitting the *i*th case from estimation. It is:

$$\text{DRESID}_{ij} = \begin{cases} \hat{e}_{ij}/(1/w_i - \pi_{ii}) & \text{if } w_i > 0 \text{ and } w_i\pi_{ii} < 1; \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

for i=1,...,n, j=1,...,r.

Standardized Residuals

The standardized residual is the residual divided by the standard deviation of data:

$$\text{ZRESID}_{ij} = \begin{cases} (y_{ij} - \hat{y}_{ij})/\left(\sqrt{\hat{\sigma}_{jj}/w_i}\right) & \text{if } w_i > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Studentized Residuals

The standard error for $\hat{e}_{ij}$ is

$$se(\hat{e}_{ij}) = \begin{cases} \sqrt{\hat{\sigma}_{jj}(1/w_i - \pi_{ii})} & \text{if } w_i > 0 \text{ and } w_i\pi_{ii} < 1; \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

for i=1,...,n, j=1,...,r. The Studentized residual is the residual divided by the standard error of the residual.

$$\text{SRESID}_{ij} = \begin{cases} \hat{e}_{ij}/se(\hat{e}_{ij}) & \text{if } w_i > 0 \text{ and } se(\hat{e}_{ij}) > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Diagnostics

The following diagnostic statistics are available.

Cook's Distance

Cook's Distance D measures the change to the solution that results from omitting each observation. The formula is

$$D_{ij} = \left(\frac{\hat{e}_{ij}}{\sqrt{\hat{\sigma}_{jj}(1/w_i - \pi_{ii})}}\right)^2 \left(\frac{\pi_{ii}}{(1/w_i - \pi_{ii})}\right)\frac{1}{r_X}$$

for i=1,...,n, j=1,...,r. This formula is equivalent to

$D_{ij} = (\hat{e}_{ij}/se(\hat{e}_{ij}))^2 (se(\hat{y}_{ij})/se(\hat{e}_{ij}))/r_X$ provided $w_i > 0$ and $se(\hat{e}_{ij}) > 0$

When $w_i \le 0$ or $se(\hat{e}_{ij}) = 0$, D_{ij} is system missing

Leverage

The leverage for the ith case (i=1,...,n) for all dependent variables is

$$\text{LEVER}_i = \begin{cases} w_i \pi_{ii} & \text{if } w_i > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Hypothesis Testing

Let **L** be an l×p known matrix, **M** be an r×m known matrix and **K** be an l×m known matrix. The test hypotheses $\text{H}_0 : \mathbf{LBM} = \mathbf{K}$ versus $\text{H}_1 : \mathbf{LBM} \neq \mathbf{K}$ are **testable** if and only if **LB** is estimable.

The following results apply to testable hypotheses only. Nontestable hypotheses are excluded.

The hypothesis SSCP matrix is $\mathbf{S}_H = \left(\mathbf{L\hat{B}M} - \mathbf{K}\right)' \left(\mathbf{LGL}'\right)^{-1} \left(\mathbf{L\hat{B}M} - \mathbf{K}\right)$and the error SSCP matrix is $\mathbf{S}_E = \mathbf{M}'\mathbf{SM}$.

Four test statistics, based on the eigenvalues of $\mathbf{S}_E^{-1}\mathbf{S}_H$, are available: Wilks' lambda, Hotelling-Lawley trace, Pillai's trace, and Roy's largest root.

Let the eigenvalues of $\mathbf{S}_E^{-1}\mathbf{S}_H$ be $\lambda_1 \ge \cdots \ge \lambda_{r_E} \ge 0$ and $\lambda_{r_{E+1}}, \ldots, \lambda_m = 0$, and let $r_E = rank(\mathbf{S}_E); s = \min(l, r_E); n_e = n - r_X; m^* = \frac{1}{2}(|r_E - l| - 1); n^* = \frac{1}{2}(n_e - r_E - 1)$.

Wilks' Lambda

$$\Lambda = \frac{det(\mathbf{S}_E)}{det(\mathbf{S}_H+\mathbf{S}_E)} = \prod_{k=1}^{m} \frac{1}{(1+\lambda_k)}$$

When H_0 is true, the F statistic

$$F = \frac{(\varsigma\tau - 2\upsilon)}{lr_E} \frac{\left(1-\Lambda^{1/\tau}\right)}{\Lambda^{1/\tau}}$$

follows asymptotically an F distribution, where

$$\varsigma = n_e - \tfrac{1}{2}(r_E - l + 1)$$
$$\upsilon = \tfrac{1}{4}(lr_E - 2)$$
$$\tau = \begin{cases} \sqrt{(l^2 r_E^2 - 4)/(l^2 + r_E^2 - 5)} & \text{if } \left(l^2 + r_E^2 - 5\right) > 0 \\ 1 & \text{otherwise} \end{cases}$$

The degrees of freedom are $(lr_E, \varsigma\tau - 2\upsilon)$. The F statistic is exact if s=1,2. See Rao (1951) and Section 8c.5 of Rao (1973) for details.

The eta-squared statistic is $\eta^2 = 1 - \Lambda^{1/s}$.

The noncentrality parameter is $\lambda = (\xi\tau - 2\upsilon)\eta^2/\left(1-\eta^2\right)$.

The power is $1 - \text{NCDF.F}(\text{F}_\alpha, lr_E, (\xi\tau - 2\upsilon), \lambda)$ where F_α is the upper 100α percentage point of the central F distribution, and α is user-specified on the ALPHA keyword on the CRITERIA subcommand.

Hotelling-Lawley Trace

In IBM® SPSS® Statistics, the name Hotelling-Lawley trace is shortened to Hotelling's trace

$$T = trace\left(\mathbf{S}_E^{-1}\mathbf{S}_H\right) = \Sigma_{k=1}^{m}\lambda_k$$

When H_0 is true, the F statistic

$$F = \frac{2(sn^*+1)}{s(2m^*+s+1)} \frac{T}{s}$$

follows asymptotically an F distribution with degrees of freedom $(s(2m^* + s + 1), 2(sn^* + 1))$. The F statistic is exact if s=1.

The eta-squared statistic is $\eta^2 = (T/s)/(T/s + 1)$.

The noncentrality parameter is $\lambda = 2(sn^* + 1)\eta^2/\left(1-\eta^2\right)$.

The power is $1 - \text{NCDF.F}(\text{F}_\alpha, s(2m^* + s + 1), 2(sn^* + 1), \lambda)$ where F_α is the upper 100α percentage point of the central F distribution, and α is user-specified on the ALPHA keyword on the CRITERIA subcommand.

Pillai's Trace

$$V = trace\left(\mathbf{S}_H(\mathbf{S}_H + \mathbf{S}_E)^{-1}\right) = \Sigma_{k=1}^{m}\lambda_k/(1+\lambda_k)$$

When H_0 is true, the F statistic

$$F = \frac{(2n^*+s+1)}{(2m^*+s+1)}\frac{V}{(s-V)}$$

follows asymptotically an F distribution with degrees of freedom $(s(2m^*+s+1), s(2n^*+s+1))$ The F statistic is exact if s=1.

The eta-squared statistic is $\eta^2 = V/s$.

The noncentrality parameter is $\lambda = s(2n^*+s+1)\eta^2/\left(1-\eta^2\right)$.

The power is 1 – NCDF.F($F_\alpha, s(2m^*+s+1), s(2n^*+s+1), \lambda$) where F_α is the upper 100α percentage point of the central F distribution, and α is user-specified on the ALPHA keyword on the CRITERIA subcommand.

Roy's Largest Root

$$\Theta = \lambda_1$$

which is the largest eigenvalue of $\mathbf{S}_E^{-1}\mathbf{S}_H$. When H_0 is true, the F statistic is

$$F = \Theta(n_e - \omega + r_H)/\omega$$

where $\omega = \max(l, r_E)$ is an upper bound of F that yields a lower bound on the significance level. The degrees of freedom are $(\omega, n_e - \omega + r_H)$. The F statistic is exact if s=1.

The eta-squared statistic is $\eta^2 = \Theta/(1+\Theta)$.

The noncentrality parameter is $\lambda = (n_e - \omega + r_H)\eta^2/\left(1-\eta^2\right)$.

The power is 1 – NCDF.F($F_\alpha, \omega.n_e - \omega + l, \lambda$) where F_α is the upper 100α percentage point of the central F distribution, and α is user-specified on the ALPHA keyword on the CRITERIA subcommand.

Individual Univariate Test

$$F = \frac{\mathbf{S}_{H;i}/l}{\mathbf{S}_{E;i}/(n-r_{\mathbf{X}})} i=1,\ldots,m$$

where $\mathbf{S}_{H;i}$ and $\mathbf{S}_{E;i}$ are the ith diagonal elements of the matrices $\mathbf{S}_H$ and $\mathbf{S}_E$ respectively. Under the null hypothesis, the F statistic has an F distribution with degrees of freedom $(l, n-r_X)$.

The eta-squared statistic is $\eta^2 = \mathbf{S}_{H;i}/(\mathbf{S}_{H;i} + \mathbf{S}_{E;i})$

The noncentrality parameter is $\lambda = (n - r_{\mathbf{X}})\mathbf{S}_{H;i}/\mathbf{S}_{E;i}$.

The power is 1 – NCDF.F$(\mathrm{F}_{\alpha}, 1, n - r_X, \lambda)$ where F_{α} is the upper 100α percentage point of the central *F* distribution, and α is user-specified on the ALPHA keyword on the CRITERIA subcommand.

Bartlett's Test of Sphericity

Bartlett's test of sphericity is printed when the Residual SSCP matrix is requested.

Hypotheses

In Bartlett's test of sphericity the null hypothesis is $H_o \; : \Sigma = \; \sigma^2 \mathbf{I}_r$ versus the alternative hypothesis $H_1 \; : \Sigma \neq \sigma^2 \mathbf{I}_r$ where $\sigma^2 > 0$ is unspecified and $\mathbf{I}_r$ is an *r*×*r* identity matrix.

Likelihood Ratio Test Statistic

$$\lambda = \begin{cases} \frac{|\mathbf{A}|^{n/2}}{(trace(\mathbf{A})/r)^{nr/2}} & \text{if } trace(\mathbf{A}) > 0 \\ \text{SYSMIS} & \text{if } trace(\mathbf{A}) \leq 0 \end{cases}$$

where $\mathbf{A} = \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right)' \mathbf{W} \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right)$ is the *r*×*r* matrix of residual sums of squares and cross products.

Chi-Square Approximation

Define $W = \lambda^{2/n}$. When *n* is large and under the null hypothesis that for $n - r_X \geq 1$ and $r \geq 2$,

$$\Pr\left(-\rho(n - r_X)\log W \leq c\right) = \Pr\left(\chi^2_f \leq c\right) + \omega_2\left(\Pr\left(\chi^2_{f+4} \leq c\right) - \Pr\left(\chi^2_f \leq c\right)\right) + O(n^{-3})$$

where

$$f = r(r+1)/2 - 1$$
$$\rho = 1 - \left(2r^2 + r + 2\right)/(6r(n - r_X))$$
$$\omega_2 = \frac{(r+2)(r-1)(r-2)\left(2r^3 + 6r^2 + 3r + 2\right)}{288r^2(n - r_X)^2\rho^2}$$

Chi-Square Statistic

$$c = \begin{cases} -\rho(n - r_X)\log W & \text{if } W > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Degrees of Freedom

$$f = r(r+1)/2 - 1$$

Significance

1 – CDF.CHISQ(c, f) – ω_2(CDF.CHISQ$(c, f+4)$ – CDF.CHISQ(c, f))

where CDF.CHISQ is the IBM® SPSS® Statistics function for the cumulative chi-square distribution. The significance is reset to zero whenever the computed value is less than zero due to floating point imprecision.

Custom Hypothesis Tests

The TEST subcommand offers custom hypothesis tests. The hypothesis term is any effect specified (either explicitly or implicitly) in the DESIGN subcommand. The error term can be a linear combination of effects that are specified in the DESIGN subcommand or a sum of squares with specified degrees of freedom. The TEST subcommand is available only for univariate analysis; therefore, an F statistic is computed. When the error term is a linear combination of effects and no value for degrees of freedom is specified, the error degrees of freedom is approximated by the Satterthwaite (1946) method.

Notation

The following notation is used in this section:

S	Number of effects in the linear combination
q_s	Coefficient of the sth effect in the linear combination, s=1,...,S.
l_s	Degrees of freedom of the sth effect in the linear combination, s=1,...,S.
MS_s	Mean square of the sth effect in the linear combination, s=1,...,S.
Q	Linear combination of effects
l_Q	Degrees of freedom of the linear combination
MS_Q	Mean square of the linear combination

Error Mean Square

If the error term is a linear combination of effects, the error mean square is

$$\mathrm{MS}_Q = \sum_{s=1}^{S} q_s \times \mathrm{MS}_s$$

If the user supplied the mean squares, $\mathrm{MS}_Q$is equal to the number specified after the keyword VS. If $\mathrm{MS}_Q < 0$, the custom error term is invalid, and $\mathrm{MS}_Q$is equal to the system-missing value and an error message is issued.

Degrees of Freedom

If $\mathrm{MS}_Q \geq 0$ and the user did not supply the error degrees of freedom, then the error degrees of freedom is approximated using the Satterthwaite (1946) method. Define

$$d_s = \begin{cases} (q_s \mathrm{MS}_s)^2 / l_s & \text{if } l_s > 0 \\ 0 & \text{otherwise} \end{cases}$$

Then $D = \sum_{s=1}^{S} d_s$. The approximate error degrees of freedom is

$$l_Q = \begin{cases} (\mathrm{MS}_Q)^2/D & \text{if } D > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

If $\mathrm{MS}_Q \geq 0$ and the user supplied the error degrees of freedom, l_Q is equal to the number following the keyword DF. If $l_Q < 0$, the custom degrees of freedom is invalid. In this case, l_Q is equal to the system-missing value and an error message is issued.

F Statistic

The null hypothesis is that all parameters of the hypothesis effect are zero. The *F* statistic is used for testing this null hypothesis. Suppose the mean square and the degrees of freedom of the hypothesis effect are MS_H and l_H; then the *F* statistic is

$$F = \begin{cases} \frac{\mathrm{MS}_H}{\mathrm{MS}_Q} & \text{if } \mathrm{MS}_Q > 0 \text{ and } \mathrm{MS}_H \geq 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Significance

$$\text{significance} = \begin{cases} 1 - \text{CDF.F}(F, l_H, l_Q) & \text{if } l_H > 0, l_Q > 0 \text{ and } F \neq \text{SYSMIS} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where CDF.F is the IBM® SPSS® Statistics function for the F cumulative distribution function.

Univariate Mixed Model

This section describes the algorithms pertaining to a random effects model. GLM offers mixed model analysis only for univariate models—that is, for *r*=1.

Notation

The following notation is used throughout this section. Unless otherwise stated, all vectors are column vectors and all quantities are known.

k	Number of random effects.
p_0	Number of parameters in the fixed effects.
p_i	Number of parameters in the *i*th random effect, *i*=1,...,*k*.
σ_i^2	Unknown variance of the ith random effect, $\sigma_i^2 \geq 0$, *i*=1,...,*k*.
σ_e^2	Unknown variance of the residual term, $\sigma_e^2 > 0$.
$\mathbf{X}_i$	The $n \times p_i$ design matrix, *i*=0,1,...,*k*.
β_0	The length p_0 vector of parameters of the fixed effects.

β_i	The length p_i vector of parameters of the *i*th random effect, *i*=1,...,*k*.
$\mathbf{L}$	The *s*×*p* full row rank matrix. The rows are estimable functions. *s*≥1

Relationships between these symbols and those defined at the beginning of the chapter are:

- $p = p_0 + p_1 + \cdots + p_k$
- $\mathbf{X} = [\mathbf{X}_0|\mathbf{X}_1|\ldots|\mathbf{X}_k]$
- $\mathbf{B} = \begin{bmatrix}\beta_0\\ \beta_1\\ \\ .\\ \beta_k\end{bmatrix}$

Model

The mixed model is represented, following Rao (1973), as

$$\mathbf{Y} = \mathbf{X}_0\beta_0 + \sum_{i=1}^{k}\mathbf{X}_i\beta_i + \mathbf{e}$$

The random vectors $\beta_1, \ldots, \beta_k$ and **e** are assumed to be jointly independent. Moreover, the random vector β_i is distributed as $N_{p_i}\left(\mathbf{0}, \sigma_i^2\mathbf{I}_{p_i}\right)$ for *i*=1,...,*k* and the residual vector **e** is distributed as $N_n\left(\mathbf{0}, \sigma_e^2\mathbf{W}^{-1}\right)$. Thus,

$$E(\mathbf{Y}) = \mathbf{X}_0\beta_o$$
$$cov(\mathbf{Y}) = \sum_{i=1}^{k}\sigma_i^2\mathbf{X}_i\mathbf{X}'_i + \sigma_e^2\mathbf{W}^{-1}$$

Expected Mean Squares

For the estimable function **L**, the expected hypothesis sum of squares is

$$E(SS_L) = E\left(\mathbf{Y}'\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{Y}\right)$$
$$= \beta_0'\mathbf{X}'_0\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X}_0\beta_0 + \sum_{i=1}^{k}\sigma_i^2 trace\left(\mathbf{X}'_k\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X}_k\right) + \sigma_e^2 trace(\mathbf{A}_L)$$

where

$$\mathbf{A}_L = \mathbf{W}^{\frac{1}{2}}\mathbf{X}\mathbf{G}\mathbf{L}'\left(\mathbf{L}\mathbf{G}\mathbf{L}'\right)^{-1}\mathbf{L}\mathbf{G}\mathbf{X}'\mathbf{W}^{\frac{1}{2}}$$

Since $\mathbf{L} = \mathbf{L}\mathbf{G}\mathbf{X}'\mathbf{W}\mathbf{X}$, $trace(\mathbf{A}_L) = s$ and $\mathbf{X}'\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X} = \mathbf{L}'\left(\mathbf{L}\mathbf{G}\mathbf{L}'\right)^{-1}\mathbf{L}$. The matrix $\mathbf{X}'\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X}$ can therefore be computed in the following way:

1. Compute an *s*×*s* upper triangular matrix **U** such that $\mathbf{U}'\mathbf{U} = \mathbf{L}\mathbf{G}\mathbf{L}'$ by the Choleskydecomposition.
2. Invert the matrix **U** to give $\mathbf{U}^{-1}$.
3. Compute $\mathbf{C} = \mathbf{L}'\mathbf{U}^{-1}$.

Now we have $\mathbf{X}'\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X} = \mathbf{C}\mathbf{C}'$. If the rows of **C** are partitioned into the same-size submatrices as those contained in **X**—that is,

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_0 \\ \mathbf{C}_1 \\ \\ . \\ \mathbf{C}_k \end{bmatrix}$$

where $\mathbf{C}_i$ is a $p_i \times s$ submatrix—then $\mathbf{X}'_k\mathbf{W}^{\frac{1}{2}}\mathbf{A}_L\mathbf{W}^{\frac{1}{2}}\mathbf{X}_k = \mathbf{C}_i\mathbf{C}'_i$, *i*=0,1,...,*k*.

Since $trace\left(\mathbf{C}_i\mathbf{C}'_i\right)$ is equal to the sum of squares of the elements in $\mathbf{C}_i$, denoted by $SSQ(\mathbf{C}_i)$, the matrices $\mathbf{C}_i\mathbf{C}'_i$ need not be formed. The preferred computational formula for the expected sum of squares is

$$E(SS_L) = \beta_0'\mathbf{C}_0\mathbf{C}_0'\beta_0 + \sum_{i=1}^{k}\sigma_i^2 SSQ(\mathbf{C}_i) + s\sigma_e^2$$

Finally the expected mean square is

$$E(MS_L) = \frac{1}{s}E(SS_L)$$

For the residual term, the expected residual mean square is: $E(MSE) = \sigma_e^2$.

Note: GLM does not compute the term $\frac{1}{s}\beta 0'\mathbf{C}_0\mathbf{C}'_0\beta_0$ but reports the fixed effects whose corresponding row block in $\mathbf{C}_0$ contains nonzero elements.

Hypothesis Test in Mixed Models

Suppose MS_L is the mean square for the effect whose estimable function is **L**, and s_L is the associated degrees of freedom. The *F* statistic for testing this effect is

$$F = \frac{MS_L}{MS_{E(L)}}$$

where $MS_{E(L)}$ is the mean square of the error term with $s_{E(L)}$ degrees of freedom.

Null Hypothesis Expected Mean Squares

If the effect being tested is a fixed effect, its expected mean square is

$$E(MS_L) = \sigma_e^2 + c_1\sigma_1^2 + \cdots + c_k\sigma_k^2 + Q(L)$$

where $c_1, \ldots, c_k$ are coefficients and $Q(L)$ is a quadratic term involving the fixed effects. Under the null hypothesis, it is assumed that $Q(L) = 0$. Although the quadratic term may involve effects that are unrelated to the effect being tested, such effects are assumed to be zero in order to draw a correct inference for the effect being tested. Therefore, under the null hypothesis, the expected mean square is

$$E(MS_L) = \sigma_e^2 + c_1\sigma_1^2 + \cdots + c_k\sigma_k^2$$

If the effect being tested is a random effect, say the *j*th $(1 \le j \le k)$ random effect, its expected mean square is

$$E(MS_L) = \sigma_e^2 + c_1\sigma_1^2 + \cdots + c_k\sigma_k^2$$

Under the null hypothesis $\sigma_j^2 = 0$; hence, the expected mean square is

$$E(MS_L) = \sigma_e^2 + \sum_{1 \le i \le k, i \ne j} c_i\sigma_i^2$$

Error Mean Squares

Let MS_i be the mean square of the *i*th $(i = 1, \ldots, k)$ random effect. Let s_i be the corresponding degrees of freedom. The error term is then found as a linear combination of the expected mean squares of the random effects:

$$MS_{E(L)} = q_1 MS_1 + \cdots + q_k MS_k + q_{k+1} MSE$$

such that

$$E\left(MS_{E(L)}\right) = q_1 E(MS_1) + \cdots + q_k E(MS_k) + q_{k+1} E(MSE) = \sigma_e^2 + c_1\sigma_1^2 + \cdots + c_k\sigma_k^2$$

If $s_i = 0 (1 \le i \le k)$ then $q_i = 0$.

The error degrees of freedom is computed using the Satterthwaite (1946) method:

$$s_{E(L)} = \frac{\left(MS_{E(L)}\right)^2}{\sum_{1 \le i \le k; s_i > 0} \left(q_i MS_i\right)^2 / s_i}$$

If the design is balanced, the above *F* statistic is approximately distributed as an *F* distribution with degrees of freedom $\left(s_L, s_{E(L)}\right)$ under the null hypothesis. The statistic is exact when only one random effect is used as the error term—that is, $q_{i_0} = 1$ and $q_i = 0$ for $i \ne i_0$. If the design is not balanced, the above approximation may not be valid (even when only one random effect is used as the error term) because the hypothesis term and the error term may not be independent.

Repeated Measures

The GLM (general linear model) procedure provides analysis of variance when the same measurement or measurements are made several times on each subject or case (repeated measures). The algorithms in this section apply solely to repeated measures designs.

Notation

The notation used in "GLM Algorithms" is used here. Additional conventions are defined below:

t	The number of within-subjects factors.
c	The number of measures.

r_k The number of levels of the kth within-subjects factor. $r_k \geq 2, k = 1, \ldots, t$

$\mathbf{M}_k$ The contrast matrix of the *k*th within-subjects factor, $k = 1, \ldots, t$. It is a square matrix with dimension r_k. Each element in the first column is usually equal to $1/r_k$. For a polynomial contrast each element is $1/\sqrt{r_k}$, or, for a user-specified contrast, a non-zero constant The other columns have zero column sums.

Number of Variables

It is required that $c \times \prod_{k=1}^{t} r_k = r$, the number of dependent variables in the model.

Covariance Structure

As usual in GLM, the data matrix is related to the parameter matrix **B** as $\mathbf{Y} = \mathbf{XB} + \mathbf{E}$. The rows of **E** are uncorrelated and the *i*th row has the distribution $N_r \left(0, w_i^{-1}\Sigma\right)$. Repeated measures analysis has two additional assumptions:

- $\Sigma = \Sigma_C \otimes \Sigma_1 \otimes \cdots \otimes \Sigma_t$ where Σ_C is the covariance matrix of the measures and is the Kronecker product operator.
- The Huynh and Feldt (1970) condition: Suppose $\sigma_{rs}^{(k)}$ is the (*r*,*s*)-th element of $\Sigma_k (k = 1, \ldots, t)$; then $\sigma_{rr}^{(k)} + \sigma_{ss}^{(k)} - 2\sigma_{rs}^{(k)} =$ constant for $r \neq s$. Matrices satisfying this condition result in orthonormally transformed variables with spherical covariance matrices; for this reason, the assumption is sometimes referred to as the sphericity assumption. A matrix that has the property of compound symmetry (that is, identical diagonal elements and identical off-diagonal elements) automatically satisfies this assumption.

Tests on the Between-Subjects Effects

The procedure for testing the hypothesis of no between-subjects effects uses the following steps:

1. Compute $\mathbf{M} = \mathbf{I}_c \otimes \mathbf{M}_{1;1} \otimes \cdots \otimes \mathbf{M}_{t;1}$ where $\mathbf{M}_{k;1}$ is the first column of the contrast matrix $\mathbf{M}_k$ of the *k*th within-subjects factors. Note that **M** is an *r*×*c* matrix.

2. For each of the between-subjects effects including the intercept, get the **L** matrix, according to the specified type of sum of squares.

3. Compute $\mathbf{S}_H = \left(\mathbf{L\hat{B}M}\right)' \left(\mathbf{LGL}'\right) \left(\mathbf{L\hat{B}M}\right)$ and $\mathbf{S}_E = \mathbf{M}'\mathbf{SM}$. Both are *c*×*c* matrices.

4. Compute the four multivariate test statistics: Wilks' lambda, Pillai's trace, Hotelling-Lawley trace, Roy's largest root, and the corresponding significance levels. Also compute the individual univariate *F* statistics.

5. Repeat steps 2 to 4 until all between-subjects effects have been tested.

Multivariate Tests on the Within-Subjects Effects

The procedure for testing the hypothesis of no within-subjects effects uses the following steps:

1. For the *k*th within-subjects factor, compute $\mathbf{M} = \mathbf{I}_c \otimes \mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_t$ where $\mathbf{A}_k = \mathbf{M}_{k;2:r_k}$ which is the second-to-last column of $\mathbf{M}_k$ when the *k*th within-subjects factor is involved in the effect. Otherwise, $\mathbf{A}_k = \mathbf{M}_{k;1}$. Note that **M** is an $r \times cd$ matrix, where *d* is the number of columns in the Kronecker product $\mathbf{A_1} \otimes ... \otimes \mathbf{A_t}$. In general, $d > 1$.
2. For each of the between-subjects effects, get the **L** matrix, according to the specified type of sum of squares.
3. Compute $\mathbf{S}_H = \left(\mathbf{L\hat{B}M}\right)' \left(\mathbf{LGL}'\right) \left(\mathbf{L\hat{B}M}\right)$ and $\mathbf{S}_E = \mathbf{M}'\mathbf{SM}$. Both are $cd \times cd$ matrices.
4. Compute the four multivariate test statistics: Wilks' lambda, Pillai's trace, Hotelling-Lawley trace, Roy's largest root, and the corresponding significance levels. Also compute the individual univariate *F* statistics.
5. Repeat steps 2 to 4 for the next between-subjects effect. When all the between-subjects effects are used, go to step 6.
6. Repeat steps 1 to 5 until all within-subjects effects have been tested.

Averaged Tests on the Within-Subjects Effects

The procedure for the averaged test of the hypothesis of no within-subjects effects uses the following steps:

1. Take $\mathbf{M}_k$ ($k = 1, \ldots, t$) the equally spaced polynomial contrast matrix.
2. Compute $\mathbf{M} = \mathbf{I}_c \otimes \mathbf{A}_1 \otimes \cdots \otimes \mathbf{A}_t$ where $\mathbf{A}_k = \mathbf{M}_{k;2:r_k}$ which is the 2nd to last column of $\mathbf{M}_k$ when the *k*th within-subjects factor is involved in the effect. Otherwise, $\mathbf{A}_k = \mathbf{1}_{r_k}/\sqrt{r_k}$. Note that **M** is an $r \times cd$ matrix, where *d* is the number of columns in the Kronecker product $\mathbf{A_1} \otimes ... \otimes \mathbf{A_t}$. In general, $d > 1$.
3. For each of the between-subjects effects, get the **L** matrix, according to the specified type of sum of squares.
4. Compute $\mathbf{S}_H = (\mathbf{L\hat{B}M})'(\mathbf{LGL}')(\mathbf{L\hat{B}M})$ and $\mathbf{S}_E = \mathbf{M}'\mathbf{SM}$. Both are $cd \times cd$ matrices.
5. Partition $\mathbf{S}_H$ into c^2 block matrices each of dimension $d \times d$. The (k,l)th block, denoted as $\mathbf{S}_{H;k,l}$, ($k=1,\ldots,c$ and $l=1,\ldots,c$), is a sub-matrix of $\mathbf{S}_H$ from row $(k-1)d+1$ to row kd, and from column $(l-1)d+1$ to column ld. Form the $c \times c$ matrix, denoted by $\overline{\mathbf{S}}_H$, whose (k, l)th element is the trace of $\mathbf{S}_{H;k,l}$. The matrix $\overline{\mathbf{S}}_E$ is obtained similarly.
6. Use $\overline{\mathbf{S}}_H$ and $\overline{\mathbf{S}}_E$ for computing the four multivariate test statistics: Wilks' lambda, Pillai's trace, Hotelling-Lawley trace, Roy's largest root, and the corresponding significance levels. *Note:* Set the degrees of freedom for $\overline{\mathbf{S}}_H$ (same as the row dimension of **L** in the test procedure) equal to dr_L and that for $\mathbf{S}_E$ equal to $d(n - r_X)$ in the computations. Also compute the individual univariate *F* statistics and their significance levels.

7. Repeat steps 3 to 6 for each between-subjects effect. When all the between-subjects effects are used, go to step 8.

8. Repeat steps 2 to 7 until all within-subjects effects have been tested.

Adjustments to Degrees of Freedom of the F Statistics

The adjustments to degrees of freedom of the univariate F test statistics are the Greenhouse-Geisser epsilon, the Huynh-Feldt epsilon, and the lower-bound epsilon.

For any of the three epsilons, the adjusted significance level is

$$1 - CDF.F(F, \epsilon d r_{\mathbf{L}}, \epsilon d(n - r_{\mathbf{X}}))$$

where ε is one of the three epsilons.

Greenhouse-Geisser epsilon

$$\epsilon_{GG} = \frac{(trace(\mathbf{S}_E))^2}{d \times trace\left(\mathbf{S}_E^2\right)}$$

Huynh-Feldt epsilon

$$\epsilon_{HF} = \min\left(\frac{nd\epsilon_{GG} - 2}{d(n - r_{\mathbf{X}}) - d^2\epsilon_{GG}}, 1\right)$$

Lower bound epsilon

$$\epsilon_{LB} = 1/d$$

Mauchly’s Test of Sphericity

Mauchly’s test of sphericity is displayed for every repeated measures model.

Hypotheses

In Mauchly’s test of sphericity the null hypothesis is $H_o : \mathrm{M}^{'}\Sigma\mathrm{M} = \sigma^2\mathbf{I}_m$, versus the alternative hypothesis $H_1 : \mathrm{M}^{'}\Sigma\mathrm{M} \neq \sigma^2\mathbf{I}_m$, where $\sigma^2 > 0$ is unspecified, **I** is an *m*×*m* identity matrix, and **M** is the *r*×*m* orthonormal matrix associated with a within-subjects effect. **M** is generated using equally spaced polynomial contrasts applied to the within-subjects factors (see the descriptions in “Averaged Tests on the Within-Subjects Effects”).

Mauchly’s W Statistic

$$W = \begin{cases} \frac{|\Xi|}{(trace(\Xi)/m)^m} & \text{if } trace(\Xi) > 0 \\ \text{SYSMIS} & \text{if } trace(\Xi) \leq 0 \end{cases}$$

where $\Xi = \mathbf{M}'\mathbf{A}\mathbf{M}$ and $\mathbf{A} = \left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right)'\mathbf{W}\left(\mathbf{Y} - \mathbf{X}\hat{\mathbf{B}}\right)$ is the $r \times r$ matrix of residual sums of squares and cross products.

Chi-Square Approximation

When n is large and under the null hypothesis that for $n - r_X \geq 1$ and $m \geq 2$,

$$\Pr\left(-\rho(n - r_X)\log W \leq c\right) = \Pr\left(\chi_f^2 \leq c\right) + \omega_2\left(\Pr\left(\chi_{f+4}^2 \leq c\right) - \Pr\left(\chi_f^2 \leq c\right)\right) + O\left(n^{-3}\right)$$

where

$$f = m(m+1)/2 - 1$$
$$\rho = 1 - \left(2m^2 + m + 2\right)/\left(6m(n - r_X)\right)$$
$$\omega_2 = \frac{(m+2)(m-1)(m-2)\left(2m^3+6m^2+3m+2\right)}{288m^2(n-r_X)^2\rho^2}$$

Chi-Square Statistic

$$c = \begin{cases} -\rho(n - r_X)\log W & \text{if } W > 0 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Degrees of Freedom

$$f = m(m+1)/2 - 1$$

Significance

$$1 - CDF.CHISQ(c, f) - \omega_2(CDF.CHISQ(c, f+4) - CDF.CHISQ(c, f))$$

where CDF.CHISQ is the IBM® SPSS® Statistics function for cumulative chi-square distribution. The significance will be reset to zero in case the computed value is less than zero due to floating point imprecision.

References

Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. *Regression diagnostics: Identifying influential data and sources of collinearity*. New York: John Wiley and Sons.

Clarke, M. R. B. 1982. Algorithm AS 178: The Gauss-Jordan sweep operator with detection of collinearity. *Applied Statistics*, 31:2, 166–168.

Goodnight, J. H. 1979. A tutorial on the SWEEP operator. *The American Statistician*, 33:3, 149–158.

Greenhouse, S. W., and S. Geisser. 1959. On methods in the analysis of profile data. *Psychometrika*, 24:2, 95–111.

Huynh, H., and L. S. Feldt. 1970. Conditions under which mean square ratios in repeated measurements designs have exact F-distributions. *Journal of the American Statistical Association*, 65, 1582–1589.

Huynh, H., and L. S. Feldt. 1976. Estimation of the Box correction for degrees of freedom from sample data in the randomized block and split plot designs. *Journal of Educational Statistics,* 1, 69–82.

Mauchly, J. W. 1940. Significance test for sphericity of a normal n-variate distribution. *Annals of Mathematical Statistics*, 11, 204–209.

Rao, C. R. 1951. An asymptotic expansion of the distribution of Wilks' criterion. *Bulletin of the International Statistical Institute,* 33:2, 177–180.

Rao, C. R. 1973. *Linear statistical inference and its applications*, 2nd ed. New York: John Wiley and Sons.

Ridout, M. S., and J. M. Cobby. 1989. A remark on algorithm AS 178. *Applied Statistics*, 38, 420–422.

Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2, 110–114.

Searle, S. R., F. M. Speed, and G. A. Milliken. 1980. Population marginal means in the linear model: an alternative to least squares means. *The American Statistician*, 34, 216–221.

Searle, S. R. 1982. *Matrix algebra useful for statistics*. New York: John Wiley & Sons, Inc..

GLM: Testing for Heteroscedasticity

Introduction

When fitting a linear regression model, researchers are always supposed to be aware of the assumption of homoscedasticity. When variance of the error is not constant across the observations, there would be loss in efficiency of the parameters estimated by ordinary least squares (OLS). While model parameters might still be estimated consistently by OLS, the inference

Although researchers may do graphic analysis to check the pattern of residuals by plotting them versus fitted values or predictors, the conclusion drawn from the plots may be subjective and implausible. To statistically test the homoscedasticity assumption, we may apply different statistical tests: White's test [White, 1980], the Breusch-Pagan test [T. S. Breusch, 1979], and the modified Breusch-Pagan test [Koenker, 1981, Koenker and Bassett Jr, 1982], all of which are based on the residuals of fitted linear regression models. However, SPSS Statistics 24 and its previous versions do not offer any procedures or options to conduct these tests.

To solve this problem and target SPSS Statistics 25, we provide in this document with the details on how to derive the statistics for both White's test and the (modified) Breusch-Pagan test. We derive the formulas of the test statistics from classic linear regression models and make illustrative examples on how to obtain the reference distributions. Through the following designed algorithms, we desire to offer White's test and two versions of the Breusch-Pagan test in SPSS Statistics 25.

Notations

The following notations defined in this section will be used for the subsequent sections.

Variables

n: Number of complete cases in the data set, which is an integer and $n \geq 1$.

k: Number of parameters, including the constant term (if exists), in the model. It is an integer and $k \geq 1$.

$\boldsymbol{Y}$: A vector $(n * 1)$ of a continuous dependent variable.

$\boldsymbol{X}$: A design matrix $(n * k)$, whose rows and columns represent the observations and the parameters, respectively.

$\boldsymbol{w}$: A vector $(n * 1)$ of regression weights.

$\boldsymbol{f}$: A vector $(n * 1)$ of frequency weights.

N: Number of the effective sample size. $N = \sum_{i=1}^{n} f_i$. If there is no $\boldsymbol{f}$, then $N = n$.

$\boldsymbol{\beta}$: A vector $(k * 1)$ of regression parameters to be estimated.

$\boldsymbol{\epsilon}$: A vector $(n * 1)$ of unobserved errors.

Models

We further define

- $\boldsymbol{y} = (y_1, y_2, \ldots, y_n)^T$ denoting the observed values of $\boldsymbol{Y}$;
- $\boldsymbol{\beta} = (\beta_1, \beta_2, \ldots, \beta_n)^T$, whose values are unknown;
- $\boldsymbol{\epsilon} = (\epsilon_1, \epsilon_2, \ldots, \epsilon_n)^T$, and assume $\boldsymbol{\epsilon} \sim (\boldsymbol{0}, \boldsymbol{W}^{-1/2}\boldsymbol{\Omega}\boldsymbol{W}^{-1/2})$, where $\boldsymbol{W}^{-1/2} = \text{diag}\left(1/\sqrt{w_1}, 1/\sqrt{w_2}, \cdots, 1/\sqrt{w_n}\right)$. Note that in ordinary linear models, ϵ_i's are assumed to be independent and homoscedastic with variance σ^2, and $\boldsymbol{\Omega} = \sigma^2\boldsymbol{I_n}$, and

- $\boldsymbol{X} = \begin{bmatrix} 1 & x_{21} & x_{31} & \cdots & x_{k1} \\ 1 & x_{22} & x_{32} & \cdots & x_{k2} \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_{2n} & x_{3n} & \cdots & x_{kn} \end{bmatrix}$. Particularly, the i^{th} row of $\boldsymbol{X}$ is denoted by $\boldsymbol{x_i}^T$, where $i = 1, 2, \cdots, n$.

Given the ordinary linear regression model $y_i = \beta_1 + \beta_2 x_{2i} + \beta_3 x_{3i} + \cdots + \beta_k x_{ki} + \epsilon_i$, or in the matrix form $\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\epsilon}$, we are interested in testing the homoscedasticity of $\boldsymbol{\epsilon}$, and will be presenting the test statistics in the following sections.

White's Test

Testing Hypothesis

The null hypothesis for White's test is

$$H_0 : \sigma_i^2 = \sigma_0^2 \quad \text{for all } i \ . \tag{1}$$

Since White's test does not make any assumptions about the form of the heteroscedasticity, the alternative hypothesis for testing is simply the complement of H_0, or H_1 : Not H_0. It has been argued that the generality of White's test may lead to a significant observed test statistic due to model misspecification other than heteroscedasticity [Thursby, 1989]. Thus, the null hypothesis would be rejected if any one of the following facts is violated:

- The regression errors are not of homoscedasticity;
- The regression errors are not independent of the predictors;
- The regression model is not correctly specified.

Test Statistic

Auxiliary regression

Let $\hat{\boldsymbol{\epsilon}}$ be the estimated $\boldsymbol{\epsilon}$. Consider the following regression model

$$\begin{aligned}\hat{\boldsymbol{\epsilon}}^2 = \theta_1 &+ \theta_2 \boldsymbol{x_2} + \theta_3 \boldsymbol{x_3} + \cdots + \theta_k \boldsymbol{x_k} \\ &+ \theta_{22} \boldsymbol{x_2}^2 + \theta_{23} \boldsymbol{x_2 x_3} + \cdots + \theta_{2k} \boldsymbol{x_2 x_k} \\ &+ \theta_{33} \boldsymbol{x_3}^2 + \theta_{34} \boldsymbol{x_3 x_4} + \cdots + \theta_{3k} \boldsymbol{x_3 x_k} \\ &+ \cdots \\ &+ \theta_{(k-1)(k-1)} \boldsymbol{x_{k-1}}^2 + \theta_{(k-1)k} \boldsymbol{x_{k-1} x_k} \\ &+ \theta_{kk} \boldsymbol{x_k}^2 + \boldsymbol{e} \ ,\end{aligned} \tag{2}$$

which regresses the squared estimated residuals on all levels, squares, and second order cross products of the design matrix $\boldsymbol{X}$ as well as a constant term. Note that the square of factors are not allowed. Actually, there should be no redundant terms in the regression model (2). As pointed as an example in [White, 1980], if $\boldsymbol{x_{i1}} = \boldsymbol{1}$ and $\boldsymbol{x_{i3}} = \boldsymbol{x_{i2}}^2$, it turns out that $\boldsymbol{x_{i1} x_{i3}} = \boldsymbol{x_{i2}}^2$ and thus only one term is allowed.

The observed test statistic

White's test is based on the estimated (constant-adjusted) correlation coefficient $R^2_{\hat{\boldsymbol{\epsilon}}^2}$ obtained from the regression model (2). We let $\boldsymbol{u} = (w_1 \hat{\epsilon}^2, w_2 \hat{\epsilon}^2, \cdots, w_n \hat{\epsilon}^2)^T$ denoting a vector $(n * 1)$ of squared weighted residuals, and set

$$\bar{u} = \frac{1}{n} \sum_{i=1}^{n} w_i \hat{\epsilon}_i^2 \quad \text{and} \quad \bar{\boldsymbol{u}} = (\underbrace{\bar{u}, \bar{u}, \cdots, \bar{u}}_{n})^T \ . \tag{3}$$

Thus, the observed test statistic for White's test is derived by

$$\begin{aligned} t_{\text{White}} &= n * R^2_{\hat{\boldsymbol{\epsilon}}^2} \\ &= n * \frac{(\boldsymbol{u}-\bar{\boldsymbol{u}})^T \boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{Z})^{-1}\boldsymbol{Z}^T(\boldsymbol{u}-\bar{\boldsymbol{u}})}{(\boldsymbol{u}-\bar{\boldsymbol{u}})^T(\boldsymbol{u}-\bar{\boldsymbol{u}})} \\ &= n * \frac{\boldsymbol{u}^T\boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{Z})^{-1}\boldsymbol{Z}^T\boldsymbol{u} - n\bar{u}^2}{\boldsymbol{u}^T\boldsymbol{u} - n\bar{u}^2} \ , \end{aligned} \tag{4}$$

where $\boldsymbol{Z}$ is a design matrix including the constant term and unique levels, squares, and cross products of $\boldsymbol{X}$. For instance, if

$$\boldsymbol{X} = \begin{bmatrix} 1 & 1 & x_{31} & x_{41} \\ 1 & 0 & x_{32} & x_{42} \\ 1 & 0 & x_{33} & x_{43} \\ 1 & 1 & x_{34} & x_{44} \end{bmatrix} \ , \tag{5}$$

where $\boldsymbol{x_2}$ is a dummy variable, and $\boldsymbol{x_3}$ is an independent continuous predictor, then

$$\boldsymbol{Z} = \begin{bmatrix} 1 & 1 & x_{31} & x_{41} & x_{31}^2 & x_{41}^2 & x_{31} & x_{41} & x_{31}x_{41} \\ 1 & 0 & x_{32} & x_{42} & x_{32}^2 & x_{42}^2 & 0 & 0 & x_{32}x_{42} \\ 1 & 0 & x_{33} & x_{43} & x_{33}^2 & x_{43}^2 & 0 & 0 & x_{33}x_{43} \\ 1 & 1 & x_{34} & x_{44} & x_{34}^2 & x_{44}^2 & x_{34} & x_{44} & x_{34}x_{44} \end{bmatrix} \ . \tag{6}$$

Set K_z = number of columns in $\boldsymbol{Z}$. Under the null hypothesis (1), $t_{\text{White}} \sim \chi^2_\nu$, where $\nu = K_z - 1$. For $\boldsymbol{Z}$ in Equation (6), $\nu = 9 - 1 = 8$.

Note that ν can also be verified by

$$\nu = \frac{k*(k+1)}{2} - \text{number of redundant or constant terms in } \boldsymbol{X} \ . \tag{7}$$

So for $\boldsymbol{X}$ in Equation (5), $\nu = 4*(4+1)/2 - 2 = 8$, since there are two constant terms including a dummy predictor in $\boldsymbol{X}$.

The Modified Breusch-Pagan Test

As an alternative approach to White' test, the Bresuch-Pagan test is based on the Lagrangian multiplier test [T. S. Breusch, 1979, Aitchison and Silvey, 1960].

Testing Hypothesis

The null hypothesis for the Breusch-Pagan test is

$$H_0 : \sigma_i^2 = \sigma^2 h(\alpha_0 + \boldsymbol{\alpha}\boldsymbol{z_i}) \text{ and } \boldsymbol{\alpha} = \boldsymbol{0} \ , \tag{8}$$

where $h(\cdot)$ is a function not indexed by i; σ_i^2 is the error variance for the i^{th} observation; and α_0 and $\boldsymbol{\alpha}$ are regression coefficients. Note that the Breusch-Pagan test assumes that the error terms are normally distributed. It tests the null hypothesis of homoscedasticity versus the alternative hypothesis that the error terms have a variance varying with the predictors.

Test Statistic

The observed test statistic for the Breusch-Pagan test is

$$t_{\text{BP}} = \frac{1}{2}\left[\boldsymbol{g}^T\boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{Z})^{-1}\boldsymbol{Z}^T\boldsymbol{g}\right] \ , \tag{9}$$

where

$$\boldsymbol{g} = \hat{\boldsymbol{\epsilon}}^2 \circ \frac{n}{\hat{\epsilon}^T\hat{\epsilon}} - \boldsymbol{1} \ , \tag{10}$$

where "$\circ$" denotes the Hadamard product, or element-wise multiplication, in matrix manipulation.

For an implementation purpose and in presence of weights, Equation (9) is equivalent to

$$t_{\text{BP}} = \frac{\boldsymbol{u}^T \boldsymbol{Z}(\boldsymbol{Z}^T \boldsymbol{Z})^{-1} \boldsymbol{Z}^T \boldsymbol{u} - n\bar{u}^2}{2\bar{u}^2} \ , \tag{11}$$

where all the variables are defined in the same way as in Equation (4). Under the null hypothesis (8), $t_{\text{BP}} \sim \chi^2_\nu$, where $\nu = K_z - 1$.

As aforementioned, the Breusch-Pagan test assumes that the residuals are normally distributed. A modified test was suggested by [Koenker, 1981] and [Koenker and Bassett Jr, 1982] based on a more robust estimator of the variance of $\boldsymbol{\epsilon}^2$. The observed test statistic for the modified Breusch-Pagan test is

$$t_{\text{MBP}} = n * \frac{\boldsymbol{u}^T \boldsymbol{Z}(\boldsymbol{Z}^T \boldsymbol{Z})^{-1} \boldsymbol{Z}^T \boldsymbol{u} - n\bar{u}^2}{\boldsymbol{u}^T \boldsymbol{u} - n\bar{u}^2} \ , \tag{12}$$

which shares the same form with White's test. Similarly, under the null hypothesis (8), $t_{\text{MBP}} \sim \chi^2_\nu$, where $\nu = K_z - 1$.

A Few Remarks

- The reference distribution is χ^2_ν , which is the same for White's test (Equation (4)), Breusch-Pagan test (Equation (11)), and the modified Breusch-Pagan test (Equation (12)). The degree of freedom $\nu = K_z - 1$ is determined by the number of columns in $\boldsymbol{Z}$.

- The difference between White's test and the modified Breusch-Pagan test lies in the regressors contained in $\boldsymbol{Z}$. In White's test, $\boldsymbol{Z}$ contains the regressors of all levels, squares, and cross products of those regressors in $\boldsymbol{X}$, or in the original regression model. For the modified Breusch-Pagan test, $\boldsymbol{Z}$ contains a set of user-specified regressors. If $\boldsymbol{Z}$ is the same for the two tests, then $t_{\text{W}} = t_{\text{MBP}}$. From this perspective, White's test is a special case of the modified Breusch-Pagan test.

- The modified Breusch-Pagan test releases the assumption of normality.

- White's test is general, since it makes no assumptions about the form of the heteroscedasticity. For the (modified) Breusch-Pagan test, the alternative hypothesis is that the variance of $\boldsymbol{\epsilon}$ varies with a set of regressors, not necessarily the design matrix $\boldsymbol{X}$ in the original model.

$\boldsymbol{F}$-Test

Testing Hypothesis

Wooldridge once proposed an F-statistic that did not require the normality assumption [Wooldridge, 2015]. Reconsider the regression model (2), and let $\boldsymbol{Z} = \boldsymbol{X}$, or

$$\hat{\boldsymbol{\epsilon}}^2 = \theta_1 + \theta_2 \boldsymbol{x_2} + \theta_3 \boldsymbol{x_3} + \cdots + \theta_k \boldsymbol{x_k} + \boldsymbol{e} \ . \tag{13}$$

If the homoscedasticity assumption holds, we would expect $\theta_2 = \theta_3 = \cdots = \theta_k = 0$. Actually, it is equivalent to test whether there is an overall significance of the regression model.

Test Statistic

The observed F-test statistic for the regression model (13) is

$$t_F = \left(\frac{R^2_{\hat{\boldsymbol{\epsilon}}^2}}{k-1} \right) / \left(\frac{1 - R^2_{\hat{\boldsymbol{\epsilon}}^2}}{n-k} \right) \ , \tag{14}$$

where $R^2_{\hat{\boldsymbol{\epsilon}}^2}$ is defined by Equation (4), and n and k are defined in Section . Under the null hypothesis that the homoscedasticity holds, $t_F \sim F_{k-1,n-k}$.

Testing by Using Predicted Values

To look at these proposed tests from a different perspective, a new idea occurs that we can regress square errors on (high-order) fitted values. Consider a second-order auxiliary regression model using the fitted $\hat{\boldsymbol{y}}$ as the regressors

$$\hat{\boldsymbol{\epsilon}}^2 = \theta_0 + \theta_1 \hat{\boldsymbol{y}} + \theta_2 \hat{\boldsymbol{y}}^2 + \boldsymbol{e} \ , \tag{15}$$

where we are interested in testing $H_0 : \theta_1 = \theta_2 = 0$.

Similar to the Breusch-Pagan test, the modified Breusch-Pagan test, and F-test, we can construct t_{BP}, t_{MBP}, and f as we have done in Equations (11), (12), and (14), respectively. The equations remain the same, but instead use the fitted values to construct the design matrix $\boldsymbol{Z}$.

Two Special Scenarios

When $\boldsymbol{Z}$ Contains Only Constant Term & Predicted Values

In this section, we talk about one special scenario in which the auxiliary regression model only contains $\hat{\boldsymbol{y}}$ and a constant term. Consider Equation 16

$$\hat{\boldsymbol{\epsilon}}^2 = \theta_0 + \theta_1 \hat{\boldsymbol{y}} + \boldsymbol{e} \tag{16}$$

the auxiliary regression model by default, if uses specifies no regressors when testing for heteroscedasticity. In this scenario, the design matrix becomes

$$\boldsymbol{Z} = \begin{bmatrix} 1 & \hat{y}_{21} \\ 1 & \hat{y}_{22} \\ 1 & \hat{y}_{23} \\ 1 & \hat{y}_{24} \end{bmatrix} \text{, which has only two columns.} \tag{17}$$

We can use this $\boldsymbol{Z}$ matrix to construct the Breusch-Pagan, the Modified Breusch-Pagan, and the F test statistics aforementioned. Under the null hypothesis, we have

$$t_{\mathrm{BP}} = \frac{\boldsymbol{u}^T \boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{Z})^{-1}\boldsymbol{Z}^T\boldsymbol{u} - n\bar{u}^2}{2\bar{u}^2} \sim \chi^2_1 \ , \tag{18}$$

$$t_{\mathrm{MBP}} = n * \frac{\boldsymbol{u}^T \boldsymbol{Z}(\boldsymbol{Z}^T\boldsymbol{Z})^{-1}\boldsymbol{Z}^T\boldsymbol{u} - n\bar{u}^2}{\boldsymbol{u}^T\boldsymbol{u} - n\bar{u}^2} \sim \chi^2_1 \ , \tag{19}$$

and

$$t_F = \frac{(n-2)R^2_{\hat{\boldsymbol{\epsilon}}^2}}{1 - R^2_{\hat{\boldsymbol{\epsilon}}^2}} \sim F_{1,n-2} \ , \tag{20}$$

since $K_z = 2$ for all of the statistics.

When $\boldsymbol{Z} = \boldsymbol{X}$

In this section, we talk about the other special scenario in which the auxiliary and the original regression model share the same regressors, or $\boldsymbol{Z} = \boldsymbol{X}$. Then we can just replace $\boldsymbol{Z}$ with $\boldsymbol{X}$ in the previously derived test statistics. Under the null hypothesis, we have

$$t_{\mathrm{BP}} = \frac{\boldsymbol{u}^T \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{u} - n\bar{u}^2}{2\bar{u}^2} \sim \chi^2_\nu \ , \tag{21}$$

$$t_{\mathrm{MBP}} = n * \frac{\boldsymbol{u}^T \boldsymbol{X}(\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{u} - n\bar{u}^2}{\boldsymbol{u}^T\boldsymbol{u} - n\bar{u}^2} \sim \chi^2_\nu \ , \tag{22}$$

and

$$t_F = \left(\frac{R^2_{\hat{\boldsymbol{\epsilon}}^2}}{k-1}\right) \Big/ \left(\frac{1 - R^2_{\hat{\boldsymbol{\epsilon}}^2}}{n-k}\right) \sim F_{1,n-2} \ . \tag{23}$$

References

[Aitchison and Silvey, 1960] Aitchison, J. and Silvey, S. D. (1960). Maximum-likelihood estimation procedures and associated tests of significance. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 154–171.

[Koenker, 1981] Koenker, R. (1981). A note on studentizing a test for heteroscedasticity. *Journal of Econometrics*, 17(1):107–112.

[Koenker and Bassett Jr, 1982] Koenker, R. and Bassett Jr, G. (1982). Robust tests for heteroscedasticity based on regression quantiles. *Econometrica: Journal of the Econometric Society*, pages 43–61.

[T. S. Breusch, 1979] T. S. Breusch, A. R. P. (1979). A simple test for heteroscedasticity and random coefficient variation. *Econometrica*, 47(5):1287–1294.

[Thursby, 1989] Thursby, J. G. (1989). A comparison of several specification error tests for a general alternative. *International Economic Review*, 30(1):217–230.

[White, 1980] White, H. (1980). A heteroskedasticity-consistent covariance matrix estimator and a direct test for heteroskedasticity. *Econometrica: Journal of the Econometric Society*, pages 817–838.

[Wooldridge, 2015] Wooldridge, J. (2015). *Introductory econometrics: A modern approach.* Nelson Education.

GLM/UNIANOVA: Robust Standard Errors

Introduction

The GLM and UNIANOVA procedures for fitting the general linear model offer ordinary least squares (OLS) estimation and weighted least squares (WLS) estimation using a user-supplied known weighting variable. The standard assumption of homoscedastic or homogeneous errors is commonly violated in unknown ways, rendering these methods inefficient. So-called robust or heteroscedasticity-consistent (HC) estimators of the covariance matrix (and therefore of the standard errors) of the parameter estimates are a popular approach to dealing with this problem, and users of SPSS Statistics have been requesting inclusion of such methods for some time. These enhancements offer a set of HC covariance matrix estimation options.

Notation

The following notation is used throughout the document unless otherwise stated:

n	Number of distinct records in the dataset. It is an integer and $n \geq 1$.
p	Number of parameters (including parameters for dummy variables but excluding the intercept). It is an integer and $p \geq 0$.
p^*	Number of non-redundant parameters (excluding intercept if it exists). It is an integer and $0 \leq p^* \leq p$.
$\mathbf{y}$	$n \times 1$ vector of single dependent variable consists of y_i .
f	$n \times 1$ vector of frequency count variable. If an element is not an integer, it is computed by rounding the value to the nearest integer. If it is less than 0.5 or if it is missing, the corresponding case is not used.
g	$n \times 1$ vector of regression weight. If there is no regression weight specified, $g = 1$. If regression weight g_i for case *i* is zero, negative or missing, the corresponding case is not used.
N	Effective sample size. it is a integer number, $N = \sum_{i=1}^{n} f_i$ .If frequency count variable *f* is not used, *N*=*n*.
$\boldsymbol{X}$	$n \times (p+1)$ design matrix. The rows represent the cases and the columns represent the parameters. The *i*th row is $\boldsymbol{x}_i = (x_{i0}, \ldots, x_{ip})$, $i = 1, 2, \ldots, n$, with $x_{i0} = 1$, The *j*th column is

	$\boldsymbol{X}_j = (x_{1j},...,x_{nj})^{\mathrm{T}}$, , $j = 0,1,...,p$, with $\boldsymbol{X}_0 = (1,...,1)^{\mathrm{T}}$. . If there is no intercept, $\boldsymbol{X} = \{\boldsymbol{X}_j\}_{j=1}^{p}$ is a $n \times p$ matrix.
$\boldsymbol{G}$	Symmetric generalized inverse of $X^T WX$, $\left(X^T WX\right)^{-}$
ε	$n \times 1$ vector of unobserved errors .
β	$(p+1) \times 1$ vector of unknown parameters. $\beta = (\beta_0, \beta_1, \cdots \beta_p)$. β_0 is the intercept, if it exists. If there is no intercept, $\beta = (\beta_1, \cdots \beta_p)^T$ is a $p \times 1$ vector.
$\hat{\beta}$	$(p+1) \times 1$ vector of estimated β . $\hat{\beta} = (\hat{\beta}_0, \hat{\beta}_1, \cdots \hat{\beta}_p)$. If there is an intercept, $\hat{\beta}_0$ is its estimate, else $\hat{\beta} = (\hat{\beta}_1, \cdots \hat{\beta}_p)^T$ is a $p \times 1$ vector.
$\hat{\mathbf{y}}$	Predicted value of $\mathbf{y}$, consists of $\hat{y}_i$
e	$n \times 1$ vector of residuals , $\mathbf{y} - \hat{\mathbf{y}}$.
h	$n \times 1$ vector of leverages

Model

The standard general linear model of variable $\boldsymbol{y}$ on the design matrix $\boldsymbol{X}$ has the form

$$\boldsymbol{y} = \boldsymbol{X\beta} + \boldsymbol{\varepsilon} \tag{1}$$

where $\boldsymbol{\varepsilon}$ follows a normal distribution with mean $\mathbf{0}$ and variance $\sigma^2 \boldsymbol{D}^{-1}$, i.e., $\boldsymbol{\varepsilon} \sim N_n\left(\mathbf{0},\ \sigma^2 \boldsymbol{D}^{-1}\right)$ with $\boldsymbol{D}^{-1} = \operatorname{diag}\left(1/g_1, \ldots, 1/g_n\right)$. Then the dependent variable $\boldsymbol{y}$ also follows a normal distribution with mean $\boldsymbol{X\beta}$ and variance $\sigma^2 \boldsymbol{D}^{-1}$, $\boldsymbol{y} \sim N_n\left(\boldsymbol{X\beta},\ \sigma^2 \boldsymbol{D}^{-1}\right)$.

Notes:

1. The elements of $\boldsymbol{\varepsilon}$ are independent with each other, so are those of $\boldsymbol{y}$.
2. $\boldsymbol{X}$ can be any combination of continuous and categorical effects and interaction effects, though in many cases only continuous covariates will be involved. See Lam (1995a) for further details on the parameterization of the design matrix $\boldsymbol{X}$.

Least Squares Coefficient Estimation

The coefficients are be estimated by the least squares (LS) method with the following closed form solution

$$\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{X}\right)^{-} \boldsymbol{X}^{\mathrm{T}}\boldsymbol{W}\boldsymbol{y}, \tag{2}$$

where $\boldsymbol{W} = \mathrm{diag}\left(w_1,\ldots,w_n\right) = \mathrm{diag}\left(g_1 f_1,\ldots,g_n f_n\right)$.

The actual computation of $\hat{\boldsymbol{\beta}}$ is done by applying sweep operations instead of applying equation (2). See Lam (1995b) for details.

Robust Covariance Matrix and Standard Errors

Homoscedasticity assumption

The homoscedasticity assumption is that variance of the error (σ^2) is constant across all cases. When the assumption is violated, the OLS coefficient estimates are still consistent, but not efficient. So for valid inference, according to Huber (1967) or White (1980), a heteroscedastic consistent (HC) or robust estimator of covariance matrix of the estimated coefficient should be used.

Robust Estimation of the Covariance Matrix of the Estimated Parameters

A robust estimator of the covariance matrix of the estimated model parameters is:

$$\hat{\boldsymbol{\Psi}} = \boldsymbol{G}\boldsymbol{X}'\boldsymbol{W}^{1/2}\hat{\boldsymbol{\Omega}}\boldsymbol{W}^{1/2}\boldsymbol{X}\boldsymbol{G}, \tag{3}$$

where $\hat{\boldsymbol{\Omega}}$ is a diagonal matrix of variance estimates of weighted residuals, $\hat{\boldsymbol{\Omega}} = diag(\omega_1,\ldots,\omega_n)$,

and there are 5 estimators differ in their choice of the ω_i:

HC0: $$\omega_i = u_i = g_i e_i^2 \tag{4}$$

HC1: $$\omega_i = \frac{N}{\left(N - rank(X)\right)} u_i \qquad (5)$$

HC2: $$\omega_i = \frac{1}{1-h_i} u_i \qquad (6)$$

HC3: $$\omega_i = \frac{1}{\left(1-h_i\right)^2} u_i \qquad (7)$$

HC4: $$\omega_i = \frac{1}{\left(1-h_i\right)^{\delta_i}} u_i \qquad (8)$$

where $\delta_i = \min\left(4, \frac{Nh_i}{rank(X)}\right)$

and h_i is the i^{th} diagonal element of $XGX^T = x_i G x_i^T$.

Notes:

- The estimator HC0 is introduced by White (1980), is justified by asymptotic arguments.
- The estimator HC1 – HC3 are suggested by MacKinnon and White (1985) to improve the performance in small samples and Long and Ervin (2000) concluded that HC3 provided the best performance in sample samples based on Monte Carlo simulation.
- The estimator HC4 was introduced by Cribari-Neto (2004) and was compared with earlier estimators via simulations and bootstrap tests and performed better than the other estimators.

Affected statistics

Many statistics computed previously would be affected by replacing the original or model-based covariance matrix $\hat{\boldsymbol{\Psi}} = s^2 \boldsymbol{G}$ with the robust estimator $\hat{\boldsymbol{\Psi}} = \boldsymbol{G}X'W^{1/2}\hat{\boldsymbol{\Omega}}W^{1/2}X\boldsymbol{G}$ (assume the (*i*, *j*) element in $\hat{\boldsymbol{\Psi}}$ is $\psi_{i,j}$) and they are listed according to areas:

- **Statistics related to coefficient estimates**:

  $\hat{\sigma}_{\hat{\beta}_j} = \psi_{j+1,j+1}$, $j = 0, \ldots, p$ (note that $\hat{\boldsymbol{\Psi}}$ includes intercept term if there is one); then t-statistics, *p*-values and confidence intervals should be updated as well.

- **Statistics related to tests of individual effects**:

When the robust estimator is used, F-statistics cannot be computed based on sums of squares any more. For each effect *j*, the F-statistic should be computed as

$$F_j = \frac{\hat{\boldsymbol{\beta}}^{\mathrm{T}} \boldsymbol{L}_j^{\mathrm{T}} \left(\boldsymbol{L}_j \hat{\boldsymbol{\Psi}} \boldsymbol{L}_j^{\mathrm{T}} \right)^{-1} \boldsymbol{L}_j \hat{\boldsymbol{\beta}}}{r_j} \qquad (9)$$

where r_j is the rank of $\boldsymbol{L_j}$.

References

[1]. Cribari-Neto, F. (2004). Asymptotic Inference Under Heteroskedasticity of Unknown Form. *Computational Statistics & Data Analysis, 45,* 215-233.

[2]. Lam, M. L. (1995a), “Constructing the Design Matrix for the β-Model,” *SPSS Internal Document*.

[3]. Lam, M. L. (1995b), “Algorithm: the symmetric sweep operator,” *SPSS Internal Document*.

[4]. Long, J. S., Irvin, L. (2000), Using Heteroscedasticity Consistent Standard Errors in the Linear Regression Model. *The American Statistician, 54(3), 217-224*.

[5]. MacKinnon, J. G., & White, H. (1985). Some Heteroskedasticity Consistent Covariance Matrix Estimators with Improved Finite Sample Properties. *Journal of Econometrics, 29,* 53-57.

[6]. White, H. (1980). A Heteroskedastic-Consistent Covariance Matrix Estimator and a Direct Test of Heteroskedasticity. *Econometrica, 48,* 817-838.

HILOGLINEAR Algorithms

`HILOGLINEAR` fits hierarchical loglinear models to multidimensional contingency tables using an iterative proportional-fitting algorithm.

The Model Minimum Configuration

Consider an $I \times J \times K$ table. Let n_{ijk} be the observed frequency and m_{ijk} the expected frequency for cell *(i, j, k)*. A simple way to construct a saturated linear model in the natural logarithms of the expected cell frequencies is by analogy with analysis of variance (ANOVA) models:

$$L_{ijk} \equiv \log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{ij}^{AB} + u_{ik}^{AC} + u_{jk}^{BC} + u_{ijk}^{ABC}$$

where $1 \le i \le I, 1 \le j \le J$, and $1 \le k \le K$. In general, each of the seven subscripted *u*-terms sums to zero over each lettered subscript.

It can be shown (Bishop, Feinberg, and Holland, 1975), p. 65, that, under the commonly encountered sampling plans, the log-likelihood is

$$\Phi + Nu + \sum_i n_{i++}u_i^A + \sum_j n_{+j+}u_j^B + \sum_k n_{++k}u_k^C + \sum_{i,j} n_{ij+}u_{ij}^{AB} + \sum_{i,k} n_{i+k}u_{ik}^{AC} + \sum_{j,k} n_{+jk}u_{jk}^{BC} + \sum_{i,j,k} n_{ijk}u_{ijk}^{ABC}$$

where Φ is independent of any parameters and *N* is total number of observations. Also, the *n*-terms adjacent to the unknown parameters are the sufficient statistics. The formulation of the above log likelihood is based on the saturated model. When we consider unsaturated models, terms drop out and those that remain give the sufficient statistics. For instance, if we assume that there is no three-factor effect, that is, $u_{ijk}^{ABC} = 0$ for all *i*, *j*, and *k*, or more briefly $u_{ijk} = 0$, then

$$\log(m_{ijk}) = u + u_i^A + u_j^B + u_k^C + u_{ij}^{AB} + u_{ik}^{AC} + u_{jk}^{BC}$$

and $n_{i++}, n_{+j+}, n_{++k}, n_{ij+}, n_{i+k}$ and n_{+jk} are the sufficient statistics for this reduced model. These statistics can be considered as tables of sums configurations and denoted by *C* with proper subscripts. For example, $\{n_{ij+}\}$ is the configuration C_{12} and $\{n_{++k}\}$ is the configuration C_3. Note that $\{n_{i++}\}, \{n_{+j+}\}$, and $\{n_{++k}\}$ can be obtained from $\{n_{ij+}\}, \{n_{i+k}\}$ and $\{n_{+jk}\}$. We then call the last three configurations C_{12}, C_{13} and C_{23}**minimal configurations** or **minimal statistics**.

Notation for Unlimited Number of Dimensions

To generalize results, we denote the complete set of subscripts by a single symbol θ. Thus, n_θ is the observed frequency in an elementary cell and w_θ is the cell weight. We add a subscript to θ to denote a reduced dimensionality so that $n_{\theta i}$ is the observed sum in a cell of $C_{\theta i}$. We use the second subscript, *i*, solely to distinguish between different configurations.

Iterative Proportional Fitting Procedure (IPFP)

We can obtain MLEs for the elementary cells under any hierarchical model by iterative fitting of the minimal sufficient configurations. To illustrate the algorithm, we consider the unsaturated model. The MLEs must fit the configurations C_{12}, C_{13} and C_{23}. The basic IPFP chooses an initial table $m_{ijk}^{(0)}$ and then sequentially adjusts the preliminary estimates to fit C_{12}, C_{13} and C_{23}. Fitting to C_{12} gives

$$\hat{m}_{ijk}^{(1)} = \hat{m}_{ijk}^{(0)} \frac{n_{ij+}}{\hat{m}_{ij+}^{(0)}}$$

Subsequent fitting to C_{13} gives

$$\hat{m}_{ijk}^{(2)} = \hat{m}_{ijk}^{(1)} \frac{n_{i+k}}{\hat{m}_{i+k}^{(1)}}$$

and similarly, after fitting C_{23} we have

$$\hat{m}_{ijk}^{(3)} = \hat{m}_{ijk}^{(2)} \frac{n_{+jk}}{\hat{m}_{+jk}^{(2)}}$$

We repeat this three-step cycle until convergence to the desired accuracy is attained. The extension of the above procedure to the general procedure for fitting *s* configurations is straightforward. Let the minimal configurations be $C_{\theta i}$ for *i*=1,...,*s*, with cell entries $n_{\theta i}$, respectively. The procedure is as follows:

Initial Cell Estimates

To start the iterations, if CWEIGHT is not specified, set

$$m_\theta^{(0)} = \begin{cases} 1 & \text{if } n_\theta \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

If CWEIGHT is specified, set

$$m_\theta^{(0)} = \begin{cases} 1 & \text{if } CWEIGHT \geq 1 \\ 0 & \text{if } CWEIGHT \leq 0 \\ CWEIGHT & \text{if } 0<CWEIGHT<1 \end{cases}$$

Intermediate Computations

After obtaining the initial cell estimates, the algorithm proceeds to fit each of these configurations in turn. After *r* cycles, the relations are

$$\hat{m}_\theta^{(sr+i)} = \hat{m}_\theta^{(sr+i-1)} \frac{n_{\theta_i}}{\hat{m}_{\theta_i}^{(sr+i-1)}} \quad \text{for } 1 \leq i \leq s; r \geq 0$$

Convergence Criteria

The computations stop either when a complete cycle, which consists of *s* steps, does not cause any cell to change by more than a preset amount ϵ; that is,

$$\left|\hat{m}_\theta^{(sr)} - \hat{m}_\theta^{(sr-s)}\right| < \epsilon \quad \text{for all } \theta$$

or the number of cycles is larger than a preset integer *max*. Both ϵ and *max* can be specified. The default for ϵ is

$$\epsilon = \max_\theta \left\{0.25; \frac{n_\theta w_\theta}{1000}\right\}$$

and the default for *max* is 20.

Goodness of Fit Tests

The Pearson chi-square statistic is

$$\chi^2 = \sum_\theta \frac{\left(n_\theta - \hat{m}_\theta\right)^2}{\hat{m}_\theta}$$

and the likelihood-ratio chi-square statistic is

$$L^2 = 2\sum_\theta n_\theta \ln\left(n_\theta / \hat{m}_\theta\right)$$

where the first summation is done over the cells with nonzero estimated cell frequencies while the second summation is done over cells with positive observed and estimated cell frequencies. The degrees of freedom for the above two statistics are computed as follows:

Adjusted Degrees of Freedom

Let T_c be the total number of the cells and *P* the number of parameters in the model. Also, let z_c be the number of cells such that $\hat{m}_\theta = 0$. The adjusted degrees of freedom is

adjusted df $= T_c - P - z_c$

Unadjusted Degrees of Freedom

unadjusted df $= T_c - P$

Parameter Estimates and Standard Errors

If a saturated model is fitted and neither $n_\theta + \delta$ nor w_θ is equal to zero for all cells, then the parameter estimates and their standard errors will be computed. Each estimate of the parameters in the saturated model can be expressed as a linear combination of the logarithms of the observed cell frequencies plus user-specified δ, where the coefficients used in the linear combination add to zero. We discuss the rule of obtaining the coefficients. Consider, in general case, a $J_1 \times J_2 \times \ldots \times J_M$ frequency table with defining variables $X_1, \ldots, X_M$. Let $u_{j_{s1},\ldots,j_{sL}}^{X_{s1},\ldots,X_{sL}}$ denote an *L*-term interaction involving. $X_{s1}, \ldots, X_{sL}$ at level $j_{s1}, \ldots, j_{sL}$ respectively. Denote *A* as a vector that is constructed in the way that its nonzero components correspond to the variables in the parameter to be estimated and are set to the level of the variable. Let $C_{j_1,\ldots,j_M}$ be a *M*-dim vector with components equal to cell IDs. That is,

$$C_{j_1,\ldots,j_M} = (j_1, \ldots, j_M); \quad 1 \le j_1 \le J_1, \quad 1 \le i \le M$$

The coefficient $\beta_{j_1,\ldots,j_M}$ is determined through the comparison of the components of *A* and $C_{j_1,\ldots,j_M}$. Let *s* be the number of nonzero components of *A* that do not match (equal) the corresponding components of $C_{j_1,\ldots,j_M}$. Also, let matching occur at component $i_1, \ldots, i_k$. Then the coefficient for cell $(j_1, \ldots, j_M)$ is

$$\beta_{j_1,\ldots,j_M} = (-1)^s (J_{i_1} - 1) \times \ldots \times (J_{i_k} - 1)$$

The estimate $\hat{u}^{X_{s1},\ldots,X_{sL}}_{j_{s1},\ldots,j_{sL}}$ of $u^{X_{s1},\ldots,X_{sL}}_{j_{s1},\ldots,j_{sL}}$ is then

$$\hat{u}^{X_{s1},\ldots,X_{sL}}_{j_{s1},\ldots,j_{sL}} = \sum_{j_1,\ldots,j_M} \beta_{j_1,\ldots,j_M} \ln(n_{j_1,\ldots,j_M} + \delta)$$

The large-sample variance of the estimate is

$$\sum_{j_1,\ldots,j_M} \beta^2 [\ln(n_{j_1,\ldots,j_M} + \delta)]^{-1}$$

For a large sample, the estimate approximately follows a normal distribution with the above mean and variance if the sampling model follows a Poisson, multinomial, or product-multinomial distribution. The confidence interval for the parameter can be computed based on the asymptotic normality.

Residuals

The following residuals are computed.

Raw Residuals

raw residual $= n_\theta - \hat{m}_\theta$

Standardized Residuals

standardized residual $= (n_\theta - \hat{m}_\theta)/\sqrt{\hat{m}_\theta}$

where $\hat{m}_\theta$ must be greater than 0.

Partial Associations and Partial Chi-squares

Partial associations of effects can be requested when a saturated model is specified. Let $\chi^2(k)$ be the chi-square for the model that contains the effects up to and including the *k*-interaction terms. The test of the significance of the *k*th-order interaction can be based on

$$\chi^2(k-1) - \chi^2(k)$$

Degrees of freedom are obtained by subtracting the degrees of freedom for the corresponding models.

Model Selection Via Backward Elimination

The selection process starts with the model specified (either via DESIGN or MAXORDER subcommand). The partial chi-square is calculated for every term in the generating class. Any term with zero partial chi-square is deleted, then the effect with the largest observed significance level for the change in chi-square is deleted, provided the significance level is larger than 0.05, the default. With the removal of a highest-order term, a new model with new generating class is generated. The above process of removing a term is repeated for the new model and is continued until no remaining terms in the model can be deleted.

References

Bishop, Y. M., S. E. Feinberg, and P. W. Holland. 1975. *Discrete multivariate analysis: Theory and practice*. Cambridge, Mass.: MIT Press.

Fienberg, S. E. 1994. *The Analysis of Cross-Classified Categorical Data*, 2nd ed. Cambridge, MA: MIT Press.

HOMALS Algorithms

The iterative HOMALS algorithm is a modernized version of Guttman (1941). The treatment of missing values, described below, is based on setting weights in the loss function equal to zero, and was first described in De Leeuw and Van Rijckevorsel (1980). Other possibilities do exist and can be accomplished by recoding the data (Gifi, 1981; Meulman, 1982).

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of cases (objects)
m	Number of variables
p	Number of dimensions

For variable *j*; $j = 1, \ldots, m$

h_j	n-vector with categorical observations
k_j	Number of valid categories (distinct values) of variable j
$\mathbf{G}_j$	Indicator matrix for variable j, of order $n \times k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i \text{ th object is in the } r \text{th category of variable } j \\ 0 & \text{when the } i \text{th object is not in the } r \text{th category of variable } j \end{cases}$$

$\mathbf{D}_j$	Diagonal matrix, containing the univariate marginals; that is, the column sums of $\mathbf{G}_j$
$\mathbf{M}_j$	Binary diagonal $n \times n$ matrix, with diagonal elements defined as

$$m_{(j)ii} = \begin{cases} 1 & \text{when the } i\text{th observation is within the range } [1, k_j] \\ 0 & \text{when the } i\text{th observation outside the range } [1, k_j] \end{cases}$$

The quantification matrices and parameter vectors are:

$\boldsymbol{X}$	Object scores, of order $n \times p$
$\mathbf{Y}_j$	Category quantifications, of order $k_j \times p$.
$\mathbf{Y}$	Concatenated category quantification matrices, of order $\Sigma_j k_j \times p$

Note: The matrices $\mathbf{G}_j$, $\mathbf{M}_j$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Objective Function Optimization

The HOMALS objective is to find object scores $\mathbf{X}$ and a set of $\mathbf{Y}_j$ (for $j = 1, \ldots, m$) so that the function

$$\sigma(\mathbf{X};\mathbf{Y}) = 1/m\Sigma_j \mathrm{tr}\Big((\mathbf{X}-\mathbf{G}_j\mathbf{Y}_j)^{'}\mathbf{M}_j(\mathbf{X}-\mathbf{G}_j\mathbf{Y}_j)\Big)$$

is minimal, under the normalization restriction $\mathbf{X}^{'}\mathbf{M}_*\mathbf{X} = mn\mathbf{I}$, where the matrix $\mathbf{M}_* = \Sigma_j\mathbf{M}_j$, and **I** is the *p*×*p* identity matrix. The inclusion of $\mathbf{M}_j$ in $\sigma(\mathbf{X};\mathbf{Y})$ ensures that there is no influence of data values outside the range $[1, k_j]$, which may be really missing or merely regarded as such; $\mathbf{M}_*$ contains the number of "active" data values for each object. The object scores are also centered; that is, they satisfy $\mathbf{u}^{'}\mathbf{M}_*\mathbf{X} = 0$, with **u** denoting an *n*-vector with ones.

Optimization is achieved through the following iteration scheme:

1. Initialization
2. Update object scores
3. Orthonormalization
4. Update category quantifications
5. Convergence test: repeat steps 2-4 or continue
6. Rotation

These steps are explained below.

Initialization

The object scores **X** are initialized with random numbers, which are normalized so that $\mathbf{u}^{'}\mathbf{M}_*\mathbf{X} = 0$ and $\mathbf{X}^{'}\mathbf{M}_*\mathbf{X} = mn\mathbf{I}$, yielding $\tilde{\mathbf{X}}$. Then the first category quantifications are obtained as $\tilde{\mathbf{Y}}_j = \mathbf{D}_j^{-1}\mathbf{G}^{'}{}_j\tilde{\mathbf{X}}$.

Update object scores

First the auxiliary score matrix **Z** is computed as

$$\mathbf{Z} \leftarrow \Sigma_j\mathbf{M}_j\mathbf{G}_j\tilde{\mathbf{Y}}_j$$

and centered with respect to $\mathbf{M}_*$:

$$\tilde{\mathbf{Z}} \leftarrow \Big\{\mathbf{M}_* - \Big(\mathbf{M}_*\mathbf{u}\mathbf{u}^{'}\mathbf{M}_*/\mathbf{u}^{'}\mathbf{M}_*\mathbf{u}\Big)\Big\}\mathbf{Z}.$$

These two steps yield locally the best updates when there are no orthogonality constraints.

Orthonormalization

The orthonormalization problem is to find an $\mathbf{M}_*$-orthonormal $\mathbf{X}^+$ that is closest to $\tilde{\mathbf{Z}}$ in the least squares sense. In HOMALS, this is done by setting

$$\mathbf{X}^+ \leftarrow m^{1/2}\mathbf{M}_*^{-1/2}GRAM\Big(\mathbf{M}_*^{-1/2}\tilde{\mathbf{Z}}\Big)$$

which is equal to the genuine least squares estimate up to a rotation. The notation GRAM( ) is used to denote the Gram-Schmidt transformation (Björk and Golub, 1973).

Update category quantifications

For *j*=1,...,*m*, the new category quantifications are computed as:

$$\mathbf{Y}_j^+ = \mathbf{D}_j^{-1}\mathbf{G}'_j\tilde{\mathbf{X}}$$

Convergence test

The difference between consecutive loss function values $\sigma\left(\tilde{\mathbf{X}};\tilde{\mathbf{Y}}\right)-\sigma(\mathbf{X}^+;\mathbf{Y}^+)$ is compared with the user-specified convergence criterion ε —a small positive number. Steps 2 to 4 are repeated as long as the loss difference exceeds ε.

Rotation

As indicated in step 3, during iteration the orientation of **X** and **Y** with respect to the coordinate system is not necessarily correct; this also reflects that $\sigma(\mathbf{X};\mathbf{Y})$ is invariant under simultaneous rotations of **X** and **Y**. From theory it is known that solutions in different dimensionality should be nested; that is, the *p*-dimensional solution should be equal to the first *p* columns of the (*p*+1)-dimensional solution. Nestedness is achieved by computing the eigenvectors of the matrix $1/m\Sigma_j\mathbf{Y}'_j\mathbf{D}_j\mathbf{Y}_j$. The corresponding eigenvalues are printed after the convergence message of the program. The calculation involves tridiagonalization with Householder transformations followed by the implicit QL algorithm (Wilkinson, 1965).

Diagnostics

The following diagnostics are available.

Maximum Rank (may be issued as a warning when exceeded)

The maximum rank $p_{\max}$ indicates the maximum number of dimensions that can be computed for any dataset. In general:

$$p_{\max} = \min\{(n-1),((\Sigma_j k_j)-\max(m_1,1))\}$$

where m_1 is the number of variables with no missing values. Although the number of nontrivial dimensions may be less than $p_{\max}$ when *m*=2, HOMALS does allow dimensionalities all the way up to $p_{\max}$.

Marginal Frequencies

The frequencies table gives the univariate marginals and the number of missing values (that is, values that are regarded as out of range for the current analysis) for each variable. These are computed as the column sums of $\mathbf{D}_j$ and the total sum of $\mathbf{M}_j$.

Discrimination Measure

These are the dimensionwise variances of the quantified variables. For variable *j* and dimension *s*:

$$\eta^2_{js} = \mathbf{y}'_{(j)s}\mathbf{D}_j\mathbf{y}_{(j)s}/n$$

where $\mathbf{y}_{(j)s}$ is the *s*th column of $\mathbf{Y}_j$, corresponding to the *s*th quantified variable $\mathbf{G}_j\mathbf{y}_{(j)s}$.

Eigenvalues

The computation of the eigenvalues that are reported after convergence is discussed in step 6. With the HISTORY option, the sum of the eigenvalues is reported during iteration under the heading "total fit." Due to the fact that the sum of the eigenvalues is equal to the trace of the original matrix, the sum can be computed as $1/m\Sigma_j\Sigma_s\eta^2_{js}$. The value of $\sigma(\mathbf{X};\mathbf{Y})$ is equal to $p - 1/m\Sigma_j\Sigma_s\eta^2_{js}$.

References

Björk, A., and G. H. Golub. 1973. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27, 579–594.

De Leeuw, J., and J. Van Rijckevorsel. 1980. HOMALS and PRINCALS—Some generalizations of principal components analysis. In: *Data Analysis and Informatics*, E. Diday,et al., ed. Amsterdam: North-Holland, 231–242.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Guttman, L. 1941. The quantification of a class of attributes: A theory and method of scale construction. In: *The Prediction of Personal Adjustment*, P. Horst, ed. New York: Social Science Research Council, 319–348.

Meulman, J. J. 1982. *Homogeneity analysis of incomplete data*. Leiden: DSWO Press.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem*. Oxford: Clarendon Press.

KM Algorithms

This procedure estimates the survival function for time to occurrence of an event. Some of the times may be “censored” in that the event does not occur during the observation period, or contact is lost with participants (loss to follow-up).

If the subjects are divided into treatment groups, KM produces a survival function for each treatment group (factor level) and a test of equality of the survival functions across treatment groups. The survival functions across treatment groups can also be compared while controlling for categories of a stratification variable.

Notation

The following notation is used throughout this chapter unless otherwise stated:

p	Number of levels (strata) for the stratification variable
g	Number of levels (treatment groups) for the factor variable

Estimation and SE for Survival Distribution

Suppose that for a given combination of the stratification and factor variables, a random sample of *n* individuals yields a sample with *k* distinct observed failure times (uncensored). Let $t_1 < \ldots < t_k$ represent the observed life times and T_L be the largest observation in the sample. (Note that $T_L = t_k$ if the largest observation is uncensored.) Define

$n_i =$ Number of subjects who are at risk at time t_i
$d_i =$ Number of failures (deaths) at t_i
$\lambda_i =$ Number of censorings in interval [t_i, t_{i+1})

Note that

$$n_0 = n$$
$$n_{i+1} = n_i - d_i - \lambda_i,\ i = 0, 1, \ldots, k-1$$
$$t_0 = 0$$
$$t_{k+1} = \infty$$
$$d_0 = 0$$
$$\lambda_0 = 0$$

The Kaplan-Meier estimate $\hat{S}(t)$ for the survival function is computed as

$$\hat{S}(t) = \prod_{t_i < t} \left(1 - \frac{d_i}{n_i}\right)$$

Note that

$$\hat{S}(t_l^+) = \prod_{i=1}^{l}\left(1-\frac{d_i}{n_i}\right),\ l = 1, 2, \ldots, k.$$

$\hat{S}(t_0^+) = 1$

$\hat{S}(t_{l+1}^+) = \hat{S}(t_l^+)\left(1-\frac{d_{l+1}}{n_{l+1}}\right)$

$\hat{S}(t_k^+) = 0$ if $n_k = d_k$ $(T_l = t_k$ and $\lambda_k = 0)$, otherwise

$$\hat{S}(t_k^+) = \prod_{l=1}^{k}\left(1-\frac{d_l}{n_l}\right),\ T_L \geq t \geq t_k$$

$\hat{S}(t_1^+), \ldots, \hat{S}(t_k^+)$ are the survival functions shown in the table.

The asymptotic standard error for $\hat{S}(t_l^+)$ is computed as the square root of

$$var\left(\hat{S}(t_l^+)\right) = \left[\hat{S}(t_l^+)\right]^2 \sum_{i=1}^{l} \frac{d_i}{n_i(n_i-d_i)}, \quad l = 1, \ldots, k.$$

Note: When $n_k = d_k(T_L = t_k$ and $\lambda_k = 0)$, $\hat{S}(t_k^+) = 0$ and $var\left(\hat{S}(t_k^+)\right) = 0$.

Estimation of Mean Survival Time and Standard Error

$$\hat{\mu} = \begin{cases} \sum_{i=0}^{k-1} \hat{S}(t_i^+)(t_{i+1}-t_i) & \text{if } T_L = t_k \\ \sum_{i=0}^{k-1} \hat{S}(t_i^+)(t_{i+1}-t_i) + \hat{S}(t_k^+)(T_L - t_k) & \text{otherwise} \end{cases}$$

The variance of the mean survival time is

$$var(\hat{\mu}) = \sum_{i=1}^{k} \frac{a_i^2 d_i}{n_i(n_i-d_i)}$$

$$a_i = \sum_{l=i}^{k-1} \hat{S}(t_l^+)(t_{l+1}-t_l) + \hat{S}(t_k^+)(T_L - t_k)$$

$$d = \sum_{i=1}^{k} d_i$$

unless there are both censored and uncensored occurrences of the largest survival time. In that case,

$$var(\hat{\mu}) = \frac{d}{d-1} \sum_{i=1}^{k-1} \frac{a_i^2 d_i}{n_i(n_i-d_i)}$$

$$a_i = \sum_{l=i}^{k-1} \hat{S}(t_l^+)(t_{l+1}-t_l)$$

The standard error is the square root of the variance.

Plots

The following plots are available.

Survival Functions versus Time

The survival function $\hat{S}(t)$ is plotted against *t*.

Log Survival Functions versus Time

$\ln\left(\hat{S}(t)\right)$ is plotted against *t*.

Cumulative Hazard Functions versus Time

$-\ln\left(\hat{S}(t)\right)$ is plotted against *t*.

Estimation of Percentiles and Standard Error

100*p* percentile of the survival time, where *p* is between 0 and 1, is computed as

$$t_p = inf\left\{t_i | \left(\hat{S}(t_i) \le p\right)\right\}$$

The asymptotic variance of t_p is estimated by

$$var(t_p) = \frac{var\left(\hat{S}(t_p)\right)}{\left(\hat{f}(t_p)\right)^2}$$

where $\hat{f}(t_p)$ is computed as

$$\hat{f}(t_p) = \frac{\hat{S}(u_{p+0.05}) - \hat{S}(t_{p-0.05})}{t_{p-0.05} - u_{p+0.05}}$$

where $u_q = sup\left\{t_i | \left(\hat{S}(t_i) \ge q\right)\right\}$.

Testing the Equality of the Survival Functions

Three statistics are computed to test the equality of survival distributions in the presence of arbitrary right censorship. These statistics are the logrank (Mantel-Cox), the modified Wilcoxon test statistic (Breslow), and an alternative test statistic proposed by Tarone and Ware (1977). Using the regression model proposed by Cox (1972), all three test statistics have been modified for testing monotonic trend in hazard functions.

Test Statistics

Let $n^{(s)}$ be the number of subjects in stratum *s*. Let

$$t_1^{(s)} < \ldots < t_{m_s}^{(s)}$$

be the observed failure times (responses) and

$n_{li}^{(s)}$ = in stratum *s* the number of individuals in group *l* at risk just prior to $t_i^{(s)}$

$d_{li}^{(s)}$ = number of deaths at $t_i^{(s)}$ in group *l*

and

$$d_i^{(s)} = \sum_{l=1}^{g} d_{li}^{(s)}$$

$$n_i^{(s)} = \sum_{l=1}^{g} n_{li}^{(s)}$$

Hence, the expected number of events in group *l* at time $t_i^{(s)}$ is given by

$$E_{li}^{(s)} = \frac{d_i^{(s)} n_{li}^{(s)}}{n_i^{(s)}}$$

Define

$$U_s = \left(U_1^{(s)}, \ldots, U_{g-1}^{(s)}\right)'$$

with

$$U_l^{(s)} = \sum_{i=1}^{m_s} w_i^{(s)} \left(d_{li}^{(s)} - E_{li}^{(s)}\right) \quad \text{for } l = 1, \ldots, g-1$$

Also, let V_s be a $(g-1) \times (g-1)$ covariance matrix with

$$V_{jl}^{(s)} = \sum_{i=1}^{m_s} \left(w_i^{(s)}\right)^2 \frac{d_i^{(s)}\left(n_i^{(s)} - d_i^{(s)}\right)}{n_i^{(s)} - 1} \frac{n_{ji}^{(s)}}{n_i^{(s)}} \left(\delta_{jl} - \frac{n_{li}^{(s)}}{n_i^{(s)}}\right) \quad \text{for } j, l = 1, \ldots, g-1$$

where

$w_i^{(s)} = 1$ for log-rank test

$w_i^{(s)} = n_i^{(s)}$ for Breslow test

$w_i^{(s)} = \sqrt{n_i^{(s)}}$ for Tarone Ware test

and

$$\delta_{jl} = \begin{cases} 1 & \text{if } j = l \\ 0 & \text{otherwise} \end{cases}$$

Define

$$U = \sum_{s=1}^{p} U_s$$

and

$$V = \sum_{s=1}^{p} V_s$$

The test statistic for the equality of the g survival functions is defined by

$$\chi^2 = U^{'} V^{-1} U$$

χ^2 has an asymptotic chi-square distribution with $g-1$ degrees of freedom.

Test Statistic for Trend

Let

$$t = (t_1, \ldots, t_g)^{'}$$

be a vector with $t_j =$ trend weighting coefficient for group j. Form the vector

$$U_{(s)} = \left(U_1^{(s)}, \ldots, U_g^{(s)}\right)^{'}$$

$U_{(s)}$ differs from U_s only in the last component.
Let $V^{(s)}$ be a $g \times g$ matrix with element $V_{lj}^{(s)}$ for $1 \le l, j \le g$. The test statistic is defined by

$$\chi_t^2 = \frac{\left(t^{'} U\right)^2}{t^{'} V t}$$

where

$$U = \sum_{s=1}^{p} U_{(s)}$$

$$V = \sum_{s=1}^{p} V_{(s)}$$

The logrank, Breslow, and Tarone Ware tests may involve trend. Each of the test statistics has a chi-square distribution with one degree of freedom.

The default trend is defined as follows:

$$t = \begin{cases} (-(g-1), \ldots, -3, -1, 1, 3, \ldots, (g-1)) & \text{if } g \text{ is even} \\ \left(-\frac{(g-1)}{2}, \ldots, \quad 1, 0, 1, \cdots, \frac{(g-1)}{2}\right) & \text{otherwise} \end{cases}$$

References

Cox, D. R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 34, 187–220.

Tarone, R. 1975. Tests for trend in life table analysis. *Biometrika*, 62, 679–682.

Tarone, R., and J. Ware. 1977. On distribution free tests for equality of survival distributions. *Biometrika*, 64, 156–160.

KNN Algorithms

Nearest Neighbor Analysis is a method for classifying cases based on their similarity to other cases. In machine learning, it was developed as a way to recognize patterns of data without requiring an exact match to any stored patterns, or cases. Similar cases are near each other and dissimilar cases are distant from each other. Thus, the distance between two cases is a measure of their dissimilarity.

Cases that are near each other are said to be "neighbors." When a new case (holdout) is presented, its distance from each of the cases in the model is computed. The classifications of the most similar cases – the nearest neighbors – are tallied and the new case is placed into the category that contains the greatest number of nearest neighbors.

You can specify the number of nearest neighbors to examine; this value is called *k*. The pictures show how a new case would be classified using two different values of *k*. When *k* = 5, the new case is placed in category *1* because a majority of the nearest neighbors belong to category *1*. However, when *k* = 9, the new case is placed in category *0* because a majority of the nearest neighbors belong to category *0*.

Nearest neighbor analysis can also be used to compute values for a continuous target. In this situation, the average or median target value of the nearest neighbors is used to obtain the predicted value for the new case.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$\mathbf{Y}$	Optional $1 \times N$ vector of responses with element y_n, where $n=1,...,N$ indexes the cases.
$\mathbf{X}^0$	$P^0 \times N$ matrix of features with element x^0_{pn}, where $p=1,...,P^0$ indexes the features and $n=1,...,N$ indexes the cases.
$\mathbf{X}$	$P \times N$ matrix of encoded features with element x_{pn}, where $p=1,...,P$ indexes the features and $n=1,...,N$ indexes the cases.
P	Dimensionality of the feature space; the number of continuous features plus the number of categories across all categorical features.
N	Total number of cases.
$N_j, j = 1, 2, \cdots, J$	The number of cases with $Y = j$, where Y is a response variable with J categories
$\hat{N}_j$	The number of cases which belong to class j and are correctly classified as j.
$\hat{N}^*_j$	The total number of cases which are classified as j.

Preprocessing

Features are coded to account for differences in measurement scale.

Continuous

Continuous features are optionally coded using adjusted normalization:

$$x_{pn} = \frac{2\left(x_{pn}^0 - \min\left(x_p^0\right)\right)}{\max\left(x_p^0\right) - \min\left(x_p^0\right)} - 1$$

where x_{pn} is the normalized value of input feature *p* for case *n*, x_p^0 is the original value of the feature for case *n*, $\min\left(x_p^0\right)$ is the minimum value of the feature for all training cases, and $\max\left(x_p^0\right)$ is the maximum value for all training cases.

Categorical

Categorical features are always temporarily recoded using one-of-*c* coding. If a feature has *c* categories, then it is is stored as *c* vectors, with the first category denoted (1,0,...,0), the next category (0,1,0,...,0), ..., and the final category (0,0,...,0,1).

Training

Training a nearest neighbor model involves computing the distances between cases based upon their values in the feature set. The nearest neighbors to a given case have the smallest distances from that case. The distance metric, choice of number of nearest neighbors, and choice of the feature set have the following options.

Distance Metric

We use one of the following metrics to measure the similarity of query cases and their nearest neighbors.

Euclidean Distance. The distance between two cases is the square root of the sum, over all dimensions, of the weighted squared differences between the values for the cases.

$$Euclidean_{ih} = \sqrt{\sum_{p=1}^{P} w_{(p)}\left(x_{(p)i} - x_{(p)h}\right)^2}$$

City Block Distance. The distance between two cases is the sum, over all dimensions, of the weighted absolute differences between the values for the cases.

$$CityBlock_{ih} = \sum_{p=1}^{P} w_{(p)}\left|x_{(p)i} - x_{(p)h}\right|$$

The feature weight $w(p)$ is equal to 1 when feature importance is not used to weight distances; otherwise, it is equal to the normalized feature importance:

$$w_{(p)} = FI_{(p)} / \sum_{p=1}^{P} FI_{(p)}$$

See “Output Statistics ” for the computation of feature importance $FI_{(p)}$.

Crossvalidation for Selection of k

Cross validation is used for automatic selection of the number of nearest neighbors, between a minimum $k_{\min}$ and maximum $k_{\max}$. Suppose that the training set has a cross validation variable with the integer values 1,2,..., *V*. Then the cross validation algorithm is as follows:

- For each $k \in$,$[k_{\min}, k_{\max}]$ compute the average error rate or sum-of square error of *k*: $CV_k = \sum_{v=1}^{V} e_v / V$, where e_v is the error rate or sum-of square error when we apply the Nearest Neighbor model to make predictions on the cases with $X = v$; that is, when we use the other cases as the training dataset.
- Select the optimal *k* as: $\hat{k} = arg\{\min CV_k : k_{\min} \leq k \leq k_{\max}\}$.

Note: If multiple values of *k* are tied on the lowest average error, we select the smallest *k* among those that are tied.

Feature Selection

Feature selection is based on the wrapper approach of Cunningham and Delany (2007) and uses forward selection which starts from J_{Forced} features which are entered into the model. Further features are chosen sequentially; the chosen feature at each step is the one that causes the largest decrease in the error rate or sum-of squares error.

Let S_J represent the set of *J* features that are currently chosen to be included, S_J^c represents the set of remaining features and e_J represents the error rate or sum-of-squares error associated with the model based on S_J .

The algorithm is as follows:

- Start with $J = J_{Forced}$ features.
- For each feature in S_J^c , fit the *k* nearest neighbor model with this feature plus the existing features in S_J and calculate the error rate or sum-of square error for each model. The feature in S_J^c whose model has the smallest error rate or sum-of square error is the one to be added to create S_{J+1}.
- Check the selected stopping criterion. If satisfied, stop and report the chosen feature subset. Otherwise, *J*=*J*+1 and go back to the previous step.

Note: the set of encoded features associated with a categorical predictor are considered and added together as a set for the purpose of feature selection.

Stopping Criteria

One of two stopping criteria can be applied to the feature selection algorithm.

Fixed number of features. The algorithm adds a fixed number of features, J_{add}, in addition to those forced into the model. The final feature subset will have $J_{add} + J_{Forced}$ features. J_{add} may be user-specified or computed automatically; if computed automatically the value is

$$J_{add} = \max\left\{\min\left(20, P^0\right) - J_{Forced},\ \ 0\right\}$$

When this is the stopping criterion, the feature selection algorithm stops when J_{add} features have been added to the model; that is, when $J_{add} = J + 1$, stop and report S_{J+1} as the chosen feature subset.

Note: if $J_{add} = 0$, no features are added and S_J with $J = J_{Forced}$ is reported as the chosen feature subset.

Change in error rate or sum of squares error. The algorithm stops when the change in the absolute error ratio indicates that the model cannot be further improved by adding more features. Specifically, if $e_{J+1} = 0$ or $e_J \geq e_{J+1}$ and

$$\frac{|e_J - e_{J+1}|}{e_J} \leq \Delta_{\min}$$

where $\Delta_{\min}$ is the specified minimum change, stop and report S_{J+1} as the chosen feature subset.

If $e_J < e_{J+1}$ and

$$\frac{|e_J - e_{J+1}|}{e_J} > 2\Delta_{\min}$$

stop and report S_J as the chosen feature subset.

Note: if $e_J = 0$ for $J = J_{Forced}$, no features are added and S_J with $J = J_{Forced}$ is reported as the chosen feature subset.

Combined k and Feature Selection

The following method is used for combined neighbors and features selection.

1. For each *k*, use the forward selection method for feature selection.
2. Select the *k*, and accompanying feature set, with the lowest error rate or the lowest sum-of-squares error.

Output Statistics

The following statistics are available.

Percent correct for class j

$$\frac{\hat{N}_j}{N_j} \times 100\%$$

Overall percent for class j

$$\frac{\hat{N}_j^*}{N} \times 100\%$$

Intersection of Overall percent and percent correct

$$\left(\sum_{j=1}^{J} \hat{N}_j / N \right) \times 100\%$$

Error rate of classification

$$\left(1 - \sum_{j=1}^{J} \hat{N}_j / N \right) \times 100\%$$

Sum-of-Square Error for continuous response

$$\sum_{n=1}^{N} (y_n - \hat{y}_n)^2$$

where $\hat{y}_n$ is the estimated value of y_n.

Feature Importance

Suppose there are $X_{(1)}, X_{(2)} \cdots X_{(m)} \left(1 \le m \le P^0\right)$ in the model from the forward selection process with the error rate or sum-of-squares error *e*. The importance of feature $X_{(p)}$ in the model is computed by the following method.

- Delete the feature $X_{(p)}$ from the model, make predictions and evaluate the error rate or sum-of-squares error$e_{(p)}$ based on features $X_{(1)}, X_{(2)} \cdots X_{(p-1)}, X_{(p+1)}, \cdots, X_{(m)}$.
- Compute the error ratio$e_{(p)} + \frac{1}{m}$.

The feature importance of $X_{(p)}$ is $FI_{(p)} = e_{(p)} + \frac{1}{m}$

Scoring

After we find the *k* nearest neighbors of a case, we can classify it or predict its response value.

Categorical response

Classify each case by majority vote of its *k* nearest neighbors among the training cases.

- If multiple categories are tied on the highest predicted probability, then the tie should be broken by choosing the category with largest number of cases in training set.
- If multiple categories are tied on the largest number of cases in the training set, then choose the category with the smallest data value among the tied categories. In this case, categories are assumed to be in the ascending sort or lexical order of the data values.

We can also compute the predicted probability of each category. Suppose k_j is the number of cases of the *j*th category among the *k* nearest neighbors. Instead of simply estimating the predicted probability for the *j*th category by $\frac{k_j}{k}$, we apply a Laplace correction as follows:

$$\frac{k_j+1}{k+J}$$

where *J* is the number of categories in the training data set.

The effect of the Laplace correction is to shrink the probability estimates towards to 1/*J* when the number of nearest neighbors is small. In addition, if a query case has *k* nearest neighbors with the same response value, the probability estimates are less than 1 and larger than 0, instead of 1 or 0.

Continuous response

Predict each case using the mean or median function.

Mean function.

Median function. Suppose that $y_m, m \in Nearest\left(n\right)$ are the values of the continuous response variable, and we arrange $y_m, m \in Nearest\left(n\right)$ from the lowest value to the highest value and denote them as $y_{(j_1)} \le y_{(m_2)} \le \cdots \le y_{(m_k)}$, then the median is

$$\hat{y}_n = \begin{cases} y_{\left(\frac{k+1}{2}\right)} & k \text{ is odd} \\ \frac{y_{\left(\frac{k}{2}\right)}+y_{\left(\frac{k}{2}\right)+1}}{2} & k \text{ is even} \end{cases}$$

References

Arya, S., and D. M. Mount. 1993. Algorithms for fast vector quantization. In: *Proceedings of the Data Compression Conference 1993*, , 381–390.

Cunningham, P., and S. J. Delaney. 2007. k-Nearest Neighbor Classifiers. *Technical Report UCD-CSI-2007-4, School of Computer Science and Informatics, University College Dublin, Ireland*, , – .

Friedman, J. H., J. L. Bentley, and R. A. Finkel. 1977. An algorithm for finding best matches in logarithm expected time. *ACM Transactions on Mathematical Software*, 3, 209–226.

Linear modeling algorithms

Linear models predict a continuous target based on linear relationships between the target and one or more predictors.

For algorithms on enhancing model accuracy, enhancing model stability, or working with very large datasets, see “Ensembles Algorithms”.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of distinct records in the dataset. It is an integer and $n \geq 1$.
p	Number of parameters (including parameters for dummy variables but excluding the intercept) in the model. It is an integer and $p \geq 0$.
p^*	Number of non-redundant parameters (excluding the intercept) currently in the model. It is an integer and $0 \leq p^* \leq p$.
p^c	Number of non-redundant parameters currently in the model. $p^c = p^* + 1$
p^e	Number of effects excluding the intercept. It is an integer and $0 \leq p^e \leq p$
$\mathbf{y}$	$n \times 1$ target vector with elements y_i.
f	$n \times 1$ frequency weight vector.
g	$n \times 1$ regression weight vector.
N	Effective sample size. It is an integer and $N = \sum_{i=1}^{n} f_i$. If there is no frequency weight vector, *N*=*n*.
$\mathbf{X}$	$n \times (p+1)$ design matrix with element x_{ij}. The rows represent the records and the columns represent the parameters.
ϵ	$n \times 1$ vector of unobserved errors.
β	$(p+1) \times 1$ vector of unknown parameters; $\beta = (\beta_0, \beta_1, \cdots \beta_p)$. β_0 is the intercept.
$\hat{\beta}$	$(p+1) \times 1$ vector of parameter estimates.
b	$(p+1) \times 1$ vector of standardized parameter estimates. It is the result of a sweep operation on matrix **R**. b_0 is the standardized estimate of the intercept and is equal to 0.
$\hat{\mathbf{y}}$	$n \times 1$ vector of predicted target values.
$\overline{X}_j$	Weighted sample mean for X_j, $j = 1, 2, \cdots p$
$\overline{y}$	Weighted sample mean for **y**.
S_{ij}	Weighted sample covariance between X_i and X_j, $i, j = 1, 2, \cdots p$.
S_{iy}	Weighted sample covariance between X_i and **y**.
S_{yy}	Weighted sample variance for **y**.
$\mathbf{R}$	$(p+1) \times (p+1)$ weighted sample correlation matrix for **X** (excluding the intercept, if it exists) and **y**.
$\tilde{\mathbf{R}}$	The resulting matrix after a sweep operation whose elements are $\tilde{r}_{ij}$.

Model

Linear regression has the form

$$y = X\beta + \varepsilon$$

where ε follows a normal distribution with mean 0 and variance $\sigma^2 D^{-1}$, where $D^{-1} = diag(1/g_1, \ldots, 1/g_n)$. The elements of ε are independent with respect to each other.

Notes:

- **X** can be any combination of continuous and categorical effects.
- Constant columns in the design matrix are not used in model building.
- If *n*=1 or the target is constant, no model is built.

Missing values

Records with missing values are deleted listwise.

Least squares estimation

The coefficients are estimated by the least squares (LS) method. First, we transform the model by pre-multiplying $D^{1/2}$ as follows:

$$D^{1/2}y = D^{1/2}X\beta + D^{1/2}\varepsilon$$

so that the new unobserved error $D^{1/2}\varepsilon$ follows a normal distribution $N_n(0, \sigma^2\mathbf{I})$, where **I** is an identity matrix and $D^{1/2} = diag(\sqrt{g_1}, \ldots, \sqrt{g_n})$. Then the least squares estimates of β can be obtained from the following formula

$$\hat{\beta} = arg \min_{\beta} \left(D^{1/2}y - D^{1/2}\mathbf{X}\beta\right)^T F\left(D^{1/2}\mathbf{y} - D^{1/2}\mathbf{X}\beta\right)$$

where $F = diag(f_1, \ldots, f_n)$. Note that

$$\begin{aligned}\left(D^{1/2}\mathbf{y} - D^{1/2}\mathbf{X}\beta\right)^T F\left(D^{1/2}\mathbf{y} - D^{1/2}\mathbf{X}\beta\right) \\ = (\mathbf{y} - \mathbf{X}\beta)^T D^{1/2} F D^{1/2}(\mathbf{y} - \mathbf{X}\beta) \\ = (\mathbf{y} - \mathbf{X}\beta)^T W(\mathbf{y} - \mathbf{X}\beta)\end{aligned}$$

where $W = diag(w_1, \ldots, w_n) = diag(g_1 f_1, \ldots, g_n f_n)$, so the closed form solution of $\hat{\beta}$ is

$$\hat{\beta} = \left(X^T W X\right)^- X^T W y$$

$\hat{\beta}$ is computed by applying sweep operations instead of the equation above. In addition, sweep operations are applied to the transformed scale of **X** and **y** to achieve numerical stability. Specifically, we construct the weighted sample correlation matrix **R** then apply sweep operations to it. The **R** matrix is constructed as follows.

First, compute weighted sample means, variances and covariances among **X**$_i$, **X**$_j$, $i, j = 1, \ldots, p$, and **y** :

Weighted sample means of **X**$_i$ and **y** are $\overline{X}_i = \frac{1}{\sum_{k=1}^{n} w_k} \sum_{k=1}^{n} w_k x_{ki}$ and $\overline{y} = \frac{1}{\sum_{k=1}^{n} w_k} \sum_{k=1}^{n} w_k y_k$;

Weighted sample covariance for **X**$_i$ and **X**$_j$ is $S_{ij} = \frac{1}{N-1} \sum_{k=1}^{n} w_k \left(x_{ki} - \overline{X}_i\right)\left(x_{kj} - \overline{X}_j\right)$;

Weighted sample covariance for **X**$_i$ and **y** is $S_{iy} = \frac{1}{N-1} \sum_{k=1}^{n} w_k \left(x_{ki} - \overline{X}_i\right)\left(y_k - \overline{y}\right)$;

Weighted sample variance for **y** is $S_{yy} = \frac{1}{N-1} \sum_{k=1}^{n} w_k (y_k - \overline{y})^2$.

Second, compute weighted sample correlations $r_{ij} = \frac{S_{ij}}{\sqrt{S_{ii} S_{jj}}}, i, j = 1, ..., p$ and y.

Then the matrix **R** is

$$R = \begin{bmatrix} r_{11} & r_{12} & \cdots & r_{1p} & r_{1y} \\ r_{21} & r_{22} & \cdots & r_{2p} & r_{2y} \\ \vdots & \vdots & \ddots & \vdots & \\ r_{p1} & r_{2p} & \cdots & r_{pp} & r_{py} \\ r_{y1} & r_{y2} & \cdots & r_{yp} & r_{yy} \end{bmatrix} = \begin{bmatrix} \mathrm{R}_{11} & \mathrm{R}_{12} \\ \mathrm{R}_{12}^{\mathrm{T}} & R_{22} \end{bmatrix}$$

If the sweep operations are repeatedly applied to each row of $\mathbf{R}_{11}$, where $\mathbf{R}_{11}$ contains the predictors in the model at the current step, the result is

$$\tilde{\mathrm{R}} = \begin{bmatrix} \mathbf{R}_{11}^{-1} & \mathbf{R}_{11}^{-1}\mathbf{R}_{12} \\ -\mathbf{R}_{12}^{\mathrm{T}}\mathbf{R}_{11}^{-1} & R_{22} - \mathbf{R}_{12}^{\mathrm{T}}\mathbf{R}_{11}^{-1}\mathbf{R}_{12} \end{bmatrix}$$

The last column $\mathrm{R}_{11}^{-1}\mathrm{R}_{12}$ contains the standardized coefficient estimates; that is, $\mathbf{b} = \mathbf{R}_{11}^{-1}\mathbf{R}_{12}$. Then the coefficient estimates, except the intercept estimate if there is an intercept in the model, are:

$$\hat{\beta}_j = b_j \sqrt{\frac{S_{yy}}{S_{jj}}}$$

Model selection

The following model selection methods are supported:

- None, in which no selection method is used and effects are force entered into the model. For this method, the singularity tolerance is set to 1e−12 during the sweep operation.

- Forward stepwise, which starts with no effects in the model and adds and removes effects one step at a time until no more can be added or removed according to the stepwise criteria.
- Best subsets, which checks "all possible" models, or at least a larger subset of the possible models than forward stepwise, to choose the best according to the best subsets criterion.

Forward stepwise

The basic idea of the forward stepwise method is to add effects one at a time as long as these additions are worthy. After an effect has been added, all effects in the current model are checked to see if any of them should be removed. Then the process continues until a stopping criterion is met. The traditional criterion for effect entry and removal is based on their *F*-statistics and corresponding *p*-values, which are compared with some specified entry and removal significance levels; however, these statistics may not actually follow an *F* distribution so the results might be questionable. Hence the following additional criteria for effect entry and removal are offered:

- Maximum adjusted R^2;
- Minimum corrected Akaike information criterion (AICC); and
- Minimum average squared error (ASE) over the overfit prevention data

Candidate statistics

Some additional notations are needed describe the addition or removal of a continuous effect X_j or categorical effect $\{X_{j_s}\}_{s=1}^{\ell}$, where ℓ is the number of categories.

ℓ^*	The number of non-redundant parameters of the eligible effect $\mathbf{X}_j$ or $\{X_{j_s}\}_{s=1}^{\ell}$.
p^c	The number of non-redundant parameters in the current model (including the intercept).
p^r	The number of non-redundant parameters in the resulting model (including the intercept). Note that $p^r = \begin{cases} p^c + \ell^* \text{ for entering an effect} \\ p^c - \ell^* \text{ for removing an effect} \end{cases}$
SSe_p	The weighted residual sum of squares for the current model.
$SSe_{p+\ell}$	The weighted residual sum of squares for the resulting model after entering the effect.
$SSe_{p-\ell}$	The weighted residual sum of squares for the resulting model after removing the effect.
r_{yy}	The last diagonal element in the current $\mathbf{R}$ matrix.
$\tilde{r}_{yy}$	The last diagonal element in the resulting $\tilde{\mathbf{R}}$ matrix.

F statistics. The *F* statistics for entering or removing an effect from the current model are:

$$F_{enter_j} = \frac{\left(SSe_p - SSe_{p+\ell}\right)/\ell^*}{SSe_{p+\ell}/\left(N - p^r\right)} = \frac{\left(r_{yy} - \tilde{r}_{yy}\right)\left(N - p^r\right)}{\tilde{r}_{yy} \times \ell^*}$$

$$F_{remove_j} = \frac{\left(SSe_{p-\ell} - SSe_p\right)/\ell^*}{SSe_p/(N - p^c)} = \frac{(\tilde{r}_{yy} - r_{yy})(N - p^c)}{r_{yy} \times \ell^*}$$

and their corresponding *p*-values are:

$$p_{enter_j} = P\left(F_{\ell^*,N-p^r} \geq F_{enter_j}\right) = 1 - P\left(F_{\ell^*,N-p^r} \leq F_{enter_j}\right)$$

$$p_{remove_j} = P\left(F_{\ell^*,N-p^c} \geq F_{remove_j}\right) = 1 - P\left(F_{\ell^*,N-p^c} \leq F_{remove_j}\right)$$

Adjusted R-squared. The adjusted R^2 value for entering or removing an effect from the current model is:

$$\text{adj.}R^2 = 1 - \frac{(N-1)\tilde{r}_{yy}}{N - p^r}$$

Corrected Akaike Information Criterion (AICC). The AICC value for entering or removing an effect from the current model is:

$$AICC = N \ln\left(\frac{(N-1)\,S_{yy} \times \tilde{r}_{yy}}{N}\right) + \frac{2p^r N}{N - p^r - 1}$$

Average Squared Error (ASE). The ASE value for entering or removing an effect from the current model is:

$$ASE = \frac{1}{\sum_{t=1}^{T} f_t} \sum_{t=1}^{T} w_t (y_t - \hat{y}_t)^2$$

where $\hat{y}_t = {}_t\hat{\beta}$ are the predicted values of y_t and T is the number of distinct testing cases in the overfit prevention set.

The Selection Process

There are slight variations in the selection process, depending upon the model selection criterion:

- The *F* statistic criterion is to select an effect for entry (removal) with the minimum (maximum) *p*-value and continue doing it until the *p*-values of all candidates for entry (removal) are equal to or greater than (less than) a specified significance level.
- The other three criteria are to compare the statistic (adjusted R^2, AICC or ASE) of the resulting model after entering (removing) an effect with that of the current model. Selection stops at a local optimal value (a maximum for the adjusted R^2 criterion and a minimum for the AICC and ASE).

The following additional definitions are needed for the selection process:

FLAG	A $p^e \times 1$ index vector which records the status of each effect. $FLAG_i$ = 1 means the effect *i* is in the current model, $FLAG_i$ = 0 means it is not. $\lvert\{i \mid FLAG_i = 1\}\rvert$ denotes the number of effects with $FLAG_i$ = 1.
MAXSTEP	The maximum number of iteration steps. The default value is $3 \times p^e$.
MAXEFFECT	The maximum number of effects (excluding intercept if exists). The default value is p^e.

P_{in}	The significance level for effect entry when the *F*-statistic criterion is used. The default is 0.05.
P_{out}	The significance level for effect removal when the *F* statistic criterion is used. The default is 0.1.
ΔF	The *F* statistic change. It is F_{enter_j} or F_{remove_j} for entering or removing an effect X_j (here X_j could represent continuous or categorical for simpler notation).
$p_{\Delta F}$	The corresponding *p*-value for ΔF.
MSC_{current}	The adjusted R^2, AICC, or ASE value for the current model.

1. Set $\{FLAG_i\}_{i=1}^{p^e} = 0$ and *iter* = 0. The initial model is $\hat{y} = \bar{y}$. If the adjusted R^2, AICC, or ASE criterion is used, compute the statistic for the initial model and denote it as MSC_{current}.

2. If $\{i|FLAG_i = 0\} \neq \emptyset$, $iter \leq MAXSTEP$ and $|\{i|FLAG_i = 1\}| < MAXEFFECT$, go to the next step; otherwise stop and output the current model .

3. Based on the current model, for every effect *j* eligible for entry (see Condition below),

   If FC (the *F* statistic criterion) is used, compute F_{enter_j} and p_{enter_j};

   If MSC (the adjusted R^2, AICC, or ASE criterion) is used, compute MSC_j.

4. If FC is used, choose the effect $X_{j^*}, j^* = arg\min_j \{p_{enter_j}\}$ and if $p_{enter_{j^*}} < P_{\text{in}}$, enter X_{j^*} to the current model.

   If MSC is used, choose the effect $X_{j^*}, j^* = arg\min_j \{MSC_j\}$ and if $MSC_{j^*} < MSC_{current}$, enter X_{j^*} to the current model. (For the adjusted R^2 criterion, replace min with max and reverse the inequality)

   If the inequality is not satisfied, stop and output the current model.

5. If the model with the new effect is the same as any previously obtained model, stop and output the current model; otherwise update the current model by doing the sweep operation on corresponding row(s) and column(s) associated with X_{j^*} in the current **R** matrix. Set $FLAG_{j*} = 1$ and $iter = iter + 1$.

   If FC is used, let $\Delta F = F_{enter_{j^*}}$ and $p_{\Delta F} = p_{enter_{j^*}}$;

   If MSC is used, let $MSC_{current} = MSC_{j^*}$.

6. For every effect *k* in the current model; that is, $FLAG_k = 1, \forall k$,

   If FC is used, compute F_{remove_k} and p_{remove_k};

   If MSC is used, compute MSC_k.

7. If FC is used, choose the effect $X_{k^*}, k^* = arg\max_k \{p_{remove_k}\}$ and if $p_{remove_{k^*}} > P_{\text{out}}$, remove X_{k^*} from the current model.

   If MSC is used, choose the effect $X_{k^*}, k^* = arg\min_k \{MSC_k\}$ and if $MSC_{j^*} < MSC_{current}$, remove X_{k^*} from the current model. (For the adjusted R^2 criterion, replace min with max and reverse the inequality)

   If the inequality is met, go to the next step; otherwise go back to step 2.

8. If the model with the effect removed is the same as any previously obtained model, stop and output the current model; otherwise update the current model by doing the sweep operation on corresponding row(s) and column(s) associated with X_{j^*} in the current **R** matrix. Set $FLAG_{j^*} = 0$ and *iter* = *iter* + 1.

   If FC is used, let $\Delta F = F_{remove_{k^*}}$ and $p_{\Delta F} = p_{remove_{k^*}}$;

   If AC is used, let $AICC_{current} = AICC_{k^*}$. Then go back to step 6.

   Condition. In order for effect *j* to be eligible for entry into the model, the following conditions must be met:

   For continuous a effect X_j, $r_{jj} \geq t$; (*t* is the singularity tolerance with a value of 1e−4)

   For categorical effect $\{X_{j_s}\}_{s=1}^{\ell}$, $max\{r_{j_1 j_1}, r_{j_2 j_2}, \ldots, r_{j_\ell j_\ell}\} \geq t$;

   where *t* is the singularity tolerance, and r_{jj} and $r_{j_s j_s}, s = 1, \ldots, \ell$ are diagonal elements in the current **R** matrix (before entering).

   For each continuous effect X_k that is currently in the model, $\tilde{r}_{kk} t \leq 1$.

   For each categorical effect $\{X_{k_s}\}_{s=1}^{\ell'}$ with ℓ' levels that is currently in the model, $max\left\{\tilde{r}_{k_1 k_1}, \tilde{r}_{k_2 k_2}, \ldots, \tilde{r}_{k_{\ell'} k_{\ell'}}\right\} t \leq 1$.

   where $\tilde{r}_{kk}$ and $\tilde{r}_{k_s k_s}, s = 1, \ldots, \ell'$, are diagonal elements in the resulting **R** matrix; that is, the results after doing the sweep operation on corresponding row(s) and column(s) associated with X_k or $\{X_{k_s}\}_{s=1}^{\ell'}$ in the current R matrix. The above condition is imposed so that entry of the effect does not reduce the tolerance of other effects already in the model to unacceptable levels.

Best subsets

Stepwise methods search fewer combinations of sub-models and rarely select the best one, so another option is to check all possible models and select the “best” based upon some criterion. The available criteria are the maximum adjusted R^2, minimum AICC, and minimum ASE over the overfit prevention set.

Since there are p^e free effects, we do an exhaustive search over 2^{p^e} models, which include intercept-only model ($\hat{y} = \bar{y}$). Because the number of calculations increases exponentially with p^e, it is important to have an efficient algorithm for carrying out the necessary computations. However, if p^e is too large, it may not be practical to check all of the possible models.

We divide the problem into 2 tiers in terms of the number of effects:

- when $p^e \leq 20$, we search all possible subsets
- when $p^e > 20$, we apply a hybrid method which combines the forward stepwise method and the all possible subsets method.

Searching All Possible Subsets

An efficient method that minimizes the number of sweep operations on the **R** matrix (Schatzoff 1968), is applied to traverse all the models and outlined as follows:

Each sweep step(s) on an effect results in a model. So 2^{p^e} models can be obtained through a sequence of exactly 2^{p^e} sweeps on effects. Assuming that the all possible models on $p^e - 1$ effects can be obtained in a sequence $S_{p^e\_1}$ of exactly 2^{p^e-1} sweeps on the first 2^{p^e-1} pivotal effects, and sweeping on the last effect will produce a new model which adds the last effect to the model produced by the sequence $S_{p^e\_1}$, then repeating the sequence $S_{p^e\_1}$ will produce another 2^{p^e-1} distinct models (including the last effect). It is a recursive algorithm for constructing the sequence; that is, $S_{p^e} = \left(\underline{S_{p^e-1}}, k, \underline{S_{p^e-1}}\right) = \left(\underline{S_{p^e-2}, k-1, S_{p^e-2}}, k, \underline{S_{p^e\_2}, k-1, S_{p^e\_2}}\right) = \ldots$, and so on.

The sequence of models produced is demonstrated in the following table:

k	S_k	Sequence of models produced
0	0	Only intercept
1	1	(1)
2	121	(1),(12),(2)
3	1213121	(1),(12),(2),(23),(123),(13),(3)
4	121312141213121	(1),(12),(2),(23),(123),(13),(3),(34),(134),(1234),(234),(24),(124),(14),(4)
...	...	...
p^e	$S_{p^e\_1}$, p^e, $S_{p^e\_1}$	All 2^{p^e} models including the intercept model.

The second column indicates the indexes of effects which are pivoted on. Each parenthesis in the third column represents a regression model. The numbers in the parentheses indicate the effects which are included in that model.

Hybrid Method

If p^e>20, we apply a hybrid method by combining the forward stepwise method with the all possible subsets method as follows:

Select the effects using the forward stepwise method with the same criterion chosen for best subsets. Say that p^s is the number of effects chosen by the forward stepwise method.

Apply one of the following approaches, depending on the value of p^s, as follows:

- If $p^s \leq 20$, do an exhaustive search of all possible subsets on these selected effects, as described above.
- If $20 < p^s \leq 40$, select $p^s - 20$ effects based on the *p*-values of type III sum of squares tests from all p^s effects (see ANOVA in “Model evaluation”) and enter them into the model, then do an exhaustive search of the remaining 20 effects via the method described above.
- If $40 < p^s$, do nothing and assume the best model is the one with these p^s effects (with a warning message that the selected model is based on the forward stepwise method).

Model evaluation

The following output statistics are available.

ANOVA

Weighted total sum of squares

$$SS_t = \sum_{i=1}^{n} w_i (y_i - \overline{y})^2 = (N-1)\,S_{yy} \text{ with d.f. } = df_t = N - 1$$

where d.f. means degrees of freedom. It is called “SS (sum of squares) for Corrected Total”.

Weighted residual sum of squares

$$SS_e = \sum_{i=1}^{n} w_i (y_i - \hat{y}_i)^2 = \tilde{r}_{yy}\,(N-1)\,S_{yy}$$

with d.f. = $df_e = N - p^c$. It is also called “SS for Error”.

Weighted regression sum of squares

$$SS_r = \sum_{i=1}^{n} w_i (\hat{y}_i - \overline{y})^2 = (1 - \tilde{r}_{yy})\,(N-1)\,S_{yy} = SS_t - SS_e$$

with d.f. = $df_r = p^*$. It is called “SS for Corrected Model” if there is an intercept.

Regression mean square error

$$SS_r / df_r$$

Residual mean square error

$$SS_e / df_e$$

F statistic for corrected model

$$F = \frac{SS_r / df_r}{SS_e / df_e} = \frac{SS_r \cdot df_e}{SS_e \cdot df_r}$$

which follows an F distribution with degrees of freedom df_r and df_e, and the corresponding p-value can be calculated accordingly.

Type III sum of squares for each effect

To compute type III SS for the effect *j*, $j = 1, \dots, p^e$, the type III test matrix $\mathbf{L}_i$ needs to be constructed first. Construction of $\mathbf{L}_i$ is based on the generating matrix $\mathrm{H}_\omega = \left(\mathrm{X}^\mathrm{T}\mathrm{DX}\right)^{-}\mathrm{X}^\mathrm{T}\mathrm{DX}$, where $\mathrm{D} = diag\left(g_1, \dots, g_n\right)$, such that $\mathbf{L}_i\beta$ is estimable. It involves parameters only for the given effect and the effects containing the given effect. For type III analysis, $\mathbf{L}_i$ doesn't depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable. For each effect *j*, the type III SS is calculated as follows

$$\mathbf{S}_j = \hat{\beta}^\mathrm{T}\mathbf{L}_j^\mathrm{T}\left(\mathbf{L}_j\mathbf{G}\mathbf{L}_j^\mathrm{T}\right)^{-1}\mathbf{L}_j\hat{\beta}$$

where $\mathbf{G} = \left(\mathbf{X}^T\mathbf{W}\mathbf{X}\right)^{-}$.

F statistic for each effect

The SS for the effect *j* is also used to compute the *F* statistic for the hypothesis test H_0: $\mathbf{L}_i\beta = \mathbf{0}$ as follows:

$$F_j = \frac{\mathbf{S}_j/r_j}{SS_e/df_e}$$

where r_j is the full row rank of $\mathbf{L}_i$. It follows an *F* distribution with degrees of freedom r_i and df_e, then the *p*-values can be calculated accordingly.

Model summary

Adjusted R square

$$\text{adj.}R^2 = 1 - \frac{SS_e/df_e}{SS_t/df_t} = R^2 - \frac{\left(1-R^2\right)p^*}{df_e} = 1 - \frac{df_t \times \tilde{r}_{yy}}{df_e}$$

where

$$R^2 = \frac{SS_r}{SS_t} = 1 - \frac{SS_e}{SS_t} = 1 - \tilde{r}_{yy}.$$

Model information criteria

Corrected Akaike information criterion (AICC)

$$AICC = N\ln\left(\frac{SS_e}{N}\right) + \frac{2p^cN}{N - p^c - 1}$$

Coefficients and statistical inference

After the model selection process, we can get the coefficients and related statistics from the swept correlation matrix. The following statistics are computed based on the **R** matrix.

Unstandardized coefficient estimates

$$\hat{\beta}_j = b_j \sqrt{\frac{S_{yy}}{S_{jj}}} = \tilde{r}_{jy} \sqrt{\frac{S_{yy}}{S_{jj}}}$$

for $j = 1, \cdots, p^*$.

Standard errors of regression coefficients

The standard error of $\hat{\beta}_j$ is

$$\hat{\sigma}_{\hat{\beta}_j} = \sqrt{var\left(\hat{\beta}_j\right)} = \sqrt{\frac{\tilde{r}_{jj}\tilde{r}_{yy}S_{yy}}{S_{jj}df_e}}$$

Intercept estimation

The intercept is estimated by all other parameters in the model as

$$\hat{\beta}_0 = \overline{y} - \sum_{j=1}^{p} \hat{\beta}_j \overline{X}_j$$

The standard error of $\hat{\beta}_0$ is estimated by

$$\hat{\sigma}_{\hat{\beta}_0} = \sqrt{\hat{\sigma}^2_{\hat{\beta}_0}}$$

where

$$\begin{aligned}\hat{\sigma}^2_{\hat{\beta}_0} &= \frac{(N-1)\tilde{r}_{yy}S_{yy}}{N(N-p^*-1)} + \sum_{j=1}^{p} \overline{X}_j^2 \hat{\sigma}^2_{\hat{\beta}_j} + 2\sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \overline{X}_k \overline{X}_j cov\left(\hat{\beta}_k, \hat{\beta}_j\right) \\ &= \frac{SS_e}{N \times df_e} + \sum_{j=1}^{p} \overline{X}_j^2 \hat{\sigma}^2_{\hat{\beta}_j} + 2\sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \overline{X}_k \overline{X}_j \frac{\tilde{r}_{kj} \times SS_e}{\sqrt{S_{kk}S_{jj}} \times (N-1)df_e}.\end{aligned}$$

$\hat{\sigma}^2_{\hat{\beta}_0} = \frac{(N-1)\tilde{r}_{yy}S_{yy}}{N(N-p^*-1)} + \sum_{j=1}^{p} \overline{X}_j^2 \hat{\sigma}^2_{\hat{\beta}_j} + 2\sum_{j=1}^{p-1} \sum_{k=j+1}^{p} \overline{X}_k \overline{X}_j cov\left(\hat{\beta}_k, \hat{\beta}_j\right)$ and $cov\left(\hat{\beta}_k, \hat{\beta}_j\right)$ is the *k*th row and *j*th column element in the parameter estimates covariance matrix.

t statistics for regression coefficients

$$t = \frac{\hat{\beta}_j}{\hat{\sigma}_{\hat{\beta}_j}} = \tilde{r}_{jy} \sqrt{\frac{df_e}{\tilde{r}_{yy}\tilde{r}_{jj}}}$$

for $j = 1, \cdots, p^*$, with degrees of freedom df_e and the *p*-value can be calculated accordingly.

100(1−α)% confidence intervals

$$\hat{\beta}_j \pm \hat{\sigma}_{\hat{\beta}_j} \times t_{\alpha/2, df_e}$$

Note: For redundant parameters, the coefficient estimates are set to zero and standard errors, t statistics, and confidence intervals are set to missing values.

Scoring

Predicted values

$$\hat{y}_k = \sum_{i=0}^{p} x_{ki}\hat{\beta}_i, k = 1, \ldots, n.$$

Diagnostics

The following values are computed to produce various diagnostic charts and tables.

Residuals

$$e_k = y_k - \hat{y}_k$$

Studentized residuals

This is the ratio of the residual to its standard error.

$$SRES_k = \frac{e_k}{s\sqrt{\frac{(1-h_k)}{g_k}}}$$

where *s* is the square root of the mean square error; that is, $s = \sqrt{SS_e/df_e}$, and h_k is the leverage value for the *k*th case (see below).

Cook’s distance

$$COOK_k = \frac{e_k^2 h_k g_k}{s^2(1-h_k)^2 p^c}$$

where the “leverage”

$$h_k = g_k \mathbf{x}_k \mathbf{G} \mathbf{x}_k^{\mathrm{T}}$$

is the *k*th diagonal element of the hat matrix

$$\mathrm{H} = \mathrm{W}^{1/2}\mathrm{X}\left(\mathrm{X}^{\mathrm{T}}\mathrm{W}\mathrm{X}\right)^{-}\mathrm{X}^{\mathrm{T}}\mathrm{W}^{1/2} = \mathrm{W}^{1/2}\mathrm{X}\mathbf{G}\mathrm{X}^{\mathrm{T}}\mathrm{W}^{1/2}$$

A record with Cook’s distance larger than $\frac{4}{N-p^c}$ is considered influential (Fox, 1997).

Predictor importance

We use the leave-one-out method to compute the predictor importance, based on the residual sum of squares (SSe) by removing one predictor at a time from the final full model.

If the final full model contains *p* predictors, $X_1, X_2, \cdots, X_p$, then the predictor importance can be calculated as follows:

1. *i*=1
2. If *i*>*p*, go to step 5.
3. Do a sweep operation on the corresponding row(s) and column(s) associated with X_i in the $\tilde{\mathbf{R}}$ matrix of the full final model.
4. Get the last diagonal element in the current $\tilde{\mathbf{R}}$ and denote it $\tilde{r}_{yy}^{(i)}$. Then the predictor importance of X_i is $VI_i = \left(\tilde{r}_{yy}^{(i)} - \tilde{r}_{yy}\right)(N-1)\,SS_{yy}$. Let *i* = *i* + 1, and go to step 2.
5. Compute the normalized predictor importance of X_i:

   $NormVI_i = \frac{VI_i}{\Sigma_{i=1}^{p} VI_i}$

References

Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. *Regression diagnostics: Identifying influential data and sources of collinearity*. New York: John Wiley and Sons.

Dempster, A. P. 1969. *Elements of Continuous Multivariate Analysis*. Reading, MA: Addison-Wesley.

Fox, J. 1997. *Applied Regression Analysis, Linear Models, and Related Methods*. Thousand Oaks, CA: SAGE Publications, Inc..

Fox, J., and G. Monette. 1992. Generalized collinearity diagnostics. *Journal of the American Statistical Association*, 87, 178–183.

Schatzoff, M., R. Tsao, and S. Fienberg. 1968. Efficient computing of all possible regressions. *Technometrics*, 10, 769–779.

Velleman, P. F., and R. E. Welsch. 1981. Efficient computing of regression diagnostics. *American Statistician*, 35, 234–242.

LOGISTIC REGRESSION Algorithms

Logistic regression regresses a dichotomous dependent (target) variable on a set of independent (predictor) variables. Several methods are implemented for selecting the independent variables.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	The number of observed cases
p	The number of parameters
y	$n \times 1$ vector with element y_i, the observed value of the *i*th case of the dichotomous dependent variable
X	$n \times p$ matrix with element x_{ij}, the observed value of the *i*th case of the *j*th parameter
β	$p \times 1$ vector with element β_j, the coefficient for the *j*th parameter
w	$n \times 1$ vector with element w_i, the weight for the *i*th case
l	Likelihood function
L	Log-likelihood function
I	Information matrix

Model

The linear logistic model assumes a dichotomous dependent variable *Y* with probability π, where for the *i*th case,

$$\pi_i = \frac{\exp(\eta_i)}{1+\exp(\eta_i)}$$

or

$$\ln\left(\frac{\pi_i}{1-\pi_i}\right) = \eta_i = \mathbf{X}_i'\beta$$

Hence, the likelihood function *l* for *n* observations $y_1, \ldots, y_n$, with probabilities $\pi_1, \ldots, \pi_n$ and case weights $w_1, \ldots, w_n$, can be written as

$$l = \prod_{i=1}^{n} \pi_i^{w_i y_i}(1-\pi_i)^{w_i(1-y_i)}$$

It follows that the logarithm of *l* is

$$L = \ln(l) = \sum_{i=1}^{n} \left(w_i y_i \ln(\pi_i) + w_i(1-y_i)\ln(1-\pi_i)\right)$$

and the derivative of *L* with respect to β_j is

$$L^*_{X_j} = \frac{\partial L}{\partial \beta_j} = \sum_{i=1}^{n} w_i(y_i - \pi_i)x_{ij}$$

Maximum Likelihood Estimates (MLE)

The maximum likelihood estimates for β satisfy the following equations

$\sum_{i=1}^{n} w_i(y_i - \hat{\pi}_i)x_{ij} = 0$, for the *j*th parameter

where $x_{i0} = 1$ for $i = 1, \ldots, n$.

Note the following:

1. A Newton-Raphson type algorithm is used to obtain the MLEs. Convergence can be based on
   - Absolute difference for the parameter estimates between the iterations
   - Percent difference in the log-likelihood function between successive iterations
   - Maximum number of iterations specified
2. During the iterations, if $\hat{\pi}_i(1-\hat{\pi}_i)$ is smaller than 10^{-8} for all cases, the log-likelihood function is very close to zero. In this situation, iteration stops and the message “All predicted values are either 1 or 0” is issued.

After the maximum likelihood estimates $\hat{\beta}$ are obtained, the asymptotic covariance matrix is estimated by I^{-1}, the inverse of the information matrix *I*, where

$$I = -\left[E\left(\frac{\partial^2 L}{\partial\beta_i\,\partial\beta_j}\right)\right] = \mathbf{X}'\mathbf{W}\hat{\mathbf{V}}\mathbf{X},$$

$$\hat{\mathbf{V}} = Diag\{\hat{\pi}_1(1-\hat{\pi}_1), \ldots, \hat{\pi}_n(1-\hat{\pi}_n)\},$$

$$\mathbf{W} = Diag\{w_1, \ldots, w_n\},$$

$$\hat{\pi}_i = \frac{\exp(\hat{\eta}_i)}{1+\exp(\hat{\eta}_i)},$$

and

$$\hat{\eta}_i = \mathbf{X}_i{}'\hat{\beta}$$

Stepwise Variable Selection

Several methods are available for selecting independent variables. With the forced entry method, any variable in the variable list is entered into the model. There are two stepwise methods: forward and backward. The stepwise methods can use either the Wald statistic, the likelihood ratio, or a conditional algorithm for variable removal. For both stepwise methods, the score statistic is used to select variables for entry into the model.

Forward Stepwise (FSTEP)

1. If FSTEP is the first method requested, estimate the parameter and likelihood function for the initial model. Otherwise, the final model from the previous method is the initial model for FSTEP.

Obtain the necessary information: MLEs of the parameters for the current model, predicted probability, likelihood function for the current model, and so on.

2. Based on the MLEs of the current model, calculate the score statistic for every variable eligible for inclusion and find its significance.

3. Choose the variable with the smallest significance. If that significance is less than the probability for a variable to enter, then go to step 4; otherwise, stop FSTEP.

4. Update the current model by adding a new variable. If this results in a model which has already been evaluated, stop FSTEP.

5. Calculate LR or Wald statistic or conditional statistic for each variable in the current model. Then calculate its corresponding significance.

6. Choose the variable with the largest significance. If that significance is less than the probability for variable removal, then go back to step 2; otherwise, if the current model with the variable deleted is the same as a previous model, stop FSTEP; otherwise, go to the next step.

7. Modify the current model by removing the variable with the largest significance from the previous model. Estimate the parameters for the modified model and go back to step 5.

Backward Stepwise (BSTEP)

1. Estimate the parameters for the full model which includes the final model from previous method and all eligible variables. Only variables listed on the BSTEP variable list are eligible for entry and removal. Let the current model be the full model.

2. Based on the MLEs of the current model, calculate the LR or Wald statistic or conditional statistic for every variable in the model and find its significance.

3. Choose the variable with the largest significance. If that significance is less than the probability for a variable removal, then go to step 5; otherwise, if the current model without the variable with the largest significance is the same as the previous model, stop BSTEP; otherwise, go to the next step.

4. Modify the current model by removing the variable with the largest significance from the model. Estimate the parameters for the modified model and go back to step 2.

5. Check to see any eligible variable is not in the model. If there is none, stop BSTEP; otherwise, go to the next step.

6. Based on the MLEs of the current model, calculate the score statistic for every variable not in the model and find its significance.

7. Choose the variable with the smallest significance. If that significance is less than the probability for variable entry, then go to the next step; otherwise, stop BSTEP.

8. Add the variable with the smallest significance to the current model. If the model is not the same as any previous models, estimate the parameters for the new model and go back to step 2; otherwise, stop BSTEP.

Stepwise Statistics

The statistics used in the stepwise variable selection methods are defined as follows.

Score Statistic

The score statistic is calculated for each variable not in the model to determine whether the variable should enter the model. Assume that there are r_1 variables, namely, $\alpha_1, \ldots, \alpha_{r_1}$ in the model and r_2 variables, $\gamma_1, \ldots, \gamma_{r_2}$, not in the model. The score statistic for γ_i is defined as

$$\mathbf{S}_i = \left(\mathbf{L}^*_{\gamma_i}\right)^2 \mathbf{B}_{22,i}$$

if γ_i is not a categorical variable. If γ_i is a categorical variable with *m* categories, it is converted to a $(m-1)$-dimension dummy vector. Denote these new $m-1$ variables as $\tilde{\gamma}_i, \ldots, \tilde{\gamma}_{i+m-2}$. The score statistic for γ_i is then

$$\mathbf{S}_i = \left(\mathbf{L}^*_{\tilde{\gamma}}\right)' \mathbf{B}_{22,i} \mathbf{L}^*_{\tilde{\gamma}}$$

where $(\mathbf{L}^*_{\tilde{\gamma}})' = \left(L^*_{\tilde{\gamma}_i}, \ldots, L^*_{\tilde{\gamma}_{i+m-2}}\right)$ and the $(m-1) \times (m-1)$ matrix $\mathbf{B}_{22,i}$ is

$$\mathbf{B}_{22,i} = \left(\mathbf{A}_{22,i} - \mathbf{A}_{21,i}\mathbf{A}_{11}^{-1}\mathbf{A}_{12,i}\right)^{-1}$$

with

$$\mathbf{A}_{11} = \underset{\sim}{\alpha}'\hat{\mathbf{V}}\underset{\sim}{\alpha},$$
$$\mathbf{A}_{12,i} = \underset{\sim}{\alpha}'\hat{\mathbf{V}}\underset{\sim}{\gamma_i},$$
$$\mathbf{A}_{22,i} = \underset{\sim i}{\gamma_i}'\hat{\mathbf{V}}\underset{\sim}{\gamma_i}$$

in which $\underset{\sim}{\alpha}$ is the design matrix for variables $\alpha_1, \ldots, \alpha_{r_1}$ and $\underset{\sim}{\gamma_i}$ is the design matrix for dummy variables $\tilde{\gamma}_i, \ldots, \tilde{\gamma}_{i+m-2}$. Note that $\underset{\sim}{\alpha}$ contains a column of ones unless the constant term is excluded from η. Based on the MLEs for the parameters in the model, **V** is estimated by $\hat{\mathbf{V}} = Diag\{\hat{\pi}_1(1-\hat{\pi}_1), \ldots, \hat{\pi}_n(1-\hat{\pi}_n)\}$. The asymptotic distribution of the score statistic is a chi-square with degrees of freedom equal to the number of variables involved.

Note the following:

1. If the model is through the origin and there are no variables in the model, $\mathbf{B}_{22,i}$ is defined by $\mathbf{A}_{22,i}^{-1}$ and $\hat{\mathbf{V}}$ is equal to $\frac{1}{4}\mathbf{I}_n$.
2. If $\mathbf{B}_{22,i}$ is not positive definite, the score statistic and residual chi-square statistic are set to be zero.

Wald Statistic

The Wald statistic is calculated for the variables in the model to determine whether a variable should be removed. If the *i*th variable is not categorical, the Wald statistic is defined by

$$Wald_i = \frac{\hat{\beta}_i^2}{\hat{\sigma}^2_{\hat{\beta}_i}}$$

If it is a categorical variable, the Wald statistic is computed as follows:

Let $\hat{\beta}_i$ be the vector of maximum likelihood estimates associated with the $m-1$ dummy variables, and $\mathbf{C}$ the asymptotic covariance matrix for $\hat{\beta}_i$. The Wald statistic is

$$Wald_i = \hat{\beta}_i' \mathbf{C}^{-1} \hat{\beta}_i$$

The asymptotic distribution of the Wald statistic is chi-square with degrees of freedom equal to the number of parameters estimated.

Likelihood Ratio (LR) Statistic

The LR statistic is defined as two times the log of the ratio of the likelihood functions of two models evaluated at their MLEs. The LR statistic is used to determine if a variable should be removed from the model. Assume that there are r_1 variables in the current model which is referred to as a full model. Based on the MLEs of the full model, *l(full)* is calculated. For each of the variables removed from the full model one at a time, MLEs are computed and the likelihood function *l(reduced)* is calculated. The LR statistic is then defined as

$$LR = -2\ln\left(\frac{l(reduced)}{l(full)}\right) = -2(L(reduced) - L(full))$$

LR is asymptotically chi-square distributed with degrees of freedom equal to the difference between the numbers of parameters estimated in the two models.

Conditional Statistic

The conditional statistic is also computed for every variable in the model. The formula for the conditional statistic is the same as the LR statistic except that the parameter estimates for each reduced model are conditional estimates, not MLEs. The conditional estimates are defined as follows. Let $\hat{\beta} = \left(\hat{\beta}_1, \ldots, \hat{\beta}_{r_1}\right)'$ be the MLE for the r_1 variables in the model and $\mathbf{C}$ be the asymptotic covariance matrix for $\hat{\beta}$. If variable x_i is removed from the model, the conditional estimate for the parameters left in the model given $\hat{\beta}$ is

$$\tilde{\beta}_{(i)} = \hat{\beta}_{(i)} - \mathbf{c}_{12}^{(i)}\left(\mathbf{c}_{22}^{(i)}\right)^{-1}\hat{\beta}_i$$

where $\hat{\beta}_i$ is the MLE for the parameter(s) associated with x_i and $\hat{\beta_{(i)}}$ is $\hat{\beta}$ with $\hat{\beta}_i$ removed, $\mathbf{c}_{12}^{(i)}$ is the covariance between $\hat{\beta_{(i)}}$ and $\hat{\beta}_i$, and $\mathbf{c}_{22}^{(i)}$ is the covariance of $\hat{\beta}_i$. Then the conditional statistic is computed by

$$-2\left(L\left(\tilde{\beta}_{(i)}\right) - L(full)\right)$$

where $L\left(\tilde{\beta}_{(i)}\right)$ is the log-likelihood function evaluated at $\hat{\beta_{(i)}}$.

Statistics

The following output statistics are available.

Initial Model Information

If β_0 is not included in the model, the predicted probability is estimated to be 0.5 for all cases and the log-likelihood function $L(0)$ is

$$L(0) = W \ln(0.5) = -0.6931472W$$

with $W = \sum_{i=1}^{n} w_i$. If β_0 is included in the model, the predicted probability is estimated as

$$\hat{\pi}_0 = \frac{\sum_{i=1}^{n} w_i y_i}{W}$$

and β_0 is estimated by

$$\hat{\beta}_0 = \ln\left(\frac{\hat{\pi}_0}{1-\hat{\pi}_0}\right)$$

with asymptotic standard error estimated by

$$\hat{\sigma}_{\hat{\beta}_0} = \frac{1}{\sqrt{W\hat{\pi}_0(1-\hat{\pi}_0)}}$$

The log-likelihood function is

$$L(0) = W\left[\hat{\pi}_0 \ln\left(\frac{\hat{\pi}_0}{1-\hat{\pi}_0}\right) + \ln(1-\hat{\pi}_0).\right]$$

Model Information

The following statistics are computed if a stepwise method is specified.

–2 Log-Likelihood

$$-2\sum_{i=1}^{n} \left(w_i y_i \ln(\hat{\pi}_i) + w_i(1-y_i)\ln(1-\hat{\pi}_i)\right)$$

Model Chi-Square

2(log-likelihood function for current model − log-likelihood function for initial model)

The initial model contains a constant if it is in the model; otherwise, the model has no terms. The degrees of freedom for the model chi-square statistic is equal to the difference between the numbers of parameters estimated in each of the two models. If the degrees of freedom is zero, the model chi-square is not computed.

Block Chi-Square

2(log-likelihood function for current model − log-likelihood function for the final model from the previous method)

The degrees of freedom for the block chi-square statistic is equal to the difference between the numbers of parameters estimated in each of the two models.

Improvement Chi-Square

2(log-likelihood function for current model − log-likelihood function for the model from the last step)

The degrees of freedom for the improvement chi-square statistic is equal to the difference between the numbers of parameters estimated in each of the two models.

Goodness of Fit

$$\sum_{i=1}^{n} \frac{w_i(y_i - \hat{\pi}_i)^2}{\hat{\pi}_i(1-\hat{\pi}_i)}$$

Cox and Snell's R-Square (Cox and Snell, 1989; Nagelkerke, 1991)

$$R^2_{CS} = 1 - \left(\frac{l(0)}{l(\hat{\beta})}\right)^{\frac{2}{W}}$$

where $l\left(\hat{\beta}\right)$ is the likelihood of the current model and *l(0)* is the likelihood of the initial model; that is, $l(0) = W\log(0.5)$ if the constant is not included in the model; $l(0) = W[\hat{\pi}_o \log\{\hat{\pi}_o/(1-\hat{\pi}_o)\} + \log(1-\hat{\pi}_o)]$ if the constant is included in the model, where $\hat{\pi}_o = \Sigma_i^n w_i y_i / W$.

Nagelkerke's R-Square (Nagelkerke, 1981)

$$R^2_N = R^2_{CS}/\max\left(R^2_{CS}\right)$$

where $\max\left(R^2_{CS}\right) = 1 - \{l(0)\}^{2/W}$.

Hosmer-Lemeshow Goodness-of-Fit Statistic

The test statistic is obtained by applying a chi-square test on a $2 \times g$ contingency table. The contingency table is constructed by cross-classifying the dichotomous dependent variable with a grouping variable (with *g* groups) in which groups are formed by partitioning the predicted probabilities using the percentiles of the predicted event probability. In the calculation, approximately 10 groups are used (*g*=10). The corresponding groups are often referred to as the "deciles of risk" (Hosmer and Lemeshow, 2000).

If the values of independent variables for observation *i* and *i'* are the same, observations *i* and *i'* are said to be in the same block. When one or more blocks occur within the same decile, the blocks are assigned to this same group. Moreover, observations in the same block are not divided when they are placed into groups. This strategy may result in fewer than 10 groups (that is, $g \leq 10$) and consequently, fewer degrees of freedom.

Suppose that there are *Q* blocks, and the *q*th block has m_q number of observations, $q = 1, \ldots, Q$. Moreover, suppose that the *k*th group ($k = 1, \ldots, g$) is composed of the q_1th, …, q_kth blocks of observations. Then the total number of observations in the *k*th group is $s_k = \Sigma_{q_1}^{q_k} m_j$. The total observed frequency of events (that is, *Y*=1) in the *k*th group, call it O_{1k}, is the total number of observations in the *k*th group with *Y*=1. Let E_{1k} be the total expected frequency of the event in the *k*th group; then E_{1k} is given by $E_{1k} = s_k \xi_k$, where ξ_k is the average predicted event probability for the *k*th group.

$$\xi_k = \Sigma_{q_1}^{q_k} m_j \hat{\pi}_j / s_k$$

The Hosmer-Lemeshow goodness-of-fit statistic is computed as

$$\chi^2_{HL} = \sum_{k=1}^{g} \frac{(O_{1k} - E_{1k})^2}{E_{1k}(1 - \xi_k)}$$

The *p* value is given by Pr$\left(\chi^2 \geq \chi^2_{HL}\right)$ where χ^2 is the chi-square statistic distributed with degrees of freedom (*g*−2).

Information for the Variables Not in the Equation

For each of the variables not in the equation, the score statistic is calculated along with the associated degrees of freedom, significance and partial *R*. Let X_i be a variable not currently in the model and S_i the score statistic. The partial *R* is defined by

$$Partial\_R = \begin{cases} \sqrt{\frac{S_i - 2\times df}{-2L(initial)}} & \text{if } S_i > 2 \times df \\ 0 & \text{otherwise} \end{cases}$$

where *df* is the degrees of freedom associated with S_i, and $L(initial)$ is the log-likelihood function for the initial model.

The residual Chi-Square printed for the variables not in the equation is defined as

$$R_{CS} = \left(L^*_g\right)' B_{22} L^*_g$$

where $L^*_g = \left(L^*_{\gamma_1}, \ldots, L^*_{\gamma_{r_2}}\right)'$

Information for the Variables in the Equation

For each of the variables in the equation, the MLE of the Beta coefficients is calculated along with the standard errors, Wald statistics, degrees of freedom, significances, and partial *R*. If X_i is not a categorical variable currently in the equation, the partial *R* is computed as

$$Partial\_R = \begin{cases} sign\left(\hat{\beta}_i\right)\sqrt{\frac{Wald_i - 2}{-2L(initial)}} & \text{if } Wald_i > 2 \\ 0 & \text{otherwise} \end{cases}$$

If X_i is a categorical variable with *m* categories, the partial *R* is then

$$Partial\_R = \begin{cases} \sqrt{\frac{Wald_i - 2(m-1)}{-2L(initial)}} & \text{if } Wald_i > 2(m-1) \\ 0 & \text{otherwise} \end{cases}$$

Casewise Statistics

The following statistics are computed for each case.

Individual Deviance

The deviance of the *i*th case, G_i, is defined as

$$G_i = \begin{cases} \sqrt{2(y_i \ln(\hat{\pi}_i) + (1-y_i)\ln(1-\hat{\pi}_i))} & \text{if } y_i > \hat{\pi}_i \\ -\sqrt{2(y_i \ln(\hat{\pi}_i) + (1-y_i)\ln(1-\hat{\pi}_i))} & \text{otherwise} \end{cases}$$

Leverage

The leverage of the *i*th case, h_i, is the *i*th diagonal element of the matrix

$$\hat{\mathbf{V}}^{\frac{1}{2}}\mathbf{X}\left(\mathbf{X}'\mathbf{C}\hat{\mathbf{V}}\mathbf{X}\right)^{-1}\mathbf{X}'\hat{\mathbf{V}}^{\frac{1}{2}}$$

where

$$\hat{\mathbf{V}} = Diag\{\hat{\pi}_1(1-\hat{\pi}_1), \ldots, \hat{\pi}_n(1-\hat{\pi}_n)\}$$

Studentized Residual

$$\tilde{G}_i^* = \frac{G_i}{\sqrt{1-h_i}}$$

Logit Residual

$$\tilde{e}_i = \frac{e_i}{\hat{\pi}_i(1-\hat{\pi}_i)}$$

where $e_i = y_i - \hat{\pi}_i$

Standardized Residual

$$z_i = \frac{e_i}{\sqrt{\hat{\pi}_i(1-\hat{\pi}_i)}}$$

Cook's Distance

$$D_i = \frac{z_i^2 h_i}{1-h_i}$$

DFBETA

Let $\Delta\beta_i$ be the change of the coefficient estimates from the deletion of case *i*. It is computed as

$$\Delta\beta_i = \frac{\left(\mathbf{X}'\mathbf{C}\hat{\mathbf{v}}\mathbf{X}\right)^{-1}\mathbf{X}'_i e_i}{1-h_i}$$

Predicted Group

If $\hat{\pi}_i \geq 0.5$, the predicted group is the group in which

y=1. Note the following:

For the unselected cases with nonmissing values for the independent variables in the analysis, the leverage $\left(\tilde{h}_i\right)$ is computed as

$$\tilde{h}_i = h_i - \frac{\hat{V}_i h_i^2}{1+\hat{V}_i h_i}$$

where

$$h_i = \hat{V}_i \mathbf{X}'_i \left(\mathbf{X}'\mathbf{C}\hat{\mathbf{V}}\mathbf{X}\right)^{-1}\mathbf{X}_i$$

For the unselected cases, the Cook’s distance and DFBETA are calculated based on $\tilde{h}_i$.

LOGLINEAR Algorithms

The LOGLINEAR procedure models cell frequencies using the multinomial response model and produces maximum likelihood estimates of parameters by the Newton-Raphson method. The contingency tables are converted to two-way *I*×*J* tables, with *I* and *J* being the dimensions of the independent and dependent categorical variables respectively.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n_{ij}	Observed frequency of cell (*i*, *j*)
I	Dimension of the row variable, associated with independent variables
J	Dimension of the column variable, associated with dependent variables
w_{ij}	Weight of cell (*i*, *j*)
β_k	Coefficients in the loglinear model; $1 \leq k \leq p$
$\beta_k^{(l)}$	Estimate of β_k at the *l*th iteration
$\hat{\beta}_k$	Final estimate of β_k
m_{ij}	Expected values of n_{ij}
$m_{ij}^{(l)}$	Estimate of m_{ij} at the *l*th iteration
$\hat{m}_{ij}$	Estimate of m_{ij} at the final iteration
$\hat{M}_{i.}$	$\sum_{j=1}^{J} \hat{m}_{ij}$
$\hat{M}_{.j}$	$\sum_{i=1}^{I} \hat{m}_{ij}$
M	$\sum_{j=1}^{J} \sum_{i=1}^{I} \hat{m}_{ij}$

Model

In the general LOGLINEAR model, the logarithms of the cell frequencies are formulated as a linear function of the parameters. The actual form of the model is determined by the contrast and the effects specified. The model has the form

$$y_{ij} \equiv \ln\left(\frac{m_{ij}}{w_{ij}}\right) = \lambda_i + \sum_{k=1}^{p} \beta_k x_{ijk} \quad 1 \leq i \leq I, 1 \leq j \leq J$$

where λ_i are chosen so that $\sum_j m_{ij} = \sum_j n_{ij}$, and x_{ijk} are the independent variables in the linear model.

Contrasts

The values of x_{ijk} are determined by the types of contrasts specified in the procedure. The default contrast is DEVIATION.

Computational Algorithm

To estimate the coefficients, a series of weighted regressions is used for iterative calculations. The iterative process is outlined (also see Haberman, 1978) as follows:

(1) Obtain initial approximations $y_{ij}^{(0)}$ and use them to obtain $\beta_k^{(0)}$.

(2) Obtain the next approximations $y_{ij}^{(1)}$ and $m_{ij}^{(1)}$.

(3) Use the updated $y_{ij}^{(1)}$ in (2) to obtain the next approximations $\beta_k^{(1)}$.

(4) Repeat steps 2 and 3, replacing $\beta_k^{(l)}$ with $\beta_k^{(l+1)}$. Continue repeating this until convergence is achieved.

The computations begin with selection of initial approximations $m_{ij}^{(0)} = n_{ij} + \delta$ for m_{ij}. The default for δ is 0.5. If the model is saturated, δ is added to n_{ij} permanently. If the model is not saturated, δ is added to n_{ij} only at the initial step and is then subtracted at the second step.

The maximum likelihood estimates $\hat{\beta}_k$ of β_k are found by the Newton-Raphson method. Let $\beta^{(l)}$ be the column vector containing the ML estimates at the lth iteration; then

$$\beta^{(0)} = \left(C^{(0)}\right)^{-1} a^{(0)}$$

$$\beta^{(l+1)} = \beta^{(l)} + \left(C^{(l+1)}\right)^{-1} a^{(l+1)}, \quad \text{for } l \geq 0,$$

where the (k, l)-element of $C^{(l)}$ is

$$c_{kl}^{(l)} = \sum_{j=1}^{J} \sum_{i=1}^{I} \left(x_{ijk} - \theta_{ik}^{(l)}\right)\left(x_{ijl} - \theta_{il}^{(l)}\right) m_{ij}^{(l)}$$

with

$$\theta_{ik}^{(l)} = \frac{\sum_j m_{ij}^{(l)} x_{ijk}}{\sum_j m_{ij}^{(l)}} \quad \text{for } 1 \leq i \leq I, 1 \leq k \leq p$$

and the kth element of $a^{(0)}$ is

$$a_k^{(0)} = \sum_{i,j} x_{ijk} y_{ij}^{(0)} m_{ij}^{(0)} - \frac{\left(\sum_{i,j} x_{ijk} m_{ij}^{(0)}\right)\left(\sum_{i,j} y_{ij} m_{ij}^{(0)}\right)}{\sum_{i,j} m_{ij}^{(0)}}$$

and the *k*th element of $a^{(l)}$ is

$$a_k^{(l)} = \sum_{i,j} x_{ijk}\left(n_{ij} - m_{ij}^{(l)}\right) \quad \text{for } l \geq 1$$

The estimated cell means are updated by

$$m_{ij}^{(l)} = \frac{T w_{ij} \exp\left(v_{ij}^{(l-1)}\right)}{\sum_{i,j} w_{ij} \exp\left(v_{ij}^{(l-1)}\right)} \quad \text{for } l \geq 1$$

where

$$T = \begin{cases} \sum_{i,j} (n_{ij} + \delta) & \text{if the model is saturated} \\ \sum_{i,j} (n_{ij}) & \text{otherwise} \end{cases}$$

and

$$v_{ij}^{(l-1)} = \sum_{k=1}^{p} x_{ijk} \beta_k^{(l-1)}$$

The iterative process stops when either the maximum number of iterations (default=20) is reached or

$$\max_{i,j}\left|v_{ij}^{(l+1)} - v_{ij}^{(l)}\right| < \epsilon \quad \text{(with default } \epsilon = 0.001\text{)}.$$

Computed Statistics

The following output statistics are available.

Correlation Matrix of Parameter Estimates

Let *C* be the final $C^{(l)}$ and $H = C^{-1}$. The correlation between $\hat{\beta}_i$ and $\hat{\beta}_j$ is computed as

$$\frac{h_{ij}}{\sqrt{h_{ii} h_{jj}}}$$

Goodness of Fit

The Pearson chi-square is computed as

$$\chi^2 = \sum_{i,j} \frac{\left(n_{ij} - \hat{m}_{ij}\right)^2}{\hat{m}_{ij}}$$

and the likelihood-ratio chi-square is

$$L = 2\sum_{i,j} n_{ij} \ln\left(\frac{n_{ij}}{\hat{m}_{ij}}\right)$$

The degrees of freedom are $I \times (J-1) - p - E$, where E is the number of cells with $n_{ij}w_{ij} \leq 0$ and p is the number of coefficients in the model.

Residuals

The following residuals are available.

Unadjusted Residuals

$$residual_{ij} = n_{ij} - \hat{m}_{ij}$$

Standardized Residuals

$$standard\ residual_{ij} = \frac{n_{ij} - \hat{m}_{ij}}{\sqrt{\hat{m}_{ij}}}$$

Adjusted Residuals

$$adjusted\ residual_{ij} = \frac{n_{ij} - \hat{m}_{ij}}{\sqrt{s_{ij}}}$$

where

$$s_{ij} = \hat{m}_{ij}\left[1 - \frac{\hat{m}_{ij}}{T} - \hat{m}_{ij}\sum_{k,l}\left(x_{ijk} - \hat{\theta}_{ik}\right)\left(x_{ijl} - \hat{\theta}_{il}\right)h_{kl}\right]$$

$$\hat{\theta}_{ik} = \frac{\sum_{j} \hat{m}_{ij}x_{ijk}}{\sum_{j} \hat{m}_{ij}}$$

Generalized Residuals

Consider a linear combination of the cell counts

$$\sum_{i,j} d_{ij}n_{ij}$$

The estimated expected value is computed as

$$\sum_{i,j} d_{ij}\hat{m}_{ij}$$

Two generalized residuals are computed.

Unadjusted Residuals

$$residual = \sum_{i,j} d_{ij}n_{ij} - \sum_{i,j} d_{ij}\hat{m}_{ij}$$

Adjusted Residuals

$$adjusted\ residual = \frac{\sum_{i,j} d_{ij} n_{ij} - \sum_{i,j} d_{ij} \hat{m}_{ij}}{\sqrt{C_1}}$$

where

$$C_1 = \sum_{i,j} d_{ij}^2 \hat{m}_{ij} - \sum_i \left[\frac{\left(\sum_j \hat{m}_{ij} d_{ij} \right)^2}{\sum_j \hat{m}_{ij}} \right] - \sum_{k=1}^{p} \sum_{l=1}^{p} f_k f_l h_{kl}$$

$$f_k = \sum_{i,j} \hat{m}_{ij} d_{ij} \left(x_{ijk} - \hat{\theta}_{ik} \right)$$

Analysis of Dispersion

Following Haberman (1982), define

$S(Y)$ = Total dispersion

$S(Y|X)$ = Conditional dispersion

$S(X)$ = Dispersion due to fit

$R = \frac{S(X)}{S(Y)}$ = Measure of association

For entropy

$$S(Y) = -M \sum_{j=1}^{J} \hat{p}_j \ln(\hat{p}_j)$$

$$S(Y|X) = -\sum_{i=1}^{I} \hat{M}_{i\bullet} \sum_{j=1}^{J} \hat{p}_{i|j} \ln(\hat{p}_{i|j})$$

$$S(X) = S(Y) - S(Y|X)$$

For concentration

$$S(Y) = M \times \left(1 - \sum_{j=1}^{J} \hat{p}_j^2 \right)$$

$$S(Y|X) = \sum_{i=1}^{I} \hat{M}_{i\bullet} \left(1 - \sum_{j=1}^{J} \hat{p}_{i|j}^2 \right)$$

$$S(X) = S(Y) - S(Y|X)$$

where

$\hat{p}_j = \frac{\hat{M}_{\bullet j}}{M}$

$\hat{p}_{j|i} = \frac{\hat{m}_{ij}}{\hat{M}_{i\bullet}}$

Haberman (1977) shows that, under the hypothesis that *Y* and *X* are independent,

$$\psi_E = 2S(X) \rightarrow \chi^2_{I(J-1)}$$

in the case of entropy, and

$$\psi_C = \frac{M(J-1)S(X)}{S(Y)} \rightarrow \chi^2_{I-1}$$

in the case of concentration.

References

Haberman, S. J. 1977. Maximum likelihood estimates in exponential response models. *Annals of Statistics*, 5, 815–841.

Haberman, S. J. 1978. *Analysis of qualitative data*. London: Academic Press.

Haberman, S. J. 1982. Analysis of dispersion of multinomial responses. *Journal of the American Statistical Association*, 77 , 568–580.

MANOVA Algorithms

The program performs univariate and multivariate analysis of variance and covariance for any crossed and/or nested design.

Analysis of Variance

The following topics detail computational formulas for MANOVA's use in the analysis of variance.

Notation

The experimental design model (the model with covariates will be discussed later) can be expressed as

$$\underset{N\times p}{\mathbf{Y}} = \underset{N\times m}{\mathbf{W}} \; \underset{m\times p}{\beta} + \underset{N\times p}{\mathbf{E}}$$

where

$\mathbf{Y}$	is the observed matrix
$\mathbf{W}$	is the design matrix
β	is the matrix of parameters
$\mathbf{E}$	is the matrix of random errors
N	is the total number of observations
p	is the number of dependent variables
m	is the number of parameters

Since the rows of **W** will be identical for all observations in the same cell, the model is rewritten in terms of cell means as

$$\underset{g\times p}{\mathbf{Y}_\bullet} = \underset{g\times m}{\mathbf{A}} \; \underset{m\times p}{\beta} + \underset{g\times p}{\mathbf{E}_\bullet}$$

where g is the number of cells and $\mathbf{Y}_\bullet$ and $\mathbf{E}_\bullet$ denote matrices of means.

Reparameterization

The reparameterization of the model (Bock, 1975; Finn, 1977) is done by factoring **A** into

$$\underset{g\times m}{\mathbf{A}} = \underset{g\times r}{\mathbf{K}} \; \underset{r\times m}{\mathbf{L}}$$

K forms a column basis for the model and has rank r. The contrast matrix **L** contains the coefficients of linear combinations of parameters and has rank r. **L** can be specified by the user. Given **L**, **K** can be obtained from $\mathbf{AL}'(\mathbf{LL}')^{-1}$. For designs with more than one factor, **L**, and hence **K**, can be constructed from Kronecker products of contrast matrices of each factor. After reparameterization, the model can be expressed as

$$\underset{g\times p}{\mathbf{Y}} = \mathbf{A}\beta + \mathbf{E}$$
$$= \mathbf{K}(\mathbf{L}\beta) + \mathbf{E}$$
$$= \underset{g\times r}{\mathbf{K}} \quad \underset{r\times p}{\mathrm{q}} \quad + \quad \underset{g\times p}{\mathbf{E}}$$

Parameter Estimation

An orthogonal decomposition (Golub, 1969) is performed on **K**. That is, **K** is represented as

$$\mathbf{K} = \mathbf{Q}\mathbf{R}$$

where **Q** is an orthonormal matrix such that $\mathbf{Q}'\mathbf{D}\mathbf{Q} = \mathbf{I}$; **D** is the diagonal matrix of cell frequencies; and **R** is an upper-triangular matrix.

The normal equation of the model is

$$\left(\mathbf{K}'\mathbf{D}\mathbf{K}\right)\hat{\theta} = \mathbf{K}'\mathbf{D}\mathbf{Y}$$

or

$$\mathbf{R}\hat{\theta} = \mathbf{Q}'\mathbf{D}\mathbf{Y} = \mathbf{U}$$

This triangular system can therefore be solved forming the cross-product matrix.

Significance Tests

The sum of squares and cross-products (SSCP) matrix due to the model is

$$\hat{\theta}'\mathbf{R}'\mathbf{R}\hat{\theta} = \mathbf{U}'\mathbf{U}$$

and since $var(\mathbf{U}) = \mathbf{R}var(\theta)\mathbf{R}' = \mathbf{I}\otimes$ S the SSCP matrix of each individual effect can be obtained from the components of

$$\mathbf{U}'\mathbf{U} = (U_1, \ldots, U_k)\begin{pmatrix} U'_1 \\ \vdots \\ U'_k \end{pmatrix} = U_1U'_1 + \ldots + U_kU'_k$$

Therefore the hypothesis SSCP matrix for testing $H_o : \theta_h = \mathbf{0}$ is

$$\underset{p\times p}{\mathbf{S}_H} = \underset{p\times n_h}{\mathbf{U}_h} \; \underset{n_h\times p}{\mathbf{U}'_h}$$

The default error SSCP matrix is the pooled within-groups SSCP:

$$\mathbf{S}_E = \mathbf{Y}'\mathbf{Y} - \mathbf{Y}'\mathbf{D}\mathbf{Y}$$

if the pooled within-groups SSCP matrix does not exist, the residual SSCP matrix is used:

$$\mathbf{S}_E = \mathbf{Y}'\mathbf{Y} - \mathbf{U}'\mathbf{U}$$

Four test criteria are available. Each of these statistics is a function of the nonzero eigenvalues λ_i of the matrix $\mathbf{S}_H\mathbf{S}_E^{-1}$. The number of nonzero eigenvalues, *s*, is equal to $\min(p, n_h)$.

