

IBM SPSS Modeler 14.2 Guida per
script e automazione



Nota: Prima di utilizzare queste informazioni e il relativo prodotto, leggere le informazioni generali disponibili in Note a pag. .

Il presente documento contiene informazioni di proprietà di SPSS Inc, una società IBM. Viene fornito su contratto di licenza ed è protetto dalle leggi sul copyright. Le informazioni contenute nella presente pubblicazione non includono alcuna garanzia di prodotto e qualsiasi dichiarazione inclusa nel presente manuale non deve essere interpretata come tale.

Al momento dell'invio di informazioni a IBM o SPSS, l'utente concede a IBM e SPSS un diritto non esclusivo a utilizzare o distribuire tali informazioni nel modo che ritiene appropriato senza incorrere in alcun obbligo nei confronti dell'utente.

© **Copyright IBM Corporation 1994, 2011..**

Prefazione

IBM® SPSS® Modeler è l'efficace workbench di data mining aziendale di IBM Corp.. SPSS Modeler consente alle organizzazioni di migliorare le relazioni con i clienti e con il pubblico grazie a un'analisi approfondita dei dati. Le organizzazioni potranno utilizzare le informazioni ottenute tramite SPSS Modeler per mantenere i clienti di valore, cogliere opportunità di vendite incrociate, attrarre nuovi clienti, individuare frodi, diminuire i rischi e migliorare l'offerta di servizi a livello statale.

L'interfaccia visiva di SPSS Modeler favorisce l'applicazione di una competenza aziendale specifica da parte degli utenti, grazie alla quale sarà possibile ottenere modelli di previsione più efficaci e una riduzione nei tempi di sviluppo delle soluzioni. SPSS Modeler offre una vasta gamma di tecniche di creazione di modelli, quali previsione, classificazione, segmentazione e algoritmi per l'individuazione delle associazioni. IBM® SPSS® Modeler Solution Publisher consente quindi di distribuire a livello aziendale i modelli creati in modo che vengano utilizzati dai responsabili dei processi decisionali oppure inseriti in un database.

Informazioni su IBM Business Analytics

Il software IBM Business Analytics fornisce informazioni complete, coerenti e accurate a cui i responsabili delle decisioni possono affidarsi per ottimizzare le prestazioni dell'azienda. Un ampio portafoglio di applicazioni di [business intelligence](#), [analisi predittiva](#), [gestione delle prestazioni e delle strategie finanziarie](#) e [analisi](#) offre una panoramica chiara, istantanea e interattiva delle prestazioni attuali e la possibilità di prevedere i risultati futuri. Utilizzato in combinazione con potenti soluzioni di settore, prassi consolidate e servizi professionali, questo software consente alle aziende di tutte le dimensioni di ottimizzare la produttività, automatizzare le decisioni senza problemi e fornire risultati migliori.

Come parte di questo portafoglio, il software IBM SPSS Predictive Analytics consente alle aziende di prevedere gli eventi futuri e di agire tempestivamente in modo da migliorare i risultati delle attività aziendali. Le aziende, gli enti governativi e le università di tutto il mondo si affidano alla tecnologia IBM SPSS perché rappresenta un vantaggio concorrenziale in termini di attrazione, retention e aumento dei clienti, riducendo al tempo stesso le frodi e limitando i rischi. Incorporando il software IBM SPSS nelle attività quotidiane, le aziende diventano imprese in grado di effettuare previsioni e di gestire e automatizzare le decisioni, per raggiungere gli obiettivi aziendali e vantaggi tangibili sulla concorrenza. Per ulteriori informazioni o per contattare un rappresentante, visitare il sito <http://www.ibm.com/spss>.

Supporto tecnico

Il supporto tecnico è a disposizione dei clienti che dispongono di un contratto di manutenzione. I clienti possono contattare il supporto tecnico per richiedere assistenza per l'utilizzo di IBM Corp. o per l'installazione di uno degli ambienti hardware supportati. Per contattare il supporto tecnico, visitare il sito Web IBM Corp. all'indirizzo <http://www.ibm.com/support>. Ricordare che durante la richiesta di assistenza sarà necessario fornire i dati di identificazione personali, i dati relativi alla propria società e il numero del contratto di manutenzione.

Contenuto

1 Informazioni su IBM SPSS Modeler 1

| | |
|--|---|
| IBM SPSS Modeler Server | 1 |
| Opzioni di IBM SPSS Modeler. | 2 |
| IBM SPSS Text Analytics | 2 |
| Documentazione di IBM SPSS Modeler | 2 |
| Esempi di applicazioni. | 4 |
| Cartella Demos. | 5 |

Parte I: Script e linguaggio di script

2 Panoramica sugli script 7

| | |
|---|----|
| Tipi di script | 7 |
| Script dello stream | 8 |
| Esempio di script dello stream: Addestramento di una rete neurale | 9 |
| Script locali | 10 |
| Esempio di script locale: Salvataggio e caricamento di un modello | 11 |
| Esempio di script locale: Generazione di un Modello di selezione funzioni | 12 |
| Script dei Supernodi | 13 |
| Esempio di script di un Supernodo | 14 |
| Esecuzione e interruzione degli script | 14 |
| Trova e sostituisci. | 15 |

3 Linguaggio di script 18

| | |
|--|----|
| Panoramica sul linguaggio di script | 18 |
| Sintassi degli script | 18 |
| Riferimenti a nodi | 19 |
| Recupero di oggetti | 21 |
| Impostazione dell'oggetto corrente. | 21 |
| Apertura di stream e altri oggetti | 22 |
| Utilizzo di più stream. | 23 |
| Variabili script locali | 23 |
| Parametri stream, sessione e Supernodo | 24 |
| Controllo dell'esecuzione degli script | 26 |

| | |
|---|----|
| Operatori negli script | 26 |
| Espressioni CLEM negli script | 27 |
| Inserimento di commenti e continuazioni | 27 |
| Blocchi di testo letterale | 28 |

4 Comandi di script

30

| | |
|--|----|
| Comandi di script generali | 30 |
| execute_all | 30 |
| execute_script | 30 |
| exit | 30 |
| for...endfor | 31 |
| if...then...else.... | 32 |
| Comando set | 32 |
| Comando var | 35 |
| Oggetti nodo | 36 |
| create NODO | 36 |
| connect NODO | 37 |
| delete NODO | 37 |
| disable NODO | 37 |
| disconnect NODO | 38 |
| duplicate NODO | 38 |
| enable NODO | 38 |
| execute NODO | 38 |
| export NODO as FILE | 39 |
| flush NODO | 39 |
| get node NODO | 40 |
| load node NOMEFILE | 40 |
| position NODO | 40 |
| rename NODO as NUOVONOME | 41 |
| retrieve node REPOSITORY_PATH | 41 |
| save node NODO as NOMEFILE | 42 |
| store node NODO as PERCORSO_REPOSITORY | 42 |
| Oggetti modello | 42 |
| Nomi degli insiemi di modelli | 42 |
| Per evitare nomi di modelli duplicati | 44 |
| delete model MODELLO | 44 |
| export model MODELLO as FILE | 45 |
| insert model MODELLO | 46 |
| load model NOMEFILE | 46 |
| retrieve model REPOSITORY_PATH | 47 |

| | |
|--|----|
| save model MODELLO as NOMEFILE | 47 |
| store model MODELLO as PERCORSO_REPOSITORY | 47 |
| Oggetti stream | 47 |
| create stream DEFAULT_FILENAME. | 47 |
| close STREAM. | 47 |
| clear stream | 48 |
| get stream STREAM. | 48 |
| load stream NOMEFILE | 48 |
| Open stream NOMEFILE. | 49 |
| retrieve stream REPOSITORY_PATH. | 49 |
| save STREAM as NOMEFILE | 49 |
| store stream as REPOSITORY_PATH | 50 |
| with stream STREAM. | 51 |
| Oggetti del progetto | 51 |
| execute_project | 51 |
| load project NOMEFILE | 51 |
| retrieve project REPOSITORY_PATH. | 51 |
| save project as NOMEFILE. | 52 |
| store project as REPOSITORY_PATH | 52 |
| Oggetti stato | 52 |
| load state NOMEFILE. | 52 |
| Oggetti risultato | 52 |
| value RESULT | 52 |
| Oggetti file | 53 |
| close FILE | 53 |
| open FILE. | 53 |
| write FILE | 54 |
| Oggetti di output. | 54 |
| Nomi dei tipi di output | 54 |
| delete output OUTPUT. | 55 |
| export output OUTPUT. | 55 |
| get output OUTPUT | 55 |
| load output NOMEFILE. | 55 |
| retrieve output REPOSITORY_PATH | 55 |
| save output OUTPUT as FILENAME | 56 |
| store output OUTPUT as REPOSITORY_PATH | 56 |

5 Suggestioni per gli script 57

| | |
|---|----|
| Modifica dell'esecuzione dello stream | 57 |
| Esecuzione di cicli sui nodi. | 57 |

| | |
|---|-----------|
| Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository | 58 |
| Generazione di una password codificata | 60 |
| Controllo degli script. | 60 |
| Script dalla riga di comando | 61 |
| Compatibilità con le versioni precedenti | 61 |
| | |
| 6 <i>Esempi di script</i> | 63 |
| Report nodo Tipo | 63 |
| Report di stream | 66 |
| | |
| 7 <i>Argomenti della riga di comando</i> | 69 |
| Modalità di richiamo del software. | 69 |
| Utilizzo degli argomenti della riga di comando. | 69 |
| Combinazione di più argomenti | 70 |
| Argomenti per la connessione al server. | 70 |
| Argomenti per la connessione a IBM SPSS Collaboration and Deployment Services Repository | 72 |
| Argomenti di sistema. | 72 |
| Argomenti dei parametri | 74 |
| | |
| 8 <i>Riferimenti al linguaggio CLEM</i> | 75 |
| Panoramica di riferimento su CLEM | 75 |
| Tipi di dati CLEM | 75 |
| Interi | 76 |
| Numeri reali. | 76 |
| Caratteri | 77 |
| Stringhe. | 77 |
| Elenchi | 77 |
| Campi | 77 |
| Date | 78 |
| Time | 79 |
| Operatori CLEM | 79 |
| Riferimenti alle funzioni. | 81 |
| Convenzioni utilizzate nelle descrizioni delle funzioni | 82 |
| Funzioni di informazione. | 83 |

| | |
|--|-----|
| Funzioni di conversione | 84 |
| Funzioni di confronto | 85 |
| Funzioni logiche | 87 |
| Funzioni numeriche | 88 |
| Funzioni trigonometriche | 89 |
| Funzioni di probabilità | 90 |
| Operazioni bitwise sui numeri interi | 90 |
| Funzioni casuali | 92 |
| Funzioni stringa | 92 |
| Funzioni SoundEx | 97 |
| Funzioni di data e ora | 98 |
| Funzioni sequenza | 102 |
| Funzioni globali | 108 |
| Funzioni per la gestione di valori vuoti e nulli | 109 |
| Campi speciali | 109 |

Parte II: Guida alle proprietà

9 Guida alle proprietà **113**

| | |
|---|-----|
| Panoramica sui riferimenti alle proprietà | 113 |
| Sintassi delle proprietà | 113 |
| Esempi di proprietà dei nodi e degli stream | 115 |
| Panoramica sulle proprietà dei nodi | 116 |
| Proprietà comuni dei nodi | 116 |

10 Proprietà stream **117**

11 Proprietà progetto **120**

12 Proprietà dei nodi di input **121**

| | |
|--|-----|
| Proprietà comuni dei nodi di input | 121 |
| Proprietà cognosimportnode | 123 |
| Proprietà databasenode | 124 |
| Proprietà datacollectionimportnode | 125 |

| | |
|--|-----|
| Proprietà excelimportnode | 127 |
| Proprietà evimportnode | 129 |
| Proprietà fixedfilenode | 129 |
| Proprietà sasimportnode | 131 |
| Proprietà statisticsimportnode | 132 |
| Proprietà userinputnode | 132 |
| Proprietà variablefilenode | 133 |
| Proprietà xmlimportnode | 136 |

13 Proprietà dei nodi Operazioni su record 138

| | |
|---------------------------------------|-----|
| Proprietà appendnode | 138 |
| Proprietà aggregatenode | 138 |
| Proprietà balancenode | 139 |
| Proprietà distinctnode | 140 |
| Proprietà mergenode | 141 |
| Proprietà rfmaggregatenode | 142 |
| Proprietà del nodo Campione | 144 |
| Proprietà selectnode | 146 |
| Proprietà sortnode | 146 |

14 Proprietà dei nodi Operazioni su campi 147

| | |
|---|-----|
| Proprietà anonymizenode | 147 |
| Proprietà autodatapreinode | 148 |
| Proprietà binningnode | 151 |
| Proprietà derivenode | 153 |
| Proprietà ensemblenode | 155 |
| Proprietà fillernode | 156 |
| Proprietà filternode | 156 |
| Proprietà historynode | 158 |
| Proprietà partitionnode | 158 |
| Proprietà del nodo Ricodifica | 160 |
| Proprietà reordernode | 161 |
| Proprietà restructurenode | 161 |
| Proprietà rfmanalysisnode | 162 |
| Proprietà settoflagnode | 163 |

| | |
|---|-----|
| Proprietà statisticstransformnode | 164 |
| Proprietà timeintervalnode | 164 |
| Proprietà transposenode | 169 |
| Proprietà typenode | 170 |

15 Proprietà dei nodi Grafici 175

| | |
|---|-----|
| Proprietà comuni dei nodi Grafici | 175 |
| Proprietà collectionnode | 176 |
| Proprietà distributionnode | 177 |
| Proprietà evaluationnode | 178 |
| Proprietà graphboardnode | 180 |
| Proprietà histogramnode | 181 |
| Proprietà multiplotnode | 182 |
| Proprietà plotnode | 183 |
| Proprietà timeplotnode | 185 |
| Proprietà webnode | 187 |

16 Proprietà dei nodi Modelli 189

| | |
|--|-----|
| Proprietà comuni dei nodi Modelli | 189 |
| Proprietà anomalydetectionnode | 190 |
| Proprietà apriorinode | 191 |
| Proprietà autotransformnode | 192 |
| Impostazione delle proprietà degli algoritmi | 194 |
| Proprietà autoclusternode | 195 |
| Proprietà autonumericnode | 197 |
| Proprietà bayesnetnode | 198 |
| Proprietà c50node | 200 |
| Proprietà carmanode | 201 |
| Proprietà cartnode | 202 |
| Proprietà chaidnode | 205 |
| Proprietà coxregnode | 207 |
| Proprietà decisionlistnode | 209 |
| Proprietà discriminantnode | 210 |
| Proprietà factornode | 212 |
| Proprietà featureselectionnode | 214 |

| | |
|---|-----|
| Proprietà genlinnode | 216 |
| Proprietà kmeansnode | 219 |
| Proprietà knnnode | 220 |
| Proprietà kohonennode | 221 |
| Proprietà linearnode | 222 |
| Proprietà logregnode | 224 |
| Proprietà neuralnetnode | 228 |
| Proprietà neuralnetworknode | 230 |
| Proprietà questnode | 232 |
| Proprietà regressionnode | 234 |
| Proprietà sequencenode | 236 |
| Proprietà slrmnode | 237 |
| Proprietà statisticsmodelnode | 238 |
| Proprietà svmnode | 238 |
| Proprietà timeseriesnode | 239 |
| Proprietà twostepnode | 241 |

17 Proprietà dei nodi degli insiemi di modelli 243

| | |
|--|-----|
| Proprietà applyanomalydetectionnode | 243 |
| Proprietà applyapriorinode | 243 |
| Proprietà applyautoclassifiernode | 244 |
| Proprietà applyautoclusternode | 244 |
| Proprietà applyautonumericnode | 245 |
| Proprietà applybayesnetnode | 245 |
| Proprietà applyc50node | 245 |
| Proprietà applycarmanode | 246 |
| Proprietà applycartnode | 246 |
| Proprietà applychaidnode | 246 |
| Proprietà applycoxregnode | 247 |
| Proprietà applydecisionlistnode | 247 |
| Proprietà applydiscriminantnode | 247 |
| Proprietà applyfactornode | 248 |
| Proprietà applyfeatureselectionnode | 248 |
| Proprietà applygeneralizedlinearnode | 248 |
| Proprietà applykmeansnode | 248 |
| Proprietà applyknnnode | 249 |
| Proprietà applykohonennode | 249 |

| | |
|--|-----|
| Proprietà applylinearnode | 249 |
| Proprietà applylogregnode | 249 |
| Proprietà applyneuralnetnode | 249 |
| Proprietà applyneuralnetworknode | 250 |
| Proprietà applyquestnode | 250 |
| Proprietà applyregressionnode | 251 |
| Proprietà applyselflearningnode | 251 |
| Proprietà applysequencenode | 251 |
| Proprietà applysvmnode | 251 |
| Proprietà applytimeseriesnode | 252 |
| Proprietà applytwostepnode | 252 |

18 Proprietà dei nodi Modelli database 253

| | |
|--|-----|
| Proprietà dei nodi Modelli Microsoft | 253 |
| Proprietà dei nodi Modelli Microsoft | 253 |
| Proprietà degli insiemi di modelli Microsoft | 256 |
| Proprietà dei nodi Modelli Oracle | 258 |
| Proprietà dei nodi Modelli Oracle | 258 |
| Proprietà degli insiemi di modelli Oracle | 263 |
| Proprietà dei nodi Modelli IBM DB2 | 264 |
| Proprietà dei nodi Modelli IBM DB2 | 264 |
| Proprietà degli insiemi di modelli IBM DB2 | 269 |
| Proprietà dei nodi Modelli IBM Netezza Analytics | 270 |
| Proprietà dei nodi Modelli Netezza | 270 |
| Proprietà degli insiemi di modelli Netezza | 272 |

19 Proprietà dei nodi Output 273

| | |
|--|-----|
| Proprietà analysisnode | 273 |
| Proprietà dataauditnode | 274 |
| Proprietà matrixnode | 276 |
| Proprietà meansnode | 277 |
| Proprietà reportnode | 279 |
| Proprietà setglobalsnode | 280 |
| Proprietà statisticsnode | 280 |
| Proprietà statisticsoutputnode | 282 |

| | |
|---|------------|
| Proprietà tablenode | 282 |
| Proprietà transformnode. | 284 |
| 20 Proprietà dei nodi di esportazione | 286 |
| Proprietà comuni dei nodi di esportazione | 286 |
| Proprietà del nodo di esportazione Cognos | 286 |
| Proprietà databaseexportnode | 287 |
| Proprietà datacollectionexportnode | 291 |
| Proprietà excelexportnode | 291 |
| Proprietà outputfilenode | 292 |
| Proprietà sasexportnode | 293 |
| Proprietà statisticsexportnode | 294 |
| Proprietà xmlexportnode | 294 |
| 21 Proprietà dei nodi IBM SPSS Statistics | 295 |
| Proprietà statisticsimportnode | 295 |
| Proprietà statisticstransformnode. | 295 |
| Proprietà statisticsmodelnode | 296 |
| Proprietà statisticsoutputnode | 297 |
| Proprietà statisticsexportnode | 297 |
| 22 Proprietà dei Supernodi | 299 |
| Appendice | |
| A Note | 301 |
| Indice | 304 |

Informazioni su IBM SPSS Modeler

IBM® SPSS® Modeler è un insieme di strumenti di data mining che consente di sviluppare rapidamente modelli predittivi con l'ausilio di competenze aziendali e di eseguirne il deployment nelle operazioni aziendali per migliorare i processi decisionali. Progettato secondo il modello CRISP-DM conforme agli standard di settore, SPSS Modeler supporta l'intero processo di data mining, dai dati a risultati aziendali migliori.

SPSS Modeler offre numerosi metodi di modellazione ricavati dall'apprendimento automatico, dall'intelligenza artificiale e dalla statistica. I metodi disponibili nella palette Modelli consentono di ricavare nuove informazioni dai dati e di sviluppare modelli predittivi. Ogni metodo ha determinati punti di forza e si presta meglio per particolari tipi di problemi.

SPSS Modeler può essere acquistato come prodotto autonomo oppure utilizzato in combinazione con SPSS Modeler Server. È inoltre disponibile una serie di opzioni, come illustrato nelle sezioni seguenti. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

IBM SPSS Modeler Server

SPSS Modeler utilizza un'architettura client/server per distribuire le richieste di operazioni che utilizzano molte risorse a potenti componenti software server, con un conseguente miglioramento della performance su insiemi di dati di grandi dimensioni. Oltre a quelli elencati di seguito, potrebbero essere disponibili altri prodotti e aggiornamenti. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

SPSS Modeler. SPSS Modeler è una versione del prodotto completa nelle funzioni, che viene installata ed eseguita sul desktop dell'utente. È possibile eseguirla in modalità locale come prodotto autonomo oppure in modalità distribuita assieme a IBM® SPSS® Modeler Server per ottenere una migliore performance su insiemi di dati di grandi dimensioni.

SPSS Modeler Server. SPSS Modeler Server viene eseguito continuamente in modalità di analisi distribuita assieme a una o più installazioni di IBM® SPSS® Modeler. Una configurazione di questo tipo consente di ottenere performance superiori quando si lavora su insiemi di dati di grandi dimensioni, in quanto le operazioni che richiedono un utilizzo consistente della memoria possono essere eseguite sul server senza necessità di scaricare i dati sul computer client. SPSS Modeler Server supporta inoltre funzionalità di ottimizzazione SQL e modellazione in-database, con ulteriori vantaggi in termini di prestazioni e grado di automazione. Per eseguire un'analisi deve essere disponibile almeno un'installazione SPSS Modeler.

Opzioni di IBM SPSS Modeler

I seguenti componenti e funzionalità possono essere acquistati separatamente e concessi in licenza per l'utilizzo con SPSS Modeler. Tenere presente che potrebbero rendersi disponibili anche prodotti e aggiornamenti aggiuntivi. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

- Accesso a SPSS Modeler Server per una maggiore scalabilità e prestazioni migliori su insiemi di dati di grandi dimensioni, nonché per il supporto di funzionalità di ottimizzazione SQL e modellazione in-database.
- SPSS Modeler Solution Publisher, per il calcolo automatico o in tempo reale del punteggio al di fuori dall'ambiente di SPSS Modeler. Per ulteriori informazioni, vedere l'argomento [IBM SPSS Modeler Solution Publisher in il capitolo 2 in IBM SPSS Modeler 14.2 Solution Publisher](#).
- Adattatori per il deployment in IBM SPSS Collaboration and Deployment Services o nell'applicazione thin-client IBM SPSS Modeler Advantage. Per ulteriori informazioni, vedere l'argomento [Archiviazione e deployment di oggetti di IBM SPSS Collaboration and Deployment Services Repository in il capitolo 9 in Manuale dell'utente di IBM SPSS Modeler 14.2](#).

IBM SPSS Text Analytics

IBM® SPSS® Text Analytics è un componente aggiuntivo completamente integrato per SPSS Modeler che utilizza tecnologie linguistiche avanzate e di Natural Language Processing (NLP) per elaborare rapidamente una grande varietà di dati di testo non strutturati, estrarre e organizzare i concetti chiave e raggruppare questi concetti in categorie. È quindi possibile combinare i concetti e le categorie estratti con dati strutturati esistenti, per esempio dati demografici, e applicarli alla modellazione utilizzando la suite completa degli strumenti di data mining di IBM® SPSS® Modeler per prendere decisioni migliori e più mirate.

- Il nodo Text Mining offre funzioni di modellazione di concetti e categorie nonché un workbench interattivo in cui è possibile eseguire esplorazioni avanzate di collegamenti e cluster di testo, creare categorie personalizzate e rifinire i modelli di risorse linguistiche.
- Sono supportati numerosi formati di importazione, inclusi blog e altre sorgenti basate sul Web.
- Sono inclusi inoltre modelli, librerie e dizionari personalizzati per domini specifici, quali CRM e genomica.

Nota: per l'accesso a questo componente, è necessaria una licenza distinta. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/modeler/>.

Documentazione di IBM SPSS Modeler

La documentazione completa nel formato guida in linea è disponibile nel menu Aiuto di SPSS Modeler. Sono incluse la documentazione per SPSS Modeler, SPSS Modeler Server e SPSS Modeler Solution Publisher, nonché la Guida alle applicazioni e altro materiale di supporto.

La documentazione completa in formato PDF dei singoli prodotti è disponibile nella cartella *Documentation* del DVD di ciascun prodotto.

- **Manuale dell'utente di IBM SPSS Modeler.** Introduzione generale all'utilizzo di SPSS Modeler che illustra come creare stream di dati, gestire valori mancanti, generare espressioni CLEM, utilizzare progetti e report e assemblare stream per il deployment tramite IBM SPSS Collaboration and Deployment Services, le applicazioni predittive o IBM SPSS Modeler Advantage.
- **IBM SPSS Modeler Source, Process, and Output Nodes.** Descrizioni di tutti i nodi utilizzati per leggere, elaborare e generare dati di output in vari formati, ovvero di nodi ad eccezione dei nodi Modelli.
- **IBM SPSS Modeler Nodi Modelli.** Descrizioni di tutti i nodi utilizzati per creare modelli di data mining. IBM® SPSS® Modeler offre numerosi metodi di modellazione ricavati dall'apprendimento automatico, dall'intelligenza artificiale e dalla statistica. [Per ulteriori informazioni, vedere l'argomento Panoramica sui nodi Modelli in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)
- **IBM SPSS Modeler Algorithms Guide.** Descrizione dei fondamenti di matematica per i metodi di modellazione utilizzati in SPSS Modeler.
- **IBM SPSS Modeler Guida alle applicazioni.** Gli esempi inclusi in questa guida forniscono indicazioni mirate e sintetiche su specifici metodi e tecniche di modellazione. Una versione in linea di questa guida è inoltre disponibile dal menu Aiuto. [Per ulteriori informazioni, vedere l'argomento Esempi di applicazioni in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)
- **IBM SPSS Modeler Script e automazione.** Informazioni sulle modalità di automazione del sistema tramite script, incluse le proprietà che è possibile utilizzare per manipolare nodi e stream.
- **IBM SPSS Modeler Deployment Guide.** Informazioni sull'esecuzione di stream e scenari SPSS Modeler come fasi dell'elaborazione di lavori in IBM® SPSS® Collaboration and Deployment Services Deployment Manager.
- **IBM SPSS Modeler Guida per lo sviluppatore CLEF.** CLEF consente di integrare programmi di terze parti (quali routine di elaborazione di dati o algoritmi di modellazione) come nodi in SPSS Modeler.
- **IBM SPSS Modeler Guida alla modellazione in-database.** Informazioni sulle modalità per utilizzare al meglio la potenza del database in uso al fine di ottenere prestazioni migliori ed estendere la gamma di funzionalità analitiche tramite algoritmi di terze parti.
- **IBM SPSS Modeler Server e Guida della performance.** Informazioni su come configurare e amministrare IBM® SPSS® Modeler Server.
- **Manuale dell'utente di IBM SPSS Modeler Administration Console.** Informazioni sull'installazione e l'utilizzo dell'interfaccia utente della console per il monitoraggio e la configurazione di SPSS Modeler Server. La console viene implementata come plug-in dell'applicazione Deployment Manager.
- **IBM SPSS Modeler Solution Publisher Guide.** SPSS Modeler Solution Publisher è un componente aggiuntivo che consente di pubblicare gli stream al di fuori dell'ambiente SPSS Modeler standard.
- **Guida CRISP-DM di IBM SPSS Modeler.** Guida passo a passo al data mining tramite la metodologia CRISP-DM con SPSS Modeler.

Esempi di applicazioni

Mentre gli strumenti per il data mining di SPSS Modeler consentono di risolvere un'ampia gamma di problemi a livello aziendale e organizzativo, gli esempi di applicazioni forniscono indicazioni mirate e sintetiche su specifici metodi e tecniche di modellazione. Gli insiemi di dati utilizzati negli esempi hanno dimensioni molto più limitate rispetto agli enormi archivi di dati gestiti da alcuni data miner, ma i concetti e i metodi coinvolti sono rapportabili alle applicazioni del mondo reale.

È possibile accedere agli esempi facendo clic su Esempi di applicazioni nel menu Aiuto di SPSS Modeler. I file di dati e gli stream di esempio sono installati nella cartella *Demos* nella directory di installazione del prodotto. [Per ulteriori informazioni, vedere l'argomento Cartella Demos in *Manuale dell'utente di IBM SPSS Modeler 14.2*.](#)

Esempi di modellazione in-database. Vedere gli esempi nella *IBM SPSS Modeler Guida alla modellazione in-database*.

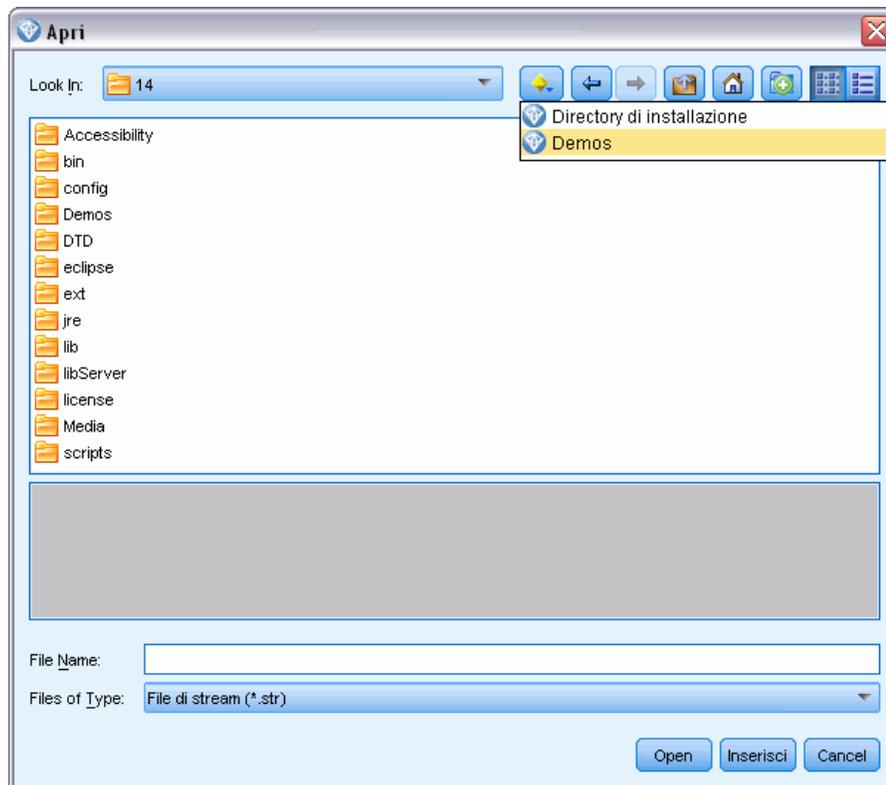
Esempi di script. Vedere gli esempi nella *IBM SPSS Modeler Guida per script e automazione*.

Cartella Demos

I file di dati e gli stream di esempio utilizzati negli esempi di applicazioni sono installati nella cartella *Demos* nella directory di installazione del prodotto. A questa cartella è possibile accedere anche dal gruppo di programmi IBM SPSS Modeler 14.2 nel menu Start di Windows oppure facendo clic su *Demos* nell'elenco delle directory recenti nella finestra di dialogo Apri file.

Figura 1-1

Selezione della cartella *Demos* dall'elenco delle directory utilizzate di recente



Parte I:
Script e linguaggio di script

Panoramica sugli script

Gli script di IBM® SPSS® Modeler sono un potente strumento per automatizzare i processi nell'interfaccia utente. Tramite gli script è possibile eseguire gli stessi tipi di azioni eseguite con il mouse o la tastiera, nonché automatizzare le attività ripetitive o la cui esecuzione manuale richiederebbe un tempo molto maggiore.

È possibile utilizzare gli script per:

- Imporre un ordine specifico per l'esecuzione dei nodi in uno stream.
- Impostare le proprietà di un nodo ed eseguire le derivazioni utilizzando un sottoinsieme di CLEM (Control Language for Expression Manipulation).
- Specificare una sequenza automatica di operazioni che in genere richiedono l'intervento dell'utente, per esempio la creazione e la verifica di un modello.
- Impostare processi di grande complessità per i quali sono necessari interventi sostanziali da parte dell'utente, per esempio le procedure di validazione incrociata che richiedono più processi di creazione e verifica dei modelli.
- Impostare i processi di manipolazione degli stream, per esempio recuperare uno stream di addestramento per un modello, eseguirlo e creare lo stream di verifica del modello corrispondente in modo automatico.

In questo capitolo sono fornite descrizioni approfondite ed esempi di script a livello di stream, script locali e script all'interno di Supernodi nell'interfaccia SPSS Modeler. Per ulteriori informazioni sul linguaggio di script, la sintassi e i comandi, consultare i capitoli che seguono.

Tipi di script

IBM® SPSS® Modeler utilizza tre tipi di script:

- Gli **script dello stream** sono archiviati come proprietà di stream e quindi salvati e caricati con uno stream specifico. Per esempio, è possibile scrivere uno script dello stream che automatizza il processo di addestramento e applicazione di un insieme di modelli. È anche possibile specificare che, ogni volta che viene eseguito un determinato stream, venga eseguito lo script anziché il contenuto dell'area di disegno dello stream.
- Gli **script locali** non sono associati ad alcuno stream particolare e vengono salvati in file di testo esterni. È possibile utilizzare uno script locale, per esempio, per manipolare insieme più stream.
- Gli **script del Supernodo** vengono archiviati come proprietà dello stream Supernodo. Gli script del Supernodo sono disponibili solo nei Supernodi terminali. È possibile utilizzare uno script del Supernodo per controllare la sequenza di esecuzione del contenuto del Supernodo. Per i Supernodi non terminali (di input o di elaborazione), è possibile definire le proprietà del Supernodo o direttamente i nodi che esso contiene nello script dello stream.

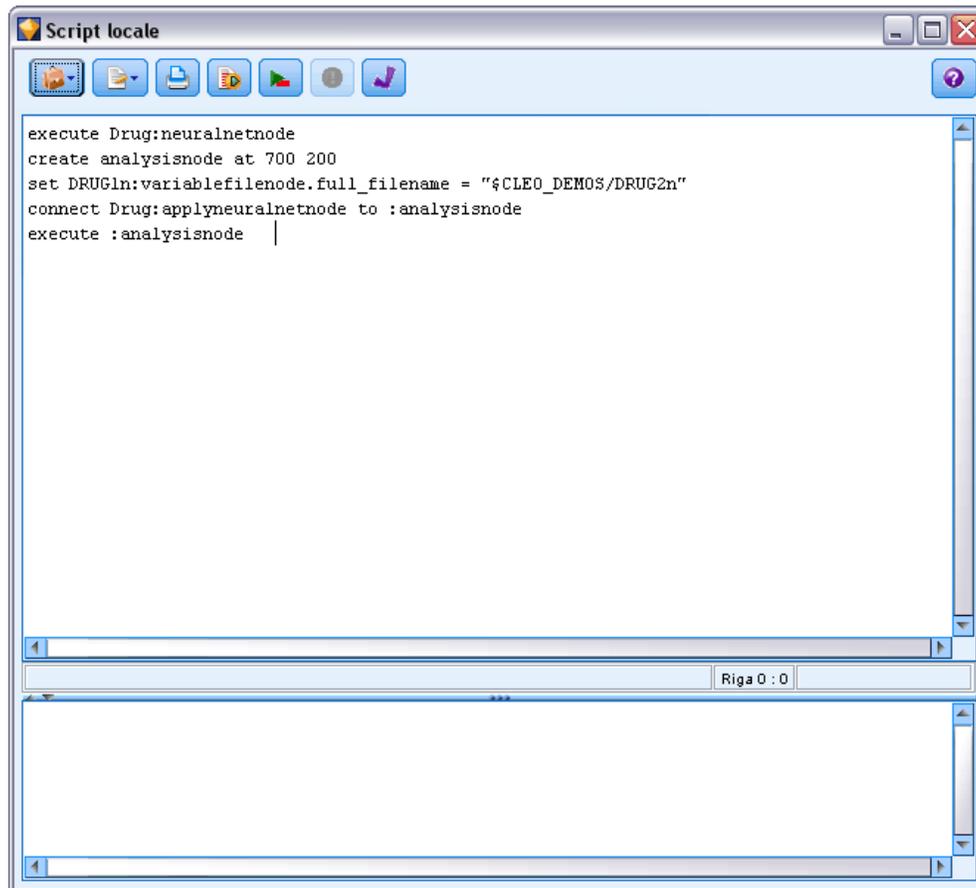
Script dello stream

È possibile utilizzare gli script per personalizzare le operazioni all'interno di uno stream specifico e salvarli insieme allo stream. Gli script dello stream possono essere utilizzati per specificare un particolare ordine di esecuzione per i nodi terminali all'interno di uno stream. La finestra di dialogo di script dello stream consente di modificare lo script salvato insieme allo stream corrente.

Per accedere alla scheda dello script dello stream nella finestra di dialogo Proprietà stream:

- ▶ Dal menu Strumenti, scegliere:
Proprietà stream > Script...
- ▶ Fare clic sulla scheda Script per utilizzare gli script nello stream corrente.

Figura 2-1
Finestra di dialogo di script dello stream



Le icone disponibili sulla barra degli strumenti nella parte superiore di questa finestra di dialogo consentono di eseguire le seguenti operazioni:

- Importare il contenuto di uno script locale preesistente nella finestra.
- Salvare lo script come file di testo.

- Stampare uno script.
- Accodare lo script di default.
- Eseguire l'intero script corrente.
- Eseguire righe selezionate di uno script.
- Verificare la sintassi dello script e, se vengono rilevati errori, visualizzarli nel riquadro inferiore della finestra di dialogo.

Inoltre, è possibile specificare se lo script debba essere o meno eseguito all'esecuzione dello stream. È possibile selezionare *Esegui questo script per eseguire lo script a ogni esecuzione dello stream* rispettando l'ordine di esecuzione dello script. L'automazione a livello di stream garantita in questo modo consente di accelerare la creazione dei modelli. Tuttavia, l'impostazione di default ignora questo script durante l'esecuzione dello stream. Anche se si seleziona l'opzione *Ignora questo script*, è sempre possibile eseguire lo script direttamente da questa finestra di dialogo.

Esempio di script dello stream: Addestramento di una rete neurale

È possibile utilizzare uno stream per addestrare una rete neurale durante l'esecuzione. La verifica del modello prevede in genere l'esecuzione del nodo di creazione modelli per aggiungere il modello allo stream, l'esecuzione delle connessioni appropriate e l'esecuzione del nodo *Analisi*.

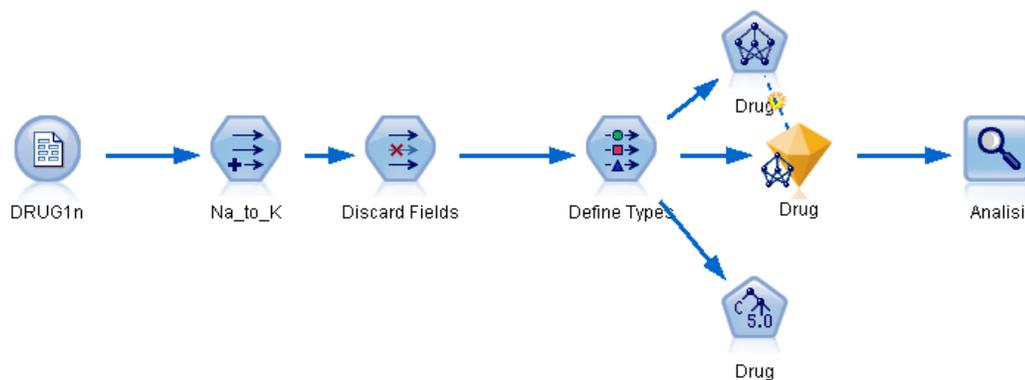
Con uno script di IBM® SPSS® Modeler, è possibile automatizzare il processo di verifica dell'insieme di modelli creato. Per esempio, il seguente script dello stream per lo stream di esempio *druglearn.str* (disponibile nella cartella */Demos/streams/* dell'installazione di SPSS Modeler) può essere eseguito dalla finestra di dialogo *Proprietà stream* (*Strumenti > Proprietà stream > Script*):

```
execute Drug:neuralnetworknode
create analysisnode at 700 200
set DRUG1n:variablefilename.full_filename = "$CLEO_DEMOS/DRUG2n"
connect :applyneuralnetworknode to :analysisnode
execute :analysisnode
```

L'elenco riportato di seguito descrive ogni riga in questo esempio di script.

- La prima riga esegue il nodo *Rete neurale* denominato *Drug* già trovato nello stream di esempio per generare un insieme di modelli e posizionarlo nell'area di disegno dello stream, collegato al nodo *Tipo* già presente nello stream.
- Nella seconda riga, lo script crea un nodo *Analisi* e lo colloca nella posizione 700 x 200 dell'area di disegno.
- Nella terza riga, la sorgente dati originale utilizzata nello stream viene quindi convertita in un insieme di dati di test denominato *DRUG2n*.
- Nella quarta riga, l'insieme di modelli *Rete neurale* viene collegato al nodo *Analisi*. Si noti che non viene utilizzato alcun nome per indicare l'insieme di modelli *Rete neurale* o il nodo *Analisi*, poiché nello stream non esistono altri nodi simili.
- Infine, viene eseguito il nodo *Analisi* per produrre il report di analisi.

Figura 2-2
Stream risultante



Questo script è destinato a funzionare insieme a uno stream esistente, poiché presuppone che un nodo Rete neurale denominato *Drug* esista già. Tuttavia, è anche possibile utilizzare uno script per creare ed eseguire uno stream nuovo, iniziando con un'area di disegno vuota. Per ulteriori informazioni sul linguaggio di script in generale, vedere *Panoramica sul linguaggio di script* a pag. 18. Per ulteriori informazioni specifiche sui comandi di script, vedere *Comandi di script* a pag. 30.

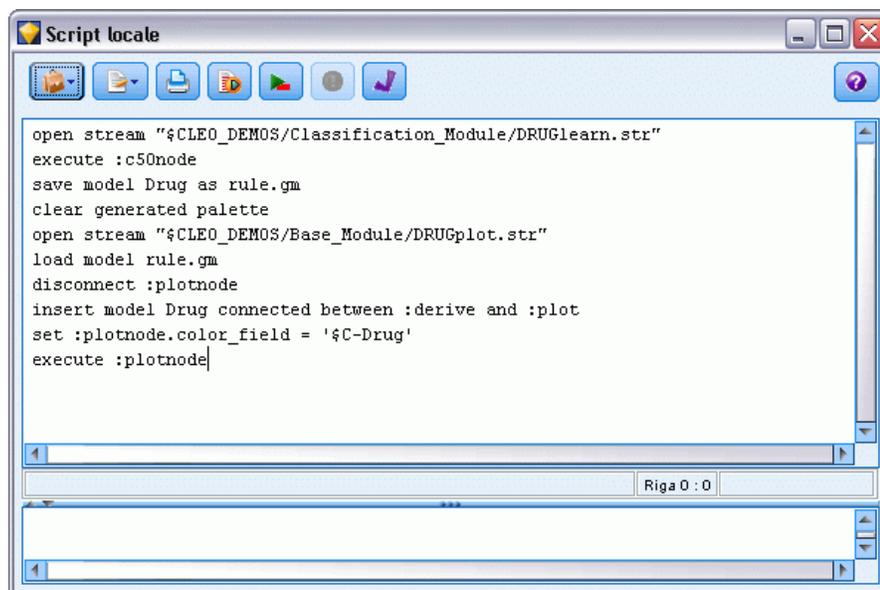
Script locali

Nella finestra di dialogo Script locale è possibile creare o modificare uno script salvato come file di testo. Nella finestra viene visualizzato il nome del file e sono disponibili funzionalità per il caricamento, il salvataggio, l'importazione e l'esecuzione degli script.

Per accedere alla finestra di dialogo dello script locale:

- Dal menu principale, scegliere:
Strumenti > Script locale

Figura 2-3
Finestra di dialogo Script locale



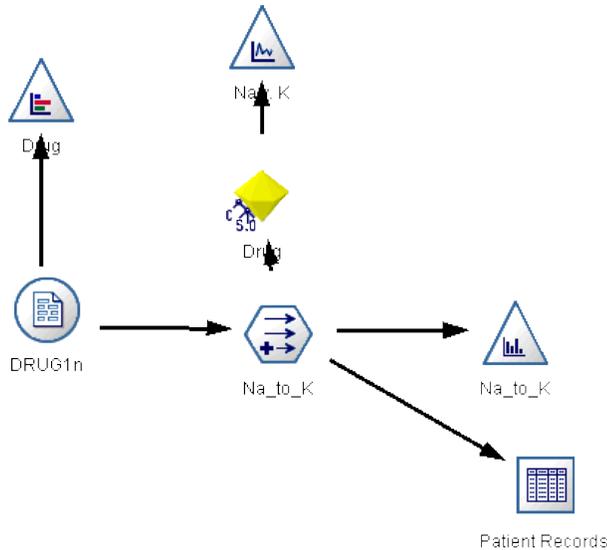
Per gli script locali e dello stream sono disponibili la stessa barra degli strumenti e le stesse opzioni di controllo della sintassi degli script. [Per ulteriori informazioni, vedere l'argomento Script dello stream a pag. 8.](#)

Esempio di script locale: Salvataggio e caricamento di un modello

Gli script locali sono utili per la manipolazione degli stream. Si supponga di avere due stream, uno che crea un modello e un altro che utilizza grafici per analizzare l'insieme di regole generato dal primo stream mediante i campi di dati esistenti. Uno script locale per questa situazione potrebbe essere simile al seguente:

```
open stream "$CLEO_DEMOS/Classification_Module/DRUGlearn.str"
execute :c50node
save model Drug as rule.gm
clear generated palette
open stream "$CLEO_DEMOS/Base_Module/DRUGplot.str"
load model rule.gm
disconnect :plotnode
insert model Drug connected between :derive and :plot
set :plotnode.color_field = '%C-Drug'
execute :plotnode
```

Figura 2-4
Stream risultante



Nota: Per ulteriori informazioni sul linguaggio di script in generale, vedere Panoramica sul linguaggio di script a pag. 18. Per ulteriori informazioni specifiche sui comandi di script, vedere Comandi di script a pag. 30.

Esempio di script locale: Generazione di un Modello di selezione funzioni

Iniziando con un'area di disegno vuota, questo esempio crea uno stream che genera un Modello di selezione funzioni, applica il modello e crea una tabella che elenca i 15 campi più importanti relativi all'obiettivo specificato.

```
create stream 'featureselection'
create statisticsimportnode
position :statisticsimportnode at 50 50
set :statisticsimportnode.full_filename = "$CLEO_DEMOS/customer_dbase.sav"
```

```
create typenode
position :typenode at 150 50
set :typenode.direction.'response_01' = Target
connect :statisticsimportnode to :typenode
```

```
create featureselectionnode
position :featureselectionnode at 250 50
set :featureselectionnode.screen_missing_values=true
set :featureselectionnode.max_missing_values=80
set :featureselectionnode.criteria = Likelihood
set :featureselectionnode.important_label = "Check Me Out!"
set :featureselectionnode.selection_mode = TopN
set :featureselectionnode.top_n = 15
connect :typenode to :featureselectionnode
execute :featureselectionnode
```

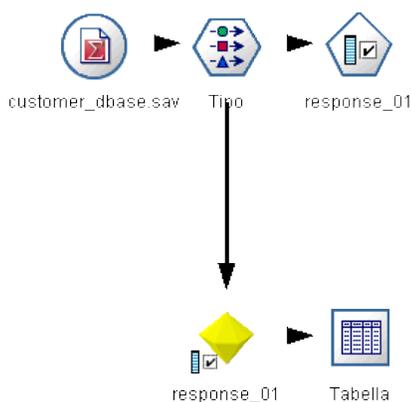
```

create tablenode
position :tablenode at 250 250
connect response_01:applyfeatureselectionnode to :tablenode
execute :tablenode

```

Questo script crea un nodo di input nel quale leggere i dati, utilizza un nodo Tipo per impostare il ruolo (direzione) del campo *response_01* su Target, quindi crea ed esegue un nodo Selezione funzioni. Inoltre, lo script connette i nodi e le posizioni nell'area di disegno dello stream per generare un layout leggibile. L'insieme di modelli così ottenuto viene quindi connesso a un nodo Tabella, che elenca i 15 campi più importanti come determinato dalle proprietà *selection_mode* e *top_n*. [Per ulteriori informazioni, vedere l'argomento Proprietà featureselectionnode in il capitolo 16 a pag. 214.](#)

Figura 2-5
Stream risultante



Script dei Supernodi

È possibile creare e salvare script all'interno di qualsiasi Supernodo terminale utilizzando il linguaggio di script di IBM® SPSS® Modeler. Questi script sono disponibili solo per i Supernodi terminali e vengono spesso utilizzati durante la creazione di modelli di stream o per imporre un ordine di esecuzione speciale per il contenuto del Supernodo. Gli script del Supernodo consentono anche l'esecuzione di più di uno script all'interno di uno stream.

Per esempio, si supponga che sia stato necessario specificare l'ordine di esecuzione di uno stream complesso e che il Supernodo contenga più nodi tra cui un nodo Globali, che deve essere eseguito prima di creare un nuovo campo utilizzato in un nodo Plot. In tal caso, è possibile creare uno script del Supernodo che esegue prima il nodo Globali. I valori calcolati da questo nodo, quali la media o la deviazione standard, possono quindi essere utilizzati quando viene eseguito il nodo Plot.

All'interno di uno script del Supernodo è possibile specificare le proprietà del nodo analogamente agli altri script. In alternativa, è possibile modificare e definire le proprietà di qualsiasi Supernodo o dei suoi nodi incapsulati direttamente da uno script dello stream. [Per ulteriori informazioni, vedere l'argomento Proprietà dei Supernodi in il capitolo 22 a pag. 299.](#) Questo metodo funziona per i Supernodi di input e di elaborazione e per i Supernodi terminali.

Nota: poiché solo i Supernodi terminali possono eseguire i propri script, la scheda Script della finestra di dialogo Supernodo è disponibile solo per i Supernodi terminali.

Per aprire la finestra di dialogo Script Supernodo dall'area di disegno principale:

- ▶ Selezionare un Supernodo terminale nell'area di disegno dello script e, dal menu Supernodo, scegliere:
Script Supernodo...

Per aprire la finestra di dialogo Script Supernodo dall'area di disegno del Supernodo in modalità Zoom avanti:

- ▶ Fare clic con il pulsante destro del mouse sull'area di disegno del Supernodo e dal menu di scelta rapida scegliere:
Script Supernodo...

Per ulteriori informazioni, vedere l'argomento Supernodi e script in il capitolo 9 in *IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output*.

Esempio di script di un Supernodo

Lo script del Supernodo riportato di seguito dichiara l'ordine in cui devono essere eseguiti i nodi terminali all'interno del Supernodo. Questo ordine assicura che il nodo Globali venga eseguito per primo, in modo che i valori calcolati da questo nodo possano successivamente essere utilizzati quando viene eseguito un altro nodo.

```
execute 'Globali'
execute 'Guadagno'
execute 'Profitto'
execute 'Età v. $CC-pep'
execute 'Tabella'
```

Esecuzione e interruzione degli script

Sono disponibili diversi sistemi per l'esecuzione degli script. Per esempio, nello script dello stream o nella finestra di dialogo dello script locale, il pulsante “Esegui questo script” esegue lo script completo:

Figura 2-6
Pulsante Esegui questo script



Il pulsante “Esegui solo righe selezionate” esegue una sola riga o un blocco di righe adiacenti selezionate nello script:

Figura 2-7
Pulsante Esegui solo righe selezionate



Per eseguire gli script è possibile utilizzare i metodi seguenti:

- Fare clic sul pulsante “Esegui questo script” o “Esegui solo righe selezionate” all’interno dello script di uno stream o nella finestra di dialogo dello script locale.
- Eseguire uno stream nel quale il metodo di esecuzione predefinito impostato è Esegui questo script.
- Utilizzare il flag `-execute` all’avvio in modalità interattiva. [Per ulteriori informazioni, vedere l'argomento Utilizzo degli argomenti della riga di comando in il capitolo 7 a pag. 69.](#)

Nota: uno script del Supernodo verrà eseguito insieme al Supernodo se nella finestra di dialogo Script Supernodo è stata selezionata l’opzione Esegui questo script.

Interruzione dell’esecuzione degli script

Nella finestra di dialogo dello script di uno stream, il pulsante rosso Interrompi viene attivato durante l’esecuzione dello script. Questo pulsante consente di interrompere l’esecuzione dello script e di qualsiasi stream corrente.

Trova e sostituisci

La finestra di dialogo Trova/Sostituisci è disponibile ogni volta che è possibile modificare il testo di script o di espressioni, compreso l’editor di script, il generatore di espressioni CLEM e quando si definisce un modello nel nodo Report. Quando si modifica un testo in una di queste aree, premere Ctrl-F per accedere alla finestra di dialogo, assicurandosi che il cursore sia posizionato in un’area di testo. In un nodo Riempimento, per esempio, è possibile accedere alla finestra di dialogo da qualsiasi area di testo della scheda Impostazioni oppure dal campo di testo nel Generatore di espressioni.

Figura 2-8

Finestra di dialogo Trova/Sostituisci



- ▶ Con il cursore posizionato in un’area di testo, premere Ctrl+F per accedere alla finestra di dialogo Trova/Sostituisci.
- ▶ Immettere il testo da cercare oppure sceglierne uno dall’elenco a discesa degli elementi cercati di recente.
- ▶ Se necessario, immettere il testo sostitutivo.

- ▶ Fare clic su Trova successivo per avviare la ricerca.
- ▶ Fare clic su Sostituisci per sostituire la selezione corrente oppure scegliere Sostituisci tutto per aggiornare tutte le istanze o quelle selezionate.
- ▶ Al termine di ogni operazione, la finestra di dialogo si chiude. Premere F3 da qualsiasi area di testo per ripetere l'ultima operazione di ricerca oppure premere Ctrl+F per accedere nuovamente alla finestra di dialogo.

Opzioni di ricerca

Maiuscole/minuscole. Specifica se l'operazione di ricerca fa distinzione tra maiuscole e minuscole, per esempio se *miavar* corrisponde a *miaVar*. Il testo sostitutivo viene sempre inserito esattamente come viene digitato, indipendentemente da questa impostazione.

Solo parole intere. Specifica se l'operazione di ricerca cerca le occorrenze che sono parole intere. Se questa opzione è selezionata, la ricerca di *palla* non consentirà di trovare per esempio *pallavolo* o *Palladio*.

Espressioni regolari. Specifica se è utilizzata la sintassi delle espressioni regolari (vedere la sezione seguente). Quando questa opzione è selezionata, l'opzione Solo parole intere è disattivata e il relativo valore viene ignorato.

Solo testo selezionato. Controlla l'ambito della ricerca quando si utilizza l'opzione Sostituisci tutto.

Sintassi delle espressioni regolari

Le espressioni regolari consentono di cercare caratteri speciali, quali tabulazioni o caratteri di nuova riga, classi o intervalli di caratteri quali *a - d*, cifre e caratteri diversi da cifre, nonché limiti, per esempio l'inizio o la fine di una riga. Sono supportati i seguenti tipi di espressioni.

Corrispondenze di caratteri

| Caratteri | Corrispondenze |
|-----------|--|
| x | Il carattere x |
| \\ | Il carattere barra rovesciata |
| \0n | Il carattere con valore ottale 0n (0 <= n <= 7) |
| \0nn | Il carattere con valore ottale 0nn (0 <= n <= 7) |
| \0mnn | Il carattere con valore ottale 0mnn (0 <= m <= 3, 0 <= n <= 7) |
| \xhh | Il carattere con valore esadecimale 0xhh |
| \uhhhh | Il carattere con valore esadecimale 0xhhhh |
| \t | Il carattere di tabulazione ('\u0009') |
| \n | Il carattere di nuova riga (avanzamento riga) ('\u000A') |
| \r | Il carattere di ritorno a capo ('\u000D') |
| \f | Il carattere di avanzamento carta ('\u000C') |
| \a | Il carattere di avviso (campanello) ('\u0007') |
| \e | Il carattere di escape ('\u001B') |
| \cx | Il carattere di controllo corrispondente a x |

Corrispondenze di classi di caratteri

| Classi di caratteri | Corrispondenze |
|---------------------|--|
| [abc] | a, b o c (classe semplice) |
| [^abc] | Qualsiasi carattere, eccetto a, b o c (sottrazione) |
| [a-zA-Z] | a-z oppure A-Z, incluse (intervallo) |
| [a-d[m-p]] | a-d oppure m-p (unione). In alternativa è possibile specificare [a-dm-p]. |
| [a-z&&[def]] | a-z + d, e oppure f (intersezione) |
| [a-z&&[^bc]] | a-z, eccetto b e c (sottrazione). In alternativa è possibile specificare [ad-z]. |
| [a-z&&[^m-p]] | a-z, eccetto m-p (sottrazione). In alternativa è possibile specificare [a-lq-z]. |

Classi di caratteri predefinite

| Classi di caratteri predefinite | Corrispondenze |
|---------------------------------|---|
| . | Qualsiasi carattere (può corrispondere o meno a terminazioni di riga) |
| \d | Qualsiasi cifra: [0-9] |
| \D | Una non cifra: [^0-9] |
| \s | Uno spazio vuoto: [\t\n\r\f] |
| \S | Uno spazio non vuoto: [^\s] |
| \w | Un carattere alfanumerico: [a-zA-Z_0-9] |
| \W | Un carattere diverso da alfanumerico: [^\w] |

Corrispondenze di limiti

| Corrispondenze di limiti | Corrispondenze |
|--------------------------|--|
| ^ | L'inizio di una riga |
| \$ | La fine di una riga |
| \b | Un limite di parola |
| \B | Un limite diverso da un limite di parola |
| \A | L'inizio dell'input |
| \Z | La fine dell'input ma per la terminazione finale, se disponibile |
| \z | La fine dell'input |

Linguaggio di script

Panoramica sul linguaggio di script

Il linguaggio di script di IBM® SPSS® Modeler include:

- Un formato per i riferimenti a nodi, stream, progetti, output e altri oggetti SPSS Modeler.
- Un insieme di istruzioni o comandi di script che può essere utilizzato per la manipolazione di questi oggetti.
- Un linguaggio di espressioni script per l'impostazione dei valori di variabili, parametri e altri oggetti.
- Supporto per commenti, continuazioni e blocchi di testo letterale.

Questa sezione illustra la sintassi di base per l'utilizzo del linguaggio di script. Le sezioni che seguono contengono informazioni su proprietà e comandi specifici.

Sintassi degli script

Per semplificare il processo di analisi, durante l'utilizzo degli script di IBM® SPSS® Modeler è consigliabile tenere presenti le seguenti regole:

- I nomi di variabili quali `income` o `referrerID` non devono essere inseriti tra virgolette.
- I nomi di variabili quali `^mystream` sono preceduti dal simbolo di accento circonflesso (^) se fanno riferimento a variabili esistenti il cui valore è già impostato. Tuttavia, tale simbolo non viene utilizzato quando viene dichiarato o impostato il valore di una variabile. [Per ulteriori informazioni, vedere l'argomento Riferimenti a nodi a pag. 19.](#)
- I riferimenti a parametri di Supernodo, sessioni e stream quali `'$P-Maxvalue'` devono essere racchiusi tra virgolette singole.
- Un'espressione con le virgolette doppie viene trattata come una stringa letterale, per esempio `"Web graph of BP and Drug"`. Ciò comporta risultati inattesi se le virgolette semplici o doppie non vengono utilizzate con cura, per esempio, `"$P-Maxvalue"` sarà una stringa anziché un riferimento al valore archiviato in un parametro.
- I nomi di file, quali `"druglearn.str"`, devono essere inseriti tra virgolette doppie.
- I nomi dei nodi, per esempio `datasourcenode` o `Na_to_K`, possono essere senza virgolette o inseriti tra virgolette singole. *Nota:* i nomi devono essere inseriti tra virgolette se includono spazi o caratteri speciali. Non è invece possibile utilizzare negli script nomi di nodo che iniziano con un numero, per esempio `'2a_referrerID'`.
- Le proprietà flag dovrebbero essere lette o impostate utilizzando i valori `true` e `false` (in caratteri minuscoli, come mostrato). Quando si impostano i valori, vengono riconosciute anche le variazioni, `includeOff`, `OFF`, `off`, `No`, `NO`, `no`, `n`, `N`, `f`, `F`, `False`, `FALSE` o `0`, ma in alcuni casi potrebbero causare degli errori durante la lettura dei valori delle proprietà. Tutti

gli altri valori vengono considerati come true. L'utilizzo coerente di true e false in lettere minuscole consente di evitare confusioni.

- Le stringhe letterali o i blocchi che comprendono interruzioni di linea, spazi o virgolette singole o doppie all'interno del blocco possono essere racchiuse tra virgolette triple. [Per ulteriori informazioni, vedere l'argomento Blocchi di testo letterale a pag. 28.](#)
- Le espressioni CLEM, quali "Age >= 55", devono essere inserite tra virgolette doppie, per esempio:

```
set :derivenode.flag_expr = "Age >= 55"
```

- Se un'espressione CLEM include le virgolette, assicurarsi di farle precedere da una barra rovesciata (\), per esempio:

```
set :node.parameter = "BP = \"HIGH\""
```

Sebbene non siano strettamente necessarie in tutte le istanze, queste indicazioni sono consigliate per una maggiore chiarezza. Lo strumento di verifica degli script disponibile in tutte le finestre di dialogo relative agli script contrassegna con un flag la sintassi ambigua.

Riferimenti a nodi

È possibile fare riferimento ai nodi negli script in vari modi:

- È possibile specificare i nodi utilizzando il nome, per esempio DRUG1n. È possibile definire il nome in base al tipo, per esempio, Drug:neuralnetworknode fa riferimento a un nodo Rete neurale denominato Drug e solo a quel tipo di nodo.
- È possibile specificare i nodi utilizzando solo il tipo, per esempio :neuralnetworknode fa riferimento a tutti i nodi Rete neurale. È possibile utilizzare qualsiasi tipo di nodo valido, per esempio samplenode, neuralnetworknode e kmeansnode. Il suffisso node è facoltativo e può essere omesso, anche se è consigliabile non farlo in quanto semplifica l'identificazione di eventuali errori negli script.
- È possibile fare riferimento ai singoli nodi mediante un ID univoco, visualizzato nella scheda Annotazioni di ciascun nodo. Utilizzare un simbolo "@" seguito dall'ID, per esempio @id5E5GJK23L.custom_name = "My Node". [Per ulteriori informazioni, vedere l'argomento Annotazioni in il capitolo 5 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Modelli generati. Le stesse regole sono valide per i nodi dei modelli generati. È possibile utilizzare il nome del nodo come viene visualizzato nella palette di modelli generati della finestra dei manager, oppure fare riferimento ai nodi dei modelli generati per tipo. Si noti che i nomi utilizzati per fare riferimento ai modelli generati nella finestra dei manager sono diversi da quelli utilizzati per i modelli aggiunti a uno stream ai fini del calcolo del punteggio (questi ultimi utilizzano un prefisso "apply"). [Per ulteriori informazioni, vedere l'argomento Nomi degli insiemi di modelli in il capitolo 4 a pag. 42.](#)

Riferimento a nodi tramite variabili

È possibile specificare i nomi e i tipi di nodi come valori di variabili script locali utilizzando nella sintassi l'accento circonflesso (^). Per esempio, se è necessario specificare il nome di un nodo, `^n` indica il nodo il cui nome è archiviato nella variabile `n` e `Drug:^t` indica il nodo denominato `Drug` il cui tipo è archiviato nella variabile `t`.

