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:

$$L(t) = \alpha Y(t) + (1-\alpha) L(t-1)$$

$$\hat{Y}_t(k) = L(t)$$

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:

$$L(t) = \alpha Y(t) + (1-\alpha) L(t-1)$$

$$T(t) = \alpha(L(t) - L(t-1)) + (1-\alpha) T(t-1)$$

$$\hat{Y}_t(k) = L(t) + \left((k-1) + \alpha^{-1}\right) T(t)$$

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:

$$L(t) = \alpha Y(t) + (1-\alpha)(L(t-1) + T(t-1))$$

$$T(t) = \gamma(L(t) - L(t-1)) + (1-\gamma) T(t-1)$$

$$\hat{Y}_t(k) = L(t) + kT(t)$$

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:

$$L(t) = \alpha Y(t) + (1-\alpha)(L(t-1) + \phi T(t-1))$$

$$T(t) = \gamma(L(t) - L(t-1)) + (1-\gamma) \phi T(t-1)$$

$$\hat{Y}_t(k) = L(t) + \sum_{i=1}^{k} \phi^i T(t)$$

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:

$$L(t) = \alpha(Y(t) - S(t-s)) + (1-\alpha)L(t-1)$$

$$S(t) = \delta(Y(t) - L(t)) + (1-\delta)S(t-s)$$

$$\hat{Y}_t(k) = L(t) + S(t+k-s)$$

It is functionally equivalent to an ARIMA(0,1,(1,s,s+1))(0,1,0) with restrictions among MA parameters.

Winters's Additive Exponential Smoothing

Winter's additive exponential smoothing has level, trend and season parameters and can be described by the following equations:

$$L(t) = \alpha(Y(t) - S(t-s)) + (1-\alpha)(L(t-1) + T(t-1))$$

$$T(t) = \gamma(L(t) - L(t-1)) + (1-\gamma)T(t-1)$$

$$S(t) = \delta(Y(t) - L(t)) + (1-\delta)S(t-s)$$

$$\hat{Y}_t(k) = L(t) + kT(t) + S(t+k-s)$$

It is functionally equivalent to an ARIMA(0,1,s+1)(0,1,0) with restrictions among MA parameters.

Winters's Multiplicative Exponential Smoothing

Winter's multiplicative exponential smoothing has level, trend and season parameters and can be described by the following equations:

$$L(t) = \alpha(Y(t)/S(t-s)) + (1-\alpha)(L(t-1) + T(t-1))$$

$$T(t) = \gamma(L(t) - L(t-1)) + (1-\gamma)T(t-1)$$

$$S(t) = \delta(Y(t)/L(t)) + (1-\delta)S(t-s)$$

$$\hat{Y}_t(k) = (L(t) + kT(t))S(t+k-s)$$

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 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 $S = (\mu + \beta) / (\Sigma \mu_i + \beta_i)$.

ARIMA and Transfer Function Models

The following notation is specific to ARIMA/TF models:

a_t (t = 1, 2, ... , 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, $\phi_p(B) = 1 - \varphi_1 B - \varphi_2 B^2 - ... - \varphi_p B^p$
$\theta_q(B)$	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 $Ba_t = a_{t-1}$
$Z\sigma_t^2$	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, $X_{1t}, X_{2t}, ..., X_{kt}$ 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:

$$Z_t = f(Y_t),$$

$$\Delta Z_t = \mu + \sum_{i=1}^{k} \frac{Num_i}{Den_i} \Delta_i f_i(X_{it}) + \frac{MA}{AR} a_t.$$

The univariate ARIMA model simply drops the predictors from the TF model; thus, it has the following form:

$$\Delta Z_t = \mu + \tfrac{MA}{AR} a_t$$

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.

- 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

  $$N_t = \Delta Z_t - \mu - \sum_{i=1}^{k} \frac{Num_i}{Den_i} \Delta_i X_{it}$$

  is assumed to be a mean zero, stationary ARMA process.

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

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

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

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

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

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.

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):

$$Y(t) = \mu(t) + wL_O(B)\,I_T(t) + \tfrac{\theta(B)}{\Delta\varphi(B)}a(t)$$

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:

$$Y(t) = \mu(t) + \sum_{k=1}^{M} w_k L_{O_k}(B)\,I_{T_{D,k}}(t) + \frac{\theta(B)}{\Delta\varphi(B)}a(t)$$

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:

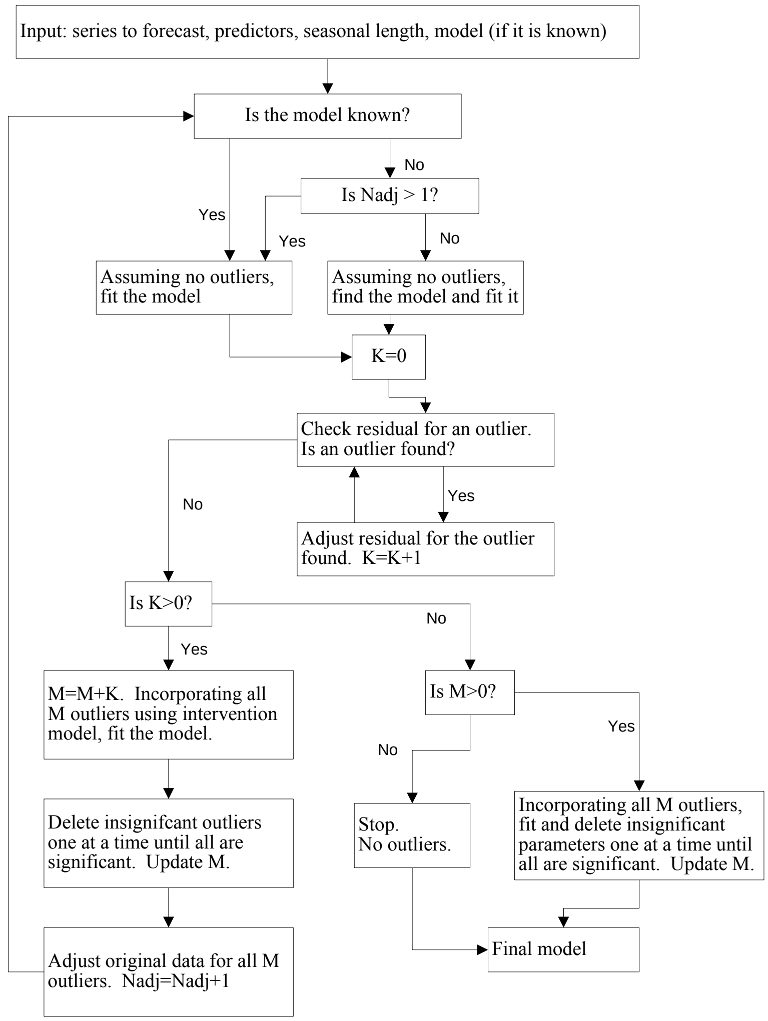
$$\chi^2(T) = \frac{\mathbf{w}^{'}(T)(X_T X_T)\mathbf{w}(T)}{\sigma^2}$$

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 1-1



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

$$\text{Normalized } BIC = \ln(MSE) + k\frac{\ln(n)}{n}$$

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).

$$d = \begin{cases} 2 & \text{Brown,Holt} \\ 1 & \text{other} \end{cases}, D = \begin{cases} 0 & s = 1 \\ 1 & s > 1 \end{cases}$$