Pillai's Criterion (Pillai, 1967)

$$T = \sum_{i=1}^{s} \lambda_i/(1+\lambda_i)$$

Approximate $F = (n_e - p - s)T/(b(s-T))$ with b_s and $s(n_e - p + s)$ degrees of freedom, where

$n_e =$ degrees of freedom S_E
$b = \max(p, n_h)$

Hotelling's Trace

$$T = \sum_{i=1}^{s} \lambda_i$$

Approximate $F = 2(sn+1)T/\left(s^2(2m+s+1)\right)$ with $s(2m+s+1)$ and $2(sn+1)$ degrees of freedom where

$m = (|n_h - p| - 1)/2$
$n = (n_e - p - 1)/2$

Wilks' Lambda (Rao, 1973)

$$T = \prod_{i=1}^{s} 1/(1+\lambda_i)$$

Approximate $F = \left(1 - T^{1/l}\right)(Ml + 1 - n_h p/2)/\left(T^{1/l} n_h p\right)$ with $n_h p$ and $(Ml + 1 - n_h p/2)$ degrees of freedom, where

$l^2 = \left(p^2 n_h^2 - 4\right)/\left(p^2 + n_h^2 - 5\right)$
$M = n_e - (p + 1 - n_h)/2$

Roy's Largest Root

$$T = \lambda_1/(1+\lambda_1)$$

Stepdown F Tests

The stepdown *F* statistics are

$$F_i = \frac{\left(t^2 - t_e^2\right)/n_h}{t_e^2/(n_e - i + 1)}$$

with n_h and $n_e - i + 1$ degrees of freedom, where t_e and *t* are the *i*th diagonal element of $\mathbf{T}_E$ and $\mathbf{T}$ respectively, and where

$\mathbf{S}_E = \mathbf{T}'_E\mathbf{T}_E$
$\mathbf{S}_E + \mathbf{S}_H = \mathbf{T}'\mathbf{T}$

Design Matrix

$$\mathbf{K}$$

Estimated Cell Means

$$\hat{\mathbf{Y}}_{\bullet} = \mathbf{K}\hat{\theta}$$

Analysis of Covariance

$$\underset{g \times p}{\mathbf{Y}_{\bullet}} = \underset{g \times r}{\mathbf{K}} \; \underset{r \times p}{\theta} + \underset{g \times q}{\mathbf{X}_{\bullet}} \; \underset{q \times p}{\mathbf{B}} + \underset{g \times p}{\mathbf{E}_{\bullet}}$$

where *g*, *p*, and *r* are as before and *q* is the number of covariates, and $\mathbf{X}_{\bullet}$ is the mean of **X**, the matrix of covariates.

Parameter Estimation and Significance Tests

For purposes of parameter estimation, no initial distinction is made between dependent variables and covariates.

Let

$$\mathbf{V} = (\mathbf{Y}\mathbf{X})$$
$$\mathbf{V}_{\bullet} = (\mathbf{Y}_{\bullet}\mathbf{X}_{\bullet})$$

The normal equation of the model

$$\underset{g \times (p+q)}{\mathbf{V}_{\bullet}} = \underset{g \times r}{\mathbf{K}} \; \underset{r \times (p+q)}{\theta} + \underset{g \times (p+q)}{\mathbf{E}_{\bullet}}$$

is

$$\left(\mathbf{K}'\mathbf{D}\mathbf{K}\right)\hat{\theta} = \mathbf{K}'\mathbf{D}\mathbf{Y}_{\bullet}$$

or

$$\mathbf{R}\hat{\theta} = \mathbf{Q}'\mathbf{D}\mathbf{V}_{\bullet} = \mathbf{U}$$

or

$$\underset{r \times (p+q)}{\hat{\theta}} = \left(\underset{r \times p}{\hat{\theta}_Y} \quad \underset{r \times q}{\hat{\theta}_X}\right)$$

If $\mathbf{S}_E$ and $\mathbf{S}_T$ are partitioned as

$$\mathbf{S}_E = \begin{pmatrix} \mathbf{S}_E^{(Y)} & \mathbf{S}_E^{(YX)} \\ \mathbf{S}_E^{(XY)} & \mathbf{S}_E^{(X)} \end{pmatrix}$$

$$\mathbf{S}_T = \begin{pmatrix} \mathbf{S}_T^{(Y)} & \mathbf{S}_T^{(YX)} \\ \mathbf{S}_T^{(XY)} & \mathbf{S}_T^{(X)} \end{pmatrix}$$

then the adjusted error SSCP matrix is

$$\mathbf{S}_E^* = \mathbf{S}_E^{(Y)} - \mathbf{S}_E^{(YX)}\left(\mathbf{S}_E^{(X)}\right)^{-1}\mathbf{S}_E^{(XY)}$$

and the adjusted total SSCP matrix is

$$\mathbf{S}_T^* = \mathbf{S}_T^{(Y)} - \mathbf{S}_T^{(YX)}\left(\mathbf{S}_T^{(X)}\right)^{-1}\mathbf{S}_T^{(XY)}$$

The adjusted hypothesis SSCP matrix is then

$$\mathbf{S}_H^* = \mathbf{S}_T^* - \mathbf{S}_E^*$$

The estimate of **B** is

$$\hat{\mathbf{B}} = \left(\mathbf{S}_T^{(X)}\right)^{-1}\mathbf{S}_T^{(XY)}$$

The adjusted parameter estimates are

$$\hat{\theta}^* = \hat{\theta}_Y - \hat{\theta}_X\hat{\mathbf{B}}$$

The adjusted cell means are

$$\hat{\mathbf{Y}}^* = \mathbf{K}\hat{\theta}^*$$

Repeated Measures

The following topics detail computational formulas for MANOVA's use in the analysis of repeated measures data.

Notation

The following notation is used within this section unless otherwise stated:

k	Degrees of freedom for the within-subject
$\mathbf{SSE}^*$	factor Orthonormal transformed error matrix
N	Total number of observations
ndfb	Degrees of freedom for all between-subject factors (including the constant)

Statistics

The following statistics are available.

Greenhouse-Geisser Epsilon

$$\text{ggeps} = \frac{\left(\text{tr}(\mathbf{SSE}^*)\right)^2}{k\times\text{tr}\left((\mathbf{SSE}^*)^2\right)}$$

Huynh-Feldt Epsilon

$$\text{hfeps} = \frac{N \times k \times \text{ggeps} - 2}{k \times (N - \text{ndfb}) - k^2 \times \text{ggeps}}$$

if hfeps>1, set hfeps=1

Lower Bound Epsilon

$$\text{lbeps}=\frac{1}{k}$$

Effect Size

The effect size gives a partial eta-squared value for each effect and parameter estimate

Notation

The following notation is used within this section unless otherwise stated:

dfh	Hypothesis degrees of freedom
dfe	Error degrees of freedom
F	F test
W	Wilks' lambda
s	Number of non-zero eigenvalues of $\mathbf{HE}^{-1}$
T	Hotelling's trace
V	Pillai's trace

Statistic

$$\text{Partial eta-squared} = \frac{dfh \times F}{dfh \times F + dfe} = \frac{\text{SS hyp}}{\text{SS hyp+SS error}}$$

$$\text{Eta} - \text{squared(Wilks')} = 1 - W^{1/s}$$

$$\text{Eta} - \text{squared(Hotelling's)} = \frac{T/s}{T/s+1}$$

$$\text{Total eta-squared} = \frac{\text{sum of squares for effect}}{\text{total(corrected)sum of squares}}$$

$$\text{Hay's omega-squared} = \frac{\text{SS for effect} - \text{df(effect)} \times \text{MSE}}{\text{corrected total SS} + \text{MSE}}$$

$$\text{Pillai} = V/S$$

Power

The following statistics pertain to the observed power of *F* and *t* tests performed by the procedure.

Univariate Non-Centrality

$$\lambda = \frac{\text{SS hyp}}{\text{SS error}} \times dfe$$

Multivariate Non-Centrality

For a single degree of freedom hypothesis

$$\lambda = T \times dfe$$

where *T* is Hotelling’s trace and *dfe* is the error degrees of freedom. Approximate power non-centrality based on Wilks’ lambda is

$$\lambda = \frac{\text{Wilks' eta square}}{1-\text{Wilks' eta square}} \times dfe(W)$$

where $dfe(W)$ is the error *df* from Rao’s *F*-approximation to the distribution of Wilks’ lambda.

Hotelling’s Trace

$$\lambda = \frac{\text{Hotelling's eta square}}{1-\text{Hotelling's eta square}} \times dfe(H)$$

where $dfe(H)$ is the error *df* from the *F*-approximation to the distribution of Hotelling’s trace.

Pillai’s Trace

$$\lambda = \frac{\text{Pillai's eta square}}{1-\text{Pillai's eta square}} \times dfe(P)$$

where $dfe(P)$ is the error *df* from the *F*-approximation to the distribution of Pillai’s trace.

Approximate Power

Approximate power is computed using an Edgeworth Series Expansion (Mudholkar, Chaubey, and Lin, 1976).

$$r = \upsilon_1 + \lambda$$
$$b = \lambda / r$$

$$K_1 = \left\{ \left(\frac{r}{\upsilon_1}\right)^{1/3} \left(1 - \frac{2(b+1)}{9r} - \frac{40b^2}{3^4 r^2} + \frac{80\left(1+3b+33b^2-77b^3\right)}{3^7 r^3} + \frac{176\left(1+4b-210b^2+2380b^3-2975b^4\right)}{3^9 r^4}\right)\right\}$$
$$-c^{1/3}\left\{\left(1 - \frac{2}{9\upsilon_2} + \frac{80}{3^7 \upsilon_2^3} + \frac{176}{3^9 \upsilon_2^4}\right)\right\}$$

$$K_2 = \left\{ \left(\frac{r}{\upsilon_1}\right)^{2/3} \left(\frac{2(b+1)}{9r} + \frac{16b^2}{3^3 r^2} - \frac{8\left(13+39b+405b^2-1025b^3\right)}{3^7 r^3} + \frac{160\left(1+4b-87b^2+1168b^3-1544b^4\right)}{3^8 r^4}\right)\right\}$$
$$+c^{2/3}\left(\frac{2}{9\upsilon_2} - \frac{104}{3^7 \upsilon_2^3} - \frac{160}{3^8 \upsilon_2^4}\right)$$

$$K_3 = \left\{ \left(\frac{-r}{\upsilon_1}\right) \left(\frac{8b^2}{27r^2} - \frac{32\left(1+3b+21b^2-62b^3\right)}{3^6 r^3} - \frac{32\left(8+32b-177b^2+4550b^3-6625b^4\right)}{3^8 r^4}\right)\right\}$$
$$-c\left(\frac{32}{3^6 \upsilon_2^3} + \frac{256}{3^8 \upsilon_2^4}\right)$$

$$K_4 = \left\{ \left(\frac{r}{\upsilon_1}\right)^{4/3} \left(\frac{16\left(1+3b+12b^2-44b^3\right)}{3^6 r^3} + \frac{256\left(1+4b+6b^2+247b^3-458b^4\right)}{3^8 r^4} \right) \right\}$$
$$-c^{4/3}\left(\frac{16}{3^6\upsilon_2^3} + \frac{256}{3^8\upsilon_2^4}\right)$$

$$Y = \frac{K_1}{\sqrt{K_2}}$$

$$\text{Power} = 1 - \Phi(Y) - \frac{1}{\sqrt{2\pi}} e^{-Y^2/2} \left\{ \frac{K_3}{6}\left(Y^2-1\right) + \frac{K_4}{24}\left(Y^3-3Y\right) \frac{K_1^2}{72}\left(Y^5-10Y^3+15Y\right) \right\}$$

Confidence Intervals

The intervals are calculated as follows:

Lower bound = parameter estimate −*k* * stderr

Upper bound = parameter estimate + *k* * stderr

where stderr is the standard error of the parameter estimate, and *k* is the critical constant whose value depends upon the type of confidence interval requested.

Univariate Individual Confidence Intervals

$$k = \sqrt{(F(a;1,ne))}$$

where

ne is the error degrees of freedom

a is the confidence level desired

F is the percentage point of the *F* distribution

Univariate Intervals Joint Confidence Intervals

For Scheffé intervals:

$$k = \sqrt{(nh * F(a;nh,ne))}$$

where

ne is the error degrees of freedom

nh is the hypothesis degrees of freedom

a is the confidence level desired

F is the percentage point of the *F* distribution

For Bonferroni intervals:

$k = t(a/(2*nh), ne)$

where

ne is the error degrees of freedom

nh is the hypothesis degrees of freedom

a is 100 minus the confidence level desired

F is the percentage point of Student's *t* distribution

Multivariate Intervals

The value of the multipliers for the multivariate case is computed as follows:

Let

$p =$ the number of dependent variables
$nh =$ the hypothesis degrees of freedom
$ne =$ the error degrees of freedom
$a =$ the desired confidence level
$s = \min(p, nh)$
$m = (|nh - p| - 1)/2$
$n = (ne - p - 1)/2$

For Roy's largest root, define

$c = G/(1 - G)$

where

G = GCR$(a; s, m, n)$; the percentage point of the largest root distribution

For Wilks' lambda, define

$t = (p*nh)^2 - 4$
$b = p*p + nh*nh - 5$
$r = \sqrt{(t/b)}$ if $b \neq 0$, else $r = 1$
$u = (p*nh - 2)/4$
$t = p*nh$
$b = (nh + ne - (p + nh + 1)/2)*r - 2*u$
$f = (t*F(a; t, b))/b$
$W = (1/(1 + c))^r$
$c = (1 - W)/W$

For Hotelling's trace, define

$t = s(2m + s + 1)$
$b = 2(sn + 1)$
$T = (stF(a; t, b))/b$
$c = T$

For Pillai’s trace, define

$t = s(\max(p, nh))$
$b = s(ne - p + s)$
$D = (F(a; t, b)t)/b$
$V = (sc)/(c + 1)$
$c = V/(1 - V)$

Now for each of the above criteria, the critical value is

$$K = \sqrt{(ne * c)}$$

For Bonferroni intervals,

$$K = t(a/(2p(nh)); ne)$$

where *t* is the percentage point of the Student’s *t* distribution.

Regression Statistics

Correlation between independent variables and predicted dependent variables

$$r\left(X_i, \hat{Y}_j\right) = \frac{r_{ij}}{R_j}$$

where

$X_i = i$ th predictor (covariate)
$\hat{Y}_j = j$ th predicted dependent variable
$r_{ij} =$ correlation between i th predictor and j th dependent variable
$R_j =$ multiple R for j th dependent variable across all predictors

References

Bock, R. D. 1975. *Multivariate statistical methods in behavioral research*. New York: McGraw-Hill.

Finn, J. D. 1977. Multivariate analysis of variance and covariance. In: *Statistical Methods for Digital Computers, Volume 3,* K. Enslein, A. Ralston, and H. Wilf, eds. New York: John Wiley & Sons, Inc.

Golub, G. H. 1969. Matrix decompositions and statistical calculations. In: *Statistical Computation,* R. C. Milton, and J. A. Nelder, eds. New York: Academic Press.

Green, P. E. 1978. *Analyzing multivariate data*. Hinsdale, Ill.: The Dryden Press.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Some Approximations for the noncentral F-distribution. *Technometrics*, 18, 351–358.

Muller, K. E., and B. L. Peterson. 1984. Practical methods for computing power in testing the multivariate general linear hypothesis. *Computational Statistics and Data Analysis*, 2, 143–158.

Pillai, K. C. S. 1967. Upper percentage points of the largest root of a matrix in multivariate analysis. *Biometrika*, 54, 189–194.

Timm, N. H. 1975. *Multivariate statistics: With applications in education and psychology*. Monterey, California: Brooks/Cole.

MEANS Algorithms

Cases are cross-classified on the basis of multiple independent variables, and for each cell of the resulting cross-classification, basic statistics are calculated for a dependent variable.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 58-1
Notation

Notation	Description
X_{ip}	Value for the *p*th independent variable for case *i*
Y_i	Value for the dependent variable for case *i*
w_i	Weight for case *i*
P	Number of independent variables
N	Number of cases

Statistics

For each value of the first independent variable (X_1), for each value of the pair (X_1, X_2), for the triple (X_1, X_2, X_3), and similarly for the *P*-tuple $(X_1, X_2, \ldots, X_P)$, the following are computed:

Sum of Case Weights for the Cell

$$W = \sum_{i=1}^{N} w_i l_i$$

where $l_i = 1$ if the *i*th case is in the cell, $l_i = 0$ otherwise.

The Sum and Corrected Sum of Squares

$$SMY = \sum_{i=1}^{N} w_i l_i Y_i$$

$$SSY = \sum_{i=1}^{N} w_i l_i Y_i^2$$

$$CSS = SSY - SMY^2/W$$

The Mean

$$\overline{Y} = \frac{\sum_{i=1}^{N} w_i l_i Y_i}{W}$$

Harmonic mean

$$\overline{Y}_h = \frac{\sum_{i=1}^{N} w_i}{\sum_{i=1}^{N} w_i y_i^{-1}}$$

Both summations are over cases with positive w_i values.

Geometric mean

$$\overline{Y}_g = \left(\prod_{i=1}^{N} y_i^{w_i} \right)^{1/W}$$

The product is taken over cases with positive w_i values.

Variance

$$S^2 = \frac{CSS}{W-1}$$

Standard Deviation

$$S = \sqrt{\text{variance}}$$

Standard Error of the Mean

$$SEM = \frac{S}{\sqrt{W}}$$

Skewness (computed if W ≥ 3 and S > 0) , and its standard error

$$g_1 = \frac{WM_3}{(W-1)(W-2)S^3} \quad se(g_1) = \sqrt{\frac{6W(W-1)}{(W-2)(W+1)(W+3)}}$$

Kurtosis (computed if W ≥ 4 and S > 0), and its standard error

$$g_2 = \frac{W(W+1)M_4 - 3(W-1)M_2^2}{(W-1)(W-2)(W-3)S^4} \quad se(g_2) = \sqrt{\frac{4(W^2-1)se(g_1)^2}{(W-3)(W+5)}}$$

Minimum

$$\min_i X_i$$

Maximum

$$\max_i X_i$$

Range

Maximum – Minimum

Percent of Total N

For each category *j* of the independent variable,

$$\%TotN_j = \left(\frac{\sum_{i=1}^{N} w_i l_i}{W} \right) \times 100$$

where $l_i = 1$ if the *i*th case is in the *j*th category, $l_i = 0$ otherwise.

Percent of Total Sum

For each category *j* of the independent variable,

$$\%TotSum_j = \left(\frac{\sum_{i=1}^{N} w_i l_i Y_i}{W} \right) \times 100$$

where $l_i = 1$ if the *i*th case is in the *j*th category, $l_i = 0$ otherwise.

Median

Find the first score interval (*x2*) containing more than *t* cases.

$$\text{median} = \begin{cases} x_2 & \text{if } t - cp_1 \geq 100/W \\ \{1 - [(W+1)/2 - cc_1]\}x_1 \\ \quad + [(W+1)/2 \quad cc_1]x_2 & \text{if } t - cp_1 < 100/W \end{cases}$$

where

$t = (W+1)/2$
$cp_1 < t < cp_2$
x_1 and x_2 are the values corresponding to cp_1 and cp_2, respectively
cc_1 is the cumulative frequency up to x_1
$cp_1$is the cumulative percent up tox_1

Grouped Median

For more information, see the topic “Grouped Percentiles”.

ANOVA and Test for Linearity

If the analysis of variance table or test for linearity are requested, only the first independent variable is used. Assume it takes on *J* distinct values (groups). The previously described statistics are calculated and printed for each group separately, as well as for all cases pooled. Symbols subscripted from 1 to *J* will denote group statistics, unsubscripted the total. Thus for group *j*,

- SMY_j is the sum of the dependent variable.

and

- X_j the value of the independent variable. Note that the standard deviation and sum of squares printed in the last row of the summary table are pooled within group values.

Analysis of Variance

Source	**Sum of Squares**	**df**
Between Groups	Total-Within Groups	$J-1$
Regression	$\dfrac{\left(\sum_{j=1}^{J} X_j SMY_j - \left(\sum_{j=1}^{J} w_j X_j\right)\left(\sum_{j=1}^{J} SMY_j\right)/W\right)^2}{\sum_{j=1}^{J} w_j X_j^2 - \left(\sum_{j=1}^{J} w_j X_j\right)^2/W}$	1
Deviation from Regression	Between-Regression	$J-2$

Source	**Sum of Squares**	**df**
Within Groups	$\sum_{j=1}^{J} CSS_j$	$W-J$
Total	$\sum_{j=1}^{J} SSY_j - \left(\sum_{j=1}^{J} SMY_j\right)^2 / W$	$W-1$

The mean squares are calculated by dividing each sum of squares by its degrees of freedom. The *F* ratios are the mean squares for each source divided by the within groups mean square. The significance level for the *F* is from the *F* distribution with the degrees of freedom for the numerator and denominator mean squares. If there is only one group the ANOVA is not done; if there are fewer than three groups or the independent variable is a string variable, the test for linearity is not done.

Correlation Coefficient

$$r = \frac{\sum_{j=1}^{J} X_j SMY_j - \left(\sum_{j=1}^{J} W_j X_j\right) SMY/W}{\sqrt{\left(\sum_{j=1}^{J} W_j X_j^2 - \left(\sum_{j=1}^{J} W_j X_j\right)^2 / W\right)(SSY - SMY^2/W)}}$$

Eta

$$(eta)^2 = \frac{\text{Sum of Squares Between Groups}}{\text{Total Sum of Squares}}$$

References

Blalock, H. M. 1972. *Social statistics*. New York: McGraw-Hill.

Bliss, C. I. 1967. *Statistics in biology, Volume 1*. New York: McGraw-Hill.

Hays, W. L. 1973. *Statistics for the social sciences*. New York: Holt, Rinehart, and Winston.

MIXED Algorithms

This document summarizes the computational algorithms for the linear mixed model (Wolfinger, Tobias, and Sall, 1994).

Notation

The following notation is used throughout this chapter unless otherwise stated:

θ	Overall covariance parameter vector
θ_G	A vector of covariance parameters associated with random effects.
θ_k	A vector of covariance parameters associated with the *k*th random effect.
θ_R	A vector of covariance parameters associated with the residual term.
K	Number of random effects.
S_r	Number of repeated subjects.
S_k	Number of subjects in *k*th random effect.
$\mathbf{V}$	The $n \times n$ covariance matrix of $\mathbf{y}$. This matrix is sometimes denoted by $V_y(\theta)$.
$\frac{\partial}{\partial \theta_s}\mathbf{V}$	First derivative of $\mathbf{V}$ with respect to the *s*th parameter in θ.
$\frac{\partial^2}{\partial \theta_s \partial \theta_t}\mathbf{V}$	Second derivative of $\mathbf{V}$ with respect to the *s*th and *t*th parameters in θ.
R	The $n \times n$ covariance matrix of $\boldsymbol{\varepsilon}$. This matrix is sometimes denoted by $\mathbf{R}(\theta_\mathbf{R})$
$\frac{\partial}{\partial \theta_s}\mathbf{R}$	First derivative of R with respect to the *s*th parameter in θ_R.
$\frac{\partial^2}{\partial \theta_s \partial \theta_t}\mathbf{R}$	Second derivative of R with respect to the *s*th and *t*th parameters in θ_R.
$\mathbf{G}$	The covariance matrix of random effects. This matrix is sometimes denoted by $\mathbf{G}(\theta_\mathbf{G})$
$\frac{\partial}{\partial \theta_s}\mathbf{G}$	First derivative of G with respect to the *s*th parameter in θ_G.
$\frac{\partial^2}{\partial \theta_s \partial \theta_t}\mathbf{G}$	Second derivative of G with respect to the *s*th and *t*th parameters in θ_G.
$\mathbf{V}_k$	The covariance matrix of the *k*th random effect for *one random subject*. This matrix is sometimes denoted by $\mathbf{V}_k(\theta_k)$.
$\frac{\partial}{\partial \theta_s}\mathbf{V}_k$	First derivative of $\mathbf{V}_k$ with respect to the *s*th parameter in θ_k.
$\frac{\partial^2}{\partial \theta_s \partial \theta_t}\mathbf{V}_k$	Second derivative of $\mathbf{V}_k$ with respect to the *s*th and *t*th parameters in θ_k.
$\mathbf{y}$	$n \times 1$ vector of dependent variable.
$\mathbf{X}$	$n \times p$ design matrix of fixed effects.
$\mathbf{Z}$	$n \times q$ design matrix of random effects.
$\mathbf{r}$	$n \times 1$ vector of residuals.
$\boldsymbol{\beta}$	$p \times 1$ vector of fixed effects parameters.
$\boldsymbol{\gamma}$	$q \times 1$ vector of random effects parameters.
$\boldsymbol{\varepsilon}$	$n \times 1$ vector of residual error.

$\mathbf{W}_c$ $n \times n$ diagonal matrix of case weights.

$\mathbf{W}_{rw}$ $n \times n$ diagonal matrix of regression weights.

Model

In this document, we assume a mixed effect model of the form

$$\mathbf{y} = \mathbf{X}\beta + \mathbf{Z}\gamma + \epsilon$$

In this model, we assume that **ε** is distributed as $N[\mathbf{0}, \mathbf{R}]$ and **γ** is independently distributed as $N[\mathbf{0}, \mathbf{G}]$. Therefore **y** is distributed as $N[\mathbf{X}\beta, \mathbf{V}]$, where $\mathbf{V} = \mathbf{ZGZ^T} + \mathbf{R}$. The unknown parameters include the regression parameters in **β** and covariance parameters in **θ**. Estimation of these model parameters relies on the use of a Newton-Ralphson or scoring algorithm. When we use either algorithm for finding MLE or REML solutions, we need to compute $\mathbf{V}^{-1}$ and its derivatives with respect to **θ**, which are computationally infeasible for large *n*. Wolfinger et.al.(1994) discussed methods that can avoid direct computation of $\mathbf{V}^{-1}$. They tackled the problem by using the SWEEP algorithm and exploiting the block diagonal structures of **G** and **R**. In the first half of this document, we will detail the algorithm for mixed models without subject blocking. In second half of the document we will refine the algorithm to exploit the structure of **G**; this is the actual implementation of the algorithm.

If there are regression weights, the covariance matrix $\mathbf{R}$ will be replaced by $\mathbf{R}^* = \mathbf{W_{rw}^{-1/2}RW_{rw}^{-1/2}}$. For simpler notations, we will assume that the weights are already included in the matrix $\mathbf{R}$ and they will not be displayed in the remainder of this document. When case weights are specified, they will be rounded to nearest integer and each case will be entered into the analysis multiple times depending on the rounded case weight. Since replicating a case will lead to duplicate repeated measures (Note: repeated measures are unique within a repeated subject), non-unity case weights will only be allowed for $\mathbf{R}$ with scaled identity structure. In MIXED, only cases with positive case weight and regression weight will be included analysis.

Fixed Effects Parameterization

The parameterization of fixed effects is the same as in the GLM procedure.

Random Effects Parameterization

If we have *K* random effects and there are S_k random subjects in *k*th random effect, the design matrix **Z** will be partitioned as

$$\mathbf{Z} = [\,\mathbf{Z}_1 \quad \mathbf{Z}_2 \quad ... \quad \mathbf{Z}_K\,]$$

where $\mathbf{Z_k}$ is the design matrix of the *k*th random effect. Each $\mathbf{Z_k}$ can be partitioned further by random subjects as below,

$$\mathbf{Zk} = [\,\mathbf{Z_{k1}} \quad \mathbf{Z_{k2}} \quad ... \quad \mathbf{Z_{kSk}}\,],\ k=1,..,K$$

The number of columns in the design matrix $\mathbf{Z_{kj}}$ (*j*th random subject of *k*th random effect) is equal to number of levels of the *k*th random effect variable.

Under this partition, the $\mathbf{G}$ will be a block diagonal matrix which can be expressed as

$$\mathbf{G} = \oplus_{k=1}^{K}[\mathbf{I_{Sk}} \otimes \mathbf{Vk}]$$

It should also be noted that each random effect has its own parameter vector θ_k, *k*=1,...,*K*, and there are no functional constraints between elements in these parameter vectors. Thus $\theta_{\mathbf{G}} = (\theta_1, ..., \theta_K)$.

When there are correlated random effects, $\mathbf{Z_{kj}}$ will be a combined design matrix of the correlated random effects. Therefore in subsequent sections, each random effect can either be one single random effect or a set of correlated random effects.

Repeated Subjects

When the REPEATED subcommand is used, $\mathbf{R}$ will be a block diagonal matrix where the *i*th block is $\mathbf{R}_i$, S_R. That is,

$$\mathbf{R} = \oplus_{i=1}^{S_R}\mathbf{R}_i$$

The dimension of $\mathbf{R}_i$ will be equal to number of cases in one repeated subject but all $\mathbf{R}_i$ share the same parameter vector $\theta_{\mathbf{R}}$.

Likelihood Functions

Recall that the –2 times log likelihood of the MLE is

$$-2l_{MLE}(\beta, \theta) = \log|\mathbf{V}| + \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} + n\log 2\pi$$

and the –2 times log likelihood of the REML is

$$-2l_{MLE}(\theta) = \log|\mathbf{V}| + \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} + \log\left|\mathbf{X}^{\prime}\mathbf{V}^{-1}\mathbf{X}\right| + (n-p)\log 2\pi$$

where *n* is the number of observations and *p* is the rank of fixed effects design matrix. The key components of the likelihood functions are

$$l_1(\theta) = \log|\mathbf{V}|$$
$$l_2(\theta) = \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r}$$
$$l_3(\theta) = \log\left|\mathbf{X}^{\prime}\mathbf{V}^{-1}\mathbf{X}\right|$$

Therefore, in each estimation iteration, we need to compute $l_1(\theta)$, $l_2(\theta)$ and $l_3(\theta)$ as well as their 1st and 2nd derivatives with respective to θ.

Newton & Scoring Algorithms

Covariance parameters in θ can be found by maximizing the MLE or REML log-likelihood; however, there are no closed form solutions in general. Therefore Newton and scoring algorithms are used to find the solution numerically. The algorithm is outlined as below,

1. Compute starting value and initial log-likelihood (REML or ML).
2. Compute gradient vector **g** and Hessian matrix **H** of the log-likelihood function using last iteration's estimate θ_{i-1}. (See later section for computation of **g** and $\mathbf{H}$)
3. Compute the new step $\mathbf{d} = -\mathbf{H}^{-1}\mathbf{g}$.
4. Let $\rho = 1$.
5. Compute estimates of *i*th iteration $\theta_\mathbf{i} = \theta_\mathbf{i-1} + \rho\mathbf{d}$
6. Check if θ_i generates valid covariance matrices and improve the likelihood. If not, reduce ρ by half and repeat step (5). If this process is repeated for pre-specified number of times and the stated conditions are still not satisfied, stop.
7. Check convergence of the parameter. If convergence criteria are met, then stop. Otherwise, go back to step (2).

Newton's algorithm performs well if the starting value is close to the solution. In order to improve the algorithm's robustness to bad starting values, the scoring algorithm is used in the first few iterations. This can be done easily be applying different formulae for the Hessian matrix at each iteration. Apart from improved robustness, the scoring algorithm is faster due to the simpler form of the Hessian matrix.

Convergence Criteria

There are three types of convergence criteria: parameter convergence, log-likelihood convergence and Hessian convergence. For parameter and log-likelihood convergence, they are subdivided into absolute and relative. If we let ε be some given tolerance level and $\theta_{s,i}$ be the *s*th parameter in *i*th iteration, l_i be the log-likelihood in the *i*th iteration, $\mathbf{g}_i$ be the gradient vector in *i*th iteration, and $\mathbf{H}_i$ be the hessian matrix in *i*th iteration, then the criteria can be written as follows:

Absolute parameter convergence: $\max_s |\theta_{s,i} - \theta_{s,i-1}| < \epsilon$

Relative parameter convergence: $\max_s \frac{|\theta_{s,i} - \theta_{s,i-1}|}{|\theta_{s,i-1}|} < \epsilon$

Absolute log-likelihood convergence: $|l_i - l_{i-1}| < \epsilon$

Relative log-likelihood convergence: $\frac{|l_i - l_{i-1}|}{|l_{i-1}|} < \epsilon$

Absolute Hessian convergence: $\mathbf{g}_i^T \mathbf{H}_i^{-1} \mathbf{g}_i < \epsilon$

Relative Hessian convergence: $\frac{\mathbf{g}_i^T \mathbf{H}_i^{-1} \mathbf{g}_i}{|l_i|} < \epsilon$

Denominator terms that equal 0 are replaced by 1.

Starting value of Newton's Algorithm

If no prior information is available, we can choose the initial values of **G** and **R** to be the identity. However, it is highly desirable to estimate the scale of the parameter. By ignoring the random effects, and assuming the residual errors are i.i.d. with variance σ^2, we can fit a GLM model and estimate σ^2 by the residual sum of squares $\hat{\sigma}^2$. Then we choose the starting value of Newton's algorithm to be

$\mathbf{G_k} = \frac{\hat{\sigma}^2}{K+1}$ and $\mathbf{R} = \frac{\hat{\sigma}^2}{K+1}$

Confidence Intervals of Covariance Parameters

The estimate $\hat{\theta}$ (ML or REML) is asymptotically normally distributed. Its variance covariance matrix can be approximated by $-2\mathbf{H}^{-1}$, where **H** is the hessian matrix of the log-likelihood function evaluated at $\hat{\theta}$A simple Wald's type confidence interval for any covariance parameter can be obtained by using the asymptotic normality of the parameter estimates, however it is not very appropriate for variance parameters and correlation parameters that have a range of $[0, \infty)$ and $[-1, 1]$ respectively. Therefore these parameters are transformed to parameters that have range $(-\infty, \infty)$. Using the uniform delta method, see for example (van der Vaart, 1998), these transformed estimates still have asymptotic normal distributions.

Suppose we are estimating a variance parameter σ^2 by $\hat{\sigma}_n^2$ that is distributed as $N\left[\sigma^2, Var\left(\hat{\sigma}_n^2\right)\right]$ asymptotically. The transformation we used is $\log\left(\sigma^2\right)$ which can correct the skewness of $\hat{\sigma}_n^2$, moreover $\log\left(\hat{\sigma}_n^2\right)$ has the range $(-\infty, \infty)$ which matches that of normal distribution. Using the delta method, one can show that the asymptotic distribution of $\log\left(\hat{\sigma}_n^2\right)$ is $N\left[\log\left(\sigma^2\right), \sigma^{-4} Var\left(\hat{\sigma}_n^2\right)\right]$. Thus, a (1−α)100% confidence interval of $\log\left(\sigma^2\right)$ is given by

$$\log\left(\hat{\sigma}_n^2\right) \pm z_{1-\alpha/2}\hat{\sigma}_n^{-2}\sqrt{Var\left(\hat{\sigma}_n^2\right)}$$

where $z_{1-\alpha/2}$ is the upper $(1 - \alpha/2)$ percentage point of standard normal distribution. By this confidence interval, a (1−α)100% confidence interval for σ^2 is given by

$$\exp\left(\log\left(\hat{\sigma}_n^2\right) \pm z_{1-\alpha/2}\hat{\sigma}_n^{-2}\sqrt{Var\left(\hat{\sigma}_n^2\right)}\right)$$

When we need a confidence interval for a correlation parameter ρ, a possible transformation will be its generalized logit $arctanh\left(\rho\right) = 0.5\log\left[\left(1+\rho\right)/\left(1-\rho\right)\right]$. The resulting confidence interval for ρ will be

$$\tanh\left(arctanh(\hat{\rho}) \pm z_{1-\alpha/2}\left(1-\hat{\rho}^2\right)^{-1}\sqrt{Var\left(\hat{\rho}\right)}\right)$$

Fixed and Random Effect Parameters: Estimation and Prediction

After we obtain an estimate of $\theta$the best linear unbiased estimator (BLUE) of **β** and the best linear unbiased predictor (BLUP) of **γ** can be found by solving the mixed model equations, Henderson (1984).

$$\begin{bmatrix} \mathbf{X}^T\hat{\mathbf{R}}^{-1}\mathbf{X} & \mathbf{X}^T\hat{\mathbf{R}}^{-1}\mathbf{Z} \\ \mathbf{Z}^T\hat{\mathbf{R}}^{-1}\mathbf{X} & \mathbf{Z}^T\hat{\mathbf{R}}^{-1}\mathbf{X} + \hat{\mathbf{G}}^{-1} \end{bmatrix} \begin{pmatrix} \beta \\ \gamma \end{pmatrix} = \begin{bmatrix} \mathbf{X}^T\hat{\mathbf{R}}^{-1}\mathbf{y} \\ \mathbf{Z}^T\hat{\mathbf{R}}^{-1}\mathbf{y} \end{bmatrix}$$

The solution of this equation can be expressed as

$$\begin{aligned}\hat{\beta} &= \left(\mathbf{X}^T\hat{\mathbf{V}}^{-1}\mathbf{X}\right)^{-}\mathbf{X}^T\hat{\mathbf{V}}^{-1}\mathbf{y}\\ \hat{\gamma} &= \hat{\mathbf{G}}\mathbf{Z}^T\hat{\mathbf{V}}^{-1}\left(\mathbf{y}-\mathbf{X}\hat{\beta}\right)\\ &= \hat{\mathbf{G}}\left[\mathbf{Z^T}\hat{\mathbf{V}}^{-1}\mathbf{y}-\mathbf{Z^T}\hat{\mathbf{V}}^{-1}\mathbf{X}\hat{\beta}\right]\end{aligned}$$

The covariance matrix **C** of $\hat{\beta}$ and $\hat{\gamma}$ is given by

$$\begin{aligned}\mathbf{C} &= Cov\left(\hat{\beta},\hat{\gamma}\right)\\ &= \begin{bmatrix}\mathbf{X}^T\hat{\mathbf{R}}^{-1}\mathbf{X} & \mathbf{X}^T\hat{\mathbf{R}}^{-1}\mathbf{Z}\\ \mathbf{Z}^T\hat{\mathbf{R}}^{-1}\mathbf{X} & \mathbf{Z}^T\hat{\mathbf{R}}^{-1}\mathbf{Z}+\hat{\mathbf{G}}^{-1}\end{bmatrix}^{-}\\ &= \begin{bmatrix}\hat{\mathbf{C}}_{11} & \hat{\mathbf{C}}^T\\ \hat{\mathbf{C}}_{21} & \hat{\mathbf{C}}_{22}\end{bmatrix}\end{aligned}$$

where

$$\hat{\mathbf{C}}_{11} = \left(\mathbf{X}^T\hat{\mathbf{V}}^{-1}\mathbf{X}\right)^{-}$$

$$\hat{\mathbf{C}}_{21} = -\hat{\mathbf{G}}\mathbf{Z}^T\hat{\mathbf{V}}^{-1}\mathbf{X}\hat{\mathbf{C}}_{11}$$

$$\hat{\mathbf{C}}_{22} = \left(\mathbf{Z}^T\hat{\mathbf{R}}^{-1}Z+\hat{\mathbf{G}}^{-1}\right)^{-1} - \hat{\mathbf{C}}_{21}\mathbf{X}^T\hat{\mathbf{V}}^{-1}\mathbf{Z}\hat{\mathbf{G}}$$

Custom Hypotheses

In general, one can construct estimators or predictors for

$$\mathbf{Lb} = \left[\mathbf{L_0} \quad \mathbf{L_1}\right]\begin{bmatrix}\beta\\ \gamma\end{bmatrix}$$

for some hypothesis matrix **L**. Estimators or predictors of **Lb** can easily be constructed by substituting $\hat{\beta}$ and $\hat{\gamma}$ into the equation for **Lb** and its variance covariance matrix can be approximated by $\mathbf{LCL^T}$. If $\mathbf{L_1}$ is zero and $\mathbf{L_0}\beta$ is estimable, $\mathbf{L}\hat{\mathbf{b}}$ is called the best linear unbiased estimator of $\mathbf{L_0}\beta$. If $\mathbf{L_1}$ is nonzero and $\mathbf{L_0}\beta$ is estimable, $\mathbf{L}\hat{\mathbf{b}}$ is called the best linear unbiased predictor of **Lb**.

To test the hypothesis $H_0:\mathbf{Lb}=\mathbf{a}$ for a given vector **a**, we can use the statistic

$$\mathbf{F} = \frac{\left(\mathbf{L}\hat{\mathbf{b}}-\mathbf{a}\right)^T\left(\mathbf{L}\hat{\mathbf{C}}\mathbf{L^T}\right)^{-1}\left(\mathbf{L}\hat{\mathbf{b}}-\mathbf{a}\right)}{q}$$

where *q* is the rank of the matrix **L**. This statistic has an approximate *F* distribution. The numerator degrees of freedom is *q* and the denominator degree of freedom can be obtained by Satterthwaite (1946) approximation. The method outlined below is similar to Giesbrecht and Burns (1985), McLean and Sanders (1988), and Fai and Cornelius (1996).

Satterthwaite's Approximation

To find the denominator degrees of freedom of the *F* statistic, first perform the spectral decomposition $\mathbf{L}\hat{\mathbf{C}}\mathbf{L}^T = \mathbf{\Gamma}^T\mathbf{D}\mathbf{\Gamma}$ where $\mathbf{\Gamma}$ is an orthogonal matrix of eigenvectors and **D** is a diagonal matrix of eigenvalues. If l_m is the *m*th row of $\mathbf{\Gamma L}$, d_m is the *m*th eigenvalues and

$$\nu_m = \frac{2d_m}{\mathbf{g_m}\Sigma\left(\hat{\theta}\right)^{-1}\mathbf{g_m}}$$

where $\mathbf{g}_m = \frac{\partial l_m \mathbf{C} l_m^T}{\partial\theta}|_{\theta=\hat{\theta}}$ and $\mathbf{\Sigma}\left(\hat{\theta}\right)^{-1}$ is the covariance matrix of the estimated covariance parameters. If

$$E = \sum_{m=1}^{q} \frac{\nu_m}{\nu_m - 2} I\left(\nu_m > 2\right)$$

then the denominator degree of freedom is given by

$$\nu = \frac{2E}{E-q}$$

Note that the degrees of freedom can only be computed when E>q.

Type I and Type III Statistics

Type I and type III test statistics are special cases of custom hypothesis tests.

Estimated Marginal Means (EMMEANS)

Estimated marginal means are special cases of custom hypothesis test. The construction of the matrix for EMMEANS can be found in “Estimated Marginal Means” section of GLM’s algorithm document. If Bonferroni or Sidak adjustment is requested for multiple comparisons, they will be computed according to the algorithm detailed in Appendix 10:Post Hoc Tests.

Saved Values

Predicted values are computed by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta} + \mathbf{Z}\hat{\gamma}$$

Fixed predicted values are be computed by

$$\hat{\mathbf{y}} = \mathbf{X}\hat{\beta}$$

Residuals are computed by

$$\mathbf{r} = \hat{\mathbf{y}} - \mathbf{y}$$

If standard errors or degrees of freedom are requested for predicted values, a **L** matrix will be constructed for each case and the formula in custom hypothesis section will be used to obtain the requested values.

Information Criteria

Information criteria are for model comparison, the following criteria are given in smaller is better form. If we let *l* be the log-likelihood of (REML or ML), *n* be total number of cases (or total of case weights if used) and *d* is number of model parameters, the formula for various criteria are given as below,

Akaike information criteria (AIC), Akaike (1974):

$$-2l+2d$$

Finite sample corrected (AICC), Hurvich and Tsai (1989):

$$-2l+\frac{2d\times n}{(n-d-1)}$$

Bayesian information criteria (BIC), Schwarz (1978):

$$-2l+d\times\log(n)$$

Consistent AIC (CAIC), Bozdogan (1987):

$$-2l+d\times(\log(n)+1)$$

For REML, the value of *n* is chosen to be total number of cases minus number fixed effect parameters and *d* is number of covariance parameters. For ML, the value of *n* is total number of cases and *d* is number of fixed effect parameters plus number of covariance parameters.

Derivatives of Log-Likelihood

In each Newton or scoring iteration we need to compute the 1st and 2nd derivatives of the components of the log-likelihood $l_k(\theta)$, k=1,2,3. Here we let $\mathbf{g}_k=\frac{\partial}{\partial\theta}l_k(\theta)$ and $\mathbf{H}_k=\frac{\partial^2}{\partial\theta^2}l_k(\theta)$, k=1,2,3, then the 1st derivatives with respect to the *s*th parameter in **θ** is given by

$$[\mathbf{g}_1]_s=tr\left(\mathbf{V}^{-1}\frac{\partial}{\partial\theta_s}\mathbf{V}\right)$$

$$[\mathbf{g}_2]_s=-\mathbf{r}\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\mathbf{r}$$

$$[\mathbf{g}_3]_s=-tr\left(\tilde{\mathbf{X}}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\right)$$

and the 2nd derivatives with respect to *s* and *t*th parameter are given by

$$[\mathbf{H}_1]_{st}=-tr\left(\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{V}\right)\right)+tr\left(\mathbf{V}^{-1}\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{V}\right)$$

$$\begin{aligned}[\mathbf{H}_2]_{st}=\;&2\mathbf{r}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}\mathbf{r}\\&-2\mathbf{r}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T\left(\frac{\partial}{\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}\mathbf{r}\\&-\mathbf{r}^T\mathbf{V}^{-1}\left(\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}r\end{aligned}$$

$$\begin{aligned}[\mathbf{H}_3]_{st} = \quad & 2tr\left(\tilde{\mathbf{X}}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\right)\\ & -tr\left(\tilde{\mathbf{X}}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\tilde{\mathbf{X}}^T\mathbf{V}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\right)\\ & -tr\left(\tilde{\mathbf{X}}^T\mathbf{V}^{-1}\left(\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{V}\right)\mathbf{V}^{-1}\tilde{\mathbf{X}}\right)\end{aligned}$$

where $\tilde{\mathbf{X}} = \mathbf{XC}$ for a matrix **C** satisfying $\mathbf{CC}^T = \left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^- = P$ and $\mathbf{r} = \mathbf{y} - \mathbf{X}\hat{\beta}$.

Derivatives: Parameters in G

Derivatives with respect to parameters in **G** can be constructed by from the entries of

$$\begin{aligned}\mathbf{W}_1(\mathbf{X};\mathbf{r};\mathbf{Z}) &= \begin{bmatrix}\mathbf{W}_1(\mathbf{X},\mathbf{X}) & \mathbf{W}_1(\mathbf{X},\mathbf{Z}) & \mathbf{W}_1(\mathbf{X},\mathbf{r})\\ \mathbf{W}_1(\mathbf{Z},\mathbf{X}) & \mathbf{W}_1(\mathbf{Z},\mathbf{Z}) & \mathbf{W}_1(\mathbf{Z},\mathbf{r})\\ \mathbf{W}_1(\mathbf{r},\mathbf{X}) & \mathbf{W}_1(\mathbf{r},\mathbf{Z}) & \mathbf{W}_1(\mathbf{r},\mathbf{r})\end{bmatrix}\\ &= \begin{bmatrix}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{X}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{X}^T\mathbf{V}^{-1}\mathbf{r}\\ \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{r}\\ \mathbf{r}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{r}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r}\end{bmatrix}\end{aligned}$$

The matrix $\mathbf{W}_1(\mathbf{X};\mathbf{r};\mathbf{Z})$ can be computed from $\mathbf{W}_1(\mathbf{X};\mathbf{y};\mathbf{Z})$ given in "Cross Product Matrices", by using the following relationship,

$$\mathbf{r} = \mathbf{y} - \mathbf{X}\mathbf{b}_0$$

where $\mathbf{b}_0$ is the current estimate of β.

Using the above formula, we can obtain the following expressions,

$$\begin{aligned}\mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} &= \mathbf{y}^T\mathbf{V}^{-1}\mathbf{y} - \mathbf{y}^T\mathbf{V}^{-1}\mathbf{X}\mathbf{b}_0\\ &= \mathbf{W}_1(\mathbf{y},\mathbf{y}) - \mathbf{W}_1(\mathbf{y},\mathbf{X})\mathbf{b}_0\end{aligned}$$

$$\begin{aligned}\mathbf{X}^T\mathbf{V}^{-1}\mathbf{r} &= \mathbf{X}^T\mathbf{V}^{-1}\mathbf{y} - \mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}\mathbf{b}_0\\ &= \mathbf{W}_1(\mathbf{X},\mathbf{y}) - \mathbf{W}_1(\mathbf{X},\mathbf{X})\mathbf{b}_0\end{aligned}$$

$$\begin{aligned}\mathbf{Z}^T\mathbf{V}^{-1}\mathbf{r} &= \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{y} - \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{X}\mathbf{b}_0\\ &= \mathbf{W}_1(\mathbf{Z},\mathbf{y}) - \mathbf{W}_1(\mathbf{Z},\mathbf{X})\mathbf{b}_0\end{aligned}$$

In terms of the elements in $\mathbf{W}_1(\mathbf{X};\mathbf{r};\mathbf{Z})$, we can write down the 1st derivatives of l_1, l_2 and l_3 with respect to a parameter θ_s of the **G** matrix,

$$[\mathbf{g}_1]_{\mathbf{G},s} = tr\left(\mathbf{W}_1(\mathbf{Z},\mathbf{Z})\frac{\partial}{\partial\theta_s}\mathbf{G}\right)$$

$$[\mathbf{g}_2]_{\mathbf{G},s} = -\mathbf{W}_1(\mathbf{Z},\mathbf{r})^T\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{r})$$

$$[\mathbf{g}_3]_{\mathbf{G},s} = -tr\left(\mathbf{W}_1(\mathbf{X},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{X})\mathbf{P}\right)$$

For the second derivatives, we first define the following simplification factors

$$\mathbf{H}_{\mathbf{G}1}^{st} = -\mathbf{W}_1(\mathbf{Z},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{Z})\left(\frac{\partial}{\partial\theta_t}\mathbf{G}\right) + \mathbf{W}_1(\mathbf{Z},\mathbf{Z})\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{G}$$

$$\mathbf{H}_{\mathbf{G}2}^{s} = \mathbf{W}_1(\mathbf{X},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{r})$$

$$\mathbf{H}_{\mathbf{G}2}^{st} = 2\mathbf{W}_1(\mathbf{r},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{Z})\left(\frac{\partial}{\partial\theta_t}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{r}) - \mathbf{W}_1(\mathbf{r},\mathbf{Z})\left(\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{r})$$

$$\mathbf{H}_{\mathbf{G}3}^{s} = \mathbf{W}_1(\mathbf{X},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{X})$$

$$\mathbf{H}_{\mathbf{G}3}^{st} = 2\mathbf{W}_1(\mathbf{X},\mathbf{Z})\left(\frac{\partial}{\partial\theta_s}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{Z})\left(\frac{\partial}{\partial\theta_t}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{X}) - \mathbf{W}_1(\mathbf{X},\mathbf{Z})\left(\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{G}\right)\mathbf{W}_1(\mathbf{Z},\mathbf{X})$$

then second derivatives of l_1, l_2 and l_3 w.r.t. θ_s and θ_t (in **G**) are given by

$$[\mathbf{H}_1]_{\mathbf{G},st} = tr\left(\mathbf{H}_{\mathbf{G}1}^{st}\right)$$

$$[\mathbf{H}_2]_{\mathbf{G},st} = \mathbf{H}_{\mathbf{G}2}^{st} - 2(\mathbf{H}_{\mathbf{G}2}^{\mathbf{s}})^T\mathbf{P}\mathbf{H}_{\mathbf{G}2}^{\mathbf{t}}$$

$$[\mathbf{H}_3]_{\mathbf{G},st} = tr\left[\mathbf{H}_{\mathbf{G}3}^{st}\mathbf{P}\right] - \left[\mathbf{H}_{\mathbf{G}3}^{s}\mathbf{P}\mathbf{H}_{\mathbf{G}3}^{t}\mathbf{P}\right]$$

Derivatives: Parameters in R

To compute **R** derivatives, we need to introduce the matrices

$$\mathbf{W}_0^{(1)s} = -\frac{\partial\mathbf{W}_0}{\partial\theta_s}$$

and

$$\mathbf{W}_0^{(2)st} = -\frac{\partial^2\mathbf{W}_0}{\partial\theta_s\partial\theta_t}$$

where θ_s and θ_t are the *s*th and *t*th parameters of **R**. Therefore,

$$\begin{aligned}
\mathbf{W}_0(\mathbf{A},\mathbf{B}) &= \mathbf{A}^T\mathbf{R}^{-1}\mathbf{B} \\
\mathbf{W}_0^{(1)s}(\mathbf{A},\mathbf{B}) &= \mathbf{A}^T\mathbf{R}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{R}\right)\mathbf{R}^{-1}\mathbf{B} \\
&= -\mathbf{A}^T\left[\frac{\partial}{\partial\theta_s}\mathbf{R}^{-1}\right]\mathbf{B} \\
\mathbf{W}_0^{(2)st}(\mathbf{A},\mathbf{B}) &= \mathbf{A}^T\Big[\mathbf{R}^{-1}\left(\frac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{R}\right)\mathbf{R}^{-1} - \mathbf{R}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{R}\right)\mathbf{R}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{R}\right)\mathbf{R}^{-1} \\
&\quad -\mathbf{R}^{-1}\left(\frac{\partial}{\partial\theta_t}\mathbf{R}\right)\mathbf{R}^{-1}\left(\frac{\partial}{\partial\theta_s}\mathbf{R}\right)\mathbf{R}^{-1}\Big]\mathbf{B} \\
&= -\mathbf{A}^T\left[\frac{\partial^2}{\partial\theta_s\theta_t}\mathbf{R}^{-1}\right]\mathbf{B}
\end{aligned}$$

The matrices **A** and **B** can be **X**, **Z**, $\tilde{\mathbf{Z}}$ or **r**, where

$$\tilde{\mathbf{Z}} = \mathbf{Z}\mathbf{M} = \mathbf{Z}\left(\mathbf{G}^{-1} + \mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z}\right)^{-1}$$

and

$$\mathbf{r} = \left[\mathbf{I} - \mathbf{X}\left(\mathbf{X}'\mathbf{V}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}'\mathbf{V}^{-1}\right]\mathbf{y} = \mathbf{y} - \mathbf{X}\mathbf{b}_0$$

Note: The matrix $\left(\mathbf{G}^{-1} + \mathbf{Z}^T R^{-1}\mathbf{Z}\right)^{-1}$ involved in $\tilde{\mathbf{Z}}$ can be obtained by pre/post multiplying $\left(\mathbf{I} + \mathbf{L}^T\mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z}\mathbf{L}\right)^{-}$ by **L** and $\mathbf{L}^T$).

Using these notations, the 1st derivatives of $l_k(\theta)$ with respect to a parameter in **R** are as follows,

$$[\mathbf{g}_1]_{\mathbf{R},s} = tr\left(\mathbf{R}^{-1}\tfrac{\partial}{\partial\theta_s}\mathbf{R}\right) - tr\left(\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{M}\right)$$

$$\begin{aligned}[\mathbf{g}_2]_{\mathbf{R},s} = \quad & -\mathbf{W}_0^{(1)s}(\mathbf{r},\mathbf{r}) + 2\mathbf{W}_0\left(\mathbf{r},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},r) \\ & -\mathbf{W}_0\left(\mathbf{r},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{r}\right)\end{aligned}$$

$$[\mathbf{g}_3]_{\mathbf{R},s} = -tr\left(\mathbf{H}_{\mathbf{R}3}^{s}\right)$$

To compute 2nd derivatives w.r.t. θ_s and θ_t (of **R**), we need to consider the following simplification factors.

$$\begin{aligned}\mathbf{H}_{\mathbf{R}1}^{st} = \quad & -\mathbf{R}^{-1}\left(\tfrac{\partial}{\partial\theta_s}\mathbf{R}\right)\mathbf{R}^{-1}\left(\tfrac{\partial}{\partial\theta_t}\mathbf{R}\right) + \mathbf{R}^{-1}\left(\tfrac{\partial^2}{\partial\theta_s\partial\theta_t}\mathbf{R}\right) \\ & -\mathbf{W}_0^{(2)st}(\mathbf{Z},\mathbf{Z})\mathbf{M} - \mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{M}\mathbf{W}_0^{(1)t}(\mathbf{Z},\mathbf{Z})\mathbf{M}\end{aligned}$$

$$\begin{aligned}\mathbf{H}_{\mathbf{R}2}^{s} = \quad & \mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{r}) + \mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{W}\left(\tilde{\mathbf{Z}},\mathbf{r}\right) \\ & -\mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},r) - \mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{r}\right)\end{aligned}$$

$$\begin{aligned}\mathbf{H}_{\mathbf{R}2}^{st} = \quad & -\mathbf{W}_0^{(2)st}(\mathbf{r},\mathbf{r}) \\ & -\mathbf{W}_0\left(\mathbf{r},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(2)st}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{r}\right) \\ & +2\mathbf{W}_0\left(\mathbf{r},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(2)st}(\mathbf{Z},\mathbf{r}) \\ & -2\left[\mathbf{W}_0\left(\mathbf{r},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z}) - \mathbf{W}_0^{(1)s}(\mathbf{r},\mathbf{Z})\right]\mathbf{M} \\ & \times\left[\mathbf{W}_0^{(1)t}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{r}\right) - \mathbf{W}_0^{(1)t}(\mathbf{Z},\mathbf{r})\right]\end{aligned}$$

$$\begin{aligned}\mathbf{H}_{\mathbf{R}3}^{s} = \quad & \mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{X}) - \mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{X}) \\ & -\mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{X}\right) \\ & +\mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{X}\right)\end{aligned}$$

$$\begin{aligned}\mathbf{H}_{\mathbf{R}3}^{st} = \quad & -\mathbf{W}_0^{(2)st}(\mathbf{X},\mathbf{X}) \\ & -\mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(2)st}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{X}\right) \\ & +2\mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(2)st}(\mathbf{Z},\mathbf{X}) \\ & -2\left[\mathbf{W}_0\left(\mathbf{X},\tilde{\mathbf{Z}}\right)\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z}) - \mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{Z})\right]\mathbf{M} \\ & \times\left[\mathbf{W}_0^{(1)t}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},X\right) - \mathbf{W}_0^{(1)t}(\mathbf{Z},\mathbf{X})\right]\end{aligned}$$

Based on these simplification terms, the entries of the Hessian matrices are given by

$$[\mathbf{H}_1]_{\mathbf{R},st} = tr\left(\mathbf{H}_{\mathbf{R}1}^{st}\right)$$

$$[\mathbf{H}_2]_{\mathbf{R},st} = \mathbf{H}_{\mathbf{R}2}^{st} - 2\left(\mathbf{H}_{\mathbf{R}2}^{\mathbf{s}}\right)^T\mathbf{P}\mathbf{H}_{\mathbf{R}2}^{t}$$

$$[\mathbf{H}_3]_{\mathbf{R},st} = tr\left(\mathbf{H}_{\mathbf{R}3}^{st}\mathbf{P} - \mathbf{H}_{\mathbf{R}3}^{s}\mathbf{P}\mathbf{H}_{\mathbf{R}3}^{t}\mathbf{P}\right)$$

G and R Cross-derivatives

This section gives expressions for the 2nd derivatives of l_1, l_2 and l_3 with respect to a parameter θ_s in **G** and a parameter θ_t in **G** . First, we introduce the following simplification terms,

$$\begin{aligned}\mathbf{H}^{st}_{\mathbf{GR}1} = \; & -\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\left(\tfrac{\partial}{\partial\theta_t}\mathbf{G}\right) \\ & +2\mathbf{W}_0^{1(s)}(\mathbf{Z},\mathbf{Z})\left(\tfrac{\partial}{\partial\theta_t}\mathbf{G}\right)\mathbf{W}_0\left(\mathbf{Z},\tilde{\mathbf{Z}}\right) \\ & -\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\mathbf{W}_0\left(\tilde{\mathbf{Z}},\mathbf{Z}\right)\left(\tfrac{\partial}{\partial\theta_t}\mathbf{G}\right)\mathbf{W}_0\left(\mathbf{Z},\tilde{\mathbf{Z}}\right)\end{aligned}$$

$$\begin{aligned}\mathbf{H}^{st}_{\mathbf{GR}2} = \; & 2\left[\mathbf{W}_0^{(1)s}(\mathbf{r},\mathbf{Z})-\mathbf{W}_0(\mathbf{r},\mathbf{Z})\mathbf{M}\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\right] \\ & \times[\mathbf{M}\mathbf{W}_0(\mathbf{Z},\mathbf{Z})-\mathbf{I}]\left(\tfrac{\partial}{\partial\theta_t}\mathbf{G}\right)[\mathbf{W}_0(\mathbf{Z},\mathbf{Z})\mathbf{M}-\mathbf{I}] \\ & \times\mathbf{W}_0(\mathbf{Z},\mathbf{r})\end{aligned}$$

$$\begin{aligned}\mathbf{H}^{st}_{\mathbf{GR}3} = \; & 2\left[\mathbf{W}_0^{(1)s}(\mathbf{X},\mathbf{Z})-\mathbf{W}_0(\mathbf{X},\mathbf{Z})\mathbf{M}\mathbf{W}_0^{(1)s}(\mathbf{Z},\mathbf{Z})\right] \\ & \times[\mathbf{M}\mathbf{W}_0(\mathbf{Z},\mathbf{Z})-\mathbf{I}]\left(\tfrac{\partial}{\partial\theta_t}\mathbf{G}\right)[\mathbf{W}_0(\mathbf{Z},\mathbf{Z})\mathbf{M}-\mathbf{I}] \\ & \times\mathbf{W}_0(\mathbf{Z},\mathbf{X})\end{aligned}$$

Based on these simplification terms, the second derivatives are given by

$$[\mathbf{H}_1]_{\mathbf{GR},st} = tr\left(\mathbf{H}^{st}_{\mathbf{GR}1}\right)$$

$$[\mathbf{H}_2]_{\mathbf{GR},st} = \mathbf{H}^{st}_{\mathbf{GR}2} - 2(\mathbf{H}^{s}_{\mathbf{G}2})^T\mathbf{H}^{t}_{\mathbf{R}2}$$

$$[\mathbf{H}_3]_{\mathbf{GR},st} = tr\left(\mathbf{H}^{st}_{\mathbf{GR}3}\mathbf{P} - \mathbf{H}^{s}_{\mathbf{G}3}\mathbf{P}\mathbf{H}^{t}_{\mathbf{R}3}\mathbf{P}\right)$$

Gradient and Hessian of REML

The restricted log likelihood is given by

$$\begin{aligned}-2l_{REML}(\theta|\mathbf{y}) = \; & \log|\mathbf{V}| + \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} \\ & + \log\left|\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}\right| + (n-p)\log 2\pi\end{aligned}$$

where *p* is equal to the rank of **X**. Therefore the *s*th element of the gradient vector is given by

$$[\mathbf{g}]_s = [\mathbf{g}_1]_s + [\mathbf{g}_2]_s + [\mathbf{g}_3]_s$$

and the (*s*,*t*)th element of the Hessian matrix is given by

$$[\mathbf{H}]_{st} = [\mathbf{H}_1]_{st} + [\mathbf{H}_2]_{st} + [\mathbf{H}_3]_{st}$$

If scoring algorithm is used, the Hessian can be simplified to

$$[\mathbf{H}]_{st} = -[\mathbf{H}_1]_{st} + [\mathbf{H}_3]_{st}$$

Gradient and Hessian of MLE

The log likelihood is given by

$$-2l_{MLE}(\theta|\mathbf{y}) = \log|\mathbf{V}| + \mathbf{r}^T\mathbf{V}^{-1}\mathbf{r} + n\log 2\pi$$

Therefore the *s*th element of the gradient vector is given by

$$[\mathbf{g}]_s = [\mathbf{g}_1]_s + [\mathbf{g}_2]_s$$

and the (*s*,*t*)th element of the Hessian matrix is given by

$$[\mathbf{H}]_{st} = [\mathbf{H}_1]_{st} + [\mathbf{H}_2]_{st}$$

If scoring algorithm is used the Hessian can be simplified to

$$[\mathbf{H}]_{st} = -[\mathbf{H}_1]_{st}$$

It should be noted that the Hessian matrices for the scoring algorithm in both ML and REML are not ‘exact’. In order to speed up calculation, some second derivative terms are dropped. Therefore, they are only used in intermediate step of optimization but not for standard error calculations.

Cross Product Matrices

During estimation we need to construct several cross product matrices in each iteration, namely: $\mathbf{W}_0(\mathbf{X};\mathbf{y};\mathbf{Z})$, $\mathbf{W}_1(\mathbf{X};\mathbf{y};\mathbf{Z})$, $\mathbf{W}_0^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z})$, $\mathbf{W}_1^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z})$, $\mathbf{W}_{\mathbf{b}0}(\mathbf{X};\mathbf{y})$, and $\mathbf{W}_{\mathbf{b}1}(\mathbf{X};\mathbf{y})$. The sweep operator (see for example Goodnight (1979)) is used in constructing these matrices. Basically, the sweep operator performs the following transformation

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}' & \mathbf{C} \end{bmatrix} \Rightarrow \begin{bmatrix} \mathbf{A}^- & \mathbf{A}^-\mathbf{B} \\ -\mathbf{B}'\mathbf{A}^- & \mathbf{C}-\mathbf{B}'\mathbf{A}^-\mathbf{B} \end{bmatrix}$$

The steps needed to construct these matrices are outlined below:

STEP 1: Construct

$$\mathbf{W}_0(\mathbf{X};\mathbf{y};\mathbf{Z}) = \begin{bmatrix} \mathbf{X}^T\mathbf{R}^{-1}\mathbf{X} & \mathbf{X}^T\mathbf{R}^{-1}\mathbf{y} & \mathbf{X}^T\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{y}^T\mathbf{R}^{-1}\mathbf{X} & \mathbf{y}^T\mathbf{R}^{-1}\mathbf{y} & \mathbf{y}^T\mathbf{R}^{-1}\mathbf{Z} \\ \mathbf{Z}^T\mathbf{R}^{-1}\mathbf{X} & \mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z} & \mathbf{y}^T\mathbf{R}^{-1}\mathbf{y} \end{bmatrix}$$

STEP 2:

Construct $\mathbf{W}_0(\mathbf{X};\mathbf{y};\mathbf{Z})$ which is an augmented version of $\mathbf{W}_0(\mathbf{X};\mathbf{y};\mathbf{Z})$. It is given by he following expression.

$$\mathbf{W}_0^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z}) = \begin{bmatrix} \mathbf{I}+\mathbf{L}^T\mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z}\mathbf{L} & \mathbf{L}^T\mathbf{W}_0(\mathbf{Z},\cdot) \\ \mathbf{W}_0(\cdot,\mathbf{Z})L & \mathbf{W}_0 \end{bmatrix}$$

where **L** is the lower-triangular Cholesky root of **G**, i.e. **G**=**LLT** and $\mathbf{W}_0(\mathbf{Z};?;?)$ is the rows of $\mathbf{W}_0$ corresponding to **Z**.

STEP 3: Sweeping $\mathbf{W}_0^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z})$ by pivoting on diagonal elements in upper-left partition will give us the matrix $\mathbf{W}_1^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z})$, which is shown below.

$$\mathbf{W}_1^{\mathbf{A}}(\mathbf{X};\mathbf{y};\mathbf{Z}) = \begin{bmatrix} \mathbf{W}_1^{\mathbf{A}}(1,1) & \mathbf{W}_1^{\mathbf{A}}(1,1)\,\mathbf{L}^T\mathbf{W}(\mathbf{Z},\cdot) \\ -\mathbf{W}_0(\cdot,\mathbf{Z})\,L\mathbf{W}_1^{\mathbf{A}}(1,1) & \mathbf{W}_1(\mathbf{X};\mathbf{y};\mathbf{Z}) \end{bmatrix}$$

where

$$\mathbf{W}_1(\mathbf{X};\mathbf{y};\mathbf{Z}) = \begin{bmatrix} \mathbf{X}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{X}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{X}^T\mathbf{V}^{-1}\mathbf{y} \\ \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{Z}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{Z}^T\mathbf{X}^{-1}\mathbf{y} \\ \mathbf{y}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{y}^T\mathbf{V}^{-1}\mathbf{Z} & \mathbf{y}^T\mathbf{V}^{-1}\mathbf{y} \end{bmatrix}$$

and

$$\mathbf{W}_1^{\mathbf{A}}(\mathbf{1},\mathbf{1}) = \left(\mathbf{I} + \mathbf{L}^T\mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z}\mathbf{L}\right)^-$$

During the sweeping, if we accumulate the log of the *i*th diagonal element just before *i*th sweep, we will obtain $\log\left|\mathbf{I} + \mathbf{L}^T\mathbf{Z}^T\mathbf{R}^{-1}\mathbf{Z}\mathbf{L}\right| = \log|\mathbf{V}| - \log|\mathbf{R}|$ as a by-product. Thus, adding to this quantity by $\log|\mathbf{R}|$ will give us $l_1(\theta)$.

STEP 4: Consider the following submatrix $\mathbf{W}_{\mathbf{b}0}\ (\mathbf{X};\mathbf{y})$ of $\mathbf{W}_1(\mathbf{X};\mathbf{y};\mathbf{Z})$,

$$\mathbf{W}_{b0}(\mathbf{X};\mathbf{y}) = \begin{bmatrix} \mathbf{X}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{X}^T\mathbf{V}^{-1}\mathbf{y} \\ \mathbf{y}^T\mathbf{V}^{-1}\mathbf{X} & \mathbf{y}^T\mathbf{V}^{-1}\mathbf{y} \end{bmatrix}$$

Sweeping $\mathbf{W}_{\mathbf{b0}}(\mathbf{X};\mathbf{y})$ by pivoting on diagonal elements of $\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}$ will give us

$$\mathbf{W}_{\mathbf{b1}}(\mathbf{X};\mathbf{y}) = \begin{bmatrix} \left(\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}\right)^- & \mathbf{b}_0 \\ \mathbf{b}_0^T & l_2(\theta) \end{bmatrix}$$

where $\mathbf{b}_0$ is an estimate of $\boldsymbol{\beta}_0$ in the current iteration. After this step, we will obtain $l_2(\theta)$ and $l_3.(\theta) = \left|\mathbf{X}^T\mathbf{V}^{-1}\mathbf{X}\right|$

References

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transaction on Automatic Control* , AC–19, 716–723.

Bozdogan, H. 1987. Model Selection and Akaike's Information Criterion (AIC): The General Theory and Its Analytical Extensions. *Psychometrika*, 52, 345–370.

Fai, A. H. T., and P. L. Cornelius. 1996. Approximate F-tests of Multiple Degree of Freedom Hypotheses in Generalized Least Squares Analyses of Unbalanced Split-plot Experiments. *Journal of Statistical Computation and Simulation*, 54, 363–378.

Giesbrecht, F. G., and J. C. Burns. 1985. Two-Stage Analysis Based on a Mixed Model: Large-sample Asymptotic Theory and Small-Sample Simulation Results. *Biometrics*, 41, 477–486.

Goodnight, J. H. 1979. A tutorial on the SWEEP operator. *The American Statistician*, 33:3, 149–158.

Henderson, C. R. 1984. *Applications of Linear Models in Animal Breeding*. Guelph, Ontario: University of Guelph.

Hurvich, C. M., and C. L. Tsai. 1989. Regression and Time Series Model Selection in Small Samples. *Biometrika* , 76, 297–307.

McLean, R. A., and W. L. Sanders. 1988. Approximating Degrees of Freedom for Standard Errors in Mixed Linear Models. In: *Proceedings of the Statistical Computing Section, American Statistical Association*, New Orleans: American StatisticalAssociation, 50–59.

Schwarz, G. 1978. Estimating the Dimension of a Model. *Annals of Statistics*, 6, 461–464.

Wolfinger, R., R. Tobias, and J. Sall. 1994. Computing Gaussian likelihoods and their derivatives for general linear mixed models. *SIAM Journal on Scientific Computing*, 15:6, 1294–1310.

MLP Algorithms

The multilayer perceptron (MLP) is a feed-forward, supervised learning network with up to two hidden layers. The MLP network is a function of one or more predictors (also called inputs or independent variables) that minimizes the prediction error of one or more target variables (also called outputs). Predictors and targets can be a mix of categorical and scale variables.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$X^{(m)} = \left(x_1^{(m)}, \ldots, x_P^{(m)}\right)$	Input vector, pattern m, m=1,...M.
$Y^{(m)} = \left(y_1^{(m)}, \ldots, y_R^{(m)}\right)$	Target vector, pattern m.
I	Number of layers, discounting the input layer.
J_i	Number of units in layer i. $J_0 = P$, $J_1 = R$, discounting the bias unit.
Γ^c	Set of categorical outputs.
Γ	Set of scale outputs.
Γ_h	Set of subvectors of $Y^{(m)}$ containing 1-of-c coded hth categorical variable.
$a_{i:j}^m$	Unit j of layer i, pattern m, $j = 0, \ldots, J_i$; $i = 0, \ldots, I$.
$w_{i:j,k}$	Weight leading from layer i−1, unit j to layer i, unit k. No weights connect $a_{i-1:j}^m$ and the bias $a_{i:0}^m$; that is, there is no $w_{i:j,0}$ for any j.
$c_{i:k}^m$	$\sum_{j=0}^{J_{i-1}} w_{i:j,k} a_{i-1:j}^m$, i=1,...,I.
$\gamma_i(c)$	Activation function for layer i.
$\mathbf{w}$	Weight vector containing all weights $\left(w_{1:0,1}, w_{1:0,2}, \ldots, w_{I:J_{I-1},J_I}\right)$.

Architecture

The general architecture for MLP networks is:

Input layer: J_0=P units, $a_{0:1}, \cdots, a_{0:J_0}$; with $a_{0:j} = x_j$.

ith hidden layer: J_i units, $a_{i:1}, \cdots, a_{i:J_i}$; with $a_{i:k} = \gamma_i(c_{i:k})$ and $c_{i:k} = \sum_{j=0}^{J_{i-1}} w_{i:j,k} a_{i\_1:j}$ where .$a_{i-1:0} = 1$

Output layer: J_I=R units, $a_{I:1}, \cdots, a_{I:J_I}$; with $a_{I:k} = \gamma_I(c_{I:k})$ and $c_{I:k} = \sum_{j=0}^{J_1} w_{I:j,k} a_{i\_1:j}$ where .$a_{i-1:0} = 1$

Note that the pattern index and the bias term of each layer are not counted in the total number of units for that layer.

Activation Functions

Hyperbolic Tangent

$$\gamma(c) = \tanh(c) = \frac{e^c - e^{-c}}{e^c + e^{-c}}$$

Sigmoid

$$\gamma(c) = \frac{1}{1 + \exp(-c)}$$

Identity

$$\gamma(c) = c$$

This is only available for output layer units.

Softmax

$$\gamma(c_k) = \frac{\exp(c_k)}{\sum_{j \in \Gamma_h} \exp(c_j)}$$

This is only available if all output layer units correspond to categorical variables and cross-entropy error is used.

Error Functions

Sum-of-Squares

$$E_T(w) = \sum_{m=1}^{M} E_m(w)$$

where

$$E_m(w) = \frac{1}{2}\sum_{r=1}^{R}\left(y_r^{(m)} - a_{I:r}^{m}\right)^2$$

Cross-Entropy

$$E_T(w) = \sum_{m=1}^{M} E_m(w)$$

where

$$E_m(w) = -\sum_{r \in \Gamma^c} y_r^{(m)} \log\left(\frac{a_{I:r}^m}{y_r^{(m)}}\right)$$

This is only available if all output layer units correspond to categorical variables and the softmax activation function is used.

Expert Architecture Selection

Expert architecture selection determines the "best" number of hidden units in a single hidden layer. The hyperbolic tangent activation function is used for the hidden layer, and the identity function is used for the output layer (softmax if the output is categorical).

A random sample is taken from the entire data set and split into training (70%) and testing samples (30%). The size of random sample is N = min(1000, *memsize*), where *memsize* is the user-controlled maximum number of cases stored in memory. If entire dataset has less than N cases, use all of them. If training and testing data sets are supplied separately, the random samples for training and testing should be taken from the respective datasets.

Given $K_{\min}$ and $K_{\max}$, the algorithm is as follows.

1. Start with an initial network of k hidden units. The default is k=min(g(R,P),20,h(R,P)), where

   $$g(R,P) = \begin{cases} \lfloor 4.5 + \sqrt{P+R} \rfloor & R < 5, P \geq 8 \\ \lfloor 0.5 + 0.5(P+R) \rfloor & \text{otherwise} \end{cases}$$

   where $\lfloor x \rfloor$ denotes the largest integer less than or equal to x. $h(R,P) = \left\lceil \frac{M-R}{P+R+1} \right\rceil$ is the maximum number of hidden units that will not result in more weights than there are cases in the entire training set.

   If $k < K_{\min}$, set $k = K_{\min}$. Else if $k > K_{\max}$, set $k = K_{\max}$. Train this network once via the alternated simulated annealing and training procedure (steps 1 to 5).

2. If k>$K_{\min}$, set *DOWN*=*TRUE*. Else if training error ratio > 0.01, *DOWN*=*FALSE*. Else stop and report the initial network.

3. If *DOWN*=*TRUE*, remove the weakest hidden unit (see below); k=k−1. Else add a hidden unit; k=k+1.

4. Using the previously fit weights as initial weights for the old weights and random weights for the new weights, train the old and new weights for the network once through the alternated simulated annealing and training procedure (steps 3 to 5) until the stopping conditions are met.

5. If the error on test data has dropped:

   If DOWN=FALSE, If k< $K_{\max}$ and training error is dropped but the error ratio is still above 0.01, return to step 3. Else stop and report the network with the minimum test error.

   Else if DOWN=TRUE, If k> $K_{\min}$, return to step 3. Else, stop and report the network with the minimum test error.

Else if the error increased, If DOWN=TRUE, If k - k_0|>1, stop and report the network with the minimum test error.

Else if training error ratio for k= k_0 is bigger than 0.01, set DOWN = FALSE, k= k_0 return to step 3. Else stop and report the initial network.

Else stop and report the network with the minimum test error.

If more than one network attains the minimum test error, choose the one with a smaller number of hidden units.

If the resulted network from above procedure has training error ratio (training error divided by error from the model using average of an output variable to predict that variable) bigger than 0.1, repeat above procedure with different initial weights until either the error ratio is <=0.1 or the procedure is repeated K times already, say K=5. If the procedure is repeated K times, pick the one with smallest test error.

The weakest hidden unit

For each hidden unit j, calculate the error on the test data when j is removed from the network. The weakest hidden unit is the one having the smallest total test error upon its removal.

Training

Given the training type (online, batch, or mini-batch), the problem of estimating the weights consists of the following parts:

- Initializing the weights. Take a random sample (as described in “Expert Architecture Selection”) and apply the alternated simulated annealing and training procedure on the random sample to derive the initial weights. Training in step 3 is performed using all default training parameters.
- Computing the derivative of the error function with respect to the weights. This is solved via the error backpropagation algorithm.
- Updating the estimated weights. This is solved by the gradient descent or scaled conjugate gradient method.

Alternated Simulated Annealing and Training

The following procedure uses simulated annealing and training alternately up to K_1 times. Simulated annealing is used to break out of the local minimum that training finds by perturbing the local minimum K_2 times. If break out is successful, simulated annealing sets a better initial weight for the next training. We hope to find the global minimum by repeating this procedure K_3 times. This procedure is rather expensive for large data sets, so it is only used on a random sample to search for initial weights and in architecture selection. Let K_1=K_2=4, K_3=3.

1. Randomly generate K_2 weight vectors between [a_0−a, a_0+a]. This is a user controllable interval with default a_0=0 and a=0.5. Calculate the training error for each weight vector. Pick the weights that give the minimum training error as the initial weights.

2. Set k_1=0.

3. Train the network with the specified initial weights. Call the trained weights **w**.

4. If the training error ratio <= 0.05, stop the k_1 loop and use **w** as the result of the loop. Else set $k_1 = k_1$+1.

5. If k_1<K_1, perturb the old weight to form K_2 new weights $\mathbf{w}' = \mathbf{w} + \mathbf{w}_n$ by adding K_2 different random noise $\mathbf{w}_n$ between [a(k_1), a(k_1)] where $a(k_1) = (0.5)^{k_1} a$. Let $\mathbf{w}_{\min}$ be the weights that give the minimum training error among all the perturbed weights. If $E_T(\mathbf{w}_{\min}) < E_T(\mathbf{w})$, set the initial weights to be $\mathbf{w}_{\min}$, return to step 3. Else stop and report **w** as the final result.

   Else stop the k_1 loop and use **w** as the result of the loop.

   If the resulting weights have training error ratio bigger than 0.1, repeat this algorithm until either the training error ratio is <=0.1 or the procedure is repeated K_3 times, then pick the one with smallest test error among the result of the k_1 loops.

Error Backpropagation

Error-backpropagation is used to compute the first partial derivatives of the error function with respect to the weights.

First note that $\gamma'(c) = \begin{cases} 1 - [\gamma(c)]^2 & \text{tanh} \\ \gamma(c)(1-\gamma(c)) & \text{sigmoid} \\ 1 & \text{identity} \end{cases}$

The backpropagation algorithm follows:

For each i,j,k, set $\frac{\partial E_T}{\partial w_{i:k,j}} = 0$.

For each m in group T; For each p=1,...,J_I, let

$$\delta^m_{I:p} = \frac{\partial E_m}{\partial c^m_{I:p}} = \begin{cases} a^m_{I:p} - y^{(m)}_p & \text{if cross-entropy error is used} \\ \gamma'_I\left(c^m_{I:p}\right)\left(a^m_{I:p} - y^{(m)}_p\right) & \text{otherwise} \end{cases}$$

For each i=I,...,1 (start from the output layer); For each j=1,...,J_i; For each k=0,...,J_{i-1}

- ► Let $\frac{\partial E_m}{\partial w_{i:k,j}} = \delta^m_{i:j} a^m_{i-1:k}$, where $\delta^m_{i:j} = \frac{\partial E_m}{\partial c^m_{i:j}}$
- ► Set $\frac{\partial E_T}{\partial w_{i:k,j}} = \frac{\partial E_T}{\partial w_{i:k,j}} + \frac{\partial E_m}{\partial w_{i:k,j}}$
- ► If k > 0 and i > 1, set $\delta^m_{i-1:k} = \gamma'_{i-1}\left(c^m_{i-1:k}\right) \sum_{j=1}^{J_i} \delta^m_{i:j} w_{i:k,j}$

This gives us a vector of $\sum_{i=0}^{I-1} (J_i + 1) J_{i+1}$ elements that form the gradient of $E_T(w_k)$.

Gradient Descent

Online or Mini-Batch

Given the learning rate parameters η_0 and η_{low}, momentum rate α, and learning rate decay factor β, the gradient descent method for online and mini-batch training is as follows.

1. Let *k*=0. Initialize the weight vector to w_0, learning rate to η_0. Let $\Delta w_0 = 0$.
2. Read records in T_k (T_k is randomly chosen) and find $E_{T_k}(w_k)$ and its gradient $g_k = \nabla E_{T_k}(w_k)$.
3. If $\eta_k|g_k| \le \alpha|\Delta w_k|$, $\alpha = 0.9\eta_k\frac{|g_k|}{|\Delta w_k|}$. This step is to make sure that the steepest gradient descent direction dominates weight change in next step. Without this step, the weight change in next step could be along the opposite direction of the steepest descent and hence no matter how small η_k is, the error will not decrease.
4. Let $v = w_k - \eta_k g_k + \alpha\Delta w_k$.
5. If $E_{T_k}(v) < E_{T_k}(w_k)$, then set $w_{k+1} = v$ and $\Delta w_{k+1} = w_{k+1} - w_k$, Else $w_{k+1} = w_k, \Delta w_{k+1} = \Delta w_k$.
6. $\eta_{k+1} = e^{-\beta}\eta_k$. If $\eta_{k+1} < \eta_{low}$, then set $\eta_{k+1} = \eta_{low}$.
7. If a stopping rule is met, exit and report the network as stated in the stopping criteria. Else let *k*=*k*+1 and return to step 2.

Batch

Given the learning rate parameter η_0 and momentum rate α, the gradient descent method for batch training is as follows.

1. Let *k*=0. Initialize the weight vector to w_0, learning rate to η_0. Let $\Delta w_0 = 0$.
2. Read all data and find $E_T(w_k)$ and its gradient $g_k = \nabla E_T(w_k)$. If $|g_k| < 10^{-6}$, stop and report the current network.
3. If $\eta_k|g_k| \le \alpha|\Delta w_k|$, $\alpha = 0.9\eta_k\frac{|g_k|}{|\Delta w_k|}$. This step is to make sure that the steepest gradient descent direction dominates weight change in next step. Without this step, the weight change in next step could be along the opposite direction of the steepest descent and hence no matter how small η_k is, the error will not decrease.
4. Let $v = w_k - \eta_k g_k + \alpha\Delta w_k$
5. If $E_T(v) < E_T(w_k)$, then set $w_{k+1} = v$, $\Delta w_{k+1} = w_{k+1} - w_k$, and $\eta_{k+1} = \eta_k$, Else $\eta_k = .5\eta_k$ and return to step 3.
6. If a stopping rule is met, exit and report the network as stated in the stopping criteria. Else let *k*=*k*+1 and return to step 2.

Scaled Conjugate Gradient

This method is only available to batch training. To begin, initialize the weight vector to $\mathbf{w}_0$, and let *N* be the total number of weights.

1. *k*=0. Choose scalars $0 < \lambda_0 < 10^{-6}, 0 < \sigma < 10^{-4}, \overline{\lambda_0} = 0$. Set $\mathbf{r}_0 = \mathbf{p}_0 = -\nabla E_T(\mathbf{w}_0)$, and *success=true*.
2. If *success=true*, find the second-order information: $\sigma_k = \frac{\sigma}{|\mathbf{p}_k|}$, $\mathbf{s}_k = \frac{\nabla E_T(\mathbf{w}_k + \sigma_k \mathbf{p}_k) - \nabla E_T(\mathbf{w}_k)}{\sigma_k}$, $\delta_k = \mathbf{p}_k^t \mathbf{s}_k$, where the superscript *t* denotes the transpose.
3. Set $\delta_k = \delta_k + \left(\lambda_k - \overline{\lambda_k}\right)|\mathbf{p}_k|^2$.
4. If $\delta_k \le 0$, make the Hessian positive definite: $\overline{\lambda_k} = 2\left(\lambda_k - \frac{\delta_k}{|\mathbf{p}_k|^2}\right)$, $\delta_k = -\delta_k + \lambda_k|\mathbf{p}_k|^2$, $\lambda_k = \overline{\lambda_k}$.
5. Calculate the step size: $\mu_k = \mathbf{p}_k^t \mathbf{r}_k$, $\alpha_k = \frac{\mu_k}{\delta_k}$.
6. Calculate the comparison parameter: $\Delta_k = 2\delta_k \frac{[E_T(\mathbf{w}_k) - E_T(\mathbf{w}_k + \alpha_k \mathbf{p}_k)]}{\mu_k}$.
7. If $\Delta_k \ge 0$, error can be reduced. Set $\mathbf{w}_{k+1} = \mathbf{w}_k + \alpha_k \mathbf{p}_k$, $\mathbf{r}_{k+1} = -\nabla E_T(\mathbf{w}_{k+1})$, If $|\mathbf{r}_{k+1}| < 10^{-6}$, return $\mathbf{w}_{k+1}$ as the final weight vector and exit. Set $\overline{\lambda_k} = 0$, *success=true*. If *k* mod *N*=0, restart the algorithm: $\mathbf{p}_{k+1} = \mathbf{r}_{k+1}$, else set $\beta_k = \frac{|\mathbf{r}_{k+1}|^2 - \mathbf{r}_{k+1}^t \mathbf{r}_k}{\mu_k}$, $\mathbf{p}_{k+1} = \mathbf{r}_{k+1} + \beta_k \mathbf{p}_k$. If $\Delta_k \ge .75$, reduce the scale parameter: $\lambda_k = \frac{1}{4}\lambda_k$. else (if $\Delta_k < 0$): Set $\overline{\lambda_k} = \lambda_k$, *success=false*.
8. If $\Delta_k < .25$, increase the scale parameter: $\lambda_k = \lambda_k + \frac{\delta_k(1-\Delta_k)}{|\mathbf{p}_k|^2}$.
9. If *success=false*, return to step 2. Otherwise if a stopping rule is met, exit and report the network as stated in the stopping criteria. Else set *k*=*k*+1 , $\overline{\lambda_{k+1}} = \overline{\lambda_k}$, $\lambda_{k+1} = \lambda_k$ and return to step 2.

Note: each iteration of batch training requires at least two data passes.

Stopping Rules

Training proceeds through at least one complete pass of the data. Then the search should be stopped according to following criteria. These stopping criteria should be checked in the listed order. For batch training, check of any stopping criteria is performed after completion of an iteration. For online or mini-batch training, check of any of stopping criterion 1, 3, 4, and 5 is performed after completion of a data pass, only check of criterion 2 is performed after an iteration. Let step mean a data pass for online and mini-batch methods, an iteration for batch method. Let E_1 denote the current minimum error and K_1 denote the step where it occurs for training data, E_2 and K_2 are that for testing data, and *K*3=min(K_1,K_2).

1. If there is no testing dataset and the training method is online or mini-batch, compute the total error for training data at the end of each step. From step K_1, if the training error does not decrease below E_1 over the next n steps, stop. Report the weights at step K_1. If there is a testing dataset, users have the following options:

   Check testing data only: at the end of each step compute the total error for testing data. From step K_2, if the testing error does not decrease below E_2 over the next *n* steps, stop. Report the weights at step K_2.

Check both training and testing data: at the end of each step simultaneously check the total error for training and testing data. From step K_1 for training and step K_2 for testing, if either training or testing error does not decrease below its current minimum over the next *n* steps, stop. Report the weights at step K_3. Notice that for batch method there is no need to check the total error for training data because a decrease in total error for training data is guaranteed by the algorithm.

2. The search has lasted beyond some maximum allotted time. For batch training, report the weights at step K_3. For on-line or mini-batch training, even though training stops before the completion of current step, treat this as a complete step. Calculate current errors for training and testing datasets and update E_1, K_1, E_2, K_2 correspondingly. Report the weights at step K_3.

3. The search has lasted more than some maximum number of data passes. Report the weights at step K_3.

4. When current training error is the minimum ($E_1 = E_T(w_k)$, always true for batch), stop if the relative change in training error is small: $\frac{|E_T(w_k)-E_T(\boldsymbol{w_{k-1}})|}{\frac{1}{2}(E_T(w_k)+E_T(w_{k-1})+\delta)} < \epsilon_1$ for $\delta = 10^{-10}$ and ϵ_1, where w_{k-1}, w_k are the weight vectors of two consecutive steps. Report weights at step K_3.

5. The current training error ratio is small compared with the initial error: $\left|\frac{E_T(w_k)}{\overline{E}_T+\delta}\right| < \epsilon_2$ for $\delta = 10^{-10}$ and ϵ_2, where $\overline{E}_T$ is the total error from the model using the average of an output variable to predict that variable; $\overline{E}_T$ is calculated by using $a^m_{I:r} = \frac{1}{M}\sum_{m=1}^{M} y_r^{(m)}$ in the error function, where w_k is the weight vector of one step. Report weights at step K_3.

Note: In criteria 4 and 5, the total error for whole training data is needed. For batch training, the error is always calculated, but for online or mini-batch training, error is not available without passing the training data one more time. So for online and mini-batch training, criterion 4 and 5 will not be checked if user decides to use testing data only in criterion 1.

Missing Values

Missing values are not allowed.

Output Statistics

The following output statistics are available. Note that, for scale variables, output statistics are reported in terms of the rescaled values of the variables.

Sum-of-Squares or Cross Entropy Error

As described in “Error Functions”. The cross entropy error is displayed if the output layer activation function is softmax, otherwise the sum-of-squares error is shown.

Relative Error

For each scale target *r*:

$$\frac{\sum_{m=1}^{M}\left(y_r^{(m)}-\hat{y}_r^{(m)}\right)^2}{\sum_{m=1}^{M}\left(y_r^{(m)}-\overline{y}_r\right)^2}$$

For each categorical target *r*, report p_r, the percent of incorrect predictions

Average Overall Relative Error

If there is at least one scale target:

$$\frac{\sum_{m=1}^{M}\sum_{r=1}^{R}\left(y_r^{(m)}-\hat{y}_r^{(m)}\right)^2}{\sum_{m=1}^{M}\sum_{r=1}^{R}\left(y_r^{(m)}-\overline{y}_r\right)^2}$$

where $\overline{y}_r$ is the mean of $y_r^{(m)}$ over patterns.

If all targets are categorical, report the average percent of incorrect predictions:

$$\frac{1}{C}\sum_{r=1}^{C}p_r$$

where *C* is the number of categorical variables.

Sensitivity Analysis

For each predictor *p* and each input pattern *m*, compute:

$$d_{pm}=\max_{x_{p_1},x_{p_2}\in S_p}||\hat{Y}_{p_1}^{(m)}-\hat{Y}_{p_2}^{(m)}||$$

where $\hat{Y}_{p_k}^{(m)}$ is the predicted output vector (standardized if standardization of output variable is used in training) using $\left(x_1^{(m)},...,x_{p-1}^{(m)},x_{p_k},x_{p+1}^{(m)},...,x_P^{(m)}\right)$ as its input, and S_p= $\left\{x_p^{\min},x_p^{(2)},x_p^{(3)},x_p^{(4)},x_p^{\max}\right\}$ for scale predictors and $\{(1,0,...,0),(0,1,0,...,0),...,(0,0,...,1)\}$ for categorical predictors.

Then compute:

$$d_p=\frac{1}{M}\sum_{m=1}^{M}d_{pm}$$

and normalize the d_ps to sum to 1, and report these normalized values as the sensitivity values for the predictors. This is the average maximum amount we can expect the output to change based on changes in the *p*th predictor. The greater the sensitivity, the more we expect the output to change when the predictor changes.

References

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition*, 3rd ed. Oxford: Oxford University Press.

Fine, T. L. 1999. *Feedforward Neural Network Methodology*, 3rd ed. New York: Springer-Verlag.

Haykin, S. 1998. *Neural Networks: A Comprehensive Foundation*, 2nd ed. New York: Macmillan College Publishing.

Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

MULTIPLE CORRESPONDENCE Algorithms

Multiple Correspondence Analysis, also known as homogeneity analysis, quantifies nominal (categorical) data by assigning numerical values to the cases (objects) and categories, such that in the low-dimensional representation of the data, objects within the same category are close together and objects in different categories are far apart. Each object is as close as possible to the category points of categories that apply to the object. In this way, the categories divide the objects into homogeneous subgroups. Variables are considered homogeneous when they classify objects that are in the same categories into the same subgroups.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of analysis cases (objects)
n_w	Weighted number of analysis cases: $\sum_{i=1}^{n} w_i$
n_{tot}	Total number of cases (analysis + supplementary)
w_i	Weight of object *i*; $w_i = 1$ if cases are unweighted; $w_i = 0$ if object *i* is supplementary.
W	Diagonal $n_{tot} \times n_{tot}$ matrix, with w_i on the diagonal.
m	Number of analysis variables
m_w	Weighted number of analysis variables ($m_w = \sum_{j=1}^{m} v_j$)
m_{tot}	Total number of variables (analysis + supplementary)
H	The data matrix (category indicators), of order $n_{tot} \times m_{tot}$, after discretization, imputation of missings, and listwise deletion, if applicable.
p	Number of dimensions

For variable *j*; $j = 1, \ldots, m_{tot}$

v_j	Variable weight; $v_j = 1$ if weight for variable *j* is not specified or if variable *j* is supplementary
k_j	Number of categories of variable *j* (number of distinct values in $\mathbf{h}_j$, thus, including supplementary objects)
$\mathbf{G}_j$	Indicator matrix for variable *j*, of order $n_{tot} \times k_j$

The elements of $\mathbf{G}_j$ are defined as $i = 1, \ldots, n_{tot}; r = 1, \ldots, k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i\text{th object is in the } r\text{th category of variable } j \\ 0 & \text{when the } i\text{th object is not in the } r\text{th category of variable } j \end{cases}$$

$\mathbf{D}_j$ Diagonal $k_j \times k_j$ matrix, containing the weighted univariate marginals; i.e., the weighted column sums of $\mathbf{G}_j$ ($\mathbf{D}_j = \mathbf{G}'_j\mathbf{W}\mathbf{G}_j$)

$\mathbf{M}_j$ Diagonal $n_{tot} \times n_{tot}$ matrix, with diagonal elements defined as

$$m_{(j)ii} = \begin{cases} 0 & \text{when the } i\text{th observation is missing and missing strategy variable } j \text{ is passive} \\ 0 & \text{when the } i\text{th object is in } r\text{th category of variable } j \text{ and } r\text{th category is only} \\ & \text{used by supplementary objects (i.e. when} d_{(j)rr} = 0) \\ v_j & \text{otherwise} \end{cases}$$

$\mathbf{M}_*$ $\Sigma_j \mathbf{M}_j$

The quantification matrices and parameter vectors are:

$\mathbf{X}$ Object scores, of order $n_{tot} \times p$

$\mathbf{X}_w$ Weighted object scores ($\mathbf{X}_w = \mathbf{W}\mathbf{X}$)

$\mathbf{X}^n$ $\mathbf{X}$ normalized according to requested normalization option

$\mathbf{Y}_j$ Category quantifications, of order $k_j \times p$.

Note: The matrices $\mathbf{W}$, $\mathbf{G}_j$, $\mathbf{M}_j$, $\mathbf{M}_*$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Discretization

Discretization is done on the unweighted data.

Multiplying

First, the original variable is standardized. Then the standardized values are multiplied by 10 and rounded, and a value is added such that the lowest value is 1.

Ranking

The original variable is ranked in ascending order, according to the alphanumerical value.

Grouping into a specified number of categories with a normal distribution

First, the original variable is standardized. Then cases are assigned to categories using intervals as defined in Max (1960).

Grouping into a specified number of categories with a uniform distribution

First the target frequency is computed as divided by the number of specified categories, rounded. Then the original categories are assigned to grouped categories such that the frequencies of the grouped categories are as close to the target frequency as possible.

Grouping equal intervals of specified size

First the intervals are defined as lowest value + interval size, lowest value + 2*interval size, etc. Then cases with values in the *k*th interval are assigned to category *k*.

Imputation of Missing Values

When there are variables with missing values specified to be treated as active (impute mode or extra category), then first the k_j's for these variables are computed before listwise deletion. Next the category indicator with the highest weighted frequency (mode; the smallest if multiple modes exist), or $k_j + 1$ (extra category) is imputed. Then listwise deletion is applied if applicable. And then the k_j's are adjusted.

Configuration

MULTIPLE CORRESPONDENCE can read a configuration from a file, to be used as the initial configuration or as a fixed configuration in which to fit variables.

For an initial configuration see step 1 in the Optimization section.

A fixed configuration **X** is centered and orthonormalized as described in the optimization section in step 3 (with **X** instead of **Z**) and step 4 (except for the factor $n_w^{1/2}$), and the result is postmultiplied with $\mathbf{\Lambda}^{1/2}$ (this leaves the configuration unchanged if it is already centered and orthogonal). The analysis variables are set to supplementary and variable weights are set to one. Then MULTIPLE CORRESPONDENCE proceeds as described in the Supplementary Variables section.

Objective Function Optimization

The MULTIPLE CORRESPONDENCE objective is to find object scores **X** and a set of $\mathbf{Y}_j$ (for *j*=1,...,*m*) — the underlining indicates that they may be restricted in various ways — so that the function

$$\sigma(\mathbf{X};\mathbf{Y}) = (n_w p)^{-1} \sum_j \text{tr}\Big((\mathbf{X} - \mathbf{G}_j \mathbf{Y}_j)' \mathbf{M}_j \mathbf{W} (\mathbf{X} - \mathbf{G}_j \mathbf{Y}_j) \Big)$$

is minimal, under the normalization restriction $\mathbf{X}'\mathbf{M}_*\mathbf{W}\mathbf{X} = n_w m_w \mathbf{I}$ (**I** is the *p*×*p* identity matrix). The inclusion of $\mathbf{M}_j$ in $\sigma(\mathbf{X};\mathbf{Y})$ ensures that there is no influence of passive missing values (missing values in variables that have missing option passive, or missing option not specified). $\mathbf{M}_*$ contains the number of active data values for each object. The object scores are also centered; that is, they satisfy $\mathbf{u}'\mathbf{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ with **u** denoting an *n*-vector with ones.

Optimization is achieved by executing the following iteration scheme:

1. Initialization
2. Update category quantifications
3. Update object scores
4. Orthonormalization
5. Convergence test: repeat (2) through (4) or continue
6. Rotation

These steps are explained below.

Initialization

If an initial configuration is not specified, the object scores **X** are initialized with random numbers. Then **X** is orthonormalized (see step 4) so that $\mathbf{u}'\mathrm{M}_*\mathbf{WX} = \mathbf{0}$ and $\mathbf{X}'\mathrm{M}_*\mathbf{WX} = n_w m_w \mathbf{I}$, yielding $\mathbf{X}_w^+$.

Update Category Quantifications; Loop Across Analysis Variables

With fixed current values $\mathbf{X}_w^+$ the unconstrained update of $\mathbf{Y}_j$ is

$$\tilde{\mathbf{Y}}_j = \mathbf{D}_j^{-1}\mathbf{G}'_j\mathbf{X}_w^+$$

Update Object Scores

First the auxiliary score matrix **Z** is computed as

$$\mathbf{Z} \leftarrow \Sigma_j \mathrm{M}_j \mathbf{G}_j \underline{\mathbf{Y}}_j^+$$

and centered with respect to **W** and M_*:

$$\mathbf{X}^* = \left(\mathbf{I} - \mathrm{M}_*\mathbf{uu}'\mathrm{W}/\left(\mathrm{u}'\mathrm{M}_*\mathrm{Wu}\right)\right)\mathbf{Z}$$

These two steps yield locally the best updates when there would be no orthogonality constraints.

Orthonormalization

To find an M_*-orthonormal $\mathbf{X}^+$ that is closest to $\mathbf{X}^*$ in the least squares sense, we use for the Procrustes rotation (Cliff, 1966) the singular value decomposition $m_w^{1/2}\mathbf{M}_*^{-1/2}\mathbf{W}^{1/2}\mathbf{X}^* = \mathbf{K\Lambda}^{1/2}\mathbf{L}'$, then yields $n_w^{1/2}m_w^{1/2}\mathbf{M}_*^{-1/2}\mathbf{W}^{1/2}\mathbf{KL}'$ -orthonormal weighted object scores: $\mathbf{X}_w^+ \leftarrow n_w^{1/2}m_w\mathbf{M}_*^{-1}\mathbf{WX}^*\mathbf{L\Lambda}^{-1/2}\mathbf{L}'$, and $\mathbf{X}^+ = \mathbf{W}^{-1}\mathbf{X}_w^+$. The calculation of **L** and **Λ** is based on tridiagonalization with Householder transformations followed by the implicit QL algorithm (Wilkinson, 1965).

Convergence Test

The difference between consecutive values of the quantity

$$\text{TFIT} = (pn_w)^{-1}\sum_j v_j \text{tr}\left(\mathbf{Y}'_j\mathbf{D}_j\mathbf{Y}_j\right)$$

is compared with the user-specified convergence criterion ε - a small positive number. It can be shown that TFIT = $m_w - \sigma(\mathbf{X};\mathbf{Y})$. Steps (2) through (4) are repeated as long as the loss difference exceeds ε.

After convergence TFIT is also equal to tr $\left(\mathbf{\Lambda}^{1/2}\right)$, with $\mathbf{\Lambda}$ as computed in step (4) during the last iteration. (See also Model Summary, and Correlations Transformed Variables for interpretation of $\mathbf{\Lambda}^{1/2}$).

Rotation

To achieve principal axes orientation, $\mathbf{X}^+$ is rotated with the matrix **L**. Then step (2) is executed, yielding the rotated quantifications.

Supplementary Objects

To compute the object scores for supplementary objects, after convergence steps (2) and (3) are repeated, with the zero's in **W** temporarily set to ones in computing **Z** and $\mathbf{X}^+$. If a supplementary object has missing values, passive treatment is applied.

Supplementary Variables

The quantifications for supplementary variables are computed after convergence by executing step (2) once.

Diagnostics

The following diagnostics are available.

Maximum Rank (may be issued as a warning when exceeded)

The maximum rank $p_{\max}$ indicates the maximum number of dimensions that can be computed for any dataset. In general

$$p_{\max} = \min\left(n-1, \left(\sum_{j\in J} k_j\right) - m\right)$$

if there are no variables with missing values to be treated as passive. If variables do have missing values to be treated as passive, the maximum rank is

$$p_{\max} = \min\left(n-1, \left(\sum_{j\in J} k_j\right) - \max\left(m_1, 1\right)\right)$$

with m_1 the number of variables without missing values to be treated as passive.

Here k_j is exclusive supplementary objects (that is, a category only used by supplementary objects is not counted in computing the maximum rank). Although the number of nontrivial dimensions may be less than $p_{\max}$ when m=2, MULTIPLE CORRESPONDENCE does allow dimensionalities all the way up to $p_{\max}$. When, due to empty categories in the actual data, the rank deteriorates below the specified dimensionality, the program stops.

Descriptives

The descriptives tables gives the weighted univariate marginals and the weighted number of missing values (system missing, user defined missing, and values less than or equal to 0) for each variable.

Fit and Loss Measures

When the HISTORY option is in effect, the following fit and loss measures are reported:

Fit (VAF). This is the quantity TFIT as defined in step (5).

Loss. This is $\sigma(\mathbf{X};\mathbf{Y})$.

Model Summary

Model summary information consists of Cronbach's alpha, the variance accounted for, and the inertia.

Cronbach's Alpha

Cronbach's Alpha per dimension (s=1,...,p):

$$\alpha_s = m_w\left(\lambda_s^{1/2} - 1\right) / \left(\lambda_s^{1/2}\left(m_w - 1\right)\right)$$

Total Cronbach's Alpha is

$$\alpha = m_w\left(\Sigma_s\lambda_s^{1/2} - 1\right)/\Sigma_s\lambda_s^{1/2}\left(m_w - 1\right)$$

with λ_s the sth diagonal element of $\mathbf{\Lambda}$ as computed in step (4) during the last iteration.

Variance Accounted For

Variance Accounted For per dimension (s=1,...,p):

$$\text{VAF}_s = n_w^{-1}\sum_{j\in J} v_j \text{tr}\left(\mathbf{y}_{(j)s}^{'}\mathbf{D}_j\mathbf{y}_{(j)s}\right), \text{(\% of variance is VAF1}_s \times 100/m_w\text{)},$$

Eigenvalue per dimension:

$\lambda_s^{1/2}$=VAF$_s$,

with λ_s the sth diagonal element of $\mathbf{\Lambda}$ as computed in step (4) during the last iteration. (See also Optimization step (5), and Correlations Transformed Variables for interpretation of $\mathbf{\Lambda}^{1/2}$).

The Total Variance Accounted For is the mean over dimensions. So, the total eigenvalue is

tr $\left(\mathbf{\Lambda}^{1/2}\right)$ =$p^{-1}\Sigma_s$VAF$_s$.

If there are no passive missing values, the eigenvalues $\mathbf{\Lambda}^{1/2}$ are those of the correlation matrix (see the Correlations and Eigenvalues section) weighted with variable weights:

$r_{jj}^{\text{W}} = v_j r_{jj}$, and $r_{jl}^{\text{W}} = r_{lj}^{\text{W}} = v_j^{1/2} r_{jl}$

If there are passive missing values, then the eigenvalues are those of the matrix $m_w \mathbf{Q}'_{\text{C}} \mathbf{M}_*^{-1} \mathbf{Q}_{\text{C}}$, with $\mathbf{Q}_{\text{C}} = n_w^{-1/2}\left(\mathbf{I} - \mathbf{M}_*\mathbf{uu}'\mathbf{W}/\left(\mathbf{u}'\mathbf{M}_*\mathbf{Wu}\right)\right)\mathbf{Q}$, (for $\mathbf{Q}$ see the Correlations and Eigenvalues section) which is not necessarily a correlation matrix, although it is positive semi-definite. This matrix is weighted with variable weights in the same way as $\mathbf{R}$.

Inertia

The inertia per dimension is the eigenvalue per dimension divided by m_w. The total inertia is the total eigenvalue divided by m_w.

Correlations and Eigenvalues

Before transformation

$\mathbf{R} = n_w^{-1}\mathbf{H}'_{\text{C}}\mathbf{W}\mathbf{H}_{\text{C}}$, with $\mathbf{H}_{\text{C}}$ weighted centered and normalized $\mathbf{H}$. For the eigenvalue decomposition of $\mathbf{R}$ (to compute the eigenvalues), first row j and column j are removed from $\mathbf{R}$ if j is a supplementary variable, and then r_{ij} is multiplied by $(v_i v_j)^{1/2}$.

If passive missing treatment is applicable for a variable, missing values are imputed with the variable mode, regardless of the passive imputation specification.

After transformation

After transformation, p correlation matrices are computed (s=1,...,p):

$\mathbf{R}_{(s)} = n_w^{-1}\mathbf{Q}'_{(s)}\mathbf{W}\mathbf{Q}_{(s)}$,

with $\mathbf{q}_{(s)j} = n_w^{1/2}\mathbf{G}_j\mathbf{Y}_{(j)s}\left(\mathbf{Y}'_{(j)s}\mathbf{D}_j\mathbf{Y}_{(j)s}\right)^{-1/2}$.

Usually, for the higher eigenvalues, the first eigenvalue of $\mathbf{R}_{(s)}$ is equal to $\lambda_s^{1/2}$ (see Model Summary section). The lower values of $\mathbf{\Lambda}^{1/2}$ are in most cases the second or subsequent eigenvalues of $\mathbf{R}_{(s)}$.

If there are missing values, specified to be treated as passive, the mode of the quantified variable or the quantification of an extra category (as specified in syntax; if not specified, default (mode) is used) is imputed before computing correlations. Then the eigenvalues of the correlation matrix do not equal $\mathbf{\Lambda}^{1/2}$ (see Model Summary section). The quantification of an extra category is computed as

$$\mathbf{Y}_{(j)_{(k_j+1)s}} = \left(\sum_{i\in I} w_i\right)^{-1} \sum_{i\in I} w_i x_{is},$$

with *I* an index set recording which objects have missing values.

For the eigenvalue decomposition of **R** (to compute the eigenvalues), first row *j* and column *j* are removed from **R** if *j* is a supplementary variable, and then r_{ij} is multiplied by $(v_i v_j)^{1/2}$.

Discrimination measures

The discrimination measures are the dimensionwise variances of the quantified variables, which are equal to the dimensionwise squared correlations of the quantified variables with the object scores. For variable *j* and dimension the discrimination measure is

$$\text{Discr}_{js} = n_w^{-1}\mathbf{y}'_{(j)s}\mathbf{D}_j\mathbf{y}_{(j)s}$$

which is equal to the squared correlation between $\mathbf{G}_j\mathbf{y}_{(j)s}$ and $\mathbf{x}_s$.

Object Scores

If $\mathbf{\Lambda}^{1/2}$ gives the eigenvalues, then $\mathbf{\Lambda}^{1/4}$ gives the singular values, that can be used to spread the inertia over the object scores **X** and the category quantifications **Y**. During the optimization phase, variable principal normalization is used, then $\mathbf{X}^{\text{n}} = \mathbf{X}$ and $\mathbf{Y}^{\text{n}} = \mathbf{Y}$, else $\mathbf{X}^{\text{n}} = \mathbf{X}\left(m_w^{-1}\mathbf{\Lambda}\right)^{a/4}$ and $\mathbf{Y}^{\text{n}} = \mathbf{Y}\left(m_w^{-1}\mathbf{\Lambda}\right)^{1/4(b-1)}$, with *a*=(1+*q*)/2, *b*=(1−*q*)/2, and *q* any real value in the closed interval [-1,1], except for independent normalization: then there is no *q* value and *a*=*b*=1. *q*=1 is equal to variable principal normalization, *q*=-1 is equal to object principal normalization, *q*=0 is equal to symmetrical normalization.

Mass

The mass of object *i* is

$$\text{Mass}_i = \frac{m_{*ii}}{\text{tr}(\mathbf{M}_*\mathbf{W})}$$

Inertia

The inertia of object *i* is

$$\text{Inertia}_i = \frac{1}{m_{*ii}} \sum_{j,h_{ij}\neq 0} \frac{v_j}{d_{(j)h_{ij}}} - \text{Mass}_i$$

where $d_{(j)h_{ij}}$ is the frequency of the category of object *i* on variable *j*, and $h_{ij} \neq 0$ indicates to exclude a variable if object *i* has a missing value on the variable and the missing option for the variable is passive.

Contribution of point to inertia of dimension

The contribution of object *i* to the inertia of dimension *s* is

$$\text{Contribution}_{is} = \frac{m_{*ii} x_{is}^2}{n_w m_w}$$

Contribution of dimension to inertia of point

The contribution of dimension *s* to the inertia of object *i* is

$$\text{Contribution}_{si} = \frac{\frac{m_{*ii} x_{is}^2}{n_w m_w} \text{Inertia}_s}{\text{Inertia}^i}$$

Quantifications

The quantifications are the centroid coordinates. If a category is only used by supplementary objects (i.e. treated as a passive missing), the centroid coordinates for this category are computed as

$$\mathbf{y}_{(j)r} = n_w^{1/2} n_{jr}^{-1} \sum_{i \in I} \mathbf{x}_i \mathbf{\Lambda}^{1/4(b-1)}$$

where $\mathbf{y}_{(j)r}$ is the *r*th row of $\mathbf{Y}_j$, n_{jr} is the number of objects that have category *r*, and I is an index set recording which objects are in category *r*.

Mass

The mass of category *r* of variable *j* is

$$\text{Mass}_{(j)r} = \frac{d_{(j)rr}}{\text{tr}(\mathbf{M}_* \mathbf{W})}$$

Inertia

The inertia of category *r* of variable *j* is

$$\text{Inertia}_{(j)r} = \frac{\sum_{i=1}^{n} w_i m_{*ii} g_{(j)ir}}{d_{(j)rr}} - \text{Mass}_{(j)r}$$

if there are no missing values with missing option passive, this is equal to $\frac{1}{m_w} - \text{Mass}_{(j)r}$, and then the total inertia for variable *j* is $\frac{v_j(k_j-1)}{m_w}$.

Contribution of point to inertia of dimension

The contribution of category *r* of variable *j* to the inertia of dimension *s* is

$$\text{Contribution}_{(j)rs} = \frac{d_{(j)rr}\frac{y^2_{(j)rs}}{n_w m_w}}{\text{Inertia}_s}$$

the total contribution of variable *j* to the inertia of dimension *s* is $v_j\frac{\text{Discr}_{js}}{\lambda_s^{1/2}}$.

Contribution of dimension to inertia of point

The contribution of dimension *s* to the inertia of category *r* of variable *j* is

$$\text{Contribution}_{s(j)r} = \frac{d_{(j)rr}\frac{y^2_{(j)rs}}{n_w m_w}}{\text{Inertia}_{(j)r}}$$

Residuals

Plots per dimension are produced of $\mathbf{G}_j\mathbf{y}^{\text{n}}_{(j)s}$ against the approximation $\mathbf{x}^{\text{n}}_s$.

References

Cliff, N. 1966. Orthogonal rotation to congruence. *Psychometrika*, 31, 33–42.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Max, J. 1960. Quantizing for minimum distortion. *Proceedings IEEE (Information Theory)*, 6, 7–12.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem*. Oxford: Clarendon Press.

Multiple Imputation Algorithms

Multiple imputation imputes missing values multiple times. This algorithm only considers the imputation phase. See “Multiple Imputation: Pooling Algorithms” for the algorithm for combining analysis results of multiply imputed data sets.

Univariate and multivariate methods are given here. Univariate methods are used in situations where only the variable to be imputed has missing values, and all variables used as predictors in the imputation have no missing values. Multivariate methods are used in situations where variables are used both as dependents and predictors during imputation.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$\mathbf{x}$	Set of variables that have no missing values.
y_{ij}	The data value for case *i*, variable *j*. It may be missing.
Y_j^{obs}	The collection of observed values of variable *j*.
Y_j^{mis}	The collection of missing values of variable *j*.
$Y^{obs} = \left(Y_1^{obs}, ..., Y_J^{obs}\right)$	The collection of all observed data.
$Y^{mis} = \left(Y_1^{mis}, ..., Y_J^{mis}\right)$	The collection of all missing data.
f_i	Frequency (replication) weight for case *i*. Must be integer.
$F = diag\left(f_1, ..., f_n\right)$	Frequency weight matrix, diagonal with case frequency weight on the diagonal.
w_i	Regression or analysis weight for case i.
$W = diag\left(w_1, ..., w_n\right)$	Regression weight matrix.
n	The total number of cases. Each case may represent more than one observation due to frequency (replication) weights.
$N = \sum_{i=1}^{n} f_i$	The total number of observations.
$N_w = \sum_{i=1}^{n} w_i f_i$	The total weight.

Univariate Methods

y: the variable to be imputed, has missing values.

x: predictor variables, no missings.

Linear Regression

The variable to be imputed, *y*, is a continuous variable and is to be used as the dependent variable in the regression model. Both frequency and regression weights are accepted.

Model $y_i = \mathbf{x}'_i\beta + e_i$ with $e_i \sim N\left(0, \frac{\sigma^2}{w_i}\right)$ is used.

Prior: $\Pr\left(\beta, \log\sigma^2\right) \propto 1$, or equivalently $\Pr\left(\beta, \sigma^2\right) \propto 1/\sigma^2$

Using the complete cases, fit the regression model, assuming that all redundant parameters are removed if there are any. Denote the fitted parameters as $\left(\hat\beta, \hat\sigma^2\right)$ such that

$$\hat\beta = \left(X_c' F_c W_c X_c\right)^{-1} X_c' F_c W_c Y_c$$
$$\hat\sigma^2 = \left(Y_c - X_c\hat\beta\right)' F_c W_c \left(Y_c - X_c\hat\beta\right) / \left(N_{obs} - p\right)$$

where $N_{obs} = \sum_{i\in obs(Y)} f_i$ is the number of complete cases, *p* is the number of parameters, and Y_c, X_c, F_c, W_c are the dependent vector, design matrix and frequency weight, regression weight matrix for complete cases.

The posterior distributions are:

$$\beta|\sigma^2, Y_c, X_c \sim N\left(\hat\beta, \left(X_c' F_c W_c X_c\right)^{-1}\sigma^2\right)$$
$$\sigma^2|Y_c, X_c \sim \left(N_{obs} - p\right)\hat\sigma^2/\chi^2_{N_{obs}-p}$$

Let A be the upper triangular matrix of Cholesky decomposition $\left(X_c' F_c W_c X_c\right)^{-1} = A'A$.

Draw parameters from the posterior distributions.

- Draw $\left(\sigma^*\right)^2$: draw a random value *u* from $\chi^2_{N_{obs}-p}$ then $\left(\sigma^*\right)^2 = \left(N_{obs} - p\right)\hat\sigma^2/u$.
- Draw β^* : draw *p* independent N(0,1) values to create a random vector *v*, $\beta^* = \hat\beta + \sigma^* A'\mathbf{v}$.

Impute missing values. For *i* in mis(Y), draw z_i from N(0,1); imputation is $y_i^* = \mathbf{x}_i'\beta^* + \frac{\sigma}{\sqrt{w_i}}^* z_i$.

Repeat the drawing of parameters and imputation of missing values to generate multiple imputations.

Incorporate restrictions

Using the linear regression method, a continuous variable may have an imputed value well outside the range of observed values, so the imputed values of continuous variables can be restricted to fall within a user-specified range, *R*. When an imputed value falls outside *R*, the algorithm draws another imputed value until a value is drawn within *R* or r_1 draws have been made (the maximum number of tries allowed for drawing each missing case under the given parameter). If the r_1 limit is reached, a new set of parameters are drawn from the posterior distributions (discarding any successfully imputed values for this variable during this imputation) and the process of imputing missing values is repeated until a set of imputed values is obtained for this variable and this imputation or r_2 sets of parameters have been drawn (the maximum number of tries allowed for drawing parameters).

If the r_2 limit is reached, the algorithm stops and issues an error.

Predictive Mean Matching

This is the same as the linear regression method, but with the following changes.

Replace the impute missing values step of linear regression by the following:

Calculate $\hat{Y}^{obs} = \left\{ \hat{y}_i^{obs} = \mathbf{x}_i^{'}\beta^* : i \in obs\,(Y) \right\}$. For *i* in mis(Y):

- ▶ $y_i^* = \mathbf{x}_i^{'}\beta^*$;
- ▶ Among Y^{obs}, find the observation whose corresponding predicted values are closest to y_i^*;
- ▶ Pick that one as the imputation.

Logistic Regression

The variable to be imputed, *y*, is a categorical variable with *K* categories taking values 1, 2, …, *K*, and is used as dependent variable in the logistic regression model. In the following, $p_{\mathbf{x}}(k) = \Pr\,(y = k|\mathbf{x})$, and $p_i(k) = p_{\mathbf{x}_i}(k)$ for case *i*.

Model: $\log \frac{p_{\mathbf{x}}(k)}{p_{\mathbf{x}}(K)} = \mathbf{x}^{'}\beta_k$ for *k* = 1, …, *K*−1.

Prior: $\Pr\,(\beta) \propto 1$, where $\beta^{'} = \left(\beta_1^{'}, ..., \beta_{K-1}^{'}\right)$

Using the complete cases, fit the logistic regression model with user specified frequency and analysis/regression weight variables. Denote the fitted parameter vector and its variance matrix as $\hat{\beta}, \hat{V}\left(\hat{\beta}\right)$. The posterior distribution is approximately: $\beta|Y_c, X_c \sim N\left(\hat{\beta}, \hat{V}\right)$. Let A be the upper triangular matrix of the Cholesky decomposition. $\hat{V} = A'A$.

Draw parameters from the posterior distributions: draw β^*: draw length (β) independent N(0,1) values to create a random vector **z**, then $\beta^* = \hat{\beta} + A\mathbf{z}$.

Impute missing values. For *i* in mis(Y):

- ▶ Calculate $p_i(k) = \frac{\exp\left(\mathbf{x}_i^{'}\beta_k^*\right)}{1+\sum_{j=1}^{K-1}\exp\left(\mathbf{x}_i^{'}\beta_j^*\right)}$ for $k = 1, \cdots, K-1$.
- ▶ Draw a random value *u* from uniform distribution [0,1].
- ▶ Imputation is $y_i^* = \begin{cases} 1 & u < P_i\,(1) \\ k & P_i\,(k-1) \le u < P_i\,(k) \\ K & u \ge P_i\,(K-1) \end{cases}$ where $P_i\,(k) = \sum_{j=1}^{k} p_i(j)$.

Repeat the drawing of parameters and imputation of missing values to generate multiple imputations.

Multivariate Methods

Multivariate methods apply to situations in which multiple variables have missing values. Patterns of missing values are important here because a fast non-iterative procedure can be used for monotone missing patterns. For general missing patterns, the fully conditional specification (FCS) is available. This is an iterative MCMC method.

Monotone Method

Missing patterns are monotone if the variables can be ordered such that, for each case, all earlier variables are observed if the later variable is observed. This method also assumes that the parameters of individual imputation models have independent priors.

Let $Y_1, ..., Y_K$ be variables with missing values in the sorted monotone order such that Y_1 has the smallest number of missing values. Let *X* be the set of variables without missing values. Starting from Y_1, sequentially use univariate method with the previous *Y* variables and *X* variables as predictors to impute.

- ► Given X, Y_1^{obs} and imputation model for Y_1, impute Y_1^{mis} by univariate method *m* times to get *m* complete variable $Y_1^{*(1)}, ..., Y_1^{*(m)}$.
- ► For *l* = 1, …, *m*, given $X, Y_1^{*(l)}, Y_2^{obs}$ and imputation model for Y_2, impute Y_2^{mis} by univariate method once to get $Y_2^{*(l)}$.
- ► For l = 1, …, *m*, given $X, Y_1^{*(l)}, Y_2^{*(l)}, Y_3^{obs}$ and imputation model for Y_3, impute Y_3^{mis} by univariate method once to get $Y_3^{*(l)}$.
- ► Continue until last variable Y_K^{mis} is imputed.

Notes:

- ■ The imputation model for variable Y_j can only use variables from $X, Y_1, ..., Y_{j-1}$ as predictors. In the case of no *X* variables, a constant model for Y_1 is used.
- ■ The posterior distribution used to draw parameters for imputing Y_j^{mis} doesn't depend on previously imputed values $\left\{\left\{Y_k^{*(l)}\right\}_{l=1}^{m}\right\}_{k=1}^{j-1}$.

Fully Conditional Specification (FCS)

In this method, an imputation model for each variable with missing values is specified. This method is an iterative MCMC procedure. In each iteration, it sequentially imputes missing values starting from the first variable with missing values.

- ► Set initial values for missing values in all variables $Y_1^{(0)}, ..., Y_K^{(0)}$ (see below).

- At iteration *t*, for *j* = 1 to *K*: Given $X, Y_1^{(t)}, ..., Y_{j-1}^{(t)}, Y_{j+1}^{(t-1)}, ..., Y_K^{(t-1)}$; that is, the most recently imputed values of all other variables, $X, Y_2^{(t-1)}, ..., Y_K^{(t-1)}$ for j =1, and $X, Y_1^{(t)}, ..., Y_{K-1}^{(t)}$ for *j* = *K*, use a univariate method to impute all missing values in the *j*th variable, $Y_j^{(t)}$.
- Continue iterations until the maximum number of iterations is reached.

We create multiple imputations by the multiple chain method; that is, we repeat above steps *m* times to get *m* imputations. Each chain starts with a different seed for random numbers and different initial values.

Initial Values

For a continuous variable with missing values, use the non-missing values to find its sample mean and standard deviation, then fill in the missing values with random draws from a normal distribution with mean and standard deviation equal to the sample values, limited within the range of the observed minimum and maximum values.

For a categorical variable with missing values, use the non-missing values to find the observed proportion of each category, then fill in the missing values with random draws from a multinomial distribution with category probabilities equal to the observed category proportions.

Assessment of Convergence

For each imputation and each iteration, missing values are imputed for each variable. Let $Y_j^{*mis(l,t)}$ be the vector of imputed values of Y_j^{mis} at iteration *t*, imputation *l*. For each (*l*, *t*), calculate the sample mean and standard deviation of $Y_j^{*mis(l,t)}$:

$$m_j^{(l,t)} = \text{mean}\left(Y_j^{*mis(l,t)}\right)$$
$$s_j^{(l,t)} = \sqrt{var\left(Y_j^{*mis(l,t)}\right)}$$

Sequence plots of $m_j^{(l,t)}$ versus *t* and $s_j^{(l,t)}$ versus *t* are useful in assessing convergence. If there are 5 imputations, then there will be 5 lines (different color) in the same plot. On convergence, for each variable *j*, the traces of different *l* should be intermingled with each other without showing any definite trends, and the variance between different sequences is no larger than the variance with each individual sequence. When frequency and analysis weights are involved, the mean and standard deviation are calculated using the weights as well.

Automatic Selection of Imputation Method

If automatic selection of the imputation method is selected, the method is chosen as follows:

- If the pattern of missing values is monotone, then the monotone method is used.
- Otherwise, the fully conditional specification (FCS) method is used.

Note: only main effects models are used during automatic selection.

Special Situations

When the variable to be imputed is constant over all its observed values, we use this constant to impute its missing values.

Missing Values

The following cases are not used during imputation.

- Cases with every variable missing.
- Cases with zero/negative replication or analysis weight.

References

Raghunathan, T. E., J. M. Lepkowski , J. van Hoewyk , and P. Solenberger . 2001. A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey Methodology* , 27, 85–95.

Rubin, R. B. 1987. *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley and Sons, Inc..

Schafer, J. L. 1997. *Analysis of Incomplete Multivariate Data*. London: Chapman and Hall.

Van Buuren, S. 2007. Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research*, 16, 219–242.

Van Buuren, S., H. C. Boshuizen, and D. L. Knook. 1999. Multiple imputation of missing blood pressure covariates in survival analysis. *Statistics in Medicine*, 18, 681–694.

Multiple Imputation: Pooling Algorithms

Analysis of missing values consists of two sequential steps: analysis of each individual complete data set to create multiple analysis results and then combining (pooling) these multiple analysis results. This algorithm only considers combining the multiple analysis results assuming that multiple complete datasets are created and the analysis of each individual complete dataset is complete. See "Multiple Imputation Algorithms" for the algorithm for creating multiply imputed data sets. See the algorithm of the analysis you're performing for details on the analysis an individual complete data set.

Notation

The following notation is used throughout this chapter unless otherwise stated:

m	The number of multiply imputed sets of complete data. $m \geq 2$ is assumed.
$\mathbf{Q} = (Q_1, \ldots, Q_k)'$	Parameter vector (with k elements) to be estimated.
$\hat{\mathbf{Q}}^{(i)} = \left(\hat{Q}_1^{(i)}, \ldots, \hat{Q}_k^{(i)}\right)'$	Estimated parameter vector using the i-th set of completed data, i=1,...,m.
$\mathbf{U}^{(i)}$	Estimated covariance matrix of $\hat{\mathbf{Q}}^{(i)}$
$\overline{\mathbf{Q}}$	The final combined estimate of $\mathbf{Q}$

Rubin's Rules

Across all the complete datasets, it is assumed that:

- the model of the same effects in the same order is fit,
- a categorical variable has the same set of categories and the reference category is the same.

Assuming that each individual analysis result $\{\hat{\mathbf{Q}}^{(i)}, \mathbf{U}^{(i)}\}_{i=1}^{m}$ is available, the goal is to derive the final combined result based on these m individual results.

Combining Results after Multiple Imputation

The final estimate of $\mathbf{Q}$ is simply the average of individual ones:

$$\overline{\mathbf{Q}} = \frac{1}{m}\sum_{i=1}^{m}\hat{\mathbf{Q}}^{(i)}$$

The estimated total variance is

$$\mathbf{T} = \overline{\mathbf{U}} + \left(1 + \frac{1}{m}\right)\mathbf{B}$$

where $\mathbf{B}$ and $\overline{\mathbf{U}}$ are respectively the between-imputation and average within-imputation variance calculated by

$$\mathbf{B} = \frac{1}{m-1}\sum_{i=1}^{m}\left(\hat{\mathbf{Q}}^{(i)} - \overline{\mathbf{Q}}\right)\left(\hat{\mathbf{Q}}^{(i)} - \overline{\mathbf{Q}}\right)'$$

$$\overline{\mathbf{U}} = \frac{1}{m}\sum_{i=1}^{m}\mathbf{U}^{(i)}$$

Special Situations

Redundant parameters. Standard procedures set redundant parameter estimates at 0 and variance/covariance as missing. If a parameter is redundant across all imputations, then the combined parameter is still redundant. If there is a parameter that is redundant in some imputations but not in others, this causes an error. The reason is that the combined results depend on the order of effects in the model (for example *x1*,*x2*,*x3* or *x3*,*x1*,*x2* when *x3*=*x1*+*x2* holds in some imputations but not in others) which makes the combined results arbitrary and useless.

Different sets of parameters. There may be situations in which some model coefficients occur in some model fits but not in others (for example, a certain combination of two categorical variables occurs in some complete datasets but not in others). If the parameters across imputations are different, this causes an error. The reason is that the combined results depend on the choice of reference categories of categorical predictors which makes the results arbitrary and useless.

Missing Elements. If there are any missing elements in $\left\{\hat{\mathbf{Q}}^{(i)}, \mathbf{U}^{(i)}\right\}_{i=1}^{m}$, then we will only use the non-missing part to do calculations.

Scalar Q

If Q is a scalar (k=1), then

$$\left(\overline{Q} - Q\right)/\sqrt{T}$$

has an approximate Student's *t* distribution with degrees of freedom

$$\nu_m = (m-1)\left[1 + r^{-1}\right]^2$$

where r is the relative increase in variance due to non-response

$$r = \frac{\left(1 + m^{-1}\right)B}{\overline{U}}$$

The fraction of missing information about Q due to missing values is

$$\lambda = \frac{r + 2/\left(\nu_m + 3\right)}{r + 1}$$

The relative efficiency (RE) of using the finite *m* imputation estimator, rather than using an infinite number for the fully efficient imputation, in units of variance, is approximately

$$RE = (1 + \lambda/m)^{-1}$$

Vector Q

If the number of imputations *m* is big enough (at least 50,000), then

$$\frac{1}{k}\left(\overline{\mathbf{Q}} - \mathbf{Q}\right)^T \mathbf{T}^{-1}\left(\overline{\mathbf{Q}} - \mathbf{Q}\right)$$

has an approximate *F* distribution with *k* numerator degrees of freedom and denominator degrees of freedom

$$\nu = (m-1)\left[1 + \overline{r}^{-1}\right]^2$$

where

$$\overline{r} = \frac{1}{k}\left(1 + m^{-1}\right) trace\left(\mathbf{B}\overline{\mathbf{U}}^{-1}\right)$$

But for small *m* (this usually is the case in practice), this approximation is bad because the estimate of **B** is unstable and when *m* ≤ *k*, **B** is not even full rank. Alternatively, we assume that **B** and $\overline{\mathbf{U}}$ are proportional to one another. Under this assumption, a more stable estimate of total variance is

$$\tilde{\mathbf{T}} = (1 + \overline{r})\,\overline{\mathbf{U}}$$

and

$$\frac{1}{k}\left(\overline{\mathbf{Q}} - \mathbf{Q}\right)^T \tilde{\mathbf{T}}^{-1}\left(\overline{\mathbf{Q}} - \mathbf{Q}\right) \sim F_{k,\tilde{\nu}}$$

has an approximate *F* distribution with *k* numerator degrees of freedom and denominator degrees of freedom $\tilde{\nu}$ (Li, Raghunathan and Rubin (1991)), let $t = k(m-1)$,

$$\tilde{\nu} = \begin{cases} t\left(1 + k^{-1}\right)\left(1 + \overline{r}^{-1}\right)^2/2 & t \le 4 \\ 4 + (t-4)\left[1 + \left(1 - 2t^{-1}\right)\overline{r}^{-1}\right]^2 & t > 4 \end{cases}$$

Note:

- When *k*=1, $\tilde{T}$ reduces to *T* for a scalar statistic.
- When *k*=1, $\tilde{\nu} = v_m$ if $m \le 5$, and $\tilde{\nu} < v_m$ if $m > 5$.

Output Statistics

Other than $\overline{\mathbf{Q}}, \overline{\mathbf{U}}, \mathbf{B}, \mathbf{T}$, we are also interested in some statistics for each individual element of **Q** (for example the vector of regression coefficients). For the *j*th element of **Q**, we calculate the following. Please notice that the following listed quantities do not use the off diagonal elements of matrix **T**, or **B**, or **U**. They are the same as treating each element as scalar and calculating them separately. In the following t_ν denotes a random variable following a Student's *t* distribution with degrees of freedom ν, and $t_{v,1-\alpha/2}$ denotes the $100\,(1-\alpha/2)$ percentile of the distribution such that $\Pr\left[t_\nu \le t_{v,1-\alpha/2}\right] = 1-\alpha/2$.

Estimate: $\overline{Q}_j$

Standard error: $se_j = \sqrt{T_{jj}}$

Degrees of freedom: ν_j

Confidence interval: $\overline{Q}_j \pm t_{\nu_j,1-\alpha/2} \quad se_j$

t-value: $\tilde{t}_j \ = \overline{Q}_j / se_j$

p-value for hypothesis test : $H_0 : Q_j = 0$: $p = 2\Pr\left[t_{\nu_j} \ge \left|\tilde{t}_j\right|\right]$

Relative increase in variance due to non-response: $r_j = \left(1+m^{-1}\right)\frac{B_{jj}}{\overline{U}_{jj}}$

Fraction of missing information: λ_j

Relative efficiency (RE): $RE_j = \ \left(1 + \lambda_j/m\right)^{-1}$

Hypothesis Tests

The *p*-value for testing $H_0 : \mathbf{Q} = \mathbf{Q}_0$ is

$$p = \Pr\left(F_{k,\nu} \ge F\right)$$

where

$$F = \frac{1}{k}\left(\overline{\mathbf{Q}} - \mathbf{Q}_0\right)^T \tilde{\mathbf{T}}^- \left(\overline{\mathbf{Q}} - \mathbf{Q}_0\right)\Big]$$

$$k = rank\left(\tilde{\mathbf{T}}\right)$$
$$\nu = \bar{\nu}$$

We will also apply this test to scalar statistics. Note that for *k* = 1 this test does not necessarily reduce to the equivalent student *t* test mentioned in the scalar Q section due to possibly different degrees of freedom.

General Linear Contrast of Model Parameters

The above test can be applied to test hypotheses about linear combinations of parameters. For a given matrix **L** and a vector **K**, $H_0 : \mathbf{L}\beta = \mathbf{K}$ can be tested, where β is a model parameter vector (regression coefficients for example). Let $\mathbf{Q} = \mathbf{L}\beta$, $\hat{\mathbf{Q}}_i = \mathbf{L}\hat{\beta}_i$, and $\mathbf{U}_i = \mathbf{L}Var\left(\hat{\beta}_i\right)\mathbf{L}'$. This test becomes testing $H_0 : \mathbf{Q} = \mathbf{K}$

It is likely that only **K**, $\hat{\mathbf{Q}}_i$ and diagonal elements of $\mathbf{U}_i$ are available, so the simultaneous test of H_0 cannot be done. Instead, we will test each row of H_0 separately. Denote the *l*-th row hypothesis of H_0 as $H_{0l} : Q_l = K_l$. Let p_l be the *p*-value for testing H_{0l} alone. If multiple comparisons are requested, the *p*-values are adjusted as usual.

In multivariate GLM, there is a parameter matrix, **B**, instead of a vector. In multivariate GLM $H_0 : \mathbf{LBM} = \mathbf{K}$ can be tested for the given matrices **L**, **M**, **K**. Where possible, we separately test each element of the hypotheses $H_{0ij} : Q_{ij} = \mathbf{l}'_i\mathbf{B}\mathbf{m}_j = k_{ij}$ where $\mathbf{l}'_i$ is the *i*-th row vector of matrix **L**, and k_{ij} is the *ij*th element of vector **K**. Again, if multiple comparisons are requested, the *p*-values are adjusted as usual.

References

Li, K. H., T. E. Raghunathan, and D. B. Rubin. 1991. Large-Sample Significance Levels from Multiply Imputed Data Using Moment-Based Statistics and an F Reference Distribution. *Journal of the American Statistical Association*, 86, 1065–1073.

Schafer, J. L. 1997. *Analysis of Incomplete Multivariate Data*. London: Chapman and Hall.

MVA Algorithms

The Missing Value procedure provides descriptions of missing value patterns; estimates of means, standard deviations, covariances, and correlations (using a listwise, pairwise, EM, or regression method); and imputation of values by either EM or regression.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$\mathbf{X}$	Data matrix
x_{ij}	Value of the *i*th case, *j*th variable
v	Number of variables
m	Number of cases
n_i	Number of nonmissing values of the *i*th variable
n_{ij}	Number of nonmissing value pairs of the *i*th and *j*th variables
n_c	Number of complete cases
J	Index of all variables
$J_{\#}=J$(condition)	Index of variables satisfying "condition"
I	Index of all cases
$I(k_1, \ldots, k_l)$	Index of cases at which variables are not missing
$I(J)$	Index of complete cases
$\mathbf{a} = [a_i]$	Vector whose *i*th element is a_i
$\mathbf{A} = [a_{ij}]$	Matrix whose *i*th row, *j*th column element is a_{ij}

Example to Illustrate Notation

$$\mathbf{X} = \begin{bmatrix} 43 & 76 & 34 \\ . & 45 & 72 \\ 44 & 15 & 52 \\ . & & 65 \\ & & 43 \\ . & 12 & \\ 54 & 67 & \\ 43 & & 34 \end{bmatrix}$$

$x_{2,3} = 72$	The 2nd row, 3rd element
$v = 3$	Number of variables
n=7	Number of cases
$n_2 = 5$	Number of nonmissing values in the 2nd variable
$n_{2,3} = 4$	Number of nonmissing value pairs in the 2nd and 3rd variables
$n_c = 3$	Number of complete cases
J={1,2,3}	Index of variables

J(2 or more missing)={1,2}	The 1st and 2nd variables have two or more missing values
I={1,2,3,4,5,6,7}	Index of cases
I(2)={1,2,3,6,7}	Index of cases at which the 2nd variable is not missing
I(2,3)={1,2,3,7}	Index of cases at which the 2nd and 3rd variables are not missing
I(*J*)={1,7}	Index of complete cases
$\overline{x}_2 = 43.0$	The 2nd element of the vector $\mathbf{\overline{x}} = [\overline{x}_1, x_2, \overline{x}_3]$

Univariate Statistics

The index *j* refers to quantitative variables.

Mean

$$\mathbf{\overline{x}} = [\overline{x}_j] = [\Sigma_i x_{ij}/n_j; i \in I(j)]$$

Standard Deviation

$$\hat{\sigma} = [\hat{\sigma}_j] = \left[\left(\Sigma_i (x_{ij} - \overline{x}_j)^2/(n_j - 1)\right)^{1/2}; i \in I(j)\right]$$

Extreme Low

$$\text{NL} = [nl_j] = [\text{number of } x_{ij} \text{ values} < low\_limit_j]$$

Extreme High

$$\text{NH} = [nh_j] = [\text{number of } x_{ij} \text{ values} > high\_limit_j]$$

where

$$low\_limit_j = \begin{cases} \overline{x}_j - 2 * \hat{\sigma}_j & \text{if } v * n * \log_{10}(n) > 150,000 \\ 25\text{th percentile of the } j\text{th varible} & \text{if } v * n * \log_{10}(n) \le 150,000 \end{cases}$$

and

$$high\_limit_j = \begin{cases} \overline{x}_j + 2 * \hat{\sigma}_j & \text{if } v * n * \log_{10}(n) > 150,000 \\ 75\text{th percentile of the } j\text{th variable} & \text{if } v * n * \log_{10}(n) \le 150,000 \end{cases}$$

Separate Variance T Test

The index *k* refers to quantitative variables, and index *j* refers to all variables.

$$t_{jk} = \frac{\overline{x}^P_{jk} - \overline{x}_{k|\text{variable } j \text{ is missing}}}{\left(\frac{\hat{\sigma}^P_{jk}}{n_{jk}} + \frac{\hat{\sigma}_{k|\text{variable } j \text{ is missing}}}{n_{kk} - n_{jk}} \right)^{1/2}}$$

where $\overline{x}^P_{jk}$ and $\hat{\sigma}^P_{jk}$ are defined in “Pairwise Statistics”.

$$\text{df}_{jk} = \frac{\left(\frac{\hat{\sigma}^P_{jk}}{n_{jk}} + \frac{\hat{\sigma}_{k|\text{variable } j \text{ is missing}}}{n_{kk} - n_{jk}} \right)^2}{\frac{\left(\hat{\sigma}^P_{jk}\right)^2}{n_{jk} - 1} + \frac{\left(\hat{\sigma}_{k|\text{variable } j \text{ is missing}}\right)^2}{n_{kk} - n_{jk} - 1}} p(\text{2-tail})_{jk} = 1 - 2 * |0.5 - \text{tcdf}(t_{jk}, \text{df}_{jk})|$$

where “tcdf” is the *t* cumulative distribution function

Listwise Statistics

The indices *j* and *k* refer to quantitative variables.

Mean

$$\mathbf{\overline{x}}^L = \left[\overline{x}^L_j\right] = \left[\Sigma_i x_{ij} / n_c ; i \in I(J)\right]$$

Covariance

$$\mathbf{C}^L = \left[c^L_{jk}\right] = \left[\Sigma_i \left(x_{ij} - \overline{x}^L_j\right) * \left(x_{ik} - \overline{x}^L_k\right) / (n_c - 1); i \in I(J)\right]$$

Correlation

$$\mathbf{R}^L = \left[r^L_{jk}\right] = \left[c^L_{jk} / \left(c^L_{jj} * c^L_{kk}\right)^{1/2}\right]$$

Pairwise Statistics

The indices *j* and *k* refer to quantitative variables, and *l* refers to all variables.

Mean

$$\mathbf{\overline{X}}^P = \left[\overline{x}^P_{lk}\right] = \left[\Sigma_i x_{ik} / n_{lk} ; i \in I(l, k)\right]$$

Standard Deviation

$$\hat{\sigma}^P = \left[\hat{\sigma}^P_{lk}\right] = \left[\left(\Sigma_i\left(x_{ik} - \overline{x}^P_{lk}\right)^2 / (n_{lk} - 1)\right)^{1/2}; i \in I(l,k)\right]$$

Covariance

$$\mathbf{C}^P = \left[c^P_{jk}\right] = \left[\Sigma_i\left(x_{ik} - \overline{x}^P_{jk}\right) * \left(x_{ij} - \overline{x}^P_{kj}\right) / \left(n_{jk} - 1\right); i \in I(j,k)\right]$$

Correlation

$$\mathbf{R}^P = \left[r^P_{jk}\right] = \left[c^P_{jk} / \left(\hat{\sigma}^P_{jk} * \hat{\sigma}^P_{kj}\right)\right]$$

Regression Estimated Statistics

The indices *j* and *k* refer to quantitative variables, and *l* refers to predictor variables.

Estimates of Missing Values

$$x^R_{ij} = \begin{cases} x_{ij} & \text{if } x_{ij} \text{ is not missing} \\ \text{regression estimated} x_{ij} & \text{if } x_{ij} \text{ is missing} \end{cases}$$

where the regression estimated x_{ij} is

$$x^R_{ij} = \beta_{0,ij} + \Sigma_l \beta_{l,ij} * x_{il} + \epsilon_{ij} \quad l \in J_1 = J(l : x_{il} \text{not missing and} l \neq j)$$

where:

$[\beta_{0,ij}, \beta_{l,ij}]$ is computed from Diag$\left(\overline{\mathbf{X}}^P\right) = \left[\overline{x}^P_{jj}\right]$ and by pivoting on the "best" "q" of the J_1 diagonals of $\mathbf{C}^P$.

"best" is forward stepwise selected.

"q" is less than or equal to the user-specified maximum number of predictors; it may also be limited by the user-specified *F*-to-enter limit.

"ϵ_{ij}" is the optional random error term, as specified:

1. residual of a randomly selected complete case
2. random normal deviate, scaled by the standard error of estimate
3. random t(df) deviate, scaled by the standard error of estimate, df is specified by the user
4. no error term adjustment

Note that for each missing value x_{ij}, a unique set of regression coefficients $(\beta_{0,ij}, \beta_{l,ij})$ and error terms ϵ_{ij} is computed.

Mean

$$\overline{\mathbf{x}}^R = \left[\overline{x}_j^R\right] = \left[\Sigma_i x_{ij}^R / n; i \in I\right]$$

Covariance

$$\mathbf{C}^R = \left[c_{jk}^R\right] = \left[\Sigma_i \left(x_{ij}^R - \overline{x}_j^R\right) * \left(x_{ik}^R - \overline{x}_k^R\right) / (n-1); i \in I\right]$$

Correlation

$$\mathbf{R}^R = \left[r_{jk}^R\right] = \left[c_{jk}^R / \left(c_{jj}^R * c_{kk}^R\right)^{1/2}\right]$$

EM Estimated Statistics

The indices *j* and *k* refer to quantitative variables, and *l* refers to predictor variables.

Estimates of Missing Values, Mean Vector, and Covariance Matrix

$$\overline{\mathbf{x}}_0 = \left[\overline{x}_j^0\right] = \text{Diag}\left(\overline{\mathbf{X}}^P\right) = \left[\overline{x}_{jj}^P\right]$$
$$\mathbf{C}_0 = \left[c_{jk}^0\right] = \mathbf{C}^P = \left[c_{jk}^P\right]$$

For *m*=1 to *M*, or until convergence is attained:

If x_{ij} is not missing then $x_{ij}^m = x_{ij}$.

If x_{ij} is missing then it is estimated in the *m*th iteration as:

$$x_{ij}^m = \beta_{0,ij}^{m-1} + \Sigma_l \beta_{l,ij}^{m-1} * x_{il}; \quad l \in J_2 = J(l : x_{il} \text{ is not missing and } l \neq j)$$

where $\left[\beta_{0,ij}^{m-1}, \beta_{l,ij}^{m-1}\right]$ is computed from $\overline{\mathbf{x}}_{m-1}$ and $\mathbf{C}_{m-1}$.

$$\overline{\mathbf{x}}_m = \left[\overline{x}_j^m\right] = \left[\Sigma_i w_i * x_{ij}^m / \Sigma_i w_i; \quad i \in I\right]$$

$$\mathbf{C}_m = \left[c_{jk}^m\right] = \left[\frac{\Sigma_i w_i * x_{ij}^m \left(x_{ij}^m - \overline{x}_j^m\right) * \left(x_{ik}^m - \overline{x}_k^m\right) + \Sigma_i \Sigma_s c_{j,s|J2}^{m-1}}{(n-1) * \Sigma_i w_i / n}; i \in J_2, s \notin J_2, \text{and } s \neq j\right]$$

where $c_{j,s|J2}^{m-1}$ is the *j*th row, *s*th element of the J_2 pivoted $\mathbf{C}_{m-1}$.

Note that some sources (Little & Rubin, 1987, for example) simply use *n* as the denominator of the formula for $\mathbf{C}_m$, which produces full maximum likelihood (ML) estimates. The formula used by MVA produces restricted maximum likelihood (REML) estimates, which are *n*/(*n*−1) times the ML estimates.

$$w_i = \begin{cases} 1 & \text{for multivariate normal} \\ \frac{1-\alpha+\alpha*\lambda^{1+p/2}*\exp\left((1-\lambda)*D^2/2\right)}{1-\alpha+\alpha*\lambda^{p/2}*\exp\left((1-\lambda)*D^2/2\right)} & \text{for contaminated normal} \\ (\mathrm{df}+p)/\left(\mathrm{df}+D^2\right) & \text{for} t(\mathrm{df}) \end{cases}$$

$$\begin{aligned} \alpha &= \text{ proportion of contamination} \\ \lambda &= \text{ ratio of standard deviations} \\ p &= \text{ number of predictors = number of indices in } J_2 \\ D^2 &= \text{Mahalanobis distance square of the current case from the mean} \\ &= \Sigma_{jk}\left(x_{ij}^m - \overline{x}_j^m\right) * \left(c_{jk}^m\right)^{-1} * \left(x_{ik}^m - \overline{x}_k^m\right) \end{aligned}$$

where $\left(c_{jk}^m\right)^{-1}$ is the *jk*th element of $\mathbf{C}_m^{-1}$.

Convergence

The algorithm is declared to have converged if, for all *j*,

$$\left|c_{jj}^m - c_{jj}^{m-1}\right| / c_{jj}^m \leq \text{CONVERGENCE}$$

Filled-In Data

$$\mathbf{X}_i^E = \left[x_{ij}^E\right] = \left[x_{ij}^{m'}\right]$$

where m' is the last value of *m*.

Mean

$$\overline{\mathbf{x}}^E = \left[\overline{x}_j^E\right] = \overline{\mathbf{x}}_{m'} = \left[\overline{x}_j^{m'}\right]$$

Covariance

$$\mathbf{C}^E = \left[c_{jk}^E\right] = \mathbf{C}_{m'} = \left[c_{jk}^{m'}\right]$$

Correlation

$$\mathbf{R}^E = \left[r_{jk}^E\right] = \left[c_{jk}^E / \left(c_{jj}^E * c_{kk}^E\right)^{1/2}\right]$$

Little’s MCAR Test

$$\chi^2_{\text{MCAR}} = \sum_{\text{each unique pattern}} (\text{no. of cases in pattern}) * (\text{MD})$$

$$\text{DF}_{\text{MCAR}} = \sum_{\text{each unique pattern}} (\text{no. of nonmissing variables}) - v$$

where

$$\text{MD} = \text{Mahalanobis } D^2 \text{ of pattern mean from } \overline{\mathbf{x}}^E$$

References

Dempster, A. P., N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B*, 39, 1–38.

Dixon, W. J. 1983. *BMDP statistical software*. Berkeley: University of California Press.

Little, R. J. A. 1988. A test of missing completely at random for multivariate data with missing values. *Journal of the American Statistical Association*, 83, 1198–1202.

Little, R. J. A., and D. B. Rubin. 1987. *Statistical analysis with missing data*. New York: John Wiley & Sons, Inc. .

Louise, T. A. 1982. Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society*, *B*, 44:2, 226–233.

Orchard, T., and M. A. Woodbury. 1972. . In: *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1,* Berkeley: Universityof California Press, 697–715.

Rubin, R. B. 1987. *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley and Sons, Inc..

NAIVE BAYES Algorithms

The Naive Bayes model is an old method for classification and predictor selection that is enjoying a renaissance because of its simplicity and stability.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 65-1
Notation

Notation	**Description**
J_0	Total number of predictors.
X	Categorical predictor vector **X'** = (X_1, ..., X_J), where J is the number of predictors considered.
M_j	Number of categories for predictor X_j.
Y	Categorical target variable.
K	Number of categories of Y.
N	Total number of cases or patterns in the training data.
N_k	The number of cases with Y= k in the training data.
N^j_{mk}	The number of cases with Y= k and X_j=m in the training data.
π_k	The probability for Y= k.
p^j_{mk}	The probability of X_j=m given Y= k.

Naive Bayes Model

The Naive Bayes model is based on the conditional independence model of each predictor given the target class. The Bayesian principle is to assign a case to the class that has the largest posterior probability. By Bayes' theorem, the posterior probability of Y given **X** is:

$$P(Y=k|\mathbf{X}=\mathbf{x}) = \frac{P(\mathbf{X}=\mathbf{x}|Y=k)P(Y=k)}{\sum_{i=1}^{K} P(\mathbf{X}=\mathbf{x}|Y=i)\,P(Y=i)}$$

Let X_1, ..., X_J be the J predictors considered in the model. The Naive Bayes model assumes that X_1, ..., X_J are conditionally independent given the target; that is:

$$P\left(\mathbf{X}=\mathbf{x}\,|Y=k) = \prod_{j=1}^{J} P(X_j = x_j|\,Y=k\right)$$

These probabilities are estimated from training data by the following equations:

$$\pi_k = P(Y=k) = \frac{N_k+\lambda}{N+K\lambda}$$

$$p^j_{mk} = P(X_j=m|Y=k) = \frac{N^j_{mk}+f}{\Sigma_{l=1}^{M_j} N^j_{lk} + M_j f}$$

Where N_k is calculated based on all non-missing Y, N^j_{mk} is based on all non-missing pairs of X_J and Y, and the factors λ and f are introduced to overcome problems caused by zero or very small cell counts. These estimates correspond to Bayesian estimation of the multinomial probabilities with Dirichlet priors. Empirical studies suggest $\lambda = f = \frac{1}{N}$(Kohavi, Becker, and Sommerfield, 1997).

A single data pass is needed to collect all the involved counts.

For the special situation in which J = 0; that is, there is no predictor at all, $P(Y = k|\mathbf{X} = \mathbf{x}) = P(Y = k)$. When there are empty categories in the target variable or categorical predictors, these empty categories should be removed from the calculations.

Preprocessing

The following steps are performed before building the Naive Bayes model.

Missing Values

A predictor is ignored if every value is missing or if it has only one observed category. A case is ignored if the value of the target variable or the values of all predictors are missing. For each case missing some, but not all, of the values of the predictors, only the predictors with nonmissing values are used to predict the case, as suggested in (Kohavi et al., 1997).

This implies the following equation:

$$P(\mathbf{X} = \mathbf{x}_i|Y = y_i) = \prod_{\{j: x_{ji} \text{not missing}\}} P(X_j = x_{ji}|Y = y_i)$$

This also implies the following equation for B(J) in average log-likelihood calculations:

$$B(J) = -\frac{1}{N}\sum_{i=1}^{N} \log\left(\sum_{k=1}^{K} \pi_k \prod_{\{j: x_{ji} \text{not missing}\}} p^j_{x_{ji}k}\right)$$

Where the log() term for case *i* is ignored if all the values of the predictors considered in the model are missing. For more information, see the topic “Average Log-likelihood”.

Continuous Variables

The Naive Bayes model assumes that the target and predictor variables are categorical. If there are continuous variables, they need to be discretized. There are many ways to discretize a continuous variable; the simplest is to divide the domain of a variable into equal width bins. This method performs well with the Naive Bayes model while no obvious improvement is found when complex methods are used (Hsu, Huang, and Wong, 2000).

Sometimes the equal width binning method may produce empty bins. In this case, empty bins are eliminated by changing bin boundary points. Let $b_1 < b_2 < ... b_n$ be the bin boundary points produced by the equal width binning method. The two end bins $(-\infty, b_1]$ and (b_n, ∞) are non-empty by design. Suppose that bin $(b_i, b_{i+1}]$ is empty, and suppose that the closest left non-empty bin has right boundary point b_j ($< b_i$) and the closest right non-empty bin has left boundary point b_k (>

b_i). Then empty bins are eliminated by deleting all boundary points from b_j to b_k, and setting a new boundary point at $(b_j+b_k)/2$.

Feature Selection

Given a total of J_0 predictors, the goal of feature selection is to choose a subset of J predictors using the Naive Bayes model (Natarajan and Pednault, 2001). This process has the following steps:

- Collect the necessary summary statistics to estimate all possible model parameters.
- Create a sequence of candidate predictor subsets that has an increasing number of predictors; that is, each successive subset is equal to the previous subset plus one more predictor.
- From this sequence, find the "best" subset.

Collect Summary Statistics

One pass through the training data is required to collect the total number of cases, the number of cases per category of the target variable, and the number of cases per category of the target variable for each category of each predictor.

Create the Sequence of Subsets

Start with an initial subset of predictors considered vital to the model, which can be empty. For each predictor not in the subset, a Naive Bayes model is fit with the predictor plus the predictors in the subset. The predictor that gives the largest average log-likelihood is added to create the next larger subset. This continues until the model includes the user-specified:

- Exact number of predictors

or

- Maximum number of predictors

Alternatively, the maximum number of predictors, J_{Max}, may be automatically chosen by the following equation:

$$J_{\text{Max}} = \min\left\{J_{\text{Must}} + \min\left\{100, \max\left(20, \tfrac{J_0}{5}\right)\right\}, J_0\right\}$$

where J_{Must} is the number of predictors in the initial subset.

Find the "Best" Subset

If you specify an exact number of predictors, the final subset in the sequence is the final model. If you specify a maximum number of predictors, the "best" subset is determined by one of the following:

- A test data criterion based on the average log-likelihood of the test data.
- A pseudo-BIC criterion that uses the average log-likelihood of the training data and penalizes overly complex models using the number of predictors and number of cases. This criterion is used if there are no test data.

Smaller values of these criteria indicate “better” models. The “best” subset is the one with the smallest value of the criterion used.

Test Data Criterion

$Q(J) = -\bar{l}_{\text{Test}}(J)$

Where $\bar{l}_{\text{Test}}(J)$ is the average log-likelihood for test data.

Pseudo-BIC Criterion

$Q(J) = -\bar{l}_{\text{Train}}(J) + \frac{1}{2}J\frac{\log(N)}{N}$

Where J denotes the number of predictors in the model, and $-\bar{l}_{\text{Train}}(J)$ is the average log-likelihood for training data.

Average Log-likelihood

The average (conditional) log-likelihood for data $\{\mathbf{x}_i, y_i\}_{i=1}^{N}$ with J predictors is

$$\begin{aligned}
\bar{l}(J) &= \frac{1}{N}\log L = \frac{1}{N}\sum_{i=1}^{N}\log P(Y=y_i|\mathbf{X}=\mathbf{x}_i) \\
&= \frac{1}{N}\sum_{i=1}^{N}\log P(Y=y_i) + \frac{1}{N}\sum_{i=1}^{N}\log P(\mathbf{X}=\mathbf{x}_i|Y=y_i) \\
&- \frac{1}{N}\sum_{i=1}^{N}\log\left(\sum_{k=1}^{K}P(\mathbf{X}=\mathbf{x}_i|Y=k)P(Y=k)\right) \\
&= \frac{1}{N}\sum_{k=1}^{K}N_k\log(\pi_k) + \frac{1}{N}\sum_{k=1}^{K}\sum_{j=1}^{J}\sum_{m=1}^{M_j}N_{mk}^{j}\log\left(p_{mk}^{j}\right) - \frac{1}{N}\sum_{i=1}^{N}\log\left(\sum_{k=1}^{K}\pi_k\prod_{j=1}^{J}p_{x_{ji}k}^{j}\right)
\end{aligned}$$

Let

$$A(J) = \frac{1}{N}\sum_{k=1}^{K}N_k\log(\pi_k) + \frac{1}{N}\sum_{k=1}^{K}\sum_{j=1}^{J}\sum_{m=1}^{M_j}N_{mk}^{j}\log\left(p_{mk}^{j}\right)$$

$$B(J) = -\frac{1}{N}\sum_{i=1}^{N}\log\left(\sum_{k=1}^{K}\pi_k\prod_{j=1}^{J}p_{x_{ji}k}^{j}\right)$$

then

$\bar{l}(J) = A(J) + B(J)$

Note: for the special situation in which J = 0; that is, there are no predictors,

$$\bar{l}(J) = \frac{1}{N}\sum_{i=1}^{N}\log P(Y=y_i) = \frac{1}{N}\sum_{k=1}^{K}N_k\log(\pi_k)$$

Calculation of average log-likelihood by sampling

When adding each predictor to the sequence of subsets, a data pass is needed to calculate B(J). When the data set is small enough to fit in the memory, this is not a problem. When the data set cannot fit in memory, this can be costly. The Naive Bayes model uses simulated data to calculate B(J). Other research has shown that this approach yields good results (Natarajan et al., 2001). The formula for B(J) can be rewritten as, for a data set of *m* cases:

$$B\left(J\right) = -\tfrac{1}{m}\sum_{i=1}^{m}\log\left(\sum_{k=1}^{K}\pi_k\prod_{j=1}^{J}p_{x_{ji}k}^{j}\right)$$

By default *m* = 1000.

Classification

The target category with the highest posterior probability is the predicted category for a given case.

$$\hat{y}(\mathbf{x}) = \arg\max_k\left\{P\left(Y = k|\mathbf{X} = \mathbf{x}\right)\right\} = \arg\max_k\left\{P\left(\mathbf{X} = \mathbf{x}|Y = k\right)P\left(Y = k\right)\right\}$$

Ties are broken in favor of the target category with greater prior probability π_k.

When cases being classified contain categories of categorical predictors that did not occur in the training data, these new categories are treated as missing.

Classification Error

If there is test data, the error equals the misclassification ratio of the test data. If there is no test data, the error equals the misclassification ratio of the training data.

References

Bayes, T. 1763. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society*, 53, 370–418.

Becker, B., R. Kohavi, and D. Sommerfield. 2001. Visualizing the Simple Bayesian Classifier. In: *Information Visualization in Data Mining and Knowledge Discovery,* U. Fayyad, G. Grinstein, and A. Wierse, eds. San Francisco: Morgan Kaufmann Publishers, 237–249.

Domingos, P., and M. J. Pazzani. 1996. Beyond Independence: conditions for the optimality of the simple Bayesian classifier. In: *Machine Learning: Proceedings of the Thirteenth International Conference,* L. Saitta, ed., 105–112.

Hsu, C., H. Huang, and T. Wong. 2000. Why Discretization Works for Naive Bayesian Classifiers. In: *Proceedings of the 17th International Conference on Machine Learning,* San Francisco: Morgan Kaufman, 399–406.

Kohavi, R., and D. Sommerfield. 1995. Feature subset selection using the wrapper model: Overfitting and dynamic search space topology. In: *The First International Conference on Knowledge Discovery and Data Mining,* Menlo Park,California: AAAI Press, 192–197.

Kohavi, R., B. Becker, and D. Sommerfield. 1997. Improving Simple Bayes. In: *Proceedings of the European Conference on Machine Learning*, , 78–87.

Natarajan, R., and E. Pednault. 2001. Using Simulated Pseudo Data to Speed Up Statistical Predictive Modeling from Massive Data Sets. In: *SIAM First International Conference on Data Mining*, .

NLR Algorithms

NLR produces the least square estimates of the parameters for models that are not linear in their parameters. Unlike in other procedures, the weight function is not treated as a case replicate in NLR.

Model

Consider the model

$$f = f(\mathbf{x}, \mathbf{\Theta})$$

where $\mathbf{\Theta}$ is a p×1 parameter vector, $\mathbf{x}$ is an independent variable vector, and *f* is a function of $\mathbf{x}$ and $\mathbf{\Theta}$.

Goal

Find the least square estimate $\mathbf{\Theta}^*$ of $\mathbf{\Theta}$ such that $\mathbf{\Theta}^*$ minimizes the objective function

$$F(\mathbf{\Theta}) = \mathbf{R}^{'}\mathbf{W}\mathbf{R}$$

where

$$\mathbf{R}^{'} = (R_1, \ldots, R_n)$$
$$R_i = y_i - f_i$$
$$f_i = f\left(x_i, \mathbf{\Theta}^*\right),\ i = 1, \ldots, n$$
$$\mathbf{W} = \text{Diag}(W_1, \ldots, W_n)$$

and *n* is the number of cases. For case *i*, y_i is the observed dependent variable, x_i is the vector of observed independent variables, W_i is the weight function which can be a function of $\mathbf{\Theta}$.

The gradient of F at $\mathbf{\Theta}$ is defined as

$$\nabla F = 2\mathbf{J}^{'}{}_{\cdot j}\mathbf{W}\mathbf{R}$$

where $\mathbf{J}_{\cdot j}$ is the *j*th column of the $n \times p$ Jacobian matrix **J** whose (i, j)th element is defined by

$$J_{ij} = \frac{R_i}{2W_i}\frac{\partial W_i}{\partial \Theta_j} - \frac{\partial f_i}{\partial \Theta_j}$$

Estimation

The modified Levenberg-Marquardt algorithm (Moré, 1977) that is contained in MINPACK is used in NLR to solve the objective function.

Given an initial value $\mathbf{\Theta}^{(0)}$ for $\mathbf{\Theta}$, the algorithm is as follows:

At stage $k + 1, k = 0, 1, 2, \ldots$

- Compute

$f_i^{(k)} = f_i\left(\mathbf{\Theta}^{(F)}\right)$, $R_i^{(k)} = y_i - f_i^{(k)}$, $F_k = F\left(\mathbf{\Theta}^{(k)}\right)$ and $J^{(k)} = J\left(\mathbf{\Theta}^{(k)}\right)$

- Choose an appropriate non-negative scalar such that

  $F(\mathbf{\Theta}^{(k)} + h_k) < F_k$

  where

  $h_k = -\left(\mathbf{J}^{(k)'}\mathbf{J}^{(k)} + \alpha_k \mathbf{I}\right)^{-1}\mathbf{J}^{(k)'}\mathbf{R}^{(k)}$

- Set

  $\mathbf{\Theta}^{(k+1)} = \mathbf{\Theta}^{(k)} + h_k$

  and compute $\mathbf{J}^{(k+1)}, \mathbf{R}^{(k+1)}, \mathbf{W}^{(k+1)}, F_{k+1}$

- Check the following conditions:

1. $1 - (F_{k-1}/F_k) < \epsilon_1 \ (SSCON)$
2. For every element of h_k

   $\left|h_{ki}/\Theta_i^{(k)}\right| < \epsilon_2(PCON)$

3. $k + 1 \geq ITER$ (maximum number of iterations)
4. For every parameter $\mathbf{\Theta}_j$, the gradient of *F* at $\mathbf{\Theta}_j$, ∇F_j, is evaluated at $\mathbf{\Theta}^{(k+1)}$ by checking

   $\left|r_j^{(k+1)}\right| < \epsilon_2(RCON)$

   where $r_j^{(k+1)}$ is the correlation between the *j*th column $\mathbf{J}_j^{(k+1)}$ of $\mathbf{J}^{(k+1)}$ and $\mathbf{W}^{(k+1)}\mathbf{R}^{(k+1)}$.

   If any of these four conditions is satisfied, the algorithm will stop. Then the final parameter estimate $\mathbf{\Theta}^*$

   $\mathbf{\Theta}^* = \mathbf{\Theta}^{(k+1)}$

   and the termination reason is reported. Otherwise, iteration continues.

Statistics

When the estimation algorithm terminates, the following statistics are printed.

Parameter Estimates and Standard Errors

The asymptotic standard error of Θ_j^* is estimated by the square root of the *j*th diagonal element a_{jj} of **A**, where

$\mathbf{A} = \frac{F(\mathbf{\Theta}^*)}{n-p}(\mathbf{J}^*\mathbf{W}^*\mathbf{J}^*)^{-1}$

and $\mathbf{J}^*$ and $\mathbf{W}^*$ are the Jacobian matrix **J** and weight function **W** evaluated at $\mathbf{\Theta}^*$, respectively.

Asymptotic 95% Confidence Interval for Parameter Values

$$\Theta_j^* \pm t(0.975, n-p)a_{ii}$$

Asymptotic Correlation Matrix of the Parameter Estimates

$$\mathbf{C} = \mathbf{D}^{-1/2}\mathbf{A}\mathbf{D}^{-1/2}$$

where

$$\mathbf{D} = \text{Diag}(a_{11,\ldots,}a_{pp})$$

and a_{ii} is the *i*th diagonal element of **A**.

Analysis of Variance Table

Source	**df**	**Sum of Squares**
Residual	*n*–*p*	$F(\mathbf{\Theta}^*)$
Regression	*p*	$SS\text{uncorrected} - F(\mathbf{\Theta}^*)$
Uncorrected Total	*n*	$SS\text{uncorrected}$
Corrected Total	*n*–1	$SS\text{uncorrected} - \overline{y}^2\sum_{i=1}^{n} W_i(\mathbf{\Theta}^*)$

where

$$SS\text{uncorrected} = \sum_{i=1}^{n} W_i(\mathbf{\Theta}^*)y_i^2$$

$$\overline{y} = \left(\sum_{i=1}^{n} W_i(\mathbf{\Theta}^*)y_i\right) / \left(\sum_{i=1}^{n} W_i(\mathbf{\Theta}^*)\right)$$

References

Moré, J. J. 1977. The Levenberg-Marquardt algorithm: implementation and theory in numerical analysis. In: *Lecture Notes in Mathematics* , G. A. Watson, ed. Berlin: Springer-Verlag, 105–116.

NOMREG Algorithms

The purpose of the Multinomial Logistic Regression procedure is to model the dependence of a nominal categorical response on a set of discrete and/or continuous predictor variables.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The response variable, which takes integer values from 1 to J.
J	The number of categories of the nominal response.
m	The number of subpopulations.
$\mathbf{X}^A$	$m \times p^A$ matrix with vector-element x_i^A, the observed values at the ith subpopulation, determined by the independent variables specified in the command.
$\mathbf{X}$	$m \times p$ matrix with vector-element x_i, the observed values of the location model's independent variables at the ith subpopulation.
f_{ijs}	The frequency weight for the sth observation which belongs to the cell corresponding to $Y=j$ at subpopulation i.
n_{ij}	The sum of frequency weights of the observations that belong to the cell corresponding to $Y=j$ at subpopulation i.
N	The sum of all n_{ij}'s.
π_{ij}	The cell probability corresponding to $Y=j$ at subpopulation i.
$\log(\pi_{ij}/\pi_{ik})$	The logit of response category j to response category k.
$\beta_j = (\beta_{j1}, ..., \beta_{jp})'$	$p \times 1$ vector of unknown parameters in the j-th logit (i.e., logit of response category j to response category J).
p	Number of parameters in each logit. $p \geq 1$.
p_j^{nr}	Number of non-redundant parameters in logit j after maximum likelihood estimation. $p \geq p_j^{nr} \geq 0$.
p^{nr}	The total number of non-redundant parameters after maximum likelihood estimation. $p^{nr} = \sum_{j=1}^{J-1} p_j^{nr}$.
$\mathbf{B} = \left(\beta_1', ..., \beta_{J-1}'\right)'$	$(J-1)\,p \times 1$ vector of unknown parameters in the model.
$\hat{\mathbf{B}} = \left(\hat{\beta}_1', ..., \hat{\beta}_{J-1}'\right)'$	The maximum likelihood estimate of **B**.
$\hat{\pi}_{ij}$	The maximum likelihood estimate of π_{ij}

Data Aggregation

Observations with negative or missing frequency weights are discarded. Observations are aggregated by the definition of subpopulations. Subpopulations are defined by the cross-classifications of either the set of independent variables specified in the command or the set of independent variables specified in the `SUBPOP` subcommand.

Let n_i be the marginal count of subpopulation i,

$$n_i = \sum_{j=1}^{J} n_{ij}$$

If there is no observation for the cell of *Y*=*j* at subpopulation *i*, it is assumed that $n_{ij} = 0$, provided that $n_i \neq 0$. A non-negative scalar $\delta \in [0,1)$ may be added to any zero cell (i.e., cell with $n_{ij} = 0$) if its marginal count n_i is nonzero. The value of δ is zero by default.

Data Assumptions

Let $(n_{i1}, ..., n_{iJ})^{\mathrm{T}}$be the $J \times 1$ vector of counts for the categories of *Y* at subpopulation. It is assumed that each $(n_{i1}, ..., n_{iJ})^{\mathrm{T}}$ is independently multinomial distributed with probability vector $(\pi_{i1}, ..., \pi_{iJ})^{\mathrm{T}}$ of dimension $J \times 1$ and fixed total n_i.

Model

NOMREG fits a generalized logit model that can also be used to model the results of 1-1 matched case-control studies.

Generalized Logit Model

In a Generalized Logit model, the probability π_{ij} of response category *j* at subpopulation *i* is

$$\pi_{ij} = \frac{\exp\left(\mathbf{x}'_i \beta_j\right)}{1 + \Sigma_{k=1}^{J-1} \exp\left(\mathbf{x}'_i \beta_k\right)}$$

where the last category *J* is assumed to be the reference category.