È possibile archiviare i riferimenti ai nodi nelle variabili script locali (dichiarate tramite un'istruzione `var`) ma non nei parametri stream, sessione oppure Supernodo. Per evitare ogni ambiguità nei riferimenti ai nodi, assegnare un ID nodo univoco a una variabile mentre si crea il nodo.

```
var x
set x = create typenode
set ^x.custom_name = "My Node"
```

- La prima riga crea una variabile denominata `x`.
- La seconda riga crea un nuovo nodo Tipo e archivia un riferimento al nodo in `x`. Si noti che `x` archivia un riferimento al nodo stesso, non il nome del nodo.
- La terza riga imposta il valore della proprietà `custom_name` per il nodo su "My Node". L'accento circonflesso viene usato per indicare che `x` è il nome di una variabile anziché un nodo. Senza tale simbolo, il sistema ricerca un nodo denominato `x`. L'accento circonflesso non è necessario quando viene dichiarata o impostata una variabile in quanto l'oggetto di un comando `var`, per esempio, può essere solo una variabile. Logicamente, però, nella terza riga `x` potrebbe essere un nome di nodo anziché una variabile, quindi l'accento circonflesso è necessario per distinguere questi due valori.

Una delle insidie più comuni è tentare di archiviare un riferimento a un nodo in una variabile senza prima dichiararla.

```
set x = create typenode
set ^x.custom_name = "My Node"
```

In tal caso, il comando `SET` tenta di creare `x` come parametro stream, sessione o Supernodo anziché come variabile e restituisce un errore perché non è possibile archiviare un riferimento a un nodo in un parametro.

Riferimento ai nodi mediante ID

È possibile anche archiviare un ID nodo univoco in una variabile. Per esempio:

```
var n
set n = "id5E5GJK23L"
set @^n.custom_name = "My Node"
```

Esecuzione di cicli sui nodi di uno stream. È anche possibile utilizzare la proprietà `stream.nodes` per restituire un elenco completo dei nodi di uno stream, e quindi eseguire un'iterazione sull'elenco per accedere ai singoli nodi. [Per ulteriori informazioni, vedere l'argomento Report di stream in il capitolo 6 a pag. 66.](#)

Esempi

NAME:TYPE

NAME è il nome di un nodo e TYPE è il relativo tipo. È necessario includere almeno il valore NAME o il valore TYPE. È possibile omettere uno di questi valori, ma non entrambi. Per esempio, il comando seguente crea un altro nodo Nuovo campo tra un nodo Testo variabile esistente denominato drug1n e un nodo Plot esistente (i nuovi nodi non usano i due punti):

```
create derivenode connected between drug1n and :plotnode
```

È anche possibile far precedere a NAME o TYPE il simbolo dell'accento circonflesso ^ per indicare la presenza di un parametro, per esempio:

Drug:^t

Questo riferimento indica un nodo denominato Drug, dove t è un parametro che specifica il tipo di nodo. Per esempio, se il valore di ^t è c50node, il riferimento sopra può essere tradotto come:

Drug:c50node

Analogamente, è possibile utilizzare un parametro per il nome del nodo. Per esempio, è possibile utilizzare entrambi i seguenti parametri in un contesto nel quale è richiesto un nome di nodo:

^n:derivenode

^n

Recupero di oggetti

Il comando `get` restituisce un riferimento a uno stream, a un nodo o a un oggetto di output, rendendo possibile la manipolazione di questi oggetti mediante gli script. Per esempio:

```
var mynode
set mynode = get node flag1:derivenode
position ^mynode at 400 400

var mytable = get output :tableoutput
export output ^mytable as c:/mytable.htm format html

set stream = get stream 'Stream1'
set ^stream.execute_method = "Script"
```

Impostazione dell'oggetto corrente

Le seguenti variabili speciali possono essere utilizzate per fare riferimento agli oggetti correnti:

- node
- stream
- output
- project

Fatta eccezione per `project`, le variabili possono anche essere reimpostate per modificare il contesto corrente. A differenza delle altre variabili di script, non devono essere dichiarate prima con il comando `var` perché sono predefinite.

```
set node = create typenode
rename ^node as "mytypenode"
```

```
set output = get output :statisticsoutput
export output ^output as c:/myoutput.htm format html
```

Poiché queste variabili speciali corrispondono ai nomi degli oggetti ai quali fanno riferimento, in alcuni casi la distinzione tra variabile e oggetto potrebbe risultare non completamente visibile, con conseguenti sottili distinzioni nell'utilizzo. [Per ulteriori informazioni, vedere l'argomento Comando set in il capitolo 4 a pag. 32.](#)

Commenti

L'assegnazione di un tipo di valore errato a una variabile speciale (per esempio l'impostazione di un nodo oggetto sulla variabile `stream`) genera un errore di esecuzione.

Nei casi in cui è possibile utilizzare la variabile speciale, è consentito anche l'utilizzo di qualsiasi variabile. Per esempio, per salvare lo stream corrente è possibile specificare:

```
save stream as 'C:/My Streams/Churn.str'
```

È tuttavia valida anche la seguente specifica:

```
save mio_stream as 'C:/My Streams/Churn.str'
```

in cui a `my_stream` è stato assegnato in precedenza un valore di stream.

Apertura di stream e altri oggetti

In uno script locale, è possibile aprire uno stream specificando il nome file e la posizione del file, per esempio:

```
open stream "c:/demos/druglearn.str"
```

Gli altri tipi di oggetti possono essere aperti utilizzando il comando `load`, per esempio:

```
load node c:/mynode.nod
```

```
load model c:/mymodel.gm
```

Comando open stream e comando load stream. Il comando `load stream` aggiunge lo stream specificato all'area di disegno senza cancellare i nodi dallo stream corrente. Questo comando era utilizzato più frequentemente nelle versioni precedenti ed è stato superato nelle versioni più recenti dalla possibilità di aprire, gestire e copiare nodi tra più stream.

Utilizzo di più stream

A parte i comandi utilizzati per accedere agli stream dal file system o dal IBM® SPSS® Collaboration and Deployment Services Repository (`open`, `load` e `retrieve`), la maggior parte dei comandi di script si applica automaticamente allo stream corrente. Negli script locali, tuttavia, può essere necessario aprire e manipolare più stream a partire dallo stesso script impostando un riferimento a qualsiasi stream aperto, oppure utilizzando il comando `with... endwith` per riassegnare temporaneamente lo stream corrente.

Per esempio, per chiudere uno stream diverso da quello corrente, è possibile utilizzare il comando `get stream` per fare riferimento allo stream desiderato:

```
set stream = get stream "druglearn"
close stream
```

Questo script riassegna lo stream variabile speciale allo stream `druglearn` (rendendolo lo stream corrente) e quindi chiude lo stream.

In alternativa, è possibile riassegnare temporaneamente lo stream corrente utilizzando l'istruzione `with stream`, per esempio:

```
with stream 'druglearn'
  create typenode
  execute_script
endwith
```

Queste istruzioni eseguono l'operazione `create`, nonché lo script dello stream per il quale lo stream specificato è stato impostato come stream corrente. Dopo che ogni istruzione è stata eseguita, viene ripristinato lo stream corrente originale. È inoltre possibile includere istruzioni condizionali e strutture cicliche, per esempio:

```
with stream 'druglearn'
  create tablenode at 500 400
  create selectnode connected between :typenode and :tablenode
  for l from 1 to 5
    set :selectnode.condition = 'Age > ' >> (l * 10)
    execute :selectnode
  endfor
endwith
```

L'istruzione di cui sopra imposta lo stream corrente su `STREAM` per tutte le espressioni all'interno del ciclo e ripristina il valore originale al termine del ciclo.

Variabili script locali

Le variabili script locali vengono dichiarate con il comando `var` e vengono impostate solo per lo script corrente. Le variabili si differenziano dai parametri, che è possibile impostare per una sessione, uno stream o un Supernodo e possono contenere solo stringhe o numeri.

```
var my_node
set my_node = create distributionnode
```

```
rename ^my_node as "Distribution of Flag"
```

Quando si riferiscono a variabili esistenti, accertarsi che il simbolo dell'accento circonflesso (^) preceda il nome del parametro. Per esempio, dato lo script precedente:

- La prima riga dichiara la variabile.
- La seconda riga ne imposta il valore.
- La terza riga rinomina il nodo a cui viene fatto riferimento dalla variabile (non la variabile stessa). L'accento circonflesso indica che ^my_node è il nome di una variabile anziché il nome letterale del nodo. Senza tale simbolo, il comando `rename` ricerca un nodo denominato `my_node`. Il simbolo dell'accento circonflesso non è necessario nella prima né nella seconda riga in quanto l'oggetto di un comando `var` può essere solo una variabile. L'accento circonflesso viene utilizzato solo quando si fa riferimento a una variabile già impostata, nel qual caso la rimozione dello stesso comporterebbe un riferimento ambiguo.
- La risoluzione dei riferimenti a variabili prevede la ricerca prima nell'elenco di variabili locali e quindi nell'elenco di parametri di sessioni, stream o Supernodo. Per esempio, se una variabile `x` è presente sia come variabile locale sia come parametro di sessione, l'utilizzo della sintassi '\$P-X' in un'istruzione di script consente di specificare l'utilizzo del parametro di sessione anziché della variabile locale.

Nota: in termini pratici, se si imposta una variabile senza prima dichiararla utilizzando un comando `var`, viene creato un parametro stream, sessione o Supernodo, in base al contesto dello script corrente. Per esempio, il codice seguente crea una variabile script locale denominata `z` e ne imposta il valore su [1 2 3]:

```
var z
set z = [1 2 3]
```

Se il comando `var` viene omesso (e presupponendo che non esista già una variabile o un nodo con questo nome), `z` viene creato come parametro anziché come variabile.

Parametri stream, sessione e Supernodo

È possibile definire parametri da utilizzare in espressioni CLEM e negli script. Si tratta a tutti gli effetti di variabili definite dall'utente che vengono salvate e mantenute con lo stream, la sessione o il Supernodo corrente e a cui è possibile accedere dall'interfaccia utente nonché attraverso script. Per esempio, durante il salvataggio di uno stream, qualsiasi parametro impostato per tale stream viene salvato contemporaneamente. In questo modo i parametri vengono distinti dalle variabili di script locali, utilizzabili solo nello script in cui sono dichiarate. In ambiente di scripting i parametri fanno spesso parte di un'espressione CLEM e il relativo valore è specificato nello script stesso.

L'ambito di un parametro dipende dalla posizione in cui è impostato:

- i parametri impostati per uno stream possono essere impostati in uno script dello stream o nella finestra di dialogo delle proprietà dello stream e sono disponibili a tutti i nodi dello stream. Vengono visualizzati nell'elenco Parametri nel generatore di espressioni.
- I parametri impostati per una sessione possono essere impostati in uno script autonomo oppure nella finestra di dialogo dei parametri della sessione e sono disponibili per tutti gli stream utilizzati nella sessione corrente, ovvero per tutti gli stream elencati nella scheda Stream del riquadro dei manager.

È inoltre possibile impostare parametri per Supernodi. In questo caso, sono visibili solo ai nodi incapsulati all'interno del Supernodo. [Per ulteriori informazioni, vedere l'argomento Definizione dei parametri di un Supernodo in il capitolo 9 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Impostazione dei parametri negli script

È possibile impostare i parametri negli script tramite il comando `set` e la sintassi seguente:

```
set foodtype = pizza
```

Se non vengono dichiarati nello script nodi o variabili denominati `foodtype`, il comando crea un parametro denominato `foodtype` con il valore predefinito `pizza`.

Interfaccia utente. In alternativa, è possibile impostare o visualizzare i parametri tramite l'interfaccia utente scegliendo Proprietà stream oppure Imposta parametri sessione dal menu Strumenti. Queste finestre di dialogo consentono di specificare opzioni aggiuntive, quali il tipo di archiviazione, che non sono disponibili tramite gli script. [Per ulteriori informazioni, vedere l'argomento Impostazione di parametri per stream e sessione in il capitolo 5 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Riga di comando. È possibile impostare i parametri dalla riga di comando, che verranno creati come parametri di sessione.

Riferimento ai parametri negli script

È possibile fare riferimento ai parametri creati in precedenza racchiudendoli tra virgolette singole e inserendo come prefisso la stringa `$P`, per esempio `'$P-minvalue'`. È possibile anche fare semplicemente riferimento al nome del parametro, per esempio `minvalue`. Il valore di un parametro è sempre una stringa o un numero. Per esempio, è possibile fare riferimento al parametro `foodtype` e impostare un nuovo valore tramite la seguente sintassi:

```
set foodtype = pasta
```

È anche possibile fare riferimento ai parametri all'interno del contesto di un'espressione CLEM utilizzata in uno script. Lo script seguente è un esempio. Tale script imposta le proprietà di un nodo Seleziona in modo da includere i record con un valore di `Age` maggiore di quello specificato

dal parametro dello stream denominato `cutoff`. Il parametro viene utilizzato in un'espressione CLEM con la sintassi appropriata per CLEM, ovvero '`$P-cutoff`':

```
set :selectnode {
mode = "Include"
condition = "Age >= '$P-cutoff'"
}
```

In questo script viene utilizzato il valore di default per il parametro dello stream denominato `cutoff`. Per specificare un nuovo valore del parametro è possibile aggiungere la seguente sintassi posizionandola sopra alle specifiche del nodo Seleziona:

```
set cutoff = 50
```

Lo script risultante seleziona tutti i record in cui il valore `Age` è maggiore di 50.

Per ulteriori informazioni, vedere l'argomento [Parametri stream, sessione e Supernodo](#) in il capitolo 7 in *Manuale dell'utente di IBM SPSS Modeler 14.2*.

Controllo dell'esecuzione degli script

L'esecuzione degli script in genere elabora un'istruzione dopo l'altra. È tuttavia possibile ignorare questo ordine di esecuzione utilizzando un'istruzione condizionale `if` e numerosi tipi di cicli `for`, per esempio:

```
if s.maxsize > 10000 then
s.maxsize = 10000
connect s to :derive
endif
```

Il ciclo `for` può avere numerosi formati, per esempio:

```
for PARAMETER in LIST
STATEMENTS
endfor
```

Questo script esegue `STATEMENTS` una volta per ogni valore di `LIST` assegnato a `PARAMETER`, in base all'ordine dell'elenco. L'elenco non è inserito tra parentesi quadre e il relativo contenuto è rappresentato da costanti. Sono anche disponibili numerosi altri formati. [Per ulteriori informazioni, vedere l'argomento Comandi di script generali](#) in il capitolo 4 a pag. 30.

Operatori negli script

Oltre ai normali operatori CLEM, è possibile modificare le variabili di script locali (dichiarate utilizzando un comando `var`) tramite gli operatori "+" e "-". L'operatore + aggiunge un elemento all'elenco mentre l'operatore - rimuove un elemento. Di seguito è riportato un esempio:

```
var z      # create a new local variable
set z = [1 2 3] # set it to the list containing 1, 2, and 3
set z = z + 4 # add an element; z now equals [1 2 3 4]
```

Non è possibile utilizzare questi operatori con i parametri stream, Supernodo o sessione (definiti negli script utilizzando il comando `set`) né al di fuori degli script nelle espressioni CLEM generali, quale una formula in un nodo Nuovo campo.

Espressioni CLEM negli script

È possibile utilizzare espressioni, funzioni e operatori CLEM all'interno degli script di IBM® SPSS® Modeler; tuttavia, l'espressione di script non può contenere chiamate a funzioni `@`, funzioni data/ora e operazioni bitwise. Per le espressioni CLEM negli script sono inoltre valide le seguenti regole:

- I parametri specificati devono essere inseriti tra virgolette singole e devono avere il prefisso `$P-`.
- Le espressioni CLEM devono essere racchiuse tra virgolette. Se l'espressione CLEM contiene stringhe o nomi di campo tra virgolette, è necessario farli precedere da una barra rovesciata (`\`). [Per ulteriori informazioni, vedere l'argomento Sintassi degli script a pag. 18.](#)

Negli script è possibile utilizzare valori globali quali `GLOBAL_MEAN(Age)`, tuttavia nell'ambiente di script non è possibile utilizzare la funzione `@GLOBAL`.

Di seguito sono riportati alcuni esempi di espressioni CLEM utilizzate negli script:

```
set :balancenode.directives = {{1.3 "Age > 60"}}

set :fillernode.condition = "(Age > 60) and (BP = \"High\")"

set :derivenode.formula_expr = "substring(5, 1, Drug)"

set Flag:derivenode.flag_expr = "Drug = X"

set :selectnode.condition = "Age >= '$P-cutoff'"

set :derivenode.formula_expr = "Age - GLOBAL_MEAN(Age)"
```

Inserimento di commenti e continuazioni

Per contrassegnare i commenti e le continuazioni negli script vengono utilizzati i seguenti caratteri.

| Carattere | Utilizzo | Esempio |
|-----------|---|-----------------------------|
| # | Il simbolo del cancelletto indica un commento. Il resto della riga viene ignorato. | #Commento di una sola riga. |
| \ | Una barra rovesciata al termine di una riga indica che l'istruzione continua nella riga successiva. | Vedere l'esempio seguente. |

| Carattere | Utilizzo | Esempio |
|-----------|--|----------------------------|
| /* | La sequenza /* indica l'inizio di un commento. Tutte le parole fino al marcatore finale del commento */ verranno ignorate. | Vedere l'esempio seguente. |
| """ | Le stringhe letterali o i blocchi che comprendono interruzioni di linea, spazi o virgolette singole o doppie all'interno del blocco possono essere racchiuse tra virgolette triple. Per ulteriori informazioni, vedere l'argomento Blocchi di testo letterale a pag. 28. | |

Esempi

```
/* This is a
multi line
comment
*/
```

```
#following is a multi-line statement
set :fixedfilenode.fields = [{"Age" 1 3}\
{"Sex" 5 7} {"BP" 9 10} {"Cholesterol" 12 22}\
{"Na" 24 25} {"K" 27 27} {"Drug" 29 32}]
```

Blocchi di testo letterale

È possibile includere negli script i blocchi di testo letterale che includono spazi, tabulazioni e interruzioni di linea racchiudendoli in virgolette triple. Qualsiasi testo all'interno del blocco tra virgolette viene mantenuto come testo letterale, inclusi gli spazi, le interruzioni di linea e le virgolette singole e doppie incorporate. Non sono necessari caratteri di escape o di continuazione della riga.

Per esempio, tale tecnica può essere utilizzata per incorporare un insieme di direttive di ingrandimento dell'albero in uno script, come segue:

```
set :cartnode.tree_directives = """
Create Root_Node
Grow Node Index 0 Children 1 2 SplitOn ("DRUG",
Group ("drugA", "drugB", "drugC" )
Group ("drugY", "drugX" ))
End Tree
"""
```

Si tratta di un'operazione utile anche per percorsi e annotazioni, per esempio:

```
set :node.annotation = """This node was built to help identify which of the following indicators
Dairy
Fish
Vegetable
Meat
Pastries
Confectionary
```

is showing unusual sales behaviour""

IBM® SPSS® Modeler ignora un'interruzione di linea che segue il simbolo letterale di apertura.
Per esempio, il codice seguente è equivalente all'esempio precedente:

```
set :node.annotation = ""  
This node was built to help identify which of the following indicators  
Etc...  
""
```

Comandi di script

In questa sezione vengono riepilogati i comandi che è possibile utilizzare negli script di IBM® SPSS® Modeler, organizzati per tipo di oggetto. Per ulteriori informazioni sul linguaggio di script, vedere [il capitolo 3](#). Per ulteriori informazioni sulle proprietà di nodi, stream, progetti e Supernodi, vedere dal il capitolo 9 al il capitolo 22.

Comandi di script generali

Se non diversamente indicato, i seguenti comandi sono disponibili in tutti gli script locali, degli stream e dei Supernodi.

execute_all

```
execute_all
```

Esegue tutti i nodi terminali dello stream corrente.

```
open stream "c:/demos/druglearn.str"
execute_all
```

execute_script

```
execute_script
```

Solo script locali. Esegue lo script dello stream associato allo stream corrente. È limitato agli script locali, perché altrimenti lo script dello stream richiamerebbe se stesso.

```
open stream "c:/demos/mysample.str"
execute_script
```

exit

```
exit CODE
```

Esce dallo script corrente. Il codice di uscita può essere utilizzato per valutare lo script o la condizione di uno stream o di un nodo, per esempio:

```
create tablenode
create variablefilenode
connect :variablefilenode to :tablenode

set :variablefilenode.full_filename = "$CLEO_DEMOS/DRUG1n"
execute 'Table'

set param = value :tablenode.output at 1 1
```

```

if ^param = 23 then
  create derivenode
else exit 2
endif

```

for...endfor

Il comando `for...endfor` scorre un insieme di istruzioni basate su una condizione. Il comando può assumere numerosi formati, ma tutti seguono la stessa struttura generale.

```

for PARAMETER in LIST
  STATEMENTS
endfor

```

for PARAMETER in LIST. Esegue `STATEMENTS` una volta per ogni valore di `LIST` assegnato a `PARAMETER`, in base all'ordine dell'elenco. Per esempio, la proprietà `Filter.include` può essere impostata su `vero` per più campi, come illustrato di seguito:

```

for f in Age Sex
  set Filter.include.^f=true
endfor

```

for PARAMETER from N to M. Esegue `STATEMENTS` una volta per ogni numero intero compreso tra `N` e `M` (inclusi), per esempio:

```

for I from 1 to 5
  set :selectnode.condition = 'Age > ' >< (I * 10)
  execute :selectnode
endfor

```

for PARAMETER in_fields_to NODE. Esegue `STATEMENTS` una volta per ogni campo dal lato a monte di `NODE`. Per esempio, il comando seguente imposta la proprietà `include` su `vero` per tutti i campi, inclusi quelli precedentemente impostati su `falso`:

```

for f in_fields_to Filter
  set Filter.include.^f = "true"
endfor

```

Nota: nei casi in cui un nodo (per esempio Unione o Accodamento) possa disporre di più campi di input con lo stesso nome, questo metodo restituisce l'elenco dei campi a valle anziché di quelli a monte, per evitare i conflitti che potrebbero altrimenti verificarsi.

for PARAMETER in_fields_at NODE. Esegue `STATEMENTS` una volta per ogni campo risultante dal `NODE` specificato o a valle dello stesso. Pertanto, nel caso di un nodo Filtro vengono inclusi solo i campi passati, e il nodo non deve essere terminale perché in tal caso non verrebbe restituito alcun campo. Per esempio, diversamente dal precedente, lo script seguente non ha effetto perché il ciclo viene eseguito solo per i campi già impostati su `vero`:

```

for f in_fields_at Filter
  set Filter.include.^f = "true"
endfor

```

for PARAMETER in_models. Esegue STATEMENTS una volta per ogni insieme di modelli nella palette Modelli. Per esempio, lo script seguente inserisce ciascun modello della palette nello stream corrente. La variabile xpos viene utilizzata per evitare che i nodi si sovrappongano l'uno sull'altro nell'area di disegno dello stream.

```
var xpos
set xpos = 100
for m in_models
  set xpos = xpos + 100
  insert model ^m at ^xpos 100
endfor
```

for PARAMETER in_streams. *Solo script locali.* Esegue STATEMENTS una volta per ogni stream caricato (come elencato nella palette Stream). Se PARAMETER è la variabile speciale stream, nel ciclo verrà impostato lo stream corrente per STATEMENTS. Al termine del ciclo, verrà ripristinato il valore originale dello stream.

if...then...else...

```
if EXPR then
  STATEMENTS 1
else
  STATEMENTS 2
endif
```

Esegue STATEMENTS 1 se l'espressione specificata è vera e STATEMENTS 2 se l'espressione è falsa. La clausola else è facoltativa.

```
if :samplenode.use_max_size = true then
  set x = "yes"
else
  set x = "no"
endif
```

Comando set

```
set VARIABLE = EXPRESSION
set PARAMETER = EXPRESSION
set PROPERTY = EXPRESSION
```

Imposta il valore di una variabile script locale, di una variabile speciale, di un parametro o di una proprietà.

Impostazione delle variabili

Per impostare il valore di una variabile script locale, dichiarare innanzitutto la variabile utilizzando il comando var, per esempio:

```
var xpos
var ypos
set xpos = 100
```

```
set ypos = 100
```

Il valore della variabile può essere un'espressione CLEM valida nello script, un comando di script che restituisce un valore (per esempio `load`, `create` o `get`), oppure un valore letterale.

```
set xpos = ^xpos + 50
```

```
var x
set x = create typenode
```

```
var s
set s = get stream 'Druglearn'
```

Impostazione delle variabili speciali per fare riferimento agli oggetti

Le variabili speciali `node`, `stream`, `output` e `project` vengono utilizzate per fare riferimento all'oggetto "corrente" in ciascun contesto. Fatta eccezione per `project`, le variabili possono anche essere reimpostate per modificare il contesto corrente. A differenza delle altre variabili di script, non devono essere dichiarate prima con il comando `var` perché sono predefinite.

```
set node = create typenode
rename ^node as "mytypenode"
```

```
set output = get output :statisticsoutput
export output ^output as c:/myoutput.htm format html
```

Sebbene siano utili, queste variabili mostrano alcune sottili distinzioni nell'utilizzo, come dimostrato dal seguente esempio:

```
set stream = get stream 'Stream7'
set ^stream.execute_method = "Script"
save stream as c:/sample7.str
close stream
```

- La prima riga reimposta lo stream corrente o, più letteralmente, imposta il valore della variabile speciale `stream`. In altre parole, `stream` è una variabile anziché una parte del comando.
- La seconda riga utilizza questa variabile per impostare una proprietà per lo stream corrente (per ulteriori informazioni sulle proprietà, vedere la sezione seguente). L'accento circonflesso viene usato per indicare che `^stream` è il nome di una variabile anziché il nome di un oggetto quale un nodo. Senza l'accento circonflesso, il comando `set` ricerca un nodo denominato *stream*.
- Le ultime due righe salvano e chiudono lo stream corrente. Come nel caso precedente, `stream` è una variabile, ma in questo caso non viene utilizzato alcun accento circonflesso perché i comandi `save` e `close` in questo esempio possono applicarsi solo a uno stream. In generale, l'accento circonflesso viene utilizzato solo nei casi in cui la sua rimozione comporterebbe un riferimento ambiguo.

Riferimenti al progetto corrente. La variabile speciale `project` può essere utilizzata per fare riferimento al progetto corrente (vedere l'esempio seguente di impostazione delle proprietà del progetto). Il valore di `project` non può essere reimpostato perché è possibile avere solo un progetto aperto (e quindi corrente) alla volta.

Impostazione di parametri

I parametri stream, sessione e Supernodo possono essere impostati nello stesso modo delle variabili ma senza utilizzare il comando `var`.

```
set p = 1
set minvalue = 21
```

Nota: in termini pratici, se l'oggetto di un comando `set` non corrisponde al nome di una variabile dichiarata, una variabile speciale o un oggetto esistente quale un nodo, allora viene creato un parametro. [Per ulteriori informazioni, vedere l'argomento Parametri stream, sessione e Supernodo in il capitolo 3 a pag. 24.](#)

Impostazione delle proprietà di nodi, stream e progetti

È anche possibile impostare anche le proprietà di nodi, stream e progetti, per esempio:

```
set :variablefilenode.full_filename = "$CLEO_DEMOS/DRUG1n"

set ^stream.execute_method = "Script"

load project "C:/myproject.cpj"
set ^project.structure = Phase
```

Per un elenco completo delle proprietà disponibili per i nodi, gli stream e i progetti, vedere [Guida alle proprietà a pag. 113](#).

Impostazione di più proprietà. È possibile assegnare più espressioni alle proprietà dei nodi o di altri oggetti mediante una singola operazione. Tale metodo risulta utile quando è necessario apportare più modifiche a un nodo prima che venga definito il modello di dati. Il formato per l'impostazione di più proprietà è il seguente:

```
set NODE {
  NODEPROPERTY1 = EXPRESSION1
  NODEPROPERTY2 = EXPRESSION2
}
```

Per esempio:

```
set :samplenode {
  max_size = 200
  mode = "Include"
  sample_type = "First"
}

set ^project {
  summary = "Initial modeling work on the latest data"
  ordering = NameAddedType
}
```

Impostazione dei valori flag (veri e falsi). Nella lettura o scrittura di proprietà tipo flag, i valori vero e falso dovrebbero essere in minuscolo, per esempio:

```
set :variablefilenode.read_field_names = true
```

Nota: le variazioni, incluse Off, OFF, off, No, NO, no, n, N, f, F, falso, False, FALSE, oppure 0 sono riconosciute durante l'impostazione dei valori, ma in alcuni casi possono generare errori durante la lettura dei valori delle proprietà. Tutti gli altri valori vengono considerati come vero. L'utilizzo coerente di vero e falso in caratteri minuscoli consente di evitare confusioni.

Esempio: Impostazione delle proprietà dei nodi

Esistono molte proprietà specifiche dei nodi (denominate talvolta parametri di configurazione) utilizzate per impostare le opzioni delle finestre di dialogo dell'interfaccia utente per ogni nodo. Per esempio, per creare uno stream e specificare le opzioni per ogni nodo, è possibile utilizzare uno script simile al seguente: Per ulteriori informazioni sulle proprietà di nodi, stream, progetti e Supernodi, vedere dal capitolo 9 al capitolo 22.

```
create varfilenode at 100 100
set :varfilenode {
full_filename = "demos/drug1n"
read_field_names = true
}
create tablenode at 400 100
create samplenode connected between :varfilenode and :tablenode
set :samplenode {
max_size = 200
mode = "Include"
sample_type = "First"
}
create plotnode at 300 300
create derivenode connected between drug1n and :plotnode
set :derivnode {
new_name = "Ratio of Na to K"
formula_expr = "'Na' / 'K'"
}
set :plotnode {
x_field = 'Ratio of Na to K'
y_field = 'Age'
color_field = 'BP'
}
```

Comando var

```
var VARNAME
```

Dichiara una variabile script locale.

```
var my_node
set my_node = create distributionnode
rename ^my_node as "Distribution of Flag"
```

Le variabili si differenziano dai parametri, che è possibile impostare per una sessione, uno stream o un Supernodo e possono contenere solo stringhe o numeri. In termini pratici, se si imposta una variabile senza prima dichiararla utilizzando un comando VAR, viene creato un parametro stream,

sessione o Supernodo, in base al contesto dello script corrente. [Per ulteriori informazioni, vedere l'argomento Variabili script locali in il capitolo 3 a pag. 23.](#)

Oggetti nodo

I seguenti comandi di script sono disponibili per gli oggetti nodo.

create NODO

```
create NODE
create NODE at X Y
create NODE between NODE1 and NODE2
create NODE connected between NODE1 and NODE2
```

Crea un nodo del tipo specificato, per esempio:

```
create statisticsimportnode
```

Se lo si desidera, è anche possibile specificare le opzioni di posizione e connessione:

```
create featureselectionnode at 400 100
```

```
create typenode between :statisticsimportnode and :featureselectionnode
```

```
create selectnode connected between :typenode and :featureselectionnode
```

È possibile creare un nodo anche utilizzando le variabili al fine di evitare ambiguità. Nell'esempio seguente, viene creato un nodo Tipo e la variabile di riferimento *x* viene impostata in modo da contenere un riferimento a tale nodo Tipo Sarà quindi possibile utilizzare la variabile *x* per restituire l'oggetto a cui *x* fa riferimento (in questo caso, il nodo Tipo) ed effettuare ulteriori operazioni, quali la ridenominazione, il posizionamento o la connessione del nuovo nodo.

```
var x
set x = create typenode
rename ^x as "mytypenode"
position ^x at 200 200
var y
set y = create varfilenode
rename ^y as "mydatasource"
position ^y at 100 200
connect ^y to ^x
```

Nell'esempio precedente i due nodi vengono creati, rinominati, posizionati e infine connessi nell'area di disegno dello stream.

Figura 4-1
Nodi creati utilizzando le variabili



In alternativa, è possibile utilizzare la variabile speciale (predefinita) `node` in modo analogo alle variabili `x` e `y` nell'esempio precedente. In tal caso, la variabile non deve essere dichiarata utilizzando il comando `var` (poiché è predefinita) e lo script risultante può rivelarsi di più facile lettura.

```
set node = create typenode
rename ^node as "mytypenode"
position ^node at 200 200
set node = create varfilenode
rename ^node as "mydatasource"
position ^node at 100 200
connect mydatasource to mytypenode
```

Nota: è possibile riutilizzare variabili speciali, quali `node`, per fare riferimento a più nodi. Utilizzare semplicemente il comando `set` per reimpostare l'oggetto a cui la variabile fa riferimento. [Per ulteriori informazioni, vedere l'argomento Impostazione dell'oggetto corrente in il capitolo 3 a pag. 21.](#)

Duplicazione dei nodi. È possibile anche utilizzare il comando `duplicate` per duplicare un nodo esistente. [Per ulteriori informazioni, vedere l'argomento duplicate NODO a pag. 38.](#)

connect NODO

```
connect NODE1 to NODE2
connect NODE1 between NODE2 and NODE3
```

Connette il NODO1 agli altri nodi come specificato.

```
connect :statisticsimportnode to :typenode
connect :selectnode between :typenode and :featureselectionnode
```

delete NODO

```
delete NODO
```

Elimina il nodo specificato dallo stream corrente.

```
delete :statisticsimportnode
delete DRUG1N:variablefilenode
```

disable NODO

```
disable NODO
```

Disabilita il nodo specificato nello stream corrente e il nodo viene ignorato durante l'esecuzione dello stream. In questo modo non è necessario rimuovere o ignorare il nodo e questo può restare connesso agli altri nodi. Le impostazioni del nodo possono ancora essere modificate; tuttavia, le modifiche non verranno applicate finché il nodo non viene nuovamente attivato.

```
disable :statisticsimportnode
```

```
disable DRUG1N:variablefilenode
```

disconnect NODO

```
disconnect NODE  
disconnect NODE1 from NODE2  
disconnect NODE1 between NODE2 and NODE3
```

Disconnette il nodo specificato da tutti gli altri nodi (di default) o dai nodi specifici come indicato.

```
disconnect :typenode
```

```
disconnect :typenode from :selectnode
```

duplicate NODO

```
duplicate NODO as NUOVONOME
```

Crea un nuovo nodo come duplicato del nodo specificato. Se lo si desidera, è possibile specificare anche la posizione in termini assoluti o relativi.

```
duplicate :derivenode as flag1 at 100 400
```

```
duplicate flag1 as flag2 connected between flag1 and flag3
```

enable NODO

```
enable NODO
```

Abilita un nodo precedentemente disabilitato nello stream corrente, con il risultato che il nodo viene incluso nell'esecuzione dello stream. Se sono state modificate le impostazioni del nodo mentre questo era disabilitato, le modifiche apportate saranno ora applicate.

```
enable :statisticsimportnode
```

```
enable DRUG1N:variablefilenode
```

execute NODO

```
execute NODO
```

Esegue il nodo specificato, per esempio:

```
execute :neuralnetworknode
```

Se non si tratta di un nodo terminale, l'esecuzione equivale all'opzione del menu di scelta rapida Esegui da qui.

Per eseguire tutti i nodi terminali dello stream corrente:

```
execute_all
```

Solo script locali. Per eseguire lo script dello stream associato allo stream corrente:

```
execute_script
```

Nota: per eseguire script associati a stream diversi è possibile impostare lo stream come stream corrente oppure utilizzare il comando `with`. [Per ulteriori informazioni, vedere l'argomento Utilizzo di più stream in il capitolo 3 a pag. 23.](#)

export NODO as FILE

```
export node NODE in DIRECTORY format FORMAT  
export node NODE as FILE format FORMAT
```

Esportazione PMML. Per esportare un modello generato in formato PMML:

```
export Drug as c:/mymodel.txt format pmml
```

Esportazione SQL. Per esportare un modello generato in formato SQL:

```
export Drug in c:/mymodels format sql
```

```
export Drug as c:/mymodel.txt format sql
```

Dettagli del nodo. Per esportare i dettagli del nodo in formato HTML o testo:

```
export Drug as c:\mymodel.htm format html
```

```
export Drug as c:\mymodel.txt format text
```

Riepilogo del nodo. Per esportare il riepilogo del nodo in formato HTML o testo:

```
export Drug summary in c:/mymodels format html
```

```
export Drug summary as c:/mymodel.txt format text
```

```
export 'assocapriori' as 'C:/temp/assoc_apriori' format html
```

flush NODO

```
flush NODO
```

Svuota la cache del nodo specificato o di tutti i nodi dello stream. Se la cache non è attivata o non è piena per un dato nodo, l'istruzione non avrà alcun effetto.

```
flush :mergenode
```

Per svuotare tutti i nodi dello stream corrente:

```
flush_all
```

get node NODO

```
get node NODO
```

Ottiene un riferimento a un nodo esistente. Può essere un modo utile per assicurare riferimenti non ambigui ai nodi.

```
var mynode  
set mynode = get node flag1:derivenode  
position ^mynode at 400 400
```

load node NOMEFILE

```
load node NOMEFILE
```

Carica un nodo salvato nello stream corrente.

```
load node c:/mynode.nod
```

position NODO

```
position NODE at X Y  
position NODE between NODE1 and NODE2  
position NODE connected between NODE1 and NODE2
```

Posiziona un nodo nell'area di disegno dello stream in termini assoluti o relativi. Se lo si desidera, è anche possibile specificare le opzioni di connessione:

```
position DRUG1n:variablefilenode at 100 100
```

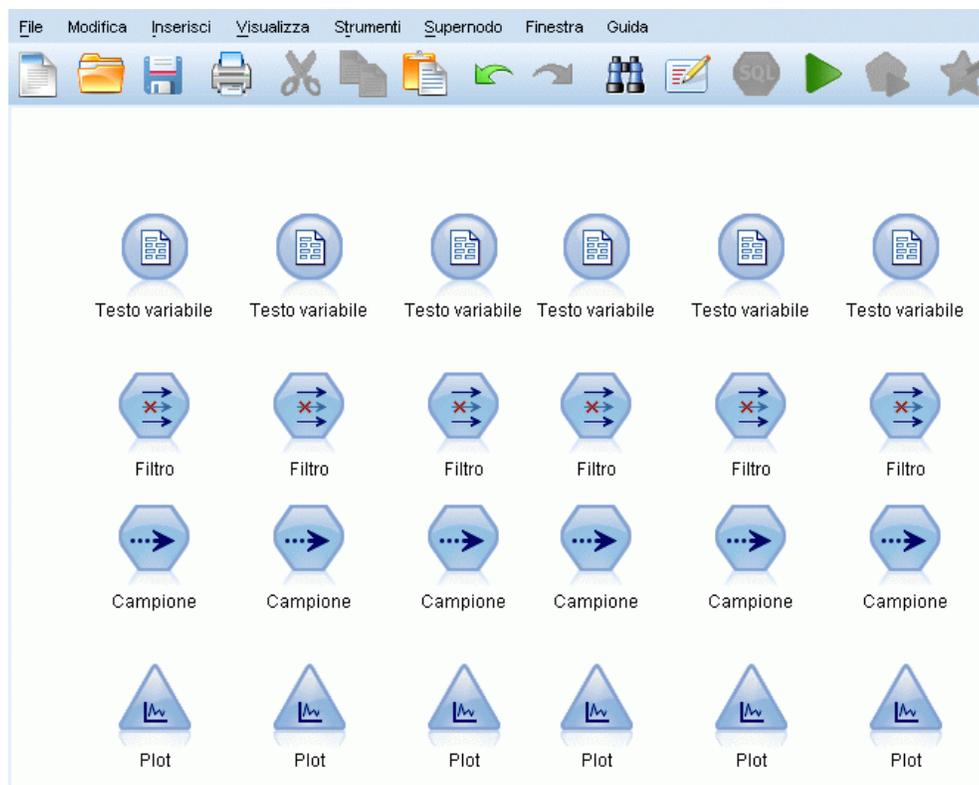
```
position Drug:net between DRUG2n and analysis
```

```
position :typenode connected between :variablefilenode and :tablenode
```

Coordinate di posizionamento

Per il posizionamento dei nodi nell'area di disegno dello stream viene utilizzata una griglia x - y invisibile. Per le coordinate della griglia x - y , è possibile fare riferimento all'immagine seguente.

Figura 4-2
Nodi creati nella posizione specificata dalle coordinate x-y



rename NODO as NUOVONOME

```
rename NODO as NUOVONOME
```

Rinomina il nodo specificato.

```
rename :derivenode as 'Flag1'
```

```
rename :varfilenode as 'testdata'
```

retrieve node REPOSITORY_PATH

```
retrieve node REPOSITORY_PATH {label LABEL | version VERSION}
```

Recupera il nodo specificato da IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
retrieve node "/samples/drugtypenode"
```

save node NODO as NOMEFILE

```
save node NODO as NOMEFILE
```

Salva il nodo specificato.

```
save node :statisticsimportnode as c:/mynode.nod
```

store node NODO as PERCORSO_REPOSITORY

```
store node NODO as PERCORSO_REPOSITORY {label ETICHETTA}
```

Memorizza un nodo in IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
store node DRUG1n as "/samples/drug1ntypenode"
```

```
store node :typenode as "/samples/drugtypenode"
```

Oggetti modello

I seguenti comandi di script sono disponibili per gli oggetti modello.

Nomi degli insiemi di modelli

Agli insiemi di modelli (detti anche modelli generati) può essere fatto riferimento per tipo, come avviene per gli oggetti nodo e output. Le seguenti tabelle elencano i nomi di riferimento degli oggetti modello.

Si noti che questi nomi vengono utilizzati specificamente per fare riferimento agli insiemi di modelli nella palette Modelli (nell'angolo superiore destro della finestra di IBM® SPSS® Modeler). Per fare riferimento ai nodi modello aggiunti a uno stream ai fini del calcolo del punteggio, viene utilizzato un insieme diverso di nomi preceduti dal prefisso apply.... [Per ulteriori informazioni, vedere l'argomento Proprietà dei nodi degli insiemi di modelli in il capitolo 17 a pag. 243.](#)

Per esempio, lo script seguente aggiunge un insieme di modelli allo stream corrente, lo connette a un nodo Tipo e crea ed esegue un nodo Tabella. Si noti il nome diverso utilizzato per inserire il modello dalla palette, distinto dal nome utilizzato per fare riferimento al nodo modello "apply" una volta aggiunto allo stream (:featureselection e :applyfeatureselectionnode).

```
insert model :featureselection at 150 250
connect Type to :applyfeatureselectionnode
create tablenode at 250 250
connect :applyfeatureselectionnode to :tablenode
execute :tablenode
```

Nota: questo è solo un esempio. In circostanze normali, si consiglia di fare riferimento ai modelli per nome e per tipo in modo da evitare confusioni (per esempio, response_01:featureselection).

Nomi degli insiemi di modelli (palette Modelli)

| Nome modello | Model |
|------------------|--|
| anomalydetection | Anomalia |
| apriori | Apriori |
| autoclassifier | Classificatore automatico |
| autocluster | Cluster automatico |
| autonumeric | Numerico automatico |
| bayesnet | Rete bayesiana |
| c50 | C5.0 |
| carma | Carma |
| C&R tree | C&R Tree |
| chaid | CHAID |
| coxreg | regressione di Cox |
| decisionlist | Elenco decisionale |
| discriminante | Discriminante |
| fattore | Fattoriale/PCA |
| featureselection | Selezione funzioni |
| genlin | Regressione lineare generalizzata |
| kmeans | K-Means |
| knn | <i>k</i> -vicino più vicino |
| kohonen | Kohonen |
| linear | Lineare |
| logreg | Regressione logistica |
| neuralnetwork | Rete neurale |
| quest | QUEST |
| regressione | Regressione lineare |
| sequence | Sequenza |
| slrm | Modello di risposta di autoapprendimento |
| statisticsmodel | Modello IBM® SPSS® Statistics |
| svm | Support vector machine |
| timeseries | Serie storiche |
| twostep | TwoStep |

Nomi degli insiemi di modelli (palette Modelli in-database)

| Nome modello | Model |
|---------------|--------------------------------|
| db2imassoc | Associazione IBM ISW |
| db2imcluster | Raggruppamento cluster IBM ISW |
| db2imreg | Regressione IBM ISW |
| db2imsequence | Sequenza IBM ISW |

| Nome modello | Model |
|-------------------|----------------------------------|
| db2imtree | Albero decisionale IBM ISW |
| msassoc | Regole di associazione Microsoft |
| msbayes | Naive Bayes Microsoft |
| mscluster | Raggruppamento cluster MS |
| mslogistic | Regressione logistica MS |
| msneuralnetwork | Rete neurale MS |
| msregression | Regressione lineare MS |
| mssequencecluster | Cluster di sequenze MS |
| mstimeseries | Serie storica Microsoft |
| mstree | Albero decisionale MS |
| oraabn | Bayes adattivi Oracle |
| oraai | Oracle AI |
| oraapriori | Apriori Oracle |
| oradecisiontree | Albero decisionale Oracle |
| oraglm | GLM Oracle |
| orakmeans | k -Means Oracle |
| oramdl | MDL Oracle |
| oranb | Bayes naive Oracle |
| oranmf | NMF Oracle |
| oraocluster | O-Cluster Oracle |
| orasvm | SVM Oracle |

Per evitare nomi di modelli duplicati

Quando si utilizzano gli script per manipolare i modelli generati, ricordare che la duplicazione dei nomi dei modelli può comportare riferimenti ambigui. Per evitare questo problema, si consiglia di richiedere nomi univoci per i modelli generati durante lo script.

Per impostare le opzioni per i nomi di modelli duplicati:

- ▶ Dai menu, scegliere:
Strumenti > Opzioni utente
- ▶ Fare clic sulla scheda Notifiche.
- ▶ Selezionare Sostituisci modello precedente per limitare la duplicazione dei nomi per i modelli generati.

delete model MODELLO

delete model MODELLO

Elimina un modello specificato (o cancella tutti i modelli) dalla palette degli insiemi di modelli.

```
delete model Drug
```

```
delete model Drug:c50
```

Per eliminare l'ultimo modello inserito dallo script corrente:

```
delete last model
```

Affinché quest'ultima istruzione funzioni, è necessario che l'istruzione `insert model` sia stata eseguita almeno una volta durante l'esecuzione dello script corrente.

Per eliminare tutti gli insiemi di modelli dalla palette Modelli:

```
clear generated palette
```

export model MODELLO as FILE

```
export model MODEL in DIRECTORY format FORMAT
```

```
export model MODEL as FILE format FORMAT
```

Esportazione PMML. Per esportare il modello generato in formato PMML:

```
export model Drug in c:/mymodels format pmml
```

```
export model Drug as c:/mymodel.xml format pmml
```

Per ulteriori informazioni, vedere l'argomento [Importazione ed esportazione di modelli come PMML in il capitolo 10 in *Manuale dell'utente di IBM SPSS Modeler 14.2*](#).

Esportazione SQL. Per esportare un modello generato in formato SQL:

```
export Drug in c:/mymodels format sql
```

```
export Drug as c:/mymodel.txt format sql
```

Nota: l'esportazione SQL è disponibile solo per determinati tipi di modelli. Per ulteriori informazioni, vedere l'argomento [Esplorazione degli insiemi di modelli in il capitolo 3 in *IBM SPSS Modeler 14.2 Nodi Modelli*](#).

Dettagli del modello. Per esportare i dettagli del modello (come visualizzati nella scheda Modello quando si sfoglia l'insieme di modelli) in formato HTML o testo:

```
export model Drug as c:\mymodel.htm format html
```

```
export model Drug as c:\mymodel.txt format text
```

Nota: Questi formati non sono disponibili per i modelli che non dispongono di una scheda Modello.

Riepilogo del modello. Per esportare il riepilogo del modello (scheda Riepilogo quando si sfoglia l'insieme di modelli) in formato HTML o testo:

```
export model Drug summary in c:/mymodels format html
```

```
export model Drug summary as c:/mymodel.txt format text
```

```
export model 'assocapriori' as 'C:/temp/assoc_apriori' format html
```

Figura 4-3
Scheda modello di associazione esportata come HTML

| | Consequente | Antecedente | Supporto | Confidenza |
|----|---------------|--------------------------|----------|------------|
| 1 | frozenmeal | beer and cannedveg | 16,7 | 87,425 |
| 2 | cannedveg | beer and frozenmeal | 17,0 | 85,882 |
| 3 | beer | frozenmeal and cannedveg | 17,3 | 84,393 |
| 4 | frozenmeal | beer | 29,3 | 58,02 |
| 5 | cannedveg | frozenmeal | 30,2 | 57,285 |
| 6 | frozenmeal | cannedveg | 30,3 | 57,096 |
| 7 | cannedveg | beer | 29,3 | 56,997 |
| 8 | beer | frozenmeal | 30,2 | 56,291 |
| 9 | beer | cannedveg | 30,3 | 55,116 |
| 10 | wine | confectionery | 27,6 | 52,174 |
| 11 | confectionery | wine | 28,7 | 50,174 |

insert model MODELLO

```
insert model MODEL
insert model MODEL at X Y
insert model MODEL between NODE1 and NODE2
insert model MODEL connected between NODE1 and NODE2
```

Aggiunge il modello allo stream corrente. Se lo si desidera, è anche possibile specificare le opzioni di posizione e connessione.

```
insert model Kohonen between :typenode and :analysisnode
```

```
insert model Drug:neuralnetwork connected between 'Define Types' and 'Analysis'
```

load model NOMEFILE

```
load model NOMEFILE
```

Carica un modello salvato nella palette Modelli.

```
load model c:/mymodel.gm
```

retrieve model REPOSITORY_PATH

```
retrieve model REPOSITORY_PATH {label LABEL | version VERSION}
```

Recupera un modello salvato da IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
retrieve model "/my folder/Kohonen.gm"
```

save model MODELLO as NOMEFILE

```
save model MODELLO as NOMEFILE
```

Salva il modello specificato come file di modello generato.

```
save model Drug as c:/mymodel.gm
```

store model MODELLO as PERCORSO_REPOSITORY

```
store model MODELLO as PERCORSO_REPOSITORY {label ETICHETTA}
```

Archivia il modello specificato in IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
store model Kohonen as "/my folder/Kohonen.gm"
```

L'estensione (*.gm) è facoltativa ma deve essere usata in modo coerente quando si memorizza e si recupera il modello. Per esempio, se viene memorizzato semplicemente come "Kohonen", il modello deve essere recuperato con lo stesso nome. In altre parole l'estensione, se è utilizzata, fa semplicemente parte del nome del modello.

Oggetti stream

I seguenti comandi di script sono disponibili per gli oggetti stream.

create stream DEFAULT_FILENAME

```
create stream DEFAULT_FILENAME
```

Solo script locali. Crea un nuovo stream nella memoria con il nome specificato. Lo stream non viene salvato automaticamente.

```
create stream 'Druglearn'
```

close STREAM

```
close STREAM
```

Solo script locali. Chiude lo stream specificato.

Per chiudere lo stream corrente, digitare il comando utilizzando caratteri minuscoli, come illustrato di seguito:

```
close stream
```

Script locali

Se si lavora con più stream, ricordare che `stream` (in caratteri minuscoli, come illustrato) è effettivamente una variabile speciale utilizzata per fare riferimento allo stream corrente. Per chiudere uno stream diverso, è possibile reimpostare il valore di questa variabile:

```
set stream = get stream 'Stream5'  
close stream
```

In alternativa, è possibile specificare qualsiasi variabile dichiarata che fa riferimento a uno stream, per esempio:

```
var s  
set s = get stream 'Stream2'  
save s as c:/stream2.str  
close s
```

Infine, è possibile riassegnare temporaneamente lo stream corrente utilizzando il comando `with stream`:

```
with stream 'Stream1'  
close stream  
endwith
```

clear stream

```
clear stream
```

Rimuove tutti i nodi dallo stream corrente.

get stream STREAM

```
get stream STREAM
```

Solo script locali. Utilizzato per ottenere un riferimento allo stream specificato, che può essere assegnato a una variabile locale (o alla variabile speciale `stream`). Lo stream specificato deve essere già aperto.

```
var s  
set s = get stream 'Druglearn'  
close s
```

load stream NOMEFILE

```
load stream NOMEFILE
```

Solo script locali. Aggiunge lo stream specificato all'area di disegno senza cancellare i nodi dallo stream corrente.

```
load stream "c:/demos/druglearn.str"
```

Comando open stream e comando load stream. Il comando load stream aggiunge lo stream specificato all'area di disegno senza cancellare i nodi dallo stream corrente. Questo comando era utilizzato più frequentemente nelle versioni precedenti di IBM® SPSS® Modeler ed è stato superato nelle versioni più recenti dalla possibilità di aprire, gestire e copiare nodi tra più stream.

Open stream NOMEFILE

```
Open stream NOMEFILE
```

Solo script locali. Apre lo stream specificato.

```
open stream "c:/demos/druglearn.str"
```

retrieve stream REPOSITORY_PATH

```
retrieve stream REPOSITORY_PATH {label LABEL | version VERSION}
retrieve stream URI [{#m.marker | #l.label}]
```

Recupera lo stream specificato da IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
retrieve stream "/myfolder/druglearn.str"
```

```
retrieve stream "spsscr:///models/drug%20model.gm#m.0:2005-10-12%2014:15:41.281"
```

save STREAM as NOMEFILE

```
save STREAM
save STREAM as FILENAME
```

Per salvare le modifiche apportate allo stream corrente (supponendo che sia stato salvato precedentemente), digitare il comando utilizzando caratteri minuscoli, come illustrato di seguito:

```
save stream
```

Per salvare uno stream per la prima volta con un nuovo nome file:

```
create stream nifty
create featureselectionnode
save stream as c:/nifty.str
```

Script locali

Se si lavora con più stream in uno script locale, ricordare che `stream` (in caratteri minuscoli, come illustrato) è effettivamente una variabile speciale utilizzata per fare riferimento allo stream corrente. Per salvare uno stream diverso, è possibile reimpostare il valore di questa variabile:

```
set stream = get stream 'Stream5'  
save stream
```

In alternativa, è possibile specificare qualsiasi variabile dichiarata che fa riferimento a uno stream, per esempio:

```
var s  
set s = get stream 'Stream2'  
save s as c:/stream2.str  
close s
```

Infine, è possibile riassegnare temporaneamente lo stream corrente utilizzando il comando `with stream`:

```
with stream 'Stream1'  
save stream  
endwith
```

[Per ulteriori informazioni, vedere l'argomento Utilizzo di più stream in il capitolo 3 a pag. 23.](#)

store stream as REPOSITORY_PATH

```
store stream as REPOSITORY_PATH {label LABEL}  
store stream as URI [#l.label]
```

```
store stream as "/folder_1/folder_2/mystream.str"
```

Archivia lo stream corrente in IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
store stream as "/folder_1/folder_2/druglearn.str"  
store stream as "spsscr:///folder_1/folder_2/mystream.str"
```

Script locali

Se si lavora con più stream in uno script locale, ricordare che `stream` (in caratteri minuscoli, come illustrato) è effettivamente una variabile speciale utilizzata per fare riferimento allo stream corrente. Per memorizzare uno stream diverso, è possibile reimpostare il valore di questa variabile:

```
set stream = get stream 'Stream5'  
store stream as "/folder_1/mystream.str"
```

In alternativa, è possibile specificare qualsiasi variabile dichiarata che faccia riferimento a uno stream oppure è possibile riassegnare temporaneamente lo stream corrente mediante il comando `with stream`:

```
with stream 'Stream6'  
  store stream as "/folder_1/mystream.str"  
endwith
```

with stream STREAM

```
with stream STREAM  
  STATEMENTS  
endwith
```

Solo script locali. Esegue `STATEMENTS` con lo `STREAM` specificato impostato come stream corrente. Dopo che le istruzioni sono state eseguite, verrà ripristinato lo stream corrente originale.

```
with stream 'druglearn'  
  create typenode  
  execute_script  
endwith
```

Oggetti del progetto

I seguenti comandi di script sono disponibili per gli oggetti progetto.

L'estensione (`*.cpj`) è facoltativa ma deve essere usata in modo coerente quando si memorizza e si recupera un determinato progetto.

execute_project

```
execute_project
```

Genera il report di default del progetto corrente.

load project NOMEFILE

```
load project NOMEFILE
```

Apri il progetto specificato.

```
load project "C:/clemdata/DrugData.cpj"  
set ^project.summary="Initial modeling work on the latest data."  
set ^project.ordering=NameAddedType  
execute_project
```

retrieve project REPOSITORY_PATH

```
retrieve project REPOSITORY_PATH {label LABEL | version VERSION}
```

Recupera un progetto da IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
retrieve project "/CRISPDM/DrugExample.cpj"
```

save project as NOMEFILE

```
save project  
save project as FILENAME
```

Salva il progetto corrente.

store project as REPOSITORY_PATH

```
store project as REPOSITORY_PATH {label ETICHETTA}
```

Archivia il progetto corrente in IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
store project as "/CRISPDM/DrugExample.cpj"
```

Oggetti stato

Uno stato salvato può essere caricato utilizzando il comando `load state`.

load state NOMEFILE

```
load state NOMEFILE
```

Carica lo stato specificato.

```
load state "c:/data/myproject.cst"
```

Oggetti risultato

È possibile accedere ai risultati utilizzando il comando `value`.

value RESULT

```
value RESULT at ROW COLUMN
```

I nodi terminali includono un parametro di sola lettura denominato `output` che consente di accedere all'oggetto generato più recentemente. Per i nodi che producono output tabulari in righe e colonne, ciò consente di accedere al valore di una cella specificata, per esempio:

```
execute :tablenode  
set last_row = :tablenode.output.row_count
```

```

set last_column = :tablenode.output.column_count
set last_value = value :tablenode.output at ^last_row ^last_column
var myresults
set myresults = open create 'C:/myresults.txt'
write myresults 'The value in the last cell is ' >< ^last_value

```

La distanza di riga e colonna è calcolata da 1. Se l'oggetto output non esiste, viene restituito un errore.

Proprietà degli oggetti risultato

Le seguenti proprietà sono comuni agli oggetti risultato (quali i risultati Tabella e Matrice) che includono i dati in righe e colonne:

| Proprietà | Descrizione |
|--------------|---|
| row_count | Restituisce il numero di righe dei dati. |
| column_count | Restituisce il numero di colonne dei dati |

Oggetti file

I seguenti comandi di script sono disponibili per gli oggetti file.

close FILE

```
close FILE
```

Questa istruzione chiude il file specificato.

open FILE

```
open create FILENAME
open append FILENAME
```

Queste istruzioni aprono il file specificato.

- **create.** Crea il file se non esiste oppure lo sovrascrive se esiste.
- **append.** Accoda a un file esistente. Genera un errore se il file non esiste.

Verrà restituito l'handle del file aperto.

```

var file
set file = open create 'C:/script.out'
for I from 1 to 3
  write file 'Stream ' >< I
endfor
close file

```

write FILE

```
write FILE TEXT_EXPRESSION
writeln FILE TEXT_EXPRESSION
```

Le espressioni sopra riportate scrivono l'espressione di testo nel file. La prima istruzione scrive il testo, mentre la seconda scrive anche una nuova riga dopo che è stata scritta l'espressione. Se FILE non è un oggetto file aperto, verrà generato un errore.

```
var file
set file = open create 'C:/hello.txt'
writeln file 'Hello'
writeln file 'World'
write file 'Would you like to play a game?'
close file
```

Oggetti di output

I seguenti comandi di script sono disponibili per gli oggetti di output.

Nomi dei tipi di output

La seguente tabella elenca tutti i tipi di oggetti output e i nodi che li creano. Per un elenco completo dei formati di esportazione disponibili per ciascun tipo di oggetto di output, vedere la descrizione delle proprietà per il nodo che crea il tipo di output, disponibile in [il capitolo 15, Proprietà dei nodi Grafici](#), e [il capitolo 19, Proprietà dei nodi Output](#).

| Tipo di oggetto di output | Nodo |
|----------------------------------|---|
| analysisoutput | Analisi |
| collectionoutput | Raccolta |
| dataauditoutput | Comandi di manipolazione dei nodi |
| distributionoutput | Distribuzione |
| evaluationoutput | Valutazione |
| histogramoutput | Histogram |
| matrixoutput | Matrice |
| meansoutput | Medie |
| multiplotoutput | Plot multiplo |
| plotoutput | Grafico |
| qualityoutput | Qualità |
| reportdocumentoutput | Questo tipo di oggetto non proviene da un nodo, ma si tratta dell'output creato da un report progetto |
| reportoutput | Report |
| statisticsprocedureoutput | StatisticsOutput |
| statisticsoutput | Statistiche |
| tableoutput | Tabella |
| timeplotoutput | Plot tempo |
| weboutput | Web |

delete output OUTPUT

```
delete output OUTPUT
```

Elimina l'output specificato dalla palette dei manager. Per esempio:

```
delete output :statisticsoutput
```

Per eliminare tutti gli elementi dell'output dalla palette dei manager:

```
clear outputs
```

export output OUTPUT

```
export output OUTPUT as FILE format FORMATO
```

Esporta l'output nel formato specificato. Si noti che i formati disponibili dipendono dal tipo di output ma devono rispecchiare quelli disponibili nel menu Esporta quando si sfoglia l'output specificato.

```
export output :statisticsoutput as "C:/output/statistics.html" format html
export output :matrixoutput as "C:/output/matrix.csv" format delimited
export output :tableoutput as "C:/output/table.tab" format transposed formatted
```

get output OUTPUT

```
get output OUTPUT
```

Ottiene un riferimento a un output specificato. Per esempio, è possibile utilizzare un ciclo per ottenere una serie di oggetti di output ed esportarne uno alla volta.

```
execute_all
for item in statisticsoutput matrixoutput tableoutput
var theoutput
set theoutput = get output :^item
set filename = 'c:/><^item ><'.htm'
export output ^theoutput as ^filename format html
endfor
```

load output NOMEFILE

```
load output NOMEFILE
```

Carica l'output specificato.

```
load output 'c:/matrix.cou'
```

retrieve output REPOSITORY_PATH

```
retrieve output REPOSITORY_PATH {label LABEL | version VERSION}
```

Recupera l'output specificato da IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
retrieve output "/results/mytable"
```

save output OUTPUT as FILENAME

```
save output as FILENAME
```

Salva l'output specificato.

```
save output :matrixoutput as 'c:/matrix.cou'
```

store output OUTPUT as REPOSITORY_PATH

```
store output OUTPUT as REPOSITORY_PATH {label LABEL}
```

Archivia l'output specificato in IBM® SPSS® Collaboration and Deployment Services Repository. [Per ulteriori informazioni, vedere l'argomento Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58.](#)

```
store output "Data Audit of [6 fields]" as "/my folder/My Audit"
```

```
store output :tableoutput as "/results/mytable"
```

Suggerimenti per gli script

Questa sezione contiene una panoramica dei suggerimenti e delle tecniche di utilizzo degli script, incluse tecniche relative a modifica dell'esecuzione dello stream, utilizzo di una password codificata in uno script e accesso a oggetti in IBM® SPSS® Collaboration and Deployment Services Repository.

Modifica dell'esecuzione dello stream

Durante l'esecuzione di uno stream, i relativi nodi terminali vengono eseguiti in base all'ordine ottimale per la situazione di default. In alcuni casi, è tuttavia preferibile utilizzare un ordine di esecuzione diverso. Per modificare l'ordine di esecuzione di uno stream, eseguire la seguente procedura nella scheda Script della finestra di dialogo Proprietà stream:

- ▶ Iniziare con uno script vuoto.
- ▶ Fare clic sul pulsante Accoda lo script di default nella barra degli strumenti per aggiungere lo script dello stream di default.
- ▶ Modificare l'ordine delle istruzioni contenute nello script dello stream di default in base all'ordine in cui si desidera che vengano eseguite le istruzioni.

Esecuzione di cicli sui nodi

È possibile utilizzare un ciclo for assieme alla proprietà `^stream.nodes` per eseguire cicli su tutti i nodi di uno stream. Per esempio, lo script seguente esegue il ciclo su tutti i nodi e cambia da minuscoli a maiuscoli i nomi dei campi di tutti i nodi Filtro.

Questo script può essere utilizzato in qualsiasi stream che dispone di un nodo Filtro, anche se non viene effettivamente filtrato alcun campo. È sufficiente aggiungere un nodo Filtro che passi tutti i campi per far sì che tutti i nomi dei campi diventino maiuscoli.