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

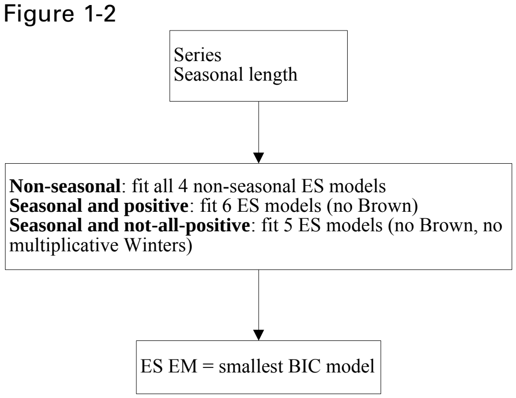
Univariate series

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

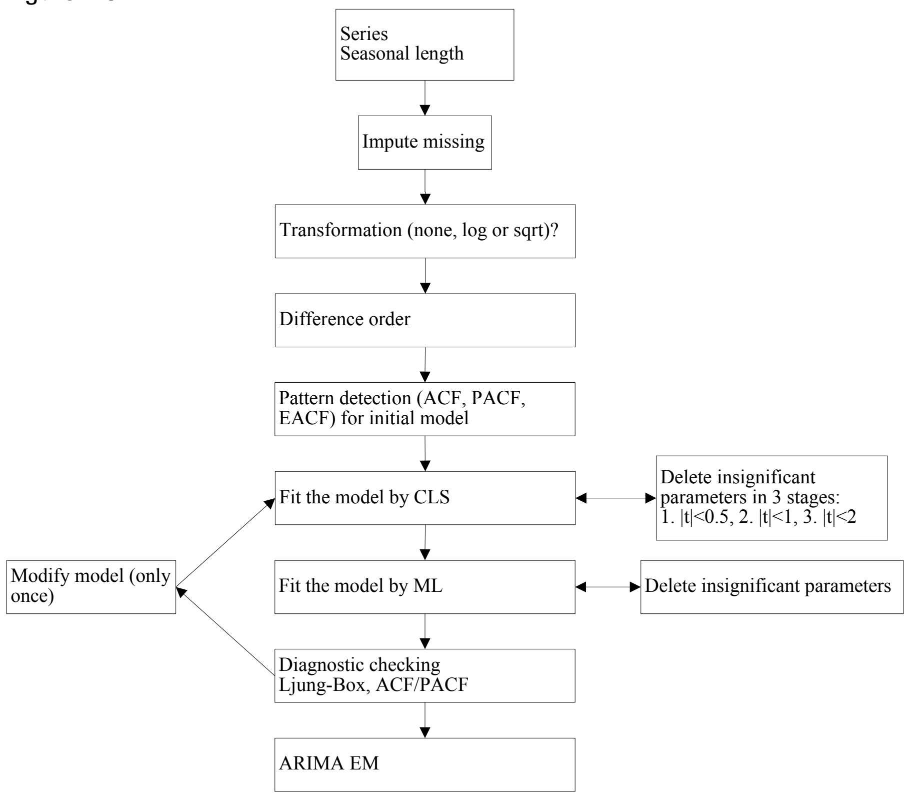
Figure 1-2



Note: for short series, 1<n≤10, fit simple ES.

ARIMA Expert Model

Figure 1-3



Note: for short series, do the following:

- If n<=10, fit AR(1) with constant term.
- 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.

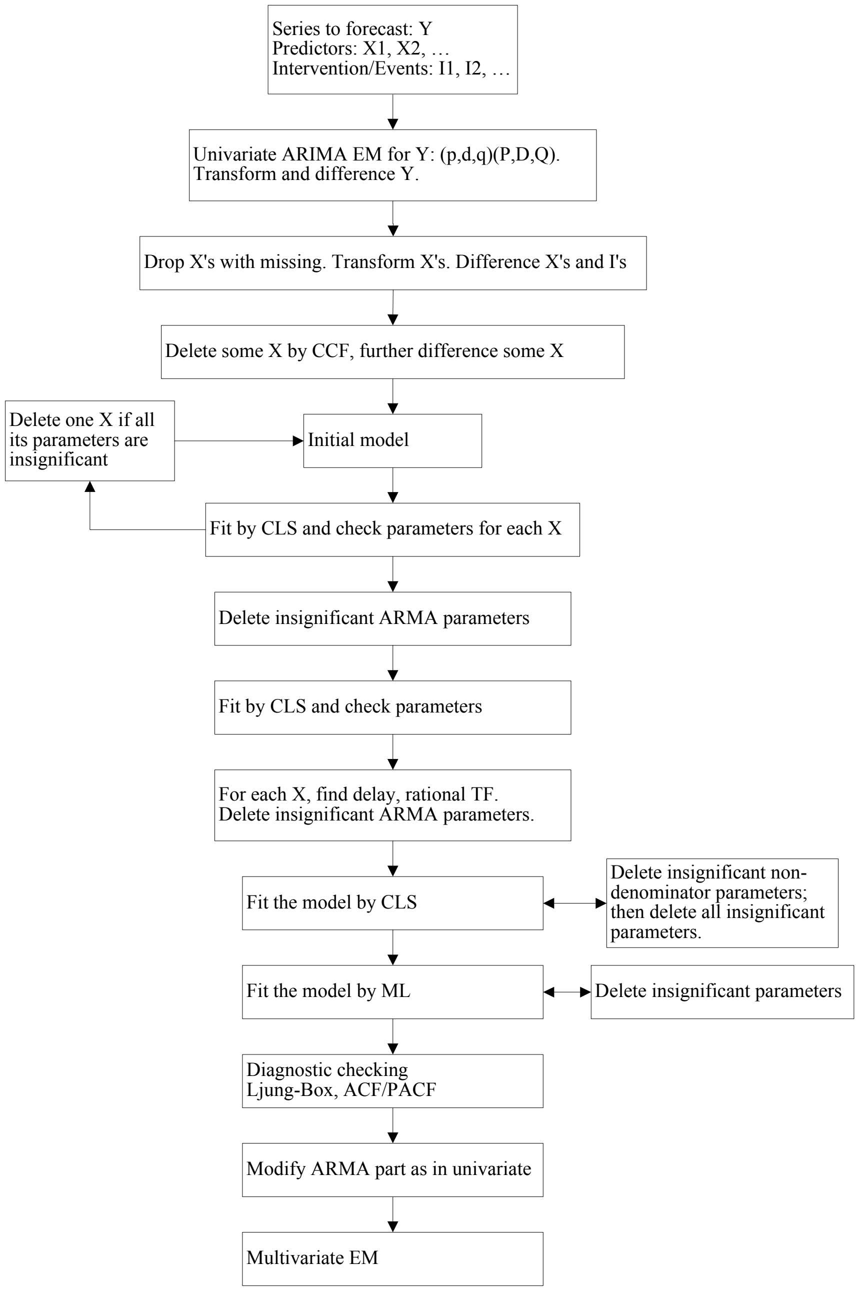
Multivariate series

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 1-4



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.