In terms of logits, the model can be expressed as

$$\log\left(\frac{\pi_{ij}}{\pi_{iJ}}\right) = \mathbf{x}'_i \beta_j$$

for *j* = 1, …, *J*–1.

When *J* = 2, this model is equivalent to the binary Logistic Regression model. Thus, the above model can be thought of as an extension of the binary Logistic Regression model from binary response to polytomous nominal response.

1-1Matched Case Control Model by Conditional Likelihood Approach

The above model can also be used to estimate the parameters in the conditional likelihood of the 1-1 Matched Case Control Model. In this case, let *m* be the number of matching pairs, x_{i1} be the vector of independent variables for the case and x_{i2} that for the control. The conditional log-likelihood for the *m* matched pairs is given by

$$l = \frac{\exp\left\{(\mathbf{x}_{i1} - \mathbf{x}_{i2})' \beta\right\}}{1 + \exp\left\{(\mathbf{x}_{i1} - \mathbf{x}_{i2})' \beta\right\}}$$

in which β is the vector of parameters for the difference between the values of independent variables of the case and those of the control. This conditional likelihood is identical to the unconditional log-likelihood of a binary (i.e., *k* = 2) logistic regression model when

- There is no intercept term in the model.
- The set of subpopulations is defined by the set of matching pairs.
- The independent variables in the model are set to equal to the differences between the values for the case and the control.
- The number of response categories is *J* = 2, and the value of the response is 1 (or a constant), i.e., *Y* = 1.

Log-likelihood

The log-likelihood of the model is given by

$$\begin{aligned} l(\mathbf{B}) &= \sum_{i=1}^{m}\sum_{j=1}^{J} n_{ij}\log(\pi_{ij}) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{J} n_{ij}\log\left(\frac{\exp\left(\mathbf{x}'_i\beta_j\right)}{1+\Sigma_{k=1}^{J-1}\exp\left(\mathbf{x}'_i\beta_k\right)}\right) \end{aligned}$$

A constant that is independent of parameters has been excluded here. The value of the constant is

$$c = \Sigma_{i=1}^{m}\log\{n_i!/(n_{i1}!\dots n_{iJ}!)\}$$

Parameter Estimation

Estimation of the model parameters proceeds as follows.

First and Second Derivatives of the Log-likelihood

For any *j* = 1, …, *J*–1, *s* = 1, …, *p*, the first derivative of *l* with respect to β_{js} is

$$\frac{\partial l}{\partial \beta_{js}} = \sum_{i=1}^{m} x_{is}\left(n_{ij} - n_i\pi_{ij}\right)$$

For any *j*, *j'*= 1, …, *J* –1, *s*, *t* = 1, …, *p*, the second derivative of *l* with respect to β_{js} and $\beta_{j't}$ is

$$\frac{\partial^2 l}{\partial \beta_{js}\partial \beta_{j't}} = -\sum_{i=1}^{m} n_i x_{is} x_{it}\pi_{ij}\left(\delta_{jj'} - \pi_{ij'}\right)$$

where $\delta_{jj'}$ = 1 if *j* = *j'*, 0 otherwise.

Maximum Likelihood Estimate

To obtain the maximum likelihood estimate of **B**, a Newton-Raphson iterative estimation method is used. Notice that this method is the same as Fisher-Scoring iterative estimation method in this model, since the expectation of the second derivative of l with respect to **B** is the same as the observed one.

Let $\partial l/\partial \mathbf{B}$ be the $(J-1)\,p \times 1$ vector of the first derivative of *l* with respect to **B**. Moreover, let $\left[\partial^2 l/\partial \mathbf{B}\partial \mathbf{B}\right]$ be the $(J-1)\,p \times (J-1)\,p$ matrix of the second derivative of *l* with respect to **B**. Notice that $-\left[\partial^2 l/\partial \mathbf{B}\partial \mathbf{B}\right] = \Sigma_{i=1}^{m} \mathbf{X}_i^* \Delta_i \mathbf{X}_i^{*\prime}$ where Δ_i is a $(J-1) \times (J-1)$ matrix

$$\Delta_i = n_i \left(\text{Diag}\left(\pi_i^{(-J)}\right) - \pi_i^{(-J)} \pi_i^{(-J)\prime}\right)$$

in which $\pi_i^{(-J)} = (\pi_{i1}, ..., \pi_{iJ-1})'$ and Diag($\pi_i^{(-J)}\big)$ is a $(J-1) \times (J-1)$ diagonal matrix of $\pi_i^{(-J)}$. Let $\mathbf{B}^{(\nu)}$ be the parameter estimate at iteration *v*, the parameter estimate $\mathbf{B}^{(\nu+1)}$ at iteration *v* + 1 is updated as

$$\mathbf{B}^{(\nu+1)} = \mathbf{B}^{(\nu)} + \xi \left(\sum_{i=1}^{m} \mathbf{X}_i^* \Delta_i^{(v)} \mathbf{X}_i^{*\prime}\right)^{-1} \frac{\partial l}{\partial \mathbf{B}^{(v)}}$$

and $\xi > 0$ is a stepping scalar such that $l\left(\mathbf{B}^{(\nu+1)}\right) - l\left(\mathbf{B}^{(\nu)}\right) \geq 0$, $\mathbf{X}^*$ is a $(J-1)\,p \times (J-1)$ matrix of independent vectors,

$$\mathbf{X_i^*} = \begin{pmatrix} \mathbf{x}_i & 0 & \cdots & 0 \\ 0 & \mathbf{x}_i & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{x}_i \end{pmatrix}$$

and $\Delta_i^{(\nu)}$ is Δ_i and $\partial l/\partial \mathbf{B}^{(\nu)}$ is $\partial l/\partial \mathbf{B}$, both evaluated at $\mathbf{B} = \mathbf{B}^{(\nu)}$.

Stepping

Use step-halving method if $l\left(\mathbf{B}^{(\nu+1)}\right) - l\left(\mathbf{B}^{(\nu)}\right) < 0$. Let *V* be the maximum number of steps in step-halving, the set of values of ξ is {1/2^v: *v* = 0, …, *V*−1}.

Starting Values of the Parameters

If intercepts are included in the model, set $\beta_j^{(0)} = \left(\beta_{j1}^{(0)}, 0, ..., 0\right)'$ where

$$\beta_{j1}^{(0)} = \log\left(\frac{\tilde{\pi}_{ij}}{\tilde{\pi}_{iJ}}\right) = \log\left(\frac{\sum_{i=1}^{m} n_{ij}}{\sum_{i=1}^{m} n_{iJ}}\right)$$

for *j* = 1, …, *J* −1.

If intercepts are not included in the model, set

$$\beta_j^{(0)} = (0, ..., 0)'$$

for $j = 1, \ldots, J-1$.

Convergence Criteria

Given two convergence criteria $\epsilon_k > 0$ and $\epsilon_p > 0$, the iteration is considered to be converged if one of the following criteria are satisfied:

1. $\left| l\left(\mathbf{B}^{(\nu+1)}\right) - l\left(\mathbf{B}^{(\nu)}\right) \right| < \epsilon_k$
2. $\max_i \left| \mathbf{B}_i^{(\nu+1)} - \mathbf{B}_i^{\nu} \right| < \epsilon_p$
3. The maximum above element in $\partial l / \partial \mathbf{B}^{(\nu+1)}$ is less than min(ϵ_l, ϵ_p).

Stepwise Variable Selection

Several methods are available for selecting independent variables. With the forced entry method, any variable in the variable list is entered into the model. The forward stepwise, backward stepwise, and backward entry methods use either the Wald statistic or the likelihood ratio statistic for variable removal. The forward stepwise, forward entry, and backward stepwise use the score statistic or the likelihood ratio statistic to select variables for entry into the model.

Forward Stepwise (FSTEP)

1. Estimate the parameter and likelihood function for the initial model and let it be our current model.
2. Based on the MLEs of the current model, calculate the score statistic or likelihood ratio statistic for every variable eligible for inclusion and find its significance.
3. Choose the variable with the smallest significance (p-value). If that significance is less than the probability for a variable to enter, then go to step 4; otherwise, stop FSTEP.
4. Update the current model by adding a new variable. If this results in a model which has already been evaluated, stop FSTEP.
5. Calculate the significance for each variable in the current model using LR or Wald's test.
6. Choose the variable with the largest significance. If its significance is less than the probability for variable removal, then go back to step 2. If the current model with the variable deleted is the same as a previous model, stop FSTEP; otherwise go to the next step.
7. Modify the current model by removing the variable with the largest significance from the previous model. Estimate the parameters for the modified model and go back to step 5.

Forward Only (FORWARD)

1. Estimate the parameter and likelihood function for the initial model and let it be our current model.
2. Based on the MLEs of the current model, calculate the score or LR statistic for every variable eligible for inclusion and find its significance.
3. Choose the variable with the smallest significance. If that significance is less than the probability for a variable to enter, then go to step 4; otherwise, stop FORWARD.
4. Update the current model by adding a new variable. If there are no more eligible variable left, stop FORWARD; otherwise, go to step 2.

Backward Stepwise (BSTEP)

1. Estimate the parameters for the full model that includes the final model from previous method and all eligible variables. Only variables listed on the BSTEP variable list are eligible for entry and removal. Let current model be the full model.
2. Based on the MLEs of the current model, calculate the LR or Wald's statistic for every variable in the BSTEP list and find its significance.
3. Choose the variable with the largest significance. If that significance is less than the probability for a variable removal, then go to step 5. If the current model without the variable with the largest significance is the same as the previous model, stop BSTEP; otherwise go to the next step.
4. Modify the current model by removing the variable with the largest significance from the model. Estimate the parameters for the modified model and go back to step 2.
5. Check to see any eligible variable is not in the model. If there is none, stop BSTEP; otherwise, go to the next step.
6. Based on the MLEs of the current model, calculate LR statistic or score statistic for every variable not in the model and find its significance.
7. Choose the variable with the smallest significance. If that significance is less than the probability for the variable entry, then go to the next step; otherwise, stop BSTEP.
8. Add the variable with the smallest significance to the current model. If the model is not the same as any previous models, estimate the parameters for the new model and go back to step 2; otherwise, stop BSTEP.

Backward Only (BACKWARD)

1. Estimate the parameters for the full model that includes all eligible variables. Let the current model be the full model.
2. Based on the MLEs of the current model, calculate the LR or Wald's statistic for all variables eligible for removal and find its significance.
3. Choose the variable with the largest significance. If that significance is less than the probability for a variable removal, then stop BACKWARD; otherwise, go to the next step.

4. Modify the current model by removing the variable with the largest significance from the model. Estimate the parameters for the modified model. If all the variables in the BACKWARD list are removed then stop BACKWARD; otherwise, go back to step 2.

Stepwise Statistics

The statistics used in the stepwise variable selection methods are defined as follows.

Score Function and Information Matrix

The score function for a model with parameter *B* is:

$$U(B) = \frac{\partial l(B)}{\partial B}$$

The (*j*,*s*)th element of the score function can be written as

$$\begin{aligned} [U(B)]_{js} &= \frac{\partial l(B)}{\partial \beta_{js}} \\ &= \sum_{i=1}^{m} x_{is}\left(n_{ij} - n_i \pi_{ij}\right) \end{aligned}$$

Similarly, elements of the information matrix are given by

$$\begin{aligned} [I(B)]_{js,j't} &= \frac{\partial^2 l(B)}{\partial \beta_{js} \partial \beta_{jt}} \\ &= -\sum_{i=1}^{m} n_i x_{is} x_{it} \pi_{ij}\left(\delta_{jj'} - \pi_{ij'}\right) \end{aligned}$$

where $\delta_{jj'}$ = 1 if $j = j^{'}$, 0 otherwise.

(Note that π_{ij} in the formula are functions of *B*)

Block Notations

By partitioning the parameter *B* into two parts, *B*$_1$ and *B*$_2$, the score function, information matrix, and inverse information matrix can be written as partitioned matrices:

$$\begin{aligned} U(B_1, B_2) &= \begin{pmatrix} U_1(B_1, B_2) \\ U_2(B_1, B_2) \end{pmatrix} \\ &= \begin{pmatrix} \frac{\partial l(B_1, B_2)}{\partial B_1} \\ \frac{\partial l(B_1, B_2)}{\partial B_2} \end{pmatrix} \end{aligned}$$

where $l(B_1, B_2) = l(B)$

$$\begin{aligned} I(B) &= I(B_1, B_2) \\ &= \begin{pmatrix} I_{11}(B_1, B_2) & I_{12}(B_1, B_2) \\ I_{21}(B_1, B_2) & I_{22}(B_1, B_2) \end{pmatrix} \\ &= \begin{pmatrix} \frac{\partial^2 l(B_1, B_2)}{\partial B_1 \partial B_1} & \frac{\partial^2 l(B_1, B_2)}{\partial B_1 \partial B_2} \\ \frac{\partial^2 l(B_1, B_2)}{\partial B_2 \partial B_1} & \frac{\partial^2 l(B_1, B_2)}{\partial B_2 \partial B_2} \end{pmatrix} \end{aligned}$$

$$J(B) = I(B_1, B_2)^{-} = \begin{pmatrix} J_{11}(B_1, B_2) & J_{12}(B_1, B_2) \\ J_{21}(B_1, B_2) & J_{22}(B_1, B_2) \end{pmatrix}$$

where

$$J_{11} = I_{11}^{-} + I_{11}^{-} I_{12} J_{22} I_{21} I_{11}^{-}$$
$$J_{12} = -I_{11}^{-} I_{12} J_{22}$$
$$J_{21} = J_{12}^{T}$$
$$J_{22} = \left[I_{22} - I_{21} I_{11}^{-} I_{12} \right]^{-}$$

Typically, B_1 and B_2 are parameters corresponding to two different sets of effects. The dimensions of the 1st and 2nd partition in U, I and J are equal to the numbers of parameters in B_1 and B_2 respectively.

Score Test

Suppose a base model with parameter vector B_{base} with the corresponding maximum likelihood estimate $\hat{B}_{base}$. We are interested in testing the significance of an extra effect E if it is added to the base model. For convenience, we will call the model with effect E the augmented model. Let B_E be the vector of extra parameters associated with the effect E, then the hypothesis can be written as

$H_0 : B_E = 0$v.s. $H_1 \quad B_E \neq \quad 0$

Using the block notations, the score function, information matrix and inverse information of the augmented model can be written as

$$U(B_{base}, B_E) = \begin{pmatrix} U_{base}(B_{base}, B_E) \\ U_E(B_{base}, B_E) \end{pmatrix}$$

$$I(B_{base}, B_E) = \begin{pmatrix} I_{base,base}(B_{base}, B_E) & I_{base,E}(B_{base}, B_E) \\ I_{E,base}(B_{base}, B_E) & I_{E,E}(B_{base}, B_E) \end{pmatrix}$$

$$J(B_{base}, B_E) = \begin{pmatrix} J_{base,base}(B_{base}, B_E) & J_{base,E}(B_{base}, B_E) \\ J_{E,base}(B_{base}, B_E) & J_{E,E}(B_{base}, B_E) \end{pmatrix}$$

Then the score statistic for testing our hypothesis will be

$$s = U_E\left(\hat{B}_{base}, 0\right)^T J_{E,E}\left(\hat{B}_{base}, 0\right) U_E\left(\hat{B}_{base}, 0\right)$$

where $U_E\left(\hat{B}_{base}, 0\right)$ and $J_{E,E}\left(\hat{B}_{base}, 0\right)$are the 2nd partition of score function and inverse information matrix evaluated at $B_{base} = \hat{B}_{base}$ and $B_E = 0$.

Under the null hypothesis, the score statistic s has a chi-square distribution with degrees of freedom equal to the rank of $J_{E,E}(B_1, B_2)$. If the rank of $J_{E,E}(B_1, B_2)$ is zero, then the score statistic will be set to 0 and the p-value will be 1. Otherwise, if the rank of $J_{E,E}(B_1, B_2)$ is r_E: $r_E > 0$, then the p-value of the test is equal to $1 - F(s; r_E)$ is the cumulative distribution function of a chi-square distribution with r_E degrees of freedom.

Computational Formula for Score Statistic

When we compute the score statistic *s*, it is not necessary to re-compute $U\left(\hat{B}_{base,}0\right)$ and $I\left(\hat{B}_{base},0\right)$ from scratch. The score function and information matrix of the base model can be reused in the calculation. Using the block notations introduced earlier, we have

$$U\left(\hat{B}_{base},0\right)=\begin{pmatrix}U_{base}\left(\hat{B}_{base},0\right)\\ U_E\left(\hat{B}_{base},0\right)\end{pmatrix}=\begin{pmatrix}U\left(\hat{B}_{base}\right)\\ U_E\left(\hat{B}_{base},0\right)\end{pmatrix}$$

and

$$I\left(\hat{B}_{base},0\right)=\begin{pmatrix}I\left(\hat{B}_{base}\right) & I_{base,E}\left(\hat{B}_{base},0\right)\\ I_{E,base}\left(\hat{B}_{base},0\right) & I_{E,E}\left(\hat{B}_{base},0\right)\end{pmatrix}$$

In stepwise logistic regression, it is necessary to compute one score test for each effect that are not in the base model. Since the 1st partition of $U\left(\hat{B}_{base},0\right)$ and $I\left(\hat{B}_{base},0\right)$ depend only on the base model, we only need to compute $U_E\left(\hat{B}_{base},0\right)$, $I_{base,E}\left(\hat{B}_{base},0\right)$ and $I_{E,E}\left(\hat{B}_{base},0\right)$ for each new effect.

If β_{js} is the *s*-th parameter of the *j*-th logit in B_{base} and β_{kt} is the *t*-th parameter of *k*-th logit in B_E, then the elements of $U_E\left(\hat{B}_{base},0\right)$, $I_{base,E}\left(\hat{B}_{base},0\right)$ and $I_{E,E}\left(\hat{B}_{base},0\right)$ can be expressed as follows:

$$\left[U_E\left(\hat{B}_{base},0\right)\right]_{kt}=\sum_{i=1}^{m}x_{it}\left(n_{ik}-n_i\hat{\pi}_{ik}\right)$$

$$\left[I_{E,E}\left(\hat{B}_{base},0\right)\right]_{kt,k't'}=-\sum_{i=1}^{m}n_ix_{it}x_{it'}\hat{\pi}_{ik}\left(\delta_{kk'}-\hat{\pi}_{ik'}\right)$$

$$\left[I_{base,E}\left(\hat{B}_{base},0\right)\right]_{js,kt}=-\sum_{i=1}^{m}n_ix_{is}x_{it}\hat{\pi}_{ij}\left(\delta_{jk}-\hat{\pi}_{ik}\right)$$

where $\hat{\pi}_{ik}$, $\hat{\pi}_{ik'}$ are computed under the base model.

Wald's Test

In backward stepwise selection, we are interested in removing an effect *F* from an already fitted model. For a given base model with parameter vector B_{base}, we want to use Wald's statistic to test if effect *F* should be removed from the base model. If the parameter vector for the effect *F* is B_F, then the hypothesis can be formulated as

$H_0: B_F=0$ vs. $H_1: B_F\neq 0$

In order to write down the expression of the Wald's statistic, we will partition our parameter vector (and its estimate) into two parts as follows:

$B_{base} = \begin{pmatrix} B_{base\backslash F} \\ B_F \end{pmatrix}$ and $\hat{B}_{base} = \begin{pmatrix} \hat{B}_{base\backslash F} \\ \hat{B}_F \end{pmatrix}$

The first partition contains parameters that we intended to keep in the model and the 2nd partition contains the parameters of the effect *F*, which may be removed from the model. The information matrix and inverse information will be partitioned accordingly,

$$I\left(B_{base}\right) = \begin{pmatrix} I_{base\backslash F,base\backslash F}\left(B_{base\backslash F}, B_{base\backslash F}\right) & I_{base\backslash F,F}\left(B_{base\backslash F}, B_F\right) \\ I_{F,base\backslash F}\left(B_{base\backslash F}, B_F\right) & I_{F,F}\left(B_{base\backslash F}, B_F\right) \end{pmatrix}$$

and

$$J\left(B_{base}\right) = \begin{pmatrix} J_{base\backslash F,base\backslash F}\left(B_{base\backslash F}, B_{base\backslash F}\right) & J_{base\backslash F,F}\left(B_{base\backslash F}, B_F\right) \\ J_{F,base\backslash F}\left(B_{base\backslash F}, B_F\right) & J_{F,F}\left(B_{base\backslash F}, B_F\right) \end{pmatrix}$$

Using the above notations, the Wald's statistic for effect *F* can be expressed as

$$w = \hat{B}_F \left[J_{F,F}\left(B_{base\backslash F}, B_F\right)\right]^{-} \hat{B}_F$$

Under the null hypothesis, *w* has a chi-square distribution with degrees of freedom equal to the rank of $J_{F,F}\left(B_{base\backslash F}, B_F\right)$. If the rank of $J_{F,F}\left(B_{base\backslash F}, B_F\right)$ is zero, then Wald's statistic will be set to 0 and the *p*-value will be 1. Otherwise, if the rank of $J_{F,F}\left(B_{base\backslash F}, B_F\right)$ is $r_F : r_F > 0$, then the *p*-value of the test is equal to $1 - F\left(w; r_F\right)$, where $F\left(w; r_F\right)$ is the cumulative distribution function of a chi-square distribution with r_F degrees of freedom.

Statistics

The following output statistics are available.

Model Information

The model information (-2 log-likelihood) is available for the initial and final model.

Initial Model, Intercept-Only

If intercepts are included in the model, the predicted probability for the initial model (that is, the model with intercepts only) is

$$\tilde{\pi}_{ij} = \frac{\sum_{i=1}^{m} n_{ij}}{N}$$

and the value of the -2 log-likelihood of the initial model is

$$-2l\left(\tilde{\pi}\right) = -2\sum_{i=1}^{m}\sum_{j=1}^{J} n_{ij}\log\left(\tilde{\pi}_{ij}\right)$$

Initial Model, Empty

If intercepts are not included in the model, the predicted probability for the initial model is

$$\tilde{\pi}_{ij} = \frac{1}{J}$$

and the value of the -2 log-likelihood of the initial model is

$$-2l\left(\tilde{\pi}\right) = -2N\log\left(\frac{1}{J}\right)$$

Final Model

The value of -2 log-likelihood of the final model is

$$-2l\left(\tilde{\pi}\right) = -2\sum_{i=1}^{m}\sum_{j=1}^{J} n_{ij}\log\left(\hat{\pi}_{ij}\right)$$

Model Chi-Square

The Model Chi-square is given by

$$-2l\left(\tilde{\pi}\right) - \left\{-2l\left(\hat{\pi}\right)\right\}$$

Model with Intercepts versus Intercept-only Model

If the final model includes intercepts, then the initial model is an intercept-only model. Under the null hypothesis that $H_0 : \beta^{\text{intercepts}} = \mathbf{0}$, the Model Chi-square is asymptotically chi-squared distributed with $p^{\text{nr}} - (J - 1)$ degrees of freedom.

Model without Intercepts versus Empty Model

If the model does not include intercepts, then the initial model is an empty model. Under the null hypothesis that $H_0 : \beta = \mathbf{0}$, the Model Chi-square is asymptotically chi-squared distributed with p^{nr} degrees of freedoms.

Pseudo R-Square

The R^2 statistic cannot be exactly computed for multinomial logistic regression models, so these approximations are computed instead.

Cox and Snell’s R-Square

$$R^2_{\text{CS}} = 1 - \left(\frac{L(\tilde{\pi})}{L(\hat{\pi})}\right)^{\frac{2}{n}}$$

Nagelkerke’s R-Square

$$R^2_{\text{N}} = \frac{R^2_{\text{CS}}}{1-L(\hat{\pi})^{2/n}}$$

McFadden’s R-Square

$$R^2_{\text{M}} = 1 - \left(\frac{l(\hat{\pi})}{l(\bar{\pi})}\right)$$

Measures of Monotone Association

When the response variable has exactly two levels; that is, $k = 2$, measures of monotone association are available.

Without loss of generality, let the predicted probability for the category which is not the base category be π_{i1}. Also, let $s_i = [500 \times \pi_{i1}]/500$ where $[x]$ is the integer part of the value x.

Take a pair of observations indexed by i_1 and i_2 with different observed responses; the smaller index corresponds to a lower predictor value. This pair is a concordant pair if $s_{i_1} < s_{i_2}$ for $i_1<i_2$. This pair is a discordant pair if $s_{i_1} > s_{i_2}$ for $i_1<i_2$. If the pair is neither concordant nor discordant, it is a tied pair. Suppose there are a total of t pairs with different responses, m_c pairs are concordant, m_d pairs are discordant, and $t - m_c - m_d$ pairs are tied. The following measures of monotone association are computed.

Somers’ D

$$D = (m_c - m_d)/t$$

Goodman-Kruskal’s Gamma

$$Gamma = (m_c - m_d)/(m_c + m_d)$$

Kendall’s Tau-a

$$Tau - a = 2(m_c - m_d)/(n(n-1))$$

where n is the total sum of all frequencies $n = \sum_{i=1}^{m}\sum_{j=1}^{k} n_{ij}$.

Concordance Index C

$$C = (m_c + (t - m_c - m_d)/2)/t$$

Goodness of Fit Measures

The following tests of the null hypothesis that the model adequately fits the data are available.

Pearson Goodness of Fit Measure

$$X^2 = \sum_{i=1}^{m} \sum_{j=1}^{J} \frac{\left(n_{ij} - n_i \hat{\pi}_{ij}\right)^2}{n_i \hat{\pi}_{ij}}$$

Under the null hypothesis, the Pearson goodness-of-fit statistic is asymptotically chi-squared distributed with $m(J-1) - p^{\mathrm{nr}}$ degrees of freedom.

Deviance Goodness of Fit Measure

$$D = 2\sum_{i=1}^{m} \sum_{j=1}^{J} n_{ij} \log\left(\frac{n_{ij}}{n_i \hat{\pi}_{ij}}\right)$$

Under the null hypothesis, the Deviance goodness-of-fit statistic is asymptotically chi-squared distributed with $m(J-1) - p^{\mathrm{nr}}$ degrees of freedom.

Overdispersion Adjustments

Let $\hat{\kappa} > 0$ be an estimate of the overdispersion parameter. Possible estimates of this parameter are:

- A positive value specified in the command. If no value is specified, 1 is assumed.
- The ratio of Pearson goodness-of-fit measure to its degrees of freedom:
  $\hat{\kappa} = \frac{X^2}{m(k-1)-p^{nr}}$
- The ratio of Deviance goodness of fit measure to its degrees of freedoms:
  $\hat{\kappa} = \frac{D}{m(k-1)-p^{nr}}$

Covariance and Correlation Matrices

The estimate of the covariance matrix of the parameters is the inverse of the negative of the second derivative of the log-likelihood evaluated at $\mathbf{B} = \mathbf{B}^{(\nu)}$, multiplied by the estimate of the overdispersion parameter.

$$\mathrm{Cov}\left(\hat{\mathbf{B}}\right) = \hat{\kappa}\left[\sum_{i=1}^{m} \mathbf{X}_i^* \hat{\Delta}_i \mathbf{X}_i^{*\prime}\right]^{-1}$$

Let $\hat{\sigma}$ be the (J-1)p 1 vector of the square roots of the diagonal elements in Cov$\left(\hat{\mathbf{B}}\right)$. The estimate of the correlation matrix of $\hat{\mathbf{B}}$ is

$$Cor\left(\hat{\mathbf{B}}\right) = Diag\left(\hat{\sigma}^{-1}\right) Cov\left(\hat{\mathbf{B}}\right) Diag\left(\hat{\sigma}^{-1}\right)$$

Parameter Statistics

An estimate of the standard deviation of $\hat{B}_{js}$ is $\hat{\sigma}_{js}$. The Wald statistic for $\hat{B}_{js}$ is

$$\text{Wald}_{js} = \frac{\hat{B}_{js}}{\hat{\sigma}_{js}}$$

Under the null hypothesis that $H_0 : B_{js} = 0$, Wald^2_{js} is asymptotically chi-square distributed with 1 degree of freedom.

Based on the asymptotic normality of the parameter estimate, a 100(1−α)% Wald confidence interval for $\hat{B}_{js}$ is

$$\hat{B}_{js} \pm z_{1-\alpha/2}\hat{\sigma}_{js}$$

where $z_{1-\alpha/2}$ is the upper (1−α/2)100th percentile of the standard normal distribution.

Predicted Cell Counts

At each subpopulation *i*, the predicted count for response category *Y*=*j* is

$$\hat{n}_{ij} = n_i\hat{\pi}_{ij}$$

The (raw) residual is $n_{ij} - \hat{n}_{ij}$ and the standardized residual is $(n_{ij} - \hat{n}_{ij}) / \sqrt{n_i\hat{\pi}_{ij}(1 - \hat{\pi}_{ij})}$.

Likelihood Based Partial Effects

A likelihood ratio test is performed for any effect (except intercept) in the model. The procedure to perform a likelihood ratio test for any effect *e* is as follows:

1. Form a submodel that has all the effects in the working model but the one (*e*) of interest.
2. Fit the submodel and calculate the value of its –2 log-likelihood, denote it by $-2l\left(\hat{\pi}_{(e)}\right)$. Moreover, let the number of non-redundant parameters in this submodel be $p^{nr}_{(e)}$.
3. Calculate the difference between the –2 log-likelihood of the submodel and that of the working model, $\left\{-2l\left(\hat{\pi}_{(e)}\right)\right\} - \{-2l(\hat{\pi})\}$.

Under the null hypothesis that the effect *e* of interest is zero, $\left\{-2l\left(\hat{\pi}_{(e)}\right)\right\} - \{-2l(\hat{\pi})\}$ is asymptotically chi-square distributed with $p^{nr} - p^{nr}_{(e)}$ degrees of freedom.

Linear Hypothesis Testing

For each $q \times p$ matrix of linear combinations **L**, *J* Wald’s tests are performed. Each of the first *J* – 1 Wald’s tests corresponds to a Wald’s test on each of the *J* – 1 logits. The last Wald’s test corresponds to a joint Wald’s test for all the *J* – 1 logits. In the following, it is assumed that $q = \text{Rank}(\mathbf{L}) \leq p$.

The Wald’s test corresponding to the *j*th logit is

$$Wald(\mathbf{L},\ j) = \left(\mathbf{L}\hat{\beta}_j\right)'\left\{\mathbf{L}Cov\left(\hat{\beta}_j\right)\mathbf{L}'\right\}^{-1}\left(\mathbf{L}\hat{\beta}_j\right)$$

Under the null hypothesis, $H_0 : \mathbf{L}\beta_j = \mathbf{0}$, Wald($\mathbf{L}, j$) is asymptotically chi-square distributed with *q* degrees of freedom.

Let **L*** be a $(J-1)q\times(J-1)p$ matrix,

$$\mathbf{L}^* = \begin{pmatrix} \mathbf{L} & 0 & \cdots & 0 \\ 0 & \mathbf{L} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{L} \end{pmatrix}$$

The Wald's joint test for all logits is

$$\text{Wald}(\mathbf{L},) = \left(\mathbf{L}^*\hat{\mathbf{B}}\right)'\left\{\mathbf{L}^*\text{Cov}\left(\hat{\mathbf{B}}\right)\mathbf{L}^{*\prime}\right\}^{-1}\left(\mathbf{L}^*\hat{\mathbf{B}}\right)$$

Under the null hypothesis, $H_0 : \mathbf{L}^*\mathbf{B} = \mathbf{0}$,Wald($\mathbf{L},$) is asymptotically chi-square distributed with $(J-1)q$ degrees of freedom.

Classification Table

Suppose that *c*(*j*, *j'*) is the (*j*, *j'*)-th element of the classification table, *j*, *j'* = 1, …, *J*. *c*(*j*, *j'*) is the sum of the frequencies for the observations whose actual response category is *j* (as row) and predicted response category is *j'* (as column) respectively.

The predicted response category for subpopulation *i* is

$$j^* : \hat{\pi}_{ij}* = \max_j\left(\hat{\pi}_{ij}\right)$$

Should there be a tie, choose the category with the smallest category number.

For *j*, *j'* = 1, …, *J*, *c*(*j*, *j'*) is given as

$$c\left(j, j'\right) = \sum_{i=1}^{m} n_{ij}\delta_{j_i^* j'}$$

The percentage of total correct predictions of the model is

$$p\left(\right) = \left(\frac{\Sigma_{j=1}^{n} c\left(j, j\right)}{n}\right)100\%$$

The percentage of correct predictions of the model for response category *j* is

$$p\left(\right) = \left(\frac{c(j,j)}{\Sigma_{i=1}^{m} n_{ij}}\right)100\%$$

Checking for Separation

The algorithm checks for separation in the data starting with iteration ν^{chksep} (20 by default). To check for separation:

1. For each subpopulation *i*, find j^* : $\hat{\pi}_{ij^*} = \max_j (\hat{\pi}_{ij})$.
2. If $n_{ij^*} = n_i$, then there is a perfect prediction for subpopulation *i*.
3. If all subpopulations have perfect prediction, then there is complete separation. If some patterns have perfect prediction and the Hessian of $\hat{\mathbf{B}}$ is singular, then there is quasi-complete separation.

References

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Cohen, A., and M. Rom. 1994. A Method for Hypothesis Tests in Polychotomous Logistic Regression. *Computational Statistics and Data Analysis*, 17, 277–288.

Cox, D. R., and E. J. Snell. 1989. *The Analysis of Binary Data*, 2nd ed. London: Chapman and Hall.

Cramer, J. S., and G. Ridder. 1988. The Logit Model in Economics. *Statistica Neerlandica*, 42, 291–314.

Haberman, S. 1974. *The Analysis of Frequency Data*. Chicago: University of Chicago Press.

Hauck, W. W., and A. Donner. 1977. Wald’s Test as Applied to Hypotheses in Logit Analysis. *Journal of the American Statistical Association*, 72, 851–853.

Hosmer, D. W., and S. Lemeshow. 2000. *Applied Logistic Regression*, 2nd ed. New York: John Wiley and Sons.

Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. New York: Advanced Quantitative Techniques in the Social Sciences Series.

Luce, R. D. 1959. *Individual Choice Behavior*. New York: John Wiley.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In: *Frontiers in Economics*, P. Zarembka, ed. New York: AcademicPress.

Nagelkerke, N. J. D. 1991. A note on the general definition of the coefficient of determination. *Biometrika*, 78:3, 691–692.

Searle, R. S. 1987. *Linear Models for Unbalanced Data*. New York: Wiley.

Zhang, J., and S. D. Hoffman. 1993. Discrete-Choice Logit Models. Testing the IIA Property. *Sociological Methods and Research*, 22:2, 193–213.

NONPAR CORR Algorithms

If a WEIGHT variable is specified, it is used to replicate a case as many times as indicated by the weight value rounded to the nearest integer. If the workspace requirements are exceeded and sampling has been selected, a random sample of cases is chosen for analysis using the algorithm described in SAMPLE. For the RUNS test, if sampling is specified, it is ignored. The tests are described in (Siegel, 1956).

Spearman Correlation Coefficient

For each of the variables *X* and *Y* separately, the observations are sorted into ascending order and replaced by their ranks. In situations where *t* observations are tied, the average rank is assigned. Each time $t > 1$, the quantity $t^3 - t$ is calculated and summed separately for each variable. These sums will be designated ST_x and ST_y.

For each of the *N* observations, the difference between the rank of *X* and rank of *Y* is computed as:

$$d_i = R(X_i) - R(Y_i)$$

Spearman's rho (ρ) is calculated as (Siegel, 1956):

$$\rho_s = \frac{T_x + T_y - \sum_{i=1}^{N} d_i^2}{2\sqrt{T_x T_y}}$$

where

$$T_x = \frac{N^3 - N - ST_x}{12} \quad \text{and} \quad T_y = \frac{N^3 - N - ST_y}{12}$$

If T_x or T_y is 0, the statistic is not computed.

The significance level is calculated assuming that, under the null hypothesis,

$$t = \rho_s \sqrt{\frac{N-2}{1-r_s^2}}$$

is distributed as a *t* with $N - 2$ degrees of freedom. A one- or two-tailed significance level is printed depending on the user-selected option.

Kendall's Tau

For each of the variables *X* and *Y* separately, the observations are sorted into ascending order and replaced by their ranks. In situations where *t* observations are tied, the average rank is assigned.

Each time $t > 1$, the following quantities are computed and summed over all groups of ties for each variable separately.

$$\tau_v = \Sigma t^2 - t$$

$$\tau'_v = \Sigma\left(t^2 - t\right)(t-2)$$

$$\tau''_v = \Sigma\left(t^2 - t\right)(2t+5),\ \text{and } v = x \text{ or } y$$

Each of the *N* cases is compared to the others to determine with how many cases its ranking of *X* and *Y* is concordant or discordant. The following procedure is used. For each distinct pair of cases $(i, j), i < j$ the quantity

$$d_{ij} = [R(X_j) - R(X_i)][R(Y_j) - R(Y_i)]$$

is computed. If the sign of this product is positive, the pair of observations (i, j) is concordant, since both members of observation *i* are either less than or greater than their respective measurement in observation *j*. If the sign is negative, the pair is discordant.

The number of concordant pairs minus the number of discordant pairs is

$$S = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \text{sign}\,(d_{ij})$$

where sign(d_{ij}) is defined as +1 or –1 depending on the sign of d_{ij}. Pairs in which $d_{ij} = 0$ are ignored in the computation of *S*.
Kendall's tau (τ) is computed as

$$\tau = \frac{S}{\sqrt{\frac{N^2 - N - \tau_x}{2}}\sqrt{\frac{N^2 - N - \tau_y}{2}}}$$

If the denominator is 0, the statistic is not computed.
The variance of *S* is estimated by (Kendall, 1955):

$$d = \frac{1}{18}\left\{K(2N+5) - \tau''_x - \tau''_y\right\} + \frac{\tau'_x \tau'_y}{9K(N-2)} + \frac{\tau_x \tau_y}{2K}$$

where

$$K = N^2 - N$$

The significance level is obtained using

$$Z = \frac{S}{\sqrt{d}}$$

which, under the null hypothesis, is approximately normally distributed. The significance level is either one- or two-sided, depending on the user specification.

References

Kendall, M. G. 1955. *Rank correlation methods*. London: Charles Griffin.

Siegel, S. 1956. *Nonparametric statistics for the behavioral sciences*. New York: McGraw-Hill.

Nonparametric Tests Algorithms

Nonparametric tests make minimal assumptions about the underlying distribution of the data. The available nonparametric tests can be grouped into three broad categories based on how the data are organized: one-sample tests, related-samples tests, and independent-samples tests. A one-sample test analyzes one field. A test for related samples compares two or more fields for the same set of records. An independent-samples test analyzes one field that is grouped by categories of another field.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$\{x_i, f_i\}_{i=1}^{n}$	Data for one sample tests: x_i is the *i*th observed value, and f_i is the frequency/replication weight for x_i.
$\{x_{i1}, \cdots, x_{iK}, f_i\}_{i=1}^{n}$	Data for *K* related samples tests: each *x*-column represents one sample, f_i is the frequency/replication weight for row/record *i*.
$\{x_i, g_i, f_i\}_{i=1}^{N}$	Data for *K* independent samples: g_i indicates the sample that observation x_i belongs to, f_i is the frequency/replication weight.
$G_j = \{i : g_i = j\}$	All record indices in the *j*th sample.
n_j	The number of records in the *j*th sample, ignoring the frequency weight.
$n_{j,f} = \sum_{i \in G_j} f_i$	The number of records in the *j*th sample, incorporating frequency weight.
$\text{rank}\left(g\left(x_i\right); D\right)$	The rank of $g\left(x_i\right)$ when all $\{g\left(x\right) : x \in D\}$ are jointly ranked. If there are ties, the average rank is used.
$rank\left(g\left(x_i\right); D.\mathbf{f}\right)$	Like $\text{rank}\left(g\left(x_i\right); D\right)$ but frequency weights are incorporated when calculating the ranks.
$F_k\left(x\right)$	The cumulative distribution function of population *k*.
$\Phi\left(z\right)$	The cumulative distribution function for the standard normal distribution such that.
α	The critical value for determining whether to reject the null hypothesis.

One-Sample Tests

The following one-sample tests are available.

Binomial Test

For a categorical field with 2 values (or recoded categorical field with more than 2 values or recoded continuous field), this tests:

H_0: The probability of success is equal to the hypothesized success probability p_0.

H_A (if p_0=0.5): The probability of success is not equal to the hypothesized success probability (use the two-tailed *p*-value)

H_A (if p_0>0.5): The probability of success is greater than the hypothesized success probability (use the one-tailed *p*-value)

H_A (if p_0<0.5): The probability of success is less than the hypothesized success probability (use the one-tailed *p*-value)

Let $n_{1,f}$ and $n_{2,f}$ be the numbers of records in the success and failure categories, incorporating the frequency weight.

If $n_{1,f} + n_{2,f} \leq 25$, then one-tailed exact probability is

$$p_1 = \min\left\{\Pr\left(T \leq n_{1,f}\,|H_0\right),\ \Pr\left(T \geq n_{1,f}\middle|\, H_0\right)\right\}$$

where

$$\Pr\left(T \leq n_{1,f}|H_0\right) = \sum_{i=0}^{n_{1,f}} \binom{n_{1,f}+n_{2,f}}{i} p_0^i (1-p_0)^{(n_{1,f}+n_{2,f})-i}$$

and

$$\Pr\left(T \geq n_{1,f}|H_0\right) = \sum_{i=n_{1,f}}^{n_{1,f}+n_{2,f}} \binom{n_{1,f}+n_{2,f}}{i} p_0^i (1-p_0)^{(n_{1,f}+n_{2,f})-i}$$

The two-tailed exact probability is $p = 2p_1$.

If $n_{1,f} + n_{2,f} > 25$, a normal approximation is used. Letting

$$Z_1 = \frac{n_{1,f} + 0.5 - \left(n_{1,f} + n_{2,f}\right) p_0}{\sqrt{\left(n_{1,f} + n_{2,f}\right) p_0 \left(1 - p_0\right)}}$$

and

$$Z_2 = \frac{n_{1,f} - 0.5 - \left(n_{1,f} + n_{2,f}\right) p_0}{\sqrt{\left(n_{1,f} + n_{2,f}\right) p_0 \left(1 - p_0\right)}}$$

the one-tailed approximate probability is

$$p_1 = \min\left\{\Phi\left(Z_1\right), 1 - \Phi\left(Z_2\right)\right\}$$

and the two-tailed approximate probability is $p = 2p_1$.

$p < \alpha$ rejects the two-tailed test if p_0 = 0.5; otherwise $p_1 < \alpha$ rejects the one-tailed test.

Confidence Interval for Binomial Success Rate

Without loss of generality we assume that $x_i = 0$ or 1 with 1 representing success. We want to estimate the success probability $p = \Pr(x = 1)$ and its confidence interval. Let $T = \sum_{i=1}^{n} f_i I(x_i = 1) = \sum_{i=1}^{n} f_i x_i$, $n_f = \sum_{i=1}^{n} f_i$. The estimate of the success probability is $\hat{p} = \frac{T}{n_f}$. For confidence interval (p_L, p_U), we provide the following three ways of calculating it. For all three methods, $p_L = 0$ if $\hat{p} = 0$ and $p_U = 1$ if $\hat{p} = 1$.

Clopper-Pearson confidence interval

The Clopper-Pearson confidence interval is an exact confidence interval based on inverting the exact equal-tailed binomial test $H_0 : p = p_0$. The lower and upper confidence limits are found by solving

$$\sum_{i=T}^{n_f} \binom{n_f}{i} p_L (1 - p_L)^{n-i} = \alpha/2$$

$$\sum_{i=0}^{T} \binom{n_f}{i} p_U (1 - p_U)^{n-i} = \alpha/2$$

The solutions to these two equations are (Leemis and Trivedi, 1996)

$$p_L = \left(1 + \frac{n_f - T + 1}{T F_{\alpha/2}\left(2T, 2\left(n_f - T + 1\right)\right)}\right)^{-1}$$

$$p_U = \left(1 + \frac{n_f - T}{(T+1) F_{1-\alpha/2}\left(2(T+1), 2\left(n_f - T\right)\right)}\right)^{-1}$$

where $F_{\alpha/2}(v_1, v_2)$ is the $\alpha/2$ percentile of the *F*-distribution $F(v_1, v_2)$.

Note: The Clopper-Pearson confidence interval is conservative (coverage probability is at least $1 - \alpha$) because of the discreteness of the binomial distribution. The coverage probability can be much larger than $1 - \alpha$ unless sample size is very big.

Jeffreys confidence interval

Jeffreys confidence interval is a Bayesian interval based on the posterior probability of *p* using the Jeffreys prior $Beta(\frac{1}{2}, \frac{1}{2})$. The resulting posterior for *p* is $Beta(T + \frac{1}{2}, n_f - T + \frac{1}{2})$. Then the lower and upper confidence limits of *p* are the $\frac{\alpha}{2}$ and $1 - \frac{\alpha}{2}$ percentiles of this beta distribution

$$p_L = B_{\frac{\alpha}{2}}\left(T + \tfrac{1}{2}, n_f - T + \tfrac{1}{2}\right)$$
$$p_U = B_{1-\frac{\alpha}{2}}\left(T + \tfrac{1}{2}, n_f - T + \tfrac{1}{2}\right)$$

Likelihood ratio confidence interval

The likelihood ratio confidence interval is constructed by inverting the acceptance region of the likelihood ratio test which accepts the null hypothesis $H_0 : p = p_0$ if

$$-2\log\left(\frac{Lik(p_0)}{\underset{p}{sup}\left(Lik(p)\right)}\right) \le \chi^2_{1-\alpha}(1)$$

or

$$-2\left(l(p_0) - \underset{p}{sup}\left(l(p)\right)\right) \le \chi^2_{1-\alpha}(1)$$

where $\chi^2_{1-\alpha}(1)$ is the $1-\alpha$ percentile of the chi-square distribution with 1 degree of freedom, $Lik(p)$ and $l(p)$ are likelihood and log likelihood functions. The log likelihood function is

$$l(p) = \log\left(Lik(p)\right) = T\log p + \left(n_f - T\right)\log(1-p)$$

with the convention $0\log(0) = 0$.

Inverting the likelihood ratio test is to find a range (p_L, p_U) for p_0 such that within this range $-2\left(l(p_0) - l(\hat{p})\right) \le \chi^2_{1-\alpha}(1)$ or equivalently $l(p_0) - l(\hat{p}) + \frac{\chi^2_{1-\alpha}(1)}{2} \ge 0$ is satisfied. The function $h(p_0) = l(p_0) - l(\hat{p}) + \frac{\chi^2_{1-\alpha}(1)}{2}$ is well behaved with $h(0) = -\infty, h(1) = -\infty$, maximum at $h(\hat{p}) = \frac{\chi^2_{1-\alpha}(1)}{2}$, increasing for $p_0 < \hat{p}$ and decreasing for $p_0 > \hat{p}$ because its first derivative is

$$h'(p_0) = \frac{T - n_f p_0}{p_0(1-p_0)} = \frac{n_f(\hat{p} - p_0)}{p_0(1-p_0)} = \begin{cases} > 0 & p_0 < \hat{p} \\ = 0 & p_0 = \hat{p} \\ < 0 & p_0 > \hat{p} \end{cases}$$

The two solutions for $h(p_0) = 0$, one on each side of $\hat{p}$, correspond to $p_L\ (< \hat{p})$ and $p_U\ (> \hat{p})$. To obtain the solutions, the Newton-Raphson iterative method is used to solve the equation $h(p_0) = 0$. Letting $p^{(v)}$ be the solution at iteration step v, the solution $p^{(v+1)}$ at iteration step $v + 1$ is updated as

$$p^{(v+1)} = p^{(v)} - \xi\frac{h\left(p^{(v)}\right)}{h'\left(p^{(v)}\right)} = p^{(v)} - \xi\frac{p^{(v)}\left(1 - p^{(v)}\right)}{T - n_f p^{(v)}} h\left(p^{(v)}\right)$$

The stepping scalar $\xi > 0$ is used to make sure $\left|h\left(p^{(v+1)}\right)\right| < \left|h\left(p^{(v)}\right)\right|$ and $0 < p^{(v+1)} < 1$. We use the step-halving method if either $\left|h\left(p^{(v+1)}\right)\right| < \left|h\left(p^{(v)}\right)\right|$ or $0 < p^{(v+1)} < 1$ is not satisfied. Let s be the maximum number of steps in step-halving, the set of values of ξ is $\{1/2^r$: $r = 0, \ldots, s-1\}$.

Iterations start with an initial value $p^{(0)}$ and continue until one of the stopping criteria is reached. Only p_U needs to be calculated when $\hat{p} = 0$ because $p_L = 0$; and only p_L needs to be calculated when $\hat{p} = 1$ because $p_U = 1$. In fact, a closed form solution exists in these special situations, $p_U = 1 - \exp\left(-\frac{\chi^2_{1-\alpha}(1)}{2n_f}\right)$ for $\hat{p} = 0$, and $p_L = \exp\left(-\frac{\chi^2_{1-\alpha}(1)}{2n_f}\right)$ for $\hat{p} = 1$.

Initial values. Any initial value $p^{(0)} \in (0, \hat{p})$ will lead to a solution for p_L, and any initial value $p^{(0)} \in (\hat{p}, 1)$ will lead to p_U. Let $p_{L,J}, p_{U,J}$ be the Jeffreys lower and upper confidence limits. We will take the following as the initial values for the lower and upper confidence limits

$$p_L^{(0)} = \begin{cases} p_{L,J} & \text{if } p_{L,J} < \hat{p} \\ \hat{p}/2 & \text{otherwise} \end{cases}$$

$$p_U^{(0)} = \begin{cases} p_{U,J} & \text{if } p_{U,J} > \hat{p} \\ (\hat{p}+1)/2 & \text{otherwise} \end{cases}$$

Stopping criteria. Let $\epsilon = 10^{-8}$. The following stopping criteria are checked in the following order.

1. Absolute argument convergence criterion: $\left|p^{(v+1)} - p^{(v)}\right| < \epsilon$
2. Relative argument convergence criterion: $\frac{\left|p^{(v+1)} - p^{(v)}\right|}{\left|p^{(v)}\right| + 10^{-12}} < \epsilon$
3. Function convergence criterion: $\left|h\left(p^{(v+1)}\right)\right| < \epsilon$
4. The maximum number of iterations, default at 50, is reached, or the maximum number of steps in step-halving is reached, default at 20.

Chi-Square Test

For a categorical field, this tests:

H_0: The probability of each category *i* equals the hypothesized probability P_i.

H_A: At least one category's probability does not equal its hypothesized probability.

The test statistic is

$$\chi^2 = \sum_{i=1}^{k} \frac{\left(O_{i,f} - EXP_i\right)^2}{EXP_i}$$

where $O_{i,f}$ and $EXP_i = P_i n_f$ are the observed and expected frequencies of category *i*.

The one-sided *p*-value is

$$p = \Pr\left(\chi^2_{k-1} \geq \chi^2\right) = 1 - \Pr\left(\chi^2_{k-1} \leq \chi^2\right)$$

where χ^2_{k-1} follows a chi-square distribution with $k-1$ degrees of freedom.

$p < \alpha$ rejects the null hypothesis.

Kolmogorov-Smirnov Test

For a continuous field, this tests:

H_0: $F(x) = F_0(x)$ for all x, where $F(x)$ is the distribution of the sample and $F_0(x)$ is the hypothesized distribution which can be the uniform, the Poisson, the normal or the exponential distribution.

H_A: $F(x) \neq F_0(x)$ for some x.

Empirical cumulative distribution function

The observations are sorted into ascending order: $x_{(1)} < x_{(2)} < \cdots < x_{(m)}$, where m is the number of distinct values of X. Then the empirical cdf is

$$\hat{F}(x) = \begin{cases} 0, -\infty < x < x_{(1)} \\ \dfrac{\sum_{i=1}^{m} f_i I\{x < x_{(k+1)}\}}{n} , x_{(k)} \le x < x_{(k+1)}, k = 1, \cdots, m-1 \\ \quad \displaystyle\sum_{i=1} f_i \\ 1, x_{(m)} \le x < \infty \end{cases}$$

Theoretical cumulative distribution function

Uniform

$$F_0(x_i) = \frac{x_i - \min}{\max - \min}$$

where min and max are user-specified (default sample minimum and maximum).

Poisson

$$F_0(x_i) = \sum_{l=0}^{x_i} \frac{e^{-\lambda}\lambda^l}{l!}$$

where λ is user-specified (default sample mean). If $\lambda \ge 100,000$, the normal approximation is used with $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$.

Normal

$$F_0(x_i) = \Phi\left(\frac{x_i - \mu}{\sigma}\right)$$

where μ and σ are user-specified (default sample mean and standard deviation).

Exponential

$$F_0(x_i) = 1 - e^{-\beta x_i}$$

where β is user-specified (default inverse sample mean).

Test statistic and p-value

The test statistic is calculated based on differences between the empirical cumulative distribution and the theoretical cumulative distribution. For the uniform, normal and exponential distributions, two differences are computed:

$$D_i = \hat{F}\left(x_{(i-1)}\right) - F_0\left(x_{(i)}\right)$$

$$\tilde{D}_i = \hat{F}\left(x_{(i)}\right) - F_0\left(x_{(i)}\right)$$

for *i*=1,...,*m*. For the Poisson:

$$D_i = \begin{cases} \hat{F}\left(x_{(i)} - 1\right) - F_0\left(x_{(i)} - 1\right), x_{(i)} > 0, \\ 0 x_{(i)} = 0 \end{cases}$$

$$\tilde{D}_i = \hat{F}\left(x_{(i)}\right) - F_0\left(x_{(i)}\right)$$

for *i*=1,...,*m*.

The test statistic is

$$Z = \sqrt{\sum_{j=1}^{n} f_j} \max_i\left(|D_i|, \left|\tilde{D}_i\right|\right)$$

The two-tailed probability level is estimated using the first three terms of the Smirnov (1948) formula.

$$p = \begin{cases} 1 & & 0 \le Z < 0.27 \\ 1 - \frac{\sqrt{2\pi}}{Z}\left(Q + Q^9 + Q^{25}\right), & Q = e^{-\pi^2/8Z^2} & 0.27 \le Z < 1 \\ 2\left(Q - Q^4 + Q^9 - Q^{16}\right), & Q = e^{-2Z^2} & 1 \le Z < 3.1 \\ 0 & & Z \ge 3.1 \end{cases}$$

$p < \alpha$ rejects the null hypothesis.

Note: If the distribution is normal and parameters are estimated from the data, then the Lilliefors method is used to compute the test statistic and p value instead of the method described in this section. For more information, see the topic "Kolmogorov-Smirnov Statistic with Lilliefors' Significance".

Kolmogorov-Smirnov Statistic with Lilliefors' Significance

In the case that the distribution is normal and parameters are estimated from the data, the Lilliefors method (Lilliefors, 1967) is used to compute the test statistic and p value. The Lilliefors significance *p* is calculated based on the formulas and critical value tables from Lilliefors (Lilliefors et al., 1967) and Dallal and Wilkinson (Dallal and Wilkinson, 1986).

The test statistic is

$$D = \max_i \left(|D_i| \, , \, \left| \tilde{D}_i \right| \right)$$

Let $n_f = \sum_{j=1}^{n} f_j$. If $n_f < 5$, the p value is set to the system missing value.

If $n_f \geq 5$, then the Lilliefors significance p is calculated as follows:

Step 1. Compute the critical value $D_{0.1}$ for upper tail probability 0.1:

$$D_{0.1} = \frac{\left(-b - \sqrt{b^2 - 4ac}\right)}{2a}$$

where, if $n_f \leq 100$, then $a = -7.01256\,(n_f + 2.78019)$, $b = 2.99587\sqrt{n_f + 2.78019}$, and $c = 2.1804661 + \frac{0.974598}{\sqrt{n_f}} + \frac{1.67997}{n_f}$

and if $n_f > 100$, then $a = -7.90289126054 * n_f{}^{0.98}$, $b = 3.180370175721 * n_f{}^{0.49}$, and $c = 2.2947256$

Step 2. If $D = D_{0.1}$, then $p = 0.1$

If $D > D_{0.1}$, then $p = \exp\left\{aD^2 + bD + c - 2.3025851\right\}$

Otherwise go to step 3

Step 3. If there is an entry in Table 69-1 for sample size n_f, then go to step 4 to compute the p value. Otherwise, linear interpolation is used to calculate the critical values for a sample size of n_f.

For example, for $n_f = 22$, which is between sample sizes $s1 = 20$ and $s2 = 25$ with critical values $c1 = 0.159$ and $c2 = 0.143$ respectively, the critical value for upper tail probability 0.2 is computed as:

$$\frac{c2 - c1}{s2 - s1} * \left(n_f - s1\right) + c1 = \frac{0.143 - 0.159}{25 - 20} * (22 - 20) + 0.159 = 0.1526$$

The critical value for upper tail probability 0.15 (for $n_f = 22$) is computed in a similar manner.

Step 4. If $D_{0.15} \leq D < D_{0.1}$ or $D_{0.2} \leq D < D_{0.15}$, then linear interpolation is used to compute the p value, where $D_{0.2}$ and $D_{0.15}$ are the critical values for upper tail probability 0.2 and probability 0.15 (corresponding to sample size n_f) respectively.

For example, for $n_f = 20$, use step 1 to compute the critical value for upper tail probability 0.1 as $D_{0.1} = 0.1772025$. If $D = 0.170$, then $0.166 = D_{0.15} < D < D_{0.1}$. The p value can be computed as:

$$p = \frac{0.1 - 0.15}{D_{0.1} - D_{0.15}} * (D - D_{0.15}) + 0.15 = 0.1321469$$

If $D < D_{0.2}$, then p is reported as > 0.2.

Table 69-1
Upper tail probability and corresponding critical values

Sample size	**p value 0.2**	**p value 0.15**
5	0.289	0.303
6	0.269	0.281
7	0.252	0.264
8	0.239	0.250
9	0.227	0.238
10	0.217	0.228
11	0.208	0.218
12	0.200	0.210
13	0.193	0.202
14	0.187	0.196
15	0.181	0.190
16	0.176	0.184
17	0.171	0.179
18	0.167	0.175
19	0.163	0.170
20	0.159	0.166
25	0.143	0.150
30	0.131	0.138
40	0.115	0.120
100	0.074	0.077
400	0.037	0.039
900	0.025	0.026
Over 900	$\frac{0.736}{n_f}$	$\frac{0.768}{n_f}$

Runs Test

For a categorical field with 2 values (or a recoded categorical field with more than 2 values or a recoded continuous field), this tests:

H_0: The observed order of observations of a field is attributable to chance variation.

Number of runs

The number of times that the category changes; that is, where x_i belongs to one category and x_{i+1} belongs to the other, as well as the number of records in category 1 ($n_{1,f}$) and category 2 ($n_{2,f}$), are determined. The number of runs, *R*, is the number of sign changes plus one.

Test statistic and p-value

The sampling distribution of the number of runs is approximately normal with

$$\mu_R = \frac{2n_{1,f}n_{2,f}}{n_{1,f}+n_{2,f}} + 1$$

$$\sigma_R = \sqrt{\frac{2n_{1,f}n_{2,f}\left(2n_{1,f}n_{2,f} - n_{1,f} - n_{2,f}\right)}{\left(n_{1,f}+n_{2,f}\right)^2\left(n_{1,f}+n_{2,f}-1\right)}}$$

The test statistic is

$z = \frac{R-\mu_R}{\sigma_R}$, if $n_f \geq 50$, otherwise

$$z = \begin{cases} \left(R-\mu_R+0.5\right)/\sigma_R & \text{if } R-\mu_R \leq -0.5 \\ \left(R-\mu_R-0.5\right)/\sigma_R & \text{if } R-\mu_R \geq 0.5 \\ 0 & \text{if } |R-\mu_R| < 0.5 \end{cases}$$

The one sided *p*-value is $p_1 = \Pr\left(Z \geq |z|\right) = 1 - \Phi\left(|z|\right)$ and the two sided *p*-value is $p_2 = 2p_1$.

$p_2 < \alpha$ rejects the null hypothesis.

Wilcoxon Signed-Rank Test

For a continuous field, this tests:

H_0: median $(X) = \theta$ where θ is user-specified (default to sample median).

Let $d_i = x_i - \theta$,$D = \{d_i : |d_i| \neq 0\}$. The test statistic is the sum of positive ranks incorporating the frequency weights:

$$T = \sum_{i \in D} f_i \text{rank}\left(|d_i|; D, \mathbf{f}\right) I\left(sgn\left(d_i\right) > 0\right)$$

The standardized test statistic is

$$T^* = \frac{T-\mu_T}{\sigma_T}$$

where

$$\mu_T = \frac{1}{4} n_f \left(n_f + 1\right)$$

$$\sigma_T^2 = \frac{1}{24} n_f \left(n_f + 1\right)\left(2n_f + 1\right) - \frac{1}{48} \sum_{j=1}^{M} \left(t_{j,f}^3 - t_{j,f}\right)$$

$$n_f = \sum_{i \in D} f_i$$

where *M* is the total number of distinct rank values of $|d_i| > 0$ and $t_{j,f}$ is the number of records tied at the *j*th distinct value, incorporating the frequency weights.

The asymptotic one-sided and two-sided *p*-values are

$$p_1 = \Pr\left(Z \geq |T^*|\right) = 1 - \Phi\left(|T^*|\right)$$

$$p = 2p_1$$

$p_1 < \alpha$ rejects the null hypothesis in favor of median $(X) > \theta$ if $T^* > 0$ and in favor of median $(X) < \theta$ if $T^* < 0$.

Note: The one-sample Wilcoxon signed-rank test is equivalent to the matched-pairs Wilcoxon signed-rank test when the second sample replaced by a constant θ.

Independent Samples Tests

The following independent-samples tests are available.

Mann-Whitney Test

For two independent samples from a continuous field, this tests:

H_0: $F_1(x) = F_2(x)$; that is, the two samples are from populations with the same distribution function

H_A: $F_1(x) \geq F_2(x)$

$H_A^{'}$: $F_1(x) \leq F_2(x)$

The first group is defined by the first value of the grouping field in ascending order.

Calculation of Sums of Ranks

The combined data from both specified groups are sorted and ranks assigned to all records, with average rank being used in the case of ties. The sum of ranks for each group is

$$S_{1,f} = \sum_{i \in G_1} f_i \text{rank}(x_i; D_1 \cup D_2, \text{f}) I(x_i \in D_1)$$
$$S_{2,f} = \sum_{i \in G_2} f_i \text{ rank}(x_i; D_1 \cup D_2, \text{f}) I(x_i \in D_2)$$

The average rank for each group is

$$\overline{S}_i = S_{i,f}/n_{i,f}$$

where $n_{i,f} = \sum_{i \in G_i^*} f_i I(x_i \in D_i)$.

If there are tied records, the number of records tied at the *j*th distinct value incorporating the frequency weight, $t_{j,f}$ are counted.

Test statistic and p-value

The Wilcoxon rank sum W statistic is $W = S_{2,f}$. The Mann-Whiney *U* statistic for group 1 is

$$\begin{aligned} U &= \textstyle\sum_{i \in G_1} \sum_{j \in G_2} f_i f_j I(x_i < x_j) + \frac{1}{2} \sum_{i \in G_1} \sum_{j \in G_2} f_i f_j I(x_i = x_j) \\ &= n_{1,f} n_{2,f} + \frac{n_{1,f}(n_{1,f}+1)}{2} - S_{1,f} \\ &= S_{2,f} - \frac{n_{2,f}(n_{2,f}+1)}{2} \end{aligned}$$

If $n_{1,f} n_{2,f} \le 400$ and $n_{1,f} n_{2,f}/2 + \min(n_{1,f}, n_{2,f}) \le 220$, the exact significance level is based on an algorithm of Dineen and Blakesley (1973), which is given as follows:

Let $f_{ij}(u)$ be the sampling frequency of the Mann-Whitney statistic for a value of *U* and with sample size *i* and *j*. Then the frequency distribution of the Mann-Whiney *U* statistic can be derived by summing two lower order distributions:

$$f_{n_{1,f}, n_{2,f}}(u) = f_{n_{1,f}-1, n_{2,f}}\left(u - n_{2,f}\right) + f_{n_{1,f}, n_{2,f}-1}(u)$$

Each of the lower order distribution is symmetrical about a different value of *U* and the sum gives a result which is also symmetrical. The algorithm starts with known distribution for *i*=1 (or *j*=1) and then uses the above equation and symmetry properties to derive the full distribution for *i*=2 (or *j*=2). This procedure is repeated until the distribution for the required value for $i = n_{1,f}$ (or $j = n_{2,f}$).

After the complete distribution of *U* is obtained, the one sided and two sided *p*-values are

$$p_1 = \begin{cases} \sum_{u=0}^{\lfloor U \rfloor} f_{n_{1,f}, n_{2,f}}(u) & \text{if } U \le \frac{n_{1,f} n_{2,f}+1}{2} \\ 1 - \sum_{u=0}^{\lfloor U \rfloor} f_{n_{1,f}, n_{2,f}}(u) & \text{if } U > \frac{n_{1,f} n_{2,f}+1}{2} \end{cases}$$
$$p = 2p_1$$

where $\lfloor x \rfloor$ is the floor integer of *x*.

The test statistic corrected for ties is

$$T = \frac{(U - \mu_T)}{\sigma_T}$$

where

$$\mu_T = \frac{n_{1,f} n_{2,f}}{2}$$

$$\sigma_T^2 = \frac{n_{1,f} n_{2,f}}{n_{1 \text{ and } 2,f}\left(n_{1 \text{ and } 2,f} - 1\right)} \left(\frac{n_{1 \text{ and } 2,f}^3 - n_{1 \text{ and } 2,f}}{12} - \sum_{i=1}^{M} T_i \right)$$

$$n_{1 \text{ and } 2,f} = n_{1,f} + n_{2,f}$$

$$T_i = \frac{t_{i,f}^3 - t_{i,f}}{12}$$

and *M* is the total number of distinct rank values. The one sided and two sided *p*-values are respectively

$$p_1 = \Pr\left(Z \geq |T|\right) = 1 - \Phi\left(|T|\right)$$

$$p = 2p_1$$

$p_1 < \alpha$ will reject the null hypothesis and in favor of $H_A^{'}$ if *T*<0 in favor of H_A if *T*>0. $p < \alpha$ will reject the null hypothesis in favor of either $H_A^{'}$ or H_A.

Wald-Wolfowitz Test

For two independent samples from a continuous field, this tests:

H_0: $F_1(x) = F_2(x)$; that is, the two samples are from populations with the same distribution function

H_A: $F_1(x) \neq F_2(x)$ for some *x*

Calculation of Number of Runs

Then all observations from the two groups G_1 and G_2 are pooled and sorted into ascending order. The number of changes in the group corresponding to the ordered data is counted. The number of runs (*R*) is the number of group changes plus one. If there are ties involving observations from the two groups, both the minimum and maximum numbers of runs possible are calculated.

Suppose that *m* distinct values in groups G_1 and G_2 are sorted into ascending order:

$$x_{(1)} < x_{(2)} < \cdots < x_{(m)}$$

Let $s_{i,f}$ and $t_{i,f}$ be the numbers of records of $x_{(i)}$ in G_1 and G_2 respectively, incorporating the frequency weight

$$s_{i,f} = \sum_{j \in G_1} f_j I\left(x_j = x_{(i)}\right)$$

and

$$t_{i,f} = \sum_{j \in G_2} f_j I\left(x_j = x_{(i)}\right)$$

Let *MinRun* and *MaxRun* be the minimum and maximum number of runs respectively, g_1 be the group indictors at the last run when computing the maximum number of runs, and g_2 be the group indicator when computing the minimum number of runs. Then the following algorithm will compute the minimum and maximum number of runs.

1. *MinRun*=0, *MaxRun*=0, g_1=0, g_2=0, *d*=0, and *i*=0
2. *i*=*i*+1. If *i*>*m*, stop and output *MinRun* and *MaxRun*.
3. $d = s_{i,f} - t_{i,f}$, $Minim = \min\left(s_{i,f}, t_{i,f}\right)$. If *Minim*=0, then go to step 6.
4. $MaxRun = MaxRun + 2Minim$, $MinRun = \max\left(MinRun + 1, 2\right)$.
5. If $d \neq 0$ and $d \times g_1 \leq 0$, then $MaxRun = MaxRun + 1$. If $d \neq 0$, then $g_1 = d$. $g_2 = -g_2$. Go to step 2.
6. If $g_2 \times d < 0$ or *i*=1, then $MinRun = MinRun + 1$. If $g_1 \times d \leq 0$, then $MaxRun = MaxRun + 1$. g_2=*d*, g_1=*d*. Go to step 2.

Test statistic and p-value

Let $n_{1,f} = \sum_{i \in G_1} f_i$ and $n_{2,f} = \sum_{i \in G_2} f_i$. The distribution of the number of runs, *R*, is approximately normal with

$$\mu_R = \frac{2n_{1,f}n_{2,f}}{n_{1,f} + n_{2,f}} + 1$$

$$\sigma_R = \sqrt{\frac{2n_{1,f}n_{2,f}\left(2n_{1,f}n_{2,f} - n_{1,f} - n_{2,f}\right)}{\left(n_{1,f} + n_{2,f}\right)^2\left(n_{1,f} + n_{2,f} - 1\right)}}$$

The test statistic is

$z = \frac{R - \mu_R}{\sigma_R}$, if $n_{1,f} + n_{2,f} \geq 50$. Otherwise

$$z_C = \begin{cases} \left(R - \mu_R + 0.5\right)/\sigma_R & \text{if } \left|R - \mu_R\right| \geq 0.5 \\ 0 & \text{if } \left|R - \mu_R\right| < 0.5 \end{cases}$$

The one sided *p*-value is $p_1 = \Pr(Z \le z) = \Phi(z)$ or $p = \Pr(Z \le z_C) = \Phi(z_C)$, but if $n_{1,f} + n_{2,f} \le 30$ we use the following exact method to compute the one sided *p*-value and do not use the above approximate normal method even if the test statistic was computed. The one-sided exact *p*-value is calculated from

$$p_1 = \Pr(r \le R) = \sum_{r=2}^{R} f_R(r)$$

where

$$f_R(r) = \frac{2\binom{n_{1,f}-1}{r/2-1}\binom{n_{2,f}-1}{r/2-1}}{\binom{n_{1,f}+n_{2,f}}{n_{1,f}}}$$

when *r* is even and when *r* is odd

$$f_R(r) = \frac{\binom{n_{1,f}-1}{(r-1)/2}\binom{n_{2,f}-1}{(r-3)/2} + \binom{n_{1,f}-1}{(r-3)/2}\binom{n_{2,f}-1}{(r-1)/2}}{\binom{n_{1,f}+n_{2,f}}{n_{1,f}}}$$

The conservative decision is made using the biggest number of runs. $p_1 < \alpha$ will reject the null hypothesis.

Kolmogorov-Smirnov Test

For two independent samples from a continuous field, this tests:

H_0: $F_1(x) = F_2(x)$; that is, the two samples are from populations with the same distribution function

H_A: $F_1(x) \ne F_2(x)$ for some *x*

Calculation of the empirical cumulative distribution functions and differences

For each of the two groups, distinct values are sorted into ascending order:

Group 1: $x^*_{(1)} < x^*_{(2)} < \cdots < x^*_{(n_1)}$,

where $x^*_{(i)} \in D_1$ and n_1 is the number of distinct values in G_1.

Group 2: $x^{**}_{(1)} < x^{**}_{(2)} < \cdots < x^{**}_{(n_2)}$,

where $x^{**}_{(i)} \in D_2$ and n_2 is the number of distinct values in G_2.

Then the empirical cumulative distribution functions for Group 1 and Group 2 are computed as:

$$\hat{F}_1(x) = \begin{cases} 0 & -\infty < x < x^*_{(1)} \\ \frac{\sum_{i\in G_1} f_i I\{x \le x^*_{(k)}\}}{n_{1,f}} & x^*_{(k)} \le x < x^*_{(k+1)} \\ 1 & x^*_{(n_1)} \le x < \infty \end{cases}$$

and

$$\hat{F}_2(x) = \begin{cases} 0 & -\infty < x < x^{**}_{(1)} \\ \frac{\sum_{i\in G_2} f_i I\{x \le x^{**}_{(k)}\}}{n_{2,f}} & x^{**}_{(k)} \le x < x^{**}_{(k+1)} \\ 1 & x^{**}_{(n_2)} \le x < \infty \end{cases}$$

where $n_{1,f} = \sum_{i\in G_2} f_i$ and $n_{2,f} = \sum_{i\in G_2} f_i$.

For each x_j, the difference between the two groups is

If $n_{1,f} \ge n_{2,f}$, $d_j = \hat{F}_1(x_j) - \hat{F}_2(x_j)$

If $n_{1,f} < n_{2,f}$, $d_j = \hat{F}_2(x_j) - \hat{F}_1(x_j)$

The maximum positive, negative and absolute differences are also computed.

Test statistic and p-value

The test statistic (Smirnov, 1948) is

$$Z = \max_j |d_j| \sqrt{\frac{n_{1,f} n_{2,f}}{n_{1,f} + n_{2,f}}}$$

The *p*-value is calculated using the Smirnov approximation described in the K-S one-sample test.

$p < \alpha$ rejects the null hypothesis.

Hodges-Lehmann Estimates

Here we assume that two samples follow the same distribution except in the location parameters; that is, if the first sample follows $F(x)$, the second sample follows $F(x+\theta)$. We want to estimate and find the confidence interval for θ.

Let $d_{ij} = x_i - x_j, i \in G_1, j \in G_2$. Incorporating the frequency weight f_i for x_i and f_j for, x_j the frequency weight for d_{ij} is $f^*_{ij} = f_i f_j$. Let $B_{(1)} \le ... \le B_{(L)}$, $L = n_1 n_2$, be the ordered values of d_{ij}, and the corresponding frequency weights are $f^*_{(1)}, ..., f^*_{(L)}$.

The Hodges-Lehmann estimator for θ is $\hat{\theta} = \text{median}\{\mathbf{B}; \mathbf{f}^*\}$.

The Moses' confidence interval for θ is $(B_{(k_1)}, B_{(k_2)}]$.

where the median, k_1 and k_2 are calculated by the same formula as that in the Hodges-Lehmann estimate for paired samples (see "Hodges-Lehmann Estimates") but with μ_T and σ_T replaced by the expected value and standard deviation of the test statistics under null hypothesis in Mann-Whitney's

$$\mu_T = \frac{n_{1,f}n_{2,f}}{2}$$

$$\sigma_T^2 = \frac{n_{1,f}n_{2,f}}{12N_f(N_f-1)}\left(N_f\left(N_f^2-1\right) - \sum_{i=1}^{M} t_{i,f}\left(t_{i,f}^2-1\right)\right)$$
$$= \frac{n_{1,f}n_{2,f}(N_f+1)}{12}\left(1-\frac{\sum_{i=1}^{M} t_{i,f}(t_{i,f}^2-1)}{N_f\left(N_f^2-1\right)}\right)$$

$$N_f = n_{1,f} + n_{2,f}$$

where M is the total number of distinct values among all combined observations, and $t_{i,f}$ is the number of occurrences of the ith distinct value, incorporating the frequency weight.

Moses Test of Extreme Reactions

For two independent samples from a continuous field, this tests:

H_0: Extreme values are equally likely in both populations

H_A: Extreme values are more likely to occur in the population from which the sample with the larger range was drawn.

Span computation

Observations from both specified groups are jointly sorted and ranked, with the average rank being assigned in the case of ties. The smallest and largest ranks of the control group (the group defined by the first value in ascending order) are determined, and the span is computed as

$$\text{SPAN} = \text{The largest rank of control group-the smallest rank of control group} + 1$$

If SPAN is not an integer, then it will be rounded to its nearest integer.

Significance Level

Let $n_{c,f}$ and $n_{e,f}$ be the numbers of records in the control group and experiment group respectively, incorporating the frequency weight, and $g = \text{SPAN} - n_{c,f} + 2h$. Then the exact one-tailed probability of span s is

$$p_1 = \Pr(s \le \text{SPAN}) = \frac{\sum_{i=0}^{g}\left[\binom{i+n_{c,f}-2h-2}{i}\binom{n_{e,f}+2h+1-i}{n_{e,f}-i}\right]}{\binom{n_{c,f}+n_{e,f}}{n_{c,f}}}$$

where *h*=0. The same formula is used below where *h* is not zero.

Censoring the Range

The previous test is repeated, dropping the *h* lowest and *h* highest ranks from the control group, where *h* is a positive user-specified integer (default at the integer part of $0.05n_{c,f}$ or 1, whichever is greater). If $2h > n_{c,f} - 2$, then the test will be implemented using the largest integer such that $2h \leq (n_{c,f} - 2)$.

The exact one-tailed probability is calculated by the formula above, and $p_1 \leq \alpha$ rejects the null hypothesis.

Kruskal-Wallis Test

For *k* independent samples from a continuous field, this tests:

H_0: The distributions of the *k* samples are the same

H_A: At least one sample is different

Sum of Ranks

Observations from all *k* nonempty groups are jointly sorted and ranked, with the average rank being assigned in the case of ties. The number of records tied at the *j*th distinct value $t_{j,f}$ is calculated incorporating the frequency weight, and the sum of $T_{j,f} = t^3_{j,f} - t_{j,f}$ is also accumulated. For each group the sum of ranks, $R_{i,f}$, as well as the number of observations, $n_{i,f}$, is obtained.

Test statistic and p-value

The test statistic unadjusted for ties is

$$H = \frac{12}{N_f\left(N_f+1\right)} \sum_{i=1}^{k} R^2_{i,f}/n_{i,f} - 3\left(N_f+1\right)$$

where $N_f = \sum_{i=1}^{k} n_{i,f}$. The statistic adjusted for ties is

$$H^{'} = \frac{H}{1 - \sum_{i=1}^{m} T_{i,f}/\left(N_f^3 - N_f\right)}$$

where *m* is the total number of tied sets.

The one-sided *p*-value is $p_1 = \Pr(x^2_{k-1} \geq H') = 1 + \Pr(x^2_{k-1} \leq H')$, where x^2_{k-1} follows a chi-square distribution with $k-1$ degrees of freedom.

$p1 < \alpha$ will reject the null hypothesis.

Median Test

For *k* independent samples from a continuous field, this tests:

H_0: $\theta_0 = \theta_1 = \theta_2 = \cdots = \theta_k$; that is, the *k* samples are from populations with the same median

H_A: At least one population median is different

Table Construction

θ_0 is user-specified (default at the sample median of the combined *k* samples).

The number of records in each of the groups that exceed the median are counted and the following table is formed, where $O_{1i,f}$ denotes the number of records that are less than or equal to the median, and $O_{2i,f}$ is the number of records that are greater than the median, in the *i*th group, incorporating the frequency weight. $n_{j,f} = O_{1j,f} + O_{2j,f}$, $R_{i,f} = \sum_{j=1}^{k} O_{ij,f}$ and $N_f = R_{1,f} + R_{2,f}$.

	1	2	...	*k*	Total
LE median	$O_{11,f}$	$O_{12,f}$	...	$O_{1k,f}$	$R_{1,f}$
GT median	$O_{21,f}$	$O_{22,f}$	...	$O_{2k,f}$	$R_{2,f}$
Total	$n_{1,f}$	$n_{2,f}$	...	$n_{k,f}$	N_f

Test statistic and p-value

The χ^2 statistic for all nonempty groups is calculated as

$$\chi^2 = \sum_{j=1}^{k}\sum_{i=1}^{2}(O_{ij} - E_{ij})^2/E_{ij}$$

where $E_{ij} = \frac{R_{i,f}n_{j,f}}{N_f}$.

If *k*=2 and $n_f > 30$, Yates' Continuity Correction for the chi-square statistic is applied

$$\chi_c^2 = \frac{\left(\left|O_{21,f}\left(n_{2,f} - O_{22,f}\right) - O_{22,f}\left(n_{1,f} - O_{21,f}\right)\right| - N_f/2\right)^2 N_f}{\left(O_{21,f} + O_{22,f}\right)\left(n_{1,f} + n_{2,f} - O_{21,f} - O_{22,f}\right)n_{1,f}n_{2,f}}$$

The one sided *p*-value is $p_1 = \Pr\left(\chi_{k-1}^2 \geq \chi^2\right) = 1 - \Pr\left(\chi_{k-1}^2 \leq \chi^2\right)$, where χ_{k-1}^2 follows a chi-square distribution with $k-1$ degrees of freedom, where *k* is the number of nonempty groups.

$p_1 < \alpha$ rejects the null hypothesis. The results may be questionable if any cell has an expected value less than one, or more than 20% of the cells have expected values less than five.

If *k*=2 and $n_f \leq 30$, the two sided *p*-value is computed using Fisher's exact test. For more information, see the topic "Significance Levels for Fisher's Exact Test".

Jonckheere-Terpstra Test

For *k* independent samples from a continuous field, this tests:

*H*0: $F_1(x) = F_2(x) = ... = F_K(x)$; that is, the *k* samples are from populations with the same distribution function

H_A: $H_A : F_1(x) \geq F_2(x) \geq ... \geq F_K(x)$ or $H'_A : F_1(x) \leq F_2(x) \leq ... \leq F_K(x)$ with at least one strict inequality.

Under the assumption that all distribution functions are the same except the location parameters; that is, $F_k(x) = F(x - \tau_k)$, $k = 1, ..., K$, the null and alternative hypotheses become:

$$H_0 : \tau_1 = \tau_2 = ... = \tau_K$$

$H_A : \tau_1 \leq \tau_2 \leq ... \leq \tau_K$ or $H'_A : \tau_1 \geq \tau_2 \geq ... \geq \tau_K$ with at least one strict inequality.

For the k_1th sample and the k_2th sample, the Mann-Whitney *U* count is

$$\begin{aligned} U_{k_1k_2} &= \sum_{i\in G_{k_1}} \sum_{j\in G_{k_2}} f_i f_j I(x_i < x_j) + \tfrac{1}{2} \sum_{i\in G_{k_1}} \sum_{j\in G_{k_2}} f_i f_j I(x_i = x_j) \\ &= n_{k_1,f} n_{k_2,f} + \tfrac{n_{k_1,f}(n_{k_1,f}+1)}{2} - S_{k_1}(k_1,k_2) \\ &= S_{k_2}(k_1,k_2) - \tfrac{n_{k_2,f}(n_{k_2,f}+1)}{2} \end{aligned}$$

where $n_{k,f} = \sum_{i\in G_k} f_i$ and $S_{k_1}(k_1,k_2)$ is the sum of ranks of sample k_1 when sample k_1 and sample k_2 are jointly ranked incorporating frequency weight; that is, $S_{k_1}(k_1,k_2) = \sum_{i\in G_{k_1}} f_i \text{rank}(x_i, (D_{k_1}, D_{k_2}), \mathbf{f})$.

The test statistics is

$$T = \sum_{k_1=1}^{K-1} \sum_{k_2=k_1+1}^{K} U_{k_1k_2}$$

The standardized test statistic is

$$T^* = \frac{T - \mu_T}{\sigma_T}$$

where

$$\mu_T = \frac{1}{4}\left(N_f^2 - \sum_{k=1}^{K} n_{k,f}^2\right)$$

$$\sigma_T^2 = \frac{1}{72}\Big(N_f\left(N_f-1\right)\left(2N_f+5\right) - \sum_{k=1}^{K} n_{k,f}\left(n_{k,f}-1\right)\left(2n_{k,f}+5\right) - 2A_2 - 5A_1\Big)$$
$$+ \frac{\left(\sum_{k=1}^{K} n_{k,f}(n_{k,f}-1)(n_{k,f}-2)\right)(A_2-2A_1)}{36N_f(N_f-1)(N_f-2)} + \frac{\left(\sum_{k=1}^{K} n_{k,f}\left(n_{k,f}-1\right)\right)A_1}{8N_f(N_f-1)}$$

$$N_f = \sum_{k=1}^{K} n_{k,f}$$

$$A_1 = \sum_{i=1}^{M} t_{i,f}\left(t_{i,f}-1\right)$$
$$A_2 = \sum_{i=1}^{M} t_{i,f}^2\left(t_{i,f}-1\right)$$

and *M* is the total number of distinct values among all combined observations, and $t_{i,f}$ is the number of occurrences of the *i*th distinct value considering the frequency weight.

The one sided and two sided *p*-values are

$$p_1 = \Pr\left(Z \geq |T^*|\right) = 1 - \Phi\left(|T^*|\right)$$

$$p = 2p_1$$

$p_1 < \alpha$ will reject the null hypothesis in favor of $H'_A : F_1(x) \leq F_2(x) \leq \ldots \leq F_K(x)$ if $T^* < 0$ and in favor of $H_A : F_1(x) \geq F_2(x) \geq \ldots \geq F_K(x)$ if $T^* > 0$.

$p < \alpha$ will reject the null hypothesis in favor of an ordered alternative (either direction of ordering).

Note: When there are only two samples, *K* = 2, the Jonckheere-Terpstra test reduces to the Mann-Whitney test.

One-sided test

If the direction of the alternative is specified, this becomes a one-sided test. The previously defined one-sided *p*-value is not the p-value for a fixed one-sided test, and cannot be used alone to make decision for one-sided test.

If the alternative is $H'_A : F_1(x) \leq F_2(x) \leq \ldots \leq F_K(x)$, the *p*-value for the one-sided test is

$$p = \Pr\left(Z \leq T^*\right) = \Phi\left(T^*\right) = \begin{cases} 1-p_1 & T^* \geq 0 \\ p_1 & T^* < 0 \end{cases}$$

If the alternative is $H_A : F_1(x) \geq F_2(x) \geq \ldots \geq F_K(x)$, the *p*-value for the one-sided test is

$$p = \Pr\left(Z \geq T^*\right) = 1 - \Phi\left(T^*\right) = \begin{cases} p_1 & T^* \geq 0 \\ 1-p_1 & T^* < 0 \end{cases}$$

Note: The one-sided test will be used in multiple comparisons for Jonckheere-Terpstra test.

Related Samples Tests

The following related samples tests are available.

McNemar's Test

For two related samples from a categorical field with 2 values (or a recoded categorical field with more than 2 values), this tests:

H_0: The two samples have the same marginal distribution.

Let $n_{1,f}$ be the number of records in which x_i is a success and y_i is a failure, and $n_{2,f}$ be the number of records in which x_i is a failure and y_i is a success, incorporating the frequency weights.

If $n_{1,f} + n_{2,f} \le 25$, the two-sided exact probability is

$$p = 2\sum_{i=0}^{r} \binom{n_{1,f} + n_{2,f}}{i} 0.5^{n_{1,f}+n_{2,f}}$$

where $r = \min(n_{1,f}, n_{2,f})$.

If $n_{1,f} + n_{2,f} > 25$, the test statistic is

$$\chi_c^2 = \frac{\left(\left|n_{1,f} - n_{2,f}\right| - 1\right)^2}{n_{1,f} + n_{2,f}}$$

The one sided *p*-value is

$$p = \Pr\left(\chi_1^2 \ge \chi^2\right) = 1 - \Pr\left(\chi_1^2 \le \chi^2\right)$$

where χ_1^2 has a chi-square distribution with 1 degree of freedom.

$p < \alpha$ will reject the null hypothesis.

Wilcoxon Signed-Rank Test

For two related samples from a continuous field, this tests:

H_0: θ = Median(X_1−X_2) = 0

H_A: $\theta < 0$or $\theta > 0$

Computing Ranked Differences

For each record, the difference $d_i = x_{i1} - x_{i2}$ is computed, as well as the absolute value. All nonzero absolute differences are sorted into ascending order, and ranks are assigned. In the case of ties, the average rank is used. Let $D = \{d_i : |d_i| \neq 0\}$, then the sums of the ranks corresponding to positive and negative differences are

$$S_p = \sum_{i \in D} f_i rank\left(|d_i|\, ; D, \mathbf{f}\right) I\left(sign\left(d_i\right) > 0\right)$$

and

$$S_n = \sum_{i \in D} f_i rank\left(|d_i|\, ; D, \mathbf{f}\right) I\left(sign\left(d_i\right) < 0\right)$$

respectively. Then the average positive rank and average negative rank are

$$\overline{X}_p = S_p / n_{p,f}$$

and

$$\overline{X}_n = S_n / n_{n,f}$$

where n_p is the number of records with positive differences and n_n the number with negative differences.

Test statistic and p-value

The test statistic is

$$T = \frac{S_p - \mu_T}{\sigma_T}$$

where

$$\mu_T = \frac{n_f\left(n_f + 1\right)}{4}$$

$$\sigma_T^2 = n_f\left(n_f + 1\right)\left(2n_f + 1\right)/24 - \sum_{j=1}^{l}\left(t_{j,f}^3 - t_{j,f}\right)/48$$

$$n_f = \sum_{i \in D} f_i$$

where *l* is the total number of distinct rank values and $t_{j,f}$ is the number of records tied at the *j*th distinct value, incorporating the frequency weight.

The one-sided and two-sided *p*-values are

$$p_1 = \Pr\left(Z \geq |T|\right) = 1 - \Phi\left(|T|\right)$$

$$p = 2p_1$$

$p_1 < \alpha$ will reject the null hypothesis in favor of $\theta > 0$ if $T > 0$ and $\theta < 0$ if $T < 0$.

$p < \alpha$ will reject the null hypothesis in favor of $\theta > 0$ or $\theta < 0$.

Sign Test

For two related samples from a continuous field, this tests:

H_0: θ = Median($X_1 - X_2$) = 0

H_A: $\theta < 0$ or $\theta > 0$

Counting Signs

For each record, the difference $d_i = x_{i1} - x_{i2}$ is computed and the number of positive($n_{p,f}$) and negative($n_{n,f}$) differences, incorporating the frequency weight, are counted:

$$n_{p,f} = \sum_{i=1}^{n} f_i I\left(d_i > 0\right)$$
$$n_{n,f} = \sum_{i=1}^{n} f_i I\left(d_i < 0\right)$$

Cases with $x_{i1} = x_{i2}$ are ignored.

Test statistic and p-value

If $n_{p,f} + n_{n,f} = 0$, then the one-sided exact probability is $p_1 = 0.5$.

If $0 < n_{p,f} + n_{n,f} \leq 25$, then p_1 is calculated recursively from the binomial distribution:

$$p_1 = \min\left\{p^*, p^{**}\right\}$$

where

$$p^* = \sum_{i=0}^{n_{p,f}} \binom{n_{p,f} + n_{n,f}}{i} 0.5^{n_{p,f}+n_{n,f}}$$

and

$$p^{**} = 1 - \sum_{i=0}^{n_{p,f}-1} \binom{n_{p,f} + n_{n,f}}{i} 0.5^{n_{p,f}+n_{n,f}}$$

If $n_{p,f} + n_{n,f} > 25$, the test statistic is

$$z_c = \frac{\max\left(n_{p,f}, n_{p,f}\right) - 0.5\left(n_{p,f} + n_{n,f}\right) - 0.5}{0.5\sqrt{n_{p,f} + n_{n,f}}}$$

The one sided and two sided *p*-values are

$$p_1 = \Pr\left(Z \geq |z_c|\right) = 1 - \Phi\left(|z_c|\right)$$

$$p = 2p_1$$

$p_1 < \alpha$ rejects the null hypothesis in favor of $\theta > 0$ if $n_{p,f} > n_{n,f}$ and $\theta < 0$ if $n_{p,f} < n_{n,f}$.

$p < \alpha$ will reject the null hypothesis in favor of $\theta > 0$ or $\theta < 0$.

Marginal Homogeneity

For two related samples from an ordinal field, this tests:

*H*0: The two samples have the same marginal distribution.

Let $n_{uv,f}$ be the cell count incorporating the frequency weight for cell ($x_1 = u, x_2 = v$)

$$n_{uv,f} = \sum_{i=1}^{n} I\left(x_{i1} = u, x_{i2} = v\right) f_i$$

The test statistic is

$$T = \sum_{u} \sum_{v \neq u} w_u n_{uv,f}$$

The standardized test statistics is

$$T^* = \frac{T - \mu_T}{\sigma_T}$$

where

$$\mu_T = \frac{1}{2} \sum_{u} \sum_{v \neq u} \left(w_u + w_v\right) n_{uv,f}$$

$$\sigma_T^2 = \sum_u \sum_{v \neq u} \left(\frac{w_u - w_v}{2} \right)^2 n_{uv,f}$$

The asymptotic one sided *p*-value is $p_1 = \Pr(Z \geq |T^*|) = 1 - \Phi(|T^*|)$.

$p_1 < \alpha$ rejects the null hypothesis in favor of $F_1(x) \geq F_2(x)$ if $T^* < 0$ or $F_1(x) \leq F_2(x)$ if $T^* > 0$ with at least one *x* gives strict inequality.

The asymptotic two sided *p*-value is $p = 2p_1$.

Note: Any linear transformation of scores produces the same standardized test statistic and *p*-value.

Hodges-Lehmann Estimates

For two related samples from a continuous field, this finds a confidence interval for the median difference: letting $d_i = x_{i1} - x_{i2}$, we assume that d_i follows a symmetric distribution with median θ.

Let $A_{ij} = \frac{d_i + d_j}{2}, i \leq j$. Incorporating the frequency weight f_i for d_i and f_j for d_j, the frequency weight for A_{ij} is

$$f_{ij}^* = \begin{cases} \frac{1}{2} f_i (f_i + 1) & i = j \\ f_i f_j & j \end{cases}$$

Let $B_{(1)} \leq ... \leq B_{(L)}, L = \frac{n(n+1)}{2}$, be the ordered values of $A_{ij}, i \leq j$, and the corresponding frequency weights are $f_{(1)}^*, ..., f_{(L)}^*$.

The Hodges-Lehmann estimator for θ is the median of $B_{(1)}, ..., B_{(L)}$ incorporating the frequency weights

$$\hat{\theta} = \text{median}\{\mathbf{B}; \mathbf{f}^*\} = \begin{cases} \frac{B_{(k)} + B_{(k+1)}}{2} & W_{(L)} = \text{even and} W_{(k)} < \frac{W_{(L)}+1}{2} < W_{(k)} + 1 \\ B_{(k)} & \text{otherwise} \frac{W_{(L)}+1}{2} \in \left[W_{(k-1)} + 1, W_{(k)}\right] \end{cases}$$

where $W_{(k)} = \sum_{i=1}^{k} f_{(i)}^*$.

The Tukey's confidence interval for θ is

$$\left(B_{(k_1)}, B_{(k_2)}\right]$$

where k_1 and k_2 are integers such that

$$i_1 \in \left[W_{(k_1-1)} + 1, W_{(k_1)}\right]$$
$$i_2 \in \left[W_{(k_2-1)} + 1, W_{(k_2)}\right]$$

with

$$i_1 = 1 + \lfloor \mu_T - z_{\alpha/2}\sigma_T \rfloor$$

$$i_2 = \lceil \mu_T + z_{\alpha/2}\sigma_T \rceil$$

and $\lfloor x \rfloor$ and $\lceil x \rceil$ are the floor and ceiling integers of *x*, μ_T and σ_T are the expected value and standard deviation of the test statistic *T* under the null hypothesis in the Wilcoxon signed rank test, and $z_{\alpha/2}$ is the right tail percentile such that $\Pr(Z > z_{\alpha/2}) = \alpha/2$ where *Z* is a random variate following a standard normal distribution.

Cochran's Q Test

For *k* related samples from a categorical field with 2 values (or recoded categorical field with more than 2 values), this tests:

H_0: The distributions of these *k* samples are the same.

For each record, the number of successes across samples is counted. The number of successes for record *i* is

$$R_{i,f} = \sum_{j=1}^{k} I\,(\text{ if } x_{ij} \text{ is success})$$

and the total number of successes for sample *l*, incorporating the frequency weights, is

$$C_{l,f} = \sum_{i=1}^{n} f_i I\,(\text{ if } x_{il} \text{ is success})$$

The test statistic is

$$Q = \frac{(k-1)\left[k\sum_{l=1}^{k} C_{l,f}^2 - \left(\sum_{l=1}^{k} C_{l,f}\right)^2\right]}{k\sum_{l=1}^{k} C_{l,f} - \sum_{i=1}^{n} f_i R_{i,f}^2}$$

The one-sided *p*-value is

$$p = \Pr\left(\chi_{k-1}^2 \geq \chi^2\right) = 1 - \Pr\left(\chi_{k-1}^2 \leq \chi^2\right)$$

where χ_{k-1}^2 follows a chi-square distribution with $k-1$ degrees of freedom.

$p < \alpha$ rejects the null hypothesis.

Friedman's Test

For *k* related samples from a continuous field, this tests:

H_0: The distributions of these *k* samples are the same.

For each record, the *k* samples are sorted and ranked, with average rank being assigned in the case of ties. For each sample, the sum of ranks over the records is calculated, incorporating the frequency weight, as follows:

$$C_{l,f} = \sum_{i=1}^{n} f_i \text{rank}\left(x_{il}, D_i, \text{f}\right)$$

where $D_i = \{x_{ij}, j = 1, \cdots, k\}$. The average rank for each sample is

$$\overline{R}_{l,f} = C_{l,f}/n_f$$

where $n_f = \sum_{i=1}^{n} f_i$.

The test statistic is

$$\chi^2 = \frac{\left(12/n_f k\left(k+1\right)\right)\sum_{l=1}^{k} C_{l,f}^2 - 3n_f\left(k+1\right)}{1 - \Sigma T/n_f k\left(k^2 - 1\right)}$$

where

$$\Sigma T = \sum_{i=1}^{n}\sum_{j=1}^{m_i}\left(t_{ij,f}^3 - t_{ij,f}\right)$$

and m_i is the total number of distinct rank values of the *i*th record and t_{ij} is the number of fields tied at the *j*th distinct value of the *i*th record, incorporating the frequency weight. The one-sided *p*-value is

$$p = \Pr\left(\chi_{k-1}^2 \geq \chi^2\right) = 1 - \Pr\left(\chi_{k-1}^2 \leq \chi^2\right)$$

where χ_{k-1}^2 follows a chi-square distribution with $k-1$ degrees of freedom.

$p < \alpha$ rejects the null hypothesis.

Kendall's Coefficient of Concordance

For *k* related samples from a continuous field, this tests:

H_0: The distributions of these *k* samples are the same.

The coefficient of concordance (W) is

$$W = \left(\frac{F}{n_f\,(k-1)}\right)$$

where *F* is the Friedman χ^2 statistic and $n_f = \sum_{i=1}^{n} f_i$.

The test statistic is

$$\chi^2 = n_f\,(k-1)\,W$$

The one-sided *p*-value is calculated as

$$p = \Pr\left(\chi^2_{k-1} \geq \chi^2\right) = 1 - \Pr\left(\chi^2_{k-1} \leq \chi^2\right)$$

where χ^2_{k-1} follows a chi-square distribution with $k-1$ degrees of freedom.

$p < \alpha$ rejects the null hypothesis.

Multiple Comparisons

Tests such as Kruskal-Wallis involve more than two samples. They test if all samples are from populations with the same characteristics. This characteristic may be the distribution, mean or median depending on the hypotheses. Denote the overall null hypothesis as $H_0 : \theta_1 = ... = \theta_K$. When this overall hypothesis is rejected at the user-specified significance level α (using two-sided *p*-values except for the Jonckheere-Terpstra test here), we may want to know where the differences are among the populations. Two multiple comparison procedures are considered to answer this question: pairwise multiple comparisons and a stepwise stepdown procedure for multiple comparisons.

Pairwise Multiple Comparisons

All possible pairwise hypotheses like $H_{0,jk} : \theta_j = \theta_k$ for $1 \leq j < k \leq K$ are tested. There are $K\,(K-1)\,/2$ of them. In order to control the familywise type I error; that is, the probability of rejecting at least one pair hypothesis given all pairwise hypotheses are true, adjusted *p*-values are calculated and used to make the decision for each pair. For pair (*j*, *k*), reject $H_{0,jk}$ at level α if $p_{adj,jk} < \alpha$. The adjusted *p*-values are calculated the following way.

Calculate the *p*-value, for each of the pairwise hypotheses.

Calculate the adjusted *p*-value as $p_{adj} = pK\,(K-1)\,/2$.

Notes

- If the adjusted *p*-value is bigger than 1, it is set to 1. The calculation of the *p*-value in step 1 depends on the specific method used to do the overall test. The details are listed below; in the following, two-sided *p*-values are used except for the Jonckheere-Terpstra test.
- The Kruskal-Wallis, Friedman and Kendall, and Cochran tests use the procedure proposed by Dunn (1964) (originally designed for the Kruskal-Wallis test). The procedure uses ranks (or successes for the Cochran test) based on considering all samples rather than just the two involved in a given comparison.

Kruskal-Wallis test

Let $R_j = \sum_{i \in G_j} f_i \text{rank}\left(x_i, D, \mathbf{f}\right)$ be the sum of ranks for sample *j*, incorporating frequency weights.

For testing $H_{0,jk} : \theta_j = \theta_k$, the test statistic is $T_{jk} = \frac{R_j}{n_{j,f}} - \frac{R_k}{n_{k,f}}$.

The standardized test statistic is $T^*_{jk} = \frac{T_{jk}}{\sigma_{jk}}$ where

$$\sigma^2_{jk} = A\left(\frac{1}{n_{j,f}} + \frac{1}{n_{k,f}}\right)$$

$$A = \left(\frac{N_f\left(N_f+1\right)}{12} - \frac{\sum_{i=1}^{M}\left(t^3_{i,f} - t_{i,f}\right)}{12\left(N_f - 1\right)}\right)$$

and *M* is the total number of distinct values among all observations, and $t_{i,f}$ is the number of occurrences of the *i*th distinct value incorporating the frequency weight.

The two-sided *p*-value is

$$p_{jk} = \Pr\left(|Z| > \left|T^*_{jk}\right|\right) = 2\left(1 - \Phi\left(\left|T^*_{jk}\right|\right)\right)$$

K-sample median test

For $H_{0,jk} : \theta_j = \theta_k$, perform the median test using data only consisting of sample *j* and sample *k* as if other samples don't exist. In this test the median of the two samples is found, and the number above and below that median is used in the test.

Jonckheere-Terpstra test for ordered alternatives

For pair (*j*, *k*), *j*<*k*, the null and alternative hypotheses are $H_{0,jk} : \theta_j = \theta_k$ vs $H_{A,jk} : \theta_j > \theta_k$ if the overall alternative hypotheses $H_A : \theta_1 \geq ... \geq \theta_K$ is specified or favored (it is favored if $T^* > 0$ where T^* is the standardized test statistic in the overall Jonckheere-Terpstra test), or vs $H_{A,jk} : \theta_j < \theta_k$ if $H_A : \theta_1 \leq ... \leq \theta_K$ is specified or favored (it is favored if $T^* < 0$). Use Mann-Whitney's *U* test on each pair of hypotheses to calculate the *p*-value for the one-sided test.

Friedman's test and Kendall's Coefficient of Concordance

For treatment *j*, let $R_{j,f} = \sum_{i=1}^{n} f_i \text{rank}(x_{ij}, \text{row} i \text{ of } D)$ be the sum of ranks for sample *j*.

For testing $H_{0,jk} : \theta_j = \theta_k$, the test statistic is $T_{jk} = \frac{R_{j,f} - R_{k,f}}{n_f}$.

The standardized test statistic is $T^*_{jk} = \frac{T_{jk}}{\sigma}$ where $\sigma^2 = \frac{K(K+1)}{6n_f}$.

The two-sided *p*-value is $p_{jk} = \Pr\left(|Z| > \left|T^*_{jk}\right|\right) = 2\left(1 - \Phi\left(\left|T^*_{jk}\right|\right)\right)$.

Cochran's Q Test

Using $x_{ij} = 1$ to represent success and $x_{ij} = 0$ to represent failure, let $C_{j,f} = \sum_{i=1}^{n} f_i x_{ij}$ be the total number of successes for sample *j* incorporating frequency weights, and $R_i = \sum_{j=1}^{K} x_{ij}$ be the total number of successes for record *i*.

The test statistic for $H_{0,jk} : \theta_j = \theta_k$ is $T_{jk} = \frac{C_{j,f} - C_{k,f}}{n_f}$.

The standardized test statistic is $T^*_{jk} = \frac{T_{jk}}{\sigma}$ where

$$\sigma^2 = 2\frac{K\sum_{i=1}^{n} f_i R_i - \sum_{i=1}^{n} f_i R_i^2}{n_f^2 K(K-1)}$$

The two-sided *p*-value is $p_{jk} = \Pr\left(|Z| > \left|T^*_{jk}\right|\right) = 2\left(1 - \Phi\left(\left|T^*_{jk}\right|\right)\right)$.

Stepwise Stepdown Multiple Comparisons

The procedure described in this section is an extension of the ad hoc procedure developed by Campbell and Skillings (1985). This procedure starts with the overall hypothesis involving all *K* populations, and if the hypothesis is rejected, then it considers the sub-hypotheses involving *K*−1 populations, continuing until the hypothesis only involves two populations or no hypotheses are rejected. If all sub-hypotheses are considered, it may be computationally too expensive when *K* is big, so a shortcut is used on the sorted samples. This procedure returns a sequence of subsets of populations with homogeneous characteristics.

Sort the samples

The *K* samples are sorted from the smallest to largest by test-specific criteria. Let (1), ..., (*K*) index the sorted samples.

- Kruskal-Wallis: average treatment rank, where rank is the joint rank of all the observations. Use the treatment median to break ties.
- Median: treatment median
- Jonckheere-Terpstra test: given by the user-specified alternative hypothesis order.
- Friedman: average treatment rank (same as using treatment rank sum) where rank is the joint within row/block ranking.

- Kendall's coefficient of concordance: same as Friedman
- Cochran's Q: average treatment mean, which is the same as using the treatment sum.

Find the homogeneous subsets

Starting with sample (1), sequentially test $H_0 : \theta_{(1)} = \theta_{(2)}$, then $H_0 : \theta_{(1)} = \theta_{(2)} = \theta_{(3)}$, and so on, until the null hypothesis is rejected when sample (*j*) is added. Samples (1) through (*j*−1) are considered homogenous. The process repeats starting with sample (*j*) and continues until sample (*K*).

References

Brown, L. D., T. Cai, and A. DasGupta. 2003. Interval estimation in exponential families. *Statistica Sinica*, 13, 19–49.

Campbell, G., and J. H. Skillings. 1985. Nonparametric Stepwise Multiple Comparison Procedures. *Journal of the American Statistical Association*, 80, 998–998.

Dineen, L. C., and B. C. Blakesley. 1973. Algorithm AS 62: Generator for the sampling distribution of the Mann-Whitney U statistic. *Applied Statistics*, 22, 269–273.

Dunn, O. J. 1964. Multiple Comparisons Using Rank Sums. *Technometrics*, 6, 241–241.

Gibbons, J. D., and S. Chakraborti. 2003. *Nonparametric Statistical Inference, 4th edition*. : Marcel Dekker.

Hochberg, Y., and A. C. Tamhane. 1987. *Multiple Comparison Procedures*. New York: John Wiley & Sons, Inc. .

Hollander, M., and D. A. Wolfe. 1999. *Nonparametric Statistical Methods, 2nd edition*. New York: John Wiley & Sons.

Lehmann, E. L. 1985. *Nonparametrics: Statistical Methods Based on Ranks*. San Francisco: McGraw-Hill.

Sheskin, D. J. 2007. *Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition*. : Chapman & Hall/CRC.

Smirnov, N. V. 1948. Table for estimating the goodness of fit of empirical distributions. *Annals of the Mathematical Statistics*, 19, 279–281.

NPAR TESTS Algorithms

If a WEIGHT variable is specified, it is used to replicate a case as many times as indicated by the weight value rounded to the nearest integer. If the workspace requirements are exceeded and sampling has been selected, a random sample of cases is chosen for analysis using the algorithm described in SAMPLE. For the RUNS test, if sampling is specified, it is ignored. The tests are described in (Siegel, 1956).

One-Sample Chi-Square Test

Cell Specification

If the (lo, hi) specification is used, each integer value in the lo to hi range is designated a cell. Otherwise, each distinct value encountered is considered a cell.

Observed Frequencies

If (lo, hi) has been selected, every observed value is truncated to an integer and, if it is in the lo to hi range, it is included in the frequency count for the corresponding cell. Otherwise, a count of the frequency of occurrence of the distinct values is obtained.

Expected Frequencies

If none or EQUAL is specified,

$$EXP_i = \frac{\text{number of observations } (N)\text{[in range]}}{\text{number of cells } (k)}$$

When the expected values (E_i) are specified either as counts, percentages, or proportions,

$$EXP_i = \left(\frac{E_i}{\sum_{i=1}^{k} E_i} \right) N$$

If there are cells with expected values less than 5, the number of such cells and the minimum expected value are printed.

If the number of user-supplied expected frequencies is not equal to the number of cells generated, or if an expected value is less than or equal to zero, the test terminates with an error message.

Chi-Square and Its Degrees of Freedom

$$\chi^2 = \sum_{i=1}^{k} \frac{(O_i - EXP_i)^2}{EXP_i}$$
$$df = k - 1$$

The significance level is from the chi-square distribution with $k-1$ degrees of freedom.

Kolmogorov-Smirnov One-Sample Test

Calculation of Empirical Cumulative Distribution Function

The observations are sorted into ascending order $X_{(1)}$ to $X_{(N)}$. The empirical cdf, $\hat{F}(X)$, is

$$\hat{F}(X) = \begin{cases} 0 & -\infty < X < X_{(1)} \\ i/N & X_{(i)} \leq X < X_{(i+1)} \quad i = 1, \ldots, N-1 \\ 1 & X_{(N)} \leq X < \infty \end{cases}$$

Estimation of Parameters for Theoretical Distribution

It is possible to test that the underlying distribution is either uniform, normal, or Poisson. If the parameters are not specified, they are estimated from the data.

Uniform

$$\begin{aligned} \text{minimum} &= X_{(1)} \\ \text{maximum} &= X_{(N)} \end{aligned}$$

Normal

$$\text{mean } (\overline{X}) = \sum_{i=1}^{N} X_i/N$$

$$\text{standard deviation } (S) = \sqrt{\left(\sum_{i=1}^{N} X_i^2 - \left(\sum_{i=1}^{N} X_i/N\right)\left(\sum_{i=1}^{N} X_i\right)\right)/(N-1)}$$

Poisson

$$\text{mean } (\lambda) = \sum_{i=1}^{N} X_i/N$$

The test is not done if, for the uniform, all data are not within the user-specified range or, for the Poisson, the data are not non-negative integers. If the variance of the normal or the mean of the Poisson is zero, the test is also not done.

Calculation of Theoretical Cumulative Distribution Functions

For Uniform

$$F_0(X_i) = \frac{X_i - \min}{\max - \min}$$

For Poisson

$$F_0(X_i) = \sum_{l=0}^{X_i} \frac{e^{-\lambda}\lambda^l}{l!}$$

If $\lambda \geq 100,000$, the normal approximation is used.

For Normal

$$F_0(X_i) = F_{0,1}\left(\frac{X_i - \overline{X}}{S}\right)$$

where the generation of $F_{0,1}(Z)$ is described in “Significance Level of a Standard Normal Deviate”.

Calculation of Differences

For the Uniform and Normal, two differences are computed:

$$\begin{aligned} D_i &= \hat{F}(X_{i-1}) - F_0(X_i) \\ \tilde{D}_i &= \hat{F}(X_i) - F_0(X_i) \quad i = 1, \ldots, N \end{aligned}$$

For the Poisson:

$$\begin{aligned} D_i &= \begin{cases} \hat{F}(X_i - 1) - F(X_i - 1) & X_i > 0 \quad i = 1, 2, \ldots, N. \\ 0 & X_i = 0 \end{cases} \\ \tilde{D}_i &= \hat{F}(X_i) - F(X_i) \end{aligned}$$

The maximum positive, negative, and absolute differences are printed.

Test Statistic and Significance

The test statistic is

$$Z = \sqrt{N}\max_i\left(|D_i|, \left|\tilde{D}_i\right|\right)$$

The two-tailed probability level is estimated using the first three terms of the Smirnov (1948) formula.

$$\begin{aligned} &\text{if } 0 \leq Z < 0.27, \quad p = 1 \\ &\text{if } 0.27 \leq Z < 1, \quad p = 1 - \tfrac{2.506628}{Z}\left(Q + Q^9 + Q^{25}\right) \end{aligned}$$

where $Q = e^{-1.233701Z^{-2}}$.

if $1 \leq Z < 3.1, \quad p = 2\big(Q - Q^4 + Q^9 - Q^{16}\big)$

where $Q = e^{-2Z^2}$.

if $Z \geq 3.1, \quad p = 0$

Note: If the distribution is normal and parameters are estimated from the data, then the Lilliefors method is used to compute the test statistic and p value instead of the method described in this section. For more information, see the topic “Kolmogorov-Smirnov Statistic with Lilliefors' Significance”.

Runs Test

Computation of Cutting Point

The cutting point which is used to dichotomize the data can be specified as a particular number, or the value of a statistic which is to be calculated. The possible statistics are

$$\text{Mean} = \sum_{i=1}^{N} X_i/N$$

$$\text{Median} = \begin{cases} \big(X_{(N/2+1)} + X_{(N/2)}\big)/2 & \text{if } N \text{ is even} \\ X_{((N+1)/2)} & \text{if } N \text{ is odd} \end{cases}$$

where the data are sorted in ascending order from $X_{(1)}$, the smallest, to $X_{(N)}$, the largest.

Mode = most frequently occurring value

If there are multiple modes, the one largest in value is selected and a warning printed.

Number of Runs

For each of the data points, in the sequence in the file, the difference

$D_i = X_i$ –CUTPOINT

is computed. If $D_i \geq 0$, the difference is considered positive, otherwise negative. The number of times the sign changes, that is, $D_i \geq 0$ and $D_{i+1} < 0$, or $D_i < 0$ and $D_{i+1} \geq 0$, as well as the number of positive (n_p) and (n_a) signs, are determined. The number of runs (R) is the number of sign changes plus one.

Significance Level

The sampling distribution of the number of runs (R) is approximately normal with

$$\mu_r = \frac{2n_p n_a}{n_p+n_a} + 1$$

$$\sigma_r = \sqrt{\frac{2n_p n_a(2n_p n_a - n_a - n_p)}{(n_p+n_a)^2(n_p+n_a-1)}}$$

The two-sided significance level is based on

$$Z = \frac{R - \mu_r}{\sigma_r}$$

unless $n < 50$; then

$$Z_c = \begin{cases} (R - \mu_r + 0.5)/\sigma_r & \text{if } R - \mu_r \le 0.5 \\ (R - \mu_r - 0.5)/\sigma_r & \text{if } R - \mu_r \ge 0.5 \\ 0 & \text{if } |R - \mu_r| < 0.5 \end{cases}$$

Binomial Test

Table 70-1
Notation

Notation	Description
n_1	Number of observations in the first (test) category
n_2	Number of observations in the second category
p	Test probability
m	$\min(n_1, n_2)$
N	$n_1 + n_2$
p^*	p if $m = n_1$, $1 - p$ if $m = n_2$

When the test probability is equal to 0.5, a two-tailed test is performed. The two-tailed probability is

$$\min\left(1, 2\left(\sum_{i=0}^{m}\binom{N}{i}0.5^N\right)\right)$$

When the test probability is not equal to 0.5, a one-tailed test is performed. The one-tailed probability is

$$\sum_{i=0}^{m}\binom{N}{i}p^{*i}(1-p^*)^{N-i}$$

McNemar's Test

Table Construction

The data values are searched to determine the two unique response categories. If the variables X and Y take on more than two values, or only one value, a message is printed and the test is not done. The number of cases that have $X_i < Y_i(n_1)$ or $X_i > Y_i(n_2)$ are counted.

Test Statistic and Significance Level

If $n_1 + n_2 \leq 25$, the exact probability of *r* or fewer "successes" occurring in $n_1 + n_2$ trials when $p = 0.5$ and $r = \min(n_1, n_2)$ is calculated recursively from the binomial.

$$p(X \leq r) = \sum_{i=0}^{r} \binom{n_1 + n_2}{i} (0.5)^{n_1+n_2}$$

The two-tailed probability level is obtained by doubling the computed value. If $n_1 + n_2 > 25$, a χ^2 approximation with a correction for continuity is used.

$$\chi_c^2 = \frac{(|n_1 - n_2| - 1)^2}{n_1 + n_2}, \quad df = 1$$

Sign Test

Count of Signs

For each case, the difference

$$D_i = X_i - Y_i$$

is computed and the number of positive (n_p) and negative (n_n) differences counted. Cases in which $X_i = Y_i$ are ignored.

Test Statistic and Significance Level

If $n_p + n_n \leq 25$, the exact probability of *r* or fewer "successes" occurring in $n_p + n_n$ trials, when $p = 0.5$ and $r = \min(n_p, n_n)$, is calculated recursively from the binomial

$$p(X \leq r) = \sum_{i=0}^{r} \binom{n_p + n_n}{i} (0.5)^{n_p+n_n}$$

If $n_p + n_n > 25$, the significance level is based on the normal approximation

$$Z_c = \frac{\max(n_p, n_n) - 0.5(n_p + n_n) - 0.5}{0.5\sqrt{n_p + n_n}}$$

A two-tailed significance level is printed.

Wilcoxon Matched-Pairs Signed-Rank Test

Computation of Ranked Differences

For each case, the difference

$$D_i = X_i - Y_i$$

is computed, as well as the absolute value of D_i. All nonzero absolute differences are then sorted into ascending order, and ranks are assigned. In the case of ties, the average rank is used. The sums of the ranks corresponding to positive differences (S_p) and negative differences (S_n) are calculated. The average positive rank is

$$\overline{X}_p = S_p/n_p$$

and the average negative rank is

$$\overline{X}_n = S_n/n_n$$

where n_p is the number of cases with positive differences and n_n the number with negative differences.

Test Statistic and Significance Level

The test statistic is

$$Z = \frac{\min(S_p, S_n) - (n(n+1)/4)}{\sqrt{n(n+1)(2n+1)/24 - \sum_{j=1}^{l}\left(t_j^3 - t_j\right)/48}}$$

where

Table 70-2
Notation

Notation	Description
n	Number of cases with non-zero differences
l	Number of ties
t_j	Number of elements in the j-th tie, $j = 1, \ldots, l$

For large sample sizes the distribution of Z is approximately standard normal. A two-tailed probability level is printed.

Cochran’s Q Test

Computation of Basic Statistics

For each of the N cases, the k variables specified may take on only one of two possible values. If more than two values, or only one, are encountered, a message is printed and the test is not done. The first value encountered is designated a “success” and for each case the number of variables that are "successes" are counted. The number of "successes" for case i will be designated R_i and the total number of "successes" for variable *l* will be designated C_l.

Test Statistic and Level of Significance

Cochran's *Q* is calculated as

$$Q = \frac{(k-1)\left[k\sum_{l=1}^{k} C_l^2 - \left(\sum_{l=1}^{k} C_l\right)^2\right]}{k\sum_{l=1}^{k} C_l - \sum_{i=1}^{N} R_i^2}$$

The significance level of *Q* is from the χ^2 distribution with $k-1$ degrees of freedom.

Friedman's Test

Sum of Ranks

For each of the *N* cases, the *k* variables are sorted and ranked, with average rank being assigned in the case of ties. For each of the *k* variables, the sum of ranks over the cases is calculated. This will be denoted as C_l. The average rank for each variable is

$$\overline{R}_l = C_l/N$$

Test Statistic and Significance Level

The test statistic is

$$\chi^2 = \frac{(12/Nk(k+1))\sum_{l=1}^{k} C_l^2 - 3N(k+1)}{1 - \Sigma T/Nk\left(k^2-1\right)}$$

where ΣT is the same as in Kendall's coefficient of concordance. See (Lehmann, 1985) p. 265.

The significance level is from the χ^2 distribution with $k-1$ degrees of freedom.

Kendall's Coefficient of Concordance

N, *k*, and *l* are the same as in Friedman, in the previous section.

Coefficient of Concordance (W)

$$W = \left(\frac{F}{N(k-1)}\right)\left(\frac{N^2k(k^2-1)/12}{N^2k(k^2-1)/12 - N\Sigma T/12}\right)$$

where $F =$ Friedman χ^2 statistic.

$$\Sigma T = \sum_{i=1}^{N}\sum_{l=1}^{k}(t^3 - t)$$

with t = number of variables tied at each tied rank for each case.

Test Statistic and Significance Level

$$\chi^2 = N(k-1)W$$

The significance level is from the χ^2 distribution with $k-1$ degrees of freedom.

The Two-Sample Median Test

Table Construction

If the median value is not specified by the user, the combined data from both samples are sorted and the median calculated.

$$Md = \begin{cases} \left(X_{[N/2]} + X_{[N/2+1]}\right)/2 & \text{if } N \text{ is even} \\ X_{[(N+1)/2]} & \text{otherwise} \end{cases}$$

where $X_{[N]}$ is the largest value and $X_{[1]}$ the smallest. The number of cases in each of the two groups which exceed the median are counted. These will be denoted as g_1 and g_2, and the corresponding sample sizes as n_1 and n_2.

Test Statistic and Significance Level

- If $N \leq 30$, the significance level is from Fisher's exact test. (See Appendix 5.)
- If $N > 30$, the test statistic is

$$\chi_c^2 = \frac{\left[\left|g_1(n_2-g_2) - g_2(n_1-g_1)\right| - N/2\right]^2 N}{(g_1+g_2)(n_1+n_2-g_1-g_2)n_1n_2}$$

which is distributed as a χ^2 with 1 degree of freedom.

Mann-Whitney U Test

Calculation of Sums of Ranks

The combined data from both groups are sorted and ranks assigned to all cases, with average rank being used in the case of ties. The sum of ranks for each of the groups (S_1 and S_2) is calculated, as well as, for tied observations, $T_i = \frac{t^3-t}{12}$, where t is the number of observations tied for rank i. The average rank for each group is

$$\overline{S}_i = S_i/n_i$$

where n_i is the sample size in group i.

Test Statistic and Significance Level

The U statistic for group 1 is

$$U = n_1n_2 + \frac{n_1(n_1+1)}{2} - S_1$$

- If $U > n_1n_2/2$, the statistic used is

$$U^{'} = n_1n_2 - U$$

- If $n_1n_2 \le 400$ and $n_1n_2/2 + \min(n_1, n_2) \le 220$ the exact significance level is based on an algorithm of Dineen and Blakesley (1973).
- The test statistic corrected for ties is

$$Z = \frac{(U - n_1n_2/2)}{\sqrt{\frac{n_1n_2}{N(N-1)}\left(\frac{N^3-N}{12} - \sum_i T_i\right)}}$$

which is distributed approximately as a standard normal. A two-tailed significance level is printed.

Wilcoxon Rank Sum W Statistic

If $U > n_1n_2/2$, then W=S_1; otherwise W=S_2.

Kolmogorov-Smirnov Two-Sample Test

Calculation of the Empirical Cumulative Distribution Functions and Differences

For each of the two groups separately the data sorted into ascending order, from $X_{[1]}$ to $X_{[n_i]}$, and the empirical cdf for group i is computed as

$$\hat{F}_i(X) = \begin{cases} 0 & -\infty < X < X_{[1]} \\ j/n_i & X_{[j]} \le X < X_{[j+1]} \\ 1 & X_{[n_1]} \le X < \infty \end{cases}$$

For all of the X_j values in the two groups, the difference between the two groups is

$$D_j = \hat{F}_1(X_j) - \hat{F}_2(X_j)$$

where $\hat{F}_1(X_j)$ is the cdf for the group with the larger sample size. The maximum positive, negative, and absolute differences are also computed.

Test Statistic and Level of Significance

The test statistic (Smirnov, 1948) is

$$Z = \max_j |D_j| \sqrt{\frac{n_1 n_2}{n_1+n_2}}_j$$

and the significance level is calculated using the Smirnov approximation described in the K-S one sample test.

Wald-Wolfowitz Runs Test

Calculation of Number of Runs

All observations from the two samples are pooled and sorted into ascending order. The number of changes in the group numbers corresponding to the ordered data are counted. The number of runs (*R*) is the number of group changes plus one.

If there are ties involving observations from the two groups, both the minimum and maximum number of runs possible are calculated.

Significance Level

If $n_1 + n_2$, the total sample size, is less than or equal to 30, the one-sided significance level is exactly calculated from

$$P(r \le R) = \frac{2}{\binom{n_1+n_2}{n_1}} \sum_{r=2}^{R} \binom{n_1-1}{r/2-1}\binom{n_2-1}{r/2-1}$$

when *R* is even. When *R* is odd

$$P(r \le R) = \frac{1}{\binom{n_1+n_2}{n_1}} \sum_{r=2}^{R} \left[\binom{n_1-1}{k-1}\binom{n_2-1}{k-2} + \binom{n_1-1}{k-2}\binom{n_2-1}{k-1}\right]$$

where

$r = 2k - 1.$

For sample sizes greater than 30, the normal approximation is used (see “Runs Test”).

Moses Test of Extreme Reaction

Span Computation

Observation from both groups are jointly sorted and ranked, with the average rank being assigned in the case of ties. The ranks corresponding to the smallest and largest control group (first group) members are determined, and the span is computed as

SPAN = Rank(Largest Control Value) – Rank(Smallest Control Value) + 1

rounded to the nearest integer.

Significance Level

The exact one-tailed probability level is computed from

$$P(\text{SPAN} \leq\ n_c - 2h + g) = \frac{\sum_{i=0}^{g}\left[\binom{i+n_c-2h-2}{i}\binom{n_e+2h+1-i}{n_e-i}\right]}{\binom{n_c+n_e}{n_c}}$$

where $h = 0$, n_c is the number of cases in the control group, and n_e is the number of cases in the experimental group. The same formula is used in the next section where *h* is not zero.

Censoring of Range

The previous test is repeated, dropping the *h* lowest and *h* highest ranks from the control group. If not specified by the user, *h* is taken to be the integer part of $0.05n_c$ or 1, whichever is greater. If *h* is user specified, the integer value is used unless it is less than one. The significance level is determined as in the previous section.

K-S ample Median Test

Table Construction

If the median value is not specified by the user, the combined data from all groups are sorted and the median is calculated.

$$\text{Md} = \begin{cases} \left(X_{[N/2]} + X_{[N/2+1]}\right)/2 & \text{if} N \text{is even} \\ X_{[(N+1)/2]} & \text{if} N \text{is odd} \end{cases}$$

where $X_{[N]}$ is the largest value and $X_{[1]}$ the smallest.

The number of cases in each of the groups that exceed the median are counted and the following table is formed.

	Group 1	**Group 2**	**Group 3**	**...**	**Group *k***	
LE Md	O_{11}	O_{12}	O_{13}	...	O_{1k}	R_1
GT Md	O_{21}	O_{22}	O_{23}	...	O_{2k}	R_2
	n_1	n_2	n_3	...	n_k	N

Test Statistic and Level of Significance

The χ^2 statistic for all nonempty groups is calculated as

$$\chi^2 = \sum_{j=1}^{k} \sum_{i=1}^{2} (O_{ij} - E_{ij})^2 / E_{ij}$$

where

$E_{ij} = \frac{R_i n_j}{N}$.

The significance level is from the χ^2 distribution with $k-1$ degrees of freedom, where k is the number of nonempty groups. A message is printed if any cell has an expected value less than one, or more than 20% of the cells have expected values less than five.

Kruskal-Wallis One-Way Analysis of Variance

Computation of Sums of Ranks

Observations from all k nonempty groups are jointly sorted and ranked, with the average rank being assigned in the case of ties. The number of tied scores in a set of ties, t_i, is also found, and the sum of $T_i = t_i^3 - t_i$ is accumulated. For each group the sum of ranks, R_i, as well as the number of observations, n_i, is obtained.

Test Statistic and Level of Significance

The test statistic unadjusted for ties is

$$H = \frac{12}{N(N+1)} \sum_{i=1}^{k} R_i^2 / n_i - 3(N+1)$$

where N is the total number of observations.
Adjusted for ties, the statistic is

$$H' = \frac{H}{1 - \sum_{i=1}^{m} T_i / (N^3 - N)}$$

where m is the total number of tied sets.

The significance level is based on the χ^2 distribution, with $k-1$ degrees of freedom.

References

Dallal, G. E., and L. Wilkinson. 1986. An analytic approximation to the distribution of Lilliefor's test statistic for normality. *The American Statistician*, 40(4): 294–296 (Correction: 41: 248), – .

Dineen, L. C., and B. C. Blakesley. 1973. Algorithm AS 62: Generator for the sampling distribution of the Mann-Whitney U statistic. *Applied Statistics*, 22, 269–273.

Lehmann, E. L. 1985. *Nonparametrics: Statistical Methods Based on Ranks*. San Francisco: McGraw-Hill.

Lilliefors, H. W. 1967. On the Kolmogorov-Smirnov tests for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62, 399–402.

Siegel, S. 1956. *Nonparametric statistics for the behavioral sciences*. New York: McGraw-Hill.

Smirnov, N. V. 1948. Table for estimating the goodness of fit of empirical distributions. *Annals of the Mathematical Statistics*, 19, 279–281.

ONEWAY Algorithms

For post hoc range tests and pairwise multiple comparisons, see *Post Hoc Tests*.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 71-1
Notation

Notation	Description
X_{lj}	Value of the *j*th observation in group *l*
w_{lj}	Weight for the *j*th observation in group *l*
$W_{l,j}$	Sum of weights of the first *j* cases in group *l*
W_l	Sum of weights of all cases in group *l*
k_{i}	Number of groups, determined as maximum group values minus minimum plus one
k'	Number of nonempty groups
n_l	Number of cases in group *l*
W	Sum of weights of cases in all groups

Group Statistics

The following group statistics are available.

Computation of Group Statistics

A weighted version of the Young-Cramer (1971) algorithm is used to compute recursively the corrected sum of squares for each group.

$$SSQ_{l,i} = SSQ_{l,i-1} + \frac{w_{li}\left(X_{li}W_{l,i-1} - \sum_{j=1}^{i-1} w_{lj}X_{lj}\right)^2}{W_{l,i-1}W_{li}}$$

The initial value is 0; the value for each group after the last observation has been processed is the corrected sum of squares.

$$SS_l = SSQ_{l,n_l}$$

The sum and mean for each group are

$$T_l = \sum_{i=1}^{n_l} X_{li}w_{li}$$

$$\overline{T}_l = T_l / W_l$$

The variance is

$$S_l^2 = SS_l / (W_l - 1)$$

The grand sum is

$$G = \sum_{i=1}^{k} T_i$$

Group Statistics from Summary Statistics

With matrix data input, the user supplies sum of weights in each group (W_l), means $(\overline{T}_l)$, and standard deviations (S_l). From these,

$$T_l = W_l \overline{T}_l$$

$$SS_l = (W_l - 1) S_l^2$$

$$G = \sum_{i=1}^{k} T_i$$

If the user supplies the pooled variance S_p^2 and its degrees of freedom (D) instead of the individual S_l, and $D < 1$, the program will reset it to

$$D = \sum_{l=1}^{k} W_l - k^{'}$$

The within-group sum of squares is

$$WSS = S_p^2 D$$

The ANOVA Table

Table 71-2
ANOVA table

Source of Variation	SS	df
Between (BSS)	$\sum_{l=1}^{k} T_l^2 / n_l - G^2 / W$	$k^{'} - 1$

Source of Variation	SS	df
Within (WSS)	$\sum_{l=1}^{k} SS_l$ ($s_p^2 D$ for matrix input)	$W - k'$ (D)
Total (TSS)	$BSS + WSS$	$W - 1$

Mean squares are calculated by dividing each sum of squares by its degree of freedom. The *F* ratio for testing equality of group means is

$$F = \frac{\text{Mean Square Between}}{\text{Mean Square Within}} = \frac{BSSM}{WSSM}$$

The significance level is obtained from the *F* distribution with numerator and denominator degrees of freedom.

Basic Statistics

The following basic statistics are available.

Descriptive Statistics

$$\text{Sample size} = W_q$$

$$\text{Mean} = \overline{T}_q$$

$$\text{Standard deviation} = S_q$$

$$\text{Standard error} = S_q / \sqrt{W_q}$$

95% Confidence Interval for the Mean

$$\overline{T}_q \pm t_{W_q-1} S_q / \sqrt{W_q}$$

where t_{W_q-1} is the upper 2.5% critical value for the *t* distribution with $W_q - 1$ degrees of freedom.

Variance Estimates and Confidence Interval for Mean

Computation depends upon whether a fixed-effects or random-effects model is fit.

Fixed-Effects Model

Fixed-effects factors are generally thought of as variables whose values of interest are all represented in the data file.

Pooled Standard Deviation

$$S_p = \sqrt{WSSM}$$

Standard Error

$$\text{Standard error} = \sqrt{WSSM/W}$$

95% Confidence Interval for the Mean

$$\overline{G} \pm t_{W-k'} \sqrt{WSSM/W}$$

where $t_{W-k'}$ is the upper 2.5% critical value for the *t* distribution with $W - k'$ degrees of freedom.

Random-Effects Model

Random-effects factors are variables whose values in the data file can be considered a random sample from a larger population of values. They are useful for explaining excess variability in the dependent variable.

Between-Groups Component of Variance (Snedecor and Cochran 1967)

$$\omega^2 = \frac{(BSSM - WSSM)\big(W(k' - 1)\big)}{\left(W^2 - \sum_{i=1}^{k} W_i^2\right)}$$

Standard Error of the Mean (Brownlee 1965)

$$V(\overline{G}) = \frac{\left(\sum_{i=1}^{k} W_i^2\right)(k' - 1)(BSSM - WSSM)}{W\left(W^2 - \sum_{i=1}^{k} W_i^2\right)} + \frac{WSSM}{W}$$

If $BSSM < WSSM$, $V(\overline{G}) = \frac{WSSM}{W}$ and a warning is printed that the variance component estimate is negative.

95% Confidence Interval for the Mean

$$\overline{G} \pm t_{k'-1}\sqrt{V(\overline{G})}$$

where $t_{k'-1}$ is the upper 2.5% critical value for the *t* distribution with $k' - 1$ degrees of freedom

Levene Test for Homogeneity of Variances

$$L = \frac{(W - k')\sum_{i=1}^{k'} W_i(\overline{Z}_i - \overline{Z})^2}{(k' - 1)\sum_{i=1}^{k'}\sum_{l=1}^{n_i} w_{il}(Z_{il} - \overline{Z}_i)^2}$$

where

$$Z_{il} = \left|X_{il} - \overline{T}_i\right|$$

$$\overline{Z}_i = \frac{\sum_{l=1}^{n_i} w_{il}Z_{il}}{W_i}$$

$$\overline{Z} = \frac{\sum_{i=1}^{k'} W_i\overline{Z}_i}{W}$$

User-Supplied Contrasts

Let C_1 through C_k be the coefficients for a particular contrast. If the sum of the coefficients is not 0, a warning is printed and the contrast number is starred. For each contrast the following are printed.

Value of the Contrast

$$V = \sum_{i=1}^{k} \overline{T}_i C_i$$

Pooled Variance Statistics

The following statistics are computed.

Standard Error

$$SE = \sqrt{S_p^2 \sum_{i=1}^{k} C_i^2 / W_i}$$

t Value

$$t = V/SE$$

Degrees of Freedom

$$W - k'$$

And a two-tailed significance level based on the *t* distribution with *W* – *k*' degrees of freedom.

Separate Variance Statistics

The following statistics are computed.

Standard Error

$$SE = \sqrt{\sum_{i=1}^{k} C_i^2 \left(S_i^2 / W_i\right)}$$

t Value

$$t = V/SE$$

Degrees of Freedom (Brownlee 1965)

$$df = \frac{\left(\sum_{i=1}^{k} C_i^2 S_i^2 / W_i\right)^2}{\sum_{i=1}^{k} \left(C_i^2 S_i^2 / W_i\right)^2 / (W_i - 1)}$$

And a two-tailed significance level based on the t distribution with df degrees of freedom

Polynomial Contrasts (Speed 1976)

If the specified degree of the polynomial (*NP*) is less than or equal to 0, or greater than 5, a message is printed and the procedure is terminated. If the degree of the polynomial specified is greater than the number of nonempty groups, it is set to $k' - 1$. If the sums of the weights in each group are equal, only the WEIGHTED contrasts will be generated. For unequal sample sizes with equal spacing between groups, both WEIGHTED and UNWEIGHTED contrasts are computed. For unequal sample sizes and unequal spacing, only WEIGHTED contrasts are computed. The metric for the polynomial is the group code.

UNWEIGHTED Contrasts and Statistics

The coefficients for the orthogonal polynomial are calculated recursively from the following relations:

$$c_{i,q} = (i - A_q)c_{i,q-1} - C_q c_{i,q-2}$$

for

$$q = 1, 2, \ldots, NP$$

$$i = 1, 2, \ldots, k$$

with the initial values

$$c_{i,-1} = 0, \quad c_{i,0} = 1$$

and

$$A_q = \frac{\sum_{i=1}^{k} i c_{i,q-1}^2}{\sum_{i=1}^{k} c_{i,q-1}^2}$$

$$C_q = \frac{\sum_{i=1}^{k} c_{i,q-1}^2}{\sum_{i=1}^{k} c_{i,q-2}^2} \quad \text{for } q \geq 2$$

$$C_q = 0 \qquad \text{for } q = 1$$

The *F* statistic for the *q*th degree contrast is computed as

$$F = \frac{\left[\sum_{i=1}^{k} (\overline{T}_i - \overline{G}) c_{i,q}\right]^2}{\sum_{i=1}^{k} c_{i,q}^2 / W_i} / WSSM$$

where *WSSM* is the mean square within. The significance level is obtained from the *F* distribution with 1 and $W - k^{'}$ degrees of freedom.

WEIGHTED Contrasts and Statistics (Emerson 1968; Robson 1959)

The contrast for the *q*th degree polynomial component is computed from the following recursive relations:

$$d_{i,q} = \left(i - A^{'}{}_q\right) d_{i,q-1} - C^{'}{}_q d_{i,q-2}$$

for

$$q = 1, 2, \ldots, NP.$$
$$i = 1, 2, \cdots, k.$$

with initial values

$$d_{i,0} = 1, d_{i,-1} = 0$$

$$A^{'}{}_q = \frac{\sum_{i=1}^{k} i W_i d_{i,q-1}^2}{\sum_{i=1}^{k} W_i d_{i,q-1}^2}$$

$$C^{'}{}_q = \frac{\sum_{i=1}^{k} i W_i d_{i,q-1} d_{i,q-2}}{\sum_{i=1}^{k} W_i d_{i,q-2}^2} \qquad \text{for } q \geq 2$$

$$C^{'}{}_q = 0 \qquad \text{for } q = 1$$

The test for the contribution of the *q*th degree orthogonal polynomial component is based on

$$F = D_q / WSSM$$

where

$$D_q = \frac{\left(\sum_{i=1}^{k} W_i \overline{T}_i d_{i,q}\right)^2}{\sum_{i=1}^{k} W_i d_{i,q}^2}$$

The significance level is computed from the *F* distribution with degrees of freedom 1 and $W - k^{'}$.

The test for deviation from the *q*th degree polynomial is based on

$$F = DD_q / WSSM$$

where

$$DD_q = \left(BSS - \sum_{j=1}^{q} D_j\right) / \left(k^{'} - q - 1\right)$$

The significance level is computed from the *F* distribution with degrees of freedom $k^{'} - q - 1$ and $W - k^{'}$. The highest degree printed will be the minimum of $\left(k^{'} - 2\right)$ and 5.

Multiple Comparisons (Winer 1971)

Generation of Ranges

The Student-Newman-Keuls (SNK), TUKEY, and TUKEYB procedures are all based on the studentized range, $S_{r,f}$, where *r* is the number of steps between means and *f* is the degrees of freedom for the within-groups mean square. For the above tests, only $\alpha = 0.05$ can be used.

The appropriate range of values for the tests are

SNK. $R_r = S_{r,f},\ r = 2, \ldots, k^{'}$

TUKEY. $R_r = S_{k^{'},f}$

TUKEYB. $R_r = \frac{\left(S_{r,f} + S_{k^{'},f}\right)}{2}$

For the DUNCAN procedure, alphas of 0.01, 0.05, and 0.10 can be used. The ranges $(D_{r,f})$ are generated using the algorithm of Gebhardt (1966).

DUNCAN. $R_r = D_{r,f},\ r = 2, \ldots, k^{'}$

The Scheffé, LSD, and modified LSD procedures all use critical points from the *F* distribution. Any $\alpha \leq 0.5$ can be used.

SCHEFFE. $R_r = \sqrt{2(k'-1)F_{1-\alpha}(k'-1,f)}$

LSD. $R_r = \sqrt{2F_{1-\alpha}(1,f)}$

MODLSD. $R_r = \sqrt{2F_{1-\alpha'}(1,f)}$

where

$$\alpha' = \frac{2\alpha}{k'(k'-1)}$$

Compute the multiplier of the ranges for the difference of means *i* and *j*.

$$M_{i,j} = S_p\sqrt{\frac{1}{2}\left(\frac{1}{n_i}+\frac{1}{n_j}\right)} \quad \text{(default)}$$

$$M_{i,j} = S_p\sqrt{\frac{\sum_{l=1}^{k} 1/n_l}{k'}} \quad \text{(harmonic mean for all groups)}$$

Establishment of Homogeneous Subsets

If the sample sizes in all groups are equal, or the harmonic mean for all groups has been selected, or the multiple comparison procedure is SNK or DUNCAN, homogeneous subsets are established as follows:

The means are sorted into ascending order from $\overline{T}_{(1)}$ to $\overline{T}_{(k')}$. Values of *i* and *q* such that

$$\left|\overline{T}_{(q)} - \overline{T}_{(i)}\right| \leq R_{q-i+1}M_{q,i} \quad (*)$$

are systematically searched for and

$$\left\{\overline{T}_{(i)}, \ldots, \overline{T}_{(q)}\right\}$$

is considered a homogeneous subset. The search procedure is as follows:

At each step *t*, the value of *i* is incremented by 1 (the starting value is 1), and $q = k'$. The value of *q* is then decremented by one until $(*)$ is true. Call this value q_t. If $q_t > q_{t-1}$ and $(*)$ is true,

$$\left\{\overline{T}_{(i)}, \ldots, \overline{T}_{q_t}\right\}$$

is considered homogeneous. Otherwise *i* is incremented and the next step is done. The procedure terminates when $i = k$ or $q_t = k$.

In all other situations, all nonredundant pairs of groups are compared using the criteria of $(*)$. A table containing all pairs of groups is printed with symbols indicating group means that are significantly different.

Welch Test

In Welch (1947,1951), he derived the an approximate test for equality of means without the homogeneous variance assumption. The statistic is given by

$$F_{Welch} = \frac{\sum_{l=1}^{k} \omega_l \left[\left(\overline{T}_l - \tilde{X} \right)^2 / (k-1) \right]}{1 + \frac{2(k-2)}{(k^2-1)} \sum_{l=1}^{k} \left[\left(1 - \frac{\omega_l}{u} \right)^2 / (W_l - 1) \right]}$$

where $\omega_l = W_l / S_l^2$, $u = \sum_{l=1}^{k} \omega_l$, and $\tilde{X} = \sum_{l=1}^{k} \omega_l \overline{T}_l / u$.

The Welch statistic has an approximate F distribution with k-1 and f degrees of freedom, where

$$f = \left[\frac{3}{k^2 - 1} \sum_{l=1}^{k} \left(1 - \frac{\omega_l}{u} \right)^2 / (W_l - 1) \right]^{-1}$$

Since the weight used in Welch statistic is $\omega_l = W_l / S_l^2$, one cannot compute the statistic if any one group has zero standard deviation. Moreover, sample sizes of all groups have to be greater than or equal to zero.

Brown-Forsythe Test

In (Brown and Forsythe, 1974a) and (Brown and Forsythe, 1974b), a test statistic for equal means was proposed. The statistic has the following form,

$$F_{BF} = \frac{\sum_{l=1}^{k} W_l \left(\overline{T}_l - \overline{G} \right)^2}{\sum_{l=1}^{k} (1 - W_l / W) S_l^2}$$

The statistic has an approximate F distribution with (k-1) and f degrees of freedom, where

$$\frac{1}{f} = \sum_{l=1}^{k} c_l^2 / (W_l - 1)$$

and

$$c_l = \frac{(1 - W_l/W)\, S_l^2}{\sum_{l=1}^{k} (1 - W_l/W)\, S_l^2}$$

When we look at the denominator of F_{BF}, we can see that it tries to estimate the 'pooled variance' by

$$S_{pool}^2 = \sum_{l=1}^{k} \omega_l^* S_l^2$$

where

$$\omega_l^* = \frac{(W - W_l)}{W\,(k-1)}$$

The Brown & Forsythe statistic cannot be computed if all groups have zero standard deviation or any group has sample size less than or equal to 1. In the situation that some groups have zero standard deviations, the statistic can be computed but the approximation may not work.

References

Brown, M. B., and A. B. Forsythe. 1974a. The small sample behavior of some statistics which test the equality of several means. *Technometrics*, 16, 129–132.

Brown, M. B., and A. B. Forsythe. 1974b. Robust tests for the equality of variances. *Journal of the American Statistical Association*, 69, 364–367.

Brownlee, K. A. 1965. *Statistical theory and methodology in science and engineering*. New York: John Wiley & Sons, Inc.

Duncan, D. B. 1955. Multiple Range and Multiple F tests. *Biometrics*, 11, 1–42.

Eisenhart, C., M. W. Hastay, and N. A. Wallis, eds. 1947. *Significance of the largest of a set of sample estimates of variance. In: Techniques of Statistical Analysis.* New York: McGraw-Hill.

Emerson, P. L. 1968. Numerical construction of orthogonal polynomials from a general recurrence formula. *Biometrics*, 24, 695–701.

Gebhardt, F. 1966. Approximation to the Critical Values for Duncan's Multiple Range Test. *Biometrics*, 22, 179–182.

James, G. S. 1951. The comparison of several groups of observations when the ratios of the population variances are unknown. *Biometrika*, 38, 324–329.

Kramer, C. Y. 1956. Extension of multiple range tests to group means with unequal numbers of replications. *Biometrics*, 12, 307–310.

Miller, R. G. J. 1966. *Simultaneous statistical inference*. New York: McGraw-Hill.

Robson, D. S. 1959. A simple method for construction of orthogonal polynomials when the independent variable is unequally spaced. *Biometrics*, 15, 187–191.

Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2, 110–114.

Snedecor, G. W., and W. G. Cochran. 1967. *Statistical methods.* Ames, Iowa: Iowa State University Press.

Speed, M. F. 1976. Response curves in the one way classification with unequal numbers of observations per cell. In: *Proceedings of the Statistical Computing Section,* Alexandria, VA: AmericanStatistical Association, 270–272.

Welch, B. L. 1947. The generalization of Student's problem when several different population variances are involved. *Biometrika*, 34, 28–35.

Welch, B. L. 1951. On the Comparison of Several Mean Values: An Alternative Approach. *Biometrika*, 38, 330–336.

Winer, B. J. 1971. *Statistical principles in experimental design*, 2nd ed. New York: McGraw-Hill.

Young, E. A., and E. W. Cramer. 1971. Some results relevant to choice of sum and sum-of-product algorithms. *Technometrics*, 13, 657–665.

OPTIMAL BINNING Algorithms

The Optimal Binning procedure performs MDLP (minimal description length principle) discretization of scale variables. This method divides a scale variable into a small number of intervals, or bins, where each bin is mapped to a separate category of the discretized variable.

MDLP is a univariate, supervised discretization method. Without loss of generality, the algorithm described in this document only considers one continuous attribute in relation to a categorical guide variable — the discretization is “optimal” with respect to the categorical guide. Therefore, the input data matrix *S* contains two columns, the scale variable *A* and categorical guide *C*.

Notation

The following notation is used throughout this chapter unless otherwise stated:

S	The input data matrix, containing a column of the scale variable A and a column of the categorical guide C. Each row is a separate observation, or instance.
A	A scale variable, also called a continuous attribute.
$S(i)$	The value of A for the ith instance in S.
N	The number of instances in S.
D	A set of all distinct values in S.
S_i	A subset of S.
C	The categorical guide, or class attribute; it is assumed to have k categories, or classes.
T	A cut point that defines the boundary between two bins.
T_A	A set of cut points.
Ent(S)	The class entropy of S.
E(A, T, S)	The class entropy of partition induced by T on A.
Gain(A, T, S)	The information gain of the cut point T on A.
n	A parameter denoting the number of cut points for the equal frequency method.
W	A weight attribute denoting the frequency of each instance. If the weight values are not integer, they are rounded to the nearest whole numbers before use. For example, 0.5 is rounded to 1, and 2.4 is rounded to 2. Instances with missing weights or weights less than 0.5 are not used.

Simple MDLP

This section describes the supervised binning method (MDLP) discussed in Fayyad and Irani (1993).

Class Entropy

Let there be k classes C_1, ..., C_k and let $P(C_i, S)$ be the proportion of instances in S that have class C_i. The class entropy Ent(S) is defined as

$$Ent\left(S\right) = -\sum_{i=1}^{k} P\left(C_i, S\right) \log_2\left(P\left(C_i, S\right)\right)$$

Class Information Entropy

For an instance set S, a continuous attribute A, and a cut point T, let $S_1 \subset S$ be the subset of instances in S with the values of $A \leq T$, and $S_2 = S - S_1$. The class information entropy of the partition induced by T, E(A, T; S), is defined as

$$E\left(A, T; S\right) = \frac{|S_1|}{|S|} Ent\left(S_1\right) + \frac{|S_2|}{|S|} Ent\left(S_2\right)$$

Information Gain

Given a set of instances S, a continuous attribute A, and a cut point T on A, the information gain of a cut point T is

$$Gain\left(A, T; S\right) = Ent\left(S\right) - E\left(A, T; S\right)$$

MDLP Acceptance Criterion

The partition induced by a cut point T for a set S of N instances is accepted if and only if

$$Gain\left(A, T; S\right) > \frac{\log_2\left(N-1\right)}{N} + \frac{\Delta\left(A, T; S\right)}{N}$$

and it is rejected otherwise.

Here $\Delta\left(A, T; S\right) = \log_2\left(3^k - 2\right) - \left[k \cdot Ent\left(S\right) - k_1 Ent\left(S_1\right) - k_2 Ent\left(S_2\right)\right]$ in which k_i is the number of classes in the subset S_i of S.

Note: While the MDLP acceptance criterion uses the association between A and C to determine cut points, it also tries to keep the creation of bins to a small number. Thus there are situations in which a high association between A and C will result in no cut points. For example, consider the following data:

D	*Class*	
	2	3
1	1	0
2	0	6

Then the potential cut point is $T = 1$. In this case:

$$Gain\left(A, T; S\right) = 0.5916728$$

$$\frac{\log_2(N-1)}{N} + \frac{\Delta(A,T;S)}{N} = 0.6530774$$

Since 0.5916728 < 0.6530774, T is not accepted as a cut point, even though there is a clear relationship between A and C.

Algorithm: BinaryDiscretization

1. Calculate E(A, d_i; S) for each distinct value $d_i \in D$ for which d_i and d_{i+1} do not belong to the same class. A distinct value belongs to a class if all instances of this value have the same class.
2. Select a cut point T for which E(A, T; S) is minimum among all the candidate cut points, that is,

   $T = arg\min_{d_i} E(A, d_i; S)$

Algorithm: MDLPCut

```
1. BinaryDiscretization(A, T; D, S).
2. Calculate Gain(A, T; S).
3. If $Gain(A,T;S) > \frac{\log_2(N-1)}{N} + \frac{\Delta(A,T;S)}{N}$ then
   a) $T_A = T_A \cup T$
   b) Split D into $D_1$ and $D_2$, and S into $S_1$ and $S_2$.
   c) MDLPCut(A, $T_A$; $D_1$, $S_1$).
   d) MDLPCut(A, $T_A$; $D_2$, $S_2$). where $S_1 \subset S$ be the subset of instances in S with A-values $\le T$, and $S_2 = S - S_1$. $D_1$ and $D_2$ are the sets of all distinct values in $S_1$ and $S_2$, respectively.
```

Also presented is the iterative version of MDLPCut(A, T_A; D, S). The iterative implementation requires a stack to store the D and S remaining to be cut.

First push D and S into *stack*. Then, while ($stack \neq \emptyset$) do

```
1. Obtain D and S by popping stack.
2. BinaryDiscretization(A, T; D, S).
3. Calculate Gain(A, T; S).
4. If $Gain(A,T;S) > \frac{\log_2(N-1)}{N} + \frac{\Delta(A,T;S)}{N}$ then
   i) $T_A = T_A \cup T$
   ii) Split D into $D_1$ and $D_2$, and S into $S_1$ and $S_2$.
   iii) Push $D_1$ and $S_1$ into stack.
   iv) Push $D_2$ and $S_2$ into stack.
```

Note: In practice, all operations within the algorithm are based on a global matrix M. Its element, m_{ij}, denotes the total number of instances that have value $d_i \in D$ and belong to the jth class in S. In addition, D is sorted in ascending order. Therefore, we do not need to push D and S into *stack*, but only two integer numbers, which denote the bounds of D, into *stack*.

Algorithm: SimpleMDLP

1. Sort the set S with N instances by the value A in ascending order.
2. Find a set of all distinct values, D, in S.
3. $T_A = \varnothing$.
4. MDLPCut(A, T_A; D, S)
5. Sort the set T_A in ascending order, and output T_A.

Hybrid MDLP

When the set D of distinct values in S is large, the computational cost to calculate E(A, d_i; S) for each $d_i \in D$ is large. In order to reduce the computational cost, the unsupervised equal frequency binning method is used to reduce the size of D and obtain a subset $D_{ef} \in D$. Then the MDLPCut(A, T_A; D_s, S) algorithm is applied to obtain the final cut point set T_A.

Algorithm: EqualFrequency

It divides a continuous attribute A into n bins where each bin contains N/n instances. n is a user-specified parameter, where $1 < n < N$.

1. Sort the set S with N instances by the value A in ascending order.
2. $D_{ef} = \varnothing$
3. j=1.
4. Use the aempirical percentile method to generate the $d_{p,i}$ which denote the $\left(\frac{i \cdot N}{n} \times 100\right)$th percentiles.
5. $D_{ef} = D_{ef} \cup d_{p,i}$; i=i+1
6. If $i \le n$, then go to step 4.
7. Delete the duplicate values in the set D_{ef}.

Note: If, for example, there are many occurrences of a single value of A, the equal frequency criterion may not be met. In this case, no cut points are produced.

Algorithm: HybridMDLP

1. $D = \varnothing$;

2. EqualFrequency(A, n, D; S).
3. $T_A = \varnothing$.
4. MDLPCut(A, T_A; D, S).
5. Output T_A.

Model Entropy

The model entropy is a measure of the predictive accuracy of an attribute A binned on the class variable C. Given a set of instances S, suppose that A is discretized into I bins given C, where the ith bin has the value A_i. Letting $S_i \subset S$ be the subset of instances in S with the value A_i, the model entropy is defined as:

$$E_m = \sum_{i=1}^{I} P(A_i) \left(- \sum_{j=1}^{J} P(C_j|A_i) \log_2 P(C_j|A_i) \right)$$

where $P(A_i) = \frac{|S_i|}{|S|}$ and $P(C_j|A_i) = \frac{P(C_j, A_i)}{P(A_i)} = P(C_j, S_i)$.

Merging Sparsely Populated Bins

Occasionally, the procedure may produce bins with very few cases. The following strategy deletes these pseudo cut points:

► For a given variable, suppose that the algorithm found n_{final} cut points, and thus n_{final}+1 bins. For bins i = 2, ..., n_{final} (the second lowest-valued bin through the second highest-valued bin), compute

$$\frac{sizeof(b_i)}{\min(sizeof(b_{i-1}), sizeof(b_{i+1}))}$$

where sizeof(bin) is the number of cases in the bin.

► When this value is less than a user-specified merging threshold, b_i is considered sparsely populated and is merged with b_{i-1} or b_{i+1}, whichever has the lower class information entropy. For more information, see the topic “Class Information Entropy”.

The procedure makes a single pass through the bins.

Example

The following example shows the process of simple MDLP using an artificial data set S with 250 instances. S is not shown here, but can be reconstructed (sorted in ascending order of values of A) from the matrix M below.

First, sort S by the value of A in ascending order. Then find a set, D, of all distinct values in S.

$|D| = 46$.

D = {-2.6, -2.4, -2.1, -2, -1.9, -1.8, -1.7, -1.6, -1.5, -1.4, -1.3, -1.2, -1.1, -1, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1, 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, 2.1, 2.3}

Compute the frequencies of instances with respect to each class for each distinct value $d_i \in D$ and construct a matrix M. Its element, m_{ij}, denotes the total number of instances that have value d_i and belong to the jth class.

Table 72-1
2-Dimensional matrix M

D	*Class*				*D*	*Class*			
	0	1	2	3		0	1	2	3
-2.6	0	1	0	0	0	0	0	0	14
-2.4	0	1	0	0	0.1	0	0	0	6
-2.1	0	2	0	0	0.2	0	0	0	10
-2	0	0	2	0	0.3	0	0	0	14
-1.9	0	0	2	0	0.4	0	0	0	12
-1.8	0	0	3	0	0.5	0	0	0	4
-1.7	0	0	2	0	0.6	0	0	0	9
-1.6	0	0	3	0	0.7	0	0	0	5
-1.5	0	0	3	0	0.8	0	0	0	3
-1.4	0	0	2	0	0.9	0	0	0	10
-1.3	0	0	4	0	1	3	0	0	0
-1.2	0	0	3	0	1.1	8	0	0	0
-1.1	0	0	3	0	1.2	5	0	0	0
-1	0	0	8	0	1.3	7	0	0	0
-0.9	0	0	6	0	1.4	2	0	0	0
-0.8	0	0	7	0	1.5	2	0	0	0
-0.7	0	0	13	0	1.6	3	0	0	0
-0.6	0	0	8	0	1.7	3	0	0	0
-0.5	0	0	4	0	1.8	4	0	0	0
-0.4	0	0	6	0	1.9	4	0	0	0
-0.3	0	0	7	0	2	2	0	0	0
-0.2	0	0	13	0	2.1	2	0	0	0
-0.1	0	0	14	0	2.3	1	0	0	0

MDLPCut(A, T_A; D, S)

Calculate E(A, d_i; S) for each $d_i \in D$ for which d_i and d_i+1 do not belong to the same class.

d_i	-2.1	-0.1	0.9
E(A, d_i; D, S)	1.4742	0.5955	0.9038
T_A = {-0.1}			

D_1 = {-2.6, -2.4, -2.1, -2, -1.9, -1.8, -1.7, -1.6, -1.5, -1.4, -1.3, -1.2, -1.1, -1, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1}

S_1 = {all instances with A-values $\leq$ -0.1}

D_2 = { 0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, 2.1, 2.3}

S_2 = {all instances with A-values > -0.1}

Calculate E(A, d_i; S_1) for each $d_i \in D_1$ for which d_i and d_i+1 do not belong to the same class.

d_i	-2.1
E(A, d_i; D_1, S_1)	0.0
T_A = {-0.1, -2.1}	

$D_{1,1}$ = {-2.6, -2.4, -2.1}

$S_{1,1}$ = {all instances with A-values between -2.6 and -2.1}

$D_{1,2}$ = { -2, -1.9, -1.8, -1.7, -1.6, -1.5, -1.4, -1.3, -1.2, -1.1, -1, -0.9, -0.8, -0.7, -0.6, -0.5, -0.4, -0.3, -0.2, -0.1}

$S_{1,2}$ = {all instances with A-values between -2 and -0.1}

All instances in $S_{1,1}$ belong to the same class, thus $S_{1,1}$ can't be split further.

All instances in $S_{1,2}$ belong to the same class, thus $S_{1,2}$ can't be split further.

Calculate E(A, d_i; S_2) for each $d_i \in D_2$ for which d_i and d_i+1 do not belong to the same class.

d_i	0.9
E(A, d_i; D_2, S_2)	0.0
T_A = {-0.1, -2.1, 0.9}	

$D_{2,1}$ = {0, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9}

$S_{2,1}$ = {all instances with A-values between 0 and 0.9}

$D_{2,2}$ = {1, 1.1, 1.2, 1.3, 1.4, 1.5, 1.6, 1.7, 1.8, 1.9, 2, 2.1, 2.3}

$S_{2,2}$ = {all instances with A-values between 1 and 2.3}

All instances in $S_{2,1}$ belong to the same class, thus $S_{2,1}$ can't be split further.

All instances in $S_{2,2}$ belong to the same class, thus $S_{2,2}$ can't be split further.

References

Fayyad, U., and K. Irani. 1993. Multi-interval discretization of continuous-value attributes for classification learning. In: *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence,* San Mateo, CA: Morgan Kaufmann, 1022–1027.

Dougherty, J., R. Kohavi, and M. Sahami. 1995. Supervised and unsupervised discretization of continuous features. In: *Proceedings of the Twelfth International Conference on Machine Learning,* Los Altos, CA: Morgan Kaufmann, 194–202.

Liu, H., F. Hussain, C. L. Tan, and M. Dash. 2002. Discretization: An Enabling Technique. *Data Mining and Knowledge Discovery*, 6, 393–423.

ORTHOPLAN Algorithms

This procedure generates an orthogonal main-effects design. It will find the smallest orthogonal plan to fit the factors having at least as many combinations as requested.

Selecting the Plan

From a library of prepared plans, select the shortest plan that can be adapted to the design and that satisfies the minimum size requirement provided by the user. If no plan exists that satisfies the minimum size requirement, pick the largest plan that can be adapted.

Adapting the Prepared Plans

Generating Multiple Factors from One Column

A four-level factor can be transformed into three two-level factors using the rule in the following table.

Table 73-1
Converting a four-level factor to three two-level factors

Original Code	**A**	**B**	**C**
0	0	0	0
1	0	1	1
2	1	0	1
3	1	1	0

An eight-level factor can be transformed into seven two-level factors using the rule in the following table.

Table 73-2
Converting an eight-level factor to seven two-level factors

Original Code	**A**	**B**	**C**	**D**	**E**	**F**	**G**
0	0	0	0	0	0	0	0
1	1	0	0	1	1	0	1
2	0	1	0	1	0	1	1
3	1	1	0	0	1	1	0
4	0	0	1	0	1	1	1
5	1	0	1	1	0	1	0
6	0	1	1	1	1	0	0
7	1	1	1	0	0	0	1

A nine-level factor can be transformed into four three-level factors using the rule in the following table.

Table 73-3
Converting a nine-level factor to four three-level factors

Original Code	**A**	**B**	**C**	**D**
0	0	0	0	0

Original Code	A	B	C	D
1	0	1	1	2
2	0	2	2	1
3	1	0	1	1
4	1	1	2	0
5	1	2	0	2
6	2	0	2	2
7	2	1	0	1
8	2	2	1	0

Changing the Number of Levels in a Column

Any factor of *m* levels can be transformed into a factor of $n<m$ levels by many-to-one mappings without changing its orthogonality. Any mapping can be used; *i* mod *n* is used here.

Library of Prepared Plans

This section describes previously developed plans.

Plackett-Burman Plans

Plackett and Burman (1946) describe a series of plans that can be generated from a single column by rotation. The general algorithm for generating any of these plans is:

- Let *L* be the number of levels for which the plan is designed. No factor in the specific design can have more than *L* levels.
- Let *N* be the number of rows (combinations) finally to be generated. Note that $N=F+1$ where *F* is defined below.
- Starting with a given column of $N-1$ level codes, rotate one position to generate each new column.
- Finally, add a row of zeroes.

$\frac{(N-1)}{(L-1)}$orthogonal columns can be generated in this fashion.

The Plackett-Burman plans used here are designated *PBL.F*, where *L* is the maximum number of levels and *F* is the number of factors:

Label	Generating Column
PB 2.7	11101 00
PB 2.11	11011 10001 0
PB 2.15	11110 10110 01000
PB 2.19	11001 11101 01000 0110
PB 2.23	11111 01011 0110 01010 000
PB 2.31	00001 01011 10110 00111 11001 10100 1
PB 2.35	01011 10001 11110 111 00 10000 10101 10010
PB 2.43	11001 01001 11011 11100 01011 10000 01000 11010 110

Label	Generating Column
PB 2.47	11111 01111 00101 01110 01001 10110 00101 01100 00100 00
PB 2.59	11011 10101 00100 11101 11100 11111 00000 11000 01000 11011 01010 0010
PB 3.4	01220 211
PB 3.13	00101 21120 11100 20212 21022 2
PB 3.40	01111 20121 12120 20221 10201 10012 22021 00200 02222 10212 21210 10112 20102 20021 11012 00100
PB 5.6	04112 10322 42014 43402 3313
PB 7.8	01262 21605 32335 20413 11430 65155 61024 54425 03646 634

Addelman Plans

Addelman (1961) described general methods for generating orthogonal main effects plans. That paper included a number of such designs, and using those methods, the authors generated more.

Table 73-4
18 rows, 7 columns of 3 levels each

0000000	0021011
0112111	0100122
0221222	0212200
1011120	1002221
1120201	1111002
1202012	1220110
2022102	2010212
2101210	2122020
2210021	2201101

Table 73-5
8 rows, 1 column of 4 levels plus 4 columns of 2 levels

0	0000
0	1111
1	0011
1	1100
2	0101
2	1010
3	0110
3	1001

Table 73-6
16 rows, 5 columns of 4 levels each

00000	02231
10111	12320
20222	22013
30333	32102
01123	03312
11032	13203

21301	23130
31210	33021

Table 73-7

32 rows, 9 columns of 4 levels each

000000000	002130213
011231111	013301302
022312222	020222031
033123333	031013120
101111032	103021221
110320123	112210330
123203210	121333003
132032301	130102112
202223102	200313311
213012013	211122200
220131320	222001133
231300231	233230022
303332130	301202323
312103021	310033232
321020312	323110101
330211203	332321010

Table 73-8

64 rows, 21 columns of 4 levels each

000000000000000000000	000222233331111022220
111111111111111100000	111333322220000122220
222222222222222200000	222000011113333222220
333333333333333300000	333111100002222322220
123012301230123012301	123230132101032030121
032103210321032112301	032321023010123130121
301230123012301212301	301012310323210230121
210321032103210312301	210103201232301330121
231023102310231023102	231201331021320001322
320132013201320123102	320310220130231101322
013201320132013223102	013023113203102201322
102310231023102323102	102132002312013301322
312031203120312031203	312213030211203013023
203120312031203131203	203302121300312113023
130213021302130231203	130031212033021213023
021302130213021331203	021120303122130313023
000111122223333011110	000333311112222033330
111000033332222111110	111222200003333133330
222333300001111211110	222111133330000233330
333222211110000311110	333000022221111333330
123103223013210003211	123321010322301021031
032012332102301103211	032230101233210121031

30132100123103220321 1	30110323210012322103 1
21023011032012330321 1	21001232301103232103 1
23113202013310203201 2	23131021320201301023 2
32002313102201313201 2	32020130231310211023 2
01331020231132023201 2	01313203102023121023 2
10220131320023133201 2	10202312013132031023 2
31212032130302102031 3	31230211203213000213 3
20303123021213012031 3	20321300312302110213 3
13030210312120322031 3	13012033021031220213 3
02121301203031232031 3	02103122130120330213 3

Table 73-9
16 rows, 1 column of 8 levels plus 8 columns of 2 levels

0	00000000	0	11111111
1	01010101	1	10101010
2	00001111	2	11110000
3	01011010	3	10100101
4	00111100	4	11000011
5	01101001	5	10010110
6	00110011	6	11001100
7	01100110	7	10011001

Table 73-10
31 rows, 1 column of 8 levels plus 8 columns of 4 levels

0	00000000	0	22222222
1	01230123	1	23012301
2	02021313	2	20203131
3	03211230	3	21033012
4	00113322	4	22331100
5	01323201	5	23101023
6	02132031	6	20310213
7	03302112	7	21120330
0	11111111	0	33333333
1	10321032	1	32103210
2	13130202	2	31312020
3	12300321	3	30122103
4	11002233	4	33220011
5	10232310	5	32010132
6	13023120	6	31201302
7	12213003	7	30031221

Table 73-11
64 rows, 9 columns of 8 levels each

000000000	202222222	404444444	606666666
011234567	213016745	415670123	617452301
022456713	220647531	426021357	624203175
033651274	231473056	437215630	635037412

044517326	246735104	440153762	642371540
055723641	257501463	451367205	653145927
066172435	264350617	462536071	660714253
077346152	275164370	473702516	671520734
101111111	303333333	505555555	707777777
110325476	312107654	514761032	7165432210
123574602	321756420	527130246	725312064
132140365	330562147	536304721	734126503
145406237	347624015	541042673	743260451
154632750	356410572	550276314	752054136
167063524	356241706	563427160	761605342
176257043	374075261	572613407	770481625

Table 73-12
27 rows, 1 column of 9 levels plus 9 columns of 3 levels

0	000000000	3	011001111	6	022002222
0	112121212	3	120122020	6	101120101
0	221212121	3	202210202	6	210211010
1	000111122	4	011112200	7	022110011
1	112202001	4	120200112	7	101201220
1	221020210	4	202021021	7	210022102
2	000222211	5	011220022	8	022221100
2	112010120	5	120011201	8	101012012
2	221101002	5	202102110	8	210100221

Table 73-13
81 rows, 10 columns of 9 levels each

0000000000	0336258147	0663174285
1011111111	1347036258	1674285063
2022222222	2358147036	2685063174
3033333333	3360582471	3606417528
4044444444	4371360582	4617528306
5055555555	5382471360	5628306417
6066666666	6303825714	6630741852
7077777777	7314603825	7641852630
8088888888	8325714603	8652630741
0112345678	0448561723	0775426831
1120453786	1456372804	1783507642
2101534867	2437480615	2764318750
3145678012	3472804156	3718750264
4153786120	4480615237	4726831075
5134867201	5461723048	5707642183
6178012345	6415237480	6742183507
7186120453	7423048561	7750264318
8167201534	8404156372	8731075426
0221687354	0557813462	0884732516

1202768435	1538624570	1865840327
2210876543	2546705381	2873651408
3254021687	3581246705	3827165840
4235102768	4562057813	4808273651
5243210876	5570138624	5816084732
6287354021	6524570138	6851408273
7268435102	7505381246	7832516084
8276543210	8513462057	8840327165

Decision Rules

Each value of *L* (the maximum number of levels in the design) has a distinct decision rule. In their descriptions, the following notation is used:

M	The user-supplied minimum number of rows desired in the plan
F	The number of factors in the design

L = 2

If all factors have two levels, simply select the smallest two-level Plackett-Burman plan for which $N_{plan} \geq \max(M, F+1)$.

L = 3

Let *P* = the number of factors with more than two levels, and let *K*=*F*+2*P*.

If *M*<9 and *F*<6 and *P*<2, base the plan on Table 73-5 “8 rows, 1 column of 4 levels plus 4 columns of 2 levels”.

If *M*<10 and *F*<5, base the plan on PB 3.4.

Otherwise, if *M*<17 and *K*<16, base it on Table 73-6 “16 rows, 5 columns of 4 levels each”.

Otherwise, if *M*<19 and *K*<8, base it on Table 73-4 “18 rows, 7 columns of 3 levels each”.

Otherwise, if *M*<28 and *K*<14, base it on PB 3.13.

Otherwise, if *M*<65 and *K*<22, use the rules for *L*=4.

Otherwise, if *F*<41, base the plan on PB 3.40.

If *F*>40, there are too many factors.

L = 4

Let *P* = the number of factors with more than two levels, and let *K*=*F*+2*P*.

If M<9 and F<6 and P<2, base the plan on Table 73-5 “8 rows, 1 column of 4 levels plus 4 columns of 2 levels”.

Otherwise, if M<17 and K<15, base it on Table 73-6 “16 rows, 5 columns of 4 levels each”.

Otherwise, if M<26 and K<19, base it on PB 5.6.

Otherwise, if M<33 and K<28, base it on Table 73-7 “32 rows, 9 columns of 4 levels each”.

Otherwise, if M<49 and K<23, use the rules for L=7.

Otherwise, if K<64, base the plan on Table 73-8 “64 rows, 21 columns of 4 levels each”.

Otherwise, there are too many factors.

A four-level factor can be transformed into three two-level factors using the rule in Table 73-1 “Converting a four-level factor to three two-level factors”.

L = 5

Create a plan based on the L=7 rules.

If that plan has 26 or more rows and M<26 and F<7, base the plan on PB 5.6.

Otherwise, use the plan generated in step 1.

L = 6

Treat this case as L=7.

L = 7

Generate the best plan based on L=8.

If that plan has more than 49 rows and M<50 and F<9, base the plan on PB 7.8.

Otherwise, use the plan generated in step 1.

L = 8

Let P be the number of factors with more than two levels, and Q be the number of factors with more than four levels.

If M<17 and F<10 and P<2, then base the plan on Table 73-9 “16 rows, 1 column of 8 levels plus 8 columns of 2 levels”.

Otherwise, if M<28 and F<11 and only one factor has more than three levels, base the plan on the L=9 rules.

Otherwise, if $M<33$ and $Q<2$ and $F+2P+4Q<32$, base the plan on Table 73-10 “31 rows, 1 column of 8 levels plus 8 columns of 4 levels”.

Otherwise, if $M<65$ and $F+6P<64$, base it on Table 73-11 “64 rows, 9 columns of 8 levels each”.