```
var my_node
var loop_me
var var_name

for my_node in ^stream.nodes
  if ^my_node.node_type = filternode then
    for loop_me in _fields_to ^my_node:filternode
      set var_name = lowertoupper(^my_node:filternode.new_name.^loop_me)
      set ^my_node:filternode.new_name.^loop_me = ^var_name
    endfor
  else
  endif
endifor
```

Lo script esegue il ciclo su tutti i nodi dello stream corrente, restituiti dalla proprietà `^stream.nodes`, e verifica se ciascun nodo sia un nodo Filtro. In caso affermativo, lo script esegue il ciclo su tutti i campi del nodo e utilizza la funzione `lowertoupper()` per modificare i nomi in lettere maiuscole.

Suggerimento: per far diventare minuscoli i nomi dei campi, utilizzare la funzione `uppertolower()`.

Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository

Nota: per l'accesso a un repository IBM® SPSS® Collaboration and Deployment Services è necessaria una licenza distinta. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/deployment/cds/>

Se si dispone di una licenza per IBM® SPSS® Collaboration and Deployment Services Repository, è possibile archiviare, recuperare, bloccare e sbloccare oggetti dal repository tramite comandi script. Il repository consente di gestire il ciclo di vita dei modelli di data mining e gli oggetti predittivi correlati nel contesto delle applicazioni, degli strumenti e delle soluzioni aziendali esistenti. [Per ulteriori informazioni, vedere l'argomento Informazioni su IBM SPSS Collaboration and Deployment Services Repository in il capitolo 9 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Connessione a IBM SPSS Collaboration and Deployment Services Repository

Per accedere al repository, è innanzitutto necessario impostare una connessione valida al repository stesso, tramite il menu Strumenti dell'interfaccia utente di IBM® SPSS® Modeler o mediante la riga di comando. ([Per ulteriori informazioni, vedere l'argomento Argomenti per la connessione a IBM SPSS Collaboration and Deployment Services Repository in il capitolo 7 a pag. 72.](#))

Archiviazione e recupero di oggetti

All'interno di uno script, i comandi `retrieve` e `store` consentono di accedere a vari oggetti, inclusi stream, modelli, output, nodi e progetti. La sintassi è la seguente:

```
store object as REPOSITORY_PATH {label LABEL}
store object as URI [#l.label]
```

```
retrieve object REPOSITORY_PATH {label LABEL | version VERSION}
retrieve object URI [{#m.marker | #l.label}]
```

`REPOSITORY_PATH` fornisce la posizione dell'oggetto nel repository. Il percorso deve essere racchiuso tra virgolette ed è necessario utilizzare le barre (/) come delimitatori. Il percorso non fa distinzione tra maiuscole e minuscole.

```
store stream as "/folder_1/folder_2/mystream.str"
store model Drug as "/myfolder/drugmodel"
store model Drug as "/myfolder/drugmodel.gm" label "final"
store node DRUG1n as "/samples/drug1ntypenode"
store project as "/CRISPD/DrugExample.cpj"
store output "Data Audit of [6 fields]" as "/my folder/My Audit"
```

È possibile aggiungere un'estensione al nome dell'oggetto, quale *.str* o *.gm*; questa operazione non è obbligatoria a condizione che il nome sia coerente. Per esempio, se un modello è stato archiviato senza estensione, deve essere recuperato con lo stesso nome.

```
store model "/myfolder/drugmodel"
retrieve model "/myfolder/drugmodel"
```

non:

```
store model "/myfolder/drugmodel.gm"
retrieve model "/myfolder/drugmodel.gm" version "0:2005-10-12 14:15:41.281"
```

Si noti che quando si recuperano oggetti, se non si specifica una versione o un'etichetta, viene sempre restituita la versione più recente dell'oggetto. Quando si recupera un oggetto nodo, il nodo viene automaticamente inserito nello stream corrente. Quando si recupera un oggetto stream, è necessario utilizzare uno script locale. Non è possibile recuperare un oggetto stream da uno script dello stream.

Blocco e sblocco degli oggetti

Da uno script, è possibile bloccare un oggetto per impedire qualsiasi aggiornamento delle sue versioni esistenti o la creazione di nuove versioni. È inoltre possibile sbloccare un oggetto che è stato bloccato.

La sintassi per bloccare e sbloccare un oggetto è:

```
lock REPOSITORY_PATH
lock URI
```

```
unlock REPOSITORY_PATH
unlock URI
```

Come nel caso dell'archiviazione e del recupero di oggetti, `REPOSITORY_PATH` fornisce la posizione dell'oggetto nel repository. Il percorso deve essere racchiuso tra virgolette ed è necessario utilizzare le barre (/) come delimitatori. Il percorso non fa distinzione tra maiuscole e minuscole.

```
lock "/myfolder/Stream1.str"
```

```
unlock "/myfolder/Stream1.str"
```

In alternativa, è possibile utilizzare un URI (Uniform Resource Identifier) anziché un percorso di repository per indicare la posizione dell'oggetto. L'URI deve includere il prefisso `spsscr:` e deve essere racchiuso tra virgolette. Come delimitatori di percorso sono consentite solo le barre (/) e gli spazi devono essere codificati, ovvero utilizzare `%20` anziché lo spazio nel percorso. L'URI non fa distinzione tra maiuscole e minuscole. Di seguito sono riportati alcuni esempi:

```
lock "spsscr:///myfolder/Stream1.str"
```

```
unlock "spsscr:///myfolder/Stream1.str"
```

Tenere presente che il blocco dell'oggetto si applica a tutte le versioni di un oggetto. Non è possibile, infatti, bloccare o sbloccare versioni singole.

Generazione di una password codificata

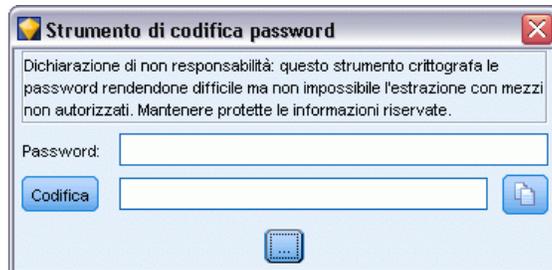
In alcuni casi, è possibile includere una password in uno script, per esempio se si desidera accedere a un'origine dati protetta da password. Le password codificate possono essere utilizzate in:

- Proprietà dei nodi per nodi di input Database e nodi output
- Argomenti della riga di comando per l'accesso al server
- Proprietà di connessione al database archiviate in un file *.par* (file del parametro generato dalla scheda Pubblica di un nodo di esportazione)

Tramite l'interfaccia utente, è disponibile uno strumento per generare password codificate in base all'algoritmo Blowfish (per ulteriori informazioni, vedere <http://www.schneier.com/blowfish.html>). Dopo la codifica, è possibile copiare e archiviare la password in file script e argomenti della riga di comando. La proprietà del nodo *epassword* utilizzata per *database* e *databaseexport* archivia la password codificata.

- ▶ Per generare una password codificata, dal menu Strumenti scegliere:
Codifica password...

Figura 5-1
Strumento di codifica password



- ▶ Specificare una password nella casella di testo Password.
- ▶ Fare clic su Codifica per generare una codifica casuale per la password.
- ▶ Fare clic sul pulsante Copia per copiare la password codificata negli Appunti.
- ▶ Incollare la password nello script o nel parametro desiderato.

Controllo degli script

Per controllare in modo rapido la sintassi di tutti i tipi di script, fare clic sul pulsante di verifica di colore rosso disponibile nella barra degli strumenti della finestra di dialogo Script locale.

Figura 5-2
Icone della barra degli strumenti Script dello stream



Durante il controllo degli script verranno segnalati gli eventuali errori del codice e forniti suggerimenti per la risoluzione. Per visualizzare la riga contenente gli errori, fare clic sul feedback visualizzato nella metà inferiore della finestra di dialogo. L'errore verrà evidenziato in rosso.

Script dalla riga di comando

Gli script consentono di eseguire operazioni che in genere vengono eseguite nell'interfaccia utente. È sufficiente specificare ed eseguire gli script locali dalla riga di comando quando si avvia IBM® SPSS® Modeler. Per esempio:

```
client -script scores.txt -execute
```

Il flag `-script` carica lo script specificato, mentre il flag `-execute` esegue tutti i comandi contenuti nel file di script.

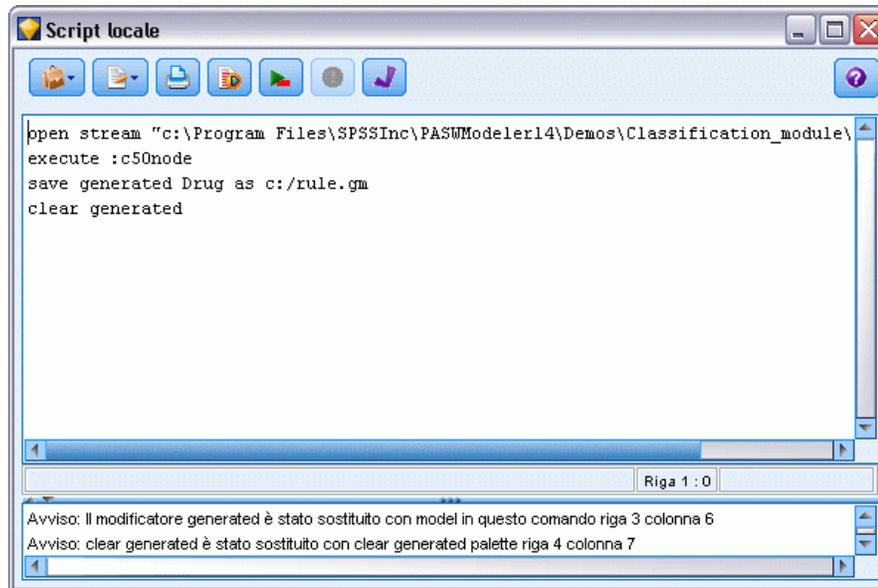
Compatibilità con le versioni precedenti

Questa versione di IBM® SPSS® Modeler supporta il funzionamento degli script creati nelle versioni precedenti. Tuttavia, ora è possibile inserire automaticamente gli insiemi di modelli nello stream (questa è l'impostazione di default) nonché sostituire o integrare un insieme di modelli esistente dello stesso tipo all'interno dello stream. Il fatto che questo accada o meno dipende dalle impostazioni delle opzioni *Aggiungi modello allo stream* e *Sostituisci modello precedente* (*Strumenti > Opzioni > Opzioni utente > Notifiche*). Per esempio, può essere necessario modificare uno script di una versione precedente in cui la sostituzione di un insieme di modelli avviene eliminando l'insieme di modelli esistenti e inserendone uno nuovo.

Gli script creati in questa versione potrebbero non funzionare in versioni precedenti.

Se uno script creato in una versione precedente utilizza un comando che è stato nel frattempo sostituito o che non è più supportato, il formato sarà supportato, ma verrà visualizzato un messaggio di avviso. Per esempio, la vecchia parola chiave `generated` è stata sostituita da `model`, e `clear generated` è stato sostituito da `clear generated palette`. Gli script che utilizzano i vecchi formati verranno eseguiti, tuttavia verrà visualizzato un messaggio di avviso.

Figura 5-3
Esecuzione di uno script che utilizza un comando obsoleto



Esempi di script

In questa sezione sono riportati numerosi esempi che illustrano l'utilizzo degli script in IBM® SPSS® Modeler.

Report nodo Tipo

Questo script crea un report HTML contenente informazioni sui campi dello stream corrente. Lo script si può utilizzare con tutti gli stream provvisti di un nodo Tipo istanziato e si può facilmente estendere per includere proprietà o nodi aggiuntivi.

- Per formattare i risultati per la visualizzazione in un browser standard vengono utilizzati tag HTML standard.
- Per accedere alle proprietà dei singoli campi viene utilizzato un nodo Tipo di IBM® SPSS® Modeler. Lo script si può facilmente estendere per elencare proprietà aggiuntive esposte mediante il nodo Tipo, quali valori mancanti o ruolo del campo. [Per ulteriori informazioni, vedere l'argomento Proprietà typenode in il capitolo 14 a pag. 170.](#)
- Per scrivere l'output in un file e per eseguire cicli sui campi al fine di accedere alle rispettive proprietà vengono utilizzati i comandi di script di SPSS Modeler. [Per ulteriori informazioni, vedere l'argomento Comandi di script in il capitolo 4 a pag. 30.](#)

Figura 6-1
Esempio di script per il report nodo Tipo

```
# This script creates an HTML file and adds data from the Type node.
var myreport
set myreport = open create "C:/typenodereport.html"

# set up the HTML page
writeln myreport "<html>"
writeln myreport "<header>Informazioni nodo Tipo da IBM SPSS Modeler</header>"
writeln myreport "<body><br><br>"

#create the table and write out the headers
writeln myreport "<table border='1'"
writeln myreport "<tr bgcolor='C0C0C0'"
writeln myreport "<td>Campo</td><td>Tipo</td><td>Valori</td>"
writeln myreport "</tr>"

# loop through fields and add a row for each
var current_field
for current_field in _fields_at Type
  writeln myreport "<tr>"
  write myreport "<td>" >> ^current_field >> "</td>"
  write myreport "<td>" >> Type:typenode.type.^current_field >> "</td>"

# add values for numeric fields
```

```

if Type:typenode.type.^current_field = Range then
  writeln myreport "<td>" << Type:typenode.values.^current_field >> "</td>"
endif

# add values for flag fields
if Type:typenode.type.^current_field = Flag then
  writeln myreport "<td>" << Type:typenode.values.^current_field >> "</td>"
endif

# add values for nominal fields
if Type:typenode.type.^current_field = Set then
  writeln myreport "<td>"
  var current_value
  for current_value in Type:typenode.values.^current_field
    writeln myreport ^current_value >> "<BR/>"
  endfor
  writeln myreport "</td>"
endif

  writeln myreport "</tr>"
endfor
writeln myreport "</table>"
writeln myreport "</body>"
writeln myreport "</html>"
close myreport

```

Creazione del file di output

Lo script inizia con la creazione di un nuovo file HTML e aggiunge i tag necessari per la creazione di una tabella con una riga di intestazione contenente i titoli di colonna *Field*, *Type* e *Values*. Ogni coppia di tag `<td></td>` crea una cella all'interno di una riga della tabella. Queste colonne saranno compilate per ogni campo in base alle proprietà del nodo Tipo.

```

# This script creates an HTML file and adds data from the Type node.
var myreport
set myreport = open create "C:/typenodereport.html"

# set up the HTML page
writeln myreport "<html>"
writeln myreport "<header>Informazioni nodo Tipo da IBM SPSS Modeler</header>"
writeln myreport "<body><br/><br/>"

#create the table and write out the headers
writeln myreport "<table border=\"1\">"
writeln myreport "<tr bgcolor=\"COCOC0\">"
writeln myreport "<td>Campo</td><td>Tipo</td><td>Valori</td>"
writeln myreport "</tr>"

```

Esecuzione di cicli sui campi

Successivamente, lo script esegue un ciclo su tutti i campi del nodo Tipo e aggiunge una riga per ogni campo, elencando il nome e il tipo di campo.

```
# loop through fields and add a row for each
var current_field
for current_field in _fields_at Type
  writeln myreport "<tr>"
  write myreport "<td>" >< ^current_field >< "</td>"
  write myreport "<td>" >< Type:typenode.type.^current_field >< "</td>"
```

Valori dei campi continui e flag

Per i campi continui (intervallo numerico), la proprietà `typenode.values` restituisce i valori minimi e massimi nel formato `[0.500517, 0.899774]`, visualizzato nella tabella. Per i campi flag, i valori vero/falso vengono visualizzati in un formato analogo.

```
# add values for numeric fields
if Type:typenode.type.^current_field = Range then
  writeln myreport "<td>" >< Type:typenode.values.^current_field >< "</td>"
endif

# add values for flag fields
if Type:typenode.type.^current_field = Flag then
  writeln myreport "<td>" >< Type:typenode.values.^current_field >< "</td>"
endif
```

Valori dei campi nominali

Per i campi nominali, la proprietà `typenode.values` restituisce l'elenco completo dei valori definiti. Lo script esegue un ciclo nell'elenco alla ricerca dei singoli campi in cui inserire di volta in volta ogni valore, con un'interruzione di riga (tag `
`) fra uno e l'altro.

```
# add values for nominal fields
if Type:typenode.type.^current_field = Set then
  writeln myreport "<td>"
  var current_value
  for current_value in Type:typenode.values.^current_field
    writeln myreport ^current_value >< "<BR/>"
  endfor
  writeln myreport "</td>"
endif
```

Chiusura del file

Infine, lo script chiude la riga, chiude i tag `<table>`, `<body>` e `<html>` e chiude il file di output.

```
writeln myreport "</tr>"
endfor
```

```
writeln myreport "</table>"
writeln myreport "</body>"
writeln myreport "</html>"
close myreport
```

Report di stream

Questo script crea un report HTML che elenca il nome, il tipo e l'annotazione per ogni nodo dello stream corrente. Oltre alle nozioni di base per la creazione di un file HTML e l'accesso alle proprietà dei nodi e degli stream, l'esempio illustra come creare un ciclo che esegue un determinato insieme di istruzioni per ogni nodo dello stream. Questo script può essere utilizzato con qualsiasi stream.

Figura 6-2
Esempio di script per il report di stream

```
# Create the HTML page with heading
var myfile
set myfile = open create "c:\stream_report.html"
writeln myfile "<HTML>"
writeln myfile "<BODY>"
writeln myfile "<HEAD>Report for stream " >> ^stream.name >> ".str</HEAD>"
writeln myfile "<p>" >> ^stream.annotation >> "</p>"

#Create the table with header row
writeln myfile "<TABLE border=\\"1\" width=\\"90%\">"
writeln myfile "<tr bgcolor=\\"lightgrey\" colspan=\\"3\">"
writeln myfile " <th>Node Name</th>"
writeln myfile " <th>Type</th>"
writeln myfile " <th>Annotation</th>"
writeln myfile "</tr>"

# Loop through nodes and add name, type, and annotation for each
# The ^stream.nodes property returns the list of nodes
var current_node
for current_node in ^stream.nodes
  writeln myfile "<tr>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.name
  writeln myfile " </td>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.node_type
  writeln myfile " </td>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.annotation >> "&nbsp;"
  writeln myfile " </td>"
  writeln myfile "</tr>"
endfor

writeln myfile "</TABLE>"
```

```
writeln myfile "</BODY>"
writeln myfile "</HTML>"
close myfile
```

Creazione del report

Lo script inizia con la creazione di un nuovo file HTML con gli elementi <BODY> e <HEAD>. La proprietà `^stream.name` restituisce il nome dello stream corrente, che viene inserito nell'intestazione. Per concatenare le stringhe viene utilizzato l'operatore `>>`.

```
# Create the HTML page with heading
var myfile
set myfile = open create "c:\stream_report.html"
writeln myfile "<HTML>"
writeln myfile "<BODY>"
writeln myfile "<HEAD>Report for stream " >> ^stream.name >> ".str</HEAD>"
writeln myfile "<p>" >> ^stream.annotation >> "</p>"
```

Successivamente, lo script crea una tabella HTML con una riga di intestazione contenente i titoli di colonna *Node Name*, *Type* e *Annotation*. Ogni coppia di tag <td></td> crea una cella all'interno di una riga della tabella.

```
#Create the table with header row
writeln myfile "<TABLE border=\"1\" width=\"90%\">"
writeln myfile "<tr bgcolor=\"lightgrey\" colspan=\"3\">"
writeln myfile " <th>Node Name</th>"
writeln myfile " <th>Type</th>"
writeln myfile " <th>Annotation</th>"
writeln myfile "</tr>"
```

Quindi, lo script esegue un ciclo su tutti i nodi dello stream corrente. Per ogni nodo viene aggiunta alla tabella una riga contenente il nome, il tipo e l'annotazione. Dopo l'annotazione viene inserito uno spazio unificatore invisibile (` `) per evitare di creare una cella vuota nei casi in cui per un determinato nodo non sia specificata alcuna annotazione (le celle vuote possono determinare una formattazione inattesa quando viene visualizzata la tabella).

```
# Loop through nodes and add name, type, and annotation for each
# The ^stream.nodes property returns the list of nodes
var current_node
for current_node in ^stream.nodes
  writeln myfile "<tr>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.name"
  writeln myfile " </td>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.node_type"
  writeln myfile " </td>"
  writeln myfile " <td>"
  writeln myfile " ^current_node.annotation >> "&nbsp;"
  writeln myfile " </td>"
  writeln myfile "</tr>"
endfor
```

Infine, lo script aggiunge i tag HTML necessari per chiudere il documento e chiude il file.

```
writeln myfile "</TABLE>"  
writeln myfile "</BODY>"  
writeln myfile "</HTML>"  
close myfile
```

Argomenti della riga di comando

Modalità di richiamo del software

È possibile utilizzare la riga di comando del sistema operativo per avviare IBM® SPSS® Modeler:

- ▶ Sul computer in cui è installato IBM® SPSS® Modeler, aprire una finestra DOS (prompt dei comandi).
- ▶ Per avviare l'interfaccia di SPSS Modeler in modalità interattiva, digitare il comando `modelerclient` seguito dagli argomenti desiderati; per esempio:

```
modelerclient -stream report.str -execute
```

Gli argomenti disponibili (flag) consentono di connettersi a un server, caricare stream, eseguire script o specificare altri parametri.

Utilizzo degli argomenti della riga di comando

È possibile aggiungere alcuni argomenti della riga di comando (denominati anche **flag**) al comando `modelerclient` iniziale per modificare il modo in cui IBM® SPSS® Modeler viene richiamato.

Per esempio, è possibile utilizzare gli argomenti della riga di comando `-server`, `-stream` e `-execute` per connettersi a un server e caricare ed eseguire uno stream, come indicato di seguito:

```
modelerclient -server -hostname myserver -port 80 -username dminer  
-password 1234 -stream mystream.str -execute
```

Si noti che in caso di esecuzione dalla riga di comando con Clementine Client installato localmente, gli argomenti di connessione al server non sono necessari.

È possibile racchiudere tra virgolette doppie i valori di parametri che contengono spazi, per esempio:

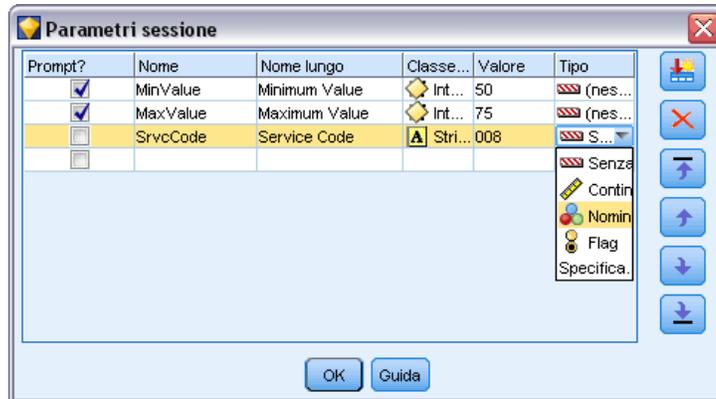
```
modelerclient -stream mystream.str -Pusername="Joe User" -execute
```

Questa soluzione consente anche di eseguire stati e script di SPSS Modeler, utilizzando rispettivamente i flag `-state` e `-script`.

Debug degli argomenti della riga di comando

Per eseguire il debug di una riga di comando, utilizzare il comando `modelerclient` per avviare SPSS Modeler con gli argomenti desiderati. Ciò consente di verificare che i comandi vengano eseguiti come previsto. È possibile confermare i valori di qualsiasi parametro passato dalla riga di comando nella finestra di dialogo Parametri sessione (menu Strumenti, Imposta parametri sessione).

Figura 7-1
Impostazione dei parametri per la sessione



Combinazione di più argomenti

È possibile combinare più argomenti in un unico file di comandi, che potrà essere specificato all'avvio utilizzando il simbolo @ seguito dal nome del file. In questo modo è possibile abbreviare il richiamo dalla riga di comando e superare eventuali limitazioni di lunghezza dei comandi previste dal sistema operativo. Per esempio, il seguente comando di avvio utilizza gli argomenti specificati nel file indicato da <commandFileName>.

```
modelerclient @<NomeFilecomando>
```

Se è necessario specificare degli spazi, racchiudere il nome del file e il percorso tra virgolette, per esempio:

```
modelerclient @"C:\Program Files\IBM\SPSS\Modeler\...\scripts\mio_file_comando.txt"
```

Il file di comando può contenere tutti gli argomenti che in precedenza venivano specificati singolarmente all'avvio, con un argomento per riga. Per esempio:

```
-stream report.str
-Porder.full_filename=APR_orders.dat
-Preport.filename=APR_report.txt
-execute
```

Quando si scrivono o si richiamano file di comando è importante attenersi alle seguenti indicazioni:

- Specificare un solo comando per riga.
- Non incorporare un argomento @CommandFile in un file di comando.

Argomenti per la connessione al server

Il flag -server indica a IBM® SPSS® Modeler che deve connettersi a un server pubblico, mentre i flag -hostname, -use_ssl, -port, -username, -password e -domain si utilizzano per indicare a SPSS Modeler come effettuare la connessione al server pubblico. Se non viene specificato un argomento -server, viene utilizzato il server di default o locale.

Esempi

Per connettersi a un server pubblico:

```
modelerclient -server -hostname myserver -port 80 -username dminer
-password 1234 -stream mystream.str -execute
```

Per connettersi a un cluster di server:

```
modelerclient -server -cluster "QA Machines" \
-spsscr_hostname pes_host -spsscr_port 8080 \
-spsscr_username asmith -spsscr_epassword xyz
```

Si noti che la connessione a un cluster di server richiede il plug-in Coordinator of Processes attraverso IBM® SPSS® Collaboration and Deployment Services, quindi l'argomento `-cluster` deve essere utilizzato insieme alle opzioni di connessione al repository (`spsscr_*`). [Per ulteriori informazioni, vedere l'argomento Argomenti per la connessione a IBM SPSS Collaboration and Deployment Services Repository a pag. 72.](#)

| Argomento | Comportamento/descrizione |
|---|---|
| <code>-server</code> | Esegue SPSS Modeler in modalità server, connettendosi a un server pubblico mediante i flag <code>-hostname</code> , <code>-port</code> , <code>-username</code> , <code>-password</code> e <code>-domain</code> . |
| <code>-hostname <name></code> | Nome host del server. Disponibile solo nella modalità server. |
| <code>-use_ssl</code> | Specifica che la connessione deve utilizzare il protocollo SSL (Secure Socket Layer). Flag facoltativo; l'impostazione predefinita <i>non</i> prevede l'uso di SSL. |
| <code>-port <numero></code> | Numero di porta del server specificato. Disponibile solo nella modalità server. |
| <code>-cluster <name></code> | Specifica una connessione a un cluster di server anziché a un server denominato. Argomento alternativo a <code>hostname</code> , <code>port</code> e <code>use_ssl</code> . Il nome è il nome del cluster o un URI univoco che identifica il cluster in IBM® SPSS® Collaboration and Deployment Services Repository. Il cluster di server viene gestito da Coordinator of Processes attraverso IBM SPSS Collaboration and Deployment Services. Per ulteriori informazioni, vedere l'argomento Argomenti per la connessione a IBM SPSS Collaboration and Deployment Services Repository a pag. 72. |
| <code>-username <name></code> | Nome utente utilizzato per l'accesso al server. Disponibile solo nella modalità server. |
| <code>-password <password></code> | Password utilizzata per l'accesso al server. Disponibile solo nella modalità server. <i>Nota:</i> Se l'argomento <code>-password</code> non viene utilizzato, verrà richiesta l'immissione di una password. |
| <code>-epassword <encodedpasswordstring></code> | Password codificata utilizzata per l'accesso al server. Disponibile solo nella modalità server. <i>Nota:</i> per creare una password codificata, utilizzare il menu Strumenti dell'applicazione SPSS Modeler. |
| <code>-domain <name></code> | Dominio utilizzato per l'accesso al server. Disponibile solo nella modalità server. |
| <code>-P <name>=<value></code> | Utilizzato per impostare un parametro di avvio. Può essere utilizzato anche per impostare le proprietà dei nodi (parametri di configurazione). |

Argomenti per la connessione a IBM SPSS Collaboration and Deployment Services Repository

Nota: per l'accesso a un repository IBM® SPSS® Collaboration and Deployment Services è necessaria una licenza distinta. Per ulteriori informazioni, vedere <http://www.ibm.com/software/analytics/spss/products/deployment/cds/>

Se si desidera archiviare o recuperare oggetti da IBM SPSS Collaboration and Deployment Services tramite la riga di comando, è necessario specificare una connessione valida a IBM® SPSS® Collaboration and Deployment Services Repository. Per esempio:

```
modelerclient -spsscr_hostname myhost -spsscr_port 8080
-spsscr_username nomeutente -spsscr_password mypassword
-stream "spsscr:///cartella_1/punteggio_stream.str" -execute
```

Nella tabella riportata di seguito sono elencati gli argomenti utilizzabili per impostare la connessione:

| Argomento | Comportamento/descrizione |
|---|--|
| -spsscr_hostname <nome host o indirizzo IP> | Nome host o indirizzo IP del server su cui è installato IBM SPSS Collaboration and Deployment Services Repository. |
| -spsscr_port <numero> | Numero di porta su cui IBM SPSS Collaboration and Deployment Services Repository accetta connessioni, generalmente è la porta 8080 di default. |
| -spsscr_use_ssl | Specifica che la connessione deve utilizzare il protocollo SSL (Secure Socket Layer). Flag facoltativo; l'impostazione predefinita <i>non</i> prevede l'uso di SSL. |
| -spsscr_username <name> | Nome utente utilizzato per l'accesso a IBM SPSS Collaboration and Deployment Services Repository. |
| -spsscr_password <password> | Password utilizzata per l'accesso a IBM SPSS Collaboration and Deployment Services Repository. |
| -spsscr_epassword <password codificata> | Password codificata utilizzata per l'accesso a IBM SPSS Collaboration and Deployment Services Repository. |
| -spsscr_domain <name> | Dominio utilizzato per l'accesso a IBM SPSS Collaboration and Deployment Services Repository. Flag facoltativo; utilizzarlo solo se si effettua l'accesso con LDAP o Active Directory. |

Argomenti di sistema

Nella tabella seguente sono illustrati gli argomenti di sistema disponibili per il richiamo dell'interfaccia utente dalla riga di comando:

| Argomento | Comportamento/descrizione |
|------------------|--|
| @ <Filecomando> | Il carattere '@' seguito da un nome di file specifica un elenco di comandi. Quando modelerclient incontra un argomento che inizia con @, utilizza i comandi del file esattamente come se fossero stati specificati nella riga di comando. Per ulteriori informazioni, vedere l'argomento Combinazione di più argomenti a pag. 70. |
| -directory <dir> | Consente di impostare la directory di default. Nella modalità locale tale directory viene utilizzata sia per i dati che per l'output. |

| Argomento | Comportamento/descrizione |
|-------------------------|---|
| -server_directory <dir> | Consente di impostare la directory del server di default per i dati. Tale directory, specificata mediante il flag -directory, viene utilizzata per l'output. |
| -execute | Dopo l'avvio, consente di eseguire eventuali stream, stati o script caricati all'avvio. Se oltre a uno stream o stato viene caricato anche uno script, verrà eseguito solamente lo script. |
| -stream <stream> | Consente di caricare all'avvio lo stream specificato. È possibile specificare più stream, ma l'ultimo stream specificato verrà impostato come stream corrente. |
| -script <script> | Consente di caricare all'avvio lo script autonomo specificato. Come illustrato di seguito, tale script può essere specificato insieme a uno stream o a uno stato, ma è possibile caricare un solo script all'avvio. |
| -model <modello> | Consente di caricare all'avvio il modello generato specificato (formato di file .gm). |
| -state <stato> | Consente di caricare all'avvio lo stato salvato specificato. |
| -project <progetto> | Consente di caricare il progetto specificato. È possibile caricare un solo progetto all'avvio. |
| -output <output> | Consente di caricare l'oggetto di output salvato (file di formato COU) all'avvio. |
| -help | Consente di visualizzare un elenco di argomenti della riga di comando. Quando si specifica questa opzione, tutti gli altri argomenti vengono ignorati e viene visualizzata la finestra della Guida in linea. |
| -P <name>=<value> | Utilizzato per impostare un parametro di avvio. Può essere utilizzato anche per impostare le proprietà dei nodi (parametri di configurazione). |

Nota: È possibile impostare le directory di default anche nell'interfaccia utente. Per accedere alle opzioni, scegliere Imposta directory o Imposta directory server dal menu File.

Caricamento di più file

Dalla riga di comando è possibile caricare più stream, stati e output all'avvio ripetendo l'argomento rilevante per ogni oggetto caricato. Per esempio, per caricare ed eseguire due stream denominati *report.str* e *train.str*, è necessario utilizzare il seguente comando:

```
modelerclient -stream report.str -stream train.str -execute
```

Caricamento di oggetti da IBM SPSS Collaboration and Deployment Services Repository

Poiché è possibile caricare determinati oggetti da un file o da IBM® SPSS® Collaboration and Deployment Services Repository (se concesso in licenza), il prefisso *spsscr:* che precede il nome file e, facoltativamente, *file:* (per oggetti su disco) indica a IBM® SPSS® Modeler dove cercare l'oggetto. Il prefisso viene utilizzato con i seguenti flag:

- -stream
- -script
- -output
- -model
- -project

Il prefisso viene utilizzato per creare un URI che specifica la posizione dell'oggetto, per esempio

-stream "spsscr:///folder_1/scoring_stream.str". La presenza del prefisso spsscr: richiede che nello stesso comando sia stata specificata una connessione valida a IBM SPSS Collaboration and Deployment Services Repository. Pertanto il comando completo si presenterà come segue:

```
modelerclient -spsscr_hostname myhost -spsscr_port 8080  
-spsscr_username nomeutente -spsscr_password mypassword  
-stream "spsscr:///cartella_1/punteggio_stream.str" -execute
```

Per ulteriori informazioni sugli URI per gli oggetti in IBM SPSS Collaboration and Deployment Services Repository, vedere [Accesso a oggetti in IBM SPSS Collaboration and Deployment Services Repository in il capitolo 5 a pag. 58](#). Si noti che dalla riga di comando è *necessario* utilizzare un URI. Non è infatti supportato il semplice REPOSITORY_PATH che viene utilizzato solo all'interno di script.

Argomenti dei parametri

I parametri possono essere utilizzati come flag durante l'esecuzione della riga di comando di IBM® SPSS® Modeler. Negli argomenti della riga di comando il flag -P consente di specificare un parametro, nel formato -P <nome>=<valore>.

I parametri possono essere dei seguenti tipi:

- **Parametri semplici** o parametri utilizzati direttamente nelle espressioni CLEM.
- **Parametri di configurazione**, detti anche **proprietà dei nodi**. Questi parametri vengono utilizzati per modificare le impostazioni dei nodi nello stream. [Per ulteriori informazioni, vedere l'argomento Panoramica sulle proprietà dei nodi in il capitolo 9 a pag. 116](#).
- **Parametri della riga di comando** che consentono di modificare il richiamo di SPSS Modeler.

Per esempio, è possibile specificare nomi utente e password per le sorgenti dei dati sotto forma di flag della riga di comando, come nel seguente esempio:

```
modelerclient -stream response.str -P:databasenode.username=george  
-P:databasenode.password=jetson
```

Riferimenti al linguaggio CLEM

Panoramica di riferimento su CLEM

Questa sezione illustra il linguaggio CLEM (Control Language for Expression Manipulation), un potente strumento per l'analisi e la manipolazione dei dati utilizzati negli stream di IBM® SPSS® Modeler. È possibile utilizzare CLEM all'interno di nodi per eseguire svariate attività, dalla valutazione di condizioni o derivazione di valori fino all'inserimento di dati in report. [Per ulteriori informazioni, vedere l'argomento Informazioni su CLEM in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

È inoltre possibile utilizzare un sottoinsieme del linguaggio CLEM quando gli script sono eseguiti nell'interfaccia utente, in modo da automatizzare numerose operazioni di manipolazione di dati dello stesso tipo. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

Le espressioni CLEM sono costituite da valori, nomi di campi, operatori e funzioni. Tramite la sintassi corretta è possibile creare una vasta gamma di potenti operazioni sui dati. [Per ulteriori informazioni, vedere l'argomento Esempi CLEM in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Tipi di dati CLEM

Nel linguaggio CLEM è possibile utilizzare i seguenti tipi di dati:

- Interi
- Numeri reali
- Caratteri
- Stringhe
- Elenchi
- Campi
- Data/ora

Regole per l'inserimento tra virgolette

Sebbene IBM® SPSS® Modeler sia flessibile quando l'utente determina campi, valori, parametri e stringhe utilizzati in un'espressione CLEM, di seguito sono suggerite alcune "procedure consigliate" da seguire durante la creazione di espressioni.

- Stringhe: utilizzare sempre virgolette doppie per la scrittura di stringhe, per esempio "Type 2". È inoltre possibile utilizzare le virgolette singole, ma il rischio è di creare confusione con i campi racchiusi tra virgolette.

- Campi: utilizzare virgolette singole solo se necessarie per racchiudere spazi o altri caratteri speciali, per esempio 'Order Number'. I campi racchiusi tra virgolette ma non definiti nell'insieme di dati verranno interpretati come stringhe.
- Parametri: utilizzare sempre virgolette singole quando si specificano parametri, per esempio '\$P-threshold'.
- Caratteri: utilizzare sempre virgolette singole a barra rovesciata (^), per esempio stripchar(`d`, "drugA").

Per ulteriori informazioni, vedere l'argomento Valori e tipi di dati in il capitolo 7 in *Manuale dell'utente di IBM SPSS Modeler 14.2*. Queste regole verranno illustrate in modo più dettagliato negli argomenti riportati di seguito.

Interi

I numeri interi vengono rappresentati come sequenze di cifre decimali. Se lo si desidera, è possibile inserire un segno meno (-) prima del numero intero per indicare un numero negativo, per esempio 1234, 999, -77.

Il linguaggio CLEM è in grado di gestire numeri interi a precisione arbitraria. La dimensione massima dei numeri interi dipende dalla piattaforma in uso. Se i valori hanno dimensioni superiori a quelle visualizzabili in un campo di numeri interi, l'impostazione del tipo di campo su Real consente in genere di ripristinare il valore.

Numeri reali

Con *numero reale* si intende un numero a virgola mobile. I numeri reali sono rappresentati da una o più cifre seguite da un punto decimale, seguito a sua volta da una o più cifre. Nel linguaggio CLEM, vengono utilizzati numeri reali con precisione doppia.

Se lo si desidera, è possibile inserire un segno meno (-) prima del numero reale per indicare un numero negativo, per esempio 1.234, 0.999, -77.001. Per esprimere un numero reale con la notazione esponenziale, utilizzare il formato <numero> e <esponente>, per esempio 1234.0e5, 1.7e- e 2. Quando le stringhe di numeri vengono lette e convertite automaticamente in numeri dall'applicazione IBM® SPSS® Modeler, vengono comunque accettati i numeri senza cifre iniziali prima del punto decimale e senza cifre dopo il punto, per esempio 999. o .11. Questi formati non sono tuttavia validi nelle espressioni CLEM.

Nota: quando si fa riferimento a numeri reali in espressioni CLEM, è necessario utilizzare il punto come separatore decimale indipendentemente da eventuali impostazioni dello stream corrente o dalle impostazioni internazionali. Per esempio, specificare

Na > 0.6

anziché

Na > 0.6

Questa regola è valida anche se nella finestra di dialogo delle proprietà dello stream è stata selezionata la virgola come simbolo decimale ed è conforme alla linea guida generale secondo cui la sintassi del codice deve essere indipendente da specifiche impostazioni internazionali o convenzioni.

Caratteri

I caratteri sono in genere visualizzati come CHAR e nelle espressioni CLEM consentono di eseguire test sulle stringhe. Per esempio, la funzione `isuppercode` consente di stabilire se il primo carattere di una stringa è maiuscolo. L'espressione CLEM seguente utilizza un carattere per indicare che il test deve essere eseguito sul primo carattere della stringa:

```
isuppercode(subscrs(1,"MiaStringa"))
```

Per esprimere il codice (anziché la posizione) di un carattere specifico in un'espressione CLEM, utilizzare virgolette singole a barra rovesciata nel formato `<carattere>`, per esempio, ``A``, ``Z``.

Nota: Non esiste un tipo di archiviazione CHAR per un campo, pertanto se un campo deriva o viene riempito con un'espressione che produce un tipo CHAR, il risultato verrà convertito in una stringa.

Stringhe

È in genere opportuno racchiudere le stringhe tra virgolette doppie. Per esempio le stringhe "c35product2" e "referrerID". Per indicare caratteri speciali in una stringa, utilizzare una barra rovesciata, per esempio "\$65443". (Per indicare una barra rovesciata, utilizzare una doppia barra rovesciata, `\\`). È possibile racchiudere una stringa tra virgolette singole, ma in questo caso il risultato (per esempio 'referrerID') non è distinguibile da un campo tra virgolette. [Per ulteriori informazioni, vedere l'argomento Funzioni stringa in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Elenchi

Un elenco rappresenta una sequenza ordinata di elementi, che possono essere di tipo eterogeneo. Gli elenchi sono racchiusi tra parentesi quadre (`[]`). Per esempio gli elenchi `[1 2 4 16]` e `["abc" "def"]`. Gli elenchi non vengono utilizzati come valori dei campi di IBM® SPSS® Modeler, bensì per fornire argomenti alle funzioni, quali `member` e `oneof`.

Campi

I nomi inclusi nelle espressioni CLEM non corrispondenti a nomi di funzioni vengono interpretati come nomi di campi. È possibile scriverli come `Power`, `val27`, `state_flag` e così via, ma se il nome inizia con una cifra o include caratteri non alfabetici, quali gli spazi (ad eccezione del carattere di sottolineatura), racchiudere il nome tra virgolette singole, per esempio, `'Power Increase'`, `'2nd answer'`, `'#101'`, `'$P-NextField'`.

Nota: I campi racchiusi tra virgolette ma non definiti nell'insieme di dati verranno interpretati come stringhe.

Date

I calcoli di data sono basati su una data di riferimento, specificata nella finestra di dialogo delle proprietà dello stream. La data di riferimento di default è 1 gennaio 1900. [Per ulteriori informazioni, vedere l'argomento Impostazione delle opzioni per gli stream in il capitolo 5 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Il linguaggio CLEM supporta i formati di data seguenti.

| Formato | Esempi |
|-------------|---|
| DDMMYY | 150163 |
| MMDDYY | 011563 |
| YYMMDD | 630115 |
| YYYYMMDD | 19630115 |
| YYYYDDD | Anno a quattro cifre seguito da numero a tre cifre rappresentante il giorno dell'anno, per esempio 2000032 rappresenta il 32esimo giorno del 2000 oppure l'1 febbraio 2000. |
| DAY | Giorno della settimana nelle opzioni internazionali correnti, per esempio Monday, Tuesday, ..., ecc., in inglese. |
| MONTH | Mese nelle opzioni internazionali correnti, per esempio January, February, |
| DD/MM/YY | 15/01/63 |
| DD/MM/YYYY | 15/01/1963 |
| MM/DD/YY | 01/15/63 |
| MM/DD/YYYY | 01/15/1963 |
| DD-MM-YY | 15-01-63 |
| DD-MM-YYYY | 15-01-1963 |
| MM-DD-YY | 01-15-63 |
| MM-DD-YYYY | 01-15-1963 |
| DD.MM.YY | 15.01.63 |
| DD.MM.YYYY | 15.01.1963 |
| MM.DD.YY | 01.15.63 |
| MM.DD.YYYY | 01.15.1963 |
| DD-MON-YY | 15-JAN-63, 15-jan-63, 15-Jan-63 |
| DD/MON/YY | 15/JAN/63, 15/jan/63, 15/Jan/63 |
| DD.MON.YY | 15.JAN.63, 15.jan.63, 15.Jan.63 |
| DD-MON-YYYY | 15-JAN-1963, 15-jan-1963, 15-Jan-1963 |
| DD/MON/YYYY | 15/JAN/1963, 15/jan/1963, 15/Jan/1963 |
| DD.MON.YYYY | 15.JAN.1963, 15.jan.1963, 15.Jan.1963 |
| MON YYYY | Jan 2004 |

| Formato | Esempi |
|------------|---|
| q Q YYYY | Data rappresentata come una cifra (1-4) rappresentante il trimestre seguito dalla lettera <i>T</i> e un anno a quattro cifre, per esempio 25 dicembre 2004 viene rappresentato nel seguente modo 4 Q 2004. |
| ww WK YYYY | Numero a due cifre rappresentante la settimana dell'anno, seguito dalle lettere <i>ST</i> e un anno a quattro cifre. La settimana dell'anno viene calcolata presupponendo che il primo giorno della settimana è Lunedì e che esiste almeno un giorno nella prima settimana. |

Time

Il linguaggio CLEM supporta i formati di ora seguenti.

| Formato | Esempi |
|----------------|------------------------------|
| HHMMSS | 120112, 010101, 221212 |
| HHMM | 1223, 0745, 2207 |
| MMSS | 5558, 0100 |
| HH:MM:SS | 12:01:12, 01:01:01, 22:12:12 |
| HH:MM | 12:23, 07:45, 22:07 |
| MM:SS | 55:58, 01:00 |
| (H)H:(M)M:(S)S | 12:1:12, 1:1:1, 22:12:12 |
| (H)H:(M)M | 12:23, 7:45, 22:7 |
| (M)M:(S)S | 55:58, 1:0 |
| HH.MM.SS | 12.01.12, 01.01.01, 22.12.12 |
| HH.MM | 12.23, 07.45, 22.07 |
| MM.SS | 55.58, 01.00 |
| (H)H.(M)M.(S)S | 12.1.12, 1.1.1, 22.12.12 |
| (H)H.(M)M | 12.23, 7.45, 22.7 |
| (M)M.(S)S | 55.58, 1.0 |

Operatori CLEM

Sono disponibili i seguenti operatori:

| Operazione | Commenti | Precedenza (vedere la sezione seguente) |
|------------|---|---|
| or | Utilizzata tra due espressioni CLEM. Restituisce il valore vero se una delle due espressioni è vera oppure se entrambe sono vere. | 10 |
| and | Utilizzata tra due espressioni CLEM. Restituisce il valore vero se entrambe le espressioni sono vere. | 9 |

| Operazione | Commenti | Precedenza (vedere la sezione seguente) |
|------------|---|---|
| = | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 è uguale a ELEMENTO2. | 7 |
| == | Identico a =. | 7 |
| /= | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 <i>non</i> è uguale a ELEMENTO2. | 7 |
| /== | Identico a /=. | 7 |
| > | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 è maggiore di ELEMENTO2. | 6 |
| >= | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 è maggiore o uguale a ELEMENTO2. | 6 |
| < | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 è minore di ELEMENTO2. | 6 |
| <= | Utilizzata tra qualsiasi coppia di elementi confrontabili. Restituisce vero se ELEMENTO1 è minore o uguale a ELEMENTO2. | 6 |
| &&=_0 | Utilizzata tra due numeri interi. Equivalente dell'espressione Booleana $INT1 \ \&\& \ INT2 = 0$. | 6 |
| &&/=_0 | Utilizzata tra due numeri interi. Equivalente dell'espressione Booleana $INT1 \ \&\& \ INT2 \neq 0$. | 6 |
| + | Somma due numeri: $NUM1 + NUM2$. | 5 |
| >< | Concatena due stringhe, per esempio $STRING1 \ >< \ STRING2$. | 5 |
| - | Sottrae un numero dall'altro: $NUM1 - NUM2$. Può inoltre precedere un numero: $- NUM$. | 5 |
| * | Utilizzata per moltiplicare due numeri: $NUM1 * NUM2$. | 4 |
| && | Utilizzata tra due numeri interi. Il risultato è l'AND bitwise di INT1 e INT2. | 4 |
| &&~~ | Utilizzata tra due numeri interi. Il risultato è l'AND bitwise di INT1 e il complemento bitwise di INT2. | 4 |
| | Utilizzata tra due numeri interi. Il risultato è l'OR inclusivo bitwise di INT1 e INT2. | 4 |
| ~~ | Utilizzata prima di un numero intero. Genera il complemento bitwise di INT. | 4 |

| Operazione | Commenti | Precedenza (vedere la sezione seguente) |
|------------|---|---|
| & | Utilizzata tra due numeri interi. Il risultato è l'OR esclusivo bitwise di INT1 e INT2. | 4 |
| INT1 << N | Utilizzata tra due numeri interi. Genera lo schema di bit di INT con spostamento a sinistra di N posizioni. | 4 |
| INT1 >> N | Utilizzata tra due numeri interi. Genera lo schema di bit di INT con spostamento a destra di N posizioni. | 4 |
| / | Utilizzata per dividere un numero per un altro: NUM1 / NUM2. | 4 |
| ** | Utilizzata tra due numeri: BASE ** POTENZA. Restituisce BASE elevata alla potenza POTENZA. | 3 |
| rem | Utilizzata tra due numeri interi: INT1 rem INT2. Restituisce il resto, INT1 - (INT1 div INT2) * INT2. | 2 |
| div | Utilizzata tra due numeri interi: INT1 div INT2. Esegue una divisione di interi. | 2 |

Precedenza degli operatori

Le precedenze influiscono sull'analisi delle espressioni complesse, in particolare delle espressioni non racchiuse tra parentesi e contenenti più operatori infissi. Ad esempio,

$3 + 4 * 5$

viene analizzato come $3 + (4 * 5)$ anziché $(3 + 4) * 5$ poiché le precedenze relative determinano che * deve essere analizzato prima di +. Nel linguaggio CLEM, a ogni operatore è associato un valore di precedenza. Minore è il valore, maggiore è l'importanza che gli viene attribuita nell'elenco di analisi. Ciò significa che verrà elaborato prima degli altri operatori con valori di precedenza maggiori.

Riferimenti alle funzioni

Per le operazioni con i dati in IBM® SPSS® Modeler sono disponibili le seguenti funzioni CLEM. È possibile inserire queste funzioni sotto forma di codice in numerose finestre di dialogo, tra cui i nodi Nuovo campo e Crea flag, oppure utilizzare il Generatore espressioni per creare espressioni CLEM valide senza che sia necessario memorizzare elenchi di funzioni o nomi di campi.

| Tipo di funzione | Descrizione |
|------------------|--|
| Informazioni | Utilizzate per ottenere informazioni sui valori dei campi. Per esempio, la funzione is_string restituisce "vero" per tutti i record il cui tipo è costituito da una stringa. |
| Conversione | Utilizzate per creare nuovi campi o convertire tipi di archiviazione. Per esempio, la funzione to_timestamp converte il campo selezionato in un timestamp. |

| Tipo di funzione | Descrizione |
|------------------|---|
| Confronto | Utilizzate per confrontare i valori dei campi tra di loro oppure con una stringa determinata. Per esempio, <= viene utilizzata per stabilire se i valori di due campi siano uno uguale o inferiore all'altro. |
| Logiche | Utilizzate per eseguire operazioni logiche, quali le operazioni if, then, else. |
| Numeriche | Utilizzate per eseguire calcoli numerici, quale il logaritmo naturale dei valori dei campi. |
| Trigonometriche | Utilizzate per eseguire calcoli trigonometrici, quale l'arcocoseno di un angolo specificato. |
| Probabilità | Restituiscono probabilità in base a varie distribuzioni, quali la probabilità che un valore dalla distribuzione <i>t</i> per studenti sia inferiore a un determinato valore. |
| Bitwise | Utilizzate per manipolare i numeri interi come schemi di bit. |
| Casuali | Utilizzate per la selezione casuale di elementi o la creazione casuale di numeri. |
| Stringa | Utilizzate per eseguire numerose operazioni sulle stringhe, per esempio stripchar, che consente di rimuovere un carattere specificato. |
| SoundEx | Utilizzate per trovare stringhe quando non se ne conosce l'esatta ortografia, in base a ipotesi fonetiche sulla pronuncia di determinate lettere. |
| Data e orario | Utilizzate per eseguire svariate operazioni sui campi di data, ora e timestamp. |
| Sequenza | Utilizzate per avere informazioni sulla sequenza di record di un insieme di dati o per eseguire operazioni in base a tale sequenza. |
| Globali | Utilizzate per accedere a valori globali creati da un nodo Globali. Per esempio, @MEAN viene utilizzata per fare riferimento alla media di tutti i valori di un campo nell'intero insieme di dati. |
| Vuoti e nulli | Utilizzate per accedere, contrassegnare e spesso riempire i vuoti specificati dall'utente o i valori mancanti del sistema. Per esempio, la funzione @BLANK(FIELD) viene utilizzata per assegnare un flag true a record contenenti valori vuoti. |
| Campi speciali | Utilizzate per denotare i campi specifici da esaminare. Per esempio, la funzione @FIELD viene utilizzata quando si derivano più campi. |

Convenzioni utilizzate nelle descrizioni delle funzioni

Le convenzioni seguenti vengono utilizzate in questa guida quando fanno riferimento agli elementi in una funzione.

| Convenzione | Descrizione |
|------------------------------|---|
| <i>BOOL</i> | Valore booleano o flag, per esempio vero o falso. |
| <i>NUM, NUM1, NUM2</i> | Qualsiasi numero. |
| <i>REALE, REALE1, REALE2</i> | Qualsiasi numero come 1.234 o -77.01. |
| <i>INT, INT1, INT2</i> | Qualsiasi numero come 1 o -77. |
| <i>CAR</i> | Un codice di carattere come 'A'. |
| <i>STRING</i> | Una stringa come "referrerID". |
| <i>LIST</i> | Un elenco di elementi come ["abc" "def"]. |
| <i>ELEMENTO</i> | Un campo come Customer o extract_concept. |

| Convenzione | Descrizione |
|-------------|---|
| <i>DATE</i> | Un campo data come <i>start_date</i> , i cui valori sono in un formato quale DD-MON-YYYY. |
| <i>ORA</i> | Un campo ora come <i>power_flux</i> , i cui valori sono in un formato quale HHMMSS. |

Le funzioni in questa guida vengono elencate con la funzione in una colonna, il tipo di risultato (intero, stringa, ecc.) in un'altra e una descrizione (se disponibile) in una terza colonna. Per esempio, di seguito è riportata la descrizione della funzione *rem*.

| Funzione | Risultato | Descrizione |
|---------------|---------------|--|
| INT1 rem INT2 | <i>Number</i> | Restituisce il resto della divisione di <i>INT1</i> diviso per <i>INT2</i> . Per esempio, $INT1 - (INT1 \text{ div } INT2) * INT2$. |

I dettagli sulle convenzioni di utilizzo, quali la modalità di elencare gli elementi oppure di specificare i caratteri in una funzione, vengono descritti altrove. [Per ulteriori informazioni, vedere l'argomento Tipi di dati CLEM in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Funzioni di informazione

Le funzioni di informazione vengono utilizzate per ottenere informazioni sui valori di un determinato campo e in genere per derivare campi flag. Per esempio, è possibile utilizzare la funzione *@BLANK* per creare un campo flag che indica i record contenenti valori vuoti per il campo selezionato. Allo stesso modo, è possibile controllare il tipo di archiviazione per un campo utilizzando qualsiasi funzione relativa al tipo di archiviazione, per esempio *is_string*.

| Funzione | Risultato | Descrizione |
|---------------------------|-----------------|--|
| <i>@BLANK</i> (FIELD) | <i>Booleana</i> | Restituisce un valore vero per tutti i record contenenti valori vuoti in base alle regole di gestione dei vuoti impostate in un nodo Input o Tipo a monte mediante la scheda Tipi. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| <i>@NULL</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record contenenti valori non definiti. I valori non definiti sono rappresentati da valori di sistema nulli, visualizzati in IBM® SPSS® Modeler come \$null\$. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| <i>is_date</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da una data. |
| <i>is_datetime</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da una data, un'ora o un timestamp. |
| <i>is_integer</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da un numero intero. |
| <i>is_number</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da un numero. |
| <i>is_real</i> (ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da un numero reale. |

| Funzione | Risultato | Descrizione |
|--------------------|-----------------|--|
| is_string(ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da una stringa. |
| is_time(ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da un'ora. |
| is_timestamp(ITEM) | <i>Booleana</i> | Restituisce vero per tutti i record il cui tipo è rappresentato da un timestamp. |

Funzioni di conversione

Le funzioni di conversione consentono di creare nuovi campi e convertire il tipo di archiviazione dei file esistenti. Per esempio, per formare nuove stringhe è possibile unire o separare stringhe. Per unire due stringhe, utilizzare l'operatore ><. Per esempio, se il campo Site include il valore "BRAMLEY", allora "xx" >< Site restituisce "xxBRAMLEY". Il risultato di >< è sempre rappresentato da una stringa, anche se gli argomenti non sono stringhe. Di conseguenza, se il campo V1 è 3 e il campo V2 è 5, allora V1 >< V2 restituisce "35" (una stringa, non un numero).

Le funzioni di conversione, e qualsiasi altra funzione che richieda un tipo di input specifico, quale un valore di data o ora, dipendono dai formati correnti specificati nella finestra di dialogo Opzioni stream. Per esempio, se si desidera convertire un campo stringa con valori *Gen 2003, Feb 2003* e così via, selezionare il formato di data corrispondente MES AAAA come formato di data di default per lo stream. [Per ulteriori informazioni, vedere l'argomento Impostazione delle opzioni per gli stream in il capitolo 5 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

| Funzione | Risultato | Descrizione |
|--------------------|------------------|---|
| ITEM1 >< ITEM2 | <i>String</i> | Esegue la concatenazione dei valori di due campi e restituisce la stringa risultante come <i>ELEMENTO1ELEMENTO2</i> . |
| to_integer(ITEM) | <i>Intero</i> | Converte il tipo di archiviazione del campo specificato in un numero intero. |
| to_real(ITEM) | <i>Reale</i> | Converte il tipo di archiviazione del campo specificato in un numero reale. |
| to_number(ITEM) | <i>Number</i> | Converte il tipo di archiviazione del campo specificato in un numero. |
| to_string(ITEM) | <i>String</i> | Converte il tipo di archiviazione del campo specificato in una stringa. |
| to_time(ITEM) | <i>Time</i> | Converte il tipo di archiviazione del campo specificato in un'ora. |
| to_date(ITEM) | <i>Date</i> | Converte il tipo di archiviazione del campo specificato in una data. |
| to_timestamp(ITEM) | <i>Timestamp</i> | Converte il tipo di archiviazione del campo specificato in un timestamp. |

| Funzione | Risultato | Descrizione |
|---------------------|------------|---|
| to_datetime(ITEM) | Data e ora | Converte il tipo di archiviazione del campo specificato in un valore di data, ora o timestamp. |
| datetime_date(ITEM) | Date | Restituisce il valore di data per un <i>numero</i> , una <i>stringa</i> o un <i>timestamp</i> . Si noti che questa è la sola funzione che consente di convertire un numero (in secondi) nuovamente in una data. Se ITEM è una stringa, crea una data analizzando una stringa nel formato di data corrente. La funzione verrà eseguita correttamente solo se il formato di data specificato nella finestra di dialogo Proprietà stream è corretto. Se ITEM è un numero, viene interpretato come numero di secondi dalla data (o epoca) di riferimento. Le frazioni di giorno vengono troncate. Se ITEM è un timestamp, viene restituita la parte relativa alla data del timestamp. Se ITEM è una data, la data viene restituita invariata. |

Funzioni di confronto

Le funzioni di confronto vengono utilizzate per confrontare i valori dei campi tra di loro oppure con una stringa specificata. Per esempio, è possibile controllare l'uguaglianza delle stringhe utilizzando =. Un esempio di verifica dell'uguaglianza delle stringhe è rappresentato da: `Class = "class 1"`.

Ai fini del confronto numerico, *maggiore* indica una maggiore vicinanza all'infinito positivo, mentre *minore* corrisponde a una maggiore prossimità all'infinito negativo. Tutti i numeri negativi sono quindi minori di qualsiasi numero positivo.

| Funzione | Risultato | Descrizione |
|---------------------------------|-----------|---|
| count_equal(ITEM1, LIST) | Intero | Restituisce il numero di valori da un elenco di campi che sono uguali a <i>ELEMENTO1</i> oppure nullo se <i>ELEMENTO1</i> è nullo. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2 . |
| count_greater_than(ITEM1, LIST) | Intero | Restituisce il numero di valori da un elenco di campi che sono maggiori di <i>ELEMENTO1</i> oppure nullo se <i>ELEMENTO1</i> è nullo. |
| count_less_than(ITEM1, LIST) | Intero | Restituisce il numero di valori da un elenco di campi che sono minori di <i>ELEMENTO1</i> oppure nullo se <i>ELEMENTO1</i> è nullo. |
| count_not_equal(ITEM1, LIST) | Intero | Restituisce il numero di valori da un elenco di campi che non sono uguali a <i>ELEMENTO1</i> oppure nullo se <i>ELEMENTO1</i> è nullo. |
| count_nulls(LIST) | Intero | Restituisce il numero di valori nulli da un elenco di campi. |
| count_non_nulls(LIST) | Intero | Restituisce il numero di valori non nulli da un elenco di campi. |
| date_before(ITEM1, ITEM2) | Booleana | Viene utilizzata per controllare l'ordine dei valori di data. Restituisce un valore vero se <i>DATA1</i> precede <i>DATA2</i> . |

| Funzione | Risultato | Descrizione |
|----------------------------|------------------|---|
| first_index(ITEM, LIST) | <i>Intero</i> | Restituisce l'indice del primo campo contenente ELEMENTO da un ELENCO di campi oppure 0 se il valore non viene trovato. Supportata solo per tipi stringa, numero intero e numero reale. Per ulteriori informazioni, vedere l'argomento Utilizzo di dati a risposta multipla in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| first_non_null(LIST) | <i>Qualsiasi</i> | Restituisce il primo valore non nullo nell'elenco di campi fornito. Sono supportati tutti i tipi di archiviazione. |
| first_non_null_index(LIST) | <i>Intero</i> | Restituisce l'indice del primo campo dell'ELENCO specificato contenente un valore non nullo oppure 0 se tutti i valori sono nulli. Sono supportati tutti i tipi di archiviazione. |
| ITEM1 = ITEM2 | <i>Booleana</i> | Restituisce vero per i record in cui ELEMENTO1 è uguale a ELEMENTO2. |
| ITEM1 /= ITEM2 | <i>Booleana</i> | Restituisce vero se le due stringhe non sono identiche e 0 se sono identiche. |
| ITEM1 < ITEM2 | <i>Booleana</i> | Restituisce vero per i record in cui ELEMENTO1 è minore di ELEMENTO2. |
| ITEM1 <= ITEM2 | <i>Booleana</i> | Restituisce vero per i record in cui ELEMENTO1 è minore o uguale a ELEMENTO2. |
| ITEM1 > ITEM2 | <i>Booleana</i> | Restituisce vero per i record in cui ELEMENTO1 è maggiore di ELEMENTO2. |
| ITEM1 >= ITEM2 | <i>Booleana</i> | Restituisce vero per i record in cui ELEMENTO1 è maggiore o uguale a ELEMENTO2. |
| last_index(ITEM, LIST) | <i>Intero</i> | Restituisce l'indice dell'ultimo campo contenente ELEMENTO da un ELENCO di campi oppure 0 se il valore non viene trovato. Supportata solo per tipi stringa, numero intero e numero reale. Per ulteriori informazioni, vedere l'argomento Utilizzo di dati a risposta multipla in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| last_non_null(LIST) | <i>Qualsiasi</i> | Restituisce l'ultimo valore non nullo nell'elenco di campi fornito. Sono supportati tutti i tipi di archiviazione. |
| last_non_null_index(LIST) | <i>Intero</i> | Restituisce l'indice dell'ultimo campo dell'ELENCO specificato contenente un valore non nullo oppure 0 se tutti i valori sono nulli. Sono supportati tutti i tipi di archiviazione. |
| max(ITEM1, ITEM2) | <i>Qualsiasi</i> | Restituisce il maggiore dei due elementi, ELEMENTO1 o ELEMENTO2. |
| max_index(LIST) | <i>Intero</i> | Restituisce l'indice del campo contenente il valore massimo da un elenco di campi numerici oppure 0 se tutti i valori sono nulli. Per esempio, se il terzo campo elencato contiene il valore massimo, viene restituito il valore di indice 3. Se più campi contengono il valore massimo, viene restituito quello elencato per primo (più a sinistra). Per ulteriori informazioni, vedere l'argomento Utilizzo di dati a risposta multipla in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| max_n(LIST) | <i>Number</i> | Restituisce il valore massimo da un elenco di campi numerici oppure nullo se tutti i valori di campo sono nulli. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |

| Funzione | Risultato | Descrizione |
|---------------------------|-----------|--|
| member(ITEM, LIST) | Booleana | Restituisce vero se <i>ELEMENTO</i> è membro dell'elenco <i>ELENCO</i> specificato. In caso contrario, verrà restituito un valore falso. È inoltre possibile specificare un elenco di nomi di campo. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| min(ITEM1, ITEM2) | Qualsiasi | Restituisce il minore dei due elementi, <i>ELEMENTO1</i> o <i>ELEMENTO2</i> . |
| min_index(LIST) | Intero | Restituisce l'indice del campo contenente il valore minimo da un elenco di campi numerici oppure 0 se tutti i valori sono nulli. Per esempio, se il terzo campo elencato contiene il valore minimo, viene restituito il valore di indice 3. Se più campi contengono il valore minimo, viene restituito quello elencato per primo (più a sinistra). Per ulteriori informazioni, vedere l'argomento Utilizzo di dati a risposta multipla in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| min_n(LIST) | Number | Restituisce il valore minimo da un elenco di campi numerici oppure nullo se tutti i valori di campo sono nulli. |
| time_before(TIME1, TIME2) | Booleana | Viene utilizzata per controllare l'ordine dei valori di ora. Restituisce un valore vero se <i>ORA1</i> precede <i>ORA2</i> . |
| value_at(INT, LIST) | | Restituisce il valore di ogni campo elencato alla distanza INT oppure NULLO se la distanza non è compresa nell'intervallo di valori validi, cioè minore di 1 o maggiore del numero dei campi elencati. Sono supportati tutti i tipi di archiviazione. |

Funzioni logiche

Le espressioni CLEM possono essere utilizzate per eseguire operazioni logiche.

| Funzione | Risultato | Descrizione |
|-----------------|-----------|--|
| COND1 and COND2 | Booleana | Questa operazione rappresenta una congiunzione logica e restituisce un valore vero se <i>COND1</i> e <i>COND2</i> sono entrambe vere. Se <i>COND1</i> è falsa, la valutazione di <i>COND2</i> non verrà eseguita. Si può quindi disporre di congiunzioni in cui <i>COND1</i> verifica innanzitutto la validità di un'operazione in <i>COND2</i> . Per esempio, <code>length(Label) >=6</code> e <code>Label(6) = 'x'</code> . |
| COND1 or COND2 | Booleana | Questa operazione rappresenta una disgiunzione logica inclusiva e restituisce un valore vero se <i>COND1</i> o <i>COND2</i> risulta vera oppure se entrambe vengono rilevate come vere. Se <i>COND1</i> è vera, la valutazione di <i>COND2</i> non verrà eseguita. |
| not(COND) | Booleana | Questa operazione rappresenta una negazione logica e restituisce un valore vero se <i>COND</i> è falsa. In caso contrario, questa operazione restituirà un valore 0. |

| Funzione | Risultato | Descrizione |
|---|-----------|---|
| if COND then EXPR1 else EXPR2 endif | Qualsiasi | Questa operazione rappresenta una valutazione condizionale. Se <i>COND</i> è vera, questa operazione restituirà il risultato di <i>EXPR1</i> . Altrimenti, verrà restituito il risultato della valutazione di <i>EXPR_N</i> . |
| if COND1 then EXPR1 elseif COND2 then EXPR2 else EXPR_N endif | Qualsiasi | Questa operazione rappresenta una valutazione condizionale con più diramazioni. Se <i>COND1</i> risulta vera, questa operazione restituirà il risultato di <i>EXPR1</i> . In caso contrario, se <i>COND2</i> è vera, restituirà il risultato della valutazione di <i>EXPR2</i> . Altrimenti, verrà restituito il risultato della valutazione di <i>EXPR_N</i> . |

Funzioni numeriche

CLEM include alcune funzioni numeriche di uso comune.

| Funzione | Risultato | Descrizione |
|---------------|-----------|--|
| -NUM | Number | Viene utilizzata per negare <i>NUM</i> . Restituisce il numero corrispondente con segno opposto. |
| NUM1 + NUM2 | Number | Restituisce la somma di <i>NUM1</i> e <i>NUM2</i> . |
| codice -NUM2 | Number | Restituisce il risultato della sottrazione del valore di <i>NUM2</i> da <i>NUM1</i> . |
| NUM1 * NUM2 | Number | Restituisce il valore di <i>NUM1</i> moltiplicato per <i>NUM2</i> . |
| NUM1 / NUM2 | Number | Restituisce il resto della divisione di <i>NUM1</i> diviso per <i>NUM2</i> . |
| INT1 div INT2 | Number | Viene utilizzata per eseguire la divisione di numeri interi. Restituisce il valore di <i>INT1</i> diviso per <i>INT2</i> . |
| INT1 rem INT2 | Number | Restituisce il resto della divisione di <i>INT1</i> diviso per <i>INT2</i> . Per esempio, $INT1 - (INT1 \text{ div } INT2) * INT2$. |
| INT1 mod INT2 | Number | Questa funzione è obsoleta. Utilizzare la funzione <i>rem</i> . |
| BASE ** POWER | Number | Restituisce il valore <i>BASE</i> elevato alla potenza <i>POTENZA</i> . Entrambi i valori possono essere rappresentati da qualsiasi numero, ma <i>BASE</i> non deve essere uguale a zero se <i>POTENZA</i> corrisponde a un valore zero di tipo diverso dal numero intero 0. Se <i>POTENZA</i> rappresenta un numero intero, il calcolo viene eseguito mediante moltiplicazioni successive delle potenze di <i>BASE</i> . Di conseguenza, se <i>BASE</i> è un numero intero, il risultato sarà rappresentato da un numero intero. Se <i>POTENZA</i> corrisponde al numero intero 0, il risultato sarà sempre rappresentato da un 1 dello stesso tipo di <i>BASE</i> . In caso contrario, ovvero se <i>POWER</i> non corrisponde a un numero intero, il risultato verrà calcolato come $\exp(POWER * \log(BASE))$. |
| abs(NUM) | Number | Restituisce il valore assoluto di <i>NUM</i> , che corrisponde sempre a un numero dello stesso tipo. |
| exp(NUM) | Reale | Restituisce <i>e</i> elevato alla potenza <i>NUM</i> , dove <i>e</i> rappresenta la base dei logaritmi naturali. |
| fracof(NUM) | Reale | Restituisce la parte frazionaria di <i>NUM</i> , definita come $NUM - \text{intof}(NUM)$. |
| intof(NUM) | Intero | Tronca l'argomento in modo da ottenere un numero intero. Restituisce il numero intero dello stesso segno di <i>NUM</i> e con l'ampiezza maggiore per cui $\text{abs}(INT) \leq \text{abs}(NUM)$. |

| Funzione | Risultato | Descrizione |
|--------------|-----------|---|
| log(NUM) | Reale | Restituisce il logaritmo naturale (base e) di NUM , che non deve essere uno zero di alcun tipo. |
| log10(NUM) | Reale | Restituisce il logaritmo in base 10 di NUM , che non deve essere uno zero di alcun tipo. Questa funzione è definita come $\log(NUM) / \log(10)$. |
| negate(NUM) | Number | Viene utilizzata per negare NUM . Restituisce il numero corrispondente con segno opposto. |
| round(NUM) | Intero | Viene utilizzata per arrotondare NUM a un numero intero come $\text{intof}(NUM+0.5)$ se NUM è positivo oppure come $\text{intof}(NUM-0.5)$ se NUM è negativo. |
| sign(NUM) | Number | Viene utilizzata per determinare il segno di NUM . Se NUM è un numero intero, questa operazione restituirà -1 , 0 o 1 . Se NUM è un numero reale, restituirà -1.0 , 0.0 , or 1.0 , a seconda che NUM sia negativo, uguale a zero o positivo. |
| sqrt(NUM) | Reale | Restituisce la radice quadrata di NUM . NUM deve essere positivo. |
| sum_n(LIST) | Number | Restituisce la somma di valori da un elenco di campi numerici oppure nullo se tutti i valori di campo sono nulli. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| mean_n(LIST) | Number | Restituisce il valore medio da un elenco di campi numerici oppure nullo se tutti i valori di campo sono nulli. |
| sdev_n(LIST) | Number | Restituisce la deviazione standard da un elenco di campi numerici oppure nullo se tutti i valori di campo sono nulli. |

Funzioni trigonometriche

Tutte le funzioni illustrate in questa sezione accettano un angolo come argomento oppure ne restituiscono uno come risultato. In entrambi i casi, le unità dell'angolo, rappresentate da radianti o gradi, vengono controllate mediante l'impostazione dell'opzione corrispondente dello stream.

| Funzione | Risultato | Descrizione |
|-----------------------|-----------|--|
| arccos(NUM) | Reale | Calcola l'arcocoseno dell'angolo specificato. |
| arccosh(NUM) | Reale | Calcola l'arcocoseno iperbolico dell'angolo specificato. |
| arcsin(NUM) | Reale | Calcola l'arcoseno dell'angolo specificato. |
| arcsinh(NUM) | Reale | Calcola l'arcoseno iperbolico dell'angolo specificato. |
| arctan(NUM) | Reale | Calcola l'arcotangente dell'angolo specificato. |
| arctan2(NUM_Y, NUM_X) | Reale | Calcola l'arcotangente di NUM_Y / NUM_X e utilizza i segni dei due numeri per derivare le informazioni sui quadranti. Il risultato è un numero reale nell'intervallo $-\pi < \text{ANGLE} \leq \pi$ (radians) – $-180 < \text{ANGLE} \leq 180$ (degrees) |
| arctanh(NUM) | Reale | Calcola l'arcotangente iperbolico dell'angolo specificato. |
| cos(NUM) | Reale | Calcola il coseno dell'angolo specificato. |
| cosh(NUM) | Reale | Calcola il coseno iperbolico dell'angolo specificato. |
| pi | Reale | Questa costante è la migliore approssimazione reale di π . |
| sin(NUM) | Reale | Calcola il seno dell'angolo specificato. |
| sinh(NUM) | Reale | Calcola il seno iperbolico dell'angolo specificato. |

| Funzione | Risultato | Descrizione |
|-----------|-----------|---|
| tan(NUM) | Reale | Calcola la tangente dell'angolo specificato. |
| tanh(NUM) | Reale | Calcola la tangente iperbolica dell'angolo specificato. |

Funzioni di probabilità

Le funzioni di probabilità restituiscono probabilità in base a varie distribuzioni, quali la probabilità che un valore dalla distribuzione t di Student sia minore di un determinato valore.

| Funzione | Risultato | Descrizione |
|-------------------------------|-----------|--|
| cdf_chisq(NUM, DF) | Reale | Restituisce la probabilità che un valore dalla distribuzione chi-quadrato con i gradi di libertà specificati sia inferiore al numero specificato. |
| cdf_f(NUM, DF1, DF2) | Reale | Restituisce la probabilità che un valore dalla distribuzione F , con gradi di libertà $DF1$ e $DF2$, sia inferiore al numero specificato. |
| cdf_normal(NUM, MEAN, STDDEV) | Reale | Restituisce la probabilità che un valore dalla distribuzione normale con la deviazione standard e media specificate sia inferiore al numero specificato. |
| cdf_t(NUM, DF) | Reale | Restituisce la probabilità che un valore dalla distribuzione t per studenti con i gradi di libertà specificati sia inferiore al numero specificato. |

Operazioni bitwise sui numeri interi

Queste funzioni consentono la manipolazione dei numeri interi come schemi di bit rappresentanti valori in complemento a due, in cui alla posizione di bit N corrisponde il peso 2^{*N} . Ai bit viene applicata una numerazione crescente a partire da 0. Queste operazioni vengono eseguite come se il bit del segno di un numero intero venisse esteso a sinistra in modo indefinito. Di conseguenza, in qualsiasi posizione precedente al bit più significativo, un numero intero positivo dispone di 0 bit e un numero intero negativo di 1 bit.

Nota: non è possibile chiamare le funzioni bitwise da script. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|---------------|-----------|---|
| ~~ INT1 | Intero | Genera il complemento bitwise del numero intero $INT1$. Nel risultato è quindi presente un 1 per ciascuna posizione di bit in cui si riscontra uno 0 in $INT1$. È sempre vero che $~~ INT = -(INT + 1)$. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| INT1 INT2 | Intero | Il risultato di questa operazione è l'OR inclusivo bitwise di $INT1$ e $INT2$. Nel risultato è quindi presente un 1 per ogni posizione di bit in cui si riscontra un 1 in $INT1$ o $INT2$ oppure in entrambi. |
| INT1 & INT2 | Intero | Il risultato di questa operazione è l'OR esclusivo bitwise di $INT1$ e $INT2$. Nel risultato è quindi presente un 1 per ogni posizione di bit in cui si riscontra un 1 in $INT1$ o in $INT2$, ma non in entrambi. |

| Funzione | Risultato | Descrizione |
|--------------------------------------|-----------------|--|
| <code>INT1 && INT2</code> | <i>Intero</i> | Genera l'AND bitwise dei numeri interi <i>INT1</i> e <i>INT2</i> . Nel risultato è quindi presente un 1 per ciascuna posizione di bit in cui si riscontra un 1 sia in <i>INT1</i> che in <i>INT2</i> . |
| <code>INT1 && ~ INT2</code> | <i>Intero</i> | Genera l'AND bitwise di <i>INT1</i> e il complemento bitwise di <i>INT2</i> . Nel risultato è quindi presente un 1 per ogni posizione di bit in cui si riscontra un 1 in <i>INT1</i> e uno 0 in <i>INT2</i> . Equivale a <code>INT1 && (~INT2)</code> e si rivela utile per cancellare i bit di <i>INT1</i> impostati in <i>INT2</i> . |
| <code>INT << N</code> | <i>Intero</i> | Genera lo schema di bit di <i>INT1</i> con spostamento a sinistra di <i>N</i> posizioni. L'utilizzo di un valore negativo per <i>N</i> determina uno spostamento a destra. |
| <code>INT >> N</code> | <i>Intero</i> | Genera lo schema di bit di <i>INT1</i> con spostamento a destra di <i>N</i> posizioni. L'utilizzo di un valore negativo per <i>N</i> determina uno spostamento a sinistra. |
| <code>INT1 &&=_0 INT2</code> | <i>Booleana</i> | Equivale all'espressione booleana <code>INT1 && INT2 == 0</code> , ma offre una maggiore efficienza. |
| <code>INT1 &&/_0 INT2</code> | <i>Booleana</i> | Equivale all'espressione booleana <code>INT1 && INT2 == 0</code> , ma offre una maggiore efficienza. |
| <code>integer_bitcount(INT)</code> | <i>Intero</i> | Conta il numero di bit 1 o 0 nella rappresentazione in complemento a due di <i>INT</i> . Se <i>INT</i> non è negativo, <i>N</i> corrisponderà al numero di bit 1. Se <i>INT</i> è negativo, rappresenterà il numero di bit 0. A causa dell'estensione del segno, in un numero intero non negativo è presente un numero infinito di bit 0 e un numero intero negativo contiene un numero infinito di bit 1. È sempre <code>integer_bitcount(INT) = integer_bitcount(-(INT+1))</code> . |
| <code>integer_leastbit(INT)</code> | <i>Intero</i> | Restituisce la posizione di bit <i>N</i> dell'insieme di bit meno significativo nel numero intero <i>INT</i> . <i>N</i> rappresenta la più elevata potenza di 2 per cui <i>INT</i> può essere diviso esattamente. |
| <code>integer_length(INT)</code> | <i>Intero</i> | Restituisce la lunghezza in bit di <i>INT</i> come intero in complemento a due. Ovvero, <i>N</i> è l'intero più piccolo tale che <code>INT < (1 << N)</code> if <code>INT >= 0</code> <code>INT >= (-1 << N)</code> if <code>INT < 0</code> . Se <i>INT</i> non è negativo, per la rappresentazione di <i>INT</i> come numero intero senza segno sarà richiesto un campo di almeno <i>N</i> bit. In alternativa, per rappresentare <i>INT</i> come numero intero con segno, sarà richiesto un minimo di <i>N+1</i> bit, indipendentemente dal segno. |
| <code>testbit(INT, N)</code> | <i>Booleana</i> | Verifica il bit in posizione <i>N</i> nel numero intero <i>INT</i> e restituisce lo stato del bit <i>N</i> sotto forma di valore booleano, ovvero vero per 1 e falso per 0. |

Funzioni casuali

Le funzioni illustrate di seguito vengono utilizzate per la selezione casuale di elementi o la generazione casuale di numeri.

| Funzione | Risultato | Descrizione |
|--------------|------------------|---|
| oneof(LIST) | <i>Qualsiasi</i> | Restituisce un elemento di <i>ELENCO</i> scelto in modo casuale. Gli elementi dell'elenco devono essere specificati come [ITEM1,ITEM2,...,ITEM_N]. Si noti che è inoltre possibile specificare un elenco di nomi di campo. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| random(NUM) | <i>Number</i> | Restituisce un numero casuale distribuito in modo uniforme dello stesso tipo (<i>INT</i> o <i>REALE</i>), da 1 a <i>NUM</i> . Se si utilizza un numero intero, verranno restituiti solo numeri interi. Se si utilizza un numero reale (decimale), verranno restituiti numeri reali la cui precisione decimale è determinata dalle opzioni dello stream. Il numero casuale più grande restituito dalla funzione può essere uguale a <i>NUM</i> . |
| random0(NUM) | <i>Number</i> | Ha le stesse proprietà di <i>random(NUM)</i> , ma a partire da 0. Il numero casuale più grande restituito dalla funzione non sarà mai uguale a <i>X</i> . |

Funzioni stringa

Nel linguaggio CLEM sono supportate le seguenti operazioni con le stringhe:

- Confronto di stringhe
- Creazione di stringhe
- Accesso ai caratteri

In un'espressione CLEM, una stringa è rappresentata da qualsiasi sequenza di caratteri racchiusa tra virgolette doppie corrispondenti ("string quotes"). I caratteri (*CHAR*) possono essere rappresentati da qualsiasi carattere alfanumerico singolo. Sono dichiarati in espressioni CLEM che utilizzano virgolette singole a barra rovesciata nel formato `<carattere>`, per esempio ``z``, ``A`` o ``2``. I caratteri che rappresentano indici fuori intervallo o negativi in una stringa determineranno un comportamento non definito.

Nota: i confronti tra stringhe che utilizzano e non utilizzano il push back SQL potrebbero generare risultati diversi in presenza di spazi finali.

| Funzione | Risultato | Descrizione |
|-------------------------------|-----------------|--|
| allbutfirst(N, STRING) | <i>String</i> | Restituisce una stringa risultante dalla rimozione dei primi <i>N</i> caratteri da <i>STRINGA</i> . |
| allbutlast(N, STRING) | <i>String</i> | Restituisce una stringa risultante dalla rimozione degli ultimi caratteri da <i>STRINGA</i> . |
| alphabefore(STRING1, STRING2) | <i>Booleana</i> | Viene utilizzata per controllare l'ordine alfabetico delle stringhe. Restituisce vero se <i>STRINGA1</i> precede <i>STRINGA2</i> . |

| Funzione | Risultato | Descrizione |
|------------------------------------|-----------|---|
| endstring(LENGTH, STRING) | String | Estrae gli ultimi <i>N</i> caratteri dalla stringa specificata. Se la lunghezza della stringa è inferiore o uguale alla lunghezza specificata, rimane invariata. |
| hasendstring(STRING, SUBSTRING) | Intero | La funzione è analoga a <code>isendstring(SUBSTRING, STRING)</code> . |
| hasmidstring(STRING, SUBSTRING) | Intero | La funzione è analoga a <code>ismidstring(SUBSTRING, STRING)</code> (sottostringa incorporata). |
| hasstartstring(STRING, SUBSTRING) | Intero | La funzione è analoga a <code>isstartstring(SUBSTRING, STRING)</code> . |
| hassubstring(STRING, N, SUBSTRING) | Intero | La funzione è analoga a <code>issubstring(SUBSTRING, N, STRING)</code> , dove il valore di default di <i>N</i> è 1. |
| count_substring(STRING, SUBSTRING) | Intero | Restituisce il numero di volte in cui la sottostringa specificata è presente all'interno della stringa. Ad esempio, <code>count_substring("foooo.txt", "oo")</code> restituisce 3. |
| hassubstring(STRING, SUBSTRING) | Intero | La funzione è analoga a <code>issubstring(SUBSTRING, 1, STRING)</code> , dove il valore di default di <i>N</i> è 1. |
| isalphacode(CHAR) | Booleana | Restituisce un valore vero se <i>CAR</i> rappresenta un carattere della stringa specificata (spesso un nome di campo) con codice alfabetico. In caso contrario, questa funzione restituirà un valore 0. Per esempio, <code>isalphacode(produce_num(1))</code> . |
| isendstring(SUBSTRING, STRING) | Intero | Se la stringa <i>STRINGA</i> termina con la sottostringa <i>SOTTOSTRINGA</i> , questa funzione restituirà il numero di indice di <i>SOTTOSTRINGA</i> in <i>STRINGA</i> . In caso contrario, questa funzione restituirà un valore 0. |
| islowercode(CHAR) | Booleana | Restituisce il valore vero se <i>CAR</i> rappresenta una lettera minuscola per la stringa specificata (spesso un nome di campo). In caso contrario, restituirà il valore 0. Per esempio, sia <code>islowercode(`)</code> che <code>islowercode(country_name(2))</code> sono espressioni valide. |
| ismidstring(SUBSTRING, STRING) | Intero | Se <i>SOTTOSTRINGA</i> è una sottostringa di <i>STRINGA</i> ma non ha inizio con il primo carattere di <i>STRINGA</i> né termina con l'ultimo, questa funzione restituirà l'indice in corrispondenza del quale inizia la sottostringa. In caso contrario, questa funzione restituirà un valore 0. |
| isnumbercode(CHAR) | Booleana | Restituisce il valore vero se <i>CAR</i> per la stringa specificata (spesso un nome di campo) rappresenta un carattere con codice numerico. In caso contrario, questa funzione restituirà un valore 0. Per esempio, <code>isnumbercode(product_id(2))</code> . |

| Funzione | Risultato | Descrizione |
|---|-----------------|--|
| isstartstring(SUBSTRING, STRING) | <i>Intero</i> | Se la stringa <i>STRINGA</i> ha inizio con la sottostringa <i>SOTTOSTRINGA</i> , questa funzione restituirà l'indice 1. In caso contrario, restituirà il valore 0. |
| issubstring(SUBSTRING, N, STRING) | <i>Intero</i> | Cerca una sottostringa uguale a <i>SOTTOSTRINGA</i> all'interno della stringa <i>STRINGA</i> , a partire dal carattere <i>N</i> . Se la ricerca ha esito positivo, questa funzione restituisce il valore di indice intero in corrispondenza del quale ha inizio la sottostringa corrispondente. In caso contrario, restituirà il valore 0. Se non viene specificato, <i>N</i> verrà impostato per default su 1. |
| issubstring(SUBSTRING, STRING) | <i>Intero</i> | Cerca una sottostringa uguale a <i>SOTTOSTRINGA</i> all'interno della stringa <i>STRINGA</i> , a partire dal carattere <i>N</i> . Se la ricerca ha esito positivo, questa funzione restituisce il valore di indice intero in corrispondenza del quale ha inizio la sottostringa corrispondente. In caso contrario, restituirà il valore 0. Se non viene specificato, <i>N</i> verrà impostato per default su 1. |
| issubstring_count(SUBSTRING, N, STRING): | <i>Intero</i> | Restituisce l'indice dell' <i>N</i> esima occorrenza della <i>SOTTOSTRINGA</i> all'interno della <i>STRINGA</i> specificata. Se le occorrenze della <i>SOTTOSTRINGA</i> sono inferiori a <i>N</i> , viene restituito 0. |
| issubstring_lim(SUBSTRING, N, STARTLIM, ENDLIM, STRING) | <i>Intero</i> | Questa funzione è equivalente a <i>issubstring</i> , ma la corrispondenza deve iniziare prima dell'indice <i>LIMINIZIO</i> o a partire da esso e terminare con l'indice <i>LIMFINE</i> o prima di esso. I vincoli <i>LIMINIZIO</i> o <i>LIMFINE</i> possono essere disattivati fornendo un valore falso per questi argomenti, per esempio <i>issubstring_lim(SUBSTRING, N, false, false, STRING)</i> equivale a <i>issubstring</i> . |
| isuppercode(CHAR) | <i>Booleana</i> | Restituisce un valore vero se <i>CAR</i> rappresenta una lettera maiuscola. In caso contrario, restituirà il valore 0. Per esempio, sia <i>isuppercode('')</i> che <i>isuppercode(country_name(2))</i> sono espressioni valide. |
| last(CHAR) | <i>String</i> | Restituisce l'ultimo carattere <i>CAR</i> di <i>STRINGA</i> (che deve contenere almeno un carattere). |
| length(STRING) | <i>Intero</i> | Restituisce la lunghezza della stringa <i>STRINGA</i> , ovvero il numero di caratteri in essa contenuti. |

| Funzione | Risultato | Descrizione |
|---|---------------|---|
| locchar(CHAR, N, STRING) | Intero | Utilizzata per identificare la posizione dei caratteri nei campi simbolici. La funzione cerca il carattere <i>CAR</i> all'interno della stringa <i>STRINGA</i> , iniziando la ricerca dal carattere <i>Nesimo</i> di <i>STRINGA</i> . Questa funzione restituisce un valore che indica la posizione (a partire da <i>N</i>) in cui si trova il carattere. Se il carattere non viene trovato, questa funzione restituirà un valore 0. Se la funzione include una distanza (<i>N</i>) non valida (per esempio una distanza superiore alla lunghezza della stringa), la funzione restituisce \$null\$. Per esempio, locchar(`n`, 2, web_page) ricerca all'interno del campo <i>web_page</i> il carattere `n` a partire dal secondo carattere del valore del campo. <i>Nota:</i> racchiudere il carattere specificato tra virgolette singole a barra rovesciata. |
| locchar_back(CHAR, N, STRING) | Intero | Funzione analoga a locchar, tranne per il fatto che la ricerca viene eseguita all'indietro, a partire dall' <i>Nesimo</i> carattere. Per esempio, locchar_back(`n`, 9, web_page) esegue la ricerca all'interno del campo <i>web_page</i> a partire dal nono carattere e procedendo all'indietro verso l'inizio della stringa. Se contiene un margine non valido, per esempio superiore alla lunghezza della stringa, questa funzione restituirà \$null\$. Idealmente, sarà necessario utilizzare locchar_back insieme alla funzione length(<field>) per utilizzare dinamicamente la lunghezza del valore corrente del campo. Per esempio, locchar_back(`n`, (length(web_page)), web_page). |
| lowertoupper(CHAR) lowertoupper (STRING) | CAR o Stringa | L'input può essere rappresentato da una stringa o da un carattere, utilizzato in questa funzione per restituire un nuovo elemento dello stesso tipo, con i caratteri minuscoli convertiti nei relativi maiuscoli. Per esempio, lowertoupper(`a`), lowertoupper("My string") e lowertoupper(field_name(2)) sono espressioni valide. |
| matches | Booleana | Restituisce vero se una stringa corrisponde allo schema specificato. Lo schema deve essere una stringa letterale e non può essere un nome di campo contenente uno schema. È possibile includere nello schema un punto interrogativo (?) corrispondente esattamente a un carattere; un asterisco (*) corrisponde a zero o più caratteri. Per indicare un punto interrogativo o un asterisco letterale, ovvero non utilizzato come carattere jolly, è possibile utilizzare la barra rovesciata (\). |

| Funzione | Risultato | Descrizione |
|--|-----------|---|
| replace(SUBSTRING, NEWSUBSTRING, STRING) | String | All'interno della <i>STRINGA</i> specificata, sostituisce tutte le istanze della <i>SOTTOSTRINGA</i> con la <i>NUOVASOTTOSTRINGA</i> . |
| replicate(COUNT, STRING) | String | Restituisce una stringa costituita dalla stringa originale copiata il numero di volte specificato. |
| stripchar(Char,STRING) | String | Consente di rimuovere i caratteri specificati da una stringa o un campo. È possibile utilizzare questa funzione, per esempio, per rimuovere dai dati i simboli aggiuntivi, come i simboli di valuta, in modo da ottenere un nome o un numero semplice. Per esempio, la sintassi stripchar('\$', 'Cost') restituisce un nuovo campo in cui il simbolo del dollaro è stato rimosso da tutti i valori. <i>Nota:</i> racchiudere il carattere specificato tra virgolette singole a barra rovesciata. |
| skipchar(Char, N, STRING) | Intero | Cerca nella stringa <i>STRINGA</i> qualsiasi carattere diverso da <i>Char</i> , a partire dall' <i>Nesimo</i> carattere. Questa funzione restituisce una sottostringa di numeri interi che indica il punto in cui è stato trovato un carattere oppure 0, se ogni carattere successivo a <i>Nesimo</i> corrisponde a <i>Char</i> . Se contiene un margine non valido, per esempio superiore alla lunghezza della stringa, questa funzione restituirà \$null\$. locchar è spesso utilizzato insieme alle funzioni skipchar per determinare il valore di <i>N</i> (punto dal quale iniziare la ricerca nella stringa). Per esempio, skipchar('s', (locchar('s', 1, "MyString")), "MyString"). |
| skipchar_back(Char, N, STRING) | Intero | Questa funzione è analoga a skipchar, tranne che per il fatto che la ricerca viene eseguita a ritroso , a partire dall' <i>Nesimo</i> carattere. |
| startstring(LENGTH, STRING) | String | Estrae i primi <i>N</i> caratteri dalla stringa specificata. Se la lunghezza della stringa è inferiore o uguale alla lunghezza specificata, rimane invariata. |
| strmember(Char, STRING) | Intero | Equivale a locchar(Char, 1, STRING). Restituisce una sottostringa di numeri interi che indica il punto in cui viene rilevata la prima occorrenza di <i>Char</i> oppure 0. Se contiene una distanza non valida, per esempio superiore alla lunghezza della stringa, questa funzione restituirà \$null\$. |
| subscrs(N, STRING) | Char | Restituisce l' <i>Nesimo</i> carattere <i>Char</i> della stringa di input <i>STRINGA</i> . Per questa funzione è inoltre disponibile la forma abbreviata STRING(N). Per esempio, lowertoupper("name"(1)) è un'espressione valida. |

| Funzione | Risultato | Descrizione |
|---|------------------|---|
| substring(N, LEN, STRING) | String | Restituisce una stringa <i>SOTTOSTRINGA</i> , rappresentata dal numero di caratteri di <i>STRINGA</i> indicati da <i>LUNGH</i> , a partire dal carattere in corrispondenza dell'indice <i>N</i> . |
| substring_between(N1, N2, STRING) | String | Restituisce la sottostringa di <i>STRINGA</i> che ha inizio in corrispondenza dell'indice <i>N1</i> e termina con l'indice <i>N2</i> . |
| trim(STRING) | String | Rimuove i caratteri spazio bianco iniziale e finale dalla stringa specificata. |
| trim_start(STRING) | String | Rimuove i caratteri spazio bianco iniziale dalla stringa specificata. |
| trimend(STRING) | String | Rimuove i caratteri spazio bianco finale dalla stringa specificata. |
| unicode_char(NUM) | CAR | Restituisce il carattere con valore Unicode <i>NUM</i> . |
| unicode_value(CHAR) | NUM | Restituisce il valore Unicode di <i>CAR</i> |
| uppertolower(CHAR) uppertolower (STRING) | CAR o Stringa | L'input può essere rappresentato da una stringa o da un carattere, utilizzato in questa funzione per restituire un nuovo elemento dello stesso tipo, con i caratteri maiuscoli convertiti nei relativi minuscoli. <i>Nota:</i> Ricordarsi di specificare le stringhe con le virgolette doppie e i caratteri con le virgolette singole a barra rovesciata. I nomi semplici di campo non devono essere racchiusi tra virgolette. |

Funzioni SoundEx

SoundEx è un metodo utilizzato per trovare stringhe quando si conosce un suono ma non l'esatta ortografia. Sviluppato nel 1918, cerca parole con suoni simili in base a ipotesi fonetiche sulla pronuncia di determinate lettere. Può essere per esempio utilizzato per cercare nomi in un database, in cui ortografia e pronuncia di nomi simili possono variare. L'algoritmo SoundEx di base è documentato in numerose fonti e nonostante i limiti conosciuti (per esempio, la combinazione iniziale ph e la lettera f non corrispondono nonostante il loro suono sia simile) è supportato in qualche forma dalla maggioranza dei database.

| Funzione | Risultato | Descrizione |
|--------------------------------------|-----------|--|
| soundex(STRING) | Intero | Restituisce il codice SoundEx di quattro caratteri per la <i>STRINGA</i> specificata. |
| soundex_difference(STRING1, STRING2) | Intero | Restituisce un numero intero compreso tra 0 e 4 che indica il numero di caratteri che sono identici nella codifica SoundEx per le due stringhe, dove 0 indica nessuna somiglianza e 4 indica una forte somiglianza o stringhe identiche. |

Funzioni di data e ora

Nel linguaggio CLEM è disponibile un gruppo di funzioni per la gestione dei campi con archiviazione di tipo datetime delle variabili che rappresentano le date e le ore. I formati di data e ora utilizzati sono specifici di ogni stream e vengono impostati nella finestra di dialogo delle proprietà dello stream. Le funzioni di data e ora analizzano le stringhe di data e ora in base al formato correntemente selezionato.

Quando si specifica un anno in una data che utilizza solo due cifre, ovvero in cui non è indicato il secolo, verrà utilizzato il secolo di default specificato nella finestra di dialogo delle proprietà dello stream.

Nota: Non è possibile chiamare le funzioni data e ora da script. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|---------------------------|------------|---|
| @TODAY | String | Se si seleziona Cambia giorni/min nella finestra di dialogo delle proprietà dello stream, questa funzione restituirà la data corrente sotto forma di stringa nel formato di data corrente. Se si utilizza un formato di data a due cifre e non si seleziona Cambia giorni/min, questa funzione restituirà \$null\$ sul server corrente. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| to_time(ITEM) | Time | Converte il tipo di archiviazione del campo specificato in un'ora. |
| to_date(ITEM) | Date | Converte il tipo di archiviazione del campo specificato in una data. |
| to_timestamp(ITEM) | Timestamp | Converte il tipo di archiviazione del campo specificato in un timestamp. |
| to_datetime(ITEM) | Data e ora | Converte il tipo di archiviazione del campo specificato in un valore di data, ora o timestamp. |
| datetime_date(ITEM) | Date | Restituisce il valore di data per un <i>numero</i> , una <i>stringa</i> o un <i>timestamp</i> . Si noti che questa è la sola funzione che consente di convertire un numero (in secondi) nuovamente in una data. Se ITEM è una stringa, crea una data analizzando una stringa nel formato di data corrente. La funzione verrà eseguita correttamente solo se il formato di data specificato nella finestra di dialogo Proprietà stream è corretto. Se ITEM è un numero, viene interpretato come numero di secondi dalla data (o epoca) di riferimento. Le frazioni di giorno vengono troncate. Se ITEM è un timestamp, viene restituita la parte relativa alla data del timestamp. Se ITEM è una data, la data viene restituita invariata. |
| date_before(DATE1, DATE2) | Booleana | Restituisce un valore vero se <i>DATE1</i> rappresenta una data o un timestamp precedente a quella rappresentata da <i>DATE2</i> . In caso contrario, questa funzione restituirà un valore 0. |

| Funzione | Risultato | Descrizione |
|---------------------------------------|---------------|--|
| date_days_difference(DATE1, DATE2) | <i>Intero</i> | Restituisce il tempo in giorni dalla data o dal timestamp rappresentato da <i>DATA1</i> a quello rappresentato da <i>DATA2</i> , sotto forma di numero intero. Se <i>DATA2</i> è antecedente a <i>DATA1</i> , questa funzione restituisce un numero negativo. |
| date_in_days(DATE) | <i>Intero</i> | Restituisce il tempo in giorni trascorso dalla data di riferimento alla data o al timestamp rappresentato da <i>DATA</i> , sotto forma di numero intero. Se il valore di <i>DATA</i> è precedente alla data di riferimento, questa funzione restituisce un numero negativo. Per ottenere un calcolo corretto è necessario includere una data valida. Per esempio, 29 febbraio 2001 non è una data valida. Infatti, dal momento che il 2001 non è un anno bisestile, questa data non esiste. |
| date_in_months(DATE) | <i>Reale</i> | Restituisce il tempo in mesi trascorso dalla data di riferimento alla data o al timestamp rappresentato da <i>DATA</i> , sotto forma di numero reale. Tale numero corrisponde a una cifra approssimativa basata su un mese formato da 30.4375 giorni. Se il valore di <i>DATA</i> è precedente alla data di riferimento, questa funzione restituisce un numero negativo. Per ottenere un calcolo corretto è necessario includere una data valida. Per esempio, 29 febbraio 2001 non è una data valida. Infatti, dal momento che il 2001 non è un anno bisestile, questa data non esiste. |
| date_in_weeks(DATE) | <i>Reale</i> | Restituisce il tempo in settimane trascorso dalla data di riferimento alla data o al timestamp rappresentato da <i>DATA</i> , sotto forma di numero reale. Tale numero è basato su una settimana formata da sette giorni. Se il valore di <i>DATA</i> è precedente alla data di riferimento, questa funzione restituisce un numero negativo. Per ottenere un calcolo corretto è necessario includere una data valida. Per esempio, 29 febbraio 2001 non è una data valida. Infatti, dal momento che il 2001 non è un anno bisestile, questa data non esiste. |
| date_in_years(DATE) | <i>Reale</i> | Restituisce il tempo in anni trascorso dalla data di riferimento alla data o al timestamp rappresentato da <i>DATA</i> , sotto forma di numero reale. Tale numero corrisponde a una cifra approssimativa basata su un anno formato da 365.25 giorni. Se il valore di <i>DATA</i> è precedente alla data di riferimento, questa funzione restituisce un numero negativo. Per ottenere un calcolo corretto è necessario includere una data valida. Per esempio, 29 febbraio 2001 non è una data valida. Infatti, dal momento che il 2001 non è un anno bisestile, questa data non esiste. |
| date_months_difference(DATE1, DATE2) | <i>Reale</i> | Restituisce il tempo in mesi dalla data o dal timestamp rappresentato da <i>DATA1</i> a quello rappresentato da <i>DATA2</i> , sotto forma di numero reale. Tale numero corrisponde a una cifra approssimativa basata su un mese formato da 30.4375 giorni. Se <i>DATA2</i> è antecedente a <i>DATA1</i> , questa funzione restituisce un numero negativo. |
| datetime_date(YEAR, MONTH, DAY) | <i>Date</i> | Crea il valore di data per gli argomenti <i>ANNO</i> , <i>MESE</i> e <i>GIORNO</i> specificati. Gli argomenti devono essere numeri interi. |

| Funzione | Risultato | Descrizione |
|--|------------------|--|
| datetime_day(DATE) | <i>Intero</i> | Restituisce il giorno del mese di una <i>DATA</i> o un timestamp. Il risultato è un numero intero nell'intervallo da 1 a 31. |
| datetime_day_name(DAY) | <i>String</i> | Restituisce il nome completo dell'argomento <i>GIORNO</i> specificato. L'argomento deve essere un numero intero nell'intervallo da 1 (domenica) a 7 (sabato). |
| datetime_hour(TIME) | <i>Intero</i> | Restituisce l'ora dal valore <i>ORA</i> o da un timestamp. Il risultato è un numero intero nell'intervallo da 0 a 23. |
| datetime_in_seconds(TIME) | <i>Reale</i> | Restituisce la parte in secondi memorizzata in <i>ORA</i> . |
| datetime_in_seconds(DATE), datetime_in_seconds(DATE-TIME) | <i>Reale</i> | Restituisce il numero accumulato, convertito in secondi, dalla differenza tra il valore <i>DATA</i> o <i>DATAORA</i> corrente e la data di riferimento (1900-01-01). |
| datetime_minute(TIME) | <i>Intero</i> | Restituisce i minuti dal valore <i>ORA</i> o da un timestamp. Il risultato è un numero intero nell'intervallo da 0 a 59. |
| datetime_month(DATE) | <i>Intero</i> | Restituisce il mese da una <i>DATA</i> o da un timestamp. Il risultato è un numero intero nell'intervallo da 1 a 12. |
| datetime_month_name(MONTH) | <i>String</i> | Restituisce il nome completo dell'argomento <i>MESE</i> specificato. L'argomento deve essere un numero intero nell'intervallo da 1 a 12. |
| datetime_now | <i>Timestamp</i> | Restituisce l'ora corrente come timestamp. |
| datetime_second(TIME) | <i>Intero</i> | Restituisce i secondi dal valore <i>ORA</i> o da un timestamp. Il risultato è un numero intero nell'intervallo da 0 a 59. |
| datetime_day_short_name(DAY) | <i>String</i> | Restituisce il nome abbreviato dell'argomento <i>GIORNO</i> specificato. L'argomento deve essere un numero intero nell'intervallo da 1 (domenica) a 7 (sabato). |
| datetime_month_short_name(MONTH) | <i>String</i> | Restituisce il nome abbreviato dell'argomento <i>MESE</i> specificato. L'argomento deve essere un numero intero nell'intervallo da 1 a 12. |
| datetime_time(HOUR, MINUTE, SECOND) | <i>Time</i> | Restituisce il valore di ora per gli argomenti <i>ORA</i> , <i>MINUTO</i> , e <i>SECONDO</i> specificati. Gli argomenti devono essere numeri interi. |
| datetime_time(ITEM) | <i>Time</i> | Restituisce il valore di ora dell' <i>ELEMENTO</i> specificato. |
| datetime_timestamp(YEAR, MONTH, DAY, HOUR, MINUTE, SECOND) | <i>Timestamp</i> | Restituisce il valore di timestamp per l' <i>ANNO</i> , il <i>MESE</i> , il <i>GIORNO</i> , l' <i>ORA</i> , i <i>MINUTE</i> e i <i>SECONDI</i> specificati. |
| datetime_timestamp(DATE, TIME) | <i>Timestamp</i> | Restituisce il valore di timestamp per la <i>DATA</i> e l' <i>ORA</i> specificate. |
| datetime_timestamp (NUMBER) | <i>Timestamp</i> | Restituisce il valore di timestamp del numero di secondi specificato. |
| datetime_weekday(DATE) | <i>Intero</i> | Restituisce il giorno della settimana dalla <i>DATA</i> o dal timestamp specificato. |
| datetime_year(DATE) | <i>Intero</i> | Restituisce l'anno da una <i>DATA</i> o da un timestamp. Il risultato è un numero intero come 2002. |

| Funzione | Risultato | Descrizione |
|--------------------------------------|-----------|--|
| date_weeks_difference (DATE1, DATE2) | Reale | Restituisce il tempo in settimane dalla data o dal timestamp rappresentato da <i>DATA1</i> a quello rappresentato da <i>DATA2</i> , sotto forma di numero reale. Tale numero è basato su una settimana formata da sette giorni. Se <i>DATA2</i> è antecedente a <i>DATA1</i> , questa funzione restituisce un numero negativo. |
| date_years_difference (DATE1, DATE2) | Reale | Restituisce il tempo in anni dalla data o dal timestamp rappresentato da <i>DATA1</i> a quello rappresentato da <i>DATA2</i> , sotto forma di numero reale. Tale numero corrisponde a una cifra approssimativa basata su un anno formato da 365.25 giorni. Se <i>DATA2</i> è antecedente a <i>DATA1</i> , questa funzione restituisce un numero negativo. |
| time_before(TIME1, TIME2) | Booleana | Restituisce un valore vero se <i>ORA1</i> rappresenta un'ora o un timestamp precedente a quello rappresentata da <i>ORA2</i> . In caso contrario, questa funzione restituirà un valore 0. |
| time_hours_difference (TIME1, TIME2) | Reale | Restituisce il tempo trascorso in ore tra le ore o timestamp rappresentati da <i>ORA1</i> e <i>ORA2</i> , sotto forma di numero reale. Se si seleziona Cambia giorni/min nella finestra di dialogo delle proprietà dello stream, un valore superiore di <i>ORA1</i> verrà interpretato come riferito al giorno precedente. Se non si seleziona l'opzione di trasformazione, un valore superiore di <i>ORA1</i> determina la restituzione di un valore negativo. |
| time_in_hours(TIME) | Reale | Restituisce il tempo in ore rappresentato dall'ora <i>TIME</i> , sotto forma di numero reale. Con il formato di ora HHMM, per esempio l'espressione <code>time_in_hours('0130')</code> restituisce 1.5. <i>TIME</i> può rappresentare un'ora o un timestamp. |
| time_in_mins(TIME) | Reale | Restituisce il tempo in minuti rappresentato da <i>ORA</i> , sotto forma di numero reale. <i>TIME</i> può rappresentare un'ora o un timestamp. |
| time_in_secs(TIME) | Intero | Restituisce il tempo in secondi rappresentato da <i>ORA</i> , sotto forma di numero intero. <i>TIME</i> può rappresentare un'ora o un timestamp. |
| time_mins_difference(TIME1, TIME2) | Reale | Restituisce il tempo trascorso in minuti tra le ore o timestamp rappresentati da <i>ORA1</i> e <i>ORA2</i> , sotto forma di numero reale. Se si seleziona Cambia giorni/min nella finestra di dialogo delle proprietà dello stream, un valore superiore a <i>ORA1</i> verrà interpretato come riferito al giorno precedente (o all'ora precedente, se per il formato corrente sono specificati solo ore e minuti). Se non si seleziona l'opzione di trasformazione, un valore superiore a <i>ORA1</i> determina la restituzione di un valore negativo. |
| time_secs_difference(TIME1, TIME2) | Intero | Restituisce la differenza in secondi tra le ore o timestamp rappresentati da <i>ORA1</i> e <i>ORA2</i> , sotto forma di numero intero. Se si seleziona Cambia giorni/min nella finestra di dialogo delle proprietà dello stream, un valore superiore a <i>ORA1</i> verrà interpretato come riferito al giorno precedente (o all'ora precedente, se per il formato corrente sono specificati solo ore e minuti). Se non si seleziona l'opzione di trasformazione, un valore superiore di <i>ORA1</i> determina la restituzione di un valore negativo. |

Conversione dei valori di data e ora

Si noti che le funzioni di conversione, e qualsiasi altra funzione che richieda un tipo di input specifico quale un valore di data o ora, dipendono dai formati correnti specificati nella finestra di dialogo Opzioni stream. Per esempio, un campo denominato *DATA*, archiviato come stringa con i valori *Gen 2003*, *Feb 2003* e così via, potrebbe essere convertito in campo di archiviazione di date nel modo seguente:

```
to_date(DATA)
```

Perché la conversione funzioni è necessario selezionare il formato di data MES AAAA corrispondente come default per lo stream. [Per ulteriori informazioni, vedere l'argomento Impostazione delle opzioni per gli stream in il capitolo 5 in Manuale dell'utente di IBM SPSS Modeler 14.2.](#)

Per un esempio di conversione di valori di stringa in date mediante un nodo Riempimento, vedere lo stream *broadband_create_models.str*, installato nella cartella *\Demos* all'interno della sottocartella *streams*. [Per ulteriori informazioni, vedere l'argomento Previsione mediante il nodo Serie storica in il capitolo 15 in Guida alle applicazioni di IBM SPSS Modeler 14.2.](#)

Date archiviate come numeri. Si noti che *DATA* nell'esempio precedente è il nome di un campo, mentre *to_date* è una funzione CLEM. È possibile convertire le date archiviate come numeri utilizzando la funzione *datetime_date*, in cui il numero viene interpretato come numero di secondi trascorsi dalla data (o epoca) di riferimento.

```
datetime_date(DATA)
```

Mediante la conversione di una data in un numero di secondi e viceversa, è possibile effettuare calcoli, quali la data corrente più o meno un numero determinato di giorni. Per esempio:

```
datetime_date((date_in_days(DATA)-7)*60*60*24)
```

Funzioni sequenza

Per alcune operazioni, la sequenza degli eventi si rivela importante. L'applicazione supporta le seguenti sequenze di record:

- Sequenze e serie storiche
- Funzioni sequenza
- Indicizzazione di record
- Calcolo della media, somma e confronto di valori
- Monitoraggio delle modifiche e differenziazione
- @SINCE
- Valori di distanza
- Funzionalità aggiuntive relative alle sequenze

In numerose applicazioni, ogni record in uno stream può essere considerato come un caso singolo, indipendente da tutti gli altri. In queste circostanze, l'ordine dei record è in genere irrilevante.

Per alcune classi di problemi, tuttavia, la sequenza di record è estremamente importante. Si tratta in genere di situazioni relative a serie storiche, in cui la sequenza di record rappresenta una sequenza ordinata di occorrenze ed eventi. Ogni record rappresenta un'istantanea di un momento specifico nel tempo. È tuttavia possibile che la maggior parte delle informazioni utili non sia contenuta nei valori del momento, bensì risieda nella variazione e nel comportamento di tali valori nel corso del tempo.

Il parametro rilevante può anche non essere rappresentato dal tempo. Gli stessi principi possono essere applicati anche a record che rappresentano, per esempio, analisi eseguite a distanze diverse lungo una linea.

Le funzioni speciali e di sequenza sono immediatamente riconoscibili grazie alle seguenti caratteristiche:

- Sono tutte precedute dal prefisso @.
- I nomi sono riportati in lettere maiuscole.

Le funzioni sequenza possono fare riferimento al record correntemente elaborato da un nodo, ai record che hanno già attraversato un nodo e anche, in un caso, ai record non ancora passati attraverso un nodo. Le funzioni sequenza possono essere combinate liberamente con altri componenti delle espressioni CLEM, benché alcune presentino restrizioni relative agli argomenti utilizzabili.

Esempi

Può rivelarsi utile conoscere da quanto tempo si è verificato un evento specifico o una condizione si è rivelata vera. A tale scopo, è possibile utilizzare la funzione @SINCE, per esempio:

```
@SINCE(Reddito > Risultati)
```

Questa funzione restituisce la distanza dell'ultimo record in cui questa condizione è stata rilevata come vera, ovvero il numero di record precedenti a quello considerato in cui la condizione era vera. Se la condizione non è mai stata rilevata come vera, @SINCE restituirà @INDEX + 1.

Talvolta può rivelarsi utile fare riferimento a un valore del record corrente nell'espressione utilizzata con @SINCE. A questo scopo, è possibile utilizzare la funzione @THIS, con cui si specifica che il nome di un campo deve essere sempre applicato al record corrente. Per trovare la distanza dell'ultimo record contenente un campo Concentration con valore due volte superiore rispetto a quello del record corrente, si può utilizzare quanto segue:

```
@SINCE(Concentrazione > 2 * @THIS(Concentrazione))
```

In alcuni casi, la condizione specificata per @SINCE è vera per il record corrente per definizione, per esempio:

```
@SINCE(ID == @THIS(ID))
```

Per questo motivo, @SINCE non valuta la condizione per il record corrente. Per valutare la condizione per il record corrente così come per i precedenti, utilizzare l'analoga funzione @SINCE0. Se la condizione è vera nel record corrente, @SINCE0 restituirà 0.

Nota: non è possibile chiamare le funzioni @ da script. Per ulteriori informazioni, vedere l'argomento [Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|-------------------------|-----------|---|
| MEAN(FIELD) | Reale | Restituisce la media dei valori per l'argomento CAMPO o CAMPI specificato. |
| @MEAN(FIELD, EXPR) | Reale | Restituisce la media dei valori per CAMPO negli ultimi record ESPR ricevuti dal nodo corrente, incluso il record corrente. CAMPO deve essere il nome di un campo numerico. ESPR può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se ESPR non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la media dei valori inclusi in tutti i record ricevuti. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| @MEAN(FIELD, EXPR, INT) | Reale | Restituisce la media dei valori per CAMPO negli ultimi record ESPR ricevuti dal nodo corrente, incluso il record corrente. CAMPO deve essere il nome di un campo numerico. ESPR può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se ESPR non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la media dei valori inclusi in tutti i record ricevuti. INT indica il numero massimo di valori da cercare all'indietro. Se si specifica questo terzo argomento, si otterrà un'operazione più precisa. |
| @DIFF1(FIELD) | Reale | Restituisce il primo differenziale di CAMPO1. La forma con un singolo argomento restituisce semplicemente la differenza tra il valore corrente e il valore precedente del campo. Restituisce 0 se per il campo specificato non esiste alcun valore precedente. |
| @DIFF1(FIELD1, FIELD2) | Reale | La forma con due argomenti fornisce il primo differenziale di CAMPO1 rispetto a CAMPO2. Restituisce 0 se per il campo specificato non esiste alcun valore precedente. |
| @DIFF2(FIELD) | Reale | Restituisce il secondo differenziale di CAMPO1. La forma con un singolo argomento restituisce semplicemente la differenza tra il valore corrente e il valore precedente del campo. Restituisce 0 se per il campo specificato non esiste alcun valore precedente. |
| @DIFF2(FIELD1, FIELD2) | Reale | La forma con due argomenti fornisce il primo differenziale di CAMPO1 rispetto a CAMPO2. Restituisce 0 se per il campo specificato non esiste alcun valore precedente. |
| @INDEX | Intero | Restituisce l'indice del record corrente. Gli indici vengono assegnati ai record mano a mano che arrivano al nodo corrente. Al primo record viene assegnato l'indice 1, che viene incrementato di 1 per ciascun record successivo. |

| Funzione | Risultato | Descrizione |
|------------------------|-----------|--|
| @LAST_NON_BLANK(FIELD) | Qualsiasi | Restituisce l'ultimo valore per <i>CAMPO</i> che non era vuoto, come definito in un nodo Input o Tipo a monte. Se i record letti non contengono valori non vuoti per <i>CAMPO</i> , viene restituito \$null\$. Si noti che è possibile definire separatamente per ogni campo valori vuoti, denominati anche valori mancanti definiti dall'utente. |
| @MAX(FIELD) | Number | Restituisce il valore massimo per il campo specificato con l'argomento <i>CAMPO</i> . |
| @MAX(FIELD, EXPR) | Number | Restituisce il valore massimo per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. |
| @MAX(FIELD, EXPR, INT) | Number | Restituisce il valore massimo per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituito il valore massimo dei valori inclusi in tutti i record ricevuti. <i>INT</i> indica il numero massimo di valori da cercare all'indietro. Se si specifica questo terzo argomento, si otterrà un'operazione più precisa. |
| @MIN(FIELD) | Number | Restituisce il valore minimo per il campo specificato con l'argomento <i>CAMPO</i> . |
| @MIN(FIELD, EXPR) | Number | Restituisce il valore minimo per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. |
| @MIN(FIELD, EXPR, INT) | Number | Restituisce il valore minimo per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituito il valore minimo dei valori inclusi in tutti i record ricevuti. <i>INT</i> indica il numero massimo di valori da cercare all'indietro. Se si specifica questo terzo argomento, si otterrà un'operazione più precisa. |

| Funzione | Risultato | Descrizione |
|---------------------------|-----------|---|
| @OFFSET(FIELD, EXPR) | Qualsiasi | Restituisce il valore di <i>CAMPO</i> nel record con una distanza dal record corrente pari al valore di <i>ESPR</i> . Una distanza positiva fa riferimento a un record già passato, mentre una distanza negativa specifica una ricerca in avanti di un record non ancora pervenuto. Per esempio, @OFFSET(Status, 1) restituisce il valore del campo Status nel record precedente mentre @OFFSET(Status, -4) viene eseguita la ricerca in avanti su quattro record nella sequenza, ovvero su record che non sono ancora passati attraverso questo nodo) per ottenere il valore. <i>Si noti che una distanza negativa (ricerca in avanti) deve essere specificata come costante.</i> Per le sole distanze positive, <i>ESPR</i> può inoltre rappresentare un'espressione CLEM arbitraria, che può essere valutata per fornire la distanza in relazione al record corrente. In tal caso, l'utilizzo di una versione a tre argomenti di questa funzione dovrebbe migliorare la performance (vedere la funzione successiva). Se questa espressione restituisce un valore diverso da un numero intero non negativo, verrà generato un errore. Non è quindi consentito il calcolo di distanze per la ricerca in avanti. <i>Nota:</i> Una funzione @OFFSET autoreferenziale non può utilizzare una "ricerca in avanti" letterale. Per esempio, in un nodo Riempimento non è possibile sostituire il valore di field1 utilizzando un'espressione come @OFFSET(field1,-2). |
| @OFFSET(FIELD, EXPR, INT) | Qualsiasi | Esegue la stessa operazione della funzione @OFFSET con l'aggiunta di un terzo argomento, <i>INT</i> , che specifica il numero massimo di valori da cercare all'indietro. In casi in cui la distanza viene calcolata da un'espressione, il terzo argomento dovrebbe migliorare le prestazioni. Per esempio in un'espressione come @OFFSET(Foo, Month, 12) il sistema conserva solo gli ultimi dodici valori di Foo, altrimenti deve archiviare ogni valore. Per i casi in cui il valore di distanza sia una costante, incluse le distanze di "ricerca in avanti" negative, che devono essere costanti, il terzo argomento è inutile ed è necessario usare la versione a due argomenti di questa funzione. Vedere anche la nota relativa alle funzioni autoreferenziali nella versione a due argomenti descritte sopra. |
| @SDEV(FIELD) | Reale | Restituisce la deviazione standard dei valori per il <i>CAMPO</i> o i <i>CAMPI</i> specificati. |
| @SDEV(FIELD, EXPR) | Reale | Restituisce la deviazione standard dei valori per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti dal nodo corrente, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la deviazione standard in tutti i record ricevuti. |

| Funzione | Risultato | Descrizione |
|-------------------------|------------------|--|
| @SDEV(FIELD, EXPR, INT) | <i>Reale</i> | Restituisce la deviazione standard dei valori per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti dal nodo corrente, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la deviazione standard in tutti i record ricevuti. <i>INT</i> indica il numero massimo di valori da cercare all'indietro. Se si specifica questo terzo argomento, si otterrà un'operazione più precisa. |
| @SINCE(EXPR) | <i>Qualsiasi</i> | Restituisce il numero di record passati da quando l'espressione CLEM arbitraria rappresentata da <i>ESPR</i> è stata rilevata come vera. |
| @SINCE(EXPR, INT) | <i>Qualsiasi</i> | L'aggiunta del secondo argomento, <i>INT</i> , consente di specificare il numero massimo di record da cercare all'indietro. Se <i>ESPR</i> non è mai stata vera <i>INT</i> corrisponderà a @INDEX+1. |
| @SINCE0(EXPR) | <i>Qualsiasi</i> | A differenza di @SINCE, considera il record corrente. @SINCE0 restituisce 0 se <i>ESPR</i> risulta vera per il record corrente. |
| @SINCE0(EXPR, INT) | <i>Qualsiasi</i> | L'aggiunta del secondo argomento, <i>INT</i> , consente di specificare il numero massimo di record da cercare all'indietro. |
| @SUM(FIELD) | <i>Number</i> | Restituisce la somma dei valori per l'argomento <i>CAMPO</i> o <i>CAMPI</i> specificato. |
| @SUM(FIELD, EXPR) | <i>Number</i> | Restituisce la somma dei valori per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti dal nodo corrente, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la somma dei valori inclusi in tutti i record ricevuti. |
| @SUM(FIELD, EXPR, INT) | <i>Number</i> | Restituisce la somma dei valori per <i>CAMPO</i> negli ultimi record <i>ESPR</i> ricevuti dal nodo corrente, incluso il record corrente. <i>CAMPO</i> deve essere il nome di un campo numerico. <i>ESPR</i> può essere qualsiasi valutazione di espressione per un numero intero maggiore di 0. Se <i>ESPR</i> non viene specificato o se il valore specificato è superiore al numero di record ricevuti, viene restituita la somma dei valori inclusi in tutti i record ricevuti. <i>INT</i> indica il numero massimo di valori da cercare all'indietro. Se si specifica questo terzo argomento, si otterrà un'operazione più precisa. |
| @THIS(FIELD) | <i>Qualsiasi</i> | Restituisce il valore del campo denominato <i>CAMPO</i> nel record corrente. Viene utilizzata solo nelle espressioni @SINCE. |

Funzioni globali

Le funzioni @MEAN,@SUM, @MIN, @MAX e @SDEV possono essere utilizzate su tutti i record letti fino al record corrente incluso. In alcuni casi, tuttavia, è utile poter confrontare i valori nel record corrente con i valori rilevati nell'intero insieme di dati. Se si utilizza un nodo Globali per generare valori nell'intero insieme di dati, è possibile accedere a tali valori in un'espressione CLEM mediante le funzioni globali.

Ad esempio,

@GLOBAL_MAX(Età)

restituisce il valore più elevato di Age nell'insieme di dati, mentre l'espressione

(Valore - @GLOBAL_MEAN(Valore)) / @GLOBAL_SDEV(Valore)

esprime la differenza tra il valore Value di questo record e la media globale come numero di deviazioni standard. È possibile utilizzare i valori globali solo dopo che sono stati calcolati in un nodo Globali. Tutti i valori globali correnti possono essere annullati facendo clic sul pulsante Cancella valori globali nella scheda Valori globali della finestra di dialogo delle proprietà dello stream.

Nota: non è possibile chiamare le funzioni @ da script. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|---------------------|-----------|---|
| @GLOBAL_MAX(FIELD) | Number | Restituisce il valore massimo per CAMPO nell'intero insieme di dati, generato in precedenza da un nodo Globali. CAMPO deve essere il nome di un campo numerico. Se il valore globale corrispondente non è stato impostato, si verifica un errore. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| @GLOBAL_MIN(FIELD) | Number | Restituisce il valore minimo per CAMPO nell'intero insieme di dati, generato in precedenza da un nodo Globali. CAMPO deve essere il nome di un campo numerico. Se il valore globale corrispondente non è stato impostato, si verifica un errore. |
| @GLOBAL_SDEV(FIELD) | Number | Restituisce la deviazione standard dei valori per CAMPO nell'intero insieme di dati, generato in precedenza da un nodo Globali. CAMPO deve essere il nome di un campo numerico. Se il valore globale corrispondente non è stato impostato, si verifica un errore. |
| @GLOBAL_MEAN(FIELD) | Number | Restituisce la media dei valori per CAMPO nell'intero insieme di dati, generato in precedenza da un nodo Globali. CAMPO deve essere il nome di un campo numerico. Se il valore globale corrispondente non è stato impostato, si verifica un errore. |
| @GLOBAL_SUM(FIELD) | Number | Restituisce la somma dei valori per CAMPO nell'intero insieme di dati, generato in precedenza da un nodo Globali. CAMPO deve essere il nome di un campo numerico. Se il valore globale corrispondente non è stato impostato, si verifica un errore. |

Funzioni per la gestione di valori vuoti e nulli

Mediante il linguaggio CLEM è possibile stabilire che valori specifici in un campo devono essere considerati come valori vuoti o mancanti. Le funzioni illustrate di seguito gestiscono i valori vuoti.

Nota: non è possibile chiamare le funzioni @ da script. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|------------------------|-----------|--|
| @BLANK(FIELD) | Booleana | Restituisce un valore vero per tutti i record contenenti valori vuoti in base alle regole di gestione dei vuoti impostate in un nodo Input o Tipo a monte mediante la scheda Tipi. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| @LAST_NON_BLANK(FIELD) | Qualsiasi | Restituisce l'ultimo valore per CAMPO che non era vuoto, come definito in un nodo Input o Tipo a monte. Se i record letti non contengono valori non vuoti per CAMPO, viene restituito \$null\$. Si noti che è possibile definire separatamente per ogni campo valori vuoti, denominati anche valori mancanti definiti dall'utente. |
| @NULL(FIELD) | Booleana | Restituisce vero se il valore di CAMPO è il valore mancante di sistema \$null\$. Restituisce falso per tutti gli altri valori, inclusi i valori vuoti definiti dall'utente. Per ricercare entrambi, utilizzare @BLANK(FIELD) e @NULL(FIELD). |
| undef | Qualsiasi | Viene utilizzata in genere in CLEM per immettere un valore \$null\$, per esempio per sostituire i valori vuoti con valori nulli nel nodo Riempimento. |

I campi vuoti possono essere “riempiti” con il nodo Riempimento. Sia nei nodi Riempimento che Nuovo campo, solo in modalità multipla, la funzione speciale CLEM@FIELD fa riferimento ai campi correnti esaminati.

Campi speciali

Le funzioni speciali vengono utilizzate per indicare i campi specifici esaminati oppure per generare un elenco di campi come input. In caso di derivazione di più campi, per esempio, è opportuno utilizzare @FIELD per indicare che l'operazione di derivazione deve essere eseguita sui campi selezionati. Se si utilizza l'espressione log(@FIELD), viene derivato un nuovo campo di logaritmo per ogni campo selezionato.

Nota: non è possibile chiamare le funzioni @ da script. [Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27.](#)

| Funzione | Risultato | Descrizione |
|-----------------------------|------------------|---|
| @FIELD | <i>Qualsiasi</i> | Esegue un'operazione su tutti i campi specificati nel contesto dell'espressione. Si noti che questa funzione non può essere chiamata da uno script. Per ulteriori informazioni, vedere l'argomento Espressioni CLEM negli script in il capitolo 3 a pag. 27. |
| @TARGET | <i>Qualsiasi</i> | In caso di utilizzo di un'espressione CLEM in una funzione di analisi definita dall'utente, @TARGET rappresenta il campo obiettivo o il "valore corretto" per la coppia obiettivo/previsto analizzata. Questa funzione viene in genere utilizzata in un nodo Analisi. |
| @PREDICTED | <i>Qualsiasi</i> | In caso di utilizzo di un'espressione CLEM in una funzione di analisi definita dall'utente, @PREDICTED rappresenta il campo previsto per la coppia obiettivo/previsto analizzata. Questa funzione viene in genere utilizzata in un nodo Analisi. |
| @PARTITION_FIELD | <i>Qualsiasi</i> | Sostituisce il nome del campo di partizione corrente. |
| @TRAINING_PARTITION | <i>Qualsiasi</i> | Restituisce il valore della partizione dell'addestramento corrente. Per esempio, per selezionare i record di addestramento tramite un nodo Seleziona, usare l'espressione CLEM: @PARTITION_FIELD = @TRAINING_PARTITION. In tal modo, il nodo Seleziona funzionerà indipendentemente dai valori utilizzati per rappresentare ogni partizione nei dati. |
| @TESTING_PARTITION | <i>Qualsiasi</i> | Restituisce il valore della partizione di test corrente. |
| @VALIDATION_PARTITION | <i>Qualsiasi</i> | Restituisce il valore della partizione di validazione corrente. |
| @FIELDS_BETWEEN(start, end) | <i>Qualsiasi</i> | Restituisce l'elenco di nomi di campo compresi tra i campi iniziale e finale (inclusi) specificati in base all'ordine naturale (cioè di inserimento) dei campi nei dati. Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |

| Funzione | Risultato | Descrizione |
|---------------------------|------------------|---|
| @FIELDS_MATCHING(pattern) | <i>Qualsiasi</i> | Restituisce un elenco di nomi di campo corrispondenti a uno schema specificato. È possibile includere nello schema un punto interrogativo (?) corrispondente esattamente a un carattere; un asterisco (*) corrisponde a zero o più caratteri. Per indicare un punto interrogativo o un asterisco letterale, ovvero non utilizzato come carattere jolly, è possibile utilizzare la barra rovesciata (\). Per ulteriori informazioni, vedere l'argomento Riepilogo di più campi in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |
| @MULTI_RESPONSE_SET | <i>Qualsiasi</i> | Restituisce l'elenco dei campi nell'insieme di risposte multiple denominato. Per ulteriori informazioni, vedere l'argomento Utilizzo di dati a risposta multipla in il capitolo 7 in Manuale dell'utente di IBM SPSS Modeler 14.2. |

Parte II:
Guida alle proprietà

Guida alle proprietà

Panoramica sui riferimenti alle proprietà

È possibile specificare numerose proprietà per nodi, stream, Supernodi e progetti. Alcune proprietà sono comuni a tutti i nodi, per esempio name, annotation e ToolTip, altre invece sono specifiche di alcuni tipi di nodi. Altre proprietà fanno riferimento a operazioni di alto livello degli stream, quali l'inserimento nella cache o il funzionamento dei Supernodi. È possibile accedere alle proprietà tramite l'interfaccia utente standard (per esempio, tramite la finestra di dialogo per la modifica delle opzioni di un nodo) e utilizzarle in molti modi.

- È possibile modificare le proprietà tramite gli script, come illustrato in questa sezione. Per ulteriori informazioni, vedere Sintassi delle proprietà di seguito.
- È possibile utilizzare le proprietà dei nodi nei parametri dei Supernodi. [Per ulteriori informazioni, vedere l'argomento Utilizzo dei parametri di un Supernodo per l'accesso alle proprietà del nodo proprietà in il capitolo 9 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)
- Le proprietà dei nodi possono inoltre essere specificate come parte di un'opzione della riga di comando (mediante il flag -P) all'avvio di IBM® SPSS® Modeler.

Per gli script di SPSS Modeler, le proprietà dei nodi e degli stream vengono spesso denominate **parametri di configurazione**. In questa guida, verranno invece definite come proprietà dei nodi o degli stream.

Per ulteriori informazioni sul linguaggio di script, vedere il capitolo 3.

Sintassi delle proprietà

La struttura della sintassi delle proprietà deve essere la seguente:

```
NAME.TYPE.PROPERTY
```

dove NAME è il nome di un nodo e TYPE è il tipo di nodo (per esempio, `multiplotnode` o `derivenode`). È possibile omettere NAME o TYPE, ma è necessario includere almeno uno dei due. PROPERTY è il nome del parametro del nodo o dello stream a cui si riferisce la propria espressione. Per esempio, la sintassi seguente consente di filtrare il campo *Age* dai dati a valle:

```
set mynode:filternode.include.Age = false
```

Per utilizzare un valore personalizzato per uno qualsiasi dei parametri (NAME, TYPE o PROPERTY), impostare per prima cosa il valore in un'istruzione, per esempio `set derive.new_name = mynewfield`. Da questo punto, è possibile utilizzare il valore `mynewfield` come parametro, facendolo precedere dal simbolo `^`. Per esempio, è possibile impostare il tipo del nodo Nuovo campo citato sopra utilizzando la sintassi seguente:

```
set ^mynewfield.result_type = "Conditional"
```

Tutti i nodi utilizzati in IBM® SPSS® Modeler possono essere specificati nel parametro TYPE della sintassi NAME:TYPE.PROPERTY.

Proprietà strutturate

Le proprietà strutturate vengono utilizzate negli script per semplificare l'analisi ed essenzialmente per due motivi:

- Per strutturare i nomi delle proprietà dei nodi complessi, quali i nodi Tipo, Filtro o Bilanciamento.
- Per rendere disponibile un formato per la specifica di più proprietà contemporaneamente.

Strutturazione per interfacce complesse

Gli script per i nodi con tabelle e altre interfacce complesse quali i nodi Tipo, Filtro e Bilanciamento devono avere una struttura particolare per poter eseguire l'analisi correttamente. A queste proprietà strutturate deve essere assegnato un nome più complesso di quello di un singolo identificatore. Per esempio, all'interno di un nodo Filtro ogni campo disponibile (nel lato a monte) è attivato o disattivato. Per fare riferimento a queste informazioni, il nodo Filtro archivia un'informazione per campo (se ogni campo è vero o falso). Per accedere a tali informazioni e aggiornarle, è possibile utilizzare una singola proprietà denominata **field**, che può avere (o alla quale può essere assegnato) il valore **true** o **false**. Si supponga che un nodo Filtro denominato **mynode** includa un campo (nel lato a monte) denominato *Age*. Per disattivarlo, impostare la proprietà **mynode.include.Age** sul valore **false**, come è illustrato di seguito:

```
set mynode.include.Age = false
```

Strutturazione per l'impostazione di proprietà multiple

Per molti nodi è possibile assegnare più di una proprietà di nodo o di stream contemporaneamente. Questo tipo di operazione è definita **comando di impostazione multipla** o **blocco di impostazioni**. [Per ulteriori informazioni, vedere l'argomento Comando set in il capitolo 4 a pag. 32.](#)

A volte le proprietà strutturate possono essere molto complesse. In questi casi, è possibile utilizzare la barra rovesciata (\) come carattere di continuazione della riga, in modo tale da allineare gli argomenti e migliorarne la leggibilità. Esempio:

```
mynode.sortnode.keys = [{ 'K' Descending} \
                        { 'Age' Ascending}\
                        { 'Na' Descending }]
```

Un altro vantaggio delle proprietà strutturate consiste nella possibilità di impostare numerose proprietà in un nodo prima che questo diventi stabile. Per default, un'impostazione multipla definisce tutte le proprietà del blocco prima di eseguire qualsiasi operazione basata sull'impostazione di una singola proprietà. Per esempio, se si definisce un nodo Testo fisso e si utilizzano due passaggi per impostare le proprietà dei campi, verranno generati degli errori perché il nodo non è coerente fino a quando entrambe le impostazioni non sono valide. La definizione

delle proprietà come un'impostazione multipla elude il problema, perché entrambe le proprietà vengono impostate prima dell'aggiornamento del modello di dati.

Abbreviazioni

Per le proprietà dei nodi, nella sintassi vengono utilizzate abbreviazioni standard il cui apprendimento può risultare utile per la creazione degli script.

| Abbreviazione | Significato |
|---------------|--|
| abs | In valore assoluto |
| len | Lunghezza |
| min | Minimo |
| max | Massimo |
| correl | Correlazione |
| covar | Covarianza |
| num | Numero o numerico |
| pct | Percento o percentuale |
| transp | Trasparenza |
| xval | Validazione incrociata |
| var | Varianza o variabile (nei nodi di input) |

Esempi di proprietà dei nodi e degli stream

IBM® SPSS® Modeler consente di utilizzare le proprietà di nodi e stream in vari modi. Nella maggior parte dei casi vengono utilizzate come parte di uno script, sia di uno **script locale** che consente di automatizzare più stream o più operazioni, sia di uno **script dello stream** che consente di automatizzare i processi all'interno di un singolo stream. È possibile specificare i parametri dei nodi anche utilizzando le proprietà dei nodi all'interno del Supernodo. A livello di base, le proprietà possono inoltre essere utilizzate come opzioni della riga di comando per l'avvio di SPSS Modeler. L'utilizzo dell'argomento -p nella chiamata alla riga di comando consente di modificare un'impostazione dello stream mediante una proprietà dello stream.

| | |
|-----------------------|--|
| s.max_size | Fa riferimento alla proprietà max_size del nodo denominato s. |
| s:samplenode.max_size | Fa riferimento alla proprietà max_size del nodo denominato s, che deve essere un nodo Campione. |
| :samplenode.max_size | Fa riferimento alla proprietà max_size del nodo Campione dello stream corrente (deve essere presente un solo nodo Campione). |
| s:sample.max_size | Fa riferimento alla proprietà max_size del nodo denominato s, che deve essere un nodo Campione. |
| t.direction.Age | Fa riferimento al ruolo del campo Age nel nodo Tipo t. |
| :.max_size | *** NON VALIDO *** È necessario specificare il nome o il tipo del nodo. |

L'esempio s:sample.max_size indica che non è necessario scrivere per esteso i tipi di nodo.

L'esempio `t.direction.Age` indica che anche i nomi di alcune configurazioni possono essere strutturati, se gli attributi di un nodo sono più complessi delle singole configurazioni con valori singoli. Queste configurazioni sono definite proprietà **strutturate** o **complesse**.

Panoramica sulle proprietà dei nodi

Per ogni tipo di nodo è disponibile un insieme specifico di proprietà valide. Questo tipo può essere generale (numero, flag o stringa), nel qual caso le impostazioni relative alla proprietà vengono forzate al tipo corretto. Se questo non è possibile, viene generato un errore. In alternativa, è possibile che il riferimento alla proprietà specifichi l'intervallo di valori validi, per esempio `Discard`, `PairAndDiscard` e `IncludeAsText`, nel qual caso verrà generato un errore se si utilizza un qualsiasi altro valore. Le proprietà flag dovrebbero essere lette o impostate utilizzando i valori `true` e `false`. (Quando si impostano i valori, vengono riconosciute anche le variazioni, incluse `Off`, `OFF`, `off`, `No`, `NO`, `no`, `n`, `N`, `f`, `F`, `false`, `False`, `FALSE` o `0`, ma in alcuni casi potrebbero causare degli errori durante la lettura dei valori delle proprietà. Tutti gli altri valori vengono considerati come vero. L'utilizzo coerente di `true` e `false` consente di evitare confusioni. Nelle tabelle di riferimento di questa guida, le proprietà strutturate vengono indicate come tali nella colonna *Descrizione proprietà* e vengono inoltre forniti i relativi formati di utilizzo.

Proprietà comuni dei nodi

Esistono numerose proprietà comuni a tutti i nodi di IBM® SPSS® Modeler, inclusi i Supernodi.

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|------------------------------|---|--|
| <code>use_custom_name</code> | <i>flag</i> | |
| <code>name</code> | <i>stringa</i> | La proprietà di sola lettura che legge il nome (automatico o personalizzato) per un nodo nell'area di disegno. |
| <code>custom_name</code> | <i>stringa</i> | Specifica un nome personalizzato per il nodo. |
| <code>tooltip</code> | <i>stringa</i> | |
| <code>annotation</code> | <i>stringa</i> | |
| <code>keywords</code> | <i>stringa</i> | Configurazione strutturata che specifica un elenco di parole chiave associate all'oggetto, per esempio, ["Keyword1" "Keyword2"]. |
| <code>cache_enabled</code> | <i>flag</i> | |
| <code>node_type</code> | <code>source_supernode</code> <code>process_supernode</code> <code>terminal_supernode</code> tutti i nomi di nodi, come specificato per gli script | Proprietà di sola lettura utilizzata per fare riferimento a un nodo in base al tipo. Per esempio, invece di fare riferimento a un nodo utilizzando solo il nome, quale <code>real_income</code> , è anche possibile specificarne il tipo, per esempio <code>userinputnode</code> o <code>filternode</code> . |

Le proprietà specifiche dei Supernodi vengono illustrate separatamente, analogamente a tutti gli altri nodi. [Per ulteriori informazioni, vedere l'argomento Proprietà dei Supernodi in il capitolo 22 a pag. 299.](#)

Proprietà stream

Gli script consentono di controllare numerose proprietà degli stream. Per fare riferimento alle proprietà dello stream è necessario utilizzare una variabile speciale dello stream, indicata dal simbolo ^ che precede lo stream:

```
set ^stream.execute_method = Script
```

Esempio

La proprietà nodes viene utilizzata per fare riferimento ai nodi nello stream corrente. Un esempio è costituito dallo script dello stream seguente:

```
var listofnodes
var thenode
set listofnodes = ^stream.nodes

set ^stream.annotation = ^stream.annotation >< "\n\nThis stream is called \"\" >< ^stream.name >
< \"\" and contains/ the following nodes\n"

for thenode in listofnodes
set ^stream.annotation = ^stream.annotation >< "\n\" >< ^thenode.node_type
endfor
```

Nell'esempio riportato sopra, la proprietà nodes viene utilizzata per creare un elenco di tutti i nodi dello stream e per scrivere tale elenco nelle annotazioni dello stream. L'annotazione generata si presenta nel modo seguente:

This stream is called "druglearn" and contains the following nodes

```
derivenode
neuralnetworknode
variablefilenode
typenode
c50node
filternode
```

Nella tabella seguente vengono illustrate le proprietà degli stream.

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|---------------------------|---|-----------------------|
| execute_method | Normal Script | |
| date_format | "GGMMAA" "MMGGAA" "AAMMGG" "AAAAMMGG" "AAAAGGG" GIORNO MESE "GG-MM-AA" "GG-MM-AAAA" "MM-GG-AA" "MM-GG-AAAA" "GG-MES-AA" "GG-MES-AAAA" "AAAA-MM-GG" "GG.MM.AA" "GG.MM.AAAA" "MM.GG.AA" "MM.GG.AAAA" "GG.MES.AA" "GG.MES.AAAA" "GG/MM/AA" "GG/MM/AAAA" "MM/GG/AA" "MM/GG/AAAA" "GG/MES/AA" "GG/MES/AAAA" MES AAAA t T AAAA ss ST AAAA | |
| date_baseline | <i>numero</i> | |
| date_2digit_baseline | <i>numero</i> | |
| time_format | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM." "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | |
| time_rollover | <i>flag</i> | |
| import_datetime_as_string | <i>flag</i> | |
| decimal_places | <i>numero</i> | |

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|----------------------|----------------------------|---|
| decimal_symbol | Default Period Comma | |
| angles_in_radians | <i>flag</i> | |
| use_max_set_size | <i>flag</i> | |
| max_set_size | <i>numero</i> | |
| ruleset_evaluation | Voting FirstHit | |
| refresh_source_nodes | <i>flag</i> | Consente di aggiornare automaticamente i nodi di input all'esecuzione dello stream. |
| script | <i>stringa</i> | |
| annotation | <i>stringa</i> | Esempio: set ^stream.annotation = "something interesting" |
| name | <i>stringa</i> | Esempio: set x = ^stream.name <i>Nota:</i> questa proprietà è di sola lettura. Se si desidera modificare il nome di uno stream, salvarlo con un nome diverso. |
| parameters | | Utilizzare questa proprietà per aggiornare i parametri degli stream dall'interno di uno script locale. Esempio: set ^stream.parameters.height = 23 |
| nodes | | Vedere le informazioni dettagliate riportate di seguito. |
| encoding | SystemDefault "UTF-8" | |

Proprietà progetto

Per gli script dei progetti sono disponibili numerose proprietà,

Esempio

```
load project "C:/clemdata/DrugData.cpj"
set ^project.summary="Initial modeling work on the latest drug data."
set ^project.ordering=NameAddedType
execute_project
```

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|-------------------|--|---|
| summary | <i>stringa</i> | Riepilogo del progetto: in genere, una versione abbreviata dell'annotazione. |
| title | <i>stringa</i> | Titolo del report. |
| author | <i>stringa</i> | Autore del report. |
| structure | Phase Class | Determina come viene organizzato il progetto: per fase di data mining o per tipo di oggetto (classe). |
| include_mode | IncludedItems ExcludedItems AllItems | Determina quali elementi includere nel report del progetto. |
| select_mode | AllItems RecentItems OldItems | Determina quali elementi includere nel report del progetto (per età). |
| recent_item_limit | <i>intero</i> | Utilizzata quando select_mode è RecentItems . |
| old_item_limit | <i>intero</i> | Utilizzata quando select_mode è OldItems . |
| ordering | TypeNameAdded TypeAddedName NameAddedType AddedNameType | Determina l'ordine secondo il quale vengono elencati gli elementi nel report. |

Proprietà dei nodi di input

Proprietà comuni dei nodi di input

Di seguito vengono elencate le proprietà comuni a tutti i nodi di input, corredate da informazioni sui nodi specifici negli argomenti che seguono.

Esempio

```
create variablefilenode
set :variablefilenode.full_filename = "$CLEO_DEMOS/DRUG4n"
set :variablefilenode.use_custom_values.Age = True
set :variablefilenode.direction.Age = Input
set :variablefilenode.type.Age = Range
#storage is read only
set :variablefilenode.check.Age = None
set :variablefilenode.values.Age = [1 100]
```

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|----------------|--|---|
| direction | Input Target Both None Partition Split Frequency RecordID | Proprietà basata su chiavi per i ruoli del campo. Formato di utilizzo: NODE.direction.FIELDNAME <i>Nota:</i> i valori In e Out ora sono dichiarati obsoleti. Nelle versioni future potrebbero non essere più supportati. |
| type | Range Flag Set Typeless Discrete Default | Tipo di campo. L'impostazione di questa proprietà su <i>Default</i> cancella qualsiasi impostazione di proprietà <i>values</i> e se <i>value_mode</i> è impostata su <i>Specify</i> , verrà reimpostata su <i>Read</i> . Se <i>value_mode</i> è già impostata su <i>Pass</i> o <i>Read</i> , non verrà influenzata dall'impostazione <i>type</i> . Formato di utilizzo: NODE.type.FIELDNAME |
| storage | Unknown String Integer Real Time Date Timestamp | Proprietà basata su chiavi in sola lettura per il tipo di archiviazione del campo. Formato di utilizzo: NODE.storage.FIELDNAME |
| check | None Nullify Coerce Discard Warn Abort | Proprietà basata su chiavi per il controllo del tipo di campo e dell'intervallo. Formato di utilizzo: NODE.check.FIELDNAME |

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|--------------------|-------------------------|---|
| values | [valore valore] | Per un campo continuo (intervallo), il primo valore corrisponde al minimo e l'ultimo valore al massimo. Per i campi nominali (insieme), specificare tutti i valori. Nel caso dei campi flag, il primo valore rappresenta <i>falso</i> e l'ultimo valore rappresenta <i>vero</i> . L'impostazione automatica di questa proprietà imposta la proprietà <i>value_mode</i> su <i>Specify</i> . Formato di utilizzo: NODE.values.FIELDNAME |
| value_mode | Read Pass Specify | Determina la modalità di impostazione dei valori per un campo nel passaggio di dati successivo. Formato di utilizzo: NODE.value_mode.FIELDNAME Si noti che non è possibile impostare questa proprietà direttamente su <i>Specify</i> . Per utilizzare valori specifici, impostare la proprietà <i>values</i> . |
| default_value_mode | Read Pass | Specifica il metodo di default per l'impostazione dei valori di tutti i campi. Formato di utilizzo: NODE.default_value_mode Esempio: set mynode.default_value_mode = Pass È possibile sovrascrivere questa impostazione per campi specifici utilizzando la proprietà <i>value_mode</i> . |
| extend_values | flag | Si applica quando <i>value_mode</i> è impostato su <i>Read</i> . Per aggiungere valori appena letti a eventuali valori esistenti per il campo, impostare su <i>T</i> . Per scartare i valori esistenti e sostituirli con i valori appena letti, impostare su <i>F</i> . Formato di utilizzo: NODE.extend_values.FIELDNAME |
| value_labels | stringa | Utilizzata per specificare l'etichetta di un valore. Esempio: set :varfilenode.value_labels.Age = [{3 three}{5 five}] Tenere presente che i valori devono essere specificati prima. |
| enable_missing | flag | Se impostato su <i>T</i> , attiva la registrazione dei valori mancanti per il campo. Formato di utilizzo: NODE.enable_missing.FIELDNAME |
| missing_values | [valore valore...] | Specifica i valori dei dati che indicano dati mancanti. Formato di utilizzo: NODE.missing_values.FIELDNAME |
| null_missing | flag | Se questa proprietà è impostata su <i>T</i> , i valori nulli (valori non definiti, visualizzati come \$null\$ nel software) vengono considerati valori mancanti. Formato di utilizzo: NODE.null_missing.FIELDNAME |
| whitespace_missing | flag | Quando questa proprietà è impostata su <i>T</i> , i valori contenenti solo uno spazio vuoto (spazi, tabulazioni e nuove righe) vengono considerati valori mancanti. Formato di utilizzo: NODE.whitespace_missing.FIELDNAME |
| description | stringa | Utilizzata per specificare una descrizione o etichetta di campo. |

| Nome proprietà | Tipo di dati | Descrizione proprietà |
|-----------------|--------------|--|
| default_include | flag | Proprietà basata su chiavi utilizzata per specificare se il comportamento di default determina il passaggio o il filtro di campi: NODE.default_include Esempio: set mynode:filternode.default_include = false |
| include | flag | Proprietà basata su chiavi utilizzata per determinare se i singoli campi vengono inclusi o filtrati: NODE.include.FIELDNAME. Esempio: set mynode:filternode.include.Age = true |
| new_name | stringa | Esempio: set mynode:filternode.new_name.'Age' = "years" |

Proprietà cognosimportnode



Il nodo di input IBM Cognos BI importa i dati dai database Cognos BI. [Per ulteriori informazioni, vedere l'argomento Nodo di input IBM Cognos BI in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create cognosimportnode
set :cognosimportnode.cognos_connection = {'http://mycogsrv1:9300/p2pd/servlet/dispatch', true, "", "", ""}
set :cognosimportnode.cognos_package_name = '/Public Folders/GOSALES'
set :cognosimportnode.cognos_items = {"[GreatOutdoors].[BRANCH].[BRANCH_CODE]",
"[GreatOutdoors].[BRANCH].[COUNTRY_CODE]"}
```

| proprietà cognosimportnode | Tipo di dati | Descrizione proprietà |
|----------------------------|---|---|
| cognos_connection | { <i>"field"</i> , <i>"field"</i> , ... , <i>"field"</i> } | Una proprietà elenco contenente i dettagli di connessione per il server Cognos. Il formato è: { <i>"URL_server_Cognos"</i> , <i>modalità_accesso</i> , <i>"spaziodeinomi"</i> , <i>"nomeutente"</i> , <i>"password"</i> } dove: <i>URL_server_URL</i> è l'URL del server Cognos contenente la sorgente <i>modalità_accesso</i> indica se viene utilizzato un accesso anonimo e può essere true o false; se impostato su true, i seguenti campi devono essere impostati su "" <i>spaziodeinomi</i> specifica il provider di protezione per l'autenticazione utilizzato per accedere al server. <i>nomeutente</i> e <i>password</i> sono i dati utilizzati per accedere al server Cognos |
| cognos_package_name | stringa | Il percorso e il nome del pacchetto Cognos da cui importare i dati, per esempio: /Public Folders/GOSALES |
| cognos_items | { <i>"field"</i> , <i>"field"</i> , ... , <i>"field"</i> } | Il percorso e il nome Cognos di uno o più oggetti da importare. Il formato di <i>field</i> è [<i>namespace</i>].[<i>query_subject</i>].[<i>query_item</i>] |

Proprietà databasenode



Il nodo Database può essere utilizzato per importare dati da numerosi altri pacchetti che utilizzano ODBC (Open Database Connectivity), tra cui Microsoft SQL Server, DB2, Oracle e altri. [Per ulteriori informazioni, vedere l'argomento Nodo di input Database in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create databasenode
set :databasenode.mode = Table
set :databasenode.query = "SELECT * FROM drug4n"
set :databasenode.datasource = "Drug4n_db"
set :databasenode.username = "spss"
set :databasenode.password = "spss"
var test_e
set test_e = :databasenode.epassword
set :databasenode.tablename = ".Drug4n"
```

| proprietà databasenode | Tipo di dati | Descrizione proprietà |
|------------------------|-------------------------------|---|
| mode | Table Query | Specificare <i>Table</i> per connettersi a una tabella di database tramite i controlli della finestra di dialogo, oppure <i>Query</i> per eseguire una query del database selezionato tramite SQL. |
| datasource | stringa | Nome database (vedere anche la nota riportata sotto). |
| username | stringa | Dettagli connessione database (vedere anche la nota riportata sotto). |
| password | stringa | |
| epassword | stringa | Specifica una password codificata come alternativa all'hardcoding di una password in uno script. Per ulteriori informazioni, vedere l'argomento Generazione di una password codificata in il capitolo 5 a pag. 60. Proprietà di sola lettura durante l'esecuzione. |
| tablename | stringa | Nome della tabella a cui si desidera accedere. |
| strip_spaces | None Left Right Both | Opzioni per scartare gli spazi iniziali e finali nelle stringhe. |
| use_quotes | AsNeeded Always Never | Specifica se si desidera che i nomi delle tabelle e delle colonne vengano racchiusi tra virgolette quando le query vengono inviate al database (per esempio nel caso in cui tali nomi contengano spazi o punteggiatura). |
| query | stringa | Specifica il codice SQL per la query che si desidera effettuare. |

Nota: se il nome del database (nella proprietà `datasource`) contiene degli spazi, anziché le singole proprietà per `datasource`, `username` e `password`, utilizzare un'unica proprietà `datasource` nel seguente formato:

| proprietà <code>databasenode</code> | Tipo di dati | Descrizione proprietà |
|-------------------------------------|----------------|---|
| <code>datasource</code> | <i>stringa</i> | Formato: {database_name,username,password[,true false]} L'ultimo parametro è destinato all'uso con le password crittografate. Se è impostato su <code>true</code> , prima dell'uso la password verrà decrittografata. |

Esempio

```
create databasenode
set :databasenode.mode = Table
set :databasenode.query = "SELECT * FROM drug4n"
set :databasenode.datasource = {"ORA 10gR2", user1, mypsw, true}
var test_e
set test_e = :databasenode.epassword
set :databasenode.tablename = ".Drug4n"
```

Utilizzare questo formato anche se si sta modificando la sorgente dati; tuttavia, se si desidera soltanto modificare il nome utente o la password, è possibile utilizzare le proprietà `username` o `password`.

Proprietà `datacollectionimportnode`



Il nodo IBM® SPSS® Data Collection Importazione dati importa dati di sondaggi basati sul Modello dati di Data Collection utilizzato dai prodotti di ricerca di mercato IBM Corp.. Per utilizzare questo nodo, è necessario che sia installata Data Collection Data Library. [Per ulteriori informazioni, vedere l'argomento Nodo Data Collection in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create datacollectionimportnode
set :datacollectionimportnode.metadata_name="mrQvDsc"
set :datacollectionimportnode.metadata_file="C:/Program Files/IBM/SPSS/DataCollection/DDL/Data/
Quanvert/Museum/museum.pkd"
set :datacollectionimportnode.casedata_name="mrQvDsc"
set :datacollectionimportnode.casedata_source_type=File
set :datacollectionimportnode.casedata_file="C:/Program Files/IBM/SPSS/DataCollection/DDL/Data/
Quanvert/Museum/museum.pkd"
set :datacollectionimportnode.import_system_variables = Common
```

```
set :datacollectionimportnode.import_multi_response = MultipleFlags
```

| proprietà datacollectionimportnode | Tipo di dati | Descrizione proprietà |
|------------------------------------|---|---|
| metadata_name | <i>stringa</i> | Nome dell'MDSC. Il valore speciale DimensionsMDD indica che deve essere utilizzato il documento metadati di Data Collection standard. Gli altri valori possibili sono: mrADODsc mrI2dDsc mrLogDsc mrQdiDrsDsc mrQvDsc mrSampleReportingMDSC mrSavDsc mrSCDsc mrScriptMDSC Il valore speciale none indica che non è presente alcun MDSC. |
| metadata_file | <i>stringa</i> | Nome del file nel quale sono archiviati i metadati. |
| casedata_name | <i>stringa</i> | Nome del CDSC. I valori possibili sono: mrADODsc mrI2dDsc mrLogDsc mrPunchDSC mrQdiDrsDsc mrQvDsc mrRdbDsc2 mrSavDsc mrScDSC mrXmlDsc Il valore speciale none indica che non è presente alcun CDSC. |
| casedata_source_type | Unknown File Folder UDL DSN | Indica il tipo di sorgente del CDSC. |
| casedata_file | <i>stringa</i> | Se casedata_source_type è <i>File</i> , specifica il file contenente i dati del caso. |
| casedata_folder | <i>stringa</i> | Se casedata_source_type è <i>Folder</i> , specifica la cartella contenente i dati del caso. |
| casedata_udl_string | <i>stringa</i> | Se casedata_source_type è <i>UDL</i> , specifica la stringa di connessione OLD-DB della sorgente dati contenente i dati del caso. |
| casedata_dsn_string | <i>stringa</i> | Se casedata_source_type è <i>DSN</i> , specifica la stringa di connessione ODBC della sorgente dati. |

| proprietà datacollectionimportnode | Tipo di dati | Descrizione proprietà |
|------------------------------------|--------------------------|--|
| casedata_project | stringa | Durante la lettura dei dati del caso da un database di Data Collection, è possibile immettere il nome del progetto. Per tutti gli altri tipi di dati del caso, questa impostazione deve essere lasciata vuota. |
| version_import_mode | All Latest Specify | Definisce quante versioni devono essere gestite. |
| specific_version | stringa | Se <i>version_import_mode</i> è <i>Specify</i> , definisce la versione dei dati del caso da importare. |
| use_language | stringa | Definisce se debbano essere utilizzate delle etichette di una lingua specifica. |
| language | stringa | Se <i>use_language</i> è impostata su true (vero), specifica il codice della lingua da utilizzare nell'importazione. Il codice della lingua deve essere uno di quelli disponibili nei dati del caso. |
| use_context | stringa | Definisce se debba essere importato un contesto specifico. I contesti vengono utilizzati per variare la descrizione associata alle risposte. |
| context | stringa | Se <i>use_context</i> è impostato su true (vero), definisce il contesto da importare. Il contesto deve essere uno di quelli disponibili nei dati del caso. |
| use_label_type | stringa | Definisce se debba essere importato un tipo specifico di etichetta. |
| label_type | stringa | Se <i>use_label_type</i> è impostato su true (vero), definisce il tipo di etichetta da importare. Il tipo di etichetta deve essere uno di quelli disponibili nei dati del caso. |
| user_id | stringa | Per i database che richiedono un login esplicito, è possibile fornire un ID utente e una password per accedere alla sorgente dati. |
| password | stringa | |
| import_system_variables | Common None All | Specifica quali variabili di sistema vengono importate. |
| import_codes_variables | flag | |
| import_sourcefile_variables | flag | |
| import_multi_response | MultipleFlags Single | |

Proprietà excelimportnode



Il nodo Importazione da Excel importa dati da qualsiasi versione di Microsoft Excel. Non è richiesta alcuna sorgente dati ODBC. [Per ulteriori informazioni, vedere l'argomento Nodo di input Excel in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
#To use a named range:
create excelimportnode
set :excelimportnode.excel_file_type = Excel2007
set :excelimportnode.full_filename = "C:/drug.xls"
set :excelimportnode.use_named_range = True
set :excelimportnode.named_range = "DRUG"
set :excelimportnode.read_field_names = True
```

```
#To use an explicit range:
create excelimportnode
set :excelimportnode.excel_file_type = Excel2007
set :excelimportnode.full_filename = "C:/drug.xls"
set :excelimportnode.worksheet_mode = Name
set :excelimportnode.worksheet_name = "Drug"
set :excelimportnode.explicit_range_start = A1
set :excelimportnode.explicit_range_end = F300
```

| proprietà excelimportnode | Tipo di dati | Descrizione proprietà |
|---------------------------|--------------------------------|---|
| excel_file_type | Excel2003 Excel2007 | |
| full_filename | <i>stringa</i> | Il nome del file completo compreso il percorso. |
| use_named_range | <i>Booleana</i> | Indica se utilizzare o meno un intervallo denominato. Se true (vero), la proprietà <code>named_range</code> viene utilizzata per specificare l'intervallo da leggere, mentre le altre impostazioni relative al foglio di lavoro e all'intervallo dati vengono ignorate. |
| named_range | <i>stringa</i> | |
| worksheet_mode | Index Name | Specifica se il foglio di lavoro è definito in base all'indice o al nome. |
| worksheet_index | <i>intero</i> | Indice dei fogli di lavoro da leggere che inizia con 0 per il primo foglio di lavoro, 1 per il secondo e così via. |
| worksheet_name | <i>stringa</i> | Nome del foglio di lavoro da leggere. |
| data_range_mode | FirstNonBlank ExplicitRange | Specifica come viene determinato l'intervallo. |
| blank_rows | StopReading ReturnBlankRows | Se <code>data_range_mode</code> è <i>FirstNonBlank</i> , specifica come vanno gestite le righe vuote. |
| explicit_range_start | <i>stringa</i> | Se <code>data_range_mode</code> è <i>ExplicitRange</i> , specifica il punto di partenza dell'intervallo da leggere. |
| explicit_range_end | <i>stringa</i> | |
| read_field_names | <i>Booleana</i> | Specifica se la prima riga dell'intervallo specificato deve essere utilizzata come nome di campo (colonna). |

Proprietà *evimportnode*



Il nodo Enterprise View crea una connessione a un IBM SPSS Collaboration and Deployment Services Repository, consentendo di leggere i dati di Enterprise View in uno stream e di assemblare un modello in uno scenario accessibile da altri utenti tramite il repository. [Per ulteriori informazioni, vedere l'argomento Nodo Enterprise View in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create evimportnode
set :evimportnode.connection = ['Training data', '/Application views/Marketing', 'LATEST', 'Analytic',
    '/Data Providers/Marketing']
set :evimportnode.tablename = "cust1"
```

| proprietà <i>evimportnode</i> | Tipo di dati | Descrizione proprietà |
|-------------------------------|----------------|--|
| connection | <i>elenco</i> | Proprietà strutturata—elenco dei parametri che costituiscono una connessione Enterprise View. Formato di utilizzo: evimportnode.connection = [description,app_view_path, app_view_version_label,environment,DPD_path] |
| tablename | <i>stringa</i> | Il nome di una tabella nella Visualizzazione applicazione. |

Proprietà *fixedfilenode*



Il nodo Testo fisso importa dati da file di testo a campi fissi, ovvero file i cui campi non vengono delimitati ma iniziano nella stessa posizione e hanno una lunghezza fissa. Nel formato a campi fissi vengono in genere archiviati dati di versioni precedenti o generati dalla macchina. [Per ulteriori informazioni, vedere l'argomento Nodo Testo fisso in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create fixedfilenode
set :fixedfilenode.full_filename = "$CLEO_DEMOS/DRUG4n"
set :fixedfilenode.record_len = 32
set :fixedfilenode.skip_header = 1
set :fixedfilenode.fields = [{'Age' 1 3} {'Sex' 5 7} {'BP' 9 10} {'Cholesterol' 12 22} {'Na' 24 25} {'K' 27 27} {'Drug' 29 32}]
set :fixedfilenode.decimal_symbol = Period
set :fixedfilenode.lines_to_scan = 30
```

| proprietà <i>fixedfilenode</i> | Tipo di dati | Descrizione proprietà |
|--------------------------------|---------------|---|
| record_len | <i>numero</i> | Specifica il numero di caratteri in ogni record. |
| line_oriented | <i>flag</i> | Ignora il carattere di nuova riga alla fine di ogni record. |

| proprietà fixedfilenode | Tipo di dati | Descrizione proprietà |
|--------------------------|---|--|
| decimal_symbol | Default Comma Period | Tipo di separatore decimale utilizzato nella sorgente dati. Esempio: set :fixedfilenode.decimal_symbol = Period |
| skip_header | numero | Specifica il numero di righe che si desidera ignorare all'inizio del primo record. Utile per ignorare le intestazioni di colonna. |
| auto_recognize_datetime | flag | Specifica se data e ora vengono identificate automaticamente nei dati di origine. |
| lines_to_scan | numero | Esempio: set :fixedfilenode.lines_to_scan = 50. |
| fields | elenco | Proprietà strutturata. Formato di utilizzo: fixedfilenode.fields = {{field start length} {field start length}} |
| full_filename | stringa | Nome completo del file da leggere, inclusa la directory. |
| strip_spaces | None Left Right Both | Scarta gli spazi iniziali e finali nelle stringhe durante l'importazione. |
| invalid_char_mode | Discard Replace | Rimuove i caratteri non validi (null, 0 o qualsiasi carattere non esistente nella codifica corrente) dall'input dei dati o sostituisce i caratteri non validi con il simbolo a un carattere specificato. |
| invalid_char_replacement | stringa | |
| use_custom_values | flag | Configurazione basata su chiavi nel formato: set :varfilenode.use_custom_values.Age = true |
| custom_storage | Unknown String Integer Real Time Date Timestamp | Configurazione basata su chiavi nel formato: set :varfilenode.custom_storage.'Age' = "Real" |
| custom_date_format | "DDMMYY" "MMDDYY" "YYMMDD" "YYYYMMDD" "YYYYDDD" DAY MONTH "DD-MM-YY" "DD-MM-YYYY" "MM-DD-YY" "MM-DD-YYYY" "DD-MON-YY" "DD-MON-YYYY" "YYYY-MM-DD" "DD.MM.YY" "DD.MM.YYYY" "MM.DD.YY" "MM.DD.YYYY" | Questa proprietà è applicabile solo se è stata specificata un'archiviazione personalizzata. Esempio: set:varfilenode.custom Configurazione basata su chiavi nel formato: set :varfilenode.custom_date_format.'LaunchDate' = "DDMMYY" |

| proprietà fixedfilenode | Tipo di dati | Descrizione proprietà |
|-------------------------|--|--|
| | "DD.MON.YY" "DD.MON.YYYY" "DD/MM/YY" "DD/MM/YYYY" "MM/DD/YY" "MM/DD/YYYY" "DD/MON/YY" "DD/MON/YYYY" MON YYYY q Q YYYY ww WK YYYY | |
| custom_time_format | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM" "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | Questa proprietà è applicabile solo se è stata specificata un'archiviazione personalizzata. Configurazione basata su chiavi nel formato: set :varfilenode.custom_time_format. 'Initialize' = "HHMM" |
| custom_decimal_symbol | <i>campo</i> | Valida solo se è stata specificata un'archiviazione personalizzata. Configurazione basata su chiavi nel formato: set :varfilenode.custom_decimal_symbol.'Revenue' = "Comma" |
| encoding | StreamDefault SystemDefault "UTF-8" | Specifica il metodo di codifica del testo. |

Proprietà sasimportnode



Il nodo File SAS importa dati SAS in IBM® SPSS® Modeler. [Per ulteriori informazioni, vedere l'argomento Nodo di input SAS in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create sasimportnode
set :sasimportnode.format = Windows
set :sasimportnode.full_filename = "C:/data/retail.sas7bdat"
set :sasimportnode.member_name = "Test"
set :sasimportnode.read_formats = False
set :sasimportnode.full_format_filename = "Test"
```

```
set :sasimportnode.import_names = True
```

| proprietà sasimportnode | Tipo di dati | Descrizione proprietà |
|-------------------------|--|--|
| format | Windows UNIX Transport SAS7 SAS8 SAS9 | Formato del file di importazione. |
| full_filename | <i>stringa</i> | Il nome del file completo che è stato specificato e il relativo percorso. |
| member_name | <i>stringa</i> | Specifica il membro da importare dal file di trasporto SAS specificato. |
| read_formats | <i>flag</i> | Legge i formati dei dati (quali etichette di variabili) dal file del formato specificato. |
| full_format_filename | <i>stringa</i> | |
| import_names | NamesAndLabels LabelsasNames | Specifica il metodo per la mappatura di nomi ed etichette di variabili durante l'importazione. |

Proprietà statisticsimportnode



Il nodo File IBM® SPSS® Statistics legge i dati dal formato di file *.sav* utilizzato da SPSS Statistics, nonché da file della cache salvati in IBM® SPSS® Modeler, che utilizzano lo stesso formato. [Per ulteriori informazioni, vedere l'argomento Nodo File Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Le proprietà di questo nodo sono descritte in [Proprietà statisticsimportnode a pag. 295.](#)

Proprietà userinputnode



Il nodo Input utente consente di ottenere in modo semplice dati sintetici creandoli oppure modificando dati esistenti. È utile, per esempio, quando si desidera creare un insieme di dati di test per la modellazione. [Per ulteriori informazioni, vedere l'argomento Nodo input utente in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create userinputnode
set :userinputnode.data.test1 = "2, 4, 8"
set :userinputnode.names = [test1 test2]
set :userinputnode.custom_storage.test1 = Integer
```

```
set :userinputnode.data_mode = "Ordered"
```

| proprietà userinputnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---|---|
| data | | Proprietà basata su chiavi nel formato: set :userinputnode.data.Age = "1 2 3 4" In alternativa, nella stringa è possibile specificare valori di dimensione di passo, alti e bassi separati da virgole. Esempio: set :userinputnode.data.Age = "10, 70, 5" I dati di ogni campo possono includere lunghezze diverse, tuttavia è necessario che siano coerenti con l'archiviazione del campo. È possibile creare un campo non presente impostandone i valori. Inoltre, se si impostano i valori di un campo su una stringa vuota (" "), tale campo verrà rimosso. |
| names | | Configurazione strutturata che imposta o restituisce un elenco di nomi di campi generati dal nodo. Esempio: ['Field1' 'Field2'] |
| custom_storage | Unknown String Integer Real Time Date Timestamp | Configurazione basata su chiavi che imposta o restituisce l'archiviazione per un campo. Esempio: set :userinputnode.custom_storage.'Age' = "Real" |
| data_mode | Combined Ordered | Se è specificato Combined, i record vengono generati per ogni combinazione di valori di insieme e valori minimi e massimi. Il numero di record generati è uguale al prodotto del numero di valori in ogni campo. Se è specificato Ordered, viene preso un solo valore da ogni colonna per ogni record allo scopo di generare una riga di dati. Il numero di record generati è uguale al numero di valori più grande associato a un campo. Tutti i campi con valori di dati inferiori verranno integrati con valori nulli. |
| values | | <i>Questa proprietà è obsoleta, non deve più essere utilizzata ed è stata sostituita dalla proprietà userinputnode.data.</i> |

Proprietà variablefilenode



Il nodo Testo variabile legge dati da file di testo a campi liberi, ovvero file i cui record contengono un numero costante di campi ma un numero variabile di caratteri. Questo nodo può essere utilizzato per file con testo di intestazione a lunghezza fissa e alcuni tipi di annotazioni. [Per ulteriori informazioni, vedere l'argomento Nodo Testo variabile in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```

create variablefilenode
set :variablefilenode.full_filename = "$CLEO_DEMOS/DRUG4n"
set :variablefilenode.read_field_names = True
set :variablefilenode.delimit_other = True
set :variablefilenode.other = ','
set :variablefilenode.quotes_1 = Discard
set :variablefilenode.decimal_symbol = Comma
set :variablefilenode.invalid_char_mode = "Replace"
set :variablefilenode.invalid_char_replacement = "|"
set :variablefilenode.use_custom_values.Age = True
set :variablefilenode.direction.Age = Input
set :variablefilenode.type.Age = Range
set :variablefilenode.values.Age = [1 100]

```

| proprietà variablefilenode | Tipo di dati | Descrizione proprietà |
|----------------------------|----------------------------|--|
| skip_header | <i>numero</i> | Specifica il numero di caratteri che si desidera ignorare all'inizio del primo record. Formato di utilizzo: variablefilenode:skip_header = 3 |
| num_fields_auto | <i>flag</i> | Determina automaticamente il numero di campi in ogni record. I record devono terminare con un carattere di nuova riga. Formato di utilizzo: variablefilenode:num_fields_auto |
| num_fields | <i>numero</i> | Specifica manualmente il numero di campi in ogni record. |
| delimit_space | <i>flag</i> | Specifica il carattere utilizzato per delimitare i delimitatori di campo nel file. |
| delimit_tab | <i>flag</i> | |
| delimit_new_line | <i>flag</i> | |
| delimit_non_printing | <i>flag</i> | |
| delimit_comma | <i>flag</i> | Nei casi in cui la virgola è sia il delimitatore di campo che il separatore decimale degli stream, impostare delimit_other su <i>true</i> (vero) e specificare una virgola come delimitatore utilizzando la proprietà <i>other</i> . |
| delimit_other | <i>flag</i> | Consente di specificare un delimitatore personalizzato utilizzando la proprietà <i>other</i> . |
| other | <i>stringa</i> | Specifica il delimitatore utilizzato quando delimit_other è <i>true</i> (vero). |
| decimal_symbol | Default Comma Period | Specifica il separatore decimale utilizzato nella sorgente dati. |
| multi_blank | <i>flag</i> | Considera più caratteri delimitatori vuoti adiacenti come un delimitatore singolo. |
| read_field_names | <i>flag</i> | Considera la prima riga del file di dati come etichette per la colonna. |

| proprietà variablefilenode | Tipo di dati | Descrizione proprietà |
|----------------------------|---|---|
| strip_spaces | None Left Right Both | Scarta gli spazi iniziali e finali nelle stringhe durante l'importazione. |
| invalid_char_mode | Discard Replace | Rimuove i caratteri non validi (null, 0 o qualsiasi carattere non esistente nella codifica corrente) dall'input dei dati o sostituisce i caratteri non validi con il simbolo a un carattere specificato. |
| invalid_char_replacement | <i>stringa</i> | |
| lines_to_scan | <i>numero</i> | Indica il numero di righe da esaminare per i tipi di dati specificati. |
| auto_recognize_datetime | <i>flag</i> | Specifica se data e ora vengono identificate automaticamente nei dati di origine. |
| quotes_1 | Discard PairAndDiscard IncludeAsText | Specifica il trattamento delle virgolette singole durante l'importazione. |
| quotes_2 | Discard PairAndDiscard IncludeAsText | Specifica il trattamento delle virgolette doppie durante l'importazione. |
| full_filename | <i>stringa</i> | Nome completo del file da leggere, inclusa la directory. |
| use_custom_values | <i>flag</i> | Configurazione basata su chiavi nel formato: set :varfilenode.use_custom_values.Age = true |
| custom_storage | Unknown String Integer Real Time Date Timestamp | Configurazione basata su chiavi nel formato: set :varfilenode.custom_storage.'Age' = "Real" |
| custom_date_format | "DDMMYY" "MMDDYY" "YYMMDD" "YYYYMMDD" "YYYYDDD" DAY MONTH "DD-MM-YY" "DD-MM-YYYY" "MM-DD-YY" "MM-DD-YYYY" "DD-MON-YY" "DD-MON-YYYY" "YYYY-MM-DD" "DD.MM.YY" "DD.MM.YYYY" "MM.DD.YY" "MM.DD.YYYY" "DD.MON.YY" "DD.MON.YYYY" "DD/MM/YY" "DD/MM/YYYY" "MM/DD/YY" | Valida solo se è stata specificata un'archiviazione personalizzata. Esempio: set:varfilenode.custom Configurazione basata su chiavi nel formato: set :varfilenode.custom_date_format. 'LaunchDate' = "DDMMYY" |

| proprietà variablefilenode | Tipo di dati | Descrizione proprietà |
|----------------------------|--|--|
| | "MM/DD/YYYY" "DD/MON/YY" "DD/MON/YYYY" MON YYYY q Q YYYY ww WK YYYY | |
| custom_time_format | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM" "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | Valida solo se è stata specificata un'archiviazione personalizzata. Configurazione basata su chiavi nel formato: set :varfilenode.custom_time_format. 'Initialize' = "HHMM" |
| custom_decimal_symbol | <i>campo</i> | Valida solo se è stata specificata un'archiviazione personalizzata. Configurazione basata su chiavi nel formato: set :varfilenode.custom_decimal_symbol.'Revenue' = "Comma" |
| encoding | StreamDefault SystemDefault "UTF-8" | Specifica il metodo di codifica del testo. |

Proprietà xmlimportnode



Il nodo di input XML importa i dati in formato XML nello stream. È possibile importare un singolo file o tutti i file in una directory. Se lo si desidera, è possibile specificare un file di schema dal quale leggere la struttura XML. [Per ulteriori informazioni, vedere l'argomento XML Source Node in il capitolo 2 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create xmlimportnode
set :xmlimportnode.full_filename = "c:\import\ebooks.xml"
set :xmlimportnode.records = "/author/name"
```

| proprietà xmlimportnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---------------------|---|
| read | single directory | Legge un singolo file di dati (default), oppure tutti i file XML in una directory. |
| recurse | <i>flag</i> | Specifica se leggere anche i file XML da tutte le sottodirectory della directory specificata. |

| proprietà xmlimportnode | Tipo di dati | Descrizione proprietà |
|--------------------------------|---------------------|--|
| full_filename | <i>stringa</i> | (obbligatorio) Percorso e nome file completi del file XML da importare (se read = single). |
| directory_name | <i>stringa</i> | (obbligatorio) Percorso completo e nome della directory dalla quale importare i file XML (se read = directory). |
| full_schema_filename | <i>stringa</i> | Percorso e nome file completi del file XSD o DTD dal quale leggere la struttura XML. Se si omette questo parametro, la struttura viene letta dal file di input XML. |
| records | <i>stringa</i> | Espressione XPath (ad esempio, /autore/nome) che indica i limiti del record. Ogni volta che si incontra questo elemento nel file di input, viene creato un nuovo record. |
| mode | read specify | Leggere tutti i dati (default), oppure specificare gli elementi da leggere. |
| fields | | Elenco di voci (elementi e attributi) da importare. Ogni voce dell'elenco è un'espressione XPath. |

Proprietà dei nodi Operazioni su record

Proprietà appendnode



Il nodo Accodamento concatena insiemi di record. Può essere utilizzato per combinare insiemi di dati con strutture simili ma dati diversi. [Per ulteriori informazioni, vedere l'argomento Nodo Accodamento in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create appendnode
set :appendnode.match_by = Name
set :appendnode.match_case = True
set :appendnode.include_fields_from = All
set :appendnode.create_tag_field = True
set :appendnode.tag_field_name = "Append_Flag"
```

| proprietà appendnode | Tipo di dati | Descrizione proprietà |
|----------------------|------------------|--|
| match_by | Position Name | È possibile accodare insiemi di dati in base alla posizione dei campi nella sorgente dati principale o al nome dei campi negli insiemi di dati di input. |
| match_case | <i>flag</i> | Attiva la distinzione tra maiuscole e minuscole quando si esegue la corrispondenza tra i nomi dei campi. |
| include_fields_from | Main All | |
| create_tag_field | <i>flag</i> | |
| tag_field_name | <i>stringa</i> | |

Proprietà aggregatenode



Il nodo Aggregazione sostituisce una sequenza di record di input con record di output aggregati di riepilogo. [Per ulteriori informazioni, vedere l'argomento Nodo Aggregazione in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create aggregatenode
connect :databasenode to :aggregatenode
set :aggregatenode.contiguous = True
set :aggregatenode.keys = ['Drug']
```

```

set :aggregatenode.aggregates.Age = [Sum Mean]
set :aggregatenode.inc_record_count = True
set :aggregatenode.count_field = "index"
set :aggregatenode.extension = "Aggregated_"
set :aggregatenode.add_as = Prefix

```

| proprietà aggregatenode | Tipo di dati | Descrizione proprietà |
|-------------------------|------------------------|--|
| keys | [campo campo... campo] | Elenca i campi che possono essere utilizzati come chiavi per l'aggregazione. Per esempio, se Sex e Region sono i campi chiave, ogni combinazione univoca di M e F con aree N e S (quattro combinazioni univoche) avranno un record aggregato. |
| contiguous | flag | Selezionare questa opzione se si sa che tutti i record con gli stessi valori chiave sono raggruppati insieme nell'input (per esempio, se l'input è ordinato in base ai campi chiave). In questo modo si migliorano le prestazioni. |
| aggregates | | Proprietà strutturata che elenca i campi numerici i cui valori verranno aggregati e le modalità di aggregazione selezionate. Esempio: set :aggregatenode. aggregates.Age = [Sum Mean Min Max SDev], dove i metodi di aggregazione desiderati sono inclusi nell'elenco. |
| extension | stringa | Specifica un prefisso o suffisso per campi aggregati duplicati (vedere l'esempio seguente). |
| add_as | Suffix Prefix | |
| inc_record_count | flag | Crea un campo aggiuntivo che specifica quanti record di input sono stati aggregati per formare ogni record aggregato. |
| count_field | stringa | Specifica il nome del campo conteggio record. |

Proprietà balancenode



Il nodo Bilanciamento corregge sbilanciamenti in un insieme di dati in modo che soddisfi una determinata condizione. La direttiva di bilanciamento regola la proporzione di record in cui una condizione è vera in base al fattore specificato. [Per ulteriori informazioni, vedere l'argomento Nodo Bilanciamento in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```

create balancenode
set :balancenode.training_data_only = true
set :balancenode.directives = \
  [{1.3 "Age > 60"}{1.5 "Na > 0.5"}]

```

| proprietà balancenode | Tipo di dati | Descrizione proprietà |
|-----------------------|--------------|---|
| directives | | Proprietà strutturata per il bilanciamento della proporzione dei valori del campo in base al numero specificato (vedere l'esempio seguente). |
| training_data_only | <i>flag</i> | Specifica che devono essere bilanciati solo i dati di addestramento. Se nello stream non è presente alcun campo di partizione, tale opzione viene ignorata. |

Esempio

```
create balancenode
set :balancenode.directives = \
  [{1.3 "Age > 60"}{1.5 "Na > 0.5"}]
```

La proprietà di questo nodo utilizza il formato:

```
[{ numero stringa } \ { numero stringa } \ ... { numero stringa }].
```

Nota: se nell'espressione sono presenti stringhe che utilizzano le virgolette doppie, tali stringhe devono essere precedute dal carattere di escape "\". Il carattere "\" è anche il carattere di continuazione della riga, che consente di allineare gli argomenti per migliorarne la leggibilità.