Otherwise, base the plan on the $L=9$ rules.

An eight-level factor can be transformed into seven two-level factors using the rule in Table 73-2 “Converting an eight-level factor to seven two-level factors”..

L = 9

Let P be the number of factors with more than three levels, and $K=F+3P$.

If $M<28$ and $F<11$ and $P<2$, then base the plan on Table 73-12 “27 rows, 1 column of 9 levels plus 9 columns of 3 levels”.

Otherwise, if $K<41$, base it on Table 73-13 “81 rows, 10 columns of 9 levels each”.

Otherwise, there are too many factors.

A nine-level factor can be transformed into four three-level factors using the rule in Table 73-3 “Converting a nine-level factor to four three-level factors”.

Randomization

After a basic plan has been selected, columns are selected at random (if possible) to fit the given design. If the basic plan is asymmetric; that is, one column has more levels than the others, then the factor in the plan with many levels must be assigned to the factor in the design with many levels, and the remaining plan factors must be assigned randomly to the remaining design factors.
If factors are to be transformed into multiple factors (for example, eight-level factors transformed into two-level factors), you can randomly assign columns from the plan to design factors with many levels first, then transform the remaining columns, and then select from the transformed columns at random the columns needed.

References

Addelman, S. 1962. Symmetrical and asymmetrical fractional factorial plans. *Technometrics*, 4, 47–58.

Plackett, R. L., and J. P. Burman. 1946. The design of optimum multifactorial experiments. *Biometrika*, 33, 305–325.

OVERALS Algorithms

The OVERALS algorithm was first described in Gifi (1981) and Van der Burg, De Leeuw and Verdegaal (1984); also see Verdegaal (1986), Van de Geer(1987), Van der Burg, De Leeuw and Verdegaal (1988), and Van der Burg (1988). Characteristic features of OVERALS, conceived by De Leeuw (1973), are the partitioning of the variables into K sets and the ability to specify any of a number of measurement levels for each variable separately. Analogously to the situation in multiple regression and canonical correlation analysis, OVERALS focuses on the relationships between sets; any particular variable contributes to the results only inasmuch as it provides information that is independent of the other variables in the same set.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of cases (objects)
m	Number of variables
p	Number of dimensions
K	Number of sets

For variable *j*; $j = 1, \ldots, m$

k_j	Number of valid categories (distinct values) of variable *j*
$\mathbf{G}_j$	Indicator matrix for variable *j*, of order $n \times k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i\text{th object is in the } r\text{th category of variable } j \\ 0 & \text{when the } i\text{th object is not in the } r\text{th category of variable } j \end{cases}$$

$\mathbf{D}_j$	Diagonal matrix, containing the univariate marginals; that is, the column sums of $\mathbf{G}_j$

For set *k*; $k = 1, \ldots, K$

J(*k*)	Index set of the variables that belong to set *k* (so that you can write $j \in J(k)$)
$\mathbf{m}_k$	Number of variables in set *k* (number of elements in *J*(*k*))
$\mathbf{M}_j$	Binary diagonal *n*×*n* matrix, with diagonal elements defined as

$$m_{(j)ii} = \begin{cases} 1 & \text{when the } i\text{th observation is within the range } [1, k_j] \text{ for all } j \in J(k) \\ 0 & \text{when the } i\text{th observation outside the range } [1, k_j] \text{ for all } j \in J(k) \end{cases}$$

The quantification matrices and parameter vectors are:

X	Object scores, of order *n*×*p*
$\mathbf{X}_j$	Auxiliary matrix of order *n*×*p*, with corrected object scores when fitting variable *j*
$\mathbf{Y}_j$	Category quantifications for multiple variables, of order $k_j \times p$
$\mathbf{y}_j$	Category quantifications for single variables, of order k_j

$\mathbf{a}_j$	Variable weights for single variables, of order *p*
$\mathbf{Q}_k$	Quantified variables of the *k*th set, of order $n \times m_k$ with columns $\mathbf{q}_j = \mathbf{G}_j\mathbf{y}_j$
$\underline{\mathbf{Y}}$	Collection of multiple and single category quantifications across variables and sets.

Note: The matrices $\mathbf{M}_k$, $\mathbf{G}_j$, $\mathbf{M}_j$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Objective Function Optimization

The OVERALS objective is to find object scores **X** and a set of $\underline{\mathbf{Y}}_j$ (for *j*=1,...,*m*) — the underlining indicates that they may be restricted in various ways — so that the function

$$\sigma(\mathbf{X};\underline{\mathbf{Y}}) = 1/K\sum_k tr\left[\left(\mathbf{X} - \sum_{j\in J(k)} \mathbf{G}_j\underline{\mathbf{Y}}_j\right)' \mathbf{M}_k \left(\mathbf{X} - \sum_{j\in J(k)} \mathbf{G}_j\underline{\mathbf{Y}}_j\right)\right]$$

is minimal, under the normalization restriction $\mathbf{X}'\mathbf{M}_*\mathbf{X} = kn\mathbf{I}$ where $\mathbf{M}_* = \sum_k \mathbf{M}_k$ and **I** is the *p*×*p* identity matrix. The inclusion of $\mathbf{M}_k$ in $\sigma(\mathbf{X};\underline{\mathbf{Y}})$ provides the following mechanism for weighting the loss: whenever any of the data values for object *i* in set *k* falls outside its particular range $[1, k_j]$, a circumstance that may indicate either genuine missing values or simulated missing values for the sake of analysis, all other data values for object *i* in set *k* are disregarded (listwise deletion per set). The diagonal of $\mathbf{M}_*$ contains the number of "active" sets for each object. The object scores are also centered; that is, they satisfy $\mathbf{u}'\mathbf{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ with **u** denoting an *n*-vector with ones.

Optimal Scaling Levels

The following optimal scaling levels are distinguished in OVERALS:

Multiple Nominal. $\underline{\mathbf{Y}}_j = \mathbf{Y}_j$ (equality restriction only).

(Single) Nominal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ (equality and rank – one restrictions).

(Single) Ordinal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{C}_j$ (equality, rank – one, and monotonicity restrictions). The monotonicity restriction $\mathbf{y}_j \in \mathbf{C}_j$ means that $\mathbf{y}_j$ must be located in the convex cone of all k_j-vectors with nondecreasing elements.

(Single) Numerical. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{L}_j$ (equality, rank – one, and linearity restrictions). The linearity restriction $\mathbf{y}_j \in \mathbf{L}_j$ means that $\mathbf{y}_j$ must be located in the subspace of all k_j-vectors that are a linear transformation of the vector consisting of k_j successive integers.

For each variable, these levels can be chosen independently. The general requirement for all options is that equal category indicators receive equal quantifications. The general requirement for the non-multiple options is $\underline{\mathbf{Y}}_j = \mathbf{y}_j \mathbf{a}'_j$; that is, $\underline{\mathbf{Y}}_j$ is of rank one; for identification purposes, $\mathbf{y}_j$ is always normalized so that $\mathbf{y}'_j \mathbf{D}_j \mathbf{y}_j = n_w$.

Optimization

Optimization is achieved by executing the following iteration scheme:

1. Initialization I or II
2. Loop across sets and variables
3. Eliminate contributions of other variables
4. Update category quantifications
5. Update object scores
6. Orthonormalization
7. Convergence test: repeat (2) through (6) or continue
8. Rotation

Steps (1) through (8) are explained below.

Initialization

I. Random

The object scores **X** are initialized with random numbers. Then **X** is normalized so that $\mathbf{u}'\mathbf{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ and $\mathbf{X}'\mathbf{M}_*\mathbf{X} = kn\mathbf{I}$, yielding $\tilde{\mathbf{X}}$. For multiple variables, the initial category quantifications are set equal to 0. For single variables, the initial category quantifications $\tilde{\mathbf{y}}_j$ are defined as the first k_j successive integers normalized in such a way that $\mathbf{u}'\mathbf{D}_j\tilde{\mathbf{y}}_j = 0$ and $\tilde{\mathbf{y}}_j\mathbf{D}_j\tilde{\mathbf{y}}_j = n$, and the initial variable weights are set equal to 0.

II. Nested

In this case, the above iteration scheme is executed twice. In the first cycle, (initialized with initialization I) all single variables are temporarily treated as single numerical, so that for the second, proper cycle, all relevant quantities can be copies from the results of the first one.

Loop across sets and variables

The next two steps are repeated for *k*=1,...,*K* and all $j \in J(k)$. During the updating of variable *j*, all parameters of the remaining variables are fixed at their current values.

Eliminate contributions of other variables

For quantifying variable *j* in set *k*, define the auxiliary matrix

$$\mathbf{V}_{(k)j} = \Sigma_{j \in J(k)} \mathbf{G}_j \underline{\mathbf{Y}}_j - \mathbf{G}_j \underline{\mathbf{Y}}_j$$

which accumulates the contributions of the other variables in set k; then in $\left(\mathbf{X} - \mathbf{V}_{(k)j}\right)$, the contributions of the other variables are eliminated from the object scores. This device enables you to write the loss $\sigma\left(\mathbf{X}; \underline{\mathbf{Y}}_j\right)$ as a function of $\mathbf{X}$ and $\underline{\mathbf{Y}}_j$ only:

$$\sigma\left(\mathbf{X}; \underline{\mathbf{Y}}_j\right) = \text{constant} + 1/K \operatorname{tr}\left[\left(\left(\mathbf{X} - \mathbf{V}_{(k)j}\right) - \mathbf{G}_j \underline{\mathbf{Y}}_j\right)' \mathbf{M}_k \left(\left(\mathbf{X} - \mathbf{V}_{(k)j}\right) - \mathbf{G}_j \underline{\mathbf{Y}}_j\right)\right]$$

With fixed current values $\tilde{\mathbf{X}}$ the unconstrained minimum over $\underline{\mathbf{Y}}_j$ is attained for the matrix

$$\tilde{\mathbf{Y}}_j = \left(\mathbf{G}'_j \mathbf{M}_k \mathbf{G}_j\right)^{-1} \mathbf{G}'_j \mathbf{M}_k \left(\tilde{\mathbf{X}} - \mathbf{V}_{(k)j}\right)$$

which forms the basis of the further computations. When switching to another variable l in the same set, the matrix $\mathbf{V}_{(k)l}$ is not computed from scratch, but updated:

$$\mathbf{V}_{(k)l} \leftarrow \mathbf{V}_{(k)j} + \mathbf{G}_j \underline{\mathbf{Y}}_j - \mathbf{G}_l \underline{\mathbf{Y}}_l$$

Update category quantifications

For multiple nominal variables, the new category quantifications are simply

$$\underline{\mathbf{Y}}_j^+ = \tilde{\underline{\mathbf{Y}}}_j$$

For single variables one cycle of an ALS algorithm (De Leeuw et al., 1976) is executed for computing the rank-one decomposition of $\tilde{Y}_j$, with restrictions on the left-hand vector. This cycle starts from the previous category quantification $\tilde{\mathbf{y}}_j$ with

$$\mathbf{a}_j^+ = \tilde{\mathbf{Y}}_j' \mathbf{D}_j \tilde{\mathbf{y}}_j$$

When the current variable is numerical, we are ready; otherwise we compute

$$\mathbf{y}_j^* = \tilde{\mathbf{Y}}_j \mathbf{a}_j^+.$$

Now, when the current variable is single nominal, you can simply obtain $\mathbf{y}_j^+$ by normalizing $\mathbf{y}_j^*$ in the way indicated below; otherwise the variable must be ordinal, and you have to insert the weighted monotonic regression process

$$\mathbf{y}_j^* \leftarrow \text{WMON}(\mathbf{y}_j^*) .$$

The notation WMON( ) is used to denote the weighted monotonic regression process, which makes $\mathbf{y}_j^*$ monotonically increasing. The weights used are the diagonal elements of $\mathbf{D}_j$ and the subalgorithm used is the up-and-down-blocks minimum violators algorithm (Kruskal, 1964; Barlow et al., 1972). The result is normalized:

$$\mathbf{y}_j^+ = n_w^{1/2} \mathbf{y}_j^* \left(\mathbf{y}'^*_j \mathbf{D}_j \mathbf{y}_j^*\right)^{-1/2}$$

Finally, we set $\underline{\mathbf{Y}}_j^+ = \mathbf{y}_j^+ \mathbf{a}'^+_j$.

Update object scores

First the auxiliary score matrix $\mathbf{W}$ is computed as

$$\mathbf{W} \leftarrow \mathbf{W} + \mathrm{M}_k \Sigma_{j \in J(k)} \mathbf{G}_j \underline{\mathbf{Y}}_j^+$$

and centered with respect to M_*:

$$\mathbf{X}^* = \left\{ \mathbf{I} - \mathrm{M}_* \mathbf{u}\mathbf{u}' / \mathbf{u}' \mathrm{M}_* \mathbf{u} \right\} \mathbf{W}$$

These two steps yield locally the best updates when there would be no orthogonality constraints.

Orthonormalization

The problem is to find an M_*-orthonormal $\mathbf{X}^+$ that is closest to $\mathrm{M}_*^{-1}\mathbf{X}^*$ in the M_*-weighted least squares sense. In OVERALS, this is done by setting

$$\mathbf{X}^+ \leftarrow m^{1/2} \mathbf{M}_*^{-1/2} \text{PROCRU} \left(\mathbf{M}_*^{-1/2} \mathbf{X}^* \right)$$

The notation PROCRU( ) is used to denote the Procrustes orthonormalization process. If the singular value decomposition of the input matrix $\mathrm{M}_*^{-1/2}\mathbf{X}^*$ is denoted by $\mathbf{K}\mathbf{\Lambda}'\mathbf{L}'$, with $\mathbf{K}'\mathbf{K} = \mathbf{I}, \mathbf{L}'\mathbf{L} = \mathbf{I}$, and $\mathbf{\Lambda}$ diagonal, then the output matrix $\mathbf{K}\mathbf{L}' = \mathrm{M}_*^{-1/2}\mathbf{X}^*\mathbf{L}\mathbf{\Lambda}'^{-1}\mathbf{L}'$ satisfies orthonormality in the metric M_*. The calculation of $\mathbf{L}$ and $\mathbf{\Lambda}$ is based on tridiagonalization with Householder transformations followed by the implicit QL algorithm (Wilkinson, 1965).

Convergence test

The difference between consecutive values of Λ^4 is compared with the user-specified convergence criterion ε - a small positive number. After convergence, the badness-of-fit values is also given. Steps (2) through (6) are repeated as long as the loss difference exceeds ε.

Rotation

The OVERALS loss function $\sigma(\mathbf{X}; \underline{\mathbf{Y}})$ is invariant under simultaneous rotations of $\mathbf{X}$ and $\underline{\mathbf{Y}}$. It can be shown that the solution is related to the principal axes of the average projection operator

$$Q_* = 1/K \Sigma_k \mathbf{M}_k \mathbf{Q}_k \left(\mathbf{Q}'_k \mathbf{M}_k \mathbf{Q}_k \right)^{-1} \mathbf{Q}'_k \mathbf{M}_k$$

In order to achieve principal axes orientation, which is useful for purposes of interpretation and comparison, it is sufficient to find a rotation matrix that makes the cross-products of the matrix $\mathrm{M}_*^{-1/2}\mathbf{X}^*$ diagonal - a matrix identical to the one used in the Procrustes orthonormalization in step (6). In the terminology of that section, we rotate the matrices $\mathbf{X}^+, \underline{\mathbf{Y}}^+$, and the vectors $\mathbf{a}_j$ with the matrix $\mathbf{L}$. The rotation matrix $\mathbf{L}$ is taken from the last PROCRU operation as described in step (6).

Diagnostics

The following diagnostics are available.

Maximum Rank

The maximum rank $\rho_{\max}$ indicates the maximum number of dimensions that can be computed for any dataset (if exceeded, OVERALS adjusts the number of dimensions if possible and issues a message). In general,

$$\rho_{\max} = \begin{cases} \min\{(n-1), r_1, r_2\} & \text{if } K = 2 \\ \min\{(n-1), \max r_k\} & \text{if } K > 2 \end{cases}$$

where the quantities r_k are defined as

$$r_k = \sum_{j \in JM(k)} k_j + m_{k1} - m_{k2}$$

Here m_{k1} is the number of multiple variables with no missing values in set k, m_{k2} is the number of single variables in set *k*, and $JM(k)$ is an index set recording which variables are multiple in set *k*. Furthermore, OVERALS stops when any one of the following conditions is not satisfied:

1. $r_k < n_k - 1$
2. $n_k > 2$
3. $\sum_k r_k \le \Sigma_k(n_k - 1) - (n_{\max} - 1)$

Here n_k denotes the number of nonmissing objects in set *k*, and $n_{\max}$ denotes the maximum across all of n_k.

Marginal Frequencies

The frequencies table gives the univariate marginals and the number of missing values (that is, values that are regarded as out of range for the current analysis) for each variable. These are computed as the column sums of $\mathbf{D}_j$ and the total sum of $\mathbf{M}_k$ for $j \in J(k)$.

Fit and Loss Measures

In the Summary of Analysis, loss and fit measures are reported.

Loss Per Set

This is *K* times $\sigma(\mathbf{X}; \underline{\mathbf{Y}})$, partitioned with respect to sets and dimensions; the means per dimension are also given.

Eigenvalue

The values listed here are 1 minus the means per dimension defined above, forming a partitioning of FIT, which is $\rho - \sigma(\mathbf{X}; \underline{\mathbf{Y}})$ when convergence is reached. These quantities are the eigenvalues of $\mathbf{Q}_*$ defined in section (8).

Multiple Fit

This measure is computed as the diagonal of the matrix $\underline{\mathbf{Y}}_j^{'}\mathbf{D}_j\underline{\mathbf{Y}}_j$, computed for all variables (rows) with dimensions given in the columns.

Single Fit

This table gives the squared weights, computed only for variables that are single. The sum of squares of the weights: $\mathbf{a}^{'}_j\mathbf{a}_j$.

Single Loss

Single loss is equal to multiple fit minus single fit for single variables only. It is the loss incurred by the imposition of the rank-one measurement level restrictions.

Component Loadings (for Single Variables)

Loadings are the lengths of the projections of the quantified (single) variables onto the object space: $\mathbf{q}^{'}_j\mathbf{X}$. When there are no missing data, the loadings are equal to the correlations between the quantified variables and the object scores (the principal components).

Category Quantifications

Single Coordinates. For single variables only: $\underline{\mathbf{Y}}_j \quad = \mathbf{y}_j\mathbf{a}^{'}_j$

Multiple Coordinates. These are $\tilde{\mathbf{Y}}_j$ defined previously; that is, the unconstrained minimizers of the loss function, for multiple variables equal to the category quantifications.

Category Centroids

The centroids of all objects that share the same category, $\mathbf{D}_j^{-1}\mathbf{G}^{'}_j\mathbf{X}$. Note that they are not necessarily equal to the multiple coordinates.

Projected Category Centroids

For single variables only, $\mathbf{y}_j\mathbf{b}^{'}_j$. These are the points on a line in the direction given by the loadings $\mathbf{b}_j$ that result from projection of the category centroids with weights $\mathbf{D}_j$.

References

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions.* New York: John Wiley and Sons.

Cliff, N. 1966. Orthogonal rotation to congruence. *Psychometrika*, 31, 33–42.

De Leeuw, J. 1984. *Canonical analysis of categorical data*, 2nd ed. Leiden: DSWO Press.

De Leeuw, J., F. W. Young, and Y. Takane. 1976. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41, 471–503.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Van de Geer, J. P. 1987. *Algebra and geometry of OVERALS: Internal Report RR-87–13*. Leiden: Department of Data Theory, University of Leiden.

Van der Burg, E. 1988. *Nonlinear canonical correlation and some related techniques*. Leiden: DSWO Press.

Van der Burg, E., J. De Leeuw, and R. Verdegaal. 1984. *Non-linear canonical correlation analysis: Internal Report RR-84–12*. Leiden: Department of Data Theory, University of Leiden.

Van der Burg, E., J. De Leeuw, and R. Verdegaal. 1988. Homogeneity analysis with k sets of variables: An alternating least squares method with optimal scaling features. *Psychometrika*, 53, 177–197.

Verdegaal, R. 1986. *OVERALS: Internal Report UG-86–01*. Leiden: Department of Data Theory, University of Leiden.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem*. Oxford: Clarendon Press.

PARTIAL CORR Algorithms

PARTIAL CORR produces partial correlation coefficients that describe the relationship between two variables while adjusting for the effects of one or more additional variables.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 75-1
Notation

Notation	**Description**
N	Number of cases
X_{kl}	Value of variable *k* for case *l*
w_l	Weight for case *l*
W_{ij}	Sum of the weights of cases used in computation of statistics for variable *i* and *j*
W_i	Sum of the weights of cases used in computation of statistics for variable *i*

Statistics

Zero-Order Correlations

$$r_{ij} = \frac{\sum_{l=1}^{N} w_l X_{il} X_{jl} - \left(\sum_{l=1}^{N} w_l X_{il}\right)\left(\sum_{l=1}^{N} w_l X_{jl}\right)/W_{ij}}{\sqrt{\left(\sum_{l=1}^{N} w_l X_{il}^2 - \left(\sum_{l=1}^{N} w_l X_{il}\right)^2/W_{ij}\right)\left(\sum_{l=1}^{N} w_l X_{jl}^2 - \left(\sum_{l=1}^{N} w_l X_{jl}\right)^2/W_{ij}\right)}}$$

Noncomputable coefficients are set to system missing. The significance level for r_{ij} is based on

$$t = r_{ij}\sqrt{\frac{W_{ij} - 2}{1 - r_{ij}^2}}$$

which, under the null hypothesis, is distributed as a *t* with $W_{ij} - 2$ degrees of freedom. By default, one-tailed significance levels are printed.

Means and Standard Deviations

$$\overline{X}_j = \sum_{i=1}^{N} w_i X_{ji}/W_j$$

$$S_j = \sqrt{\left(\sum_{i=1}^{N} w_i X_{ji}^2 - \overline{X}_j^2 W_j\right)/(W_j - 1)}$$

If pairwise deletion is selected, means and standard deviations are based on *all* nonmissing cases. For listwise deletion, only cases with no missing values on any specified variables are included.

Partial Correlations

Partial correlations are calculated recursively from the lower-order coefficients using

$$r_{ij.k}=\frac{r_{ij}-r_{ki}r_{kj}}{\sqrt{\left(1-r_{ki}^2\right)\left(1-r_{kj}^2\right)}}\text{(first order)}$$

$$r_{ij.kl}=\frac{r_{ij.k}-r_{il.k}r_{jl.k}}{\sqrt{\left(1-r_{il.k}^2\right)\left(1-r_{jl.k}^2\right)}}\text{(second order)}$$

and similarly for higher orders ((Morrison, 1976) p. 94).

If the denominator is less than 10^{-20}, or if any of the lower-order coefficients necessary for calculations are system missing, the coefficient is set to system missing. If a coefficient in absolute value is greater than 1, it is set to system missing. (This may occur with pairwise deletion.)

Significance Level

The significance level is based on

$$t=r\sqrt{\frac{df}{1-r^2}}$$

The degrees of freedom are

$$df=M-\theta-2$$

where θ is the order of the coefficient and M is the minimum sum of weights from which the zero-order coefficients involved in the computations were calculated. Thus, for $r_{ij.kl}$

$$M=\min\left(W_{ij},W_{ki},W_{kj},W_{il},W_{lk},W_{jl}\right)$$

where W_{ij} is the sum of weights of the cases used to calculated r_{ij}. If listwise deletion of missing values (default) was used, all W_{ij} are equal. By default, the significance level is one-tailed.

References

Morrison, D. F. 1976. *Multivariate statistical methods*. New York: McGraw-Hill.

PLS Algorithms

Partial least squares (PLS) regression fits a model for one or more dependent variables based upon one or more predictors. It is especially useful when the predictors exhibit multicollinearity, or there are more predictors than cases.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 76-1
Notation

Notation	Description
X	$N \times n$ design matrix of independent variables, centered and perhaps standardized. Note that there is no intercept term.
Y	$N \times m$ matrix of dependent variables, centered and perhaps standardized
c	$m \times 1$ column vector of weights
u	$N \times 1$ column vector of Y scores
w	$n \times 1$ column vector of weights
t	$N \times 1$ column vector of X scores
d	number of PLS factors to extract
p	$n \times 1$ loading vector
q	$m \times 1$ loading vector
P	$n \times d$ loading matrix
Q	$m \times d$ loading matrix
T	$N \times d$ score matrix, $\mathbf{T} = \mathbf{X}\mathbf{W}^*$
U	$N \times d$ score matrix
W	$n \times d$ matrix of X-weights
$\mathbf{W}^*$	$n \times d$ matrix of X-weights in original coordinates; these weights can be directly applied to **X**, $\mathbf{W}^* = \mathbf{W}\left(\mathbf{P}'\mathbf{W}\right)^{-1}$
C	$m \times d$ matrix of Y-weights; these weights can be directly applied to **Y**.
B	$n \times m$ matrix of regression parameters, $\mathbf{B} = \mathbf{W}^*\mathbf{C}'$
E	$N \times n$ matrix of residuals, $\mathbf{E} = \mathbf{X} - \mathbf{TP}'$
F	$N \times m$ matrix of residuals, $\mathbf{F} = \mathbf{Y} - \mathbf{UQ}' = \mathbf{Y} - \mathbf{XB}$
DModX	$N \times 1$ vector of distances of **X** variables to the model
DModY	$N \times 1$ vector of distances of **Y** variables to the model
VIP	$n \times d$ matrix of Variable Importance in the Projection

Preprocessing

The following steps are performed before the estimation algorithm commences.

Design Matrix

The design matrix **X** is constructed from the independent variables as in GLM models without an intercept.

Categorical Variable Encoding

The procedure temporarily recodes categorical dependent variables using one-of-*c* coding for the duration of the procedure. If there are *c* categories of a variable, then the variable is stored as *c* vectors, with the first category denoted (1,0,...,0), the next category (0,1,0,...,0), ..., and the final category (0,0,...,0,1).

Categorical dependent variables are represented using dummy coding; that is, simply omit the indicator corresponding to the reference category. In particular, when there is a single dependent variable with exactly two levels, there will be a single indicator, and convergence will occur in a single NIPALS iteration.

Missing Values

Cases with user- or system-missing values are handled as follows:

Listwise Deletion. Only cases with complete values for all **X** and **Y** variables will be used.

Center and Standardize Variables

Given a matrix of independent variables **X** and of dependent variables **Y** (with the design matrix, categorical variable encoding, and missing values), compute the mean and standard deviation of each variable, and replace **X** with the centered and standardized variates $\mathbf{X} \coloneqq (\mathbf{X} - \mu_{\mathbf{X}})\boldsymbol{\Sigma}_{\mathbf{X}}^{-1}$ where $\boldsymbol{\Sigma}_{\mathbf{X}}$ is a diagonal matrix of standard deviations and $\mu_{\mathbf{X}}$ is the vector of means; similarly for $\widehat{\mathbf{Y}} \coloneqq \mathbf{Y}\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} + \mu_{\mathbf{Y}}$. This change of coordinates must be reversed after all components have been extracted; $\widehat{\mathbf{Y}} \coloneqq \mathbf{Y}\boldsymbol{\Sigma}_{\mathbf{Y}}^{-1} + \mu_{\mathbf{Y}}$.

Estimation

When there is only one dependent variable (*m*=1), use the NIPALS algorithm. Only one iteration will be required. When there is more than one dependent variable (*m*>1), solve the equivalent eigenproblem, solving for the vector with the smallest dimension. Use the resulting eigenvector as the input to NIPALS, checking the vector with the greatest length for convergence. (This check may turn out to be unneeded, in which case one iteration of NIPALS will still be needed to obtain all the required vectors.)

This diagram illustrates the relationship between the vectors and matrices used in the NIPALS algorithm, where the vectors should be taken as determined only up to scalar multiples:

$$\begin{tikzcd}
 & & \mathbf{c} \arrow[dr, "\mathbf{Y}"] & & \\
\mathbf{p}=\mathbf{X}'\mathbf{t}/(\mathbf{t}'\mathbf{t}) & \mathbf{t} \arrow[l, "\mathbf{X}'"'] \arrow[ur, "\mathbf{Y}'"] & & \mathbf{u} \arrow[r, "\mathbf{Y}'"'] \arrow[dl, "\mathbf{X}'"] & \mathbf{q}=\mathbf{Y}'\mathbf{u}/(\mathbf{u}'\mathbf{u}) \\
 & & \mathbf{w} \arrow[ul, "\mathbf{X}"] & &
\end{tikzcd}$$

NonLinear Iterative Partial Least Squares (NIPALS) Algorithm

The classical NIPALS algorithm explicitly takes **c** and **w** to have unit norm. In particular, note that if there is only one dependent variable **Y**, then **c** is a 1×1 unit vector so **c** = 1, and this will be the most useful starting point: initialize **u** = **Y**; otherwise, initialize **u** or any of the vectors to some random starting value. Also, when **c** = 1, then NIPALS converges in only one iteration.

The following loop may be entered at any point which is most convenient, most especially when $m = 1$, **c** = 1, begin at step 1 with **u** = **Y**:

Repeat until convergence:

1. **w** = **X'u**/(**u'u**)
2. **w** := **w**/||**w**||
3. **t** = **Xw**
4. **c** = **Y't**/(**t't**)
5. **c** := **c**/||**c**||
6. **u** = **Yc**

Although the NIPALS algorithm will in practice be replaced with the solution of an eigenproblem (see "NIPALS-Equivalent Eigenproblem") the relationships defined in the sequence above will be used to obtain all the matrices and vectors required.

Regress **X** on **t** and **Y** on **u**:

1. **p** = **X't**/(**t't**)
2. **q** = **Y'u**/(**u'u**)

Deflate **X** and **Y** matrices:

1. **X** := **X** – **tp'**
2. **Y** := **Y** – **tc'** (use **c** from step 4, not step 5, above)

Note that the deflated matrices are the errors **E**, **F** at that stage.

Repeat d times, assembling the **t**, **p**, **u**, **q** vectors into matrices to obtain the desired factorizations into scores **T**, **U**, loadings **P**, **Q**, weights **W**, **C**, and residuals **E**, **F**:

- **X** = **TP'** + **E**
- **Y** = **UQ'** + **F**

Since the matrices **X**, **Y** are centered, note that $\left(\mathrm{t}'\mathrm{t}\right)^{-1}\mathrm{t}'\mathbf{Y}$ is the normal equation for a regression of **Y** on **t**, likewise $\left(\mathrm{u}'\mathrm{u}\right)^{-1}\mathbf{u}'\mathrm{X}$ regresses **X** on **u**. Thus the NIPALS algorithm alternates between regression and projection. If vectors are considered to be determined only up to length, there is no longer any distinction between the two.

The matrix of regression coefficients for predicting **Y** from **X** is given by either any of the following expressions, and is independent of the scalings of **T** and **U**:

$$\mathrm{B} = \mathrm{W}^{*}\mathrm{C}^{'}$$
$$\mathrm{B} = \mathrm{W}\left(\mathrm{P}^{'}\mathrm{W}\right)^{-1}\mathrm{C}^{'}$$
$$\mathrm{B} = \mathrm{X}^{'}\mathrm{U}\left(\mathrm{T}^{'}\mathrm{X}\mathrm{X}^{'}\mathrm{U}\right)^{-1}\mathrm{T}^{'}\mathrm{Y}$$

W and **C** are obtained by assembling the **w** and **c** vectors into $n\times d$ and $m\times d$ matrices. This solves the PLS Regression equation:

Y = **XB** + **F**

Until now the **X** and **Y** matrices have been assumed to be centered, and (optionally) standardized. The parameters **B** and residuals **E** and **F** must be restored to their original coordinates $\mathrm{B}^{*}=\mathbf{\Sigma}_{\mathbf{X}}^{-1}\mathrm{B}\mathbf{\Sigma}_{\mathbf{Y}}$, $\mathbf{E}^{*}=\mathbf{E}\mathbf{\Sigma}_{\mathbf{X}}$, with the final regression equation in the original coordinates given by $\hat{\mathrm{Y}}=\mathrm{XB}^{*}+(\mu_{\mathbf{Y}}-\mu_{\mathbf{X}}\mathrm{B}^{*})$. Also, the residuals **F** left over after deflating the **Y** matrix should not be used, but are recalculated from the predictions in the centered and rescaled coordinates as **F** = **Y** – **XB**; $\mathbf{F}^{*}=\mathbf{F}\mathbf{\Sigma}_{\mathbf{Y}}$ in the original coordinates.

NIPALS-Equivalent Eigenproblem

Regarding the vectors as determined only up to length allows the NIPALS loop to be replaced by an eigenproblem. One can choose to solve any of the following; typically selecting the matrix with the smallest dimension, which will often be the first equation:

$$\mathbf{Y}^{'}\mathbf{X}\mathbf{X}^{'}\mathbf{Y}\mathbf{c}=\lambda\mathbf{c}$$
$$\mathbf{Y}\mathbf{Y}^{'}\mathbf{X}\mathbf{X}^{'}\mathbf{u}=\lambda\mathbf{u}$$
$$\mathbf{X}^{'}\mathbf{Y}\mathbf{Y}^{'}\mathbf{X}\mathrm{w}=\lambda\mathrm{w}$$
$$\mathbf{X}\mathbf{X}^{'}\mathbf{Y}\mathbf{Y}^{'}\mathbf{t}=\lambda\mathbf{t}$$

Once **c** (or any of the others) are determined, the rest of the vectors can be determined; at this point it is important to keep track of the lengths.

The eigenproblem can be solved by the Power Method.

Power Method

$$\mathbf{x}_{i+1}=\mathbf{A}\mathbf{x}_{i}$$
$$\lambda_{i}=\mathbf{x}^{'}_{i}\mathbf{A}\mathbf{x}_{i}=\mathbf{x}^{'}_{i}\mathbf{x}_{i+1}$$
$$\mathbf{x}_{i+1}:=\mathbf{x}_{i+1}/\|\mathbf{x}_{i+1}\|$$

Initialize a vector $\mathbf{x}_0$ say to the vector **1**, normalize to unit length, then iterate until convergence. The sequence of iterates is guaranteed to converge to the eigenvector associated to the dominant (that is, the largest) eigenvalue. Moreover the dominant eigenvalue is guaranteed to be unique.

Rather than continue to iterate using the power method, switch to Rayleigh Quotient Iteration (RQI).

Rayleigh Quotient Iteration

Begin with initial estimates of $\mathbf{x}_0$ and λ_0 obtained from one or two iterations of the Power Method. Then repeat until convergence:

$$\begin{aligned}&(\mathbf{A}-\lambda_i\mathbf{I})\mathrm{w}=\mathbf{x}_i \quad \text{(solve for w)}\\ &\mathbf{x}_{i+1}=\mathbf{w}/\|\mathbf{w}\|\\ &\lambda_{i+1}=\mathbf{x}'_{i+1}\mathbf{A}\mathbf{x}_{i+1}\end{aligned}$$

The conjugate gradient method may be used to solve for **w**.

The eigenproblem is considered solved when the difference between two iterations is small enough. However, the eigenproblem is typically solved for **c**, but the vector of interest is **t**. One iteration of NIPALS is used to obtain the vectors (**c**, **u**, **w**, **t**).

Output Statistics

The following output statistics are available.

Proportion of Variance Explained

The proportion of variance explained by the extraction of factor *k* is given by computing:

$$\begin{aligned}SS_k(\mathbf{Y}) &= \left(\mathbf{t}'_{(k)}\mathbf{t}_{(k)}\right)\cdot trace\left(\mathbf{c}_{(k)}\mathbf{c}'_{(k)}\right)\\ &= \left(\mathbf{t}'_{(k)}\mathbf{t}_{(k)}\right)\cdot\left(\mathbf{c}'_{(k)}\mathbf{c}_{(k)}\right)\end{aligned}$$

$$VarProp_k(\mathbf{Y})=\frac{SS_k(\mathbf{Y})}{trace(\mathbf{Y}'\mathbf{Y})}$$

The cumulative proportion of variance explained is

$$CumVarProp_k(\mathbf{Y})=\sum_{i=1}^{k}Var_i(\mathbf{Y})$$

Here $\mathbf{t}_{(k)}$ and $\mathbf{c}_{(k)}$ are the column vectors obtained after *k* factors have been extracted; that is, the *k*th columns of **T** and **C**. Note that $\mathbf{c}_{(k)}$ is taken from step 4 of the NIPALS algorithm, and is not rescaled to unit length as in step 5.

The proportion of variance explained in **X** is similar:

$$\begin{aligned}SS_k(\mathbf{X}) &= \left(\mathbf{t}'_{(k)}\mathbf{t}_{(k)}\right)\cdot trace\left(\mathbf{p}_{(k)}\mathbf{p}'_{(k)}\right)\\ &= \left(\mathbf{t}'_{(k)}\mathbf{t}_{(k)}\right)\cdot\left(\mathbf{p}'_{(k)}\mathbf{p}_{(k)}\right)\end{aligned}$$

$$VarProp_k(\mathbf{X})=\frac{SS_k(\mathbf{X})}{trace(\mathbf{X}'\mathbf{X})}$$

$$CumVarProp_k(\mathbf{X}) = \sum_{i=1}^{k} Var_i(\mathbf{X})$$

Variable Importance in the Projection (VIP)

The VIP statistic is computed for each variable and latent factor as

$$\text{VIP}_{jk} = \sqrt{\frac{n\sum_{l=1}^{k} {w^*_{jl}}^2 \cdot SS_l(\mathbf{Y})}{\sum_{l=1}^{k} SS_l(\mathbf{Y})}}$$

Here $1 \le j \le n$ and $1 \le k \le d$; w^*_{jk} is the *j*th element of $\mathbf{w}^*_{(k)}$, where $\mathbf{w}^*_{(k)}$ is the *k*th column of **W**.

Distance to the Model

Distance to the model, sometimes denoted DModX and DModY, is given by:

$$DModX_i = \sqrt{\mathbf{e}'_i\mathbf{e}_i}$$
$$DModY_i = \sqrt{\mathbf{f}_i\mathbf{f}_i}$$

for each row $\mathbf{e}_i$ of **E** and $\mathbf{f}_i$ of **F**. This may be normalized to:

$$DModX_i = \sqrt{\tfrac{N}{N-d-1}\mathbf{e}'_i\mathbf{e}_i}$$
$$DModY_i = \sqrt{\tfrac{N}{N-d-1}\mathbf{f}_i\mathbf{f}_i}$$

PRESS Statistic

The PRESS residuals are $\mathbf{f}_i{}'\mathbf{f}$, that is, $DModY_i^2$ before any normalizations are carried out. The PRESS statistic is simply

$$\text{PRESS} = \sum_{i=1}^{N} DModY_i^2$$

"Jackknifed", or more correctly, leave-one-out PRESS residuals are calculated as $\mathbf{f}_{(i)} = \frac{\mathbf{Y}_i - \hat{\mathbf{Y}}_i}{\sqrt{1-h_{ii}}}$ where $\mathbf{Y}_i$ is the *i*th row of **Y**, $\hat{\mathbf{Y}}_i$ is the predicted value for that row, and h_{ii} is the *i*th diagonal element of the "hat" matrix $\mathbf{X}\left(\mathbf{X}'\mathbf{X}\right)^{-1}\mathbf{X}'$. Leave-one-out PRESS residuals are not available when there are more variables than cases, or when **X'X** is not invertible for any other reason. The Jackknifed PRESS statistic for model selection is the sum of the squared norm of the Jackknifed PRESS residuals:

$$PRESS = \sum_{i=1}^{N} \mathbf{f}'_{(i)}\mathbf{f}_{(i)}$$

References

Rosipal, R., and N. Krämer. 2006. Overview and Recent Advances in Partial Least Squares. In: *Subspace, Latent Structure and Feature Selection Techniques,* C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, eds. Berlin: Springer-Verlag, 34–51.

PLUM Algorithms

The purpose of the PLUM procedure is to model the dependence of an ordinal categorical response variable on a set of categorical and scale independent variables.

Since the choice and the number of response categories can be quite arbitrary, it is essential to model the dependence such that the choice of the response categories does not affect the conclusion of the inference. That is, the final conclusion should be the same if any two or more adjacent categories of the old scale are combined. Such considerations lead to modeling the dependence of the response on the independent variables by means of the cumulative response probability.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 77-1
Notation

Notation	Description
Y	The response variable, which takes integer values from 1 to J.
J	The number of categories of the ordinal response.
m	The number of subpopulations.
$\mathbf{X}^A$	$m \times p^A$ matrix with vector-element x_i^A, the observed values at the ith subpopulation, determined by the independent variables specified in the command.
$\mathbf{X}$	$m \times p$ matrix with vector-element x_i, the observed values of the location model's independent variables at the ith subpopulation.
$\mathbf{Z}$	$m \times p$ matrix with vector-element x_i, the observed values of the scale model's independent variables at the ith subpopulation.
f_{ijs}	The frequency weight for the sth observation which belongs to the cell corresponding to $Y=j$ at subpopulation i.
n_{ij}	The sum of frequency weights of the observations that belong to the cell corresponding to $Y=j$ at subpopulation i.
r_{ij}	The cumulative total up to and including $Y=j$ at subpopulation i.
n_i	The marginal frequency of subpopulation i.
n	The sum of all frequency weights.
π_{ij}	The cell probability corresponding to $Y=j$ at subpopulation i.
γ_{ij}	The cumulative response probability up to and including $Y=j$ at subpopulation i.
$\boldsymbol{\theta}$	$(J-1)\times 1$ vector of threshold parameters in the location part of the model.
$\boldsymbol{\beta}$	$p\times 1$ vector of location parameters in the location part of the model.
$\boldsymbol{\tau}$	$q\times 1$ vector of scale parameters in the scale part of the model.
$\mathbf{B}=(\boldsymbol{\theta}^T,\boldsymbol{\beta}^T,\boldsymbol{\tau}^T)^T$	The $\{(J-1)+p+q\}\times 1$ vector of unknown parameters in the general model.
$\hat{\mathbf{B}}=\left(\hat{\theta}^T,\hat{\beta}^T,\hat{\tau}^T\right)^T$	The $\{(J-1)+p+q\}\times 1$ vector of maximum likelihood estimates of the parameters in the general model.
$\breve{\mathbf{B}}=\left(\breve{\theta}^T,\breve{\beta}^T\right)^T$	The $\{(J-1)+p\}\times 1$ vector of maximum likelihood estimates of the parameters in the location-only model.
$\hat{\gamma}_{ij}$	The cumulative response probability estimate based on the maximum likelihood estimate $\hat{\mathbf{B}}$ in the general model.

Notation	Description
$\breve{\gamma}_{ij}$	The cumulative response probability estimate based on the maximum likelihood estimate $\breve{\mathbf{B}}$ in the location-only model.
$\hat{\pi}_{ij}$	The cell response probability estimate based on the maximum likelihood estimate $\hat{\mathbf{B}}$ in the general model.
$\breve{\pi}_{ij}$	The cell response probability estimate based on the maximum likelihood estimate $\breve{\mathbf{B}}$ in the location-only model.
$\hat{e}$	Number of non-redundant parameters in the general model. If all parameters are non-redundant, $\hat{e} = (J-1) + p + q$.
$\breve{e}$	Number of non-redundant parameters in the location-only model. If all parameters are non-redundant, $\breve{e} = (J-1) + p$.

Data Aggregation

Observations with negative or missing frequency weights are discarded. Observations are aggregated by the definition of subpopulations. Subpopulations are defined by the cross-classifications of either the set of independent variables specified in the command or the set of independent variables specified in the subpopulation command.

Let n_i be the marginal count of subpopulation i,

$$n_i = \sum_{j=1}^{J} n_{ij}$$

If there is no observation for the cell of $Y=j$ at subpopulation i, it is assumed that $n_{ij} = 0$, provided that $n_i \neq 0$. A non-negative scalar $\delta \in [0,1)$ may be added to any zero cell (i.e., cell with $n_{ij} = 0$) if its marginal count n_i is nonzero. The value of δ is zero by default.

Data Assumptions

Let $(n_{i1}, ..., n_{iJ})^{\mathrm{T}}$ be the $J \times 1$ vector of counts for the categories of Y at subpopulation. It is assumed that each $(n_{i1}, ..., n_{iJ})^{\mathrm{T}}$ is independently multinomial distributed with probability vector $(\pi_{i1}, ..., \pi_{iJ})^{\mathrm{T}}$ of dimension $J \times 1$ and fixed total n_i.

Model

Let $\gamma_{ij} = Prob(Y \leq j|x_i)$ be the cumulative response probability for Y; that is,

$$\gamma_{ij} = \sum_{l=1}^{j} \pi_{il}$$

for j = 1, …, J–1. Note that $\gamma_{iJ} = 1$, hence only the first J–1 γ's are needed in the model.

General Model

The general model is given by

$\eta_{ij} = \frac{\theta_j - b^T x_i}{\sigma(z_i)}$

where

η_{ij}= link(γ_{ij})

Possible link functions are

$$\text{link}(\gamma) = \begin{cases} \log\left(\frac{\gamma}{1-\gamma}\right) & \text{Logit link} \\ \log(-\log(1-\gamma)) & \text{Complementary log-log link} \\ -\log(-\log(\gamma)) & \text{Negative Log-log link} \\ \Phi^{-1}(\gamma) & \text{Probit link} \\ \tan(\pi(\gamma - 0.5)) & \text{Cauchit (Inverse Cauchy) link} \end{cases}$$

The numerator in the right hand side of the general model specifies the location of the model, $\theta_j - b^T x_i$. In the location part of the model, **θ** is the vector of thresholds. Values of the thresholds are subject to a monotonicity property $\theta_1 \leq \ldots \leq \theta_{J-1}$. **β** is the vector of location

parameters. The denominator is the scale part of the model, $\sigma(z)$. Possible forms are:

$$\sigma(z) = \begin{cases} 1 & \text{if unity scale is assumed} \\ \exp\left(\tau^T z\right) & \text{if non-constant scale is assumed} \end{cases}$$

τ is the vector of scale parameters.

Location-Only Model

If unity scale is assumed, then the general model is said to reduce to the location-only model. The parameter **B** reduces to **B**=(**θ**^{T},**β**^{T})^T.

Log-likelihood Function

The log-likelihood of the model is

$$l = \sum_{i=1}^{m} \sum_{j=1}^{J-1} r_{ij}\varphi_{ij} - r_{i(j+1)} g(\varphi_{ij})$$

where

$$r_{ij} = \sum_{k=1}^{j} nk$$

and

$\varphi_{ij} = \log\left(\frac{\gamma_{ij}}{\gamma_{ij+1} - \gamma_{ij}}\right)$

and

$g(\varphi) = \log(1 + \exp(\varphi)) = \log\left(\frac{\gamma_{ij+1}}{\gamma_{ij+1} - \gamma_{ij}}\right)$

Note: a constant term $c = \Sigma_{i=1}^{m} \log\left\{n_i!/\left(n_{i1}!\ldots n_{iJ}!\right)\right\}$ which is independent of the unknown parameters has been excluded here. Thus, *l* is in fact the kernel of the true log-likelihood function.

Derivatives of the Log-likelihood Function

The derivatives of the log-likelihood function are used in the iterative parameter estimation algorithm.

First Derivative

The first derivative of *l* with respect to B_k, $k = 1, \ldots, (J-1)+p+q$, is

$$\frac{\partial l}{\partial B_k} = \sum_{i=1}^{m} \sum_{j=1}^{J-1} \frac{\partial l_i}{\partial \varphi_{ij}} U_{ij} Q_{ijk}$$

where

$$\frac{\partial l_i}{\partial \varphi_{ij}} = r_{ij} - r_{i(j+1)} \frac{\gamma_{ij}}{\gamma_{ij+1}}$$

$$U_{ij} = \frac{\gamma_{ij+1}}{\gamma_{ij}\left(\gamma_{ij+1} - \gamma_{ij}\right)}$$

and

$$Q_{ij} = P_{ijk} \frac{\partial \gamma_{ij}}{\partial \eta_{ij}} - P_{ij+1k} \frac{\gamma_{ij}}{\gamma_{ij+1}} \frac{\partial \gamma_{ij+1}}{\partial \eta_{ij+1}}$$

where

$$P_{ijk} = \frac{\partial \eta_{ij}}{\partial B_k} = \begin{cases} \dfrac{\delta_{jk}}{\exp\left(\mathrm{T}_{z_i}\right)} & \text{if } 1 \le k \le (J-1) \\ \dfrac{-x_{i[k-(J-1)]}}{\exp\left(\mathrm{T}_{z_i}\right)} & \text{if } (J-1)+1 \le k \le (J-1)+p \\ -z_{i[k-\{(J-1)+p\}]}\eta_{ij} & \text{if } (J-1)+p+1 \le k \le (J-1)+p+q \end{cases}$$

$\delta_{jk} = 1$ if $j = k$, 0 otherwise, and $P_{jJk} = 0$. For $i = 1, \ldots, m$, $j = 1, \ldots, J{-}1$,

$$\frac{\partial \gamma_{ij}}{\partial \eta_{ij}} = \begin{cases} \gamma_{ij}\left(1-\gamma_{ij}\right) & \text{Logit link} \\ -\left(1-\gamma_{ij}\right)\log\left(1-\gamma_{ij}\right) & \text{Complementary log-log link} \\ -\gamma_{ij}\log\left(\gamma_{ij}\right) & \text{Negative Log-log link} \\ \phi\left(\Phi^{-1}\left(\gamma_{ij}\right)\right) & \text{Probit link} \\ \cos^2\left(\pi\left(\gamma_{ij}-0.5\right)\right)/\pi & \text{Cauchit link} \end{cases}$$

and $\partial\gamma_{iJ}/\partial\eta_{iJ} = 0$.

Second Derivative

The second derivative is

$$\frac{\partial^2 l}{\partial B_s \partial B_k} = \sum_{i=1}^{m} \sum_{j=1}^{J-1} \left(\frac{\partial^2 l_i}{\partial B_s \partial \varphi_{ij}} U_{ij} Q_{ijk} + \frac{\partial l_i}{\partial \varphi_{ij}} \frac{\partial U_{ij}}{\partial B_s} Q_{ijk} + \frac{\partial l_i}{\partial \varphi_{ij}} U_{ij} \frac{\partial Q_{ijk}}{\partial B_s} \right)$$

for s, $k = 1, \ldots, (J-1) + p + q$. The first term of the equation is

$$\frac{\partial^2 l_i}{\partial B_s \partial \varphi_{ij}} U_{ij} Q_{ijk} = -\frac{r_{ij+1}}{\gamma_{ij+1}} U_{ij} Q_{ijs} Q_{ijk}$$

The second term is

$$\frac{\partial l_i}{\partial \varphi_{ij}} \frac{\partial U_{ij}}{\partial B_s} Q_{ijk} = -\left(r_{ij} - r_{ij+1}\frac{\gamma_{ij}}{\gamma_{ij+1}}\right)\left(\frac{1}{\gamma_{ij}^2} U_{ij} Q_{ijs} + \frac{1}{(\gamma_{ij+1}-\gamma_{ij})^2}\left(U_{ij+1}Q_{ij+1s} - U_{ij}Q_{ijs}\right)\right) Q_{ijk}$$

To calculate the third term, notice that

$$\begin{aligned}\frac{\partial Q_{ijk}}{\partial B_s} = {} & \frac{\partial P_{ijk}}{\partial B_s}\frac{\partial \gamma_{ij}}{\partial \eta_{ij}} + P_{ijk}\frac{\partial^2 \gamma_{ij}}{\partial B_s \partial \eta_{ij}} - \frac{\partial P_{ij+1k}}{\partial B_s}\frac{\gamma_{ij}}{\gamma_{ij+1}}\frac{\partial \gamma_{ij+1}}{\partial \eta_{ij+1}} \\ & - P_{ij+1k}\frac{Q_{ijs}}{\gamma_{ij+1}}\frac{\partial \gamma_{ij+1}}{\partial \eta_{ij+1}} - P_{ij+1l}\frac{\gamma_{ij}}{\gamma_{ij+1}}\frac{\partial^2 \gamma_{ij+1}}{\partial B_s \partial \eta_{ij+1}}\end{aligned}$$

where

$$\frac{\partial Pk}{\partial B_s} = \begin{cases} 0 & 1 \le k \le (J-1)+p \text{ and } 1 \le s \le (J-1)+p \\ -z_{i[s-\{(J-1)+p\}]}P_{ijk} & 1 \le k \le (J-1)+p \text{ and } (J-1)+p+1 \le s \le (J-1)+p+q \\ -z_{i[k-\{(J-1)+p\}]}P_{ijs} & (J-1)+p+1 \le k \le (J-1)+p+q \end{cases}$$

and $\partial P_{iJk}/\partial B_s = 0$. Moreover,

$$\frac{\partial^2 \gamma_{ij}}{\partial B_s \partial \eta_{ij}} = R_{ij}\frac{\partial \gamma_{ij}}{\partial \eta_{ij}} P_{ijs}$$

and $\partial^2 \gamma_{iJ}/\partial B_s \partial \eta_{iJ} = 0$. R_{ij} has the following form:

$$R_{ij} = \begin{cases} 1 - 2\gamma_{ij} & \text{Logit link} \\ 1 + \log(1-\gamma_{ij}) & \text{Complementary log-log link} \\ -(1 + \log \gamma_{ij}) & \text{Negative Log-log link} \\ -\phi\left(\Phi^{-1}(\gamma_{ij})\right)\Phi^{-1}(\gamma_{ij}) & \text{Probit link} \\ \sin(2\pi\gamma_{ij}) & \text{Cauchit link} \end{cases}$$

The third term can be calculated by applying these equations.

Expectation of the Second Derivative

For s, $k = 1, \ldots, (J-1) + p + q$.

$$\begin{aligned} \mathrm{E}\left(\frac{\partial^2 l}{\partial B_s \partial B_k}\right) &= \sum_{i=1}^{m}\sum_{j=1}^{J-1} \mathrm{E}\left(\frac{\partial^2 l_i}{\partial B_s \partial \varphi_{ij}} U_{ij} Q_{ijk}\right) \\ &= \sum_{i=1}^{m}\sum_{j=1}^{J-1} \mathrm{E}\left(-\frac{r_{ij+1}}{\gamma_{ij+1}} U_{ij} Q_{ijs} Q_{ijk}\right) \\ &= -\sum_{i=1}^{m} n_i \sum_{j=1}^{J-1} U_{ij} Q_{ijs} Q_{ijk} \end{aligned}$$

Parameter Estimation

Further details of parameter estimation are described here.

Maximum Likelihood Estimate

To obtain the maximum likelihood estimate of **B**, a Fisher Scoring iterative estimation method or Newton-Raphson iterative estimation method can be used. Let $\mathbf{B}^{(t)}$ be the parameter vector at iteration *t* and $\partial l/\partial\mathbf{B}^{(t)}$ be a vector of the first derivatives of *l* evaluated at $\mathbf{B}=\mathbf{B}^{(t)}$. Moreover, let $\mathbf{A}^{(t)}$ be a {(*J*–1)+*p*+*q*}×{(*J*–1)+*p*+*q*} matrix such that

$$\left[\mathbf{A}^{(t)}\right]_{sk} = \begin{cases} -\frac{\partial^2 l}{\partial B_s \partial B_k}\Big|_{\mathbf{B}=\mathbf{B}^{(t)}} & \text{Newton-Raphson approach} \\ -\mathrm{E}\left(\frac{\partial^2 l}{\partial B_s \partial B_k}\right)\Big|_{\mathbf{B}=\mathbf{B}^{(t)}} & \text{Fisher Scoring approach} \end{cases}$$

For a location-only model, the corresponding formulas use the first (*J*−1)+*p* elements of $\partial l/\partial\mathbf{B}^{(t)}$ and the upper {(*J*–1)+*p*}×{(*J*–1)+*p*} submatrix of $\mathbf{A}^{(t)}$.

The parameter vector **B** at iteration $t+1$ is updated by $\mathbf{B}^{(t+1)}$ where

$$\mathbf{A}^{(t)}\mathbf{B}^{(t+1)} = \mathbf{A}^{(t)}\mathbf{B}^{(t)} + \xi\frac{\partial l}{\partial\mathbf{B}^{(t)}}$$

and $\xi > 0$ is a stepping scalar such that $l\left(\mathbf{B}^{(t+1)}\right) - l\left(\mathbf{B}^{(t)}\right) \geq 0$.

Stepping

Use the step-halving method if $l\left(\mathbf{B}^{(t+1)}\right) - l\left(\mathbf{B}^{(t)}\right) < 0$. Let *V* be the maximum number of steps in step-halving; then the set of values of ξ is $\{1/2^v: v = 0, \ldots, V-1\}$.

Starting Values of the Parameters

Location-Only Model

If a location-only model is specified, set $\mathbf{B}^{(0)} = \left(\mathrm{q}^{(0)\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right)^{\mathrm{T}}$ where

$$\theta_j^{(0)} = \text{link}\left(\frac{\sum_{i=1}^{m}\sum_{k=1}^{j} n_{ik}}{\sum_{i=1}^{m} n_i}\right)$$

for *j* = 1, …, *J*–1.

General Model

If a general model is specified, first ignore the scale part; that is, by assuming that $\boldsymbol{\tau} = 0$ and treating the model as if it is a location-only model, and use $\mathbf{B}^{(0)} = \left(\mathrm{q}^{(0)\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right)^{\mathrm{T}}$ as the starting value to obtain the maximum likelihood estimate $\breve{\mathbf{B}}$. After $\breve{\mathbf{B}}$ is obtained, find the maximum likelihood estimate $\hat{\mathbf{B}}$ of the general model by starting at $\left(\breve{\theta}^{\mathrm{T}}, \breve{\beta}^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right)^{\mathrm{T}}$.

The above practice is essentially the same as taking $\mathbf{B}^{(0)} = \left(\mathrm{q}^{(0)\mathrm{T}}, \mathbf{0}^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right)^{\mathrm{T}}$. The advantage is that the maximum likelihood estimate $\breve{\mathbf{B}}$ can be obtained in the process of finding $\hat{\mathbf{B}}$.

Ordinal Adjustments for the Threshold Parameters

If the monotonicity property $\theta_1 \leq \ldots \leq \theta_{J-1}$ is not preserved at the end of any iteration, an ad hoc adjustment will be taken before the next iteration starts. If $\theta_j^{(t)} > \theta_{j+1}^{(t)}$ for some *j*, then both $\theta_j^{(t)}$ and $\theta_{j+1}^{(t)}$ are set to $\left(\theta_j^{(t)} + \theta_{j+1}^{(t)}\right)/2$ before the next iteration. This value is then compared with $\theta_{j+2}^{(t)}$ and so on.

Convergence Criteria

Given convergence criteria $\epsilon_k > 0$ and $\epsilon_p > 0$, the iteration is considered to be converged if one of the following criteria are satisfied:

$$\left| l\left(\mathbf{B}^{(t+1)}\right) - l\left(\mathbf{B}^{(t)}\right) \right| < \epsilon_k$$

$$\max_i \left| \mathbf{B}_i^{(t+1)} - \mathbf{B}_i^t \right| < \epsilon_p$$

Statistics

The following statistics are available.

Model Information

The model information is the –2 log-likelihood of the model, computed for a given vector of parameter estimates.

Final Model, General

The value of –2 log-likelihood of the model is given by

$$-2l\left(\hat{\mathbf{B}}\right)$$

where $l\left(\hat{\mathbf{B}}\right)$ is the value of the log-likelihood evaluated at $\hat{\mathbf{B}}$.

Final Model, Location-Only

If unity scale is assumed, the general model reduces to the location-only model. The value of –2 log-likelihood of the model is given by

$$-2l\left(\breve{\mathbf{B}}\right)$$

Initial Model, Intercept-Only

In the initial model, when the intercepts are the only parameters in the model, the parameter vector is $\mathbf{B}^{(0)} = \left(\mathrm{q}^{(0)\,\mathrm{T}}, \mathbf{0}^{\mathrm{T}}, \mathbf{0}^{\mathrm{T}}\right)^{\mathrm{T}}$. The value of the –2 log-likelihood is

$$-2l\left(\mathbf{B}^{(0)}\right)$$

Model Chi-Square

The value of the Model Chi-square statistic is given by the difference between any two nesting models of interest.

General Model versus Intercept-Only Model

The following statistic is available when a general model is specified. The Model Chi-square statistic is given by

$$-2l\left(\mathbf{B}^{(0)}\right) - 2l\left(\hat{\mathbf{B}}\right)$$

Under that null hypothesis that H_0 : b$=\mathbf{0}$ and t $=\mathbf{0}$, the Model Chi-square is asymptotically chi-squared distributed with $\hat{e} - (J-1)$ degrees of freedoms.

Location-Only Model versus Intercept-Only Model

The following statistic is available when a location-only model is specified. The Model Chi-square statistic is given by

$$-2l\left(\mathbf{B}^{(0)}\right) - 2l\left(\breve{\mathbf{B}}\right)$$

Under that null hypothesis that H_0 : b $=\mathbf{0}$, the Model Chi-square is asymptotically chi-squared distributed with $\breve{e} - (J-1)$ degrees of freedoms.

General Model versus Location-Only Model

The following statistic is available when a general model is specified. The Model Chi-square statistic is given by

$$-2l\left(\breve{\mathbf{B}}\right) - 2l\left(\hat{\mathbf{B}}\right)$$

Under that null hypothesis that H_0 : t$=\mathbf{0}$, the Model Chi-square is asymptotically chi-squared distributed with $\hat{e} - \breve{e}$ degrees of freedoms.

Likelihood Ratio Test for Equal Slopes Assumption

For location-only model, a likelihood ratio test of parallel lines in the location is performed. If the regression lines are not parallel, the location can be specified as

$$\eta_{ij} = \theta_j - \mathrm{b}_j x_i$$

for $j = 1, \ldots, J-1$. That is, the location parameters b_j (or slopes) vary with the levels of the response. The parameter for the above "non-parallel" location-only model is $\mathbf{B} = \left(\mathrm{q}^{\mathrm{T}}, \mathrm{b}_j^{\mathrm{T}}, \ldots, \mathrm{b}_{J-1}^{\mathrm{T}}\right)^{\mathrm{T}}$ which is of dimension $\{(J-1)+(J-1)p\}\times 1$. The first derivative $\partial l / \partial \mathbf{B}$ of the log-likelihood is the same as in the "parallel" model, except that $P_{ijk} = \partial \eta_{ij} / \partial B_k$ is replaced by the following:

$$P_{ijk} = \frac{\partial \eta_{ij}}{\partial B_k} = \begin{cases} \delta_{jk} & 1 \le k \le (J-1) \\ -x_{i[k-\{(J-1)+sp\}]} & (J-1)+sp \le k \le (J-1)+sp+p, s = 1, \ldots, (J-2) \end{cases}$$

Similarly, the expected value of the second derivative is the same as in the parallel model, except that the P_{ijk} is replaced by the above equation.

To test the null hypothesis of parallelism $H_0 : \mathrm{b}_1 = \ldots = \mathrm{b}_{J-1}$, find the maximum likelihood estimate $\breve{\mathbf{B}}$ of the parallel location-only model and the maximum likelihood estimate $\breve{\breve{\mathbf{B}}}$ of the non-parallel model. The Model Chi-square statistic is given by

$$-2l\left(\breve{\mathbf{B}}\right) - 2l\left(\breve{\breve{\mathbf{B}}}\right)$$

Under the null hypothesis, the Model Chi-square statistic is asymptotically chi-squared distributed with $(k-2)p$ degrees of freedoms.

Pseudo R-Squares

Replace $\hat{\mathbf{B}}$ by $\breve{\mathbf{B}}$ for a location-only model in the equations below.

Cox and Snell's R-Square

$$R^2_{\mathrm{CS}} = 1 - \left(\frac{L\left(\mathbf{B}^{(0)}\right)}{L\left(\hat{\mathbf{B}}\right)}\right)^{\frac{2}{n}}$$

Nagelkerke's R-Square

$$R^2_{\mathrm{N}} = \frac{R^2_{\mathrm{CS}}}{1 - L\left(\mathbf{B}^{(0)}\right)^{2/n}}$$

McFadden's R-Square

$$R^2_{\mathrm{M}} = 1 - \left(\frac{l\left(\hat{\mathbf{B}}\right)}{l\left(\mathbf{B}^{(0)}\right)}\right)$$

Predicted Cell Counts

The estimated cell response probability based on the maximum likelihood estimate for the general model is

$$\hat{\pi}_{ij} = \begin{cases} \hat{\gamma}_{i1} & j = 1 \\ \hat{\gamma}_{ij} - \hat{\gamma}_{ij-1} & j = 2, ..., J-1 \\ 1 - \hat{\gamma}_{iJ-1} & j = J \end{cases}$$

At each subpopulation i, the predicted count for response category *Y=j* is

$$\hat{n}_{ij} = n_i \hat{\pi}_{ij}$$

The (raw) residual is $n_{ij} - \hat{n}_{ij}$ and the standardized residual is $(n_{ij} - \hat{n}_{ij}) / \sqrt{n_i \hat{\pi}_{ij} (1 - \hat{\pi}_{ij})}$.

Replace $\hat{\gamma}_{ij}$ by $\breve{\gamma}_{ij}$, $\hat{\pi}_{ij}$ by $\breve{\pi}_{ij}$, and $\hat{n}_{ij}$ by $\breve{n}_{ij}$ for a location-only model.

Predicted Cumulative Totals

The predicted cumulative total up to and including *Y=j* is

$$\hat{r}_{ij} = n_i \hat{\gamma}_{ij}$$

The (raw) residual is $r_{ij} - \hat{r}_{ij}$ and the standardized residual is $(r_{ij} - \hat{r}_{ij}) / \sqrt{n_i \hat{\gamma}_{ij} (1 - \hat{\gamma}_{ij})}$.

Replace $\hat{\gamma}_{ij}$ by $\breve{\gamma}_{ij}$ and $\hat{r}_{ij}$ by $\breve{r}_{ij}$ for a location-only model.

Goodness of Fit Measures

These are chi-square statistics used to test whether the model adequately fits the data.

Pearson Goodness of Fit Measure

The Pearson goodness of fit measure for a general model is

$$X^2 = \sum_{i=1}^{m} \sum_{j=1}^{J} \frac{(n_{ij} - n_i \hat{\pi}_{ij})^2}{n_i \hat{\pi}_{ij}}$$

Under the null hypothesis, the Pearson goodness-of-fit statistic is asymptotically chi-squared distributed with $m(J-1) - \hat{e}$ degrees of freedom.

Replace $\hat{\pi}_{ij}$ by $\breve{\pi}_{ij}$ and $\hat{e}$ by $\breve{e}$ for a location-only model.

Deviance Goodness of Fit Measure

The Deviance goodness of fit measure for a general model is

$$D = 2 \sum_{i=1}^{m} \sum_{j=1}^{J} n_{ij} \log \left(\frac{n_{ij}}{n_i \hat{\pi}_{ij}} \right)$$

Under the null hypothesis, the Deviance goodness-of-fit statistic is asymptotically chi-squared distributed with $m(J-1) - \hat{e}$ degrees of freedom.

Replace $\hat{\pi}_{ij}$ by $\breve{\pi}_{ij}$ and $\hat{e}$ by $\breve{e}$ for a location-only model.

Covariance and Correlation Matrices

The estimate of the covariance matrix of $\hat{\mathbf{B}}$ is

$$\mathrm{Cov}\left(\hat{\mathbf{B}}\right) = \begin{cases} -\frac{\partial^2 l}{\partial \mathbf{B} \partial \mathbf{B}}\Big|_{\mathbf{B}=\hat{\mathbf{B}}} & \text{Newton-Raphson method} \\ -\mathrm{E}\left(\frac{\partial^2 l}{\partial \mathbf{B} \partial \mathbf{B}}\right)\Big|_{\mathbf{B}=\hat{\mathbf{B}}} & \text{Fisher Scoring method} \end{cases}$$

Let be the $\{(J-1)+p+q\}\times 1$ vector of the square roots of the diagonal elements in $\hat{\mathbf{B}}\Big)$. The estimate of the correlation matrix of $\hat{\mathbf{B}}$ is

Replace $\hat{\mathbf{B}}$ by $\breve{\mathbf{B}}$ and by (a $\{(J-1)+p\}\times 1$ vector) for a location-only model.

Parameter Statistics

An estimate of the standard deviation of $\hat{B}_k$ is $\hat{\sigma}_k$. The Wald statistic for $\hat{B}_k$ is

$$\mathrm{Wald}_k = \frac{\hat{B}_k}{\hat{\sigma}_k}$$

Under the null hypothesis that $H_0 : B_k = 0$, Wald $_k$ is asymptotically chi-squared distributed with 1 degree of freedom.

Based on the asymptotic normality of the parameter estimate, a 100(1−α) % Wald confidence interval for $\hat{B}_k$ is

$$\hat{B}_k \pm z_{1-\alpha/2}\hat{\sigma}_k$$

where $z_{1-\alpha/2}$ is the upper (1−α /2)100th percentile of the standard normal distribution.

Replace $\hat{B}_k$ by $\breve{B}_k$ and $\hat{\sigma}_k$ by $\breve{\sigma}_k$ for a location-only model.

Linear Hypothesis Testing

For a general model, let $\mathbf{L}$ be a matrix of coefficients for the linear hypotheses

$$H_0 : \mathbf{LB} = \mathbf{c}$$

where $\mathbf{c}$ is a $k\times 1$ vector of constants. The Wald statistic for H_0 is

$$\mathrm{Wald}\,(\mathbf{L}, \mathbf{c}) = \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{c}\right)^{\mathrm{T}} \left\{\mathbf{L}\mathrm{Cov}\left(\hat{\mathbf{B}}\right)\mathbf{L}^{\mathrm{T}}\right\}^{-1} \left(\mathbf{L}\hat{\mathbf{B}} - \mathbf{c}\right)$$

Under the null hypothesis, Wald $(\mathbf{L}, \mathbf{c})$ is asymptotically chi-squared distributed with l degrees of freedom, where l is the rank of $\mathbf{L}$.

Replace $\hat{\mathbf{B}}$ by $\breve{\mathbf{B}}$ for a location-only model.

References

Cox, D. R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 34, 187–220.

Goodman, L. A. 1979. Simple models for the analysis of association in cross-classifications having ordered categories. *Journal of the American Statistical Association*, 74, 537–552.

Goodman, L. A. 1981. Association Models and Canonical Correlation in the Analysis of Cross-Classifications having Ordered Categories. *Journal of American Statistical Association*, 76, 320–334.

Greenland, S. 1994. Alternative Models for Ordinal Logistic Regression. *Statistics in Medicine* , 13, 1665–1677.

Hosmer, D. W. J., and S. Lemeshow. 1981. Applied Logistic Regression Models. *Biometrics*, 34, 318–327.

Magidson, J. 1995. Introducing a New Graphical Model for the Analysis of an Ordinal Categorical Response – Part I. *Journal of Targeting, Measurement and Analysis for Marketing*, 4:2, 133–148.

McCullagh, P. 1980. Regression Models for Ordinal Data. *Journal of the Royal Statistical Society B*, 42:2, 109–142.

Pregibon, D. 1981. Logistic Regression Diagnostics. *Annals of Statistics*, 9, 705–724.

Williams, D. A. 1982. Extra-Binomial Variation in Logistic Linear Models. *Applied Statistics*, 31, 144–148.

PPLOT Algorithms

PPLOT produces probability plots of one or more sequence or time series variables. The variables can be standardized, differenced, and/or transformed before plotting. Expected normal values or deviations from expected normal values can be plotted. PPLOT can be used to investigate whether the data are from a specified distribution: normal, lognormal, logistic, exponential, Weibull, gamma, beta, uniform, Pareto, Laplace, half normal, chi-square and Student's *t*.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 78-1
Notation

Notation	Description
$\overline{X}$	Sample mean
S	Sample standard deviation
$\overline{LX}$	Sample mean for $\ln(x_i)$
LS	Sample standard deviation for $\ln(x_i)$
x_i	Value of the *i*th observation
$x_{(i)}$	The *i*th smallest observation
R_i	Corresponding rank for x_i
n	Sample size
$fr_{dist}(x_i)$	Fractional rank of x_i for the specified distribution function
$a_{dist}(x_i)$	Score for the specified distribution function
α	Location parameter
β	Scale parameter
γ	Shape parameter
ν	Degrees of freedom

Fractional Ranks

Based on the rank R_i for the observation x_i, the fractional rank $fr_{dist}(x_i)$ is computed and used to estimate the expected cumulative distribution function of *X*. One of four methods can be selected to calculate the fractional rank $fr_{dist}(x_i)$:

$$fr_{dist}(x_i) = \begin{cases} \left(R_i - \frac{3}{8}\right) / \left(n + \frac{1}{4}\right) & \text{Blom} \\ \left(R_i - \frac{1}{2}\right) / n & \text{Rankit} \\ \left(R_i - \frac{1}{3}\right) / \left(n + \frac{1}{3}\right) & \text{Tukey} \\ R_i / (n+1) & \text{VanderWaerden} \end{cases}$$

Scores

The score of the specified distribution for case *i* is defined as

$$a_{dist}(x_i) = F_{dist}^{-1}(fr_{dist}(x_i)) \quad i = 1, \ldots, n$$

where F_{dist} is the inverse cumulative specified distribution function.

P-P Plot

For a P-P plot, the fractional rank and the cumulative specified distribution function F_{dist} are plotted:

$$(fr_{dist}(x_i), F_{dist}(x_i)) \quad i = 1, \ldots, n$$

Q-Q Plot

For a Q-Q plot, the observations and the score for the specified distribution function are plotted.

$$(x_i, a_{dist}(x_i)) \quad i = 1, \ldots, n$$

Distributions

The distributions and their parameters are listed below. Parameters may be either specified by users or estimated from the data. Any parameter values specified by the user should satisfy the conditions indicated.

Table 78-2
Distributions

Distribution	**Description**
Beta(β_1,β_2)	$\beta_1(>0)$ and $\beta_2(>0)$ are scale parameters.
Chi-square(ν)	$\nu(>0)$ is the degrees of freedom.
Exponential(β)	$\beta(>0)$ is a scale parameter.
Gamma(γ, β)	$\gamma(>0)$ is a shape parameter and $\beta(>0)$ is the scale parameter.
Half Normal(β)	$\beta(>0)$ is a scale parameter and the location parameter is 0.
Laplace(α, β)	α is the location parameter and $\beta(>0)$ is the scale parameter.
Logistic(α, β)	α is the location parameter and $\beta(>0)$ is the scale parameter.
Lognormal(β, γ)	$\beta(>0)$ is a scale parameter and $\gamma(>0)$ is a shape parameter.
Normal(α, β)	α is the location parameter and $\beta(>0)$ is the scale parameter.
Pareto(β,b):	$\beta(>0)$ is scale parameter and $b(>0)$ is an index of inequality.
Student's $t(\nu)$	$v(>0)$ is the degrees of freedom specified by the user.
Uniform(a,b)	a is a lower bound and b is an upper bound.
Weibull(β,γ)	$\beta(>0)$ is a scale parameter and $\gamma(>0)$ is a shape parameter.

Estimates of the Parameters

The estimates for parameters for each distribution are defined below.

Table 78-3
Parameter estimates for distributions

Distribution	Description	Parameter type
Beta(β_1,β_2)	$\hat{\beta}_1 = \overline{X}\left\{\frac{\overline{X}(1-\overline{X})}{S^2} - 1\right\}$	scale parameter
	$\hat{\beta}_2 = (1-\overline{X})\left\{\frac{\overline{X}(1-\overline{X})}{S^2} - 1\right\}$	scale parameter
Chi-square(ν)	ν is the degrees of freedom specified by the user.	
Exponential(β)	$\hat{\beta} = \frac{1}{\overline{X}}$	scale parameter
Gamma(γ, β)	$\hat{\gamma} = \frac{\overline{X}^2}{S^2}$	shape parameter
	$\hat{\beta} = \frac{\overline{X}}{S^2}$	scale parameter
Half Normal(β)	$\hat{\beta} = \sqrt{x_1^2 + ... + x_n^2}$	scale parameter
Laplace(α, β)	$\hat{\alpha} = \overline{X}$	location parameter
	$\hat{\beta} = \sqrt{\frac{S^2}{2}}$	scale parameter
Logistic(α, β)	$\hat{\alpha} = \overline{X}$	location parameter
	$\hat{\beta} = \sqrt{3}(\frac{S}{\pi}), \ \pi = 3.1415927$	scale parameter
Lognormal	$\hat{\beta} = \exp(L\overline{X})$	scale parameter
	$\hat{\gamma} = LS$	shape parameter
Normal(α, β)	$\hat{\alpha} = \overline{X}$	location parameter
	$\hat{\beta} = S$	scale parameter
Pareto(β,b);	$\hat{\beta} = \min\{x_1, \ldots, x_n\}$	scale parameter
	$\hat{b} = \frac{1}{L\overline{X} - \ln(\hat{\beta})}$	index of inequality
Student's t(ν)	ν is the degrees of freedom specified by the user.	
Uniform(a,b)	$\hat{a} = \min\{x_1, \ldots, x_n\}$	lower bound
	$\hat{b} = \max\ \{x_1, \ldots, x_n\}$	upper bound
Weibull(β,γ)	$\hat{\beta} = \frac{\sum_{i=1}^{n} U_i Y_i - n\overline{UY}}{\sum_{i=1}^{n} (U_i - \overline{U})^2}$	scale parameter

Distribution	Description	Parameter type
	$\hat{\gamma} = \exp\left(-\left(\left(\overline{Y} - \hat{\beta}\overline{U}\right)/\hat{\beta}\right)\right)$ where $Y_i = \ln(-\ln(1 - fr_{dist}(x_i)))$ and $U_i = \ln(x_i)$	shape parameter

References

Kotz, S., and N. L. Johnson, eds. 1988. *Encyclopedia of statistical sciences.* New York: John Wiley & Sons, Inc.

PRINCALS Algorithms

The PRINCALS algorithm was first described in Van Rijckevorsel and De Leeuw (1979) and De Leeuw and Van Rijckevorsel (1980); also see Gifi (1981, 1985). Characteristic features of PRINCALS are the ability to specify any of a number of measurement levels for each variable separately and the treatment of missing values by setting weights in the loss function equal to 0.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	Number of cases (objects)
m	Number of variables
p	Number of dimensions

For variable *j*; $j = 1, \ldots, m$

h_j	n-vector with categorical observations
k_j	Number of valid categories (distinct values) of variable *j*
$\mathbf{G}_j$	Indicator matrix for variable *j*, of order $n \times k_j$

$$g_{(j)ir} = \begin{cases} 1 & \text{when the } i\text{th object is in the } r\text{th category of variable } j \\ 0 & \text{when the } i\text{th object is not in the } r\text{th category of variable } j \end{cases}$$

$\mathbf{D}_j$	Diagonal matrix, containing the univariate marginals; that is, the column sums of $\mathbf{G}_j$
$\mathbf{M}_j$	Binary diagonal $n \times n$ matrix, with diagonal elements defined as

$$m_{(j)ii} = \begin{cases} 1 & \text{when the } i\text{th observation is within the range } [1, k_j] \\ 0 & \text{when the } i\text{th observation outside the range } [1, k_j] \end{cases}$$

The quantification matrices and parameter vectors are:

$\mathbf{X}$	Object scores, of order $n \times p$
$\mathbf{Y}_j$	Multiple category quantifications, of order $k_j \times p$
$\mathbf{y}_j$	Single category quantifications, of order k_j
$\mathbf{a}_j$	Variable weights (equal to component loadings), of order p
$\mathbf{Q}$	Transformed data matrix of order $n \times m$ with columns $\mathbf{q}_j = \mathbf{G}_j \mathbf{y}_j$
$\underline{\mathbf{Y}}$	Collection of multiple and single category quantifications.

Note: The matrices $\mathbf{G}_j$, $\mathbf{M}_j$, and $\mathbf{D}_j$ are exclusively notational devices; they are stored in reduced form, and the program fully profits from their sparseness by replacing matrix multiplications with selective accumulation.

Objective Function Optimization

The PRINCALS objective is to find object scores **X** and a set of $\underline{\mathbf{Y}}_j$ (for *j*=1,...,*m*) — the underlining indicates that they may be restricted in various ways — so that the function

$$\sigma(\mathbf{X};\underline{\mathbf{Y}}) = 1/m\Sigma_j \mathrm{tr}\Big((\mathbf{X}-\mathbf{G}_j\underline{\mathbf{Y}}_j)' M_j(\mathbf{X}-\mathbf{G}_j\underline{\mathbf{Y}}_j)\Big)$$

is minimal, under the normalization restriction $\mathbf{X}'\mathrm{M}_*\mathbf{X} = mn\mathbf{I}$ where $\mathrm{M}_* = \sum_j \mathrm{M}_j$ and **I** is the *p*×*p* identity matrix. The inclusion of M_j in $\sigma(\mathbf{X};\underline{\mathbf{Y}})$ ensures that there is no influence of data values outside the range $[1,k_j]$, a circumstance that may indicate either genuine missing values or simulated missing values for the sake of analysis. M_* contains the number of "active" data values for each object. The object scores are also centered; that is, they satisfy $\mathbf{u}'\mathrm{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ with **u** denoting an *n*-vector with ones.

Optimal Scaling Levels

The following optimal scaling levels are distinguished in PRINCALS:

Multiple Nominal. $\underline{\mathbf{Y}}_j = \mathbf{Y}_j$ (equality restriction only).

(Single) Nominal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ (equality and rank – one restrictions).

(Single) Ordinal. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{C}_j$ (equality, rank – one, and monotonicity restrictions). The monotonicity restriction $\mathbf{y}_j \in \mathbf{C}_j$ means that $\mathbf{y}_j$ must be located in the convex cone of all k_j-vectors with nondecreasing elements.

(Single) Numerical. $\underline{\mathbf{Y}}_j = \mathbf{y}_j\mathbf{a}'_j$ and $\mathbf{y}_j \in \mathbf{L}_j$ (equality, rank – one, and linearity restrictions). The linearity restriction $\mathbf{y}_j \in \mathbf{L}_j$ means that $\mathbf{y}_j$ must be located in the subspace of all k_j-vectors that are a linear transformation of the vector consisting of k_j successive integers.

For each variable, these levels can be chosen independently. The general requirement for all options is that equal category indicators receive equal quantifications. The general requirement for the non-multiple options is $\underline{\mathbf{Y}}_j = \mathrm{y}_j\mathrm{a}'_j$; that is, $\underline{\mathbf{Y}}_j$ is of rank one; for identification purposes, y_j is always normalized so that $\mathrm{y}'_j\mathbf{D}_j\mathrm{y}_j = n_w$.

Optimization

Optimization is achieved by executing the following iteration scheme:

1. Initialization I or II
2. Update object scores
3. Orthonormalization
4. Update category quantifications
5. Convergence test: repeat (2) through (4) or continue

6. Rotation

Steps (1) through (6) are explained below.

Initialization

I. Random

The object scores $\mathbf{X}$ are initialized with random numbers. Then $\mathbf{X}$ is normalized so that $\mathbf{u}'\mathrm{M}_*\mathbf{W}\mathbf{X} = \mathbf{0}$ and $\mathbf{X}'\mathrm{M}_*\mathbf{X} = mn\mathbf{I}$, yielding $\tilde{\mathbf{X}}$. For multiple variables, the initial category quantifications are obtained as $\tilde{\mathbf{Y}}_j = \mathbf{D}_j^{-1}\mathbf{G}'_j\tilde{\mathbf{X}}$. For single variables, the initial category quantifications $\tilde{\mathbf{y}}_j$ are defined as the first k_j successive integers normalized in such a way that $\mathbf{u}'\mathbf{D}_j\tilde{\mathbf{y}}_j = 0$ and $\tilde{\mathbf{y}}_j\mathbf{D}_j\tilde{\mathbf{y}}_j = n$, and the initial variable weights are calculated as the vector $\tilde{\mathbf{a}}_j = \tilde{\mathbf{X}}'\mathbf{G}_j\tilde{\mathbf{y}}_j$, rescaled to unit length.

II. All relevant quantities are copied from the results of the first cycle.

Update object scores

First the auxiliary score matrix $\mathbf{Z}$ is computed as

$$\mathbf{Z} \leftarrow \Sigma_j\mathbf{M}_j\mathbf{G}_j\tilde{\mathbf{Y}}_j$$

and centered with respect to M_*:

$$\tilde{\mathbf{Z}} \leftarrow \left\{\mathbf{M}_* - \left(\mathbf{M}_*\mathbf{u}\mathbf{u}'\mathbf{M}_*/\mathbf{u}'\mathbf{M}_*\mathbf{u}\right)\right\}\mathbf{Z}$$

These two steps yield locally the best updates when there would be no orthogonality constraints.

Orthonormalization

The problem is to find an M_*-orthonormal $\mathbf{X}^+$ that is closest to $\tilde{\mathbf{Z}}$ in the least squares sense. In PRINCALS, this is done by setting

$$\mathbf{X}^+ \leftarrow m^{1/2}\mathbf{M}_*^{-1/2}\text{GRAM}\left(\mathbf{M}_*^{-1/2}\tilde{\mathbf{Z}}\right)$$

which is equal to the genuine least squares estimate up to a rotation—see (6). The notation GRAM( ) is used to denote the Gram-Schmidt transformation (Björk and Golub, 1973).

Update category quantifications

For multiple nominal variables, the new category quantifications are computed as:

$$\mathbf{Y}_j^+ = \mathbf{D}_j^{-1}\mathbf{G}'_j\tilde{\mathbf{X}}$$

For single variables one cycle of an ALS algorithm (De Leeuw et al., 1976) is executed for computing the rank-one decomposition of $\tilde{Y}_j$, with restrictions on the left-hand vector. This cycle starts from the previous category quantification $\tilde{\mathbf{y}}_j$ with

$$\mathbf{a}_j^{+} = \tilde{\mathbf{Y}}'_j \mathbf{D}_j \tilde{\mathbf{y}}_j$$

When the current variable is numerical, we are ready; otherwise we compute

$$\mathbf{y}_j^{*} = \tilde{\mathbf{Y}}_j \mathbf{a}_j^{+}.$$

Now, when the current variable is single nominal, you can simply obtain $\mathbf{y}_j^{+}$ by normalizing $\mathbf{y}_j^{*}$ in the way indicated below; otherwise the variable must be ordinal, and you have to insert the weighted monotonic regression process

$$\mathbf{y}_j^{*} \leftarrow \text{WMON}(\mathbf{y}_j^{*}) .$$

The notation WMON( ) is used to denote the weighted monotonic regression process, which makes $\mathbf{y}_j^{*}$ monotonically increasing. The weights used are the diagonal elements of $\mathbf{D}_j$ and the subalgorithm used is the up-and-down-blocks minimum violators algorithm (Kruskal, 1964; Barlow et al., 1972). The result is normalized:

$$\mathbf{y}_j^{+} = n_w^{1/2} \mathbf{y}_j^{*} \left(\mathbf{y}'^{*}_j \mathbf{D}_j \mathbf{y}_j^{*} \right)^{-1/2}$$

Finally, we set $\underline{\mathbf{Y}}_j^{+} = \mathbf{y}_j^{+} \mathbf{a}'^{+}_j$.

Convergence test

The difference between consecutive values of the quantity

$$\text{TFIT} = 1/m \sum_s \left[\sum_{j \in J} \mathbf{y}'_{(j)s} \mathbf{D}_j \mathbf{y}_{(j)s} + \sum_{j \notin J} \mathbf{a}'_j \mathbf{a}_j \right]$$

where $\mathbf{y}_{(j)s}$ denotes the *s*th column of $\mathbf{Y}_j$ and *J* is an index set recording which variables are multiple, is compared with the user-specified convergence criterion ε - a small positive number. It can be shown that TFIT $= \mathbf{p} - \sigma(\mathbf{X}; \underline{\mathbf{Y}})$. Steps (2) through (4) are repeated as long as the loss difference exceeds ε.

Rotation

As remarked in (3), during iteration the orientation of **X** and **Y** with respect to the coordinate system is not necessarily correct; this also reflects that $\sigma(\mathbf{X}; \underline{\mathbf{Y}})$ is invariant under simultaneous rotations of **X** and **Y**. From the theory of principal components, it is known that if all variables would be single, the matrix **A** — which can be formed by stacking the row vectors $\mathbf{a}'_j$—has the property that **A'A** is diagonal. Therefore you can rotate so that the matrix

$$1/m \mathbf{A}' \mathbf{A} = 1/m \sum_j \mathbf{a}_j \mathbf{a}'_j = 1/m \sum_j \mathbf{Y}'_j \mathbf{D}_j \mathbf{Y}_j$$

becomes diagonal. The corresponding eigenvalues are printed after the convergence message of the program. The calculation involves tridiagonalization with Householder transformations followed by the implicit QL algorithm (Wilkinson, 1965).

Diagnostics

The following diagnostics are available.

Maximum Rank (may be issued as a warning when exceeded)

The maximum rank $p_{\max}$ indicates the maximum number of dimensions that can be computed for any dataset. In general

$$p_{\max} = \min\left\{(n-1), \left(\left(\sum_{j\in J} k_j + m_2\right) - \max\left(m_1,\ \max\left(0, 1-m_2\right)\right)\right)\right\}$$

where m_1 is the number of multiple variables with no missing values, m_2 is the number of single variables, and J is an index set recording which variables are multiple. Although the number of nontrivial dimensions may be less than $p_{\max}$ when m=2, PRINCALS does allow dimensionalities all the way up to $p_{\max}$. When, due to empty categories in the actual data, the rank deteriorates below the specified dimensionality, the program stops.

Marginal Frequencies

The frequencies table gives the univariate marginals and the number of missing values (that is, values that are regarded as out of range for the current analysis) for each variable. These are computed as the column sums of $\mathbf{D}_j$ and the total sum of $\mathbf{M}_j$.

Fit and Loss Measures

When the HISTORY option is in effect, the following fit and loss measures are reported:

Total fit. This is the quantity TFIT defined in (5).

Total loss. This is $\sigma(\mathbf{X};\underline{\mathbf{Y}})$, computed as the sum of multiple loss and single loss defined below.

Multiple loss. This measure is computed as

$$\text{TMLOSS} = p - 1/m\Sigma_j\text{tr}\left[\mathbf{Y}'_j\mathbf{D}_j\mathbf{Y}_j\right]$$

Single loss. This measure is computed only when some of the variables are single:

$$\text{SLOSS} = 1/m\sum_{j\notin J}\text{tr}\left[\mathbf{Y}'_j\mathbf{D}_j\mathbf{Y}_j\right] + \sum_{j\notin J}\mathbf{a}'_j\mathbf{a}_j$$

Eigenvalues and Correlations between Optimally Scaled Variables

If there are no missing data, the eigenvalues printed by PRINCALS are those of $1/m\mathbf{R}(\mathbf{Q})$, where **R**(**Q**) denotes the matrix of correlations between the optimally scaled variables in the columns of **Q**. For multiple variables, $\mathbf{q}_j$ is defined here as $\mathbf{G}_j\mathbf{y}_{(j)1}$. When all variables are single or when p=1, **R**(**Q**) itself is also printed. If there are missing data, then the eigenvalues are those of the matrix with elements $\mathbf{q}'_j \mathbf{M}_*^{-1} \mathbf{q}_1$, which is not necessarily a correlation matrix, although it is positive semidefinite.

References

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions.* New York: John Wiley and Sons.

Björk, A., and G. H. Golub. 1973. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27, 579–594.

De Leeuw, J., and J. Van Rijckevorsel. 1980. HOMALS and PRINCALS—Some generalizations of principal components analysis. In: *Data Analysis and Informatics*, E. Diday,et al., ed. Amsterdam: North-Holland, 231–242.

De Leeuw, J., F. W. Young, and Y. Takane. 1976. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41, 471–503.

Gifi, A. 1990. *Nonlinear multivariate analysis.* Chichester: John Wiley and Sons.

Gifi, A. 1985. *PRINCALS. Research Report UG-85-02*. Leiden: Department of Data Theory, University of Leiden.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Van Rijckevorsel, J., and J. De Leeuw. 1979. *An outline of PRINCALS: Internal Report RB 002–'79*. Leiden: Department of Data Theory, University of Leiden.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem.* Oxford: Clarendon Press.

PROBIT Algorithms

The Probit procedure is used to estimate the effects of one or more independent variables on a dichotomous dependent variable. The program is designed for dose-response analyses and related models, but Probit can also estimate logistic regression models.

Notation

The following notation is used throughout this chapter unless otherwise stated:

m	Number of covariate patterns
n_i	Number of subjects for ith covariate pattern
r_i	Number of responses for ith covariate pattern
p	Number of independent variables
q	Number of levels of the grouping variable. q=0 when there is no grouping variable
c	Natural response rate
X	$n \times (p+q)$ matrix with element x_{ij}, which represents the *j*th covariate for the *i*th covariate pattern
$\boldsymbol{\gamma}$	p×1 vector with element γ_j, which represents the slope parameter of the *j*th independent variable
$\boldsymbol{\alpha}$	q×1 vector with element α_j, which represents the parameter for the *j*th level of the grouping variable
$\boldsymbol{\beta}$	$(p+q) \times 1$ vector which is a composite of $\boldsymbol{\gamma}$ and $\boldsymbol{\alpha}$
s	Total number of parameters in the model, equal to p+q if the natural response rate is set to a constant, p+q+1 if the natural response rate is to be estimated by the model

Model

The model assumes a dichotomous dependent variable with probability *P* for the event of interest. Since the procedure assumes aggregated data for every covariate pattern, the random variable y_i takes a binomial distribution.

$$P(y_i = r_i) = \binom{n_i}{r_i} P_i^{r_i} (1 - P_i)^{n_i - r_i} \quad i = 1, \ldots, m$$

Hence, the log likelihood, *L*, for *m* observations after ignoring the constant factor can be written as

$$L = \sum_{i=1}^{m} r_i \ln P_i + (n_i - r_i) \ln (1 - P_i)$$

For dose-response models, it is further assumed that

$$P_i = c + (1 - c) F\left(\mathbf{X}_i' \beta\right)$$

where $\mathbf{X}_i'$ is the vector of covariates for the *i*th covariate pattern and $F\left(\mathbf{X}_i'\beta\right)$ has two forms:

$$F\left(\mathbf{X}_i'\beta\right) = \begin{cases} \frac{e^{\mathbf{X}_i'\beta}}{1+e^{\mathbf{X}_i'\beta}} & \text{if logit model} \\ \int_{-\infty}^{\mathbf{X}_i'\beta} \frac{1}{\sqrt{2\pi}} e^{-z^2/2} \mathrm{dz} & \text{if probit model} \end{cases}$$

When there is no grouping variable, x_{ij} is simply the observed value of the *j*th independent variable for the *i*th covariate pattern, and **β**=**γ**. When there is a grouping variable, a set of indicator variables is constructed. There will be *q* indicator variables $l_{i1}, \ldots, l_{iq}$ added to the **X** matrix and *q* parameters $\alpha_1, \ldots, \alpha_q$ added to the **β** vector.

$$l_{ij} = \begin{cases} 1 & \text{if the } i\text{th covariate pattern is in the } j\text{th level} \\ 0 & \text{otherwise} \end{cases}$$

Hence, the $\mathbf{X}_i$ vector has *p*+*q* elements and the associated parameter vector **β** is expanded to $(\beta_1, \ldots, \beta_p, \beta_{p+1}, \ldots, \beta_{p+q})$, where $\alpha_j = \beta_{p+j}$.

Maximum-Likelihood Estimates (MLE)

To obtain the maximum likelihood estimates for *c*, and $\beta_1, \ldots, \beta_{p+q}$, set the following equations equal to 0:

$$L_c^* = \sum_{i=1}^{m} \frac{r_i - n_i P_i}{P_i(1-P_i)} \left[1 - F\left(\mathbf{X}_i'\beta\right)\right]$$

$$L_{\beta_j}^* = \begin{cases} (1-c)\sum_{i=1}^{m} \frac{r_i - n_i P_i}{P_i(1-P_i)} x_{ij} F\left(\mathbf{X}_i'\beta\right)\left(1 - F\left(\mathbf{X}_i'\beta\right)\right) & \text{if logit model} \\ (1-c)\sum_{i=1}^{m} \frac{r_i - n_i P_i}{P_i(1-P_i)} x_{ij} \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\mathbf{X}_i'\beta\right)^2\right\} & \text{if probit model} \end{cases}$$

where $L_{\beta_j}^*$ is the derivative of *L* with respect to β_j.

Algorithm

Probit uses the algorithms proposed and implemented in NPSOL by Gill, Murray, Saunders, and Wright. The loss function for this procedure is the negative of the log-likelihood described in the model. The derivatives for the parameters are described above. The only bound for the parameters is $0 < c < 1$. For more details of the NPSOL algorithms, see CNLR (constrained nonlinear regression).

Natural Response Rate

When the user specifies a fixed number for the natural response rate, L_c^* is set to 0 for iterations and the bound for *c* is set equal to the fixed number.

Initial Values

The initial value for each β is set to 0. If there is a control group, the initial value of *c*, designated by c_0, is set to the ratio of the response to the number of subjects for the control group. If there is no control group, then c_0 is set to the minimum ratio of the response to the number of subjects, over all covariate patterns.

Criteria

Users can control two criteria, ITER and CONV. ITER is the maximum number of iterations allowed. The default value is $\max(50, 3(s+1))$. CONV (criterion of convergence) is the same as the OPTOLERANCE criterion in CNLR.

Asymptotic Covariance Matrix

The asymptotic covariance matrix for the MLE $\left(\hat{c}, \hat{\beta}_1, \ldots, \hat{\beta}_{p+q}\right)$ is estimated by $\mathbf{I}^{-1}$, where $\mathbf{I}$ is the information matrix containing the negatives of the second partial derivatives of *L*.

$$\frac{\partial^2 L}{\partial c^2} = \sum_{i=1}^{m} \left[\frac{r_i - n_i P_i}{P_i(1-P_i)^2} - \frac{r_i}{P_i(1-P_i)}\right]\left(1 - F\left(\mathbf{X}_i'\beta\right)\right)^2$$

$$\frac{\partial^2 L}{\partial c \partial \beta_j} = \sum_{i=1}^{m} x_{ij}\left((1-c)\left(1 - F\left(\mathbf{X}_i'\beta\right)\right)\left[\frac{r_i - n_i P_i}{P_i(1-P_i)^2} - \frac{r_i}{P_i(1-P_i)}\right] - \frac{r_i - n_i P_i}{P_i(1-P_i)}\right)\frac{dF\left(\mathbf{X}_i'\beta\right)}{d\mathbf{X}_i'\beta}$$

where

$$\frac{dF\left(\mathbf{X}_i'\beta\right)}{d\mathbf{X}_i'\beta} = \begin{cases} F\left(\mathbf{X}_i'\beta\right)\left(1 - F\left(\mathbf{X}_i'\beta\right)\right) & \text{if logit model} \\ \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{1}{2}\left(\mathbf{X}_i'\beta\right)^2\right\} & \text{if probit model} \end{cases}$$

$$\begin{aligned}\frac{\partial^2 L}{\partial \beta_j \partial \beta_h} &= (1-c^2)\sum_{i=1}^{m} \left[\frac{r_i - n_i P_i}{P_i(1-P_i)^2} - \frac{r_i}{P_i(1-P_i)}\right] x_{ij} x_{ih} \left(\frac{dF\left(\mathbf{X}_i'\beta\right)}{d\mathbf{X}_i'\beta}\right)^2 \\ &\quad + (1-c)\sum_{i=1}^{m} \left[\frac{r_i - n_i P_i}{P_i(1-P_i)}\right] x_{ij} x_{ih} \frac{d^2F\left(\mathbf{X}_i'\beta\right)}{d^2\mathbf{X}_i'\beta}\end{aligned}$$

where

$$\frac{d^2F\left(\mathbf{X}_i'\beta\right)}{d^2\mathbf{X}_i'\beta} = \begin{cases} F\left(\mathbf{X}_i'\beta\right)\left(1 - F\left(\mathbf{X}_i'\beta\right)\right)\left(1 - 2F\left(\mathbf{X}_i'\beta\right)\right) & \text{if logit model} \\ \frac{1}{\sqrt{2\pi}}\left(-\mathbf{X}_i'\beta\right) \exp\left(-\frac{1}{2}\left(\mathbf{X}_i'\beta\right)^2\right) & \text{if probit model} \end{cases}$$

Frequency Table and Goodness of Fit

For every covariate pattern *i*, *i*=1,...,*m*, compute

$$\hat{F}_i = \begin{cases} \frac{e^{\mathbf{x}_i'\beta}}{1+e^{\mathbf{x}_i'\beta}} & \text{if logit model} \\ \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\mathbf{x}_i'\beta} e^{-z^2/2}\text{dz} & \text{if probit model} \end{cases}$$

$$\hat{P}_i = \hat{c} + (1-\hat{c})\hat{F}_i$$

Then the expected frequency is equal to

$$\hat{E}_i = n_i\hat{P}_i$$

The Pearson chi-square statistic is defined by

$$\chi^2 = \sum_{i=1}^{m} \frac{\left(r_i - \hat{E}_i\right)^2}{\hat{E}_i\left(1-\hat{P}_i\right)}$$

and the degrees of freedom (*df*) is

$$df = \begin{cases} (q-1)m - s & \text{if } q \geq 2 \\ m - s & \text{if } q = 1 \end{cases}$$

Fiducial Limits, RMP, and Parallelism

The parallelism test statistic, fiducial limits, and relative median potency are available when there is only one covariate (predictor variable). Assuming that $\hat{\alpha}_1, \ldots, \hat{\alpha}_q$ are the MLE's for $\alpha_1, \ldots, \alpha_q$ and $\hat{\gamma}$ is the MLE for γ, $v\left(\hat{\alpha}_j\right)$ is the asymptotic variance for $\hat{\alpha}_j$, $v\left(\hat{\gamma}\right)$ is the asymptotic variance for $\hat{\gamma}$, and $cov\left(\hat{\alpha}_j, \hat{\gamma}\right)$ is the asymptotic covariance for $\hat{\alpha}_j$ and $\hat{\gamma}$.

Fiducial Limits for Effective dose x

For level of the grouping variable *j* and *P* = 0.01 through 0.09, 0.10 through 0.90 (by 0.05), and 0.91 through 0.99, compute

$$y = \begin{cases} \ln\left(P/(1\text{-}P)\right) & \text{if logit model} \\ \text{probit}\left(P\right) & \text{if probit model} \end{cases}$$

Then the effective dose x_j to obtain probability *P* of response for level *j* is defined by

$$x_j = \left((y - \hat{\alpha}_j)/\hat{\gamma}\right)$$

and the 95% fiducial limit for effective dose x_j is computed by

$$x_j + \frac{g}{1-g}\left(x + \frac{cov(\hat{\alpha}_j,\hat{\gamma})}{v(\hat{\gamma})}\right)$$
$$\pm\frac{t}{\hat{\gamma}(1-g)}\sqrt{\left\{v(\hat{\alpha}_j) + 2x_j cov(\hat{\alpha}_j,\hat{\gamma}) + x_j^2 v(\hat{\gamma}) - g\left(v(\hat{\alpha}_j) - \frac{(cov(\hat{\alpha}_j,\hat{\gamma}))^2}{v(\hat{\gamma})}\right)\right\}h^*}$$

where

$$g = \begin{cases} \frac{t^2 v(\hat{\gamma})}{\hat{\gamma}^2} & \text{without heterogeneity factor} \\ \frac{t^2 v(\hat{\gamma})}{\hat{\gamma}^2} h & \text{with heterogeneity factor} \end{cases}$$

$$t = \begin{cases} 1.96 & \text{without heterogeneity factor} \\ t_{(0.025,df)} & \text{with heterogeneity factor} \end{cases}$$

$$h = x_j^2/(df)$$

$$h^* = \begin{cases} 1 & \text{without heterogeneity factor} \\ h & \text{with heterogeneity factor} \end{cases}$$

The heterogeneity factor is used if the Pearson chi-square statistic is significant.

Note: If the covariate (predictor variable) *x* is transformed, transform it back to the original metrics for the estimate and its two limits. For example, if $\log_{10}$ is applied to the predictor for the analysis and $\hat{x}_L, \hat{x}, \hat{x}_U$ are the lower limit, the estimate, and the upper limit on the $\log_{10}$ scale, then $10^{\hat{x}_L}$and $10^{\hat{x}_U}$ are the lower and upper limits on the original scale.

Relative Median Potency

The relative median potency is available when there is a factor variable and a single covariate. It is not available if there is no factor variable or if there is more than one covariate.

The estimate of relative median potency for group *j* versus group *k* is

$$M_{jk} = (\hat{\alpha}_k - \hat{\alpha}_j)/\hat{\gamma}$$

and its 95% confidence limit is

$$M_{jk} + \frac{g}{1-g}\left(M_{jk} - \frac{v_{12}}{v_{22}}\right) \pm \frac{t}{\hat{\gamma}(1-g)}\sqrt{\left\{v_{11} - 2M_{jk}v_{12} + M_{jk}^2 v_{22} - g\left(v_{11} - \frac{v_{12}^2}{v_{22}}\right)\right\}h^*}$$

where

$$\begin{aligned} v_{11} &= v(\hat{\alpha}_j) + v(\hat{\alpha}_k) - 2cov(\hat{\alpha}_j, \hat{\alpha}_k) \\ v_{12} &= cov(\hat{\alpha}_j, \hat{\gamma}) - cov(\hat{\alpha}_k, \hat{\gamma}) \\ v_{22} &= v(\hat{\gamma}) \end{aligned}$$

Note: If the covariate (predictor variable) *x* is transformed, transform it back to the original metrics for the relative median potency.

Parallelism Test Chi-Square Statistic

The parallelism test is available only if there is a factor variable.

$$\chi^2 = \chi_0^2 - \sum_{j=1}^{q} \chi_j^2$$

where χ_0^2 is the Pearson chi-square statistic, assuming that the group variable is in the model and χ_j^2 is the Pearson chi-square for the *j*th group and the degrees of freedom for χ^2 is *q*−1.

References

Finney, D. J. 1971. *Probit analysis.* Cambridge: Cambridge University Press.

Gill, P. E., W. M. Murray, M. A. Saunders, and M. H. Wright. 1986. *User's guide for NPSOL (version 4.0): A FORTRAN package for nonlinear programming. Technical Report SOL 86-2*. Stanford University: Department of Operations Research.

PROXIMITIES Algorithms

PROXIMITIES computes a variety of measures of similarity, dissimilarity, or distance between pairs of cases or pairs of variables.

Standardizing Cases or Variables

Either cases or variables can be standardized. The following methods of standardization are available:

Z

PROXIMITIES subtracts the mean from each value for the variable or case being standardized and then divides by the standard deviation of the values. If a standard deviation is 0, PROXIMITIES sets all values for the case or variable to 0.

RANGE

PROXIMITIES divides each value for the variable or case being standardized by the range of the values. If the range is 0, PROXIMITIES leaves all values unchanged.

RESCALE

From each value for the variable or case being standardized, PROXIMITIES subtracts the minimum value and then divides by the range. If a range is 0, PROXIMITIES sets all values for the case or variable to 0.50.

MAX

PROXIMITIES divides each value for the variable or case being standardized by the maximum of the values. If the maximum of a set of values is 0, PROXIMITIES uses an alternate process to produce a comparable standardization: it divides by the absolute magnitude of the smallest value and adds 1.

MEAN

PROXIMITIES divides each value for the variable or case being standardized by the mean of the values. If a mean is 0, PROXIMITIES adds one to all values for the case or variable to produce a mean of 1.

SD

PROXIMITIES divides each value for the variable or case being standardized by the standard deviation of the values. PROXIMITIES does not change the values if their standard deviation is 0.

Transformations

Three transformations are available for the values PROXIMITIES computes or reads:

ABSOLUTE

Take the absolute values of the proximities.

REVERSE

Transform similarity values into dissimilarities, or vice versa, by changing the signs of the coefficients.

RESCALE

RESCALE standardizes the proximities by first subtracting the value of the smallest and then dividing by the range.

If you specify more than one transformation, PROXIMITIES does them in the order listed above: first ABSOLUTE, then REVERSE, then RESCALE.

Proximities Measures

Measure defines the formula for calculating distance. For example, the Euclidean distance measure calculates the distance as a "straight line" between two clusters.

Measures for Continuous Data

Measures for continuous data, also called interval measures, assume that the variables are scale.

EUCLID

The distance between two items, x and y, is the square root of the sum of the squared differences between the values for the items.

$$\text{EUCLID}(x,y) = \sqrt{\Sigma_i (x_i - y_i)^2}$$

SEUCLID

The distance between two items is the sum of the squared differences between the values for the items.

$$\text{SEUCLID}(x,y) = \Sigma_i (x_i - y_i)^2$$

CORRELATION

This is a pattern similarity measure.

$$\text{CORRELATION}(x,y)=\frac{\Sigma_i(Z_{xi}Z_{yi})}{N}$$

where Z_{xi} is the (standardized) *Z*-score value of *x* for the *i*th case or variable, and *N* is the number of cases or variables.

COSINE

This is a pattern similarity measure.

$$\text{COSINE}(x,y)=\frac{\Sigma_i(x_iy_i)}{\sqrt{((\Sigma_i x_i^2)(\Sigma_i y_i^2))}}$$

CHEBYCHEV

The distance between two items is the maximum absolute difference between the values for the items.

$$\text{CHEBYCHEV}(x,y)=\max_i|x_i-y_i|$$

BLOCK

The distance between two items is the sum of the absolute differences between the values for the items.

$$\text{BLOCK}(x,y)=\Sigma_i|x_i-y_i|$$

MINKOWSKI(p)

The distance between two items is the pth root of the sum of the absolute differences to the pth power between the values for the items.

$$\text{MINKOWSKI}(x,y)=\left(\Sigma_i|x_i-y_i|^p\right)^{1/p}$$

POWER(p,r)

The distance between two items is the *r*th root of the sum of the absolute differences to the *p*th power between the values for the items.

$$\text{POWER}(x,y)=\left(\Sigma_i|x_i-y_i|^p\right)^{1/r}$$

Measures for Frequency Count Data

Frequency count measures assume that the variables are discrete numeric.

CHISQ

The magnitude of this dissimilarity measure depends on the total frequencies of the two cases or variables whose proximity is computed. Expected values are from the model of independence of cases (or variables), *x* and *y*.

$$\text{CHISQ}(x,y)=\sqrt{\sum_i \frac{(x_i-E(x_i))^2}{E(x_i)}+\sum_i \frac{(y_i-E(y_i))^2}{E(y_i)}}$$

PH2

This is the CHISQ measure normalized by the square root of the combined frequency. Therefore, its value does not depend on the total frequencies of the two cases or variables whose proximity is computed.

$$\text{PH2}(x,y)=\frac{\text{CHISQ}(x,y)}{\sqrt{N}}$$

Measures for Binary Data

Binary measures assume that the variables take only two values.

PROXIMITIES constructs a 2×2 contingency table for each pair of items in turn. It uses this table to compute a proximity measure for the pair.

Table 81-1
2 x 2 Contingency table

	Item 2 Present	Item 2 Absent
Item 1 Present	*a*	*b*
Item 1 Absent	*c*	*d*

PROXIMITIES computes all binary measures from the values of *a*, *b*, *c*, and *d*. These values are tallies across variables (when the items are cases) or tallies across cases (when the items are variables).

Russel and Rao Similarity Measure

This is the binary dot product.

$$\text{RR}(x,y)=\frac{a}{a+b+c+d}$$

Simple Matching Similarity Measure

This is the ratio of the number of matches to the total number of characteristics.

$$\text{SM}(x,y) = \frac{a+d}{a+b+c+d}$$

Jaccard Similarity Measure

This is also known as the similarity ratio.

$$\text{JACCARD}(x,y) = \frac{a}{a+b+c}$$

Dice or Czekanowski or Sorenson Similarity Measure

$$\text{DICE}(x,y) = \frac{2a}{2a+b+c}$$

Sokal and Sneath Similarity Measure 1

$$\text{SS1}(x,y) = \frac{2(a+d)}{2(a+d)+b+c}$$

Rogers and Tanimoto Similarity Measure

$$\text{RT}(x,y) = \frac{a+d}{a+d+2(b+c)}$$

Sokal and Sneath Similarity Measure 2

$$\text{SS2}(x,y) = \frac{a}{a+2(b+c)}$$

Kulczynski Similarity Measure 1

This measure has a minimum value of 0 and no upper limit. It is undefined when there are no nonmatches $(b = 0 \text{and} c = 0)$. Therefore, PROXIMITIES assigns an artificial upper limit of 9999.999 to K1 when it is undefined or exceeds this value.

$$\text{K1}(x,y) = \frac{a}{b+c}$$

Sokal and Sneath Similarity Measure 3

This measure has a minimum value of 0, has no upper limit, and is undefined when there are no nonmatches $(b=0 \text{and} c=0)$. As with K1, PROXIMITIES assigns an artificial upper limit of 9999.999 to SS3 when it is undefined or exceeds this value.

$$\text{SS3}(x,y)=\frac{a+d}{b+c}$$

Conditional Probabilities

The following three binary measures yield values that you can interpret in terms of conditional probability. All three are similarity measures.

Kulczynski Similarity Measure 2

This yields the average conditional probability that a characteristic is present in one item given that the characteristic is present in the other item. The measure is an average over both items acting as predictors. It has a range of 0 to 1.

$$\text{K2}(x,y)=\frac{a/(a+b)+a/(a+c)}{2}$$

Sokal and Sneath Similarity Measure 4

This yields the conditional probability that a characteristic of one item is in the same state (present or absent) as the characteristic of the other item. The measure is an average over both items acting as predictors. It has a range of 0 to 1.

$$\text{SS4}(x,y)=\frac{a/(a+b)+a/(a+c)+d/(b+d)+d/(c+d)}{4}$$

Hamann Similarity Measure

This measure gives the probability that a characteristic has the same state in both items (present in both or absent from both) minus the probability that a characteristic has different states in the two items (present in one and absent from the other). HAMANN has a range of –1 to +1 and is monotonically related to SM, SS1, and RT.

$$\text{HAMANN}(x,y)=\frac{(a+d)-(b+c)}{a+b+c+d}$$

Predictability Measures

The following four binary measures assess the association between items as the predictability of one given the other. All four measures yield similarities.

Goodman and Kruskal Lambda (Similarity)

This coefficient assesses the predictability of the state of a characteristic on one item (presence or absence) given the state on the other item. Specifically, lambda measures the proportional reduction in error using one item to predict the other, when the directions of prediction are of equal importance. Lambda has a range of 0 to 1.

$$t_1 = \max(a,b) + \max(c,d) + \max(a,c) + \max(b,d)$$
$$t_2 = \max(a+c, b+d) + \max(a+b, c+d)$$
$$\text{LAMBDA}(x,y) = \frac{t_1 - t_2}{2(a+b+c+d) - t_2}$$

Anderberg's D (Similarity)

This coefficient assesses the predictability of the state of a characteristic on one item (presence or absence) given the state on the other. *D* measures the actual reduction in the error probability when one item is used to predict the other. The range of *D* is 0 to 1.

$$t_1 = \max(a,b) + \max(c,d) + \max(a,c) + \max(b,d)$$
$$t_2 = \max(a+c, b+d) + \max(a+b, c+d)$$
$$\text{D}(x,y) = \frac{t_1 - t_2}{2(a+b+c+d)}$$

Yule's Y Coefficient of Colligation (Similarity)

This is a function of the cross-product ratio for a 2×2 table. It has a range of –1 to +1.

$$\text{Y}(x,y) = \frac{\sqrt{ad} - \sqrt{bc}}{\sqrt{ad} + \sqrt{bc}}$$

Yule's Q (Similarity)

This is the 2×2 version of Goodman and Kruskal's ordinal measure *gamma*. Like Yule's *Y, Q* is a function of the cross-product ratio for a 2×2 table and has a range of –1 to +1.

$$\text{Q}(x,y) = \frac{ad - bc}{ad + bc}$$

Other Binary Measures

The remaining binary measures available in PROXIMITIES are either binary equivalents of association measures for continuous variables or measures of special properties of the relation between items.

Ochiai Similarity Measure

This is the binary form of the cosine. It has a range of 0 to 1 and is a similarity measure.

$$\text{OCHIAI}(x,y) = \sqrt{\left(\frac{a}{a+b}\right)\left(\frac{a}{a+c}\right)}$$

Sokal and Sneath Similarity Measure 5

This is a similarity measure. Its range is 0 to 1.

$$\text{SS5}(x,y) = \frac{ad}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$$

Fourfold Point Correlation (Similarity)

This is the binary form of the Pearson product-moment correlation coefficient. Phi is a similarity measure, and its range is 0 to 1.

$$\text{PHI}(x,y) = \frac{ad-bc}{\sqrt{(a+b)(a+c)(b+d)(c+d)}}$$

Binary Euclidean Distance

This is a distance measure. Its minimum value is 0, and it has no upper limit.

$$\text{BEUCLID}\,(x,y) = \sqrt{b+c}$$

Binary Squared Euclidean Distance

This is also a distance measure. Its minimum value is 0, and it has no upper limit.

$$\text{BSEUCLID}\,(x,y) = b+c$$

Size Difference

This is a dissimilarity measure with a minimum value of 0 and no upper limit.

$$\text{SIZE}(x,y) = \frac{(b-c)^2}{(a+b+c+d)^2}$$

Pattern Difference

This is also a dissimilarity measure. Its range is 0 to 1.

$$\text{PATTERN}(x,y) = \frac{bc}{(a+b+c+d)^2}$$

Binary Shape Difference

This dissimilarity measure has no upper or lower limit.

$$\text{BSHAPE}(x,y) = \frac{(a+b+c+d)(b+c)-(b-c)^2}{(a+b+c+d)^2}$$

Dispersion Similarity Measure

This similarity measure has a range of –1 to +1.

$$\text{DISPER}(x,y) = \frac{ad-bc}{(a+b+c+d)^2}$$

Variance Dissimilarity Measure

This dissimilarity measure has a minimum value of 0 and no upper limit.

$$\text{VARIANCE}(x,y) = \frac{b+c}{4(a+b+c+d)}$$

Binary Lance-and-Williams Nonmetric Dissimilarity Measure

Also known as the Bray-Curtis nonmetric coefficient, this dissimilarity measure has a range of 0 to 1.

$$\text{BLWMN}(x,y) = \frac{b+c}{2a+b+c}$$

References

Anderberg, M. R. 1973. *Cluster analysis for applications*. New York: Academic Press.

Romesburg, H. C. 1984. *Cluster analysis for researchers*. Belmont, Calif.: Lifetime Learning Publications.

PROXSCAL Algorithms

PROXSCAL performs multidimensional scaling of proximity data to find a least-squares representation of the objects in a low-dimensional space. Individual differences models can be specified for multiple sources. A majorization algorithm guarantees monotone convergence for optionally transformed, metric and nonmetric data under a variety of models and constraints.

Detailed mathematical derivations concerning the algorithm can be found in Commandeur and Heiser (1993).

Notation

The following notation is used throughout this chapter unless otherwise stated. For the dimensions of the vectors and matrices:

n	Number of objects
m	Number of sources
p	Number of dimensions
s	Number of independent variables
h	Maximum(s, p)
l	Length of transformation vector
r	Degree of spline
t	Number of interior knots for spline

The input and input-related variables are:

Δ_k	n×n matrix with raw proximities for source k
$\mathbf{W}_k$	n×n matrix with weights for source k
$\mathbf{E}$	n×s matrix with raw independent variables
$\mathbf{F}$	n×p matrix with fixed coordinates

Output and output-related variables are:

$\hat{\mathbf{D}}_k$	n×n matrix with transformed proximities for source k
$\mathbf{Z}$	n×p matrix with common space coordinates
$\mathbf{A}_k$	p×p matrix with space weights for source k
$\mathbf{X}_k$	n×p matrix with individual space coordinates for source k
$\mathbf{Q}$	n×h matrix with transformed independent variables
$\mathbf{B}$	h×p matrix with regression weights for independent variables
$\mathbf{S}$	l×(r+t) matrix of coefficients for the spline basis

Special matrices and functions are:

$\mathbf{J}$	$\mathbf{I} - \mathbf{1}\mathbf{1}^{\mathrm{T}}/\mathbf{1}^{\mathrm{T}}\mathbf{1}$, centering matrix of appropriate size
$D(\mathbf{X}_k)$	n×n matrix with distances, with elements $\{d_{ijk}\}$, where $d_{ijk} = \sqrt{(\mathbf{x}_{ik} - \mathbf{x}_{jk})(\mathbf{x}_{ik} - \mathbf{x}_{jk})}$

$\mathbf{V}_k$ $n \times n$ matrix with elements $\{v_{ijk}\}$, where $v_{ijk} = \begin{cases} -w_{ijk} \text{ for } i \neq j \\ \sum_{l \neq i}^{n} w_{ilk} \text{ for } i = j \end{cases}$

$\mathbf{B}(\mathbf{X}_k)$ $n \times n \times m$ matrix with elements $\{b_{ijk}\}$, where

$$b_{ijk} = \begin{cases} \frac{-w_{ijk}(\delta_{ijk})}{ij(\mathbf{X}_k)} \text{ if } d_{ij}(\mathbf{X}_k) > 0 \text{ and } i \neq j \\ 0 \text{ if } d_{ij}(\mathbf{X}_k) = 0 \text{ and } i \neq j \\ -\sum_{l \neq i}^{n} b_{ilk} \text{ if } i = j \end{cases}$$

Introduction

The following loss function is minimized by PROXSCAL,

$$\sigma^2 \equiv \frac{1}{m} \sum_{k=1}^{m} \sum_{i<j}^{n} w_{ijk} \left[\hat{d}_{ijk} - d_{ij}(\mathbf{X}_k) \right]^2$$

which is the weighted mean squared error between the transformed proximities and the distances of *n* objects within *m* sources. The transformation function for the proximities provides nonnegative, monotonically nondecreasing values for the transformed proximities $\hat{d}_{ijk}$. The distances d$_{ij}(\mathbf{X}_k)$ are simply the Euclidean distances between the object points, with the coordinates in the rows of $\mathbf{X}_k$.

The main algorithm consists of the following major steps:

1. find initial configurations $\mathbf{X}_k$, and evaluate the loss function;
2. find an update for the configurations $\mathbf{X}_k$;
3. find an update for the transformed proximities $\hat{d}_{ijk}$;
4. evaluate the loss function; if some predefined stop criterion is satisfied, stop; otherwise, go to step 2.

Preliminaries

At the start of the procedure, several preliminary computations are performed to handle missing weights or proximities, and initialize the raw proximities.

Missing Values

On input, missing values may occur for both weights and proximities. If a weight is missing, it is set equal to zero. If a proximity is missing, the corresponding weight is set equal to zero.

Proximities

Only the upper or lower triangular part (without the diagonal) of the proximity matrix is needed. In case both triangles are given, the weighted mean of both triangles is used. Next, the raw proximities are transformed such that similarities become dissimilarities by multiplying with -1, taking into account the conditionality, and setting the smallest dissimilarity equal to zero.

Transformations

For ordinal transformations, the nonmissing proximities are replaced by their ascending rank numbers, also taking into account the conditionality. For spline transformations, the spline basis **S** is computed.

Normalization

The proximities are normalized such that the weighted squared proximities equal the sum of the weights, again, taking into account the conditionality.

Step 1: Initial Configuration

PROXSCAL allows for several initial configurations. Before determining the initial configuration, missings are handled, and the raw proximities are initialized. Finally, after one of the starts described below, the common space **Z** is centered on the origin and optimally dilated in accordance with the normalized proximities.

Simplex Start

The simplex start consists of a rank-*p* approximation of the matrix $\mathbf{V}^{-}\mathbf{B}(\mathbf{J})$. Set **H**, an *n*×*p* columnwise orthogonal matrix, satisfying $\mathbf{H}^{\mathrm{T}}\mathbf{H} = \mathbf{I}_p$, where $\mathbf{I}_p$ denotes the matrix with the first *p* columns of the identity matrix. The nonzero rows are selected in such a way that the first **Z**=**B**(**J**)**H** contains the *p* columns of **B**(**J**) with the largest diagonal elements. The following steps are computed in turn, until convergence is reached:

1. For a fixed **Z**, $\mathbf{H}=\mathbf{P}\mathbf{Q}^{\mathrm{T}}$, where $\mathbf{P}\mathbf{Q}^{\mathrm{T}}$ is taken from the singular value decomposition $\mathbf{B}(\mathbf{J})\mathbf{Z}=\mathbf{P}\mathbf{L}\mathbf{Q}^{\mathrm{T}}$;
2. For a fixed **H**, $\mathbf{Z} = 2^{-1/2}\mathbf{V}^{-}\mathbf{B}(\mathbf{J})\mathbf{H}$, where $\mathbf{V}^{-}$ is the pseudo-inverse of **V**.

For a restricted common space **Z**, the second step is adjusted in order to fullfill the restictions. This procedure was introduced in Heiser (1985).

Torgerson Start

The proximities are aggregated over sources, squared, double centered and multiplied with −0.5, after which an eigenvalue decomposition is used to determine the coordinate values, thus

$$-0.5\mathbf{J}\mathbf{D}^{*}\mathbf{J} = \mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^{\mathrm{T}}$$

where elements of $\mathbf{D}^*$ are defined as

$$d_{ij}^* = \left(\sum_{k=1}^{m} w_{ijk}\hat{d}_{ijk}^2\right)\left(\sum_{k=1}^{m} w_{ijk}\right)^{-1}$$

followed by $\mathbf{Z} = \mathbf{Q}\mathbf{\Lambda}^{1/2}$, where only the first *p* positive ordered eigenvalues $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_n$) and eigenvectors are used. This technique, classical scaling, is due to Torgerson (1952, 1958) and Gower (1966) and also known under the names Torgerson scaling or Torgerson-Gower scaling.

(Multiple) Random Start

The coordinate values are randomly generated from a uniform distribution using the default random number generator from IBM® SPSS® Statistics.

User-Provided Start

The coordinate values provided by the user are used.

Step 2: Configuration Update

The coordinates of the common space and the space weights (if applicable) are updated.

Update for the Common Space

The common space **Z** is related to the individual spaces $\mathbf{X}_k$ through the model $\mathbf{X}_k = \mathbf{Z}\mathbf{A}_k$, where $\mathbf{A}_k$ are matrices containing space weights. Assume that weight matrix $\mathbf{A}_k$ is of full rank. Only considering **Z** defines the loss function as

$$\sigma^2(\mathbf{z}) = c + \mathbf{z}^T\mathbf{H}\mathbf{z} - 2\mathbf{z}^T\mathbf{t},$$

where

$$\mathbf{z} \equiv \text{vec}(\mathbf{Z}),$$
$$\mathbf{H} \equiv \tfrac{1}{m}\sum_{k=1}^{m}\left(\mathbf{A}_k\mathbf{A}_k^T \otimes \mathbf{V}_k\right),$$
$$\mathbf{t} \equiv \text{vec}\left(\tfrac{1}{m}\sum_{k=1}^{m}\mathbf{B}(\mathbf{X}_k)\mathbf{X}_k\mathbf{A}_k^T\right),$$

for which a solution is found as

$$\mathbf{z} = \mathbf{H}^{-}\mathbf{t}$$

Several special cases exist for which the solution can be simplified. First, the weights matrices $\mathbf{W}_k$ may all be equal, or even all equal to one. In these cases **H** will simplify, as will the pseudo-inverse of **H**. Another simplification is concerned with the different models, reflected in restrictions for the space weights. This model is the generalized Euclidean model, also known as IDIOSCAL (Carroll and Chang, 1972). The weighted Euclidean model, or INDSCAL, restricts

$\mathbf{A}_k$ to be diagonal, which does simplify $\mathbf{H}$, but not the pseudo-inverse. The identity model requires $\mathbf{A}_k = \mathbf{I}$ for all *k*, and does simplify $\mathbf{H}$ and its pseudo-inverse, for the kronecker product vanishes.

To avoid computing the pseudo-inverse of a large matrix, PROXSCAL uses three technical simplifications when appropriate. First, the pseudo-inverse can be replaced by a proper inverse by adding the nullspace, taking the proper inverse and then subtracting the nullspace again as

$$\mathbf{H}^- = (\mathbf{H} + \mathbf{N})^{-1} - \mathbf{N}$$

where $\mathbf{N} = \left(\mathbf{1}\mathbf{1}^T\right) / \left(\mathbf{1}^T\mathbf{1}\right)$. Furthermore, a dimensionwise approach (Heiser and Stoop, 1986) is used which results in a solution for dimension *a* of $\mathbf{Z}$ as

$$\mathbf{z}_a = \mathbf{V}_a^- \overline{\mathbf{z}}_a,$$

where

$$\mathbf{V}_a = \frac{1}{m} \sum_{k=1}^{m} \mathbf{V}_k \mathbf{e}_a^T \mathbf{A}_k \mathbf{A}_k^T \mathbf{e}_a,$$

where $\mathbf{e}_a$ is the *a*th column of an identity matrix, and

$$\overline{\mathbf{z}}_a = \frac{1}{m} \sum_{k=1}^{m} \left[\mathbf{B}\left(\mathbf{X}_k\right) \mathbf{X}_k \mathbf{A}_k^T - \mathbf{V}_k \mathbf{P}_a \mathbf{A}_k \mathbf{A}_k^T\right] \mathbf{e}_a,$$

with $\mathbf{P}_a$ an *n*×*p* matrix equal to **Z**, but with the *a*th column containing zeros.

Still, the proper inverse of a *n*×*n* matrix is required. The final simplification is concerned with a majorization function in which the largest eigenvalue of **V** allows for an easy update (Heiser, 1987; Groenen, Heiser, and Meulman, 1999). Instead of the largest eigenvalue itself, an upper bound is used for this scalar (Wolkowicz and Styan, 1980).

Update for the Space Weights

An update for the space weights $\mathbf{A}_k (k = 1, ..., m)$ for the generalized Euclidean model is given by

$$\mathbf{A}_k = \left(\mathbf{Z}^T \mathbf{V}_k \mathbf{Z}\right)^{-1} \left(\mathbf{Z}^T \mathbf{B}(\mathbf{X}_k) \mathbf{X}_k\right)$$

Suppose $\mathbf{P}_k \mathbf{L}_k \mathbf{Q}_k^T$ is the singular value decomposition of $\mathbf{A}_k$ for which the diagonal matrix with singular values $\mathbf{L}_k$ is in nonincreasing order. Then, for the reduced rank model, the best *r*(*r*<*p*) rank approximation of $\mathbf{A}_k$ is given by $\mathbf{R}_k \mathbf{T}_k^T$, where $\mathbf{R}_k$ contains the first *r* columns of $\mathbf{P}_k \mathbf{L}_k$, and $\mathbf{T}_k$ contains the first *r* columns of $\mathbf{Q}_k$.

For the weighted Euclidean model, the update reduces to a diagonal matrix

$$\mathbf{A}_k = \text{diag}\left(\mathbf{Z}^T \mathbf{V}_k \mathbf{Z}\right)^{-1} \text{diag}\left(\mathbf{Z}^T \mathbf{B}(\mathbf{X}_k) \mathbf{X}_k\right)$$

The space weights for the identity model need no update, since $\mathbf{A}_k = \mathbf{I}$ for all *k*. Simplifications can be obtained if all weights **W** are equal to one and for the reduced rank model, which can be done in *r* dimensions, as explained in Heiser and Stoop (1986).

Restrictions

The user can impose restrictions on the common space by fixing some of the coordinates or specifying that the common space is a weighted sum of independent variables.

Fixed Coordinates

If some of the coordinates of **Z** are fixed by the user, then only the free coordinates of **Z** need to be updated. The dimensionwise approach is taken one step further, which results in an update for object *i* on dimension *a* as

$$z_{ia}^{+} = \frac{1}{\mathbf{e}_i^{\mathrm{T}}\overline{\mathbf{V}}_a\mathbf{e}_i}\mathbf{e}_i^{\mathrm{T}}\left[\frac{1}{m}\sum_{k=1}^{m}\mathbf{B}\left(\mathbf{X}_k\right)\mathbf{X}_k\mathbf{A}_k^{\mathrm{T}}\mathbf{e}_a - \frac{1}{m}\sum_{j\neq a}^{p}\left(\sum_{k=1}^{m}\mathbf{e}_j^{\mathrm{T}}\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}\mathbf{e}_a\mathbf{V}_k\right)\mathbf{z}_j\right] - \frac{1}{\mathbf{e}_i^{\mathrm{T}}\overline{\mathbf{V}}_a\mathbf{e}_i}\mathbf{e}_i^{\mathrm{T}}\overline{\mathbf{V}}_a\tilde{\mathbf{z}}_{ia}$$

where the *a*th column of **Z** is divided into $\mathbf{z}_a = \tilde{\mathbf{z}}_{ia} + z_{ia}\mathbf{e}_i$, with $\mathbf{e}_i$ the *i*th column of the identity matrix, and $\overline{\mathbf{V}}_a = \frac{1}{m}\sum_{k=1}^{m}\mathbf{e}_j^{\mathrm{T}}\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}\mathbf{e}_a\mathbf{V}_k$.

This update procedure will only locally minimize the loss function, and repeatedly cycling through all free coordinates until convergence is reached, will provide global optimization. After all free coordinates have been updated, **Z** is centered on the origin. On output, the configuration is adapted as to coincide with the initial fixed coordinates.

Independent Variables

Independent variables **Q** are used to express the coordinates of the common space **Z** as a weighted sum of these independent variables as

$$\mathbf{Z} = \mathbf{QB} = \sum_{j=1}^{h}\mathbf{q}_j\mathbf{b}_j^{\mathrm{T}}$$

An update for **Z** is found by performing the following calculations for *j*=1,...,*h*:

1. $\mathbf{U}_j = \sum_{k\neq j}^{h}\mathbf{q}_k\mathbf{b}_k^{\mathrm{T}}$
2. $\mathbf{T}_j = \mathbf{C}^{-}\ \frac{1}{m}\sum_{k=1}^{m}\mathbf{V}_k\mathbf{U}_j\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}$, where $\mathbf{C} = \frac{1}{m}\sum_{k=1}^{m}\mathbf{B}(\mathbf{X}_k)\mathbf{X}_k\mathbf{A}_k^{\mathrm{T}}$
3. update $\mathbf{b}_j$ as $\mathbf{b}_j = \left(\frac{1}{m}\sum_{k=1}^{m}\mathbf{q}_j\mathbf{V}_k\mathbf{q}_j\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}\right)^{-1}\mathbf{T}_j^{\mathrm{T}}\mathbf{q}_j$
4. optionally, compute optimally transformed variables by regressing $\tilde{\mathbf{q}}_j = \frac{1}{k_1}\mathbf{T}_j\mathbf{b}_j + \left(\mathbf{I} - \frac{1}{k_1}\overline{\mathbf{V}}_j\right)\mathbf{q}_j$, where $\overline{\mathbf{V}}_j = \frac{1}{m}\sum_{k=1}^{m}\mathbf{b}_j^{\mathrm{T}}\mathbf{A}_k\mathbf{A}_k^{\mathrm{T}}\mathbf{b}_j\mathbf{V}_k$ and k_1 is greater than or equal to the largest eigenvalue of $\overline{\mathbf{V}}_j$, on the original variable $\mathbf{q}_j$. Missing elements in the original variable are replaced with the corresponding values from $\tilde{\mathbf{q}}_j$.

Finally, set $\mathbf{Z} = \mathbf{QB} = \sum_{j=1}^{h} \mathbf{q}_j \mathbf{b}_j^{\mathrm{T}}$.

Independent variables restrictions were introduced for the MDS model in Bentler and Weeks (1978), Bloxom (1978), de Leeuw and Heiser (1980) and Meulman and Heiser (1984). If there are more dimensions (*p*) than independent variables (*s*), *p*−*s* dummy variables are created and treated completely free in the analysis. The transformations for the independent variables from Step 4 are identical to the transformations of the proximities, except that the nonnegativety constraint does not apply. After transformation, the variables **q** are centered on the origin, normalized on *n*, and the reverse normalization is applied to the regression weights **b**.

Step 3: Transformation Update

The values of the transformed proximities are updated.

Conditionality

Two types of conditionalities exist in PROXSCAL. Conditionality refers to the possible comparison of proximities in the transformation step. For unconditional transformations, all proximities are allowed to be compared with each other, irrespective of the source. Matrix-conditional transformations only allow for comparison of proximities within one matrix *k*, in PROXSCAL refered to as one source *k*. Here, the transformation is computed for each source seperately (thus *m* times).

Transformation Functions

All transformation functions in PROXSCAL result in nonnegative values for the transformed proximities. After the transformation, the transformed proximities are normalized and the common space is optimally dilated accordingly. The following transformations are available.

Ratio. $\hat{\mathbf{D}} = \mathbf{\Delta}$. No transformation is necessary, since the scale of $\hat{\mathbf{D}}$ is adjusted in the normalization step.

Interval. $\hat{\mathbf{D}} = \alpha + \beta\mathbf{\Delta}$ Both α and β are computed using linear regression, in such a way that both parameters are nonnegative.

Ordinal. $\hat{\mathbf{D}}$ =WMON $(\mathbf{\Delta}, \mathbf{W})$. Weighted monotone regression (WMON) is computed using the up-and-down-blocks minimum violators algorithm (Kruskal, 1964; Barlow et al., 1972). For the secondary approach to ties, ties are kept tied, the proximities within tieblocks are first contracted and expanded afterwards.

Spline. vec$\left(\hat{\mathbf{D}}\right) = \mathbf{S}\mathbf{b}$ PROXSCAL uses monotone spline transformations (Ramsay, 1988). In this case, the spline transformation gives a smooth nondecreasing piecewise polynomial transformation. It is computed as a weighted regression of **D** on the spline basis **S**. Regression weights **b** are restricted to be nonnegative and computed using nonnegative alternating least squares (Groenen, van Os and Meulman, 2000).

Normalization

After transformation, the transformed proximities are normalized such that the sum-of-squares of the weighted transformed proximities are equal to $mn(n-1)/2$ in the unconditional case and equal to $n(n-1)/2$ in the matrix-conditional case.

Step 4: Termination

After evaluation of the loss function, the old function value and new function values are used to decide whether iterations should continue. If the new function value is smaller than or equal to the minimum Stress value MINSTRESS, provided by the user, iterations are terminated. Also, if the difference in consecutive Stress values is smaller than or equal to the convergence criterion DIFFSTRESS, provided by the user, iterations are terminated. Finally, iterations are terminated if the current number of iterations, exceeds the maximum number of iterations MAXITER, also provided by the user. In all other cases, iterations continue.

Acceleration

For the identity model without further restrictions, the common space can be updated with acceleration as $\mathbf{Z}^{\text{new}} = 2\mathbf{Z}^{\text{update}} - \mathbf{Z}^{\text{old}}$, also referred to as the relaxed update.

Lowering Dimensionality

For a restart in $p-1$ dimensions, the $p-1$ most important dimensions need to be identified. For the identity model, the first $p-1$ principal axes are used. For the weighted Euclidean model, the $p-1$ most important space weights are used, and for the generalized Euclidean and reduced rank models, the $p-1$ largest singular values of the space weights determine the remaining dimensions.

Stress Measures

The following statistics are used for the computation of the Stress measures:

$$\eta^2\left(\hat{\mathbf{D}}\right) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\hat{d}_{ijk}^2$$

$$\eta^4\left(\hat{\mathbf{D}}\right) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\hat{d}_{ijk}^4$$

$$\eta^2(\mathbf{X}) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\mathrm{d}_{ij}^2(\mathbf{X}_k)$$

$$\eta^4(\mathbf{X}) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\mathrm{d}_{ij}^4(\mathbf{X}_k)$$

$$\rho(\mathbf{X}) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\hat{d}_{ijk}\mathrm{d}_{ij}(\mathbf{X}_k)$$

$$\rho^2(\mathbf{X}) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\hat{d}_{ijk}^2\mathrm{d}_{ij}^2(\mathbf{X}_k)$$

$$\kappa^2(\mathbf{X}) = \sum_{k=1}^{m}\sum_{i<j}^{n} w_{ijk}\left(\mathrm{d}_{ij}(\mathbf{X}_k) - \bar{\mathrm{d}}(\mathbf{X})\right)^2$$

where $\bar{\mathrm{d}}(\mathbf{X})$ is the average distance.

The loss function minimized by PROXSCAL, normalized raw Stress, is given by:

$\sigma^2 = \frac{\eta^2(\hat{\mathbf{D}})+\eta^2(\alpha\mathbf{X})-2\rho(\alpha\mathbf{X})}{\eta^2(\hat{\mathbf{D}})}$, with $\alpha = \frac{\rho(\mathbf{X})}{\eta^2(\mathbf{X})}$.

Note that at a local minimum of **X**, α is equal to one. The other Fit and Stress measures provided by PROXSCAL are given by:

Stress-I: $\frac{\eta^2(\hat{\mathbf{D}})+\eta^2(\alpha\mathbf{X})-2\rho(\alpha\mathbf{X})}{\eta^2(\alpha\mathbf{X})}$, with $\alpha = \frac{\eta^2(\hat{\mathbf{D}})}{\rho(\mathbf{X})}$.

Stress-II: $\frac{\eta^2(\hat{\mathbf{D}})+\eta^2(\alpha\mathbf{X})-2\rho(\alpha\mathbf{X})}{\kappa^2(\alpha\mathbf{X})}$, with $\alpha = \frac{\eta^2(\hat{\mathbf{D}})}{\rho(\mathbf{X})}$.

S-Stress: $\eta^4\left(\hat{\mathbf{D}}\right) + \eta^4(\alpha\mathbf{X}) - 2\rho^2(\alpha\mathbf{X})$, with $\alpha^2 = \frac{\rho^2(\mathbf{X})}{\eta^4(\mathbf{X})}$.

Dispersion Accounted For (DAF): $1 - \sigma^2$.

Tucker's coefficient of congruence: $\sqrt{1-\sigma^2}$.

Decomposition of Normalized Raw Stress

Each part of normalized raw Stress, as described before, is assigned to objects and sources. Either sum over objects or sum over sources are equal to total normalized raw Stress.

Transformations on Output

On output, whenever fixed coordinates or independent variables do not apply, the models are not unique. In these cases transformations of the common space and the space weights are in order.

For the identity model, the common space $\mathbf{Z}$ is rotated to principal axes. For the weighted Euclidean model, $\mathbf{Z} = \sqrt{n}\mathbf{Z}\left(\text{diag}\,\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\right)^{-1/2}$ so that diag $\left(\mathbf{Z}^{\mathrm{T}}\mathbf{Z}\right) = n\mathbf{I}$, and reverse transformations are applied to the space weights $\mathbf{A}_k$. Further, the sum over sources of the squared space weights are put in descending order as to specify the importance of the dimensions. For the generalized Euclidean model, the Cholesky decomposition $\mathbf{Z}^{\mathrm{T}}\mathbf{Z} = \mathbf{L}\mathbf{L}^{\mathrm{T}}$ specifies the common space on output as $\mathbf{Z} = \sqrt{n}\mathbf{Z}\left(\mathbf{L}^{\mathrm{T}}\right)^{-1}$, so that $\mathbf{Z}^{\mathrm{T}}\mathbf{Z} = n\mathbf{I}$.

References

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions.* New York: John Wiley and Sons.

Bentler, P. M., and D. G. Weeks. 1978. Restricted multidimensional scaling models. *Journal of Mathematical Psychology*, 17, 138–151.

Bloxom, B. 1978. Contrained multidimensional scaling in n spaces. *Psychometrika*, 43, 397–408.

Carroll, J. D., and J. J. Chang. 1972. *IDIOSCAL (Individual differences in orientation scaling). Paper presented at the spring meeting of the Psychometric Society, Princeton, New Jersey.* : .

Commandeur, J. J. F., and W. J. Heiser. 1993. *Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices.* Leiden: Department of Data Theory, University of Leiden.

De Leeuw, J. 1977. Applications of convex analysis to multidimensional scaling. In: *Recent developments in statistics,* J. R. Barra, F. Brodeau, G. Romier, and B. van Cutsem, eds. Amsterdam,The Netherlands: North-Holland, 133–145.

De Leeuw, J., and W. J. Heiser. 1980. Multidimensional scaling with restrictions on the configuration. In: *Multivariate Analysis, Vol. V,* P. R. Krishnaiah, ed. Amsterdam: North-Holland, 501–522.

Gower, J. C. 1966. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53, 325–338.

Groenen, P. J. F., W. J. Heiser, and J. J. Meulman. 1999. Global optimization in least squares multidimensional scaling by distance smoothing. *Journal of Classification*, 16, 225–254.

Groenen, P. J. F., B. van Os, and J. J. Meulman. 2000. Optimal scaling by alternating length-constained nonnegative least squares, with application to distance-based analysis. *Psychometrika*, 65, 511–524.

Heiser, W. J. 1985. *A general MDS initialization procedure using the SMACOF algorithm-model with constraints: Technical Report No. RR-85-23*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J. 1987. Joint ordination of species and sites: The unfolding technique. In: *Developments in numerical ecology,* P. Legendre, and L. Legendre, eds. Berlin,Heidelberg: Springer-Verlag, 189–221.

Heiser, W. J., and J. De Leeuw. 1986. *SMACOF-I: Technical Report No. UG-86-02*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J., and I. Stoop. 1986. *Explicit SMACOF algorithms for individual differences scaling: Technical Report No. RR-86-14*. Leiden: Department of Data Theory, University of Leiden.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Meulman, J. J., and W. J. Heiser. 1984. Constrained Multidimensional Scaling: more Directions than Dimensions. In: *COMPSTAT 1984,* T. Havranek, ed. Wien: Physica Verlag, 137–142.

Ramsay, J. O. 1989. Monotone regression splines in action. *Statistical Science*, 4, 425–441.

Stoop, I., W. J. Heiser, and J. De Leeuw. 1981. *How to use SMACOF-IA*. Leiden: Department of Data Theory.

Stoop, I., and J. De Leeuw. 1982. *How to use SMACOF-IB*. Leiden: Department of Data Theory.

Torgerson, W. S. 1952. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17, 401–419.

Torgerson, W. S. 1958. *Theory and methods of scaling*. New York: Wiley.

Wolkowicz, H., and G. P. H. Styan. 1980. Bounds for eigenvalues using traces. *Linear algebra and its applications*, 29, 471–506.

QUICK CLUSTER Algorithms

When the desired number of clusters is known, QUICK CLUSTER groups cases efficiently into clusters.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 83-1
Notation

Notation	**Description**
NC	Number of clusters requested
$\mathbf{M}_i$	Mean of *i*th cluster
$\mathbf{x}_k$	Vector of *k*th observation
$d(\mathbf{x}_i, \mathbf{x}_j)$	Euclidean distance between vectors $\mathbf{x}_i$ and $\mathbf{x}_j$
d_{mn}	$\min_{i,j} d(\mathbf{M}_i, \mathbf{M}_j)$
ϵ	Convergence criteria

Algorithm

The first iteration involves three steps.

Step 1: Select Initial Cluster Centers

To select the initial cluster centers, a single pass of the data is made. The values of the first *NC* cases with no missing values are assigned as cluster centers, then the remaining cases are processed as follows:

- If $\min_i d(\mathbf{x}_k, \mathbf{M}_i) > d_{mn}$ and $d(\mathbf{x}_k, \mathbf{M}_m) > d(\mathbf{x}_k, \mathbf{M}_n)$, then $\mathbf{x}_k$ replaces $\mathbf{M}_n$. If $\min_i d(\mathbf{x}_k, \mathbf{M}_i) > d_{mn}$ and $d(\mathbf{x}_k, \mathbf{M}_m) < d(\mathbf{x}_k, \mathbf{M}_n)$, then $\mathbf{x}_k$ replaces $\mathbf{M}_m$; that is, if the distance between $\mathbf{x}_k$ and its closest cluster mean is greater than the distance between the two closest means ($\mathbf{M}_m$ and $\mathbf{M}_n$), then $\mathbf{x}_k$ replaces either $\mathbf{M}_m$ or $\mathbf{M}_n$, whichever is closer to $\mathbf{x}_k$.

- If $\mathbf{x}_k$ does not replace a cluster mean in (a), a second test is made:
  Let $\mathbf{M}_q$ be the closest cluster mean to $\mathbf{x}_k$.
  Let $\mathbf{M}_p$ be the second closest cluster mean to $\mathbf{x}_k$.
  If $d(\mathbf{x}_k, \mathbf{M}_p) > \min_i d(\mathbf{M}_q, \mathbf{M}_i)$, then $\mathbf{M}_q = \mathbf{x}_k$;
  That is, if $\mathbf{x}_k$ is further from the second closest cluster's center than the closest cluster's center is from any other cluster's center, replace the closest cluster's center with $\mathbf{x}_k$.

At the end of one pass through the data, the initial means of all *NC* clusters are set. Note that if NOINITIAL is specified, the first *NC* cases with no missing values are the initial cluster means.

Step 2: Update Initial Cluster Centers

Starting with the first case, each case in turn is assigned to the nearest cluster, and that cluster mean is updated. Note that the initial cluster center is included in this mean. The updated cluster means are the classification cluster centers.

Note that if NOUPDATE is specified, this step is skipped.

Step 3: Assign Cases to the Nearest Cluster

The third pass through the data assigns each case to the nearest cluster, where distance from a cluster is the Euclidean distance between that case and the (updated) classification centers. Final cluster means are then calculated as the average values of clustering variables for cases assigned to each cluster. Final cluster means do not contain classification centers.

When the number of iterations is greater than one, the final cluster means in step 3 are set to the classification cluster means in the end of step 2, and QUICK CLUSTER repeats step 3 again. The algorithm stops when either the maximum number of iterations is reached or the maximum change of cluster centers in two successive iterations is smaller than ε times the minimum distance among the initial cluster centers.

References

Hartigan, J. A. 1975. *Clustering algorithms*. New York: John Wiley and Sons.

RANK Algorithms

RANK produces new variables containing ranks, normal scores, and Savage and related scores for numeric variables.

Notation

Let $y_1 < y_2 < \cdots < y_m$ be *m* distinct ordered observations for the sample and $C_1, C_2, \ldots, C_m$ be the corresponding sum of caseweights for each value. Define

$$CC_i = \sum_{k=1}^{i} C_k = \text{cumulative sum of caseweights up to } y_i$$

$$W = CC_m = \sum_{k=1}^{m} C_k = \text{total sum of caseweights}$$

Statistics

The following statistics are available.

Rank

A rank is assigned to each case based on four different ways of treating ties or caseweights not equal to 1.

For every *i*, $i = 1, \ldots, m$,

(a) if $C_i \geq 1$

Calculation	**Condition**
$R_i = CC_{i-1} + 1$	if TIES = LOW
$R_i = CC_i$	if TIES = HIGH
$R_i = CC_{i-1} + (C_i + 1)/2$	if TIES = MEAN
$R_i = i$	if TIES = CONDENSE

(b) if $C_i < 1$

Calculation	**Condition**
$R_i = CC_{i-1}$	if TIES = LOW
$R_i = CC_i$	if TIES = HIGH
$R_i = CC_{i-1} + C_i/2$	if TIES = MEAN
$R_i = i$	if TIES = CONDENSE

Note: $CC_0 = 0$

RFRACTION

Fractional rank:

$$RF_i = R_i / W \ , \ i = 1, \ldots, m$$

PERCENT

Fractional rank as a percentage:

$$P_i = \frac{R_i}{W} \times 100 \ , \ i = 1, \ldots, m$$

PROPORTION Estimate for Cumulative Proportion

The proportion is calculated for each case based on four different methods of estimating fractional rank:

Calculation	**Method**
$F_i = (R_i - \frac{3}{8})/(W + \frac{1}{4})$	(BLOM)
$F_i = (R_i - \frac{1}{2})/W$	(RANKIT)
$F_i = (R_i - \frac{1}{3})/(W + \frac{1}{3})$	(TUKEY)
$F_i = R_i/(W + 1)$	(Van der Waerden)

Note: F_i will be set to SYSMIS if the calculated value of F_i by the formula is negative.

NORMAL (a)

Normal scores that are the *Z*-scores from the standard normal distribution that corresponds to the estimated cumulative proportion *F*. The normal score is defined by

$$a_i = \Psi(F_i) \ , \ i = 1, \ \ldots, m$$

where Ψ is the inverse cumulative standard normal distribution (PROBIT).

NTILES (K)

Assign group membership for the requested number of groups. If *K* groups are requested, the *n* tile (N_i) for case *i* is defined by

$$N_i = \left[\frac{R_i K}{W + 1} \right] + 1$$

where $\left[\frac{R_i K}{W+1}\right]$ is the greatest integer that is less than or equal to $R_i K/(W+1)$.

SAVAGE (S)

Savage scores based on exponential distribution. The Savage score is calculated by

$$S_i = \begin{cases} \left\{\left[(1-g_{i_1})l_{i_1+1} + g_{i_2}l_{i_2+1} + \sum_{j=i_1+2}^{i_2} l_j\right]/C_i\right\} - 1 & i_1+2 \le i_2 \\ \{[(1-g_{i_1})l_{i_1+1} + g_{i_2}l_{i_2+1}]/C_i\} - 1 & i_1+1 = i_2 \\ l_{i_1+1} - 1 & i_1 = i_2 \end{cases}$$

where

$$i_1 = [CC_{i-1}], \quad i_2 = [CC_i], \quad W^* = \begin{cases} W & \text{if } W \text{ is an integer} \\ [W]+1 & \text{if } W \text{ is not an integer} \end{cases}$$
$$g_{i_1} = CC_{i-1} - i_1, \quad g_{i_2} = CC_i - i_2$$

and $l_1, \ldots, l_{w^*}$ are defined as the expected values of the order statistics from an exponential distribution; that is

$$l_j = \sum_{K=1}^{j} \frac{1}{W^* - K + 1}$$

References

Blom, G. 1958. *Statistical estimates and transformed beta variables*. New York: John Wiley and Sons.

Chambers, J. M., W. S. Cleveland, B. Kleiner, and P. A. Tukey. 1983. *Graphical methods for data analysis*. Boston: Duxbury Press.

Lehmann, E. L. 1975. *Nonparametrics: Statistical methods based on ranks*. San Francisco: Holden-Day.

Tukey, J. W. 1962. The future of data analysis. *Annals of Mathematical Statistics*, 33:22, 1–67.

(Correction: 33:812)

RATIO STATISTICS Algorithms

This procedure provides a variety of descriptive statistics for the ratio of two variables.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 85-1
Notation

Notation	Description
n	Number of observations
A_i	Numerator of the *I*th ratio (*i*=1,...,*n*). This is usually the appraisal roll value.
S_i	Denominator of the *i*th ratio (*i*=1,...,*n*). This is usually the sale price.
R_i	The *i*th ratio (*i*=1,...,*n*). Often called the appraisal ratio.
f_i	Case weight associated with the *i*th ratio (*i*=1,...,*n*).

Data

This procedure requires for *i* = 1, …, *n* that:

- $A_i > 0$,
- $S_i > 0$,
- $f_i > 0$, and
- w_i is a whole number. If the Weight variable contains fractional values, then only the integral parts are used.

A case is considered valid if it satisfies all four requirements above. This procedure will use only valid cases in computing the requested statistics.

Ratio Statistics

The following statistics are available.

Ratio

$$R_i = \frac{A_i}{S_i}, i = 1, \ldots, n$$

Minimum

The smallest ratio and is denoted by $R_{\min}$.

Maximum

The largest ratio and is denoted by $R_{\max}$.

Range

The difference between the largest and the smallest ratios. It is equal to $R_{\max} - R_{\min}$.

Median

The middle number of the sorted ratios if *n* is odd. The mean (average) of the two middle ratios if the *n* is even. The median is denoted as $\tilde{R}$.

Average Absolute Deviation (AAD)

$$AAD = \sum_{i=1}^{n} f_i \left| R_i - \tilde{R} \right| / \sum_{i=1}^{n} f_i$$

Coefficient of Dispersion (COD)

$$COD = 100\% \times \frac{AAD}{\tilde{R}}$$

Coefficient of Concentration (COC)

Given a percentage 100% × *g*, the coefficient of concentration is the percentage of ratios falling within the interval $[\,(1-g)\tilde{R}, \quad (1+g)\tilde{R}\,]$. The higher this coefficient, the better uniformity.

Mean

$$\overline{A/S} = \overline{R} = \sum_{i=1}^{n} f_i R_i / \sum_{i=1}^{n} f_i$$

Standard Deviation (SD)

$$s = \sqrt{\frac{1}{(F-1)} \sum_{i=1}^{n} f_i \left(R_i - \overline{R}\right)^2}$$

where $F = \sum_{i=1}^{n} f_i$.

Coefficient of Variation (COV)

$$COV = 100\% \times \frac{s}{\overline{R}}$$

Weighted Mean

$$\overline{A}/\overline{S} = \frac{\sum_{i=1}^{n} f_i A_i}{\sum_{i=1}^{n} f_i S_i} = \frac{\sum_{i=1}^{n} f_i S_i R_i}{\sum_{i=1}^{n} f_i S_i}$$

This is the weighted mean of the ratios weighted by the sales prices in addition to the usual case weights.

Price Related Differential (a.k.a. Index of Regressivity)

$$PRD = \frac{\overline{A/S}}{\overline{A}/\overline{S}}$$

This is quotient by dividing the Mean by the Weighted Mean.

Property appraisals sometimes result in unequal tax burden between high-value and low-value properties in the same property group. Appraisals are considered *regressive* if high-value properties are under-appraised relative to low-value properties. On the contrary, appraisals are considered *progressive* if high-value properties are relatively over-appraised. The price related differential is a measure for measuring assessment regressivity or progressivity. Hence the price related differential is also known as the index of regressivity.

Recall that the [unweighted] mean weights the ratios equally, whereas the weighted mean high-value properties are under-appraised, thus pulling the weighted mean below the mean. On the other hand, if the PRD is less than 1, high-value properties are relatively over-appraised, pulling the weighted mean above the mean.

Confidence Interval for the Median

The confidence interval can be computed under the assumption that the ratios follow a normal distribution or nonparametrically.

Distribution free (nonparametric)

Given the confidence level $100\%\times(1-\alpha)$, the confidence interval for the median is an $\left(R_{[r]}, R_{[n-r+1]}\right)$ interval such that

$$1 - \alpha = 1 - 2I_{0.5}(n - r + 1, r) = \frac{1}{2^n}\sum_{k=r}^{n-r}\binom{n}{k},$$

where $R_{[k]}$ is the $100\%\times k/n$ quantile, and $I_{0.5}(n - r + 1, r)$ is the incomplete Beta function.

An equivalent formula is

$$\frac{\alpha}{2} = I_{0.5}(n - r + 1, r) = \frac{1}{2^n}\sum_{k=0}^{r-1}\binom{n}{k}.$$

Since the rightmost term is the cumulative Binomial distribution and it is discrete, *r* is solved as the largest value such that

$$\frac{\alpha}{2} \le \frac{1}{2^n}\sum_{k=0}^{r-1}\binom{n}{k}.$$

Thus the confidence interval has coverage probability of at least $1 - \alpha$.

Normal distribution

Assuming the ratios follow a normal distribution, a two-sided $100\%\times(1 - \alpha)$ confidence interval for a median of a normal distribution is

$$\left(\overline{R} + g_{(\alpha/2;0.5,d)} \times s, \overline{R} + g_{(1-\alpha/2;0.5,d)} \times s\right)$$

where $g_{(\gamma;p,d)}$ are values defined in Table 1 of Odeh and Owen (1980).

The value $g_{(\gamma;p,d)}$ is, in fact, the solution to the following equations:

$$\Pr\left(T_d \le g\sqrt{n} | \delta = K_p\sqrt{n}\right) = \gamma$$

with T_d follows a noncentral Student *t*-distribution where *d* is degrees of freedom associated with the standard deviation *s*, *δ* is noncentrality parameter, γ is the probability, *n* is the sample size, and K_p is the upper *p* percentile point of a standard normal distribution.

Confidence Interval for the Mean

The normal distribution is used to approximate the distribution of the ratios. The 100%×$(1 - \alpha)$ confidence interval for the mean is:

$$\overline{R} \pm t_{\alpha/2;F-1} \times s/\sqrt{F}$$

where $t_{\alpha/2;F-1}$ is the upper $\alpha/2$ percentage point of the *t* distribution with $F - 1$ degrees of freedom, and where $F = \sum_{i=1}^{n} f_i$.

Confidence Interval for the Weighted Mean

Using the Delta method, variance of the weighted mean is approximated as

$$var\left(\frac{\overline{A}}{\overline{S}}\right) \approx \frac{var\left(\overline{A}\right)}{\overline{S}^2} - \frac{2\overline{A}cov\left(\overline{A},\overline{S}\right)}{\overline{S}^3} + \frac{\overline{A}^2 var\left(\overline{S}\right)}{\overline{S}^4}.$$

where

$$var\left(\overline{A}\right) = \frac{1}{(F-1)}\sum_{i=1}^{n} f_i\left(A_i - \overline{A}\right)^2 \times \sum_{i=1}^{n} f_i^2/F^2,$$

$$var\left(\overline{S}\right) = \frac{1}{(F-1)}\sum_{i=1}^{n} f_i\left(S_i - \overline{S}\right)^2 \times \sum_{i=1}^{n} f_i^2/F^2, \text{ and}$$

$$cov\left(\overline{A},\overline{S}\right) = \frac{1}{(F-1)}\sum_{i=1}^{n} f_i\left(A_i - \overline{A}\right)\left(S_i - \overline{S}\right) \times \sum_{i=1}^{n} f_i^2/F^2.$$

References

International Association of Assessing Officers, . 1990. *Property Appraisal and Assessment Administration*. International Association of Assessing Officers: Chicago, Illinois.

Odeh, R. E., and D. B. Owen. 1980. *Tables for Normal Tolerance Limits, Sampling Plans, and Screening*. New York: Marcel Dekker.

RBF Algorithms

A radial basis function (RBF) network is a feed-forward, supervised learning network with only one hidden layer, called the radial basis function layer. The RBF network is a function of one or more predictors (also called inputs or independent variables) that minimizes the prediction error of one or more target variables (also called outputs). Predictors and targets can be a mix of categorical and scale variables.

Notation

The following notation is used throughout this chapter unless otherwise stated:

$X^{(m)} = \left(x_1^{(m)}, \ldots, x_P^{(m)}\right)$	Input vector, pattern *m*, *m*=1,..*M*.
$Y^{(m)} = \left(y_1^{(m)}, \ldots, y_R^{(m)}\right)$	Target vector, pattern *m*.
I	Number of layers, discounting the input layer. For an RBF network, *I*=2.
J_i	Number of units in layer *i*. $J_0 = P$, $J_1 = R$, discounting the bias unit. J_1 is the number of RBF units.
$\phi_j\left(X^{(m)}\right)$	*j*th RBF unit for input $X^{(m)}$, *j*=1, …J_1.
μ_j	center of ϕ_j, it is *P*-dimensional.
σ_j	width of ϕ_j, it is *P*-dimensional.
h	the RBF overlapping factor.
$a_{i:j}^m$	Unit *j* of layer *i*, pattern *m*, $j = 0, \ldots, J_i$; $i = 0, \ldots, I$.
w_{rj}	weight connecting *r*th output unit and *j*th hidden unit of RBF layer.

Architecture

There are three layers in the RBF network:

Input layer: J_0=*P* units, $a_{0:1}, \cdots, a_{0:J_0}$; with $a_{0:j} = x_j$.

RBF layer: J_1 units, , $a_{1:1}, \cdots, a_{1:J_1}$; with $a_{1:j} = \phi_j(X)$ and $\phi_j(X)$ described below.

Output layer: J_2=*R* units, $a_{I:1}, \cdots, a_{I:J_2}$; with $a_{I:r} = w_{r0} + \sum_{j=1}^{J_1} w_{rj}\phi_j(X)$.

There are many types of radial basis functions; there are two distinct types of Gaussian RBF architectures that we support:

Ordinary RBF (ORBF): This type uses the exp activation function, so the activation of the RBF unit is a Gaussian "bump" as a function of the inputs. In ORBF, the Gaussian basis function takes form

$$\phi_j(X) = \exp\left(-\sum_{p=1}^{P} \frac{1}{2\sigma_{jP}^2}(x_p - \mu_{jp})^2\right)$$

Normalized RBF (NRBF): This type uses the softmax activation function, so the activation of all the RBF units are normalized to sum to one. In NRBF networks, the basis function takes form

$$\phi_j\left(X\right) = \exp\left(-\sum_{p=1}^{P}\frac{1}{2\sigma_{jp}^2}\left(x_p-\mu_{jp}\right)^2\right)\Big/\sum_{j=1}^{J_1}\exp\left(-\sum_{p=1}^{P}\frac{1}{2\sigma_{jp}^2}\left(x_p-\mu_{jp}\right)^2\right)$$

Error Function

Sum-of-squares error is used:

$$E_T\left(w\right)=\sum_{m=1}^{M}E_m\left(w\right)$$

where

$$E_m\left(w\right)=\frac{1}{2}\sum_{r=1}^{R}\left(y_r^{(m)}-a_{I:r}^{m}\right)^2$$

The sum-of-squares error function with identity activation function for output layer can be used for both scale and categorical targets. For scale targets, $a_{I:r}^{m}$ approximates the conditional expectation of the target value $E\left(y_r|X^{(m)}\right)$. For categorical targets, $a_{I:r}^{m}$ approximates the posterior probability of class *k*: $P\left(y_r=1|X^{(m)}\right)$.

Note: though $\Sigma a_{I:r}^{m}=1$ (the sum is over all classes of the same categorical target variable), $a_{I:r}^{m}$ may not lie in the range [0, 1].

Training

The network is trained in two stages:

1. **Determine the basis functions by clustering methods.** The center and width for each basis function is computed.

2. **Determine the weights given the basis functions.** For the given basis functions, compute the ordinary least-squares regression estimates of the weights.

The simplicity of these computations allows the RBF network to be trained very quickly.

Determining Basis Functions

The two-step clustering algorithm is used to find the RBF centers and widths. For each cluster, the mean and standard deviation for each scale variable and proportion of each category for each categorical variable are derived. Using the results from clustering, the center of the *j*th RBF is set as:

$$\mu_{jp} = \begin{cases} \overline{x}_{jp} & \text{if pth variable is scale} \\ \pi_{jp} & \text{if pth variable is a dummy variable of a categorical variable} \end{cases}$$

where $\overline{x}_{jp}$ is the *j*th cluster mean of the *p*th input variable if it is scale, and π_{jp} is the proportion of the category of a categorical variable that the *p*th input variable corresponds to. The width of the *j*th RBF is set as

$$\sigma_{jp} = h^{1/2} \begin{cases} s_{jp} & \text{if pth variable is scale} \\ \sqrt{p_{jp}\left(1 - p_{jp}\right)} & \text{if pth variable is a dummy variable of a categorical variable} \end{cases}$$

where s_{jp} is the *j*th cluster standard deviation of the *p*th variable and *h*>0 is the RBF overlapping factor that controls the amount of overlap among the RBFs. Since some σ_{jp} may be zeros, we use spherical shaped Gaussian bumps; that is, a common width

$$\sigma_j = \sqrt{\frac{1}{P}\sum_{p=1}^{P} \sigma_{jp}^2}$$

in for all predictors. In the case that σ_j is zero for some *j*, set it to be $\min\{\sigma_j : \sigma_j \neq 0,\}_{j=1}^{J_1}$. If all σ_j are zero, set all of them to be $\sqrt{h}$.

When there are a large number of predictors, $\sum_{p=1}^{P} \left(x_p - \mu_{jp}\right)^2$ could be easily very large and hence $\exp\left(-\sum_{p=1}^{P} \frac{1}{2\sigma_j^2}\left(x_p - \mu_{jp}\right)^2\right)$ is practically zero for every record and every RBF unit if σ_j is relatively small. This is especially bad for ORBF because there would be only a constant term in the model when this happens. To avoid this, σ_j is increased by setting the default overlapping factor *h* proportional to the number of inputs: *h*=1 + 0.1 *P*.

For more information, see the topic “TWOSTEP CLUSTER Algorithms”.

Automatic Selection of Number of Basis Functions

The algorithm tries a reasonable range of numbers of hidden units and picks the “best”. By default, the reasonable range [K_1, K_2] is determined by first using the two-step clustering method to automatically find the number of clusters, *K*. Then set K_1 = min(*K*, *R*) for ORBF and K_1 =max{2, min(*K*, *R*)} for NRBF and K_2=max(10, 2*K*, *R*).

If a test data set is specified, then the "best" model is the one with the smaller error in the test data. If there is no test data, the BIC (Bayesian information criterion) is used to select the "best" model. The BIC is defined as

$$BIC = MR\ln(MSE) + k\ln(M)$$

where $MSE = \frac{1}{MR}\sum_{m=1}^{M}\sum_{r=1}^{R}\left(y_r^{(m)} - a_{I:r}^{m}\right)^2$ is the mean squared error and k= $(P+1+R)J_1$ for NRBF and $(P+1+R)J_1+R$ for ORBF is the number of parameters in the model.

Output Statistics

The following output statistics are available. Note that, for scale variables, output statistics are reported in terms of the rescaled values of the variables.

Sum-of-Squares Error

As described in "Error Function". The cross entropy error is displayed if the output layer activation function is softmax, otherwise the sum-of-squares error is shown.

Relative Error

For each scale target *r*:

$$\frac{\sum_{m=1}^{M}\left(y_r^{(m)} - \hat{y}_r^{(m)}\right)^2}{\sum_{m=1}^{M}\left(y_r^{(m)} - \overline{y}_r\right)^2}$$

For each categorical target *r*, report p_r, the percent of incorrect predictions

Average Overall Relative Error

If there is at least one scale target:

$$\frac{\sum_{m=1}^{M}\sum_{r=1}^{R}\left(y_r^{(m)} - \hat{y}_r^{(m)}\right)^2}{\sum_{m=1}^{M}\sum_{r=1}^{R}\left(y_r^{(m)} - \overline{y}_r\right)^2}$$

where $\overline{y}_r$ is the mean of $y_r^{(m)}$ over patterns.

If all targets are categorical, report the average percent of incorrect predictions:

$$\frac{1}{C}\sum_{r=1}^{C} p_r$$

where *C* is the number of categorical variables.

Sensitivity Analysis

For each predictor *p* and each input pattern *m*, compute:

$$d_{pm} = \max_{x_{p_1}, x_{p_2} \in S_p} \|\hat{Y}_{p_1}^{(m)} - \hat{Y}_{p_2}^{(m)}\|$$

where $\hat{Y}_{p_k}^{(m)}$ is the predicted output vector (standardized if standardization of output variable is used in training) using $\left(x_1^{(m)}, ..., x_{p-1}^{(m)}, x_{p_k}, x_{p+1}^{(m)}, ..., x_P^{(m)}\right)$ as its input, and $S_p = \left\{x_p^{\min}, x_p^{(2)}, x_p^{(3)}, x_p^{(4)}, x_p^{\max}\right\}$ for scale predictors and {(1,0,…,0),(0,1,0,…,0),…,(0,0,…,1)} for categorical predictors.
Then compute:

$$d_p = \frac{\ }{M}\sum_{m=1}^{M} d_{pm}$$

and normalize the d_ps to sum to 1, and report these normalized values as the sensitivity values for the predictors. This is the average maximum amount we can expect the output to change based on changes in the *p*th predictor. The greater the sensitivity, the more we expect the output to change when the predictor changes.

References

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition*, 3rd ed. Oxford: Oxford University Press.

Fine, T. L. 1999. *Feedforward Neural Network Methodology*, 3rd ed. New York: Springer-Verlag.

Haykin, S. 1998. *Neural Networks: A Comprehensive Foundation*, 2nd ed. New York: Macmillan College Publishing.

Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

Tao, K. K. 1993. A closer look at the radial basis function (RBF) networks. In: *Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems, and Computers,* A. Singh, ed. Los Alamitos, Calif.: IEEE Comput. Soc. Press, 401–405.

Uykan, Z., C. Guzelis, M. E. Celebi, and H. N. Koivo. 2000. Analysis of input-output clustering for determining centers of RBFN. *IEEE Transactions on Neural Networks*, 11, 851–858.

REGRESSION Algorithms

This procedure performs multiple linear regression with five methods for entry and removal of variables. It also provides extensive analysis of residual and influential cases. Caseweight (CASEWEIGHT) and regression weight (REGWGT) can be specified in the model fitting.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 87-1
Notation

Notation	Description
y_i	Dependent variable for case i with variance σ^2/g_i
c_i	Caseweight for case *i*; $c_i = 1$ if CASEWEIGHT is not specified
g_i	Regression weight for case *i*; $g_i = 1$ if REGWGT is not specified
l	Number of distinct cases
w_i	$c_i g_i$
W	$\sum_{i=1}^{l} w_i$
P	Number of independent variables
C	Sum of caseweights: $\sum_{i=1}^{l} c_i$
x_{ki}	The *k*th independent variable for case *i*
$\overline{X}_k$	Sample mean for the *k*th independent variable: $\overline{X}_k = \left(\sum_{i=1}^{l} w_i x_{ki}\right)/W$
$\overline{Y}$	Sample mean for the dependent variable: $\overline{Y} = \left(\sum_{i=1}^{l} w_i y_i\right)/W$
h_i	Leverage for case *i*
$\tilde{h}_i$	$\frac{g_i}{W} + h_i$
S_{kj}	Sample covariance for X_k and X_j
S_{yy}	Sample variance for *Y*
S_{ky}	Sample covariance for X_k and *Y*
p^*	Number of coefficients in the model. $p^* = p$ if the intercept is not included; otherwise $p^* = p + 1$
R	The sample correlation matrix for $X_1, \ldots, X_p$ and *Y*

Descriptive Statistics

$$\mathbf{R} = \begin{bmatrix} r_{11} & \cdots & r_{1p} r_{1y} \\ r_{21} & \cdots & r_{2p} r_{2y} \\ \cdot & \cdots & \cdot \ \cdot \\ r_{y1} & \cdots & r_{yp} r_{yy} \end{bmatrix}$$

where

$$r_{kj} = \frac{S_{kj}}{\sqrt{S_{kk} S_{jj}}}$$

and

$$r_{yk} = r_{ky} = \frac{S_{ky}}{\sqrt{S_{kk} S_{yy}}}$$

The sample mean $\overline{X}_i$ and covariance S_{ij} are computed by a provisional means algorithm. Define

$W_k = \sum_{i=1}^{k} w_i =$ cumulative weight up to case *k*.

then

$$\overline{X}_{i(k)} = \overline{X}_{i(k-1)} + \left(x_{ik} - \overline{X}_{i(k-1)}\right) \frac{w_k}{W_k}$$

where

$$\overline{X}_{i(1)} = x_{i1}$$

If the intercept is included,

$$C_{ij(k)} = C_{ij(k-1)} + \left(x_{ik} - \overline{X}_{i(k-1)}\right)\left(x_{jk} - \overline{X}_{j(k-1)}\right)\left(w_k - \frac{w_k^2}{W_k}\right)$$

where

$$C_{ij(1)} = 0$$

Otherwise,

$$C_{ij(k)} = C_{ij(k-1)} + w_k x_{ik} x_{jk}$$

where

$$C_{ij(1)} = w_1 x_{i1} x_{j1}$$

The sample covariance S_{ij} is computed as the final C_{ij} divided by $C-1$.

Sweep Operations (Dempster, 1969)

For a regression model of the form

$$Y_i = \beta_0 + \beta_1 X_{1i} + \beta_2 X_{2i} + \cdots + \beta_p X_{pi} + e_i$$

sweep operations are used to compute the least squares estimates **b** of β and the associated regression statistics. The sweeping starts with the correlation matrix **R**. Let $\tilde{\mathbf{R}}$ be the new matrix produced by sweeping on the *k*th row and column of **R**. The elements of $\tilde{\mathbf{R}}$ are

$$\begin{aligned} \tilde{r}_{kk} &= \frac{1}{r_{kk}} \\ \tilde{r}_{ik} &= \frac{r_{ik}}{r_{kk}}, \quad i \neq k \\ \tilde{r}_{kj} &= -\frac{r_{kj}}{r_{kk}}, \quad j \neq k \end{aligned}$$

and

$$\tilde{r}_{ij} = \frac{r_{ij} r_{kk} - r_{ik} r_{kj}}{r_{kk}}, \quad i \neq k, j \neq k$$

If the above sweep operations are repeatedly applied to each row of $\mathbf{R}_{11}$ in

$$\mathbf{R} = \begin{pmatrix} \mathbf{R}_{11} & \mathbf{R}_{12} \\ \mathbf{R}_{21} & \mathbf{R}_{22} \end{pmatrix}$$

where $\mathbf{R}_{11}$ contains independent variables in the equation at the current step, the result is

$$\tilde{\mathbf{R}} = \begin{pmatrix} \mathbf{R}_{11}^{-1} & -\mathbf{R}_{11}^{-1}\mathbf{R}_{12} \\ \mathbf{R}_{21}\mathbf{R}_{11}^{-1} & \mathbf{R}_{22} - \mathbf{R}_{21}\mathbf{R}_{11}^{-1}\mathbf{R}_{12} \end{pmatrix}$$

The last row of

$$\mathbf{R}_{21}\mathbf{R}_{11}^{-1}$$

contains the standardized coefficients (also called BETA), and

$$\mathbf{R}_{22} - \mathbf{R}_{21}\mathbf{R}_{11}^{-1}\mathbf{R}_{12}$$

can be used to obtain the partial correlations for the variables not in the equation, controlling for the variables already in the equation. Note that this routine is its own inverse; that is, exactly the same operations are performed to remove a variable as to enter a variable.

Note: When the stepwise or forward entry method is used, the variable order in the swept correlation matrix described above might differ from the variable order in the Swept Correlation Matrix table displayed in the output.

Variable Selection Criteria

Let r_{ij} be the element in the current swept matrix associated with X_i and X_j. Variables are entered or removed one at a time. X_k is eligible for entry if it is an independent variable not currently in the model with

$r_{kk} \geq t$ (tolerance with a default of 0.0001)

and also, for each variable X_j that is currently in the model,

$$\left(r_{jj} - \frac{r_{jk} r_{kj}}{r_{kk}} \right) t \leq 1$$

The above condition is imposed so that entry of the variable does not reduce the tolerance of variables already in the model to unacceptable levels.

The *F*-to-enter value for X_k is computed as

$$F - to - enter_k = \frac{(C - p^* - 1)V_k}{r_{yy} - V_k}$$

with 1 and $C - p^* - 1$ degrees of freedom, where p^* is the number of coefficients currently in the model and

$$V_k = \frac{r_{yk} r_{ky}}{r_{kk}}$$

The *F*-to-remove value for X_k is computed as

$$F - to - remove_k = \frac{(C - p^*)|V_k|}{r_{yy}}$$

with 1 and $C - p^*$ degrees of freedom.

Methods for Variable Entry and Removal

Five methods for entry and removal of variables are available. The selection process is repeated until the maximum number of steps (MAXSTEP) is reached or no more independent variables qualify for entry or removal. The algorithms for these five methods are described in the following sections.

Stepwise

If there are independent variables currently entered in the model, choose X_k such that $F - to - remove_k$ is minimum. X_k is removed if $F - to - remove_k < F_{out}$ (default = 2.71) or, if probability criteria are used, $P(F - to - remove_k) > P_{out}$ (default = 0.1). If the inequality does not hold, no variable is removed from the model.

If there are no independent variables currently entered in the model or if no entered variable is to be removed, choose X_k such that $F-to-enter_k$ is maximum. X_k is entered if $F-to-enter_k > F_{in}$ (default = 3.84) or, $P(F-to-enter_k) < P_{in}$ (default = 0.05). If the inequality does not hold, no variable is entered.

At each step, all eligible variables are considered for removal and entry.

Forward

This procedure is the entry phase of the stepwise procedure.

Backward

This procedure is the removal phase of the stepwise procedure and can be used only after at least one independent variable has been entered in the model.

Enter (Forced Entry)

Choose X_k such that r_{kk} is maximum and enter X_k. Repeat for all variables to be entered.

Remove (Forced Removal)

Choose X_k such that r_{kk} is minimum and remove X_k. Repeat for all variables to be removed.

Statistics

The following statistics are available.

Summary

For the summary statistics, assume *p* independent variables are currently entered in the equation, of which a block of *q* variables have been entered or removed in the current step.

Multiple R

$$R = \sqrt{1 - r_{yy}}$$

R Square

$$R^2 = 1 - r_{yy}$$

Adjusted R Square

$$R^2_{adj} = R^2 - \frac{\left(1 - R^2\right)p}{C - p^*}$$

R Square Change (when a block of q independent variables was added or removed)

$$\Delta R^2 = R^2_{current} - R^2_{previous}$$

F Change and Significance of F Change

$$\Delta F = \begin{cases} \frac{\Delta R^2(C-p^*)}{q(1-R^2_{current})} & \text{for the addition of } q \text{ independent variables} \\ \frac{\Delta R^2(C-p^*-q)}{q\left(R^2_{previous}-1\right)} & \text{for the removal of } q \text{ independent variables} \end{cases}$$

the degrees of freedom for the addition are q and $C - p^*$, while the degrees of freedom for the removal are q and $C - p^* - q$.

Residual Sum of Squares

$$SS_e = r_{yy}(C-1)S_{yy}$$

with degrees of freedom $C - p^*$.

Sum of Squares Due to Regression

$$SS_R = R^2(C-1)S_{yy}$$

with degrees of freedom p.

ANOVA Table

Table 87-2
ANOVA table

Analysis of Variance	***df***	***Sum of Squares***	***Mean Square***
Regression	p	SS_R	$(SS_R)/p$
w	$C-p^*$	SS_e	$(SS_e)/(C-p^*)$

Standard Error of Estimate

Also known as the standard error of regression, this is simply the square root of the mean square residual from the ANOVA table, or $\sqrt{(SS_e)/(C-p^*)}$.

Variance-Covariance Matrix for Unstandardized Regression Coefficient Estimates

A square matrix of size *p* with diagonal elements equal to the variance, the below diagonal elements equal to the covariance, and the above diagonal elements equal to the correlations:

$$var(b_k) = \frac{r_{kk} r_{yy} S_{yy}}{S_{kk}(C-p^*)}$$

$$cov(b_k, b_j) = \frac{r_{kj} r_{yy} S_{yy}}{\sqrt{S_{kk} S_{jj}}(C-p^*)}$$

$$cor(b_k, b_j) = \frac{r_{kj}}{\sqrt{r_{kk} r_{jj}}}$$

Selection Criteria

The following selection criteria are available.

Akaike Information Criterion (AIC)

$$AIC = C \ln\left(\frac{SS_e}{C}\right) + 2p^*$$

Amemiya's Prediction Criterion (PC)

$$PC = \frac{\left(1 - R^2\right)(C + p^*)}{C - p^*}$$

Mallow's CP

$$CP = \frac{SS_e}{\hat{\sigma}^2} + 2p^* - C$$

where $\hat{\sigma}^2$ is the mean square error from fitting the model that includes all variables specified or implied across all `METHOD` subcommands.

Schwarz Bayesian Criterion (SBC)

$$SBC = C \ln\left(\frac{SS_e}{C}\right) + p^* \ln(C)$$

Collinearity

The following measures of collinearity are available.

Variance Inflation Factors

$$VIF_i = \frac{1}{r_{ii}}$$

Tolerance

$$Tolerance_i = r_{ii}$$

Eigenvalues

The eigenvalues of scaled and uncentered cross-product matrix for the independent variables in the equation are computed by the QL method (Wilkinson and Reinsch, 1971).

Condition Indices

$$\eta_k = \frac{\max \lambda_j}{\lambda_k}$$

Variance-Decomposition Proportions

Let

$$\mathbf{v}_i = (v_{i1}, \ldots, v_{ip})$$

be the eigenvector associated with eigenvalue λ_i. Also, let

$$\Phi_{ij} = v_{ij}^2/\lambda_i \text{ and } \Phi_j = \sum_{i=1}^{p} \Phi_{ij}$$

The variance-decomposition proportion for the *j*th regression coefficient associated with the *i*th component is defined as

$$\pi_{ij} = \Phi_{ij}/\Phi_j$$

Statistics for Variables in the Equation

The following statistics are computed for each variable in the equation.

Regression Coefficient

$$b_k = \frac{r_{yk}\sqrt{S_{yy}}}{\sqrt{S_{kk}}} \text{ for } k = 1, \ldots, p$$

The standard error of b_k is computed as

$$\hat{\sigma}_{b_k} = \sqrt{\frac{r_{kk} r_{yy} S_{yy}}{S_{kk}(C - p^*)}}$$

95% confidence interval for coefficient

$$b_k \pm \hat{\sigma}_{b_k} t_{0.975, C-p^*}$$

If the model includes the intercept, the intercept is estimated as

$$b_0 = \overline{y} - \sum_{k=1}^{p} b_k \overline{X}_k$$

The variance of b_0 is estimated by

$$\hat{\sigma}_{b_0}^2 = \frac{(C-1) r_{yy} S_{yy}}{C(C-p^*)} + \sum_{k=1}^{p} \overline{X}_k^2 \hat{\sigma}_{b_k}^2 + 2 \sum_{k=j+1}^{p} \sum_{j=1}^{p-1} \overline{X}_k \overline{X}_j est.\text{cov}(b_k, b_j)$$

Beta Coefficients

$$Beta_k = r_{yk}$$

The standard error of $Beta_k$ is estimated by

$$\hat{\sigma}_{Beta_k} = \sqrt{\frac{r_{yy} r_{kk}}{C - p^*}}$$

F-test for $Beta_k$

$$F = \left(\frac{Beta_k}{\hat{\sigma}_{Beta_k}} \right)^2$$

with 1 and $C - p^*$ degrees of freedom.

Part Correlation

$$Part - Corr(X_k) = \frac{r_{yk}}{\sqrt{r_{kk}}}$$

Partial Correlation

$$Partial - Corr(X_k) = \frac{r_{yk}}{\sqrt{r_{kk} r_{yy} - r_{yk} r_{ky}}}$$

Statistics for Variables Not in the Equation

The following statistics are computed for each variable not in the equation.

Standardized regression coefficient Beta if predictor enters the equation at the next step

$$Beta_k^* = \frac{r_{yk}}{r_{kk}}$$

The *F*-test for $Beta_k^*$

$$F = \frac{(C - p^* - 1) r_{yk}^2}{r_{kk} r_{yy} - r_{yk}^2}$$

with 1 and $C - p^*$ degrees of freedom

Partial Correlation

$$Partial(X_k) = \frac{r_{yk}}{\sqrt{r_{yy} r_{kk}}}$$

Tolerance

$$Tolerance_k = r_{kk}$$

Minimum tolerance among variables already in the equation if predictor enters at the next step is

$$\min_{1 \le j \le p} \left(\frac{1}{r_{jj} - \left(r_{kj} r_{jk}\right) / r_{kk}}, r_{kk} \right)$$

Residuals and Associated Statistics

There are 19 temporary variables that can be added to the active system file. These variables can be requested with the `RESIDUAL` subcommand.

Centered Leverage Values

For all cases, compute

$$h_i = \begin{cases} \frac{g_i}{(C-1)} \sum_{j=1}^{p} \sum_{k=1}^{p} \frac{(X_{ji} - \overline{X}_j)(X_{ki} - \overline{X}_k) r_{jk}}{\sqrt{S_{jj} S_{kk}}} & \text{if intercept is included} \\ \frac{g_i}{(C-1)} \sum_{j=1}^{p} \sum_{k=1}^{p} \frac{X_{ji} X_{ki} r_{jk}}{\sqrt{S_{jj} S_{kk}}} & \text{otherwise} \end{cases}$$

For selected cases, leverage is h_i; for unselected case *i* with positive caseweight, leverage is

$$h'_i = \begin{cases} g_i \left[\left(\frac{1}{W} + h_i \right) / \left(1 + \frac{1}{W} + h_i \right) - \frac{1}{W+1} \right] & \text{if intercept is included} \\ h_i / (1 + h_i / g_i) & \text{otherwise} \end{cases}$$

Unstandardized Predicted Values

$$\hat{Y}_i = \begin{cases} \sum_{k=1}^{p} b_k X_{ki} & \text{if no intercept} \\ b_0 + \sum_{k=1}^{p} b_k X_{ki} & \text{otherwise} \end{cases}$$

Unstandardized Residuals

$$e_i = Y_i - \hat{Y}_i$$

Standardized Residuals

$$ZRESID_i = \begin{cases} \frac{e_i}{s} & \text{if no regression weight is specified} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where *s* is the square root of the residual mean square.

Standardized Predicted Values

$$ZPRED_i = \begin{cases} \frac{\hat{Y}_i - \overline{Y}}{sd} & \text{if no regression weight is specified} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

where *sd* is computed as

$$sd = \sqrt{\sum_{i=1}^{l} \frac{c_i \left(\hat{Y}_i - \overline{Y} \right)^2}{C - 1}}$$

Studentized Residuals

$$SRES_i = \begin{cases} \frac{e_i/s}{\sqrt{(1-\tilde{h}_i)/g_i}} & \text{for selected cases with } c_i > 0 \\ \frac{e_i/s}{\sqrt{(1+\tilde{h}_i)/g_i}} & \text{otherwise} \end{cases}$$

Deleted Residuals

$$DRESID_i = \begin{cases} e_i/\left(1-\tilde{h}_i\right) & \text{for selected cases with } c_i > 0 \\ e_i & \text{otherwise} \end{cases}$$

Studentized Deleted Residuals

$$SDRESID_i = \begin{cases} \frac{DRESID_i}{s^*_{(i)}} & \text{for selected cases with } c_i > 0 \\ \frac{e_i}{s\sqrt{(1+\tilde{h}_i)/g_i}} & \text{otherwise} \end{cases}$$

where

$$s^*_{(i)} = \frac{1}{\sqrt{C-p^*-1}}\sqrt{\frac{(C-p^*)s^2}{1-\tilde{h}_i} - DRESID_i^2}$$

Adjusted Predicted Values

$$ADJPRED_i = Y_i - DRESID_i$$

DfBeta

$$DFBETA_i = b - b(i) = \frac{g_i e_i \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{-1}\mathbf{X}_i^t}{1-\tilde{h}_i}$$

where

$$\mathbf{X}_i^t = \begin{cases} (1, X_{1i}, \ldots, X_{pi}) & \text{if intercept is included} \\ (X_{1i}, \ldots, X_{pi}) & \text{otherwise} \end{cases}$$

and $\mathbf{W} = diag(w_1, \ldots, w_l)$.

This is only computed for selected cases with case weight greater than or equal to 1.

Standardized DfBeta

$$SDBETA_{ij} = \frac{b_j - b_j(i)}{s_{(i)}\sqrt{(\mathbf{X}^t\mathbf{W}\mathbf{X})^{-1}_{jj}}}$$

where $b_j - b_j(i)$ is the *j*th component of $\mathbf{b} - \mathbf{b}(i)$, and

$$s_{(i)} = s^*_{(i)}\sqrt{1 - \tilde{h}_i}$$

This is only computed for selected cases with case weight greater than or equal to 1.

DfFit

$$DFFIT_i = \mathbf{X}_i[\mathbf{b} - \mathbf{b}(i)] = \frac{\tilde{h}_i e_i}{1 - \tilde{h}_i}$$

This is only computed for selected cases with case weight greater than or equal to 1.

Standardized DfFit

$$SDFIT_i = \frac{DFFIT_i}{s_{(i)}\sqrt{\tilde{h}_i}}$$

This is only computed for selected cases with case weight greater than or equal to 1.

Covratio

$$COVRATIO_i = \left(\frac{s_{(i)}}{s}\right)^{2p^*} \times \frac{1}{1 - \tilde{h}_i}$$

This is only computed for selected cases with case weight greater than or equal to 1.

Mahalanobis Distance

For selected cases with $c_i > 0$

$$MAHAL_i = \begin{cases} (C-1)h_i & \text{if intercept is included} \\ Ch_i & \text{otherwise} \end{cases}$$

For unselected cases with $c_i > 0$

$$MAHAL_i = \begin{cases} Ch'_i & \text{if intercept is included} \\ (C+1)h'_i & \text{otherwise} \end{cases}$$

Cook's Distance (Cook, 1977)

For selected cases with $c_i > 0$

$$COOK_i = \begin{cases} \left(DRESID_i^2 \tilde{h}_i g_i\right) / \left[s^2(p+1)\right] & \text{if intercept is included} \\ \left(DRESID_i^2 h_i g_i\right) / \left(s^2 p\right) & \text{otherwise} \end{cases}$$

For unselected cases with $c_i > 0$

$$COOK_i = \begin{cases} \left(DRESID_i^2 \left(h'_i + \frac{1}{W}\right)\right) / \left[\tilde{s}^2(p+1)\right] & \text{if intercept is included} \\ \left(DRESID_i^2 h'_i\right) / \left(\tilde{s}^2 p\right) & \text{otherwise} \end{cases}$$

where h'_i is the leverage for unselected case *i*, and $\tilde{s}^2$ is computed as

$$\tilde{s}^2 = \begin{cases} \frac{1}{C-p}\left[SS_e + e_i^2\left(1 - h'_i - \frac{1}{1+W}\right)\right] & \text{if intercept is included} \\ \frac{1}{C-p+1}\left[SS_e + e_i^2\left(1 - h'_i\right)\right] & \text{otherwise} \end{cases}$$

Standard Errors of the Mean Predicted Values

For all the cases with positive caseweight,

$$SEPRED_i = \begin{cases} s\sqrt{\tilde{h}_i / g_i} & \text{if intercept is included} \\ s\sqrt{h_i / g_i} & \text{otherwise} \end{cases}$$

95% Confidence Interval for Mean Predicted Response

$$LMCIN_i = \hat{Y}_i - t_{0.975,C-p^*} SEPRED_i$$
$$UMCIN_i = \hat{Y}_i + t_{0.975,C-p^*} SEPRED_i$$

95% Confidence Interval for a Single Observation

$$LICIN_i = \begin{cases} \hat{Y}_i - t_{0.975,C-p^*} s\sqrt{\left(\tilde{h}_i + 1\right)/g_i} & \text{if intercept is included} \\ \hat{Y}_i - t_{0.975,C-p} s\sqrt{(h_i + 1)/g_i} & \text{otherwise} \end{cases}$$

$$UICIN_i = \begin{cases} \hat{Y}_i + t_{0.975,C-p^*} s\sqrt{\left(\tilde{h}_i + 1\right)/g_i} & \text{if intercept is included} \\ \hat{Y}_i + t_{0.975,C-p} s\sqrt{(h_i + 1)/g_i} & \text{otherwise} \end{cases}$$

Durbin-Watson Statistic

$$DW = \frac{\sum_{i=2}^{l} \left(\tilde{e}_i - \tilde{e}_{i-1}\right)^2}{\sum_{i=1}^{l} c_i \tilde{e}_i^2}$$

where $\tilde{e}_i = e_i \sqrt{g_i}$.

Note: the Durbin-Watson statistic cannot be computed if there are fractional case weights. Even with integer case weights, the formula is only valid if the case weights represent contiguous case replications in the original sample.

Partial Residual Plots

The scatterplots of the residuals of the dependent variable and an independent variable when both of these variables are regressed on the rest of the independent variables can be requested in the RESIDUAL branch. The algorithm for these residuals is described in (Velleman and Welsch, 1981).

Missing Values

By default, a case that has a missing value for any variable is deleted from the computation of the correlation matrix on which all consequent computations are based. Users are allowed to change the treatment of cases with missing values.

References

Cook, R. D. 1977. Detection of influential observations in linear regression. *Technometrics*, 19, 15–18.

Dempster, A. P. 1969. *Elements of Continuous Multivariate Analysis*. Reading, MA: Addison-Wesley.

Velleman, P. F., and R. E. Welsch. 1981. Efficient computing of regression diagnostics. *American Statistician*, 35, 234–242.

Wilkinson, J. H., and C. Reinsch. 1971. Linear Algebra. In: *Handbook for Automatic Computation, Volume II,* J. H. Wilkinson, and C. Reinsch, eds. New York: Springer-Verlag.

RELIABILITY Algorithms

The RELIABILITY procedure employs one of two different computing methods, depending upon the MODEL specification and options and statistics requested.

Method 1 does not involve computing a covariance matrix. It is faster than method 2 and, for large problems, requires much less workspace. However, it can compute coefficients only for ALPHA and SPLIT models, and it does not allow computation of a number of optional statistics, nor does it allow matrix input or output. Method 1 is used only when alpha or split models are requested and only FRIEDMAN, COCHRAN, DESCRIPTIVES, SCALE, and/or ANOVA are specified on the STATISTICS subcommand and/or TOTAL is specified on the SUMMARY subcommand.

Method 2 requires computing a covariance matrix of the variables. It is slower than method 1 and requires more space. However, it can process all models, statistics, and options.

The two methods differ in one other important respect. Method 1 will continue processing a scale containing variables with zero variance and leave them in the scale. Method 2 will delete variables with zero variance and continue processing if at least two variables remain in the scale. If item deletion is required, method 2 can be selected by requesting the covariance method.

Notation

There are *N* persons taking a test that consists of *k* items. A score X_{ji} is given to the *j*th person on the *i*th item.

$$\begin{array}{c|cccccc|c}
 & 1 & 2 & \cdots & i & \cdots & k & \\
\hline
1 & X_{11} & X_{12} & & & & X_{1k} & P_1 \\
\vdots & & & & & & & \vdots \\
j & & & & X_{ji} & & & P_j \\
\vdots & & & & & & & \vdots \\
N & X_{N1} & X_{N2} & & & & X_{Nk} & P_N \\
\hline
 & T_1 & T_2 & \cdots & T_i & \cdots & T_k & G
\end{array}$$

If the model is SPLIT, k_1 items are in part 1 and $k_2 = k - k_1$ are in part 2. If the number of items in each part is not specified and *k* is even, the program sets $k_1 = k_2 = k/2$. If *k* is odd, $k_1 = (k+1)/2$. It is assumed that the first k_1 items are in part 1.

Table 88-1
Notation

Notation	**Description**
$W = \sum_{j=1}^{N} w_j$	Sum of the weights, where w_j is the weight for case *j*

Notation	**Description**
$P_j = \sum_{i=1}^{k} X_{ji}$	The total score of the *j*th person
$\overline{P}_j = P_j/k$	Mean of the observations for the *j*th person
$T_i = \sum_{j=1}^{N} X_{ji} w_j$	The total score for the *i*th item
$G = \sum_{i=1}^{k} \sum_{j=1}^{N} X_{ji} w_j$	Grand sum of the scores
$\overline{G} = \frac{G}{Wk}$	Grand mean of the observations

Scale and Item Statistics—Method 1

Item Means and Standard Deviations

Mean for the ith Item

$$\overline{T}_i = T_i/W$$

Standard Deviation for the ith Item

$$S_i = \sqrt{\frac{\sum_{j=1}^{N} w_j X_{ji}^2 - W\overline{T}_i^2}{W-1}}$$

Scale Mean and Scale Variance

Scale Mean

$$M = G/W$$

For the split model:

Mean Part 1

$$M_1 = \sum_{i=1}^{k_1} \overline{T}_i$$

Mean Part 2

$$M_2 = \sum_{i=k_1+1}^{k} \overline{T}_i$$

Scale Variance

$$S_p^2 = \frac{1}{(W-1)}\left[\sum_{j=1}^{N} P_j^2 w_j - W\left(\sum_{i=1}^{k} \overline{T}_i\right)^2\right]$$

For the split model:

Variance Part 1

$$S_{p1}^2 = \frac{1}{W-1}\left[\sum_{j=1}^{N} w_j\left(\sum_{i=1}^{k_1} X_{ji}\right) - W\left(\sum_{i=1}^{k_1} \overline{T}_i\right)^2\right]$$

Variance Part 2

$$S_{p2}^2 = \frac{1}{W-1}\left[\sum_{j=1}^{N} w_j\left(\sum_{i=k_1+1}^{k} X_{ji}\right)^2 - W\left(\sum_{i=k_1+1}^{k} \overline{T}_i\right)^2\right]$$

Item-Total Statistics

Scale Mean if the ith Item is Deleted

$$\tilde{M}_i = M - \overline{T}_i$$

Scale Variance if the ith Item is Deleted

$$\tilde{S}_i^2 = S_p^2 + S_i^2 - 2cov(X_i, P)$$

where the covariance between item *i* and the case score is

$$cov(X_i, P) = \frac{1}{W-1}\left(\sum_{j=1}^{N} P_j X_{ji} w_j - \sum_{l=1}^{k} \overline{T}_l T_i\right)$$

Alpha if the ith Item Deleted

$$\overline{A}_i = \frac{k-1}{k-2}\left(1 - \sum_{\substack{l=1 \\ l \neq i}}^{k} S_l^2/\tilde{S}_i^2\right)$$

Correlation between the ith Item and Sum of Others

$$R_i = \frac{cov(X_i, P) - S_i^2}{S_i \tilde{S}_i}$$

The ANOVA Table (Winer, 1971)

Table 88-2
ANOVA table

Source of variation	**Sum of Squares**	**df**
Between people	$\sum_{j=1}^{N} P_j^2 w_j/k - G^2/Wk$	$W-1$
Within people	$\sum_{i=1}^{k}\sum_{j=1}^{N} w_j X_{ji}^2 - \sum_{j=1}^{N} P_j^2 w_j/k$	$W(k-1)$
Between measures	$\sum_{i=1}^{k} T_i^2/W - G^2/Wk$	$k-1$
Residual	$\sum_{i=1}^{k}\sum_{j=1}^{N} w_j X_{ji}^2 - \sum_{j=1}^{N} P_j^2 w_j/k - \sum_{i=1}^{k} T_i^2/W - G^2/Wk$	$(W-1)(k-1)$
Total	$\sum_{i=1}^{k}\sum_{j=1}^{N} w_j X_{ji}^2 - G^2/Wk$	$Wk-1$

Each of the mean squares is obtained by dividing the sum of squares by the corresponding degrees of freedom. The *F* ratio for between measures is

$$F = \frac{MS_{\text{between measures}}}{MS_{\text{residual}}}, \quad df = (k-1, (W-1)(k-1))$$

Friedman Test or Cochran Test

$$\chi^2 = \frac{SS_{\text{between measures}}}{MS_{\text{within people}}}, \quad df = k-1$$

Note: Data must be ranks for the Friedman test and a dichotomy for the Cochran test.

Kendall’s Coefficient of Concordance

$$W = \frac{SS_{\text{between measures}}}{SS_{\text{total}}}$$

(Will not be printed if Cochran is also specified.)

Tukey’s Test for Nonadditivity

The residual sums of squares are further subdivided to

$$SS_{\text{nonadd}} = M^2/D, \quad df = 1$$

where

$$D = \left(SS_{\text{bet. meas}} SS_{\text{bet. people}}\right)/(Wk)$$
$$\left(= \left[\sum_{i=1}^{k} \left(\overline{T}_i - \overline{G}\right)^2\right]\left[\sum_{j=1}^{N} w_j\left(\overline{P}_j - \overline{G}\right)^2\right]\right)$$
$$M = \sum_{i=1}^{k} \overline{T}_i \sum_{j=1}^{N} \overline{P}_j X_{ji} w_j - \overline{G}\sum_{j=1}^{N} P_j^2 w_j/k - \overline{G}SS_{\text{bet. meas}}$$
$$\left(= \sum_{j=1}^{N} w_j\left(\overline{P}_j - \overline{G}\right)\left[\sum_{i=1}^{k} X_{ji}\left(\overline{T}_i - \overline{G}\right)\right]\right)$$
$$SS_{\text{bal}} = SS_{\text{res}} - SS_{\text{nonadd}}, \quad df = (W-1)(k-1)-1$$

The test for nonadditivity is

$$F = \frac{MS_{\text{nonadd}}}{MS_{\text{balance}}} \quad df = (1,(W-1)(k-1)-1)$$

The regression coefficient for the nonadditivity term is

$$\hat{B} = M/D,$$

and the power to transform to additivity is

$$\hat{p} = 1 - \hat{B}\overline{G}$$

Scale Statistics

Reliability coefficient alpha (Cronbach 1951)

$$A = \frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} S_i^2}{S_p^2}\right)$$

If the model is split, separate alphas are computed:

$$A_1 = \frac{k_1}{k_1-1}\left(1 - \sum_{i=1}^{k_1} \frac{S_i^2}{S_{p1}^2}\right)$$

$$A_2 = \frac{k_2}{k_2-1}\left(1 - \sum_{i=k_1+1}^{k_2} \frac{S_i^2}{S_{p2}^2}\right)$$

For Split Model Only

Correlation Between the Two Parts of the Test

$$R = \frac{\frac{1}{2}\left(S_p^2 - S_{p1}^2 - S_{p2}^2\right)}{S_{p1}S_{p2}}$$

Equal Length Spearman-Brown Coefficient

$$Y = \frac{2R}{1+R}$$

Guttman Split Half

$$G = \frac{2\left(S_p^2 - S_{p1}^2 - S_{p2}^2\right)}{S_p^2}$$

Unequal Length Spearman-Brown

$$ULY = \frac{-R^2 + \sqrt{R^4 + 4R^2(1-R^2)k_1k_2/k^2}}{2(1-R^2)k_1k_2/k^2}$$

Basic Computations—Method 2

Items with zero variance are deleted from the scale and from k, k_1, and k_2. The inverses of matrices, when needed, are computed using the sweep operator described by Dempster (1969). If $|V| < 10^{-30}$, a warning is printed and statistics that require V^{-1} are skipped.

Covariance Matrix V and Correlation Matrix R

$$v_{ij} = \begin{cases} \left(\frac{1}{W-1}\left(\sum_{l=1}^{N} X_{li}X_{lj}w_j - W\overline{T}_i\overline{T}_j\right), i,j = 1,\ldots,k\right) & \text{if raw data input} \\ r_{ij}S_iS_j & \text{if correlation matrix and SD input} \end{cases}$$

$r_{ij} = \frac{v_{ij}}{S_iS_i}$, where $S_i^2 = \frac{v_{ij}}{W-1}$

Scale Variance

$$S_p^2 = \sum_{i=1}^{k} S_i^2 + 2\sum_{i<}^{k}\sum_{j}^{k} v_{ij}$$

If the model is split,

$$S_{p1}^2 = \sum_{i=1}^{k_1} S_i^2 + 2\sum_{i<j}^{k_1}\sum^{k_1} v_{ij}$$

$$S_{p2}^2 = \sum_{i=k_1+1}^{k} S_i^2 + 2\sum_{i=k_1+1}^{k}\sum_{j>i}^{k} v_{ij}$$

where the first k_1 items are in part 1.

Scale Statistics—Method 2

Alpha Model

Estimated Reliability

$$\frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} S_i^2}{S_p^2}\right)$$

Standardized Item Alpha

$$\frac{k\overline{Corr}}{1+(k-1)\overline{Corr}}$$

where

$$\overline{\text{Corr}} = \frac{2}{k(k-1)} \sum_{i<j}^{k} \sum^{k} r_{ij}$$

Split Model

Correlation between Forms

$$\frac{\sum_{i=1}^{k_1} \sum_{j=k_1+1}^{k} v_{ij}}{S_{p1}S_{p2}}$$

Guttman Split-Half

$$G = \frac{\sum_{i=1}^{k_1} \sum_{j=k_1+1}^{k} v_{ij}}{S_p^2}$$

Alpha and Spearman-Brown equal and unequal length are computed as in method 1.

Guttman Model (Guttman 1945)

$$L_1 = 1 - \frac{\sum_{i=1}^{k} S_i^2}{S_p^2}$$

$$L_2 = L_1 + \frac{\sqrt{\frac{2k}{k-1} \sum_{i<j}^{k} \sum^{k} v_{ij}^2}}{S_p^2}$$

$$L_3 = \frac{k}{k-1} L_1$$

$$L_4 = \frac{4 \overset{k}{\Sigma} \underset{i<j}{\overset{k}{\Sigma}} v_{ij}^2}{S_p^2}$$

$$L_5 = L_1 + \frac{2\sqrt{\max_i \sum_{j \neq i}^{k} v_{ij}^2}}{S_p^2}$$

$$L_6 = 1 - \sum_{i=1}^{k} \epsilon_i^2 / S_p^2; \text{ where } \epsilon_i^2 = \left(V^{-1}\right)_{ii}^{-1}$$

Parallel Model (Kristof 1963)

Common Variance

$$CV = \overline{var} = \frac{1}{k}\sum_{i=1}^{k} S_i^2$$

True Variance

$$TV = \overline{cov} = \frac{2}{k(k-1)} \overset{k}{\Sigma} \underset{i<j}{\overset{k}{\Sigma}} v_{ij}$$

Error Variance

$$EV = \overline{var} - \overline{cov}$$

Common Inter-Item Correlation

$$\hat{R} = \overline{cov} / \overline{var}$$

Reliability of the Scale

$$A = \frac{k}{k-1}\left(1 - \frac{\sum_{i=1}^{k} S_i^2}{S_p^2}\right)$$

Unbiased Estimate of the Reliability

$$\hat{A} = \frac{2 + (W-3)A}{(W-1)}$$

where A is defined above.

Test for Goodness of Fit

$$\chi^2 = -(W-1)\left(1 - \frac{k(k+1)^2(2k-3)}{12(k-1)\left(\frac{k(k+1)}{2} - 2\right)(W-1)}\right)\log L$$

where

$$L = \frac{|V|}{(\overline{var} - \overline{cov})^{k-1}(\overline{var} + (k+1)\overline{cov})}$$

$$df = \frac{k(k+1)}{2} - 2$$

Log of the Determinant of the Unconstrained Matrix

$$\log UC = \log |V|$$

Log of the Determinant of the Constrained Matrix

$$\log C = \log\left((\overline{var} - \overline{cov})^{k-1}(\overline{var} + (k-1)\overline{cov})\right)$$

Strict Parallel (Kristof 1963)

Common Variance

$$CV = \overline{var} + \frac{1}{k}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2$$

Error Variance

$$EV = MS_{\text{within people}}$$

All mean squares are calculated as described in the analysis of variance table.

True Variance

$$TV = \overline{var} + \frac{1}{k}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2 - EV$$

Common Inter-Item Correlation

$$\hat{R} = \frac{\overline{cov} - \frac{1}{(k-1)k}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2}{\overline{var} + \frac{1}{k}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2}$$

Reliability of the Scale

$$Rel = \frac{k\hat{R}}{1 + (k-1)\hat{R}}$$

Unbiased Estimate of the Reliability

$$Rel = \frac{3 + (W-3)Rel}{W}$$

Test for Goodness of Fit

$$\chi^2 = -(W-1)\left(1 - \frac{k(k+1)^2(2k-3)}{12(k-1)(k(k+3)/2-3)(W-1)}\right)\log L$$

where

$$L = \frac{|V|}{(\overline{var} + (k-1)\overline{cov})\left(\overline{var} - \overline{cov} + \frac{1}{k}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2\right)^{k-1}}$$

$$df = k(k+3)/2 - 3$$

Log of the Determinant of the Unconstrained Matrix

$$\log UC = \log |V|$$

Log of the Determinant of the Constrained Matrix

$$\log C = \log\left(\overline{var} + (k-1)\overline{cov}\right)\left(\overline{var} - \overline{cov} + \frac{1}{k-1}\sum_{i=1}^{k}\left(\overline{T}_i - \overline{G}\right)^2\right)^{k-1}$$

Additional Statistics—Method 2

Descriptive and scale statistics and Tukey's test are calculated as in method 1. Multiple R^2 if an item is deleted is calculated as

$$\tilde{R}_i^2 = 1 - \frac{\epsilon_i^2}{S_i^2} \quad \epsilon_i^2 = \frac{1}{\left(V^{-1}\right)_{ii}}$$

Analysis of Variance Table

Table 88-3
Analysis of variance table

Source of variation	**Sum of Squares**	**df**
Between people	$(W-1)\left[\frac{1}{k}\left(\underset{i=1}{\overset{k}{\Sigma\Sigma}}\ S_i^2 - \frac{2}{k-1}\underset{i<j}{\overset{k\ \ k}{\Sigma\ \Sigma}}\ v_{ij}\right)\right] + \frac{2}{(k-1)}\sum_{i<j} v_{ij}$	$W-1$
Within people	$\frac{(W-1)(k-1)}{k}\left[\sum_{i=1}^{k} S_i^2 - \frac{2}{k\ -1}\underset{i<j}{\overset{k\ \ k}{\Sigma\ \Sigma}}\ v_{ij}\right] + (W-1)SS$bet. people	$W(k-1)$
Between measures	$W\left(\sum_{i=1}^{k}\overline{T}_i^2 - \frac{1}{k}\left(\sum_{i=1}^{k}\overline{T}_i\right)^2\right)$	$k-1$
Residual	$\frac{(W-1)(k-1)}{k}\left[\sum_{i=1}^{k} S_i^2 - \frac{2}{k-1}\underset{i<j}{\overset{k\ \ k}{\Sigma\ \Sigma}}\ v_{ij}\right]$	$(W-1)(k-1)$
Total	Between *SS* + Within *SS*	$Wk-1$

Hotelling's T-Squared (Winer, 1971)

$$T^2 = W\mathbf{Y}'\mathbf{B}^{-1}\mathbf{Y}$$

where

$$\mathbf{Y} = \begin{bmatrix} \overline{T}_1 - \overline{T}_k \\ \overline{T}_2 - \overline{T}_k \\ \vdots \\ \overline{T}_{k-1} - \overline{T}_k \end{bmatrix}$$
$$\mathbf{B} = \mathbf{CVC}'$$

where **C** is an identity matrix of rank $k-1$ augmented with a column of –1 on the right.

$$b_{ij} = v_{ij} - v_{ik} - v_{jk} + S_k^2$$

The test will not be done if $W < k$ or $|\mathbf{B}| < 10^{-30}$.

The significance level of T^2 is based on

$$F = \frac{W-k+1}{(W-1)(k-1)}T^2, \text{ with } df = (k-1, W-k+1)$$

Item Mean Summaries

$$\text{Mean} = \sum_{i=1}^{k} \overline{T}_i / k$$

$$\text{Variance} = \frac{\sum_{i=1}^{k} \overline{T}_i^2 - \left(\sum_{i=1}^{k} \overline{T}_i\right)^2 / k}{(k-1)}$$

$$\text{Maximum} = \max_i \overline{T}_i$$

$$\text{Minimum} = \min_i \overline{T}_i$$

$$Range = Maximum - Minimum$$

$$\text{Ratio} = \frac{\text{Maximum}}{\text{Minimum}}$$

Item Variance Summaries

Same as for item means except that S_i^2 is substituted for $\overline{T}_i$ in all calculations.

Inter-Item Covariance Summaries

$$\text{Mean} = \frac{\underset{i<j}{\overset{k}{\Sigma}\ \overset{k}{\Sigma}}\ v_{ij}}{k(k-1)}$$

$$\text{Variance} = \frac{1}{k(k-1)-1}\left[\underset{i<j}{\overset{k}{\Sigma}\ \overset{k}{\Sigma}}\ v_{ij} - \frac{1}{k(k-1)}\left(\underset{i<j}{\overset{k}{\Sigma}\ \overset{k}{\Sigma}}\ v_{ij}\right)^2\right]$$

$$\text{Maximum} = \max_{i,j} v_{ij}$$

$$\text{Minimum} = \min_{i,j} v_{ij}$$

$$Range = Maximum - Minimum$$

$$\text{Ratio} = \frac{\text{Maximum}}{\text{Minimum}}$$

Inter-Item Correlations

Same as for inter-item covariances, with v_{ij} being replaced by r_{ij}.

If the model is split, statistics are also calculated separately for each scale.

Intraclass Correlation Coefficients

Intraclass correlation coefficients are always discussed in a random/mixed effects model setting. McGraw and Wong (1996) is the key reference for this document. See also Shrout and Fleiss (1979).

In this document, two measures of correlation are given for each type under each model: **single measure** and **average measure**. Single measure applies to single measurements, for example, the ratings of judges, individual item scores, or the body weights of individuals, whereas average measure applies to average measurements, for example, the average rating for *k* judges, or the average score for a *k*-item test.

One-Way Random Effects Model: People Effect Random

Let X_{ji} be the response to the *i*th measure given by the *j*th person, $i = 1, \ldots, k$, $j = 1, \ldots, W$. Suppose that X_{ji} can be expressed as $X_{ji} = \mu + p_j + w_{ji}$, where p_j is the between-people effect which is normal distributed with zero mean and a variance of σ_p^2, and w_{ji} is the within-people effect which is also normal distributed with zero mean and a variance of σ_w^2.

Let MS_{BP} and MS_{WP} be the respective between-people Mean Squares and within-people Mean Squares. These two quantities can be computed by dividing the corresponding Sum of Squares with its degrees of freedom. For more information, see the topic “Analysis of Variance Table”.

Single Measure Intraclass Correlation

The single measure intraclass correlation is defined as

$$\rho_{(1)} = \frac{\sigma_p^2}{\sigma_p^2 + \sigma_w^2}$$

Estimate

The single measure intraclass correlation coefficient is estimated by

$$ICC(1) = \frac{MS_{\text{BP}} - MS_{\text{WP}}}{MS_{\text{BP}} + (k-1)MS_{\text{WP}}}.$$

In general,

$$\frac{-1}{k-1} < ICC(1) \le 1.$$

Confidence Interval

For 0 < α < 1, a (1- α)100% confidence interval for $\rho_{(1)}$ is given by

$$\frac{F_{p/w} - F_{\alpha/2, W-1, W(k-1)}}{F_{p/w} + (k-1)F_{\alpha/2, W-1, W(k-1)}} < \rho_{(1)} < \frac{F_{p/w} - F_{1-\alpha/2, W-1, W(k-1)}}{F_{p/w} + (k-1)F_{1-\alpha/2, W-1, W(k-1)}},$$

where

$$F_{p/w} = \frac{MS_{\text{BP}}}{MS_{\text{WP}}}$$

and F_{α', v_1, v_2} is the upper α' point of a *F*-distribution with degrees of freedom v_1 and v_2.

Hypothesis Testing

The test statistic $F^{(1)}$ for $H_0 : \rho_{(1)} = \rho_0$, where $1 > \rho_0 \ge 0$ is the hypothesized value, is

$$F^{(1)} = F_{p/w} \frac{1-\rho_0}{1+(k-1)\rho_0}.$$

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1, W(k-1)$ degrees of freedom.

Average Measure Intraclass Correlation

The average measure intraclass correlation is defined as

$$\rho_{(k)} = \frac{\sigma_p^2}{\sigma_p^2 + \sigma_w^2/k}.$$

Estimate

The average measure intraclass correlation coefficient is estimated by

$$ICC(k) = \frac{MS_{\text{BP}} - MS_{\text{WP}}}{MS_{\text{BP}}}.$$

Confidence Interval

A (1- α)100% confidence interval for $\rho_{(k)}$ is given by

$$\frac{F_{p/w} - F_{\alpha/2, W-1, W(k-1)}}{F_{p/w}} < \rho_{(k)} < \frac{F_{p/w} - F_{1-\alpha/2, W-1, W(k-1)}}{F_{p/w}}.$$

Hypothesis Testing

The test statistic $F^{(k)}$ for $H_0 : \rho_{(k)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$$F^{(k)} = F_{p/w}(1 - \rho_0).$$

Under the null hypothesis, the test statistic has an *F*-distribution with $W - 1, W(k - 1)$ degrees of freedom.

Two-Way Random Effects Model: People and Measures Effects Random

Let X_{ji} be the response to the *i*-th measure given by the *j*-th person, $i = 1, \ldots, k$, $j = 1, \ldots, W$. Suppose that X_{ji} can be expressed as $X_{ji} = \mu + p_j + m_i + pm_{ji} + e_{ji}$, where p_j is the *people effect* which is normal distributed with zero mean and a variance of σ_p^2, m_i is the *measures effect* which is normal distributed with zero mean and a variance of σ_m^2, pm_{ji} is the interaction effect which is normal distributed with zero mean and a variance of σ_{pm}^2, and e_{ji} is the error effect which is again normal distributed with zero mean and a variance of σ_e^2.

Let MS_{BP}, MS_{BM} and MS_{Res} be the respective between-people Mean Squares, between-measures Mean Squares and Residual Mean Squares. These quantities can be computed by dividing the corresponding Sum of Squares with its degrees of freedom. For more information, see the topic "Analysis of Variance Table".

Type A Single Measure Intraclass Correlation

The type A single measure intraclass correlation is defined as

$$\rho_{(A,1,r)} = \begin{cases} \frac{\sigma_p^2}{\sigma_p^2+\sigma_m^2+\sigma_{pm}^2+\sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2+\sigma_m^2+\sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type A single measure intraclass correlation coefficient is estimated by

$$ICC(A,1,r) = \frac{MS_{\text{BP}} - MS_{\text{Res}}}{MS_{\text{BP}} + (k-1)MS_{\text{Res}} + k\left(MS_{\text{BM}} - MS\right)/W}.$$

Notice that the same estimator is used whether or not the interaction effect pm_{ji} is present.

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{W\left(MS_{\text{BP}} - F_{\alpha/2,W-1,v} \cdot MS_{\text{Res}}\right)}{F_{\alpha/2,W-1,v}\left[k \cdot MS_{\text{BM}} + (kW-k-W)MS_{\text{Res}}\right] + W \cdot MS_{\text{BP}}} < \rho_{(A,1,r)} < \frac{W\left(MS_{\text{BP}} - F_{1-\alpha/2,W-1,v} \cdot MS_{\text{Res}}\right)}{F_{1-\alpha/2,W-1,v}\left[k \cdot MS_{\text{BM}} + (kW-k-W)MS_{\text{Res}}\right] + W \cdot MS_{\text{BP}}},$$

where

$$\nu = \frac{\left(aMS_{\text{BM}}+bMS_{\text{Res}}\right)^2}{\left[\frac{\left(aMS_{\text{BM}}\right)^2}{k-1}+\frac{\left(bMS_{\text{Res}}\right)^2}{(W-1)(k-1)}\right]}$$

$$a = \frac{k\cdot ICC(A,1,r)}{W(1-ICC(A,1,r))}$$

and

$$b = 1+\frac{k\cdot ICC(A,1,r)\cdot(W-1)}{W(1-ICC(A,1,r))}.$$

Hypothesis Testing

The test statistic $F^{(A,1,r)}$ for $H_0 : \rho_{(A,1,r)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$$F^{(A,1,r)} = \frac{MS_{\text{BP}}}{a_0 MS_{\text{BM}} + b_0 MS_{\text{Res}}}$$

where

$$a_0 = \frac{k\rho_0}{W(1-\rho_0)}$$

and

$$b_0 = 1+\frac{k\rho_0(W-1)}{W(1-\rho_0)}.$$

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1, \nu_0^{(1)}$ degrees of freedom.

$$\nu_0^{(1)} = \frac{\left(a_0MS_{\text{BM}}+b_0MS_{\text{Res}}\right)^2}{\left[\frac{\left(a_0MS_{\text{BM}}\right)^2}{k-1}+\frac{\left(b_0MS_{\text{Res}}\right)^2}{(W-1)(k-1)}\right]}.$$

Type A Average Measure Intraclass Correlation

The type A average measure intraclass correlation is defined as

$$\rho_{(A,k,r)} = \begin{cases} \frac{\sigma_p^2}{\sigma_p^2+\left(\sigma_m^2+\sigma_{pm}^2+\sigma_e^2\right)/k} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2+(\sigma_m^2+\sigma_e^2)/k} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type A average measure intraclass correlation coefficient is estimated by

$$ICC(A,k,r) = \frac{MS_{\text{BP}}-MS_{\text{Res}}}{MS_{\text{BP}}+\left(MS_{\text{BM}}-MS\right)/W}.$$

Notice that the same estimator is used whether or not the interaction effect pm_{ji} is present.

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{W\left(MS_{BP}-F_{\alpha/2,W-1,v}MS_{Res}\right)}{F_{\alpha/2,W-1,v}\left(MS_{BM}-MS_{Res}\right)+W\cdot MS_{BP}} < \rho_{(A,k,r)} < \frac{W\left(MS_{BP}-F_{1-\alpha/2,W-1,v}MS_{Res}\right)}{F_{1-\alpha/2,W-1,v}\left(MS_{BM}-MS_{Res}\right)+W\cdot MS_{BP}}$$

where

$$\nu=\frac{\left(cMS_{BM}+dMS_{Res}\right)^2}{\left[\frac{\left(cMS_{BM}\right)^2}{k-1}+\frac{\left(dMS_{Res}\right)^2}{(W-1)(k-1)}\right]}$$

$$c=\frac{ICC(A,k,r)}{W(1-ICC(A,k,r))}$$

and

$$d=1+\frac{ICC(A,k,r)\cdot(W-1)}{W(1-ICC(A,k,r))}.$$

Hypothesis Testing

The test statistic for $H_0:\rho_{(A,k,r)}=\rho_0$, where $1>\rho_0\geq 0$ is the hypothesized value, is

$$F^{(A,k,r)}=\frac{MS_{BP}}{c_0MS_{BM}+d_0MS_{Res}}$$

where

$$c_0=\frac{\rho_0}{W\left(1-\rho_0\right)}$$

and

$$d_0=1+\frac{\rho_0(W-1)}{W(1-\rho_0)}.$$

Under the null hypothesis, the test statistic has an F-distribution with $W-1,\nu_0^{(k)}$ degrees of freedom.

$$\nu_0^{(k)}=\frac{\left(c_0MS_{BM}+d_0MS_{Res}\right)^2}{\left[\frac{\left(c_0MS_{BM}\right)^2}{k-1}+\frac{\left(d_0MS_{Res}\right)^2}{(W-1)(k-1)}\right]}.$$

Type C Single Measure Intraclass Correlation

The type C single measure intraclass correlation is defined as

$$\rho_{(C,1,r)} = \begin{cases} \frac{\sigma_p^2}{\sigma_p^2+\sigma_{pm}^2+\sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2+\sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type C single measure intraclass correlation coefficient is estimated by

$$ICC\left(C,1,r\right) = \frac{MS_{\text{BP}}-MS_{\text{Res}}}{MS_{\text{BP}}+(k-1)MS_{\text{Res}}}.$$

Notice that the same estimator is used whether or not the interaction effect *pm*$_{\text{ji}}$ is present.

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{F_{p/r}-F_{\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r}+(k-1)F_{\alpha/2,W-1,(W-1)(k-1)}} < \rho_{(C,1,r)} < \frac{F_{p/r}-F_{1-\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r}+(k-1)F_{1-\alpha/2,W-1,(W-1)(k-1)}}$$

where

$$F_{p/r} = \frac{MS_{\text{BP}}}{MS_{\text{Res}}}.$$

Hypothesis Testing

The test statistic for $H_0 : \rho_{(C,1,r)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$$F^{(C,1,r)} = F_{p/r}\frac{1-\rho_0}{1+(k-1)\rho_0}.$$

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1, (W-1)(k-1)$ degrees of freedom.

Type C Average Measure Intraclass Correlation

The type C average measure intraclass correlation is defined as

$$\rho_{(C,k,r)} = \begin{cases} \frac{\sigma_p^2}{\sigma_p^2+\left(\sigma_{pm}^2+\sigma_e^2\right)/k} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2+\sigma_e^2/k} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type C average measure intraclass correlation coefficient is estimated by

$$ICC\left(C,k,r\right) = \frac{MS_{\text{BP}}-MS_{\text{Res}}}{MS_{\text{BP}}}.$$

Notice that the same estimator is used whether or not the interaction effect *pm*$_{\text{ji}}$ is present.

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{F_{p/r}-F_{\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r}} < \rho_{(C,k,r)} < \frac{F_{p/r}-F_{1-\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r}}.$$

Hypothesis Testing

The test statistic for $H_0 : \rho_{(C,k,r)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$$F^{(C,k,r)} = F_{p/r}\left(1-\rho_0\right).$$

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1,(W-1)(k-1)$ degrees of freedom.

Two-Way Mixed Effects Model: People Effects Random, Measures Effects Fixed

Let X_{ji} be the response to the *i*-th measure given by the *j*-th person, $i = 1, \ldots, k$, $j = 1, \ldots, W$. Suppose that X_{ji} can be expressed as $X_{ji} = \mu + p_j + m_i + pm_{ji} + e_{ji}$, where p_j is the *people effect* which is normal distributed with zero mean and a variance of σ_p^2, m_i is considered as a fixed effect, pm_{ji} is the interaction effect which is normal distributed with zero mean and a variance of σ_{pm}^2, and e_{ji} is the error effect which is again normal distributed with zero mean and a variance of σ_e^2. Denote θ_m^2 as the expected measure square of between measures effect m_i.

Let MS_{BP} and MS_{Res} be the respective between-people Mean Squares and Residual Mean Squares. These quantities can be computed by dividing the corresponding Sum of Squares with its degrees of freedom. For more information, see the topic "Analysis of Variance Table".

Type A Single Measure Intraclass correlation

The type A single measure intraclass correlation is defined as

$$\rho_{(A,1,m)} = \begin{cases} \dfrac{\sigma_p^2-\sigma_{pm}^2/(k-1)}{\sigma_p^2+\theta_m^2+\left(\sigma_{pm}^2+\sigma_e^2\right)} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \dfrac{\sigma_p^2}{\sigma_p^2+\theta_m^2+\sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type A single measure intraclass correlation is estimated by

$$ICC(A,1,m) = \frac{MS_{\text{BP}}-MS_{\text{Res}}}{MS_{\text{BP}}+(k-1)MS_{\text{Res}}+k\left(MS_{\text{BM}}-MS_{\text{Res}}\right)/W}.$$

Notice that the same estimator is used whether or not the interaction effect pm_{ji} is present.

Confidence Interval

A (1- α)100% confidence interval for $\rho_{(A,1,m)}$ is the same as that for $\rho_{(A,1,r)}$, with $ICC(A,1,r)$ replaced by $ICC(A,1,m)$.

Hypothesis Testing

The test statistic for $H_0 : \rho_{(A,1,m)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is the same as that for $\rho_{(A,1,r)}$, with the same distribution under the null hypothesis.

Type A Average Measure Intraclass Correlation

The type A average measure intraclass correlation is defined as

$$\rho_{(A,k,m)} = \begin{cases} \frac{\sigma_p^2 - \sigma_{pm}^2/(k-1)}{\sigma_p^2 + \left(\theta_m^2 + \sigma_{pm}^2 + \sigma_e^2\right)/k} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2 + (\theta_m^2 + \sigma_e^2)/k} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases} .$$

Estimate

The type A single measure intraclass correlation is estimated by

$$ICC(A,k,m) = \begin{cases} \text{Not estimable} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{MS_{\text{BP}} - MS_{\text{Res}}}{MS_{\text{BP}} + \left(MS_{\text{BM}} - MS_{\text{Res}}\right)/W} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases} .$$

Confidence Interval

A (1- α)100% confidence interval for $\rho_{(A,k,m)}$ is the same as that for $\rho_{(A,k,r)}$, with $ICC(A,k,r)$ replaced by $ICC(A,k,m)$. Notice that the hypothesis test is not available when the interaction effect pm_{ji} is present.

Hypothesis Testing

The test statistic for $H_0 : \rho_{(A,k,m)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is the same as that for $\rho_{(A,k,r)}$, with the same distribution under the null hypothesis. Notice that the hypothesis test is not available when the interaction effect pm_{ji} is present.

Type C Single Measure Intraclass Correlation

The type C single measure intraclass correlation is defined as

$$\rho_{(C,1,m)} = \begin{cases} \frac{\sigma_p^2 - \sigma_{pm}^2/(k-1)}{\sigma_p^2 + \left(\sigma_{pm}^2 + \sigma_e^2\right)} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2 + \sigma_e^2} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases} .$$

Estimate

The type C single measure intraclass correlation is estimated by

$$ICC(C,1,m) = \frac{MS_{\text{Between people}} - MS_{\text{Residual}}}{MS_{\text{Between people}} + (k-1)MS_{\text{Residual}}}.$$

Notice that the same estimator is used whether or not the interaction effect pm_{ji} is present.

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{F_{p/r} - F_{\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r} + (k-1)F_{\alpha/2,W-1,(W-1)(k-1)}} < \rho_{(C,1,m)} < \frac{F_{p/r} - F_{1-\alpha/2,W-1,(W-1)(k-1)}}{F_{p/r} + (k-1)F_{1-\alpha/2,W-1,(W-1)(k-1)}}.$$

where

$F_{p/r} = \frac{MS_{\text{BP}}}{MS_{\text{Res}}}$.

Hypothesis Testing

The test statistic for $H_0 : \rho_{(C,1,m)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$F^{(C,1,m)} = F_{p/r} \frac{1-\rho_0}{1+(k-1)\rho_0}$.

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1, (W-1)(k-1)$ degrees of freedom.

Type C Average Measure Intraclass Correlation

The type C average measure intraclass correlation is defined as

$$\rho_{(C,k,m)} = \begin{cases} \frac{\sigma_p^2 - \sigma_{pm}^2/(k-1)}{\sigma_p^2 + \left(\sigma_{pm}^2 + \sigma_e^2\right)/k} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{\sigma_p^2}{\sigma_p^2 + \sigma_e^2/k} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Estimate

The type C average measure intraclass correlation is estimated by

$$ICC(C,k,m) = \begin{cases} \text{Not estimable} & \text{if interaction effect } pm_{ji} \text{ is present} \\ \frac{MS_{\text{BP}} - MS_{\text{Res}}}{MS_{\text{BP}}} & \text{if interaction effect } pm_{ji} \text{ is absent} \end{cases}.$$

Confidence Interval

A (1- α)100% confidence interval is given by

$$\frac{F_{p/r} - F_{\alpha/2, W-1, (W-1)(k-1)}}{F_{p/r}} < \rho_{(C,k,m)} < \frac{F_{p/r} - F_{1-\alpha/2, W-1, (W-1)(k-1)}}{F_{p/r}}.$$

Notice that the confidence interval is not available when the interaction effect pm_{ji} is present.

Hypothesis Testing

The test statistic for $H_0 : \rho_{(C,k,m)} = \rho_0$, where $1 > \rho_0 \geq 0$ is the hypothesized value, is

$F^{(C,1,m)} = F_{p/r}(1 - \rho_0)$.

Under the null hypothesis, the test statistic has an *F*-distribution with $W-1, (W-1)(k-1)$ degrees of freedom. Notice that the *F*-test is not available when the interaction effect pm_{ji} is present.

References

Cronbach, L. J. 1951. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16:3, 297–334.

Dempster, A. P. 1969. *Elements of Continuous Multivariate Analysis*. Reading, MA: Addison-Wesley.

Guttman, L. 1945. A basis for analyzing test-retest reliability. *Psychometrika*, 10:4, 255–282.

Kristof, W. 1963. The statistical theory of stepped-up reliability coefficients when a test has been divided into several equivalent parts. *Psychometrika*, 28:3, 221–238.

Kristof, W. 1969. Estimation of true score and error variance for tests under various equivalence assumptions. *Psychometrika*, 34:4, 489–507.

McGraw, K. O., and S. P. Wong. 1996. Forming inferences about some intraclass correlation coefficients. *Psychological Methods*, 1:1, 30–46.

Novick, M. R., and C. Lewis. 1967. Coefficient alpha and the reliability of composite measurements. *Psychometrika*, 32:1, 1–13.

Shrout, P. E., and J. L. Fleiss. 1979. Intraclass correlations: Uses in assessing reliability. *Psychological Bulletin*, 86, 420–428.

Winer, B. J. 1971. *Statistical principles in experimental design*, 2nd ed. New York: McGraw-Hill.

RMV Algorithms

Missing values in a time series are estimated.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 89-1
Notation

Notation	Description
$X=(X_1,\ldots,X_n)$	Original series
$\hat{X}_i$	Estimate for spans
p	Number of spans
k	The number of consecutive missing values
X_i to X_{i+k-1}	Set of consecutive missing values

Methods for Estimating Missing Values

The following methods are available.

Linear Interpolation (LINT(X))

$$\hat{X}_{i+l}=\begin{cases} X_{i-1}+\frac{l+1}{k+1}(X_{i+k}-X_{i-1}) & l=0,\ldots,k-1 \\ \text{SYSMIS} & i=1 \text{ or } i+k-1=n \end{cases}$$

If $k=1$ (that is, only one consecutive missing observation), then

$$\hat{X}_i=\begin{cases} \frac{1}{2}(X_{i-1}+X_{i+1}) & i=2,\ldots,n-1 \\ \text{SYSMIS} & i=1 \text{ or } i=n \end{cases}$$

Mean of p Nearest Preceding and p Subsequent Values (MEAN (X,p))

If the number of nonmissing observations in $(X_1,\ldots,X_{i-1})$ or $(X_{i+k},\ldots,X_n)$ is less than *p*, then set $\hat{X}_{i+l}=$ SYSMIS; otherwise, set $\hat{X}_{i+l}=$ average of *p* nonmissing observations preceding X_i and *p* nonmissing observations following X_{i+k-1}.

Median of p Nearest Preceding and p Subsequent Values (MEDIAN (X,p))

If the number of nonmissing observations in $(X_1,\ldots,X_{i-1})$ or $(X_{i+k},\ldots,X_n)$ is less than *p*, then set $\hat{X}_{i+l}$ = SYSMIS; otherwise, set $\hat{X}_{i-1}$ median of *p* nonmissing observations preceding X_i and *p* nonmissing observations following X_{i+k-1}.

Series Mean (SMEAN (X))

$\hat{X}_{i+l} =$ average of all nonmissing observations in the series.

Linear Trend (TREND(X))

1. Use all the nonmissing observations in the series to fit the regression line of the form

   $$\hat{X}_t = a + bt$$

   The least squares estimates are

   $$b = \frac{\Sigma\left(X_t - \overline{X}\right)\left(t - \bar{t}\right)}{\Sigma\left(t - \bar{t}\right)^2}$$
   $$a = \overline{X} - b\bar{t}$$

2. Apply the regression equation to replace the missing values

   $$\hat{X}_{i+l} = a + b(i + l)$$

ROC Algorithms

ROC produces a receiver operating characteristic (ROC) curve.

Notation and Definitions

Table 90-1
Notation

Notation	Description
d_i	Actual state for case *i*, it is either positive or negative; positive usually means that a test detected some evidence for a condition to exist.
x_i	Test result score for case *i*.
n_{TP}	Number of true positive decisions
n_{FN}	Number of false negative decisions
n_{TN}	Number of true negative decisions
n_{FP}	Number of false positive decisions
Sensitivity	Probability of correctly identifying a positive
Specificity	Probability of correctly identifying a negative
C	Cutoff or criterion value on the test result variable
n_-	Number of cases with negative actual state
n_+	Number of cases with positive actual state
$n_{-=j}$	Number of true negative cases with test result equal to *j*.
$n_{+>j}$	Number of true positive cases with test result greater than *j*.
$n_{+=j}$	Number of true positive cases with test result equal to *j*.
$n_{-<j}$	Number of true negative cases with test result less than *j*.
Q_1	The probability that two randomly chosen positive state subjects will both get a more positive test result than a randomly chosen negative state subject.
Q_2	The probability that one randomly chosen positive state subject will get a more positive test result than two randomly chosen negative state subjects.

Construction of the ROC Curve

The ROC plot is merely the graph of points defined by sensitivity and (1 – specificity). Customarily, sensitivity takes the y axis and (1 – specificity) takes the x axis.

Computation of Sensitivity and Specificity

The ROC procedure fixes the set of cutoffs to be the set defined by the values half the distance between each successive pair of observed test scores, plus $max(x_i) + 1$ and $min(x_i) - 1$.

Given a set of cutoffs, the actual state values, and test result values, one can classify each observation into one of TP, FN, TN, and FP according to a classification rule. Then, the computation of sensitivity and specificity is immediate from their definitions.

Four classification or decision rules are possible:

Table 90-2
Classification of decision rules

ClassificRaetsiounlt	
(1)	a test result is positive if the test result value is greater than or equal to C and that a test result is negative if the test result is less than C;
(2)	a test result is positive if the test result value is greater than C and that a test result is negative if the test result is less than or equal to C;
(3)	a test result is positive if the test result value is less than or equal to C and that a test result is negative if the test result is greater than C; and
(4)	a test result is positive if the test result value is less than C and that a test result is negative if the test result is greater than or equal to C.

Specificity

Specificity is defined by

$$\frac{n_{\text{TN}}}{n_{\text{TN}} + n_{\text{FP}}}$$

Sensitivity

Sensitivity is defined by

$$\frac{n_{\text{TP}}}{n_{\text{TP}} + n_{\text{FN}}}$$

Interpolation of the Points

When the test result variable is a scale variable, the number of distinct test result values and thus the number of cutoff points tend to increase as the number of observations (or test results) increases. Theoretically, in the “limit” the pairs of sensitivity and (1 – specificity) values form a dense set of points in itself and in some continuous curve, the ROC curve. A continuous interpolation of the points may be reasonable in this sense.

Note: The domain of the test result variable need only be a positive-measure subset of the real line. For example, it could be defined only on (-1, 0] and (1, $+\infty$). As long as the variable is not discrete, the ROC curve will be continuous.

When the test result variable is an ordinal discrete variable, the points never become dense, even when there are countably infinite number of (ordinal discrete) values. Thus, a continuous interpolation may not be justifiable. But, when it is reasonable to assume there is some underlying or latent continuous variable, an interpolation such as a linear interpolation, though imprecise, may be attempted. From now on, the test result variable is assumed continuous or practically so.

The problem is related to having ties, but not the same. In the continuous case, when values are tied, they are identical but unique. In the ordinal case with the grouped/discretized continuous interpretation, values in some underlying continuous scale range may be grouped together and represented by a certain value, usually the mid range value. Those values are represented as if they were ties, but in fact they are a collection of unordered values. Now, even if each category/group contains only one observation, the problem still exists unless the observation’s latent value is identical to the representing value of the observation.

Case 1: No ties between actual positive and actual negative groups

If there are ties within a group, the vertical/horizontal distance between the points is simply multiplied by the number of ties. If not, all the points are uniformly spaced within each of the vertical and horizontal directions, because as a cutoff value changes, only one observation at a time switches the test result.

Case 2: Some ties between actual positive and actual negative groups

For ties between actual positive and actual negative groups, both of the n_{TP} and n_{FP} change simultaneously, and we do not know “the correct path between two adjacent points” (Zweig and Campbell, 1993, p. 566). “It could be the minimal path (horizontal first, then vertical) or the maximal path (vice versa). The straight diagonal line segment is the average of the two most extreme paths and tends to underestimate the plot for diagnostically accurate test” (Zweig and Campbell, 1993, p. 566). But, it is our choice here. In passing, the distance and angle of this diagonal line depend on the numbers of ties within D+ and D- groups.

The Area Under the ROC Curve

Let x represent the scale of the test result variable, with its low values suggesting a negative result and the high values a positive result. Denote by x_+ the x values for cases with positive actual states. Similarly, denote by x the x values for cases with negative actual states. Then, the “true” area under the ROC curve is

$$\theta = \Pr\left(x_+ > x_-\right).$$

The nonparametric approximation of θ is

$$W = \frac{1}{n_+ n_-} \sum_{\substack{all possible \\ combinations \\ of\left(x_+, x_-\right)}} s\left(x_+, x_-\right),$$

where n_+ is the sample size of D+, n is the sample size of D-, and

$$s\left(x_+, x_-\right) = \begin{cases} 1 & \text{if } x_+ > x_- \\ \frac{1}{2} & \text{if } x_+ = x_-. \\ 0 & \text{if } x_+ < x_- \end{cases}$$

Note that W is the observed area under the ROC curve, which connects successive points by a straight line, i.e., by the trapezoidal rule.

An alternative way to compute W is as follows:

$$W = \frac{1}{n_+ n_-} \sum_{\substack{x \in \{\text{set of all test} \\ \text{result values}\}}} \left\{ n_{-=j} \times n_{+>j} + \frac{n_{-=j} \times n_{+=j}}{2} \right\}.$$

When a low value of x suggests a positive test result and a high value a negative test result

If a low value of x suggests a positive test result and a high a negative test result, compute W as above and then

$$W' = 1 - W,$$

where W' is the estimated area under the curve when a low test result score suggests a positive test result.

SE of the area under the ROC curve statistic, nonparametric assumption

The standard deviation of W is estimated by:

$$\text{SE}\left(W\right) = \sqrt{\frac{W(1-W)+(n_+-1)\left(\hat{Q}_1-W^2\right)+(n_--1)\left(\hat{Q}_2-W^2\right)}{n_+n_-}}.$$

where

$$\hat{Q}_1 = \frac{1}{n_- n_+^2} \sum_x n_{-=j} \times \left[n_{+>j}^2 + n_{+>j} \times n_{+=j} + \frac{n_{+=j}^2}{3} \right]$$

and

$$\hat{Q}_2 = \frac{1}{n_-^2 n} \sum_x n_{+=j} \times \left[n_{-<j}^2 + n_{-<j} \times n_{-=j} + \frac{n_{-=j}^2}{3} \right].$$

When a low value of x suggests a positive test result and a high value a negative test result

If we assume that a low value of x suggests a positive test result and a high value a negative test result, then we estimate the standard deviation of W' by SE $\left(W'\right) = \text{SE}\left(W\right)$.

Under the bi-negative exponential distribution assumption, given the number of negative results equal number of positive results

Under the bi-negative exponential distribution assumption when $n_+ = n$,

$$\hat{Q}_1 = \frac{W}{2-W}$$

and

$$\hat{Q}_2 = \frac{2W^2}{1+W}.$$

SE (W) is then computed as before.

When a low value of x suggests a positive test result and a high value a negative test result

Once again, $\text{SE}(W') = \text{SE}(W)$.

The asymptotic confidence interval of the area under the ROC curve

A 2-sided asymptotic $c\% = (100 - \alpha)\%$ confidence interval for the true area under the ROC curve is

$W \pm Z_\alpha \text{SE}(W)$.

When a low value of x suggests a positive test result and a high value a negative test result

$$W' \pm Z_\alpha \text{SE}\left(W'\right)$$

Asymptotic P-value

Since W is asymptotically normal under the null hypothesis that $\theta = 0.5$, we can calculate the asymptotic *P*-value under the null hypothesis that $\theta = 0.5$ vs. the alternative hypothesis that $\theta \neq 0.5$:

$$\text{Pr}\left(|Z| > \left|\frac{W - 0.5}{\text{SD}(W)|_{\theta=0.5}}\right|\right) = 2\text{Pr}\left(Z > \left|\frac{W - 0.5}{\text{SD}(W)|_{\theta=0.5}}\right|\right)$$

In the nonparametric case,

$$\text{SD}(W)|_{\theta=0.5} = \sqrt{\frac{\theta(1-\theta)+(n_+-1)(Q_1-\theta^2)+(n_--1)(Q_2-\theta^2)}{n_+n_-}}|_{\theta=0.5}$$
$$= \sqrt{\frac{0.5(1-0.5)+(n_+-1)(1/3-0.5^2)+(n_--1)(1/3-0.5^2)}{n_+n_-}}$$
$$\left(= \sqrt{\frac{n_++n_-+1}{12n_+n_-}} = \sqrt{\frac{n_+n_-(n_++n_-+1)}{12}}/n_+n_-\right),$$

because we can deduce that $Q_1 = 1/3$ and $Q_2 = 1/3$ under the null hypothesis that $\theta = 0.5$. The argument for $Q_1 = 1/3$ is as follows. $\theta = 0.5$ implies that the distribution of test results of positive actual state subjects is identical to the distribution of test results of negative actual state subjects. So, the mixture of the two distributions is identical to either one of the distributions. Then, we can reinterpret Q_1 as the probability that, given three randomly chosen subjects from the (mixture) distribution, the subject with the lowest test result was selected, say, first. (One may consider this subject as a negative state subject and the other two as positive state subjects.) From here on, we can pursue a purely combinatorial argument, irrespective of the distribution of subjects' test results, because the drawings are independent and given. There are $3! = 3 \times 2 \times 1 = 6$ ways to order the three subjects, and there are two ways in which the subject with the lowest test result comes first. So, if $\theta = 0.5$, $Q_1 = 2/6 = 1/3$. The argument for $Q_2 = 1/3$ is similar.

In the bi-negative exponential case,

$$\text{SD}\,(W)|_{\theta=0.5} = \sqrt{\frac{\theta(1-\theta)+(n_+-1)(Q_1-\theta^2)+(n_--1)(Q_2-\theta^2)}{n_+n_-}}\Big|_{\theta=0.5}$$
$$= \sqrt{\frac{\theta(1-\theta)+(n_+-1)\left(\frac{\theta}{2-\theta}-\theta^2\right)+(n_--1)\left(\frac{2\theta^2}{1+\theta}-\theta^2\right)}{n_+n_-}}\Big|_{\theta=0.5}$$
$$= \sqrt{\frac{0.5(1-0.5)+(n_+-1)\left(\frac{0.5}{2-0.5}-0.5^2\right)+(n_--1)\left(\frac{2\times0.5^2}{1+0.5}-0.5^2\right)}{n_+n_-}}$$
$$= \sqrt{\frac{0.5(1-0.5)+(n_+-1)(1/3-0.5^2)+(n_--1)(1/3-0.5^2)}{n_+n_-}},$$

where $n_+ = n$. (Note that this formula is identical to the nonparametric one except for the sample size restriction.)

When a low value of x suggests a positive test result and a high value a negative test result

The asymptotic *P*-value under the null hypothesis that $\theta' = 0.5$ vs. the alternative hypothesis that $\theta' \neq 0.5$, if desired, may be computed, using W' and SD $\left(W'\right)|_{\theta=0.5}$ = SD $(W)|_{\theta=0.5}$.

References

Bamber, D. 1975. The area above the ordinal dominance graph and the area below the receiver operating graph. *Journal of Mathematical Psychology*, 12, 387–415.

Beck, R. J., and E. Shultz. 1986. The use of relative operating characteristic (ROC) curves in test performance evaluation. *Arch. Pathol. Lab. Med.*, 110, 13–20.

Centor, R. M., and J. S. Schwartz. 1985. An evaluation of methods for estimating the area under the receiver operating statistic (ROC) curve. *Med. Decis. Making*, 5, 149–156.

Dorfman, D. D., and E. J. Alf. 1968. Maximum likelihood estimation of parameters of signal detection theory—A direct solution. *Psychometrika*, 33, 117–124.

Dorfman, D. D., and E. J. Alf. 1969. Maximum likelihood estimation of parameters of signal-detection theory and determination of confidence intervals—Rating-method data. *Journal of Mathematical Psychology*, 6, 487–496.

Green, D., and J. Swets. 1966. *Signal Detection Theory and Psychophysics*. New York: John Wiley & Sons.

Griner, P. F., R. J. Mayewski, A. I. Mushlin, and P. Greenland. 1981. Selection and interpretation of diagnostic tests and procedures: Principles in applications. *Annals of Internal Medicine*, 94, 553–600.

Hanley, J. A., and B. J. McNeil. 1982. The Meaning and Use of the Area Under a Receiver Operating Characteristic (ROC) Curve. *Radiology*, 143:1, 29–36.

Hanley, J. A., and B. J. McNeil. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148, 839–843.

Metz, C. E. 1978. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 8, 283–298.

Metz, C. E. 1993. *ROC Software Package*. : .

Metz, C. E., and H. B. Kronman. 1980. Statistical significance tests for binormal ROC curves. *Journal of Mathematical Psychology*, 22, 218–243.

Schoonjans, F. 1993–1998. *MedCalc Version 4.20.011*. : .

Woods, K., and K. W. Bowyer. 1997. Generating ROC curves for artificial neural networks. *IEEE Transactions on Medical Imaging*, 16, 329–337.

Zweig, M. H., and G. Campbell. 1993. Receiver Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine. *Clinical Chemistry*, 39:4, 561–577.

SAMPLE Algorithms

SAMPLE permanently draws a random sample of cases for processing in all subsequent procedures.

Selection of a Proportion p

For each case, a random uniform number in the range 0 to 1 is generated. If it is less than *p*, the case is included in the sample.

Selection of a Sample

(a) $p = \frac{n_1}{n}$
$n = n - 1$

Select a case if its uniform (0,1) number is less than *p*. If selected, $n_1 = n_1 - 1$, and return to (a).

Selection of Cases in Nonparametric Procedures

The sampling procedure is as follows:

Each time a case is encountered after the limit imposed by the size of the workspace has been reached, the program decides whether to include it in the sample or not at random. The probability that the new cases will enter the sample is equal to the number of cases that can be held in the workspace divided by the number of cases so far encountered.

If the program decides to accept a case, it then picks at random one of the cases previously stored in the workspace and drops it from the analysis, replacing it with the new case. Each case has the same probability of being in the sample.

If case weighting is used, the nonparametric procedures can use a case more than once. For example, if the weight of a case is 2.3, the program will use that case twice, and may choose at random, with a probability of 0.3, to use it a third time. If sampling is in effect, each of these two or three cases is a candidate for sampling.

SEASON Algorithms

Based on the multiplicative or additive model, the SEASON procedure decomposes the existing series into three components: trend-cycle, seasonal, and irregular.

Model

Multiplicative Model

$$X_t = TC_t S_t I_t, \quad t = 1, \ldots, n$$

Additive Model

$$X_t = TC_t + S_t + I_t, \quad t = 1, \ldots, n$$

where TC_t is the "trend-cycle" component, S_t is the "seasonal" component, and I_t is the "irregular" or "random" component.

The procedure for estimating the seasonal component is:

(1) Smooth the series by the moving average method; the moving average series reflects the trend-cycle component.

(2) Obtain the seasonal-irregular component by dividing the original series by the smoothed values if the model is multiplicative, or by subtracting the smoothed values from the original series if the model is additive.

(3) Isolate the seasonal component from the seasonal-irregular component by computing the medial average (average) of the specific seasonal relatives for each unit of periods if the model is multiplicative (additive).

Moving Average Series

Based on the specified method and period *p*, the moving average series Z_t for X_t is defined as follows:

p is even, weight all points equally

$$Z_t = \begin{cases} \sum_{j=t-\frac{p}{2}}^{t+\frac{p}{2}-1} X_j/p & t = \frac{p}{2}+1, \ldots, n-\frac{p}{2}+1 \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

p is even, weights unequal

$$Z_t = \begin{cases} \left(X_{t-\frac{p}{2}} + X_{t+\frac{p}{2}}\right)/2p + \left(\sum_{j=t-\frac{p}{2}+1}^{t+\frac{p}{2}-1} X_j\right)/p, & t = \frac{p}{2}+1, \ldots, n-\frac{p}{2} \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

p is odd

$$Z_t = \begin{cases} \left(\sum_{j=t-\left[\frac{p}{2}\right]}^{t+\left[\frac{p}{2}\right]} X_j \right) / p, & t = \left[\frac{p}{2}\right] + 1, \ldots, n - \left[\frac{p}{2}\right] \\ \text{SYSMIS} & \text{otherwise} \end{cases}$$

Ratios or Differences (Seasonal-Irregular Component)

Multiplicative Model

$$SI_t = \begin{cases} \text{SYSMIS}, & \text{if } Z_t = \text{SYSMIS} \\ (X_t / Z_t) \times 100, & \text{otherwise} \end{cases}$$

Additive Model

$$SI_t = \begin{cases} \text{SYSMIS}, & \text{if } Z_t = \text{SYSMIS} \\ X_t - Z_t, & \text{otherwise} \end{cases}$$

Seasonal Factors (Seasonal Components)

Multiplicative Model

$$F_t = \begin{cases} \text{medial average}(SI_{t+p}, SI_{t+2p}, \ldots, SI_{t+qp}), & 1 \le t \le L - \left[\frac{L}{p}\right] p \\ \text{medial average}\left(SI_{t+p}, SI_{t+2p}, \ldots, SI_{t+(q-1)p}\right), & L - \left[\frac{L}{p}\right] p < t \le \left[\frac{p}{2}\right] \\ \text{medial average}\left(SI_t, SI_{t+p}, \ldots, SI_{t+(q-1)p}\right), & \left[\frac{p}{2}\right] < t \le p \end{cases}$$

where

$$\begin{array}{lll} L = n - \frac{p}{2} + 1, & q = \left[\frac{L}{p}\right], & \text{if } p \text{ is even and all points are weighted equally} \\ L = n - \left[\frac{p}{2}\right], & q = [(n - p/2)/p], & \text{otherwise} \end{array}$$

and the medial average of a series is the mean value of the series after the smallest and the largest values are excluded. The seasonal factor is defined as

$$SAF_t = F_t \frac{100p}{\sum_{t=1}^{p} F_t}, \quad t = 1, \ldots, p$$

Additive Model

F_t is defined as the arithmetic average of the series shown above. Then

$$SAF_t = F_t - \overline{F},$$

where

$$\overline{F} = \sum_{t=1}^{p} F_t / p$$

Seasonally Adjusted Series (SAS)

$$SAS_t = \begin{cases} (X_t / SAF_m)100, & \text{if model is multiplicative} \\ X_t - SAF_m, & \text{if model is additive} \end{cases}$$

where

$$m = t - [t/p]p$$

Smoothed Trend-Cycle Series

The smoothed trend-cycle series (STC) is obtained by applying a 3×3 moving average on seasonally adjusted series (SAS). Thus,

$$STC_t = \tfrac{1}{9}\left[(SAS)_{t-2} + 2(SAS)_{t-1} + 3(SAS)_t + 2(SAS)_{t+1} + (SAS)_{t+2}\right],$$
$$t = 2, \ldots, n-2$$

and for the two end points on the beginning and end of the series

$$(STC)_2 = \tfrac{1}{3}[(SAS)_1 + (SAS)_2 + (SAS)_3]$$
$$(STC)_{n-1} = \tfrac{1}{3}\left[(SAS)_{n-2} + (SAS)_{n-1} + (SAS)_n\right]$$

$$(STC)_1 = (STC)_2 + \tfrac{1}{2}[(STC)_2 - (STC)_3]$$
$$(STC)_n = (STC)_{n-1} + \tfrac{1}{2}\left[(STC)_{n-1} - (STC)_{n-2}\right]$$

Irregular Component

For $t = 1, \ldots, n$

$$I_t = \begin{cases} (SAS)_t / (STC)_t, & \text{if model is multiplicative} \\ (SAS)_t - (STC)_t, & \text{if model is additive} \end{cases}$$

References

Makridakis, S., S. C. Wheelwright, and V. E. McGee. 1983. *Forecasting: Methods and applications*. New York: John Wiley and Sons.

SELECTPRED

Data mining problems often involve hundreds, or even thousands, of variables. As a result, the majority of time and effort spent in the model-building process involves examining which variables to include in the model. Fitting a computationally intensive model to a set of variables this large may require more time than is practical.

Predictor selection allows the variable set to be reduced in size, creating a more manageable set of attributes for modeling. Adding predictor selection to the analytical process has several benefits:

- Simplifies and narrows the scope of the variables essential to building a predictive model.
- Minimizes the computational time and memory requirements for building a predictive model because focus can be directed to a subset of predictors.
- Leads to more accurate and/or more parsimonious models.
- Reduces the time for generating scores because the predictive model is based upon only a subset of predictors.

Screening

This step removes variables and cases that do not provide useful information for prediction and issues warnings about variables that may not be useful.

The following variables are removed:

- Variables that have all missing values.
- Variables that have all constant values.
- Variables that represent case ID.

The following cases are removed:

- Cases that have missing target values.
- Cases that have missing values in all its predictors.

The following variables are removed based on user settings:

- Variables that have more than m_1% missing values.
- Categorical variables that have a single category counting for more than m_2% cases.
- Continuous variables that have standard deviation < m_3%.
- Continuous variables that have a coefficient of variation $|CV| < m_4$%. CV = standard deviation / mean.
- Categorical variables that have a number of categories greater than m_5% of the cases.

Values m_1, m_2, m_3, m_4, and m_5 are user-controlled parameters.

Ranking Predictors

This step considers one predictor at a time to see how well each predictor alone predicts the target variable. The predictors are ranked according to a user-specified criterion. Available criteria depend on the measurement levels of the target and predictor.

Categorical Target

This section describes ranking of predictors for a categorical target under the following scenarios:

- All predictors categorical
- All predictors continuous
- Some predictors categorical, some continuous

All Categorical Predictors

The following notation applies:

Table 93-1
Notation

Notation	Description
X	The predictor under consideration with I categories.
Y	Target variable with J categories.
N	Total number of cases.
N_{ij}	The number of cases with $X = i$ and $Y = j$.
$N_{i\cdot}$	The number of cases with $X = i$. $N_{i\cdot} = \sum_{j=1}^{J} N_{ij}$
$N_{\cdot j}$	The number of cases with $Y = j$. $N_{\cdot j} = \sum_{i=1}^{I} N_{ij}$

The above notations are based on nonmissing pairs of (X, Y). Hence J, N, and $N{\cdot}j$ may be different for different predictors.

P Value Based on Pearson's Chi-square

Pearson's chi-square is a test of independence between X and Y that involves the difference between the observed and expected frequencies. The expected cell frequencies under the null hypothesis of independence are estimated by $\hat{N}_{ij} = N_{i\cdot}N_{\cdot j}/N$. Under the null hypothesis, Pearson's chi-square converges asymptotically to a chi-square distribution χ^2_d with degrees of freedom $d = (I-1)(J-1)$.

The p value based on Pearson's chi-square X^2 is calculated by p value = Prob($\chi^2_d > X^2$), where

$$X^2 = \sum_{i=1}^{I}\sum_{j=1}^{J}\left(N_{ij} - \hat{N}_{ij}\right)^2/\hat{N}_{ij}.$$

Predictors are ranked by the following rules.

1. Sort the predictors by p value in the ascending order
2. If ties occur, sort by chi-square in descending order.
3. If ties still occur, sort by degree of freedom d in ascending order.

4. If ties still occur, sort by the data file order.

P Value Based on Likelihood Ratio Chi-square

The likelihood ratio chi-square is a test of independence between *X* and *Y* that involves the ratio between the observed and expected frequencies. The expected cell frequencies under the null hypothesis of independence are estimated by $\hat{N}_{ij} = N_{i\cdot}N_{\cdot j}/N$. Under the null hypothesis, the likelihood ratio chi-square converges asymptotically to a chi-square distribution χ^2_d with degrees of freedom *d* = (*I*−1)(*J*−1).

The *p* value based on likelihood ratio chi-square G^2 is calculated by *p* value = Prob($\chi^2_d > G^2$), where

$$G^2 = 2\sum_{i=1}^{I}\sum_{j=1}^{J} G^2_{ij}, \text{ with } G^2_{ij} = \begin{cases} N_{ij}\ln\left(N_{ij}/\hat{N}_{ij}\right) & N_{ij} > 0, \\ 0 & \text{else.} \end{cases}$$

Predictors are ranked according to the same rules as those for the *p* value based on Pearson’s chi-square.

Cramer’s V

Cramer’s *V* is a measure of association, between 0 and 1, based upon Pearson’s chi-square. It is defined as

$$V = \left(\frac{X^2}{N(\min\{I,J\}-1)}\right)^{1/2}.$$

Predictors are ranked by the following rules:

1. Sort predictors by Cramer’s *V* in descending order.
2. If ties occur, sort by chi-square in descending order.
3. If ties still occur, sort by data file order.

Lambda

Lambda is a measure of association that reflects the proportional reduction in error when values of the independent variable are used to predict values of the dependent variable. A value of 1 means that the independent variable perfectly predicts the dependent variable. A value of 0 means that the independent variable is no help in predicting the dependent variable. It is computed as

$$\lambda(Y|X) = \frac{\sum_i \max_j (N_{ij}) - \max_j (N_{\cdot j})}{N - \max_j(N_{\cdot j})}.$$

Predictors are ranked by the following rules:

1. Sort predictors by lambda in descending order.
2. If ties occur, sort by *I* in ascending order.
3. If ties still occur, sort by data file order.

All Continuous Predictors

If all predictors are continuous, *p* values based on the *F* statistic are used. The idea is to perform a one-way ANOVA *F* test for each continuous predictor; this tests if all the different classes of *Y* have the same mean as *X*.

The following notation applies:

Table 93-2
Notation

Notation	**Description**
N_j	The number of cases with *Y* = *j*.
$\overline{x}_j$	The sample mean of predictor *X* for target class *Y* = *j*.
s_j^2	The sample variance of predictor *X* for target class *Y* = *j*. $s_j^2 = \sum_{i=1}^{N_j} (x_{ij} - \overline{x}_j)^2/(N_j - 1)$
$\overline{x}$	The grand mean of predictor *X*. $\overline{x} = \sum_{j=1}^{J} N_j \overline{x}_j / N$

The above notations are based on nonmissing pairs of (*X*, *Y*).

P Value Based on the F Statistic

The *p* value based on the *F* statistic is calculated by *p* value = Prob{*F*(*J*−1, *N*−*J*)> *F*}, where

$$F = \frac{\sum_{j=1}^{J} N_j \left(\overline{x}_j - \overline{\overline{x}}\right)^2 / (J-1)}{\sum_{j=1}^{J} \left(N_j - 1\right) s_j^2 / (N-J)},$$

and *F*(*J*−1, *N*−*J*) is a random variable that follows an *F* distribution with degrees of freedom *J*−1 and *N*−*J*. If the denominator for a predictor is zero, set the *p* value = 0 for the predictor.

Predictors are ranked by the following rules:

1. Sort predictors by *p* value in ascending order.
2. If ties occur, sort by *F* in descending order.
3. If ties still occur, sort by *N* in descending order.
4. If ties still occur, sort by the data file order.

Mixed Type Predictors

If some predictors are continuous and some are categorical, the criterion for continuous predictors is still the *p* value based on the *F* statistic, while the available criteria for categorical predictors are restricted to the *p* value based on Pearson's chi-square or the *p* value based on the likelihood ratio chi-square. These *p* values are comparable and therefore can be used to rank the predictors.

Predictors are ranked by the following rules:

1. Sort predictors by *p* value in ascending order.
2. If ties occur, follow the rules for breaking ties among all categorical and all continuous predictors separately, then sort these two groups (categorical predictor group and continuous predictor group) by the data file order of their first predictors.

Continuous Target

This section describes ranking of predictors for a continuous target under the following scenarios:

- All predictors categorical
- All predictors continuous
- Some predictors categorical, some continuous

All Categorical Predictors

If all predictors are categorical and the target is continuous, *p* values based on the *F* statistic are used. The idea is to perform a one-way ANOVA *F* test for the continuous target using each categorical predictor as a factor; this tests if all different classes of *X* have the same mean as *Y*.

The following notation applies:

Table 93-3
Notation

Notation	**Description**
X	The categorical predictor under consideration with I categories.
Y	The continuous target variable. y_{ij} represents the value of the continuous target for the jth case with $X = i$.
Ni	The number of cases with $X = i$.
$\overline{y}_i$	The sample mean of target Y in predictor category $X = i$.
$s(y)_i^2$	The sample variance of target Y for predictor category $X = i$. $s(y)i2 = \sum_{j=1}^{N_i} (y_{ij} - \overline{y}_i)^2 / (N_i - 1)$
$\overline{\overline{y}}$	The grand mean of target Y. $\overline{\overline{y}} = \Sigma_{i=1}^{I} N_i \overline{y}_i / N$

The above notations are based on nonmissing pairs of (X, Y).

The *p* value based on the *F* statistic is *p* value = Prob$\{F(I-1, N-I) > F\}$, where

$$F = \frac{\sum_{i=1}^{I} N_i \left(\overline{y}_i - \overline{\overline{y}}\right)^2 {}_{/(I-1)}}{\sum_{i=1}^{I} \left(N_i - 1\right) s(y)_i^2 {}_{/(N-I)}},$$

in which $F(I-1, N-I)$ is a random variable that follows a *F* distribution with degrees of freedom $I-1$ and $N-I$. When the denominator of the above formula is zero for a given categorical predictor *X*, set the *p* value = 0 for that predictor.

Predictors are ranked by the following rules:

1. Sort predictors by *p* value in ascending order.
2. If ties occur, sort by *F* in descending order.
3. If ties still occur, sort by *N* in descending order.
4. If ties still occur, sort by the data file order.

All Continuous Predictors

If all predictors are continuous and the target is continuous, the *p* value is based on the asymptotic *t* distribution of a transformation *t* on the Pearson correlation coefficient *r*.

The following notation applies:

Table 93-4
Notation

Notation	**Description**
X	The continuous predictor under consideration.
Y	The continuous target variable.
$\overline{x} = \Sigma_{i=1}^{N} x_i / N$	The sample mean of predictor variable *X*.
$\overline{y} = \Sigma_{i=1}^{N} y_i / N$	The sample mean of target *Y*.
$s(x)^2$	The sample variance of predictor variable *X*.
$s(y)^2$	The sample variance of target variable *Y*.

The above notations are based on nonmissing pairs of (*X*, *Y*).

The Pearson correlation coefficient *r* is

$$r = \frac{\Sigma_{i=1}^{N} (x_i - \overline{x})(y_i - \overline{y}) / (N-1)}{\sqrt{s(x)^2 s(y)^2}}.$$

The transformation *t* on *r* is given by

$$t = r\sqrt{\frac{N-2}{1-r^2}}.$$

Under the null hypothesis that the population Pearson correlation coefficient $\rho = 0$, the *p* value is calculated as

$$p\ value = \begin{cases} 0 & \text{if } r^2 = 1, \\ 2\ \text{Prob}\{T > |t|\} & \text{else.} \end{cases}$$

T is a random variable that follows a *t* distribution with $N-2$ degrees of freedom. The *p* value based on the Pearson correlation coefficient is a test of a linear relationship between *X* and *Y*. If there is some nonlinear relationship between *X* and *Y*, the test may fail to catch it.

Predictors are ranked by the following rules:

1. Sort predictors by *p* value in ascending order.
2. If ties occur in, sort by r^2 in descending order.
3. If ties still occur, sort by *N* in descending order.
4. If ties still occur, sort by the data file order.

Mixed Type Predictors

If some predictors are continuous and some are categorical in the dataset, the criterion for continuous predictors is still based on the *p* value from a transformation and that for categorical predictors from the *F* statistic.

Predictors are ranked by the following rules:

1. Sort predictors by *p* value in ascending order.
2. If ties occur, follow the rules for breaking ties among all categorical and all continuous predictors separately, then sort these two groups (categorical predictor group and continuous predictor group) by the data file order of their first predictors.

Selecting Predictors

If the length of the predictor list has not been prespecified, the following formula provides an automatic approach to determine the length of the list.

Let L_0 be the total number of predictors under study. The length of the list *L* may be determined by

$$L = \left[\min\left(\max\left(30, 2\sqrt{L_0}\right), L_0\right)\right],$$

where [*x*] is the closest integer of *x*. The following table illustrates the length *L* of the list for different values of the total number of predictors L_0.

L_0	*L*	L/L_0(%)
10	10	100.00%
15	15	100.00%
20	20	100.00%
25	25	100.00%

L_0	L	L/L_0(%)
30	30	100.00%
40	30	75.00%
50	30	60.00%
60	30	50.00%
100	30	30.00%
500	45	9.00%
1000	63	6.30%
1500	77	5.13%
2000	89	4.45%
5000	141	2.82%
10,000	200	2.00%
20,000	283	1.42%
50,000	447	0.89%

Simulation algorithms

Simulation in IBM® SPSS® Statistics refers to simulating input data to predictive models using the Monte Carlo method and evaluating the model based on the simulated data. The distribution of predicted target values can then be used to evaluate the likelihood of various outcomes.

The algorithms described here are used by the `SIMPLAN` and `SIMRUN` commands.

Simulation algorithms: create simulation plan

Creating a simulation plan includes specifying distributions for all inputs to a predictive model that are to be simulated. When historical data are present, the distribution that most closely fits the data for each input can be determined using the algorithms described in this section.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 94-1
Notation

Notation	**Description**
x_i	Value of the input variable in the *i*th case of the historical data
w_i	Frequency weight associated with the *i*th case of the historical data
W	Total effective sample size accounting for frequency weights
$\overline{x}_{obs}$	Sample mean
s^2_{obs}	Sample variance
s_{obs}	Sample standard deviation

Distribution fitting

The historical data for a given input is denoted by:

$$\vec{x} = x_1,\ \ x_2,\ \ \ldots,\ \ x_n$$

The total effective sample size is:

$$W = \sum_{i=1}^{n} w_i$$

The observed sample mean, sample variance and sample standard deviation are:

$$\overline{x}_{obs} = \ \frac{1}{W}\ \sum_{i=1}^{n} w_i x_i$$

$$s_{obs}^2 = \frac{1}{W-1} \sum_{i=1}^{n} w_i (x_i - \overline{x}_{obs})^2$$

$$s_{obs} = \sqrt{s_{obs}^2}$$

Parameter estimation for most distributions is based on the maximum likelihood (ML) method, and closed-form solutions for the parameters exist for many of the distributions. There is no closed-form ML solution for the distribution parameters for the following distributions: negative binomial, beta, gamma and Weibull. For these distributions, the Newton-Raphson method is used. This approach requires the following information: the log-likelihood function, the gradient vector, the Hessian matrix, and the initial values for the iterative Newton-Raphson process.

Discrete distributions

Distribution fitting is supported for the following discrete distributions: binomial, categorical, Poisson and negative binomial.

Binomial distribution: parameter estimation

The probability mass function for a random variable x with a binomial distribution is:

$$Bin\,(x; N, P) = \binom{N}{x} P^x (1-P)^{N-x}, for\ x = 0, 1, \ldots, N$$

where $0 \leq P \leq 1$ is the probability of success. The binomial distribution is used to describe the total number of successes in a sequence of N independent Bernoulli trials. The parameter estimates for the binomial distribution using the method of moments (see Johnson & Kotz (2005) for details) are:

$$\hat{P} = \begin{cases} 1 - \frac{s_{obs}^2}{\overline{x}_{obs}}, & \overline{x}_{obs} > s_{obs}^2 \\ NaN, & \overline{x}_{obs} < s_{obs}^2 \end{cases}$$

where *NaN* implies that the binomial distribution would not be an appropriate distribution to fit the data under this criterion, and where

$$\hat{N} = \frac{\overline{x}_{obs}}{\hat{P}}$$

If $\hat{N}$ is not an integer, then the parameter estimates are:

$$\hat{N}^* = \left[\hat{N} + 0.5\right]$$

$$\hat{P}^* = \frac{\overline{x}_{obs}}{\hat{N}^*}$$

where $[x]$ denotes the integer part of x.

Categorical distribution: parameter estimation

The categorical distribution can be considered a special case of the multinomial distribution in which $N = 1$. Suppose x_i, $i = 1, 2, \ldots, n$, has the categorical distribution and its categorical values are denoted as 1, 2, …, J. Then an indicator variable of x_i for category j can be denoted as

$$x_{i,j} = \begin{cases} 1 & \text{if } x_i = j \\ 0 & \text{otherwise} \end{cases}$$

and the corresponding probability is P_j. Then the probability mass function for a random variable x_i with the categorical distribution can be described based on $x_{i,j}$ and P_j as follows:

$$Categorical(x_i; P_1, \ldots, P_J) = \prod_{j=1}^{J} P_j^{x_{i,j}}, \text{ with} \sum_{j=1}^{J} P_j = 1$$

The parameter estimates for $P_j, j = 1, \ldots, J$, are:

$$\hat{P}_j = \frac{\sum_{i=1}^{n} w_i x_{i,j}}{W}, j = 1, \ldots, J$$

Poisson distribution: parameter estimation

The probability mass function for a random variable x with a Poisson distribution is:

$$Pois(x; \lambda) = \frac{e^{-\lambda}\lambda^x}{x!}, \ for\ x = 0, 1, \cdots$$

where $\lambda > 0$ is the rate parameter of the Poisson distribution. The parameter of the Poisson distribution can be estimated as:

$$\hat{\lambda} = \overline{x}_{obs}$$

Negative binomial distribution: parameter estimation

The distribution fitting component for simulation supports the parameterization of the negative binomial distribution that describes the distribution of the number of failures before the r^{th} success. For this parameterization, the probability mass function for a random variable x is:

$$NB(x; r, \theta) = \binom{x + r - 1}{x} \theta^r (1 - \theta)^x, \text{ for } x = 0, 1, \ldots$$

where $r \geq 0,\ 0 \leq \theta \leq 1$ are the two distribution parameters. There is no closed-form solution for the parameters *r* and *θ*, so the Newton-Raphson method with step-halving will be used. The method requires the following information:

(1) The log likelihood function

$$L = \sum_{i=1}^{n} w_i \ln\Gamma(x_i + r) - \sum_{i=1}^{n} w_i \ln x_i! - W\ln\Gamma(r) + Wr\ln(\theta) + \ln(1-\theta)\sum_{i=1}^{n} w_i x_i$$

(2) The gradient (1st derivative) vector with respect to *r* and *θ*

$$s = \begin{bmatrix} \frac{\partial L}{\partial \theta} \\ \frac{\partial L}{\partial r} \end{bmatrix} = \begin{bmatrix} \frac{Wr}{\theta} - \frac{1}{(1-\theta)}\sum_{i=1}^{n} w_i x_i \\ \sum_{i=1}^{n} w_i \psi(x_i + r) - W\psi(r) + W\ln(\theta) \end{bmatrix}$$

where $\psi(\alpha) = \frac{\Gamma'(\alpha)}{\Gamma(\alpha)}$ is a digamma function, which is the derivative of the logarithm of the gamma function, evaluated at α.

(3) The Hessian (2nd derivative) matrix with respect to *r* and *θ* (since the Hessian matrix is symmetric, only the lower triangular portion is displayed)

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial \theta^2} & \\ \frac{\partial^2 L}{\partial r \partial \theta} & \frac{\partial^2 L}{\partial r^2} \end{bmatrix} = \begin{bmatrix} -\frac{Wr}{\theta^2} - \frac{1}{(1-\theta)^2}\sum_{i=1}^{n} w_i x_i & \\ \frac{W}{\theta} & \sum_{i=1}^{n} w_i \psi'(x_i + r) - W\psi'(r) \end{bmatrix}$$

where $\psi'(\alpha)$ is the trigamma function, or the derivative of the digamma function.

(4) The initial values of *θ* and *r* can be obtained from the closed-form estimates using the method of moments:

$$r^{(0)} = \begin{cases} \frac{\overline{x}_{obs}^2}{s_{obs}^2 - \overline{x}_{obs}} & \text{if } s_{obs}^2 > \overline{x}_{obs} \\ 1 & \text{otherwise} \end{cases}$$

$$\theta^{(0)} = \frac{r^{(0)}}{r^{(0)} + \overline{x}_{obs}}$$

Note

An alternative parameterization of the negative binomial distribution describes the distribution of the number of trials before the r^{th} success. Although it is not supported in distribution fitting, it is supported in simulation when explicitly specified by the user. The probability mass function for this parameterization, for a random variable x is:

$$NB(x; r, \theta) = \binom{x-1}{r-1}\theta^r(1-\theta)^{x-r}, \text{ for } x \geq r$$

where $r \geq 0,\ 0 \leq \theta \leq 1$ are the two distribution parameters.

Continuous distributions

Distribution fitting is supported for the following continuous distributions: triangular, uniform, normal, lognormal, exponential, beta, gamma and Weibull.

Triangular distribution: parameter estimation

The probability density function for a random variable x with a triangular distribution is:

$$Triag(x;a,m,b) = \begin{cases} \frac{2}{(b-a)}\frac{(x-a)}{(m-a)}, & x \in [a,m) \\ \frac{2}{(b-a)}, & x = m \\ \frac{2}{(b-a)}\frac{(b-x)}{(b-m)}, & x \in (m,b] \\ 0, & x \notin [a,b] \end{cases}$$

such that $a \leq m \leq b$. Parameter estimates of the triangular distribution are:

$$\hat{a} = \min\{x_1, x_2, \ldots, x_n\}$$

$$\hat{b} = \max\{x_1, x_2, \ldots, x_n\}$$

$$\hat{m} = mode\{x_1, x_2, \ldots, x_n\}$$

Since the calculation of the mode for continuous data may be ambiguous, we transform the parameter estimates and use the method of moments as follows (see Kotz and Rene van Dorp (2004) for details):

$$z_i = \frac{x_i - \hat{a}}{\hat{b} - \hat{a}}$$

$$\theta = \frac{m - \hat{a}}{\hat{b} - \hat{a}}$$

$$\overline{z} = \frac{1}{W}\sum_{i=1}^{n} w_i z_i$$

From the method of moments we obtain

$$\hat{\theta} = 3\overline{z} - 1$$

from which it follows that

$$\hat{m} = \hat{a} + \left(\hat{b} - \hat{a}\right) \times (3\overline{z} - 1)$$

Note: For very skewed data or if the actual mode equals a or b, the estimated mode, $\hat{m}$, may be less than â or greater than b̂. In this case, the adjusted mode, defined as below, is used:

$$Adj.\,\hat{m} = \begin{cases} \hat{a} & \text{if } \hat{m} < \hat{a} \\ \hat{b} & \text{if } \hat{m} > \hat{b} \end{cases}$$

Uniform distribution: parameter estimation

The probability density function for a random variable x with a uniform distribution is:

$$U\left(x; a, b\right) = \begin{cases} \frac{1}{b-a}, & x \in [a, b] \\ 0, & x \notin [a, b] \end{cases}$$

where a is the minimum and b is the maximum among the values of $\overrightarrow{x}$. Hence, the parameter estimates of the uniform distribution are:

$$\hat{a} = \min\left\{x_1, x_2, \ldots, x_n\right\}$$

$$\hat{b} = \max\left\{x_1, x_2, \ldots, x_n\right\}$$

Normal distribution: parameter estimation

The probability density function for a random variable x with a normal distribution is:

$$Nor(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \ -\infty < x < \infty$$

Here, μ is the measure of centrality and σ is the measure of dispersion of the normal distribution. The parameter estimates of the normal distribution are:

$$\hat{\mu} = \overline{x}_{obs}$$

$$\hat{\sigma} = \sqrt{\frac{1}{W} \sum_{i=1}^{n} w_i (x_i - \overline{x}_{obs})^2} = \sqrt{\frac{(W-1)\, s_{obs}^2}{W}}$$

Lognormal distribution: parameter estimation

The lognormal distribution is a probability distribution where the natural logarithm of a random variable follows a normal distribution. In other words, if x has a lognormal(μ, σ) distribution, then ln(x) has a normal(ln(μ), σ) distribution. The probability density function for a random variable x with a lognormal distribution is:

$$LN(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma x} \ e^{-\frac{1}{2}\left(\frac{\ln x - \ln \mu}{\sigma}\right)^2}, 0 < x < \infty$$

Note: The form of the probability density function is the same as that used in IBM® SPSS® Statistics.

Define $\overline{lx}_{obs} = \frac{1}{W}\sum_{i=1}^{n} w_i \ln x_i$

Parameter estimates for the lognormal distribution are:

$$\hat{\mu} = e^{\overline{lx}_{obs}}$$

$$\hat{\sigma} = \sqrt{\frac{1}{W} \sum_{i=1}^{n} w_i(\ln x_i - \ln \hat{\mu})^2}$$

Exponential distribution: parameter estimation

The probability density function for a random variable x with an exponential distribution is:

$$Exp(x;\lambda) = \lambda e^{-\lambda x},\ \text{for } x \geq 0 \text{ and } \lambda > 0$$

The estimate of the parameter for the exponential distribution is:

$$\hat{\lambda} = \frac{1}{\bar{x}_{obs}}$$

Beta distribution: parameter estimation

The probability density function for a random variable x with a beta distribution is:

$$Beta(x;\alpha,\beta) = \frac{1}{\mathrm{B}(\alpha,\beta)}\ x^{\alpha-1}(1-x)^{\beta-1},\ \ \alpha,\beta > 0$$

where,

$$B(\alpha,\beta) = \frac{\Gamma(\alpha)\Gamma(\beta)}{\Gamma(\alpha+\beta)}$$

There is no closed-form solution for the parameters α and β, so the Newton-Raphson method with step-halving will be used. The method requires the following information:

(1) The log likelihood function

$$L = W\ln(\Gamma(\alpha+\beta)) - W\ln(\Gamma(\alpha)) - W\ln(\Gamma(\beta))$$

$$+(\alpha-1)\sum_{i=1}^{n} w_i \ln x_i + (\beta-1)\sum_{i=1}^{n} w_i \ln(1-x_i)$$

(2) The gradient (1st derivative) vector with respect to α and β

$$s = \begin{bmatrix} \frac{\partial L}{\partial \alpha} \\ \frac{\partial L}{\partial \beta} \end{bmatrix} = \begin{bmatrix} W\psi(\alpha+\beta) - W\psi(\alpha) + \sum_{i=1}^{n} w_i \ln(x_i) \\ W\psi(\alpha+\beta) - W\psi(\beta) + \sum_{i=1}^{n} w_i \ln(1-x_i) \end{bmatrix}$$

where $\psi(\alpha) = \frac{\Gamma'(\alpha)}{\Gamma(\alpha)}$ is a digamma function, which is the derivative of the logarithm of the gamma function, evaluated at α.

(3) The Hessian (2nd derivative) matrix with respect to α and β (since the Hessian matrix is symmetric, only the lower triangular portion is displayed)

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial \alpha^2} & \\ \frac{\partial^2 L}{\partial \alpha \partial \beta} & \frac{\partial^2 L}{\partial \beta^2} \end{bmatrix} = \begin{bmatrix} W(\psi'(\alpha+\beta) - \psi'(\alpha)) & \\ W\psi'(\alpha+\beta) & W(\psi'(\alpha+\beta) - \psi'(\beta)) \end{bmatrix}$$

where $\psi'(\alpha)$ is the trigamma function, or the derivative of the digamma function.

(4) The initial values of α and β can be obtained from the closed-form estimates using the method of moments:

$$\alpha^{(0)} = \overline{x}_{obs}\left(\frac{\overline{x}_{obs}(1-\overline{x}_{obs})}{s_{obs}{}^2} - 1\right)$$

$$\beta^{(0)} = (1-\overline{x}_{obs})\left(\frac{\overline{x}_{obs}(1-\overline{x}_{obs})}{s_{obs}{}^2} - 1\right)$$

Gamma distribution: parameter estimation

The probability density function for a random variable x with a gamma distribution is:

$$Gamma(x;\alpha,\beta) = \frac{\beta^\alpha}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \text{ for } x \geq 0 \text{ and } \alpha, \beta > 0$$

If α is a positive integer, then the gamma function is given by: $\Gamma(\alpha) = (\alpha-1)!$

There is no closed-form solution for the parameters α and β, so the Newton-Raphson method with step-halving will be used. The method requires the following information:

(1) The log likelihood function

$$L = W\alpha \ln\beta - W \ln\Gamma(\alpha) + (\alpha-1)\sum_{i=1}^{n} w_i \ln x_i - \beta \sum_{i=1}^{n} w_i x_i$$

(2) The gradient (1st derivative) vector with respect to α and β

$$s = \begin{bmatrix} \frac{\partial L}{\partial \alpha} \\ \frac{\partial L}{\partial \beta} \end{bmatrix} = \begin{bmatrix} W \ln\beta - W\psi(\alpha) + \sum_{i=1}^{n} w_i \ln x_i \\ \frac{W\alpha}{\beta} - \sum_{i=1}^{n} w_i x_i \end{bmatrix}$$

where $\psi(\alpha) = \frac{\Gamma'(\alpha)}{\Gamma(\alpha)}$ is a digamma function, which is the derivative of the logarithm of the gamma function, evaluated at α.

(3) The Hessian (2nd derivative) matrix with respect to α and β (since the Hessian matrix is symmetric, only the lower triangular portion is displayed)

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial \alpha^2} & \\ \frac{\partial^2 L}{\partial \alpha \partial \beta} & \frac{\partial^2 L}{\partial \beta^2} \end{bmatrix} = \begin{bmatrix} -W\psi'(\alpha) & \\ \frac{W}{\beta} & -\frac{W\alpha}{\beta^2} \end{bmatrix}$$

where $\psi'(\alpha)$ is the trigamma function, or the derivative of the digamma function.

(4) The initial values of α and β can be obtained from the closed-form estimates using the method of moments:

$$\alpha^{(0)} = \left(\frac{\overline{x}_{obs}}{s_{obs}}\right)^2$$

$$\beta^{(0)} = \frac{\overline{x}_{obs}}{s_{obs}^2}$$

Weibull distribution: parameter estimation

Distribution fitting for the Weibull distribution is restricted to the two-parameter Weibull distribution, whose probability density function is given by:

$$Weib(x;\beta,\gamma) = \frac{\gamma}{\beta}\left(\frac{x}{\beta}\right)^{\gamma-1} e^{-\left(\frac{x}{\beta}\right)^{\gamma}}, \quad \text{for } x \geq 0 \text{ and } \beta,\gamma > 0$$

There is no closed-form solution for the parameters β and γ, so the Newton-Raphson method with step-halving will be used. The method requires the following information:

(1) The log likelihood function

$$L = W(\ln\gamma - \gamma\ln\beta) + (\gamma - 1)\sum_{i=1}^{n} w_i \ln(x_i) - \sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma}$$

(2) The gradient (1st derivative) vector with respect to β and γ

$$s = \begin{bmatrix} \frac{\partial L}{\partial \beta} \\ \frac{\partial L}{\partial \gamma} \end{bmatrix} = \begin{bmatrix} -\frac{W\gamma}{\beta} + \frac{\gamma}{\beta}\sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma} \\ \frac{W}{\gamma} - W\ln\beta + \sum_{i=1}^{n} w_i \ln(x_i) - \sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma} \ln\left(\frac{x_i}{\beta}\right) \end{bmatrix}$$

(3) The Hessian (2nd derivative) matrix with respect to β and γ (since the Hessian matrix is symmetric, only the lower triangular portion is displayed)

$$H = \begin{bmatrix} \frac{\partial^2 L}{\partial \beta^2} & \\ \frac{\partial^2 L}{\partial \beta \partial \gamma} & \frac{\partial^2 L}{\partial \gamma^2} \end{bmatrix}$$

where

$$\frac{\partial^2 L}{\partial \beta^2} = \frac{\gamma}{\beta^2}\left[W - (\gamma+1)\sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma}\right]$$

$$\frac{\partial^2 L}{\partial \beta \partial \gamma} = -\frac{1}{\beta}\left[W - \sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma} - \gamma \sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma} \ln\left(\frac{x_i}{\beta}\right)\right]$$

$$\frac{\partial^2 L}{\partial \gamma^2} = -\frac{W}{\gamma^2} - \sum_{i=1}^{n} w_i \left(\frac{x_i}{\beta}\right)^{\gamma}\left[\ln\left(\frac{x_i}{\beta}\right)\right]^2$$

(4) The initial values of β and γ are given by:

$$\gamma^{(0)} = 1$$

$$\beta^{(0)} = 1$$

Goodness of fit measures

Goodness of fit measures are used to determine the distribution that most closely fits the data. For discrete distributions, the Chi-Square test is used. For continuous distributions, the Anderson-Darling test or the Kolmogorov-Smirnov test is used.

Discrete distributions

The Chi-Square goodness of fit test is used for discrete distributions (Dirk P. Kroese, 2011). The Chi-Square test statistic has the following form:

$$T = \sum_{i=1}^{k} \frac{(O_i - E_i)^2}{E_i}$$

where,

Table 94-2
Notation

Notation	**Description**
k	The number of classes, as defined in the table below for each discrete distribution
O_i	The total observed frequency for class *i*

Notation	**Description**
PDF(i)	Probability density function of the fitted distribution. For the Poisson and negative binomial distributions, the density function for the last class is computed as $\text{PDF}(k) = 1 - \sum_{i=1}^{k-1} \text{PDF}(i)$
E_i	Expected frequency for class *i*: E_i = *W*PDF(i)*
W	The total effective sample size

For large *W*, the above statistic follows the Chi-Square distribution:

$$T \sim \chi^2_{k-1-r}$$

where *r = number of parameters estimated from the data*. The following table provides the values of *k* and *r* for the various distributions. The value *Max* in the table is the observed maximum value.

Distribution	**Notation**	***k* (classes)**	***r* (parameters)**
Binomial	$Bin(x; N, P)$	*N+1*	2
Categorical	$Categorical(x; p_1, \ldots, p_J)$	*J*	*J*-1
Poisson	$Pois(x;\lambda)$	*Max* + 1	1
Negative binomial	$NB(x; r, \theta)$	*Max* + 1	2

This Chi-Square test is valid only if all values of $E_i \geq 5$.

The p-value for the Chi-Square test is then calculated as:

$$p = 1 - F\left(T, \chi^2_{k-1-r}\right)$$

where $F\left(T, \chi^2_{k-1-r}\right) =$ *CDF* of the Chi-Square distribution.

Note: The p-value cannot be calculated for the Categorical distribution since the number of degrees of freedom is zero.

Continuous distributions

For continuous distributions, the Anderson-Darling test or the Kolmogorov-Smirnov test is used to determine goodness of fit. The calculation consists of the following steps:

1. Transform the data to a Uniform(0,1) distribution
2. Sort the transformed data to generate the Order Statistics
3. Calculate the Anderson-Darling or Kolmogorov-Smirnov test statistic
4. Compute the approximate p-value associated with the test statistic

The first two steps are common to both the Anderson-Darling and Kolmogorov-Smirnov tests. The original data are transformed to a Uniform(0,1) distribution using the transformation:

$$y = F\left(x\right)$$

where the transformation function $F\left(x\right)$ is given in the table below for each of the supported distributions.

Distribution	**Transformation F(x)**
$\boldsymbol{Triag(x;a,m,b)}$	$\begin{cases} \frac{1}{(b-a)}\frac{(x-a)^2}{(m-a)}, & x \in [a,m) \\ \frac{(m-a)}{(b-a)}, & x = m \\ 1-\frac{1}{(b-a)}\frac{(b-x)^2}{(b-m)}, & x \in (m,b] \\ 0, & x \quad [a,b] \end{cases}$
$U\left(x;a,b\right)$	$\frac{x-a}{b-a}$
$Nor(x,\mu,\sigma)$	$\Phi\left(\frac{x-\mu}{\sigma}\right)$
$LN(x,\mu,\sigma)$	$\Phi\left(\frac{\ln x-\ln\mu}{\sigma}\right)$
$Exp(x;\lambda)$	$1-e^{-\lambda x}$
$Beta(x;\alpha,\beta)$	$\int_0^x \frac{1}{\mathrm{B}(\alpha,\beta)} t^{\alpha-1}(1-t)^{\beta-1}dt$
$Gamma(x;\alpha,\beta)$	$\int_0^x \frac{\beta^\alpha}{\Gamma(\alpha)} t^{\alpha-1}e^{-\beta t}dt$
$Weib(x;\beta,\gamma)$	$1-e^{-\left(\frac{x}{\beta}\right)^\gamma}$

The transformed data points y_i are sorted in ascending order to generate the Order Statistics:

$$y_{(1)} \leq y_{(2)} \leq \ldots \leq y_{(n-1)} \leq y_{(n)}$$

Define w_i^* to be the corresponding frequency weight for $y_{(i)}$. The cumulative frequency up to and including $y_{(i)}$ is defined as:

$$W_i^* = \sum_{k=1}^{i} w_i^*$$

and where we define $W_0^* = 0$.

Anderson-Darling test

The Anderson-Darling test statistic is given by:

$$z = -W_n^* - \frac{1}{W_n^*}\sum_{i=1}^{n} w_i^*\left(2W_{i-1}^* + w_i^*\right)\ln\left(y_{(i)}\right) + \frac{1}{W_n^*}\sum_{i=1}^{n} w_i^*\left(2W_{i-1}^* + w_i^*\right)\ln\left(1 - y_{(i)}\right)$$
$$-2\sum_{i=1}^{n} w_i^* \ln\left(1 - y_{(i)}\right)$$

For more information, see the topic “Anderson-Darling statistic with frequency weights”.

The approximate p-value for the Anderson-Darling statistic can be computed for the following distributions: uniform, normal, lognormal, exponential, Weibull and gamma. The p-value is not available for the triangular and beta distributions.

Uniform distribution: p-value

The p-value for the Anderson-Darling statistic is computed based on the following result, provided by Marsaglia (2004):

$$p = \begin{cases} 1 - z^{-1/2} e^{-1.2337141/z} g\left(z\right) & for\ z \in (0,2) \\ 1 - e^{-e^{(1.0776-(2.30695-(0.43424-(0.082433-(0.008056-0.0003146z)z)z)z)z)}} & for\ z \in [2,\infty) \end{cases}$$

where

$$g\left(z\right) = (2.00012 + (0.247105 - (0.0649821 - (0.0347962 - (0.0116720 - 0.00168691z)z)z)z)z)$$

Normal and lognormal distributions: p-value

The p-value for the Anderson-Darling statistic is computed based on the following result, provided by D’Agostino and Stephens (1986):

$$p = \begin{cases} 1 - \exp\left(-13.436 + 101.14z^* - 223.73z^{*2}\right), & z^* \le 0.2 \\ 1 - \exp\left(-8.318 + 42.796z^* - 59.938z^{*2}\right), & 0.2 < z^* \le 0.34 \\ \exp\left(0.9177 - 4.279z^* - 1.38z^{*2}\right), & 0.34 < z^* \le 0.6 \\ \exp\left(1.2937 - 5.709z^* + 0.0186z^{*2}\right), & 0.6 < z^* \le 153.467 \\ 0 & z^* > 153.467 \end{cases}$$

where

$$z^* = z\left(1.0 + 0.75/W_n^* + 2.25/W_n^{*2}\right)$$

Exponential distribution: p-value

The p-value for the Anderson-Darling statistic is computed based on the following result, provided by D'Agostino and Stephens (1986):

$$p = \begin{cases} 1 - \exp\left(-12.2204 + 67.459z^* - 110.3z^{*2}\right), & z^* \le 0.260 \\ 1 - \exp\left(-6.1327 + 20.218z^* - 18.663z^{*2}\right), & 0.260 < z^* \le 0.510 \\ \exp\left(0.9209 - 3.353z^* + 0.3z^{*2}\right), & 0.510 < z^* \le 0.950 \\ \exp\left(0.731 - 3.009z^* + 0.15z^{*2}\right), & 0.950 < z^* \le 10.03 \\ 0 & z^* > 10.03 \end{cases}$$

where

$$z^* = z\left(1.0 + 0.6/W_n^*\right)$$

Weibull distribution: p-value

The p-value for the Anderson-Darling statistic is computed based on Table 94-3 below, provided by D'Agostino and Stephens (1986). First, the adjusted Anderson-Darling statistic is computed from:

$$z^* = z\left(1 + 0.2/\sqrt{W_n^*}\right)$$

If the value of z^* is between two probability levels (in the table), then linear interpolation is used to estimate the p-value. For example, if $z^* = 0.543$ which is between $z_1^* = 0.474$ and $z_2^* = 0.637$ then the corresponding probabilities of z_1^* and z_2^* are $p_1 = 0.25$ and $p_2 = 0.1$ respectively. Then the p-value of z^* is computed as

$$p = \frac{p_2 - p_1}{z_2^* - z_1^*}(z^* - z_1^*) + p_1 = \frac{0.1 - 0.25}{0.637 - 0.474}(0.543 - 0.474) + 0.25 = 0.1865$$

If the value of z^* is less than the smallest critical value in the table, then the p-value is $\ge$ 0.25; and if z^* is greater than the largest critical value in the table, then the p-value is $\le$ 0.01.

Table 94-3
Upper tail probability and corresponding critical values for the Anderson-Darling test, for the Weibull distribution

p-value	**0.25**	**0.10**	**0.05**	**0.025**	**0.01**
$z\left(1 + 0.2/\sqrt{W_n^*}\right)$	0.474	0.637	0.757	0.877	1.038

Gamma distribution: p-value

Table 94-4, which is provided by D'Agostino and Stephens (1986), is used to compute the p-value of the Anderson-Darling test for the gamma distribution. First, the appropriate row in the table is determined from the range of the parameter α. Then linear interpolation is used to compute the p-value, as done for the Weibull distribution. For more information, see the topic "Weibull distribution: p-value".

If the test statistic is less than the smallest critical value in the row, then the p-value is $\geq$ 0.25; and if the test statistic is greater than the largest critical value in the row, then the p-value is $\leq$ 0.005.

Table 94-4
Upper tail probability and corresponding critical values for the Anderson-Darling test, for the gamma distribution with estimated parameter α

p-value	**0.25**	**0.10**	**0.05**	**0.025**	**0.01**	**0.005**
$\alpha \in (0, 1]$	0.486	0.657	0.786	0.917	1.092	1.227
$\alpha \in (1, 8]$	0.473	0.637	0.759	0.883	1.048	1.173
$\alpha \in (8, \infty)$	0.470	0.631	0.752	0.873	1.035	1.159

Kolmogorov-Smirnov test

The Kolmogorov-Smirnov test statistic, D_n, is given by:

$$D^+ = max_i \left| y_{(i)} - \frac{W_i^*}{W_n^*} \right| \qquad D^- = max_i \left| y_{(i)} - \frac{W_i^* - 1}{W_n^*} \right|$$

$$D_n = max\left(D^+, D^-\right)$$

Computation of the p-value is based on the modified Kolmogorov-Smirnov statistic, which is distribution specific.

Uniform distribution: p-value

The procedure proposed by Kroese (2011) is used to compute the p-value of the Kolmogorov-Smirnov statistic for the uniform distribution. First, the modified Kolmogorov-Smirnov statistic is computed as

$$D = \sqrt{W_n^*}\ D_n$$

The corresponding p-value is computed as follows:

1. Set k=100
2. Define $\alpha_k = \sum_{i=-k}^{k} (-1)^i e^{-2(iD)^2}$
3. Calculate α_k and α_{k+1}
4. If $|\alpha_k - \alpha_{k+1}| \geq 10^{-5}$ set k=k+1 and repeat step 2; otherwise, go to step 5.
5. p-value $= 1 - \alpha_k$

Normal and lognormal distributions: p-value

The modified Kolmogorov-Smirnov statistic is

$$D = D_n \left(\sqrt{W_n^*} - 0.01 + \frac{0.85}{\sqrt{W_n^*}} \right)$$

The p-value for the Kolmogorov-Smirnov statistic is computed based on Table 94-5 below, provided by D'Agostino and Stephens (1986). If the value of D is between two probability levels, then linear interpolation is used to estimate the p-value. For more information, see the topic "Weibull distribution: p-value".

If D is less than the smallest critical value in the table, then the p-value is $\geq$ 0.15; and if D is greater than the largest critical value in the table, then the p-value is $\leq$ 0.01.

Table 94-5
Upper tail probability and corresponding critical values for the Kolmogorov-Smirnov test, for the Normal and Lognormal distributions

***p*-value**	**0.15**	**0.10**	**0.05**	**0.025**	**0.01**
D	0.775	0.819	0.895	0.995	1.035

Exponential distribution: p-value

The modified Kolmogorov-Smirnov statistic is

$$D = (D_n - 0.2/W_n^*) \left(\sqrt{W_n^*} + 0.26 + 0.5/\sqrt{W_n^*} \right)$$

The p-value for the Kolmogorov-Smirnov statistic is computed based on Table 94-6 below, provided by D'Agostino and Stephens (1986). If the value of D is between two probability levels, then linear interpolation is used to estimate the p-value. For more information, see the topic "Weibull distribution: p-value".

If D is less than the smallest critical value in the table, then the p-value is $\geq$ 0.15; and if D is greater than the largest critical value in the table, then the p-value is $\leq$ 0.01.

Table 94-6
Upper tail probability and corresponding critical values for the Kolmogorov-Smirnov test, for the Exponential distribution

***p*-value**	**0.15**	**0.10**	**0.05**	**0.025**	**0.01**
D	0.926	0.995	1.094	1.184	1.298

Weibull distribution: p-value

The modified Kolmogorov-Smirnov statistic is

$$D = \sqrt{W_n^*} D_n$$

The p-value for the Kolmogorov-Smirnov statistic is computed based on Table 94-7 below, provided by D'Agostino and Stephens (1986). If the value of D is between two probability levels, then linear interpolation is used to estimate the p-value. For more information, see the topic "Weibull distribution: p-value".

If D is less than the smallest critical value in the table, then the p-value is $\geq$ 0.10; and if D is greater than the largest critical value in the table, then the p-value is $\leq$ 0.01.

Table 94-7
Upper tail probability and corresponding critical values for the Kolmogorov-Smirnov test, for the Weibull distribution

***p*-value**	**0.10**	**0.05**	**0.025**	**0.01**
D	1.372	1.477	1.557	1.671

Gamma distribution: p-value

The modified Kolmogorov-Smirnov statistic is

$$D = D_n \quad \left(\sqrt{W_n^*} + 0.3/\sqrt{W_n^*}\right)$$

The p-value for the Kolmogorov-Smirnov statistic is computed based on Table 94-8 below, provided by D'Agostino and Stephens (1986). If the value of D is between two probability levels, then linear interpolation is used to estimate the p-value. For more information, see the topic "Weibull distribution: p-value".

If D is less than the smallest critical value in the table, then the p-value is $\geq$ 0.25; and if D is greater than the largest critical value in the table, then the p-value is $\leq$ 0.005.

Table 94-8
Upper tail probability and corresponding critical values for the Kolmogorov-Smirnov test, for the Gamma distribution

***p*-value**	**0.25**	**0.20**	**0.15**	**0.10**	**0.05**	**0.025**	**0.01**	**0.005**
D	0.74	0.780	0.800	0.858	0.928	0.990	1.069	1.13

Determining the recommended distribution

The distribution fitting module is invoked by the user, who may specify an explicit set of distributions to test or rely on the default set, which is determined from the measurement level of the input to be fit. For continuous inputs, the user specifies either the Anderson-Darling test (the default) or the Kolmogorov-Smirnov test for the goodness of fit measure (for ordinal and nominal inputs, the Chi-Square test is always used). The distribution fitting module then returns the values of the specified test statistic along with the calculated p-values (if available) for each of the tested distributions, which are then presented to the user in ascending order of the test statistic. The recommended distribution is the one with the minimum value of the test statistic.

The above approach yields the distribution that most closely fits the data. However, if the p-value of the recommended distribution is less than 0.05, then the recommended distribution may not provide a close fit to the data.

Anderson-Darling statistic with frequency weights

To obtain the expression for the Anderson-Darling statistic with frequency weights, we first give the expression where the frequency weight of each value is 1:

$$\begin{aligned} z &= -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\ln\left(y_{(i)}\left(1-y_{(n+1-i)}\right)\right)\right] \\ &= -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\ln\left(y_{(i)}\right)+\ln\left(1-y_{(n+1-i)}\right)\right] \\ &= -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\ln\left(y_{(i)}\right)\right] - \frac{1}{n}\sum_{i=1}^{n}\left(2\left(n+1-i\right)-1\right)\left[\ln\left(1-y_{(i)}\right)\right] \\ &= -n - \frac{1}{n}\sum_{i=1}^{n}(2i-1)\left[\ln\left(y_{(i)}\right)\right] - \frac{1}{n}\sum_{i=1}^{n}(1-2i)\left[\ln\left(1-y_{(i)}\right)\right] - \sum_{i=1}^{n}2\left[\ln\left(1-y_{(i)}\right)\right] \\ &= A+B+C+D \end{aligned}$$

If there is a frequency weight variable, then the corresponding four terms of the above expression are given by:

$$A = -W_n^*$$

$$\begin{aligned} B &= -\frac{1}{W_n^*}\sum_{i=1}^{n}\sum_{j=1}^{w_i^*}\left(2\left(W_{i-1}^*+j\right)-1\right)\left[\ln\left(y_{(i)}\right)\right] \\ &= -\frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*\left(2W_{i-1}^*-1\right)\ln\left(y_{(i)}\right) - \frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*(w_i^*+1)\ln\left(y_{(i)}\right) \\ &= -\frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*\left(2W_{i-1}^*+w_i^*\right)\ln\left(y_{(i)}\right) \end{aligned}$$

$$\begin{aligned} C &= -\frac{1}{W_n^*}\sum_{i=1}^{n}\sum_{j=1}^{w_i^*}\left(1-2\left(W_{i-1}^*+j\right)\right)\left[\ln\left(1-y_{(i)}\right)\right] \\ &= -\frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*\left(1-2W_{i-1}^*\right)\ln\left(1-y_{(i)}\right) + \frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*(w_i^*+1)\ln\left(1-y_{(i)}\right) \\ &= \frac{1}{W_n^*}\sum_{i=1}^{n}w_i^*\left(2W_{i-1}^*+w_i^*\right)\ln\left(1-y_{(i)}\right) \end{aligned}$$

$$D = -2\sum_{i=1}^{n}w_i^*\ln\left(1-y_{(i)}\right)$$

where w_i^* and W_i^* are defined in the section on goodness of fit measures for continuous distributions. For more information, see the topic “Continuous distributions”.

References

D’Agostino, R., and M. Stephens. 1986. *Goodness-of-Fit Techniques*. New York: Marcel Dekker.

Johnson, N. L., S. Kotz, and A. W. Kemp. 2005. *Univariate Discrete Distributions*, 3rd ed. Hoboken, New Jersey: John Wiley & Sons.

Kotz, S., and J. Rene Van Dorp. 2004. *Beyond Beta, Other Continuous Families of Distributions with Bounded Support and Applications*. Singapore: World Scientific Press.

Kroese, D. P., T. Taimre, and Z. I. Botev. 2011. *Handbook of Monte Carlo Methods*. Hoboken, New Jersey: John Wiley & Sons.

Marsaglia, G., and J. Marsaglia. 2004. Evaluating the Anderson-Darling Distribution. *Journal of Statistical Software*, 9:2, .

Simulation algorithms: run simulation

Running a simulation involves generating data for each of the simulated inputs, evaluating the predictive model based on the simulated data (along with values for any fixed inputs), and calculating metrics based on the model results.

Generating correlated data

Simulated values of input variables are generated so as to account for any correlations between pairs of variables. This is accomplished using the NORTA (Normal-To-Anything) method described by Biller and Ghosh (2006). The central idea is to transform standard multivariate normal variables to variables with the desired marginal distributions and Pearson correlation matrix.

Suppose that the desired variables are $X_j,\ j = 1, \cdots, k$, with the desired Pearson correlation matrix Σ_X, where the elements of Σ_X are given by ρ_{ij}. Then the NORTA algorithm is as follows:

1. For each pair X_i and X_j, where $i < j$, use a stochastic root finding algorithm (described in the following section) and the correlation ρ_{ij} to search for an approximate correlation ρ^*_{ij} of standard bivariate normal variables.
2. Construct the symmetric matrix Σ_z whose elements are given by ρ^*_{ij}, where $\rho^*_{ii} = 1$ and $\rho^*_{ij} = \rho^*_{ji}$.
3. Generate the standard multivariate normal variables $Z_1, \cdots, Z_k$ with Pearson correlation matrix Σ_z.
4. Transform the variables $Z_1, \cdots, Z_k$ to $X_1, \cdots, X_k$ using

   $$X_i = F_i^{-1}(\Phi(Z_i)), i = 1, \cdots, k$$

   where F_i is the desired marginal cumulative distribution, and $\Phi\,()$ is the cumulative standard normal distribution function. Then the correlation matrix of $X_1, \cdots, X_k$ will be close to the desired Pearson correlation matrix Σ_z.

Stochastic root finding algorithm

Given a correlation ρ_{ij}, a stochastic root finding algorithm is used to find an approximate correlation ρ^*_{ij} such that if standard bivariate normal variables Z_i and Z_j have the Pearson correlation ρ^*_{ij}, then after transforming Z_i and Z_j to X_i and X_j (using the transformation described in Step 4 of the previous section) the Pearson correlation between X_i and X_j is close to ρ_{ij}. The stochastic root finding algorithm is as follows:

1. Let $LowCorr = -1$ and $HighCorr = 1$
2. Simulate N samples of standard normal variables $Z_i^{(L)}$ and $Z_j^{(L)}$, $Z_i^{(H)}$ and $Z_j^{(H)}$, such that the Pearson correlation between $Z_i^{(L)}$ and $Z_j^{(L)}$ is *LowCorr* and the Pearson correlation between $Z_i^{(H)}$ and $Z_j^{(H)}$ is *HighCorr*. The sample size N is set to 1000.
3. Transform the variables $Z_i^{(L)}$, $Z_j^{(L)}$, $Z_i^{(H)}$ and $Z_j^{(H)}$ to the variables $X_i^{(L)}$, $X_j^{(L)}$, $X_i^{(H)}$ and $X_j^{(H)}$ using the transformation described in Step 4 of the previous section.

4. Compute the Pearson correlation between $X_i^{(L)}$ and $X_j^{(L)}$ and denote it as ρ_{ij}^L. Similarly, compute the Pearson correlation between $X_i^{(H)}$ and $X_j^{(H)}$ and denote it as ρ_{ij}^H.
5. If the desired correlation $\rho_{ij} \le \rho_{ij}^L$ or $\rho_{ij} \ge \rho_{ij}^H$ then stop and set $\rho_{ij}^* = LowCorr$ if $\rho_{ij} \le \rho_{ij}^L$ or set $\rho_{ij}^* = HighCorr$ if $\rho_{ij} \ge \rho_{ij}^H$. Otherwise go to Step 6.
6. Simulate N samples of standard bivariate normal variables $Z_i^{(M)}$ and $Z_j^{(M)}$ with a Pearson correlation of $MidCorr = \frac{1}{2}(LowCorr + HighCorr)$. As in Steps 3 and 4, transform $Z_i^{(M)}$ and $Z_j^{(M)}$to $X_i^{(M)}$ and $X_j^{(M)}$ and compute the Pearson correlation between $X_i^{(M)}$ and $X_j^{(M)}$, which will be denoted ρ_{ij}^M.
7. If $|\rho_{ij} - \rho_{ij}^M| \le \epsilon$ or $|HighCorr - LowCorr| \le \epsilon$ where ε is the tolerance level (set to 0.01), then stop and set $\rho_{ij}^* = MidCorr$. Otherwise go to Step 8.
8. If $\rho_{ij} > \rho_{ij}^M$, set $LowCorr = MidCorr$, else set $HighCorr = MidCorr$ and return to Step 6.

Inverse CDF for binomial, Poisson and negative binomial distributions

Use of the NORTA method for generating correlated data requires the inverse cumulative distribution function for each desired marginal distribution. This section describes the method for computing the inverse CDF for the binomial, Poisson and negative binomial distributions. Two parameterizations of the negative binomial distribution are supported. The first parameterization describes the distribution of the number of trials before the rth success, whereas the second parameterization describes the distribution of the number of failures before the rth success.

The choice of method for determining the CDF depends on the mean μ of the distribution. If $\mu \ge Threshold$, where *Threshold* is set to 20, the following approximate normal method will be used to compute the inverse CDF for the binomial distribution, the Poisson distribution and the second parameterization of the negative binomial distribution.

$$X = \left[F^{-1}(\Phi(Z))\right] = \left[\sigma\left(\Phi^{-1}(\Phi(Z))\right) + \mu\right] = [\sigma Z + \mu]$$

For the first parameterization of the negative binomial distribution, the formula is as follows:

$$X = [\sigma Z + \mu] + r$$

The parameters μ and σ are given by:

- **Binomial distribution.** $\mu = NP$ and $\sigma = \sqrt{NP(1-P)}$, where N is the number of trials and P is the probability of success.
- **Poisson distribution.** $\mu = \lambda$ and $\sigma = \sqrt{\lambda}$, where λ is the rate parameter.
- **Negative binomial distribution (both parameterizations).** $\mu = r\frac{1-\theta}{\theta}$ and $\sigma = \sqrt{r\frac{1-\theta}{\theta^2}}$, where r is the specified number of successes and θ is the probability of success.

The notation $[x]$ used above denotes the integer part of x.

If $\mu \le Threshold$ then the bisection method will be used.

Suppose that *x* and *z* are the values of *X* and *Z* respectively, where *X* is a random variable with a binomial, Poisson or negative binomial distribution, and *Z* is a random variable with the standard

$f(x)$

normal distribution. The objective function to be used in the bisection search method is as follows:

- **Binomial distribution.** $f(x) = 1 - \Pr(B(x+1, N-x) \le P) - \Phi(z)$
- **Poisson distribution.** $f(x) = 1 - \Pr(G(x+1, 1) \le \lambda) - \Phi(z)$
- **Negative binomial distribution (second parameterization).** $f(x) = \Pr(B(r, x+1) \le \theta) - \Phi(z)$

where $B(\alpha, \beta)$ and $G(\alpha, \beta)$ are random variables with the beta distribution and gamma distribution, respectively, with parameters α and β.

The bisection method is as follows:

1. If $f(\mu) = 0$ then stop and set $x = [\mu + 0.5]$. Otherwise go to step 2 to determine two values x_1 and x_2 such that $f(x_1) \times f(x_2) \le 0$.
2. If $f(\mu) > 0$ then let $x_1 = 0$ and $x_2 = \mu$. If $f(\mu) < 0$ then let $x_1 = 2^{J-1} \times \mu$ and $x_2 = 2^J \times \mu$, where J is the minimum integer such that $f(x_1) \times f(x_2) \le 0$.
3. Let $m = \frac{1}{2}(x_1 + x_2)$. If $|f(m)| < \epsilon$ or $|x_1 - x_2 < 1$ where ϵ is a tolerance level, which is set to 10^{-6}, then stop and set $x = [m + 0.5]$. Otherwise go to step 4.
4. If $f(m) > 0$, let $x_2 = m$, else let $x_1 = m$ and return to step 3.

Note: The inverse CDF for the first parameterization of the negative binomial distribution is determined by taking the inverse CDF for the second parameterization and adding the distribution parameter r, where r is the specified number of successes.

Sensitivity measures

Sensitivity measures provide information on the relationship between the values of a target and the values of the simulated inputs that give rise to the target. The following sensitivity measures are supported (and rendered as Tornado charts in the output of the simulation):

- **Correlation.** Measures the Pearson correlation between a target and a simulated input.
- **One-at-a-time measure.** Measures the effect on the target of modulating a simulated input by plus or minus a specified number of standard deviations of the input.
- **Contribution to variance.** Measures the contribution to the variance of the target from a simulated input.

Notation

The following notation is used throughout this section unless otherwise stated:

Table 94-9
Notation

Notation	Description
n	Number of records of simulated data

X	An $n \times p$ matrix of values of the inputs to the predictive model. The rows $x_i = (x_{i1}, \ldots, x_{ip})$; $i = 1, \ldots, n$ contain the values of the inputs for each simulated record, excluding the target value. The columns $x_j^T = (x_{1j}, \ldots, x_{nj})$; $j = 1, \ldots, p$ represent the set of inputs.
y	An $n \times 1$ vector of values of the target variable, consisting of $y_i, i = 1, \ldots, n$
$F(X)$	A known model which can generate y from X
sa_j	The value of a sensitivity measure for the input x_j

Correlation measure

The correlation measure is the Pearson correlation coefficient between the values of a target and one of its simulated predictors. The correlation measure is not supported for targets with a nominal measurement level or for simulated inputs with a categorical distribution. For more information, see the topic “Pearson Correlation”.

One-at-a-time measure

The one-at-a-time measure is the change in the target due to modulating a simulated input by plus or minus a specified number of standard deviations of the distribution associated with the input. The one-at-a-time measure is not supported for targets with an ordinal or nominal measurement level, or for simulated inputs with any of the following distributions: categorical, Bernoulli, binomial, Poisson, or negative binomial.

The procedure is to modulate the values of a simulated input by the specified number of standard deviations and recompute the target with the modulated values, without changing the values of the other inputs. The mean change in the target is then taken to be the value of the one-at-a-time sensitivity measure for that input.

For each simulated input x_j for which the one-at-a-time measure is supported:

1. Define the temporary data matrix $X^{'} = X$
2. Add the specified number of standard deviations of the input’s distribution to each value of x_j in $X^{'}$.
3. Calculate $y^{'} = \mathrm{F}(X')$
4. Calculate $sa_j = \frac{1}{n}\sum_{i=1}^{n}\left(y^{'}{}_i - y_i\right)$
5. Repeat Step 2, but now subtracting the specified number of standard deviations from each value of x_j. Continue with Steps 3 and 4 to obtain the value of sa_j in this case.

Contribution to variance measure

The contribution to variance measure uses the method of Sobol (2001) to calculate the total contribution to the variance of a target due to a simulated input. The total contribution to variance, as defined by Sobol, automatically includes interaction effects between the input of interest and the other inputs in the predictive model.

The contribution to variance measure is not supported for targets with an ordinal or nominal measurement level, or for simulated inputs with any of the following distributions: categorical, Bernoulli, binomial, Poisson, or negative binomial.

Let X' be an additional set of simulated data, in the same form as X and with the same number of simulated records.

Define the following:

$$f_0 = \frac{1}{n}\sum_{i=1}^{n} y_i$$

$$D = \frac{1}{n}\sum_{i=1}^{n} y_i^2 - (f_0)^2$$

For each simulated input x_j for which the contribution to variance measure is supported, calculate

$$D_{x\setminus x_j} = \frac{1}{n}\sum_{i=1}^{n} y_i\left[F\left(X'_{x_j} + X_{x\setminus x_j}\right)\right]_i - (f_0)^2$$

where:

- $x \setminus x_j$ denotes the set of all inputs excluding x_j
- $X'_{x_j} + X_{x\setminus x_j}$ is a derived data matrix where the column associated with x_j is taken from X' and the remaining columns (for all inputs excluding x_j) are taken from X

The total contribution to variance from x_j is then given by

$$sa_j = \frac{D - D_{x\setminus x_j}}{D}$$

Note: When interaction terms are present, the sum of the sa_j over all simulated inputs for which the contribution of variance is supported, may be greater than 1.

References

Biller, B., and S. Ghosh. 2006. Multivariate input processes. In: *Handbooks in Operations Research and Management Science: Simulation*, B. L. Nelson, and S. G. Henderson, eds. Amsterdam: Elsevier Science, 123–153.

Sobol, I. M. 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55, 271–280.

SPATIAL ASSOCIATION RULES Algorithms

1. Introduction

Since association rule mining (Agrawal and Srikant, 1994) has been proposed, many algorithms have emerged and been successfully applied to real-world applications. In recent years, because of the importance and necessity of analyzing geospatial data in different industries, spatial data mining approaches have gained lots of interests. Among the existing spatial data mining approaches, the spatial association rule mining proposed by Koperski and Han (1995) is one of the most typical approaches for spatial pattern discovery.

As defined by Koperski and Han (1995), a spatial association rule is a rule that describes the implication of one or a set of spatial objects by another set of spatial objects in spatial databases. The spatial objects involved therefore can be classified into two groups: the *event object*s and the *geo-context objects*.

- The **event object**s are the research targets of rule mining, which means that the rules discovered are about the spatial patterns of the event objects.
- The **geo-context objects** are used to describe the patterns of the event objects.

The patterns are represented by *spatial relationships* (e.g., topological relationships) defined between each pair of an event object and a geo-context object.

An example of a spatial association rule, event and geo-context objects, and spatial relationships is given below.

Example 1
Rule: *Most crime cases within census tract No. 1 are close to Freya St (street).*

This is a spatial association rule discovered in a spatial database containing crime cases and map elements. The crime cases are event objects. *Census tract No. 1* and *Freya St* are specific geo-context objects. The whole set of geo-context objects may include all the census tracts, streets and roads, and other map elements in the database. *Within* and *close to* are spatial relationships defined between the crime cases and the census tract and the road.

The spatial relationships can be symbolically represented by the spatial predicates of event objects.

Definition 1: Spatial Predicate
A spatial predicate can be seen as a spatial attribute of an event object. It is defined by an ordered 2-tuple <*r*, *o*>, where *r* is a spatial relationship, *o* is a geo-context object.

By using the spatial predicates, the rule in Example 1 can be written as:

<Within, Tract1> ➔ <Close to, Freya St> (*a*%, *l*%) (1)

where <Within, Tract1> is the condition of the rule, and <Close to, Freya St> is the prediction of the rule, both of which are spatial predicates. In parenthesis, a% denotes condition support, implying how many crime cases (in percentage) satisfy the condition of the rule, i.e., <Within, Tract1>. Symbol l% denotes the rule lift, implying the ratio of confidence for the rule (the probability that the prediction is true given that the condition is true) to the prior probability of having the prediction of the rule. Therefore, lift measures the gain in prediction accuracy by using the rule. Rules without sufficiently large a% and l% could not be regarded as significant. Besides condition support and lift, there are other statistics controlling the rule generation, which will be explained in details in section 2.5.

A spatial association rule describes the spatial distribution pattern of a set of event objects by their spatial relationships with the geo-context objects. However, as analyzed by Dong et al. (2012b), existing spatial association rule mining approaches have a major limitation that they cannot effectively involve all available non-spatial information of the spatial objects. As a result, many interesting rules expressing richer information (e.g., the combinations of spatial and non-spatial information) cannot be found even if non-spatial information that could be useful for rule discovery is available.

This document describes the Generalized Spatial Association Rule (GSAR) mining algorithm, which remedies this significant shortcoming. GSAR is capable of exploiting all available information of the spatial objects, including spatial and non-spatial information.

2. Generalized Spatial Association Rule (GSAR)

GSAR combines and extends the merits of the traditional spatial association rule (Koperski and Han, 1995) and the generalized association rule (Srikant and Agrawal, 1995, Han and Fu, 1995) so that much more information can be involved in analysis than ever before.

The overall flow of the Generalized Spatial Association Rule (GSAR) mining algorithm includes the following steps.

Step 1: User specifies event objects and geo-context objects, as well as spatial predicates and criteria for mining association rules.

Step 2: Compute spatial relationships between event objects and geo-context objects, and construct spatial predicate transaction table.

Step 3: Involve non-spatial attributes of event objects, if provided.

Step 4: Involve non-spatial attributes of geo-context objects, if provided.

Step 5: Rule mining, and the output will be GSAR.

Each of the steps is described in the following subsections. We use the crime analysis in Example 1 as a sample scenario to explain each step.

2.1. Initialization

First, the user needs to specify which spatial objects in the inputs are event objects, and which are geo-context objects. Suppose we are given a set of crime history, where each crime case has latitude and longitude coordinates, and multiple map layers are available. To analyze the crime patterns using the map layers, a user can specify the crime cases as event objects, and all the census tracts (may appear as polygons in the database), streets and roads (may appear as polylines) as geo-context objects. Usually, some attributes of all these spatial objects are also available (e.g., the type of crime, the population density of census tracts, etc.). Such information is important for discovering interesting patterns. Typically, an attribute table is associated with each layer of spatial objects.

2.2. Construction of Spatial Predicate Transaction Table

To mine spatial association rules, a spatial predicate transaction table of event objects needs to be constructed. Each row of the transaction table contains the spatial predicates of one event object. Let us consider the crime analysis scenario in Example 1. A sample spatial predicate transaction table of the crime cases is given in Table 1, where the ID column represents crime case identifier. The "*close to*" relationship can be defined by a condition such as a distance less than 500 feet. The spatial relationships "*within*" and "*close to*" used here are two typical topological relationships. Other types of spatial relationships, such as directional relationships, can also be used.

Table 1: A Sample Spatial Predicate Transaction Table

ID	Spatial Predicates
1	<Within, Tract1>, <Close to, Freya St>, <Close to, Wellesley Av>
2	<Within, Tract2>
3	<Within, Tract1>, <Close to, Freya St>

By treating spatial predicates as items[1], a traditional association rule mining algorithm, such as Apriori, can be applied to the spatial predicate transaction table, producing spatial patterns of crimes as rules. Nonetheless, the spatial predicates only express the spatial attributes of event objects. Non-spatial attributes of reference and geo-context objects, which can also be useful for finding interesting rules, are not included in the transaction table. In fact, involving non-spatial attributes of event objects is straightforward, which is done in the next step.

2.3. Involving Non-Spatial Attributes of Event objects

In some cases where non-spatial attributes of event objects are not available, or the user does not want to involve them in analysis, this step can be omitted. Otherwise, the information can be involved by expanding the transaction table by joining available non-spatial attributes[2] of event objects by their unique

[1] Items can be flag-type conditions that indicate the presence or absence of a particular thing in a specific transaction or simply categories of categorical variables.

[2] The non-spatial information is either categorical, or discretized according to user specified cutpoints, or discretized automatically through equal width binning.

identifiers. Suppose each crime case has two non-spatial attributes, crime type and day of week on which it was reported. After joining, the resulting expanded transaction table will look like Table 2.

Table 2: Spatial Predicate Transaction Table Expanded from Table 1

ID	Non-spatial Attributes	Spatial Predicates
1	Drugs, Tuesday	<Within, Tract1>, <Close to, Freya St>, <Close to, Wellesley Av>
2	Robbery, Friday	<Within, Tract2>
3	Vehicle Theft, Monday	<Within, Tract1>, <Close to, Freya St>

2.4. Involving Non-Spatial Attributes of Geo-Context Objects

Available non-spatial attributes of geo-context objects are also important for finding interesting patterns. However, if such non-spatial attributes are involved (as above) and a traditional association rule mining algorithm is applied, a large number of redundant patterns that provide nothing interesting can be generated. For instance, suppose Freya St has a non-spatial attribute Load=Heavy. If we simply append this attribute to Table 2, treat it as a usual item, and run Apriori, the following itemset[3] of length two can be found as frequent:

{<Close to, Freya St>, <Close to, Load=Heavy>}

This itemset correlates Freya St with its attribute Load=Heavy. Nonetheless, it provides nothing new to the known information and therefore is redundant. Any rule generated from it thus is also meaningless. Ideally, such patterns should be prevented from pattern generation. However, traditional association rule mining treats items equally and independently. Therefore, in the above case, Freya St and Load=Heavy cannot be prevented from appearing within the same itemset or rule.

GSAR supports user-specified pairs of fields to exclude, so that results can be more relevant and interesting. After removing those pairs, generalized spatial predicates are inferred from spatial predicates. For example, as given in table 3, <Within, POPDEN=Low> and <Within, RMF=Avg> are generalized spatial predicates of <Within, Tract1>, and <Close to, Road> is a generalized spatial predicate of <Close to, Freya St>. We can find that a non-spatial attribute (or a concept) of a geo-context object can only appear in a generalized spatial predicate by replacing the corresponding geo-context object.

Table 3: Transaction Table Further Expanded from Table 2 with Generalized Spatial Predicates

ID	Non-spatial Attributes	Spatial Predicates	Generalized Spatial Predicates
1	Drugs, Tuesday	<Within, Tract1>, <Close to, Freya St>, <Close to, Wellesley Av>	<Within, POPDEN=Low>, <Within, RMF=Avg>, <Close to, Road>
2	Robbery, Friday	<Within, Tract2>	<Within, POPDEN=VeryHigh>, <Within, RMF=Avg>
3	Vehicle Theft, Monday	<Within, Tract1>, <Close to, Freya St>	<Within, POPDEN=Low>, <Within, RMF=Avg>, <Close to, Road>

[3] An itemset is a group of items which may or may not tend to co-occur within transactions.

2.5. Rule Mining

All the above steps can be regarded as data preparation for Generalized Spatial Association Rules (GSAR) mining. Now we give a definition of GSAR. A Generalized Spatial Association Rule (GSAR) extends the traditional spatial association rule so that it can contain (1) spatial relationships, (2) non-spatial attributes of event objects, and (3) non-spatial attributes of geo-context objects.

As can be seen, in the GSAR mining algorithm, above points (3) are expressed via the generalized spatial predicates. As described in previous sections, the expanded spatial predicate transaction table which contains non-spatial attributes of event objects (denoted by T), is further expanded with generalized spatial predicates and becomes T'. Taking this as input, frequent itemsets are generated from T', and pruning is applied on itemsets with length two. Then rules are generated. Details are summarized in Algorithm 1.

Algorithm 1: Finding Generalized Spatial Association Rules (GSAR)

Input: An expanded spatial predicate transaction table T; user input excluded field pairs S, minimum condition support a_{min}; minimum lift l_{min}; minimum rule support s_{min}; minimum confidence c_{min}; maximum rule length L.

Output: A set of rules

```
// Rule mining with redundant pattern pruning
1)  Start from iteration $k$=1. Each item is a candidate itemset at this level.
2)  Treat all the elements in $T'$ as ordinary items, and scan all candidate itemsets of size $k$ to see if there
    are any with frequency exceeding a predetermined threshold of minimum rule support $s_{min}$. If yes,
    continue; otherwise, the iteration ends.
3)  Find all candidate itemsets of length $k$+1 (for $k$=1 the itemsets covered by user input excluded field
    pairs $S$ are pruned). If such candidate itemsets are found, continue; otherwise, the iteration ends.
4)  From candidate itemsets, find frequent itemsets with support above or equal to $a_{min}$.
5)  Based on the frequent itemsets, generate association rules with lift values above or equal to $l_{min}$, and
    condition support above or equal to $a_{min}$, and confidence values above or equal to minimum
    confidence $c_{min}$.
6)  Increase $k$ by one. If $k \leq L$ then go to step 2); otherwise, the iteration ends.
```

Note that GSAR turns to simple Apriori if geo-context objects or pairs of fields to exclude are not given. More details of the GSAR algorithm, as well as its time complexity analysis, can be found in Dong et al. (2012).

2.6. Evaluation Measures

The different measures emphasize different aspects of the rules.

Name	Range	Comments
Condition support	[0,100%]	Proportion of input transactions that contain the condition.
Rule Support	[0,100%]	Proportion of input transactions that contain the entire rule: conditions, and prediction. This indicates what percentage of the prediction in the inputs can be predicted through the condition.
Confidence	[0,100%]	Ratio of rule support to condition support. This is the probability of having the prediction in the condition population.
Lift	[1.0, infinity)	Ratio of confidence for the rule to the prior probability of having the prediction. For example, if 10% of the entire population buys bread, then a rule that predicts whether people will buy bread with 20% confidence will have a lift of 20/10 = 2. If another rule tells you that people will buy bread with 11% confidence, then the rule has a lift of close to 1, meaning that having the condition does not make a lot of difference in the probability of having the prediction. In general, rules with lift different from 1 will be more interesting than rules with lift close to 1. The GSAR component requires minimum lift to be no less than 1.
Deployability	[0,100%]	The percentage of the transactions that contain the condition but do not contain the prediction. In product purchase terms, it basically means what percentage of the total customer base owns (or has purchased) the condition but has not yet purchased the prediction.

3. Appendix

3.1. Detection of most interesting rules

Denote $x_i^{(m)}$ as the value of rule i $(i = 1, \dots, \mathrm{I})$ along rule measurement dimension m $(m = 1, \dots, M)$. The rule i is considered "interesting" along dimension m if

$$x_i^{(m)} > \bar{X}^{(m)} + T * SD^{(m)}$$

where T is a constant, by default set to 3. $\bar{X}^{(m)}$ and $SD^{(m)}$ is the mean and standard deviation for dimension m, respectively. They are computed using the following steps:

1. Start with: $W_0^{(m)} = \bar{X}_0^{(m)} = 0$,

2. Compute statistics below for $i = 1$ to I. (Skip to next data record when $x_i^{(m)}$ is missing):

$$W_i^{(m)} = W_{i-1}^{(m)} + 1,$$

$$V_i^{(m)} = \frac{1}{W_i^{(m)}}(x_i^{(m)} - \overline{X}_{i-1}^{(m)})$$

$$\overline{X}_i^{(m)} = \overline{X}_{i-1}^{(m)} + V_i^{(m)}$$

3. After the last rule $i = \mathrm{I}$ has been processed, return the following result

$$\overline{X}^{(m)} = \overline{X}_I^{(m)}$$

$$SD^{(m)} = \sqrt{\frac{1}{I-1}\sum_{i=1}^{I}(x_i^{(m)} - \overline{X}_I^{(m)})^2}$$

The effect size of interestingness $Q_i^{(m)}$ is evaluated by

$$Q_i^{(m)} = \begin{cases} 0, & \text{if } x_i^{(m)} \le \overline{X}^{(m)} + T * SD^{(m)} \\ \frac{x_i^{(m)} - \overline{X}^{(m)}}{SD^{(m)}} - \mathrm{T}, & \text{if } x_i^{(m)} > \overline{X}^{(m)} + T * SD^{(m)} \end{cases}$$

A rule with greater $Q_i^{(m)}$ is more interesting.

References

[1]. Agrawal, R. and Srikant, R. (1994), "Fast algorithms for mining association rules," in *VLDB*, 1994, pp. 487–499.

[2]. Dong, W., Li, L., Zhou, C., Wang, Y., Li, M., Tian, C., and Sun, W. (2012), "Discovery of generalized spatial association rules," in *IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI)*, 2012, pp. 60–65.

[3]. Han, J. and Fu, Y. (1995) "Discovery of multiple-level association rules from large databases," in *VLDB*, 1995, pp. 420–431.

[4]. Koperski, K. and Han, J. (1995), "Discovery of spatial association rules in geographic information databases," in *Advances in spatial databases*, 1995, pp. 47–66.

[5]. Srikant, R. and Agrawal, R. (1995), "Mining generalized association rules," in *VLDB*, 1995, pp. 407–419.

SPATIAL TEMPORAL PREDICTION Algorithms

1. Introduction

1.1 Background

Spatio-temporal statistical analysis has many applications. For example, energy management for buildings or facilities, performance analysis and forecasting for service branches, or public transport planning. In these applications, measurements such as energy usage are often taken over space and time. The key questions here are what factors will affect future observations, what can we do to effect a desired change, or to better manage the system. In order to address these questions, we need to develop statistical techniques which can forecast future values at different locations, and can explicitly model adjustable factors to perform what-if analyses.

However, these analytical needs are not the focus of traditional spatio-temporal statistical research. In traditional statistical research, spatio-temporal analysis is treated just as an extension of spatial analysis and focuses more on looking for patterns in past data rather than forecasting future values. The traditional spatio-temporal research targets different application areas such as environmental research. There are, however, different types of spatio-temporal problems in which time is the key component. We therefore need to treat spatio-temporal analysis as a unique type of problem itself, not an extension to spatial analysis. Moreover, we need to explicitly model these factors to allow for what-if analysis. Although these kinds of problems could be addressed by traditional methods, the emphasis is quite different.

This algorithm assumes a fixed set of spatial locations (either point location or center of an area) and equally spaced time stamps common across locations. It can issue predicted or interpolated values at locations with no response measurements (but with available covariates). We call our model spatio-temporal prediction (STP).

The goal of the STP algorithm is to address the needs for solving the spatio-temporal problems. STP can generate predictions at any location within a 3D space for any future time. It also explicitly models the external factors so we can perform what-if analysis.

1.2 Handling of missing data

The algorithm is designed to accommodate missing values in the response variable, as well as in the predictors. We consider an observation at a given time point and location ‘complete’ if all predictors and the response are observed at that time and location. To allow for model fitting in spite of missing data, all of the following conditions must be met:

1. At each location, observations need to be complete for at least one sequence of at least $L + 2$ *consecutive* time points.
2. At each location s_i, for any pair of locations s_i, s_j, $s_j \neq s_i$, observations must be complete at *both* locations simultaneously for at least two sequences of $L + 2$ consecutive time points.

3. Overall, at least L sequences of at least $L+2$ consecutive time points must be present in the data (to allow for estimation of α).
4. The total number of complete samples must be at least equal to $D+L+2$, where D is the number of predictors, including the intercept, and L the user-specified lag.
5. After removing locations according to the rules above, no more than 5% of the remaining records should be incomplete. As an example, if after removing locations, n locations and m time stamps remain, no more than $n \times m \times .05$ records should be incomplete.

The above conditions should be verified in the following order:

Step 1. Remove locations that do not meet condition 1.

Step 2. Remove locations that violate condition 2 in the following order:

(a) Let $\mathcal{J}$ be the set of points that violate condition 2.

(b) Eliminate from the data set the observation(s) that violate condition 2 for the greatest number of pairs. In case of a tie, remove all observations that are tied.

(c) Update $\mathcal{J}$ by removing any observations that now no longer violate Condition 2. That is, remove observation that only violated the condition 2 in a pair with the observations that were removed in Step 2b.

(d) Iterate steps 2b and 2c until $\mathcal{J}$ is empty.

Step 3. If after Steps 1 and 2, conditions 3-5 are violated, the model cannot be fit.

2 Model

2.1 Notation

The following notation is used for the model inputs:

Name	Symbol	Type	Dimensions
Number of time stamps	$m > L$	integer	1
Number of measurement locations	$n \geq 3$	integer	1
Number of prediction grid points	N	integer	1
Number of predictors (including intercept)	D	integer	1
Index of time stamps	$t \in \{1, \dots, m\}$	integer	1
Spatial coordinates	$s \in \{s_1, \dots, s_n\}; s_j = (u_j, v_j, w_j)'$	vector	3×1
Targets observed at location s and time t	$Y_t(s)$	scalar	1
Targets observed at location s	$Y(s)$	vector	$m \times 1$
Targets observed at time t	Y_t	vector	$n \times 1$
Predictors observed at location s and time t	$X_t(s) = (X_{t,1}(s), \dots, X_{t,D}(s))'$	vector	$D \times 1$
Predictors observed at location s	$X(s) = (X_1(s), \dots, X_m(s))'$	matrix	$m \times D$
Predictors observed at time t	$X_t = (X_t(s_1), \dots, X_t(s_n))'$	matrix	$n \times D$
Maximum autoregressive time lag	$L > 0$	integer	1
Length of prediction steps	$H > 0$	integer	1

Notes

i. For a predictor that does not vary over space, $X_{t,d}(s_1) = X_{t,d}(s_2) = \cdots = X_{t,d}(s_n)$;

ii. For a predictor that does not evolve over time, $X_{1,d}(s) = X_{2,d}(s) = \cdots = X_{m,d}(s)$.

The following notation is used for model definition and computation:

Name	Symbol	Type	Dimension
Coefficient vector for linear model	$\boldsymbol{\beta} = (\beta_1, \dots, \beta_D)$	vector	D
Coefficient vector for AR model	$\boldsymbol{\alpha} = (\alpha_1, \dots, \alpha_L)$	vector	L
Vector of 1's	$1 = (1, \dots, 1)'$	vector	variable
Kronecker product	$\otimes$	operator	NA

2.1 Model structure

$$Y_t(s) = \sum_{d=1}^{D} \beta_d X_{t,d}(s) + Z_t(s) \qquad (1)$$

where $Z_t(s)$ is mean-zero space-time correlated random process. Users can specify whether an "intercept" term needs to be included in the model. The inference algorithm works with general "continuous" variables, and with or without intercept.

- Autoregressive model, AR(L) for time autocorrelation (Brockwell and Davis, 2002):

$$Z_t(s) = \sum_{l=1}^{L} \alpha_l Z_{t-l}(s) + \epsilon_t(s) \qquad (2)$$

Note that users need to specify the maximum AR lag L.

Let $\epsilon_t = (\epsilon_t(s_1), \dots, \epsilon_t(s_n))'$ be the AR residual vector at time t. Since the time autocorrelation effect has already been removed, $\epsilon_{L+1}, \dots, \epsilon_m$ are independent.

- Parametric or nonparametric covariance model for spatial dependence:

$$V(\epsilon_t) = \Sigma_S, t = L + 1, \dots, m \qquad (3)$$

where $\Sigma_S = \{R(s_i, s_j)\}_{i,j=1,\dots,n}$ is a $n \times n$ covariance matrix of spatial covariance functions $R(s, s') = Cov(Y_t(s), Y_t(s'))$ at observed locations. Two alternative ways of modeling the spatial covariance function $R(s_i, s_j)$ are implemented - a *variogram-based parametric* model (Cressie, 1993) and a *Empirical Orthogonal Functions (EOF)-based nonparametric* model (Cohen and Johnes, 1969; Creutin and Obled, 1982).

Note that users can specify which covariance model to be used.

- If a "parametric model" is chosen, the algorithm will automatically test for the goodness-of-fit. If the test suggests a parametric model is not adequate, the algorithm switch to EOF model fitting and issue prediction based on EOF model.
- If a EOF model is chosen, the switching test part will be skipped, and both model fitting and prediction will follow EOF-based algorithm.

Under this model decomposition, the covariance structure for the spatio-temporal process $Y = (Y'_{L+1}, \dots, Y'_m)'$ is of separable form

$$V(Y) = V(Z) = \Sigma = \Sigma_T \otimes \Sigma_S \qquad (4)$$

where $\Sigma_T = \{\gamma_T(t - t')\}_{t=L+1,\dots,m;t'=L+1,\dots,m}$ is the $(m - L) \times (m - L)$ AR(L) covariance matrix with the autocovariance function.

3 Estimation algorithm

This section provides details on the multi-step procedure to fit the STP model (see Figure 1) when the user specifies a "parametric model". If an "empirical model" is specified, the switching test part will be skipped, and both model fitting and prediction follows EOF-based algorithm.

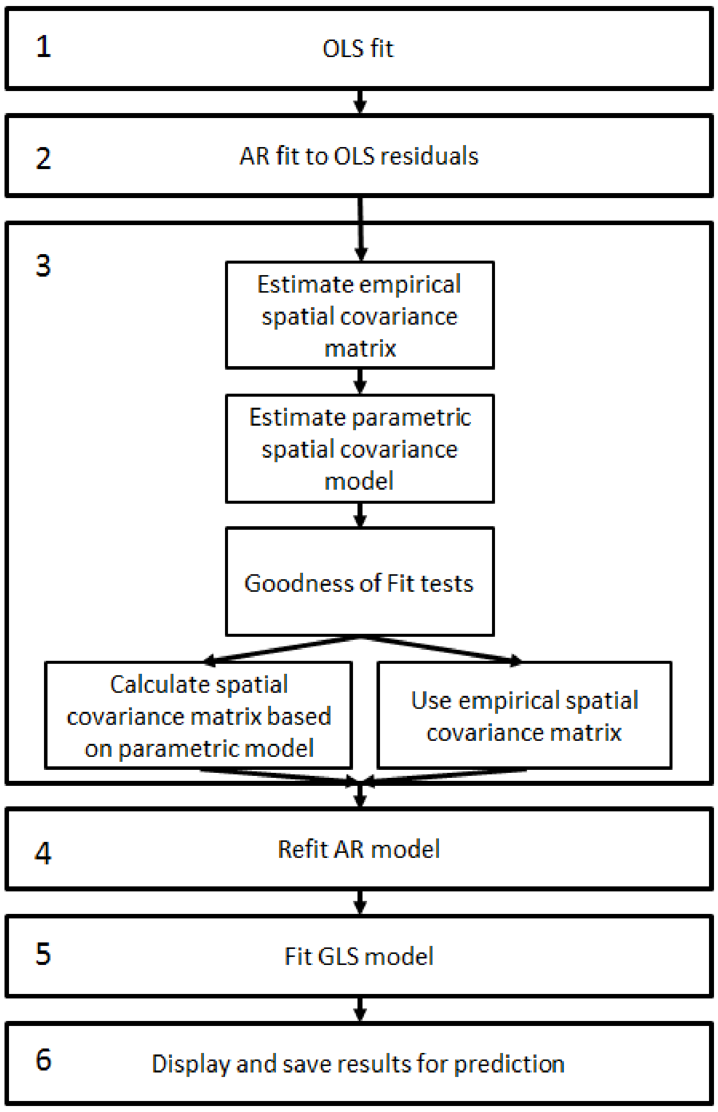

1 OLS fit
2 AR fit to OLS residuals
3
Estimate empirical spatial covariance matrix
Estimate parametric spatial covariance model
Goodness of Fit tests
Calculate spatial covariance matrix based on parametric model
Use empirical spatial covariance matrix
4 Refit AR model
5 Fit GLS model
6 Display and save results for prediction

Figure 1. Flowchart of algorithm steps for model fitting when a “parametric model” is specified.