Proprietà distinctnode

Il nodo Elimina duplicati rimuove record duplicati passando il primo record distinto nello stream di dati oppure scartando il primo record e passando nello stream tutti i duplicati. [Per ulteriori informazioni, vedere l'argomento Nodo Elimina duplicati in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create distinctnode
set :distinctnode.mode = Include
set :distinctnode.fields = ['Age' 'Sex']
set :distinctnode.keys_pre_sorted = True
```

| proprietà distinctnode | Tipo di dati | Descrizione proprietà |
|------------------------|------------------------------|---|
| mode | Include Discard | È possibile includere il primo record distinto nello stream di dati oppure scartare il primo record distinto e passare invece tutti i record duplicati allo stream di dati. |
| fields | [<i>campo campo campo</i>] | Elenca i campi utilizzati per stabilire se i record sono identici. |
| low_distinct_key_count | <i>flag</i> | Indica che i record e/o i valori univoci dei campi chiave sono in numero ridotto. |
| keys_pre_sorted | <i>flag</i> | Specifica che tutti i record con gli stessi valori chiave sono raggruppati insieme nell'input. |
| | | |

Proprietà mergenode



Il nodo Unione prende più record di input e crea un singolo record di output contenente tutti o alcuni campi di input. È utile per unire dati da sorgenti diverse, per esempio dati interni sui clienti e dati demografici acquistati. [Per ulteriori informazioni, vedere l'argomento Nodo Unione in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create mergenode
connect customerdata to :mergenode
connect salesdata to :mergenode
set :mergenode.method = Keys
set :mergenode.key_fields = ['id']
set :mergenode.common_keys = true
set :mergenode.join = PartialOuter
set :mergenode.outer_join_tag.2 = true
set :mergenode.outer_join_tag.4 = true
set :mergenode.single_large_input = true
set :mergenode.single_large_input_tag = '2'
set :mergenode.use_existing_sort_keys = true
set :mergenode.existing_sort_keys = [{'id' Ascending}]
```

| proprietà mergenode | Tipo di dati | Descrizione proprietà |
|---------------------|------------------------------|--|
| method | Order Keys | Specifica se i record vengono uniti nell'ordine secondo cui sono elencati nei file di dati o se verranno utilizzati uno o più campi chiave per unire i record con lo stesso valore nei campi chiave. |
| key_fields | [<i>campo campo campo</i>] | |
| common_keys | <i>flag</i> | |

| proprietà mergenode | Tipo di dati | Descrizione proprietà |
|------------------------|--|---|
| join | Inner FullOuter PartialOuter Anti | Esempio: set :merge.join = FullOuter |
| outer_join_tag.n | flag | In questa proprietà, <i>n</i> è il nome del tag come viene visualizzato nella finestra di dialogo Seleziona insieme di dati. Si noti che è possibile specificare più nomi di tag, in quanto qualsiasi numero di insiemi di dati può contribuire ai record incompleti. |
| single_large_input | flag | Specifica se verrà utilizzata l'ottimizzazione per avere un input relativamente grande rispetto agli altri input. |
| single_large_input_tag | stringa | Specifica il nome del tag come viene visualizzato nella finestra di dialogo Seleziona insieme di dati grande. Si noti che l'utilizzo di questa proprietà differisce leggermente rispetto alla proprietà <code>outer_join_tag</code> (flag vs. stringa) perché è possibile specificare solo un insieme di dati di input. |
| use_existing_sort_keys | flag | Specifica se gli input sono già ordinati in base a uno o più campi chiave. |
| existing_sort_keys | [{stringa Ascending} \ {stringa Descending}] | Specifica i campi già ordinati e la direzione nella quale sono ordinati. |

Proprietà rfmaggatenode



Il nodo Aggregazione RFM (Recency, Frequency, Monetary, Passato recente, Frequenza, Monetario) consente di prendere in considerazione i dati storici delle transazioni dei clienti, eliminare i dati non utilizzati e combinare tutti i dati delle transazioni rimanenti in un'unica riga che indica quanto tempo è trascorso dall'ultima transazione, il numero di transazioni effettuate e il valore monetario totale delle transazioni. [Per ulteriori informazioni, vedere l'argomento Nodo Aggregazione RFM in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create rfmaggatenode
connect :fillernode to :rfmaggatenode
set :rfmaggatenode.relative_to = Fixed
set :rfmaggatenode.reference_date = "2007-10-12"
set :rfmaggatenode.id_field = "CardID"
set :rfmaggatenode.date_field = "Date"
set :rfmaggatenode.value_field = "Amount"
set :rfmaggatenode.only_recent_transactions = True
set :rfmaggatenode.transaction_date_after = "2000-10-01"
```

| proprietà rfmaggregatenode | Tipo di dati | Descrizione proprietà |
|----------------------------|----------------------------------|--|
| relative_to | Fixed Today | Specifica la data a partire da cui viene calcolato il passato recente delle transazioni. |
| reference_date | <i>date</i> | Disponibile solo se viene scelto Fixed in <i>relative_to</i> . |
| contiguous | <i>flag</i> | Se i dati sono preordinati in modo che tutti i record con lo stesso ID appaiano insieme nello stream di dati, selezionare questa opzione per accelerare l'elaborazione. |
| id_field | <i>campo</i> | Specifica il campo da utilizzare per identificare il cliente e le relative transazioni. |
| date_field | <i>campo</i> | Specifica il campo data da utilizzare per calcolare il passato recente. |
| value_field | <i>campo</i> | Specifica il campo da utilizzare per calcolare il valore monetario. |
| extension | <i>stringa</i> | Specifica un prefisso o suffisso per campi aggregati duplicati. |
| add_as | Suffix Prefix | Specifica se <i>extension</i> viene aggiunta come suffisso o prefisso. |
| discard_low_value_records | <i>flag</i> | Attiva l'utilizzo dell'impostazione <i>discard_records_below</i> . |
| discard_records_below | <i>numero</i> | Specifica un valore minimo al di sotto del quale non vengono utilizzati i dettagli delle transazioni nel calcolo dei totali RFM. Le unità di valore sono relative al campo <i>value</i> selezionato. |
| only_recent_transactions | <i>flag</i> | Attiva l'uso dell'impostazione <i>specify_transaction_date</i> o <i>transaction_within_last settings</i> . |
| specify_transaction_date | <i>flag</i> | |
| transaction_date_after | <i>data</i> | Disponibile solo se è selezionata <i>specify_transaction_date</i> . Specifica la data della transazione dopo la quale i record saranno inclusi nell'analisi. |
| transaction_within_last | <i>numero</i> | Disponibile solo se è selezionata <i>transaction_within_last</i> . Specifica il numero e il tipo di periodi (giorni, settimane, mesi o anni) dalla data di Calcola passato recente relativo a dopo la quale i record saranno inclusi nell'analisi. |
| transaction_scale | Days Weeks Months Years | Disponibile solo se è selezionata <i>transaction_within_last</i> . Specifica il numero e il tipo di periodi (giorni, settimane, mesi o anni) dalla data di Calcola passato recente relativo a dopo la quale i record saranno inclusi nell'analisi. |
| save_r2 | <i>flag</i> | Visualizza la data della seconda transazione più recente per ogni cliente. |
| save_r3 | <i>flag</i> | Disponibile solo se è selezionata <i>save_r2</i> . Visualizza la data della terza transazione più recente per ogni cliente. |

Proprietà del nodo Campione



Il nodo Campione seleziona un sottoinsieme di record. Sono supportati vari tipi di campioni, inclusi campioni stratificati, raggruppati e non casuali (strutturati). Il campionamento può essere utile per migliorare le prestazioni e per selezionare gruppi di record correlati o transazioni per un'analisi. [Per ulteriori informazioni, vedere l'argomento Nodo Campione in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
/* Create two Sample nodes to extract
different samples from the same data */

create variablefilenode
set :variablefilenode.full_filename = "$CLEO_DEMOS/DRUG1n"

set node = create samplenode at 300 100
rename ^node as 'First 500'
connect :variablefilenode to 'First 500'
set 'First 500':samplenode.method = Simple
set 'First 500':samplenode.mode = Include
set 'First 500':samplenode.sample_type = First
set 'First 500':samplenode.first_n = 500

set node = create samplenode at 300 200
rename ^node as 'Custom Strata'
connect :variablefilenode to 'Custom Strata'
set 'Custom Strata':samplenode.method = Complex
set 'Custom Strata':samplenode.stratify_by = ['Sex' 'Cholesterol']
set 'Custom Strata':samplenode.sample_units = Proportions
set 'Custom Strata':samplenode.sample_size_proportions = Custom
set 'Custom Strata':samplenode.sizes_proportions= \
  [{"M" "High" "Default"}{"M" "Normal" "Default"} \
  {"F" "High" "0.3"}{"F" "Normal" "0.3"}]
```

| proprietà samplenode | Tipo di dati | Descrizione proprietà |
|----------------------|------------------------------|---|
| method | Simple Complex | |
| mode | Include Discard | Include o scarta i record che soddisfano la condizione specificata. |
| sample_type | First OneInN RandomPct | Specifica il metodo di campionamento. Esempio: set :samplenode.sample_type = First set :samplenode.first_n = 100 |
| first_n | <i>intero</i> | I record fino al punto di interruzione specificato verranno inclusi o scartati. |
| one_in_n | <i>numero</i> | Include o scarta ogni <i>n</i> record. |
| rand_pct | <i>numero</i> | Specifica la percentuale di record da includere o scartare. |
| use_max_size | <i>flag</i> | Attiva l'utilizzo dell'impostazione maximum_size. |

| proprietà samplenode | Tipo di dati | Descrizione proprietà |
|-----------------------------|--|--|
| maximum_size | <i>intero</i> | Specifica la dimensione massima del campione da includere nello stream di dati o da scartare. Questa opzione è ridondante e risulta pertanto disattivata se vengono specificati First e Include. |
| set_random_seed | <i>flag</i> | Attiva l'utilizzo dell'impostazione del seme casuale. |
| random_seed | <i>intero</i> | Specifica il valore utilizzato come seme casuale. |
| complex_sample_type | Random Systematic | |
| sample_units | Proportions Counts | |
| sample_size_proportions | Fixed Custom Variable | |
| sample_size_counts | Fixed Custom Variable | |
| fixed_proportions | <i>numero</i> | |
| fixed_counts | <i>intero</i> | |
| variable_proportions | <i>campo</i> | |
| variable_counts | <i>campo</i> | |
| use_min_stratum_size | <i>flag</i> | |
| minimum_stratum_size | <i>intero</i> | Questa opzione è valida solo quando con Sample units=Proportions viene acquisito un campione Complesso. |
| use_max_stratum_size | <i>flag</i> | |
| maximum_stratum_size | <i>intero</i> | Questa opzione è valida solo quando con Sample units=Proportions viene acquisito un campione Complesso. |
| clusters | <i>campo</i> | |
| stratify_by | <i>[campo1 ... campoN]</i> | |
| specify_input_weight | <i>flag</i> | |
| input_weight | <i>campo</i> | |
| new_output_weight | <i>stringa</i> | |
| sizes_proportions | <i>[{stringvalore di stringa} {stringvalore di stringa} ...]</i> | Se sample_units=proportions e sample_size_proportions=Custom, specifica un valore per ogni possibile combinazione di valori di campi di stratificazione. |
| default_proportion | <i>numero</i> | |
| sizes_counts | <i>[{stringvalore di stringa} {stringvalore di stringa} ...]</i> | Specifica un valore per ogni possibile combinazione di valori di campi di stratificazione. L'utilizzo è simile a quello della proprietà sizes_proportions, con la differenza che viene specificato un numero intero anziché una proporzione. |
| default_count | <i>numero</i> | |

Proprietà selectnode



Il nodo Seleziona consente di selezionare o scartare un sottoinsieme di record dallo stream dei dati basato su una condizione specifica. Per esempio, è possibile selezionare i record relativi a una determinata area vendite. [Per ulteriori informazioni, vedere l'argomento Nodo Seleziona in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create selectnode
set :selectnode.mode = Include
set :selectnode.condition = "Age < 18"
```

| proprietà selectnode | Tipo di dati | Descrizione proprietà |
|----------------------|--------------------|--|
| mode | Include Discard | Indica se includere o scartare i record selezionati. |
| condition | <i>stringa</i> | Condizione per includere o scartare i record. |

Proprietà sortnode



Il nodo Ordina ordina record in ordine crescente o decrescente in base ai valori di uno o più campi. [Per ulteriori informazioni, vedere l'argomento Nodo Ordina in il capitolo 3 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create sortnode
set :sortnode.keys = [{"Age" Ascending} {"Sex" Descending}]
set :sortnode.default_ascending = False
set :sortnode.use_existing_keys = True
set :sortnode.existing_keys = [{"Age" Ascending}]
```

| proprietà sortnode | Tipo di dati | Descrizione proprietà |
|--------------------|--|---|
| keys | [{ <i>stringa</i> Ascending} \ { <i>stringa</i> Descending}] | Specifica i campi in base a cui si desidera ordinare (vedere l'esempio seguente). Se non viene specificata una direzione, viene utilizzata quella di default. |
| default_ascending | <i>flag</i> | Specifica l'ordinamento di default. |
| use_existing_keys | <i>flag</i> | Specifica se l'ordinamento è ottimizzato utilizzando l'ordinamento precedente per i campi già ordinati. |
| existing_keys | | Specifica i campi già ordinati e la direzione nella quale sono ordinati. Utilizza lo stesso formato della proprietà keys. |

Proprietà dei nodi Operazioni su campi

Proprietà anonymizenode



Il nodo Anonimizza consente di mascherare i nomi o i valori dei campi, quando si utilizzano dati da includere in un modello a valle del nodo, permettendo di nascondere i dati originali. Questa funzionalità può essere utile se si desidera consentire ad altri utenti di creare modelli utilizzando dati riservati, quali nomi di clienti o altri dettagli. [Per ulteriori informazioni, vedere l'argomento Nodo Anonimizza in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create anonymizenode
set: anonymizenode.enable_anonymize = age
set: anonymizenode.use_prefix = true
set: anonymizenode.prefix = "myprefix"
set: anonymizenode.transformation = Random
set: anonymizenode.set_random_seed = true
set: anonymizenode.random_seed = "123"
```

| proprietà anonymizenode | Tipo di dati | Descrizione proprietà |
|-------------------------|-----------------|--|
| enable_anonymize | <i>flag</i> | Quando è impostata su V, attiva l'anonimizzazione dei valori dei campi (equivale alla selezione di Sì per quel campo nella colonna Anonimizza valori). |
| use_prefix | <i>flag</i> | Quando è impostata su V, viene utilizzato un prefisso personalizzato se ne è stato definito uno. È valida per i campi che saranno anonimizzati con il metodo hash ed equivale alla selezione del pulsante di scelta Personalizzato nella finestra di dialogo Sostituisci valori di quel campo. |
| prefix | <i>stringa</i> | Equivale alla digitazione di un prefisso nella casella di testo della finestra di dialogo Sostituisci valori. Se non sono stati specificati altri valori, il prefisso di default è il valore di default. |
| transformation | Random Fixed | Determina se i parametri di trasformazione di un campo anonimizzato con il metodo Trasformazioni saranno casuali o fissi. |
| set_random_seed | <i>flag</i> | Quando è impostata su T, sarà utilizzato il valore seme specificato (se la proprietà transformation è impostata su Random). |
| random_seed | <i>intero</i> | Quando set_random_seed è impostata su T, questo è il seme per il numero casuale. |
| scale | <i>numero</i> | Quando transformation è impostata su Fixed, questo è il valore utilizzato per la scala. Il valore di scala massimo in genere è 10, ma può essere ridotto per evitare l'overflow. |
| translate | <i>numero</i> | Quando transformation è impostata su Fixed, questo è il valore utilizzato per "translate". Il valore di translate massimo in genere è 1000, ma può essere ridotto per evitare l'overflow. |

Proprietà autodatapreinode



Il nodo Preparazione automatica dati (ADP) può analizzare i dati e individuare le correzioni, escludere i campi problematici o probabilmente inutili e derivare all'occorrenza nuovi attributi, migliorando le performance grazie allo screening intelligente e alle tecniche di campionamento. Il nodo si può utilizzare in modo completamente automatico, permettendogli di scegliere e di applicare le correzioni, oppure visualizzando in anteprima le modifiche prima dell'applicazione e accettandole, respingendole o modificandole a seconda dei casi. [Per ulteriori informazioni, vedere l'argomento Preparazione automatica dati in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create autodatapreinode
set: autodatapreinode.objective = Balanced
set: autodatapreinode.excluded_fields = Filter
set: autodatapreinode.prepare_dates_and_times = true
set: autodatapreinode.compute_time_until_date = true
set: autodatapreinode.reference_date = Today
set: autodatapreinode.units_for_date_durations = Automatic
```

| proprietà autodatapreinode | Tipo di dati | Descrizione proprietà |
|----------------------------|---|---|
| objective | Balanced Speed Accuracy Custom | |
| custom_fields | <i>flag</i> | Se vera, consente di specificare i campi obiettivo, di input e di altro tipo per il nodo corrente. Se falsa, vengono utilizzate le impostazioni correnti di un nodo Tipo a monte. |
| target | <i>campo</i> | Specifica un singolo campo obiettivo. |
| inputs | [<i>campo1 ... campoN</i>] | I campi di input o predittore utilizzati dal modello. |
| use_frequency | <i>flag</i> | |
| frequency_field | <i>campo</i> | |
| use_weight | <i>flag</i> | |
| weight_field | <i>campo</i> | |
| excluded_fields | Filter None | |
| if_fields_do_not_match | StopExecution ClearAnalysis | |
| prepare_dates_and_times | <i>flag</i> | Controllo dell'accesso a tutti i campi data e ora |
| compute_time_until_date | <i>flag</i> | |
| reference_date | Today Fixed | |
| fixed_date | <i>date</i> | |
| units_for_date_durations | Automatic Fixed | |

| proprietà autodatapreinode | Tipo di dati | Descrizione proprietà |
|--|-----------------------------|------------------------------|
| fixed_date_units | Years Months Days | |
| compute_time_until_time | <i>flag</i> | |
| reference_time | CurrentTime Fixed | |
| fixed_time | <i>time</i> | |
| units_for_time_durations | Automatic Fixed | |
| fixed_date_units | Hours Minutes Seconds | |
| extract_year_from_date | <i>flag</i> | |
| extract_month_from_date | <i>flag</i> | |
| extract_day_from_date | <i>flag</i> | |
| extract_hour_from_time | <i>flag</i> | |
| extract_minute_from_time | <i>flag</i> | |
| extract_second_from_time | <i>flag</i> | |
| exclude_low_quality_inputs | <i>flag</i> | |
| exclude_too_many_missing | <i>flag</i> | |
| maximum_percentage_missing | <i>numero</i> | |
| exclude_too_many_categories | <i>flag</i> | |
| maximum_number_categories | <i>numero</i> | |
| exclude_if_large_category | <i>flag</i> | |
| maximum_percentage_category | <i>numero</i> | |
| prepare_inputs_and_target | <i>flag</i> | |
| adjust_type_inputs | <i>flag</i> | |
| adjust_type_target | <i>flag</i> | |
| reorder_nominal_inputs | <i>flag</i> | |
| reorder_nominal_target | <i>flag</i> | |
| replace_outliers_inputs | <i>flag</i> | |
| replace_outliers_target | <i>flag</i> | |
| replace_missing_continuous_in- puts | <i>flag</i> | |
| replace_missing_continuous_tar- get | <i>flag</i> | |
| replace_missing_nominal_inputs | <i>flag</i> | |
| replace_missing_nominal_target | <i>flag</i> | |
| replace_missing_ordinal_inputs | <i>flag</i> | |
| replace_missing_ordinal_target | <i>flag</i> | |
| maximum_values_for_ordinal | <i>numero</i> | |
| minimum_values_for_continuous | <i>numero</i> | |
| outlier_cutoff_value | <i>numero</i> | |
| outlier_method | Replace Delete | |
| rescale_continuous_inputs | <i>flag</i> | |

| proprietà autodatapreinode | Tipo di dati | Descrizione proprietà |
|-----------------------------------|---------------------|------------------------------|
| rescaling_method | MinMax ZScore | |
| min_max_minimum | <i>numero</i> | |
| min_max_maximum | <i>numero</i> | |
| z_score_final_mean | <i>numero</i> | |
| z_score_final_sd | <i>numero</i> | |
| rescale_continuous_target | <i>flag</i> | |
| target_final_mean | <i>numero</i> | |
| target_final_sd | <i>numero</i> | |
| transform_select_input_fields | <i>flag</i> | |
| maximize_association_with_target | <i>flag</i> | |
| p_value_for_merging | <i>numero</i> | |
| merge_ordinal_features | <i>flag</i> | |
| merge_nominal_features | <i>flag</i> | |
| minimum_cases_in_category | <i>numero</i> | |
| bin_continuous_fields | <i>flag</i> | |
| p_value_for_binning | <i>numero</i> | |
| perform_feature_selection | <i>flag</i> | |
| p_value_for_selection | <i>numero</i> | |
| perform_feature_construction | <i>flag</i> | |
| transformed_target_name_extension | <i>stringa</i> | |
| transformed_inputs_name_extension | <i>stringa</i> | |
| constructed_features_root_name | <i>stringa</i> | |
| years_duration_name_extension | <i>stringa</i> | |
| months_duration_name_extension | <i>stringa</i> | |
| days_duration_name_extension | <i>stringa</i> | |
| hours_duration_name_extension | <i>stringa</i> | |
| minutes_duration_name_extension | <i>stringa</i> | |
| seconds_duration_name_extension | <i>stringa</i> | |
| year_cyclical_name_extension | <i>stringa</i> | |
| month_cyclical_name_extension | <i>stringa</i> | |
| day_cyclical_name_extension | <i>stringa</i> | |
| hour_cyclical_name_extension | <i>stringa</i> | |
| minute_cyclical_name_extension | <i>stringa</i> | |
| second_cyclical_name_extension | <i>stringa</i> | |

Proprietà binningnode



Il nodo Discretizza crea automaticamente nuovi campi nominali (insieme) basati sui valori di uno o più campi continui (intervallo numerico) esistenti. Per esempio, è possibile trasformare un campo continuo relativo al reddito in campo categoriale contenente gruppi di reddito come deviazioni dalla media. Dopo aver creato bin per il nuovo campo, è possibile generare un nodo Nuovo campo basato sui punti di taglio. Per ulteriori informazioni, vedere l'argomento [Nodo Discretizza in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create binningnode
set :binningnode.fields = [Na K]
set :binningnode.method = Rank
set :binningnode.fixed_width_name_extension = "_binned"
set :binningnode.fixed_width_add_as = Suffix
set :binningnode.fixed_bin_method = Count
set :binningnode.fixed_bin_count = 10
set :binningnode.fixed_bin_width = 3.5
set :binningnode.tile10 = true
```

| proprietà binningnode | Tipo di dati | Descrizione proprietà |
|----------------------------|---|---|
| fields | <i>[campo1 campo2 ... campon]</i> | Campi continui (intervalli numerici) in attesa di trasformazione. È possibile eseguire la discretizzazione di più campi contemporaneamente. |
| method | FixedWidth EqualCount Rank SDev Optimal | Metodo utilizzato per determinare i punti di taglio per i nuovi bin di campo (categorie). |
| rcalculate_bins | Always IfNecessary | Specifica se i bin vengono ricalcolati e i dati collocati nel bin corrispondente ogni volta che viene eseguito il nodo o se i dati vengono semplicemente inseriti nei bin esistenti e negli eventuali nuovi bin aggiunti. |
| fixed_width_name_extension | <i>stringa</i> | L'estensione di default è <i>_BIN</i> . |
| fixed_width_add_as | Suffix Prefix | Specifica se l'estensione viene aggiunta alla fine (suffisso) del nome del campo oppure all'inizio (prefisso). L'estensione di default è <i>income_BIN</i> . |
| fixed_bin_method | Width Count | |
| fixed_bin_count | <i>intero</i> | Specifica un numero intero utilizzato per determinare il numero di bin a larghezza fissa (categorie) per i nuovi campi. |
| fixed_bin_width | <i>reale</i> | Valore (numero intero o reale) utilizzato per calcolare la "larghezza" del bin. |
| equal_count_name_extension | <i>stringa</i> | L'estensione di default è <i>_TILE</i> . |

| proprietà binningnode | Tipo di dati | Descrizione proprietà |
|--------------------------------|---------------------------|---|
| equal_count_add_as | Suffix Prefix | Specifica un'estensione, un suffisso o un prefisso, utilizzata per il nome del campo generato con p-tili standard. L'estensione di default è <i>_TILE</i> preceduta da <i>N</i> , dove <i>N</i> è il numero percentile. |
| tile4 | <i>flag</i> | Genera quattro bin quantile, ognuno contenente il 25% dei casi. |
| tile5 | <i>flag</i> | Genera cinque bin quintile. |
| tile10 | <i>flag</i> | Genera 10 bin decile. |
| tile20 | <i>flag</i> | Genera 20 bin ventile. |
| tile100 | <i>flag</i> | Genera 100 bin percentile. |
| use_custom_tile | <i>flag</i> | |
| custom_tile_name_extension | <i>stringa</i> | L'estensione di default è <i>_TILEN</i> . |
| custom_tile_add_as | Suffix Prefix | |
| custom_tile | <i>intero</i> | |
| equal_count_method | RecordCount ValueSum | Il metodo RecordCount cerca di assegnare un numero uguale di record a ciascun bin, mentre ValueSum assegna i record in modo che la somma dei valori in ogni bin sia uguale. |
| tied_values_method | Next Current Random | Specifica quali dati relativi ai valori pari merito dei bin devono essere inseriti. |
| rank_order | Ascending Descending | Questa proprietà include Ascending (il valore più basso viene indicato con 1) o Descending (il valore più alto viene indicato con 1). |
| rank_add_as | Suffix Prefix | Questa opzione è applicabile a rango, rango frazionario e percentuale rango. |
| rank | <i>flag</i> | |
| rank_name_extension | <i>stringa</i> | L'estensione di default è <i>_RANK</i> . |
| rank_fractional | <i>flag</i> | Opzioni dei ranghi in cui il valore del nuovo campo equivale al rango diviso per la somma dei pesi dei casi non mancanti. I ranghi frazionari sono compresi nell'intervallo 0-1. |
| rank_fractional_name_extension | <i>stringa</i> | L'estensione di default è <i>_F_RANK</i> . |
| rank_pct | <i>flag</i> | Ogni rango è diviso in base al numero di record con valori validi e moltiplicato per 100. I ranghi frazionari in percentuale sono compresi nell'intervallo 1-100. |
| rank_pct_name_extension | <i>stringa</i> | L'estensione di default è <i>_P_RANK</i> . |
| sdev_name_extension | <i>stringa</i> | |
| sdev_add_as | Suffix Prefix | |
| sdev_count | One Two Three | |

| proprietà binningnode | Tipo di dati | Descrizione proprietà |
|-----------------------------|------------------------|---|
| optimal_name_extension | stringa | L'estensione di default è <i>_OPTIMAL</i> . |
| optimal_add_as | Suffix Prefix | |
| optimal_supervisor_field | campo | Campo scelto come campo supervisore a cui sono correlati i campi selezionati per la discretizzazione. |
| optimal_merge_bins | flag | Specifica che tutti i bin con conteggi di casi ridotti vengono aggiunti a bin più grandi adiacenti. |
| optimal_small_bin_threshold | intero | |
| optimal_pre_bin | flag | Indica che deve essere eseguita la discretizzazione preventiva dell'insieme di dati. |
| optimal_max_bins | intero | Specifica un limite superiore per evitare di creare un numero eccessivamente elevato di bin. |
| optimal_lower_end_point | Inclusive Exclusive | |
| optimal_first_bin | Unbounded Bounded | |
| optimal_last_bin | Unbounded Bounded | |

Proprietà derivenode



Il nodo Nuovo campo modifica valori di dati o crea nuovi campi da uno o più campi esistenti. Crea campi di tipo Formula, Flag, Nominale, Stato, Conteggio e Condizionale. [Per ulteriori informazioni, vedere l'argomento Nuovo campo in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
# Create and configure a Flag Derive field node
create derivenode
rename derive:derivenode as "Flag"
set Flag:derivenode.new_name = "DrugX_Flag"
set Flag:derivenode.result_type = Flag
set Flag:derivenode.flag_true = 1
set Flag:derivenode.flag_false = 0
set Flag:derivenode.flag_expr = "Drug = X"
# Create and configure a Conditional Derive field node
create derivenode
rename derive:derivenode as "Conditional"
set Conditional:derivenode.result_type = Conditional
set Conditional:derivenode.cond_if_cond = "@OFFSET('\Age', 1) = '\Age'"
set Conditional:derivenode.cond_then_expr = "(@OFFSET('\Age', 1) = '\Age') >< @INDEX"
```

```
set Conditional:derivenode.cond_else_expr = "\'Age\'"
```

| proprietà derivenode | Tipo di dati | Descrizione proprietà |
|-----------------------------|---|---|
| new_name | stringa | Nome del nuovo campo. |
| mode | Single Multiple | Specifica campi singoli o multipli. |
| fields | [campo campo campo] | Utilizzata nella modalità Multiple solo per selezionare più campi. |
| name_extension | stringa | Specifica l'estensione del nome del nuovo campo. |
| add_as | Suffix Prefix | Aggiunge l'estensione come prefisso (all'inizio) o come suffisso (alla fine) del nome del campo. |
| result_type | Formula Flag Set State Count Conditional | Sei tipi di nuovi campi che è possibile creare. |
| formula_expr | stringa | Espressione per il calcolo del nuovo valore del campo in qualsiasi nodo Nuovo campo. |
| flag_expr | stringa | |
| flag_true | stringa | |
| flag_false | stringa | |
| set_default | stringa | |
| set_value_cond | stringa | Strutturata per fornire la condizione associata a un valore specificato. Formato di utilizzo: set :derivenode. set_value_cond. Retired = 'age > 65' |
| state_on_val | stringa | Specifica il valore per il nuovo campo quando viene soddisfatta la condizione Attivato. |
| state_off_val | stringa | Specifica il valore per il nuovo campo quando viene soddisfatta la condizione Disattivato. |
| state_on_expression | stringa | |
| state_off_expression | stringa | |
| state_initial | On Off | Assegna a ogni record del nuovo campo un valore iniziale di On o Off. Questo valore può cambiare quando viene soddisfatta ciascuna condizione. |
| count_initial_val | stringa | |
| count_inc_condition | stringa | |
| count_inc_expression | stringa | |
| count_reset_condition | stringa | |
| cond_if_cond | stringa | |
| cond_then_expr | stringa | |
| cond_else_expr | stringa | |

Proprietà ensemblenode



Il nodo Risultati classificatore binario combina due o più insiemi di campi al fine di ottenere previsioni più precise di quelle ricavabili dai singoli modelli. [Per ulteriori informazioni, vedere l'argomento Nodo Risultati classificatore binario in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
# Create and configure an Ensemble node
# Use this node with the models in demos\streams\pm_binaryclassifier.str
create ensemblenode
set :ensemblenode.ensemble_target_field = response
set :ensemblenode.filter_individual_model_output = false
set :ensemblenode.flag_ensemble_method = ConfidenceWeightedVoting
set :ensemblenode.flag_voting_tie_selection = HighestConfidence
```

| proprietà ensemblenode | Tipo di dati | Descrizione proprietà |
|--------------------------------|---|---|
| ensemble_target_field | campo | Specifica il campo obiettivo per tutti i modelli utilizzati nel classificatore binario. |
| filter_individual_model_output | flag | Specifica se i risultati del calcolo del punteggio dei singoli modelli devono essere esclusi. |
| flag_ensemble_method | Voting ConfidenceWeightedVoting RawPropensityWeightedVoting AdjustedPropensityWeightedVoting HighestConfidence AverageRawPropensity AverageAdjustedPropensity | Specifica il metodo utilizzato per determinare il punteggio del classificatore binario. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo flag. |
| set_ensemble_method | Voting ConfidenceWeightedVoting HighestConfidence | Specifica il metodo utilizzato per determinare il punteggio del classificatore binario. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo nominale. |
| flag_voting_tie_selection | Random HighestConfidence RawPropensity AdjustedPropensity | Se è selezionato un metodo di confronto, specifica le modalità di risoluzione delle situazioni di pari merito. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo flag. |

| proprietà ensemblenode | Tipo di dati | Descrizione proprietà |
|--------------------------|-----------------------------|---|
| set_voting_tie_selection | Random HighestConfidence | Se è selezionato un metodo di confronto, specifica le modalità di risoluzione delle situazioni di pari merito. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo nominale. |
| calculate_standard_error | flag | Se il campo obiettivo è continuo viene eseguito per default il calcolo dell'errore standard per calcolare la differenza fra i valori misurati o stimati e i valori veri e per evidenziare il grado di corrispondenza di tali stime. |

Proprietà fillernode



Il nodo Riempimento sostituisce valori di campo e modifica l'archiviazione. È possibile scegliere di sostituire i valori in base a una condizione CLEM, per esempio @BLANK(@FIELD). In alternativa, si può scegliere di sostituire tutti i valori nulli o vuoti con un valore specifico. Il nodo Riempimento è utilizzato spesso in combinazione con il nodo Tipo per sostituire valori mancanti. [Per ulteriori informazioni, vedere l'argomento Nodo Riempimento in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create fillernode
set :fillernode.fields = ['Age']
set :fillernode.replace_mode = Always
set :fillernode.condition = "(\Age' > 60) and (\Sex' = 'M\')"
set :fillernode.replace_with = "\old man\"
```

| proprietà fillernode | Tipo di dati | Descrizione proprietà |
|----------------------|--|--|
| fields | [<i>campo campo</i> <i>campo</i>] | Campi dell'insieme di dati i cui valori saranno esaminati e sostituiti. |
| replace_mode | Always Conditional Blank Null BlankAndNull | È possibile sostituire tutti i valori, i valori vuoti, i valori nulli oppure sostituire i valori basati su una condizione specifica. |
| condition | stringa | |
| replace_with | stringa | |

Proprietà filternode



Il nodo Filtro filtra (ignora) campi, rinomina campi e mappa campi tra i nodi di input. [Per ulteriori informazioni, vedere l'argomento Applicazione di filtri o ridenominazione di campi in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create filternode
set :filternode.default_include = True
set :filternode.new_name.'Drug' = 'Chemical'
set :filternode.include.'Drug' = off
```

Utilizzo della proprietà default_include. Si noti che l'impostazione del valore della proprietà `default_include` non include o esclude automaticamente tutti i campi, ma determina semplicemente l'impostazione di default della selezione corrente. Dal punto di vista funzionale, equivale a fare clic sul pulsante Include i campi per default nella finestra di dialogo Nodo Filtro. Per esempio, si supponga di eseguire lo script seguente:

```
set Filter.default_include=False
# Include only fields in the list
for f in Age Sex
  set Filter.include.^f=True
endfor
```

Il nodo passerà i campi *Età* e *Sesso* e scarterà tutti gli altri. Si supponga ora di eseguire di nuovo lo stesso script ma indicando due campi diversi:

```
set Filter.default_include=False
# Include only fields in the list
for f in Pressione Na
  set Filter.include.^f=True
endfor
```

Verranno aggiunti altri due campi al filtro, per un totale di quattro campi passati (*Età*, *Sesso*, *Pressione*, *Na*). In altre parole, il fatto di reimpostare il valore di `default_include` su `False` non reimposta automaticamente tutti i campi.

In alternativa, se si cambia `default_include` in `True` utilizzando uno script o dalla finestra di dialogo Nodo Filtro, si inverte il comportamento in modo che i quattro campi sopraindicati vengano scartati anziché inclusi. In caso di dubbio, potrebbe essere utile sperimentare con i controlli della finestra di dialogo Nodo Filtro per capire questa interazione.

| proprietà filternode | Tipo di dati | Descrizione proprietà |
|----------------------|--------------|---|
| default_include | <i>flag</i> | Proprietà basata su chiavi utilizzata per specificare se il comportamento di default determina il passaggio o il filtro di campi: NODE.include.FIELDNAME Esempio: set mynode:filternode.default_include = false Si noti che l'impostazione di questa proprietà non include o esclude automaticamente tutti i campi, ma determina semplicemente se i campi selezionati sono inclusi o esclusi per default. Per commenti aggiuntivi, vedere l'esempio seguente. |
| include | <i>flag</i> | Proprietà basata su chiavi utilizzata per l'inclusione e la rimozione. Formato di utilizzo: |

| proprietà filternode | Tipo di dati | Descrizione proprietà |
|----------------------|--------------|---|
| | | NODE.include.FIELDNAME Esempio: set mynode: filternode.include.Age = false |
| new_name | stringa | Esempio: set mynode: filternode.new_name. Age = "age" |

Proprietà historynode



Il nodo Cronologia crea nuovi campi contenenti dati dei campi di record precedenti. I nodi Cronologia sono utilizzati in genere per dati sequenziali, per esempio per dati di serie storica. Prima di utilizzare un nodo Cronologia, può essere utile ordinare i dati con un nodo Ordina. [Per ulteriori informazioni, vedere l'argomento Nodo Cronologia in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create historynode
set :historynode.fields = ['Drug']
set :historynode.offset = 1
set :historynode.span = 3
set :historynode.unavailable = Discard
set :historynode.fill_with = "undef"
```

| proprietà historynode | Tipo di dati | Descrizione proprietà |
|-----------------------|--------------------------|---|
| fields | [campo campo campo] | Campi di cui si desidera creare una cronologia. |
| offset | numero | Specifica il record che precede quello corrente dal quale si desidera estrarre i valori di campo cronologici. |
| span | numero | Specifica il numero di record precedenti a partire dal quale si desidera estrarre i valori. |
| unavailable | Discard Leave Fill | Per la gestione di record senza valori di cronologia, in genere riferiti ai primi record (all'inizio dell'insieme di dati), per i quali non esistono record precedenti da utilizzare come cronologia. |
| fill_with | String Number | Specifica il valore o la stringa da utilizzare per i record in cui non sia disponibile un valore cronologico. |

Proprietà partitionnode



Il nodo Partizione genera un campo di partizione che suddivide i dati in sottoinsiemi separati per le fasi di addestramento, verifica e validazione della creazione del modello. [Per ulteriori informazioni, vedere l'argomento Nodo Partizione in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```

create partitionnode
set :partitionnode.create_validation = True
set :partitionnode.training_size = 33
set :partitionnode.testing_size = 33
set :partitionnode.validation_size = 33
set :partitionnode.set_random_seed = True
set :partitionnode.random_seed = "123"
set :partitionnode.value_mode = System

```

| proprietà partitionnode | Tipo di dati | Descrizione proprietà |
|-------------------------|-----------------------------------|--|
| new_name | <i>stringa</i> | Nome del campo di partizione generato dal nodo. |
| create_validation | <i>flag</i> | Specifica se deve essere creata una partizione di validazione. |
| training_size | <i>intero</i> | Percentuale di record (0–100) da allocare alla partizione di addestramento. |
| testing_size | <i>intero</i> | Percentuale di record (0–100) da allocare alla partizione di test. |
| validation_size | <i>intero</i> | Percentuale di record (0–100) da allocare alla partizione di validazione. Viene ignorata se non viene creata alcuna partizione di validazione. |
| training_label | <i>stringa</i> | Etichetta per la partizione di validazione. |
| testing_label | <i>stringa</i> | Etichetta per la partizione di test. |
| validation_label | <i>stringa</i> | Etichetta per la partizione di validazione. Viene ignorata se non viene creata alcuna partizione di validazione. |
| value_mode | System SystemAndLabel Label | Specifica i valori utilizzati per rappresentare ogni partizione nei dati. Per esempio, il campione di addestramento può essere rappresentato dal valore intero di sistema 1, dall'etichetta Training o da una combinazione dei due 1_Training. |
| set_random_seed | <i>Booleana</i> | Specifica se è necessario utilizzare un seme casuale definito dall'utente. |
| random_seed | <i>intero</i> | Valore del seme casuale definito dall'utente. Per l'utilizzo di questo valore è necessario che <code>set_random_seed</code> sia impostata su True. |
| enable_sql_generation | <i>Booleana</i> | Specifica se utilizzare il push back SQL per assegnare i record alle partizioni. |
| unique_field | | Specifica il campo di input utilizzato per garantire che i record vengano assegnati alle partizioni in modo casuale ma ripetibile. Per l'utilizzo di questo valore è necessario che <code>enable_sql_generation</code> sia impostata su True. |

Proprietà del nodo Ricodifica



Il nodo Ricodifica trasforma un insieme di valori categoriali in un altro. L'operazione di ricodifica consente di comprimere categorie o raggruppare dati per l'analisi. Per ulteriori informazioni, vedere l'argomento [Nodo Ricodifica](#) in il capitolo 4 in *IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output*.

Esempio

```
create reclassifynode
set :reclassifynode.mode = Multiple
set :reclassifynode.replace_field = true
set :reclassifynode.field = "Drug"
set :reclassifynode.new_name = "Chemical"
set :reclassifynode.fields = [Drug, BP]
set :reclassifynode.name_extension = "reclassified"
set :reclassifynode.add_as = Prefix
set :reclassifynode.reclassify.'drugA' = 'Yes'
set :reclassifynode.use_default = True
set :reclassifynode.default = "BrandX"
set :reclassifynode.pick_list = [BrandX, Placebo, Generic]
```

| proprietà reclassifynode | Tipo di dati | Descrizione proprietà |
|--------------------------|--------------------------------------|---|
| mode | Single Multiple | Single riclassifica le categorie per un campo. Multiple attiva le opzioni che consentono la trasformazione di più di un campo contemporaneamente. |
| replace_field | <i>flag</i> | |
| field | <i>stringa</i> | Utilizzata solo in modalità Singola. |
| new_name | <i>stringa</i> | Utilizzata solo in modalità Singola. |
| fields | <i>[campo1 campo2 ... campon]</i> | Utilizzata solo in modalità Multipla. |
| name_extension | <i>stringa</i> | Utilizzata solo in modalità Multipla. |
| add_as | Suffix Prefix | Utilizzata solo in modalità Multipla. |
| reclassify | <i>stringa</i> | Proprietà strutturata per i valori dei campi. Formato di utilizzo: NODE.reclassify. OLD_VALUE Esempio: set :reclassifynode.reclassify.'drugB' = 'Yes' |
| use_default | <i>flag</i> | Utilizza il valore di default. |
| default | <i>stringa</i> | Specificare un valore di default. |
| pick_list | <i>[stringa stringa ... stringa]</i> | Consente all'utente di importare un elenco di nuovi valori noti per popolare l'elenco a discesa nella tabella. Esempio: set :reclassify.pick_list = [fruit dairy cereals] |

Proprietà reordernode



Il nodo Ordina campi definisce l'ordine naturale utilizzato per visualizzare i campi a valle. Tale ordine incide sulla visualizzazione dei campi in vari contesti, quali tabelle, elenchi e Selezione campi. Questa operazione risulta utile se si desidera rendere più visibili i campi interessanti in insiemi di dati di grandi dimensioni. [Per ulteriori informazioni, vedere l'argomento Nodo Ordina campi in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create reordernode
set :reordernode.mode = Custom
set :reordernode.sort_by = Storage
set :reordernode.ascending = "false"
set :reordernode.start_fields = [Age Cholesterol]
set :reordernode.end_fields = [Drug]
```

| proprietà reordernode | Tipo di dati | Descrizione proprietà |
|-----------------------|------------------------------|--|
| mode | Custom Auto | È possibile ordinare i valori automaticamente oppure specificare un ordine personalizzato. |
| sort_by | Name Type Storage | |
| ascending | flag | |
| start_fields | [campo1 campo2 ...campon] | Dopo questi campi vengono inseriti altri nuovi campi. |
| end_fields | [campo1 campo2 ...campon] | Prima di questi campi vengono inseriti altri nuovi campi. |

Proprietà restructurenode



Il nodo Riorganizza converte un campo nominale o flag in un gruppo di campi in cui è possibile inserire i valori di un altro campo. Per esempio, dato un campo denominato *tipo di pagamento*, con valori di *credito*, *contanti* e *debito*, verrebbero creati tre nuovi campi (*credito*, *contanti*, *debito*), ognuno dei quali può contenere il valore del pagamento effettuato. [Per ulteriori informazioni, vedere l'argomento Nodo Riorganizza in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create restructurenode
connect :typenode to :restructurenode
set :restructurenode.fields_from.Drug = ["drugA" "drugX"]
set :restructurenode.include_field_name = "True"
set :restructurenode.value_mode = "OtherFields"
```

```
set :restructurenode.value_fields = ["Age" "BP"]
```

| proprietà restructurenode | Tipo di dati | Descrizione proprietà |
|---------------------------|---|---|
| fields_from | [<i>categoria categoria categoria</i>] all | Per esempio, set :restructurenode.fields_from.Drug = [drugA drugB] crea campi chiamati Drug_drugA e Drug_drugB. Per utilizzare tutte le categorie del campo specificato: set :restructurenode.fields_from.Drug = all |
| include_field_name | <i>flag</i> | Indica se utilizzare il nome del campo nel nome del campo riorganizzato. |
| value_mode | OtherFields Flags | Indica la modalità per specificare i valori dei campi riorganizzati. Con OtherFields, è necessario specificare quali campi utilizzare (vedere sezione seguente). Con Flags, i valori sono flag numerici. |
| value_fields | [<i>campo campo campo</i>] | Necessario se value_mode è OtherFields. Specifica quali campi utilizzare come campi valore. |

Proprietà rfmanalysisnode



Il nodo Analisi RFM (Recency, Frequency, Monetary, Passato recente, Frequenza, Monetario) consente di determinare in modo quantitativo i clienti potenzialmente migliori verificando quanto tempo è trascorso dal loro ultimo acquisto (passato recente), con quale frequenza hanno effettuato acquisti (frequenza) e quanto hanno speso per tutte le transazioni (monetario). [Per ulteriori informazioni, vedere l'argomento Nodo Analisi RFM in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create rfmanalysisnode
connect :rfmaggregatenode to :rfmanalysisnode
set :rfmanalysisnode.recency = Recency
set :rfmanalysisnode.frequency = Frequency
set :rfmanalysisnode.monetary = Monetary
set :rfmanalysisnode.tied_values_method = Next
set :rfmanalysisnode.recalculate_bins = IfNecessary
set :rfmanalysisnode.recency_thresholds = [1, 500, 800, 1500, 2000, 2500]
```

| proprietà rfmanalysisnode | Tipo di dati | Descrizione proprietà |
|---------------------------|---------------|---|
| recency | <i>campo</i> | Specifica il campo Passato recente, il cui valore può essere una data, un timestamp o un semplice numero. |
| frequency | <i>campo</i> | Specifica il campo Frequenza. |
| monetary | <i>campo</i> | Specifica il campo Monetario. |
| recency_bins | <i>intero</i> | Specifica il numero di bin di passato recente da generare. |

| proprietà rfanalysisnode | Tipo di dati | Descrizione proprietà |
|--------------------------|----------------------------------|--|
| recency_weight | numero | Specifica la ponderazione da applicare ai dati di passato recente. Il valore di default è 100. |
| frequency_bins | intero | Specifica il numero di bin di frequenza da generare. |
| frequency_weight | numero | Specifica la ponderazione da applicare ai dati di frequenza. Il valore di default è 10. |
| monetary_bins | intero | Specifica il numero di bin monetari da generare. |
| monetary_weight | numero | Specifica la ponderazione da applicare ai dati monetari. Il valore di default è 1. |
| tied_values_method | Next Current | Specifica quali dati relativi ai valori pari merito dei bin devono essere inseriti. |
| recalculate_bins | Always IfNecessary | |
| add_outliers | flag | Disponibile solo se recalculate_bins è impostata su IfNecessary. Se la proprietà è impostata, i record di valore inferiore a quello del bin più basso vengono aggiunti a tale bin e quelli di valore superiore a quello del bin più alto vengono aggiunti a tale bin. |
| binned_field | Recency Frequency Monetary | |
| recency_thresholds | valore valore | Disponibile solo se recalculate_bins è impostata su Always. Specifica la soglia superiore e inferiore per i bin di passato recente. La soglia superiore di un bin viene utilizzata come soglia inferiore del bin successivo: per esempio, [10 30 60] definirebbe due bin, il primo con soglia superiore e inferiore rispettivamente di 10 e 30 e il secondo con soglia superiore e inferiore rispettivamente di 30 e 60. |
| frequency_thresholds | valore valore | Disponibile solo se recalculate_bins è impostata su Always. |
| monetary_thresholds | valore valore | Disponibile solo se recalculate_bins è impostata su Always. |

Proprietà settoflagnode



Il nodo Crea flag crea campi flag in base ai valori categoriali di uno o più campi nominali. [Per ulteriori informazioni, vedere l'argomento Nodo Crea flag in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create settoflagnode
connect :typenode to :settoflag
set :settoflagnode.fields_from.Drug = ["drugA" "drugX"]
```

```

set :settoflagnode.true_value = "1"
set :settoflagnode.false_value = "0"
set :settoflagnode.use_extension = "True"
set :settoflagnode.extension = "Drug_Flag"
set :settoflagnode.add_as = Suffix
set :settoflagnode.aggregate = True
set :settoflagnode.keys = ['Cholesterol']

```

| proprietà settoflagnode | Tipo di dati | Descrizione proprietà |
|-------------------------|--|---|
| fields_from | [<i>categoria</i> <i>categoria categoria</i>] all | Per esempio, set :settoflagnode.fields_from.Drug = [drugA drugB] crea campi flag chiamati Drug_drugA e Drug_drugB. Per utilizzare tutte le categorie del campo specificato: set :settoflagnode.fields_from.Drug = all |
| true_value | <i>stringa</i> | Specificare il valore vero utilizzato dal nodo durante l'impostazione di un flag. L'impostazione di default è T. |
| false_value | <i>stringa</i> | Specifica il valore falso utilizzato dal nodo durante l'impostazione di un flag. L'impostazione di default è F. |
| use_extension | <i>flag</i> | Utilizzare un'estensione come suffisso o prefisso al nuovo campo flag. |
| extension | <i>stringa</i> | |
| add_as | Suffix Prefix | Specifica se l'estensione viene aggiunta come suffisso o prefisso. |
| aggregate | <i>flag</i> | Raggruppa i record in base ai campi chiave. Tutti i campi flag di un gruppo vengono attivati se un record è impostato su vero. |
| keys | [<i>campo campo</i> <i>campo</i>] | Campi chiave. |

Proprietà statisticstransformnode



Il nodo Trasformazioni Statistics esegue una selezione di comandi di sintassi IBM® SPSS® Statistics rispetto alle sorgenti dati in IBM® SPSS® Modeler. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. [Per ulteriori informazioni, vedere l'argomento Nodo Trasformazioni Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Le proprietà di questo nodo sono descritte in [Proprietà statisticstransformnode a pag. 295](#).

Proprietà timeintervalsnode



Il nodo Intervalli di tempo specifica intervalli e, se necessario, crea etichette per la modellazione di dati di serie storica. Se i valori non sono spaziati in modo uniforme, il nodo può integrare o aggregare i valori in base alle proprie esigenze per generare un intervallo uniforme tra i record. [Per ulteriori informazioni, vedere l'argomento Nodo Intervalli di tempo in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```

create timeintervalsnode
set :timeintervalsnode.interval_type=SecondsPerDay
set :timeintervalsnode.days_per_week=4
set :timeintervalsnode.week_begins_on=Tuesday
set :timeintervalsnode.hours_per_day=10
set :timeintervalsnode.day_begins_hour=7
set :timeintervalsnode.day_begins_minute=5
set :timeintervalsnode.day_begins_second=17
set :timeintervalsnode.mode=Label
set :timeintervalsnode.year_start=2005
set :timeintervalsnode.month_start=January
set :timeintervalsnode.day_start=4
set :timeintervalsnode.pad.AGE=MeanOfRecentPoints
set :timeintervalsnode.agg_mode=Specify
set :timeintervalsnode.agg_set_default=Last

```

| proprietà timeintervalsnode | Tipo di dati | Descrizione proprietà |
|-----------------------------|--|---|
| interval_type | None Periods CyclicPeriods Years Quarters Months DaysPerWeek DaysNonPeriodic HoursPerDay HoursNonPeriodic MinutesPerDay MinutesNonPeriodic SecondsPerDay SecondsNonPeriodic | |
| mode | Label Create | Specifica se si desidera identificare i record consecutivamente o creare le serie in base a un campo data, timestamp o ora specifico. |
| field | <i>campo</i> | Quando si creano le serie a partire dai dati, specifica il campo che indica la data o l'ora di ciascun record. |
| period_start | <i>intero</i> | Specifica l'intervallo iniziale dei periodi o dei periodi ciclici. |
| cycle_start | <i>intero</i> | Ciclo iniziale dei periodi ciclici. |
| year_start | <i>intero</i> | Per i tipi di intervalli, ove applicabile, anno nel quale cade il primo intervallo. |
| quarter_start | <i>intero</i> | Per i tipi di intervalli, ove applicabile, trimestre nel quale cade il primo intervallo. |

| proprietà <i>timeintervalsnode</i> | Tipo di dati | Descrizione proprietà |
|------------------------------------|--|--|
| <i>month_start</i> | gennaio febbraio marzo aprile maggio giugno luglio agosto settembre ottobre novembre dicembre | |
| <i>day_start</i> | <i>intero</i> | |
| <i>hour_start</i> | <i>intero</i> | |
| <i>minute_start</i> | <i>intero</i> | |
| <i>second_start</i> | <i>intero</i> | |
| <i>periods_per_cycle</i> | <i>intero</i> | Per i periodi ciclici, numero entro ciascun ciclo. |
| <i>fiscal_year_begins</i> | gennaio febbraio marzo aprile maggio giugno luglio agosto settembre ottobre novembre dicembre | Per gli intervalli trimestrali, specifica il mese nel quale inizia l'anno fiscale. |
| <i>week_begins_on</i> | Sunday Monday Tuesday Wednesday Thursday Friday Saturday Sunday | Per gli intervalli periodici (giorni alla settimana, ore al giorno, minuti al giorno e secondi al giorno), specifica il giorno in cui inizia la settimana. |
| <i>day_begins_hour</i> | <i>intero</i> | Per gli intervalli periodici (ore al giorno, minuti al giorno e secondi al giorno), specifica l'ora in cui inizia il giorno. Può essere utilizzata in combinazione con <i>day_begins_minute</i> e <i>day_begins_second</i> per specificare un orario esatto, per esempio <i>8:05:01</i> . Vedere l'esempio di utilizzo seguente. |
| <i>day_begins_minute</i> | <i>intero</i> | Per gli intervalli periodici (ore al giorno, minuti al giorno e secondi al giorno), specifica il minuto in cui inizia il giorno (per esempio, il <i>5</i> in <i>8:05</i>). |
| <i>day_begins_second</i> | <i>intero</i> | Per gli intervalli periodici (ore al giorno, minuti al giorno e secondi al giorno), specifica il secondo in cui inizia il giorno (per esempio, il <i>17</i> in <i>8:05:17</i>). |

| proprietà timeintervalsnode | Tipo di dati | Descrizione proprietà |
|--------------------------------|---|---|
| days_per_week | <i>intero</i> | Per gli intervalli periodici (giorni alla settimana, ore al giorno, minuti al giorno e secondi al giorno), specifica il numero di giorni per settimana. |
| hours_per_day | <i>intero</i> | Per gli intervalli periodici (ore al giorno, minuti al giorno e secondi al giorno), specifica il numero di ore del giorno. |
| interval_increment | 1 2 3 4 5 6 10 15 20 30 | Per i minuti al giorno e i secondi al giorno, specifica il numero di minuti o secondi da incrementare per ogni record. |
| field_name_extension | <i>stringa</i> | |
| field_name_extension_as_prefix | <i>flag</i> | |
| date_format | "GGMMAA" "MMGGAA" "AAMMGG" "AAAAMMGG" "AAAAGGG" GIORNO MESE "GG-MM-AA" "GG-MM-AAAA" "MM-GG-AA" "MM-GG-AAAA" "GG-MES-AA" "GG-MES-AAAA" "AAAA-MM-GG" "GG.MM.AA" "GG.MM.AAAA" "MM.GG.AA" "MM.GG.AAAA" "GG.MES.AA" "GG.MES.AAAA" "GG/MM/AA" "GG/MM/AAAA" "MM/GG/AA" "MM/GG/AAAA" "GG/MES/AA" "GG/MES/AAAA" MES AAAA t T AAAA ss ST AAAA | |

| proprietà <i>timeintervalsnode</i> | Tipo di dati | Descrizione proprietà |
|------------------------------------|---|--|
| <i>time_format</i> | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM." "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | |
| <i>aggregate</i> | Mean Sum Mode Min Max First Last TruelfAnyTrue | Specifica il metodo di aggregazione di un campo, per esempio: <i>aggregate.AGE=Mean</i>). |
| <i>pad</i> | Blank MeanOfRecentPoints True False | Specifica il metodo di integrazione di un campo, per esempio: <i>pad.AGE=MeanOfRecentPoints</i>). |
| <i>agg_mode</i> | All Specify | Specifica se aggregare o riempire tutti i campi con le funzioni di default quando necessario oppure specifica i campi e le funzioni da utilizzare. |
| <i>agg_range_default</i> | Mean Sum Mode Min Max | Specifica la funzione di default da utilizzare durante l'aggregazione dei campi continui. |
| <i>agg_set_default</i> | Mode First Last | Specifica la funzione di default da utilizzare durante l'aggregazione dei campi nominali. |
| <i>agg_flag_default</i> | TruelfAnyTrue Mode First Last | |
| <i>pad_range_default</i> | Blank MeanOfRecentPoints | Specifica la funzione di default da utilizzare durante la spaziatura dei campi continui. |
| <i>pad_set_default</i> | Blank MostRecentValue | |
| <i>pad_flag_default</i> | Blank True False | |
| <i>max_records_to_create</i> | <i>intero</i> | Specifica il numero massimo di record da creare durante l'integrazione delle serie. |
| <i>estimation_from_beginning</i> | <i>flag</i> | |

| proprietà timeintervalsnode | Tipo di dati | Descrizione proprietà |
|-----------------------------|----------------|-----------------------|
| estimation_to_end | <i>flag</i> | |
| estimation_start_offset | <i>intero</i> | |
| estimation_num_holdouts | <i>intero</i> | |
| create_future_records | <i>flag</i> | |
| num_future_records | <i>intero</i> | |
| create_future_field | <i>flag</i> | |
| future_field_name | <i>stringa</i> | |

Proprietà transposenode



Il nodo Trasponi scambia i dati delle righe e delle colonne in modo da trasporre i campi in record e i record in campi. [Per ulteriori informazioni, vedere l'argomento Nodo Trasponi in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create transposenode
set :transposenode.transposed_names=Read
set :transposenode.read_from_field="TimeLabel"
set :transposenode.max_num_fields="1000"
set :transposenode.id_field_name="ID"
```

| proprietà transposenode | Tipo di dati | Descrizione proprietà |
|-------------------------|------------------------------|--|
| transposed_names | Prefix Read | È possibile generare automaticamente i nomi dei nuovi campi in base a un prefisso specificato, oppure leggerli da un campo esistente nei dati. |
| prefix | <i>stringa</i> | |
| num_new_fields | <i>intero</i> | Quando si utilizza un prefisso, specifica il numero massimo di nuovi campi da creare. |
| read_from_field | <i>campo</i> | Campo dal quale vengono letti i nomi. Deve essere un campo istanziato, altrimenti si verificherà un errore al momento dell'esecuzione del nodo. |
| max_num_fields | <i>intero</i> | Quando si leggono i nomi da un campo, specifica un limite superiore per evitare di creare un numero eccessivamente elevato di campi. |
| transpose_type | Numeric String Custom | Per default vengono trasposti solo i campi continui (intervalli numerici), ma è possibile scegliere un sottoinsieme personalizzato di campi numerici oppure trasporre tutti i campi stringa. |
| transpose_fields | [<i>campo campo campo</i>] | Specifica i campi da trasporre quando viene utilizzata l'opzione Custom. |
| id_field_name | <i>campo</i> | |

Proprietà typenode



Il nodo Tipo specifica proprietà e metadati di campo. Per esempio, è possibile specificare un livello di misurazione (continuo, nominale, ordinale o flag) per ogni campo, impostare opzioni relative alla gestione dei valori mancanti e dei valori nulli di sistema, impostare il ruolo di un campo per la modellazione, specificare le etichette di campo e valore e specificare i valori per un campo. [Per ulteriori informazioni, vedere l'argomento Nodo Tipo in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create typenode
connect :variablefilenode to :typenode
set :typenode.check.'Cholesterol' = Coerce
set :typenode.direction.'Drug' = Input
set :typenode.type.K = Range
set :typenode.values.Drug = [drugA drugB drugC drugD drugX drugY drugZ]
set :typenode.null_missing.BP = false
set :typenode.whitespace_missing.BP = "false"
set :typenode.description.BP = "Blood Pressure"
set :typenode.value_labels.BP = {{HIGH 'High Blood Pressure'}}{NORMAL 'normal blood pressure'}}
set :typenode.display_places.K = 5
set :typenode.export_places.K = 2
set :typenode.grouping_symbol.Drug = None
set :typenode.column_width.Cholesterol = 25
set :typenode.justify.Cholesterol = Right
```

Si noti che in alcuni casi potrebbe essere necessario istanziare il nodo Tipo per consentire il corretto funzionamento di altri nodi, quali la proprietà fields from del nodo Crea flag. È possibile collegare semplicemente un nodo Tabella ed eseguirlo per istanziare i campi:

```
create tablenode
connect :typenode to :tablenode
execute :tablenode
delete :tablenode
```

| proprietà typenode | Tipo di dati | Descrizione proprietà |
|--------------------|--|--|
| direction | Input Target Both None Partition Split Frequency RecordID | Proprietà basata su chiavi per i ruoli del campo. Formato di utilizzo: NODE.direction.FIELDNAME <i>Nota:</i> i valori In e Out ora sono dichiarati obsoleti. Nelle versioni future potrebbero non essere più supportati. |
| type | Range Flag Set Typeless Discrete Default | Tipo di campo. L'impostazione di type su Default cancellerà qualsiasi impostazione di parametro values e se value_mode presenta il valore Specify, verrà reimpostato su Read. Se value_mode è impostato su Pass o Read, l'impostazione type non avrà effetti su value_mode. Formato di utilizzo: NODE.type.FIELDNAME |

| proprietà typenode | Tipo di dati | Descrizione proprietà |
|--------------------|---|---|
| storage | Unknown String Integer Real Time Date Timestamp | Proprietà basata su chiavi in sola lettura per il tipo di archiviazione del campo. Formato di utilizzo: NODE.storage.FIELDNAME |
| check | None Nullify Coerce Discard Warn Abort | Proprietà basata su chiavi per il controllo del tipo di campo e dell'intervallo. Formato di utilizzo: NODE.check.FIELDNAME |
| values | [valore valore] | Per un campo continuo, il primo valore corrisponde al minimo e l'ultimo valore al massimo. Per i campi nominali, specificare tutti i valori. Nel caso dei campi flag, il primo valore rappresenta <i>falso</i> e l'ultimo valore rappresenta <i>vero</i> . L'impostazione automatica di questa proprietà imposta la proprietà value_mode su Specify. Formato di utilizzo: NODE.values.FIELDNAME |
| value_mode | Read Pass Specify | Determina la modalità di impostazione dei valori. Si noti che non è possibile impostare questa proprietà direttamente su Specify. Per utilizzare valori specifici, impostare la proprietà values. Formato di utilizzo: NODE.value_mode.FIELDNAME |
| extend_values | flag | Si applica quando value_mode è impostato su Read. Per aggiungere valori appena letti a eventuali valori esistenti per il campo, impostare su T. Per scartare i valori esistenti e sostituirli con i valori appena letti, impostare su F. Formato di utilizzo: NODE.extend_values.FIELDNAME |
| enable_missing | flag | Se impostato su T, attiva la registrazione dei valori mancanti per il campo. Formato di utilizzo: NODE.enable_missing.FIELDNAME |
| missing_values | [valore valore...] | Specifica i valori dei dati che indicano dati mancanti. Formato di utilizzo: NODE.missing_values.FIELDNAME |
| range_missing | flag | Specifica se viene definito un intervallo di valori mancanti (vuoti) per un campo. |
| missing_lower | stringa | Se range_missing è impostata su true (vero), specifica il limite inferiore dell'intervallo di valori mancanti. |
| missing_upper | stringa | Se range_missing è impostata su true (vero), specifica il limite superiore dell'intervallo di valori mancanti. |

| proprietà typenode | Tipo di dati | Descrizione proprietà |
|--------------------|---|---|
| null_missing | <i>flag</i> | Se impostata su T, i valori <i>null</i> (valori non definiti, visualizzati come \$null\$ nel software) vengono considerati valori mancanti. Formato di utilizzo: NODE.null_missing.FIELDNAME |
| whitespace_missing | <i>flag</i> | Se impostata su T, i valori contenenti solo uno spazio bianco (spazi, tabulazioni e nuove righe) vengono considerati valori mancanti. Formato di utilizzo: NODE.whitespace_missing.FIELDNAME |
| description | <i>stringa</i> | Specifica la descrizione di un campo. |
| value_labels | <i>[[{Valore EtichettaStringa} { Valore EtichettaStringa} ...]]</i> | Utilizzata per specificare etichette per coppie di valori. Esempio: set :typenode.value_labels.'Drug'=[{drugA label1} {drugB label2}] |
| display_places | <i>intero</i> | Imposta il numero di decimali del campo per la visualizzazione (valida solo per campi con archiviazione REAL). Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.display_places.FIELDNAME |
| export_places | <i>intero</i> | Imposta il numero di decimali del campo per l'esportazione (valida solo per campi con archiviazione REAL). Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.export_places.FIELDNAME |
| decimal_separator | DEFAULT PERIOD COMMA | Imposta il separatore decimale per il campo (valido solo per campi con archiviazione REAL). Formato di utilizzo: NODE.decimal_separator.FIELDNAME |
| date_format | "GGMMAA" "MMGGAA" "AAMMGG" "AAAAMMGG" "AAAAGGG" GIORNO MESE "GG-MM-AA" "GG-MM-AAAA" "MM-GG-AA" "MM-GG-AAAA" "GG-MES-AA" "GG-MES-AAAA" "AAAA-MM-GG" "GG.MM.AA" "GG.MM.AAAA" "MM.GG.AA" "MM.GG.AAAA" | Imposta il formato della data per il campo (valida solo per i campi con archiviazione di tipo DATE o TIMESTAMP). Formato di utilizzo: NODE.date_format.FIELDNAME Esempio: set :tablenode.date_format.'LaunchDate' = "DDMMYY" |

| proprietà typenode | Tipo di dati | Descrizione proprietà |
|--------------------|---|--|
| | "GG.MES.AA" "GG.MES.AAAA" "GG/MM/AA" "GG/MM/AAAA" "MM/GG/AA" "MM/GG/AAAA" "GG/MES/AA" "GG/MES/AAAA" MES AAAA t T AAAA ss ST AAAA | |
| time_format | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM." "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | Imposta il formato dell'ora per il campo (valida solo per i campi con archiviazione di tipo TIME o TIMESTAMP). Formato di utilizzo: NODE.time_format.FIELDNAME Esempio: set :tablnode.time_format.'BOF_enter' = "HHMMSS" |
| number_format | DEFAULT STANDARD SCIENTIFIC CURRENCY | Imposta il formato di visualizzazione dei numeri per il campo. Formato di utilizzo: NODE.number_format.FIELDNAME |
| standard_places | <i>intero</i> | Imposta il numero di decimali del campo per la visualizzazione in formato standard. Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Si noti che la configurazione display_places esistente consente di ottenere lo stesso risultato, tuttavia tale configurazione è ora dichiarata obsoleta. Formato di utilizzo: NODE.standard_places.FIELDNAME |
| scientific_places | <i>intero</i> | Imposta il numero di decimali del campo per la visualizzazione in formato scientifico. Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.scientific_places.FIELDNAME |
| currency_places | <i>intero</i> | Imposta il numero di decimali del campo per la visualizzazione in formato valuta. Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.currency_places.FIELDNAME |

| proprietà typenode | Tipo di dati | Descrizione proprietà |
|---------------------------|---|--|
| grouping_symbol | DEFAULT NONE LOCALE PERIOD COMMA SPACE | Imposta il simbolo di raggruppamento per il campo. Formato di utilizzo: NODE.grouping_symbol.FIELDNAME |
| column_width | <i>intero</i> | Imposta la larghezza delle colonne per il campo. Se viene specificato il valore -1, la larghezza delle colonne verrà impostata su Auto. Formato di utilizzo: NODE.column_width.FIELDNAME |
| justify | AUTO CENTER LEFT RIGHT | Imposta la giustificazione delle colonne per il campo. Formato di utilizzo: NODE.justify.FIELDNAME |

Proprietà dei nodi Grafici

Proprietà comuni dei nodi Grafici

In questa sezione vengono illustrate le proprietà disponibili per i nodi Grafici, incluse le proprietà comuni e quelle specifiche per ogni tipo di nodo.

| Proprietà comuni dei nodi Grafici | Tipo di dati | Descrizione proprietà |
|-----------------------------------|---|---|
| title | <i>stringa</i> | Specifica il titolo. Esempio: "Questo è un titolo." |
| caption | <i>stringa</i> | Specifica la didascalia. Esempio: "Questa è una didascalia." |
| output_mode | Screen File | Specifica se l'output del nodo Grafico viene visualizzato o scritto su un file. |
| output_format | BMP JPEG PNG HTML output (.cou) | Specifica il tipo di output. Il tipo di output consentito varia da nodo a nodo. |
| full_filename | <i>stringa</i> | Specifica il percorso di destinazione e il nome file per l'output generato dal nodo Grafico. |
| use_graph_size | <i>flag</i> | Controlla se le dimensioni del grafico vengono indicate esplicitamente, utilizzando le proprietà di larghezza e altezza seguenti. Influisce solo sui grafici per i quali viene visualizzato l'output su schermo. Non disponibile per il nodo Distribuzione. |
| graph_width | <i>numero</i> | Quando use_graph_size è Vero, imposta la larghezza del grafico in pixel. |
| graph_height | <i>numero</i> | Quando use_graph_size è Vero, imposta l'altezza del grafico in pixel. |

Nota

Disattivazione dei campi facoltativi. È possibile disattivare i campi facoltativi, per esempio un campo di sovrapposizione per i plot, impostando il valore della proprietà su "" (stringa vuota), come illustrato nell'esempio seguente.

```
set :plotnode.color_field = ""
```

Specificare i colori. Il colore dei titoli, delle didascalie, degli sfondi e delle etichette può essere specificato utilizzando le stringhe esadecimali che iniziano con il simbolo del cancelletto (#). Per esempio, per impostare uno sfondo di colore azzurro per il grafico è possibile utilizzare l'istruzione seguente:

```
set mygraph.graph_background="#87CEEB"
```

Le prime due cifre, 87, specificano il contenuto rosso; le due cifre intermedie, CE, specificano il contenuto verde e le ultime due cifre, EB, specificano il contenuto blu. Ogni cifra può assumere un valore compreso nell'intervallo 0–9 o A–F. Utilizzando la combinazione di questi valori è possibile specificare un colore RGB (red-green-blue).