Step 1: Fit regression model by ordinary least squares (OLS) regression using only observations that have no missing values (see Section 3.1).

We first ignore the spatio-temporal dependence in the data and simply estimate the fixed regression part by OLS and obtain the regression residuals $Z_t(s)$.

Step 2: Fit autoregressive model using only data without missing values (see Section 3.2).

Ignoring spatial dependence in OLS residuals $Z_t(s)$, we estimate autoregressive coefficients by fitting the regression model (2) and obtain the AR residuals $\epsilon_t(s)$.

Step 3: Fit spatial covariance model and test for goodness of fit on data without missing values (see Section 3.3).

We fit a parametric spatial covariance model. We perform two Goodness of Fit tests to decide whether to continue with the parametric covariance model or the empirical covariance matrix.

Step 4: Refit autoregressive model using augmented data (see Section 3.4).

We refit autoregressive model accounting for spatial dependence by generalized least squares (GLS) and obtain improved AR coefficients α.

Step 5: Refit Regression model using augmented data (see Section 3.5).

We obtain improved regression coefficients β by GLS to account for spatio-temporal correlation in the data.

Step 6: Save the results for use in output and prediction.

3.1 Fit regression model

We first ignore the spatio-temporal dependence in the data and simply estimate the fixed regression part by OLS. Assume that out of nm location-time combinations, q samples have missing values in either X or Y. Let $Y = (Y', \dots, Y')'$, a $(nm - q) \times 1$-vector and $X = (X_1', \dots, X_m')'$, a $(mn - q) \times D$ matrix, such that X and Y contain only complete observations, i.e., observations without any missing values. The OLS estimates of the regression coefficients are:

$$\widehat{\boldsymbol{\beta}} = (X'X)^{-1}X'Y \tag{5}$$

The residuals are:

$$\hat{Z} = Y - X\widehat{\boldsymbol{\beta}}. \tag{6}$$

3.2 Fit autoregressive model

We estimate autoregressive coefficients by OLS assuming no spatial correlation and AR(L) as model for time-series autocorrelation,

$$\hat{Z}_t = \alpha_1\hat{Z}_{t-1} + \cdots + \alpha_L\hat{Z}_{t-L} + \boldsymbol{\epsilon}_t, \tag{7}$$

where $\hat{Z}_t$ is a $n_t \times 1$ vector. Note that due to the existence of missing values, the number of locations n_t varies among different time points. Moreover, for each time points t, only locations with no missing values at $L + 1$ consecutive time points, i.e., $(t, t-1, \dots, t-L)$ can be used for model fitting, therefore, $\sum_{t=L+1}^{m} n_t \leq [n(m-L) - q]$.

Step 1: Construct $n_t \times L$ time lag matrix

Step 2: Let $\hat{Z}_{lag} = (\hat{Z}'_{L+1-lag}, \dots, \hat{Z}'_{m-lag})'$ and $\hat{Z}^* = (\hat{Z}'_{L+1}, \dots, \hat{Z}'_m)'$. Solve the linear system

$$(\hat{Z}'_{lag}\hat{Z}_{lag})\boldsymbol{\alpha} = \hat{Z}'_{lag}\hat{Z}^* \tag{9}$$

which is equivalent to solving

$$\left(\sum_{t=L+1}^{m} \hat{Z}'_{t-lag}\hat{Z}_{t-lag}\right)\boldsymbol{\alpha} = \sum_{t=L+1}^{m} \hat{Z}'_{t-lag}\hat{Z}_t \tag{10}$$

using the sweep operation to find estimate $\hat{\alpha}$.

Step 3: Compute the de-autocorrelated AR(L) residuals

$$\hat{\epsilon}_t = \hat{Z}_t - \hat{\alpha}_1\hat{Z}_{t-1} - \cdots - \hat{\alpha}_L\hat{Z}_{t-L}, t = L+1, \dots, m \tag{11}$$

3.3 Fit model and check goodness of fit for spatial covariance structure

We explicitly model the spatial covariance structure among locations, rather than using variogram estimation.

Under the assumption of the model (stationarity, AR-relationship removed), the mean of the residuals is 0 at all locations. We therefore estimate the unadjusted empirical covariances s_{ij} and correlations r_{ij} assuming mean 0, i.e.,

$$\boldsymbol{S} = [s_{ij}]_{i,j=1,\dots,n}, s_{ij} = \frac{1}{t_{ij}} \sum_{t} \hat{\epsilon}_t(s_i)\hat{\epsilon}_t(s_j) \qquad (12)$$

where t_{ij} is the number of complete residual pairs between locations s_i and s_j, and t indexes these pairs, i.e., the time points for which both $\hat{\epsilon}_t(s_i)$ and $\hat{\epsilon}_t(j)$ are non-missing.

$$r_{ij} = \frac{s_{ij}}{\sqrt{s_{ii}s_{jj}}} \qquad (13)$$

To determine whether to model the spatial covariance structure parametrically or to use the nonparametric EOF model, we perform the following two tests sequentially:

1. Fit parametric model to covariances using the parameter vector $\boldsymbol{\psi} = (\sigma^2, \theta, \tau^2)$ (Cressie 1993)

   $$Cov(\epsilon_t(s_i), \epsilon_t(s_j); \hat{\psi}) = \begin{cases} \hat{\sigma}^2 exp(-(h_{ij}/\hat{\theta})^p), & if\ h_{ij} > 0; \\ \hat{\sigma}^2 + \hat{\tau}^2, & otherwise. \end{cases} \qquad (14)$$

   where $h_{ij} = \|s_i - s_j\|_2$ is the Euclidean distance between locations s_i and s_j. Users need to specify the values for the order parameter p.

   $p \in [1, 2]$ is a user-defined parameter that determines the class of covariance models to be fit. $p = 1$ corresponds to an exponential covariance model, $p = 2$ results in a Gaussian covariance model and $p \in (1, 2)$ belongs to the powered exponential family.

   Next, determine if there is a significant decay over space by testing $H_0: -1/\theta^p \geq 0$. If we fail to reject H_0, we conclude that the decay over space is not significant, and EOF estimation will be used. If EOF estimation is used, there is not need to calculate θ, σ or τ, as we have concluded that they are invalid descriptions of the covariance matrix. In fact, there may not be valid solutions for these parameters, therefore they should not be estimated.

2. If the previous test rejects H_0, test for homogeneity of variances among locations: if homogeneity of variances is rejected, EOF estimation will be used. Otherwise, the parametric covariance model will be used.

3.3.1 Fit and test parametric model

a) Enforce a minimum correlation of +.01: if $r_{ij} < .01$, set $s_{ij} = .01\sqrt{s_{ii}s_{jj}}$ and $r_{ij} = .01$.

b) Let $\boldsymbol{s}$ be the vectorized lower triangular of the covariance matrix (excluding the diagonal, i.e., excluding variances), $\boldsymbol{r}$ be the vectorized lower triangular of the correlation matrix (excl. diagonal), and $\boldsymbol{h}$ the corresponding vector of pairwise distances between the n locations. $\boldsymbol{s}, \boldsymbol{r}$ and $\boldsymbol{h}$ are each vectors of length $n(n-1)/2$.

Define $\varphi = -1/\theta^p$. Fit the linear model $\ln \boldsymbol{s} = \ln \sigma^2 + \varphi \boldsymbol{h}^p$ using a GLS fit:

$$\boldsymbol{A} = [1, \boldsymbol{h}^p] \tag{15}$$

$$\boldsymbol{V}^{-1} = \frac{1}{2}\boldsymbol{T}(\boldsymbol{B}^{-1} - cbb')\boldsymbol{T} \tag{16}$$

where$\boldsymbol{b} = 2\boldsymbol{r}^2/(1 - \boldsymbol{r}^2)$, $\boldsymbol{r}^2$ is obtained by squaring each element of vector $\boldsymbol{r}$, $\boldsymbol{B}^{-1} =$ diag$(\boldsymbol{b})$, and scalar $c = 1/(1 + \boldsymbol{1}'\boldsymbol{B}^{-1}\boldsymbol{1})$. Also, let $\boldsymbol{T} =$ diag$[\sqrt{t_k}]$, $k = 1, \dots, n(n-1)/2$, where t_k is the number of pairs of de-autocorrelated residuals in the calculation of the corresponding element r_k in $\boldsymbol{r}$, i.e., the number of observations pairs that went into calculating r_k, which may be different for each entry of the covariance matrix, depending on missing values. Note that t_k corresponds to the vectorized lower triangular of $[t_{ij}]_{i,j=1,\dots,n}$, where t_{ij} are as defined in (12).

Let $\boldsymbol{\eta} = (\ln \sigma^2, \varphi)$, the GLS estimator can be calculated as

$$\hat{\boldsymbol{\eta}} = (\boldsymbol{A}'\boldsymbol{V}^{-1}\boldsymbol{A})^{-1}\boldsymbol{A}'\boldsymbol{V}^{-1}\ln \boldsymbol{s}$$

The standard error for $\hat{\boldsymbol{\eta}}$ will be $se(\hat{\boldsymbol{\eta}}) = \sqrt{\text{diag}[(\boldsymbol{A}'\boldsymbol{V}^{-1}\boldsymbol{A})^{-1}]}$.

Calculate the test statistic $z_1 = \frac{\hat{\varphi}}{se(\hat{\varphi})}$. If $z_1 \geq z_{.05}$, where $z_{.05}$ is the .05 quantile of the standard normal distribution (or critical value for selected level of significance γ_1), then all following calculations will be performed using the empirical spatial covariance matrix, i.e., $\Sigma_{\boldsymbol{S}} = \boldsymbol{S}$, and the nonparametric EOF model will be used for prediction. Equivalently, a p-value p_1 can be calculated by evaluating the standard Normal cumulative distribution function (CDF) at z_1 (i.e., $p_1 = P(Z < z_1)$). If $p_1 \geq$ level of significance γ_1, then all following calculations will be performed using the empirical covariance matrix.

c) If the previous test does reject H_0 (i.e., we have not yet decided to continue with the empirical covariance matrix), continue to perform the following test: Let $v = (s_{11}, s_{22}, \dots, s_{nn})'$ be the $(n \times 1)$-vector of location-specific variances. Calculate the weighted mean variance $\bar{v}$

$$\bar{v} = 1'\boldsymbol{W}^{-1}v/(1'\boldsymbol{W}^{-1}1) = 1'\boldsymbol{W}^{-1}v \Big/ \sum_{i,j} w^*_{ij} \tag{17}$$

where $\boldsymbol{W} = [w_{ij}] = [s^2_{ij}/t_{ij}]_{i,j=1,\dots,n}$ is an $n \times n$ matrix, where t_{ij} is defined as in (12), and $\boldsymbol{W}^{-1} = [w^*_{ij}]_{i,j=1,\dots,n}$.

Calculate the test statistic $z_2 = (\boldsymbol{v} - \bar{v})'\boldsymbol{W}^{-1}(\boldsymbol{v} - \bar{v})$. If $z_2 \geq \chi^2_{n-1,.95}$ (or critical value for $[1 -$ selected level of significance $\gamma_2]$), all following calculations will be performed using the empirical spatial covariance matrix, i.e., $\Sigma_{\boldsymbol{S}} = \boldsymbol{S}$, and the nonparametric EOF model will be used for prediction. Equivalently, one may compute a p-value p_2 by evaluating 1 minus the χ^2_{n-1} – CDF: $p_2 = P(\chi^2_{n-1} > z_2)$. If $p_2 <$ level of significance γ_2, then all following calculations will be performed using the empirical spatial covariance matrix.

d) If the two tests in b) and c) do not indicate a switch to the EOF model, all following calculations will be performed using the parametric covariance model, i.e., the spatial covariance matrix $\Sigma_S$is constructed according to (14). Recall that $\eta = (ln\, \sigma^2, -1/\theta^p)$.

The missing parameter τ^2 is derived as $\widehat{\tau^2} = max\left\{0, \frac{1}{n}\sum_{i=1,\dots,n} s_{ii} - exp\left[\widehat{ln\, \sigma^2}\right]\right\}$.

3.4 Re-fit autoregressive model

We refit the autoregressive model accounting for spatial dependence using GLS with augmented data:

Step 1: Compute the Cholesky factorization $\mathbf{\Sigma}_S = \boldsymbol{H}_S\boldsymbol{H}'_S$ and the inverse matrix $\boldsymbol{H}'_S$.

Step 2: Substitute 0 for missing values such that $\hat{\boldsymbol{Z}}_{t-lag,impute}$ is an $n \times L$ matrix and $\hat{\boldsymbol{Z}}_{t,impute}$ is a vector of length n.

Step 3: Augment predictor matrix as follows. Let $\hat{\boldsymbol{Z}}_{lag,impute} = \left(\hat{\boldsymbol{Z}}'_{L+1-lag,impute}, \cdots, \hat{\boldsymbol{Z}}'_{m-lag,impute}\right)'$ be a $n(m-L) \times L$ matrix and $\hat{\boldsymbol{Z}}_{impute} = \left(\hat{\boldsymbol{Z}}'_{L+1,impute}, \cdots, \hat{\boldsymbol{Z}}'_{m,impute}\right)'$ is a vector of length $n(m-L)$, then

$$\hat{\boldsymbol{Z}}_{lag,aug} = \left(\hat{\boldsymbol{Z}}_{lag,impute}, \cdots, \mathbf{I}_{Zmiss}\right)$$

where $\mathbf{I}_{Zmiss}$ is a $n(m-L) \times q_Z$ indicator matrix given q_Z the total number of rows with missing values in either $\hat{\boldsymbol{Z}}^*$ or $\hat{\boldsymbol{Z}}_{lag}$. If there is a missing value in the ith row of either $\hat{\boldsymbol{Z}}^*$ or $\hat{\boldsymbol{Z}}_{lag}$, and if this is the jth out of all q_Z rows that have missing values, then the jth column of $\mathbf{I}_{Zmiss}$ is all 0 except for the ith element, which is set to 1.

Step 4: Remove the spatial correlation: $\tilde{\boldsymbol{Z}}_{t-lag,aug} = \boldsymbol{H}_S^{-1}\hat{\boldsymbol{Z}}_{t-lag,aug}$ and $\tilde{\boldsymbol{Z}}_{t,impute} = \boldsymbol{H}_S^{-1}\hat{\boldsymbol{Z}}_{t,impute}$, where $\hat{\boldsymbol{Z}}_{t-lag,aug}$ are the submatrices of $\hat{\boldsymbol{Z}}_{lag,aug}$ that correspond to the rows of the matrices $\hat{\boldsymbol{Z}}_{t-lag,impute}$.

Step 5: Use the same computational steps as for the linear system in equation (10) to solve the linear system

$$\left(\sum_{t=L+1}^{m} \tilde{\boldsymbol{Z}}'_{t-lag,aug}\tilde{\boldsymbol{Z}}_{t-lag,aug}\right)\boldsymbol{\alpha}_{aug} = \sum_{t=L+1}^{m} \tilde{\boldsymbol{Z}}'_{t-lag,aug}\tilde{\boldsymbol{Z}}_{t,impute} \qquad (18)$$

where $\boldsymbol{\alpha}_{aug}$ is a vector of length $L + q_Z$, and there are $L^* + q^*_Z$ non-redundant parameters in above linear system. The AR coefficient estimate $\hat{\boldsymbol{\alpha}}$ is the subvector consisting of the first L elements of $\hat{\boldsymbol{\alpha}}_{aug}$, there are L^* non-redundant parameters in first D elements of $\hat{\boldsymbol{\alpha}}_{aug}$, and q_Z^* non-redundant parameters in last q_Z elements of $\hat{\boldsymbol{\alpha}}_{aug}$.

3.5 Re-fit Regression model

Refit regression model by GLS using augmented data to account for spatio-temporal correlation in the data.

Step 1: Substitute the following for missing values such that $\boldsymbol{X}_{impute}$ is a $nm \times D$ matrix and $\boldsymbol{Y}_{impute}$ is a vector of length nm: at location s_i, use the mean of $\boldsymbol{Y}(s_i)$ and the mean of each predictor in $\boldsymbol{X}(s_i)$.

Step 2: Augment predictor matrix as follows.

$$\boldsymbol{X}_{aug} = (\boldsymbol{X}_{impute}, \ \boldsymbol{I}_{Xmiss})$$

where $\mathbf{I}_{Xmiss}$ is a $nm \times q$ indicator matrix given q the total number of rows with missing values in either $\boldsymbol{X}$ or $\boldsymbol{Y}$. If there is a missing value in ith row of either $\boldsymbol{X}$ or $\boldsymbol{Y}$, and if this is the jth out of all q rows that have missing value, then the jth column of $\mathbf{I}_{Xmiss}$ is all 0 except for the ith element, which is 1.

Step 3: Remove the spatial correlation: $\tilde{\boldsymbol{X}}_{t,aug} = \boldsymbol{H}_S^{-1}\boldsymbol{X}_{t,aug}$ and $\tilde{\boldsymbol{Y}}_{t,impute} = \boldsymbol{H}_S^{-1}\boldsymbol{Y}_{t,impute}$.

Step 4: Remove the autocorrelation:

$$\breve{\boldsymbol{X}}_{t,aug} = \tilde{\boldsymbol{X}}_{t,aug} - \hat{\alpha}_1\tilde{\boldsymbol{X}}_{t-1,aug} - \cdots - \hat{\alpha}_L\tilde{\boldsymbol{X}}_{t-L,aug}, t = L+1, \dots, m \tag{19}$$

$$\breve{\boldsymbol{Y}}_{t,impute} = \tilde{\boldsymbol{Y}}_{t,impute} - \hat{\alpha}_1\tilde{\boldsymbol{Y}}_{t-1,impute} - \cdots - \hat{\alpha}_L\tilde{\boldsymbol{Y}}_{t-L,impute}, t = L+1, \dots, m \tag{20}$$

Step 5: Solve the linear system

$$(\breve{\boldsymbol{X}}'_{aug}\breve{\boldsymbol{X}}_{aug})\boldsymbol{\beta}_{aug} = \breve{\boldsymbol{X}}'_{aug}\breve{\boldsymbol{Y}}_{impute} \tag{21}$$

where $\breve{\boldsymbol{Y}}_{impute} = (\breve{\boldsymbol{Y}}'_{L+1,impute}, \dots, \breve{\boldsymbol{Y}}'_{m,impute})'$, an $n(m-L)\times 1$-vector and $\breve{\boldsymbol{X}}_{aug} = (\breve{\boldsymbol{X}}'_{L+1,aug}, \dots, \breve{\boldsymbol{X}}'_{m,aug})'$, a $n(m-L)\times(D+q)$ matrix, $\boldsymbol{\beta}_{aug}$ is a vector of length $D+q$, and there are $D^* + q^*$ non-redundant parameters in above linear system. The regression coefficients estimate $\hat{\boldsymbol{\beta}}$ is the subvector consisting of first D elements of $\hat{\boldsymbol{\beta}}_{aug}$, there are D^* non-redundant parameters in first D elements of $\hat{\boldsymbol{\beta}}_{aug}$, and q^* non-redundant parameters in last q elements of $\hat{\boldsymbol{\beta}}_{aug}$.

3.6 Statistics to display

3.6.1 Goodness of Fit statistics

We present statistics referring to the three main elements of the model: the mean structure, the spatial covariance structure, and the temporal structure.

1. **Goodness of fit mean structure model $\boldsymbol{X\beta}$:**

   Let Q be the set of observations $(Y_t(s), \boldsymbol{X}_t(s))$ that have missing values in either $Y_t(s)$ or $\boldsymbol{X}_t(s)$. Note that q has been defined as the number of observations in Q.

   Calculate the mean squared error (MSE) and an R^2 statistic based only on complete observations:

$$\text{MSE} = \sum_{\substack{s\in\{s_1,\dots,s_n\};\\ t=1,\dots,m;\\ Y_t(\boldsymbol{S})\notin Q}} \left(Y_t(s) - \hat{Y}_t(s)\right)^2/(nm - q - D^*) \tag{22}$$

$$R^2 = \begin{cases} 1 - \sum\limits_{\substack{s\in\{s_1,\dots,s_n\};\\ t=1,\dots,m;\\ Y_t(\boldsymbol{S})\notin Q}} \left(Y_t(s)-\hat{Y}_t(s)\right)^2 \Big/ \sum\limits_{\substack{s\in\{s_1,\dots,s_n\};\\ t=1,\dots,m;\\ Y_t(\boldsymbol{S})\notin Q}} Y_t(s)^2, & \text{if there is no intercept} \\ 1 - \sum\limits_{\substack{s\in\{s_1,\dots,s_n\};\\ t=1,\dots,m;\\ Y_t(\boldsymbol{S})\notin Q}} \left(Y_t(s)-\hat{Y}_t(s)\right)^2 \Big/ \sum\limits_{\substack{s\in\{s_1,\dots,s_n\};\\ t=1,\dots,m;\\ Y_t(\boldsymbol{S})\notin Q}} (Y_t(s)-\bar{Y}_t(s))^2, & \text{if there is an intercept} \end{cases} \tag{23}$$

   where $\hat{Y}_t(s) = \boldsymbol{X}'_t(s)\boldsymbol{\beta}$, D^* is the number of non-redundant parameters of re-fitted regression in first D elements of $\hat{\boldsymbol{\beta}}_{aug}$, and $\bar{Y}_t(s)$ is the mean of Y only on complete observations. Note that for this calculation the original (untransformed) observations $\boldsymbol{Y}$ and covariates $\boldsymbol{X}$ are used. Alternatively, we can calculate the adjusted R^2

2. **Goodness of fit for AR model:**

   Present *t*-tests for AR parameters based on variance estimates in item 3 in Section 3.6.2.

3. **Goodness of fit of spatial covariance model:**

   Present the test statistics listed in item 5 in Section 3.6.2.

3.6.2 Model and parameter estimates

The following information should be displayed as a summary of the model:

1. Model coefficients $\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\alpha}}$ obtained in Sections 3.4 and 3.5
2. Standard errors of elements of $\boldsymbol{\beta}$ based on $V(\widehat{\boldsymbol{\beta}})$, the covariance matrix of $\widehat{\boldsymbol{\beta}}$, which is the upper $D \times D$ submatrix of $V(\widehat{\boldsymbol{\beta}}_{aug})$:

$$V(\widehat{\boldsymbol{\beta}}_{aug}) = \frac{SS_e}{df_e} \times \left(\breve{\boldsymbol{X}}'_{aug}\breve{\boldsymbol{X}}_{aug}\right)^{-1} = \frac{SS_e}{df_e} \times \left(\sum_{t=L+1}^{m} \breve{\boldsymbol{X}}'_{t,aug}\breve{\boldsymbol{X}}_{t,aug}\right)^{-1} \qquad (25)$$

   where

   - $SS_e = \sum_{i=1}^{N}\left(\breve{\boldsymbol{Y}}_{impute} - \left(\breve{\boldsymbol{Y}}_{impute}\right)^*\right)^2 = \tilde{r}_{\breve{Y}\breve{Y}}(n(m-L)-1)V(\breve{\boldsymbol{Y}}_{impute})$,
     - $\left(\breve{\boldsymbol{Y}}_{impute}\right)^*$ is the predicted value based on estimated $\widehat{\boldsymbol{\beta}}$,
     - $\tilde{r}_{\breve{Y}\breve{Y}}$ is corresponding element of $\breve{\boldsymbol{Y}}_{impute}$ in the correlation matrix of re-fitted regression after sweep operation,
     - $n(m-L)$ is number of transformed records used in equation (21) for re-fit regression ,
     - and $V(\breve{\boldsymbol{Y}}_{impute})$ is variance of $\breve{\boldsymbol{Y}}_{impute}$.
   - $df_e = n(m-L) - p$, and $p = D^* + q^*$ is the number of non-redundant parameters in re-fitted regression.

   Based on these standard errors, t-test statistics and/or p-values may be computed and displayed according to standard definitions and output scheme of linear models (please refer to linear model documentation):

   (a) For each element β_j of $\widehat{\boldsymbol{\beta}}$ and the corresponding j-th diagonal element of $V(\widehat{\boldsymbol{\beta}})$, $j = 1, \dots, D$, compute the t-statistic $t_j = \beta_j / \sqrt{V(\widehat{\boldsymbol{\beta}})_{jj}}$

   (b) The p-value corresponding to t_j is $2 \times$ the value of the cumulative distribution function of a t-distribution with $nm - q - D^*$ degrees of freedom, i.e., $p_j = 2 \cdot$

$(1 - P(t_{nm-q-D^*} \leq |t_j|))$.

Note that depending on the implementation of the GLS estimation in Section 3.5, $(\breve{X}'_{aug}\breve{X}_{aug})^{-1}$ may have already been computed, in which case this expression does not need to be recalculated.

3. Standard errors of $\boldsymbol{\alpha}$ based on $V(\hat{\boldsymbol{\alpha}})$, the covariance matrix of $\hat{\boldsymbol{\alpha}}$, which is the upper $L \times L$ submatrix of $V(\hat{\boldsymbol{\alpha}}_{aug})$:

$$V(\hat{\boldsymbol{\alpha}}_{aug}) = \frac{SS_e^*}{df_e^*} \times \left(\sum_{t=L+1}^{m} \tilde{\mathbf{Z}}'_{t-lag,aug}\tilde{\mathbf{Z}}_{t-lag,aug} \right)^{-1} \qquad (26)$$

where

- $SS_e^* = \sum_{i=1}^{N}(\tilde{Z}_{t,impute} - (\tilde{Z}_{t,impute})^*)^2 = \tilde{r}_{\tilde{Z}\tilde{Z}}(n(m-L)-1)V(\tilde{Z}_{t,impute})$,
  - $(\tilde{Z}_{t,impute})^*$ is the predicted value based on estimated $\hat{\alpha}$ and $\tilde{Z}_{t-lag,aug}$
  - $\tilde{r}_{\tilde{Z}\tilde{Z}}$is corresponding element of $\tilde{Z}_{t,impute}$ in the correlation matrix of re-fitted autoregressive model after sweep operation,
  - $n(m-L)$ is number of transformed records used in equation (18) for re-fit autoregressive,
  - and $V(\tilde{Z}_{t,impute})$ is variance of $\tilde{Z}_{t,impute}$.
- $df_e^* = n(m-L) - p_{AR}$, and $p_{AR} = L^* + q_Z^*$ is the number of non-redundant parameters in re-fitted autoregressive model.

Based on these standard errors, t-test statistics and/or p-values may be computed and displayed according to standard definitions and output scheme of linear models.

(a) For each element α_j of $\hat{\boldsymbol{\alpha}}$ and the corresponding j-th diagonal element of $V(\hat{\boldsymbol{\alpha}})$, $j = 1, \dots, L$, compute the t-statistic $t_j = \alpha_j / \sqrt{V(\hat{\boldsymbol{\alpha}})_{jj}}$

(b) The p-value corresponding to t_j is $2 \times$the value of the cumulative distribution function of a t-distribution with $\sum_{t=1}^{m} n_t - L^*$ degrees of freedom, i.e., $p_j = 2 \cdot (1 - P(t_{\sum_{t=1}^{m} n_t - L^*} \leq |t_j|))$.

4. Indicator of which method has been automatically chosen to model spatial covariances, either empirical covariance (EOF) or parametric variogram model.

5. Test statistics from goodness of fit tests for parametric model:

   - Test statistic z_1, p-value p_1, level of significance γ_1 used for automated test for fit of slope parameter

   - Test statistic z_2, p-value p_2, level of significance γ_2 used for testing homogeneity of variances

6. Parametric covariance parameters $\hat{\boldsymbol{\psi}}$ if parametric model has been chosen

3.6.3 Tests of effects in Mean Structure Model (Type III)

For each effect specified in the model, type III test matrix L is constructed and H_0: $L_i\beta = 0$ is tested. Construction of type III matrix L as well as generating estimable function (GEF) is based on the generating matrix H, which is the upper $D \times D$ submatrix of $(\breve{X}'_{aug}\breve{X}_{aug})^{-1}\breve{X}'_{aug}\breve{X}_{aug}$, such that $L_i\beta$ is estimable. It involves parameters only for the given effect. For type III analysis, L does not depend on the order of effects specified in the model. If such a matrix cannot be constructed, the effect is not testable.

Then the L matrix is then used to construct the test statistic

$$F = \frac{\hat{\beta}'L'(L\Sigma L')^{-1}L\hat{\beta}}{r_c}$$

where

- $\hat{\beta}$ is the subvector of the first D elements of $\hat{\beta}_{aug}$ obtained in Step 5 of Section 3.5,
- $r_c = rank(L\Sigma L')$,
- Σ is the covariance matrix of $\hat{\beta}$, which is the upper $D \times D$ submatrix of $V(\hat{\beta}_{aug})$ defined in equation (25).

The statistic has an approximate F distribution. The numerator degrees of freedom $df1$ is r_c and the denominator degrees of freedom $df2$ is $nm - q - D^*$, where D^* is the number of non-redundant parameters in the first D parameters of refitted regression model obtained in Section 3.5. Then the p-values can be calculated accordingly.

An additional test also should be computed, which is similar to “corrected model” if there is an intercept or “model” if there is no intercept in ANOVA table in linear regression. Essentially, the null hypothesis is regression parameters (except intercept if there is on) are zeros. The test statistic would be the same as the above F statistic except the L matrix is from GEF. If there is no intercept, the L matrix is the whole GEF. If there is an intercept, the L matrix is GEF without the first row which corresponds to the intercept.

Statistics saved for Test of effects in Mean Structure Model (including corrected model or model):

- F statistics
- $df1$
- $df2$
- p-value

3.6.4 Location clustering for spatial structure visualization

Large spatial covariance matrix or correlation matrix are not suitable to demonstrate the relation among the locations. Grouping method, also called community detection or position analysis (Wasserman, 1994), can be used to identify some representative location clusters. To simplify the implementation, hierarchical clustering (Johnson, 1967) is used to detect clusters among locations based on STP model spatial statistics.

Please note location clustering is only supported when empirical nonparametric covariance model is used.

Given a set of n locations $\{s_1, \dots, s_n\}$ in STP to be clustered, and their corresponding spatial correlation matrix R, a $n*n$ matrix, as the similarity matrix

$$R = [r_{ij}]_{i,j=1,\dots,n}$$

SPATIAL TEMPORAL PREDICTION Algorithms

Given similarity threshold α with default value 0.2, and N_C with default value 10, the process of location clustering is described in following steps, which is based on the basic process of hierarchical clustering.

Step 1. Initialize the clusters and similarities:

- Assign each location s_i to a cluster $C_i (i = 1, \dots, n)$. So that for n locations, the total number of clusters $n_C = n$ at the beginning, and each cluster has just one location,
- Define the set of clusters: C,
- Define similarity matrix

$$R^C = [r_{ij}^C]_{i,j=1,\dots,n}$$

where the similarity r_{ij}^C between the clusters C_i and C_j is the similarity r_{ij} between location s_i and s_j.

Step 2. Find 2 clusters C_i and C_j in C with largest similarity $\max(r_{ij}^c)$,

If $max(r_{ij}^C) > \alpha$:

- Merge C_i and C_j into a new cluster $C_{\langle i,j \rangle}$ to include all locations in C_i and C_j,
- Compute similarities between the new cluster $C_{\langle i,j \rangle}$ and other clusters C_k , $k \neq i \text{ and } j$

$$r_{\langle i,j \rangle,k}^C = min(r_{ik}^C, r_{jk}^C)$$

- Update C by adding $C_{\langle i,j \rangle}$, discarding C_j and C_i. So $n_C = n_C - 1$.
- Update similarity matrix R^C by adding $r_{<i,j>,k}^c$, discarding r_{ik}^c and r_{jk}^c, go to step 3.

If $max(r_{ij}^C) \leq \alpha$, go to step 4.

Step 3. Repeat step 2.

Step 4. For all the detected clusters with more than 1 location, compute following statistics:

- Cluster size: n_{C_i} is the number of locations in C_i,
- Closeness:

$$d_i = \frac{1}{n_{C_i}(n_{C_i} - 1)/2} \sum r_{kl}, \forall\, s_k, s_l \in C_i, and\ k \neq l.$$

Step 5. Define clusters for interactive visualization:

- $C_{closeness}$: The first N_C clusters sorted by descending closeness d_i,
- C_{size}: The first N_C clusters sorted by descending cluster size n_{C_i}.

Step 6. Output the union for location cluster visualization:

$$C^* = C_{closeness} \cup C_{size}$$

Statistics saved for spatial structure visualization including:

4. Number of excluded locations during handling of missing data
5. Spatial correlation matrix $\boldsymbol{R} = [r_{ij}]_{i,j=1,\dots,n}$
6. Statistics of each output location cluster in C^*:
   - Closeness d_i
   - Cluster size n_{C_i}
   - Coordinates of locations in this cluster

3.7 Results saved for prediction

1. Model coefficients $\widehat{\boldsymbol{\beta}}, \widehat{\boldsymbol{\alpha}}$ the covariance estimate $V(\widehat{\boldsymbol{\beta}})$ as defined in (25).
2. Transformed regression residuals and predictors of L most recent observations for prediction:

$$\breve{\boldsymbol{Z}}_{m-l+1} = \boldsymbol{H'}_S^{-1}\boldsymbol{H}_S^{-1}\left(\boldsymbol{Y}_{m-l+1,impute} - \boldsymbol{X}_{m-l+1,aug}\widehat{\boldsymbol{\beta}}_{aug}\right), l = 1, \dots, L \qquad (27)$$

$$\breve{\boldsymbol{X}}_{m-l+1,impute} = \boldsymbol{H'}_S^{-1}\boldsymbol{H}_S^{-1}\boldsymbol{X}_{m-l+1,impute}, l = 1, \dots, L \qquad (28)$$

3. Indicator of which method has been chosen to model spatial covariances, either empirical covariance (EOF) or parametric variogram model.
4. Parametric covariance parameters $\breve{\psi}$ if parametric model has been chosen.
5. Coordinates of locations s.
6. Number of unique time points used for model build, m.
7. Number of records with missing values in the data set used in model building, q.
8. Spatial covariance matrix Σ_S.
9. H_S^{-1}, inverse of Cholesky factor of spatial covariance matrix.

4 Prediction

We perform the following procedure to issue predictions for future time $m+1, \dots, m+H$ at prediction locations $\boldsymbol{G} = (\boldsymbol{g}_1, \dots, \boldsymbol{g}_N)$ using the results saved in the output file (see Figure 2). The input data set format should include location $\boldsymbol{G}$, predictors $\boldsymbol{X}$ for $t = m+1, \dots, m+H$.

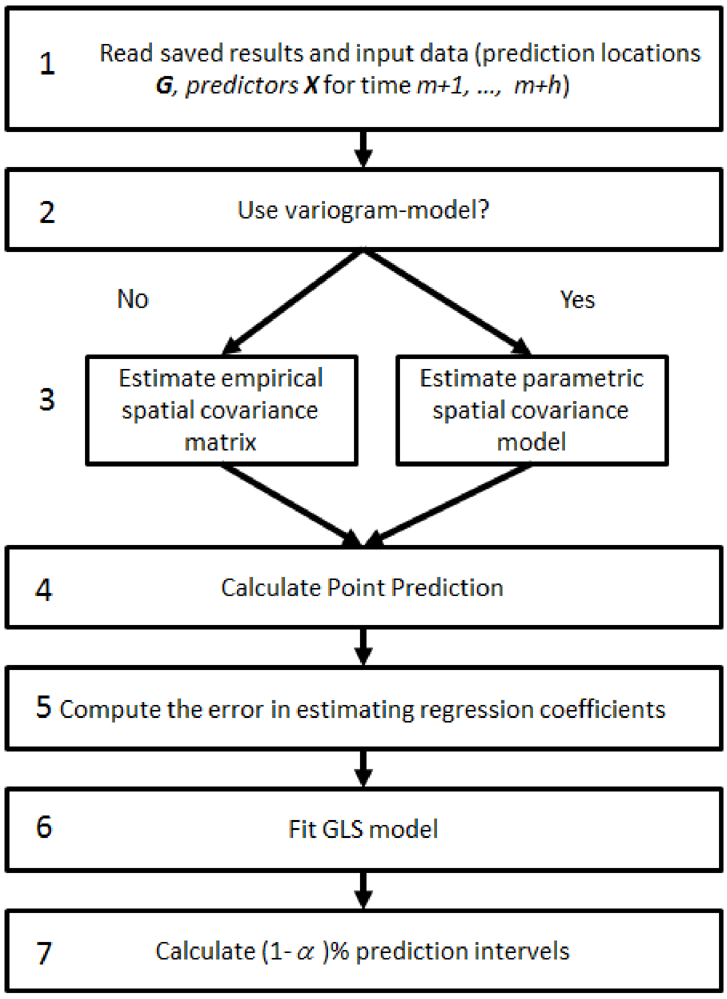

1 Read saved results and input data (prediction locations G, predictors X for time m+1, ..., m+h)
2 Use variogram-model?
No
Yes
3
Estimate empirical spatial covariance matrix
Estimate parametric spatial covariance model
4 Calculate Point Prediction
5 Compute the error in estimating regression coefficients
6 Fit GLS model
7 Calculate (1- α)% prediction intervels

Figure 2. Flowchart of algorithm steps for model prediction

4.1 Point prediction

Step 1: Construct the $N \times n$ spatial covariance matrix to capture the spatial dependence between prediction grids $\boldsymbol{g} \in \boldsymbol{G}$ and original sample locations $\boldsymbol{s}$.

- If variogram-based spatial covariance matrix

$$V_S(\boldsymbol{g}) = V(\epsilon_t(\boldsymbol{g})) = \sigma^2 + \tau^2 \tag{29}$$

and

$$\boldsymbol{C}_S(\boldsymbol{G}) = \{Cov(\epsilon_t(\boldsymbol{g}_i), \epsilon_t(\boldsymbol{s}_j); \hat{\psi})\}_{i=1,\dots,N; j=1,\dots,n} \tag{30}$$

according to (14) for all locations $\boldsymbol{g}$ (whether locations were included in the model build or not).

- If EOF-based spatial covariance function is used:

For locations g_i that are included in the original sample locations s, $Cov_{EOF}(\epsilon_t(g_i), \epsilon_t(s))$ is equal to the row corresponding to location g_i in the empirical covariance matrix Σ_S and $V_S(g_i)$ is equal to the empirical variance at that location, i.e., the diagonal element of Σ_S corresponding to that location.

For locations g_i that were not included in the model build, calculate the spatial covariance in the following way:

(a) Perform eigendecomposition on the empirical covariance matrix

$$\boldsymbol{S} = \boldsymbol{\Phi\Lambda\Phi}'$$

where $\boldsymbol{\Phi} = (\phi_1, \dots, \phi_n)$ with $\Phi_k = (\phi_k(s_1), \dots, \phi_k(s_n))'$ is the $n \times n$ matrix of eigenvectors and $\boldsymbol{\Lambda} = \text{diag}(\lambda_1, \dots, \lambda_n)$ is the $n \times n$ matrix of eigenvalues.

(b) Apply inverse distance weighting (IDW) (Shepard 1968) to interpolate eigenvectors to locations with no observations.

$$\phi_k(\boldsymbol{g}) = \sum_{i=1}^{n} \frac{w_i(\boldsymbol{g})\phi_k(\boldsymbol{s}_i)}{\sum_{j=1}^{n} w_j(\boldsymbol{g})}, k = 1, \dots, n$$

where

$$w_i(\boldsymbol{g}) = \frac{1}{\text{dist}(\boldsymbol{g}, \boldsymbol{s}_i)^\rho}$$

is an Inverse Distance Weighting (IDW) function with $\rho \leq d$ for d-dimensional space and $\text{dist}(\boldsymbol{g}, \boldsymbol{s}_i)$ may be any distance function. As a default value, use Euclidean distance with $\rho = 2$ and $\text{dist}(\boldsymbol{g}, \boldsymbol{s}_i)^2 = (\boldsymbol{g} - \boldsymbol{s}_i)'(\boldsymbol{g} - \boldsymbol{s}_i)$.

(c) The EOF-based spatial variance-covariance functions are

$$V_S(\boldsymbol{g}) = V(\epsilon_t(\boldsymbol{g})) = \sum_{k=1}^{n} \lambda_n \phi_k^2(\boldsymbol{g}) \tag{31}$$

and

$$Cov\ (\epsilon_t(\boldsymbol{g}_i), \epsilon_t(\boldsymbol{s}_j)) = \sum_{k=1}^{n} \lambda_n \phi_k(\boldsymbol{g}_i)\phi_k(\boldsymbol{s}_j) \tag{32}$$

and the corresponding $N \times n$ spatial covariance matrix

$$\boldsymbol{C}_S(\boldsymbol{G}) = \{Cov_{EOF}\ (\epsilon_t(\boldsymbol{g}_i), \epsilon_t(\boldsymbol{s}_j))\}_{i=1,\dots,N; j=1,\dots,n} \tag{33}$$

Note that under the EOF model, we allow for space-varying variances.

Step 2: Spatial interpolation to prediction locations g for the most recent L time units, $Z_{m-L+1}, \dots, Z_m$

$$\widehat{\boldsymbol{Z}}_{m-l+1}(\boldsymbol{G}) = \boldsymbol{C}_S(\boldsymbol{G})\boldsymbol{\Sigma}_S^{-1}\boldsymbol{Z}_{m-l+1} = \boldsymbol{C}_S(\boldsymbol{G})\breve{\boldsymbol{Z}}_{m-l+1}, l = 1, \dots, L \tag{34}$$

where $\widehat{\boldsymbol{Z}}_{m-l+1}(\boldsymbol{G})$ is a vector of length N.

Step 3: Iteratively forecast for future time $\text{m} + 1, \dots, \text{m} + \text{H}$ at prediction locations $\boldsymbol{G}$.

$$\widehat{\boldsymbol{Z}}_{m+1}(\boldsymbol{G}) = \hat{\alpha}_1\widehat{\boldsymbol{Z}}_m(\boldsymbol{G}) + \dots + \hat{\alpha}_L\widehat{\boldsymbol{Z}}_{m-L+1}(\boldsymbol{G}) \tag{35}$$

$$\widehat{\boldsymbol{Z}}_{m+2}(\boldsymbol{G}) = \hat{\alpha}_1\widehat{\boldsymbol{Z}}_{m+1}(\boldsymbol{G}) + \dots + \hat{\alpha}_L\widehat{\boldsymbol{Z}}_{m-L+2}(\boldsymbol{G}) \tag{36}$$

$$\widehat{\boldsymbol{Z}}_{m+H}(\boldsymbol{G}) = \hat{\alpha}_1\widehat{\boldsymbol{Z}}_{m+H-1}(\boldsymbol{G}) + \dots + \hat{\alpha}_L\widehat{\boldsymbol{Z}}_{m+H-L}(\boldsymbol{G}) \tag{37}$$

where $\widehat{\boldsymbol{Z}}_{m+H}(\boldsymbol{G}), h = 1, \dots, H$ are vectors of length N.

Step 4: Incorporate predicted systematic effect

$$\widehat{\boldsymbol{Y}}_{m+H}(\boldsymbol{G}) = \widehat{\boldsymbol{Z}}_{m+H}(\boldsymbol{G}) + X_{m+h}(\boldsymbol{G})\widehat{\boldsymbol{\beta}}, \qquad h = 1, \dots, H \tag{38}$$

where $\widehat{\boldsymbol{Y}}_{m+H}(\boldsymbol{G}), h = 1, \dots, H$ are vectors of length N.

4.2 Prediction intervals

Under the assumption of Gaussian Process and known variance components, the prediction error $\hat{Y}_{m+H}(\boldsymbol{g}_i) - Y_{m+h}(\boldsymbol{g}_i)$ comes from two sources:

- The prediction error that would be incurred even if regression coefficients $\boldsymbol{\beta}$ were known.
- The error in estimating regression coefficients $\boldsymbol{\beta}$

  The variance of prediction error is thus

$$V[\hat{Y}_{m+H}(\boldsymbol{g}_i) - Y_{m+h}(\boldsymbol{g}_i)]$$
$$= (\boldsymbol{X'}_{m+h}(\boldsymbol{g}_i) - \boldsymbol{C'}_{m+h}(\boldsymbol{g}_i)\boldsymbol{\Sigma}^{-1}\boldsymbol{X}_{impute})\boldsymbol{V}(\widehat{\boldsymbol{\beta}})(\boldsymbol{X'}_{m+h}(\boldsymbol{g}_i) - \boldsymbol{C'}_{m+h}(\boldsymbol{g}_i)\boldsymbol{\Sigma}^{-1}\boldsymbol{X}_{impute})' \tag{39}$$
$$+\boldsymbol{V}_{m+h}(\boldsymbol{g}_i) - \boldsymbol{C'}_{m+h}(\boldsymbol{g}_i)\boldsymbol{\Sigma}^{-1}\boldsymbol{C}_{m+h}(\boldsymbol{g}_i) \tag{40}$$

  Expression (39) arises from the variance expression for universal kriging, while (40) is the variance of a predicted random effect with known variance of the random effects (McCulloch et al. 2008, p.171).

- $C_{m+h}(g_i) = C_T(m+h) \otimes C_S(g_i)$ is the covariance vector of length nm between the prediction $Y_{m+h}(g_i)$ and measurements $Y_1(s), \dots, Y_m(s)$. Note that $C_T(m+h) = \{\gamma_T(m+h-t)\}_{t=1,\dots,m}$ is the AR(L) covariance vector of length m and $C_S(g_i) = \{Cov\ (Y_t(g_i), Y_t(s_j))\}_{j=1,\dots,n}$ is the spatial covariance vector of length n.
- The nm × nm covariance matrix Σ is defined as to $\Sigma = \Sigma_T \otimes \Sigma_S$ and $\Sigma_T = \{\gamma_T|t-t'|\}_{t,t'=1,\dots,m}$. Note that Σ_S is a quantity stored after the model build step.
- $V_{m+h}(g_i) = V(Y_{m+h}(g_i)) = \gamma_T(0)V_S(g_i)$ is the variance of $Y_{m+h}(g_i)$.
- Note that expressions (39) and (40) are not computed explicitly, but instead are implemented as described in the following.

Computational process:

Step 1: Compute the error in estimating regression coefficients β in (39).

For $l = 1, \dots, L$, interpolate $\boldsymbol{X}$ to prediction locations $\boldsymbol{g}$ for the most recent L time units

$$\boldsymbol{P}_{m+1-l}(\boldsymbol{g}_i) = \boldsymbol{X'}_{m+1-l,impute}\boldsymbol{\Sigma}_{\boldsymbol{S}}^{-1}\boldsymbol{C}_S(\boldsymbol{g}_i) = \check{\boldsymbol{X}}'_{m+1-l,impute}\boldsymbol{C}_S(\boldsymbol{g}_i) \tag{41}$$

where $\boldsymbol{P}_{m+1-l}(\boldsymbol{g}_i)$ is a vector of dimension $D \times 1$. Define

$$\widehat{\boldsymbol{X}}_{m+h-l}(\boldsymbol{g}_i) = \begin{cases}\boldsymbol{P}_{m+h-l}(\boldsymbol{g}_i), & if\ h-l \le 0;\\ \boldsymbol{X}_{m+h-l}(\boldsymbol{g}_i), & \text{otherwise.}\end{cases} \tag{42}$$

For $t = m - L + 1, \dots, m\ (h \le l)$, we only have X at sample locations s, so $\hat{X}_t(g_i) = P_t(g_i)$, the interpolated values from $X_t(s)$; for $t > m$ (or $h > l$), we already input X at prediction locations g, so there is no need to interpolate and $\hat{X}_t(g_i) = X_t(g_i)$.

Then, for $h = 1, \dots, H$, recursively compute the $D \times 1$ vectors $W_{m+h}(g_i)$

$$W_{m+h}(g_i) = X_{m+h}(g_i) + \sum_{l=1}^{L} \hat{\alpha}_l\left(\widehat{W}_{m+h-l}(g_i) - \hat{X}_{m+h-l}(g_i)\right) \tag{43}$$

where

$$\widehat{W}_{m+h-l}(g_i) = \begin{cases}0, & \text{if } h-l \le 0; (7)\\ W_{m+h-l}(g_i), & \text{otherwise.}\end{cases} \tag{44}$$

The prediction error in estimating β, that is, expression (39) is thus

$$W'_{m+h}(g_i)V(\hat{\beta})W_{m+h}(g_i) \tag{45}$$

where $V(\hat{\beta})$ is computed in (25).

Step 2: Compute the prediction error that would be incurred if regression coefficients β were known, i.e., equation (40).

- Compute $C_T(m+h)$ by AR(L) autocovariance function $\gamma_T(k)$ (McLeod 1975).

First, compute $\gamma_T(0), \dots, \gamma_T(L)$ by solving a linear system $AX = b$,

$$\begin{pmatrix} 1 & -\hat{\alpha}_1 & -\hat{\alpha}_2 & \cdots & -\hat{\alpha}_{L-1} & -\hat{\alpha}_L \\ -\hat{\alpha}_1 & 1-\hat{\alpha}_2 & -\hat{\alpha}_3 & \cdots & -\hat{\alpha}_L & 0 \\ -\hat{\alpha}_2 & -(\hat{\alpha}_1+\hat{\alpha}_3) & 1-\hat{\alpha}_4 & \cdots & 0 & 0 \\ -\hat{\alpha}_3 & -(\hat{\alpha}_2+\hat{\alpha}_4) & -(\hat{\alpha}_1+\hat{\alpha}_5) & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & & \vdots \\ -\hat{\alpha}_{L-2} & -(\hat{\alpha}_{L-3}+\hat{\alpha}_{L-1}) & -(\hat{\alpha}_{L-4}+\hat{\alpha}_L) & \cdots & 0 & 0 \\ -\hat{\alpha}_{L-1} & -(\hat{\alpha}_{L-2}+\hat{\alpha}_L) & -\hat{\alpha}_{L-3} & \cdots & 1 & 0 \\ -\hat{\alpha}_L & -\hat{\alpha}_{L-1} & -\hat{\alpha}_{L-2} & \cdots & -\hat{\alpha}_1 & 1 \end{pmatrix} \begin{pmatrix} \gamma_T(0) \\ \gamma_T(1) \\ \gamma_T(2) \\ \gamma_T(3) \\ \vdots \\ \gamma_T(L-2) \\ \gamma_T(L-1) \\ \gamma_T(L) \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{46}$$

Note that the first element of the vector on the right hand side (the variance of the measurement error) is fixed to be one, to account for the normalization through the spatial variance-covariance structure.

For $k = L + 1, \dots, m + H - 1$, recursively compute

$$\gamma_T(k) = \hat{\alpha}_1\gamma_T(k-1) + \cdots + \hat{\alpha}_L\gamma_T(k-L) \tag{47}$$

Remark: To construct the $(L+1) \times (L+1)$ matrix A,

$$A_{ij} = \begin{cases}-[\alpha_{i-1}], & j = 1; i = 1, \dots, L+1\\ -[\alpha_{i-j}] - [\alpha_{i+j-2}], & j = 2, \dots, L+1; i = 1, \dots, L+1.\end{cases} \tag{48}$$

where

$$[\alpha_k] = \begin{cases} -1, & k = 0; \\ 0, & k < 0 \text{ or } k > L; \\ \hat{\alpha}_k, & 0 < k \le L. \end{cases} \qquad (49)$$

- Compute the approximated factorization of Σ_T^{-1} such that $R'R \approx \Sigma_T^{-1}$, where R is a $(m-L) \times m$ matrix (follows from Cholesky or Gram-Schmidt orthogonalization, see for example Fuller 1975):

$$R = \begin{pmatrix} -\hat{\alpha}_L & \dots & -\hat{\alpha}_1 & 1 & 0 & 0 & \dots & \vdots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots & \vdots & \vdots \\ \dots & 0 & -\hat{\alpha}_L & \dots & -\hat{\alpha}_1 & 1 & 0 & 0 \\ \dots & \dots & 0 & -\hat{\alpha}_L & \dots & -\hat{\alpha}_1 & 1 & 0 \\ \dots & \dots & \dots & 0 & -\hat{\alpha}_L & \dots & -\hat{\alpha}_1 & 1 \end{pmatrix} \qquad (50)$$

- Compute the value of expression (40):

$$\gamma_T(0)V_S(g_i) - (C'_T(m+h) \otimes C'_S(g_i))(R'R \otimes H_S^{-1'} H_S^{-1})(C_T(m+h) \otimes C_S(g_i)) \qquad (51)$$

where $C'_S(g_i)$ is a the row of $C_S(G)$ corresponding to location g_i.

Step 3: The $(1-\alpha\%)$ prediction interval is

$$\hat{Y}_{m+h}(g_i) \pm t_{nm-q-D^*,\alpha/2}\sqrt{V[\hat{Y}_{m+h}(g_i) - Y_{m+h}(g_i)]} \qquad (55)$$

where $V[\hat{Y}_{m+h}(g_i) - Y_{m+h}(g_i)]$ is the sum of equations (39) and (40) as computed in expressions (45) and (51), respectively. $t_{nm-q-D,\alpha/2}$ is defined as $P(X \le t_{nm-q-D^*,\alpha/2}) = 1 - \alpha/2$ where X follows t-distribution with degree freedom $nm - q - D^*$. The default value for α is 0.05.

As final output from the prediction step, point prediction, variances of point predictions and prediction interval (lower and upper bounds) are issued for each specified (location, time).

We remark that to perform what-if-analysis, a set of ***X*** variables under the new settings need to be provided. Then we re-run the prediction algorithm described in Section 4 to obtain prediction results under adjusted settings.

References

[1] Brockwell, P., Davis, R.A. (2002), *Introduction to Time Series and Forecasting*, Second Edition, New York: Springer.

[2] Cohen, A., Johnes, R. (1969), “Regression on a Random Field”, *Journal of the American Statistical Association*, 64 (328), 1172-1182.

[3] Cressie, N. (1993), *Statistics for Spatial Data*, Revised Edition, Wiley-Interscience.

[4] Creutin, J.D., Obled, C. (1982), “Objective Analyses and Mapping Techniques for Rainfall Fields: an Objective Comparison”, *Water Resources Research*, 18(2), 413-431.

[5] Fuller, W.A. (1975), *Introduction to Statistical Time Series*, John Wiley & Sonse, New York, New York.

[6] Johnson S. (1967), “Hierarchical Clustering Schemes”, *Psychometrika*, 32(3), 241-254.

[7] McCulloch, C.E., Searle, S.R., Neuhaus, J.M. (2008), *Generalized, Linear and Mixed Models*, Second Edition, John Wiley & Sons, Hoboken, New Jersey.

[8] McLeod, I. (1975), “Derivation of the Theoretical Autocovariance Function of Autoregressive-Moving Average Time Series”, *Applied Statistics*, 24(2), 255-256.

[9] Shepard, D. (1968), “A two-dimensional interpolation function for irregularly-spaced data”, *Proceedings of the 1968 ACM National Conference*, 517-524.

[10] Wasserman S. (1994), *Social network analysis: Methods and applications*. Cambridge university press.

SPCHART Algorithms

Nine types of Shewhart control charts can be created. In this chapter, the charts are grouped into five sections:

- X-Bar and R Charts
- X-Bar and s Charts
- Individual and Moving Range Charts
- p and np Charts
- u and c Charts

For each type of control chart, the process, the center line, and the control limits (upper and lower) are described.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 95-1
Notation

Notation	Description
σ	Population standard deviation for measurements X
A	Number of sigmas specified by the user, $0 \leq A \leq 9$
K	Number of subgroups
n_i	Number of units (samples) for subgroup i
N	Total sample size, equal to $n_1 + \ldots + n_K$
x_{ij}	Measurement (observation) for the jth unit (sample) of subgroup i
x_i	Mean of measurements for subgroup i, $\overline{x}_i = \left(\sum_{i=1}^{n_i} x_{ij}\right)/n_i$
S_i	Sample standard deviation for subgroup i, $S_i^2 = \left(\sum_{i=1}^{n_i} (x_{ij} - \overline{x}_i)^2\right)/(n_i - 1)$
R_i	Sample range for subgroup i, $R_i = \max(x_{i1}, \ldots, x_{in_i}) - \min(x_{i1}, \ldots, x_{in_i})$
LCL	Lower Control Limit
UCL	Upper Control Limit

Weight

Weights can be used when the data organization is Cases are units.

- Each value for weight must be a positive integer.
- Cases with either non-positive or fractional weights are dropped.
- When weight is in effect, n_i is a weighted sum for all the units in subgroup i and x_i and x are weighted means.

X-Bar and R Charts

When X-Bar and R charts are paired, the sample range statistic *R* is used to construct the control limits for the X-Bar chart.

Note: Subgroups whose sample sizes are less than the specified minimum value are dropped.

Equal Sample Sizes

Assume that $n_i = n$ for $i = 1, \ldots, K$. The process for the X-Bar chart is $\{x_i : i = 1, \ldots, K\}$. The center line for an X-Bar chart is the grand mean statistic:

$$\overline{x} = \tfrac{1}{K}\sum_{i=1}^{K} \overline{x}_i$$

and the control limits are

$$\text{LCL} = \overline{x} - A\overline{R}/(d_2(n)\sqrt{n})$$
$$\text{UCL} = \overline{x} + A\overline{R}/(d_2(n)\sqrt{n})$$

where

$$\overline{R} = \tfrac{1}{K}\sum_{i=1}^{K} R_i$$

is the mean range statistic. The process for an R chart is $\{R_i : i = 1, \ldots, K\}$. The center line for an R chart is $\overline{R}$ and the control limits are

$$\text{LCL} = \max\left(\overline{R}(1 - Ad_3(n)/d_2(n)), 0\right)$$
$$\text{UCL} = \overline{R}(1 + Ad_3(n)/d_2(n))$$

The auxiliary functions are

$$d_2(n) = \int_{-\infty}^{\infty} (1 - (1 - \Phi(x))^n - (\Phi(x))^n)dx$$

$$d_3(n) = \left(2\int_{-\infty}^{\infty}\int_{-\infty}^{x} (1 - (\Phi(x))^n - (1 - \Phi(y))^n + (\Phi(x) - \Phi(y))^n)dydx - (d_2(n))^2\right)^{\frac{1}{2}}$$

$$\Phi(z) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du$$

Unequal Sample Sizes

The processes for X-Bar and R charts are the same as described in the section "Equal Sample Sizes" above. The center line for an X-Bar chart is the grand mean statistic (numerically identical to that in the section "Equal Sample Sizes"):

$$\overline{x} = \tfrac{1}{N}\sum_{i=1}^{K} n_i \overline{x}_i$$

and the control limits for subgroup *i* are

$$\text{LCL} = \overline{x} - A\hat{\sigma}/\sqrt{n_i}$$
$$\text{UCL} = \overline{x} + A\hat{\sigma}/\sqrt{n_i}$$

The center line for an R chart for subgroup *i* is $R_i = \hat{\sigma} d_2(n_i)$ for $i = 1, \ldots, K$ where

$$\hat{\sigma} = \frac{1}{K}\sum_{i=1}^{K} (R_i/d_2(n_i))$$

and the control limits for subgroup *i* are

$$\text{LCL} = \max(R_i - A\hat{\sigma} d_3(n_i), 0)$$
$$\text{UCL} = R_i + A\hat{\sigma} d_3(n_i)$$

X-Bar and s Charts

When X-Bar and s charts are paired, the sample standard deviation is used to construct the control limits for the X-Bar chart.

Equal Sample Sizes

Assume $n_i = n$. The process for the X-Bar chart is $\{x_i \ : i = 1, \ldots, K\}$. The center line for an X-Bar chart is $\overline{x}$ and the control limits are

$$\text{LCL} = \overline{x} - A\overline{S}/(c_4(n)\sqrt{n})$$
$$\text{UCL} = \overline{x} + A\overline{S}/(c_4(n)\sqrt{n})$$

The process for an s chart is $\{S_i \ : i = 1, \ldots, K\}$. The center line for an s chart is

$$\overline{S} = \frac{1}{K}\sum_{i=1}^{K} S_i$$

and the control limits are

$$\text{LCL} = \max\left(\overline{S}\left(1 - A\sqrt{\left(1 - (c_4(n))^2\right)}/c_4(n)\right), 0\right)$$
$$\text{UCL} = \overline{S}\left(1 + A\sqrt{\left(1 - (c_4(n))^2\right)}/c_4(n)\right)$$

The auxiliary function is

$$c_4(n) = \sqrt{\frac{2}{n-1}}\frac{\Gamma(n/2)}{\Gamma((n-1)/2)}$$

where Γ(.) is the complete Gamma function.

Note: When $n \geq 25$, $c_4(n)\sqrt{n}$ can be approximated by $\sqrt{n - 0.5}$, $\sqrt{\left(1 - (c_4(n))^2\right)}/c_4(n)$ can be approximated by $1/\sqrt{2n - 2.5}$, and $c_4(n)$ can be approximated by $\sqrt{(4n - 5)/(4n - 3)}$.

Unequal Sample Sizes

The processes for X-Bar and s charts are the same as the processes in the section “Equal Sample Sizes” above. The center line for an X-Bar chart is $\overline{x}$ and the control limits are

$$\text{LCL} = \overline{x} - A\hat{\sigma}/\sqrt{n_i}$$
$$\text{UCL} = \overline{x} + A\hat{\sigma}/\sqrt{n_i}$$

or

$$\text{LCL} = \overline{x} - AS_i/\left(c_4(n_i)\sqrt{n_i}\right)$$
$$\text{UCL} = \overline{x} + AS_i/\left(c_4(n_i)\sqrt{n_i}\right)$$

where

$$\hat{\sigma} = \tfrac{1}{K}\sum_{i=1}^{K} S_i/c_4(n_i)$$

However, the center line for an s chart for subgroup *i* is $S_i = \hat{\sigma}c_4(n_i)$ for *i*=1,...,*K* and the control limits are

$$\text{LCL} = \max\left(S_i - A\hat{\sigma}\sqrt{\left(1-(c_4(n_i))^2\right)}, 0\right)$$
$$\text{UCL} = S_i + A\hat{\sigma}\sqrt{\left(1-(c_4(n_i))^2\right)}$$

or

$$\text{LCL} = \max\left(S_i - AS_i\sqrt{\left(1-(c_4(n_i))^2\right)}/c_4(n_i), 0\right)$$
$$\text{UCL} = S_i + AS_i\sqrt{\left(1-(c_4(n_i))^2\right)}/c_4(n_i)$$

Individual and Moving Range Charts

When a weight variable is specified, each unit of the process is expanded to multiple units based on the case weight associated with this particular unit. The span (specified by the user) is associated with the expanded process. If the span is greater than *N* (the total number of units of the expanded process), an error message is displayed and neither an Individual nor a Moving Range chart is generated.

Since each subgroup has only one unit, the process for an Individual chart is $\{y_i : i = 1, \ldots, N\}$ where y_i is the *i*th unit of the expanded process. For a span of length *m*, the moving ranges, are

$$R_i = \begin{cases} \max\left(y_{i-m+1}, \ldots, y_i\right) - \min\left(y_{i-m+1}, \ldots, y_i\right) & \text{if } i = m, \ldots, N \\ \text{SYSMIS} & \text{if } i = 1, \ldots, m-1 \end{cases}$$

The average moving range is

$$\overline{R} = \frac{1}{(N-m+1)}\sum_{m}^{N} R_i$$

The center line for an Individual chart is $\overline{x}$ and the control limits for an Individual chart are

$$\text{LCL} = \overline{x} - A\overline{R}/d_2(m)$$
$$\text{UCL} = \overline{x} + A\overline{R}/d_2(m)$$

The process for a moving range chart is {R_i, *i* = *m*,..., *N*}. The center line for a moving range chart is $\overline{R}$. The control limits for a moving range chart are

$$\text{LCL} = \max\left(\overline{R}(1 - Ad_3(m)/d_2(m)), 0\right)$$
$$\text{UCL} = \overline{R}(1 + Ad_3(m)/d_2(m))$$

p and np Charts

The data for p and np charts are attribute data. Each measurement x_{ij} is either 0 or 1, where 1 indicates a non-conforming measurement. Therefore,

$$x_{i+} = \sum_{j=1}^{n_i} x_{ij}$$

is the count of non-conforming units for subgroup *i*. When a weight variable is specified, x_{i+} is a weighted sum of non-conforming units. If the data are aggregated and the value of the count variable is greater than the total number of units for any subgroup, this subgroup is dropped.

Equal Sample Sizes

Assume $n_i = n$. The process for a p chart is $\{p_i : i = 1, \ldots, K\}$ where $p_i = x_{i+}/n$. The center line for a p chart is

$$\overline{p} = \tfrac{1}{K}\sum_{i=1}^{K} p_i$$

and the control limits are

$$\text{LCL} = \max\left(\overline{p} - A\sqrt{(\overline{p}(1-\overline{p}))/n}, 0\right)$$
$$\text{UCL} = \min\left(\overline{p} + A\sqrt{(\overline{p}(1-\overline{p}))/n}, 1\right)$$

The process for an np chart is $\{x_{i+} : i = 1, \ldots, K\}$. The center line for an np chart is

$$\overline{x} = \tfrac{1}{K}\sum_{i=1}^{K} x_{i+}$$

and the control limits are

$$\text{LCL} = \max\left(\overline{x} - A\sqrt{n\overline{p}(1-\overline{p})}, 0\right)$$
$$\text{UCL} = \min\left(\overline{x} + A\sqrt{n\overline{p}(1-\overline{p})}, n\right)$$

Unequal Sample Sizes

The process for a p chart is $\{p_i : i = 1, \ldots, K\}$ where $p_i = x_{i+}/n_i$. The center line for a p chart is

$$\overline{p} = \tfrac{1}{N}\sum_{i=1}^{K} x_{i+} = \frac{1}{N}\sum_{i=1}^{K} n_i p_i$$

and the control limits for subgroup *i* are

$$\begin{aligned}\text{LCL} &= \max\left(\overline{p} - A\sqrt{(\overline{p}(1-\overline{p}))/n_i}, 0\right)\\ \text{UCL} &= \min\left(\overline{p} + A\sqrt{(\overline{p}(1-\overline{p}))/n_i}, 1\right)\end{aligned}$$

The process for an np chart is $\{x_{i+} : i = 1, \ldots, K\}$. However, the center line for an np chart for subgroup *i* is $n_i\overline{p}$. The control limits for subgroup *i* are

$$\begin{aligned}\text{LCL} &= \max\left(n_i\overline{p} - A\sqrt{(n_i\overline{p}(1-\overline{p}))}, 0\right)\\ \text{UCL} &= \min\left(n_i\overline{p} + A\sqrt{(n_i\overline{p}(1-\overline{p}))}, n_i\right)\end{aligned}$$

Note: A warning message is issued when an np chart is requested for subgroups of unequal sample sizes.

u and c Charts

Measurements x_{ij} show the number of defects for the *j*th unit for subgroup *i*. Hence,

$$x_{i+} = \sum_{j=1}^{n_i} x_{ij}$$

is the total number of defects for subgroup *i*. When a weight variable is used, x_{i+} is a weighted sum of defects.

Equal Sample Size

Assume $n_i = n$. The process for a u chart is $\{u_i : i = 1, \ldots, K\}$ where $u_i = x_{i+}/n$. The center line for a u chart is

$$\overline{u} = \tfrac{1}{K}\sum_{i=1}^{K} u_i$$

and the control limits are

$$\begin{aligned}\text{LCL} &= \max\left(\overline{u} - A\sqrt{\overline{u}/n}, 0\right)\\ \text{UCL} &= \overline{u} + A\sqrt{\overline{u}/n}\end{aligned}$$

The process for a c chart is $\{x_{i+} : i = 1, \ldots, K\}$. The center line for a c chart is

$$\overline{c} = \tfrac{1}{K}\sum_{i=1}^{K} x_{i+}$$

and the control limits for a c chart are

$$\text{LCL} = \max\left(\overline{c} - A\sqrt{\overline{c}}, 0\right)$$
$$\text{UCL} = \overline{c} + A\sqrt{\overline{c}}$$

Unequal Sample Size

The process for a u chart is $\{u_i : i = 1, \ldots, K\}$ where $u_i = x_{i+}/n$. The center line for a u chart is

$$\overline{u} = \frac{1}{N}\sum_{i=1}^{K} x_{i+}$$

and the control limits are

$$\text{LCL} = \max\left(\overline{u} - A\sqrt{\overline{u}/n_i}, 0\right)$$
$$\text{UCL} = \overline{u} + A\sqrt{\overline{u}/n_i}$$

The process for a c chart is $\{x_{i+} : i = 1, \ldots, K\}$. The center line for subgroup *i* is $n_i\overline{u}$ and the control limits are

$$\text{LCL} = \max\left(n_i\overline{u} - A\sqrt{n_i\overline{u}}, 0\right)$$
$$\text{UCL} = n_i\overline{u} + A\sqrt{n_i\overline{u}}$$

Note: A warning message is issued when a c chart is requested for subgroups of unequal sample sizes.

Statistics

This section discusses the capability and performance statistics that can be requested through SPCHART, and uses the following notation.

Table 95-2
Notation

Notation	**Description**
$\overline{x}$	the total sample mean.
s	the total sample/process standard deviation.
$\hat{\sigma}$	the estimated sigma in the Process Capability Indices.
μ_o	the nominal or the target value, given by the user.
LSL	the lower specification limit, given by the user.
USL	the upper specification limit, given by the user.

Assumptions

- The process is in control. ($\overline{x}$ and *s* are finitely estimated.)
- The measured variable is normally distributed.

Prerequisites

- For the Process Capability Indices except *CpK* and the Process Performance Indices except *PpK*, both LSL and USL must be specified by the user, satisfying LSL < USL. For *CpK* and *PpK*, at least one of LSL and USL must be specified by the user.
- A target value μ_o such that LSL $\leq \mu_o \leq$ USL must be given by the user for *CpM* and *PpM* to be computed.

Process Capability Indices

The estimated capability sigma $\hat{\sigma}$ may be computed in one of four ways.

(1). If it is to be based on the sample within-subgroup variance,

$$\hat{\sigma} = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{n_i}\left(x_{ij} - \overline{x}_i\right)^2 / \sum_{i=1}^{k}\left(n_i - 1\right)}$$

(2). If it is to be based on the mean range,

$$\hat{\sigma} = \frac{\sum_{i=1}^{k} \frac{R_i}{d_2\left(n_i\right)}}{k}$$

where $d_2\left(n_i\right) = \int_{-\infty}^{\infty} 1 - \left(1 - \Phi\left(x\right)\right)^{n_i} - \left(\Phi\left(x\right)\right)^{n_i} dx$ with $\Phi\left(x\right) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du$

Note that n_i may or may not be equal for different subgroups. If they are all equal, we may write

$$\hat{\sigma} = \frac{\overline{R}}{d_2(n_i)}$$

where $\overline{R} = \Sigma_{i=1}^{k} R_i / k$, the mean range.

(3). If it is to be based on the mean standard deviation,

$$\hat{\sigma} = \frac{\sum_{i=1}^{k} \frac{s_i}{c_4\left(n_i\right)}}{k}$$

where $c_4\left(n_i\right) = \sqrt{\frac{2}{n_i - 1}} \frac{\Gamma(n_i/2)}{\Gamma((n_i-1)/2)}$, with the complete Gamma function Γ().

Note that n_i may or may not be equal for different subgroups. If they are all equal, we may write

$$\hat{\sigma} = \frac{\overline{s}}{c_4(n_i)}$$

(4). If it is to be based on the mean moving range,

$$\hat{\sigma} = \frac{\sum_{i=m}^{n} \frac{MR_i}{d_2\left(m\right)}}{n-m+1} = \frac{\overline{MR}}{d_2(m)}$$

where

$$MR_i = \begin{cases} \max\left(y_{i-m+1}, \ldots, y_i\right) - \min\left(y_{i-m+1}, \ldots, y_i\right), \text{if} i = m, \ldots, n \\ \text{sysmis, if} i = 1, \ldots, m-1 \end{cases}$$

and *n* is the total sample size, *m* is the user-given length of span, and MR_i is the *i*th moving range for the data.

All of the capability indices, except *K*, require $\hat{\sigma}$, and in order to define them, we must have $\hat{\sigma}$ >0.

CP: Capability of the process

$$CP = \frac{\text{USL}-\text{LSL}}{6\hat{\sigma}}$$

CpL: The distance between the process mean and the lower specification limit scaled by capability sigma

$$CpL = \frac{\overline{x}-\text{LSL}}{3\hat{\sigma}}$$

CpU: The distance between the process mean and the upper specification limit scaled by capability sigma

$$CpU = \frac{\text{USL}-\overline{x}}{3\hat{\sigma}}$$

K: The deviation of the process mean from the midpoint of the specification limits

$$K = \frac{2\left|(\text{USL}+\text{LSL})/2-\overline{x}\right|}{\text{USL}-\text{LSL}}$$

Note this is computed independently of the estimated capability sigma, so it does not need to be greater than 0 or even specified.

CpK: Capability of process related to both dispersion and centeredness

$$CpK = \min\left(CpU, CpL\right)$$

If only one specification limit is provided, we compute and report a unilateral *CpK* instead of taking the minimum.

CR: The reciprocal of CP

$$CR = \frac{1}{CP}$$

CpM: An index relating capability sigma and the difference between the process mean and the target value

$$CpM = \frac{\text{USL}-\text{LSL}}{6\sqrt{\hat{\sigma}^2+\left(\overline{x}-\mu_o\right)^2}}$$

μ_o must be given by the user.

Z-lower (Cap): The number of capability sigmas between the process mean and the lower specification limit

$$CZ_L = \frac{\overline{x} - \mathrm{LSL}}{\hat{\sigma}}$$

Z-upper (Cap): The number of capability sigmas between the process mean and the upper specification limit

$$CZ_U = \frac{\mathrm{USL} - \overline{x}}{\hat{\sigma}}$$

Z-min (Cap): The minimum number of capability sigmas between the process mean and the specification limits

$$CZmin = \min(CZ_U, CZ_L)$$

Note that unlike *CpK*, this index is undefined unless both specification limits are given and valid.

Z-max (Cap): The maximum number of capability sigmas between the process mean and the specification limits

$$CZ\max = \max(CZ_U, CZ_L)$$

Note that unlike *CpK*, this index is undefined unless both specification limits are given and valid.

The estimated percentage outside the specification limits (Cap)

$$(1 - \Phi(CZ_U) + \Phi(-CZ_L)) \times 100\%$$

where $\Phi(x)$ is the cumulative distribution function of the standard normal distribution.

Process Performance Indices

The estimated performance sigma is always the process standard deviation *s*. None of the indices in this chapter is defined unless *s*>0.

PP: Performance of the process

$$PP = \frac{\mathrm{USL} - \mathrm{LSL}}{6s}$$

PpL: The distance between the process mean and the lower specification limit scaled by process standard deviation

$$PpL = \frac{\overline{x} - \mathrm{LSL}}{3s}$$

PpU: The distance between the process mean and the upper specification limit scaled by process standard deviation

$$PpU = \frac{\text{USL} - \overline{x}}{3s}$$

PpK: Performance of process related to both dispersion and centeredness

$$PpK = \min\left(PpU, PpL\right)$$

If only one specification limit is provided, we compute and report a unilateral *PpK* instead of taking the minimum.

PR: The reciprocal of PP

$$PR = \frac{1}{PP}$$

PpM: An index relating process variance and the difference between the process mean and the target value

$$PpM = \frac{\text{USL} - \text{LSL}}{6\sqrt{s^2 + (\overline{x} - \mu_o)^2}}$$

μ_o must be given by the user.

Z-lower (Perf): The number of standard deviations between the process mean and the lower specification limit

$$PZ_L = \frac{\overline{x} - \text{LSL}}{s}$$

Z-upper (Perf): The number of standard deviations between the process mean and the upper specification limit

$$PZ_U = \frac{\text{USL} - \overline{x}}{s}$$

Z-min (Perf): The minimum number of standard deviations between the process mean and the specification limits

$$PZmin = \min\left(PZ_U, PZ_L\right)$$

Note that unlike *PpK*, this index is undefined unless both specification limits are given and valid.

Z-max (Perf): The maximum number of standard deviations between the process mean and the specification limits

$$PZ\max = \max\left(PZ_U, PZ_L\right)$$

Note that unlike *PpK*, this index is undefined unless both specification limits are given and valid.

The estimated percentage outside the specification limits (Perf)

$$(1 - \Phi(PZ_U) + \Phi(-PZ_L)) \times 100\%$$

where $\Phi(x)$ is the cumulative distribution function of the standard normal distribution.

Measure(s) for Assessing Normality: The observed percentage outside the specification limits

This is the percentage of individual observations in the process which lie outside the specification limits. A point is defined as outside the specification limits when its value is greater than the USL or is less than the LSL.

References

Committee E-11 on Quality and Statistics, . 1990. *ASTM Special Technical Publication (STP) 15D: Manual on presentation of data and control chart analysis*, 6 ed. Philadelphia: American Society for Testing and Materials.

Grant, E. L., and R. S. Leavenworth. 1980. *Statistical quality control*, 5 ed. New York: McGraw-Hill.

Harter, H. L. 1969. *Order statistics and their use in testing and estimation, Volumes 1 and 2*. Washington, D.C.: U.S. Government Printing Office: Aerospace Research Laboratories, United States Air Force.

SPECTRA Algorithms

SPECTRA plots the periodogram and spectral density function estimates for one or more series.

Univariate Series

For all *t*, the series X_t can be represented by

$$X_t = a_0^x + \sum_{K=1}^{q} \left(a_K^x \cos 2\pi f_K(t-1) + b_K^x \sin 2\pi f_K(t-1)\right)$$

where

$$t = 1, 2, \ldots, N$$

$$a_0^x = \overline{X}, \overline{X} = \sum_{t=1}^{N} X_t / N$$

$$a_K^x = \tfrac{2}{N}\left[\sum_{t=1}^{N} \left(X_t \cos 2\pi f_K(t-1)\right)\right]$$

$$b_K^x = \tfrac{2}{N}\left[\sum_{t=1}^{N} \left(X_t \sin 2\pi f_K(t-1)\right)\right]$$

$$f_K = K/N$$

$$q = \begin{cases} N/2, & \text{if } N \text{ is even} \\ (N-1)/2, & \text{if } N \text{ is odd} \end{cases}$$

The following statistics are calculated:

Frequency

$$f_K = K/N, K = 1, \ldots, q$$

Period

$$1/f_K = N/K, K = 1, \ldots, q$$

Fourier Cosine Coefficient

$$a_K^x, K = 1, \ldots, q$$

Fourier Sine Coefficient

$$b_K^x = (a_K^x - ib_K^x)(a_K^x + ib_K^x)$$

Periodogram

$$l_K^x = \left[(a_K^x)^2 + (b_K^x)^2\right]N/2, K = 1, \ldots, q$$

spectral density estimate

$$s_K^x = \sum_{j=p}^{p} w_j l_{K+j}^x,\ \text{where } 2p+1 = m \text{ (number of spans)}$$

and

$$l_{-K}^x = l_K^x, K = 1, \ldots, q$$
$$l_0^x = l_1^x$$
$$l_K^x = l_{N+1-K} \text{ for } K > q$$

$w_{-p}, w_{-p+1}, \ldots, w_0, w_1, \ldots, w_p$ are the periodogram weights defined by different data windows.

Bivariate Series

For the bivariate series X_t and Y_t

$$X_t = a_0^x + \sum_{K=1}^{q} (a_K^x \cos 2\pi f_K t + b_K^x \sin 2\pi f_K t) \quad t = 1, \ldots, N$$

$$Y_t = a_0^y + \sum_{K=1}^{q} \left(a_K^y \cos 2\pi f_K t + b_K^y \sin 2\pi f_K t\right)$$

Cross-Periodogram of X and Y

$$l_K^{xy} = \frac{N}{2}\left(a_K^x - ib_K^x\right)\left(a_K^y + ib_K^y\right)$$
$$= \frac{N}{2}\left\{\left(a_K^x a_K^y + b_K^x b_K^y\right) + i\left(a_K^x b_K^y - b_K^x a_K^y\right)\right\}$$

Real

$$(RC)_K = \frac{N}{2}\left(a_K^x a_K^y + b_K^x b_K^y\right)$$

Imaginary

$$(IC)_K = \frac{N}{2}\left(a_K^x b_K^y - b_K^x a_K^y\right)$$

Cospectral Density Estimate

$$C_K = \sum_{j=-p}^{p} w_j (RC)_{K+j}$$

Quadrature Spectrum Estimate

$$Q_K = \sum_{j=-p}^{p} w_j (IC)_{K+j}$$

Cross-amplitude Values

$$A_K = \left(Q_K^2 + C_K^2\right)^{1/2}$$

Squared Coherency Values

$$K_K = \frac{A_K^2}{s_K^x s_K^y}$$

Gain Values

$$G_K = \begin{cases} A_K / s_K^x & \text{(gain of } Y_t \text{ over } X_t \text{ at } f_K) \\ A_K / s_K^y & \text{(gain of } X_t \text{ over } Y_t \text{ at } f_K) \end{cases}$$

Phase Spectrum Estimate

$$\Psi_K = \begin{cases} \tan^{-1}(Q_K/C_K) & \text{if } \begin{matrix} Q_K > 0, C_K > 0 \\ Q_K < 0, C_K > 0 \end{matrix} \\ \tan^{-1}(Q_K/C_K) + \pi & \text{if } Q_K > 0, C_K < 0 \\ \tan^{-1}(Q_K/C_K) - \pi & \text{if } Q_K < 0, C_K < 0 \end{cases}$$

Data Windows

The following spectral windows can be specified. Each formula defines the upper half of the window. The lower half is symmetric with the upper half. In all formulas, *p* is the integer part of the number of spans divided by 2. To be concise, the formulas are expressed in terms of the Fejer kernel:

$$F_q(\theta) = \begin{cases} q & \theta = 0,\ \pm 2\pi,\ \pm 4\pi,\ \ldots \\ \frac{1}{q}\left(\frac{\sin(q\theta/2)}{\sin(\theta/2)}\right)^2 & \text{otherwise} \end{cases}$$

and the Dirichlet kernel:

$$D_q(\theta) = \begin{cases} 2q+1 & \theta = 0,\ \pm 2\pi,\ \pm 4\pi,\ \ldots \\ \frac{\sin((2q+1)\theta/2)}{\sin(\theta/2)} & \text{otherwise} \end{cases}$$

where q is any positive real number.

HAMMING

Tukey-Hamming window. The weights are

$$W_k = 0.54D_p(2\pi f_k) + 0.23D_p\left(2\pi f_k + \frac{\pi}{p}\right) + 0.23D_p\left(2\pi f_k - \frac{\pi}{p}\right)$$

for $k = 0, \ldots, p$.

TUKEY

Tukey-Hanning window. The weights are

$$W_k = 0.5D_p(2\pi f_k) + 0.25D_p\left(2\pi f_k + \frac{\pi}{p}\right) + 0.25D_p\left(2\pi f_k - \frac{\pi}{p}\right)$$

for $k = 0, \ldots, p$.

PARZEN

Parzen window. The weights are

$$W_k = \frac{1}{p}(2 + \cos(2\pi f_k))\left(F_{p/2}(2\pi f_k)\right)^2$$

for $k = 0, \ldots, p$.

BARTLETT

Bartlett window. The weights are

$$W_k = F_p(2\pi f_k)$$

for $k = 0, \ldots, p$.

DANIELL UNIT

Daniell window or rectangular window. The weights are

$$W_k = 1$$

for $k = 0, \ldots, p$.

NONE

No smoothing. If NONE is specified, the spectral density estimate is the same as the periodogram. It is also the case when the number of span is 1.

$$W_{-p}, \ldots, W_0, \ldots, W_p$$

User-specified weights. If the number of weights is odd, the middle weight is applied to the periodogram value being smoothed and the weights on either side are applied to preceding and following values. If the number of weights are even (it is assumed that W_p is not supplied), the weight after the middle applies to the periodogram value being smoothed. It is required that the weight W_0 must be positive.

References

Bloomfield, P. 1976. *Fourier analysis of time series*. New York: John Wiley and Sons.

Fuller, W. A. 1976. *Introduction to statistical time series*. New York: John Wiley and Sons.

SURVIVAL Algorithms

Although life table analysis may be useful in many differing situations and disciplines, for simplicity, the usual survival-time-to-death terminology will be used here.

Notation

The following notation is used throughout this chapter unless otherwise stated:

X_j	Time from starting event to terminal event or censoring for case *j*
w_j	Weight for case *j*
k	Total number of intervals
t_i	Beginning time for *i*th interval
h_i	Width of interval *i*
c_i	Sum of weights of cases censored in interval *i*
d_i	Sum of weights of cases experiencing the terminal event in interval *i*

Construction of Life Table (Gehan, 1975)

The following sections detail the construction of the life table.

Computation of Intervals

The widths of the intervals for the actuarial calculations must be defined by the user. In addition to the last interval specified, an additional interval is automatically created to take care of any times exceeding the last. If the upper limits are not in ascending order, a message is printed and the procedure stops. If the interval width does not divide the time range into an integral number of intervals, a warning is printed and the interval width is reset so that the number of intervals will be the nearest integer to that resulting from the user specification.

Count of Events and Censoring

For each case, the interval i into which the survival time falls is determined.

$$t_i \leq X_j < t_{i+1}$$

If X_j exceeds t_k, the starting time for the last interval, it is included in the last interval. The status code is examined to determine whether the observed time is time to event or time to censoring. If it is time to censoring, that is, the terminal event did not occur, c_i is incremented by the case weight. If it is time to event, d_i is incremented by the case weight.

Calculation of Survival Functions

For each interval, the following are calculated.

Number Alive at the Beginning

$$l_i = l_{i-1} - c_{i-1} - d_{i-1}$$

where l_1 is the sum of weights of all cases in the table.

Number Exposed to Risk of an Event

$$r_i = l_i - c_i/2$$

Proportion Terminating

$$q_i = \frac{d_i}{r_i}$$

Proportion Surviving

$$p_i = 1 - q_i$$

Cumulative Proportion Surviving at End of Interval

$$P_i = P_{i-1} p_i$$

where

$$P_0 = 1$$

Probability Density Function

$$f_i = \frac{P_{i-1} - P_i}{h_i}$$

Hazard Rate

$$\lambda_i = \frac{2q_i}{h_i(1+p_i)}$$

Standard Error of Probability Surviving

$$se(P_i) = P_i \sqrt{\sum_{j=1}^{i} q_j/(r_j p_j)}$$

Standard Error of Probability Density

$$se(f_i) = \frac{P_i q_i}{h_i}\sqrt{\sum_{j=1}^{i-1} q_j/(r_j p_j) + p_i/(r_i q_i)}$$

For the first interval

$$se(f_1) = \frac{P_1 q_1}{h_1}\sqrt{\frac{p_1}{r_1 q_1}}$$

Standard Error of the Hazard Rate

$$se(\lambda_i) = \sqrt{\frac{\lambda_i^2}{r_i q_i}\left\{1 - \left(\frac{\lambda_i h_i}{2}\right)^2\right\}}$$

If $q_i = 0$, the standard error for interval *i* is set to 0.

Median Survival Time

If $P_k > 0.5$ the value printed for median survival time is

$$t_k +$$

Otherwise, let *i* be the interval for which $P_i < 0.5$ and $P_{i-1} \geq 0.5$. The estimate of the median survival time is then

$$Md = (t_i) + \frac{h_{i-1}(P_{i-1} - 0.5)}{P_{i-1} - P_i}$$

Comparison of Survival Distributions

The survival times from the groups to be compared are jointly sorted into ascending order. If survival times are equal, the uncensored is taken to be less than the censored. When approximate comparisons are done, they are based on the lifetables, with the beginning of the interval determining the length of survival for cases censored or experiencing the event in that interval.

Notation

The following notation is used throughout this section unless otherwise stated:

N	Number of cases
$X_{(k)}$	Survival time for case *k*, where times are sorted into ascending order so that case 1 has the shortest time and case *N* the longest
w_k	Weight for case *k*
g	Number of nonempty groups in the comparison
W_j	Sum of weights of cases in group *j*
W_c	Sum of weights of censored cases

W_u	Sum of weights of uncensored cases
W	Sum of weights of all cases

Computations

For each case the following are computed:

- ULE_k: Sum of weights of uncensored cases with survival times less than or equal to that of case k.
- CLE_k: Same as above, but for censored cases.
- UE_k: Sum of weights of uncensored cases with survival times equal to that of case k.
- CE_k: Same as above, but for censored cases.

The score for case *k* is:

$$S_k = \begin{cases} ULE_k & \text{if } X_k \text{ is censored} \\ A_1 - A_2 - A_3 & \text{if } X_k \text{ is uncensored} \end{cases}$$

where

$A_1 = ULE_k - UE_k$ uncensored cases surviving shorter than case k
$A_2 = W_c - CLE_k + CE_k$ censored cases surviving longer than or equal to case k
$A_3 = W_u - ULE_k$ uncensored cases surviving longer than case k

Test Statistic and Significance (Wilcoxon (Gehan))

The test statistic is

$$D = \frac{(W-1)B}{T}$$

where

$$B = \sum_{j=1}^{g} SS_j^2 / W_j$$

SS_j = the sum of scores of cases in group j

$$T = \sum_{i=1}^{N} S_i^2$$

Under the hypothesis that the groups are samples from the same survival distribution, *D* is asymptotically distributed as a chi square with (g−1) degrees of freedom.

References

Gehan, E. A. 1975. Statistical methods for survival time studies. In: *Cancer Therapy: Prognostic Factors and Criteria*, M. J. Staquet, ed. New York: Raven Press, 7–35.

Lee, E., and M. Desu. 1972. A computer program for comparing k samples with right censored data. *Computer Programs in Biomedicine*, 2, 315–321.

T Test Algorithms

The T Test procedure compares the means of two groups or (one-sample) compares the means of a group with a constant.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 98-1
Notation

Notation	Description
X_{ki}	Value for *i*th case of group *k*
w_{ki}	Weight for *i*th case of group *k*
n_k	Number of cases in group *k*
W_k	Sum of weights of cases in group *k*

Basic Statistics

The following statistics are computed.

Means

$$\overline{X}_k = \frac{\sum_{i=1}^{n_k} X_{ki} w_{ki}}{W_k} \quad k = 1, 2$$

Variances

$$S_k^2 = \frac{\sum_{i=1}^{n_k} X_{ki}^2 w_{ki} - \left(\sum_{i=1}^{n_k} X_{ki} w_{ki}\right)^2 / W_k}{(W_k - 1)}$$

Standard Errors of the Mean

$$SEM_k = S_k / \sqrt{W_k}$$

Differences of the Means for Groups 1 and 2

$$D = \overline{X}_1 - \overline{X}_2$$

Unpooled (Separate Variance) Standard Error of the Difference

$$S_D = \sqrt{\frac{S_1^2}{W_1} + \frac{S_2^2}{W_2}}$$

The 95% confidence interval for mean difference is

$$D \pm t_{df'} S_D$$

where $t_{df'}$ is the upper 2.5% critical value for the *t* distribution with df' degrees of freedom.

Pooled Standard Error of the Difference

$$S'_D = S_p \sqrt{\frac{1}{W_1} + \frac{1}{W_2}}$$

where the pooled estimate of the variance is

$$S_p^2 = \frac{(W_1-1)S_1^2 + (W_2-1)S_2^2}{W_1+W_2-2}$$

The 95% confidence interval for mean difference

$$D \pm t_{df} S'_D$$

where *df* is defined in the following.

The t Statistics for Equality of Means

Separate Variance

$$t = D/S_D$$
$$df' = \frac{1}{Z_1+Z_2}$$

where

$$Z_k = \left(\frac{S_k^2/W_k}{S_1^2/W_1+S_2^2/W_2}\right)^2 / (W_k - 1)$$

Pooled Variance

$$t' = D/S'_D$$
$$df = W_1 + W_2 - 2$$

The two-tailed significance levels are obtained from the *t* distribution separately for each of the computer *t* values.

The Test for Equality of Variances

The Levene statistic is used and defined as

$$L = \frac{(W-2)\sum_{k=1}^{2} W_k\left(\overline{Z}_k - \overline{Z}\right)^2}{\sum_{k=1}^{2}\sum_{i=1}^{n_k} w_{ki}\left(Z_{ki} - \overline{Z}_k\right)^2}$$

where

$$Z_{ki} = \left|X_{ki} - \overline{X}_k\right|$$

$$\overline{Z}_k = \frac{\sum_{i=1}^{n_k} w_{ki} Z_{ki}}{W_k}$$

$$\overline{Z} = \frac{\sum_{k=1}^{2} W_k \overline{Z}_k}{W_1 + W_2}$$

The t Test for Paired Samples

The following notation is used throughout this section unless otherwise stated:

Table 98-2
Notation

Notation	**Description**
X_i	Value of variable *X* for case *i*
Y_i	Value of variable *Y* for case *i*
w_i	Weight for case *i*
W	Sum of the weights
N	Number of cases

Means

$$\overline{X} = \sum_{i=1}^{N} w_i X_i / W$$

$$\overline{Y} = \sum_{i=1}^{N} w_i Y_i / W$$

Variances

$$S_X^2 = \frac{\sum_{i=1}^{N} w_i X_i^2 - \left(\sum_{i=1}^{N} w_i X_i\right)^2 / W}{W-1}$$

Similarly for S_Y^2.

Covariance between X and Y

$$S_{XY} = \frac{1}{W-1}\left(\sum_{k=1}^{N} X_k Y_k w_k - \left(\sum_{k=1}^{N} w_k X_k\right)\left(\sum_{k=1}^{N} w_k Y_k\right)/W\right)$$

Difference of the Means

$$D = \overline{X} - \overline{Y}$$

Standard Error of the Difference

$$S_D = \sqrt{(S_X^2 + S_Y^2 - 2S_{XY})/W}$$

t statistic for Equality of Means

$$t = D/S_D$$

with (W–1) degrees of freedom. A two-tailed significance level is printed.

95% Confidence Interval for Mean Difference

$$D \pm t_{W-1} S_D$$

Correlation Coefficient between X and Y

$$r = \frac{S_{XY}}{S_X S_Y}$$

The two-tailed significance level is based on

$$t = r\sqrt{\frac{W-2}{1-r^2}}$$

with (W–2) degrees of freedom.

One-Sample t Test

The following notation is used throughout this chapter unless otherwise stated:

Table 98-3
Notation

Notation	Description
N	Number of cases
X_i	Value of variable X for case i
w_i	Weight for case i
v	Test value

Mean

$$\overline{X} = \frac{1}{W}\sum_{i=1}^{N} w_i X_i$$

where $W = \sum_{i=1}^{N} w_i$ is the sum of the weights.

Variance

$$S_X^2 = \frac{1}{W-1}\sum_{i=1}^{N} w_i\left(X_i - \overline{X}\right)^2$$

Standard Deviation

$$S_X = \sqrt{S_X^2}$$

Standard Error of the Mean

$$S_{\overline{X}} = S_X/\sqrt{W}$$

Mean Difference

$$D = \overline{X} - v$$

The t value

$$t = D/S_{\overline{X}}$$

with (W–1) degrees of freedom. A two-tailed significance level is printed.

100p% Confidence Interval for the Mean Difference

$$\mathrm{CI} = D \pm t_{W-1,(p+1)/2} S_{\overline{X}}$$

where $t_{W-1,(p+1)/2}$ is the $100((p+1)/2)$% percentile of a Student's *t* distribution with (W–1) degrees of freedom.

References

Blalock, H. M. 1972. *Social statistics*. New York: McGraw-Hill.

Temporal Causal Modeling Algorithms

1. Introduction

Forecasting and prediction are important tasks in real world applications that involve decision making. In such applications, it is important to go beyond discovering statistical correlations and unravel the key variables that influence the behaviors of other variables using an algebraic approach. Many real world data, such as stock price data, are temporal in nature; that is, the values of a set of variables depend on the values of another set of variables at several time points in the past. Temporal causal modeling, or TCM, refers to a suite of methods that attempt to discover key temporal relationships in time series data. This chapter describes a particular method to discover temporal relationships using a combination of Granger causality and regression algorithms for variable selection. Although this treatment strives to be self-contained, a minimal set of papers describing the design principles behind the method can be found in [Lozano et al., 2011, Lozano et al., 2009, Arnold et al., 2007][1].

The rest of the chapter is organized as follows. Section 2 lays the groundwork for the TCM algorithm (notation and brief history) and explains the greedy orthogonal matching pursuit (GOMP) [Lozano et al., 2011] algorithm that is used. Section 3 describes the techniques used to fit and forecast time series and compute approximated forecasting intervals. Section 4 describes scenario analysis, which refers to a capability of the TCM product to "play-out" the repercussions of artificially setting the value of a time series. Section 5 describes the detection of outliers, and Section 6 discusses how potential causes for outliers can be established using root cause analysis.

2. Model

Introduced by Clive Granger [Granger, 1980], Granger causality in time series is based on the intuition that a cause should necessarily precede its effect, and that if time series a causally affects time series b, then the past values of a should be useful in predicting the future values of a. More specifically, time series a is said to "Granger cause" time series b if the accuracy of regressing for b in terms of past values of both a and b is statistically significantly better than regressing just with past values of b. If the time series have T time points and are denoted by $\{a_t\}_{t=1}^{T}$ and $\{b_t\}_{t=1}^{T}$, then the following regressions are performed:

$$b_t \approx \sum_{j=1}^{L} \alpha_j \quad _{t-j} + \sum_{j=1}^{L} \beta_j \, b_{t-j} \tag{1}$$

$$b_t \approx \sum_{j=1}^{L} \beta_j \, b_{t-j} \tag{2}$$

Here L is the number of lags; that is, the value of b at time t can only be determined by values of other time series at times $\{t-1, t-2, \dots, t-L\}$. If Equation (1) is statistically more significant (using some test for significance) than Equation (2), then a is deemed to Granger cause b.

[1] The methods described in this chapter are particularly useful for under-determined systems, where the number of time series (n) far exceeds the number of samples (m); that is $n >> m$. Although these methods function for both over-determined ($m >> n$) and fully-determined ($n == m$) systems, there are other approaches to pursue for such systems.

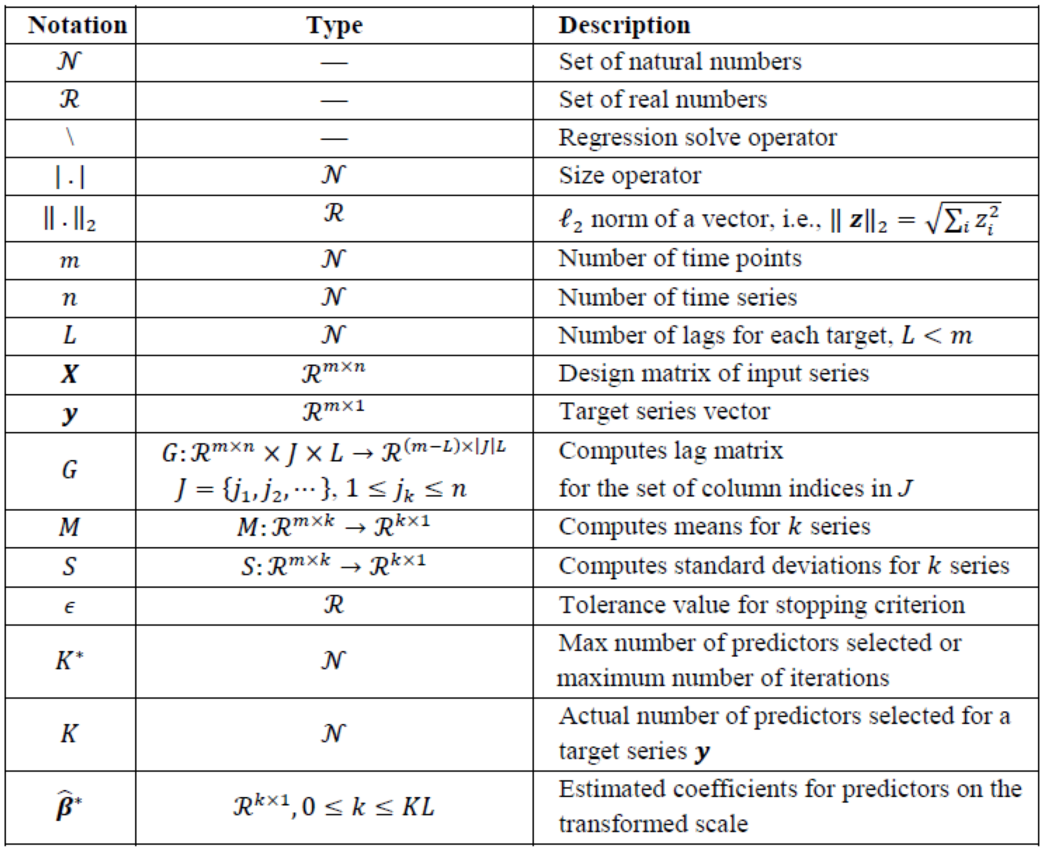
2.1 Graphical Granger Modeling

The classical definition of Granger causality is defined for a pair of time series. In the real world, we are interested in finding not one, but *all* the significant time series that influence the target time series. In order to accomplish this, we use group greedy (ℓ_0) regression algorithms with variable selection (see Section 2.3). An important feature of our TCM algorithm is that it groups influencer/predictor variables; that is, we are interested in predicting whether time series as a whole $-\{\alpha_{t-1}, \alpha_{t-2}, \ldots, \alpha_{t-L}\}$ has influence over time series b. Such grouping is a more natural interpretation of causality and also helps sparsify the solution set. For example, without such grouping we may select the time-lagged series α_{t-2} to model b_t but not select any other value of a, which increases the number of choices for variable selection L-fold, where L is the number of lags that is allowed.

2.2 Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 1: Notation



Notation	Type	Description
$\mathcal{N}$	—	Set of natural numbers
$\mathcal{R}$	—	Set of real numbers
$\backslash$	—	Regression solve operator
$\lvert . \rvert$	$\mathcal{N}$	Size operator
$\lVert . \rVert_2$	$\mathcal{R}$	ℓ_2 norm of a vector, i.e., $\lVert \boldsymbol{z} \rVert_2 = \sqrt{\sum_i z_i^2}$
m	$\mathcal{N}$	Number of time points
n	$\mathcal{N}$	Number of time series
L	$\mathcal{N}$	Number of lags for each target, $L < m$
$\boldsymbol{X}$	$\mathcal{R}^{m \times n}$	Design matrix of input series
$\boldsymbol{y}$	$\mathcal{R}^{m \times 1}$	Target series vector
G	$G: \mathcal{R}^{m \times n} \times J \times L \rightarrow \mathcal{R}^{(m-L) \times \lvert J \rvert L}$ $J = \{j_1, j_2, \cdots\}$, $1 \le j_k \le n$	Computes lag matrix for the set of column indices in J
M	$M: \mathcal{R}^{m \times k} \rightarrow \mathcal{R}^{k \times 1}$	Computes means for k series
S	$S: \mathcal{R}^{m \times k} \rightarrow \mathcal{R}^{k \times 1}$	Computes standard deviations for k series
ϵ	$\mathcal{R}$	Tolerance value for stopping criterion
K^*	$\mathcal{N}$	Max number of predictors selected or maximum number of iterations
K	$\mathcal{N}$	Actual number of predictors selected for a target series $\boldsymbol{y}$
$\widehat{\boldsymbol{\beta}}^*$	$\mathcal{R}^{k \times 1}, 0 \le k \le KL$	Estimated coefficients for predictors on the transformed scale

In this section, we introduce the algorithm that is used to construct the temporal causal model. The list of symbols used in the rest of this chapter is summarized in Table 1. Most of the symbols are self-explanatory; however, the function G, which stands for grouping, requires some additional explanation. G is a function that takes a matrix ($R^{m \times n}$), a set of column indices J, and a lag value L and constructs a lag matrix that has $(m - L)$ rows and ($|J|L$) columns. Basically, for every column index $j \in J, G$ constructs a $(m - L) \times L$ lag matrix by carefully unrolling the jth column of the input matrix. An example of G's action is shown below:

$$G\left(\boldsymbol{X} = \begin{bmatrix} a_1 & b_1 & c_1 & d_1 \\ a_2 & b_2 & c_2 & d_2 \\ a_3 & b_3 & c_3 & d_3 \\ a_4 & b_4 & c_4 & d_4 \\ a_5 & b_5 & c_5 & d_5 \end{bmatrix}, J = \{1\}, L = 2\right) \rightarrow \begin{bmatrix} a_2 & a_1 \\ a_3 & a_2 \\ a_4 & a_3 \end{bmatrix}$$

In this example, the input matrix $\boldsymbol{X} \in R^{5\times 4}$ has 4 time series ($n = 4$) and five time points per time series ($m = 5$). The lag matrix associated with the time series in column 1, when L (lag) is 2, is produced by invoking $G(\boldsymbol{X}, \{1\}, 2)$. Note that the lag matrix consists of the lag-1 vector $\boldsymbol{X}$ of as the first column, the lag-2 vector as the second column, up to the lag-L vector as the Lth column. Similarly, the functions (M,S) accept any input matrix and compute the mean and the standard deviation, respectively, of the matrix's columns. For purposes of numerical stability, and to increase interpretability during modeling, columns of the lagged matrix are both centered by the column means and scaled by the column standard deviations [2]. On the other hand, the target $\boldsymbol{y}$ is *only centered*. An example of mean centering and scaling for the lagged matrices is shown below:

$$\left(\begin{bmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \end{bmatrix}\right) \rightarrow \begin{bmatrix} \frac{a_1 - a_\mu}{a_\sigma} & \frac{b_1 - b_\mu}{b_\sigma} \\ \frac{a_2 - a_\mu}{a_\sigma} & \frac{b_2 - b_\mu}{b_\sigma} \\ \frac{a_3 - a_\mu}{a_\sigma} & \frac{b_3 - b_\mu}{b_\sigma} \end{bmatrix}$$

Here, $(\alpha_\mu, \alpha_\sigma)$ and (b_μ, b_σ) are the means and standard deviations of the first and the second columns, (a, b) respectively.

2.3 Group Orthogonal Matching Pursuit (GOMP)

Algorithm 1: GOMP

```
Input: $\boldsymbol{X}, \boldsymbol{y}, G, M, S, L, \epsilon, K^*, J_{sel}^0, J_{sel}^{\sim}$.
Output: $J_{sel}, \widehat{\boldsymbol{\beta}}^*$.
1  $\boldsymbol{X}_{aug}^0 = G(X, J_{sel}^0, L)$;
2  for $i \in [1, (m-L)]$ do $\boldsymbol{X}_{aug}^0(i,:) = \frac{\boldsymbol{X}_{aug}^0(i,:) - M(\boldsymbol{X}_{aug}^0)^T}{S(\boldsymbol{X}_{aug}^0)^T}$;
3  $\widehat{\boldsymbol{\beta}}^{*0} = \boldsymbol{X}_{aug}^0 \setminus (\boldsymbol{y} - M(\boldsymbol{y}))$;
4  $\boldsymbol{r}^0 = \boldsymbol{y} - M(\boldsymbol{y}) - \boldsymbol{X}_{aug}^0 \widehat{\boldsymbol{\beta}}^{*0}$;
5  if any redundant series are found, delete them in $J_{sel}^0$;
6  if ($|J_{sel}^0| \geq K^*$), then $J_{sel}^0 = J_{sel}^0(1{:}K^*)$, update $\widehat{\boldsymbol{\beta}}^{*0}$, return $J_{sel}^0$, $\widehat{\boldsymbol{\beta}}^{*0}$ and stop;
7  otherwise update $\widehat{\boldsymbol{\beta}}^{*0}$ and $\boldsymbol{r}^0$;
8  for $k \in 1, 2, 3 \ldots (K^* - |J_{sel}^0|)$ do
9      $j^k = argmin(\boldsymbol{X}, \boldsymbol{r}^{k-1}, G, M, S, L, \epsilon, J_{sel}^0, J_{sel}^{\sim})$;
10     if $j^k = -1$, return $J_{sel}^{(k-1)}$ and $\widehat{\boldsymbol{\beta}}^{*(k-1)}$ and stop;
11     $\boldsymbol{X}_{aug}^k = G(\boldsymbol{X}, J_{sel}^{k-1} \cup j^k, L)$;
```

[2] Although each column of the lagged matrix has a different mean and standard deviation, due to the structure of these columns, it is possible to compute the mean and the standard deviation of the time series itself and use those to center and scale the lagged columns.

```
12   for $i \in [1, (m-L)]$ do
13       $\boldsymbol{X}^k_{aug}(i,:) = \frac{\boldsymbol{X}^k_{aug}(i,:) - M(\boldsymbol{X}^k_{aug})^T}{S(\boldsymbol{X}^k_{aug})^T}$;
14   $\widehat{\boldsymbol{\beta}}^{*k} = \boldsymbol{X}^k_{aug} \setminus (\boldsymbol{y} - M(\boldsymbol{y}))$;
15   $\boldsymbol{r}^k = \boldsymbol{y} - M(\boldsymbol{y}) - \boldsymbol{X}^k_{aug}\widehat{\boldsymbol{\beta}}^{*k}$;
16   $J^k_{sel} = J^{k-1}_{sel} \cup j^k$;
17   if $\|\boldsymbol{r}^k\|_2 \le \epsilon$, break;
18 return $J^k_{sel}$, $\widehat{\boldsymbol{\beta}}^{*k}$.
```

We begin by describing Algorithm 1: GOMP, which will be used to establish causality of time-series data. This algorithm receives the variables $\boldsymbol{X}, \boldsymbol{y}, G, M, S, L, \epsilon, K^*$(described in Table 1) as input. Briefly, $\boldsymbol{y} \in \boldsymbol{R}^{(m-L)\times 1}$ is a target vector for which we want to establish the Granger causality (note that we have excluded the first L values of $\boldsymbol{y}$). In contrast, $\boldsymbol{X} \in \boldsymbol{R}^{m\times n}$ is the input *unlagged* time series data. L is the number of lags for each predictor in each target series, K^* is the maximum number of predictors to be selected per-target, and ϵ determines whether a new predictor needs to be added. In addition, G, M and S are grouping, centering, and scaling functions which have been described in Section 2.2. J^0_{sel} is the set of pre-selected predictor indices for $\boldsymbol{y}$, and always contains the lagged $\boldsymbol{y}$. $J^{\sim}_{sel}$ is the set of forbidden predictors, if any, for $\boldsymbol{y}$. If there are no forbidden predictors, then $J^{\sim}_{sel} = \emptyset$. Given these, the goal is to greedily find predictors that solve the system $\boldsymbol{X\beta} = \boldsymbol{y}$ subject to sparsity constraints.

The greedy algorithm approximates an ℓ_0–sparse solution by iteratively choosing the best predictor for addition at each iteration. We use superscripts to denote the iteration number in Algorithm 1. For example, J^0_{sel} represents the initial values of J_{sel} at the 0*th* iteration (before the actual iteration starts). The first part of the algorithm (lines 1 – 4) constructs and solves a linear system consisting of the predictors in J^0_{sel} to obtain $\boldsymbol{\beta}^{*0}$, the coefficient vector for predictors on the transformed scale. At the end of this first part, we have $\boldsymbol{r}^0$, the initial residual. Then check whether there are redundant predictor series in J^0_{sel}. If yes, then delete them. If the number of predictor series in the (updated) J^0_{sel} is equal to or larger than the maximum number of iterations (i.e., $|J^0_{sel}| \ge K^*$) then keep the first K^* predictor series in J^0_{sel}, update $\boldsymbol{\beta}^{*0}$, return J^0_{sel} and $\boldsymbol{\beta}^{*0}$, and stop the process (line 6); otherwise (i.e., $|J^0_{sel}| < K^*$), update $\boldsymbol{\beta}^{*0}$ and $\boldsymbol{r}^0$ (line 7) if any redundant predictor series were deleted. Then start the iterative process to add one predictor series at a time (line 8). The first step in predictor selection (line 9) consists of an **argmin** function that systematically goes over each eligible predictor and evaluates its goodness (see Algorithm 2). This step is the performance critical portion of the algorithm and can be searched in parallel. At the end of the step, j^k, the index corresponding to the best predictor is available. However, if no suitable predictor is found in the **argmin** function (i.e., $j^k = -1$), then return J^{k-1}_{sel} and $\boldsymbol{\beta}^{*(k-1)}$ and stop (line 10). The next part (lines 11 – 14) re-estimates the model coefficients by adding j^k to the model. Line 15 updates the residual, $\boldsymbol{r}^k$, for this model and line 16 adds j^k to the model. Finally, if the ℓ_2 norm of the current residuals is equal to or smaller than the tolerance value (i.e., ($||\boldsymbol{r}^k||_2 \le \epsilon$)), then the iterative process is terminated.

Note that if the tolerance ϵ is achieved by adding j^k, then no new iterations are required and the iterative process is terminated. Thus the actual number of predictors selected, K, can be less than the maximum number of iterations, (i.e., $K \le K^*$). However, if the tolerance ϵ is set very small, then it is highly unlikely that such a situation will happen.

Algorithm 2: argmin

Input: $\boldsymbol{X}, \boldsymbol{r}, G, M, S, L, \epsilon_2, K^*, J^0_{sel}, J^{\sim}_{sel}$.

Output: J_{sel}: Selected group index.

```
1  $cost = \|\boldsymbol{r}\|_2^2, j_{sel} = -1$;
2  for $j \in 1,2,3 \ldots n$ do
3    if $j \in J_{sel}$ || $j \in J_{sel}^{\sim}$ continue;
4    $\boldsymbol{X}_{G_j} = G(\boldsymbol{X}, j, L)$;
5    for $i \in [1,(m-L)]$ do $\boldsymbol{X}_{G_j}(i,:) = \frac{\boldsymbol{X}_{G_j}(i,:) - M(\boldsymbol{X}_{G_j})^T}{S(\boldsymbol{X}_{G_j})^T}$;
6    $\widehat{\boldsymbol{\beta}}_j = \boldsymbol{X}_{G_j} \setminus \boldsymbol{r}$;
7    $\boldsymbol{r}_j = \boldsymbol{r} - \left(\boldsymbol{X}_{G_j}\widehat{\boldsymbol{\beta}}_j\right)_{G_j}$;
8    if $\|\boldsymbol{r}_j\|_2^2 < (cost - \epsilon_2)$, then $(cost, j_{sel}) = \left(\|\boldsymbol{r}_j\|_2^2, j\right)$;
9  return $j_{sel}$.
```

The implementation of the **argmin** function (line 8, Algorithm 1) is shown in Algorithm 2. The algorithm first assigns the initial cost to be the square of the ℓ_2 norm of the current residuals, and the selected group index to be -1 (line 1). Then it loops over each series group, first checking if the time series being considered for addition (*j*) has already been added to the solution J_{sel} or if it is a forbidden predictor (line 3). If the current group (*j*) is not yet selected, the lagged transformed matrix corresponding to this time series ($\boldsymbol{X}_{G_j}$) is constructed using the *G*, *M* and *S* functions (lines 4 and 5). After grouping and transforming $\boldsymbol{X}_{G_j}$, the residual ($\boldsymbol{r}_j$) corresponding to the candidate time series *j* is computed by first regressing $\boldsymbol{r}$ on $\boldsymbol{X}_{G_j}$ (line 6), and then computing the residual (line 7). Finally, the current time series is selected as the leading candidate if the square of the ℓ_2 norm of its residual ($\boldsymbol{r}_j$) is lower than the previous estimate minus a threshold value, ϵ_2. Including such a threshold value prevents selecting an (almost) identical series.

The loop in Algorithm 2 (line 2) can be thought of as iterating over all candidate series. For each candidate series, the following computations are carried out: (1) a filter is applied in line 3 to ensure that it is a valid candidate; (2) lines 4 and 5 map the current candidate to the transformed matrix ($\boldsymbol{X}_{G_j}$) that represents the lag matrix to be used; (3) lines 6 and 7 evaluate the goodness of the current candidate by first solving a dense linear system and then computing the residual; (4) line 8 applies a predicate to check if the current candidate series is better than previously evaluated candidates. Notice that the predicate (line 8) is associative and commutative; therefore, Algorithm 2 can be parallelized by dividing the iteration space ([1,*n*]) into chunks and executing each chunk in parallel. To get the globally best group, it is sufficient to *reduce* the groups that were selected by each parallel instance in a tree-like fashion by applying the predicate in line 8.

2.4 Selecting *L*

Both Algorithms 1 and 2 accept *L* as an input parameter which can be specified by user. If *L* is not explicitly specified then the following heuristic approach can be used to determine *L* based on (# of time points) and *s* (periodicity or seasonal length):

(1) If $s > 1$ and $m \geq 4s$, then $L = \min(s, 20)$.

(2) If $s = 1$ or $m < 4s$, then $L = 5$.

2.5 AR(L) Model

Out of the n series in the data, some series may be used as predictors only, so no TCM models are built for them. However, if they are selected as predictors for some target series, then simple models need to be built for them in order to do forecasting. For example, suppose that time series 1 is a selected predictor for time series 2, but there is no model built for time series 1. While a model for time series 1 is not needed in order to forecast time series 2 at time (t + 1) (where t is the latest time in the data), forecasts for time (t + 2) require values of time series 1 for time (t + 1), which then requires a model for time series 1.

Hence, for each predictor-only series, a simple auto-regressive (AR) model is built using the same lag, L, as used for the target series. This model, called an AR(L) model, can be constructed using Algorithm 1 by specifying J_{sel}^{0} to be the target itself and setting the maximum number of predictors to be 1.

2.6 Post-estimation steps

Algorithm 1 selects the best predictors (time series) to model a target series $\boldsymbol{y}$. Without loss of generality, we assume that the model for $\boldsymbol{y}$ is $\boldsymbol{y} = \overline{\boldsymbol{y}} + \boldsymbol{X}_G^* \widehat{\boldsymbol{\beta}} + \boldsymbol{r} = \widehat{\boldsymbol{y}} + \boldsymbol{r}$, where $\boldsymbol{X}_G^*$ is the selected predictor series matrix with the lagged terms on the transformed scale, $\widehat{\boldsymbol{\beta}}^*$ is the estimated standardized coefficient vector, and $\boldsymbol{r} = \boldsymbol{y} - \widehat{\boldsymbol{y}}$ is the residual vector**.**

However, this is not the end of modeling. Several post processing steps are needed in order to complete the modeling process for $\boldsymbol{y}$. The steps include three parts: (1) coefficients and statistics inference; (2) tests of model effects; (3) model quality measures.

2.6.1 Coefficients and statistical inference

The results of Algorithm 1 include $\widehat{\boldsymbol{\beta}}^*$ and $(\boldsymbol{X}^{*T}\boldsymbol{X}^*)^-$ (by solving the linear system from Cholesky decomposition), where superscript T means the transpose of a matrix or vector, and $(\boldsymbol{z})^-$ is a generalized inverse of the $\boldsymbol{z}$ matrix. Based on these quantities, the first step is to compute coefficient estimates, their standard errors, and statistical inference on the original scale.

Table 2: Additional notation

Notation	Description
K	Actual number of predictors selected (including target itself) for $\boldsymbol{y}$, i.e., $K = \lvert J_{sel} \rvert$.
p	Number of coefficient estimates in $\widehat{\boldsymbol{\beta}}^*$, i.e., $p = K \times L$
$\boldsymbol{p^c}$	Number of non-redundant coefficient estimates in $\widehat{\boldsymbol{\beta}}^*$, $\boldsymbol{p^c} \leq p$
$\boldsymbol{X}_G^*$	Selected predictor series matrix with lagged terms on the transformed scale. This is an $(m-L) \times p$ matrix as $\boldsymbol{X}_G^* = \left[\boldsymbol{X}_{G_1}^*, \dots, \boldsymbol{X}_{G_K}^*\right]$ with $\boldsymbol{X}_{G_j}^* = G(\boldsymbol{X}^*, j, L) = \left[\boldsymbol{X}_{G_{j1}}^*, \dots, \boldsymbol{X}_{G_{jL}}^*\right]$ (an $(m-L) \times L$ matrix).
$\boldsymbol{X}_G$	Selected predictor series matrix on the original scale. This is an $(m-L) \times (p+1)$ matrix as $\boldsymbol{X}_G = \left[\boldsymbol{1}, \boldsymbol{X}_{G_1}, \dots, \boldsymbol{X}_{G_K}\right] = \left[\boldsymbol{1}, \boldsymbol{X}_{G_{11}}, \dots, \boldsymbol{X}_{G_{1L}}, \cdots, \boldsymbol{X}_{G_{K1}}, \dots, \boldsymbol{X}_{G_{KL}}\right]$, where $\boldsymbol{1}$ is a column vector of 1's corresponding to an intercept.
$\widehat{\boldsymbol{\beta}}$	Unstandardized coefficient estimates vector (corresponding to $\boldsymbol{X}_G$), which is a $(p+1) \times$

	1 vector. The first element, $\hat{\beta}_0$, is the intercept estimate.
$\hat{\sigma}^2$	Estimated variance of the model based on residuals.
$\boldsymbol{\Sigma}^*$	Covariance matrix of standardized coefficient estimates on the transformed scale, i.e., $\boldsymbol{\Sigma}^* = \hat{\sigma}^2(\boldsymbol{X}_G^{*\mathrm{T}}\boldsymbol{X}_G^*)^-$. The j^{th} diagonal element is $\hat{\sigma}^2_{\hat{\beta}_j^*}$ and its square root, $\hat{\sigma}_{\hat{\beta}_j^*}$, is the standard error of the j^{th} standardized coefficent estimate.
$\boldsymbol{\Sigma}$	Covariance matrix of unstandardized coefficient estimates on the original scale. The j^{th} diagonal element is $\hat{\sigma}^2_{\hat{\beta}_j}$ and its square root, $\hat{\sigma}_{\hat{\beta}_j}$, is the standard error of the j^{th} unstandardized coefficent estimate.
$\mathbf{M}$	Centering vector of $\boldsymbol{X}$, i.e., $\mathbf{M} = [\mathrm{M}_1, \dots, \mathrm{M}_p]^{\mathrm{T}}$, where $\mathrm{M}_j = M(\boldsymbol{X}_j)$ is the mean of $\boldsymbol{X}_j$.
$\mathbf{S}$	Scaling matrix of $\boldsymbol{X}$, i.e., $\mathbf{S} = \mathrm{diag}[\mathrm{S}_1, \dots, \mathrm{S}_p]$, where $\mathrm{S}_j = S(\boldsymbol{X}_j)$ is the standard deviation of $\boldsymbol{X}_j$.
$\boldsymbol{A}$	Transformation matrix of $\boldsymbol{X}$ to $\boldsymbol{X}^*$, i.e., $\boldsymbol{A} = \begin{bmatrix} -\mathbf{M}^{\mathrm{T}}\mathbf{S}^{-1} \\ \mathbf{S}^{-1} \end{bmatrix}$, which is a $(p+1) \times p$ vector. Note that $\boldsymbol{X}_G^* = \boldsymbol{X}_G\boldsymbol{A}$.

The relationship between $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\beta}}^*$ is $\hat{\boldsymbol{\beta}} = \boldsymbol{A}\hat{\boldsymbol{\beta}}^* + [\bar{y}, 0, \dots, 0]^{\mathrm{T}}$ and the relationship between $\boldsymbol{\Sigma}$ and $\boldsymbol{\Sigma}^*$ is $\boldsymbol{\Sigma} = \boldsymbol{A}\boldsymbol{\Sigma}^*\boldsymbol{A}^{\mathrm{T}}$. The relevant statistics are computed as follows:

- **Unstandardized coefficient estimates**

$$\hat{\beta}_j = \mathrm{S}_j^{-1}\hat{\beta}_j^*, \quad j = 1, \dots, p \tag{3}$$

$$\hat{\beta}_0 = \bar{\boldsymbol{y}} - \mathbf{M}^{\mathrm{T}}\mathbf{S}^{-1}\hat{\boldsymbol{\beta}}^* \tag{4}$$

- **Standard errors of unstandardized coefficient estimates**

$$\hat{\sigma}_{\hat{\beta}_j} = \frac{\hat{\sigma}_{\hat{\beta}_j^*}}{\mathrm{S}_j}, \quad j = 1, \dots, p \tag{5}$$

$$\hat{\sigma}_{\hat{\beta}_0} = sqrt\left(\left[\frac{\mathrm{M}_1}{\mathrm{S}_1}, \dots, \frac{\mathrm{M}_p}{\mathrm{S}_p}\right]\boldsymbol{\Sigma}^*\begin{bmatrix}\frac{\mathrm{M}_1}{\mathrm{S}_1} \\ \vdots \\ \frac{\mathrm{M}_p}{\mathrm{S}_p}\end{bmatrix}\right) \tag{6}$$

where $\boldsymbol{\Sigma}^* = \hat{\sigma}^2(\boldsymbol{X}_G^{*\mathrm{T}}\boldsymbol{X}_G^*)^-$ and $\hat{\sigma}^2 = SS_e/df_e$ with $SS_e = \|\boldsymbol{y} - \hat{\boldsymbol{y}}\|_2^2 = \sum_{t=1}^{t=(m-L)}(y_t - \hat{y}_t)^2$ and $df_e = m - L - p^c - 1$.

- **t-statistics for coefficient estimates**

$$t_j = \frac{\hat{\beta}_j}{\hat{\sigma}_{\hat{\beta}_j}}, \quad j = 0, 1, \dots p, \tag{7}$$

which follows an asymptotic t distribution with df_edegrees of freedom. Then the p-value is computed as

$$p_{t_j} = 2 \times \left(1 - prob\left(t_{df_e} \leq |t_j|\right)\right) \tag{8}$$

- **$100(1-\alpha)\%$ confidence internals**

$$\hat{\beta}_j \pm \hat{\sigma}_{\hat{\beta}_j} \times t_{\alpha/2,df_e} \tag{9}$$

where α is the significance level and $t_{\alpha/2,df_e}$ is the $100(1-\alpha/2)^{\text{th}}$ percentile of the t distribution with df_e degrees of freedom.

2.6.2 Tests of model effects

For each selected predictor series for $\boldsymbol{y}$, there are L lagged columns associated with it. The columns can be grouped together, considered as an effect, and tested with a null hypothesis of zero for all coefficients. This is similar to the test of a categorical effect with all dummy variables in a (generalized) linear model setting. Only type III tests are conducted here. For each selected predictor series $\boldsymbol{X}_{G,i}$, the type III test matrix $\boldsymbol{L}_i$ is constructed and H_0: $\boldsymbol{L}_i\boldsymbol{\beta} = \boldsymbol{0}$ is tested based on an F-statistic.

- **F-statistics for effects**

$$F_i = \frac{\hat{\boldsymbol{\beta}}^{\mathrm{T}}\boldsymbol{L}_i^{\mathrm{T}}\left(\boldsymbol{L}_i\boldsymbol{\Sigma}\boldsymbol{L}_i^{\mathrm{T}}\right)^{-1}\boldsymbol{L}_i\hat{\boldsymbol{\beta}}}{r_i} \tag{10}$$

where $r_i = rank(\boldsymbol{L}_i\boldsymbol{\Sigma}\boldsymbol{L}_i^T)$. The statistic follows an approximate F distribution with the numerator degrees of freedom r_i and the denominator degrees of freedom df_e. Then the p-value is computed as follows:

$$p_{F_i} = 1 - prob\left(F_{r_i,df_e} \leq |F_i|\right) \tag{11}$$

2.6.3 Model quality measures

In addition to statistical inferences, the goodness of the model can be evaluated. The following model quality measures are provided:

- **Root Mean Squared Error (RMSE)**

$$RMSE = \sqrt{MSE} = \sqrt{\frac{SS_e}{df_e}} \tag{12}$$

Note that $RMSE = \hat{\sigma}$.

- **Root Mean Squared Percentage Error (RMSPE)**

$$RMSPE = \sqrt{MSPE} = \sqrt{\frac{\sum_{t=L+1}^{m}\left(\frac{y_t - \hat{y}_t}{y_t}\right)^2}{(m-L)}} \tag{13}$$

- **R squared**

$$R^2 = 1 - \frac{\sum_{t=1}^{t=(m-L)}(y_t - \hat{y}_t)^2}{\sum_{t=1}^{t=(m-L)}(y_t - \bar{y})^2} = 1 - \frac{SS_e}{SS_t} \tag{14}$$

- **Bayesian Information Criterion (BIC)**

$$BIC = (m-L)ln\left(\frac{SS_e}{(m-L)}\right) + \left((p^c + 1)ln(m-L)\right) \tag{15}$$

- **Akaike Information Criterion (AIC)**

$$AIC = (m-L)ln\left(\frac{SS_e}{(m-L)}\right) + 2(p^c + 1) \tag{15'}$$

3. Scoring

Once the models $(\widehat{\boldsymbol{\beta}}, J_{sel})$ for all the required targets $(\boldsymbol{y})$ are built and post-estimation statistics are computed, the next task is to use these models to do scoring. There are two types of scoring: (1) fit: in-sample prediction for the past and current values of the target series; (2) forecast: out-of-sample prediction for future values of the target series.

3.1 Fit

Without loss of generality, we assume $\boldsymbol{X}$ and $\boldsymbol{X}_G$ are the selected predictor series matrices without lagged terms and with lagged terms, respectively; and $\widehat{\boldsymbol{\beta}}$ is the coefficient estimates vector for the target $\boldsymbol{y}$, so $\boldsymbol{X} = [\boldsymbol{X}_1, \dots, \boldsymbol{X}_K], \boldsymbol{X}_G = [\boldsymbol{1}, \boldsymbol{X}_{G_{11}}, \dots, \boldsymbol{X}_{G_{1L}}, \dots, \boldsymbol{X}_{G_{K1}}, \dots, \boldsymbol{X}_{G_{KL}}]$ and $\widehat{\boldsymbol{\beta}} = [\hat{\beta}_0, \hat{\beta}_{11}, \dots, \hat{\beta}_{1L}, \dots, \hat{\beta}_{K1}, \dots, \hat{\beta}_{KL}]^T$. Given that all series have *m* time points, in-sample prediction of $\boldsymbol{y}$ is one-step ahead prediction and can be written as

$$\hat{y}_t = \boldsymbol{X}_{G,t}\widehat{\boldsymbol{\beta}} = \hat{\beta}_0 + \sum_{j \in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot X_{G_{j\ell},t} \tag{16}$$

$$= \hat{\beta}_0 + \sum_{j \in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot X_{j,t-\ell}; \quad t = L+1, \dots, m. \tag{17}$$

The corresponding $100(1-\alpha)\%$ confidence interval of $\boldsymbol{y}$ is

$$\left[\hat{y}_t - t_{\alpha/2, df_e} \times \hat{\sigma},\ \hat{y}_t - t_{\alpha/2, df_e} \times\ \hat{\sigma}\right];\ \ t = L+1, \dots, m. \tag{18}$$

3.2 Forecast

Given that data is available up to time interval *m*, the one-step ahead forecast for $\boldsymbol{y}$ is

$$\hat{y}_m(1) = \hat{\beta}_0 + \sum_{j \in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot X_{j,m+1-\ell} \tag{19}$$

The h-step ahead forecast for $\boldsymbol{y}$ is

$$\hat{y}_m(h) = \hat{\beta}_0 + \sum_{j \in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot \hat{X}_{j,m+h-\ell} \tag{20}$$

where

$$\hat{X}_{j,m+h-\ell} = \begin{cases} X_{j,m+h-\ell}, & h \le \ell \\ \hat{X}_{j,m}(h-\ell), & h > \ell \end{cases}$$

Thus, forecasting the value of y_{m+2} requires us to first forecast the values of all the predictors up to time (*m* + 1). Forecasting the values of all the predictors up to time (*m* + 1) requires us to use Equation (19) on all the predictors $j \in J_{sel}$. Similarly, to predict the value of y_{m+3}, we need to forecast the values of predictors $j \in J_{sel}$ at time (*m* + 2) by using Equation (20). This task poses a bigger problem; to forecast the values of $j \in J_{sel}$ at time (*m* + 2), we first need to forecast the values of the predictors of $j \in J_{sel}$ at time (*m* + 1). That is, as we increasingly look into the future, we need to forecast more and more values to determine the value of y_{m+h}.

3.3 Approximated forecasting variances and intervals

In this subsection, we outline how forecasting variances and intervals can be computed for TCM models. We start by using the following representation for the linear model built by TCM for target y_{m+h}:

$$y_{m+h} = \hat{\beta}_0 + \sum_{j\in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot X_{j,m+h-\ell} + \varepsilon_{m+h} \tag{21}$$

where $\varepsilon_{m+h} \sim N(0, \sigma^2)$ and σ^2 is estimated as $\hat{\sigma}^2$ (computed in Section 2.6.1). Please note that we don't include parameter estimation error when defining forecasting error in TCM.

The forecasting error at *m* + 1 is defined as the difference between y_{m+1} and $\hat{y}_m(1)$, which can be written as

$$e_{y,m}(1) = y_{m+1} - \hat{y}_m(1) = \varepsilon_{m+1} \tag{22}$$

The forecasting variance for one-step ahead forecasts is computed as σ^2. For multi-step ahead forecasts, the forecasting error at *m* + *h* is

$$e_{y,m}(h) = y_{m+h} - \hat{y}_m(h) = \sum_{j\in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}_{j,\ell} \cdot e_{X_{j},m}(h-\ell) + \varepsilon_{m+h} \tag{23}$$

where $e_{X_{j,m}}(h-\ell) = X_{j,m+h-\ell} - \hat{X}_{j,m}(h-\ell)$ and $e_{X_{j,m}}(h-\ell) = 0$ if $h \le \ell$.

In general, $e_{X_{j,m}}(1), \dots, e_{X_{j,m}}(h-\ell)$ are not independent of each other. The larger the *h* is, the more complex the dependence is. In addition, $e_{X_{j,m}}(h-\ell)$ and $e_{X_{j,m}}(h-\ell)$ might not be independent for $j, i \in J_{sel}$. In order to fully consider the dependence, we need to write all time series in vector autoregressive (VAR) format. Since we assume the number of series *n* is usually large, the parameter matrix, which is an $n \times n$ matrix, might be too large to handle in computation of the forecasting variances. Therefore, we make the assumption that all forecasting error terms in Equation (23), $e_{X_{j,m}}(h-\ell), j \in J_{sel}, \ell = 1, \dots, L$, are independent, so it is easier to compute the forecasting variances.

Based on the above independence assumption, the approximated variance of the forecasting error, $e_{y,m}(h)$, is

$$\hat{\sigma}^2_{e_{y,m,h}} = \sum_{j\in J_{sel}} \sum_{\ell=1}^{L} \hat{\beta}^2_{j,\ell} \hat{\sigma}^2_{e_{X_j,m,h-\ell}} + \hat{\sigma}^2 \tag{24}$$

where $\hat{\sigma}^2_{e_{X_{j,m,h-\ell}}}$ is the variance of the forecasting error in the series X_j at $m + h - \ell$.

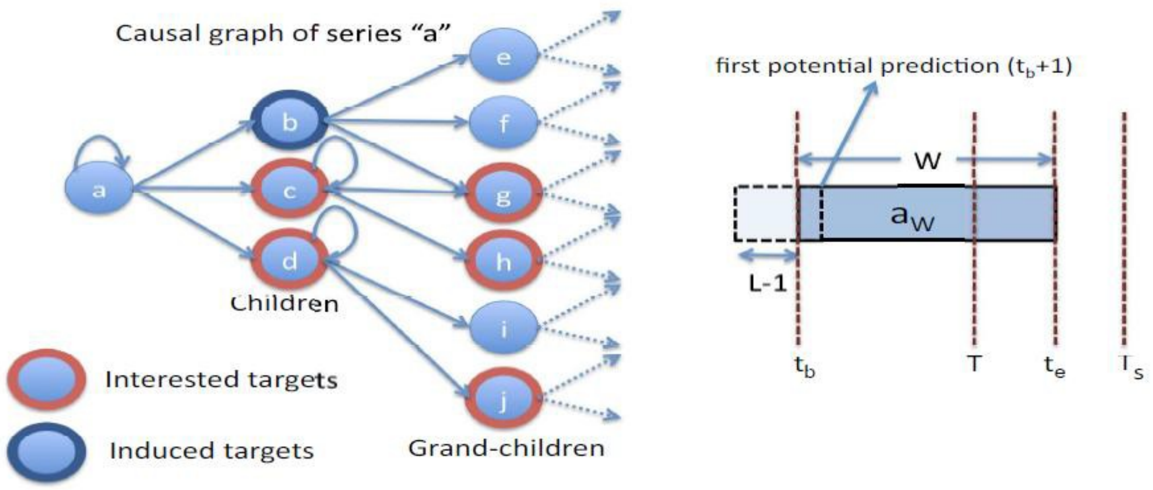
Then the corresponding $100(1 - \alpha)\%$ approximated forecasting interval of y_{m+h} can be expressed as

$$\left[\hat{y}_m(h) - t_{\alpha/2,df_e} \times \hat{\sigma}_{e_{y,m,h}}, \hat{y}_m(h) + t_{\alpha/2,df_e} \times \hat{\sigma}_{e_{y,m,h}}\right] \quad (25)$$

4. Scenario analysis

Scenario analysis refers to a capability of TCM to "play-out" the repercussions of artificially setting the value of a time series. A scenario is the set of forecasts that are generated by substituting the values of a *root* time series by a vector of substitute values, as illustrated in Figure 1.

Figure 1: Causal graph of a root time series and the specification of the vector of substitute values

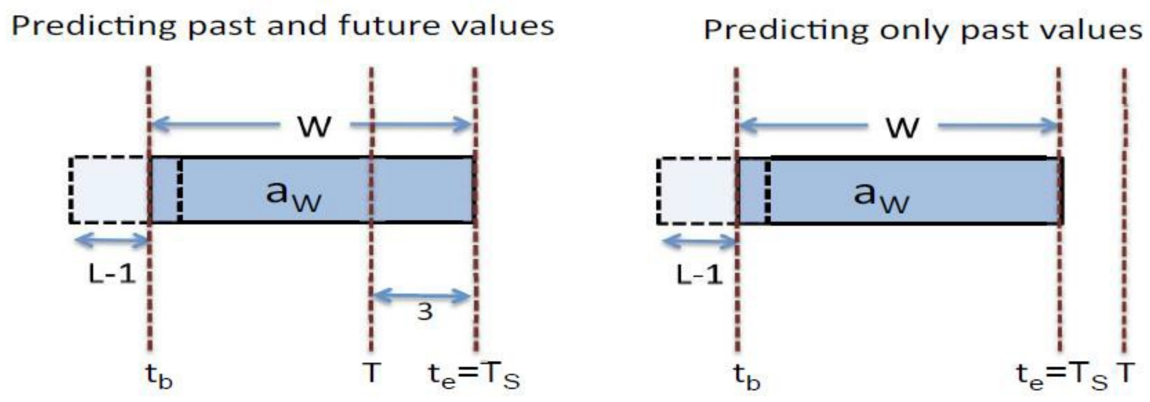


During scenario analysis, we specify the targets that we want to analyze as a response to changes in the values of the root series ("a" in Figure 1), along with the time window. In Figure 1, we are interested in the behavior of time series "c", "d", "g", "h", and "j" *only*. The rest of the time series are ignored. The figure also depicts the vector $\mathbf{a}_W$ of values for "a" that should be used instead of the observed or predicted values of "a". The values $(\boldsymbol{t_b}, \boldsymbol{t_e}, \boldsymbol{T}, \boldsymbol{T_s})$ specify the beginning and end of the replacement values for the root series, the current time, and the farthest time for analysis, respectively.

The partial Granger causal graph of time series "a" is shown in Figure 1. That is, "a" is the parent of itself, "b", "c", and "d". Similarly, it is the grand-parent of "e", "f", "g", "h", "i", and "j". Further descendents are possible, but only two generations suffice for the sake of explanation. Figure 1 also displays the specification of the vector $\mathbf{a}_W$, of length W, that contains the replacement values of the root series. In the example shown in the figure, $\mathbf{a}_W$ starts at time $t_b < T$, where T is the current time, and ends at $t_e > T$, which is in the future. We are also given T_s, the last time point $(t_e \leq T_s)$ for which we want to perform scenario analysis on the target variables. Finally, we are given a set of time series for which the scenario predictions are carried out. In the figure, these are "c", "d", "g", "h", and "j", which are marked with a thick red border. Since "b" is required to model "g", "b" is marked with a thick blue border to signify that it is an induced target. Given this information, the goal of scenario analysis is to forecast the values of the target time series ("c", "d", "g", "h", and "j") up to time T_s, based on the values of the root time series $\mathbf{a}_W$.

Notice that we have to predict values of targets up to time T_s, where T_s can be $> (T+1)$ or $\leq (T+1)$. When $T_s = (T+2)$, we need to compute the values of the predictors of the target time series at time $(T+1)$. Similarly, when $T_s = (T+3)$, we need to compute the values of the predictors' predictors at time $(T+1)$ and the values of the predictors at time $(T+2)$ before predicting the values of the target time series at time $(T+3)$.

Figure 2: Scenarios with and without predicting future values


Predicting past and future values
W
a_W
L-1
3
t_b
T
t_e=T_S
Predicting only past values
W
a_W
L-1
t_b
t_e=T_S
T

The left-hand panel in Figure 2 depicts a scenario where the values of ancestors of targets of interest also have to be predicted. In this particular case, $\boldsymbol{T_s} = (\boldsymbol{T}+\mathbf{3})$ and therefore it is necessary to predict the values of the predictors of the targets at $(\boldsymbol{T}+\mathbf{1})$ and $(\boldsymbol{T}+\mathbf{2})$, and values of the predictors' predictors at time $(\boldsymbol{T}+\mathbf{1})$. The right-hand panel depicts a scenario where the entire period of prediction is earlier than the current time $\boldsymbol{T}$ (i.e., $\boldsymbol{T_s} < \boldsymbol{T}$). In this case, all the values of the predictors and their ancestors are readily available.

Determining $\mathbf{a}_W$

In the discussion above, we have neglected the issue of $\mathbf{a}_W$, the substitute values for time series "a", which is the root time series. For purposes of scenario analysis, it is sufficient to consider that $\mathbf{a}_W$ is readily available. In a typical use case for scenario analysis, $\mathbf{a}_W$ will come from the values specified by the user's direct input, although its values could also come as input from a calling meta-process (as is the case with the use of scenario analysis as a sub- procedure in root cause analysis, as shown in Section 6).

Caveat on scenario analysis

It is possible to carry out scenario analysis for a time period that is entirely in the future; that is $t_b > T$. However, forecasting errors in the remaining predictors may make such scenario analysis inherently low-precision. That is, if $\theta = t_b - T$ and $t_b > T$, then the precision of scenario analysis decreases with an increase in θ.

4.1 *SA*, the scenario analysis algorithm

Input:

The inputs to *SA* are: (1) $\boldsymbol{r}$: the root time series; (2) $\boldsymbol{r}_W$: the vector of replacement values for time series $\boldsymbol{r}$; (3) (t_b, t_e, T, T_s): the beginning and end time for the modified values of $\boldsymbol{r}$, the current time, and the last time point for which target values need to be predicted, respectively; (4) D: a set of descendant target time series of interest along with their relation to $\boldsymbol{r}$ (which may be input as the Granger causal graph, *G*). Notice that the length of $\boldsymbol{r}_W$ is $t_e - t_b + 1$ and $t_e \leq T_s$. Furthermore, it is erroneous to have a target $\boldsymbol{d} \in D$, where $\boldsymbol{r}$ is not an ancestor of $\boldsymbol{d}$.

Output:

For each $\boldsymbol{d}$ in D, we output a vector $\boldsymbol{d}_{sa}$ containing values that pertain to the scenario analysis of these time series and the corresponding confidence intervals (when $T_s \leq T$) or apprxomiated forecasting intervals (when $T_s > T$). Please note that the time period for the children series in D is $[t_b + 1, T_s]$, for the grand-children series is $[t_b + 2, T_s]$, etc.

Preparation:

To prepare for *SA*, we first calculate the closure on the set of targets D^* that need to be predicted, which is determined by the relationship between $\boldsymbol{r}$ and each of the targets in D. Essentially, D^* is computed by iteratively looking at the path from each $\boldsymbol{d} \in D$ and adding all those intermediate nodes that are ancestors of $\boldsymbol{d}$ and are also descendents of $\boldsymbol{r}$. In the example shown in Figure 1, the time series “b” is itself not of primary interest, but since it is a parent of “g”, which is of interest, “b” is also added as a target of interest to the set {“c”, “d”, “g”, “h”, “j”}.

Next, we compute *M*, the set of models that need to be included in order to perform scenario analysis on D^*. Obviously, M contains the models for each of the series in D^*, i.e., $D^* \subset M$; however, depending on the time span of the scenario analysis, additional models of some time series might have to be brought in (see Figure 2). Basically, depending on how far ahead T_s is from *T*, we may need to compute the values of the ancestors (other than $\boldsymbol{r}$) of the targets of interest at time points $(T + 1), \ldots, (T_s - 1)$. That is, the set $\{M - D^*\}$(which may be $\emptyset$) contains all series that are needed for scenario analysis and are not descendants of $\boldsymbol{r}$.

At the end of the preparation phase we have D^* and *M*, which allows us to predict all the time series of interest.

Computation:

The computation in scenario analysis is exactly that of scoring the values of a set of time series (see Section 3). For each target in D^*, we have a range of time points for which we need to fit/forecast values. For example, for immediate children of the root (“c”, “d”, and the induced child “b” in Figure 1), this range is $[t_b + 1, T_s]$. Similarly, for grand-children (“g”, “h”, and “j” in Figure 1), this range is $[t_b + 2, T_s]$. Using the models in *M* and substituted values $\boldsymbol{r}_W$ for $\boldsymbol{r}$, this task can be carried out.

5. Outlier detection

One of the advantages of building TCM models is the ability to detect model-based outliers. Outliers can be defined in several ways. For now, we shall define an outlier in a time series to be a value that strays too far from its expected (fitted) value based on the TCM models. The detection process is based on the normal distribution assumption for series $\boldsymbol{y}$. Consider the value of a time series $\boldsymbol{y}$ at time *t*. Let y_t and $\hat{y}_t$ be the observed and expected values of $\boldsymbol{y}$ at time *t*, respectively; and $\hat{\sigma}^2$ be the variance of $\boldsymbol{y}$ from the TCM model (based on residuals). Given these inputs, we call y_t an outlier if the likelihood of y_t when modeled as a normal random variable with mean $\hat{y}_t$ and variance $\hat{\sigma}^2$ is below a particular threshold.

Input:

The inputs to OD (outlier detection) are: (1) $y_t, \forall\, t$; (2) $\hat{y}_t, \forall\, t$; (3) $\hat{\sigma}^2$; (4) the outlier threshold value $\kappa \in (0,1]$ (the default is 0.95).

Computation:

a) Under the assumption that the observed value y_t is a normal random variable with mean $\hat{y}_t$ and variance $\hat{\sigma}^2$, compute the square score at time *t* as

$$s_{sqr,t} = \frac{(y_t - \hat{y}_t)^2}{\hat{\sigma}^2} \tag{26}$$

b) Compute the outlier probability as

$$p_{sqr,t} = prob\left(\chi_1^2 \le s_{sqr,t}\right) \tag{27}$$

where χ_1^2 is a random variable with a chi-squared distribution with 1 degree of freedom.

c) Flag y_t as an outlier if $p_{sqr,t} \ge \kappa$.

Output:

The output to OD for series $\boldsymbol{y}$ is a set of time points with their corresponding outlier probabilities.

6. Outlier root cause analysis

In Section 5, we saw how to detect outliers. The next logical step is to find the likely causes for a time series whose value has been flagged as an outlier. Outlier root cause analysis refers to the capability to explore the Granger causal graph in order to analyse the key/root values that resulted in the outlier under question. To formalize this notion, consider a time series $\boldsymbol{y}$, whose observed value at time *t* (that is, y_t) has been flagged as an outlier due to its abnormal deviation from its expected value $\hat{y}_t$. The goal of outlier root cause analysis (*ORCA*) is to output the set of time series $\mathcal{A}$ that can be considered as root causes of the anomalous value of y_t. The idea is that setting the values of time series in the predictor set $\boldsymbol{X}$ to their normal/expected values, instead of their observed values, will bring the outlying y_t back to normal. The normal value of y_t is unknown so we specify it with the expected value of $\boldsymbol{y}$ at time *t* as predicted by $\boldsymbol{y}$'s univariate model, which is an AR(L) model, and denoted as $\tilde{y}_t$.

The result of *ORCA* has the following objective function with a constraint as follows:

$$\arg\max_{x \in \mathcal{A}_y} |\hat{y}_t - \tilde{y}_t| - |\hat{y}_{t|x=\hat{x}} - \tilde{y}_t| \tag{28}$$

$$\text{s.t.}\ |\hat{y}_t - \tilde{y}_t| \ge |\hat{y}_{t|x=\hat{x}} - \tilde{y}_t|$$

where $\mathcal{A}_y$ corresponds to the set of ancestors of $\boldsymbol{y}$ according to the Granger causal graph *G*. The quantity $\hat{y}_{t|x=\hat{x}}$ should be interpreted as the likely predicted value of *y* at time *t* had the value of its ancestor *x* been set to its expected value of $\hat{x}$. We see that Equation (28) is made up of two parts: (1) the portion $|\hat{y}_t - \tilde{y}_t|$, which is the degree of “outlier-ness” of y at t as predicted by the “Granger model”, where the outlier-ness is judged based on what is expected from the history of $\boldsymbol{y}$; (2) the portion $|\hat{y}_{t|x=\hat{x}} - \tilde{y}_t|$, which is the degree of “outlier-ness” of y at t as predicted by the “Granger model”, if x was corrected. In other words, Equation (28) amounts to replacing the observed value y_t by its “expected” value, given by a simpler, univariate model. Therefore Equation (28) expresses the reduction in the degree of outlier-ness in y_t brought about by correcting x.

6.1 *ORCA*, the outlier root cause analysis algorithm

Input:

The inputs to *ORCA* are: (1) $\boldsymbol{y}$, the anomalous time series; (2) t, the time at which the anomaly was detected; (3) y_t, the anomalous value; (4) $\hat{y}_t$, the expected value of y_t; (5) k, the oldest generation of ancestors to search based on the Granger causal graph, G.

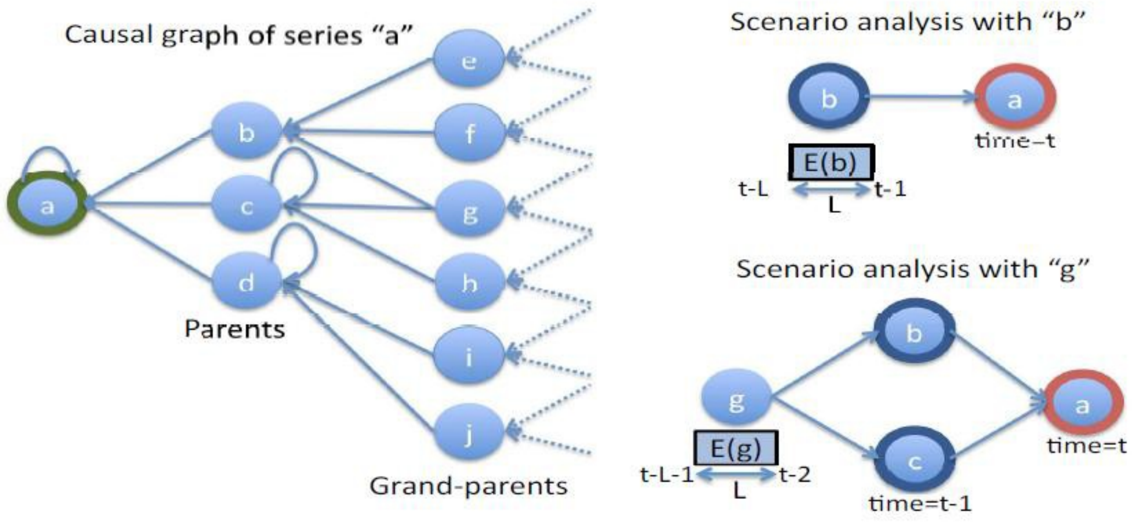
Output:

ORCA outputs the set of root causes $\mathcal{A}$ of the anomaly in y_t, where each $\boldsymbol{x} \in \mathcal{A}$ maximizes the objective function in Equation (28) by the same amount.

Preparation:

To prepare for *ORCA*, we first compute $\mathcal{A}_y$, the set of ancestors that need to be examined as the potential root causes of the anomaly in y_t.

Figure 3: Outlier root cause analysis for a time series



In the example shown in Figure 3, assuming that $\boldsymbol{y}$=“a” and $k = 2$, then $\mathcal{A}_y$ = { “b”, “c”, “d”, “e”, “f”, “g”, “h”, “i”, “j”}. $\mathcal{A}_y$ can be computed by performing a reverse breadth-first search from $\boldsymbol{y}$ to k levels.

Second, each potential root cause $\boldsymbol{x} \in \mathcal{A}_y$ is prepped for scenario analysis by computing the vector of substitute values of $\boldsymbol{x}$ to be used during scenario analysis. Note that the length of this substitute vector is L, the lag. For example, consider $\boldsymbol{b}_L$, the substitute for time series “b” in Figure 3. As “b” is a parent of “a”, we need to compute the fits of “b” from $(t-L)$ to $(t-1)$. On the other hand, as “g” is a grand-parent of “a”, $\boldsymbol{g}_L$ contains the fits for “g” from the time $(t-L-1)$ to $(t-2)$ (see Section 3.1 for computation of fits). Please note that this approach assumes that any anomalies are purely in “b” (the parent series) or “g” (the grandparent series). In particular, it is assumed that anomalies in “b” are not caused by values in the grandparent series, including anomalous values in the grandparent series.

Third, for each potential root cause $\boldsymbol{x} \in \mathcal{A}_y$, scenario analysis is carried out (see Section 4) using the substitute values computed in the previous step. For the example in Figure 3, scenario analysis is called for series “b” with the parameters $(\boldsymbol{r}-\boldsymbol{b}, \boldsymbol{r}_W = b_L, t_b = (t-L), t_e = (t-1), T = t, D = \{\boldsymbol{a}\}, T_s = t)$. And the result of scenario analysis is $\hat{y}_{t|x=\hat{x}}$.

Computation:

The process of *ORCA* is as follows:

- Initiaize $\mathcal{A}$, the set of potential root causes for y_t, to $\emptyset$.
  Initialize obj_{max}, the maximum objective function value, to 0.
- Suppose there are J series in $\mathcal{A}_y$, $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_J$.
  For each $\boldsymbol{x}_j$, $j \in 1, \ldots, J$, compute $obj_j = |\hat{y}_t - \tilde{y}_t| - |\hat{y}_{t|x_j=\hat{x}_j} - \tilde{y}_t|$.
  If $obj_j \geq obj_{max}$, set $obj_{max} = obj_j$ and store $\boldsymbol{x}_j$ in $\mathcal{A}$.

References

[1]. Arnold, A., Liu, Y., and Abe, N. (2007). Temporal causal modeling with graphical granger methods. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, pages 66–75, New York, NY, USA. ACM.

[2]. Darema, F., George, D. A., Norton, V. A., and Pfister, G. F. (1988). A single-program-multiple-data computational model for EPEX/FORTRAN. *Parallel Computing*, 7(1):11–24.

[3]. Dean, J. and Ghemawat, S. (2008). Mapreduce: simplified data processing on large clusters. volume 51.

[4]. Duchi, J., Gould, S., and Koller, D. (2008). Projected subgradient methods for learning sparse gaussians. In *Proceedings of the Twenty-fourth Conference on Uncertainty in AI (UAI).*

[5]. Friedman, J., Hastie, T., and Tibshirani, R. (2008). Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441.

[6]. Granger, C. W. J. (1980). Testing for causality : A personal viewpoint. *Journal of Economic Dynamics and Control*, 2(1):329–352.

[7]. Hsieh, C.-J., Sustik, M. A., Dhillon, I. S., and Ravikumar, P. (2011). Sparse inverse covariance matrix estimation using quadratic approximation. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K., editors, *Advances in Neural Information Processing Systems 24*, pages 2330–2338. http://nips.cc/.

[8]. Kambadur, P. and Lozano, A. C. (2013). A parallel, block greedy method for sparse inverse covariance estimation for ultra-high dimensions. In *Sixteenth International Conference on Artificial Intelligence and Statistics (AISTATS).*

[9]. Li, L. and chuan Toh, K. (2010). An inexact interior point method for l1-regularized sparse covariance selection. Technical report, National University Of Singapore.

[10]. Lozano, A. C., Abe, N., Liu, Y., and Rosset, S. (2009). Grouped graphical granger modeling methods for temporal causal modeling. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 577–586, New York, NY, USA. ACM.

[11]. Lozano, A. C., Swirszcz, G., and Abe, N. (2011). Group orthogonal matching pursuit for logistic regression. *Journal of Machine Learning Research - Proceedings Track*, 15:452–460.

[12]. MPI Forum (1995). Message Passing Interface. http://www.mpi-forum.org/.

[13]. MPI Forum (1997). Message Passing Interface-2. http://www.mpi-forum.org/.

[14]. O.Banerjee, El Ghaoui, L., and d'Aspremont, A. (2008). Model selection through sparse maximum likelihood estimation for multivariate gaussian or binary data. *Journal of Machine Learning Research*, 9:485–516.

[15]. Scheinberg, K., Ma, S., and Goldfarb, D. (2010). Sparse inverse covariance selection via alternating linearization methods. *CoRR*, abs/1011.0097.

[16]. Scheinberg, K. and Rish, I. (2010). Learning sparse gaussian markov networks using a greedy coordinate ascent approach. In *Proceedings of the 2010 European conference on Machine learning and knowledge discovery in databases: Part III*, ECML PKDD'10, pages 196–212, Berlin, Heidelberg. Springer-Verlag.

[17]. Strang, G. (1993). *Introduction to Linear Algebra*. Wellesley-Cambridge Press.

TREE Algorithms

The TREE procedure creates a tree-based classification model using the CART, CHAID, or QUEST algorithm.

CART Algorithms

The CART algorithm is based on Classification and Regression Trees by Breiman et al (1984). A CART tree is a binary decision tree that is constructed by splitting a node into two child nodes repeatedly, beginning with the root node that contains the whole learning sample.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The dependent, or target, variable. It can be ordinal categorical, nominal categorical or continuous. If *Y* is categorical with *J* classes, its class takes values in *C* = {1, …, *J*}.
X_m, *m*=1,...,*M*	The set of all predictor variables. A predictor can be ordinal categorical, nominal categorical or continuous.
$\hbar = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$	The whole learning sample.
$\hbar(t)$	The learning samples that fall in node *t*.
w_n	The case weight associated with case *n*.
f_n	The frequency weight associated with case *n*. Non-integral positive value is rounded to its nearest integer.
π(*j*), *j*=1,...,*J*	Prior probability of *Y* = *j*, *j* = 1, …, *J*.
p(*j*,*t*), *j*=1,...,*J*	The probability of a case in class *j* and node *t*.
p(*t*)	The probability of a case in node *t*.
p(*j*|*t*), *j*=1,...,*J*	The probability of a case in class *j* given that it falls into node *t*.
C(*i*|*j*)	The cost of miss-classifying a class *j* case as a class *i* case. *C*(*j*|*j*)=0

Tree Growing Process

The basic idea of tree growing is to choose a split among all the possible splits at each node so that the resulting child nodes are the "purest". In this algorithm, only univariate splits are considered. That is, each split depends on the value of only one predictor variable. All possible splits consist of possible splits of each predictor. If *X* is a nominal categorical variable of *I* categories, there are $2^{I-1} - 1$ possible splits for this predictor. If *X* is an ordinal categorical or continuous variable with *K* different values, there are *K*−1 different splits on *X*. A tree is grown starting from the root node by repeatedly using the following steps on each node.

1. Find each predictor's best split.

For each continuous and ordinal predictor, sort its values from the smallest to the largest. For the sorted predictor, go through each value from top to examine each candidate split point (call it *v*, if *x*≤*v*, the case goes to the left child node, otherwise, it goes to the right) to determine the best.

The best split point is the one that maximize the splitting criterion the most when the node is split according to it. The definition of splitting criterion is in a later section.

For each nominal predictor, examine each possible subset of categories (call it A, if $x \in A$, the case goes to the left child node, otherwise, it goes to the right) to find the best split.

2. Find the node's best split.

   Among the best splits found in step 1, choose the one that maximizes the splitting criterion.

3. Split the node using its best split found in step 2 if the stopping rules are not satisfied.

Splitting Criteria and Impurity Measures

At node *t*, the best split *s* is chosen to maximize a splitting criterion $\Delta i\left(s,t\right)$. When the impurity measure for a node can be defined, the splitting criterion corresponds to a decrease in impurity. $\Delta I\left(s,t\right)=p\left(t\right)\Delta i\left(s,t\right)$is referred to as the improvement.

Categorical Dependent Variable

If *Y* is categorical, there are three splitting criteria available: Gini, Twoing, and ordered Twoing criteria. At node *t*, let probabilities *p*(*j*,*t*), *p*(*t*) and *p*(*j*|*t*) be estimated by

$$p\left(j,t\right)=\frac{\pi\left(j\right)N_{w,j}\left(t\right)}{N_{w,j}}$$

$$p\left(t\right)=\sum_{j}p\left(j,t\right)$$

$$p\left(j|t\right)=\frac{p\left(j,t\right)}{p\left(t\right)}=\frac{p\left(j,t\right)}{\sum_{j}p\left(j,t\right)}$$

where

$$N_{w,j}=\sum_{n\in\hbar}w_nf_nI\left(y_n=j\right)$$

$$N_{w,j}\left(t\right)=\sum_{n\in\hbar\left(t\right)}w_nf_nI\left(y_n=j\right)$$

with *I*(*a*=*b*) being the indicator function taking value 1 when *a*=*b*, 0 otherwise.

Gini Criterion

The Gini impurity measure at a node *t* is defined as

$$i\left(t\right)=\Sigma_{i,j}C\left(i|j\right)p\left(i|t\right)p\left(j|t\right)$$

The Gini splitting criterion is the decrease of impurity defined as

$$\Delta i\left(s,t\right)=i\left(t\right)-p_L i\left(t_L\right)-p_R i\left(t_R\right)$$

where p_L and p_R are probabilities of sending a case to the left child node t_L and to the right child node t_R respectively. They are estimated as $p_L=p\left(t_L\right)/p\left(t\right)$ and $p_R=p\left(t_R\right)/p\left(t\right)$.

Note: When user-specified costs are involved, the altered priors can optionally be used to replace the priors. When altered priors are used, the problem is considered as if no costs are involved. The altered prior is defined as $\pi'\left(j\right)=\frac{C(j)\pi(j)}{\sum_j C\left(j\right)\pi\left(j\right)}$, where $C\left(j\right)=\Sigma_i C\left(i|j\right)$.

Note: When the Gini index is used to find the improvement for a split during tree growth, only those records in node *t* and the root node with valid values for the split-predictor are used to compute $N_j(t)$ and N_j, respectively.

Twoing Criterion

$$\Delta i\left(s,t\right)=p_L p\left[\sum_j \left|p\left(j|\,t_L\right)-p\left(j\,|t_R\right)\right|\right]^2$$

Ordered Twoing Criterion

Ordered Twoing is used only when *Y* is ordinal categorical. Its algorithm is as follows:

1. First separate the class $C=\{1,\ldots,J\}$ of *Y* as two super-classes C_1 and $C_2=C-C_1$ such that C_1 is of the form $C_1=\{1,\ldots,j_1\}$, $j_1=1,\ldots,J-1$.
2. Using the 2-class measure $i(t)=p(C_1\mid t)p(C_2\mid t)$, find the split $s^*(C_1)$ that maximizes

$$\Delta i\left(s,t\right)=i\left(t\right)-p_L i\left(t_L\right)-p_R i\left(t_R\right)=p_L p_R\left[\sum_{j\in C_1}\left\{p\left(j\,|t_L\right)-p\left(j|\,t_R\right)\right\}\right]^2$$

3. Find the super-class C^*_1 of C_1 which maximizes $\Delta i\left(s^*\left(C_1\right),t\right)$.

Continuous Dependent Variable

When *Y* is continuous, the splitting criterion $\Delta i\left(s,t\right)=i\left(t\right)-p_L i\left(t_L\right)-p_R i\left(t_R\right)$ is used with the Least Squares Deviation (LSD) impurity measures

$$i\left(t\right)=\frac{\sum_{n\in\hbar(t)} w_n f_n (y_n-\overline{y}\left(t\right))^2}{\sum_{n\in\hbar(t)} w_n f_n}$$

where

$$p_L = N_w(t_L)/N_w(t), p_R = N_w(t_R)/N_w(t),\ N_w(t) = \sum_{n \in \hbar(t)} w_n f_n$$

$$\overline{y}(t) = \frac{\sum_{n \in \hbar(t)} w_n f_n y_n}{N_w(t)}$$

Stopping Rules

Stopping rules control if the tree growing process should be stopped or not. The following stopping rules are used:

- If a node becomes pure; that is, all cases in a node have identical values of the dependent variable, the node will not be split.
- If all cases in a node have identical values for each predictor, the node will not be split.
- If the current tree depth reaches the user-specified maximum tree depth limit value, the tree growing process will stop.
- If the size of a node is less than the user-specified minimum node size value, the node will not be split.
- If the split of a node results in a child node whose node size is less than the user-specified minimum child node size value, the node will not be split.
- If for the best split s^* of node t, the improvement $\Delta I(s^*, t) = p(t)\Delta i(s^*, t)$ is smaller than the user-specified minimum improvement, the node will not be split.

Surrogate Splits

Given a split $X^* \le s^*$, its surrogate split is a split using another predictor variable X, $X \le s_X$ (or $X > s_X$), such that this split is most similar to it and is with positive predictive measure of association. There may be multiple surrogate splits. The bigger the predictive measure of association is, the better the surrogate split is.

Predictive measure of association

Let $\hbar_{X^* \cap X}$ (resp. $\hbar_{X^* \cap X}(t)$) be the set of learning cases (resp. learning cases in node t) that has non-missing values of both X^* and X. Let $p(s^* \approx s_X|t)$ be the probability of sending a case in $\hbar_{X^* \cap X}(t)$ to the same child by both s^* and s_X, and $\tilde{s}_X$ be the split with maximized probability $p(s^* \approx \tilde{s}_X|t) = \max_{s_X}(p(s^* \approx s_X|t))$.

The predictive measure of association $\lambda(s^* \approx \tilde{s}_X|t)$ between s^* and $\tilde{s}_X$ at node t is

$$\lambda(s^* \approx \tilde{s}_X|t) = \frac{\min(p_L, p_R) - (1 - p(s^* \approx \tilde{s}_X|t))}{\min(p_L, p_R)}$$

where p_L (resp. p_R) is the relative probability that the best split s^* at node t sends a case with non-missing value of X^* to the left (resp. right) child node. And where

$$p\left(s^* \approx s_X | t\right) = \begin{cases} \sum_j \dfrac{\pi(j)\, N_{w,j}\left(s^* \approx s_X, t\right)}{N_{w,j}(X^* \cap X)} & \text{if} Y \text{ is categorical} \\ \frac{N_w(s^* \approx s_X, t)}{N_w(X^* \cap X)} & \text{if} Y \text{ is continuous} \end{cases}$$

with

$$N_w\left(X^* \cap X\right) = \sum_{n \in \hbar_{X^* \cap X}} w_n f_n,\ N_w\left(X^* \cap X, t\right) = \sum_{n \in \hbar_{X^* \cap X}(t)} w_n f_n$$

$$N_w\left(s^* \approx s_X, t\right) = \sum_{n \in \hbar_{X^* \cap X}(t)} w_n f_n I\left(n : s^* \approx s_X\right)$$

$$N_{w,j}\left(X^* \cap X\right) = \sum_{n \in \hbar_{X^* \cap X}} w_n f_n I\left(y_n = j\right),\ N_{w,j}\left(X^* \cap X\right) = \sum_{n \in \hbar_{X^* \cap X}(t)} w_n f_n I\left(y_n = j\right)$$

$$N_{w,j}\left(s^* \approx s_X, t\right) = \sum_{n \in \hbar_{X^* \cap X}(t)} w_n f_n I\left(y_n = j\right) I\left(n : s^* \approx s_X\right)$$

and $I\left(n : s^* \approx s_X\right)$ being the indicator function taking value 1 when both splits s^* and s_X send the case n to the same child, 0 otherwise.

Missing Value Handling

If the dependent variable of a case is missing, this case will be ignored in the analysis. If all predictor variables of a case are missing, this case will also be ignored. If the case weight is missing, zero, or negative, the case is ignored. If the frequency weight is missing, zero, or negative, the case is ignored.

The surrogate split method is otherwise used to deal with missing data in predictor variables. Suppose that $X^* < s^*$ is the best split at a node. If value of X^* is missing for a case, the best surrogate split (among all non-missing predictors associated with surrogate splits) will be used to decide which child node it should go. If there are no surrogate splits or all the predictors associated with surrogate splits for a case are missing, the majority rule is used.

Variable Importance

The Measure of Importance $M(X)$ of a predictor variable X in relation to the final tree T is defined as the (weighted) sum across all splits in the tree of the improvements that X has when it is used as a primary or surrogate (but not competitor) splitter. That is,

$$M(X) = \sum_{t \in T} \Delta\left(\tilde{s}_X, t\right)$$

If, for a given t, the rank of the surrogate is larger than the maximum number of surrogates to keep in each node, then the contribution of that split is set to 0.

The Variable Importance *VI*(*X*) of X is expressed in terms of a normalized quantity relative to the variable having the largest measure of importance. It ranges from 0 to 100, with the variable having the largest measure of importance scored as 100. That is,

$$VI(X) = \frac{M(X)}{\max_X M(X)} \times 100$$

References

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees*. New York: Chapman & Hall/CRC.

CHAID and Exhaustive CHAID Algorithms

The CHAID algorithm is originally proposed by Kass (1980) and the Exhaustive CHAID is by Biggs et al (1991). Algorithm CHAID and Exhaustive CHAID allow multiple splits of a node.

Both CHAID and exhaustive CHAID algorithms consist of three steps: merging, splitting and stopping. A tree is grown by repeatedly using these three steps on each node starting from the root node.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The dependent variable, or target variable. It can be ordinal categorical, nominal categorical or continuous. If *Y* is categorical with *J* classes, its class takes values in *C* = {1, …, *J*}.
X_m, *m*=1, ..., *M*	The set of all predictor variables. A predictor can be ordinal categorical, nominal categorical or continuous.
$\hbar = \{\mathbf{x}_n, y_n\}_{n=1}^N$	The whole learning sample.
w_n	The case weight associated with case *n*.
f_n	The frequency weight associated with case *n*. Non-integral positive value is rounded to its nearest integer.

CHAID Algorithm

The following algorithm only accepts nominal or ordinal categorical predictors. When predictors are continuous, they are transformed into ordinal predictors before using the following algorithm.

Binning Continuous Predictors

For a given set of break points $a_1, a_2, ..., a_{K-1}$ (in ascending order), a given *x* is mapped into category *C*(*x*) as follows:

$$C(x) = \begin{cases} 1 & x \le a_1 \\ k+1 & a_k < x \le a_{k+1}, k = 1, ..., K-2 \\ K & a_{K-1} < x \end{cases}$$

If *K* is the desired number of bins, the break points are computed as follows:

Calculate the rank of x_i. Frequency weights are incorporated when calculating the ranks. If there are ties, the average rank is used. Denote the rank and the corresponding values in ascending order as $\{r_{(i)}, x_{(i)}\}_{i=1}^{n}$.

For $k = 0$ to $(K-1)$, set $I_k = \left\{ i : \left\lfloor r_{(i)} \frac{K}{N_f+1} \right\rfloor = k \right\}$ where $\lfloor x \rfloor$ denotes the floor integer of x. If I_k is not empty, $i_k = \max\{i : i \in I_k\}$. The break points are set equal to the x values corresponding to the i_k, excluding the largest.

Merging

For each predictor variable *X*, merge non-significant categories. Each final category of *X* will result in one child node if *X* is used to split the node. The merging step also calculates the adjusted *p*-value that is to be used in the splitting step.

1. If *X* has 1 category only, stop and set the adjusted *p*-value to be 1.
2. If *X* has 2 categories, go to step 8.
3. Else, find the allowable pair of categories of *X* (an allowable pair of categories for ordinal predictor is two adjacent categories, and for nominal predictor is any two categories) that is least significantly different (i.e., most similar). The most similar pair is the pair whose test statistic gives the largest *p*-value with respect to the dependent variable *Y*. How to calculate *p*-value under various situations will be described in later sections.
4. For the pair having the largest *p*-value, check if its *p*-value is larger than a user-specified alpha-level α_{merge}. If it does, this pair is merged into a single compound category. Then a new set of categories of *X* is formed. If it does not, then go to step 7.
5. (Optional) If the newly formed compound category consists of three or more original categories, then find the best binary split within the compound category which *p*-value is the smallest. Perform this binary split if its *p*-value is not larger than an alpha-level $\alpha_{\text{split-merge}}$.
6. Go to step 2.
7. (Optional) Any category having too few observations (as compared with a user-specified minimum segment size) is merged with the most similar other category as measured by the largest of the *p*-values.
8. The adjusted *p*-value is computed for the merged categories by applying Bonferroni adjustments that are to be discussed later.