Nota: quando si specificano colori RGB, è possibile utilizzare il pulsante di selezione dei campi disponibile nell'interfaccia utente per definire il codice di colore corretto. È sufficiente posizionare il puntatore del mouse sul colore per visualizzare una descrizione contenente le informazioni desiderate.

Proprietà *collectionnode*



Il nodo Raccolta mostra la distribuzione dei valori di un campo numerico in relazione ai valori di un altro, ovvero crea grafici simili a istogrammi. È utile per illustrare una variabile o un campo i cui valori vengono modificati nel tempo. La grafica 3-D consente inoltre di includere un asse simbolico che visualizza le distribuzioni per categoria. [Per ulteriori informazioni, vedere l'argomento Scheda Plot del nodo Raccolta in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create collectionnode
position :collectionnode at ^posX ^posY
# "Plot" tab
set :collectionnode.three_D = True
set :collectionnode.collect_field = 'Drug'
set :collectionnode.over_field = 'Age'
set :collectionnode.by_field = 'BP'
set :collectionnode.operation = Sum
# "Overlay" section
set :collectionnode.color_field = 'Drug'
set :collectionnode.panel_field = 'Sex'
set :collectionnode.animation_field = ''
# "Options" tab
set :collectionnode.range_mode = Automatic
set :collectionnode.range_min = 1
set :collectionnode.range_max = 100
set :collectionnode.bins = ByNumber
set :collectionnode.num_bins = 10
set :collectionnode.bin_width = 5
```

| Proprietà <i>collectionnode</i> | Tipo di dati | Descrizione proprietà |
|---------------------------------|----------------|-----------------------|
| over_field | <i>campo</i> | |
| over_label_auto | <i>flag</i> | |
| over_label | <i>stringa</i> | |
| collect_field | <i>campo</i> | |
| collect_label_auto | <i>flag</i> | |
| collect_label | <i>stringa</i> | |
| three_D | <i>flag</i> | |

| Proprietà collectionnode | Tipo di dati | Descrizione proprietà |
|--------------------------|-----------------------------------|--|
| by_field | campo | |
| by_label_auto | flag | |
| by_label | stringa | |
| operation | Sum Mean Min Max SDev | |
| color_field | stringa | |
| panel_field | stringa | |
| animation_field | stringa | |
| range_mode | Automatic UserDefined | |
| range_min | numero | |
| range_max | numero | |
| bins | ByNumber ByWidth | |
| num_bins | numero | |
| bin_width | numero | |
| use_grid | flag | |
| graph_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| page_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |

Proprietà distributionnode



Il nodo Distribuzione mostra l'occorrenza di valori simbolici (categoriali), per esempio tipo o genere di ipoteca. In genere è possibile utilizzare un nodo Distribuzione per mostrare sbilanciamenti nei dati, che possono essere successivamente corretti con un nodo Bilanciamento prima di creare un modello. [Per ulteriori informazioni, vedere l'argomento Nodo Distribuzione in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create distributionnode
# "Plot" tab
set :distributionnode.plot = Flags
set :distributionnode.x_field = 'Age'
set :distributionnode.color_field = 'Drug'
set :distributionnode.normalize = True
set :distributionnode.sort_mode = ByOccurrence
set :distributionnode.use_proportional_scale = True
```

| Proprietà distributionnode | Tipo di dati | Descrizione proprietà |
|----------------------------|-------------------------|-----------------------|
| plot | SelectedFields Flags | |
| x_field | campo | |

| Proprietà <i>distributionnode</i> | Tipo di dati | Descrizione proprietà |
|-----------------------------------|----------------------------|-----------------------|
| color_field | <i>campo</i> | Campo sovrapposto. |
| normalize | <i>flag</i> | |
| sort_mode | ByOccurrence Alphabetic | |
| use_proportional_scale | <i>flag</i> | |

Proprietà *evaluationnode*



Il nodo Valutazione facilita la valutazione e il confronto di modelli predittivi. Il grafico di valutazione mostra il comportamento dei modelli nella previsione di particolari risultati. Ordina i record in base al valore previsto e alla confidenza della previsione, quindi li suddivide in gruppi di uguale dimensione (**quantili**) e infine rappresenta il valore del criterio commerciale per ciascun quantile, dal più alto al più basso. I modelli multipli sono mostrati nel grafico come linee separate. [Per ulteriori informazioni, vedere l'argomento Nodo Valutazione in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create evaluationnode
position :evaluationnode at ^posX ^posY
#"Plot" tab
set :evaluationnode.chart_type = Gains
set :evaluationnode.cumulative = False
set :evaluationnode.field_detection_method = Name
set :evaluationnode.inc_baseline = True
set :evaluationnode.n_tile = Deciles
set :evaluationnode.style = Point
set :evaluationnode.point_type = Dot
set :evaluationnode.use_fixed_cost = True
set :evaluationnode.cost_value = 5.0
set :evaluationnode.cost_field = 'Na'
set :evaluationnode.use_fixed_revenue = True
set :evaluationnode.revenue_value = 30.0
set :evaluationnode.revenue_field = 'Age'
set :evaluationnode.use_fixed_weight = True
set :evaluationnode.weight_value = 2.0
set :evaluationnode.weight_field = 'K'
```

| Proprietà <i>evaluationnode</i> | Tipo di dati | Descrizione proprietà |
|---------------------------------|--|-----------------------|
| chart_type | Gains Response Lift Profit ROI | |
| inc_baseline | <i>flag</i> | |
| field_detection_method | Metadata Name | |
| use_fixed_cost | <i>flag</i> | |
| cost_value | <i>numero</i> | |

| Proprietà evaluationnode | Tipo di dati | Descrizione proprietà |
|--------------------------|---|-----------------------|
| cost_field | <i>stringa</i> | |
| use_fixed_revenue | <i>flag</i> | |
| revenue_value | <i>numero</i> | |
| revenue_field | <i>stringa</i> | |
| use_fixed_weight | <i>flag</i> | |
| weight_value | <i>numero</i> | |
| weight_field | <i>campo</i> | |
| n_tile | Quartiles Quintles Deciles Vingtiles Percentiles 1000-tiles | |
| cumulative | <i>flag</i> | |
| style | Line Point | |
| point_type | Rettangolo Punto Triangolo Esagono Segno più Pentagono Stella Farfallino Trattino orizzontale Trattino verticale Croce Fabbrica Casa Cattedrale Cupola Triangolo concavo Geoide Occhio di gatto Cuscino quadrato Rettangolo arrotondato Ventaglio | |
| export_data | <i>flag</i> | |
| data_filename | <i>stringa</i> | |
| delimiter | <i>stringa</i> | |
| new_line | <i>flag</i> | |
| inc_field_names | <i>flag</i> | |
| inc_best_line | <i>flag</i> | |
| inc_business_rule | <i>flag</i> | |
| business_rule_condition | <i>stringa</i> | |
| plot_score_fields | <i>flag</i> | |
| score_fields | <i>[campo1 ... campoN]</i> | |
| target_field | <i>campo</i> | |
| use_hit_condition | <i>flag</i> | |

| Proprietà evaluationnode | Tipo di dati | Descrizione proprietà |
|--------------------------|--------------|-----------------------|
| hit_condition | stringa | |
| use_score_expression | flag | |
| score_expression | stringa | |
| caption_auto | flag | |

Proprietà graphboardnode



Il nodo Lavagna grafica offre numerosi tipi di grafici diversi in un unico nodo. Con questo nodo è possibile scegliere i campi di dati da esplorare e selezionare quindi un grafico fra quelli disponibili per i dati selezionati. Il nodo esclude automaticamente tutti i tipi di grafici non adatti ai campi selezionati. [Per ulteriori informazioni, vedere l'argomento Nodo Lavagna grafica in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Nota: se si imposta una proprietà non valida per il tipo di grafico, per esempio specificando y_field per un istogramma, tale proprietà viene ignorata.

Esempio

```
create graphboardnode
connect DRUG4n to :graphboardnode
set :graphboardnode.graph_type="Line"
set :graphboardnode.x_field = "K"
set :graphboardnode.y_field = "Na"
execute :graphboardnode
```

| Proprietà graphboard | Tipo di dati | Descrizione proprietà |
|----------------------|---|--|
| graph_type | Histogram Dotplot Scatterplot BinnedScatter HexBinScatter Line Path Area 3DHistogram 3DDensity Bubble PieCounts Pie 3DPie BarCounts Bar Surface Ribbon 3DArea Boxplot Heatmap SPLM Parallel LinkAnalysis | Identifica il tipo di grafico. |
| x_field | campo | Specifica un'etichetta personalizzata per l'asse x. Disponibile solo per le etichette. |

| Proprietà graphboard | Tipo di dati | Descrizione proprietà |
|----------------------|--------------|--|
| y_field | campo | Specifica un'etichetta personalizzata per l'asse y. Disponibile solo per le etichette. |
| z_field | campo | |
| color_field | campo | Utilizzata nelle mappe termiche. |
| size_field | campo | Utilizzata nei grafici a bolle. |
| categories_field | campo | |
| values_field | campo | |
| rows_field | campo | |
| columns_field | campo | |
| fields | campo | |

Proprietà histogramnode



Il nodo Istogramma mostra l'occorrenza dei valori per i campi numerici. Viene spesso utilizzato per analizzare i dati prima delle manipolazioni e della generazione del modello. Come il nodo Distribuzione, anche il nodo Istogramma viene frequentemente utilizzato per rivelare sbilanciamenti nei dati. [Per ulteriori informazioni, vedere l'argomento Scheda Plot del nodo Istogramma in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create histogramnode
position :histogramnode at ^posX ^posY
# "Plot" tab
set :histogramnode.field = 'Drug'
set :histogramnode.color_field = 'Drug'
set :histogramnode.panel_field = 'Sex'
set :histogramnode.animation_field = ''
# "Options" tab
set :histogramnode.range_mode = Automatic
set :histogramnode.range_min = 1.0
set :histogramnode.range_max = 100.0
set :histogramnode.num_bins = 10
set :histogramnode.bin_width = 10
set :histogramnode.normalize = True
set :histogramnode.separate_bands = False
```

| Proprietà histogramnode | Tipo di dati | Descrizione proprietà |
|-------------------------|--------------------------|-----------------------|
| field | campo | |
| color_field | campo | |
| panel_field | campo | |
| animation_field | campo | |
| range_mode | Automatic UserDefined | |
| range_min | numero | |
| range_max | numero | |

| Proprietà histogramnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---------------------|---|
| bins | ByNumber ByWidth | |
| num_bins | numero | |
| bin_width | numero | |
| normalize | flag | |
| separate_bands | flag | |
| x_label_auto | flag | |
| x_label | stringa | |
| y_label_auto | flag | |
| y_label | stringa | |
| use_grid | flag | |
| graph_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| page_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| normal_curve | flag | Indica se la curva di distribuzione normale deve essere visualizzata nell'output. |

Proprietà multiplotnode



Un nodo Plot multiplo crea un grafico che consente di visualizzare più campi *Y* in un singolo campo *X*. I campi *Y* sono rappresentati come linee colorate e ognuno di essi equivale a un nodo Plot con lo Stile impostato su Linea e la Modalità *X* impostata su Ordina. I plot multipli sono utili quando si desidera esplorare la fluttuazione di numerose variabili nel tempo. [Per ulteriori informazioni, vedere l'argomento Nodo Plot multiplo in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create multiplotnode
# "Plot" tab
set :multiplotnode.x_field = 'Age'
set :multiplotnode.y_fields = ['Drug' 'BP']
set :multiplotnode.panel_field = 'Sex'
# "Overlay" section
set :multiplotnode.animation_field = ''
set :multiplotnode.tooltip = "test"
set :multiplotnode.normalize = True
set :multiplotnode.use_overlay_expr = False
set :multiplotnode.overlay_expression = "test"
set :multiplotnode.records_limit = 500
set :multiplotnode.if_over_limit = PlotSample
```

| Proprietà multiplotnode | Tipo di dati | Descrizione proprietà |
|-------------------------|------------------------|-----------------------|
| x_field | campo | |
| y_fields | [campo campo campo] | |

| Proprietà multiplotnode | Tipo di dati | Descrizione proprietà |
|-------------------------|-----------------------------------|--|
| panel_field | campo | |
| animation_field | campo | |
| normalize | flag | |
| use_overlay_expr | flag | |
| overlay_expression | stringa | |
| records_limit | numero | |
| if_over_limit | PlotBins PlotSample PlotAll | |
| x_label_auto | flag | |
| x_label | stringa | |
| y_label_auto | flag | |
| y_label | stringa | |
| use_grid | flag | |
| graph_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| page_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |

Proprietà plotnode



Il nodo Plot mostra la relazione tra campi numerici. È possibile creare un grafico plot utilizzando punti (un grafico a dispersione) oppure linee. [Per ulteriori informazioni, vedere l'argomento Nodo Plot in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create plotnode
# "Plot" tab
set :plotnode.three_D = True
set :plotnode.x_field = 'BP'
set :plotnode.y_field = 'Cholesterol'
set :plotnode.z_field = 'Drug'
# "Overlay" section
set :plotnode.color_field = 'Drug'
set :plotnode.size_field = 'Age'
set :plotnode.shape_field = ''
set :plotnode.panel_field = 'Sex'
set :plotnode.animation_field = 'BP'
set :plotnode.transp_field = ''
set :plotnode.style = Point
# "Output" tab
set :plotnode.output_mode =
set :plotnode.output_format = JPEG
```

```
set :plotnode.full_filename = "C:/Temp/Graph_Output/plot_output.jpeg"
```

| Proprietà plotnode | Tipo di dati | Descrizione proprietà |
|--------------------|---|--|
| x_field | campo | Specifica un'etichetta personalizzata per l'asse x. Disponibile solo per le etichette. |
| y_field | campo | Specifica un'etichetta personalizzata per l'asse y. Disponibile solo per le etichette. |
| three_D | flag | Specifica un'etichetta personalizzata per l'asse y. Disponibile solo per le etichette nei grafici 3-D. |
| z_field | campo | |
| color_field | campo | Campo sovrapposto. |
| size_field | campo | |
| shape_field | campo | |
| panel_field | campo | Specifica un campo nominale o flag da utilizzare per creare un grafico distinto per ciascuna categoria. I grafici vengono rappresentati in pannelli in un'unica finestra di output. |
| animation_field | campo | Specifica un campo nominale o flag per rappresentare le categorie per i valori dei dati creando una serie di grafici visualizzati in sequenza tramite l'animazione. |
| transp_field | campo | Specifica un campo per rappresentare le categorie per i valori dei dati utilizzando un livello di trasparenza diverso per ciascuna categoria. Non disponibile per i grafici lineari. |
| overlay_type | None Smoother Function | Specifica se viene visualizzata una funzione sovrapposta o un livellamento di LOESS. |
| overlay_expression | stringa | Specifica l'espressione utilizzata quando overlay_type è impostata su Function. |
| style | Point Line | |
| point_type | Rettangolo Punto Triangolo Esagono Segno più Pentagono Stella Farfallino Trattino orizzontale Trattino verticale Croce Fabbrica Casa Cattedrale Cupola Triangolo concavo Geoide Occhio di gatto Cuscino quadrato Rettangolo arrotondato Ventaglio | |

| Proprietà plotnode | Tipo di dati | Descrizione proprietà |
|--------------------|-----------------------------------|--|
| x_mode | Sort Overlay AsRead | |
| x_range_mode | Automatic UserDefined | |
| x_range_min | numero | |
| x_range_max | numero | |
| y_range_mode | Automatic UserDefined | |
| y_range_min | numero | |
| y_range_max | numero | |
| z_range_mode | Automatic UserDefined | |
| z_range_min | numero | |
| z_range_max | numero | |
| jitter | flag | |
| records_limit | numero | |
| if_over_limit | PlotBins PlotSample PlotAll | |
| x_label_auto | flag | |
| x_label | stringa | |
| y_label_auto | flag | |
| y_label | stringa | |
| z_label_auto | flag | |
| z_label | stringa | |
| use_grid | flag | |
| graph_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| page_background | colore | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| use_overlay_expr | flag | Reso obsoleto a favore di overlay_type. |

Proprietà timeplotnode



Il nodo Plot tempo visualizza uno o più insiemi di dati di serie storica. In genere, si utilizza prima un nodo Intervalli di tempo per creare un campo *EtichettaTempo*, che viene utilizzato per attribuire un'etichetta all'asse x. [Per ulteriori informazioni, vedere l'argomento Nodo Plot tempo in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create timeplotnode
set :timeplotnode.y_fields = ['sales' 'men' 'women']
set :timeplotnode.panel = True
set :timeplotnode.normalize = True
set :timeplotnode.line = True
```

```

set :timeplotnode.smoother = True
set :timeplotnode.use_records_limit = True
set :timeplotnode.records_limit = 2000
# Appearance settings
set :timeplotnode.symbol_size = 2.0

```

| Proprietà timeplotnode | Tipo di dati | Descrizione proprietà |
|------------------------|---|---|
| plot_series | Series Models | |
| use_custom_x_field | <i>flag</i> | |
| x_field | <i>campo</i> | |
| y_fields | [<i>campo campo</i> <i>campo</i>] | |
| panel | <i>flag</i> | |
| normalize | <i>flag</i> | |
| line | <i>flag</i> | |
| points | <i>flag</i> | |
| point_type | Rettangolo Punto Triangolo Esagono Segno più Pentagono Stella Farfallino Trattino orizzontale Trattino verticale Croce Fabbrica Casa Cattedrale Cupola Triangolo concavo Geoide Occhio di gatto Cuscino quadrato Rettangolo arrotondato Ventaglio | |
| smoother | <i>flag</i> | È possibile aggiungere livellamenti al plot solo se si imposta panel su True. |
| use_records_limit | <i>flag</i> | |
| records_limit | <i>intero</i> | |
| symbol_size | <i>numero</i> | Specifica la dimensione di un simbolo. Ad esempio, <code>set :webnode.symbol_size = 5.5</code> crea una dimensione del simbolo maggiore di quella di default. |
| panel_layout | Horizontal Vertical | |

Proprietà webnode



Il nodo Web illustra l'intensità della relazione tra valori di due o più campi simbolici (categoriali). Il grafico utilizza linee di spessore diverso per indicare l'intensità della connessione. Un nodo Web può essere utilizzato, per esempio, per analizzare la relazione tra l'acquisto di vari oggetti in un sito di e-commerce. [Per ulteriori informazioni, vedere l'argomento Nodo Web in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create webnode
# "Plot" tab
set :webnode.use_directed_web = True
set :webnode.to_field = 'Drug'
set :webnode.fields = ['BP' 'Cholesterol' 'Sex' 'Drug']
set :webnode.from_fields = ['BP' 'Cholesterol' 'Sex']
set :webnode.true_flags_only = False
set :webnode.line_values = Absolute
set :webnode.strong_links_heavier = True
# "Options" tab
set :webnode.max_num_links = 300
set :webnode.links_above = 10
set :webnode.num_links = ShowAll
set :webnode.discard_links_min = True
set :webnode.links_min_records = 5
set :webnode.discard_links_max = True
set :webnode.weak_below = 10
set :webnode.strong_above = 19
set :webnode.link_size_continuous = True
set :webnode.web_display = Circular
```

| Proprietà webnode | Tipo di dati | Descrizione proprietà |
|----------------------|---|-----------------------|
| use_directed_web | <i>flag</i> | |
| fields | [<i>campo campo campo</i>] | |
| to_field | <i>campo</i> | |
| from_fields | [<i>campo campo campo</i>] | |
| true_flags_only | <i>flag</i> | |
| line_values | Absolute OverallPct PctLarger PctSmaller | |
| strong_links_heavier | <i>flag</i> | |
| num_links | ShowMaximum ShowLinksAbove ShowAll | |
| max_num_links | <i>numero</i> | |
| links_above | <i>numero</i> | |
| discard_links_min | <i>flag</i> | |

| Proprietà webnode | Tipo di dati | Descrizione proprietà |
|----------------------|---|--|
| links_min_records | <i>numero</i> | |
| discard_links_max | <i>flag</i> | |
| links_max_records | <i>numero</i> | |
| weak_below | <i>numero</i> | |
| strong_above | <i>numero</i> | |
| link_size_continuous | <i>flag</i> | |
| web_display | Circular Network Directed Grid | |
| graph_background | <i>colore</i> | I colori standard dei grafici sono descritti all'inizio di questa sezione. |
| symbol_size | <i>numero</i> | Specifica la dimensione di un simbolo. Ad esempio, <code>set :webnode.symbol_size = 5.5</code> crea una dimensione del simbolo maggiore di quella di default. |

Proprietà dei nodi Modelli

Proprietà comuni dei nodi Modelli

Le seguenti proprietà sono comuni ad alcuni o a tutti i nodi Modelli. Le eventuali eccezioni sono segnalate, ove necessario, nella documentazione relativa ai singoli nodi Modelli.

| Proprietà | Valori | Descrizione proprietà |
|------------------------|---|---|
| custom_fields | <i>flag</i> | Se vera, consente di specificare i campi obiettivo, di input e di altro tipo per il nodo corrente. Se falsa, vengono utilizzate le impostazioni correnti di un nodo Tipo a monte. |
| target o targets | <i>campo</i> o [<i>campo1 ... campoN</i>] | Specifica un unico campo obiettivo o più campi obiettivo, a seconda del tipo di modello. |
| inputs | [<i>campo1 ... campoN</i>] | I campi di input o predittore utilizzati dal modello. |
| partition | <i>campo</i> | |
| use_partitioned_data | <i>flag</i> | Se è definito un campo di partizione, questa opzione garantisce che per la creazione del modello verranno utilizzati solo i dati della partizione di addestramento. |
| use_split_data | <i>flag</i> | |
| splits | [<i>campo1 ... campoN</i>] | Specifica il campo o i campi da usare per la creazione di modelli suddivisi. Ha effetto solo se <i>use_split_data</i> è impostata su True. |
| use_frequency | <i>flag</i> | I campi peso e frequenza vengono utilizzati da determinati modelli, come riportato per ogni tipo di modello. |
| frequency_field | <i>campo</i> | |
| use_weight | <i>flag</i> | |
| weight_field | <i>campo</i> | |
| use_model_name | <i>flag</i> | |
| model_name | <i>stringa</i> | Nome personalizzato per il nuovo modello. |
| mode | Simple Expert | |

Proprietà anomalydetectionnode



Il nodo Rilevamento anomalie identifica casi insoliti, o valori anomali, non conformi a schemi di dati “normali”. Con questo nodo è possibile identificare valori anomali anche se questi non rientrano in schemi precedentemente conosciuti e anche se l’utente non sa esattamente ciò che sta cercando. [Per ulteriori informazioni, vedere l’argomento Nodo Rilevamento anomalie in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create anomalydetectionnode
set :anomalydetectionnode.anomaly_method=PerRecords
set :anomalydetectionnode.percent_records=95
set :anomalydetectionnode.mode=Expert
set :anomalydetectionnode.peer_group_num_auto=true
set :anomalydetectionnode.min_num_peer_groups=3
set :anomalydetectionnode.max_num_peer_groups=10
```

| Proprietà anomalydetectionnode | Valori | Descrizione proprietà |
|--------------------------------|--|---|
| inputs | <i>[campo1 ... campoN]</i> | I modelli Rilevamento anomalie effettuano lo screening dei record in base ai campi di input specificati. Non utilizzano un campo obiettivo, né i campi peso e frequenza. Per ulteriori informazioni, vedere l’argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| mode | Expert Simple | |
| anomaly_method | IndexLevel PerRecords NumRecords | Specifica il metodo utilizzato per determinare il valore di interruzione per contrassegnare i record come anomali. |
| index_level | <i>numero</i> | Specifica il valore di interruzione minimo per contrassegnare le anomalie. |
| percent_records | <i>numero</i> | Imposta la soglia per contrassegnare i record in base alla percentuale di record nei dati di addestramento. |
| num_records | <i>numero</i> | Imposta la soglia per contrassegnare i record in base al numero di record nei dati di addestramento. |
| num_fields | <i>intero</i> | Numero di campi da segnalare per ciascun record anomalo. |
| impute_missing_values | <i>flag</i> | |
| adjustment_coeff | <i>numero</i> | Valore utilizzato per bilanciare il peso relativo attribuito ai campi continui e categoriali nel calcolo della distanza. |
| peer_group_num_auto | <i>flag</i> | Calcola automaticamente il numero dei gruppi di lavoro. |

| Proprietà anomalydetectionnode | Valori | Descrizione proprietà |
|--------------------------------|---------------|--|
| min_num_peer_groups | <i>intero</i> | Specifica il numero minimo di gruppi di lavoro utilizzati quando peer_group_num_auto è impostata su True. |
| max_num_per_groups | <i>intero</i> | Specifica il numero massimo di gruppi di lavoro. |
| num_peer_groups | <i>intero</i> | Specifica il numero di gruppi di lavoro utilizzati quando peer_group_num_auto è impostata su False. |
| noise_level | <i>numero</i> | Determina come trattare i valori anomali durante il raggruppamento tramite cluster. Specificare un valore compreso tra 0 e 0.5. |
| noise_ratio | <i>numero</i> | Specifica la parte di memoria allocata per il componente da utilizzare per la memorizzazione del rumore. Specificare un valore compreso tra 0 e 0.5. |

Proprietà apriorinode



Il nodo Apriori estrae un insieme di regole dai dati, estrapolando le regole con il più alto contenuto di informazioni. Apriori offre cinque diversi metodi per la selezione delle regole e utilizza uno schema di indicizzazione sofisticato per elaborare in modo efficiente insiemi di dati di grandi dimensioni. In caso di problemi complessi, l'addestramento di Apriori è in genere più rapido. Apriori non ha un limite arbitrario per quanto riguarda il numero di regole che possono essere mantenute e può gestire regole con un massimo di 32 precondizioni. Apriori richiede che tutti i campi di input e output siano categoriali ma garantisce prestazioni migliori perché è ottimizzato per questo tipo di dati. [Per ulteriori informazioni, vedere l'argomento Nodo Apriori in il capitolo 12 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create apriorinode
# "Fields" tab
set :apriorinode.custom_fields = True
set :apriorinode.use_transactional_data = True
set :apriorinode.id_field = 'Age'
set :apriorinode.contiguous = True
set :apriorinode.content_field = 'Drug'
# These seem to have changed, used to be:
#help set :apriorinode.consequents = ['Age']
#help set :apriorinode.antecedents = ['BP' 'Cholesterol' 'Drug']
# now it seems we have;
#help set :apriorinode.content = ['Age']
set :apriorinode.partition = Test
# "Model" tab
set :apriorinode.use_model_name = False
set :apriorinode.model_name = "Apriori_bp_choles_drug"
set :apriorinode.min_supp = 7.0
```

```

set :apriorinode.min_conf = 30.0
set :apriorinode.max_antecedents = 7
set :apriorinode.true_flags = False
set :apriorinode.optimize = Memory
# "Expert" tab
set :apriorinode.mode = Expert
set :apriorinode.evaluation = ConfidenceRatio
set :apriorinode.lower_bound = 7

```

| Proprietà apriorinode | Valori | Descrizione proprietà |
|------------------------|--|---|
| consequents | <i>campo</i> | I modelli Apriori utilizzano antecedenti e conseguenti al posto dei campi obiettivo e di input standard. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| antecedents | [<i>campo1 ... campoN</i>] | |
| min_supp | <i>numero</i> | |
| min_conf | <i>numero</i> | |
| max_antecedents | <i>numero</i> | |
| true_flags | <i>flag</i> | |
| optimize | Speed Memory | |
| use_transactional_data | <i>flag</i> | |
| contiguous | <i>flag</i> | |
| id_field | <i>stringa</i> | |
| content_field | <i>stringa</i> | |
| mode | Simple Expert | |
| evaluation | RuleConfidence DifferenceToPrior ConfidenceRatio InformationDifference NormalizedChiSquare | |
| lower_bound | <i>numero</i> | |
| optimize | Speed Memory | Utilizzare per specificare se ottimizzare la velocità o la memoria durante la creazione del modello. |

Proprietà autoclassifiernode



Il nodo Classificatore automatico crea e confronta svariati tipi di modelli per risultati binari (sì o no, abbandono oppure no e così via), consentendo di scegliere l'approccio migliore per una determinata analisi. Sono supportati numerosi algoritmi di modellazione ed è possibile selezionare i metodi da utilizzare, le opzioni specifiche per ognuno di essi e i criteri per confrontare i risultati. Il nodo genera un insieme di modelli basato sulle opzioni specificate e classifica i candidati migliori in base ai criteri indicati. [Per ulteriori informazioni, vedere l'argomento Nodo Classificatore automatico in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```

create autotclassifiernode
set :autotclassifiernode.ranking_measure=Accuracy
set :autotclassifiernode.ranking_dataset=Training
set :autotclassifiernode.enable_accuracy_limit=true
set :autotclassifiernode.accuracy_limit=0.9
set :autotclassifiernode.calculate_variable_importance=true
set :autotclassifiernode.use_costs=true
set :autotclassifiernode.svm=false

```

| Proprietà autotclassifiernode | Valori | Descrizione proprietà |
|----------------------------------|---|--|
| target | <i>campo</i> | Per i campi obiettivo, il nodo Classificatore automatico richiede un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare i campi frequenza e peso. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| ranking_measure | Accuracy Area_under_curve Profit Lift Num_variables | |
| ranking_dataset | Training Test | |
| number_of_models | <i>intero</i> | Numero dei modelli da includere nell'insieme di modelli. Specificare un numero intero compreso fra 1 e 100. |
| calculate_variable_importance | <i>flag</i> | |
| enable_accuracy_limit | <i>flag</i> | |
| accuracy_limit | <i>intero</i> | Numero intero compreso fra 0 e 100. |
| enable_area_under_curve_limit | <i>flag</i> | |
| area_under_curve_limit | <i>numero</i> | Numero reale compreso tra 0.0 e 1.0. |
| enable_profit_limit | <i>flag</i> | |
| profit_limit | <i>numero</i> | Numero intero maggiore di 0. |
| enable_lift_limit | <i>flag</i> | |
| lift_limit | <i>numero</i> | Numero reale maggiore di 1,0. |
| enable_number_of_variables_limit | <i>flag</i> | |
| number_of_variables_limit | <i>numero</i> | Numero intero maggiore di 0. |
| use_fixed_cost | <i>flag</i> | |
| fixed_cost | <i>numero</i> | Numero reale maggiore di 0.0. |
| variable_cost | <i>campo</i> | |
| use_fixed_revenue | <i>flag</i> | |
| fixed_revenue | <i>numero</i> | Numero reale maggiore di 0.0. |
| variable_revenue | <i>campo</i> | |
| use_fixed_weight | <i>flag</i> | |

| Proprietà autotassificatore | Valori | Descrizione proprietà |
|--|---------|---|
| fixed_weight | numero | Numero reale maggiore di 0,0. |
| variable_weight | campo | |
| lift_percentile | numero | Numero intero compreso fra 0 e 100. |
| enable_model_build_time_limit | flag | |
| model_build_time_limit | numero | Numero intero impostato sul numero di minuti per limitare il tempo impiegato per creare ogni singolo modello. |
| enable_stop_after_time_limit | flag | |
| stop_after_time_limit | numero | Numero reale impostato sul numero di ore per limitare il tempo complessivo impiegato per l'esecuzione di un classificatore automatico. |
| enable_stop_after_valid_model_produced | flag | |
| use_costs | flag | |
| <algorithm> | flag | Attiva o disattiva l'utilizzo di un algoritmo specifico, ad esempio: set :autotassificatore.chaid=true |
| <algorithm>.<property> | stringa | Imposta il valore di una proprietà di un algoritmo specifico. Per ulteriori informazioni, vedere l'argomento Impostazione delle proprietà degli algoritmi a pag. 194. |

Impostazione delle proprietà degli algoritmi

Per i nodi Classificatore automatico, Numerico automatico e Cluster automatico, le proprietà degli algoritmi specifici utilizzati dal nodo si possono impostare utilizzando il formato generico:

```
set :autotassificatore.<algoritmo>.<proprietà> = <valore>
```

```
set :autonumericonodo.<algoritmo>.<proprietà> = <valore>
```

```
set :autoclusternodo.<algoritmo>.<proprietà> = <valore>
```

Per esempio:

```
set :autotassificatore.neuralnetwork.method = MultilayerPerceptron
```

I nomi degli algoritmi per il nodo Classificatore automatico sono cart, chaid, quest, c50, logreg, decisionlist, bayesnet, discriminant, svm e knn.

I nomi degli algoritmi per il nodo Numerico automatico sono cart, chaid, neuralnetwork, genlin, svm, regression, linear e knn.

I nomi degli algoritmi per il nodo Cluster automatico sono twostep, k-means e kohonen.

I nomi delle proprietà sono standard, come documentato per i nodi dei singoli algoritmi.

Le proprietà degli algoritmi che contengono punti o altri tipi di punteggiatura devono essere racchiuse tra virgolette singole, per esempio:

```
set :autoclassifiernode.logreg.tolerance = '1.0E-5'
```

Come proprietà è possibile assegnare anche valori multipli, per esempio:

```
set :autoclassifiernode.decisionlist.search_direction = [Up Down]
```

Per attivare o disattivare l'utilizzo di un algoritmo specifico:

```
set :autoclassifiernode.chaid=true
```

Note:

- Quando si impostano i valori true e false è necessario utilizzare lettere minuscole (anziché False).
- Nei casi in cui determinate opzioni di algoritmi non siano disponibili nel nodo Classificatore automatico o quando è possibile specificare un solo valore anziché un intervallo di valori, per gli script si applicano gli stessi limiti validi per l'accesso al nodo con la normale procedura.

Proprietà *autoclusternode*



Il nodo Cluster automatico stima e confronta i modelli di cluster che identificano gruppi di record con caratteristiche simili. Il nodo funziona in modo analogo ad altri nodi Modelli automatici e consente di sperimentare varie combinazioni di opzioni in un singolo passaggio di modellazione. I modelli si possono confrontare utilizzando misure di base con cui tentare di filtrare e classificare l'utilità dei modelli di cluster e fornire una misura in base all'importanza di determinati campi. [Per ulteriori informazioni, vedere l'argomento Nodo Cluster automatico in il capitolo 5 in *IBM SPSS Modeler 14.2 Nodi Modelli*.](#)

Esempio

```
create autoclusternode  
set :autoclusternode.ranking_measure=Silhouette  
set :autoclusternode.ranking_dataset=Training  
set :autoclusternode.enable_silhouette_limit=true
```

```
set :autoclusternode.silhouette_limit=5
```

| Proprietà autoclusternode | Valori | Descrizione proprietà |
|-----------------------------------|--|--|
| evaluation | <i>campo</i> | <i>Nota:</i> solo nodo Cluster automatico. Identifica il campo per cui verrà calcolato un valore di importanza. In alternativa, può essere utilizzato per identificare il modo in cui il cluster distingue il valore di questo campo e, pertanto, la validità della previsione di questo campo da parte del modello. |
| ranking_measure | Silhouette Num_clusters Size_smallest_cluster Size_largest_cluster Smallest_to_largest Importance | |
| ranking_dataset | Training Test | |
| summary_limit | <i>intero</i> | Numero dei modelli da elencare nel report. Specificare un numero intero compreso fra 1 e 100. |
| enable_silhouette_limit | <i>flag</i> | |
| silhouette_limit | <i>intero</i> | Numero intero compreso fra 0 e 100. |
| enable_number_less_limit | <i>flag</i> | |
| number_less_limit | <i>numero</i> | Numero reale compreso tra 0.0 e 1.0. |
| enable_number_greater_limit | <i>flag</i> | |
| number_greater_limit | <i>numero</i> | Numero intero maggiore di 0. |
| enable_smallest_cluster_limit | <i>flag</i> | |
| smallest_cluster_units | Percentage Counts | |
| smallest_cluster_limit_percentage | <i>numero</i> | |
| smallest_cluster_limit_count | <i>intero</i> | Numero intero maggiore di 0. |
| enable_largest_cluster_limit | <i>flag</i> | |
| largest_cluster_units | Percentage Counts | |
| largest_cluster_limit_percentage | <i>numero</i> | |
| largest_cluster_limit_count | <i>intero</i> | |
| enable_smallest_largest_limit | <i>flag</i> | |
| smallest_largest_limit | <i>numero</i> | |
| enable_importance_limit | <i>flag</i> | |
| importance_limit_condition | Greater_than Less_than | |
| importance_limit_greater_than | <i>numero</i> | Numero intero compreso fra 0 e 100. |
| importance_limit_less_than | <i>numero</i> | Numero intero compreso fra 0 e 100. |

| Proprietà autoclusternode | Valori | Descrizione proprietà |
|---------------------------|----------------|---|
| <algorithm> | <i>flag</i> | Attiva o disattiva l'utilizzo di un algoritmo specifico, ad esempio: set :autoclusternode.kohonen=true |
| <algorithm>.<property> | <i>stringa</i> | Imposta il valore di una proprietà di un algoritmo specifico. Per ulteriori informazioni, vedere l'argomento Impostazione delle proprietà degli algoritmi a pag. 194. |

Proprietà autonumericnode



Il nodo Numerico automatico stima e confronta i modelli per i risultati di intervalli numerici continui utilizzando svariati metodi. Il nodo funziona in modo analogo al nodo Classificatore automatico e consente di scegliere gli algoritmi da utilizzare e di sperimentare più combinazioni di opzioni in un singolo passaggio di modellazione. Gli algoritmi supportati includono reti neurali, C&R Tree, CHAID, regressione lineare, regressione lineare generalizzata e SVM (Support Vector Machine). I modelli si possono confrontare in base a correlazione, errore relativo o numero di variabili utilizzato. [Per ulteriori informazioni, vedere l'argomento Nodo Numerico automatico in il capitolo 5 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create autonumericnode
set :autonumericnode.ranking_measure=Correlation
set :autonumericnode.ranking_dataset=Training
set :autonumericnode.enable_correlation_limit=true
set :autonumericnode.correlation_limit=0.8
set :autonumericnode.calculate_variable_importance=true
set :autonumericnode.neuralnetwork=true
set :autonumericnode.chaid=false
```

| Proprietà autonumericnode | Valori | Descrizione proprietà |
|---------------------------|----------------------------|---|
| custom_fields | <i>flag</i> | Se True (vero), saranno utilizzate le impostazioni personalizzate dei campi anziché le impostazioni del nodo Tipo. |
| target | <i>campo</i> | Il nodo Numerico automatico richiede un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare i campi frequenza e peso. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| inputs | <i>[campo1 ... campo2]</i> | |
| partition | <i>campo</i> | |
| use_frequency | <i>flag</i> | |
| frequency_field | <i>campo</i> | |
| use_weight | <i>flag</i> | |
| weight_field | <i>campo</i> | |

| Proprietà autonumericnode | Valori | Descrizione proprietà |
|-------------------------------|-------------------------------|---|
| use_partitioned_data | <i>flag</i> | Se è definito un campo di partizione, per la creazione del modello verranno utilizzati solo i dati di addestramento. |
| ranking_measure | Correlation NumberOfFields | |
| ranking_dataset | Test Training | |
| number_of_models | <i>intero</i> | Numero dei modelli da includere nell'insieme di modelli. Specificare un numero intero compreso fra 1 e 100. |
| calculate_variable_importance | <i>flag</i> | |
| enable_correlation_limit | <i>flag</i> | |
| correlation_limit | <i>intero</i> | |
| enable_number_of_fields_limit | <i>flag</i> | |
| number_of_fields_limit | <i>intero</i> | |
| enable_relative_error_limit | <i>flag</i> | |
| relative_error_limit | <i>intero</i> | |
| enable_model_build_time_limit | <i>flag</i> | |
| model_build_time_limit | <i>intero</i> | |
| enable_stop_after_time_limit | <i>flag</i> | |
| stop_after_time_limit | <i>intero</i> | |
| stop_if_valid_model | <i>flag</i> | |
| <algorithm> | <i>flag</i> | Attiva o disattiva l'utilizzo di un algoritmo specifico, ad esempio: set :autonumericnode.chaid=true |
| <algorithm>.<property> | <i>stringa</i> | Imposta il valore di una proprietà di un algoritmo specifico. Per ulteriori informazioni, vedere l'argomento Impostazione delle proprietà degli algoritmi a pag. 194. |

Proprietà bayesnetnode



Il nodo Rete bayesiana consente di generare un modello di probabilità combinando elementi osservati e registrati con conoscenze del mondo reale per stabilire la probabilità di occorrenze. Il nodo si concentra sulle reti TAN (Tree Augmented Naïve Bayes) e Markov Blanket, che sono prevalentemente utilizzate a scopo di classificazione. [Per ulteriori informazioni, vedere l'argomento Nodo Rete bayesiana in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create bayesnetnode
set :bayesnetnode.continue_training_existing_model = True
set :bayesnetnode.structure_type = MarkovBlanket
set :bayesnetnode.use_feature_selection = True
# Expert tab
set :bayesnetnode.mode = Expert
set :bayesnetnode.all_probabilities = True
```

set :bayesnetnode.independence = Pearson

| Proprietà bayesnetnode | Valori | Descrizione proprietà |
|----------------------------------|----------------------------|---|
| inputs | <i>[campo1 ... campoN]</i> | I modelli di rete bayesiana utilizzano un solo campo obiettivo e uno o più campi di input. I campi continui vengono automaticamente discretizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue_training_existing_model | <i>flag</i> | |
| structure_type | TAN MarkovBlanket | Selezionare la struttura da utilizzare per la creazione della rete bayesiana. |
| use_feature_selection | <i>flag</i> | |
| parameter_learning_method | Likelihood Bayes | Specifica il metodo utilizzato per stimare le tabelle di probabilità condizionale tra i nodi in cui i valori dei nodi genitore sono noti. |
| mode | Expert Simple | |
| missing_values | <i>flag</i> | |
| all_probabilities | <i>flag</i> | |
| independence | Likelihood Pearson | Specifica il metodo utilizzato per determinare se le osservazioni accoppiate su due variabili sono indipendenti tra loro. |
| significance_level | <i>numero</i> | Specifica il valore di interruzione per determinare l'indipendenza. |
| maximal_conditioning_set | <i>numero</i> | Imposta il numero massimo di variabili di condizionamento da utilizzare per il test dell'indipendenza. |
| inputs_always_selected | <i>[campo1 ... campoN]</i> | Specifica quali campi dell'insieme di dati devono sempre essere utilizzati nella creazione della rete bayesiana. <i>Nota:</i> il campo obiettivo è sempre selezionato. |
| maximum_number_inputs | <i>numero</i> | Specifica il numero massimo di campi di input da utilizzare nella creazione della rete bayesiana. |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà c50node



Il nodo C5.0 crea un albero decisionale o un insieme di regole. Il modello suddivide il campione in base al campo che fornisce il massimo guadagno di informazioni a ogni livello. Il campo obiettivo deve essere categoriale. Sono consentite suddivisioni multiple in più di due sottogruppi. [Per ulteriori informazioni, vedere l'argomento nodo C5.0 in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create c50node
# "Model" tab
set :c50node.use_model_name = False
set :c50node.model_name = "C5_Drug"
set :c50node.use_partitioned_data = True
set :c50node.output_type = DecisionTree
set :c50node.use_xval = True
set :c50node.xval_num_folds = 3
set :c50node.mode = Expert
set :c50node.favor = Generality
set :c50node.min_child_records = 3
# "Costs" tab
set :c50node.use_costs = True
set :c50node.costs = [{"drugA" "drugX" 2}]
```

| Proprietà c50node | Valori | Descrizione proprietà |
|--------------------|-------------------------|--|
| target | <i>campo</i> | I modelli C50 utilizzano un solo campo obiettivo e uno o più campi di input. È anche possibile specificare un campo peso. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| output_type | DecisionTree RuleSet | |
| group_symbolics | <i>flag</i> | |
| use_boost | <i>flag</i> | |
| boost_num_trials | <i>numero</i> | |
| use_xval | <i>flag</i> | |
| xval_num_folds | <i>numero</i> | |
| mode | Simple Expert | |
| favor | Accuracy Generality | Precisione o generalità della preferenza. |
| expected_noise | <i>numero</i> | |
| min_child_records | <i>numero</i> | |
| pruning_severity | <i>numero</i> | |
| use_costs | <i>flag</i> | |
| costs | <i>strutturato</i> | Si tratta di una proprietà strutturata. |
| use_winning | <i>flag</i> | |
| use_global_pruning | <i>flag</i> | Impostato su (True) (vero) per default. |

| Proprietà c50node | Valori | Descrizione proprietà |
|---------------------------------|--------------------|-----------------------|
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà carmanode



Il modello CARMA estrae un insieme di regole dai dati senza che venga richiesto all'utente di specificare i campi di input o obiettivo. A differenza di Apriori il nodo CARMA fornisce le impostazioni di creazione per il supporto delle regole (sia per l'antecedente che per il conseguente) anziché solo per il supporto antecedente. Pertanto, le regole generate possono essere utilizzate per una gamma più vasta di applicazioni, per esempio per trovare un elenco di prodotti o di servizi (antecedenti) il cui conseguente è rappresentato dall'articolo che si desidera promuovere per le festività correnti. [Per ulteriori informazioni, vedere l'argomento Nodo CARMA in il capitolo 12 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create carmanode
# "Fields" tab
set :carmanode.custom_fields = True
set :carmanode.use_transactional_data = True
set :carmanode.inputs = ['BP' 'Cholesterol' 'Drug']
set :carmanode.partition = Test
# "Model" tab
set :carmanode.use_model_name = False
set :carmanode.model_name = "age_bp_drug"
set :carmanode.use_partitioned_data = False
set :carmanode.min_supp = 10.0
set :carmanode.min_conf = 30.0
set :carmanode.max_size = 5
# Expert Options
set :carmanode.mode = Expert
#help set :carmanode.exclude_simple = True
set :carmanode.use_pruning = True
set :carmanode.pruning_value = 300
set :carmanode.vary_support = True
set :carmanode.estimated_transactions = 30
```

```
set :carmanode.rules_without_antecedents = True
```

| Proprietà carmanode | Valori | Descrizione proprietà |
|---------------------------|-----------------------------|---|
| inputs | <i>[campo1 ... campon]</i> | I modelli CARMA utilizzano un elenco di campi di input, ma nessun campo obiettivo. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| id_field | <i>campo</i> | Campo utilizzato come campo ID per la creazione del modello. |
| contiguous | <i>flag</i> | Utilizzata per specificare se gli ID del campo ID sono contigui. |
| use_transactional_data | <i>flag</i> | |
| content_field | <i>campo</i> | |
| min_supp | <i>numero (percentuale)</i> | Si riferisce al supporto della regola più che al supporto antecedente. L'impostazione di default è 20%. |
| min_conf | <i>numero (percentuale)</i> | L'impostazione di default è 20%. |
| max_size | <i>numero</i> | Il valore di default è 10. |
| mode | Simple Expert | L'impostazione di default è Simple. |
| exclude_multiple | <i>flag</i> | Esclude le regole con più conseguenti. L'impostazione di default è False. |
| use_pruning | <i>flag</i> | L'impostazione di default è False. |
| pruning_value | <i>numero</i> | Il valore di default è 500. |
| vary_support | <i>flag</i> | |
| estimated_transactions | <i>intero</i> | |
| rules_without_antecedents | <i>flag</i> | |

Proprietà cartnode



Il nodo Albero di classificazione e regressione (C&R) genera un albero decisionale che consente di prevedere o classificare osservazioni future. Il metodo utilizza l'esecuzione ricorsiva di partizioni per suddividere i record di addestramento in segmenti riducendo l'impurità ad ogni passaggio. Un nodo dell'albero è considerato "puro" quando il 100% dei casi nel nodo fa parte di una categoria specifica del campo obiettivo. I campi obiettivo e di input possono essere intervalli numerici o categoriali (nominali, ordinali o flag); tutte le suddivisioni sono binarie (solo due sottogruppi). [Per ulteriori informazioni, vedere l'argomento Nodo C&R Tree in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create cartnode
# "Fields" tab
set :cartnode.custom_fields = True
set :cartnode.target = 'Drug'
set :cartnode.inputs = ['Age' 'BP' 'Cholesterol']
# "Build Options" tab, 'Objective' panel
```

```

set :cartnode.model_output_type = InteractiveBuilder
set :cartnode.use_tree_directives = True
set :cartnode.tree_directives = ""Grow Node Index 0 Children 1 2
Grow Node Index 2 Children 3 4""
# "Build Options" tab, 'Basics' panel
set :cartnode.prune_tree = False
set :cartnode.use_std_err_rule = True
set :cartnode.std_err_multiplier = 3.0
set :cartnode.max_surrogates = 7
# "Build Options" tab, 'Stopping Rules' panel
set :cartnode.use_percentage = True
set :cartnode.min_parent_records_pc = 5
set :cartnode.min_child_records_pc = 3
# "Build Options" tab, 'Costs & Priors' panel
set :cartnode.use_costs = True
set :cartnode.costs = [{"drugA" "drugX" 2}]
set :cartnode.priors = Custom
# custom priors must add to 1
set :cartnode.custom_priors = [{"drugA" 0.3}{drugX 0.7}]
set :cartnode.adjust_priors = True
# "Build Options" tab, 'Advanced' panel
set :cartnode.min_impurity = 0.0003
set :cartnode.impurity_measure = Twoing
# "Model Options" tab
set :cartnode.use_model_name = False
set :cartnode.model_name = "Cart_Drug"

```

| Proprietà cartnode | Valori | Descrizione proprietà |
|----------------------------------|--|--|
| target | <i>campo</i> | I modelli C&R Tree richiedono un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare un campo frequenza. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue_training_existing_model | <i>flag</i> | |
| objective | Standard Boosting Bagging psm | psm viene utilizzato per insiemi di dati di grandi dimensioni e richiede una connessione Server. |
| model_output_type | Single InteractiveBuilder | |
| use_tree_directives | <i>flag</i> | |

| Proprietà cartnode | Valori | Descrizione proprietà |
|------------------------|--|---|
| tree_directives | <i>stringa</i> | Specifica le direttive per l'ingrandimento di alberi. È possibile racchiudere le direttive tra virgolette triple per evitare il carattere escape in nuove righe o virgolette. Si noti che le direttive possono essere molto sensibili a piccole modifiche nelle opzioni di modellazione e nei dati e possono non essere generalizzate ad altri insiemi di dati. Per ulteriori informazioni, vedere l'argomento Direttive di ingrandimento dell'albero in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi Modelli. |
| use_max_depth | Default Custom | |
| max_depth | <i>intero</i> | Profondità massima dell'albero, da 0 a 1000. Valore utilizzato solo se use_max_depth = Custom. |
| prune_tree | <i>flag</i> | Taglia albero per evitare sovradattamento. |
| use_std_err | <i>flag</i> | Utilizza differenza massima di rischio (in errori standard). |
| std_err_multiplier | <i>numero</i> | Differenza massima. |
| max_surrogates | <i>numero</i> | Numero massimo surrogati. |
| use_percentage | <i>flag</i> | |
| min_parent_records_pc | <i>numero</i> | |
| min_child_records_pc | <i>numero</i> | |
| min_parent_records_abs | <i>numero</i> | |
| min_child_records_abs | <i>numero</i> | |
| use_costs | <i>flag</i> | |
| costs | <i>strutturato</i> | Proprietà strutturata che utilizza il formato: <code>{drugA drugB 1.5} {drugA drugC 2.1}</code> , dove gli argomenti racchiusi tra parentesi graffe ({}) sono i costi previsti effettivi. |
| priors | Data Equal Custom | |
| custom_priors | <i>strutturato</i> | Proprietà strutturata che utilizza il formato: <code>set :cartnode. custom_priors = [{ drugA 0.3 } { drugB 0.6 }]</code> |
| adjust_priors | <i>flag</i> | |
| trails | <i>numero</i> | Numero di modelli di componenti per boosting o bagging. |
| set_ensemble_method | Voting HighestProbability HighestMeanProbability | Regola di combinazione di default per obiettivi categoriali. |
| range_ensemble_method | Mean Median | Regola di combinazione di default per obiettivi continui. |

| Proprietà cartnode | Valori | Descrizione proprietà |
|---------------------------------|---------------------------|---|
| large_boost | <i>flag</i> | Applica il boosting a insiemi di dati di grandi dimensioni. |
| min_impurity | <i>numero</i> | |
| impurity_measure | Gini Twoing Ordered | |
| train_pct | <i>numero</i> | Insieme di prevenzione del sovradattamento. |
| set_random_seed | <i>flag</i> | Opzione Replica risultati. |
| seed | <i>numero</i> | |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà chaidnode



Il nodo CHAID genera un albero decisionale utilizzando statistiche chi-quadrato per identificare suddivisioni ottimali. A differenza dei nodi C&R Tree e QUEST, CHAID può generare alberi non binari e pertanto alcune suddivisioni possono avere più di due rami. I campi obiettivo e di input possono essere intervallo numerico (continui) o categoriali. Un CHAID completo è una modificazione di CHAID che esegue operazioni avanzate per l'analisi di tutte le suddivisioni possibili, ma richiede tempi di elaborazione maggiori. [Per ulteriori informazioni, vedere l'argomento Nodo CHAID in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create chaidnode
set :chaidnode.custom_fields = True
set :chaidnode.target = Drug
set :chaidnode.inputs = [Age Na K Cholesterol BP]
set :chaidnode.use_model_name = true
set :chaidnode.model_name = "CHAID"
set :chaidnode.method = Chaid
set :chaidnode.model_output_type = InteractiveBuilder
set :chaidnode.use_tree_directives = True
set :chaidnode.tree_directives = "Test"
set :chaidnode.mode = Expert
set :chaidnode.split_alpha = 0.03
set :chaidnode.merge_alpha = 0.04
set :chaidnode.chi_square = Pearson
set :chaidnode.use_percentage = True
set :chaidnode.min_parent_records_abs = 40
set :chaidnode.min_child_records_abs = 30
set :chaidnode.epsilon = 0.003
set :chaidnode.max_iterations = 75
set :chaidnode.split_merged_categories = true
```

```
set :chaidnode.bonferroni_adjustment = true
```

| Proprietà chaidnode | Valori | Descrizione proprietà |
|----------------------------------|--|---|
| target | <i>campo</i> | I modelli CHAID richiedono un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare un campo frequenza. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue_training_existing_model | <i>flag</i> | |
| objective | Standard Boosting Bagging psm | psm viene utilizzato per insiemi di dati di grandi dimensioni e richiede una connessione Server. |
| model_output_type | Single InteractiveBuilder | |
| use_tree_directives | <i>flag</i> | |
| tree_directives | <i>stringa</i> | |
| method | Chaid ExhaustiveChaid | |
| use_max_depth | Default Custom | |
| max_depth | <i>intero</i> | Profondità massima dell'albero, da 0 a 1000. Valore utilizzato solo se use_max_depth = Custom. |
| use_percentage | <i>flag</i> | |
| min_parent_records_pc | <i>numero</i> | |
| min_child_records_pc | <i>numero</i> | |
| min_parent_records_abs | <i>numero</i> | |
| min_child_records_abs | <i>numero</i> | |
| use_costs | <i>flag</i> | |
| costs | <i>strutturato</i> | Proprietà strutturata che utilizza il formato: {{drugA drugB 1.5} {drugA drugC 2.1}}, dove gli argomenti racchiusi tra parentesi graffe ({}) sono i costi previsti effettivi. |
| trails | <i>numero</i> | Numero di modelli di componenti per boosting o bagging. |
| set_ensemble_method | Voting HighestProbability HighestMeanProbability | Regola di combinazione di default per obiettivi categoriali. |
| range_ensemble_method | Mean Median | Regola di combinazione di default per obiettivi continui. |
| large_boost | <i>flag</i> | Applica il boosting a insiemi di dati di grandi dimensioni. |
| split_alpha | <i>numero</i> | Livello di significatività per suddivisione. |
| merge_alpha | <i>numero</i> | Livello di significatività per unione. |

| Proprietà chaidnode | Valori | Descrizione proprietà |
|---------------------------------|--------------------|---|
| bonferroni_adjustment | <i>flag</i> | Adegua valori di significatività tramite il metodo di Bonferroni. |
| split_merged_categories | <i>flag</i> | Consenti risuddivisione di categorie unite. |
| chi_square | Pearson LR | Metodo utilizzato per calcolare la statistica chi-quadrato: Pearson o Rapporto di verosimiglianza |
| epsilon | <i>numero</i> | Modifica minima nelle frequenze di cella previste. |
| max_iterations | <i>numero</i> | Numero massimo di iterazioni per la convergenza. |
| set_random_seed | <i>intero</i> | |
| seed | <i>numero</i> | |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |
| maximum_number_of_models | <i>intero</i> | |

Proprietà coxregnode



Il nodo Regressione di Cox consente di generare un modello di sopravvivenza per i dati della relazione tempo-evento in presenza di record censurati. Il modello produce una funzione di sopravvivenza che prevede la probabilità che l'evento di interesse sia verificato a una determinata ora (t) per i valori dati delle variabili di input. [Per ulteriori informazioni, vedere l'argomento Nodo Cox in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create coxregnode
set :coxregnode.survival_time = tenure
set :coxregnode.method = BackwardsStepwise
# Expert tab
set :coxregnode.mode = Expert
set :coxregnode.removal_criterion = Conditional
set :coxregnode.survival = True
```

| Proprietà coxregnode | Valori | Descrizione proprietà |
|----------------------|--------------|---|
| survival_time | <i>campo</i> | I modelli di regressione di Cox richiedono un solo campo contenente i tempi di sopravvivenza. |
| target | <i>campo</i> | I modelli di regressione di Cox richiedono un solo campo obiettivo e uno o più campi di input. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |

| Proprietà coxregnode | Valori | Descrizione proprietà |
|----------------------|---|---|
| method | Enter Stepwise BackwardsStepwise | |
| groups | <i>campo</i> | |
| model_type | MainEffects Custom | |
| custom_terms | [<i>"BP*Sex" "BP*Age"</i>] | Esempio: set :coxregnode. custom_terms = [<i>"BP*Sex" "BP" "Age"</i>] |
| mode | Expert Simple | |
| max_iterations | <i>numero</i> | |
| p_converge | 1.0E-4 1.0E-5 1.0E-6 1.0E-7 1.0E-8 0 | |
| p_converge | 1.0E-4 1.0E-5 1.0E-6 1.0E-7 1.0E-8 0 | |
| l_converge | 1.0E-1 1.0E-2 1.0E-3 1.0E-4 1.0E-5 0 | |
| removal_criterion | LR Wald Conditional | |
| probability_entry | <i>numero</i> | |
| probability_removal | <i>numero</i> | |
| output_display | EachStep LastStep | |
| ci_enable | <i>flag</i> | |
| ci_value | 90 95 99 | |
| correlation | <i>flag</i> | |
| display_baseline | <i>flag</i> | |
| survival | <i>flag</i> | |
| hazard | <i>flag</i> | |
| log_minus_log | <i>flag</i> | |
| one_minus_survival | <i>flag</i> | |

| Proprietà coxregnode | Valori | Descrizione proprietà |
|----------------------|--------------------------------|---|
| separate_line | <i>campo</i> | |
| value | <i>numero</i> o <i>stringa</i> | Se per un campo non viene specificato alcun valore sarà utilizzata l'opzione di default "Media". Utilizzo per un campo numerico: coxnode.value = [{"age" "35.8"}] Utilizzo per un campo categoriale: coxnode.value = [{"color" "pink"}] |