Splitting

The "best" split for each predictor is found in the merging step. The splitting step selects which predictor to be used to best split the node. Selection is accomplished by comparing the adjusted *p*-value associated with each predictor. The adjusted *p*-value is obtained in the merging step.

1. Select the predictor that has the smallest adjusted *p*-value (i.e., most significant).
2. If this adjusted *p*-value is less than or equal to a user-specified alpha-level α_{split}, split the node using this predictor. Else, do not split and the node is considered as a terminal node.

Stopping

The stopping step checks if the tree growing process should be stopped according to the following stopping rules.

1. If a node becomes pure; that is, all cases in a node have identical values of the dependent variable, the node will not be split.
2. If all cases in a node have identical values for each predictor, the node will not be split.
3. If the current tree depth reaches the user specified maximum tree depth limit value, the tree growing process will stop.
4. If the size of a node is less than the user-specified minimum node size value, the node will not be split.
5. If the split of a node results in a child node whose node size is less than the user-specified minimum child node size value, child nodes that have too few cases (as compared with this minimum) will merge with the most similar child node as measured by the largest of the *p*-values. However, if the resulting number of child nodes is 1, the node will not be split.

Exhaustive CHAID Algorithm

Splitting and stopping steps in Exhaustive CHAID algorithm are the same as those in CHAID. Merging step uses an exhaustive search procedure to merge any similar pair until only a single pair remains.

Also like CHAID, only nominal or ordinal categorical predictors are allowed, continuous predictors are first transformed into ordinal predictors before using the following algorithm.

Merging

1. If *X* has 1 category only, then set the adjusted *p*-value to be 1.
2. Set *index* = 0. Calculate the *p*-value based on the set of categories of *X* at this time. Call the *p*-value *p*(*index*) = *p*(0).
3. Else, find the allowable pair of categories of *X* that is least significantly different; that is, most similar. This can be determined by the pair whose test statistic gives the largest *p*-value with

respect to the dependent variable *Y*. How to calculate *p*-value under various situations will be described in a later section.

4. Merge the pair that gives the largest *p*-value into a compound category.
5. (Optional) If the compound category just formed contains three or more original categories, search for a binary split of this compound category that gives the smallest *p*-value. If this *p*-value is larger than the one in forming the compound category by merging in the previous step, perform the binary split on that compound category.
6. Update the *index* = *index* + 1, calculate the *p*-value based on the set of categories of *X* at this time. Denote *p*(*index*) as the *p*-value.
7. Repeat 3 to 6 until only two categories remain. Then among all the indices, find the set of categories such that *p*(*index*) is the smallest.
8. (Optional) Any category having too few observations (as compared with a user-specified minimum segment size) is merged with the most similar other category as measured by the largest *p*-value.
9. The adjusted *p*-value is computed by applying Bonferroni adjustments which are to be discussed in a later section.

Unlike CHAID algorithm, no user-specified alpha-level is needed. Only the alpha-level α_{split} is needed in the splitting step.

p-Value Calculations

Calculations of (unadjusted) *p*-values in the above algorithms depend on the type of dependent variable.

The merging step of both CHAID and Exhaustive CHAID sometimes needs the *p*-value for a pair of *X* categories, and sometimes needs the *p*-value for all the categories of *X*. When the *p*-value for a pair of *X* categories is needed, only part of data in the current node is relevant. Let *D* denote the relevant data. Suppose in *D* there are *I* categories of *X*, and *J* categories of *Y* (if *Y* is categorical). The *p*-value calculation using data in *D* is given below.

Scale Dependent Variable

If the dependent variable *Y* is scale, perform an ANOVA *F* test that tests if the means of *Y* for different categories of *X* are the same. This ANOVA *F* test calculates the *F*-statistic and hence derives the *p*-value as

$$F = \frac{\sum_{i=1}^{I} \sum_{n \in D} w_n f_n I(x_n = i)\left(\bar{y}_i - \bar{y}\right)^2 / (I-1)}{\sum_{i=1}^{I} \sum_{n \in D} w_n f_n I(x_n = i)\left(y_n - \bar{y}_i\right)^2 / \left(N_f - I\right)}$$

$$p = \Pr\left(F\left(I-1, N_f - I\right) > F\right)$$

where

$$\overline{y}_i = \frac{\sum_{n \in D} w_n f_n y_n I(x_n = i)}{\sum_{n \in D} w_n f_n I(x_n = i)}, \overline{y} = \frac{\sum_{n \in D} w_n f_n y_n}{\sum_{n \in D} w_n f_n}, N_f = \sum_{n \in D} f_n$$

and $F(I-1, N_f - I)$ is a random variable following a *F*-distribution with degrees of freedom *I*−1 and $N_f - I$.

Nominal Dependent Variable

If the dependent variable *Y* is nominal categorical, the null hypothesis of independence of *X* and *Y* is tested. To perform the test, a contingency (or count) table is formed using classes of *Y* as columns and categories of the predictor *X* as rows. The expected cell frequencies under the null hypothesis are estimated. The observed cell frequencies and the expected cell frequencies are used to calculate the Pearson chi-squared statistic or likelihood ratio statistic. The *p*-value is computed based on either one of these two statistics.

The Pearson's Chi-square statistic and likelihood ratio statistic are, respectively,

$$X^2 = \sum_{j=1}^{J} \sum_{i=1}^{I} \frac{(n_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}}$$

$$G^2 = 2\sum_{j}^{J} \sum_{i=1}^{I} n_{ij} \ln(n_{ij}/\hat{m}_{ij})$$

where $n_{ij} = \sum_{n \in D} f_n I(x_n = i \wedge y_n = j)$ is the observed cell frequency and $\hat{m}_{ij}$ is the estimated expected cell frequency for cell $(x_n = i, y_n = j)$ following the independence model. The corresponding *p*-value is given by $p = \Pr(\chi_d^2 > X^2)$ for Pearson's Chi-square test or $p = \Pr(\chi_d^2 > G^2)$ for likelihood ratio test, where χ_d^2 follows a chi-squared distribution with degrees of freedom *d* = (*J*−1)(*I*−1).

Estimation of Expected Cell Frequencies without Case Weights

$$\hat{m}_{ij} = \frac{n_{i.} n_{.j}}{n_{..}}$$

where

$$n_{i.} = \sum_{j=1}^{J_t} n_{ij}, n_{.j} = \sum_{i=1}^{I_t} n_{ij}, n_{..} = \sum_{j=1}^{J_t} \sum_{i=1}^{I_t} n_{ij}$$

Estimation of Expected Cell Frequencies with Case Weights

If case weights are specified, the expected cell frequency under the null hypothesis of independence is of the form

$$m_{ij} = \overline{w}_{ij}^{-1}\alpha_i\beta_j$$

where α_i and β_j are parameters to be estimated, and

$$\overline{w}_{ij} = \frac{w_{ij}}{n_{ij}},\ w_{ij} = \sum_{n\in D} w_n f_n I\left(x = i \wedge y_n = j\right)$$

Parameters estimates $\hat{\alpha}_i$, $\hat{\beta}_j$, and hence $\hat{m}_{ij}$, are resulted from the following iterative procedure.

1. $k = 0, \alpha_i^{(0)} = \beta_j^{(0)} = 1,\ m_{ij}^{(0)} = \overline{w}_{ij}^{-1}$
2. $\alpha_i^{(k+1)} = \dfrac{n_{i.}}{\sum_j \overline{w}_{ij}^{-1}\beta_j^{(k)}} = \alpha_i^{(k)}\dfrac{n_{i.}}{\sum_j m_{ij}^{(k)}}$
3. $\beta_j^{(k+1)} = \dfrac{n_{.j}}{\sum_i \overline{w}_{ij}^{-1}\alpha_i^{(k+1)}}$
4. $m_{ij}^{(k+1)} = \overline{w}_{ij}^{-1}\alpha_i^{(k+1)}\beta_j^{(k+1)}$
5. If $\max_{i,j}\left|m_{ij}^{(k+1)} - m_{ij}^{(k)}\right| < \epsilon$, stop and output $\alpha_i^{(k+1)}, \beta_j^{(k+1)}$ and $m_{ij}^{(k+1)}$ as the final estimates. Otherwise, *k*=*k*+1, go to step 2.

Ordinal Dependent Variable

If the dependent variable *Y* is categorical ordinal, the null hypothesis of independence of *X* and *Y* is tested against the row effects model, with the rows being the categories of *X* and columns the classes of *Y*, proposed by Goodman (1979). Two sets of expected cell frequencies, $\hat{m}_{ij}$ (under the hypothesis of independence) and $\hat{\hat{m}}_{ij}$ (under the hypothesis that the data follow a row effects model), are both estimated. The likelihood ratio statistic and the *p*-value are

$$H^2 = 2\sum_{i=1}^{I}\sum_{j=1}^{J}\hat{\hat{m}}_{ij}\ln\left(\hat{\hat{m}}_{ij}/\hat{m}_{ij}\right)$$

$$p = \Pr\left(\chi^2_{I-1} > H^2\right)$$

Estimation of Expected Cell Frequencies under Row Effects Model

In the row effects model, scores for classes of *Y* are needed. By default, the order of a class of *Y* is used as the class score. Users can specify their own set of scores. Scores are set at the beginning of the tree and kept unchanged afterward. Let s_j be the score for class *j* of *Y*, *j* = 1, …, *J*. The expected cell frequency under the row effects model is given by

$$m_{ij} = \overline{w}_{ij}^{-1}\alpha_i\beta_j\gamma_i$$

where

$$\overline{s} = \sum_{j=1}^{J} w_{.j} s_j / \sum_{j=1}^{J} w_{.j}$$

in which $w_{.j} = \Sigma_i w_{ij}$, α_i, β_j and γ_i are unknown parameters to be estimated. Parameters estimates $\hat{\hat{\alpha}}_i, \hat{\hat{\beta}}_j, \hat{\hat{\gamma}}_i$ and hence $\hat{\hat{m}}_{ij}$ are resulted from the following iterative procedure.

1. $k = 0, \alpha_i^{(0)} = \beta_j^{(0)} = \gamma_i^{(0)} = ,1 m_{ij}^{(0)} = \overline{w}_{ij}^{-1}$

2. $\alpha_i^{(k+1)} = \dfrac{n_{.j}}{\sum_j \overline{w}_{ij}^{-1} \beta_j^{(k)} \left(\gamma_i^{(k)}\right)^{(s_j - \overline{s})}} = \alpha_i^{(k)} \dfrac{n_{i.}}{\sum_j m_{ij}^{(k)}}$

3. $\beta_j^{(k+1)} = \dfrac{n_{.j}}{\sum_i \overline{w}_{ij}^{-1} \alpha_i^{(k+1)} \left(\gamma_i^{(k)}\right)^{(s_j - \overline{s})}}$

4. $m_{ij}^{*} = \overline{w}_{ij}^{-1} \alpha_i^{(k+1)} \beta_j^{(k+1)} \left(\gamma_i^{(k)}\right)^{(s_j - \overline{s})}, \ G_i = 1 + \dfrac{\sum_j (s_j - \overline{s})\left(n_{ij} - m_{ij}^{*}\right)}{\sum_j (s_j - \overline{s})^2 m_{ij}^{*}}$

5. $\gamma_i^{(k+1)} = \begin{cases} \gamma_i^{(k)} G_i & G_i > 0 \\ \gamma_i^{(k)} & \text{otherwise} \end{cases}$

6. $m_{ij}^{(k+1)} = \overline{w}_{ij}^{-1} \alpha_i^{(k+1)} \beta_j^{(k+1)} \left(\gamma_i^{(k+1)}\right)^{(s_j - \overline{s})}$

7. If $\max_{i,j} \left| m_{ij}^{(k+1)} - m_{ij}^{(k)} \right| < \epsilon$, stop and output $\alpha_i^{(k+1)}, \beta_j^{(k+1)}, \gamma_i^{(k+1)}$ and $m_{ij}^{(k+1)}$ as the final estimates. Otherwise, k=k+1, go to step 2.

Bonferroni Adjustments

The adjusted *p*-value is calculated as the *p*-value times a Bonferroni multiplier. The Bonferroni multiplier adjusts for multiple tests.

CHAID

Suppose that a predictor variable originally has *I* categories, and it is reduced to *r* categories after the merging step. The Bonferroni multiplier *B* is the number of possible ways that *I* categories can be merged into *r* categories. For *r* = *I*, *B* = 1. For 2≤*r*<*I*, use the following equation.

$$B = \begin{cases} \binom{I-1}{r-1} & \text{Ordinal predictor} \\ \sum_{v=0}^{r-1} (-1)^v \frac{(r-v)^I}{v!\,(r-v)!} & \text{Nominal predictor} \\ \binom{I-2}{r-2} + r\binom{I-2}{r-1} & \text{Ordinal with a missing category} \end{cases}$$

Exhaustive CHAID

Exhaustive CHAID merges two categories iteratively until only two categories left. The Bonferroni multiplier *B* is the sum of number of possible ways of merging two categories at each iteration.

$$B = \begin{cases} \frac{I(I-1)}{2} & \text{Ordinal predictor} \\ \frac{I(I^2-1)}{2} & \text{Nominal predictor} \\ \frac{I(I-1)}{2} & \text{Ordinal with a missing category} \end{cases}$$

Missing Values

If the dependent variable of a case is missing, it will not be used in the analysis. If all predictor variables of a case are missing, this case is ignored. If the case weight is missing, zero, or negative, the case is ignored. If the frequency weight is missing, zero, or negative, the case is ignored.

Otherwise, missing values will be treated as a predictor category. For ordinal predictors, the algorithm first generates the “best” set of categories using all non-missing information from the data. Next the algorithm identifies the category that is most similar to the missing category. Finally, the algorithm decides whether to merge the missing category with its most similar category or to keep the missing category as a separate category. Two *p*-values are calculated, one for the set of categories formed by merging the missing category with its most similar category, and the other for the set of categories formed by adding the missing category as a separate category. Take the action that gives the smallest *p*-value.

For nominal predictors, the missing category is treated the same as other categories in the analysis.

References

Biggs, D., B. de Ville, and E. Suen. 1991. A method of choosing multiway partitions for classification and decision trees. *Journal of Applied Statistics*, 18, 49–62.

Goodman, L. A. 1979. Simple models for the analysis of association in cross-classifications having ordered categories. *Journal of the American Statistical Association*, 74, 537–552.

Kass, G. 1980. An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29:2, 119–127.

QUEST Algorithms

QUEST is proposed by Loh and Shih (1997) as a Quick, Unbiased, Efficient, Statistical Tree. It is a tree-structured classification algorithm that yields a binary decision tree. A comparison study of QUEST and other algorithms was conducted by Lim et al (2000).

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The dependent, or target, variable. It must be nominal categorical. If Y is categorical with J classes, its class takes values in $C = \{1, \ldots, J\}$.
X_m, m=1, ..., M	The set of all predictor variables. A predictor can be nominal categorical or continuous (including ordinal categorical).
$\hbar = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$	The whole learning sample.
$\hbar(t)$	The learning samples that fall in node t.
f_n	The frequency weight associated with case n. Non-integral positive value is rounded to its nearest integer.
N_f	Total number of learning cases, $N_f = \sum_{n\in\hbar} f_n$
$N_{f,j}$	Total number of class j learning cases, $N_{f,j} = \sum_{n\in\hbar} f_n I(y_n = j)$
$N_f(t)$	Total number of learning cases in node t, $N_f(t) = \sum_{n\in\hbar(t)} f_n$
$N_{f,j}(t)$	Total number of class j learning cases in node t, $N_{f,j}(t) = \sum_{n\in\hbar(t)} f_n I(y_n = j)$.
$\pi(j)$, j=1,...,J	Prior probability of $Y = j$, $j = 1, \ldots, J$.
$p(j,t)$, j=1,...,J	The probability of a case in class j and node t.
$p(t)$	The probability of a case in node t.
$p(j\|t)$, j=1,...,J	The probability of a case in class j given that it falls into node t.
$C(i\|j)$	The cost of miss-classifying a class j case as a class i case. $C(j\|j)$=0

Tree Growing Process

The QUEST tree growing process consists of the selection of a split predictor, selection of a split point for the selected predictor, and stopping. In this algorithm, only univariate splits are considered.

Selection of Split Predictor

1. For each continuous predictor X, perform an ANOVA F test that tests if all the different classes of the dependent variable Y have the same mean of X, and calculate the p-value according to the

F statistics. For each categorical predictor, perform a Pearson's chi-square test of *Y* and *X*'s independence, and calculate the *p*-value according to the chi-square statistics.

2. Find the predictor with the smallest *p*-value and denote it X^*.
3. If this smallest *p*-value is less than α / M, where $\alpha \in (0,1)$ is a user-specified level of significance and *M* is the total number of predictor variables, predictor X^* is selected as the split predictor for the node. If not, go to 4.
4. For each continuous predictor *X*, compute a Levene's *F* statistic based on the absolute deviation of *X* from its class mean to test if the variances of *X* for different classes of *Y* are the same, and calculate the *p*-value for the test.
5. Find the predictor with the smallest *p*-value and denote it as X^{**}.
6. If this smallest *p*-value is less than $\alpha/(M + M_1)$, where M_1 is the number of continuous predictors, X^{**} is selected as the split predictor for the node. Otherwise, this node is not split.

ANOVA F Test

Suppose, for node *t*, there are J_t classes of dependent variable *Y*. The *F* statistic for a continuous predictor *X* is given by

$$F_X = \frac{\sum_{j=1}^{J_t} N_{f,j}(t)\left(\overline{x}^{(j)}(t) - \overline{x}(t)\right)^2 / (J_t - 1)}{\sum_{n \in \hbar(t)} f_n\left(x_n - \overline{x}^{(y_n)}(t)\right)^2 / \left(N_f(t) - J_t\right)}$$

where

$$\overline{x}^{(j)}(t) = \frac{\sum_{n\in\hbar(t)} f_n x_n I(y_n = j)}{N_{f,j}(t)}, \quad \overline{x}(t) = \frac{\sum_{n\in\hbar(t)} f_n x_n}{N_f(t)}$$

Its corresponding *p*-value is given by

$$p_X = \Pr\left(F\left(J_t - 1, N_f(t) - J_t\right) > F_X\right)$$

where $F(J_t - 1, N_f(t) - J_t)$ follows an *F* distribution with $J_t - 1$ and $N_f(t) - J_t$ degrees of freedom.

Pearson's Chi-Square Test

Suppose, for node *t*, there are J_t classes of dependent variable *Y*. The Pearson's Chi-Square statistic for a categorical predictor *X* with I_t categories is given by

$$X^2 = \sum_{j=1}^{J_t} \sum_{i=1}^{I_t} \frac{(n_{ij} - \hat{m}_{ij})^2}{\hat{m}_{ij}}$$

where

$$n_{ij} = \sum_{n \in \hbar(t)} f_n I(y_n = j \wedge x_n = i),\ \hat{m}_{ij} = \frac{n_{i.} n_{.j}}{n_{..}}$$

with

$$n_{i.} = \sum_{j=1}^{J_t} n_{ij},\ n_{.j} = \sum_{i=1}^{I_t} n_{ij},\ n_{..} = \sum_{j=1}^{J_t} \sum_{i=1}^{I_t} n_{ij}$$

where $I(y_n = j \wedge x_n = i)$=1 if case *n* has $y_n = j$ and $x_n = i$; 0 otherwise.

The corresponding *p*-value is given by $p_X = \Pr\left(\chi_d^2 > X^2\right)$ where χ_d^2 follows a chi-squared distribution with degrees of freedom *d* = (J_t−1)(I_t−1).

Levene's F Test

For continuous predictor *X*, calculate $z_n = \left|x_n - \overline{x}_{\cdot}^{(y_n)}(t)\right|$. The Levene's *F* statistics for predictor *X* is the ANOVA *F* statistic for z_n.

Selection of Split Point

At a node, suppose that a predictor variable *X* has been selected for splitting. The next step is to determine the split point. If *X* is a continuous predictor variable, a split point *d* in the split *X*≤*d* is to be determined. If *X* is a nominal categorical predictor variable, a subset *K* of the set of all values taken by *X* in the split *X*∈*K* is to be determined. The algorithm is as follows.

Continuous Splitting Predictor

If the selected predictor variable *X* is continuous:

1. Group classes of dependent variable *Y* into two super-classes. If there are only two classes of *Y*, go to step 2. Otherwise, calculate the sample mean of *X* for each class of *Y*. If all class means are identical, the class with the most cases is gathered as super-class *A* and the other classes as super-class *B*. If there are two or more classes with the same maximum number of cases, the one with the smallest class index *j* is chosen to form *A* and the rest to *B*. If not all the class means are identical, a *k*-means clustering method, with the initial cluster centers set at the two most extreme class means, is applied to class means to divide classes of *Y* into two super-classes: *A* and *B*. Let $\overline{x}_A$ and s_A^2 denote the sample mean and variance for super-class *A*, $\overline{x}_B$ and s_B^2 the sample mean and variance for super-class *B*.
2. If $\min\left(s_A^2, s_B^2\right) = 0$, order the two super-classes by their variance in increasing order and denote the variances by $s_1^2 \le s_2^2$, and the corresponding means by $\overline{x}_1, \overline{x}_2$. Let ε be a very small positive number, say ε=10^{-12}. If $\overline{x}_1 < \overline{x}_2$, $d = \overline{x}_1(1+\epsilon)$. Else, $d = \overline{x}_1(1-\epsilon)$.

3. If $\min\left(s_A^2, s_B^2\right) \neq 0$, quadratic discriminant analysis (QDA) is applied to determine the split point *d*. QDA assumes that *X* follows a normal distributions in each super-class with the calculated sample mean and variance. The split point is among the roots that make probability $\Pr(x, A|t) = \Pr(x, B|t)$ for node *t*, where

$$\Pr(x, A|t) = P(x|A,t)\,P(A|t) = P(A|t)\,\frac{1}{\sqrt{2\pi s_A^2}}\exp\left\{-\frac{(x-\overline{x}_A)^2}{2s_A^2}\right\}$$

with

$$p(A|t) = \sum_{j\in A} p(j|t) = \sum_{j\in A}\frac{p(j,t)}{\sum_j p(j,t)},\ p(j,t) = \frac{\pi(j)N_{f,j}(t)}{N_{f,j}}$$

Solving $P(X, A|t) = P(X, B|t)$ is equivalent to solving the following quadratic equation

$$ax^2 + bx + c = 0$$

where

$$a = s_A^2 - s_B^2,\ b = 2\left(\overline{x}_A s_B^2 - \overline{x}_B s_A^2\right)$$

$$c = \overline{x}_B s_A^2 - \overline{x}_A s_B^2 + 2s_A^2 s_B^2 \log\frac{p(A|t)s_B}{p(B|t)s_A}$$

If there is only one real root, it is chosen to be the split point, provided this yields two non-empty nodes. If there are two real roots, choose the one that is closer to $\overline{x}_A$, provided this yields two non-empty nodes. Otherwise use the mean $(\overline{x}_A + \overline{x}_B)/2$ as split point.

Note: In step 3, the prior probability distribution for the dependent variable is needed. When user specified costs are involved, the altered priors can be used to replace the priors (optional). The altered prior is defined as $\pi'(j) = \frac{C(j)\pi(j)}{\sum_j C(j)\pi(j)}$, where *C*(*j*) = $\Sigma_i C(i|j)$.

Nominal Splitting Predictor

If the selected predictor variable *X* is nominal and with more than two categories (if *X* is binary, the split point is clear), QUEST first transforms it into a continuous variable (call it ξ) by assigning the largest discriminant coordinates to categories of the predictor. QUEST then applies the split point selection algorithm for continuous predictor on ξ to determine the split point.

Transforming a Categorical Predictor into a Continuous Predictor

Let *X* be a nominal categorical predictor taking values in the set $\{b_1, \ldots, b_I\}$. Transform *X* into a continuous variable ξ such that the ratio of between-classes to within-classes sum squares of ξ is maximized (the classes here refer to the classes of dependent variable). The details are as follows:

- Transform each value *x* of *X* in ℏ into an *I*-dimensional dummy vector $v = (v_1, \ldots, v_I)'$, where $v_i = \begin{cases} 1 & x = b_i \\ 0 & \text{otherwise} \end{cases}$.
- Calculate the overall and class *j* mean of **v**

$$\overline{v} = \frac{\sum_{n \in \hbar} f_n v_n}{N_f}, \ \overline{v}^{(j)} = \frac{\sum_{n \in \hbar} f_n v_n I(y_n = j)}{N_{f,j}}$$

- Calculate the following $I \times I$ matrices.

$$\mathbf{B} = \sum_{j=1}^{J} N_{f,j} \left(\overline{v}^{(j)} - \overline{v} \right) \left(\overline{v}^{(j)} - \overline{v} \right)'$$
$$\mathbf{T} = \sum_{n \in \hbar} f_n (v_n - \overline{v})(v_n - \overline{v})'$$

- Perform single value decomposition on **T** to obtain **T** = **QDQ**', where **Q** is an $I \times I$ orthogonal matrix, **D** = diag(d_1, …, d_I) such that $d_1 \geq \ldots \geq d_I \geq 0$. Let $D^{-\frac{1}{2}}$ = diag(d_1^* , …, d_I^*) where $d_i^* = d_i^{-1/2}$ if $d_i > 0$, 0 otherwise. Perform single value decomposition on $D^{-\frac{1}{2}} Q' B Q D^{-\frac{1}{2}}$ to obtain its eigenvector a which is associated with its largest eigenvalue.
- The largest discriminant coordinate of **v** is the projection

$$\xi = a' D^{-\frac{1}{2}} Q' v$$

Note: The original QUEST by Loh and Shih (1997) transforms a categorical predictor into a continuous predictor at a considered node based on the data in the node. This implementation of QUEST does the transformation only once at the very beginning based on the whole learning sample.

Stopping

The stopping step checks if the tree growing process should be stopped according to the following stopping rules.

1. If a node becomes pure; that is, all cases belong to the same dependent variable class at the node, the node will not be split.
2. If all cases in a node have identical values for each predictor, the node will not be split.
3. If the current tree depth reaches the user-specified maximum tree depth limit value, the tree growing process will stop.
4. If the size of a node is less than the user-specified minimum node size value, the node will not be split.
5. If the split of a node results in a child node whose node size is less than the user-specified minimum child node size value, the node will not be split.

Missing Values

If the dependent variable of a case is missing, this case will be ignored in the analysis. If all predictor variables of a case are missing, this case will be ignored. If the frequency weight is missing, zero or negative, the case will be ignored.

Otherwise, the surrogate split method will be used to deal with missing data in predictor variables. If a case has a missing value at the selected predictor, the assignment will be done based on the surrogate split. The method of defining and calculating surrogate splits is the same as that in CART. For more information, see the topic "Missing Value Handling".

References

Lim, T. S., W. Y. Loh, and Y. S. Shih. 2000. A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms. *Machine Learning*, 40:3, 203–228.

Loh, W. Y., and Y. S. Shih. 1997. Split selection methods for classification trees. *Statistica Sinica*, 7, 815–840.

Assignment and Risk Estimation Algorithms

This section discusses how a class or a value is assigned to a node and to a case and three methods of risk estimation: the resubstitution method, test sample method and cross validation method. The information is applicable to the tree growing algorithms CART, CHAID, exhaustive CHAID and QUEST. Materials in this document are based on Classification and Regression Trees by Breiman, et al (1984). It is assumed that a CART, CHAID, exhaustive CHAID or QUEST tree has been grown successfully using a learning sample.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The dependent variable, or target variable. It can be either categorical (nominal or ordinal) or continuous. If Y is categorical with J classes, its class takes values in $C = \{1, \dots, J\}$.
$\hbar = \{\mathbf{x}_n, y_n\}_{n=1}^{N}$	The learning sample where $\mathbf{x}_n$ and y_n are the predictor vector and dependent variable for case n.
$\hbar(t)$	The learning samples that fall in node t.
f_n	The frequency weight associated with case n. Non-integral positive value is rounded to its nearest integer.
w_n	The case weight associated with case n.
$\pi(j)$, j=1,...,J	Prior probability of $Y = j$
$C(i \mid j)$	The cost of miss-classifying a class j case as a class i case, $C(j \mid j)$=0.

Assignment

Once the tree is grown, an assignment (also called action or decision) is given to each node based on the learning sample. To predict the dependent variable value for an incoming case, we first find in which terminal node it falls, then use the assignment of that terminal node for prediction.

Assignment of a Node

For any node *t*, let d_t be the assignment given to node *t*,

$$d_t = \begin{cases} j^*(t) & Y \text{ is categorical} \\ \overline{y}(t) & Y \text{ is continuous} \end{cases}$$

$$j^*(t) = arg \min_i \Sigma_j C(i|j)\, p(j|t)$$

$$\overline{y}(t) = \frac{1}{N_w(t)} \sum_{n \in \hbar(t)} w_n f_n y_n$$

where

$$p(j|t) = \frac{p(j,t)}{\sum_j p(j,t)},\ p(j,t) = \pi(j)\,\frac{N_{w,j}(t)}{N_{w,j}}$$

$$N_w = \sum_{n \in \hbar} w_n f_n,\ N_{w,j} = \sum_{n \in \hbar} w_n f_n I(y_n = j)$$

$$N_w(t) = \sum_{n \in \hbar(t)} w_n f_n,\ N_{w,j}(t) = \sum_{n \in \hbar(t)} w_n f_n I(y_n = j)$$

If there is more than one class *j* that achieves the minimum, choose $j^*(t)$ to be the smallest such *j* for which $N_{f,j}(t) = \sum_{n \in \hbar(t)} f_n I(y_n = j)$ is greater than 0, or the absolute smallest if $N_{f,j}(t)$ is zero for all of them. For CHAID and exhaustive CHAID, use $\pi(j) = N_{w,j}/N_w$ in the equation.

Assignment of a Case

For a case with predictor vector **x**, the assignment or prediction $d_T(\mathbf{x})$ for this case by the tree *T* is

$$d_T(\mathbf{x}) = \begin{cases} j^*(t(\mathbf{x})) & Y \text{ is categorical} \\ \overline{y}(t(\mathbf{x})) & Y \text{ is continuous} \end{cases}$$

where $t(\mathbf{x})$ is the terminal node the case falls in.

Risk Estimation

Note that case weight is not involved in risk estimation, though it is involved in tree growing process and class assignment.

Loss Function

A loss function $L(y, a)$ is a real-valued function in which *y* is the actual value of *Y* and *a* is the assignment taken. Throughout this document, the following types of loss functions are used.

$$L(y,a)=\begin{cases} C(a|y) & Y \text{ is categorical} \\ (y-a)^2 & Y \text{ is continuous} \end{cases}$$

Risk Estimation of a Tree

Suppose that a tree *T* is grown and assignments have been given to each node. Let $\tilde{T}$ denote the set of terminal nodes of the tree. Let *D* be the data set used to calculate the risk. Dropping all cases in *D* to *T*, let *D*(*t*) denote the set of cases that fall in node *t*. The risk of the tree based on data *D* is estimated by

$$R(T|D)=\begin{cases} \sum_j \pi(j)\overline{L}_j & Y \text{ categorical} \\ \overline{L} & Y \text{ continuous} \end{cases} = \begin{cases} \overline{L} & Y \text{ categorical, M1} \\ \sum_j \pi(j)\overline{L}_j & Y \text{ categorical, M2} \\ \overline{L} & Y \text{ continuous} \end{cases}$$

where M1 represents empirical prior situation, and M2 non-empirical prior, and

$$\overline{L}=\tfrac{1}{N_f}\sum_{n\in D} f_n L(y_n,d_T(x_n)),\ \overline{L}_j=\tfrac{1}{N_{f,j}}\sum_{n\in D} f_n L(y_n,d_T(x_n))\,I(y_n=j)$$

$$N_f=\sum_{n\in D} f_n,\ N_{f,j}=\sum_{n\in D} f_n I(y_n=j)$$

Assuming that $L(y_n,d_T(x_n))$ are independent of each other, then the variance of *R*(*T*) is estimated by

$$\text{Var}(R(T))=\begin{cases} \sum_j \pi(j)^2 \dfrac{s_j^2}{N_{f,j}} & Y \text{ categorical, M2} \\ \frac{s^2}{N_f} & Y \text{ con, or, } Y \text{ cat and M1} \end{cases}$$

where

$$\begin{aligned} s_j^2 &= \tfrac{1}{N_{f,j}}\sum_{n\in D} f_n\big(L(y_n,d_T(x_n))-\overline{L}_j\big)^2 I(y_n=j) \\ &= \tfrac{1}{N_{f,j}}\sum_{n\in D} f_n L^2(y_n,d_T(x_n))\,I(y_n=j)-\overline{L}_j \end{aligned}$$

$$s^2=\frac{1}{N_f}\sum_{n\in D} f_n\big(L(y_n,d_T(x_n))-\overline{L}\big)^2=\frac{1}{N_f}\sum_{n\in D} f_n L^2(y_n,d_T(x_n))-\overline{L}^2$$

Putting everything together:

$$R(T|D) = \begin{cases} \frac{1}{N_f}\sum_{t\in\tilde{T}}\sum_{j} C(j^*(t)|j)\, N_{f,j}(t) & Y \text{ categorical, M1} \\ \sum_{j}\frac{\pi(j)}{N_{f,j}}\sum_{t\in\tilde{T}} C(j^*(t)|j)\, N_{f,j}(t) & Y \text{ categorical, M2} \\ \frac{1}{N_f}\sum_{t\in\tilde{T}}\sum_{n\in D(t)} f_n(y_n - \overline{y}(t))^2 & Y \text{ continuous} \end{cases}$$

$$\text{Var}(R(T|D)) = \begin{cases} \frac{1}{(N_f)^2}\left\{\sum_{j}\sum_{t\in\tilde{T}} N_{f,j}(t)\, C(j^*(t)|j)^2 - N_f R(T|D)^2\right\} & Y \text{ cat, M1} \\ \sum_{j}\left(\frac{\pi(j)}{N_{f,j}}\right)^2\left[\sum_{t\in\tilde{T}} N_{f,j}(t)\, C(j^*(t)|j)^2 - \frac{\left\{\sum_{t\in\tilde{T}} N_{f,j}(t)\, C(j^*(t)|j)\right\}^2}{N_{f,j}}\right] & Y \text{ cat, M2} \\ \frac{1}{N_f}\left\{\sum_{t\in\tilde{T}}\sum_{n\in D(t)} f_n(y_n - \overline{y}(t))^4 - N_f R(T|D)^2\right\} & Y \text{ con} \end{cases}$$

where

$$N_{f,j}(t) = \sum_{n\in D(t)} f_n I(y_n = j)$$

The estimated standard error of *R*(*T*|*D*) is given by se $(R(T|D)) = \sqrt{var(R(T|D))}$.

Risk estimation of a tree is often written as $R(T|D) = \sum_{t\in\tilde{T}} R(t|D)$ with $R(t|D)$ being the contribution from node *t* to the tree risk such that

$$R(t|D) = \begin{cases} \frac{1}{N_f}\sum_{j} N_{f,j}(t)\, C(j^*(t)|j) & Y \text{ categorical, M1} \\ \sum_{j}\frac{\pi(j)\, N_{f,j}(t)}{N_{f,j}} C(j^*(t)|j) & Y \text{categorical, M2} \\ \frac{1}{N_f}\sum_{n\in D(t)} f_n(y_n - \overline{y}(t))^2 & Y \text{ continuous} \end{cases}$$

Resubstitution Estimate of the Risk

The resubstitution risk estimation method uses the same set of data (learning sample) that is used to grow the tree *T* to calculate its risk, that is:

$$R(t) = R(t|\hbar)$$
$$R(T) = R(T|\hbar) = \sum_{t \in \tilde{T}} R(t)$$
$$\text{Var}(R(T)) = \text{Var}(R(T|\hbar))$$

Test Sample Estimate of the Risk

The idea of test sample risk estimation is that the whole data set is divided into 2 mutually exclusive subsets $\hbar$ and $\hbar'$. $\hbar$ is used as a learning sample to grow a tree *T* and $\hbar'$ is used as a test sample to check the accuracy of the tree. The test sample estimate is

$$R^{ts}(T) = R\left(T|\hbar'\right)$$

$$\text{Var}\left(R^{ts}(T)\right) = \text{Var}\left(R\left(T|\hbar'\right)\right)$$

Cross Validation Estimate of the Risk

Cross validation estimation is provided only when a tree is grown using the automatic tree growing process. Let *T* be a tree which has been grown using all data from the whole data set $\hbar^0$. Let $V \geq 2$ be a positive integer.

1. Divide $\hbar^0$ into *V* mutually exclusive subsets $\hbar'_v$, *v* = 1, …, *V*. Let $\hbar_v$ be $\hbar^0 - \hbar'_v$, *v* = 1, …, *V*.

2. For each *v*, consider $\hbar_v$ as a learning sample and grow a tree T_v on $\hbar_v$ by using the same set of user specified stopping rules which was applied to grow *T*.

3. After T_v is grown and assignment $j_v^*(t)$ or $\overline{y}_v(t)$ for node *t* of T_v is done, consider $\hbar'_v$ as a test sample and calculate its test sample risk estimate $R^{ts}(T_v)$.

4. Repeat above for each *v* = 1, …, *V*. The weighted average of these test sample risk estimates is used as the *V*-fold cross validation risk estimate of *T*.

The *V*-fold cross validation estimate, $R^{cv}(T)$, of the risk of a tree *T* and its variance are estimated by

$$R^{CV}(T) = \begin{cases} \sum_j \pi(j) \frac{1}{N^0_{f,j}} \sum_v N'_{v,f,j} R^{ts}(T_v|j) & \text{Y categorical, M2} \\ \frac{1}{N^0_f} \sum_v N'_{v,f} R^{ts}(T_v) & Y \text{ con, or, Y cat and M1} \end{cases}$$

$$\mathrm{Var}\left(R^{CV}\left(T\right)\right)$$

$$= \begin{cases} \frac{1}{\left(N_f^0\right)^2}\left\{\sum_v \sum_j \sum_{t\in\tilde{T}_v} N'_{v,f,j}\left(t\right) C(j_v^*\left(t\right)|j)^2 - N_f^0 R^{cv}(T)^2\right\} & Y \text{ cat, M1} \\ \sum_j \left(\frac{\pi\left(j\right)}{N_{f,j}^0}\right)^2 \left[\sum_v \sum_{t\in\tilde{T}_v} N'_{v,f,j}\left(t\right) C(j_v^*\left(t\right)|j)^2 - \frac{\left\{\sum_v N'_{v,f,j} R^{ts}\left(T_v|Y=j\right)\right\}^2}{N_{f,j}^0}\right] & Y \text{ cat, M2} \\ \frac{1}{\left(N_f^0\right)^2}\left\{\sum_v \sum_{t\in\tilde{T}_v} \sum_{n\in\hbar'_v(t)} f_n(y_n - \overline{y}_v\left(t\right))^4 - N_f^0 R^{cv}(T)^2\right\} & Y \text{ con} \end{cases}$$

where

$$R^{ts}\left(T_v|j\right) = \frac{1}{N'_{v,f,j}} \sum_{t\in\tilde{T}_v} N'_{v,f,j}\left(t\right) C\left(j_v^*\left(t\right)|j\right)$$

$$N_f^0 = \sum_{n\in\hbar^0} f_n,\ N_{f,j}^0 = \sum_{n\in\hbar^0} f_n I\left(y_n = j\right)$$

$$N'_{v,f} = \sum_{n\in\hbar'_v} f_n,\ N'_{v,f,j} = \sum_{n\in\hbar'_v} f_n I\left(y_n = j\right),\ N'_{v,f,j}\left(t\right) = \sum_{n\in\hbar'_v(t)} f_n I\left(y_n = j\right)$$

References

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees*. New York: Chapman & Hall/CRC.

Gain Summary Algorithms

The Gain Summary summarizes a tree by displaying descriptive statistics for each terminal node. This allows users to recognize the relative contribution of each terminal node and identify the subsets of terminal nodes that are most useful. This document can be used for all tree growing algorithms CART, CHAID, exhaustive CHAID and QUEST.

Note that case weight is not involved in gain summary calculations though it is involved in tree growing process and class assignment.

Types of Gain Summaries

Depending on the type of dependent variable, different statistics are given in the gain summary.

Average Oriented Gain Summary (Y continuous). Statistics related to the node mean of *Y* are given. Through this summary, users may identify the terminal nodes that give the largest (or smallest) average of the dependent variable.

Target Class Gain Summary (Y categorical). Statistics related to an interested dependent variable class (target class) are given. Users may identify the terminal nodes that have a large relative contribution to the target class.

Average Profit Value Gain Summary (Y categorical). Statistics related to average profits are given. Users may be interested in identifying the terminal nodes that have relatively large average profit values.

Node-by-Node, Cumulative, Percentile Gain Summary. To assist users in identifying the interesting terminal nodes and in understanding the result of a tree, three different ways (node-by-node, cumulative and percentile) of looking at the gain summaries mentioned above are provided.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y	The dependent, or target, variable. It can be either categorical (nominal or ordinal) or continuous. If Y is categorical with J classes, its class takes values in $C = \{1, \ldots, J\}$.
D	Data set used to calculate gain statistics. It can be either learning sample data set or test sample data set.
$D(t)$	Cases in D in node t.
y_n	The dependent variable value for case n.
f_n	The frequency weight associated with case n. Non-integral positive value is rounded to its nearest integer.
N_f	The number of cases in D, $N_f = \sum_{n \in D} f_n$
$N_f(t)$	The number of cases in $D(t)$, $N_f(t) = \sum_{n \in D(t)} f_n$
$N_{f,j}$	The number of class j cases in D, $N_{f,j} = \sum_{n \in D} f_n I(y_n = j)$
$N_{f,j}(t)$	The number of class j cases in $D(t)$, $N_{f,j}(t) = \sum_{n \in D(t)} f_n I(y_n = j)$
$\overline{y}(t)$	The mean of dependent variable in $D(t)$, $\overline{y}(t) = \frac{1}{N_f(t)} \sum_{n \in D(t)} f_n y_n$
j''	Target class of interest; it is any value in $\{1, \ldots, J\}$. If not user-specified, the default target class is 1.
$r(j)$, $e(j)$	Respectively, the revenue and expense associated with class j.
$pv(j)$	The profit value associated with class j, $pv(j) = r(j) - e(j)$
$j^*(\tilde{t})$	Class assignment given by terminal node $\tilde{t}$.
$\pi(j)$	Prior probability of class j, $j = 1, \ldots, J$.

M1	For categorical *Y*, denotes the empirical prior situation. CHAID and exhaustive CHAID are always considered as having an empirical prior.
M2	For categorical *Y*, denotes the non-empirical prior situation.

Node by Node Summary

The node-by-node gain summary includes statistics for each node that are defined as follows.

Terminal Node

The identity of a terminal node. It is denoted by $\tilde{t}$

Size: n

Total number of cases in the terminal node. It is denoted by $N_f(t)$.

Size: %

Percentage of cases in the node. It is denoted by $p_f(t)$100%, where $p_f(t)$ is given by

$$p_f\left(\tilde{t}\right) = \begin{cases} \frac{N_f(\tilde{t})}{N_f} & \text{M1, or, Y continuous} \\ \sum_j \frac{\pi(j)\, N_{f,j}\left(\tilde{t}\right)}{N_{f,j}} & \text{M2} \end{cases}$$

Gain: n

Total number of target class j'' cases in the node, $N_{f,j''}\left(\tilde{t}\right)$.

This is only computed for the target class gain summary type.

Gain: %

Percentage of target class j'' cases in the sample that belong to the node. It is denoted by $p_f\left(\tilde{t}|j''\right)$100%, where

$$p_f\left(\tilde{t}|j''\right) = \frac{N_{f,j''}\left(\tilde{t}\right)}{N_{f,j''}}$$

This is only computed for the target class gain summary type.

Score

Depending on the type of gain summary, the score is defined and named differently. But they are all denoted by $s(\tilde{t})$.

Response: % (for target class gain summary only)

The ratio of the number of target class j'' cases in the node to the total number of cases in the node.

$$s\left(\tilde{t}\right) = \begin{cases} \frac{N_{f,j''}\left(\tilde{t}\right)}{N_f\left(\tilde{t}\right)} & \text{M1} \\ \frac{1}{p_f\left(\tilde{t}\right)} \cdot \frac{\pi\left(j''\right) N_{f,j''}\left(\tilde{t}\right)}{N_{f,j''}} & \text{M2} \end{cases}$$

Average Profit (for average profit value gain summary only)

The average profit value for the node.

$$s\left(\tilde{t}\right) = \begin{cases} \frac{\sum_j N_{f,j}\left(\tilde{t}\right) \cdot pv\left(j\right)}{N_f\left(\tilde{t}\right)} & \text{M1} \\ \frac{1}{p_f\left(\tilde{t}\right)} \cdot \sum_j \frac{\pi\left(j\right) N_{f,j}\left(\tilde{t}\right) \cdot pv\left(j\right)}{N_{f,j}} & \text{M2} \end{cases}$$

Mean (for average oriented gain summary only)

The respective mean of the continuous dependent variable *Y* at the node.

$$s\left(\tilde{t}\right) = \overline{y}\left(\tilde{t}\right)$$

ROI (Return on Investment)

ROI for a node is calculated as average profit divided by average expense.

$$ROI\left(\tilde{t}\right) = \frac{s\left(\tilde{t}\right)}{s_0\left(\tilde{t}\right)}$$

Where $s_0\left(\tilde{t}\right)$ is the average expense for node $\tilde{t}$ and is calculated using equation for $s\left(\tilde{t}\right)$ with *pv*(*j*) replaced by *e*(*j*).

This is only computed for the average profit value gain summary type.

Index (%)

For the target class gain summary, it is the ratio of the score for the node to the proportion of class j'' cases in the sample. It is denoted by *is* ($\tilde{t}$)100%, where *is* ($\tilde{t}$) is

$$is\left(\tilde{t}\right) = \begin{cases} \frac{s\left(\tilde{t}\right)}{N_{f,j''}/N_f} & \text{M1} \\ \frac{s\left(\tilde{t}\right)}{\pi\left(j''\right)} & \text{M2} \end{cases}$$

For the average profit value gain summary, it is the ratio of the score for the node to the average profit value for the sample.

$$is(\tilde{t}) = \begin{cases} \dfrac{s(\tilde{t})}{\sum_j N_{f,j} pv(j) / N_f} & \text{M1} \\ \dfrac{s(\tilde{t})}{\sum_j \pi(j) pv(j)} & \text{M2} \end{cases}$$

For the average oriented gain summary, it is the ratio of the gain score for the node to the gain score $s(t = 1)$ for root node $t = 1$.

$$is\left(\tilde{t}\right) = \frac{s\left(\tilde{t}\right)}{s\left(t = 1\right)}$$

Note: if the denominator is 0, the index is not available.

Cumulative Summary

In the cumulative gain summary, all nodes are first sorted with respect to the values of the score $s\left(\tilde{t}\right)$. To simplify the formulas, we assume that nodes in the collection { $\tilde{t}_1$, $\tilde{t}_2$, …, $\tilde{t}_{|\tilde{T}|}$ } are already sorted either in descending or ascending order.

Terminal Node

The identity of a terminal node. It is denoted by $\tilde{t}_s$.

Cumulative Size: n

$$\oplus N_f\left(\tilde{t}_s\right) = \sum_{i=1}^{s} N_f\left(\tilde{t}_s\right)$$

Cumulative Size: %

$$\oplus p_f\left(\tilde{t}_s\right) = \sum_{i=1}^{s} p_f\left(\tilde{t}_s\right)$$

Cumulative Gain: n

$$\oplus N_{f,j''}\left(\tilde{t}_s\right) = \sum_{i=1}^{s} N_{f,j''}\left(\tilde{t}_s\right)$$

Cumulative Gain: %

$$\oplus p_f\left(\tilde{t}_s|j''\right)=\sum_{i=1}^{s}p_f\left(\tilde{t}_s|j''\right)$$

Score

For Cumulative response, it is the ratio of the number of target class j'' cases up to the node to the total number of cases up to the node. For cumulative average profit, it is the average profit value up to the node. For cumulative mean, it is the mean of all y_n's up to the terminal nodes. In all cases, the same formula is used, but the appropriate formulas for $s(\tilde{t}_s)$ and $p_f(\tilde{t}_s)$ should be used in the calculations. This cumulative score is denoted by:

$$\oplus s(\tilde{t}_s)=\begin{cases}\dfrac{\sum_{i=1}^{s}s(\tilde{t}_i)\quad N_f\left(\tilde{t}_i\right)}{\sum_{i=1}^{s}N_f\left(\tilde{t}_i\right)} & \text{M1, or, Y continuous}\\[2ex] \dfrac{\sum_{i=1}^{s}s\left(\tilde{t}_i\right)\cdot p_f(\tilde{t}_i)}{\sum_{i=1}^{s}p_f\left(\tilde{t}_i\right)} & \text{M2}\end{cases}$$

Cumulative ROI

$$\oplus ROI\left(\tilde{t}_s\right)=\frac{\oplus s\left(\tilde{t}_s\right)}{\oplus s_0\left(\tilde{t}_s\right)}$$

Where $\oplus s_0\left(\tilde{t}_s\right)$ is the cumulative expense and calculated by equation for $\oplus s\left(\tilde{t}_s\right)$ with $pv\left(\tilde{t}\right)$ replaced by $e\left(\tilde{t}\right)$.

This is only computed for the average profit value gain summary type.

Cumulative Index %

For the target class cumulative gain summary, it is the ratio of the cumulative gain score for the node to the proportion of class j'' cases in the sample. It is denoted by $\oplus is(\tilde{t}_s)$100%, where:

$$\oplus is(\tilde{t}_s)=\begin{cases}\dfrac{\oplus s(\tilde{t}_s)}{N_{f,j''}/N_f} & \text{M1}\\[2ex] \dfrac{\oplus s(\tilde{t}_s)}{\pi(j'')} & \text{M2}\end{cases}$$

For the average profit value cumulative gain summary, it is the ratio of the cumulative gain score for the node to the average profit value for the sample.

$$\oplus is\left(\tilde{t}_s\right) = \begin{cases} \dfrac{\oplus s(\tilde{t}_s)}{\sum_j N_{f,j} \cdot pv\left(j\right)/N_f} & \text{M1} \\ \dfrac{\oplus \mathrm{s}(\tilde{t}_s)}{\sum_j \pi\left(j\right) \cdot pv\left(j\right)} & \text{M2} \end{cases}$$

For the average oriented cumulative gain summary, it is the ratio of the cumulative score for the node to the score *s*(*t* = 1) for root node *t* = 1.

$$\oplus is\left(\tilde{t}_s\right) = \frac{\oplus s\left(\tilde{t}_s\right)}{s\left(t=1\right)} = \sum_{i=1}^{s} is\left(\tilde{t}_i\right)$$

Note: if the denominator is 0, the index is not available.

Percentile Summary

Like cumulative gain summary, all nodes are first sorted with respect to the values of their scores. To simplify the formulas, we assume that nodes in the collection { $\tilde{t}_1$, $\tilde{t}_2$, …, $\tilde{t}_{|\tilde{T}|}$} are already sorted in either descending or ascending order. Let *q* be any positive integer divisible to 100. The value of *q* will be used as the percentage increment for percentiles, and is user-specified (default *q* = 10). For fixed q, the number of percentiles to be studied is 100/*q*. The *p*th percentile to be studied is the *pq*%-tile, and its size is $N_{f \cdot pq} = N_f \cdot pq\%$, *p* = 1, …, 100/*q*. For any *pq*%-tile, let s_p and s'_p be the two smallest integers in {1, …, $|\tilde{T}|$} such that

$$N_{f \cdot pq} \in \left(\oplus N_f\left(\tilde{t}_{s_p - 1}\right), \oplus N_f\left(\tilde{t}_{s_p}\right)\right], N_{f \cdot pq} \in \left[\oplus N_f\left(\tilde{t}_{s'_p - 1}\right), \oplus N_f\left(\tilde{t}_{s'_p}\right)\right)$$

where $\oplus N_f\left(\tilde{t}_0\right) \equiv 0$

Terminal Node

The identity of all terminal nodes that belong to the *p*th increment. Node $\tilde{t}$ belongs to the *p*th increment if $\tilde{t} \in \left[s'_{p-1}, s_p\right]$.

Percentile (%)

Percentile being studied. The *p*th percentile is the *pq*%-tile.

Percentile: n

Total number of cases in the percentile, $N^*_{f \cdot pq} = [N_f \cdot pq\%]$, where [*x*] denotes the nearest integer of *x*.

Gain: n

Total number of class j'' cases in the *pq*-percentile.

$$?N_{f,j''}(p) = \oplus N_{f,j''}\left(\tilde{t}_{s_p-1}\right) + \frac{N_{f\cdot pq} - \oplus N_f\left(\tilde{t}_{s_p-1}\right)}{N_f\left(t_{s_p}\right)} N_{f,j''}\left(\tilde{t}_{s_p}\right)$$

where $\oplus N_{f,j''}(\tilde{t}_0)$ is defined to be 0.

This is only computed for the target class percentile gain summary type.

Gain: %

Percentage of class j'' cases in the sample that belong to the *pq*%-tile. It is denoted by $?p_{f,j''}(p)$100%, where

$$?p_{f,j''}(p) = \frac{?N_{f,j''}(p)}{N_{f,j''}}$$

This is only computed for the target class percentile gain summary type.

Percentile Score

For the target class percentile gain summary, it is an estimate of the ratio of the number of class j'' cases in the *pq*-percentile to the total number of cases in the percentile. For the average profit value percentile gain summary, it is an estimate of the average profit value in the *pq*-percetile. For the average oriented percentile gain summary, it is an estimate of the average of the gain score for all nodes in the percentile. In all charts, the same formula is used.

$$?s(p) = \begin{cases} \dfrac{\oplus N_f\left(\tilde{t}_{s_p-1}\right)\cdot\oplus s\left(\tilde{t}_{s_p-1}\right)+\left\{N_{f\cdot pq}-\oplus N_f\left(\tilde{t}_{s_p-1}\right)\right\}\cdot s\left(\tilde{t}_{s_p}\right)}{N_{f\cdot pq}} & \text{M1} \\ \dfrac{\oplus p_f\left(\tilde{t}_{s_p-1}\right)\cdot\oplus s\left(\tilde{t}_{s_p-1}\right)+\left\{p_{f\cdot pq}-\oplus p_f\left(\tilde{t}_{s_p-1}\right)\right\}\cdot s\left(\tilde{t}_{s_p}\right)}{p_{f\cdot pq}} & \text{M2} \end{cases}$$

where

$$p_{f.pq} = \oplus p_f\left(\tilde{t}_{s_p-1}\right) + \frac{N_{f.pq} - \oplus N_f\left(\tilde{t}_{s_p-1}\right)}{N_f\left(\tilde{t}_{s_p}\right)} p_f\left(\tilde{t}_{s_p}\right)$$

Percentile ROI

$$?ROI(p) = \frac{?s(p)}{?s_0(p)}$$

where $?s_0(p)$ is the percentile expense and calculated through equation $?s(p)$ with $pv(\tilde{t})$ replaced by $e(\tilde{t})$.

This is only computed for the average profit value gain summary type.

Percentile Index

For the target class percentile gain summary, it is the ratio of the percentile gain score for the *pq*-percentile to the proportion of class j'' cases in the sample. It is denoted by $?is(p)$100 percent, where

$$?is(p) = \begin{cases} \frac{?s(p)}{N_{f,j''}/N_f} & \text{M1} \\ \frac{?\,\mathrm{s}(p)}{\pi(j'')} & \text{M2} \end{cases}$$

For the average profit value percentile gain summary, it is the ratio of the percentile gain score for the *pq*-percentile to the average of the profit values for the sample.

$$?is(p) = \begin{cases} \frac{?s(p)}{\sum_j N_{f,j} \cdot pv(j)/N_f} & \text{M1} \\ \frac{?s(p)}{\sum_j \pi(j) \cdot pv(j)} & \text{M2} \end{cases}$$

For the average oriented cumulative gain summary, it is the ratio of the percentile gain score in the *pq*-percentile to the gain score $s(t = 1)$ for root node $t = 1$.

$$?is(p) = \frac{?s(p)}{s(t=1)}$$

Note: if the denominator, which is the average or the median of y_n's in the sample, is 0, the index is not available.

Cost—Complexity Pruning Algorithms

Assuming a CART or QUEST tree has been grown successfully using a learning sample, this document describes the automatic cost-complexity pruning process for both CART and QUEST trees. Materials in this document are based on Classification and Regression Trees by Breiman et al (1984). Calculations of the risk estimates used throughout this document are given in *TREE Algorithms*.

Cost-Complexity Risk of a Tree

Given a tree *T* and a real number α, the cost-complexity risk of *T* with respect to α is

$$R_\alpha(T) = R(T) + \alpha\left|\tilde{T}\right|$$

where $\left|\tilde{T}\right|$ is the number of terminal nodes and *R*(*T*) is the resubstitution risk estimate of *T*.

Smallest Optimally Pruned Subtree

Pruned subtree. For any tree *T*, $T^{'}$ is a pruned subtree of *T* if $T^{'}$ is a tree with the same root node as *T* and all nodes of $T^{'}$ are also nodes of *T*. Denote $T^{'}\underline{?}T$ if $T^{'}$ is a pruned subtree of *T*.

Optimally pruned subtree. Given α, a pruned T'subtree of *T* is called an optimally pruned subtree of *T* with respect to α if $R_{\alpha}\left(T'\right) = \min_{T"?T} R_a(T")$. The optimally pruned subtree may not be unique

Smallest optimally pruned subtree. If $T^{'}\underline{?}T^{''}$ for any optimally pruned subtree $T^{''}\underline{?}T_0$ such that $R_{\alpha}\left(T^{'}\right) = R_{\alpha}\left(T^{''}\right)$, then $T^{'}$ is the smallest optimally pruned subtree of T_0 with respect to α, and is denoted by T_0(α).

Cost-Complexity Pruning Process

Suppose that a tree T_0 was grown. The cost-complexity pruning process consists of two steps:

1. Based on the learning sample, find a sequence of pruned subtrees $\{T_k\}_{k=0}^{K}$ of T_0 such that $T_0 \succ T_1 \succ T_2 \succ \ldots \succ T_K$, where T_K has only the root node of T_0.
2. Find an "honest" risk estimate $\hat{R}(T_k)$ of each subtree. Select a right sized tree from the sequence of pruned subtrees.

Generate a sequence of smallest optimally pruned subtrees

To generate a sequence of pruned subtrees, the cost-complexity pruning technique developed by Breiman et. al. (1984) is used. In generating the sequence of subtrees, only the learning sample is used. Given any real value $\alpha_{\min}$ ($\alpha_{\min} = 0$) and an initial tree T_0, there exists a sequence of real values $-\infty < \alpha_1 = \alpha_{\min} < \alpha_2 < \cdots < \alpha_K < +\infty$ and a sequence of pruned subtrees $T_0?T_1?\cdots?T_K$, such that the smallest optimally pruned subtree of T_0 for a given α is

$$T_0(\alpha) = \begin{cases} T_0 & \alpha < \alpha_1 \\ T_0(\alpha_k) = T_k & \alpha_k \le \alpha < \alpha_{k+1} \quad 1 \le k < K \\ T_0(\alpha_K) = T_K & \alpha_K \le \alpha \end{cases}$$

where

$\alpha_{k+1} = \min_{t \in T_k} g_k(t)$, $T_{k+1} = \{t \in T_k : g_k(s) > \alpha_{k+1}$for all ancestors of t$\}$

$$g_k(t) = \begin{cases} \frac{R(t) - R(T_{k,t})}{\left|\tilde{T}_{k,t}\right| - 1} & t \in T_k - \tilde{T}_k \\ +\infty & t \in \tilde{T}_k \end{cases}$$

$\tilde{T}_{k,t}$ is the branch of T_k stemming from node *t*, and *R*(*t*) is the resubstitution risk estimate of node *t* based on the learning sample.

Explicit algorithm

For node *t*, let

$$lt(t) = \begin{cases} 0 & t \text{ is terminal} \\ \text{left child of } t & \text{otherwise} \end{cases}$$

$$rt(t) = \begin{cases} 0 & t \text{ is terminal} \\ \text{right child of } t & \text{otherwise} \end{cases}$$

$$pa(t) = \begin{cases} 0 & t \text{ is root node} \\ \text{parent of } t & \text{otherwise} \end{cases}$$

$$\tilde{N}(t) = \begin{cases} 1 & t \text{ is terminal} \\ \left|\tilde{T}_{k,t}\right| & \text{otherwise} \end{cases}$$

$$S(t) = \begin{cases} R(t) & t \text{ is terminal} \\ R\left(T_{k,t}\right) & \text{otherwise} \end{cases}$$

$$G(t) = \min_{s \in T_{k,t}} g_k(s)$$

Then the explicit algorithm is as follows:

Initialization. Set k=1, $\alpha=\alpha_{\min}$

For $t=\#T_0$ to 1:

if *t* is a terminal node, set

$\tilde{N}(t)=1$, $S(t)=R(t)$, $g(t)=G(t)=+\infty$

else set

$\tilde{N}(t)=\tilde{N}(lt(t))+\tilde{N}(rt(t))$

$S(t) = S(lt(t)) + S(rt(t))$

$g(t) = (R(t) - S(t))/(\tilde{N}(t) - 1)$

$G(t) = \min\{g(t), G(lt(t)), G(rt(t))\}$

Main algorithm. If $G(1) > \alpha$, set

$\alpha_k = \alpha$ and $T_k = \{t \in T_{k-1}$: $g(s) > \alpha_k$ for all ancestor s of t$\}$

$\alpha=G(1)$, $k=k+1$

else if $\tilde{N}(1)=1$, terminate the process.

Set t=1.

While $G(t) < g(t)$, $t = \begin{cases} lt(t) & G(t) = G(lt(t)) \\ rt(t) & \text{otherwise} \end{cases}$

Make the current node t terminal by setting $\tilde{N}(t)=1$, $S(t)=R(t)$, $g(t)=G(t)=+\infty$

Update ancestor's information of current node t; while $t>1$

$t=pa(t)$

$\tilde{N}(t)= \tilde{N}(lt(t))+ \tilde{N}(rt(t))$

$S(t) = S(lt(t)) + S(rt(t))$

$g(t) = (R(t) - S(t))/(\tilde{N}(t) - 1)$

$G(t) = \min\{g(t), G(lt(t)), G(rt(t))\}$

Repeat the main algorithm until the process terminates.

Selecting the Right Sized Subtree

To select the right sized pruned subtree from the sequence of pruned subtrees $\{T_k\}_{k=0}^{K}$ of T_0, an "honest" method is used to estimate the risk $\hat{R}(T_k)$ and its standard error $se\left(\hat{R}(T_k)\right)$ of each subtree T_k. Two methods can be used: the resubstitution estimation method and the test sample estimation method. Resubstitution estimation is used if there is no test sample. Test sample estimation is used if there is a testing sample. Select the subtree T_{k*} as the right sized subtree of T_0 based on one of the following rules.

Simple rule

The right sized tree is selected as the $k^* \in \{0, 1, 2, \ldots, K\}$ such that

$$\hat{R}(T_{k^*}) = \min_k \hat{R}(T_k)$$

b-SE rule

For any nonnegative real value b (default $b = 1$), the right sized tree is selected as the largest $k^{**} \in \{0, 1, 2, \ldots, K\}$ such that

$$\hat{R}(T_{k^{**}}) \le \hat{R}(T_{k^*}) + b \quad se\left(\hat{R}(T_{k^*})\right)$$

References

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees*. New York: Chapman & Hall/CRC.

TSMODEL Algorithms

The TSMODEL procedure builds univariate exponential smoothing, ARIMA (Autoregressive Integrated Moving Average), and transfer function (TF) models for time series, and produces forecasts. The procedure includes an Expert Modeler that identifies and estimates an appropriate model for each dependent variable series. Alternatively, you can specify a custom model.

This algorithm is designed with help from professor Ruey Tsay at The University of Chicago.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Y_t (t=1, 2, ..., n)	Univariate time series under investigation.
n	Total number of observations.
$\hat{Y}_t(k)$	Model-estimated k-step ahead forecast at time t for series Y.
S	The seasonal length.

Models

TSMODEL estimates exponential smoothing models and ARIMA/TF models.

Exponential Smoothing Models

The following notation is specific to exponential smoothing models:

α	Level smoothing weight
γ	Trend smoothing weight
ϕ	Damped trend smoothing weight
δ	Season smoothing weight

Simple Exponential Smoothing

Simple exponential smoothing has a single level parameter and can be described by the following equations:

$$L(t) = \begin{cases} \alpha Y(t) + (1-\alpha)L(t-1), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ L(t-1), & else \end{cases}$$

$$\hat{Y}_t(k) = L(t)$$

It is functionally equivalent to an ARIMA(0,1,1) process.

Brown’s Exponential Smoothing

Brown’s exponential smoothing has level and trend parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha Y(t) + (1-\alpha)L(t-1), & if\ \ Y(t)\ \ is\ \ not\ \ missing \\ L(t-1) + T(t-1), & else \end{cases}$$

$$T(t) = \begin{cases} \alpha(L(t) - L(t-1)) + (1-\alpha)T(t-1), & if\ \ Y(t)\, is\ \ not\ \ missing \\ T(t-1), & else \end{cases}$$

$$\hat{Y}_t(k) = L(t) + \left((k-1) + \alpha^{-1}\right)T(t)$$

It is functionally equivalent to an ARIMA(0,2,2) with restriction among MA parameters.

Holt’s Exponential Smoothing

Holt’s exponential smoothing has level and trend parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha Y(t) + (1-\alpha)(L(t-1) + T(t-1)), & if\ \ Y(t)\, is\ \ not\ \ missing \\ L(t-1) + T(t-1), & else \end{cases}$$

$$T(t) = \begin{cases} \gamma(L(t) - L(t-1)) + (1-\gamma)T(t-1), & if\ \ Y(t)\ \ is\ \ not\ \ missing \\ T(t-1), & else \end{cases}$$

$$\hat{Y}_t(k) = L(t) + kT(t)$$

It is functionally equivalent to an ARIMA(0,2,2).

Damped-Trend Exponential Smoothing

Damped-Trend exponential smoothing has level and damped trend parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha Y(t) + (1-\alpha)(L(t-1) + \varphi T(t-1)), & if\ \ Y(t)\ \ is\ \ not\ \ missing \\ L(t-1) + \varphi T(t-1), & else \end{cases}$$

$$T(t) = \begin{cases} \gamma(L(t) - L(t-1)) + (1-\gamma)\varphi T(t-1), & if\ \ Y(t)\, is\ \ not\ \ missing \\ \varphi T(t-1), & else \end{cases}$$

$$\hat{Y}_t(k) = L(t) + \sum_{i=1}^{k} \phi^i T(t)$$

It is functionally equivalent to an ARIMA(1,1,2).

Simple Seasonal Exponential Smoothing

Simple seasonal exponential smoothing has level and season parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha(Y(t) - S(t-s)) + (1-\alpha)L(t-1), & if \;\; Y(t) \; is \;\; not \;\; missing \\ L(t-1), & else \end{cases}$$

$$S(t) = \begin{cases} \delta(Y(t) - L(t)) + (1-\delta)S(t-s), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ S(t-s) & else \end{cases}$$

$$\hat{Y}_t(k) = L(t) + S(t + \boldsymbol{k} - s)$$

It is functionally equivalent to an ARIMA(0,1,(1,s,s+1))(0,1,0) with restrictions among MA parameters.

Winters' Additive Exponential Smoothing

Winters' additive exponential smoothing has level, trend, and season parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha(Y(t) - S(t-s)) + (1-\alpha)(L(t-1) + T(t-1)), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ L(t-1) + T(t-1), & else \end{cases}$$

$$T(t) = \begin{cases} \gamma(L(t) - L(t-1)) + (1-\gamma)T(t-1), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ T(t-1), & else \end{cases}$$

$$S(t) = \begin{cases} \delta(Y(t) - L(t)) + (1-\delta)S(t-s), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ S(t-s) & else \end{cases}$$

$$\hat{Y}_t(k) = L(t) + kT(t) + S(t + \boldsymbol{k} - s)$$

It is functionally equivalent to an ARIMA(0,1,s+1)(0,1,0) with restrictions among MA parameters.

Winters' Multiplicative Exponential Smoothing

Winters' multiplicative exponential smoothing has level, trend and season parameters and can be described by the following equations:

$$L(t) = \begin{cases} \alpha(Y(t) / S(t-s)) + (1-\alpha)(L(t-1) + T(t-1)), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ L(t-1) + T(t-1), & else \end{cases}$$

$$T(t) = \begin{cases} \gamma(L(t) - L(t-1)) + (1-\gamma)T(t-1), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ T(t-1), & else \end{cases}$$

$$S(t) = \begin{cases} \delta(Y(t) / L(t)) + (1-\delta)S(t-s), & if \;\; Y(t) \;\; is \;\; not \;\; missing \\ S(t-s) & else \end{cases}$$

$$\hat{Y}_t(k) = (L(t) + kT(t))S(t + \boldsymbol{k} - s)$$

There is no equivalent ARIMA model.

Estimation and Forecasting of Exponential Smoothing

The sum of squares of the one-step ahead prediction error, $\Sigma\left(Y_t - \hat{Y}_{t-1}(1)\right)^2$, is minimized to optimize the smoothing weights.

Initialization of Exponential Smoothing

Let *L* denote the level, *T* the trend and, *S*, a vector of length *s*, denote the seasonal states. The initial smoothing states are made by back-casting from t=n to t=0. Initialization for back-casting is described here.

For all the models $L = y_n$.

For all non-seasonal models with trend, *T* is the negative of the slope of the line (with intercept) fitted to the data with time as a regressor.

For the simple seasonal model, the elements of *S* are seasonal averages minus the sample mean; for example, for monthly data the element corresponding to January will be average of all January values in the sample minus the sample mean.

For the additive Winters' model, fit $y = \alpha t + \sum_{i=1}^{s} \beta_i I_i(t)$ to the data where *t* is time and $I_i(t)$ are seasonal dummies. Note that the model does not have an intercept. Then $T = -\alpha$, and $S = \beta - mean(\beta)$.

For the multiplicative Winters' model, fit a separate line (with intercept) for each season with time as a regressor. Suppose μ is the vector of intercepts and β is the vector of slopes (these vectors will be of length *s*). Then $T = -mean(\beta)$ and $S = (\mu + \beta) / (mean(\mu) + mean(\beta))$.

The initial smoothing states are:

$$L' = L(0)$$

$$T' = -T(0)$$

$$S^{'} = (S(1-s), S(2-s), \ldots S(-1), S(0)) = (S(1), S(2), \ldots, S(-1+s), S(0))$$

ARIMA and Transfer Function Models

The following notation is specific to ARIMA/TF models:

a_t ($t = 1, 2, \ldots, n$)	White noise series normally distributed with mean zero and variance σ^2
p	Order of the non-seasonal autoregressive part of the model
q	Order of the non-seasonal moving average part of the model
d	Order of the non-seasonal differencing
P	Order of the seasonal autoregressive part of the model
Q	Order of the seasonal moving-average part of the model
D	Order of the seasonal differencing

s	Seasonality or period of the model
$\phi_p(B)$	AR polynomial of B of order p, $\phi_p(B) = 1 - \varphi_1 B - \varphi_2 B^2 - \ldots - \varphi_p B^p$
$\theta_q(B)$	MA polynomial of B of order q, $\theta_q(B) = 1 - \vartheta_1 B - \vartheta_2 B^2 - \ldots - \vartheta_q B^q$
$\Phi_P(B^s)$	Seasonal AR polynomial of BS of order P, $\Phi_P(B^s) = 1 - \Phi_1 B^s - \Phi_2 B^{s2} - \ldots - \Phi_P B^{sP}$
$\Theta_Q(B^s)$	Seasonal MA polynomial of BS of order Q, $\Theta_Q(B^s) = 1 - \Theta_1 B^s - \Theta_2 B^{s2} - \ldots - \Theta_Q B^{sQ}$
Δ	Differencing operator $\Delta = (1-B)^d (1-B^s)^D$
B	Backward shift operator with $BY_t = Y_{t-1}$ and $Ba_t = a_{t-1}$
$Z\sigma_t^2$	Prediction variance of Z_t
$N\sigma_t^2$	Prediction variance of the noise forecasts

Transfer function (TF) models form a very large class of models, which include univariate ARIMA models as a special case. Suppose Y_t is the dependent series and, optionally, $X_{1t}, X_{2t}, \ldots, X_{kt}$ are to be used as predictor series in this model. A TF model describing the relationship between the dependent and predictor series has the following form:

$$Z_t = f(Y_t),$$

$$\Delta Z_t = \mu + \sum_{i=1}^{k} \frac{Num_i}{Den_i} \Delta_i B^{b_i} f_i(X_{it}) + \frac{MA}{AR} a_t.$$

The univariate ARIMA model simply drops the predictors from the TF model; thus, it has the following form:

$$\Delta Z_t = \mu + \frac{MA}{AR} a_t$$

The main features of this model are:

- An initial transformation of the dependent and predictor series, *f* and *f*$_i$. This transformation is optional and is applicable only when the dependent series values are positive. Allowed transformations are log and square root. These transformations are sometimes called variance-stabilizing transformations.
- A constant term μ.
- The unobserved i.i.d., zero mean, Gaussian error process a_t with variance σ^2.
- The moving average lag polynomial *MA*=$\theta_q(B)\Theta_Q(B^s)$ and the auto-regressive lag polynomial *AR*=$\phi_p(B)\Phi_P(B^s)$.
- The difference/lag operators Δ and Δ_i.
- A delay term, B^{b_i}, where b_i is the order of the delay
- Predictors are assumed given. Their numerator and denominator lag polynomials are of the form: $Num_i = (\omega_{i0} - \omega_{i1}B - \cdots - \omega_{iu}B^u)(1 - \Omega_{i1}B^s - \cdots - \Omega_{iv}B^{vs})B^b$ and $Den_i = (1 - \delta_{i1}B - \cdots - \delta_{ir}B^r)(1 - \Delta_{i1}B^s - \cdots)$.
- The “noise” series

$$N_t = \Delta Z_t - \mu - \sum_{i=1}^{k} \frac{Num_i}{Den_i} \Delta_i B^{b_i} X_{it}$$

is assumed to be a mean zero, stationary ARMA process.

Interventions and Events

Interventions and events are handled like any other predictor; typically they are coded as 0/1 variables, but note that a given intervention variable's exact effect upon the model is determined by the transfer function in front of it.

Estimation and Forecasting of ARIMA/TF

There are two forecasting algorithms available: Conditional Least Squares (CLS) and Exact Least Squares (ELS) or Unconditional Least Squares forecasting (ULS). These two algorithms differ in only one aspect: they forecast the noise process differently. The general steps in the forecasting computations are as follows:

1. Computation of noise process N_t through the historical period.
2. Forecasting the noise process N_t up to the forecast horizon. This is one step ahead forecasting during the historical period and multi-step ahead forecasting after that. The differences in CLS and ELS forecasting methodologies surface in this step. The prediction variances of noise forecasts are also computed in this step.
3. Final forecasts are obtained by first adding back to the noise forecasts the contributions of the constant term and the transfer function inputs and then integrating and back-transforming the result. The prediction variances of noise forecasts also may have to be processed to obtain the final prediction variances.

Let $\hat{N}_t(k)$ and $\sigma_t^2(k)$ be the k-step forecast and forecast variance, respectively.

Conditional Least Squares (CLS) Method

$\hat{N}_t(k) = E(N_{t+k} | N_t, N_{t-1}, \cdots)$ assuming $N_t = 0$ for t<0.

$$\sigma_t^2(k) = \sigma^2 \sum_{j=0}^{k-1} \psi_j^2$$

where ψ_j are coefficients of the power series expansion of $MA/(\Delta \times AR)$.

Minimize $S = \Sigma\left(N_t - \hat{N}_t(1)\right)^2$.

Missing values are imputed with forecast values of N_t.

Maximum Likelihood (ML) Method (Brockwell and Davis, 1991)

$$\hat{N}_t(k) = E(N_{t+k} | N_t, N_{t-1}, \cdots, N_1)$$

Maximize likelihood of $\left\{N_t - \hat{N}_t(1)\right\}_{t=1}^{n}$; that is,

$$L = -\ln(S/n) - (1/n)\sum_{j=1}^{n}\ln(\eta_j)$$

where $S = \Sigma\left(N_t - \hat{N}_t(1)\right)^2/\eta_t$, and $\sigma_t^2 = \sigma^2\eta_t$ is the one-step ahead forecast variance.

When missing values are present, a Kalman filter is used to calculate $\hat{N}_t(k)$.

Error Variance

$$\hat{\sigma}^2 = S/(n-k)$$

in both methods. Here *n* is the number of non-zero residuals and *k* is the number of parameters (excluding error variance).

Initialization of ARIMA/TF

A slightly modified Levenberg-Marquardt algorithm is used to optimize the objective function. The modification takes into account the “admissibility” constraints on the parameters. The admissibility constraint requires that the roots of AR and MA polynomials be outside the unit circle and the sum of denominator polynomial parameters be non-zero for each predictor variable. The minimization algorithm requires a starting value to begin its iterative search. All the numerator and denominator polynomial parameters are initialized to zero except the coefficient of the 0th power in the numerator polynomial, which is initialized to the corresponding regression coefficient.

The ARMA parameters are initialized as follows:

Assume that the series Y_t follows an ARMA(p,q)(P,Q) model with mean 0; that is:

$$Y_t - \varphi_1 Y_{t-1} - \cdots - \varphi_p Y_{t-p} = a_t - \theta_1 a_{t-1} - \cdots - \theta_q a_{t-q}$$

In the following c_l and ρ_l represent the *l*th lag autocovariance and autocorrelation of Y_t respectively, and $\hat{c}_l$ and $\hat{\rho}_l$ represent their estimates.

Non-Seasonal AR Parameters

For AR parameter initial values, the estimated method is the same as that in appendix A6.2 of (Box, Jenkins, and Reinsel, 1994). Denote the estimates as $\hat{\varphi}_1^{'}, \cdots, \hat{\varphi}_{p+q}^{'}$.

Non-Seasonal MA Parameters

Let

$$w_t = Y_t - \varphi_1 Y_{t-1} - \cdots - \varphi_p Y_{t-p} = a_t - \theta_1 a_{t-1} - \cdots - \theta_q a_{t-q}$$

The cross covariance

$$\lambda_l = E(w_{t+l}a_t) = E((a_{t+l} - \theta_1 a_{t+l-1} - \cdots - \theta_q a_{t+l-q})a_t) = \begin{cases} \sigma_a^2 & l = 0 \\ -\theta_1\sigma_a^2 & l = 1 \\ \cdots & \cdots \\ -\theta_q\sigma_a^2 & l = q \\ 0 & l > q \end{cases}$$

Assuming that an AR(p+q) can approximate Y_t, it follows that:

$$Y_t - \varphi'_1 Y_{t-1} - \cdots - \varphi'_p Y_{t-p} - \varphi'_{p+1} Y_{t-p-1} - \cdots - \varphi'_{p+q} Y_{t-p-q} = a_t$$

The AR parameters of this model are estimated as above and are denoted as $\hat{\varphi}'_1, \cdots, \hat{\varphi}'_{p+q}$.

Thus λ_l can be estimated by

$$\lambda_l \approx E\Big((Y_{t+l} - \varphi_1 Y_{t+l-1} - \cdots - \varphi_p Y_{t+l-p})\left(Y_t - \varphi'_1 Y_{t-1} - \cdots - \varphi'_{p+q} Y_{t-p-q}\right)\Big)$$
$$= \left(\rho_l - \sum_{j=1}^{p+q} \varphi_j \rho_{l+j} - \sum_{i=1}^{p} \varphi_i \rho_{l-i} + \sum_{i=1}^{p}\sum_{j=1}^{p+q} \varphi_i \varphi_j \rho_{l+j-i}\right) c_0$$

And the error variance σ_a^2 is approximated by

$$\hat{\sigma}_a^2 = Var\left(-\sum_{j=0}^{p+q} \varphi'_j Y_{t-j}\right) = \sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \varphi'_i \varphi'_j c_{i-j} = c_0 \sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \varphi'_i \varphi'_j \rho_{i-j}$$

with $\hat{\varphi}'_0 = -1$.

Then the initial MA parameters are approximated by $\theta_l = -\lambda_l/\sigma_a^2$ and estimated by

$$\hat{\theta}_l = -\hat{\lambda}_l/\hat{\sigma}_a^2 = \frac{\rho_l - \sum_{j=1}^{p+q} \hat{\varphi}_j \rho_{l+j} - \sum_{i=1}^{p} \hat{\varphi}_i \rho_{l-i} + \sum_{i=1}^{p}\sum_{j=1}^{p+q} \hat{\varphi}_i \hat{\varphi}_j \rho_{l+j-i}}{\sum_{i=0}^{p+q}\sum_{j=0}^{p+q} \hat{\varphi}'_i \hat{\varphi}'_j \rho_{i-j}}$$

So $\hat{\theta}_l$ can be calculated by $\hat{\varphi}'_j, \hat{\varphi}_i$, and $\{\hat{\rho}_l\}_{l=1}^{p+2q}$. In this procedure, only $\{\hat{\rho}_l\}_{l=1}^{p+q}$ are used and all other parameters are set to 0.

Seasonal parameters

For seasonal AR and MA components, the autocorrelations at the seasonal lags in the above equations are used.

Calculation of the Transfer Function

The transfer function needs to be calculated for each predictor series. For the predictor series X_{it}, let the transfer function be:

$$V_{it} = \frac{Num_i}{Den_i} \Delta_i B^{b_i} f_i(X_{it})$$

It can be calculated as follows:

1. Calculate $U_{it} = \Delta_i B^{b_i} f_i(X_{it})$

2. Recursively calculate

$$V_{it} = -\sum_{j=1}^{D_i} Cden_i(j) * V_{it-j} + \sum_{j=0}^{N_i} Cnum_i(j) * U_{it-j}$$

where $Cden_i(j)$ and $Cnum_i(j)$ are the coefficients of B^j in the polynomials Den_i and Num_i respectively. Likewise, the summation limits D_i and N_i are the maximum degree of B^j in the polynomials Den_i and Num_i respectively.

All missing V_{it-j} in the first term of V_{it} are taken to be $V_{i,-\infty}$ and missing U_{it-j} in the second term are taken to be $U_{i,-\infty}$, where $U_{i,-\infty}$ is the first non-missing measurement of U_{it}. $V_{i,-\infty}$ is given by

$$V_{i,-\infty} = \frac{Num_i(1)}{Den_i(1)} * U_{i,-\infty}$$

where $Num_i(1)$ and $Den_i(1)$ are the Num_i and Den_i polynomials evaluated at $B = 1$.

Diagnostic Statistics

ARIMA/TF diagnostic statistics are based on residuals of the noise process, $R(t) = N(t) - \hat{N}(t)$.

Ljung-Box Statistic

$$Q(K) = n(n+2)\sum_{k=1}^{K} r_k^2/(n-k)$$

where r_k is the kth lag ACF of residual.

Q(K) is approximately distributed as $\chi^2(K-m)$, where m is the number of parameters other than the constant term and predictor related-parameters.

Outlier Detection in Time Series Analysis

The observed series may be contaminated by so-called outliers. These outliers may change the mean level of the uncontaminated series. The purpose of outlier detection is to find if there are outliers and what are their locations, types, and magnitudes.

TSMODEL considers seven types of outliers. They are additive outliers (AO), innovational outliers (IO), level shift (LS), temporary (or transient) change (TC), seasonal additive (SA), local trend (LT), and AO patch (AOP).

Notation

The following notation is specific to outlier detection:

U(t) or U_t	The uncontaminated series, outlier free. It is assumed to be a univariate ARIMA or transfer function model.

Definitions of Outliers

Types of outliers are defined separately here. In practice any combination of these types can occur in the series under study.

AO (Additive Outliers)

Assuming that an AO outlier occurs at time *t=T*, the observed series can be represented as

$$Y(t) = U(t) + wI_T(t)$$

where $I_T(t) = \begin{cases} 0 & t \neq T \\ 1 & t = T \end{cases}$ is a pulse function and w is the deviation from the true *U(T)* caused by the outlier.

IO (Innovational Outliers)

Assuming that an IO outlier occurs at time *t=T*, then

$$Y(t) = \mu(t) + \frac{\theta(B)}{\Delta\varphi(B)}(a(t) + wI_T(t))$$

LS (Level Shift)

Assuming that a LS outlier occurs at time *t=T*, then

$$Y(t) = U(t) + wS_T(t)$$

where $S_T(t) = \frac{1}{1-B} I_T(t) = \begin{cases} 0 & t < T \\ 1 & t \geq T \end{cases}$ is a step function.

TC (Temporary/Transient Change)

Assuming that a TC outlier occurs at time *t=T*, then

$$Y(t) = U(t) + wD_T(t)$$

where $D_T(t) = \frac{1}{1-\delta B} I_T(t)$, $0 < \delta < 1$ is a damping function.

SA (Seasonal Additive)

Assuming that a SA outlier occurs at time *t=T*, then

$$Y(t) = U(t) + wSS_T(t)$$

where $SS_T(t) = \frac{1}{1-B^s} I_T(t) = \begin{cases} 1 & t = T + ks, k \geq 0 \\ 0 & o.w. \end{cases}$ is a step seasonal pulse function.

LT (Local Trend)

Assuming that a LT outlier occurs at time *t*=*T*, then

$$Y(t) = U(t) + wT_T(t)$$

where $T_T(t) = \frac{1}{(1-B)^2} I_T(t) = \begin{cases} t + 1 - T & t \geq T \\ 0 & o.w. \end{cases}$ is a local trend function.

AOP (AO patch)

An AO patch is a group of two or more consecutive AO outliers. An AO patch can be described by its starting time and length. Assuming that there is a patch of AO outliers of length *k* at time *t*=*T*, the observed series can be represented as

$$Y(t) = U(t) + \sum_{i=1}^{k} w_i I_{T-1+i}(t)$$

Due to a masking effect, a patch of AO outliers is very difficult to detect when searching for outliers one by one. This is why the AO patch is considered as a separate type from individual AO. For type AO patch, the procedure searches for the whole patch together.

Summary

For an outlier of type O at time *t*=*T* (except AO patch):

$$Y(t) = \mu(t) + wL_O(B) I_T(t) + \frac{\theta(B)}{\Delta\varphi(B)} a(t)$$

where

$$L_O(B) = \begin{cases} 1 & O = AO \\ 1/(\Delta\pi(B)) & O = IO \\ 1/(1-B) & O = LS \\ 1/(1-\delta B) & O = TC \\ 1/(1-B^s) & O = SA \\ 1/(1-B)^2 & O = LT \end{cases}$$

with $\pi(B) = \varphi(B)/\theta(B)$. A general model for incorporating outliers can thus be written as follows:

$$Y(t) = \mu(t) + \sum_{k=1}^{M} w_k L_{O_k}(B) I_{T_k}(t) + \frac{\theta(B)}{\Delta\varphi(B)} a(t)$$

where *M* is the number of outliers.

Estimating the Effects of an Outlier

Suppose that the model and the model parameters are known. Also suppose that the type and location of an outlier are known. Estimation of the magnitude of the outlier and test statistics are as follows.

The results in this section are only used in the intermediate steps of outlier detection procedure. The final estimates of outliers are from the model incorporating all the outliers in which all parameters are jointly estimated.

Non-AO Patch Deterministic Outliers

For a deterministic outlier of any type at time T (except AO patch), let $e(t)$ be the residual and $x(t) = \pi(B) L(B) \Delta I_T(t)$, so:

$$e(t) = wx(t) + a(t)$$

From residuals *e(t)*, the parameters for outliers at time T are estimated by simple linear regression of *e(t)* on *x(t)*.

For j = 1 (AO), 2 (IO), 3 (LS), 4 (TC), 5 (SA), 6 (LT), define test statistics:

$$\lambda_j(\mathrm{T}) = \frac{w_j(T)}{\sqrt{\mathrm{Var}(w_j(T))}}$$

Under the null hypothesis of no outlier, λ_j(T) is distributed as N(0,1) assuming the model and model parameters are known.

AO Patch Outliers

For an AO patch of length k starting at time T, let $x_i(t;T) = \pi(B) \Delta I_{T+i-1}(t)$ for i = 1 to k, then

$$e(t) = \sum_{i=1}^{k} w_i(T) x_i(t;T) + a(t)$$

Multiple linear regression is used to fit this model. Test statistics are defined as:

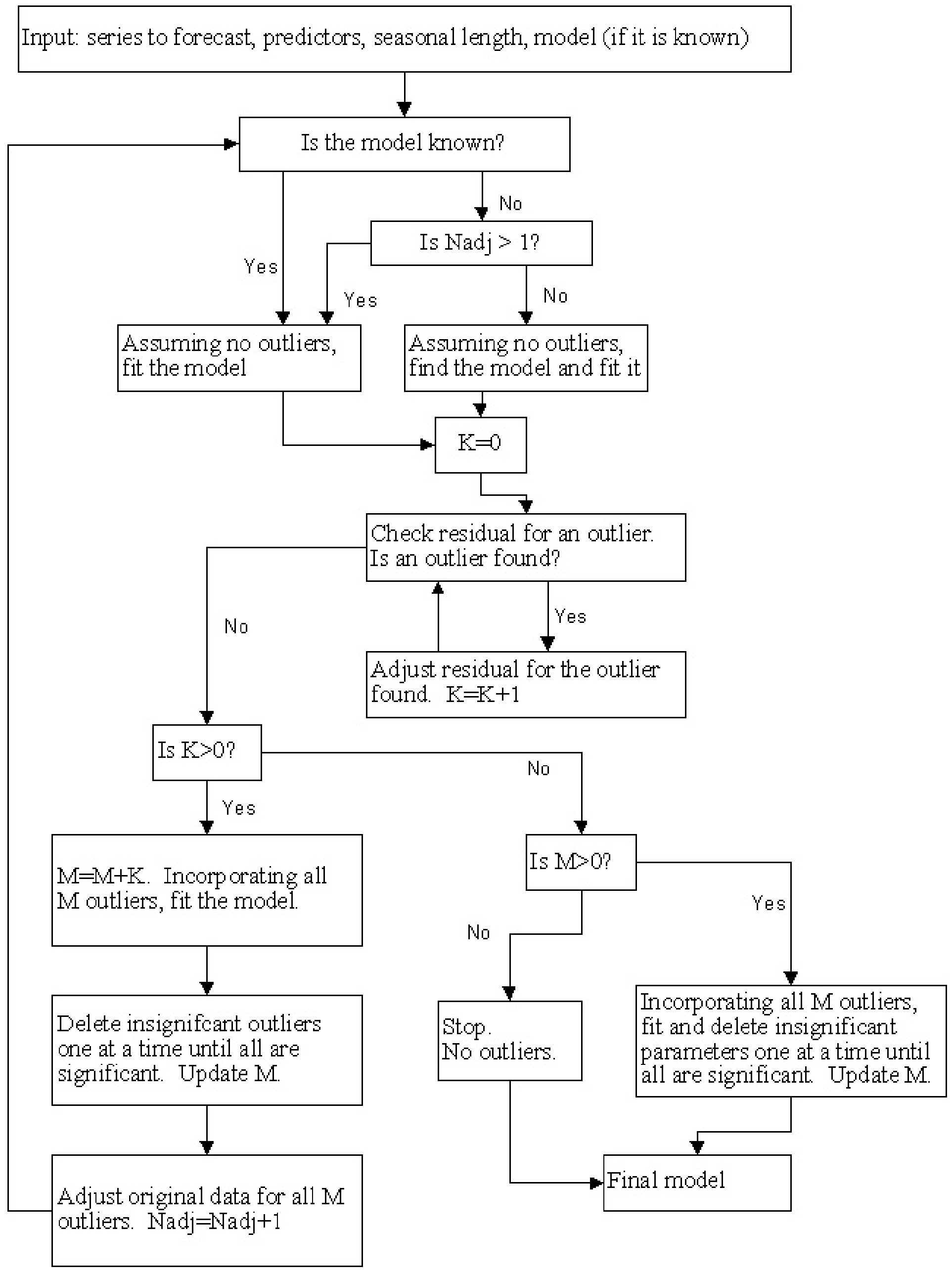
$$\chi^2(T) = \frac{\mathbf{w}'(T)(X_T X_T)\mathbf{w}(T)}{\sigma^2}$$

Assuming the model and model parameters are known, $\chi^2(T)$ has a Chi-square distribution with k degrees of freedom under the null hypothesis $w_1(T) = \cdots = w_k(T) = 0$.

Detection of Outliers

The following flow chart demonstrates how automatic outlier detection works. Let M be the total number of outliers and Nadj be the number of times the series is adjusted for outliers. At the beginning of the procedure, M = 0 and Nadj = 0.

Figure 100-1


Input: series to forecast, predictors, seasonal length, model (if it is known)
Is the model known?
Yes
No
Is Nadj > 1?
Yes
No
Assuming no outliers, fit the model
Assuming no outliers, find the model and fit it
K=0
Check residual for an outlier. Is an outlier found?
No
Yes
Adjust residual for the outlier found. K=K+1
Is K>0?
No
Yes
M=M+K. Incorporating all M outliers, fit the model.
Is M>0?
No
Yes
Delete insignifcant outliers one at a time until all are significant. Update M.
Stop. No outliers.
Incorporating all M outliers, fit and delete insignificant parameters one at a time until all are significant. Update M.
Final model
Adjust original data for all M outliers. Nadj=Nadj+1

Goodness-of-Fit Statistics

Goodness-of-fit statistics are based on the original series Y(t). Let k= number of parameters in the model, n = number of non-missing residuals.

Mean Squared Error

$$MSE = \frac{\Sigma\left(Y(t) - \hat{Y}(t)\right)^2}{n-k}$$

Mean Absolute Percent Error

$$MAPE = \frac{100}{n}\Sigma\left|\left(Y(t) - \hat{Y}(t)\right)/Y(t)\right|$$

Maximum Absolute Percent Error

$$MaxAPE = 100\max\left(\left|\left(Y(t) - \hat{Y}(t)\right)/Y(t)\right|\right)$$

Mean Absolute Error

$$MAE = \frac{1}{n}\Sigma\left|Y(t) - \hat{Y}(t)\right|$$

Maximum Absolute Error

$$MaxAE = \max\left(\left|Y(t) - \hat{Y}(t)\right|\right)$$

Normalized Bayesian Information Criterion

Normalized $BIC = \ln(MSE) + k\frac{\ln(n)}{n}$

R-Squared

$$R^2 = 1 - \frac{\Sigma\left(Y(t) - \hat{Y}(t)\right)^2}{\Sigma\left(Y(t) - \overline{Y}\right)^2}$$

Stationary R-Squared

A similar statistic was used by Harvey (Harvey, 1989).

$$R_S^2 = 1 - \frac{\sum_t \left(Z(t) - \hat{Z}(t)\right)^2}{\sum_t \left(\Delta Z(t) - \overline{\Delta Z}\right)^2}$$

where

The sum is over the terms in which both $Z(t) - \hat{Z}(t)$ and $\Delta Z(t) - \overline{\Delta Z}$ are not missing.

$\overline{\Delta Z}$ is the simple mean model for the differenced transformed series, which is equivalent to the univariate baseline model ARIMA(0,d,0)(0,D,0).

For the exponential smoothing models currently under consideration, use the differencing orders (corresponding to their equivalent ARIMA models if there is one).

$$d = \begin{cases} 2 & \text{Brown, Holt} \\ 1 & \text{other} \end{cases}, \quad D = \begin{cases} 0 & s = 1 \\ 1 & s > 1 \end{cases}$$

Note: Both the stationary and usual R-squared can be negative with range $(-\infty, 1]$. A negative R-squared value means that the model under consideration is worse than the baseline model. Zero R-squared means that the model under consideration is as good or bad as the baseline model. Positive R-squared means that the model under consideration is better than the baseline model.

Expert Modeling

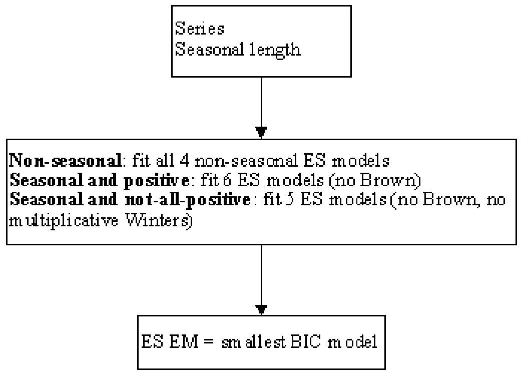
Univariate Series

Users can let the Expert Modeler select a model for them from:

- All models (default).
- Exponential smoothing models only.
- ARIMA models only.

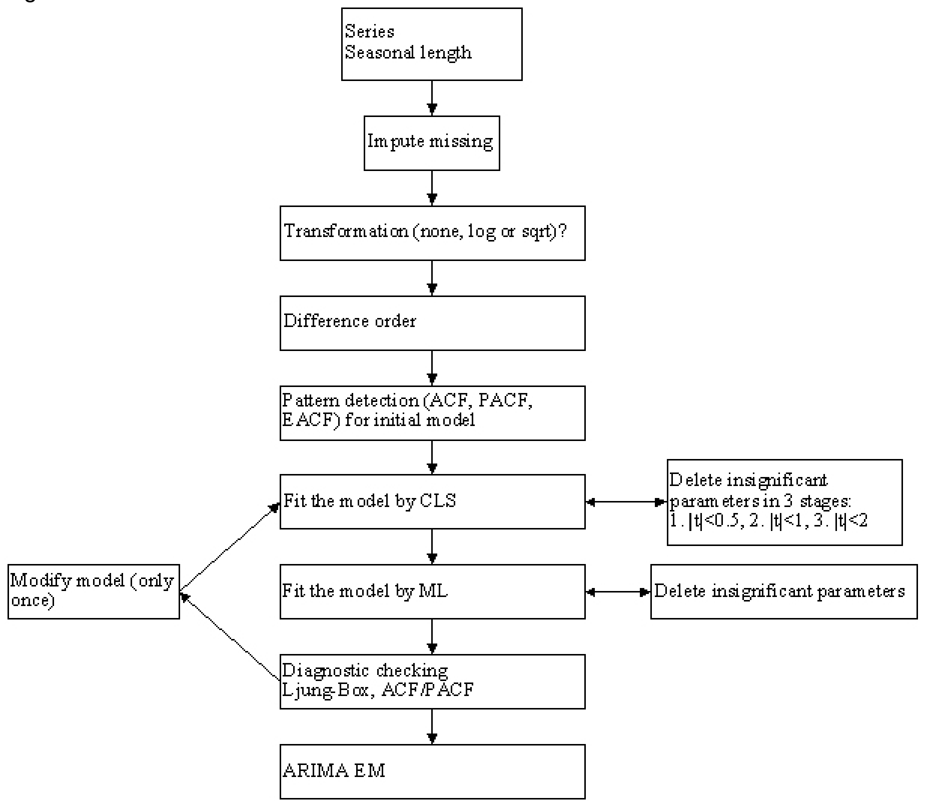
Exponential Smoothing Expert Model

Figure 100-2


Series
Seasonal length
Non-seasonal: fit all 4 non-seasonal ES models
Seasonal and positive: fit 6 ES models (no Brown)
Seasonal and not-all-positive: fit 5 ES models (no Brown, no multiplicative Winters)
ES EM = smallest BIC model

ARIMA Expert Model

Figure 100-3


Series
Seasonal length
Impute missing
Transformation (none, log or sqrt)?
Difference order
Pattern detection (ACF, PACF, EACF) for initial model
Fit the model by CLS
Delete insignificant parameters in 3 stages: 1. |t|<0.5, 2. |t|<1, 3. |t|<2
Modify model (only once)
Fit the model by ML
Delete insignificant parameters
Diagnostic checking Ljung-Box, ACF/PACF
ARIMA EM

Note: If 10<n<3s, set s=1 to build a non-seasonal model.

All Models Expert Model

In this case, the Exponential Smoothing and ARIMA expert models are computed, and the model with the smaller normalized BIC is chosen.

Note: For short series, n<max(20,3s), use Exponential Smoothing Expert Model.

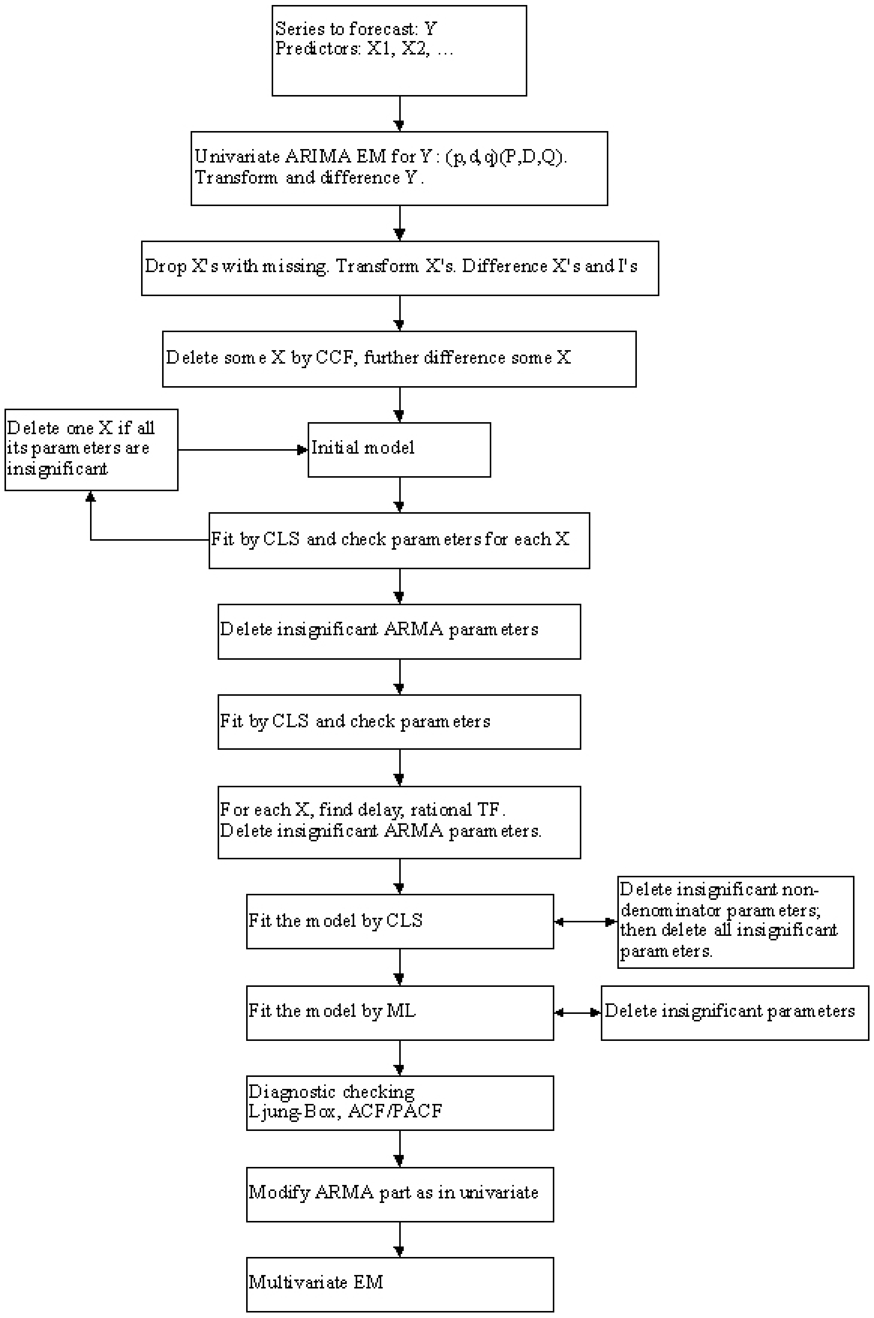
Multivariate Series

In the multivariate situation, users can let the Expert Modeler select a model for them from:

- All models (default). Note that if the multivariate expert ARIMA model drops all the predictors and ends up with a univariate expert ARIMA model, this univariate expert ARIMA model will be compared with expert exponential smoothing models as before and the Expert Modeler will decide which is the best overall model.
- ARIMA models only.

Transfer Function Expert Model

Figure 100-4


Series to forecast: Y
Predictors: X1, X2, ...
Univariate ARIMA EM for Y : (p,d,q)(P,D,Q). Transform and difference Y.
Drop X's with missing. Transform X's. Difference X's and I's
Delete some X by CCF, further difference some X
Delete one X if all its parameters are insignificant
Initial model
Fit by CLS and check parameters for each X
Delete insignificant ARMA parameters
Fit by CLS and check parameters
For each X, find delay, rational TF. Delete insignificant ARMA parameters.
Fit the model by CLS
Delete insignificant non-denominator parameters; then delete all insignificant parameters.
Fit the model by ML
Delete insignificant parameters
Diagnostic checking Ljung-Box, ACF/PACF
Modify ARMA part as in univariate
Multivariate EM

Note: For short series, n<max(20,3s), fit a univariate expert model.

References

Box, G. E. P., G. M. Jenkins, and G. C. Reinsel. 1994. *Time series analysis: Forecasting and control*, 3rd ed. Englewood Cliffs, N.J.: Prentice Hall.

Brockwell, P. J., and R. A. Davis. 1991. *Time Series: Theory and Methods*, 2 ed. : Springer-Verlag.

Gardner, E. S. 1985. Exponential smoothing: The state of the art. *Journal of Forecasting*, 4, 1–28.

Harvey, A. C. 1989. *Forecasting, structural time series models and the Kalman filter*. Cambridge: Cambridge University Press.

Makridakis, S. G., S. C. Wheelwright, and R. J. Hyndman. 1997. *Forecasting: Methods and applications*, 3rd ed. ed. New York: John Wiley and Sons.

Melard, G. 1984. A fast algorithm for the exact likelihood of autoregressive-moving average models. *Applied Statistics*, 33:1, 104–119.

Pena, D., G. C. Tiao, and R. S. Tsay, eds. 2001. *A course in time series analysis*. New York: John Wiley and Sons.

TWOSTEP CLUSTER Algorithms

The TwoStep cluster method is a scalable cluster analysis algorithm designed to handle very large data sets. It can handle both continuous and categorical variables or attributes. It requires only one data pass. It has two steps 1) pre-cluster the cases (or records) into many small sub-clusters; 2) cluster the sub-clusters resulting from pre-cluster step into the desired number of clusters. It can also automatically select the number of clusters.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table 101-1
Notation

Notation	**Description**
K^A	Total number of continuous variables used in the procedure.
K^B	Total number of categorical variables used in the procedure.
L_k	Number of categories for the *k*th categorical variable.
R_k	The range of the *k*th continuous variable.
N	Number of data records in total.
N_k	Number of data records in cluster *k*.
$\hat{\mu}_k$	The estimated mean of the *k*th continuous variable across the entire dataset.
$\hat{\sigma}_k^2$	The estimated variance of the *k*th continuous variable across the entire dataset.
$\hat{\mu}_{jk}$	The estimated mean of the *k*th continuous variable in cluster *j*.
$\hat{\sigma}_{jk}^2$	The estimated variance of the *k*th continuous variable in cluster *j*.
N_{jkl}	Number of data records in cluster *j* whose *k*th categorical variable takes the *l*th category.
N_{kl}	Number of data records in the *k*th categorical variable that take the *l*th category.
d(j, s)	Distance between clusters *j* and *s*.
$< j, s >$	Index that represents the cluster formed by combining clusters *j* and *s*.

TwoStep Clustering Procedure

The TwoStep clustering procedure consists of the following steps:

- Pre-clustering,
- Outlier handling (optional),
- Clustering

Pre-cluster

The pre-cluster step uses a sequential clustering approach. It scans the data records one by one and decides if the current record should be merged with the previously formed clusters or starts a new cluster based on the distance criterion (described below).

The procedure is implemented by constructing a modified cluster feature (CF) tree. The CF tree consists of levels of nodes, and each node contains a number of entries. A leaf entry (an entry in the leaf node) represents a final sub-cluster. The non-leaf nodes and their entries are used to guide a new record quickly into a correct leaf node. Each entry is characterized by its CF that consists of the entry's number of records, mean and variance of each range field, and counts for each category of each symbolic field. For each successive record, starting from the root node, it is recursively guided by the closest entry in the node to find the closest child node, and descends along the CF tree. Upon reaching a leaf node, it finds the closest leaf entry in the leaf node. If the record is within a threshold distance of the closest leaf entry, it is absorbed into the leaf entry and the CF of that leaf entry is updated. Otherwise it starts its own leaf entry in the leaf node. If there is no space in the leaf node to create a new leaf entry, the leaf node is split into two. The entries in the original leaf node are divided into two groups using the farthest pair as seeds, and redistributing the remaining entries based on the closeness criterion.

If the CF tree grows beyond allowed maximum size, the CF tree is rebuilt based on the existing CF tree by increasing the threshold distance criterion. The rebuilt CF tree is smaller and hence has space for new input records. This process continues until a complete data pass is finished. For details of CF tree construction, see the BIRCH algorithm (Zhang, Ramakrishnon, and Livny, 1996).

All records falling in the same entry can be collectively represented by the entry's CF. When a new record is added to an entry, the new CF can be computed from this new record and the old CF without knowing the individual records in the entry. These properties of CF make it possible to maintain only the entry CFs, rather than the sets of individual records. Hence the CF-tree is much smaller than the original data and can be stored in memory more efficiently.

Note that the structure of the constructed CF tree may depend on the input order of the cases or records. To minimize the order effect, randomly order the records before building the model.

Outlier Handling

An optional outlier-handling step is implemented in the algorithm in the process of building the CF tree. Outliers are considered as data records that do not fit well into any cluster. We consider data records in a leaf entry as outliers if the number of records in the entry is less than a certain fraction (25% by default) of the size of the largest leaf entry in the CF tree. Before rebuilding the CF tree, the procedure checks for potential outliers and sets them aside. After rebuilding the CF tree, the procedure checks to see if these outliers can fit in without increasing the tree size. At the end of CF tree building, small entries that cannot fit in are outliers.

Cluster

The cluster step takes sub-clusters (non-outlier sub-clusters if outlier handling is used) resulting from the pre-cluster step as input and then groups them into the desired number of clusters. Since the number of sub-clusters is much less than the number of original records, traditional clustering methods can be used effectively. TwoStep uses an agglomerative hierarchical clustering method, because it works well with the auto-cluster method (see the section on auto-clustering below).

Hierarchical clustering refers to a process by which clusters are recursively merged, until at the end of the process only one cluster remains containing all records. The process starts by defining a starting cluster for each of the sub-clusters produced in the pre-cluster step. (For more information, see the topic “Pre-cluster”.) All clusters are then compared, and the pair of clusters with the smallest distance between them is selected and merged into a single cluster. After merging, the new set of clusters is compared, the closest pair is merged, and the process repeats until all clusters have been merged. (If you are familiar with the way a decision tree is built, this is a similar process, except in reverse.) Because the clusters are merged recursively in this way, it is easy to compare solutions with different numbers of clusters. To get a five-cluster solution, simply stop merging when there are five clusters left; to get a four-cluster solution, take the five-cluster solution and perform one more merge operation, and so on.

Accuracy

In general, the larger the number of sub-clusters produced by the pre-cluster step, the more accurate the final result is. However, too many sub-clusters will slow down the clustering during the second step. The maximum number of sub-clusters should be carefully chosen so that it is large enough to produce accurate results and small enough not to slow down the second step clustering.

Distance Measure

A log-likelihood or Euclidean measure can be used to calculate the distance between clusters.

Log-Likelihood Distance

The log-likelihood distance measure can handle both continuous and categorical variables. It is a probability based distance. The distance between two clusters is related to the decrease in log-likelihood as they are combined into one cluster. In calculating log-likelihood, normal distributions for continuous variables and multinomial distributions for categorical variables are assumed. It is also assumed that the variables are independent of each other, and so are the cases. The distance between clusters *j* and *s* is defined as:

$$d(i,j) = \xi_i + \xi_j - \xi_{\langle i,j \rangle}$$

where

$$\xi_v = -N_v \left(\sum_{k=1}^{K^A} \frac{1}{2} \log \left(\hat{\sigma}_k^2 + \hat{\sigma}_{vk}^2 \right) + \sum_{k=1}^{K^B} \hat{E}_{vk} \right)$$

and

$$\hat{E}_{vk} = -\sum_{l=1}^{L_k} \frac{N_{vkl}}{N_v} \log \frac{N_{vkl}}{N_v}$$

If $\hat{\sigma}_k^2$ is ignored in the expression for ξ_v, the distance between clusters *i* and *j* would be exactly the decrease in log-likelihood when the two clusters are combined. The $\hat{\sigma}_k^2$ term is added to solve the problem caused by $\hat{\sigma}_{vk}^2 = 0$, which would result in the natural logarithm being undefined. (This would occur, for example, when a cluster has only one case.)

Euclidean Distance

This distance measure can only be applied if all variables are continuous. The Euclidean distance between two points is clearly defined. The distance between two clusters is here defined by the Euclidean distance between the two cluster centers. A cluster center is defined as the vector of cluster means of each variable.

Number of Clusters (auto-clustering)

TwoStep can use the hierarchical clustering method in the second step to assess multiple cluster solutions and automatically determine the optimal number of clusters for the input data. A characteristic of hierarchical clustering is that it produces a sequence of partitions in one run: 1, 2, 3, … clusters. In contrast, a *k*-means algorithm would need to run multiple times (one for each specified number of clusters) in order to generate the sequence. To determine the number of clusters automatically, TwoStep uses a two-stage procedure that works well with the hierarchical clustering method. In the first stage, the BIC for each number of clusters within a specified range is calculated and used to find the initial estimate for the number of clusters. The BIC is computed as

$$BIC(J) = -2\sum_{j=1}^{J} \xi_j + m_J \log(N)$$

where

$$m_J = J\left\{2K^A + \sum_{k=1}^{K^B} (L_K - 1)\right\}$$

and other terms defined as in “Distance Measure”. The ratio of change in BIC at each successive merging relative to the first merging determines the initial estimate. Let $dBIC(J)$ be the difference in BIC between the model with J clusters and that with (J + 1) clusters, $dBIC(J) = BIC(J) - BIC(J+1)$. Then the change ratio for model J is

$$R_1(J) = \frac{dBIC(J)}{dBIC(1)}$$

If $dBIC(1) < 0$, then the number of clusters is set to 1 (and the second stage is omitted). Otherwise, the initial estimate for number of clusters *k* is the smallest number for which $R_1(J) < 0.04$

In the second stage, the initial estimate is refined by finding the largest relative increase in distance between the two closest clusters in each hierarchical clustering stage. This is done as follows:

- Starting with the model C_k indicated by the BIC criterion, take the ratio of minimum inter-cluster distance for that model and the next larger model C_{k+1}, that is, the previous model in the hierarchical clustering procedure,

$$R_2(k) = \frac{d_{\min}(C_k)}{d_{\min}(C_{k+1})}$$

where C_k is the cluster model containing *k* clusters and $d_{\min}(C)$ is the minimum inter-cluster distance for cluster model *C*.

- Now from model C_{k-1}, compute the same ratio with the following model C_k, as above. Repeat for each subsequent model until you have the ratio $R_2(2)$.
- Compare the two largest R_2 ratios; if the largest is more that 1.15 times the second largest, then select the model with the largest R_2 ratio as the optimal number of clusters; otherwise, from those two models with the largest R_2 values, select the one with the larger number of clusters as the optimal model.

Cluster Membership Assignment

Records are assigned to clusters based upon the specified outlier handling and distance measure options.

Without Outlier-Handling

Assign a record to the closest cluster according to the distance measure.

With Outlier-Handling

With outlier handling, records are assigned depending upon the distance measure specified.

Log-Likelihood Distance

Assume outliers or noises follow a uniform distribution. Calculate both the log-likelihood resulting from assigning a record to a noise cluster and that resulting from assigning it to the closest non-noise cluster. The record is then assigned to the cluster which leads to the larger log-likelihood. This is equivalent to assigning a record to its closest non-noise cluster if the distance between them is smaller than a critical value $C = \log(V)$, where $V = \prod_m R_m \prod_m L_m$. Otherwise, designate it as an outlier.

Euclidean Distance

Assign a record to its closest non-noise cluster if the Euclidean distance between them is smaller than a critical value $C = 2\sqrt{\frac{1}{JK_A}\sum_{j=1}^{J}\sum_{k=1}^{K_A}\hat{\sigma}_{jk}^2}$. Otherwise, designate it as an outlier.

Missing Values

No missing values are allowed. Cases with missing values are deleted on a listwise basis.

References

Zhang, T., R. Ramakrishnon, and M. Livny. 1996. BIRCH: An efficient data clustering method for very large databases. In: *Proceedings of the ACM SIGMOD Conference on Management of Data,* Montreal, Canada: ACM, 103–114.

Chiu, T., D. Fang, J. Chen, Y. Wang, and C. Jeris. 2001. A Robust and Scalable Clustering Algorithm for Mixed Type Attributes in Large Database Environment. In: *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining,* SanFrancisco, CA: ACM, 263–268.

VARCOMP Algorithms

The Variance Components procedure provides estimates for variances of random effects under a general linear model framework. Four types of estimation methods are available in this procedure.

Notation

The following notation is used throughout this chapter. Unless otherwise stated, all vectors are column vectors and all quantities are known.

n	Number of observations, $n \geq 1$
k	Number of random effects, $k \geq 0$
m_0	Number of parameters in the fixed effects, $m_0 \geq 0$
m_i	Number of parameters in the *i*th random effect, $m_i \geq 0$, *i*=1,...,*k*
m	Total number of parameters, $m = m_0 + m_1 + \cdots + m_k$
σ_i^2	Unknown variance of the *i*th random effect, $\sigma_i^2 \geq 0$, *i*=1,...,*k*
σ_e^2	Unknown variance of the residual term, same as σ_{k+1}^2, $\sigma_e^2 > 0$
γ_i^2	Unknown variance ratio of the *i*th random effect, $\gamma_i^2 = \sigma_i^2/\sigma_e^2$, $\gamma_i^2 \geq 0$, *i*=1,...,*k*, and $\gamma_{k+1}^2 = 1$
y	The length *n* vector of observations
e	The length *n* vector of residuals
$\mathbf{X}_i$	The $n \times m_i$ design matrix, *i*=0,1,...,*k*
β_0	The length m_0 vector of parameters of the fixed effects
β_i	The length m_i vector of parameters of the *i*th random effect, *i*=1,...,*k*

Unless otherwise stated, a *p*×*p* identity matrix is denoted as $\mathbf{I}_p$, a *p*×*q* zero matrix is denoted as $\mathbf{0}_{p\times q}$, and a zero vector of length *p* is denoted as $\mathbf{0}_p$.

Weights

For the sake of clarity and simplicity, the algorithms described in this chapter assume unit frequency weight and unit regression weight for all cases. Weights can be applied as described in the following two sections.

Frequency Weight

The WEIGHT command specifies frequency weights.

- Cases with nonpositive frequency are excluded from all calculations in the procedure.
- Non-integral frequency weight is rounded to the nearest integer.
- The total sample size is equal to the sum of positive rounded frequency weights.

Regression Weight

The REGWGT subcommand specifies regression weights. Suppose the *l*th case has a regression weight $w_i > 0$ (cases with nonpositive regression weights are excluded from all calculations in the procedure). Let $\mathbf{W} = diag(w_1, \ldots, w_n)$ be the *n*×*n* diagonal weight matrix. Then the VARCOMP procedure will perform all calculations as if **y** is physically transformed to $\mathbf{W}^{\frac{1}{2}}\mathbf{y}$ and $\mathbf{X}_i$to $\mathbf{W}^{\frac{1}{2}}\mathbf{X}_i$, *i*=0,1,...,*k*; and then the pertinent algorithm is applied to the transformed data.

Model

The mixed model is represented, following Rao (1973), as

$$\mathbf{y} = \mathbf{X}_0\beta_0 + \sum_{i=1}^{k} \mathbf{X}_i\beta_i + \mathbf{e}$$

The random vectors $\beta_1, \ldots, \beta_k$ and **e** are assumed to be jointly independent. Moreover, the random vector β_i is distributed as $N_{m_i}\left(\mathbf{0}, \sigma_i^2\mathbf{I}_{m_i}\right)$ for *i*=1,...,*k* and the residual vector **e** is distributed as $N_n\left(\mathbf{0}, \sigma_e^2\mathbf{I}_n\right)$. It follows from these assumptions that **y** is distributed as $N_n\left(\mathbf{X}_0\mathrm{b}_0, \sigma_e^2\mathbf{V}\right)$ where

$$\mathbf{V} = \sum_{i=1}^{k} \gamma_i^2\mathbf{X}_i\mathbf{X}'_i + \mathbf{I}_n = \sum_{i=1}^{k+1} \gamma_i^2\mathbf{V}_i$$

where $\mathbf{V}_i = \mathbf{X}_i\mathbf{X}'_i$, *i*=1,...,*k*, and $\mathbf{V}_i = \mathbf{I}_n$.

Minimum Norm Quadratic Unbiased Estimate (MINQUE)

Given the initial guess or the prior values $\gamma_i^2 = \alpha_i(\alpha_i \geq 0)$, *i*=1,...,*k*+1, the MINQUE ofσ are obtained as a solution of the linear system of equations:

$$\mathbf{S}\sigma = \mathbf{q}$$

where $\mathbf{S} = \{s_{ij}\}$ is a (*k*+1)×(*k*+1) symmetric matrix, $\mathbf{q} = \{q_i\}$ is a (*k*+1) vector, and $\sigma = \left(\sigma_1^2, \ldots, \sigma_{k+1}^2\right)$. Define

$$\mathbf{R} = \mathbf{V}^{-1} - \mathbf{V}^{-1}\mathbf{X}_0\left(\mathbf{X}'_0\mathbf{V}^{-1}\mathbf{X}_0\right)^{-}\mathbf{X}'_0\mathbf{V}^{-1}$$

The elements of **S** and **q** are

$$s_{ij} = \begin{cases} SSQ\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_j\right) & i = 1, \ldots, k, \quad j = 1, \ldots, k \\ SSQ\left(\mathbf{X}'_i\mathbf{R}\right) & i = 1, \ldots, k, \quad j = k+1 \\ SSQ(\mathbf{R}\mathbf{X}_j) & i = k+1, \quad j = 1, \ldots, k \\ SSQ(\mathbf{R}) & i = k+1, \quad j = k+1 \end{cases}$$

and

$$q_i = \begin{cases} SSQ\left(\mathbf{X}'_i\mathbf{R}\mathbf{y}\right) & i = 1, \ldots, k \\ SSQ(\mathbf{R}\mathbf{y}) & i = k+1 \end{cases}$$

where SSQ(**A**) is the sum of squares of all elements of a matrix **A**.

MINQUE(0)

The prior values are $\alpha_i = 0$, *i*=1,...,*k*, and $\alpha_{k+1} = 1$. Under this set of prior values, $\mathbf{V} = \mathbf{I}_n$ and $\mathbf{R} = \mathbf{I}_n - \mathbf{X}_0\left(\mathbf{X}'_0\mathbf{X}_0\right)^-\mathbf{X}'_0$. Since this **R** is an idempotent matrix, some of the elements of **S** and **q** can be simplified to

$$\begin{aligned}
&s_{i,k+1} = trace\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_i\right) \quad i = 1,\ldots,k;\\
&s_{k+1,j} = trace\left(\mathbf{X}'_j\mathbf{R}\mathbf{X}_j\right) \quad j = 1,\ldots,k;\\
&s_{k+1,k+1} = n - rank(\mathbf{X}_0)\\
&q_{k+1} = \mathbf{y}'\mathbf{R}\mathbf{y}
\end{aligned}$$

Using the algorithm by Goodnight (1978), the elements of S and q are obtained without explicitly computing R. The steps are described as follows:

Step 1. Form the symmetric matrix:

$$\begin{bmatrix}
\mathbf{X}'_0\mathbf{X}_0 & \mathbf{X}'_0\mathbf{X}_1 & \cdots & \mathbf{X}'_0\mathbf{X}_k & \mathbf{X}'_0\mathbf{y}\\
\mathbf{X}'_1\mathbf{X}_0 & \mathbf{X}'_1\mathbf{X}_1 & \cdots & \mathbf{X}'_1\mathbf{X}_k & \mathbf{X}'_1\mathbf{y}\\
\vdots & \vdots & & \vdots & \vdots\\
\mathbf{X}'_k\mathbf{X}_0 & \mathbf{X}'_k\mathbf{X}_1 & \cdots & \mathbf{X}'_k\mathbf{X}_k & \mathbf{X}'_k\mathbf{y}\\
\mathbf{y}'\mathbf{X}_0 & \mathbf{y}'\mathbf{X}_1 & \cdots & \mathbf{y}'\mathbf{X}_k & \mathbf{y}'\mathbf{y}
\end{bmatrix}$$

Step 2. Sweep the above matrix by pivoting on each diagonal of $\mathbf{X}'_0\mathbf{X}_0$. This produces the following matrix:

$$\begin{bmatrix}
\mathbf{G} & \mathbf{G}\mathbf{X}'_0\mathbf{X}_1 & \cdots & \mathbf{G}\mathbf{X}'_0\mathbf{X}_k & \mathbf{G}\mathbf{X}'_0\mathbf{y}\\
\mathbf{X}'_1\mathbf{X}_0\mathbf{G} & \mathbf{X}'_1\mathbf{R}\mathbf{X}_1 & \cdots & \mathbf{X}'_1\mathbf{R}\mathbf{X}_k & \mathbf{X}'_1\mathbf{R}\mathbf{y}\\
\vdots & \vdots & & \vdots & \vdots\\
\mathbf{X}'_k\mathbf{X}_0\mathbf{G} & \mathbf{X}'_k\mathbf{R}\mathbf{X}_1 & \cdots & \mathbf{X}'_k\mathbf{R}\mathbf{X}_k & \mathbf{X}'_k\mathbf{R}\mathbf{y}\\
\mathbf{y}'\mathbf{X}_0\mathbf{G} & \mathbf{y}'\mathbf{R}\mathbf{X}_1 & \cdots & \mathbf{y}'\mathbf{R}\mathbf{X}_k & \mathbf{y}'\mathbf{R}\mathbf{y}
\end{bmatrix}$$

where $\mathbf{G} = \left(\mathbf{X}'_0\mathbf{X}_0\right)^-$. In the process of computing the above matrix, the rank of $\mathbf{X}_0$ is obtained as the number of nonzero pivots found.

Step 3. Form **S** and **q**. The MINQUE(0) of **σ** are $\hat{\sigma} = \mathbf{S}^-\mathrm{q}$.

MINQUE(1)

The prior values are $\alpha_i = 1$, *i*=1,...,*k*+1. Under this set of prior values, $\mathbf{V} = \sum_{i=1}^{k+1}\mathbf{X}'_i\mathbf{X}_i$. Using Giesbrecht (1983), the matrix **S** and the vector **q** are obtained through an iterative procedure. The steps are described as follows:

Step 1. Construct the augmented matrix $\mathbf{A} = [\mathbf{X}_0|\mathbf{X}_1|\cdots|\mathbf{X}_k|\mathbf{y}]$. Then compute the $(m+1)\times(m+1)$ matrix $\mathbf{T}_{(k+1)} = \mathbf{A}'\mathbf{A}$.

Step 2. Define $\mathbf{H}_{(l)} = \sum_{i=l}^{k+1} \mathbf{X}'_i \mathbf{X}_i$, and $\mathbf{T}_{(l)} = \mathbf{A}'\mathbf{H}_{(l)}^{-1}\mathbf{A}$, l=1,...,k. Update $\mathbf{T}_{(l+1)}$ to $\mathbf{T}_{(l)}$ using the *W* Transform given in Goodnight and Hemmerle (1979). The updating formula is

$$\mathbf{T}_{(l)} = \mathbf{T}_{(l+1)} - \mathbf{A}'\mathbf{H}_{(l+1)}^{-1}\mathbf{X}_l\left(\mathbf{I}_{m_l} + \mathbf{X}'_l\mathbf{H}_{(l+1)}^{-1}\mathbf{X}_l\right)^{-}\mathbf{X}'_l\mathbf{H}_{(l+1)}^{-1}\mathbf{A}$$

Step 3. Once $\mathbf{T}_{(1)} = \mathbf{A}'\mathbf{H}_{(1)}^{-1}\mathbf{A} = \mathbf{A}'\mathbf{V}^{-1}\mathbf{A}$ is obtained, apply the Sweep operation to the diagonal elements of upper left $m_0 \times m_0$ submatrix of $\mathbf{T}_{(1)}$. The resulting matrix will contain the quadratic form $\mathbf{y}'\mathbf{R}\mathbf{y}$, the vectors $\mathbf{y}'\mathbf{R}\mathbf{X}_j$, j=1,...,k, and the matrices $\mathbf{X}'_i\mathbf{R}\mathbf{X}_j$, i, j=1,...,k .

Step 4. Compute the elements of **S** and **q**. Since $\mathbf{RVR} = \mathbf{R}$, then

$$SSQ(\mathbf{R}\mathbf{X}_j) = tr\left(\mathbf{X}'_j\mathbf{R}\mathbf{X}_j\right) - \sum_{i=1}^{k} SSQ\left(\mathbf{X}'_j\mathbf{R}\mathbf{X}_i\right) \quad j = 1, \ldots, k$$
$$SSQ(\mathbf{R}) = n - rank(\mathbf{X}_0) - \sum_{i=1}^{k} tr\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_i\right) - \sum_{j=1}^{k} SSQ(\mathbf{R}\mathbf{X}_j)$$
$$SSQ(\mathbf{R}\mathbf{y}) = \mathbf{y}'\mathbf{R}\mathbf{y} - \sum_{j=1}^{k} SSQ\left(\mathbf{y}'\mathbf{R}\mathbf{X}_j\right)$$

The MINQUE(1) of **σ** are $\hat{\sigma} = \mathbf{S}^{-}\mathbf{q}$.

Maximum Likelihood Estimate (MLE)

The maximum likelihood estimates are obtained using the algorithm by Jennrich and Sampson (1976). The algorithm is an iterative procedure that combines Newton-Raphson steps and Fisher scoring steps.

Parameters

The parameter vector is $\theta = \begin{bmatrix} \beta_0 \\ \gamma^2 \\ \sigma_e^2 \end{bmatrix}$ where $\gamma^2 = \begin{bmatrix} \gamma_1^2 \\ \\ . \\ \gamma_k^2 \end{bmatrix}$.

Likelihood Function

The likelihood function is

$$L \equiv L(\theta) = (2\pi)^{-n/2}\left|\sigma_e^2\mathbf{V}\right|^{-1/2} \exp\left(-\tfrac{1}{2}(\mathbf{y} - \mathbf{X}_0\beta_0)'\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}_0\beta_0)/\sigma_e^2\right).$$

The log-likelihood function is

$$l = \log L = -\tfrac{n}{2}\log(2\pi) - \tfrac{n}{2}\log\left(\sigma_e^2\right) - \tfrac{1}{2}\log|\mathbf{V}| - \tfrac{1}{2\sigma_e^2}(\mathbf{y} - \mathbf{X}_0\beta_0)'\mathbf{V}^{-1}(\mathbf{y} - \mathbf{X}_0\beta_0).$$

Gradient Vector

$$\frac{\partial l}{\partial \beta_0} = \frac{1}{\sigma_e^2}\mathbf{X}'_0\mathbf{V}^{-1}\mathbf{r},$$
$$\frac{\partial l}{\partial \gamma_i^2} = \frac{1}{2\sigma_e^2}\mathbf{r}'\mathbf{V}^{-1}\mathbf{X}_i\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{r} - \frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{X}_i\right), \quad i = 1, \ldots, k\ ,$$
$$\frac{\partial l}{\partial \sigma_e^2} = \frac{1}{2\sigma_e^4}\mathbf{r}'\mathbf{V}^{-1}\mathbf{r} - \frac{n}{2\sigma_e^2}.$$

where $\mathbf{r} = \mathbf{y} - \mathbf{X}_0\beta_0$. The gradient vector is

$$\frac{dl}{d\theta} = \begin{bmatrix} \frac{\partial l}{\partial \beta_0} \\ \frac{\partial l}{\partial \gamma^2} \\ \frac{\partial l}{\partial \sigma_e^2} \end{bmatrix}$$

Hessian Matrix

$$\frac{\partial^2 l}{\partial \beta_0 \partial \beta_0} = -\frac{1}{\sigma_e^2}\mathbf{X}'_0\mathbf{V}^{-1}\mathbf{X}_0$$

$$\frac{\partial^2 l}{\partial \gamma_i^2 \partial \beta_0} = -\frac{1}{\sigma_e^2}\mathbf{r}'\mathbf{V}^{-1}\mathbf{X}_i\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{X}_0 \quad i = 1, \ldots, k\ ,$$
$$\frac{\partial^2 l}{\partial \gamma_i^2 \partial \gamma_j^2} = \frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{X}_j\mathbf{X}'_j\mathbf{V}^{-1}\mathbf{X}_i\right) - \frac{1}{\sigma_e^2}\mathbf{r}'\mathbf{V}^{-1}\mathbf{X}_i\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{X}_j\mathbf{X}'_j\mathbf{V}^{-1}\mathbf{r} \quad i = 1, \ldots, k; j = 1, \ldots, k\ ,$$
$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \beta_0} = -\frac{1}{\sigma_e^4}\mathbf{r}'\mathbf{V}^{-1}\mathbf{X}_0$$
$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma_j^2} = -\frac{1}{2\sigma_e^4}\mathbf{r}'\mathbf{V}^{-1}\mathbf{X}_j\mathbf{X}'_j\mathbf{V}^{-1}\mathbf{r} \quad j = 1, \ldots, k$$
$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \sigma_e^2} = \frac{n}{2\sigma_e^4} - \frac{1}{\sigma_e^6}\mathbf{r}'\mathbf{V}^{-1}\mathbf{r}$$

The Hessian matrix is

$$\frac{d^2 l}{d\theta d\theta} = \begin{bmatrix} \frac{\partial^2 l}{\partial \beta_0 \partial \beta_0} & \frac{\partial^2 l}{\partial \beta_0 \partial \gamma^2} & \frac{\partial^2 l}{\partial \beta_0 \partial \sigma_e^2} \\ \frac{\partial^2 l}{\partial \gamma^2 \partial \beta_0} & \frac{\partial^2 l}{\partial \gamma^2 \partial \gamma^2} & \frac{\partial^2 l}{\partial \gamma^2 \partial \sigma_e^2} \\ \frac{\partial^2 l}{\partial \sigma_e^2 \partial \beta_0} & \frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma^2} & \frac{\partial^2 l}{\partial \sigma_e^2 \partial \sigma_e^2} \end{bmatrix}$$

where

$$\frac{\partial^2 l}{\partial \gamma^2 \partial \beta_0} = \begin{bmatrix} \frac{\partial^2 l}{\partial \gamma_1^2 \partial \beta_0} \\ \vdots \\ \frac{\partial^2 l}{\partial \gamma_k^2 \partial \beta_0} \end{bmatrix},$$

$$\frac{\partial^2 l}{\partial \gamma^2 \partial \gamma^2} = \begin{bmatrix} \frac{\partial^2 l}{\partial \gamma_1^2 \partial \gamma_1^2} & \cdots & \frac{\partial^2 l}{\partial \gamma_1^2 \partial \gamma_k^2} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 l}{\partial \gamma_k^2 \partial \gamma_1^2} & \cdots & \frac{\partial^2 l}{\partial \gamma_k^2 \partial \gamma_k^2} \end{bmatrix}$$

and

$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma^2} = \begin{bmatrix} \frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma_1^2} \\ \vdots \\ \frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma_k^2} \end{bmatrix}$$

Fisher Information Matrix

As $E(\mathbf{r}) = \mathbf{0}_n$ and $E\left(\mathbf{r}'\mathbf{V}^{-1}\mathbf{r}\right) = n\sigma_e^2$, the expected second derivatives are

$$E\left(\frac{\partial^2 l}{\partial\beta_0\partial\beta_0}\right) = -\frac{1}{\sigma_e^2}\mathbf{X}'_0\mathbf{V}^{-1}\mathbf{X}_0$$

$$E\left(\frac{\partial^2 l}{\partial\gamma_i^2\partial\beta_0}\right) = \mathbf{0}'_{m_0} \quad i = 1,\ldots,k$$
$$E\left(\frac{\partial^2 l}{\partial\gamma_i^2\partial\gamma_j^2}\right) = -\frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{V}^{-1}\mathbf{X}_j\mathbf{X}'_j\mathbf{V}^{-1}\mathbf{X}_i\right) \quad i = 1,\ldots,k, j = 1,\ldots,k$$

$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\beta_0}\right) = \mathbf{0}'_{m_0}$$
$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_j^2}\right) = -\frac{1}{2\sigma_e^2}tr\left(\mathbf{X}'_j\mathbf{V}^{-1}\mathbf{X}_j\right) \quad j = 1,\ldots,k,$$
$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\sigma_e^2}\right) = -\frac{n}{2\sigma_e^4}$$

The Fisher Information matrix is

$$E\left(\frac{d^2 l}{d\theta d\theta}\right) = \begin{bmatrix} \frac{\partial^2 l}{\partial\beta_0\partial\beta_0} & \mathbf{0}_{m_0\times(m-m_0)} & \mathbf{0}_{m_0} \\ \mathbf{0}_{(m-m_0)\times m_0} & E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\gamma^2}\right) & E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\sigma_e^2}\right) \\ \mathbf{0}'_{m_0} & E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma^2}\right) & E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\sigma_e^2}\right) \end{bmatrix}$$

where

$$E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\gamma^2}\right) = \begin{bmatrix} E\left(\frac{\partial^2 l}{\partial\gamma_1^2\partial\gamma_1^2}\right) & \ldots & E\left(\frac{\partial^2 l}{\partial\gamma_1^2\partial\gamma_k^2}\right) \\ \vdots & \ddots & \vdots \\ E\left(\frac{\partial^2 l}{\partial\gamma_k^2\partial\gamma_1^2}\right) & \ldots & E\left(\frac{\partial^2 l}{\partial\gamma_k^2\partial\gamma_k^2}\right) \end{bmatrix}$$

and

$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma^2}\right) = \begin{bmatrix} E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_1^2}\right) \\ \vdots \\ E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_k^2}\right) \end{bmatrix}$$

Iteration Procedure

The iterative estimation algorithm proceeds according to the details in the following sections.

Initial Values

- Fixed Effect Parameters: $\hat{\beta}_0 = \left(\mathbf{X}'_0\mathbf{X}_0\right)^-\mathbf{X}'_0\mathbf{y}$
- Random Effect Variance Components: For the *i*th random effect, compute $\hat{\beta}_i = \left(\mathbf{X}'_i\mathbf{X}_i\right)^-\mathbf{X}'_i\mathbf{y}$. Then assign the variance of the m_i elements of $\hat{\beta}_i$ using divisor $(m_i - 1)$ to the estimate $\hat{\sigma}_i^2$ if $m_i \geq 2$; otherwise $\hat{\sigma}_i^2 = 0$.
- Residual Variance: $\hat{\sigma}_e^2 = \mathbf{r}'\mathbf{r}/n$ where $\mathbf{X} = [\mathbf{X}_0|\mathbf{X}_1|\cdots|\mathbf{X}_k]$ and $\mathbf{r} = \mathbf{y} - \left(\mathbf{X}'\mathbf{X}\right)^-\mathbf{X}'\mathbf{y}$. If $\hat{\sigma}_e^2 = 0$ but *k*≥1 then reset $\hat{\sigma}_e^2 = 10^{-8}$ so that the iteration can continue.

The variance ratios are then computed as $\hat{\gamma}_i^2 = \hat{\sigma}_i^2/\hat{\sigma}_e^2$. Following the same method in which the residual variance is initialized, $\hat{\sigma}_e^2 > 0$ for $k \geq 1$.

Updating

At the *s*th iteration, *s*=0,1,..., the parameter vector is updated as

$$\hat{\theta}_{(s+1)} = \hat{\theta}_s + \rho\Delta\hat{\theta}_s$$

where $\Delta\hat{\theta}_s$ is the value of increment $\Delta\theta$ evaluated at $\theta = \hat{\theta}_s$, and ρ>0 is a step size such that $l\left(\hat{\theta}_{(s+1)}\right) > l\left(\hat{\theta}_s\right)$. The increment vector depends on the choice of step type—Newton-Raphson versus Fisher scoring. The step size is determined by the step-halving technique with ρ=1 initially and a maximum of 10 halvings.

Choice of Step

Following Jennrich and Sampson (1976), the first iteration is always the Fisher scoring step because it is more robust to poor initial values. For subsequent iteration the Newton-Raphson step is used if:

1. The Hessian matrix is nonnegative definite, and
2. The increment in the log-likelihood function of step 1 is less than or equal to one.

Otherwise the Fisher scoring step is used. The increment vector for each type of step is:

- Newton-Raphson Step: $\Delta\theta = \left(-\frac{d^2 l}{d\theta d\theta}\right)^{-1}\frac{dl}{d\theta}$.
- Fisher Scoring Step: $\Delta\theta = \left(-E\left(\frac{d^2 l}{d\theta d\theta}\right)\right)^{-1}\frac{dl}{d\theta}$.

Convergence Criteria

Given the convergence criterion $\epsilon > 0$, the iteration is considered converged when the following criteria are satisfied:

1. $\left|l\left(\hat{\theta}_{(s+1)}\right) - l\left(\hat{\theta}_s\right)\right| < \epsilon \times \max\left(1, \left|\hat{\theta}_s\right|\right)$, and
2. $< l\left(\hat{\theta}_{(s+1)}\right) - l\left(\hat{\theta}_s\right) >< \epsilon \times \max\left(1, < \hat{\theta}_s >\right)$ where <**a**> is the sum of absolute values of elements of the vector **a**.

Negative Variance Estimates

Negative variance estimates can occur at the end of an iteration. An *ad hoc* method is to set those estimates to zero before the next iteration.

Covariance Matrix

Let $\hat{\theta}$ be the vector of maximum likelihood estimates. Their covariance matrix is given by

$$cov\left(\hat{\theta}\right) = \left(-E\left(\frac{d^2 l}{d\theta d\theta}\right)\Big|_{\theta=\hat{\theta}}\right)^{-1}$$

Let

$$\mathbf{\Psi} = \begin{bmatrix} \beta_0 \\ \sigma_1^2 \\ \\ . \\ \sigma_k^2 \\ \sigma_e^2 \end{bmatrix}$$

be the original parameters. Their maximum likelihood estimates are given by

$$\hat{\mathbf{\Psi}} = \begin{bmatrix} \hat{\beta}_0 \\ \hat{\sigma}_e^2 \hat{\gamma}_1^2 \\ \\ . \\ \hat{\sigma}_e^2 \hat{\gamma}_k^2 \\ \hat{\sigma}_e^2 \end{bmatrix}$$

and their covariance matrix is estimated by

$$cov\left(\hat{\mathbf{\Psi}}\right) = \mathbf{J} cov\left(\hat{\theta}\right)\mathbf{J}'$$

where

$$\mathbf{J} = \begin{bmatrix} \mathbf{I}_{m_0} & \mathbf{0}_{m_0 \times k} & \mathbf{0}_{m_0} \\ \mathbf{0}_{k \times m_0} & \sigma_e^2 \mathbf{I}_k & \gamma \\ 0 & 0 & 1 \end{bmatrix}$$

which is the $(m_0 + k + 1) \times (m_0 + k + 1)$ Jacobian matrix of transforming **θ** to **ψ**.

Restricted Maximum Likelihood Estimate (REML)

The restricted maximum likelihood method finds a linear transformation on **y** such that the resulting vector does not involve the fixed effect parameter vector $b_0$regardless of their values. It has been shown that these linear combinations are the residuals obtained after a linear regression on the fixed effects. Suppose *r* is the rank of $\mathbf{X}_0$; then there are at most $n - r$ linearly independent combinations. Let **K** be an $n \times (n - r)$ matrix whose columns are these linearly independent combinations. Then the properties of **K** are (Searle et al., 1992, Chapter 6):

$$\mathbf{K}'\mathbf{X}_0 = \mathbf{0}_{(n-r)\times m_0}$$
$$\mathbf{K}' = \mathbf{TM}$$

where **T** is a $(n - r) \times n$ matrix with linearly independent rows and

$$\mathbf{M} = \mathbf{I}_n - \mathbf{X}_0\left(\mathbf{X}'_0\mathbf{X}_0\right)^{-}\mathbf{X}'_0$$

It can be shown that REML estimation is invariant to **K** (Searle et al., 1992, Chapter 6); thus, we can choose **K** such that $\mathbf{K}'\mathbf{K} = \mathbf{I}_{n-r}$ to simplify calculations. It follows that the distribution of $\mathbf{K}'\mathbf{y}$ is $N_{n-r}\left(\mathbf{0}, \sigma_e^2 \mathbf{K}'\mathbf{V}\mathbf{K}\right)$.

Parameters

The parameter vector is $\theta = \begin{bmatrix} \gamma^2 \\ \sigma_e^2 \end{bmatrix}$ where $\gamma^2 = \begin{bmatrix} \gamma_1^2 \\ . \\ \gamma_k^2 \end{bmatrix}$.

Likelihood Function

The likelihood function of $\mathbf{K}'\mathbf{y}$ is

$$L \equiv L(\theta) = (2\pi)^{-(n-r)/2} \left|\sigma_e^2 \mathbf{K}'\mathbf{V}\mathbf{K}\right|^{-1/2} \exp\left(-\tfrac{1}{2}\mathbf{y}'\mathbf{K}\left(\mathbf{K}'\mathbf{V}\mathbf{K}\right)^{-1}\mathbf{K}'\mathbf{y}/\sigma_e^2\right).$$

It can be shown (Searle et al., 1992) that

$$\mathbf{R} \equiv \mathbf{V}^{-1} - \mathbf{V}^{-1}\mathbf{X}_0\left(\mathbf{X}'_0\mathbf{V}^{-1}\mathbf{X}_0\right)^{-}\mathbf{X}'_0\mathbf{V}^{-1} = \mathbf{K}\left(\mathbf{K}'\mathbf{V}\mathbf{K}\right)^{-1}\mathbf{K}'$$

Thus, the log-likelihood function is

$$l = \log L = -\tfrac{n-r}{2}\log(2\pi) - \tfrac{n-r}{2}\log\left(\sigma_e^2\right) - \tfrac{1}{2}\log\left|\mathbf{K}'\mathbf{V}\mathbf{K}\right| - \tfrac{1}{2\sigma_e^2}\mathbf{y}'\mathbf{R}\mathbf{y}.$$

Gradient Vector

$$\frac{\partial l}{\partial \gamma_i^2} = \frac{1}{2\sigma_e^2}\mathbf{y}'\mathbf{R}\mathbf{X}_i\mathbf{X}'_i\mathbf{R}\mathbf{y} - \frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_i\right) \quad i = 1, \ldots, k$$
$$\frac{\partial l}{\partial \sigma_e^2} = \frac{1}{2\sigma_e^4}\mathbf{y}'\mathbf{R}\mathbf{y} - \frac{(n-r)}{2\sigma_e^2}$$

The gradient vector is

$$\frac{dl}{d\theta} = \begin{bmatrix} \frac{\partial l}{\partial \gamma^2} \\ \frac{\partial l}{\partial \sigma_e^2} \end{bmatrix}$$

Hessian Matrix

$$\frac{\partial^2 l}{\partial \gamma_i^2 \partial \gamma_j^2} = -\frac{1}{\sigma_e^2}\mathbf{y}'\mathbf{R}\mathbf{X}_i\mathbf{X}'_i\mathbf{R}\mathbf{X}_j\mathbf{X}'_j\mathbf{R}\mathbf{y} + \frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_j\mathbf{X}'_j\mathbf{R}\mathbf{X}_i\right) \quad i = 1, \ldots, k; j = 1, \ldots, k$$
$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma_j^2} = -\frac{1}{2\sigma_e^4}\mathbf{y}'\mathbf{R}\mathbf{X}_j\mathbf{X}'_j\mathbf{R}\mathbf{y} \quad j = 1, \ldots, k$$
$$\frac{\partial^2 l}{\partial \sigma_e^2 \partial \sigma_e^2} = -\frac{1}{\sigma_e^6}\mathbf{y}'\mathbf{R}\mathbf{y} + \frac{n-r}{2\sigma_e^4}$$

The Hessian matrix is

$$\frac{d^2 l}{d\theta d\theta} = \begin{bmatrix} \frac{\partial^2 l}{\partial \gamma^2 \partial \gamma^2} & \frac{\partial^2 l}{\partial \gamma^2 \partial \sigma_e^2} \\ \frac{\partial^2 l}{\partial \sigma_e^2 \partial \gamma^2} & \frac{\partial^2 l}{\partial \sigma_e^2 \partial \sigma_e^2} \end{bmatrix}$$

where

$$\frac{\partial^2 l}{\partial \gamma^2 \partial \gamma^2} = \begin{bmatrix} \frac{\partial^2 l}{\partial \gamma_1^2 \partial \gamma_1^2} & \cdots & \frac{\partial^2 l}{\partial \gamma_1^2 \partial \gamma_k^2} \\ \vdots & \ddots & \vdots \\ \frac{\partial^2 l}{\partial \gamma_k^2 \partial \gamma_1^2} & \cdots & \frac{\partial^2 l}{\partial \gamma_k^2 \partial \gamma_k^2} \end{bmatrix}$$

and

$$\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma^2} = \begin{bmatrix} \frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_1^2} \\ \vdots \\ \frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_k^2} \end{bmatrix}$$

Fisher Information Matrix

Since $\mathbf{K}'\mathbf{X}_0 = \mathbf{0}_{(n-r)\times m_0}$ and $trace(\mathbf{RV}) = n - r$, the expected second derivatives are

$$E\left(\frac{\partial^2 l}{\partial\gamma_i^2\partial\gamma_j^2}\right) = -\frac{1}{2}tr\left(\mathbf{X}'_i\mathbf{R}\mathbf{X}_j\mathbf{X}'_j\mathbf{R}\mathbf{X}_i\right) \quad i = 1,\ldots,k, j = 1,\ldots,k$$

$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_j^2}\right) = -\frac{1}{2\sigma_e^2}tr\left(\mathbf{X}'_j\mathbf{R}\mathbf{X}_j\right) \quad j = 1,\ldots,k$$
$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\sigma_e^2}\right) = -\frac{n-r}{2\sigma_e^4}$$

The Fisher Information matrix is

$$E\left(\frac{d^2 l}{d\theta d\theta}\right) = \begin{bmatrix} E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\gamma^2}\right) & E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\sigma_e^2}\right) \\ E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma^2}\right) & E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\sigma_e^2}\right) \end{bmatrix}$$

where

$$E\left(\frac{\partial^2 l}{\partial\gamma^2\partial\gamma^2}\right) = \begin{bmatrix} E\left(\frac{\partial^2 l}{\partial\gamma_1^2\partial\gamma_1^2}\right) & \cdots & E\left(\frac{\partial^2 l}{\partial\gamma_1^2\partial\gamma_k^2}\right) \\ \vdots & \ddots & \vdots \\ E\left(\frac{\partial^2 l}{\partial\gamma_k^2\partial\gamma_1^2}\right) & \cdots & E\left(\frac{\partial^2 l}{\partial\gamma_k^2\partial\gamma_k^2}\right) \end{bmatrix}$$

and

$$E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial g^2}\right) = \begin{bmatrix} E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_1^2}\right) \\ \vdots \\ E\left(\frac{\partial^2 l}{\partial\sigma_e^2\partial\gamma_k^2}\right) \end{bmatrix}$$

Iteration Procedure

The iterative estimation algorithm proceeds according to the details in the following sections.

Initial Values

- Random Effect Variance Components: For the *i*th random effect, compute $\hat\beta_i = \left(\mathbf{X}'_i\mathbf{X}_i\right)^{-}\mathbf{X}'_i\mathbf{y}$. Then assign the variance of the m_i elements of b$_i$ using divisor $(m_i - 1)$ to the estimate $\hat\sigma_i^2$ if $m_i \geq 2$; otherwise $\hat\sigma_i^2 = 0$.
- Residual Variance: $\hat\sigma_e^2 = \mathbf{r}'\mathbf{r}/n$ where $\mathbf{X} = [\mathbf{X}_0|\mathbf{X}_1|\cdots|\mathbf{X}_k]$ and $\mathbf{r} = \mathbf{y} - \left(\mathbf{X}'\mathbf{X}\right)^{-}\mathbf{X}'\mathbf{y}$. If $\hat\sigma_e^2 = 0$ but $k \geq 1$ then reset $\hat\sigma_e^2 = 10^{-8}$ so that the iteration can continue.

The variance ratios are then computed as $\hat\gamma_i^2 = \hat\sigma_i^2/\hat\sigma_e^2$. Following the same method in which the residual variance is initialized, $\hat\sigma_e^2 > 0$ for $k \geq 1$.

Updating

At the *s*th iteration, *s*=0,1,..., the parameter vector is updated as

$$\hat{\theta}_{(s+1)} = \hat{\theta}_s + \rho\Delta\hat{\theta}_s$$

where $\Delta\hat{\theta}_s$ is the value of increment $\Delta\theta$ evaluated at $\theta = \hat{\theta}_s$, and ρ>0 is a step size such that $l\left(\hat{\theta}_{(s+1)}\right) > l\left(\hat{\theta}_s\right)$. The increment vector depends on the choice of step type—Newton-Raphson versus Fisher scoring. The step size is determined by the step-halving technique with ρ=1 initially and a maximum of 10 halvings.

Choice of Step

Following Jennrich and Sampson (1976), the first iteration is always the Fisher scoring step because it is more robust to poor initial values. For subsequent iterations the Newton-Raphson step is used if:

1. The Hessian matrix is nonnegative definite, and
2. The increment in the log-likelihood function of step 1 is less than or equal to one.

Otherwise the Fisher scoring step is used. The increment vector for each type of step is:

- Newton-Raphson Step: $\Delta\theta = \left(-\frac{d^2 l}{d\theta d\theta}\right)^{-1}\frac{dl}{d\theta}$.
- Fisher Scoring Step: $\Delta\theta = \left(-E\left(\frac{d^2 l}{d\theta d\theta}\right)\right)^{-1}\frac{dl}{d\theta}$.

Convergence Criteria

Given the convergence criterion $\epsilon > 0$, the iteration is considered converged when the following criteria are satisfied:

Given the convergence criterion $\epsilon > 0$, the iteration is considered converged when the following criteria are satisfied:

1. $\left| l\left(\hat{\theta}_{(s+1)}\right) - l\left(\hat{\theta}_s\right)\right| < \epsilon \times \max\left(1, \left|\hat{\theta}_s\right|\right)$, and
2. $< l\left(\hat{\theta}_{(s+1)}\right) - l\left(\hat{\theta}_s\right) >< \epsilon \times \max\left(1, <\hat{\theta}_s>\right)$ where <**a**> is the sum of absolute values of elements of the vector **a**.

Negative Variance Estimates

Negative variance estimates can occur at the end of an iteration. An *ad hoc* method is to set those estimates to zero before the next iteration.

Covariance Matrix

Let $\hat{\theta}$ be the vector of maximum likelihood estimates. Their covariance matrix is given by

$$cov\left(\hat{\theta}\right) = \left(-E\left(\frac{d^2 l}{d\theta d\theta}\right)\Big|_{\theta=\hat{\theta}}\right)^{-1}$$

Let

$$\mathbf{\Psi} = \begin{bmatrix} \sigma_1^2 \\ \vdots \\ \sigma_k^2 \\ \sigma_e^2 \end{bmatrix}$$

be the original parameters. Their maximum likelihood estimates are given by

$$\hat{\mathbf{\Psi}} = \begin{bmatrix} \hat{\sigma}_e^2 \hat{\gamma}_1^2 \\ \vdots \\ \hat{\sigma}_e^2 \hat{\gamma}_k^2 \\ \hat{\sigma}_e^2 \end{bmatrix}$$

and their covariance matrix is estimated by

$$cov\left(\hat{\mathbf{\Psi}}\right) = \mathbf{J} cov\left(\hat{\theta}\right)\mathbf{J}'$$

where

$$\mathbf{J} = \begin{bmatrix} \sigma_e^2 \mathbf{I}_k & \gamma \\ 0 & 1 \end{bmatrix}$$

which is the $(k+1) \times (k+1)$ Jacobian matrix of transforming **θ** to **ψ**.

ANOVA

The ANOVA variance component estimates are obtained by equating the expected mean squares of the random effects to their observed mean squares. The VARCOMP procedure offers two types of sum of squares: Type I and Type III (see Appendix 11 for details).

Let

$$\mathbf{\Psi} = \begin{bmatrix} \sigma_1^2 \\ \\ \cdot \\ \sigma_k^2 \\ \sigma_e^2 \end{bmatrix}$$

be the vector of variance components.

Let

$$\mathbf{q} = \begin{bmatrix} MS_1 \\ \\ \cdot \\ MS_k \\ MSE \end{bmatrix}$$

where MS_i is the observed mean squares of the *i*th random effect, and *MSE* is the residual mean squares.

Let

$$\mathbf{S} = \begin{bmatrix} \mathbf{s}'_1 \\ \vdots \\ \mathbf{s}'_k \\ \mathbf{s}'_{k+1} \end{bmatrix}$$

be a $(k+1) \times (k+1)$ matrix whose rows are coefficients for the expected mean squares. For example, the expected mean squares of the *i*th random effect is $\mathbf{s}'_i$y. Algorithms for computing the expected mean squares can be found in the section "Univariate Mixed Model" in the chapter GLM Univariate and Multivariate. The ANOVA variance component estimates are then obtained by solving the system of linear equations:

$$\mathbf{S}\text{y} = \mathbf{q}$$

References

Giesbrecht, F. G. 1983. An efficient procedure for computing MINQUE of variance components and generalized least squares estimates of fixed effects. *Communications in Statistics, Part A - Theory and Methods*, 12, 2169–2177.

Goodnight, J. H. 1978. Computing MIVQUE0 Estimates of Variance Components. *SAS Technical Report*, R-105, – .

Goodnight, J. H., and W. J. Hemmerle. 1979. A simplified algorithm for the W transformation in variance component estimation. *Technometrics*, 21, 265–267.

Jennrich, R. I., and P. F. Sampson. 1976. Newton-Raphson and related algorithms for maximum likelihood variance component estimation. *Technometrics*, 18, 11–17.

Rao, C. R. 1973. *Linear statistical inference and its applications*, 2nd ed. New York: John Wiley and Sons.

Searle, S. R., G. Casella, and C. E. McCulloch. 1992. *Variance Components*. New York: John Wiley and Sons.

WLS Algorithms

WLS estimates regression model with different weights for different cases.

Notation

The following notation is used throughout this chapter unless otherwise stated:

n	The number of cases
p	The number of parameters for the model
y	*n*×1 vector with element y_i, which represents the observed dependent variable for case *i*
X	*n*×*p* matrix with element x_{ij}, which represents the observed value of the *i*th case of the *j*th independent variable
β	*p*×1 vector with element β_j, which represents the regression coefficient of the *j*th independent variable
w	*n*×1 vector with element w_i, which represents the weight for case *i*

Model

The linear regression model has the form of

$$y_i = \mathbf{X}_i'\beta + \epsilon_i, \quad i = 1, \ldots, n$$

where $\mathbf{X}_i$ is the vector of covariates for the *i*th case, $\mathbf{E}(\epsilon_i) = 0$, and $var(\epsilon_i) = w_i^{-1}\sigma^2$. Assuming that $\epsilon_1, \ldots, \epsilon_n$ follow a normal distribution, the log-likelihood function is

$$L = 0.5\left\{ -n\ln 2\pi - n\ln\sigma^2 + \sum_{i=1}^{n}\ln w_i - \frac{\sum_{i=1}^{n} w_i\left(y_i - \mathbf{X}_i'\beta\right)^2}{\sigma^2} \right\}$$

Computational Details

The algorithm used to obtain the weighted least-square estimates for the parameters in the model is the same as the REGRESSION procedure with regression weight. For details of the algorithm and statistics (the ANOVA table and the variables in the equation), see REGRESSION.

After the estimation is finished, the log-likelihood function is estimated by

$$\hat{L} = 0.5\left\{ -n\ln 2\pi - n\ln\hat{\sigma}^2 + \sum_{i=1}^{n}\ln w_i - (n-p) \right\}$$

where $\hat{\sigma}^2$ is the mean square error in the ANOVA table.

Appendix

A

Significance Level of a Standard Normal Deviate

The significance level is based on a polynomial approximation.

Notation

The following notation is used throughout this section unless otherwise stated:

Table A-1
Notation

Notation	Description
X	Value of the standard normal deviate
Q	One-sided significance level

Computation

$$Q(X) = 0.5\{1 + Z(a_1 + Z(a_2 + Z(a_3 + Z(a_4 + Z(a_5 + Za_6)))))\}^{-16}$$

where

$$Z = \begin{cases} 0.7071067812|X| & \text{if} |X| \le 14.14 \\ 10 & \text{otherwise} \end{cases}$$

$$\begin{array}{ll} a_1 = 0.070523078\ldots & a_4 = 0.0001520143 \\ a_2 = 0.0422820123 & a_5 = 0.0002765672 \\ a_3 = 0.0092705272 & a_6 = 0.0000430638 \end{array}$$

References

Abramowitz, M., and I. A. Stegun, eds. 1970. *Handbook of mathematical functions*. New York: Dover Publications.

Ling, R. E. 1978. A study of the accuracy of some approximations for t, chi-square, and F tail probabilities. *Journal of American Statistical Association*, 73, 274–283.

Appendix

B

Significance Levels for Fisher's Exact Test

The procedure described in this appendix is used to calculate the exact one-tailed and two-tailed significance levels of Fisher's exact test for a 2×2 table under the assumption of independence of rows and columns and conditional on the marginal totals. All cell counts are rounded to the nearest integers.

Background

Consider the following observed 2×2 table:

Table B-1
2 x 2 table

	Column 1	**Column 2**	**Column total**
Row 1	n1	n2	n1+n2
Row 2	n3	n4	n3+n4
Row total	n1+n3	n2+n4	N

Conditional on the observed marginal totals, the values of the four cell counts can be expressed as the observed count of the first cell n_1 only. Under the hypothesis of independence, the count of the first cell N_1 follows a hypergeometric distribution with the probability of $N_1 = n_1$ given by

$$\text{Prob}(N_1 = n_1) = \frac{(n_1+n_2)!(n_3+n_4)!(n_1+n_3)!(n_2+n_4)!}{N!n_1!n_2!n_3!n_4!}$$

where N_1 ranges from $\max(0, n_1 - n_4)$ to $\min(n_1 + n_2, n_1 + n_3)$ and $N = n_1 + n_2 + n_3 + n_4$.

The exact one-tailed significance level p_1 is defined as

$$p_1 = \begin{cases} \text{Prob}(N_1 \geq n_1) & \text{if } n_1 > E(N_1) \\ \text{Prob}(N_1 \leq n_1) & \text{if } n_1 \leq E(N_1) \end{cases}$$

where $E(N_1) = (n_1 + n_2)(n_1 + n_3)/N$.

The exact two-tailed significance level p_2 is defined as the sum of the one-tailed significance level p_1 and the probabilities of all points in the other side of the sample space of N_1 which are not greater than the probability of N_1=n_1.

Computations

To begin the computation of the two significance levels p_1 and p_2, the counts in the observed 2×2 table are rearranged. Then the exact one-tailed and two-tailed significance levels are computed using the CDF.HYPER cumulative distribution function.

Table Rearrangement

The following steps are used to rearrange the table:

1. Check whether $n_1 > E(N_1)$, which can be done by checking whether $n_1 n_4 > n_2 n_3$. If so, rearrange the table so that the first cell contains the minimum of n_2 and n_3, maintaining the row and column totals; otherwise, rearrange the table so that the first cell contains the minimum of n_1 and n_4, again maintaining the row and column totals.

2. Without loss of generality, we assume that the count of the first cell is n_1 after the above rearrangement. Calculate the first row total, the first column total, and the overall total, and name them *SAMPLE*, *HITS*, and *TOTAL*, respectively.

One-Tailed Significance Level

The following steps are used to calculate the one-tailed significance level:

1. If *TOTAL*=0, set the one-tailed significance level p_1 equal to 1; otherwise, obtain p_1 by using the CDF.HYPER cumulative distribution function with arguments n_1, *SAMPLE*, *HITS*, and *TOTAL*.

2. Also calculate the probability of the first cell count equal to n_1 by finding the difference between p_1 and the value obtained from CDF.HYPER with n_1-1, *SAMPLE*, *HITS*, and *TOTAL* as its arguments, provided that $n_1>0$. Call this probability *PEXACT*.

3. If $n_1=0$, set *PEXACT*=p_1. *PEXACT* will be used in the next step to find the points for which the probabilities are not greater than *PEXACT*.

Two-Tailed Significance Level

The following steps are used to calculate the two-tailed significance level:

1. If *TOTAL*=0, set the two-tailed significance level p_2 equal to 1; otherwise, start searching backwards from $\min(n_1+n_2, n_1+n_3)$ to (n_1+1), and find the first point x with its point probability greater than *PEXACT*. (Notice that this backward search takes advantage of the unimodal property of the hypergeometric distribution.)

2. If such an x exists between $\min(n_1+n_2, n_1+n_3)$ and (n_1+1), calculate the probability value obtained from CDF.HYPER with arguments x, *SAMPLE*, *HITS*, and *TOTAL*. Call this probability p_x.

3. The two-tailed significance level p_2 is obtained by finding the sum of p_1 and $(1 - p_x)$. If no qualified x exists, the two-tailed significance level is equal to 1.

Appendix C
Sorting and Searching

Sorting and searching have a significant impact on the performance of a number of procedures. For those procedures, the methods used are identified here.

CROSSTABS

In the general mode, the table of cells is searched using an unordered open scatter table search and insertion algorithm similar to Knuth's Algorithm *L* (Knuth, 1973, p. 518). The scatter table contains only pointers to the actual cell contents and is twice as large as it need be (that is, if there is room for *m* cells, the scatter table has room for 2*m* pointers). This means it can never be more than half full. Collisions are resolved by sequential search from the initial location until an empty pointer is found.

Letting

k be the table number

p be the dimension of the table

v(*i*), *i*=1,...,*p*, be the bit string used to represent the value of the *i*th variable defining table *k*

m be the length of the scatter table

n be the resulting hash value, to be used as an index in the scatter table

The hash function used is given by the following algorithm:

```
j:=k
for i:=1 to p
  j:=j rotated left 3
  bits j:=j
  EXCLUSIVE OR
  v(i)
end
n:=(j modulo m)+1
```

When the tables have been completed, the cells are sorted by table numbers and the values of the defining variables using the algorithm described by Singleton (1969).

FREQUENCIES

FREQUENCIES uses the same search and sort algorithms as CROSSTABS, except that its hashing function is given by:

$$h = \left(\left((k + 16807v)\text{modulo } 2^{31}\right)\text{modulo} m\right) + 1$$

where

h is the hash value, to be used as an index in the scatter table

k is the table number

v is the integer value of the bits representing the value to be tabulated

m is the length of the scatter table

NONPAR CORR and NPAR TESTS

Both use the method of Singleton to sort cases for computing ranks.

SURVIVAL

SURVIVAL uses a modified Quicksort similar to Knuth's algorithm *Q* (Knuth, 1973, p. 116) to sort cases.

References

Knuth, D. E. 1973. *The Art of Computer Programming, volume3: Sorting and Searching*. Reading, MA: Addison-Wesley.

Singleton, R. C. 1969. Algorithm 347: Efficient sorting in minimal storage. *Communications of the ACM*, 12, 185–187.

Appendix

D

Generation of Uniform Random Numbers

Two different random number generators are available:

- **Version 12 Compatible.** The random number generator used in version 12 and previous releases. If you need to reproduce randomized results generated in previous releases based on a specified seed value, use this random number generator.
- **Mersenne Twister.** A newer random number generator that is more reliable for simulation purposes. If reproducing randomized results from version 12 or earlier is not an issue, use this random number generator.

Specifically, the Mersenne Twister has a far longer period (number of draws before it repeats) and far higher order of equidistribution (its results are “more uniform”) than the IBM® SPSS® Statistics 12 Compatible generator. The Mersenne Twister is also very fast and uses memory efficiently.

IBM SPSS Statistics 12 Compatible Random Number Generator

Uniform numbers are generated using the algorithm of (Fishman and Moore, 1981). It is a multiplicative congruential generator that is simply stated as:

```
seed(t+1) = (a * seed(t))
modulo p rand = seed(t+1) /
(p+1)
```

where a = 397204094 and p = $2^{31}-1$ = 2147483647, which is also its period. Seed(t) is a 32-bit integer that can be displayed using `SHOW SEED`. `SET SEED=`*number* sets seed(t) to the specified number, truncated to an integer. `SET SEED=RANDOM` sets seed(t) to the current time of day in milliseconds since midnight.

Mersenne Twister Random Number Generator

The Mersenne Twister (MT) algorithm generates uniform 32-bit pseudorandom integers. The algorithm provides a period of $2^{19937}-1$, assured 623-dimensional equal distribution, and 32-bit accuracy. Following the description given by Matsumoto and Nishimura (1998), the algorithm is based on the linear recurrence:

$$\mathbf{x}_{k+n} = \mathbf{x}_{k+m} \oplus \left(\mathbf{x}_k^u | \mathbf{x}_{k+1}^l\right) \mathbf{A},\ k = 0, 1, \cdots$$

where

Table D-1
Notation

Notation	**Description**
$\mathbf{x}$	is a word vector; a *w*-dimensional row vector over the two-element field $\mathbf{F}_2 = \{0, 1\}$
n	is the degree of recurrence (recursion)
r	is an integer, $0 \le r \le w - 1$, the separation point of one word
m	is an integer, $1 \le m \le n$, the middle term
$\mathbf{A}$	is a constant $w \times w$ matrix with entries in $\mathbf{F}_2$
$\mathbf{x}_k^u$	is the upper (*w*–*r*) bits of $\mathbf{x}_k$
$\mathbf{x}_{k+1}^l$	is the lower *r* bits of $\mathbf{x}_{k+1}$; thus
$\mathbf{x}_k^u \vert \mathbf{x}_{k+1}^l$	is the word vector obtained by concatenating the upper (*w*–*r*) bits of $\mathbf{x}_k$ and the lower *r* bits of $\mathbf{x}_{k+1}$
$\oplus$	Bitwise addition modulo two (XOR)

Given initial seeds $\mathbf{x}_0, \mathbf{x}_1, \cdots, \mathbf{x}_{n-1}$, the algorithm generates $\mathbf{x}_{n+k}$ by the above recurrence for *k*=0, 1, ...

A form of the matrix **A** is chosen so that multiplication by **A** is very fast. **A** candidate is

$$\mathbf{A} = \begin{pmatrix} & 1 & & & \\ & & 1 & & \\ & & & \ddots & \\ & & & & 1 \\ a_{w-1} & a_{w-2} & \cdots & \cdots & a_0 \end{pmatrix}$$

where $\mathbf{a} = (a_{w-1}, a_{w-2}, \cdots, a_0)$ and $\mathbf{x} = (x_{w-1}, x_{w-2}, \cdots, x_0)$; then $\mathbf{xA}$ can be computed using only bit operations

$$\mathbf{xA} = \begin{cases} shiftright(\mathbf{x}) & if \quad x_0 = 0 \\ shiftright(\mathbf{x}) \oplus \mathbf{a} & if \quad x_0 = 1 \end{cases}$$

Thus calculation of the recurrence is realized with bitshift, bitwise EXCLUSIVE-OR, bitwise OR, and bitwise AND operations.

For improving the *k*-distribution to *v*-bit accuracy, we multiply each generated word by a suitable $w \times w$ invertible matrix $\mathbf{T}$ from the right (called tempering in (Matsumoto and Kurita, 1994)). For the tempering matrix $\mathbf{z} = \mathbf{xT}$, we choose the following successive transformations

$$\mathbf{y} := \mathbf{x} \oplus (\mathbf{x} >> u)$$

$$\mathbf{y} := \mathbf{y} \oplus ((\mathbf{y} << s) \;\; AND \; \mathbf{b})$$

$$\mathbf{y} := \mathbf{y} \oplus ((\mathbf{y} << t) \;\; AND \; \mathbf{c})$$

$$\mathbf{z} := \mathbf{y} \oplus ((\mathbf{y} >> l)$$

where

Table D-2
Notation

Notation	Description
l, s, t, u	are integers
b, **c**	are suitable bitmasks of word size
$\mathbf{x} >> u$	denotes the u-bit shiftright
$\mathbf{x} << u$	denotes the u-bit shiftleft

To execute the recurrence, let **x**[0:*n*−1]be an array of *n* unsigned integers of word size, *i* be an integer variable, and $\mathbf{x}, \mathbf{v}, \mathbf{a}$ be unsigned constant integers of word size vectors.

Step	Description
Step 0	$\mathbf{u} \leftarrow \underset{w-r}{1\cdots1}\;\underset{r}{0\cdots0}$; bitmask for upper (*w*−*r*) bits $\mathbf{v} \leftarrow \underset{w-r}{0\cdots0}\;\underset{r}{1\cdots1}$; bitmask for lower r bits $\mathbf{a} \leftarrow a_{w-1}a_{w-2}\cdots a_0$; the last row of matrix $\mathbf{A}$
Step 1	$i \leftarrow 0$ Initialize the state space vector array $\mathbf{x}[0], \mathbf{x}[1], \cdots, \mathbf{x}[n-1]$.
Step 2	$\mathbf{y} \leftarrow (\mathbf{x}[i]\ AND\ \mathbf{u})\ OR\ (\mathbf{x}[(i+1)\ mod\, n\ ])AND\ \mathbf{v}$; computing $(\mathbf{x}_i^u \mid \mathbf{x}_{i+1}^l)$
Step 3	If the least significant bit of $\mathbf{y}$ equals to zero then $\mathbf{x}[i] \leftarrow \mathbf{x}[(i+m)\ mod\, n] \oplus (\mathbf{y} >> 1) \oplus 0$ If the least significant bit of $\mathbf{y}$ equals to one then $\mathbf{x}[i] \leftarrow \mathbf{x}[(i+m)\ mod\, n] \oplus (\mathbf{y} >> 1) \oplus \mathbf{a}$
Step 4	calculate $\mathbf{x}[i]\mathbf{T}$ $\mathbf{y} \leftarrow \mathbf{x}[i]$ $\mathbf{y} \leftarrow \mathbf{y} \oplus (\mathbf{y} >> u)$ $\mathbf{y} \leftarrow \mathbf{y} \oplus ((\mathbf{y} << s)\ AND\, \mathbf{b})$ $\mathbf{y} \leftarrow \mathbf{y} \oplus ((\mathbf{y} << t)\ AND\, \mathbf{c})$ $\mathbf{y} \leftarrow \mathbf{y} \oplus (\mathbf{y} >> l)$
Step 5	$i \leftarrow (i+1)\ mod\, n$
Step 6	Go to **Step 2**.

IBM SPSS Statistics Usage

The MT algorithm provides 32 random bits in each draw. IBM® SPSS® Statistics draws 64-bit floating-point numbers in the range [0..1] with 53 random bits in the mantissa using

Draw = $(2^{26}$*[k(t)/2^5]+[k(t+1)/2^6])/2^{53}

There are two options for initializing the state space vector array. `SET RNG=MT MTINDEX=`*x* accepts a 64-bit floating point number *x* to set the seed. `SET RNG=MT MTINDEX=RANDOM` uses the current time of day in milliseconds since midnight to set the seed.

init_genrand(unsigned32 s,unsigned32 &x[])
{

$\mathbf{x}[0] = s$;
f = 1812433253; f is an unsigned long interger from i=0 to n

$$\mathbf{x}[i+1] = f(\mathbf{x}[i] >> 30)\, mod 2^{w}$$

k[0]: 8*d+4*c+2*b+a

k[1]: y = trunc(z*2^{26})

k[2]: z*2^{53} - y*2^{27}

where

- x is the argument a is 1 if x == 0, or 0 otherwise
- b is 1 if x<0, or 0 otherwise
- c is 1 if |x| >= 1, or 0 otherwise
- d is an integer such that
- if |x| > 1, .5 <= |x|/2^{d} < 1,

  else if |x| > 0, .5 <= |x|*2^{d} < 1

  else x == 0 and d == 0.
- e is d if |x| <= 1, else -d
- z is |x|*2^{e}

```
init_by_array(unsigend32 init_key[ ] ,int key_length, unsigned32 &x[])
{
 init_genrand(19650218,
 x); i=1, j=0,
 k=max(key_length,n) for
 (;k;k--)
  x[i] = (x[i]□((x[i-1]□(x[i-1]>>30))f1))
  +init_key[j]+
  j; if i>=n
  then x[0] =
  x[n-1]
   i=1
  if (j>=key_length)
   then j=0
 end for
 for (k=n-1;k;k--)
  x[i] = (x[i]□((x[i-1]□(x[i-
  1]>>30))f2))-i; if i>=n then
   x[0]=x[n-1];
   i=1;
 end for
}
f1=1664525 is an unsigned long
interger; f2=1566083941 is an
unsigned long interger;
```

References

Fishman, G., and L. R. I. Moore. 1981. In search of correlation in multiplicative congruential generators with modulus 2**31-1. In: *Computer Science and Statistics, Proceedings of the 13th Symposium on the Interface*, W. F. Eddy, ed. New York: Springer-Verlag, 155–157.

Knuth, D. E. 1981. *The Art of Computer Programming, volume 2, p. 106*. Reading, MA: Addison-Wesley.

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister, A 623–dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. on Modeling and Computer Simulation*, 8:1, 3–30.

Matsumoto, M., and Y. Kurita. 1994. Twisted GFSR generators II. *ACM Trans. on Modeling and Computer Simulation*, 4:3, 254–266.

Appendix

E

Grouped Percentiles

Two summary functions, GMEDIAN and GPTILE are used in procedures such as Frequencies and Graph, to calculate the percentiles for the data which are grouped by specifying a value for each grouping. It is assumed that the actual data values give represent midpoints of the grouped intervals.

Notation

The following notation is used throughout this section unless otherwise stated:

Table E-1
Notation

Notation	Description
$x_i < \ldots < x_k$	Distinct observed values with frequencies (caseweights) $c_1, \ldots, c_k$
k	Number of distinct observed data points
p	percentile/100 (a number between 0 and 1)
cc_l	Cumulative frequency up to and including x_l

$$cc_l = \sum_{i=1}^{l-1} c_i + 0.5 * c_l l = 1, \ldots, n$$

Finding Percentiles

To find the 100*p*th grouped percentile, first find *i* such that $cc_{i-1} \le wp < cc_i$, where $w=\sum_{j=1}^{k} c_j$, the total sum of caseweights. Then the grouped percentile is

$$(1-R)x_{i-1} + Rx_i$$

where

$$R = \frac{wp - cc_{i-1}}{cc_i - cc_{i-1}}$$

Note the following:

- If $wp < cc_1$, the grouped percentile is system missing and a warning message “Since the lower bound of the first interval is unknown, some percentiles are undefined” is produced.
- If $wp > cc_k$, the grouped percentile is system missing and a warning message “Since the upper bound of the last interval is unknown, some percentiles are undefined” is produced.
- If $wp = cc_k$, the grouped percentile is equal to x_k.

Appendix

F

Indicator Method

The indicator method is used in the GENLOG and the GLM procedures to generate the design matrix corresponding to the design specified. Under this method, each parameter (either non-redundant or redundant) in the model corresponds uniquely to a column in the design matrix. Therefore, the terms parameter and design matrix column are often used interchangeably without ambiguity.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table F-1
Notation

Notation	**Description**
n	Number of valid observations
p	Number of parameters
$\mathbf{X}$	$n \times p$ design matrix (also known as model matrix)
x_{ij}	Elements of $\mathbf{X}$

Row Dimension

The design matrix has as many rows as the number of valid observations. In the GLM procedure, an observation is a case in the data file. In the GENLOG procedure, an observation is a cell. In both procedures, the observations are uniquely identified by the factor-level combination. Therefore, rows of the design matrix are also uniquely identified by the factor-level combination.

Column Dimension

The design matrix has as many columns as the number of parameters in the model. Columns of the design matrix are uniquely indexed by the parameters, which are in turn related to factor-level combinations.

Elements

A factor-level combination is contained in another factor-level combination if the following conditions are true:

- All factor levels in the former combination appear in the latter combination.
- There are factor levels in the latter combination which do not appear in the former combination.

For example, the combination [A=1] is contained in [A=1]*[B=3] and so is the combination [B=3]. However, neither [A=3] nor [C=1] is contained in [A=1]*[B=3].

The design matrix **X** is generated by rows. Elements of the *i*th row are generated as follows:

- If the *j*th column corresponds to the intercept term, then $x_{ij} = 1$.
- If the *j*th column is a parameter of a factorial effect which is constituted of factors only, then $x_{ij} = 1$ if the factor-level combination of the *j*th column is contained in that of the *i*th row. Otherwise $x_{ij} = 0$.
- If the *j*th column is a parameter of an effect involving covariates (or, in the GLM procedure, a product of covariates), then x_{ij} is equal to the covariate value (or the product of the covariate values in GLM) of the *i*th row if the levels combination of the factors of the *j*th column is contained in that of the *i*th row. Otherwise $x_{ij} = 0$.

Redundancy

A parameter is redundant if the corresponding column in the design matrix is linearly dependent on other columns. Linear dependent columns are detected using the SWEEP algorithm by Clarke (1982) and Ridout and Cobby (1989). Redundant parameters are permanently set to zero and their standard errors are set to system missing.

References

Clarke, M. R. B. 1982. Algorithm AS 178: The Gauss-Jordan sweep operator with detection of collinearity. *Applied Statistics*, 31:2, 166–168.

Ridout, M. S., and J. M. Cobby. 1989. A remark on algorithm AS 178. *Applied Statistics*, 38, 420–422.

Post Hoc Tests

Post hoc tests are available in more than one procedure, including ONEWAY and GLM.

Notation

The following notation is used throughout this section unless otherwise stated:

Table G-1
Notation

Notation	**Description**
k	Number of levels for an effect
n_i	Number of observations at level *i*
$\overline{x}_i$	Mean at level *i*
s_i	Standard deviation of level *i*
v_i	Degrees of freedom for level *i*, $n_i - 1$
s_{pp}	Square root of the mean square error
ε	Experimentwise error rate under the complete null hypothesis
α	Comparisonwise error rate
r	Number of steps between means
f	Degrees of freedom for the within-groups mean square $\sum_{i=1}^{k}(n_i - 1)$
$v_{i,j}$	Absolute difference between the ith and jth means $\overline{x}_i - \overline{x}_j$
k^*	$k(k-1)/2$
$Q_{i,j}$	$s_{pp}\sqrt{\frac{1}{2}\left(\frac{1}{n_i}+\frac{1}{n_j}\right)}$
n_h	Harmonic mean of the sample size $n_h = \frac{k}{\sum_{1 \le i \le k} n_i^{-1}}$
Q_h	$s_{pp}/\sqrt{n_h}$

Studentized Range and Studentized Maximum Modulus

Let $x_1, x_2, \ldots, x_r$ be independent and identically distributed $N(\mu, \sigma)$. Let s_m be an estimate of σ with *m* degrees of freedom, which is independent of the $\{x_i\}$, and $ms_m^2/\sigma^2 \sim \chi^2$. Then the quantity

$$S_{r,m} = \frac{\max(x_1,\ldots,x_r) - \min(x_1,\ldots,x_r)}{s_m}$$

is called the Studentized range. The upper-ε critical point of this distribution is denoted by $S_{\epsilon,r,m}$.

The quantity

$$M_{r,m} = \frac{\max(|x_1|,\ldots,|x_r|)}{s_m}$$

is called the Studentized maximum modulus. The upper-ε critical point of this distribution is denoted as $M_{\epsilon,r,m}$.

Methods

The tests are grouped as follows according to assumptions about sample sizes and variances.

Equal Variances

The tests in this section are based on the assumption that variances are equal.

Waller-Duncan t Test

The Waller-Duncan *t* test statistic is given by

$$v_{i,j} = \overline{x}_i - \overline{x}_j \geq t_B\,(w,F,q,f)\,S\sqrt{2/n}$$

where $t_B(w, F, q, f)$ is the Bayesian *t* value that depends on *w* (a measure of the relative seriousness of a Type I error versus a Type II error), the *F* statistic for the one-way ANOVA,

$$F = \frac{MS_{treat}}{MS_{error}}$$

and

$$S^2 = MS_{error}$$

Here $f = k(n-1)$ and $q = k-1$. MS_{error} and MS_{treat} are the usual mean squares in the ANOVA table.

Only homogeneous subsets are given for the Waller-Duncan *t* test. This method is for equal sample sizes. For unequal sample sizes, the harmonic mean n_h is used instead of *n*.

Constructing Homogeneous Subsets

For many tests assuming equal variances, homogeneous subsets are constructed using a range determined by the specific test being used. The following steps are used to construct the homogeneous subsets:

1. Rank the *k* means in ascending order and denote the ordered means as $\overline{x}_{(1)}, \ldots, \overline{x}_{(k)}$.
2. Determine the range value, $R_{\epsilon,k,f}$, for the specific test, as shown in Range Values.
3. If $\overline{x}_{(k)} - \overline{x}_{(1)} > Q_h R_{\epsilon,k,f}$, there is a significant range and the ranges of the two sets of $k-1$ means $\{\overline{x}_{(1)}, \cdots, \overline{x}_{(k-1)}\}$ and $\{\overline{x}_{(2)}, \ldots, \overline{x}_{(k)}\}$ are compared with $Q_h R_{\epsilon,k-1,f}$. Smaller subsets of means

are examined as long as the previous subset has a significant range. For some tests, $Q_{i,j}$ is used instead of Q_h. For more information, see the topic "Range Values".

4. Each time a range proves nonsignificant, the means involved are included in a single group—a homogeneous subset.

Range Values

Following are range values for the various types of tests.

Student-Newman-Keuls (SNK)

$$R_{\epsilon,r,f} = S_{\epsilon,r,f}$$

Tukey's Honestly Significant Difference Test (TUKEY)

$$R_{\epsilon,r,f} = S_{\epsilon,k,f}$$

The confidence intervals of the mean difference are calculated using $Q_{i,j}$ instead of Q_h.

Tukey's b (TUKEYB)

$$R_{\epsilon,r,f} = \frac{S_{\epsilon,r,f}+S_{\epsilon,k,f}}{2}$$

Duncan's Multiple Range Test (DUNCAN)

$R_{\epsilon,r,f} = S_{\alpha_r,r,f}$ where $\alpha_r = 1-(1-\epsilon)^{r-1}$

Scheffé Test (SCHEFFE)

$$R_{\epsilon,r,f} = \sqrt{2\,(\mathrm{k}-1)\,F_{1-\epsilon}\,(\mathrm{k}-1, f)}$$

The confidence intervals of the mean difference are calculated using $Q_{i,j}$ instead of Q_h.

Hochberg's GT2 (GT2)

$$R_{\epsilon,r,f} = \sqrt{2}M_{\epsilon,k^*,f}$$

The confidence intervals of the mean difference are calculated using $Q_{i,j}$ instead of Q_h.

Gabriel's Pairwise Comparisons Test (GABRIEL)

The test statistic and the critical point are as follows:

$$|\overline{x}_i - \overline{x}_j| \geq s_{pp}(\frac{1}{\sqrt{2n_i}} + \frac{1}{\sqrt{2n_j}})M_{\epsilon,k^*,f}$$

For homogeneous subsets, n_h is used instead of n_i and n_j. The confidence intervals of the mean difference are calculated based on the above equation.

Least Significant Difference (LSD), Bonferroni, and Sidak

For the least significant difference, Bonferroni, and Sidak tests, only pairwise confidence intervals are given. The test statistic is

$$\overline{x}_{\mathrm{i}} - \overline{x}_{\mathrm{j}} > Q_{i,j} R_{\epsilon,k,f}$$

where the range, $R_{\epsilon,k,f}$, for each test is provided below.

Least Significant Difference (LSD)

$$R_{\alpha,r,f} = \sqrt{2F_{1-\alpha}(1,f)}$$

Bonferroni t Test (BONFERRONI or MODLSD)

$$R_{\epsilon,r,f} = \sqrt{2F_{1-\alpha'}(1,f)}$$

where $\alpha^{'} = \epsilon / k^{*}$.

Sidak t Test (SIDAK)

$$R_{\epsilon,r,f} = \sqrt{2F_{1-\alpha,1,f}}$$

where $\alpha = 1 - (1 - \epsilon)^{\frac{2}{k(k-1)}}$.

Dunnett Tests

For the Dunnett tests, confidence intervals are given only for the difference between the control group and the other groups.

Dunnett’s Two-Tailed t Test (DUNNETT)

When a set of new treatments ($\overline{x}_i$) is compared with a control ($\overline{x}_0$), Dunnett’s two-tailed *t* test is usually used under the equal variances assumption.

For two-tailed tests,

$$v_{i,0} = |\overline{x}_i - \overline{x}_0| > d^{\epsilon}_{k,v} s_{dd} \sqrt{\frac{1}{n_0} + \frac{1}{n_i}}$$

where $d^{\epsilon}_{k,v}$ is the upper 100ε percentage point of the distribution of

$$T = \max_{1 \le i \le k} \{|T_i|\}$$

where $T_i = \frac{(\bar{x}_i - \bar{x}_0)}{s_{dd}\sqrt{\frac{1}{n_0}+\frac{1}{n_i}}}$ and $s_{dd}^2 = \frac{\Sigma_{i=0}^{k}\Sigma_{j=1}^{n_i}(x_{ij} - \bar{x}_{i.})^2}{\Sigma_{i=0}^{k}(n_i - 1)}$

Dunnett's One-Tailed t Test (DUNNETTL)

This Dunnett's one-tailed t test indicates whether the mean at any level is smaller than a reference category.

$$\bar{x}_i - \bar{x}_0 > dU_{k,v}^{\epsilon} s_{dd}\sqrt{\frac{1}{n_0} + \frac{1}{n_i}}$$

where $dU_{k,v}^{\epsilon}$ is the upper 100ε percentage point of the distribution of

$$T = \max_{1 \le i \le k} T_i$$

Confidence intervals are given only for the difference between the control group and the other groups.

Dunnett's One-Tailed t Test (DUNNETTR)

This Dunnett's one-tailed t test indicates whether the mean at any level is larger than a reference category.

$$\bar{x}_i - \bar{x}_0 < dL_{k,v}^{\epsilon} s_{dd}\sqrt{\frac{1}{n_0} + \frac{1}{n_i}}$$

where $dL_{k,v}^{\epsilon}$ is the upper 100ε percentage point of the distribution of

$$T = \max_{1 \le i \le k} \{T_i\}$$

Confidence intervals are given only for the difference between the control group and the other groups.

Ryan-Einot-Gabriel-Welsch (R-E-G-W) Multiple Stepdown Procedures

For the R-E-G-W F test and the R-E-G-W Q test, a new significant level, γ_r, based on the number of steps between means is introduced:

$$\gamma_r = \begin{cases} 1 - (1-\epsilon)^{r/k} & \text{if } r < k-1 \\ \epsilon & \text{if } r \ge k-1 \end{cases}$$

Note: For homogeneous subsets, the n_i and n_j are used for the R-E-G-W *F* test and the R-E-G-W *Q* test. To apply these methods, the procedures are same as in "Constructing Homogeneous Subsets", using the tests provided below.

Ryan-Einot-Gabriel-Welsch Based on the Studentized Range Test (QREGW)

The R-E-G-W *Q* test is based on

$$\max_{i,j \in R}\left\{(\bar{x}_i - \bar{x}_j)\sqrt{\min(n_i, n_j)}\right\}/s_{pp} \ge S_{\gamma_r,r,f}$$

Ryan-Einot-Gabriel-Welsch Procedure Based on an F Test (FREGW)

The R-E-G-W *F* test is based on

$$\frac{\left(\sum_{i\in R} n_i\overline{x}_i^2 - \left(\sum_{i\in R} n_i\overline{x}_i\right)^2 / \sum_{i\in R} n_i\right)}{(r-1)s_{pp}^2} \geq F_{\gamma_r, r-1, f}$$

where $r = j - i + 1$ and summations are over $R = \{i, \ldots, j\}$.

Unequal Sample Sizes and Unequal Variances

The tests in this section are based on assumptions that variances are unequal and sample sizes are unequal. An estimate of the degrees of freedom is used. The estimator is

$$v = \frac{\left(s_i^2/n_i + s_j^2/n_j\right)^2}{s_i^4/n_i^2 v_i + s_j^4/n_j^2 v_j}$$

Two means are significantly different if

$$|\overline{x}_i - \overline{x}_j| \geq Q_{i,j}^* R_{\epsilon,r,v}$$

where

$$Q_{i,j}^* = \sqrt{\frac{s_i^2}{n_i} + \frac{s_j^2}{n_j}}$$

and $R_{\epsilon,r,\gamma}$ depends on the specific test being used, as listed below.

For the Games-Howell, Tamhane’s T2, Dunnett’s T3, and Dunnett’s C tests, only pairwise confidence intervals are given.

Games-Howell Pairwise Comparison Test (GH)

$$R_{\epsilon,r,v} = S_{\epsilon,k,v}/\sqrt{2}$$

Tamhane’s T2 (T2)

$R_{\epsilon,r,v} = \sqrt{F_{\gamma,1,v}}=t_{\gamma,v}$ where $\gamma = 1 - (1-\epsilon)^{1/k^*}$

Dunnett’s T3 (T3)

$$R_{\epsilon,r,v} = M_{\epsilon,k^*,v}$$

Dunnett’s C (C)

$$R_{\epsilon,r,v} = \frac{\left(S_{\epsilon,k,n_i-1} s_i^2/n_i + S_{\epsilon,k,n_j-1} s_j^2/n_j\right)/\sqrt{2}}{s_i^2/n_i + s_j^2/n_j}$$

References

Cheng, P. H., and C. Y. K. Meng. 1992. A New Formula for Tail probabilities of DUNNETT's T with Unequal Sample Sizes. *ASA Proc. Stat. Comp.*, , 177–182.

Duncan, D. B. 1955. Multiple Range and Multiple F tests. *Biometrics*, 11, 1–42.

Duncan, D. B. 1975. t Tests and Intervals for Comparisons Suggested by the Data. *Biometrics*, 31, 339–360.

Dunnett, C. W. 1955. A Multiple Comparisons Procedure for Comparing Several Treatments with a Control. *Journal of the American Statistical Association*, 50, 1096–1121.

Dunnett, C. W. 1980. Pairwise Multiple Comparisons in Homogeneous Variance, Unequal Sample Size Case. *Journal of the American Statistical Association*, 75, 789–795.

Dunnett, C. W. 1980. Pairwise Multiple Comparisons in the Unequal Variance Case. *Journal of the American Statistical Association*, 75, 796–800.

Dunnett, C. W. 1989. Multivariate Normal Probability Integrals with Product Correlation Structure. *Applied Statistics*, 38, 564–571.

Einot, I., and K. R. Gabriel. 1975. A Study of the powers of Several Methods of Multiple Comparisons. *Journal of the American Statistical Association*, 70, 574–783.

Gabriel, K. R. 1978. A Simple method of Multiple Comparisons of Means. *Journal of the American Statistical Association*, 73, 724–729.

Games, P. A., and J. F. Howell. 1976. Pairwise Multiple Comparison Procedures with Unequal N's and/or Variances: A Monte Carlo Study. *Journal of Educational Statistics*, 1, 113–125.

Gebhardt, F. 1966. Approximation to the Critical Values for Duncan's Multiple Range Test. *Biometrics*, 22, 179–182.

Hochberg, Y. 1974. Some Generalizations of the T-method in Simultaneous Inference. *Journal of Multivariate Analysis*, 4, 224–234.

Hochberg, Y., and A. C. Tamhane. 1987. *Multiple Comparison Procedures*. New York: John Wiley & Sons, Inc. .

Hsu, J. C. 1989. *Multiple Comparison Procedures*. : American Statistical Association Short Course.

Miller, R. G. 1980. *Simultaneous Statistical Inference*, 2 ed. New York: Springer-Verlag.

Milliken, G., and D. Johnson. 1992. *Analysis of Messy Data: Volume 1. Designed Experiments*. New York: Chapman & Hall.

Ramsey, P. H. 1978. Power Differences Between Pairwise Multiple Comparisons. *Journal of the American Statistical Association*, 73, 479–485.

Ryan, T. A. 1959. Multiple Comparisons in Psychological Research. *Psychological Bulletin*, 56, 26–47.

Ryan, T. A. 1960. Significance Tests for Multiple Comparison of Proportions, Variances, and Other Statistics. *Psychological Bulletin*, 57, 318–328.

Scheffe, H. 1953. A method for judging all contrasts in the analysis of variance. *Biometrika*, 40, 87–104.

Scheffe, H. 1959. *The Analysis of Variance*. New York: John Wiley & Sons, Inc..

Searle, S. R. 1971. *Linear Models*. New York: John Wiley & Sons, Inc.

Sidak, Z. 1967. Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62, 626–633.

SAS Institute, Inc., . 1990. *SAS/STAT User's Guide, Version 6*, 4 ed. Cary, NC: SAS Institute Inc..

Tamhane, A. C. 1977. Multiple Comparisons in Model I One-Way ANOVA with Unequal Variances. *Communications in Statistics*, 6, 15–32.

Tamhane, A. C. 1979. A Comparison of Procedures for Multiple Comparisons of Means with Unequal Variances. *Journal of the American Statistical Association*, 74, 471–480.

Waller, R. A., and D. B. Duncan. 1969. A Bayes Rule for the Symmetric Multiple Comparison Problem. *Journal of the American Statistical Association*, 64, 1484–1499.

Waller, R. A., and D. B. Duncan. 1972. . *Journal of the American Statistical Association*, 67, 253–255.

Waller, R. A., and K. E. Kemp. 1975. Computations of Bayesian t-value for Multiple Comparison. *Journal of statistical computation and simulation*, 4, 169–172.

Welsch, R. E. 1977. Stepwise Multiple Comparison Procedures. *Journal of the American Statistical Association*, 72, 566–575.

Appendix

H

Sums of Squares

This appendix describes methods for computing sums of squares.

Notation

The notation used in this appendix is the same as that in the GLM Univariate and Multivariate chapter.

Type I Sum of Squares and Hypothesis Matrix

The Type I sum of squares is computed by fitting the model in steps according to the order of the effects specified in the design and recording the difference in error sum of squares (ESS) at each step.

By applying the SWEEP operator on the rows and columns of the augmented matrix $\mathbf{Z}'\mathbf{WZ}$, of dimension $(p+r)\times(p+r)$, the Type I sum of squares and its hypothesis matrix for each effect (except for the intercept effect, if any) is obtained.

Calculating the Sum of Squares

The following procedure is used to find the Type I sum of squares for effect F:

Let the order of effects specified in the design be F_0, F_1, F_2, ..., F_m. The columns of X are partitioned into X_0, X_1, X_2, ..., X_m, where $\mathbf{X}_0 = \mathbf{1}$ corresponds to the intercept effect F_0, and the columns in the submatrix X_j correspond to effect F_j, j=0,1,...,m.

Let F_j be the effect F of interest. Let $ESS_{j-1}(l)$ and $ESS_j(l)$ be the lth diagonal elements of the $r\times r$ lower diagonal submatrix of $\mathbf{Z}'\mathbf{WZ}$ after the SWEEP operator is applied to the columns associated with X_0, X_1, X_2, ..., X_j, . When the lth column of **Y** is used as the dependent variable, the Type I sum of squares for effect F_j is $\text{ESS}_{j-1}(l) - \text{ESS}_j(l), l = 1,\ldots,r$, where $ESS_{-1}(l)$ is defined as 0.

Constructing a Hypothesis Matrix

The hypothesis matrix **L** is constructed using the following steps:

1. Let $\mathbf{L}_0$ be the upper diagonal $p\times p$ submatrix of $\mathbf{Z}'\mathbf{WZ}$ after the SWEEP operator is applied to the columns associated with the effects preceding F. Set the columns and rows of $\mathbf{L}_0$, which are associated with the effects preceding F, to 0.

2. For the rows of $\mathbf{L}_0$ associated with the effects ordered after F_j, if any, set the corresponding rows of $\mathbf{L}_0$ to 0. Remove all of the 0 rows in the matrix $\mathbf{L}_0$. The row dimension of $\mathbf{L}_0$ is then less than p.

3. Use row operations on the rows of $\mathbf{L}_0$ to remove any linearly dependent rows. The set of all nonzero rows of $\mathbf{L}_0$ forms a Type I hypothesis matrix L.

Type II Sum of Squares and Hypothesis Matrix

A Type II sum of squares is the reduction in ESS due to adding an effect after all other terms have been added to the model except effects that contain the effect being tested.

For any two effects F and F', F is contained in F' if the following conditions are true:

- Both effects F and F' involve the same covariate, if any.
- F' consists of more factors than F.
- All factors in F also appear in F'.

Intercept Effect. The intercept effect μ is contained in all the pure factor effects. However, it is not contained in any effect involving a covariate. No other effect is contained in the intercept effect.

Calculating the Sum of Squares

To find the Type II (and also Type III and IV) sum of squares associated with any effect F, you must distinguish which effects in the model contain F and which do not. The columns of **X** can then be partitioned into three groups: $\mathbf{X}_1$, $\mathbf{X}_2$ and $\mathbf{X}_3$, where:

• $\mathbf{X}_1$ consists of columns of **X** that are associated with effects that do not contain F.

• $\mathbf{X}_2$ consists of columns that are associated with F.

• $\mathbf{X}_3$ consists of columns that are associated with effects that contain F.

The SWEEP operator applied on the augmented matrix $\mathbf{Z}'\mathbf{W}\mathbf{Z}$ is used to find the Type II sum of squares for each effect. The order of sweeping is determined by the "contained" relationship between the effect being tested and all other effects specified in the design.

Once the ordering is defined, the Type II sum of squares and its hypothesis matrix **L** can be obtained by a procedure similar to that used for the Type I sum of squares.

Constructing a Hypothesis Matrix

A hypothesis matrix **L** for the effect F has the form

$$\mathbf{L} = (\mathbf{0} \quad \mathbf{C}\mathbf{X}'_2\mathbf{W}^{\frac{1}{2}}\mathbf{M}_1\mathbf{W}^{\frac{1}{2}}\mathbf{X}_2 \quad \mathbf{C}\mathbf{X}'_2\mathbf{W}^{\frac{1}{2}}\mathbf{M}_1\mathbf{W}^{\frac{1}{2}}\mathbf{X}_3)$$

where

$$\mathbf{M}_1 = \mathbf{I} - \mathbf{W}^{\frac{1}{2}}\mathbf{X}_1\left(\mathbf{X}'_1\mathbf{W}\mathbf{X}_1\right)^*\mathbf{X}_1\mathbf{W}^{\frac{1}{2}}$$

$$\mathbf{C} = \left(\mathbf{X}'_2\mathbf{W}^{\frac{1}{2}}\mathbf{M}_1\mathbf{W}^{\frac{1}{2}}\mathbf{X}_2\right)^*$$

A* is a g2 generalized inverse of a symmetric matrix **A**.

Type III Sum of Squares and Hypothesis Matrix

The Type III sum of squares for an effect F can best be described as the sum of squares for F adjusted for effects that do not contain it, and orthogonal to effects (if any) that contain it.

Constructing a Hypothesis Matrix

A Type III hypothesis matrix **L** for an effect F is constructed using the following steps:

1. The design matrix **X** is reordered such that the columns can be grouped in three parts as described in the Type II approach. Compute $\mathbf{H} = \left(\mathbf{X}'\mathbf{W}\mathbf{X}\right)^{*}\mathbf{X}'\mathbf{W}\mathbf{X}$. Notice that the columns of **H** can also be partitioned into three parts: the columns corresponding to effects not containing F, the columns corresponding to the effect F, and the columns corresponding to the effects containing F (if any).
2. The columns of those effects not containing F (except F) are set to 0 by means of the row operation. That is:

   a) For each of those columns that is not already 0, fix any nonzero element in that column and call this nonzero element the pivot element.

   b) Divide the row that corresponds to the pivot element by the value of the pivot element itself.

   c) Use row operations to introduce zeros to all other nonzero elements (except the pivot element itself) in that column.

   d) Set the whole row containing the pivot element to 0. The column and the row corresponding to this pivot element should now be 0.

   e) Continue the process for the next column that is not 0 until all columns corresponding to those effects that do not contain F are 0.
3. For each column associated with effect F, find a nonzero element, use it as pivot, and perform the Gaussian elimination method as described in a, b, and c of step 2. After all such columns are processed, remove all of the 0 rows from the resulting matrix. If there is no column corresponding to effects containing F (which is the case when F contains all other effects), the matrix just constructed is the Type III hypothesis matrix for effect F. If there are columns corresponding to effects that contain F, continue with step 4.
4. The rows of the resulting matrix in step 3 can now be categorized into two groups. In one group, the columns corresponding to the effect F are all 0; call this group of rows G_0. In the other group, those columns are nonzero; call this group of rows G_1. Notice that the rows in G_0 form a generating basis for the effects that contain F. Transform the rows in G_1 such that they are orthogonal to any rows in G_0.

Calculating the Sum of Squares

Once a hypothesis matrix is constructed, the corresponding sum of squares can be calculated by $\left(\mathbf{L}\hat{\mathbf{B}}\right)'\left(\mathbf{L}\mathbf{G}\mathbf{L}'\right)^{*}\mathbf{L}\hat{\mathbf{B}}$.

Type IV Sum of Squares and Hypothesis Matrix

A hypothesis matrix **L** of a Type IV sum of squares for an effect F is constructed such that for each row of **L**, the columns corresponding to effect F are distributed equitably across the columns of effects containing F. Such a distribution is affected by the availability and the pattern of the nonmissing cells.

Constructing a Hypothesis Matrix

A Type IV hypothesis matrix L for effect F is constructed using the following steps:

1. Perform steps 1, 2, and 3 as described for the Type III sum of squares.
2. If there are no columns corresponding to the effects containing F, the resulting matrix is a Type IV hypothesis matrix for effect F. If there are columns corresponding to the effects containing F, the following step is needed.
3. First, notice that each column corresponding to effect F represents a level in F. Moreover, the values in these columns can be viewed as the coefficients of a contrast for comparing different levels in F. For each row, the values of the columns corresponding to the effects that contain F are based on the values in that contrast. The final hypothesis matrix **L** consists of rows with nonzero columns corresponding to effect A. For each row with nonzero columns corresponding to effect F:

   a) If the value of any column (or level) corresponding to effect F is 0, set to 0 all values of columns corresponding to effects containing F and involving that level of F.

   b) For columns (or levels) of F that have nonzero values, count the number of times that those levels occur in one or more common levels of the other effects. This count will be based on the availability of the nonmissing cells in the data. Then set each column corresponding to an effect that contains F and involves that level of F to the value of the column that corresponds to that level of F divided by the count.

   c) If any value of the column corresponding to an effect that contains F and involves a level (column) of F is undetermined, while the value for that level (column) of F is nonzero, set the value to 0 and claim that the hypothesis matrix created may not be unique.

Calculating the Sum of Squares

Once a hypothesis matrix is constructed, the corresponding sum of squares can be calculated by $\left(\mathbf{L}\hat{\mathbf{B}}\right)'\left(\mathbf{LGL}'\right)^{*}\mathbf{L}\hat{\mathbf{B}}$. The corresponding degrees of freedom for this test is the row rank of the hypothesis matrix.

Appendix

I

Distribution and Special Functions

The functions described in this appendix are used in more than one procedure. They are grouped into the following categories:

- **Continuous Distributions.** Beta, Cauchy, chi-square, exponential, F, gamma, Laplace, logistic, lognormal, normal, noncentral beta, noncentral chi-square, noncentral F, noncentral Student's t, Pareto, Student's t, uniform, and Weibull
- **Discrete Distributions.** Bernoulli, binomial, geometric, hypergeometric, negative binomial, and Poisson
- **Special Functions.** Gamma function, beta function, incomplete gamma function (ratio), incomplete beta function (ratio), and standard normal function

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table I-1
Notation

Notation	Description
$f(x)$	Density function of continuous random variable X or probability mass function of discrete random variable X
$F(x)$	Cumulative distribution function of continuous or discrete variable X
$F^{-1}(x)$	Inverse cumulative distribution function of X

Continuous Distributions

These are functions of a single scale variable.

Beta

The beta distribution takes values in the range $0<x<1$ and has two shape parameters, α and β. Both α and β must be positive, and they have the property that the mean of the distribution is α/(α+β).

Common uses. The beta distribution is used in Bayesian analyses as a conjugate to the binomial distribution.

Functions. The `CDF`, `IDF`, `PDF`, `NCDF`, `NPDF`, and `RV` functions are available.

The beta distribution has PDF, CDF, and IDF

$$f(x;\alpha,\beta) = \frac{1}{\mathrm{B}_{(\alpha,\beta)}} x^{\alpha-1}(1-x)^{\beta-1}$$

$$F(x;\alpha,\beta) = \mathrm{IB}(x;\alpha,\beta)$$

$F^{-1}(p;\alpha,\beta) = \mathrm{IB}^{-1}(p;\alpha,\beta)$

where $\mathrm{B}(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1}dx$ is the beta function and

$\mathrm{IB}(x;a,b) = \int_0^x \frac{1}{\mathrm{B}(a,b)} t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- When *α*=*β*=1, the beta(*α*,*β*) distribution is equivalent to the uniform(0,1) distribution.
- The beta(*α*,*β*) distribution is the distribution of *X*/(*X*+*Y*) where *X* and *Y* are variables that have chi-square distributions with degrees of freedom parameters 2*α* and 2*β*, respectively.

Random Number Generation

Special case (*a*=1 or *b*=1)

1. Generate *U* from Uniform(0,1).
2. If *a*=1, set $X = 1-(1-U)^{1/b}$.
3. If *b*=1, set $X = U^{1/a}$.
4. If both *a*=1 and *b*=1, set *X*=*U*.

Algorithm BN due to Ahrens and Dieter (1974) for *a* > 1 and *b* > 1

1. Set $e = a-1, f = b-1, c = e+f, g = c\ln(c), u = e/c,$ and s=0.5/√c.
2. Generate *Y* from N(0,1) and set $X = sY + u$.
3. If $X < 0$ or $X > 1$, go to step 2.
4. Generate *U* from Uniform(0,1).
5. If $\ln(U) \le \left(e\ln(X/e) + f\ln((1-X)/f) + g + 0.5Y^2\right)$, finish; otherwise go to step 2.

References. CDF: AS 63 (1973); ICDF: AS 64 (1973) and AS 109 (1977); RV: AS 134 (1979), (Ahrens and Dieter, 1974), and (Cheng, 1978). (See the Algorithm Index and References.)

Bivariate Normal

The bivariate normal distribution takes real values and has one correlation parameter, ρ, which must be between –1 and 1, inclusive.

Functions. The `CDF` and `PDF` functions are available and require two quantiles, *x1* and *x2*.

The bivariate normal distribution has PDF

$$f(x_1,x_2;\rho) = \frac{1}{2\pi(1-\rho^2)^{1/2}}\exp\left(\frac{-1}{2(1-\rho^2)}\left(x_1^2 - 2\rho x_1 x_2 + x_2^2\right)\right)$$

The CDF does not have a closed form and is computed by approximation.

Relationship to other distributions.

- Two variables with a bivariate normal(ρ) distribution with correlation ρ have marginal normal distributions with a mean of 0 and a standard deviation of 1.

Numerical algorithms for computing the CDF are described in the references.

References. AS 462 (1973) and AS 195. (See the Algorithm Index and References.)

Cauchy

The Cauchy distribution takes real values and has a location parameter, θ, and a scale parameter, ς; ς must be positive. The Cauchy distribution is symmetric about the location parameter, but has such slowly decaying tails that the distribution does not have a computable mean.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are

available. The Cauchy distribution has PDF, CDF, and IDF

$$f(x;\theta,\varsigma) = \frac{1}{\pi\varsigma}\left(1+\left(\frac{x-\theta}{\varsigma}\right)^2\right)^{-1}$$

$$F(x;\theta,\varsigma) = \frac{1}{2}+\frac{1}{\pi}\tan^{-1}\left(\frac{x-\theta}{\varsigma}\right)$$

$$F^{-1}(p;\theta,\varsigma) = \theta+\varsigma\tan\left(\pi(p-1/2)\right)$$

Relationship to other distributions.

- A “standardized” Cauchy variate, $(x-\theta)/\varsigma$, has a *t* distribution with 1 degree of freedom.

Random Number Generation

Inverse CDF algorithm

1. Generate *U* from Uniform(0,1).
2. Set $X = a + b\tan\left(\pi(U-1/2)\right)$

Chi-Square

The chi-square(ν) distribution takes values in the range x>=0 and has one degrees of freedom parameter, ν; it must be positive and has the property that the mean of the distribution is ν.

Functions. The `CDF`, `IDF`, `PDF`, `RV`, `NCDF`, `NPDF`, and `SIG` functions are available.

The chi-square distribution has PDF, CDF, and IDF

$$f(x;\nu) = \frac{1}{2^{\nu/2}\Gamma(\nu/2)}x^{(\nu/2)-1}e^{-x/2}$$

$$F(x;\nu) = \mathrm{IG}\left(\frac{x}{2};\frac{\nu}{2}\right)$$

$F^{-1}(p;\nu) = 2IG^{-1}\left(p; \frac{\nu}{2}\right)$

where $\Gamma(a) = \int_0^\infty x^{a-1}e^{-x}dx$ is the gamma function and $IG(x;a) = \int_0^x \frac{1}{\Gamma(a)}t^{a-1}e^{-t}dt$ is the incomplete gamma function.

Relationship to other distributions.

- The chi-square(*v*) distribution is the distribution of the sum of squares of ν independent normal(0,1) random variates.
- The chi-square(*v*) distribution is equivalent to the gamma(*v*/2, 1/2) distribution.

Random Number Generation

Generate *X* from the Gamma(*a*/2, 1/2) distribution.

References. CDF: CACM 299 (1967); ICDF: AS 91 (1975), AS R85(1991), and CACM 451 (1973). (See the Algorithm Index and References.)

Exponential

The exponential distribution takes values in the range x>=0 and has one scale parameter, β, which must be greater than 0 and has the property that the mean of the distribution is 1/β.

Common uses. In life testing, the scale parameter a represents the rate of decay.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The exponential distribution has PDF, CDF, and IDF

$f(x;\beta) = \beta e^{-\beta x}$

$F(x;\beta) = 1 - e^{-\beta x}$

$F^{-1}(p;\beta) = -\frac{1}{\beta}\ln(1-p)$

Relationship to other distributions.

- The exponential(*β*) distribution is equivalent to the gamma(1,*β*) distribution.

Random Number Generation

Inverse CDF algorithm

Generate *U* from Uniform(0,1); $X = -\ln(1-U)/a$.

F

The F distribution takes values in the range x>=0 and has two degrees of freedom parameters, ν1 and ν2, which are the “numerator” and “denominator” degrees of freedom, respectively. Both ν1 and ν2 must be positive.

Common uses. The *F* distribution is commonly used to test hypotheses under the Gaussian assumption.

Functions. The `CDF`, `IDF`, `IDF`, `RV`, `NCDF`, `NPDF`, and `SIG` functions are available.

The F distribution has PDF, CDF, and IDF

$$f(x;\nu_1,\nu_2)=\frac{1}{\mathrm{B}(\nu_1/2,\nu_2/2)}\left(\frac{\nu_1}{\nu_2}\right)^{\nu_1/2}x^{(\nu_1/2)-1}\left(1+\frac{\nu_1}{\nu_2}x\right)^{-(\nu_1+\nu_2)/2}$$

$$F(x;\nu_1,\nu_2)=\mathrm{IB}\left(\frac{\nu_1 x}{\nu_2+\nu_1 x};\frac{\nu_1}{2},\frac{\nu_2}{2}\right)$$

$$F^{-1}(p;\nu_1,\nu_2)=\frac{\nu_2}{\nu_1}\left(\frac{IB^{-1}(p;\nu_1/2,\ \nu_2/2)}{1-IB^{-1}(p;\nu_1/2,\ \nu_2/2)}\right)$$

where $\mathrm{B}(a,b)=\int_0^1 x^{a-1}(1-x)^{b-1}dx$ is the beta function and

$\mathrm{IB}(x;a,b)=\int_0^x \frac{1}{\mathrm{B}(a,b)}t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- The F(*v1*,*v2*) distribution is the distribution of (*X*/ν1)/(*Y*/ν2), where *X* and *Y* are independent chi-square random variates with *v1* and *v2* degrees of freedom, respectively.

Random Number Generation

Using the chi-square distribution

1. Generate *Y* and *Z* independently from chi-square(*a*) and chi-square(*b*), respectively.
2. Set *X*=(*Y*/*a*) / (*Z*/*b*).

References. CDF: CACM 332 (1968). ICDF: use inverse of incomplete beta function. (See the Algorithm Index and References.)

Gamma

The gamma distribution takes values in the range x>=0 and has one shape parameter, α, and one scale parameter, β. Both parameters must be positive and have the property that the mean of the distribution is α/β.

Common uses. The gamma distribution is commonly used in queuing theory, inventory control, and precipitation processes.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The gamma distribution has PDF, CDF, and IDF

$$f(x;\alpha,\beta)=\frac{\beta^\alpha}{\Gamma(\alpha)}x^{\alpha-1}e^{-\beta x}$$

$$F(x;\alpha,\beta)=\mathrm{IG}(\beta x;\alpha)$$

$F^{-1}(p;\alpha,\beta) = \frac{1}{\beta}\mathrm{IG}^{-1}(p;\alpha)$

where $\Gamma(a) = \int_0^\infty x^{a-1}e^{-x}dx$ is the gamma function and $\mathrm{IG}(x;a) = \int_0^x \frac{1}{\Gamma(a)}t^{a-1}e^{-t}dt$ is the incomplete gamma function.

Relationship to other distributions.

- When *α*=1, the gamma(*α*,*β*) distribution reduces to the exponential(*β*) distribution.
- When *β*=1/2, the gamma(*α*,*β*) distribution reduces to the chi-square(2*α*) distribution.
- When *α* is an integer, the gamma distribution is also known as the Erlang distribution.

Random Number Generation

Special case

If *a* = 1 and *b* > 0, generate *X* from an exponential distribution with parameter *b*.

Algorithm GS due to Ahrens and Dieter (1974) for 0<*a*<1 and *b*=1

1. Generate *U* from Uniform(0,1). Set *c*=(*e*+*a*)/*e*, where *e*=exp(1).
2. Set *P*=*cU*. If *P*>1, go to step 4.
3. (*P*≤1) Set $X = P^{1/a}$. Generate *V* from Uniform(0,1). If *V*>exp(−*x*), go to step 1; otherwise finish.
4. (*P*>1) Set *X*=−ln((*c*−*P*)/*a*). If *X*<0, go to step 1; otherwise go to step 5.
5. Generate *V* from Uniform(0,1). If $V > X^{a-1}$, go to step 1; otherwise finish.

Algorithm due to Fishman (1976) for *a*>1 and *b*=1

1. Generate *Y* from Exponential (1).
2. Generate *U* from Uniform(0,1).
3. If ln*U*≤(*a*−1)(1−*Y*+ln*Y*), *X*=*aY*; otherwise go to Step 1.

References. CDF: AS 32 (1970) and AS 239 (1988); ICDF: Use the relationship between gamma and chi-square distributions. RV: (Ahrens et al., 1974), (Fishman, 1976), and (Tadikamalla, 1978). (See the Algorithm Index and References.)

Half-normal

The half-normal distribution takes values in the range x>=μ and has one location parameter, μ, and one scale parameter, σ. Parameter σ must be positive.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The half-normal distribution has PDF, CDF, and IDF

$f(x;\mu,\sigma) = \phi\left(\frac{x-\mu}{\sigma}\right)/\sigma$

$$F\left(x;\mu,\sigma\right)=2\Phi\left(\tfrac{x-\mu}{\sigma}\right)-1$$

$$F^{-1}\left(p;\mu,\sigma\right)=\mu+\sigma\Phi^{-1}\left(\tfrac{1+p}{2}\right)$$

Relationship to other distributions.

- If X has a normal(μ,σ) distribution, then $|X-\mu|$ has a half-normal(μ,σ) distribution.

Random Number Generation

1. Generate X from a normal(a,b) distribution.
2. Then $|X-a|$ has a half normal distribution.

Inverse Gaussian

The inverse Gaussian, or Wald, distribution takes values in the range x>0 and has two parameters, μ and λ, both of which must be positive. The distribution has mean μ.

Common uses. The inverse Gaussian distribution is commonly used to test hypotheses for model parameter estimates.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The inverse Gaussian distribution has PDF and CDF

$$f\left(x;\mu,\lambda\right)=\left(\tfrac{\lambda}{2\pi x^3}\right)^{1/2}\exp\left(-\tfrac{\lambda(x-\mu)^2}{2\mu^2 x}\right)$$

$$F\left(x;\mu,\lambda\right)=\Phi\left(\sqrt{\tfrac{\lambda}{x}}\left(-1+\tfrac{x}{\mu}\right)\right)+e^{(2\lambda/\mu)}\Phi\left(-\sqrt{\tfrac{\lambda}{x}}\left(1+\tfrac{x}{\mu}\right)\right)$$

The IDF is computed by approximation.

Inverse CDF Approximation

For the upper tail, an inverse Gaussian variable X can be approximated by

$$X=\alpha\chi_v^2+\beta$$

where

$$\alpha=3a^2/4b$$

$$\beta=a/3$$

$$v=8b/9a$$

For the lower tail, one can use the approximation

$$X=\left(\alpha\chi_v^2+\beta\right)^{-1}$$

where

$$\alpha = (3b + 8a) / [4b (b + 2a)]$$

$$\beta = (b + 3a) / [a (3b + 8a)]$$

$$v = 8(b + 2a)^3 / \left[a(8a + 3b)^2\right]$$

Random Number Generation

1. Generate a standard normal variate *Z*.
2. Let $w = aZ^2$
3. Let $x = a + \frac{a}{2b}\left(w - \sqrt{w(4b + w)}\right)$
4. Let $p = a / (a + x)$
5. Then the inverse Gaussian variate will take value *x* with probability $1 - p$ and value a^2/x with probability *p*.

References.(Mudholkar and Natarajan, 1999) and (Michael, Schucany, and Haas, 1976). (See the Algorithm Index and References.)

Laplace

The Laplace distribution takes real values and has one location parameter, μ, and one scale parameter, β. Parameter β must be positive. The distribution is symmetric about μ and has exponentially decaying tails.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The Laplace distribution has PDF, CDF, and IDF

$$f(x; \mu, \beta) = \frac{1}{2b} e^{-|x-\mu|/\beta}$$

$$F(x; \mu, \beta) = \begin{cases} \frac{1}{2} e^{(x-\mu)/\beta} & x \le \mu \\ 1 - \frac{1}{2} e^{(\mu-x)/\beta} & x > \mu \end{cases}$$

$$F^{-1}(p; \mu, \beta) = \begin{cases} \mu + \beta \ln(2p) & 0 \le p \le \frac{1}{2} \\ \mu - \beta \ln(2(1-p)) & \frac{1}{2} < p \le 1 \end{cases}$$

Random Number Generation

Inverse CDF algorithm

1. Generate *Y* and *U* independently from Exponential(1/ *b*) and Uniform(0,1), respectively.
2. If *U*≥0.5, set *X*=*a*+*Y*; otherwise set *X*=*a*−*Y*.

Logistic

The logistic distribution takes real values and has one location parameter, μ, and one scale parameter, ς. Parameter ς must be positive. The distribution is symmetric about μ and has longer tails than the normal distribution.

Common uses. The logistic distribution is used to model growth curves.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The logistic distribution has PDF, CDF, and IDF

$$f(x;\mu,\varsigma)=\tfrac{1}{\varsigma}e^{-(x-\mu)/\varsigma}\left(1+e^{-(x-\mu)/\varsigma}\right)^{-2}$$

$$F(x;\mu,\varsigma)=\tfrac{1}{1+e^{-(x-\mu)/\varsigma}}$$

$$F^{-1}(p;\mu,\varsigma)=\mu+\varsigma\ln\left(\tfrac{p}{1-p}\right)$$

Random Number Generation

Inverse CDF algorithm

1. Generate *U* from Uniform(0,1).
2. Set $X=a+b\ln\left(U/(1-U)\right)$.

Lognormal

The lognormal distribution takes values in the range x>=0 and has two parameters, η and σ, both of which must be positive.

Common uses. Lognormal is used in the distribution of particle sizes in aggregates, flood flows, concentrations of air contaminants, and failure time.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The lognormal distribution has PDF, CDF, and IDF

$$f(x;\eta,\sigma)=\tfrac{1}{x\sigma\sqrt{2\pi}}e^{-(\ln(x/\eta))^2/\left(2\sigma^2\right)}$$

$$F(x;\eta,\sigma)=\Phi\left(\tfrac{1}{\sigma}\ln\left(\tfrac{x}{\eta}\right)\right)$$

$$F^{-1}(p;\eta,\sigma)=\eta e^{\sigma\Phi^{-1}(p)}$$

Relationship to other distributions.

- If *X* has a lognormal(η,σ) distribution, then ln(*X*) has a normal(ln(η),σ) distribution.

Random Number Generation

Inverse CDF algorithm

1. Generate Z from N(0,1).
2. Set $X = a\exp(bZ)$.

Noncentral Beta

The noncentral beta distribution is a generalization of the beta distribution that takes values in the range 0<*x*<1 and has an extra noncentrality parameter, λ, which must be greater than or equal to 0.

Functions.

The noncentral beta distribution has PDF, CDF, and IDF

$$f(x;\alpha,\beta,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\frac{x^{\alpha+j-1}(1-x)^{\beta-1}}{\mathrm{B}(\alpha+j;\beta)}$$

$$F(x;\alpha,\beta,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\mathrm{IB}(x;\alpha=j,\beta)$$

where $\mathrm{B}(a,b)=\int_0^1 x^{a-1}(1-x)^{b-1}dx$ is the beta function and

$\mathrm{IB}(x;a,b)=\int_0^x \frac{1}{\mathrm{B}(a,b)}t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- When λ equals 0, this distribution reduces to the beta distribution.
- The noncentral beta(α,β,λ) distribution is the distribution of $X/(X+Y)$ where X is a variable that has a noncentral chi-square($2\alpha,\lambda$) distribution, and Y is a variable that has a central chi-square(2β) distribution.

References. CDF: (Abramowitz and Stegun, 1970) Chapter 26, AS 226 (1987), and AS R84 (1990). (See the Algorithm Index and References.)

Noncentral Chi-Square

The noncentral chi-square distribution is a generalization of the chi-square distribution that takes values in the range x>=0 and has an extra noncentrality parameter, λ, which must be greater than or equal to 0.

Functions.

The noncentral chi-square distribution has PDF and CDF

$$f(x;\nu,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\frac{x^{\nu/2+j-1}e^{-x/2}}{2^{\nu/2+j}\Gamma(\nu/2+j)}$$

$$F(x;\nu,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\mathrm{IG}\left(\frac{x}{2};\frac{\nu}{2}+j\right)$$

where $\Gamma(a)=\int_0^{\infty}x^{a-1}e^{-x}dx$ is the gamma function and $\mathrm{IG}(x;a)=\int_0^{x}\frac{1}{\Gamma(a)}t^{a-1}e^{-t}dt$ is the incomplete gamma function.

Relationship to other distributions.

- When λ equals 0, this distribution reduces to the chi-square distribution.
- The noncentral chi-square(ν,λ) distribution is the distribution of the sum of squares of ν independent normal(μ_i,1) random variates. Then $\lambda=\Sigma\mu_i^2$.

References. CDF: (Abramowitz et al., 1970) Chapter 26, AS 170 (1981), AS 231 (1987). Density: AS 275 (1992). (See the Algorithm Index and References.)

Noncentral F

The noncentral F distribution is a generalization of the F distribution that takes values in the range x>=0 and has an extra noncentrality parameter, λ, which must be greater than or equal to 0.

Functions.

The noncentral *F* distribution has PDF and CDF

$$f(x;\nu_1,b,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\frac{(\nu_1/b)^{\nu_1/2+j}}{\mathrm{B}(\nu_1/2+j,b/2)}x^{\nu_1/2+j-1}\left(1+\frac{\nu_1}{b}x\right)^{-((\nu_1+b)/2+j)}$$

$$F(x;\nu_1,b,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\frac{\lambda}{2}\right)^{j}e^{-\lambda/2}\mathrm{IB}\left(\frac{\nu_1 x}{b+\nu_1 x};\frac{\nu_1}{2}+j,\frac{b}{2}\right)$$

where $\mathrm{B}(a,b)=\int_0^{1}x^{a-1}(1-x)^{b-1}dx$ is the beta function and $\mathrm{IB}(x;a,b)=\int_0^{x}\frac{1}{\mathrm{B}(a,b)}t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- When λ equals 0, this distribution reduces to the *F* distribution.
- The noncentral *F* distribution is the distribution of (*X*/v1)/(*Y*/v2), where *X* and *Y* are independent variates with noncentral chi-square(*v1*, λ) and central chi-square(*v2*) distributions, respectively.

References. CDF: (Abramowitz et al., 1970) Chapter 26. (See the Algorithm Index and References.)

Noncentral Student's t

The noncentral t distribution is a generalization of the t distribution that takes real values and has an extra noncentrality parameter, λ, which must be greater than or equal to 0. When λ equals 0, this distribution reduces to the t distribution.

Functions.

The noncentral t distribution has PDF and CDF

$$f(x;\nu,\lambda)=\sum_{j=0}^{\infty}\frac{1}{j!}\left(\lambda\sqrt{2}\right)^{j}e^{-\lambda^2/2}\frac{\Gamma((\nu+j+1)/2)}{\Gamma(\nu/2)\Gamma(1/2)}\frac{x^j}{\nu^{(j+1)/2}}\left(1+\frac{x^2}{\nu}\right)^{-(\nu+j+1)/2}$$

$$F(x;\nu,\lambda)=\begin{cases}\frac{1}{2}\sum_{j=0}^{\infty}\frac{1}{j!}\left(-\lambda\sqrt{2}\right)^{j}e^{-\lambda^2/2}\frac{\Gamma((j+1)/2)}{\Gamma(1/2)}\mathrm{IB}\left(\frac{\nu}{\nu+x^2};\frac{\nu}{2},\frac{j+1}{2}\right) & x\le 0\\ 1-\frac{1}{2}\sum_{j=0}^{\infty}\frac{1}{j!}\left(\lambda\sqrt{2}\right)^{j}e^{-\lambda^2/2}\frac{\Gamma((j+1)/2)}{\Gamma(1/2)}\mathrm{IB}\left(\frac{\nu}{\nu+x^2};\frac{a}{2},\frac{j+1}{2}\right) & x>0\end{cases}$$

where $\mathrm{B}(a,b)=\int_0^1 x^{a-1}(1-x)^{b-1}dx$ is the beta function and

$\mathrm{IB}(x;a,b)=\int_0^x \frac{1}{\mathrm{B}(a,b)}t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- The noncentral $t(\nu,\lambda)$ distribution is the distribution of X/Y, where X is a normal(λ,1) variate and Y is a central chi-square(ν) variate divided by ν.

Special case

$$F(0)=1-\Phi(c)$$

References. CDF: (Abramowitz et al., 1970) Chapter 26, AS 5 (1968), AS 76 (1974), and AS 243 (1989). (See the Algorithm Index and References.)

Normal

The normal, or Gaussian, distribution takes real values and has one location parameter, μ, and one scale parameter, σ. Parameter σ must be positive. The distribution has mean μ and standard deviation σ.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The normal distribution has PDF, CDF, and IDF

$$f(x;\mu,\sigma)=\frac{1}{\sigma\sqrt{2\pi}}e^{-(x-\mu)^2/\left(2\sigma^2\right)}$$

$$F(x;\mu,\sigma)=\Phi\left(\frac{x-\mu}{\sigma}\right)$$

$$F^{-1}(p;\mu,\sigma)=\mu+\sigma\Phi^{-1}(p)$$

Relationship to other distributions.

- If X has a normal(μ,σ) distribution, then exp(X) has a normal(exp(μ),σ) distribution.

For Φ and Φ^{-1}, see “Standard Normal”

Random Number Generation

Kinderman and Ramage (1976) method

1. Generate as $X=a+bz$, where z is an $N(0,1)$ random number.

References. CDF: AS 66 (1973); ICDF: AS 111 (1977) and AS 241 (1988); RV: CACM 488 (1974) and (Kinderman and Ramage, 1976). (See the Algorithm Index and References.)

Pareto

The Pareto distribution takes values in the range xmin<x and has a threshold parameter, xmin, and a shape parameter, α. Both parameters must be positive.

Common uses. Pareto is commonly used in economics as a model for a density function with a slowly decaying tail.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The Pareto distribution has PDF, CDF, and IDF

$$f(x;x_{\text{min}},\alpha)=\frac{\alpha}{x_{\text{min}}}\left(\frac{x_{\text{min}}}{x}\right)^{\alpha+1}$$

$$F(x;x_{\text{min}},\alpha)=1-\left(\frac{x_{\text{min}}}{x}\right)^{\alpha}$$

$$F^{-1}(p;x_{\text{min}},\alpha)=x_{\text{min}}(1-p)^{-1/\alpha}$$

Random Number Generation

Inverse CDF

1. Generate U from Uniform(0,1).
2. Set $X=a(1-U)^{-1/b}$.

Studentized Maximum Modulus

The Studentized maximum modulus distribution takes values in the range x>0 and has a number of comparisons parameter, k*, and degrees of freedom parameter, ν, both of which must be greater than or equal to 1.

Common uses. The Studentized maximum modulus is commonly used in post hoc multiple comparisons for GLM and ANOVA.

Functions. The `CDF` and `IDF` functions are available, and are computed by approximation.

The Studentized maximum modulus distribution has CDF

$$F(x) = \int_0^\infty [2\Phi(xu) - 1]^{k*} \mathrm{dg}_\nu(u)$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the PDF and CDF of the standard normal distribution and

$$\mathrm{dg}_\nu(u) = \frac{\nu^{\nu/2}}{\Gamma\left(\frac{\nu}{2}\right)2} u^{\nu-1} \exp\left(-\nu u^2/2\right) \mathrm{d}u$$

The IDF does not have a closed form. The CDF can be computed by using numerical integration. The inverse CDF can be found by solving F(*x*) = *p* numerically for given *p*.

Studentized Range

The Studentized range distribution takes values in the range x>0 and has a number of samples parameter, k, and degrees of freedom parameter, ν, both of which must be greater than or equal to 1.

Common uses. The Studentized range is commonly used in post hoc multiple comparisons for GLM and ANOVA.

Functions. The `CDF` and `IDF` functions are available, and are computed by approximation.

The Studentized range distribution has CDF

$$F(x) = \int_0^\infty \int_{-\infty}^\infty a\phi(t)[\Phi(t) - \Phi(t - xu)]^{k-1} \mathrm{d}t\, \mathrm{dg}_\nu(u)$$

where $\phi(\cdot)$ and $\Phi(\cdot)$ are the PDF and CDF of the standard normal distribution and

$$\mathrm{dg}_\nu(u) = \frac{\nu^{\nu/2}}{\Gamma\left(\frac{\nu}{2}\right)2} u^{\nu-1} \exp\left(-\nu u^2/2\right) \mathrm{d}u$$

The IDF does not have a closed form. Both the CDF and IDF have to be computed numerically (see the following references).

References. AS 190, plus correction and remark. (See the Algorithm Index and References.)

Student's t

The Student t distribution takes real values and has one degrees of freedom parameter, ν, which must be positive. The Student t distribution is symmetric about 0.

Common uses. The major uses of the Student *t* distribution are to test hypotheses and construct confidence intervals for means of data.

Functions. The `CDF`, `IDF`, `PDF`, `RV`, `NCDF`, and `NPDF` functions are available.

The *t* distribution has PDF, CDF, and IDF

$$f(x;\nu) = \frac{1}{\sqrt{\nu}\mathrm{B}(\nu/2,1/2)}\left(1+\frac{x^2}{\nu}\right)^{-(\nu+1)/2}$$

$$F(x;\nu) = \begin{cases} \frac{1}{2}\mathrm{IB}\left(\frac{\nu}{\nu+x^2};\frac{\nu}{2},\frac{1}{2}\right) & \mathrm{x} \le 0 \\ 1\text{-}\frac{1}{2}\mathrm{IB}\left(\frac{\nu}{\nu+x^2};\frac{\nu}{2},\frac{1}{2}\right) & x > 0 \end{cases}$$

$$F^{-1}(p;\nu) = \begin{cases} -\sqrt{\nu\left(1/\left(\mathrm{IB}^{-1}\left(2p;\frac{\nu}{2},\frac{1}{2}\right)\right)-1\right)} & 0 \le p \le \frac{1}{2} \\ \sqrt{\nu\left(1/\left(\mathrm{IB}^{-1}\left(2(1-p);\frac{\nu}{2},\frac{1}{2}\right)\right)-1\right)} & \frac{1}{2} < p \le 1 \end{cases}$$

where $\mathrm{B}(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1}dx$ is the beta function and
$\mathrm{IB}(x;a,b) = \int_0^x \frac{1}{\mathrm{B}(a,b)} t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- The *t*(*v*) distribution is the distribution of *X*/*Y*, where *X* is a normal(0,1) variate and *Y* is a chi-square(*v*) variate divided by *v*.
- The square of a *t*(*v*) variate has an *F*(1,*v*) distribution.
- The *t*(*v*) distribution approaches the normal(0,1) distribution as *v* approaches infinity.

Random Number Generation

Special case

If *a*=1, generate *X* from a Cauchy (0, 1) distribution.

Using the normal and the chi-square distributions

1. Generate *Z* from N(0,1) and *V* from Chi-square(*a*) independently.
2. Set $X = Z/\sqrt{V/a}$.

References. CDF: AS 3 (1968), AS 27 (1970), and CACM 395 (1970); ICDF: CACM 396 (1970).
(See the Algorithm Index and References.)

Uniform

The uniform distribution takes values in the range a<x<b and has a minimum value parameter, a, and a maximum value parameter, b.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The uniform distribution has PDF, CDF, and IDF

$$f(x;a,b) = \frac{1}{b-a}$$

$F(x;a,b) = \frac{x-a}{b-a}$

$F^{-1}(p;a,b) = a + (b-a)p$

Random Number Generation

Inverse CDF algorithm

1. Generate *U* from Uniform(0,1).
2. Set $X = a + (b-a)U$.

References. Uniform of (0,1) is generated by the method in (Schrage, 1979).

Weibull

The Weibull distribution takes values in the range x>=0 and has one scale parameter, β, and one shape parameter, α, both of which must be positive.

Common uses. The Weibull distribution is commonly used in survival analysis.

Functions. The `CDF`, `IDF`, `PDF`, and `RV` functions are available.

The Weibull distribution has PDF, CDF, and IDF

$f(x;\beta,\alpha) = \frac{\alpha}{\beta}\left(\frac{x}{\beta}\right)^{\alpha-1} e^{-(x/\beta)^{\alpha}}$

$F(x;\beta,\alpha) = 1 - e^{-(x/\beta)^{\alpha}}$

$F^{-1}(p;\beta,\alpha) = \beta(-\ln(1-p))^{1/\alpha}$

Relationship to other distributions.

- A Weibull(β,1) distribution is equivalent to an exponential(β) distribution.

Random Number Generation

Inverse CDF algorithm

1. Generate *U* from Uniform(0,1).
2. Set $X = a(-\ln(1-U))^{1/b}$

Discrete Distributions

These are functions of a single variable that takes integer values.

Bernoulli

The Bernoulli distribution takes values 0 or 1 and has one success probability parameter, θ, which must be between 0 and 1, inclusive.

Functions. The `CDF`, `PDF`, and `RV` functions are available.

The Bernoulli distribution has PDF and CDF

$$f(x;\theta) = \theta^x (1-\theta)^{1-x}$$

$$F(x;\theta) = \begin{cases} 1-\theta & x=0 \\ 1 & x=1 \end{cases}$$

Relationship to other distributions.

- The Bernoulli distribution is a special case of the binomial distribution and is used in simple success-failure experiments.

Random Number Generation

Special case

If *a*=0, *X*=0. If *a*=1, *X*=1.

Direct algorithm for 0<*a*<1

1. Generate *U* from Uniform(0,1).
2. Set *X*=1 if *U*≤*a* (a success) and *X*=0 if *U*>*a* (a failure).

Binomial

The binomial distribution takes integer values 0<=x<=n, representing the number of successes in n trials, and has one number of trials parameter, n, and one success probability parameter, θ. Parameter n must be a positive integer and parameter θ must be between 0 and 1, inclusive.

Common uses. The binomial distribution is used in independently replicated success-failure experiments.

Functions. The `CDF`, `PDF`, and `RV` functions are available.

The binomial distribution has PDF and CDF

$$f(x;n,\theta) = \binom{n}{x} \theta^x (1-\theta)^{n-x}$$

$$F(x;n,\theta) = \begin{cases} 1 - \mathrm{IB}(\theta; x+1, n-x) & x = 0,1,\ldots,n-1 \\ 1 & x = n \end{cases}$$

where $\mathrm{IB}(x;a,b) = \int_0^x \frac{1}{\mathrm{B}(a,b)} t^{a-1}(1-t)^{b-1} dt$ is the incomplete beta function.

Random Number Generation

Special case

If b = 0, X = 0. If b = 1, X = a.

Algorithm BB due to Ahrens and Dieter (1974) for 0 < b < 1

1. Set $c = a, d = b, k = 0, y = 0,$ and $h = 1$.
2. If c<40, generate J from Binomial(c, d) using algorithm BU. X=k+J.
3. If c is odd, go to step 4. If c is even, set c=c−1 and generate U from Uniform(0,1). If U≤d, set k=k+1.
4. Set s=(c+1)/2 and generate S from Beta(s, s). Set G=hs and Z=y+G.
5. If Z≤b, set $y = Z, h = h - d, d = (b - Z)/h,$ and $k = k + s$; otherwise set $h = G$ and $d = (b - y)/h$.
6. Set $c = s - 1$ and go to step 2.

Computation time for algorithm BB is O(log a).

References. RV: (Ahrens et al., 1974).

Geometric

The geometric distribution takes integer values x>=1, representing the number of trials needed (including the last trial) before a success is observed, and has one success probability parameter, θ, which must be between 0 and 1, inclusive.

Functions. The `CDF`, `PDF`, and `RV` functions are available.

The geometric distribution has PDF and CDF

$$f(x;\theta) = \theta(1-\theta)^{x-1}$$

$$F(x;\theta) = 1-(1-\theta)^{x}$$

Relationship to other distributions.

- The geometric(θ) distribution is equivalent to the negative binomial (1,θ) distribution.

Random Number Generation

Special case

If a=1, X=1.

Direct algorithm for 0 < a < 1

1. Set X=1.
2. Generate U from Uniform(0,1).

3. If *U*>*a*, set *X*=*X*+1 and go to step 2; otherwise finish.

Hypergeometric

The hypergeometric distribution takes integer values in the range max(0, Np+n−N)<=x<=min(Np,n), and has three parameters, N, n, and Np, where N is the total number of objects in an urn model, n is the number of objects randomly drawn without replacement from the urn, Np is the number of objects with a given characteristic, and x is the number of objects with the given characteristic observed out of the withdrawn objects. All three parameters are positive integers, and both n and Np must be less than or equal to N.

Functions. The `CDF`, `PDF`, and `RV` functions are available.

The hypergeometric distribution has PDF and CDF

$$f\left(x;N,n,Np\right)=\frac{\binom{Np}{x}\binom{N-Np}{n-x}}{\binom{N}{n}}$$

$$F\left(x;N,n,Np\right)=\sum_{k=\max(0,n+Np-N)}^{x}\text{Prob}(X=k)$$

Random Number Generation

Special case

If *b*=*a*, *X*=*c*. If *c*=*a*, *X*=*b*.

Direct algorithm

```
1. (Initialization) X=0, g=c, h=b, t=a.
2. Do the following loop exactly b times:
   Begin Loop
   i. Generate U from Uniform(0,1).
   ii. If $U \le (g/t)$, set $X = X + 1, g = g - 1,$ else $h = h - 1$.
   iii. If g=0, go to step 3.
   iv. If h=0, set $X = X + b - i$, where i (from 1 to b) is the loop index. Go to step 3.
   v. Set t=t−1.
   End Loop
3. Finish.
```

References. CDF: AS 152 (1989), AS R77 (1989), and AS R86 (1991). (See the “Algorithm Index” and “References”)

Negative Binomial

The negative binomial distribution takes integer values in the range x>=r, where x is the number of trials needed (including the last trial) before r successes are observed, and has one threshold parameter, r, and one success probability parameter, θ. Parameter r must be a positive integer and parameter θ must be greater than 0 and less than or equal to 1.

Functions. The `CDF`, `PDF`, and `RV` functions are available.

The negative binomial distribution has PDF and CDF

$$f(x;r,\theta) = \binom{x\ -1}{r-1}\theta^r(1-\theta)^{x-r}$$

$$F(x;r,\theta) = \mathrm{IB}(\theta;r,x-r+1)$$

where $\mathrm{IB}(x;a,b) = \int_0^x \frac{1}{\mathrm{B}(a,b)} t^{a-1}(1-t)^{b-1}dt$ is the incomplete beta function.

Relationship to other distributions.

- The negative binomial(1,θ) distribution is equivalent to the geometric(θ) distribution.

Random Number Generation

Special case

If *b*=1, *X*=*a*.

Direct algorithm

1. Generate *G* from Gamma(*a*, *b*/(1−*b*)).
2. If *G*=0, go to step 1. Otherwise generate *P* from Poisson (*G*).
3. Compute *X*=*P*+*a*.

Poisson

The Poisson distribution takes integer values in the range x>=0 and has one rate or mean parameter, λ. Parameter λ must be positive.

Common uses. The Poisson distribution is used in modeling the distribution of counts, such as traffic counts and insect counts.

Functions. The `CDF`, `PDF`, and `RV` functions are

available. The Poisson distribution has PDF and CDF

$$f(x;\lambda) = \tfrac{\lambda^x}{x!}e^{-\lambda}$$

$$F(x;\lambda) = 1 - \mathrm{IG}(\lambda;x+1)$$

where IG $(x;a) = \int_0^x \frac{1}{\Gamma(a)} t^{a-1} e^{-t} dt$ is the incomplete gamma function.

Random Number Generation

Algorithm PG due to Ahrens and Dieter (1974)

1. (Initialization) Set *X*=0 and *w*=*a*.
2. If *w*>16, go to step 6.
3. Set *c*=exp(−*w*) and *p*=1.
4. Generate *U* from Uniform(0,1). Set *p*=*pU*.
5. If *p*<*c*, continue with step 6; otherwise set *X*=*X*+1 and go to step 4.
6. Set $n = [7w/8]$. Generate *G* from Gamma(*n*, 1).
7. If *G*>*w*, generate *Y* from Binomial(*n*−1, *w*/*G*), set *X*=*X*+*Y*.
8. If *G*≤*w*, set $X = X + n, w = w - G,$ and go to step 2.

Notes. [*y*] means the integer part of *y*.

Steps 3 to 5 of Algorithm PG are in fact the direct algorithm.

References. RV: (Ahrens et al., 1974).

Special Functions

These are not distribution functions, but are used in the functional definition of one or more distributions.

Gamma Function

$$\Gamma(a) = \int_0^\infty x^{a-1} e^{-x} dx \quad a > 0$$

The gamma function has the following properties:

- $\Gamma(1) = 1$
- $\Gamma(1/2) = \sqrt{\pi}$
- $\Gamma(a) = (a-1)\Gamma(a-1) \quad a > 1$
- $\Gamma(a) = (a-1)!$ when *a* is a positive integer

Note. Since Γ(*a*) can be very large even for a moderate value of *a*, the (natural) logarithm of Γ(*a*) is computed instead.

References. The ln(Γ(*a*)) function: CACM 291 (1966) and AS 245 (1989). (See the “Algorithm Index” and “References”)

Beta Function

$$\mathrm{B}(a,b) = \int_0^1 x^{a-1}(1-x)^{b-1}dx \quad a>0, b>0$$

The beta function has the following properties:

- $\mathrm{B}(a,1) = 1/a$
- $\mathrm{B}\left(\frac{1}{2},\frac{1}{2}\right) = \pi$
- $\mathrm{B}(a,b) = \Gamma(a)\Gamma(b)/\Gamma(a+b)$
- $\mathrm{B}(b,a) = \mathrm{B}(a,b)$
- $\mathrm{B}(a,b) = (b-1)\mathrm{B}(a+1,b-1)/a$
- $\mathrm{B}(a,b) = (a+b)\mathrm{B}(a+1,b)/a$

Note. Usually, $B(x, y)$ is calculated as:

$$\mathrm{B}(x,y) = \exp\left(\ln(\Gamma(x)) + \ln(\Gamma(y)) - \ln(\Gamma(x+y))\right)$$

Incomplete Gamma Function (Ratio)

$$\mathrm{IG}(x;a) = \int_0^x \frac{1}{\Gamma(a)} t^{a-1}e^{-t}dt \quad x \geq 0$$
$$x_p = \mathrm{IG}^{-1}(p;a) \Leftrightarrow p = \mathrm{IG}(x_p;a)\, 0 \leq p \leq 1$$

for a>0

The incomplete gamma function has the following properties:

- $\mathrm{IG}(x;1) = 1 - e^{-x}$
- Using integration by parts, for a>1

$$\mathrm{IG}(x;a) = \frac{1}{\Gamma(a+1)} x^a e^{-x} + \mathrm{IG}(x;a+1)$$

Note. $\mathrm{IG}^{-1}(1,a) = \infty$.

References. AS 32 (1970), AS 147 (1980), and AS 239 (1988). (See the “Algorithm Index” and “References”)

Incomplete Beta Function (Ratio)

$$\mathrm{IB}(x;a,b) = \int_0^x \frac{1}{\mathrm{B}(a,b)} t^{a-1}(1-t)^{b-1}dt 0 \leq x \leq 1$$
$$x_p = \mathrm{IB}^{-1}(p;a,b) \Leftrightarrow p = \mathrm{IB}(x_p;a.b)\, 0 \leq p \leq 1$$
for $a > 0$ and $b > 0$

The incomplete beta function has the following properties:

- $\mathrm{IB}(x;a,1) = x^a$
- Using the transformation $t = \sin^2\theta$, we get $\mathrm{IB}\left(x;\frac{1}{2},\frac{1}{2}\right) = \frac{2}{\pi}\sin^{-1}\sqrt{x}$

- $\mathrm{IB}(x;a,b) = 1 - (1 \quad - x;b,a)$
- Using integration by parts, we get, for $b>1$,

  $\mathrm{IB}(x;a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}x^a(1-x)^{b-1} + \mathrm{IB}(x;a+1,b-1)$
- Using the fact that $\frac{d}{dx}x^a(1-x)^b = ax^{a-1}(1-x)^{b-1} - (a+b)x^a(1-x)^{b-1}$ we have

  $\mathrm{IB}(x;a,b) = \frac{\Gamma(a+b)}{\Gamma(a+1)\Gamma(b)}x^a(1-x)^b + \mathrm{IB}(x;a+1,b)$

References. AS 63 (1973); Inverse: AS 64 (1973), AS 109 (1977). (See the “Algorithm Index” and “References”)

Standard Normal

$$\Phi(x) = \int_{-\infty}^{x} \frac{1}{\sqrt{2\pi}} e^{-u^2/2} du - \infty < x < \infty$$
$$x_p = \Phi^{-1}(p) \Leftrightarrow \Phi(x) = p$$

For Φ^{-1}, the Abramowitz and Stegun method is used.

References. AS 66 (1973); Inverse: AS 111 (1977) and AS 241 (1988). See (Patel and Read, 1982) for related distributions, and see the “Algorithm Index” and “References”.

Algorithm Index

AS 3: (Cooper, 1968)a

AS 5: (Cooper, 1968)b

AS 27: (Taylor, 1970)

AS 32: (Bhattacharjee, 1970)

AS 63: (Majumder and Bhattacharjee, 1973)a

AS 64: (Majumder and Bhattacharjee, 1973)b

AS 66: (Hill, 1973)

AS 76: (Young and Minder, 1974)

AS 91: (Best and Roberts, 1975)

AS 109: (Cran, Martin, and Thomas, 1977)

AS 111: (Beasley and Springer, 1977)

AS 134: (Atkinson and Whittaker, 1979)

AS 147: (Lau, 1980)

AS 152: (Lund, 1980)

AS 170: (Narula and Desu, 1981)

AS 190: (Lund and Lund, 1983) , Correction (Lund and Lund, 1985), Remark (Royston, 1987)

AS 195: (Schervish, 1984)

AS 226: (Lenth, 1987)

AS 231: (Farebrother, 1987)

AS 239: (Shea, 1988)

AS 241: (Wichura, 1988)

AS 243: (Lenth, 1989)

AS 245: (Macleod, 1989)

AS 275: (Ding, 1992)

AS 462: (Donnelly, 1973)

AS R85: Shea (1991)

CACM 291: (Pike and Hill, 1966)

CACM 299: (Hill and Pike, 1967)

CACM 332: (Dorrer, 1968)

CACM 395: (Hill, 1970)a

CACM 396: (Hill, 1970)b

CACM 451: (Goldstein, 1973)

CACM 488: (Brent, 1974)

References

Abramowitz, M., and I. A. Stegun, eds. 1970. *Handbook of mathematical functions*. New York: Dover Publications.

Ahrens, J. H., and U. Dieter. 1974. Computer methods for sampling from gamma, beta, Poisson and binomial distributions. *Computing*, 12, 223–246.

Atkinson, A. C., and J. Whittaker. 1979. Algorithm AS 134: The generation of beta random variables with one parameter greater than and one parameter less than 1. *Applied Statistics*, 28, 90–93.

Beasley, J. D., and S. G. Springer. 1977. Algorithm AS 111: The percentage points of the normal distribution. *Applied Statistics*, 26, 118–121.

Berger, R. L. 1991. AS R86: A remark on algorithm AS 152. *Applied Statistics*, 40, 374–375.

Best, D. J., and D. E. Roberts. 1975. Algorithm AS 91: The percentage points of the c2 distribution. *Applied Statistics*, 24, 385–388.

Bhattacharjee, G. P. 1970. Algorithm AS 32: The incomplete gamma integral. *Applied Statistics*, 19, 285–287.

Box, G. E. P., and M. E. Muller. 1958. A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, 29, 610–611.

Bratley, P., and L. E. Schrage. 1987. *A Guide to Simulation*. New York: Springer-Verlag.

Brent, R. P. 1974. Algorithm 488: A Gaussian pseudo–random number generator. *Communications of the ACM*, 17, 704–706.

Cheng, R. C. H. 1978. Generating beta variates with nonintegral shape parameters. *Communications of the ACM*, 21, 317–322.

Cooper, B. E. 1968. Algorithm AS 3: The integral of Student's t distribution. *Applied Statistics*, 17, 189–190.

Cooper, B. E. 1968. Algorithm AS 5: The integral of the noncentral t distribution. *Applied Statistics*, 17, 193–194.

Cran, G. W., K. J. Martin, and G. E. Thomas. 1977. Algorithm AS 109: A remark on algorithms: AS 63 and AS 64 (replacing AS 64). *Applied Statistics*, 26, 111–114.

Ding, C. G. 1992. Algorithm AS 275: Computing the noncentral chi-squared distribution function. *Applied Statistics*, 41, 478–482.

Donnelly, T. G. 1973. Algorithm 462: Bivariate Normal Distribution. *Communications of ACM*, 16, 638.

Dorrer, E. 1968. Algorithm 332: F-distribution. *Communications of the ACM*, 11, 115–116.

Farebrother, R. W. 1987. Algorithm AS 231: The distribution of a noncentral c2 variable with nonnegative degrees of freedom (Correction: 38: 204). *Applied Statistics*, 36, 402–405.

Fishman, G. S. 1976. Sampling from the gamma distribution on a computer. *Communications of the ACM*, 19, 407–409.

Frick, H. 1990. Algorithm AS R84: A remark on algorithm AS 226. *Applied Statistics*, 39, 311–312.

Goldstein, R. B. 1973. Algorithm 451: Chi-square quantiles. *Communications of the ACM*, 16, 483–485.

Hill, G. W. 1970. Algorithm 395: Student's t-distribution. *Communications of the ACM*, 13, 617–619.

Hill, G. W. 1970. Algorithm 396: Student's t-quantiles. *Communications of the ACM*, 13, 619–620.

Hill, I. D. 1973. Algorithm AS 66: The normal integral. *Applied Statistics*, 22, 424–424.

Hill, I. D., and A. C. Pike. 1967. Algorithm 299: Chi-squared integral. *Communications of the ACM*, 10, 243–244.

Jöhnk, M. D. 1964. Erzeugung von Betaverteilten und Gammaverteilten Zufallszahlen. *Metrika*, 8, 5–15.

Johnson, N. L., S. Kotz, and A. W. Kemp. 1992. *Univariate Discrete Distributions*, 2 ed. New York: John Wiley.

Johnson, N. L., S. Kotz, and N. Balakrishnan. 1994. *Continuous Univariate Distributions*, 2 ed. New York: John Wiley.

Kennedy, W. J., and J. E. Gentle. 1980. *Statistical computing*. New York: Marcel Dekker.

Kinderman, A. J., and J. G. Ramage. 1976. Computer generation of normal random variables (Correction: 85: 212). *Journal of the American Statistical Association*, 71, 893–896.

Lau, C. L. 1980. Algorithm AS 147: A simple series for the incomplete gamma integral. *Applied Statistics*, 29, 113–114.

Lenth, R. V. 1987. Algorithm AS 226: Computing noncentral beta probabilities (Correction: 39: 311–312). *Applied Statistics*, 36, 241–244.

Lenth, R. V. 1989. Algorithm AS 243: Cumulative distribution function of the noncentral t distribution. *Applied Statistics*, 38, 185–189.

Lund, R. E. 1980. Algorithm AS 152: Cumulative hypergeometric probabilities. *Applied Statistics*, 29, 221–223.

Lund, R. E., and J. R. Lund. 1983. Algorithm AS 190: Probabilities and upper quantiles for the studentized range. , 32, 204–210.

Lund, R. E., and J. R. Lund. 1985. Correction to Algorithm AS 190. , 34, 104–.

Macleod, A. J. 1989. Algorithm AS 245: A robust and reliable algorithm for the logarithm of the gamma function. *Applied Statistics*, 38, 397–402.

Majumder, K. L., and G. P. Bhattacharjee. 1973. Algorithm AS 63: The incomplete beta integral.. *Applied Statistics*, 22, 409–411.

Majumder, K. L., and G. P. Bhattacharjee. 1973. Algorithm AS 64: Inverse of the incomplete beta function ratio. *Applied Statistics*, 22, 412–414.

Marsaglia, G. 1962. Random variables and computers. In: *Information theory statistical decision functions random processes: Transactions of the third Prague conference,* J. Kozesnik, ed. Prague, Czechoslovak: Czechoslovak Academy of Science, 499–510.

Michael, J., W. Schucany, and R. Haas. 1976. Generating random variates using transformation with multiple roots. *American Statistician*, 30, 88–90.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Approximations for the doubly noncentral F-distribution. *Communications in Statistics, Part A*, 5, 49–53.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Some Approximations for the noncentral F-distribution. *Technometrics*, 18, 351–358.

Mudholkar, G., and R. Natarajan. 1999. Approximations for the inverse Gaussian probabilities and percentiles. *Communications in Statistics - Simulation and Computation*, 28:4, 1051–1071.

Narula, S. C., and M. M. Desu. 1981. Computation of probability and noncentrality parameter of a noncentral chi-square distribution. *Applied Statistics*, 30, 349–352.

Patel, J. K., and C. B. Read. 1982. *Handbook of the normal distribution*. New York: Marcel Dekker.

Pike, M. C., and I. D. Hill. 1966. Algorithm 291: Logarithm of gamma function. *Communications of the ACM*, 9, 684–684.

Royston, J. P. 1987. AS R69: A remark on Algorithm AS 190. *Applied Statistics*, 36, 119.

Schervish, M. J. 1984. Algorithm AS 195: Multivariate normal probabilities with error bound. *Applied Statistics*, 33, 81–94.

Schrage, L. 1979. A more portable Fortran random number generator. *ACM Transactions on Mathematical Software*, 5:2, 132–132.

Shea, B. L. 1988. Algorithm AS 239: Chi-squared and incomplete gamma integral. *Applied Statistics*, 37, 466–473.

Shea, B. L. 1989. AS R77: A remark on algorithm AS 152. *Applied Statistics*, 38, 199–204.

Tadikamalla, P. R. 1978. Computer generation of gamma random variables. *Communications of the ACM*, 21, 419–422.

Taylor, G. A. R. 1970. Algorithm AS 27: The integral of Student’s t-distribution. *Applied Statistics*, 19, 113–114.

Von Neumann, J. 1951. Various techniques used in connection with random digits. *National Bureau of Standards Applied Mathematics*, 12, 36–38.

Wichura, M. J. 1988. Algorithm AS 241: The percentage points of the normal distribution. *Applied Statistics*, 37, 477–484.

Young, J. C., and C. E. Minder. 1974. Algorithm AS 76: An integral useful in calculating noncentral t and bivariate normal probabilities. *Applied Statistics*, 23, 455–457.

Appendix
J

Box’s M Test

Box’s *M* statistic is used to test for homogeneity of covariance matrices. The *j*th set of *r* dependent variables in the *i*th cell are $y'_{ij} = x'_{ij}\mathbf{B} + e'_{ij}$ where $e_{ij} \sim N_r\left(0, w_{ij}^{-1}\Sigma_i\right)$ for *i*=1,...,*g* and $j = 1, \ldots, n_i$. The null hypothesis of the test for homogeneity of covariance matrices is $H_o : \Sigma_1 = \cdots = \Sigma_g$. Box (1949) derived a test statistic based on the likelihood-ratio test. The test statistic is called Box’s *M* statistic. For moderate to small sample sizes, an *F* approximation is used to compute its significance.

Box’s *M* statistic is not designed to be used in a linear model context; therefore the observed cell means are used in computing the statistic.

Note: Although Anderson (Anderson, 1958) mentioned that the population cell means can be expressed as linear combinations of parameters, he assumed that the combination coefficients are different for different cells, which is not the model assumed for GLM.

Notation

The following notation is used throughout this chapter unless otherwise stated:

Table J-1
Notation

Notation	**Description**
g	Number of cells with non-singular covariance matrices.
n_i	Number of cases in the *i*th cell.
n	Total sample size, $n = n_1 + \cdots + n_g$
$\mathbf{y}_{ij}$	The *j*th set of dependent variables in the *i*th cell. A column vector of length *r*.
w_{ij}	Regression weight associated with $\mathbf{y}_{ij}$. It is assumed $w_{ij} > 0$.

Means

$$\bar{\mathbf{y}}_i = \Sigma_{j=1}^{n_i}\mathbf{y}_{ij}/n_i$$

Cell Covariance Matrix

$$\mathbf{S}_i = \begin{cases} \Sigma_{j=1}^{n_i} w_{ij}(\mathbf{y}_{ij} - \bar{\mathbf{y}}_i)(\mathbf{y}_{ij} - \bar{\mathbf{y}}_i)'/(n_i - 1) & \text{if } n_i > 1 \\ \mathbf{0} & \text{if } n_i \le 1 \end{cases}$$

Pooled Covariance Matrix

$$\mathbf{S} = \begin{cases} \Sigma_{i=1}^{g}(n_i - 1)\mathbf{S}_i/(n - g) & \text{if } n > g \\ \mathbf{0} & \text{if } n \le g \end{cases}$$

Box's M Statistic

$$M = \begin{cases} (n-g)\log|\mathbf{S}| - \sum_{i=1}^{g}(n_i-1)\log|\mathbf{S}_i| & \text{if } |\mathbf{S}|>0 \\ \text{SYSMIS} & \text{if } |\mathbf{S}|\le 0 \end{cases}$$

Significance

$$1 - CDF.F(\gamma M, f_1, f_2)$$

where CDF.F is the IBM® SPSS® Statistics function for the cumulative *F* distribution and

$$f_1 = (g-1)r(r+1)/2$$
$$\rho = 1 - \tfrac{2r^2+3r-1}{6(r+1)(g-1)}\left(\sum_{i=1}^{g}\frac{1}{(n_i-1)} - \frac{1}{(n-g)}\right)$$
$$\tau - \tfrac{(r-1)(r+2)}{6(g-1)}\left(\sum_{i=1}^{g}\frac{1}{(n_i-1)^2} - \frac{1}{(n-g)^2}\right)$$
$$f_2 = \tfrac{f_1+2}{\left|\tau-(1-\rho)^2\right|}$$
$$\gamma = \tfrac{(\rho - f_1/f_2)}{f_1}$$

The significance is a system-missing value whenever the denominator is zero in the above expression.

References

Anderson, T. W. 1958. *Introduction to multivariate statistical analysis*. New York: John Wiley & Sons, Inc..

Box, G. E. P. 1949. A general distribution theory for a class of likelihood criteria. *Biometrika*, 36, 317–346.

Seber, G. A. F. 1984. *Multivariate observations*. New York: John Wiley & Sons, Inc.

Appendix

K

Confidence Intervals for Percentages and Counts Algorithms

This document describes the algorithms for computing confidence intervals for percentages and counts for bar charts. The data are assumed to be from a simple random sample, and each confidence interval is a separate or individual interval, based on a binomial proportion of the total count. The computed binomial intervals are equal-tailed Jeffreys prior intervals (see Brown, Cai, & DasGupta, 2001, 2002, 2003). Note that they are generally not symmetric around the observed proportion. Therefore, the plotted interval bounds are generally not symmetric around the observed percentage or count.

Notation

The following notation is used throughout this section unless otherwise noted:

Table K-1
Notation

Notation	**Description**
X_i	Distinct values of the category axis variable X_i
w_i	Rounded sum of weights for cases with value
$W = \sum_i w_i$	Total sum of weights over values of X
P_i	Population proportion of cases at X_i
•	Specified error level for 100(1- •)% confidence intervals

IDF.BETA(p,shape1,shape2) in COMPUTE gives the p^{th} quantile of the beta distribution or incomplete beta function with shape parameters shape1 and shape2. For a precise mathematical definition, see “Beta Function”.

Confidence Intervals for Counts

Lower bound for $W p_i$ = W [IDF.BETA(•/2,wi +.5,W−wi +.5)].

Upper bound for $W p_i$ = W [IDF.BETA(1-•/2,wi +.5,W−wi +.5)].

Standard error for $W p_i$ = $W \times \sqrt{(w_i/W)(1-(w_i/W))/W}$

Confidence Intervals for Percentages

Lower bound for 100 p_i = 100 [IDF.BETA(•/2,wi +.5,W−wi +.5)].

Upper bound for 100 p_i = 100 [IDF.BETA(1-•/2,wi +.5,W−wi +.5)].

Standard error for $p_i = 100 \times \sqrt{(w_i/W)(1-(w_i/W))/W}$

References

Brown, L. D., T. Cai, and A. DasGupta. 2001. Interval estimation for a binomial proportion. *Statistical Science*, 16(2), 101–133.

Brown, L. D., T. Cai, and A. DasGupta. 2002. Confidence intervals for a binomial Proportion and asymptotic expansions. *The Annals of Statistics*, 30(4), 160–201.

Brown, L. D., T. Cai, and A. DasGupta. 2003. Interval estimation in exponential families. *Statistica Sinica*, 13, 19–49.

Appendix
L

Notices

Trademarks

Bibliography

Abraham, B., and J. Ledolter. 1983. *Statistical methods of forecasting*. New York: John Wiley and Sons.

Abramowitz, M., and I. A. Stegun, eds. 1970. *Handbook of mathematical functions*. New York: Dover Publications.

Addelman, S. 1962. Symmetrical and asymmetrical fractional factorial plans. *Technometrics*, 4, 47–58.

Agresti, A. 2002. *Categorical Data Analysis*, 2nd ed. New York: John Wiley and Sons.

Agresti, A. 1996. *An Introduction to Categorical Data Analysis*. New York: John Wiley and Sons.

Agresti, A. 1990. *Categorical Data Analysis*. New York: John Wiley.

Agresti, A., J. G. Booth, and B. Caffo. 2000. Random-effects Modeling of Categorical Response Data. *Sociological Methodology*, 30, 27–80.

Ahrens, J. H., and U. Dieter. 1974. Computer methods for sampling from gamma, beta, Poisson and binomial distributions. *Computing*, 12, 223–246.

Aitkin, M., D. Anderson, B. Francis, and J. Hinde. 1989. *Statistical Modelling in GLIM*. Oxford: Oxford Science Publications.

Akaike, H. 1974. A new look at the statistical model identification. *IEEE Transaction on Automatic Control* , AC–19, 716–723.

Albert, A., and J. A. Anderson. 1984. On the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, 71, 1–10.

Anderberg, M. R. 1973. *Cluster analysis for applications*. New York: Academic Press.

Anderson, T. W. 1958. *Introduction to multivariate statistical analysis*. New York: John Wiley & Sons, Inc..

Andrews, F., J. Morgan, J. Sonquist, and L. Klein. 1973. *Multiple classification analysis*, 2nd ed. Ann Arbor: University of Michigan.

Arya, S., and D. M. Mount. 1993. Algorithms for fast vector quantization. In: *Proceedings of the Data Compression Conference 1993*, , 381–390.

Atkinson, A. C., and J. Whittaker. 1979. Algorithm AS 134: The generation of beta random variables with one parameter greater than and one parameter less than 1. *Applied Statistics*, 28, 90–93.

Bamber, D. 1975. The area above the ordinal dominance graph and the area below the receiver operating graph. *Journal of Mathematical Psychology*, 12, 387–415.

Barlow, R. E., D. J. Bartholomew, D. J. Bremner, and H. D. Brunk. 1972. *Statistical inference under order restrictions*. New York: John Wiley and Sons.

Bartlett, M. S. 1946. On the theoretical specification of sampling properties of autocorrelated time series. *Journal of Royal Statistical Society, Series B*, 8, 27–27.

Bayes, T. 1763. An essay towards solving a problem in the doctrine of chances. *Philosophical Transactions of the Royal Society*, 53, 370–418.

Beasley, J. D., and S. G. Springer. 1977. Algorithm AS 111: The percentage points of the normal distribution. *Applied Statistics*, 26, 118–121.

Beck, R. J., and E. Shultz. 1986. The use of relative operating characteristic (ROC) curves in test performance evaluation. *Arch. Pathol. Lab. Med.*, 110, 13–20.

Becker, B., R. Kohavi, and D. Sommerfield. 2001. Visualizing the Simple Bayesian Classifier. In: *Information Visualization in Data Mining and Knowledge Discovery,* U. Fayyad, G. Grinstein, and A. Wierse, eds. San Francisco: Morgan Kaufmann Publishers, 237–249.

Belsley, D. A., E. Kuh, and R. E. Welsch. 1980. *Regression diagnostics: Identifying influential data and sources of collinearity*. New York: John Wiley and Sons.

Bentler, P. M., and D. G. Weeks. 1978. Restricted multidimensional scaling models. *Journal of Mathematical Psychology*, 17, 138–151.

Benzécri, J. P. 1969. Statistical analysis as a tool to make patterns emerge from data. In: *Methodologies of Pattern Recognition,* S. Watanabe, ed. New York: Academic Press, 35–74.

Berger, R. L. 1991. AS R86: A remark on algorithm AS 152. *Applied Statistics,* 40, 374–375.

Best, D. J., and D. E. Roberts. 1975. Algorithm AS 91: The percentage points of the c2 distribution. *Applied Statistics*, 24, 385–388.

Bhattacharjee, G. P. 1970. Algorithm AS 32: The incomplete gamma integral. *Applied Statistics*, 19, 285–287.

Biggs, D., B. de Ville, and E. Suen. 1991. A method of choosing multiway partitions for classification and decision trees. *Journal of Applied Statistics,* 18, 49–62.

Biller, B., and S. Ghosh. 2006. Multivariate input processes. In: *Handbooks in Operations Research and Management Science: Simulation,* B. L. Nelson, and S. G. Henderson, eds. Amsterdam: Elsevier Science, 123–153.

Binder, D. A. 1983. On the variances of asymptotically normal estimators from complex surveys. *International Statistical Review*, 51, 279–292.

Binder, D. A. 1992. Fitting Cox’s Proportional Hazards Models from Survey Data. *Biometrika*, 79, 139–147.

Bishop, Y. M., S. E. Feinberg, and P. W. Holland. 1975. *Discrete multivariate analysis: Theory and practice*. Cambridge, Mass.: MIT Press.

Bishop, Y. M., S. E. Fienberg, and P. W. Holland. 1977. *Discrete Multivariate Analysis: Theory and Practice*. Cambridge, MA: MIT Press.

Bishop, C. M. 1995. *Neural Networks for Pattern Recognition*, 3rd ed. Oxford: Oxford University Press.

Björk, A., and G. H. Golub. 1973. Numerical methods for computing angles between linear subspaces. *Mathematics of Computation*, 27, 579–594.

Blalock, H. M. 1972. *Social statistics*. New York: McGraw-Hill.

Bliss, C. I. 1967. *Statistics in biology, Volume 1*. New York: McGraw-Hill.

Blom, G. 1958. *Statistical estimates and transformed beta variables*. New York: John Wiley and Sons.

Bloomfield, P. 1976. *Fourier analysis of time series*. New York: John Wiley and Sons.

Bloxom, B. 1978. Contrained multidimensional scaling in n spaces. *Psychometrika*, 43, 397–408.

Bock, R. D. 1975. *Multivariate statistical methods in behavioral research*. New York: McGraw-Hill.

Bowker, A. H. 1948. A test for symmetry in contingency tables. *Journal of the American Statistical Association*, 43, 572–574.

Box, G. E. P., and D. R. Cox. 1964. An analysis of transformations. *Journal of the Royal Statistical Society, Series B*, 26, 211–246.

Box, G. E. P. 1949. A general distribution theory for a class of likelihood criteria. *Biometrika*, 36, 317–346.

Box, G. E. P., and M. E. Muller. 1958. A note on the generation of random normal deviates. *Annals of Mathematical Statistics*, 29, 610–611.

Box, G. E. P., and G. M. Jenkins. 1976. *Time series analysis: Forecasting and control*, Rev. ed. San Francisco: Holden-Day.

Box, G. E. P., G. M. Jenkins, and G. C. Reinsel. 1994. *Time series analysis: Forecasting and control*, 3rd ed. Englewood Cliffs, N.J.: Prentice Hall.

Bozdogan, H. 1987. Model Selection and Akaike’s Information Criterion (AIC): The General Theory and Its Analytical Extensions. *Psychometrika*, 52, 345–370.

Bratley, P., and L. E. Schrage. 1987. *A Guide to Simulation*. New York: Springer-Verlag.

Breiman, L., J. H. Friedman, R. A. Olshen, and C. J. Stone. 1984. *Classification and Regression Trees*. New York: Chapman & Hall/CRC.

Brent, R. P. 1974. Algorithm 488: A Gaussian pseudo–random number generator. *Communications of the ACM*, 17, 704–706.

Breslow, N. E. 1974. Covariance analysis of censored survival data. *Biometrics*, 30, 89–99.

Breslow, N. E., and N. E. Day. 1980. *Statistical Methods in Cancer Research, 1, The Analysis of Case-Control Studies*. : International Agency for Research on Cancer, Lyon..

Brewer, K. W. R. 1963. A Model of Systematic Sampling with Unequal Probabilities. *Australian Journal of Statistics*, 5, 93–105.

Brigham, E. O. 1974. *The fast Fourier transform*. Englewood Cliffs, N.J.: Prentice-Hall.

Brockwell, P. J., and R. A. Davis. 1991. *Time Series: Theory and Methods*, 2 ed. : Springer-Verlag.

Brown, M. B., and A. B. Forsythe. 1974b. Robust tests for the equality of variances. *Journal of the American Statistical Association*, 69, 364–367.

Brown, M. B. 1975. The asymptotic standard errors of some estimates of uncertainty in the two-way contingency table. *Psychometrika*, 40(3), 291.

Brown, M. B., and J. K. Benedetti. 1977. Sampling behavior of tests for correlation in two-way contingency tables. *Journal of the American Statistical Association*, 72, 309–315.

Brown, L. D., T. Cai, and A. DasGupta. 2001. Interval estimation for a binomial proportion. *Statistical Science*, 16(2), 101–133.

Brown, L. D., T. Cai, and A. DasGupta. 2002. Confidence intervals for a binomial Proportion and asymptotic expansions. *The Annals of Statistics*, 30(4), 160–201.

Brown, L. D., T. Cai, and A. DasGupta. 2003. Interval estimation in exponential families. *Statistica Sinica*, 13, 19–49.

Brown, M. B., and A. B. Forsythe. 1974a. The small sample behavior of some statistics which test the equality of several means. *Technometrics*, 16, 129–132.

Brownlee, K. A. 1965. *Statistical theory and methodology in science and engineering*. New York: John Wiley & Sons, Inc.

Cain, K. C., and N. T. Lange. 1984. Approximate case influence for the proportional hazards regression model with censored data. *Biometrics*, 40, 493–499.

Cameron, A. C., and P. K. Trivedi. 1998. *Regression Analysis of Count Data*. Cambridge: Cambridge University Press.

Campbell, G., and J. H. Skillings. 1985. Nonparametric Stepwise Multiple Comparison Procedures. *Journal of the American Statistical Association*, 80, 998–998.

Carroll, J. D., and J. J. Chang. 1970. Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition. *Psychometrika*, 35, 238–319.

Carroll, J. D., and J. J. Chang. 1972. *IDIOSCAL (Individual differences in orientation scaling). Paper presented at the spring meeting of the Psychometric Society, Princeton, New Jersey*. : .

Carroll, J. D., S. Pruzansky, and J. B. Kruskal. 1980. CANDELINC: A general approach to multidimensional analysis with linear constraints on parameters. *Psychometrika*, 45, 3–24.

Centor, R. M., and J. S. Schwartz. 1985. An evaluation of methods for estimating the area under the receiver operating statistic (ROC) curve. *Med. Decis. Making*, 5, 149–156.

Chambers, R., and C. Skinner, eds. 2003. *Analysis of Survey Data*. New York: John Wiley& Sons.

Chambers, J. M., W. S. Cleveland, B. Kleiner, and P. A. Tukey. 1983. *Graphical methods for data analysis*. Boston: Duxbury Press.

Cheng, R. C. H. 1978. Generating beta variates with nonintegral shape parameters. *Communications of the ACM*, 21, 317–322.

Cheng, P. H., and C. Y. K. Meng. 1992. A New Formula for Tail probabilities of DUNNETT's T with Unequal Sample Sizes. *ASA Proc. Stat. Comp.*, , 177–182.

Chiu, T., D. Fang, J. Chen, Y. Wang, and C. Jeris. 2001. A Robust and Scalable Clustering Algorithm for Mixed Type Attributes in Large Database Environment. In: *Proceedings of the seventh ACM SIGKDD international conference on knowledge discovery and data mining,* SanFrancisco, CA: ACM, 263–268.

Christensen, R. 1990. *Log-linear models*. New York: Springer-Verlag.

Chromy, J. R. 1979. . *Sequential Sample Selection MethodsProceedings of the American Statistical Association, Survey Research Methods Section*, , 401–406.

Clarke, M. R. B. 1982. Algorithm AS 178: The Gauss-Jordan sweep operator with detection of collinearity. *Applied Statistics*, 31:2, 166–168.

Cliff, N. 1966. Orthogonal rotation to congruence. *Psychometrika*, 31, 33–42.

Cochran, W. G. 1977. *Sampling Techniques*, 3rd ed. New York: John Wiley and Sons.

Cochran, W. G. 1954. Some methods of strengthening the common chi-square tests. *Biometrics*, 10, 417–451.

Cohen, A., and M. Rom. 1994. A Method for Hypothesis Tests in Polychotomous Logistic Regression. *Computational Statistics and Data Analysis*, 17, 277–288.

Collett, D. 2003. *Modelling survival data in medical research*, 2 ed. Boca Raton: Chapman & Hall/CRC.

Commandeur, J. J. F., and W. J. Heiser. 1993. *Mathematical derivations in the proximity scaling (PROXSCAL) of symmetric data matrices*. Leiden: Department of Data Theory, University of Leiden.

Committee E-11 on Quality and Statistics, . 1990. *ASTM Special Technical Publication (STP) 15D: Manual on presentation of data and control chart analysis*, 6 ed. Philadelphia: American Society for Testing and Materials.

Cook, R. D. 1977. Detection of influential observations in linear regression. *Technometrics*, 19, 15–18.

Cooley, W. W., and P. R. Lohnes. 1971. *Multivariate data analysis*. New York: John Wiley & Sons, Inc..

Cooper, B. E. 1968. Algorithm AS 3: The integral of Student's t distribution. *Applied Statistics*, 17, 189–190.

Cooper, B. E. 1968. Algorithm AS 5: The integral of the noncentral t distribution. *Applied Statistics*, 17, 193–194.

Cox, D. R. 1972. Regression models and life tables (with discussion). *Journal of the Royal Statistical Society, Series B*, 34, 187–220.

Cox, D. R., and E. J. Snell. 1989. *The Analysis of Binary Data*, 2nd ed. London: Chapman and Hall.

Cramer, J. S., and G. Ridder. 1988. The Logit Model in Economics. *Statistica Neerlandica*, 42, 291–314.

Cran, G. W., K. J. Martin, and G. E. Thomas. 1977. Algorithm AS 109: A remark on algorithms: AS 63 and AS 64 (replacing AS 64). *Applied Statistics*, 26, 111–114.

Cronbach, L. J. 1951. Coefficient alpha and the internal structure of tests. *Psychometrika*, 16:3, 297–334.

Cryer, J. D. 1986. *Time series analysis*. Boston, Mass.: Duxbury Press.

Cunningham, P., and S. J. Delaney. 2007. k-Nearest Neighbor Classifiers. *Technical Report UCD-CSI-2007-4, School of Computer Science and Informatics, University College Dublin, Ireland*, , – .

Dallal, G. E., and L. Wilkinson. 1986. An analytic approximation to the distribution of Lilliefor's test statistic for normality. *The American Statistician*, 40(4): 294–296 (Correction: 41: 248), – .

Davison, A. C., and D. V. Hinkley. 2006. *Bootstrap Methods and their Application*. : Cambridge University Press.

De Leeuw, J. 1977. Applications of convex analysis to multidimensional scaling. In: *Recent developments in statistics*, J. R. Barra, F. Brodeau, G. Romier, and B. van Cutsem,   eds. Amsterdam,The Netherlands: North-Holland, 133–145.

De Leeuw, J. 1984. *Canonical analysis of categorical data*, 2nd ed. Leiden: DSWO Press.

De Leeuw, J., and W. J. Heiser. 1980. Multidimensional scaling with restrictions on the configuration. In: *Multivariate Analysis, Vol. V*, P. R. Krishnaiah, ed. Amsterdam: North-Holland, 501–522.

De Leeuw, J., and J. Van Rijckevorsel. 1980. HOMALS and PRINCALS—Some generalizations of principal components analysis. In: *Data Analysis and Informatics*, E. Diday,et al., ed. Amsterdam: North-Holland, 231–242.

De Leeuw, J., F. W. Young, and Y. Takane. 1976. Additive structure in qualitative data: An alternating least squares method with optimal scaling features. *Psychometrika*, 41, 471–503.

Dempster, A. P., N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, B*, 39, 1–38.

Dempster, A. P. 1969. *Elements of Continuous Multivariate Analysis*. Reading, MA: Addison-Wesley.

Diggle, P. J., P. Heagerty, K. Y. Liang, and S. L. Zeger. 2002. *The analysis of Longitudinal Data,* 2 ed. Oxford: Oxford University Press.

Dineen, L. C., and B. C. Blakesley. 1973. Algorithm AS 62: Generator for the sampling distribution of the Mann-Whitney U statistic. *Applied Statistics*, 22, 269–273.

Ding, C. G. 1992. Algorithm AS 275: Computing the noncentral chi-squared distribution function. *Applied Statistics*, 41, 478–482.

Dixon, W. J. 1973. *BMD Biomedical computer programs*. Los Angeles: University of California Press.

Dixon, W. J. 1983. *BMDP statistical software*. Berkeley: University of California Press.

Dobson, A. J. 2002. *An Introduction to Generalized Linear Models*, 2 ed. Boca Raton, FL: Chapman & Hall/CRC.

Domingos, P., and M. J. Pazzani. 1996. Beyond Independence: conditions for the optimality of the simple Bayesian classifier. In: *Machine Learning: Proceedings of the Thirteenth International Conference,* L. Saitta, ed., 105–112.

Donnelly, T. G. 1973. Algorithm 462: Bivariate Normal Distribution. *Communications of ACM*, 16, 638.

Dorfman, D. D., and E. J. Alf. 1968. Maximum likelihood estimation of parameters of signal detection theory—A direct solution. *Psychometrika*, 33, 117–124.

Dorfman, D. D., and E. J. Alf. 1969. Maximum likelihood estimation of parameters of signal-detection theory and determination of confidence intervals—Rating-method data. *Journal of Mathematical Psychology*, 6, 487–496.

Dorrer, E. 1968. Algorithm 332: F-distribution. *Communications of the ACM*, 11, 115–116.

Dougherty, J., R. Kohavi, and M. Sahami. 1995. Supervised and unsupervised discretization of continuous features. In: *Proceedings of the Twelfth International Conference on Machine Learning,* Los Altos, CA: Morgan Kaufmann, 194–202.

Drucker, H. 1997. Improving regressor using boosting techniques. In: *Proceedings of the 14th International Conferences on Machine Learning* , D. H. Fisher,Jr., ed. San Mateo, CA: Morgan Kaufmann, 107–115.

Duncan, D. B. 1955. Multiple Range and Multiple F tests. *Biometrics*, 11, 1–42.

Duncan, D. B. 1975. t Tests and Intervals for Comparisons Suggested by the Data. *Biometrics*, 31, 339–360.

Dunn, O. J. 1964. Multiple Comparisons Using Rank Sums. *Technometrics*, 6, 241–241.

Dunn, P. K., and G. K. Smyth. 2005. Series Evaluation of Tweedie Exponential Dispersion Model Densities. *Statistics and Computing*, 15, 267–280.

Dunn, P. K., and G. K. Smyth. 2001. Tweedie Family Densities: Methods of Evaluation. In: *Proceedings of the 16th International Workshop on Statistical Modelling,* Odense, Denmark: .

Dunnett, C. W. 1955. A Multiple Comparisons Procedure for Comparing Several Treatments with a Control. *Journal of the American Statistical Association*, 50, 1096–1121.

Dunnett, C. W. 1980. Pairwise Multiple Comparisons in Homogeneous Variance, Unequal Sample Size Case. *Journal of the American Statistical Association*, 75, 789–795.

Dunnett, C. W. 1980. Pairwise Multiple Comparisons in the Unequal Variance Case. *Journal of the American Statistical Association,* 75, 796–800.

Dunnett, C. W. 1989. Multivariate Normal Probability Integrals with Product Correlation Structure. *Applied Statistics*, 38, 564–571.

Dziuban, C. D., and E. C. Shirkey. 1974. When is a correlation matrix appropriate for factor analysis?. *Psychological Bulletin*, 81, 358–361.

D'Agostino, R., and M. Stephens. 1986. *Goodness-of-Fit Techniques*. New York: Marcel Dekker.

Eckart, C., and G. Young. 1936. The approximation of one matrix by another one of lower rank. *Psychometrika*, 1, 211–218.

Einot, I., and K. R. Gabriel. 1975. A Study of the powers of Several Methods of Multiple Comparisons. *Journal of the American Statistical Association*, 70, 574–783.

Eisenhart, C., M. W. Hastay, and N. A. Wallis, eds. 1947. *Significance of the largest of a set of sample estimates of variance. In: Techniques of Statistical Analysis*. New York: McGraw-Hill.

Emerson, P. L. 1968. Numerical construction of orthogonal polynomials from a general recurrence formula. *Biometrics*, 24, 695–701.

Fahrmeir, L., and G. Tutz. 2001. *Multivariate Statistical Modelling Based on Generalized Linear Models*, 2nd ed. New York: Springer-Verlag.

Fai, A. H. T., and P. L. Cornelius. 1996. Approximate F-tests of Multiple Degree of Freedom Hypotheses in Generalized Least Squares Analyses of Unbalanced Split-plot Experiments. *Journal of Statistical Computation and Simulation*, 54, 363–378.

Fan, C. T., M. E. Muller, and I. Rezucha. 1962. Development of Sampling Plans by Using Sequential (Item by Item) Selection Techniques and Digital Computers. *Journal of the American Statistical Association*, 57, 387–402.

Farebrother, R. W. 1987. Algorithm AS 231: The distribution of a noncentral c2 variable with nonnegative degrees of freedom (Correction: 38: 204). *Applied Statistics*, 36, 402–405.

Fayyad, U., and K. Irani. 1993. Multi-interval discretization of continuous-value attributes for classification learning. In: *Proceedings of the Thirteenth International Joint Conference on Artificial Intelligence,* San Mateo, CA: Morgan Kaufmann, 1022–1027.

Fellegi, I. P. 1980. Approximate tests of independence and goodness of fit based on stratified multistage samples. *Journal of the American Statistical Association,* 75, 261–268.

Fienberg, S. E. 1994. *The Analysis of Cross-Classified Categorical Data*, 2nd ed. Cambridge, MA: MIT Press.

Fine, T. L. 1999. *Feedforward Neural Network Methodology*, 3rd ed. New York: Springer-Verlag.

Finn, J. D. 1977. Multivariate analysis of variance and covariance. In: *Statistical Methods for Digital Computers, Volume 3,* K. Enslein, A. Ralston, and H. Wilf, eds. New York: John Wiley & Sons, Inc.

Finney, D. J. 1971. *Probit analysis*. Cambridge: Cambridge University Press.

Fishman, G. S. 1976. Sampling from the gamma distribution on a computer. *Communications of the ACM*, 19, 407–409.

Fishman, G., and L. R. I. Moore. 1981. In search of correlation in multiplicative congruential generators with modulus 2**31-1. In: *Computer Science and Statistics, Proceedings of the 13th Symposium on the Interface,* W. F. Eddy, ed. New York: Springer-Verlag, 155–157.

Fox, D. R. 1989. Computer Selection of Size-Biased Samples. *The American Statistician*, 43:3, 168–171.

Fox, J., and G. Monette. 1992. Generalized collinearity diagnostics. *Journal of the American Statistical Association*, 87, 178–183.

Fox, J. 1997. *Applied Regression Analysis, Linear Models, and Related Methods*. Thousand Oaks, CA: SAGE Publications, Inc..

Freund, Y., and R. E. Schapire. 1995. A decision theoretic generalization of on-line learning and an application to boosting. In: *Computational Learning Theory: 7 Second European Conference, EuroCOLT '95,* , 23–37.

Frick, H. 1990. Algorithm AS R84: A remark on algorithm AS 226. *Applied Statistics*, 39, 311–312.

Friedman, J. H., J. L. Bentley, and R. A. Finkel. 1977. An algorithm for finding best matches in logarithm expected time. *ACM Transactions on Mathematical Software*, 3, 209–226.

Frigge, M., D. C. Hoaglin, and B. Iglewicz. 1987. Some implementations for the boxplot. In: *Computer Science and Statistics Proceedings of the 19th Symposium on the Interface,* R. M. Heiberger, and M. Martin, eds. Alexandria, Virginia: AmericanStatistical Association.

Fuller, W. A. 1975. Regression analysis for sample survey. *Sankhya, Series C*, 37, 117–132.

Fuller, W. A. 1976. *Introduction to statistical time series*. New York: John Wiley and Sons.

Gabriel, K. R. 1978. A Simple method of Multiple Comparisons of Means. *Journal of the American Statistical Association*, 73, 724–729.

Games, P. A., and J. F. Howell. 1976. Pairwise Multiple Comparison Procedures with Unequal N's and/or Variances: A Monte Carlo Study. *Journal of Educational Statistics*, 1, 113–125.

Gardner, E. S. 1985. Exponential smoothing: The state of the art. *Journal of Forecasting*, 4, 1–28.

Gebhardt, F. 1966. Approximation to the Critical Values for Duncan's Multiple Range Test. *Biometrics*, 22, 179–182.

Gehan, E. A. 1975. Statistical methods for survival time studies. In: *Cancer Therapy: Prognostic Factors and Criteria*, M. J. Staquet, ed. New York: Raven Press, 7–35.

Gibbons, J. D., and S. Chakraborti. 2003. *Nonparametric Statistical Inference, 4th edition*. : Marcel Dekker.

Giesbrecht, F. G. 1983. An efficient procedure for computing MINQUE of variance components and generalized least squares estimates of fixed effects. *Communications in Statistics, Part A - Theory and Methods*, 12, 2169–2177.

Giesbrecht, F. G., and J. C. Burns. 1985. Two-Stage Analysis Based on a Mixed Model: Large-sample Asymptotic Theory and Small-Sample Simulation Results. *Biometrics*, 41, 477–486.

Gifi, A. 1985. *PRINCALS. Research Report UG-85-02*. Leiden: Department of Data Theory, University of Leiden.

Gifi, A. 1990. *Nonlinear multivariate analysis*. Chichester: John Wiley and Sons.

Gill, P. E., W. M. Murray, and M. A. Saunders. 1981. *Practical Optimization*. London: Academic Press.

Gill, P. E., W. M. Murray, M. A. Saunders, and M. H. Wright. 1984. Procedures for optimization problems with a mixture of bounds and general linear constraints. *ACM Transactions on Mathematical Software*, 10:3, 282–296.

Gill, P. E., W. M. Murray, M. A. Saunders, and M. H. Wright. 1986. *User's guide for NPSOL (version 4.0): A FORTRAN package for nonlinear programming. Technical Report SOL 86-2.* Stanford University: Department of Operations Research.

Gill, J. 2000. *Generalized Linear Models: A Unified Approach*. Thousand Oaks, CA: Sage Publications.

Glaser, R. E. 1983. Levene's Robust Test of Homogeneity of Variances. In: *Encyclopedia of Statistical Sciences 4,* New York: Wiley, p608–610.

Goldstein, R. B. 1973. Algorithm 451: Chi-square quantiles. *Communications of the ACM*, 16, 483–485.

Golmant, J. 1990. Correction: Computer Selection of Size-Biased Samples. *The American Statistician*, , 194–194.

Golub, G. H., and C. Reinsch. 1971. Linear Algebra. In: *Handbook for Automatic Computation, Volume II*, J. H. Wilkinson, and C. Reinsch, eds. New York: Springer-Verlag.

Golub, G. H. 1969. Matrix decompositions and statistical calculations. In: *Statistical Computation*, R. C. Milton, and J. A. Nelder, eds. New York: Academic Press.

Goodman, L. A. 1981. Association Models and Canonical Correlation in the Analysis of Cross-Classifications having Ordered Categories. *Journal of American Statistical Association*, 76, 320–334.

Goodman, L. A. 1979. Simple models for the analysis of association in cross-classifications having ordered categories. *Journal of the American Statistical Association*, 74, 537–552.

Goodman, L. A., and W. H. Kruskal. 1954. Measures of association for cross-classification.. *Journal of the American Statistical Association*, 49, 732–764.

Goodman, L. A., and W. H. Kruskal. 1972. Measures of association for cross-classification, IV: simplification and asymptotic variances. *Journal of the American Statistical Association,* 67, 415–421.

Goodnight, J. H. 1978. Computing MIVQUE0 Estimates of Variance Components. *SAS Technical Report,* R-105, – .

Goodnight, J. H., and W. J. Hemmerle. 1979. A simplified algorithm for the W transformation in variance component estimation. *Technometrics*, 21, 265–267.

Goodnight, J. H. 1979. A tutorial on the SWEEP operator. *The American Statistician*, 33:3, 149–158.

Gower, J. C. 1966. Some distance properties of latent root and vector methods used in multivariate analysis. *Biometrika*, 53, 325–338.

Grambsch, P., and T. Therneau. 1994. Proportional hazards tests and diagnostics based on weighted residuals. *Biometrika*, 81, 515–526.

Grant, E. L., and R. S. Leavenworth. 1980. *Statistical quality control*, 5 ed. New York: McGraw-Hill.

Graubard, B. I., and E. L. Korn. Graubard. Hypothesis testing with complex survey data: The use of classical quadratic test statistics with particular reference to regression problems. *Journal of the American Statistical Association*, 88, 629–641.

Green, P. E. 1978. *Analyzing multivariate data*. Hinsdale, Ill.: The Dryden Press.

Green, D., and J. Swets. 1966. *Signal Detection Theory and Psychophysics*. New York: John Wiley & Sons.

Greenacre, M. J. 1984. *Theory and applications of correspondence analysis*. London: Academic Press.

Greenhouse, S. W., and S. Geisser. 1959. On methods in the analysis of profile   data. *Psychometrika*, 24:2, 95–111.

Greenland, S. 1994. Alternative Models for Ordinal Logistic Regression. *Statistics in Medicine* , 13, 1665–1677.

Griner, P. F., R. J. Mayewski, A. I. Mushlin, and P. Greenland. 1981. Selection and interpretation of diagnostic tests and procedures: Principles in applications. *Annals of Internal Medicine*, 94, 553–600.

Groenen, P. J. F., W. J. Heiser, and J. J. Meulman. 1999. Global optimization in least squares multidimensional scaling by distance smoothing. *Journal of Classification*, 16, 225–254.

Groenen, P. J. F., B. van Os, and J. J. Meulman. 2000. Optimal scaling by alternating length-constained nonnegative least squares, with application to distance-based analysis. *Psychometrika*, 65, 511–524.

Guttman, L. 1941. The quantification of a class of attributes: A theory and method of scale construction. In: *The Prediction of Personal Adjustment*, P. Horst, ed. New York: Social Science Research Council, 319–348.

Guttman, L. 1945. A basis for analyzing test-retest reliability. *Psychometrika*, 10:4, 255–282.

Haberman, S. 1974. *The Analysis of Frequency Data*. Chicago: University of Chicago Press.

Haberman, S. J. 1982. Analysis of dispersion of multinomial responses. *Journal of the American Statistical Association*, 77 , 568–580.

Haberman, S. J. 1977. Maximum likelihood estimates in exponential response models. *Annals of Statistics*, 5, 815–841.

Haberman, S. J. 1973. The analysis of residuals in cross-classified tables. *Biometrics*, 29, 205–220.

Haberman, S. J. 1978. *Analysis of qualitative data*. London: Academic Press.

Hanley, J. A., and B. J. McNeil. 1982. The Meaning and Use of the Area Under a Receiver Operating Characteristic (ROC) Curve. *Radiology*, 143:1, 29–36.

Hanley, J. A., and B. J. McNeil. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology*, 148, 839–843.

Hansen, M. H., and W. N. Hurwitz. 1943. On the theory of sampling from finite populations. *Annals of Mathematical Statistics*, 14, 333–362.

Hansen, M. H., W. N. Hurwitz, and W. G. Madow. 1953. *Sample Survey Methods and Theory, Volume II Theory*. New York: John Wiley & Sons.

Hanurav, T. V. 1967. Optimum Utilization of Auxiliary Information: PPS Sampling of Two Units from a Stratum. *Journal of the Royal Statistical Society, Series B*, 29, 374–391.

Hardin, J. W., and J. M. Hilbe. 2003. *Generalized Linear Models and Extension*. Station, TX: Stata Press.

Hardin, J. W., and J. M. Hilbe. 2001. *Generalized Estimating Equations*. Boca Raton, FL: Chapman & Hall/CRC.

Harman, H. H. 1976. *Modern Factor Analysis*, 3rd ed. Chicago: University of Chicago Press.

Harshman, R. A. 1970. *Foundations of the PARAFAC procedure: Models and conditions for an explanatory multi-model factor analysis*, Working Papers in Phonetics No. 16 ed. Los Angeles: University of California.

Harter, H. L. 1969. *Order statistics and their use in testing and estimation, Volumes 1 and 2*. Washington, D.C.: U.S. Government Printing Office: Aerospace Research Laboratories, United States Air Force.

Hartigan, J. A. 1975. *Clustering algorithms*. New York: John Wiley and Sons.

Hartzel, J., A. Agresti, and B. Caffo. 2001. Multinomial Logit Random Effects Models. *Statistical Modelling*, 1, 81–102.

Harvey, A. C. 1989. *Forecasting, structural time series models and the Kalman filter*. Cambridge: Cambridge University Press.

Hauck, W. W., and A. Donner. 1977. Wald’s Test as Applied to Hypotheses in Logit Analysis. *Journal of the American Statistical Association*, 72, 851–853.

Hauck, W. 1989. Odds ratio inference from stratified samples. *Commun. Statis.-Theory Meth.*, 18, 767–800.

Haykin, S. 1998. *Neural Networks: A Comprehensive Foundation*, 2nd ed. New York: Macmillan College Publishing.

Hays, W. L. 1973. *Statistics for the social sciences*. New York: Holt, Rinehart, and Winston.

Hedeker, D. 1999. Generalized Linear Mixed Models. In: *Encyclopedia of Statistics in Behavioral Science,* B. Everitt, and D. Howell, eds. London: Wiley, 729–738.

Heiser, W. J. 1981. *Unfolding analysis of proximity data*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J. 1985. *A general MDS initialization procedure using the SMACOF algorithm-model with constraints: Technical Report No. RR-85-23*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J., and J. De Leeuw. 1986. *SMACOF-I: Technical Report No. UG-86-02*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J., and I. Stoop. 1986. *Explicit SMACOF algorithms for individual differences scaling: Technical Report No. RR-86-14*. Leiden: Department of Data Theory, University of Leiden.

Heiser, W. J. 1987. Joint ordination of species and sites: The unfolding technique. In: *Developments in numerical ecology,* P. Legendre, and L. Legendre, eds. Berlin,Heidelberg: Springer-Verlag, 189–221.

Henderson, C. R. 1984. *Applications of Linear Models in Animal Breeding*. Guelph, Ontario: University of Guelph.

Hendrickson, A. E., and P. O. White. 1964. Promax: a quick method for rotation to oblique simple structure. *British Journal of Statistical Psychology*, 17, 65–70.

Hettmansperger, T. P., and S. J. Sheather. 1986. Confidence Interval Based on Interpolated Order Statistics. *Statistical Probability Letters*, 4, 75–79.

Hill, G. W. 1970. Algorithm 395: Student's t-distribution. *Communications of the ACM*, 13, 617–619.

Hill, G. W. 1970. Algorithm 396: Student's t-quantiles. *Communications of the ACM*, 13, 619–620.

Hill, I. D. 1973. Algorithm AS 66: The normal integral. *Applied Statistics*, 22, 424–424.

Hill, I. D., and A. C. Pike. 1967. Algorithm 299: Chi-squared integral. *Communications of the ACM*, 10, 243–244.

Hoaglin, D. C., F. Mosteller, and J. W. Tukey. 1983. *Understanding robust and exploratory data analysis*. New York: John Wiley and Sons.

Hoaglin, D. C., F. Mosteller, and J. W. Tukey. 1985. *Exploring data tables, trends, and shapes*. New York: John Wiley and Sons.

Hochberg, Y. 1974. Some Generalizations of the T-method in Simultaneous Inference. *Journal of Multivariate Analysis*, 4, 224–234.

Hochberg, Y., and A. C. Tamhane. 1987. *Multiple Comparison Procedures*. New York: John Wiley & Sons, Inc. .

Hollander, M., and D. A. Wolfe. 1999. *Nonparametric Statistical Methods, 2nd edition*. New York: John Wiley & Sons.

Holm, S. 1979. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6, 65–70.

Horst, P. 1963. *Matrix algebra for social scientists*. New York: Holt, Rinehart and Winston.

Horton, N. J., and S. R. Lipsitz. 1999. Review of Software to Fit Generalized Estimating Equation Regression Models. *The American Statistician*, 53, 160–169.

Horwitz, D. G., and D. J. Thompson. 1952. A generalization of sampling without replacement from a finite universe. *Journal of the American Statistical Association*, 47, 663–685.

Hosmer, D. W. J., and S. Lemeshow. 1981. Applied Logistic Regression Models. *Biometrics*, 34, 318–327.

Hosmer, D. W., and S. Lemeshow. 2000. *Applied Logistic Regression*, 2nd ed. New York: John Wiley and Sons.

Hsu, J. C. 1989. *Multiple Comparison Procedures*. : American Statistical Association Short Course.

Hsu, C., H. Huang, and T. Wong. 2000. Why Discretization Works for Naive Bayesian Classifiers. In: *Proceedings of the 17th International Conference on Machine Learning*, San Francisco: Morgan Kaufman, 399–406.

Huber, P. J. 1967. The Behavior of Maximum Likelihood Estimates under Nonstandard Conditions. In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability*, Berkeley, CA: University of California Press, 221–233.

Hurvich, C. M., and C. L. Tsai. 1989. Regression and Time Series Model Selection in Small Samples. *Biometrika* , 76, 297–307.

Huynh, H., and L. S. Feldt. 1970. Conditions under which mean square ratios in repeated measurements designs have exact F-distributions. *Journal of the American Statistical Association*, 65, 1582–1589.

Huynh, H., and L. S. Feldt. 1976. Estimation of the Box correction for degrees of freedom from sample data in the randomized block and split plot designs. *Journal of Educational Statistics*, 1, 69–82.

International Association of Assessing Officers, . 1990. *Property Appraisal and Assessment Administration*. International Association of Assessing Officers: Chicago, Illinois.

Israëls, A. 1987. *Eigenvalue techniques for qualitative data*. Leiden: DSWO Press.

James, G. S. 1951. The comparison of several groups of observations when the ratios of the population variances are unknown. *Biometrika*, 38, 324–329.

Jennrich, R. I., and P. F. Sampson. 1976. Newton-Raphson and related algorithms for maximum likelihood variance component estimation. *Technometrics*, 18, 11–17.

Jennrich, R. I., and P. F. Sampson. 1966. Rotation for simple loadings. *Psychometrika*, 31, 313–323.

Jöhnk, M. D. 1964. Erzeugung von Betaverteilten und Gammaverteilten Zufallszahlen. *Metrika*, 8, 5–15.

Johnson, N. L., S. Kotz, and A. W. Kemp. 1992. *Univariate Discrete Distributions*, 2 ed. New York: John Wiley.

Johnson, N. L., S. Kotz, and N. Balakrishnan. 1994. *Continuous Univariate Distributions*, 2 ed. New York: John Wiley.

Johnson, N. L., S. Kotz, and A. W. Kemp. 2005. *Univariate Discrete Distributions*, 3rd ed. Hoboken, New Jersey: John Wiley & Sons.

Jones, L. E., and F. W. Young. 1972. Structure of a social environment: longitudinal individual differences scaling of an intact group. *Journal of Personality and Social Psychology*, 24, 108–121.

Jöreskog, K. G. 1977. Factor analysis by least-square and maximum-likelihood method. In: *Statistical Methods for Digital Computers, volume 3,* K. Enslein, A. Ralston, and R. S. Wilf, eds. New York: John Wiley andSons.

Kaiser, H. F. 1963. Image analysis. In: *Problems in Measuring Change,* C. W. Harris, ed. Madison: Universityof Wisconsin Press.

Kalbfleisch, J. D., and R. L. Prentice. 2002. *The statistical analysis of failure time data*, 2 ed. New York: John Wiley & Sons, Inc.

Kass, G. 1980. An exploratory technique for investigating large quantities of categorical data. *Applied Statistics*, 29:2, 119–127.

Kaufman, L., and P. J. Rousseeuw. 1990. *Finding groups in data: An introduction to cluster analysis*. New York: John Wiley and Sons.

Kendall, M. G. 1955. *Rank correlation methods*. London: Charles Griffin.

Kennedy, W. J., and J. E. Gentle. 1980. *Statistical computing*. New York: Marcel Dekker.

Kinderman, A. J., and J. G. Ramage. 1976. Computer generation of normal random variables (Correction: 85: 212). *Journal of the American Statistical Association*, 71, 893–896.

Kish, L., and M. R. Frankel. 1974. Inference from complex samples. *Journal of the Royal Statistical Society B*, 36, 1–37.

Kish, L. 1965. *Survey Sampling*. New York: John Wiley and Sons.

Kish, L. 1995. Methods for Design Effects. *Journal of Official Statistics*, 11, 119–127.

Knuth, D. E. 1973. *The Art of Computer Programming, volume3: Sorting and Searching*. Reading, MA: Addison-Wesley.

Knuth, D. E. 1981. *The Art of Computer Programming, volume 2, p. 106*. Reading, MA: Addison-Wesley.

Koch, G. G., D. H. Freeman, and J. L. Freeman. 1975. Strategies in the multivariate analysis of data from complex surveys. *International Statistical Review*, 43, 59–78.

Kohavi, R., and D. Sommerfield. 1995. Feature subset selection using the wrapper model: Overfitting and dynamic search space topology. In: *The First International Conference on Knowledge Discovery and Data Mining*, Menlo Park,California: AAAI Press, 192–197.

Kohavi, R., B. Becker, and D. Sommerfield. 1997. Improving Simple Bayes. In: *Proceedings of the European Conference on Machine Learning*, , 78–87.

Kohn, R., and C. Ansley. 1986. Estimation, prediction, and interpolation for ARIMA models with missing data. *Journal of the American Statistical Association*, 81, 751–761.

Kohn, R., and C. Ansley. 1985. Efficient estimation and prediction in time series regression models. *Biometrika*, 72:3, 694–697.

Korn, E. L., and B. L. Graubard. 1990. Simultaneous testing of regression coefficients with complex survey data: Use of Bonferroni t statistics. *American Statistician*, 44, 270–276.

Kotz, S., and N. L. Johnson, eds. 1988. *Encyclopedia of statistical sciences*. New York: John Wiley & Sons, Inc.

Kotz, S., and J. Rene Van Dorp. 2004. *Beyond Beta, Other Continuous Families of Distributions with Bounded Support and Applications*. Singapore: World Scientific Press.

Kramer, C. Y. 1956. Extension of multiple range tests to group means with unequal numbers of replications. *Biometrics*, 12, 307–310.

Kristof, W. 1963. The statistical theory of stepped-up reliability coefficients when a test has been divided into several equivalent parts. *Psychometrika*, 28:3, 221–238.

Kristof, W. 1969. Estimation of true score and error variance for tests under various equivalence assumptions. *Psychometrika*, 34:4, 489–507.

Kroese, D. P., T. Taimre, and Z. I. Botev. 2011. *Handbook of Monte Carlo Methods*. Hoboken, New Jersey: John Wiley & Sons.

Kruskal, J. B. 1964. Nonmetric multidimensional scaling: A numerical method. *Psychometrika*, 29, 115–129.

Lane, P. W., and J. A. Nelder. 1982. Analysis of Covariance and Standardization as Instances of Prediction. *Biometrics*, 38, 613–621.

Lau, C. L. 1980. Algorithm AS 147: A simple series for the incomplete gamma integral. *Applied Statistics*, 29, 113–114.

Lawless, R. F. 1982. *Statistical models and methods for lifetime data*. New York: John Wiley & Sons, Inc..

Lawless, J. E. 1984. Negative Binomial and Mixed Poisson Regression. *The Canadian Journal of Statistics*, 15, 209–225.

Ledolter, J., and B. Abraham. 1984. Some comments on the initialization of exponential smoothing. *Journal of Forecasting*, 3, 79–84.

Lee, E., and M. Desu. 1972. A computer program for comparing k samples with right censored data. *Computer Programs in Biomedicine*, 2, 315–321.

Lehmann, E. L. 1975. *Nonparametrics: Statistical methods based on ranks*. San Francisco: Holden-Day.

Lehmann, E. L. 1985. *Nonparametrics: Statistical Methods Based on Ranks*. San Francisco: McGraw-Hill.

Lenth, R. V. 1987. Algorithm AS 226: Computing noncentral beta probabilities (Correction: 39: 311–312). *Applied Statistics*, 36, 241–244.

Lenth, R. V. 1989. Algorithm AS 243: Cumulative distribution function of the noncentral t distribution. *Applied Statistics*, 38, 185–189.

Levene, H. 1960. Robust tests for equality of variances. In: *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, I. Olkin, ed. Palo Alto, Calif.: Stanford University Press, 278–292.

Li, K. H., T. E. Raghunathan, and D. B. Rubin. 1991. Large-Sample Significance Levels from Multiply Imputed Data Using Moment-Based Statistics and an F Reference Distribution. *Journal of the American Statistical Association*, 86, 1065–1073.

Liang, K. Y., and S. L. Zeger. 1986. Longitudinal Data Analysis Using Generalized Linear Models. *Biometrika*, 73, 13–22.

Lilliefors, H. W. 1967. On the Kolmogorov-Smirnov tests for normality with mean and variance unknown. *Journal of the American Statistical Association*, 62, 399–402.

Lim, T. S., W. Y. Loh, and Y. S. Shih. 2000. A Comparison of Prediction Accuracy, Complexity, and Training Time of Thirty-three Old and New Classification Algorithms. *Machine Learning*, 40:3, 203–228.

Lin, D. Y. 2000. On fitting Cox's proportional hazards models to survey data. *Biometrika*, 87, 37–47.

Ling, R. E. 1978. A study of the accuracy of some approximations for t, chi-square, and F tail probabilities. *Journal of American Statistical Association*, 73, 274–283.

Link, C. L. 1984. Confidence intervals for the survival function using Cox's proportional hazards model with covariates. *Biometrics*, 40, 601–610.

Link, C. L. 1986. Confidence intervals for the survival function in the presence of covariates. *Biometrics*, 42, 219–220.

Lipsitz, S. H., K. Kim, and L. Zhao. 1994. Analysis of Repeated Categorical Data Using Generalized Estimating Equations. *Statistics in Medicine*, 13, 1149–1163.

Little, R. J. A. 1988. A test of missing completely at random for multivariate data with missing values. *Journal of the American Statistical Association*, 83, 1198–1202.

Little, R. J. A., and D. B. Rubin. 1987. *Statistical analysis with missing data*. New York: John Wiley & Sons, Inc. .

Liu, H., F. Hussain, C. L. Tan, and M. Dash. 2002. Discretization: An Enabling Technique. *Data Mining and Knowledge Discovery*, 6, 393–423.

Loh, W. Y., and Y. S. Shih. 1997. Split selection methods for classification trees. *Statistica Sinica*, 7, 815–840.

Loh, W. Y. 1987. Some Modifications of Levene's Test of Variance Homogeneity. *Journal of the Statistical Computation and Simulation*, 28, 213–226.

Long, J. S. 1997. *Regression Models for Categorical and Limited Dependent Variables*. New York: Advanced Quantitative Techniques in the Social Sciences Series.

Louise, T. A. 1982. Finding the observed information matrix when using the EM algorithm. *Journal of the Royal Statistical Society, B*, 44:2, 226–233.

Luce, R. D. 1959. *Individual Choice Behavior*. New York: John Wiley.

Lund, R. E. 1980. Algorithm AS 152: Cumulative hypergeometric probabilities. *Applied Statistics*, 29, 221–223.

Lund, R. E., and J. R. Lund. 1983. Algorithm AS 190: Probabilities and upper quantiles for the studentized range. , 32, 204–210.

Lund, R. E., and J. R. Lund. 1985. Correction to Algorithm AS 190. , 34, 104–.

MacCallum, R. C. 1977. Effects of conditionality on INDSCAL and ALSCAL weights. *Psychometrika*, 42, 297–305.

Macleod, A. J. 1989. Algorithm AS 245: A robust and reliable algorithm for the logarithm of the gamma function. *Applied Statistics*, 38, 397–402.

Magidson, J. 1995. Introducing a New Graphical Model for the Analysis of an Ordinal Categorical Response – Part I. *Journal of Targeting, Measurement and Analysis for Marketing*, 4:2, 133–148.

Majumder, K. L., and G. P. Bhattacharjee. 1973. Algorithm AS 63: The incomplete beta integral.. *Applied Statistics*, 22, 409–411.

Majumder, K. L., and G. P. Bhattacharjee. 1973. Algorithm AS 64: Inverse of the incomplete beta function ratio. *Applied Statistics*, 22, 412–414.

Makridakis, S. G., S. C. Wheelwright, and R. J. Hyndman. 1997. *Forecasting: Methods and applications*, 3rd ed. ed. New York: John Wiley and Sons.

Makridakis, S., S. C. Wheelwright, and V. E. McGee. 1983. *Forecasting: Methods and applications*. New York: John Wiley and Sons.

Mantel, N., and W. Haenszel. 1959. Statistical aspects of the analysis of data from retrospective studies of disease. *J. Natl. Cancer Inst.*, 22, 719–748.

Mardia, K. V. 1972. *Statistics of Directional Data*. New York: Academic Press.

Mardia, K. V., J. T. Kent, and J. M. Bibby. 1979. *Multivariate Analysis*. New York: Academic Press.

Marsaglia, G. 1962. Random variables and computers. In: *Information theory statistical decision functions random processes: Transactions of the third Prague conference*, J. Kozesnik, ed. Prague, Czechoslovak: Czechoslovak Academy of Science, 499–510.

Marsaglia, G., and J. Marsaglia. 2004. Evaluating the Anderson-Darling Distribution. *Journal of Statistical Software*, 9:2, .

Matsumoto, M., and T. Nishimura. 1998. Mersenne Twister, A 623–dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. on Modeling and Computer Simulation*, 8:1, 3–30.

Matsumoto, M., and Y. Kurita. 1994. Twisted GFSR generators II. *ACM Trans. on Modeling and Computer Simulation*, 4:3, 254–266.

Mauchly, J. W. 1940. Significance test for sphericity of a normal n-variate distribution. *Annals of Mathematical Statistics*, 11, 204–209.

Max, J. 1960. Quantizing for minimum distortion. *Proceedings IEEE (Information Theory)*, 6, 7–12.

McCullagh, P. 1980. Regression Models for Ordinal Data. *Journal of the Royal Statistical Society B*, 42:2, 109–142.

McCullagh, P. 1983. Quasi-Likelihood Functions. *Annals of Statistics*, 11, 59–67.

McCullagh, P., and J. A. Nelder. 1989. *Generalized Linear Models*, 2nd ed. London: Chapman & Hall.

McCulloch, C. E., and S. R. Searle. 2001. *Generalized, Linear, and Mixed Models*. New York: John Wiley and Sons.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In: *Frontiers in Economics,* P. Zarembka, ed. New York: AcademicPress.

McGraw, K. O., and S. P. Wong. 1996. Forming inferences about some intraclass correlation coefficients. *Psychological Methods*, 1:1, 30–46.

McLean, R. A., and W. L. Sanders. 1988. Approximating Degrees of Freedom for Standard Errors in Mixed Linear Models. In: *Proceedings of the Statistical Computing Section, American Statistical Association,* New Orleans: American StatisticalAssociation, 50–59.

McNemar, Q. 1947. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12, 153–157.

Melard, G. 1984. A fast algorithm for the exact likelihood of autoregressive-moving average models. *Applied Statistics*, 33:1, 104–119.

Metz, C. E. 1978. Basic principles of ROC analysis. *Seminars in Nuclear Medicine*, 8, 283–298.

Metz, C. E. 1993. *ROC Software Package*. : .

Metz, C. E., and H. B. Kronman. 1980. Statistical significance tests for binormal ROC curves. *Journal of Mathematical Psychology*, 22, 218–243.

Meulman, J. J., and W. J. Heiser. 1984. Constrained Multidimensional Scaling: more Directions than Dimensions. In: *COMPSTAT 1984,* T. Havranek, ed. Wien: Physica Verlag, 137–142.

Meulman, J. J. 1982. *Homogeneity analysis of incomplete data*. Leiden: DSWO Press.

Michael, J., W. Schucany, and R. Haas. 1976. Generating random variates using transformation with multiple roots. *American Statistician*, 30, 88–90.

Miller, R. G. 1980. *Simultaneous Statistical Inference*, 2 ed. New York: Springer-Verlag.

Miller, R. G. J. 1966. *Simultaneous statistical inference*. New York: McGraw-Hill.

Miller, M. E., C. S. Davis, and J. R. Landis. 1993. The Analysis of Longitudinal Polytomous Data: Generalized Estimating Equations and Connections with Weighted Least Squares. *Biometrics*, 49, 1033–1044.

Milliken, G., and D. Johnson. 1992. *Analysis of Messy Data: Volume 1. Designed Experiments*. New York: Chapman & Hall.

Monro, D. M. 1975. Algorithm AS 83: Complex discrete fast Fourier transform. *Applied Statistics*, 24, 153–160.

Monro, D. M., and J. L. Branch. 1977. Algorithm AS 117: The Chirp discrete Fourier transform of general length. *Applied Statistics*, 26, 351–361.

Moré, J. J. 1977. The Levenberg-Marquardt algorithm: implementation and theory in numerical analysis. In: *Lecture Notes in Mathematics* , G. A. Watson, ed. Berlin: Springer-Verlag, 105–116.

Morf, M., G. S. Sidhu, and T. Kailath. 1974. Some new algorithms for recursive estimation in constant, linear, discrete-time systems. *IEEE Transactions on Automatic Control*, AC-19:4, 315–323.

Morrison, D. F. 1976. *Multivariate statistical methods*. New York: McGraw-Hill.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Approximations for the doubly noncentral F-distribution. *Communications in Statistics, Part A*, 5, 49–53.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Some Approximations for the noncentral F-distribution. *Technometrics*, 18, 351–358.

Mudholkar, G. S., Y. P. Chaubrey, and C. Lin. 1976. Some Approximations for the noncentral F-distribution. *Technometrics*, 18, 351–358.

Mudholkar, G., and R. Natarajan. 1999. Approximations for the inverse Gaussian probabilities and percentiles. *Communications in Statistics - Simulation and Computation*, 28:4, 1051–1071.

Muller, K. E., and B. L. Peterson. 1984. Practical methods for computing power in testing the multivariate general linear hypothesis. *Computational Statistics and Data Analysis*, 2, 143–158.

Murthy, M. N. 1957. Ordered and Unordered Estimators in Sampling Without Replacement. *Sankhya*, 18, 379–390.

Nagelkerke, N. J. D. 1991. A note on the general definition of the coefficient of determination. *Biometrika*, 78:3, 691–692.

Narula, S. C., and M. M. Desu. 1981. Computation of probability and noncentrality parameter of a noncentral chi-square distribution. *Applied Statistics*, 30, 349–352.

Natarajan, R., and E. Pednault. 2001. Using Simulated Pseudo Data to Speed Up Statistical Predictive Modeling from Massive Data Sets. In: *SIAM First International Conference on Data Mining*, .

Nelder, J. A., and R. W. M. Wedderburn. 1972. Generalized Linear Models. *Journal of the Royal Statistical Society Series A*, 135, 370–384.

Nishisato, S. 1980. *Analysis of categorical data: Dual scaling and its applications*. Toronto: University of Toronto Press.

Novick, M. R., and C. Lewis. 1967. Coefficient alpha and the reliability of composite measurements. *Psychometrika*, 32:1, 1–13.

Null, C. H., and W. S. Sarle. 1982. Multidimensional scaling by least squares. *Proceedings of the 7th Annual SAS User's Group International*, , – .

Odeh, R. E., and D. B. Owen. 1980. *Tables for Normal Tolerance Limits, Sampling Plans, and Screening*. New York: Marcel Dekker.

Orchard, T., and M. A. Woodbury. 1972. . In: *Proceedings of the Sixth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1,* Berkeley: Universityof California Press, 697–715.

Pan, W. 2001. Akaike's Information Criterion in Generalized Estimating Equations. *Biometrics*, 57, 120–125.

Patel, J. K., and C. B. Read. 1982. *Handbook of the normal distribution*. New York: Marcel Dekker.

Pearlman, J. G. 1980. An algorithm for the exact likelihood of a high-order autoregressive-moving average process. *Biometrika,* 67:1, 232–233.

Pena, D., G. C. Tiao, and R. S. Tsay, eds. 2001. *A course in time series analysis*. New York: John Wiley and Sons.

Peterson, B., and F. Harrell. 1990. Partial proportional odds models for ordinal response variables. *Applied Statistics*, 39, 205–217.

Pierce, D. A., and D. W. Schafer. 1986. Residuals in generalized linear models. *Journal of the American Statistical Association*, 81, 977–986.

Pike, M. C., and I. D. Hill. 1966. Algorithm 291: Logarithm of gamma function. *Communications of the ACM*, 9, 684–684.

Pillai, K. C. S. 1967. Upper percentage points of the largest root of a matrix in multivariate analysis. *Biometrika*, 54, 189–194.

Plackett, R. L., and J. P. Burman. 1946. The design of optimum multifactorial experiments. *Biometrika*, 33, 305–325.

Pratt, J. W. 1987. Dividing the indivisible: Using simple symmetry to partition variance explained. In: *Proceedings of the Second International Conference in Statistics,* T. Pukkila, and S. Puntanen, eds. Tampere, Finland: Universityof Tampere, 245–260.

Pregibon, D. 1981. Logistic Regression Diagnostics. *Annals of Statistics*, 9, 705–724.

Quenouville, M. H. 1949. Approximate tests of correlation in time series. *Journal of the Royal Statistical Society, Series B*, 11, 68–68.

Raghunathan, T. E., J. M. Lepkowski , J. van Hoewyk , and P. Solenberger . 2001. A multivariate technique for multiply imputing missing values using a sequence of regression models. *Survey Methodology* , 27, 85–95.

Ramsay, J. O. 1989. Monotone regression splines in action. *Statistical Science*, 4, 425–441.

Ramsey, P. H. 1978. Power Differences Between Pairwise Multiple Comparisons. *Journal of the American Statistical Association*, 73, 479–485.

Rao, C. R. 1973. *Linear statistical inference and its applications*, 2nd ed. New York: John Wiley and Sons.

Rao, C. R. 1980. Matrix approximations and reduction of dimensionality in multivariate statistical analysis. In: *Multivariate Analysis, Vol. 5,* P. R. Krishnaiah, ed. Amsterdam: North-Holland, 3–22.

Rao, J. N. K., and A. J. Scott. 1981. The analysis of categorical data from complex sample surveys: chi-squared tests for goodness of fit and independence in two-way tables. *Journal of the American Statistical Association*, 76, 221–230.

Rao, J. N. K., and A. J. Scott. 1984. On chi-squared tests for multiway contingency tables with cell proportions estimated from survey data. *Annals of Statistics*, 12, 46–60.

Rao, J. N. K., and D. R. Thomas. 2003. Analysis of categorical response data from complex surveys: an Appraisal and update. In: *Analysis of Survey Data*, R. Chambers, and C. Skinner, eds. New York: John Wiley & Sons.

Rao, C. R. 1951. An asymptotic expansion of the distribution of Wilks' criterion. *Bulletin of the International Statistical Institute*, 33:2, 177–180.

Ridout, M. S., and J. M. Cobby. 1989. A remark on algorithm AS 178. *Applied Statistics*, 38, 420–422.

Ripley, B. D. 1996. *Pattern Recognition and Neural Networks*. Cambridge: Cambridge University Press.

Robins, J., N. Breslow, and S. Greenland. 1986. Estimators of the Mantel-Haenszel variance consistent in both sparse data and large-strata limiting models. *Biometrics*, 42, 311–323.

Robson, D. S. 1959. A simple method for construction of orthogonal polynomials when the independent variable is unequally spaced. *Biometrics*, 15, 187–191.

Romesburg, H. C. 1984. *Cluster analysis for researchers*. Belmont, Calif.: Lifetime Learning Publications.

Rosipal, R., and N. Krämer. 2006. Overview and Recent Advances in Partial Least Squares. In: *Subspace, Latent Structure and Feature Selection Techniques*, C. Saunders, M. Grobelnik, S. Gunn, and J. Shawe-Taylor, eds. Berlin: Springer-Verlag, 34–51.

Royston, J. P. 1987. AS R69: A remark on Algorithm AS 190. *Applied Statistics*, 36, 119.

Rubin, R. B. 1987. *Multiple Imputation for Nonresponse in Surveys*. New York: John Wiley and Sons, Inc..

Rummel, R. J. 1970. *Applied factor analysis*. Evanston: Ill.: Northwestern University Press.

Ryan, T. A. 1959. Multiple Comparisons in Psychological Research. *Psychological Bulletin*, 56, 26–47.

Ryan, T. A. 1960. Significance Tests for Multiple Comparison of Proportions, Variances, and Other Statistics. *Psychological Bulletin*, 57, 318–328.

Sampford, M. R. 1967. On Sampling without Replacement with Unequal Probabilities of Selection. *Biometrika*, 54, 499–513.

Santner, T. J., and E. D. Duffy. 1986. A Note on A. Albert and J.A. Anderson's Conditions for the Existence of Maximum Likelihood Estimates in Logistic Regression Models. *Biometrika*, 73, 755–758.

Särndal, C., B. Swensson, and J. Wretman. 1992. *Model Assisted Survey Sampling*. New York: Springer-Verlag.

SAS Institute, Inc., . 1990. *SAS/STAT User's Guide, Version 6*, 4 ed. Cary, NC: SAS Institute Inc..

Satterthwaite, F. E. 1946. An approximate distribution of estimates of variance components. *Biometrics Bulletin*, 2, 110–114.

Schafer, J. L. 1997. *Analysis of Incomplete Multivariate Data*. London: Chapman and Hall.

Schatzoff, M., R. Tsao, and S. Fienberg. 1968. Efficient computing of all possible regressions. *Technometrics*, 10, 769–779.

Scheffe, H. 1953. A method for judging all contrasts in the analysis of variance. *Biometrika*, 40, 87–104.

Scheffe, H. 1959. *The Analysis of Variance*. New York: John Wiley & Sons, Inc..

Schervish, M. J. 1984. Algorithm AS 195: Multivariate normal probabilities with error bound. *Applied Statistics*, 33, 81–94.

Schiffman, S. S., M. L. Reynolds, and F. W. Young. 1981. *Introduction to multidimensional scaling: theory, methods and applications*. New York: Academic Press.

Schoonjans, F. 1993–1998. *MedCalc Version 4.20.011*. : .

Schrage, L. 1979. A more portable Fortran random number generator. *ACM Transactions on Mathematical Software*, 5:2, 132–132.

Schwarz, G. 1978. Estimating the Dimension of a Model. *Annals of Statistics*, 6, 461–464.

Searle, R. S. 1987. *Linear Models for Unbalanced Data*. New York: Wiley.

Searle, S. R., F. M. Speed, and G. A. Milliken. 1980. Population marginal means in the linear model: an alternative to least squares means. *The American Statistician*, 34, 216–221.

Searle, S. R. 1971. *Linear Models*. New York: John Wiley & Sons, Inc.

Searle, S. R. 1982. *Matrix algebra useful for statistics*. New York: John Wiley & Sons, Inc..

Searle, S. R., G. Casella, and C. E. McCulloch. 1992. *Variance Components*. New York: John Wiley and Sons.

Searle, S. R. 1966. *Matrix algebra for the biological sciences*. New York: John Wiley& Sons, Inc.

Seber, G. A. F. 1984. *Multivariate observations*. New York: John Wiley & Sons, Inc.

Sen, A. R. 1953. On the estimate of the variance in sampling with varying probabilities. *Journal of the Indian Society of Agricultural Statistics*, 5, 55–77.

Shah, B. V., M. M. Holt, and R. E. Folsom. 1977. Inference about regression models from sample survey data. *Bulletin of the International Statistical Institute*, 67:3, 43–57.

Shao, J., and D. Tu. 1995. *The Jackknife and Bootstrap*. New York: Springer.

Shapiro, S. S., and M. B. Wilk. 1965. An analysis of variance test for normality. *Biometrika*, 52:3, 591–599.

Shea, B. L. 1988. Algorithm AS 239: Chi-squared and incomplete gamma integral. *Applied Statistics*, 37, 466–473.

Shea, B. L. 1989. AS R77: A remark on algorithm AS 152. *Applied Statistics*, 38, 199–204.

Sheskin, D. J. 2007. *Handbook of Parametric and Nonparametric Statistical Procedures, 4th edition*. : Chapman & Hall/CRC.

Shrout, P. E., and J. L. Fleiss. 1979. Intraclass correlations: Uses in assessing reliability. *Psychological Bulletin*, 86, 420–428.

Sidak, Z. 1967. Rectangular confidence regions for the means of multivariate normal distributions. *Journal of the American Statistical Association*, 62, 626–633.

Siegel, S. 1956. *Nonparametric statistics for the behavioral sciences*. New York: McGraw-Hill.

Singleton, R. C. 1969. Algorithm 347: Efficient sorting in minimal storage. *Communications of the ACM*, 12, 185–187.

Skrondal, A., and S. Rabe-Hesketh. 2004. *Generalized Latent Variable Modeling: Multilevel, Longitudinal, and Structural Equation Models*. Boca Raton, FL: Chapman & Hall/CRC.

Smirnov, N. V. 1948. Table for estimating the goodness of fit of empirical distributions. *Annals of the Mathematical Statistics*, 19, 279–281.

Smyth, G. K., and B. Jorgensen. 2002. Fitting Tweedie's Compound Poisson Model to Insurance Claims Data: Dispersion Modelling. *ASTIN Bulletin*, 32, 143–157.

Snedecor, G. W., and W. G. Cochran. 1980. *Statistical Methods*, 7th ed. Ames, Iowa: Iowa University Press.

Snedecor, G. W., and W. G. Cochran. 1967. *Statistical methods*. Ames, Iowa: Iowa State University Press.

Sobol, I. M. 2001. Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates. *Mathematics and Computers in Simulation*, 55, 271–280.

Somes, G. W., and K. F. O'Brien. 1985. *Mantel-Haenszel statistic. In Encyclopedia of Statistical Sciences, Vol. 5 (S. Kotz and N. L. Johnson, eds.) 214–217*. New York: John Wiley.

Speed, M. F. 1976. Response curves in the one way classification with unequal numbers of observations per cell. In: *Proceedings of the Statistical Computing Section*, Alexandria, VA: AmericanStatistical Association, 270–272.

Spicer, C. C. 1972. Algorithm AS 52: Calculation of power sums of deviations about the mean. *Applied Statistics*, 21, 226–227.

Stewart, G. W. 1973. *Introduction to matrix computations*. New York: Academic Press.

Stoop, I., W. J. Heiser, and J. De Leeuw. 1981. *How to use SMACOF-IA*. Leiden: Department of Data Theory.

Stoop, I., and J. De Leeuw. 1982. *How to use SMACOF-IB*. Leiden: Department of Data Theory.

Storer, B. E., and J. Crowley. 1985. A diagnostic for Cox regression and general conditional likelihoods. *Journal of the American Statistical Association*, 80, 139–147.

Tadikamalla, P. R. 1978. Computer generation of gamma random variables. *Communications of the ACM*, 21, 419–422.

Takane, Y., F. W. Young, and J. de Leeuw. 1977. Nonmetric individual differences multidimensional scaling: An alternating least squares method with optimal scaling features. *Psychometrika*, 42, 7–67.

Tamhane, A. C. 1977. Multiple Comparisons in Model I One-Way ANOVA with Unequal Variances. *Communications in Statistics*, 6, 15–32.

Tamhane, A. C. 1979. A Comparison of Procedures for Multiple Comparisons of Means with Unequal Variances. *Journal of the American Statistical Association*, 74, 471–480.

Tan, P., M. Steinbach, and V. Kumar. 2006. *Introduction to Data Mining*. : Addison-Wesley.

Tao, K. K. 1993. A closer look at the radial basis function (RBF) networks. In: *Conference Record of the Twenty-Seventh Asilomar Conference on Signals, Systems, and Computers*, A. Singh, ed. Los Alamitos, Calif.: IEEE Comput. Soc. Press, 401–405.

Tarone, R. 1975. Tests for trend in life table analysis. *Biometrika*, 62, 679–682.

Tarone, R., and J. Ware. 1977. On distribution free tests for equality of survival distributions. *Biometrika*, 64, 156–160.

Tarone, R. E. 1985. On heterogeneity tests based on efficient scores. *Biometrika*, 72, 91–95.

Tatsuoka, M. M. 1971. *Multivariate analysis*. New York: John Wiley & Sons, Inc. .

Taylor, G. A. R. 1970. Algorithm AS 27: The integral of Student's t-distribution. *Applied Statistics*, 19, 113–114.

Theil, H. 1953. *Repeated least square applied to complete equation systems*. Netherlands: The Hague: Central Planning Bureau.

Theil, H. 1953. *Estimation and simultaneous correlation in complete equation systems*. Netherlands: The Hague: Central Planning Bureau.

Therneau, T., and P. Grambsch. 2000. *Modeling Survival Data: Extending the Cox Model*. New York: Springer.

Thomas, D. R., and J. N. K. Rao. 1987. Small-sample comparisons of level and power for simple goodness-of-fit statistics under cluster sampling. *Journal of the American Statistical Association*, 82, 630–636.

Timm, N. H. 1975. *Multivariate statistics: With applications in education and psychology*. Monterey, California: Brooks/Cole.

Torgerson, W. S. 1952. Multidimensional scaling: I. Theory and method. *Psychometrika*, 17, 401–419.

Torgerson, W. S. 1958. *Theory and methods of scaling*. New York: Wiley.

Tucker, L. R. 1972. Relations between multidimensional scaling and three mode factor analysis. *Psychometrika*, 37, 3–28.

Tuerlinckx, F., F. Rijmen, G. Molenberghs, G. Verbeke, D. Briggs, W. Van den Noortgate, M. Meulders, and P. De Boeck. 2004. Estimation and Software. In: *Explanatory Item Response Models: A Generalized Linear and Nonlinear Approach*, P. De Boeck, and M. Wilson, eds. New York: Springer-Verlag, 343–373.

Tukey, J. W. 1977. *Exploratory data analysis*. Reading, MA: Addison-Wesley.

Tukey, J. W. 1962. The future of data analysis. *Annals of Mathematical Statistics*, 33:22, 1–67.

Uykan, Z., C. Guzelis, M. E. Celebi, and H. N. Koivo. 2000. Analysis of input-output clustering for determining centers of RBFN. *IEEE Transactions on Neural Networks*, 11, 851–858.

Van Buuren, S. 2007. Multiple imputation of discrete and continuous data by fully conditional specification. *Statistical Methods in Medical Research*, 16, 219–242.

Van Buuren, S., H. C. Boshuizen, and D. L. Knook. 1999. Multiple imputation of missing blood pressure covariates in survival analysis. *Statistics in Medicine*, 18, 681–694.

Van de Geer, J. P. 1987. *Algebra and geometry of OVERALS: Internal Report RR-87–13*. Leiden: Department of Data Theory, University of Leiden.

Van der Burg, E., J. De Leeuw, and R. Verdegaal. 1984. *Non-linear canonical correlation analysis: Internal Report RR-84–12*. Leiden: Department of Data Theory, University of Leiden.

Van der Burg, E. 1988. *Nonlinear canonical correlation and some related techniques*. Leiden: DSWO Press.

Van der Burg, E., J. De Leeuw, and R. Verdegaal. 1988. Homogeneity analysis with k sets of variables: An alternating least squares method with optimal scaling features. *Psychometrika*, 53, 177–197.

Van der Kooij, A. J., J. J. Meulman, and W. J. Heiser. 2006. Local Minima in Categorical Multiple Regression. *Computational Statistics and Data Analysis*, 50, 446–462.

Van der Kooij, A. J. 2007. *Prediction Accuracy and Stability of Regression with Optimal Scaling Transformations (Thesis)*. : Leiden University.

Van Rijckevorsel, J., and J. De Leeuw. 1979. *An outline of PRINCALS: Internal Report RB 002–'79*. Leiden: Department of Data Theory, University of Leiden.

Velleman, P. F., and D. C. Hoaglin. 1981. *Applications, basics, and computing of exploratory data analysis*. Boston, Mass.: Duxbury Press.

Velleman, P. F., and R. E. Welsch. 1981. Efficient computing of regression diagnostics. *American Statistician*, 35, 234–242.

Velleman, P. F. 1980. Definition and comparison of robust nonlinear data smoothing algorithms. *Journal of the American Statistical Association*, 75, 609–615.

Verdegaal, R. 1986. *OVERALS: Internal Report UG-86–01*. Leiden: Department of Data Theory, University of Leiden.

Vijayan, K. 1968. An Exact PPS Sampling Scheme: Generalization of a Method of Hanurav. *Journal of the Royal Statistical Society, Series B*, 54, 556–566.

Von Neumann, J. 1951. Various techniques used in connection with random digits. *National Bureau of Standards Applied Mathematics*, 12, 36–38.

Waller, R. A., and D. B. Duncan. 1969. A Bayes Rule for the Symmetric Multiple Comparison Problem. *Journal of the American Statistical Association*, 64, 1484–1499.

Waller, R. A., and D. B. Duncan. 1972. . *Journal of the American Statistical Association*, 67, 253–255.

Waller, R. A., and K. E. Kemp. 1975. Computations of Bayesian t-value for Multiple Comparison. *Journal of statistical computation and simulation*, 4, 169–172.

Watts, D. L. 1991. Correction: Computer Selection of Size-Biased Samples. *The American Statistician*, 45:2, 172–172.

Welch, B. L. 1951. On the Comparison of Several Mean Values: An Alternative Approach. *Biometrika*, 38, 330–336.

Welch, B. L. 1947. The generalization of Student's problem when several different population variances are involved. *Biometrika*, 34, 28–35.

Welsch, R. E. 1977. Stepwise Multiple Comparison Procedures. *Journal of the American Statistical Association*, 72, 566–575.

White, H. 1980. A Heteroskedasticity-Consistent Covariance Matrix Estimator and a Direct Test for Heteroskedasticity. *Econometrica*, 48, 817–836.

Wichura, M. J. 1988. Algorithm AS 241: The percentage points of the normal distribution. *Applied Statistics*, 37, 477–484.

Wilkinson, J. H. 1965. *The algebraic eigenvalue problem*. Oxford: Clarendon Press.

Wilkinson, J. H., and C. Reinsch. 1971. Linear Algebra. In: *Handbook for Automatic Computation, Volume II,* J. H. Wilkinson, and C. Reinsch, eds. New York: Springer-Verlag.

Williams, D. A. 1987. Generalized Linear Models Diagnostics Using the Deviance and Single Case Deletions. *Applied Statistics*, 36, 181–191.

Williams, D. A. 1982. Extra-Binomial Variation in Logistic Linear Models. *Applied Statistics*, 31, 144–148.

Winer, B. J. 1971. *Statistical principles in experimental design*, 2nd ed. New York: McGraw-Hill.

Wolfinger, R., R. Tobias, and J. Sall. 1994. Computing Gaussian likelihoods and their derivatives for general linear mixed models. *SIAM Journal on Scientific Computing*, 15:6, 1294–1310.

Wolfinger, R., and M. O’Connell. 1993. Generalized Linear Mixed Models: A Pseudo-Likelihood Approach. *Journal of Statistical Computation and Simulation*, 4, 233–243.

Wolkowicz, H., and G. P. H. Styan. 1980. Bounds for eigenvalues using traces. *Linear algebra and its applications*, 29, 471–506.

Wolter, K. M. 1985. *Introduction to variance estimation*. Berlin: Springer-Verlag.

Woodruff, R. S. 1971. A Simple Method for Approximating the Variance of a Complicated Estimate. *Journal of the American Statistical Association*, 66, 411–414.

Woods, K., and K. W. Bowyer. 1997. Generating ROC curves for artificial neural networks. *IEEE Transactions on Medical Imaging*, 16, 329–337.

Wright, S. P. 1992. Adjusted P-values for simultaneous inference. *Biometrics*, 48, 1005–1013.

Yates, F., and P. M. Grundy. 1953. Selection without replacement from within strata with probability proportional to size. *Journal of the Royal Statistical Society Series B*, 15, 253–261.

Young, J. C., and C. E. Minder. 1974. Algorithm AS 76: An integral useful in calculating noncentral t and bivariate normal probabilities. *Applied Statistics*, 23, 455–457.

Young, F. W., and R. Lewyckyj. 1979. *ALSCAL-4 user’s guide*. Carrboro: N.C.: Data Analysis and Theory Associates.

Young, F. W., Y. Takane, and R. Lewyckyj. 1978. ALSCAL: A nonmetric multidimensional scaling program with several different options. *Behavioral Research Methods and Instrumentation*, 10, 451–453.

Young, F. W., D. V. Easterling, and B. N. Forsyth. 1984. *The general Euclidean model for scaling three mode dissimilarities: theory and application. In: Research Methods for Multi-mode Data Analysis in the Behavioral Sciences, H. G. Law, G. W. Snyder, Jr., J. Hattie, and R. P. McDonald, eds*. New York: Praeger.

Young, E. A., and E. W. Cramer. 1971. Some results relevant to choice of sum and sum-of-product algorithms. *Technometrics*, 13, 657–665.

Zeger, S. L., and K. Y. Liang. 1986. Longitudinal Data Analysis for Discrete and Continuous Outcomes. *Biometrics*, 42, 121–130.

Zhang, J., and S. D. Hoffman. 1993. Discrete-Choice Logit Models. Testing the IIA Property. *Sociological Methods and Research*, 22:2, 193–213.

Zhang, T., R. Ramakrishnon, and M. Livny. 1996. BIRCH: An efficient data clustering method for very large databases. In: *Proceedings of the ACM SIGMOD Conference on Management of Data*, Montreal, Canada: ACM, 103–114.

Zhang, D. 2005. "Analysis of Survival Data, lecture notes, Chapter 10." Available at http://www4.stat.ncsu.edu/%7Edzhang2/st745/chap10.pdf.

Zweig, M. H., and G. Campbell. 1993. Receiver Operating Characteristic (ROC) Plots: A Fundamental Evaluation Tool in Clinical Medicine. *Clinical Chemistry*, 39:4, 561–577.