Proprietà decisionlistnode



Il nodo Elenco decisionale identifica i sottogruppi o i segmenti che mostrano una probabilità maggiore o minore che si verifichi un determinato risultato binario rispetto alla popolazione globale. Per esempio, è possibile che si cerchino i clienti non a rischio di abbandono o quelli che più probabilmente rispondano in modo favorevole a una campagna. È possibile incorporare le proprie conoscenze aziendali nel modello aggiungendo propri segmenti personalizzati e visualizzando in anteprima modelli alternativi uno accanto all'altro per confrontarne i risultati. I modelli Elenco decisionale consistono in un elenco di regole in cui ogni regola ha una condizione e un risultato. Le regole vengono applicate in ordine e la prima regola corrispondente determina il risultato. [Per ulteriori informazioni, vedere l'argomento Elenco decisionale in il capitolo 9 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create decisionlistnode
set :decisionlistnode.search_direction=Down
set :decisionlistnode.target_value=1
set :decisionlistnode.max_rules=4
set :decisionlistnode.min_group_size_pct = 15
```

| Proprietà decisionlistnode | Valori | Descrizione proprietà |
|----------------------------|-----------------------------|---|
| target | <i>campo</i> | I modelli Elenco decisionale utilizzano un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare un campo frequenza. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| model_output_type | Model InteractiveBuilder | |
| search_direction | Up Down | Si riferisce alla ricerca di segmenti, dove Up è l'equivalente di Probabilità elevata e Down è l'equivalente di Probabilità bassa. |
| target_value | <i>stringa</i> | Se non specificata, presuppone il valore vero per i flag. |
| max_rules | <i>intero</i> | Il numero massimo di segmenti escluso il resto. |
| min_group_size | <i>intero</i> | Dimensione minima del segmento. |
| min_group_size_pct | <i>numero</i> | Dimensioni minime del segmento espresse come percentuale. |

| Proprietà decisionlistnode | Valori | Descrizione proprietà |
|---------------------------------|--------------------------|--|
| confidence_level | numero | Soglia minima di cui un campo di input deve migliorare la probabilità di risposta (lift) perché valga la pena aggiungerlo alla definizione di un segmento. |
| max_segments_per_rule | intero | |
| mode | Simple Expert | |
| bin_method | EqualWidth EqualCount | |
| bin_count | numero | |
| max_models_per_cycle | intero | Larghezza di ricerca per gli elenchi. |
| max_rules_per_cycle | intero | Larghezza di ricerca per le regole dei segmenti. |
| segment_growth | numero | |
| include_missing | flag | |
| final_results_only | flag | |
| reuse_fields | flag | Consente il riutilizzo degli attributi (campi di input che compaiono nelle regole). |
| max_alternatives | intero | |
| calculate_raw_propensities | flag | |
| calculate_adjusted_propensities | flag | |
| adjusted_propensity_partition | Test Validation | |

Proprietà discriminantnode



L'analisi discriminante prevede presupposti più rigidi rispetto alla regressione logistica, ma può essere una valida alternativa o un complemento dell'analisi di regressione logistica quando vengono soddisfatti tali presupposti. [Per ulteriori informazioni, vedere l'argomento Nodo Discriminante in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create discriminantnode
set :discriminantnode.target = custcat
set :discriminantnode.use_partitioned_data = False
```

set :discriminantnode.method = Stepwise

| Proprietà discriminantnode | Valori | Descrizione proprietà |
|----------------------------|---|---|
| target | <i>campo</i> | I modelli Discriminante richiedono un solo campo obiettivo e uno o più campi di input. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| method | Enter Stepwise | |
| mode | Simple Expert | |
| prior_probabilities | AllEqual ComputeFromSizes | |
| covariance_matrix | WithinGroups SeparateGroups | |
| means | <i>flag</i> | Opzioni relative alle statistiche nella finestra di dialogo Output avanzato. |
| univariate_anovas | <i>flag</i> | |
| box_m | <i>flag</i> | |
| within_group_covariance | <i>flag</i> | |
| within_groups_correlation | <i>flag</i> | |
| separate_groups_covariance | <i>flag</i> | |
| total_covariance | <i>flag</i> | |
| fishers | <i>flag</i> | |
| unstandardized | <i>flag</i> | |
| casewise_results | <i>flag</i> | Opzioni relative alla classificazione nella finestra di dialogo Output avanzato. |
| limit_to_first | <i>numero</i> | Il valore di default è 10. |
| summary_table | <i>flag</i> | |
| leave_one_classification | <i>flag</i> | |
| combined_groups | <i>flag</i> | |
| separate_groups_covariance | <i>flag</i> | Opzione delle matrici Covarianza per gruppi separati. |
| territorial_map | <i>flag</i> | |
| combined_groups | <i>flag</i> | Opzione del nodo Plot Gruppi combinati. |
| separate_groups | <i>flag</i> | Opzione del nodo Plot Gruppi separati. |
| summary_of_steps | <i>flag</i> | |
| F_pairwise | <i>flag</i> | |
| stepwise_method | WilksLambda UnexplainedVariance MahalanobisDistance SmallestF RaosV | |
| V_to_enter | <i>numero</i> | |
| criteria | UseValue UseProbability | |

| Proprietà discriminantnode | Valori | Descrizione proprietà |
|---------------------------------|--------------------|------------------------------|
| F_value_entry | numero | Il valore di default è 3,84. |
| F_value_removal | numero | Il valore di default è 2,71. |
| probability_entry | numero | Il valore di default è 0,05. |
| probability_removal | numero | Il valore di default è 0,10. |
| calculate_variable_importance | flag | |
| calculate_raw_propensities | flag | |
| calculate_adjusted_propensities | flag | |
| adjusted_propensity_partition | Test Validation | |

Proprietà factornode



Il nodo Fattoriale offre potenti tecniche di riduzione dei dati che consentono di diminuirne la complessità. L'analisi dei componenti principali (PCA, Principal Components Analysis) trova le combinazioni lineari dei campi di input che catturano meglio la varianza nell'intero insieme di campi, dove i componenti sono ortogonali (perpendicolari) l'uno rispetto all'altro. L'analisi fattoriale tenta di identificare i concetti sottostanti, o fattori, che spiegano lo schema delle correlazioni all'interno dell'insieme di campi osservati. Entrambi gli approcci mirano a trovare un numero ridotto di campi derivati che riassumono in modo efficace le informazioni presenti nell'insieme originale di campi. [Per ulteriori informazioni, vedere l'argomento Nodo Fattoriale in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create factornode
# "Fields" tab
set :factornode.custom_fields = True
set :factornode.inputs = ['BP' 'Na' 'K']
set :factornode.partition = Test
# "Model" tab
set :factornode.use_model_name = True
set :factornode.model_name = "Factor_Age"
set :factornode.use_partitioned_data = False
set :factornode.method = GLS
# Expert options
set :factornode.mode = Expert
set :factornode.complete_records = true
set :factornode.matrix = Covariance
set :factornode.max_iterations = 30
set :factornode.extract_factors = ByFactors
set :factornode.min_eigenvalue = 3.0
set :factornode.max_factor = 7
set :factornode.sort_values = True
set :factornode.hide_values = True
set :factornode.hide_below = 0.7
# "Rotation" section
set :factornode.rotation = DirectOblimin
set :factornode.delta = 0.3
```

set :factornode.kappa = 7.0

| Proprietà factornode | Valori | Descrizione proprietà |
|----------------------|--|--|
| inputs | [<i>campo1 ... campoN</i>] | I modelli Fattoriale utilizzano un elenco di campi di input, ma nessun campo obiettivo. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| method | PC ULS GLS ML PAF Alpha Image | |
| mode | Simple Expert | |
| max_iterations | <i>numero</i> | |
| complete_records | <i>flag</i> | |
| matrix | Correlation Covariance | |
| extract_factors | ByEigenvalues ByFactors | |
| min_eigenvalue | <i>numero</i> | |
| max_factor | <i>numero</i> | |
| rotation | None Varimax DirectOblimin Equamax Quartimax Promax | |
| delta | <i>numero</i> | Se si seleziona DirectOblimin come tipo di dati di rotazione, è possibile specificare un valore per delta. Se non si specifica un valore, per delta verrà utilizzato il valore di default. |
| kappa | <i>numero</i> | Se si seleziona Promax come tipo di dati di rotazione, è possibile specificare un valore per kappa. Se non si specifica un valore, per kappa verrà utilizzato il valore di default. |
| sort_values | <i>flag</i> | |
| hide_values | <i>flag</i> | |
| hide_below | <i>numero</i> | |

Proprietà *featureselectionnode*



Il nodo Selezione funzioni effettua lo screening dei campi di input, rimuovendoli in base a un insieme di criteri quali la percentuale di valori mancanti. Classifica quindi gli input restanti in ordine di importanza rispetto a un determinato obiettivo. Per esempio, dato un insieme di dati con centinaia di input potenziali, quali sono quelli con la maggiore probabilità di essere utili nella modellazione di risultati clinici? [Per ulteriori informazioni, vedere l'argomento Nodo Selezione funzioni in il capitolo 4 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create featureselectionnode
set :featureselectionnode.screen_single_category=true
set :featureselectionnode.max_single_category=95
set :featureselectionnode.screen_missing_values=true
set :featureselectionnode.max_missing_values=80
set :featureselectionnode.criteria = Likelihood
set :featureselectionnode.unimportant_below = 0.8
set :featureselectionnode.important_above = 0.9
set :featureselectionnode.important_label = "Check Me Out!"
set :featureselectionnode.selection_mode = TopN
set :featureselectionnode.top_n = 15
```

Per un esempio più dettagliato di creazione e applicazione di un modello di selezione funzioni, vedere [Esempio di script locale: Generazione di un Modello di selezione funzioni in il capitolo 2 a pag. 12.](#)

| Proprietà <i>featureselectionnode</i> | Valori | Descrizione proprietà |
|---------------------------------------|---------------|--|
| target | <i>campo</i> | I modelli di selezione funzioni classificano i predittori rispetto all'obiettivo specificato. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| screen_single_category | <i>flag</i> | Se True, esegue lo screening dei campi che hanno troppi record che rientrano nella stessa categoria rispetto al numero totale di record. |
| max_single_category | <i>numero</i> | Specifica la soglia utilizzata quando <i>screen_single_category</i> è True. |
| screen_missing_values | <i>flag</i> | Se True, esegue lo screening dei campi con troppi valori mancanti, espressi come percentuale del numero totale di record. |
| max_missing_values | <i>numero</i> | |
| screen_num_categories | <i>flag</i> | Se True, esegue lo screening dei campi con troppe categorie rispetto al numero totale di record. |
| max_num_categories | <i>numero</i> | |

| Proprietà featureselectionnode | Valori | Descrizione proprietà |
|--------------------------------|---|---|
| screen_std_dev | <i>flag</i> | Se True, esegue lo screening dei campi con una deviazione standard inferiore o uguale al minimo specificato. |
| min_std_dev | <i>numero</i> | |
| screen_coeff_of_var | <i>flag</i> | Se True, esegue lo screening dei campi con un coefficiente di varianza inferiore o uguale al minimo specificato. |
| min_coeff_of_var | <i>numero</i> | |
| criteria | Pearson Likelihood CramersV Lambda | Quando si classificano i predittori categoriali rispetto a un obiettivo categoriale, specifica la misura sulla quale si basa il valore di importanza. |
| unimportant_below | <i>numero</i> | Specifica i valori p di soglia utilizzati per classificare variabili quali importante, marginale o non importante. Accetta i valori compresi fra 0.0 e 1.0. |
| important_above | <i>numero</i> | Accetta i valori compresi fra 0.0 e 1.0. |
| unimportant_label | <i>stringa</i> | Specifica l'etichetta per la classificazione non importante. |
| marginal_label | <i>stringa</i> | |
| important_label | <i>stringa</i> | |
| selection_mode | ImportanceLevel ImportanceValue TopN | |
| select_important | <i>flag</i> | Quando selection_mode è impostata su ImportanceLevel, specifica se selezionare i campi importanti. |
| select_marginal | <i>flag</i> | Quando selection_mode è impostata su ImportanceLevel, specifica se selezionare i campi marginali. |
| select_unimportant | <i>flag</i> | Quando selection_mode è impostata su ImportanceLevel, specifica se selezionare i campi non importanti. |
| importance_value | <i>numero</i> | Quando selection_mode è impostata su ImportanceValue, specifica il valore di interruzione da utilizzare. Accetta i valori compresi fra 0 e 100. |
| top_n | <i>intero</i> | Quando selection_mode è impostata su TopN, specifica il valore di interruzione da utilizzare. Accetta i valori compresi fra 0 e 1000. |

Proprietà genlinnode



Il modello Lineare generalizzato amplia il modello lineare generale in modo che la variabile dipendente venga linearmente correlata ai fattori e alle covariate tramite una funzione di collegamento specifica. Inoltre, il modello consente alla variabile dipendente di avere una distribuzione non normale. Copre la funzionalità di un grande numero di modelli statistici, inclusi modelli di regressione lineare, modelli di regressione logistica, modelli loglineari per dati dei conteggi e modelli di sopravvivenza censurati per intervallo. [Per ulteriori informazioni, vedere l'argomento Nodo GenLin in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create genlinnode
set :genlinnode.model_type = MainAndAllTwoWayEffects
set :genlinnode.offset_type = Variable
set :genlinnode.offset_field = Claimant
```

| Proprietà genlinnode | Valori | Descrizione proprietà |
|--------------------------|--|--|
| target | <i>campo</i> | I modelli lineari generalizzati richiedono un solo campo obiettivo (che deve essere nominale o flag) e uno o più campi di input. È anche possibile specificare un campo peso. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| use_weight | <i>flag</i> | |
| weight_field | <i>campo</i> | Il tipo di campo è solo continuo. |
| target_represents_trials | <i>flag</i> | |
| trials_type | Variable FixedValue | |
| trials_field | <i>campo</i> | Il tipo di campo è continuo, flag o ordinale. |
| trials_number | <i>numero</i> | Il valore di default è 10. |
| model_type | MainEffects MainAndAllTwoWayEffects | |
| offset_type | Variable FixedValue | |
| offset_field | <i>campo</i> | Il tipo di campo è solo continuo. |
| offset_value | <i>numero</i> | Deve essere un numero reale. |
| base_category | Last First | |
| include_intercept | <i>flag</i> | |
| mode | Simple Expert | |

| Proprietà genlinnode | Valori | Descrizione proprietà |
|-----------------------|---|--|
| distribution | BINOMIAL GAMMA IGAUSS NEGBIN NORMAL POISSON TWEEDIE MULTINOMIAL | IGAUSS: gaussiana inversa. NEGBIN: binomiale negativa. |
| negbin_para_type | Specify Estimate | |
| negbin_parameter | <i>numero</i> | Il valore di default è 1. Deve contenere un numero reale non negativo. |
| tweedie_parameter | <i>numero</i> | |
| link_function | IDENTITY CLOGLOG LOG LOGC LOGIT NEGBIN NLOGLOG ODDSPOWER PROBIT POWER CUMCAUCHIT CUMCLOGLOG CUMLOGIT CUMNLOGLOG CUMPROBIT | CLOGLOG: log-log complementare. LOGC: logaritmica complementare. NEGBIN: binomiale negativa. NLOGLOG: log-log negativa. CUMCAUCHIT: Cauchit cumulativa. CUMCLOGLOG: log-log complementare cumulativa. CUMLOGIT: logit cumulativa. CUMNLOGLOG: log-log negativa cumulativa. CUMPROBIT: probit cumulativa. |
| power | <i>numero</i> | Il valore deve essere un numero reale diverso da zero. |
| method | Hybrid Fisher NewtonRaphson | |
| max_fisher_iterations | <i>numero</i> | Il valore di default è 1; sono consentiti solo numeri interi positivi. |
| scale_method | MaxLikelihoodEstimate Deviance PearsonChiSquare FixedValue | |
| scale_value | <i>numero</i> | Il valore di default è 1; deve essere maggiore di 0. |
| covariance_matrix | ModelEstimator RobustEstimator | |
| max_iterations | <i>numero</i> | Il valore di default è 100; solo numeri interi non negativi. |
| max_step_halving | <i>numero</i> | Il valore di default è 5; solo numeri interi positivi. |
| check_separation | <i>flag</i> | |
| start_iteration | <i>numero</i> | Il valore di default è 20; sono consentiti solo numeri interi positivi. |
| estimates_change | <i>flag</i> | |
| estimates_change_min | <i>numero</i> | Il valore di default è 1E-006; sono consentiti solo numeri positivi. |

| Proprietà genlinnode | Valori | Descrizione proprietà |
|-------------------------------|--|--|
| estimates_change_type | Absolute Relative | |
| loglikelihood_change | <i>flag</i> | |
| loglikelihood_change_min | <i>numero</i> | Sono consentiti solo numeri positivi. |
| loglikelihood_change_type | Absolute Relative | |
| hessian_convergence | <i>flag</i> | |
| hessian_convergence_min | <i>numero</i> | Sono consentiti solo numeri positivi. |
| hessian_convergence_type | Absolute Relative | |
| case_summary | <i>flag</i> | |
| contrast_matrices | <i>flag</i> | |
| descriptive_statistics | <i>flag</i> | |
| estimable_functions | <i>flag</i> | |
| model_info | <i>flag</i> | |
| iteration_history | <i>flag</i> | |
| goodness_of_fit | <i>flag</i> | |
| print_interval | <i>numero</i> | Il valore di default è 1; deve essere un numero intero positivo. |
| model_summary | <i>flag</i> | |
| lagrange_multiplier | <i>flag</i> | |
| parameter_estimates | <i>flag</i> | |
| include_exponential | <i>flag</i> | |
| covariance_estimates | <i>flag</i> | |
| correlation_estimates | <i>flag</i> | |
| analysis_type | TypeI TypeIII TypeIAndTypeIII | |
| statistics | Wald LR | |
| citype | Wald Profile | |
| tolerancelevel | <i>numero</i> | Il valore di default è 0,0001. |
| confidence_interval | <i>numero</i> | Il valore di default è 95. |
| loglikelihood_function | Full Kernel | |
| singularity_tolerance | 1E-007 1E-008 1E-009 1E-010 1E-011 1E-012 | |
| value_order | Ascending Descending DataOrder | |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |

| Proprietà genlinnode | Valori | Descrizione proprietà |
|---------------------------------|--------------------|-----------------------|
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà kmeansnode



Il nodo K-Means raggruppa l'insieme di dati in gruppi distinti (o cluster). Il metodo definisce un numero fisso di cluster, esegue un'assegnazione iterativa dei record ai cluster e modifica i centri dei cluster finché un'ulteriore ridefinizione non consente più un miglioramento del modello. Invece di tentare di prevedere un risultato, il nodo K-Means utilizza un processo denominato apprendimento non supervisionato per scoprire gli schemi nell'insieme di campi di input. [Per ulteriori informazioni, vedere l'argomento Nodo K-Means in il capitolo 11 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create kmeansnode
# "Fields" tab
set :kmeansnode.custom_fields = True
set :kmeansnode.inputs = ['Cholesterol' 'BP' 'Drug' 'Na' 'K' 'Age']
# "Model" tab
set :kmeansnode.use_model_name = False
set :kmeansnode.model_name = "Kmeans_allinputs"
set :kmeansnode.num_clusters = 9
set :kmeansnode.gen_distance = True
set :kmeansnode.cluster_label = "Number"
set :kmeansnode.label_prefix = "Kmeans_"
set :kmeansnode.optimize = Speed
# "Expert" tab
set :kmeansnode.mode = Expert
set :kmeansnode.stop_on = Custom
set :kmeansnode.max_iterations = 10
set :kmeansnode.tolerance = 3.0
set :kmeansnode.encoding_value = 0.3
```

| Proprietà kmeansnode | Valori | Descrizione proprietà |
|----------------------|------------------------------|---|
| inputs | [<i>campo1 ... campoN</i>] | I modelli K-Means eseguono l'analisi dei cluster su un insieme di campi di input ma non utilizzano un campo obiettivo. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| num_clusters | <i>numero</i> | |
| gen_distance | <i>flag</i> | |
| cluster_label | String Number | |
| label_prefix | <i>stringa</i> | |

| Proprietà kmeansnode | Valori | Descrizione proprietà |
|----------------------|-------------------|--|
| mode | Simple Expert | |
| stop_on | Default Custom | |
| max_iterations | <i>numero</i> | |
| tolerance | <i>numero</i> | |
| encoding_value | <i>numero</i> | |
| optimize | Speed Memory | Utilizzare per specificare se ottimizzare la velocità o la memoria durante la creazione del modello. |

Proprietà knnnode



Il nodo Vicini più vicini K (KNN) associa un nuovo caso alla categoria o valore degli oggetti K più vicini ad esso nello spazio di predittori, dove K è un numero intero. I casi simili sono vicini gli uni agli altri, mentre i casi dissimili sono distanti gli uni dagli altri. [Per ulteriori informazioni, vedere l'argomento Nodo KNN in il capitolo 16 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create knnnode
# Objectives tab
set: knnnode.objective = Custom
# Settings tab - Neighbors panel
set: knnnode.automatic_k_selection = false
set: knnnode.fixed_k = 2
set: knnnode.weight_by_importance = True
# Settings tab - Analyze panel
set: knnnode.save_distances = True
```

| Proprietà knnnode | Valori | Descrizione proprietà |
|-----------------------|--|--|
| analysis | PredictTarget IdentifyNeighbors | |
| objective | Balance Speed Accuracy Custom | |
| normalize_ranges | <i>flag</i> | |
| use_case_labels | <i>flag</i> | Selezionare la casella per abilitare l'opzione successiva. |
| case_labels_field | <i>campo</i> | |
| identify_focal_cases | <i>flag</i> | Selezionare la casella per abilitare l'opzione successiva. |
| focal_cases_field | <i>campo</i> | |
| automatic_k_selection | <i>flag</i> | |
| fixed_k | <i>intero</i> | Attivata solo se automatic_k_selectio è False. |

| Proprietà knnnode | Valori | Descrizione proprietà |
|---------------------------------|------------------------------|--|
| minimum_k | <i>intero</i> | Attivata solo se automatic_k_selectio è True. |
| maximum_k | <i>intero</i> | |
| distance_computation | Euclidean CityBlock | |
| weight_by_importance | <i>flag</i> | |
| range_predictions | Mean Median | |
| perform_feature_selection | <i>flag</i> | |
| forced_entry_inputs | [<i>campo1 ... campoN</i>] | |
| stop_on_error_ratio | <i>flag</i> | |
| number_to_select | <i>intero</i> | |
| minimum_change | <i>numero</i> | |
| validation_fold_assign_by_field | <i>flag</i> | |
| number_of_folds | <i>intero</i> | Attivata solo se validation_fold_assign_by_field è False |
| set_random_seed | <i>flag</i> | |
| random_seed | <i>numero</i> | |
| folds_field | <i>campo</i> | Attivata solo se validation_fold_assign_by_field è True |
| all_probabilities | <i>flag</i> | |
| save_distances | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà kohonennode



Il nodo Kohonen genera un tipo di rete neurale che può essere utilizzato per raggruppare l'insieme di dati in gruppi distinti. Al termine dell'apprendimento della rete, i record analoghi dovranno essere vicini nella mappa di output, mentre i record diversi saranno a notevole distanza. Per identificare le unità forti, è possibile controllare il numero di osservazioni catturate da ciascuna unità nell'insieme di modelli. In questo modo è possibile avere un'idea del numero appropriato di cluster. [Per ulteriori informazioni, vedere l'argomento Nodo Kohonen in il capitolo 11 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create kohonennode
# "Model" tab
set :kohonennode.use_model_name = False
set :kohonennode.model_name = "Symbolic Cluster"
set :kohonennode.stop_on = Time
set :kohonennode.time = 1
set :kohonennode.set_random_seed = True
set :kohonennode.random_seed = 12345
set :kohonennode.optimize = Speed
# "Expert" tab
```

```

set :kohonenode.mode = Expert
set :kohonenode.width = 3
set :kohonenode.length = 3
set :kohonenode.decay_style = Exponential
set :kohonenode.phase1_neighborhood = 3
set :kohonenode.phase1_eta = 0.5
set :kohonenode.phase1_cycles = 10
set :kohonenode.phase2_neighborhood = 1
set :kohonenode.phase2_eta = 0.2
set :kohonenode.phase2_cycles = 75

```

| Proprietà kohonenode | Valori | Descrizione proprietà |
|----------------------|-----------------------|---|
| inputs | [campo1 ... campoN] | I modelli Kohonen utilizzano un elenco di campi di input, ma nessun campo obiettivo. I campi frequenza e peso non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue | flag | |
| show_feedback | flag | |
| stop_on | Default Time | |
| time | numero | |
| optimize | Speed Memory | Utilizzare per specificare se ottimizzare la velocità o la memoria durante la creazione del modello. |
| cluster_label | flag | |
| mode | Simple Expert | |
| width | numero | |
| length | numero | |
| decay_style | Linear Exponential | |
| phase1_neighborhood | numero | |
| phase1_eta | numero | |
| phase1_cycles | numero | |
| phase2_neighborhood | numero | |
| phase2_eta | numero | |
| phase2_cycles | numero | |

Proprietà linearnode



I modelli di regressione lineare prevedono un obiettivo continuo basato sulle relazioni lineari tra l'obiettivo e uno o più predittori. [Per ulteriori informazioni, vedere l'argomento Modelli lineari in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```

create linearnode
# Build Options tab - Objectives panel
set: linearnode.objective = Standard
# Build Options tab - Model Selection panel
set: linearnode.model_selection = BestSubsets
set: linearnode.criteria_best_subsets = ASE
# Build Options tab - Ensembles panel
set: linearnode.combining_rule_categorical = HighestMeanProbability

```

| Proprietà linearnode | Valori | Descrizione proprietà |
|----------------------------------|---|--|
| target | <i>campo</i> | Specifica un singolo campo obiettivo. |
| inputs | [<i>campo1 ... campoN</i>] | I campi predittore utilizzati dal modello. |
| continue_training_existing_model | <i>flag</i> | |
| objective | Standard Bagging Boosting psm | psm viene utilizzato per insiemi di dati di grandi dimensioni e richiede una connessione Server. |
| use_auto_data_preparation | <i>flag</i> | |
| confidence_level | <i>numero</i> | |
| model_selection | ForwardStepwise BestSubsets None | |
| criteria_forward_stepwise | AICC Fstatistics AdjustedRSquare ASE | |
| probability_entry | <i>numero</i> | |
| probability_removal | <i>numero</i> | |
| use_max_effects | <i>flag</i> | |
| max_effects | <i>numero</i> | |
| use_max_steps | <i>flag</i> | |
| max_steps | <i>numero</i> | |
| criteria_best_subsets | AICC AdjustedRSquare ASE | |
| combining_rule_continuous | Mean Median | |
| component_models_n | <i>numero</i> | |
| use_random_seed | <i>flag</i> | |
| random_seed | <i>numero</i> | |
| use_custom_model_name | <i>flag</i> | |
| custom_model_name | <i>stringa</i> | |
| use_custom_name | <i>flag</i> | |
| custom_name | <i>stringa</i> | |
| tooltip | <i>stringa</i> | |
| keywords | <i>stringa</i> | |
| annotation | <i>stringa</i> | |

Proprietà logregnode



La regressione logistica, una tecnica statistica che consente di classificare i record in base ai valori dei campi di input, è analoga alla regressione lineare ma, al posto di un intervallo numerico, prende un campo obiettivo categoriale. [Per ulteriori informazioni, vedere l'argomento Nodo Logistica in il capitolo 10 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio multinomiale

```

create logregnode
# "Fields" tab
set :logregnode.custom_fields = True
set :logregnode.target = 'Drug'
set :logregnode.inputs = ['BP' 'Cholesterol' 'Age']
set :logregnode.partition = Test
# "Model" tab
set :logregnode.use_model_name = False
set :logregnode.model_name = "Log_reg Drug"
set :logregnode.use_partitioned_data = True
set :logregnode.method = Stepwise
set :logregnode.logistic_procedure = Multinomial
set :logregnode.multinomial_base_category = BP
set :logregnode.model_type = FullFactorial
set :logregnode.custom_terms = [{BP Sex}{Age}{Na K}]
set :logregnode.include_constant = False
# "Expert" tab
set :logregnode.mode = Expert
set :logregnode.scale = Pearson
set :logregnode.scale_value = 3.0
set :logregnode.all_probabilities = True
set :logregnode.tolerance = "1.0E-7"
# "Convergence..." section
set :logregnode.max_iterations = 50
set :logregnode.max_steps = 3
set :logregnode.l_converge = "1.0E-3"
set :logregnode.p_converge = "1.0E-7"
set :logregnode.delta = 0.03
# "Output..." section
set :logregnode.summary = True
set :logregnode.likelihood_ratio = True
set :logregnode.asymptotic_correlation = True
set :logregnode.goodness_fit = True
set :logregnode.iteration_history = True
set :logregnode.history_steps = 3
set :logregnode.parameters = True
set :logregnode.confidence_interval = 90
set :logregnode.asymptotic_covariance = True
set :logregnode.classification_table = True
# "Stepping" options
set :logregnode.min_terms = 7
set :logregnode.use_max_terms = true

```

```
set :logregnode.max_terms = 10
set :logregnode.probability_entry = 3
set :logregnode.probability_removal = 5
set :logregnode.requirements = Containment
```

Esempio binomiale

```
create logregnode
# "Fields" tab
set :logregnode.custom_fields = True
set :logregnode.target = 'Cholesterol'
set :logregnode.inputs = ['BP' 'Drug' 'Age']
set :logregnode.partition = Test
# "Model" tab
set :logregnode.use_model_name = False
set :logregnode.model_name = "Log_reg Cholesterol"
set :logregnode.multinomial_base_category = BP
set :logregnode.use_partitioned_data = True
set :logregnode.binomial_method = Forwards
set :logregnode.logistic_procedure = Binomial
set :logregnode.binomial_categorical_input = Sex
set :logregnode.binomial_input_contrast.Sex = Simple
set :logregnode.binomial_input_category.Sex = Last
set :logregnode.include_constant = False
# "Expert" tab
set :logregnode.mode = Expert
set :logregnode.scale = Pearson
set :logregnode.scale_value = 3.0
set :logregnode.all_probabilities = True
set :logregnode.tolerance = "1.0E-7"
# "Convergence..." section
set :logregnode.max_iterations = 50
set :logregnode.l_converge = "1.0E-3"
set :logregnode.p_converge = "1.0E-7"
# "Output..." section
set :logregnode.binomial_output_display = at_each_step
set :logregnode.binomial_goodness_fit = True
set :logregnode.binomial_iteration_history = True
set :logregnode.binomial_parameters = True
set :logregnode.binomial_ci_enable = True
set :logregnode.binomial_ci = 85
# "Stepping" options
set :logregnode.binomial_removal_criterion = LR
```

```
set :logregnode.binomial_probability_removal = 0.2
```

| Proprietà logregnode | Valori | Descrizione proprietà |
|----------------------------|---|--|
| target | <i>campo</i> | I modelli di regressione logistica richiedono un solo campo obiettivo e uno o più campi di input. I campi frequenza e peso non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| logistic_procedure | Binomial Multinomial | |
| include_constant | <i>flag</i> | |
| mode | Simple Expert | |
| method | Enter Stepwise Forwards Backwards BackwardsStepwise | |
| binomial_method | Enter Forwards Backwards | |
| model_type | MainEffects FullFactorial Custom | Se FullFactorial è specificato come tipo di modello, i criteri di controllo non verranno eseguiti, anche se sono specificati. Verrà invece utilizzato il metodo Enter. Se il tipo di modello è impostato su Custom, ma non sono stati specificati campi personalizzati, verrà creato un modello effetti principali. |
| custom_terms | <i>[{Pressione Sesso}{Pressione}{Età}]</i> | Esempio: set :logregnode. custom_terms = [Na] K Na K |
| multinomial_base_category | <i>stringa</i> | Specifica come viene determinata la categoria di riferimento. |
| binomial_categorical_input | <i>stringa</i> | |
| binomial_input_contrast | Indicator Simple Difference Helmert Repeated Polynomial Deviation | Proprietà basata su chiavi per input categoriali che indica come viene determinato il confronto. Formato di utilizzo: NODE.binomial_input_contrast.FIELD-NAME |
| binomial_input_category | First Last | Proprietà basata su chiavi per input categoriali che indica come viene determinata la categoria di riferimento. Formato di utilizzo: NODE.binomial_input_category.FIELD-NAME |
| scale | None UserDefined Pearson Deviance | |

| Proprietà logregnode | Valori | Descrizione proprietà |
|------------------------------|---|------------------------------|
| scale_value | <i>numero</i> | |
| all_probabilities | <i>flag</i> | |
| tolerance | 1.0E-5 1.0E-6 1.0E-7 1.0E-8 1.0E-9 1.0E-10 | |
| min_terms | <i>numero</i> | |
| use_max_terms | <i>flag</i> | |
| max_terms | <i>numero</i> | |
| entry_criterion | Score LR | |
| removal_criterion | LR Wald | |
| probability_entry | <i>numero</i> | |
| probability_removal | <i>numero</i> | |
| binomial_probability_entry | <i>numero</i> | |
| binomial_probability_removal | <i>numero</i> | |
| requirements | HierarchyDiscrete HierarchyAll Containment None | |
| max_iterations | <i>numero</i> | |
| max_steps | <i>numero</i> | |
| p_converge | 1.0E-4 1.0E-5 1.0E-6 1.0E-7 1.0E-8 0 | |
| l_converge | 1.0E-1 1.0E-2 1.0E-3 1.0E-4 1.0E-5 0 | |
| delta | <i>numero</i> | |
| iteration_history | <i>flag</i> | |
| history_steps | <i>numero</i> | |
| summary | <i>flag</i> | |
| likelihood_ratio | <i>flag</i> | |
| asymptotic_correlation | <i>flag</i> | |
| goodness_fit | <i>flag</i> | |
| parameters | <i>flag</i> | |
| confidence_interval | <i>numero</i> | |
| asymptotic_covariance | <i>flag</i> | |
| classification_table | <i>flag</i> | |
| stepwise_summary | <i>flag</i> | |

| Proprietà logregnode | Valori | Descrizione proprietà |
|--------------------------------|------------------------------|-----------------------|
| info_criteria | <i>flag</i> | |
| monotonicity_measures | <i>flag</i> | |
| binomial_output_display | at_each_step at_last_step | |
| binomial_goodness_of_fit | <i>flag</i> | |
| binomial_parameters | <i>flag</i> | |
| binomial_iteration_history | <i>flag</i> | |
| binomial_classification_plots | <i>flag</i> | |
| binomial_ci_enable | <i>flag</i> | |
| binomial_ci | <i>numero</i> | |
| binomial_residual | outliers all | |
| binomial_residual_enable | <i>flag</i> | |
| binomial_outlier_threshold | <i>numero</i> | |
| binomial_classification_cutoff | <i>numero</i> | |
| binomial_removal_criterion | LR Wald Conditional | |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |

Proprietà neuralnetnode

Attenzione: in questa versione è disponibile una nuova versione del modello Rete neurale con funzionalità avanzate, descritta nella sezione che segue (*neuralnetwork*). Sebbene sia ancora possibile creare e calcolare il punteggio di un modello con la versione precedente, si consiglia di aggiornare gli script in modo da utilizzare la nuova versione. I dettagli della versione precedente sono riportati a scopo informativo.

Esempio

```
create neuralnetnode
# "Fields" tab
set :neuralnetnode.custom_fields = True
set :neuralnetnode.targets = ['Drug']
set :neuralnetnode.inputs = ['Age' 'Na' 'K' 'Cholesterol' 'BP']
# "Model" tab
set :neuralnetnode.use_partitioned_data = True
set :neuralnetnode.method = Dynamic
set :neuralnetnode.train_pct = 30
set :neuralnetnode.set_random_seed = True
set :neuralnetnode.random_seed = 12345
set :neuralnetnode.stop_on = Time
set :neuralnetnode.accuracy = 95
set :neuralnetnode.cycles = 200
set :neuralnetnode.time = 3
set :neuralnetnode.optimize = Speed
```

```
# "Multiple Method Expert Options" section
set :neuralnetnode.m_topologies = "5 30 5; 2 20 3, 1 10 1"
set :neuralnetnode.m_non_pyramids = False
set :neuralnetnode.m_persistence = 100
```

| Proprietà neuralnetnode | Valori | Descrizione proprietà |
|-------------------------|--|---|
| targets | [<i>campo1</i> ... <i>campoN</i>] | Il nodo Rete neurale richiede uno o più campi obiettivo e uno o più campi di input. I campi frequenza e peso vengono ignorati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| method | Quick Dynamic Multiple Prune ExhaustivePrune RBFN | |
| prevent_overtrain | <i>flag</i> | |
| train_pct | <i>numero</i> | |
| set_random_seed | <i>flag</i> | |
| random_seed | <i>numero</i> | |
| mode | Simple Expert | |
| stop_on | Default Accuracy Cycles Time | Modalità di arresto. |
| accuracy | <i>numero</i> | Precisione di arresto. |
| cycles | <i>numero</i> | Cicli di apprendimento. |
| time | <i>numero</i> | Tempo di addestramento (minuti). |
| continue | <i>flag</i> | |
| show_feedback | <i>flag</i> | |
| binary_encode | <i>flag</i> | |
| use_last_model | <i>flag</i> | |
| gen_logfile | <i>flag</i> | |
| logfile_name | <i>stringa</i> | |
| alpha | <i>numero</i> | |
| initial_eta | <i>numero</i> | |
| high_eta | <i>numero</i> | |
| low_eta | <i>numero</i> | |
| eta_decay_cycles | <i>numero</i> | |
| hid_layers | One Two Three | |
| hl_units_one | <i>numero</i> | |
| hl_units_two | <i>numero</i> | |
| hl_units_three | <i>numero</i> | |

| Proprietà neuralnetnode | Valori | Descrizione proprietà |
|---------------------------------|---------------------|--|
| persistence | <i>numero</i> | |
| m_topologies | <i>stringa</i> | |
| m_non_pyramids | <i>flag</i> | |
| m_persistence | <i>numero</i> | |
| p_hid_layers | One Two Three | |
| p_hl_units_one | <i>numero</i> | |
| p_hl_units_two | <i>numero</i> | |
| p_hl_units_three | <i>numero</i> | |
| p_persistence | <i>numero</i> | |
| p_hid_rate | <i>numero</i> | |
| p_hid_pers | <i>numero</i> | |
| p_inp_rate | <i>numero</i> | |
| p_inp_pers | <i>numero</i> | |
| p_overall_pers | <i>numero</i> | |
| r_persistence | <i>numero</i> | |
| r_num_clusters | <i>numero</i> | |
| r_eta_auto | <i>flag</i> | |
| r_alpha | <i>numero</i> | |
| r_eta | <i>numero</i> | |
| optimize | Speed Memory | Utilizzare per specificare se ottimizzare la velocità o la memoria durante la creazione del modello. |
| calculate_variable_importance | <i>flag</i> | Nota: la proprietà <i>sensitivity_analysis</i> utilizzata nelle versioni precedenti è obsoleta ed è stata sostituita da questa proprietà. La vecchia proprietà è ancora supportata, ma si consiglia di utilizzare <i>calculate_variable_importance</i> . |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà neuralnetworknode



Il nodo Rete neurale utilizza un modello semplificato del modo in cui il cervello umano elabora le informazioni. Funziona simulando un elevato numero di semplici unità di elaborazione interconnesse che assomigliano a versioni astratte di neuroni. Le reti neurali sono potenti strumenti di valutazione delle funzioni generali e richiedono una conoscenza statistica o matematica minima per l'addestramento o l'applicazione.

Esempio

```
create neuralnetworknode
# scheda Opzioni di creazione - riquadro Obiettivi
set: neuralnetworknode.objective = Standard
```

```
# scheda Opzioni di creazione - riquadro Regole di arresto
set: neuralnetworknode.model_selection = BestSubsets
set: neuralnetworknode.criteria_best_subsets = ASE
# scheda Opzioni di creazione - riquadro Risultati classificatore binario
set: neuralnetworknode.combining_rule_categorical = HighestMeanProbability
```

| Proprietà neuralnetworknode | Valori | Descrizione proprietà |
|------------------------------|--|---|
| targets | [campo1 ... campoN] | Specifica i campi obiettivo. |
| inputs | [campo1 ... campoN] | I campi predittore utilizzati dal modello. |
| splits | [campo1 ... campoN] | Specifica il campo o i campi da usare per la creazione di modelli suddivisi. |
| use_partition | flag | Se è definito un campo di partizione, questa opzione garantisce che per la creazione del modello verranno utilizzati solo i dati della partizione di addestramento. |
| continue | flag | Addestramento continuo modello esistente. |
| objective | Standard Bagging Boosting psm | psm viene utilizzato per insiemi di dati di grandi dimensioni e richiede una connessione Server. |
| method | MultilayerPerceptron RadialBasisFunction | |
| use_custom_layers | flag | |
| first_layer_units | numero | |
| second_layer_units | numero | |
| use_max_time | flag | |
| max_time | numero | |
| use_max_cycles | flag | |
| max_cycles | numero | |
| use_min_accuracy | flag | |
| min_accuracy | numero | |
| combining_rule_categorical | Voting HighestProbability HighestMeanProbability | |
| combining_rule_continuous | Mean Median | |
| component_models_n | numero | |
| overfit_prevention_pct | numero | |
| use_random_seed | flag | |
| random_seed | numero | |
| missing_values | listwiseDeletion missingValueImputation | |
| use_custom_model_name | flag | |
| custom_model_name | stringa | |
| confidence | onProbability onIncrease | |
| score_category_probabilities | flag | |

| Proprietà neuralnetworknode | Valori | Descrizione proprietà |
|-----------------------------|----------------|-----------------------|
| max_categories | <i>numero</i> | |
| score_propensity | <i>flag</i> | |
| use_custom_name | <i>flag</i> | |
| custom_name | <i>stringa</i> | |
| tooltip | <i>stringa</i> | |
| keywords | <i>stringa</i> | |
| annotation | <i>stringa</i> | |

Proprietà questnode



Il nodo QUEST offre un metodo di classificazione binario per la creazione di alberi decisionali, progettato per ridurre i tempi di elaborazione necessari per le analisi C&R Tree più complesse, riducendo inoltre la tendenza dei metodi per gli alberi di classificazione a favorire gli input che consentono un numero maggiore di suddivisioni. I campi di input possono essere intervalli numerici (continui), ma il campo obiettivo deve essere categoriale. Tutte le suddivisioni sono binarie. [Per ulteriori informazioni, vedere l'argomento Nodo QUEST in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create questnode
set :questnode.custom_fields = True
set :questnode.target = Drug
set :questnode.inputs = [Age Na K Cholesterol BP]
set :questnode.model_output_type = InteractiveBuilder
set :questnode.use_tree_directives = True
set :questnode.mode = Expert
set :questnode.max_surrogates = 5
set :questnode.split_alpha = 0.03
set :questnode.use_percentage = False
set :questnode.min_parent_records_abs = 40
set :questnode.min_child_records_abs = 30
set :questnode.prune_tree = True
set :questnode.use_std_err = True
set :questnode.std_err_multiplier = 3
set :questnode.priors = Custom
set :questnode.custom_priors = [{drugA 0.3}{drugB 0.4}]
set :questnode.adjust_priors = true
```

| Proprietà questnode | Valori | Descrizione proprietà |
|----------------------------------|--------------|---|
| target | <i>campo</i> | I modelli QUEST richiedono un solo campo obiettivo e uno o più campi di input. È inoltre possibile specificare un campo frequenza. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue_training_existing_model | <i>flag</i> | |

| Proprietà questnode | Valori | Descrizione proprietà |
|------------------------|--|---|
| objective | Standard Boosting Bagging psm | psm viene utilizzato per insiemi di dati di grandi dimensioni e richiede una connessione Server. |
| model_output_type | Single InteractiveBuilder | |
| use_tree_directives | <i>flag</i> | |
| tree_directives | <i>stringa</i> | |
| use_max_depth | Default Custom | |
| max_depth | <i>intero</i> | Profondità massima dell'albero, da 0 a 1000. Valore utilizzato solo se use_max_depth = Custom. |
| prune_tree | <i>flag</i> | Taglia albero per evitare sovradattamento. |
| use_std_err | <i>flag</i> | Utilizza differenza massima di rischio (in errori standard). |
| std_err_multiplier | <i>numero</i> | Differenza massima. |
| max_surrogates | <i>numero</i> | Numero massimo surrogati. |
| use_percentage | <i>flag</i> | |
| min_parent_records_pc | <i>numero</i> | |
| min_child_records_pc | <i>numero</i> | |
| min_parent_records_abs | <i>numero</i> | |
| min_child_records_abs | <i>numero</i> | |
| use_costs | <i>flag</i> | |
| costs | <i>strutturato</i> | Proprietà strutturata che utilizza il formato: {{drugA drugB 1.5} {drugA drugC 2.1}}, dove gli argomenti racchiusi tra parentesi graffe ({}) sono i costi previsti effettivi. |
| priors | Data Equal Custom | |
| custom_priors | <i>strutturato</i> | Proprietà strutturata che utilizza il formato: set :cartnode. custom_priors = [{ drugA 0.3 } { drugB 0.6 }] |
| adjust_priors | <i>flag</i> | |
| trails | <i>numero</i> | Numero di modelli di componenti per boosting o bagging. |
| set_ensemble_method | Voting HighestProbability HighestMeanProbability | Regola di combinazione di default per obiettivi categoriali. |
| range_ensemble_method | Mean Median | Regola di combinazione di default per obiettivi continui. |
| large_boost | <i>flag</i> | Applica il boosting a insiemi di dati di grandi dimensioni. |

| Proprietà questnode | Valori | Descrizione proprietà |
|---------------------------------|--------------------|--|
| split_alpha | <i>numero</i> | Livello di significatività per suddivisione. |
| train_pct | <i>numero</i> | Insieme di prevenzione del sovradattamento. |
| set_random_seed | <i>flag</i> | Opzione Replica risultati. |
| seed | <i>numero</i> | |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà regressionnode



La regressione lineare è una tecnica statistica molto comune per riassumere i dati ed eseguire previsioni individuando un'area o una linea retta in grado di ridurre le discrepanze tra i valori di output previsti e quelli osservati.

Nota: il nodo Regressione verrà sostituito dal nodo Lineare nella prossima versione. Da questo momento si consiglia di utilizzare i modelli lineari per la regressione lineare.

Esempio

```
create regressionnode
# "Fields" tab
set :regressionnode.custom_fields = True
set :regressionnode.target = 'Age'
set :regressionnode.inputs = ['Na' 'K']
set :regressionnode.partition = Test
set :regressionnode.use_weight = True
set :regressionnode.weight_field = 'Drug'
# "Model" tab
set :regressionnode.use_model_name = False
set :regressionnode.model_name = "Regression Age"
set :regressionnode.use_partitioned_data = True
set :regressionnode.method = Stepwise
set :regressionnode.include_constant = False
# "Expert" tab
set :regressionnode.mode = Expert
set :regressionnode.complete_records = False
set :regressionnode.tolerance = "1.0E-3"
# "Stepping..." section
set :regressionnode.stepping_method = Probability
set :regressionnode.probability_entry = 0.77
set :regressionnode.probability_removal = 0.88
set :regressionnode.F_value_entry = 7.0
set :regressionnode.F_value_removal = 8.0
# "Output..." section
```

```

set :regressionnode.model_fit = True
set :regressionnode.r_squared_change = True
set :regressionnode.selection_criteria = True
set :regressionnode.descriptives = True
set :regressionnode.p_correlations = True
set :regressionnode.collinearity_diagnostics = True
set :regressionnode.confidence_interval = True
set :regressionnode.covariance_matrix = True
set :regressionnode.durbin_watson = True

```

| Proprietà regressionnode | Valori | Descrizione proprietà |
|--------------------------|---|---|
| target | <i>campo</i> | I modelli di regressione richiedono un solo campo obiettivo e uno o più campi di input. È anche possibile specificare un campo peso. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| method | Enter Stepwise Backwards Forwards | |
| include_constant | <i>flag</i> | |
| use_weight | <i>flag</i> | |
| weight_field | <i>campo</i> | |
| mode | Simple Expert | |
| complete_records | <i>flag</i> | |
| tolerance | 1.0E-1 1.0E-2 1.0E-3 1.0E-4 1.0E-5 1.0E-6 1.0E-7 1.0E-8 1.0E-9 1.0E-10 1.0E-11 1.0E-12 | Utilizzare le virgolette doppie per gli argomenti. |
| stepping_method | useP useF | useP : usa probabilità di F useF: usa valore di F |
| probability_entry | <i>numero</i> | |
| probability_removal | <i>numero</i> | |
| F_value_entry | <i>numero</i> | |
| F_value_removal | <i>numero</i> | |
| selection_criteria | <i>flag</i> | |
| confidence_interval | <i>flag</i> | |
| covariance_matrix | <i>flag</i> | |
| collinearity_diagnostics | <i>flag</i> | |
| regression_coefficients | <i>flag</i> | |
| exclude_fields | <i>flag</i> | |

| Proprietà regressionnode | Valori | Descrizione proprietà |
|-------------------------------|-------------|-----------------------|
| durbin_watson | <i>flag</i> | |
| model_fit | <i>flag</i> | |
| r_squared_change | <i>flag</i> | |
| p_correlations | <i>flag</i> | |
| descriptives | <i>flag</i> | |
| calculate_variable_importance | <i>flag</i> | |

Proprietà sequencenode



Il nodo Sequenza consente di scoprire le regole di associazione nei dati sequenziali o basati su valori temporali. Per sequenza si intende un elenco di insiemi di elementi che tendono a ricorrere secondo un ordine prevedibile. Ad esempio, un cliente che acquista un rasoio e la lozione dopobarba potrebbe in seguito acquistare la schiuma da barba. Il nodo Sequenza si basa sull'algoritmo delle regole di associazione CARMA, che utilizza un metodo efficiente in due passaggi per trovare le sequenze. [Per ulteriori informazioni, vedere l'argomento Nodo Sequenza in il capitolo 12 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create sequencenode
connect :databasenode to :sequencenode
#"Fields" tab
set :sequencenode.id_field = 'Age'
set :sequencenode.contiguous = True
set :sequencenode.use_time_field = True
set :sequencenode.time_field = 'Date1'
set :sequencenode.content_fields = ['Drug' 'BP']
set :sequencenode.partition = Test
#"Model" tab
set :sequencenode.use_model_name = True
set :sequencenode.model_name = "Sequence_test"
set :sequencenode.use_partitioned_data = False
set :sequencenode.min_supp = 15.0
set :sequencenode.min_conf = 14.0
set :sequencenode.max_size = 7
set :sequencenode.max_predictions = 5
#"Expert" tab
set :sequencenode.mode = Expert
set :sequencenode.use_max_duration = True
set :sequencenode.max_duration = 3.0
set :sequencenode.use_pruning = True
set :sequencenode.pruning_value = 4.0
set :sequencenode.set_mem_sequences = True
set :sequencenode.mem_sequences = 5.0
set :sequencenode.use_gaps = True
set :sequencenode.min_item_gap = 20.0
```

```
set :sequencenode.max_item_gap = 30.0
```

| Proprietà sequencenode | Valori | Descrizione proprietà |
|------------------------|----------------------------|--|
| id_field | <i>campo</i> | Per creare un modello Sequenza, è necessario specificare un campo ID, un campo ora facoltativo e uno o più campi contenuto. I campi peso e frequenza non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| time_field | <i>campo</i> | |
| use_time_field | <i>flag</i> | |
| content_fields | <i>[campo1 ... campon]</i> | |
| contiguous | <i>flag</i> | |
| min_supp | <i>numero</i> | |
| min_conf | <i>numero</i> | |
| max_size | <i>numero</i> | |
| max_predictions | <i>numero</i> | |
| mode | Simple Expert | |
| use_max_duration | <i>flag</i> | |
| max_duration | <i>numero</i> | |
| use_gaps | <i>flag</i> | |
| min_item_gap | <i>numero</i> | |
| max_item_gap | <i>numero</i> | |
| use_pruning | <i>flag</i> | |
| pruning_value | <i>numero</i> | |
| set_mem_sequences | <i>flag</i> | |
| mem_sequences | <i>intero</i> | |

Proprietà slrmnode



Il nodo Modello risposta autoapprendimento consente di creare un modello in cui è possibile utilizzare un unico nuovo caso oppure un numero limitato di nuovi casi per eseguire una nuova stima del modello senza doverlo riaddestrare con tutti i dati. [Per ulteriori informazioni, vedere l'argomento Nodo SLRM in il capitolo 14 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create slrmnode
set :slrmnode.target = Offer
set :slrmnode.target_response = Response
```

```
set :slrmnode.inputs = ['Cust_ID' 'Age' 'Ave_Bal']
```

| Proprietà slrmnode | Valori | Descrizione proprietà |
|----------------------------------|----------------------------|---|
| target | <i>campo</i> | Il campo obiettivo deve essere nominale o flag. È inoltre possibile specificare un campo frequenza. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| target_response | <i>campo</i> | Il tipo deve essere Flag. |
| continue_training_existing_model | <i>flag</i> | |
| target_field_values | <i>flag</i> | Utilizza tutto: utilizza tutti i valori dalla sorgente. Specifica: è necessario selezionare valori. |
| target_field_values_specify | <i>[campo1 ... campoN]</i> | |
| include_model_assessment | <i>flag</i> | |
| model_assessment_random_seed | <i>numero</i> | Deve essere un numero reale. |
| model_assessment_sample_size | <i>numero</i> | Deve essere un numero reale. |
| model_assessment_iterations | <i>numero</i> | Numero di iterazioni. |
| display_model_evaluation | <i>flag</i> | |
| max_predictions | <i>numero</i> | |
| randomization | <i>numero</i> | |
| scoring_random_seed | <i>numero</i> | |
| sort | Ascending Descending | Specifica se verranno visualizzate per prime le offerte con i punteggi più alti o più bassi. |
| model_reliability | <i>flag</i> | |
| calculate_variable_importance | <i>flag</i> | |

Proprietà statisticsmodelnode



Il nodo Modello Statistics consente di analizzare e operare con i dati eseguendo le procedure IBM® SPSS® Statistics che generano PMML. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. [Per ulteriori informazioni, vedere l'argomento Nodo Modello Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Le proprietà di questo nodo sono descritte in [Proprietà statisticsmodelnode a pag. 296.](#)

Proprietà svmnode



Il nodo SVM (Support Vector Machine) consente di classificare i dati in uno di due gruppi senza sovradattamento. Il nodo SVM è particolarmente indicato per l'utilizzo con insiemi di dati di grandi dimensioni, cioè quelli con un elevato numero di campi di input. [Per ulteriori informazioni, vedere l'argomento Nodo SVM in il capitolo 15 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create svmnode
# Expert tab
set :svmnode.mode=Expert
set :svmnode.all_probabilities=True
set :svmnode.kernel=Polynomial
set :svmnode.gamma=1.5
```

| Proprietà svmnode | Valori | Descrizione proprietà |
|---------------------------------|--|--|
| all_probabilities | <i>flag</i> | |
| stopping_criteria | 1.0E-1 1.0E-2 1.0E-3 (default) 1.0E-4 1.0E-5 1.0E-6 | Determina quando interrompere l'algoritmo di ottimizzazione. |
| regularization | <i>numero</i> | Nota anche come parametro C. |
| precision | <i>numero</i> | Utilizzata solo se il livello di misurazione del campo obiettivo è Continuous. |
| kernel | RBF (default) Polynomial Sigmoid Linear | Tipo di funzione Kernel utilizzata per la trasformazione. |
| rbf_gamma | <i>numero</i> | Utilizzata solo se kernel è RBF. |
| gamma | <i>numero</i> | Utilizzata solo se kernel è Polynomial o Sigmoid. |
| bias | <i>numero</i> | |
| degree | <i>numero</i> | Utilizzata solo se kernel è Polynomial. |
| calculate_variable_importance | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |
| adjusted_propensity_partition | Test Validation | |

Proprietà timeseriesnode

Il nodo Serie storica stima i modelli di livellamento esponenziale, i modelli ARIMA (Autoregressive Integrated Moving Average, autoregressivi integrati a media mobile) univariati e ARIMA (o a funzione di trasferimento) multivariati per i dati di serie storica e genera previsioni di prestazioni future. Il nodo Serie storica deve sempre essere preceduto da un nodo Intervalli di tempo. [Per ulteriori informazioni, vedere l'argomento Nodo Modelli Serie storica in il capitolo 13 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```
create timeseriesnode
set :timeseriesnode.method = Exsmooth
```

```
set :timeseriesnode.exsmooth_model_type = HoltsLinearTrend
```

```
set :timeseriesnode.exsmooth_transformation_type = None
```

| Proprietà timeseriesnode | Valori | Descrizione proprietà |
|----------------------------------|--|--|
| targets | <i>campo</i> | Il nodo Serie storica prevede uno o più obiettivi, utilizzando in via facoltativa uno o più campi di input come predittori. I campi frequenza e peso non sono utilizzati. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| continue | <i>flag</i> | |
| method | ExpertModeler Exsmooth Arima Reuse | |
| expert_modeler_method | <i>flag</i> | |
| consider_seasonal | <i>flag</i> | |
| detect_outliers | <i>flag</i> | |
| expert_outlier_additive | <i>flag</i> | |
| expert_outlier_level_shift | <i>flag</i> | |
| expert_outlier_innovational | <i>flag</i> | |
| expert_outlier_level_shift | <i>flag</i> | |
| expert_outlier_transient | <i>flag</i> | |
| expert_outlier_seasonal_additive | <i>flag</i> | |
| expert_outlier_local_trend | <i>flag</i> | |
| expert_outlier_additive_patch | <i>flag</i> | |
| exsmooth_model_type | Simple HoltsLinearTrend BrownsLinearTrend DampedTrend SimpleSeasonal WintersAdditive WintersMultiplicative | |
| exsmooth_transformation_type | None SquareRoot NaturalLog | |
| arima_p | <i>intero</i> | |
| arima_d | <i>intero</i> | |
| arima_q | <i>intero</i> | |
| arima_sp | <i>intero</i> | |
| arima_sd | <i>intero</i> | |
| arima_sq | <i>intero</i> | |

| Proprietà timeseriesnode | Valori | Descrizione proprietà |
|--|----------------------------------|--|
| arima_transformation_type | None SquareRoot NaturalLog | |
| arima_include_constant | flag | |
| tf_arima_p.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_d.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_q.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_sp.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_sd.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_sq.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_delay.nomecampo | intero | Per le funzioni di trasferimento. |
| tf_arima_transformation_type.nomecampo | None SquareRoot NaturalLog | Per le funzioni di trasferimento. |
| arima_detect_outlier_mode | None Automatic | |
| arima_outlier_additive | flag | |
| arima_outlier_level_shift | flag | |
| arima_outlier_innovational | flag | |
| arima_outlier_transient | flag | |
| arima_outlier_seasonal_additive | flag | |
| arima_outlier_local_trend | flag | |
| arima_outlier_additive_patch | flag | |
| conf_limit_pct | reale | |
| max_lags | intero | |
| events | campi | |
| scoring_model_only | flag | Utilizzato per i modelli con grandi quantità (decine di migliaia) di serie storiche. |

Proprietà twostepnode



Il nodo TwoStep è un metodo di raggruppamento tramite cluster in due fasi. La prima fase esegue un singolo passaggio nei dati per comprimere i dati di input non elaborati in un insieme gestibile di sottocluster. Nella seconda fase viene utilizzato un metodo di raggruppamento tramite cluster gerarchico per unire progressivamente i sottocluster in cluster sempre più grandi. Il nodo TwoStep offre il vantaggio di stimare automaticamente il numero ottimale di cluster per i dati di addestramento. Può gestire in modo efficiente tipi di campo misti e insiemi di dati di grandi dimensioni. [Per ulteriori informazioni, vedere l'argomento Nodo Cluster TwoStep in il capitolo 11 in IBM SPSS Modeler 14.2 Nodi Modelli.](#)

Esempio

```

create twostep
set :twostep.custom_fields = True
set :twostep.inputs = ['Age' 'K' 'Na' 'BP']
set :twostep.partition = Test
set :twostep.use_model_name = False
set :twostep.model_name = "TwoStep_Drug"
set :twostep.use_partitioned_data = True
set :twostep.exclude_outliers = True
set :twostep.cluster_label = "String"
set :twostep.label_prefix = "TwoStep_"
set :twostep.cluster_num_auto = False
set :twostep.max_num_clusters = 9
set :twostep.min_num_clusters = 3
set :twostep.num_clusters = 7

```

| Proprietà twostepnode | Valori | Descrizione proprietà |
|-----------------------|-------------------------------------|--|
| inputs | [<i>campo1</i> ... <i>campoN</i>] | I modelli TwoStep utilizzano un elenco di campi di input, ma nessun campo obiettivo. I campi peso e frequenza non vengono riconosciuti. Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi Modelli a pag. 189. |
| standardize | <i>flag</i> | |
| exclude_outliers | <i>flag</i> | |
| percentage | <i>numero</i> | |
| cluster_num_auto | <i>flag</i> | |
| min_num_clusters | <i>numero</i> | |
| max_num_clusters | <i>numero</i> | |
| num_clusters | <i>numero</i> | |
| cluster_label | String Number | |
| label_prefix | <i>stringa</i> | |
| distance_measure | Euclidean Loglikelihood | |
| clustering_criterion | AIC BIC | |

Proprietà dei nodi degli insiemi di modelli

I nodi degli insiemi di modelli condividono le stesse proprietà comuni agli altri nodi. [Per ulteriori informazioni, vedere l'argomento Proprietà comuni dei nodi in il capitolo 9 a pag. 116.](#)

Proprietà *applyanomalydetectionnode*

I nodi Modelli Rilevamento anomalie si possono utilizzare per generare un insieme di modelli Rilevamento anomalie. Il nome di script di questo insieme di modelli è *applyanomalydetectionnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà anomalydetectionnode nel il capitolo 16 a pag. 190.](#)

| Proprietà <i>applyanomalydetectionnode</i> | Valori | Descrizione proprietà |
|--|---------------------------------------|--|
| <code>anomaly_score_method</code> | FlagAndScore FlagOnly ScoreOnly | Determina quali output sono creati per il calcolo del punteggio. |
| <code>num_fields</code> | <i>intero</i> | Campi da inserire nel report. |
| <code>discard_records</code> | <i>flag</i> | Indica se i record sono scartati o meno dall'output. |
| <code>discard_anomalous_records</code> | <i>flag</i> | Indica se scartare i record anomali o <i>non</i> anomali. L'impostazione di default è off, ad indicare che i record <i>non</i> anomali vengono scartati. Altrimenti, se l'impostazione è su on, verranno scartati i record anomali. Questa proprietà è attivata solo se è attivata la proprietà <code>discard_records</code> . |

Proprietà *applyapriorinode*

I nodi Modelli Apriori si possono utilizzare per generare un insieme di modelli Apriori. Il nome di script di questo insieme di modelli è *applyapriorinode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà apriorinode nel il capitolo 16 a pag. 191.](#)

| Proprietà <i>applyapriorinode</i> | Valori | Descrizione proprietà |
|-----------------------------------|------------------------|-----------------------|
| <code>max_predictions</code> | <i>numero (intero)</i> | |
| <code>ignore_unmattached</code> | <i>flag</i> | |
| <code>allow_repeats</code> | <i>flag</i> | |

| Proprietà <i>applyapriorinode</i> | Valori | Descrizione proprietà |
|-----------------------------------|--|-----------------------|
| check_basket | NoPredictions Predictions NoCheck | |
| criterion | Confidenza Supporto RuleSupport Lift Distribuibilità | |

Proprietà *applyautoclassifiernode*

I nodi Modelli Classificatore automatico si possono utilizzare per generare un insieme di modelli Classificatore automatico. Il nome di script di questo insieme di modelli è *applyautoclassifiernode*. Per ulteriori informazioni sugli script per il nodo Modelli, [vedere Proprietà autoclassifiernode in il capitolo 16 a pag. 192.](#)

| Proprietà <i>applyautoclassifiernode</i> | Valori | Descrizione proprietà |
|--|--|---|
| flag_ensemble_method | Voting ConfidenceWeightedVoting RawPropensityWeightedVoting HighestConfidence AverageRawPropensity | Specifica il metodo utilizzato per determinare il punteggio del classificatore binario. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo flag. |
| flag_voting_tie_selection | Random HighestConfidence RawPropensity | Se è selezionato un metodo di confronto, specifica le modalità di risoluzione delle situazioni di pari merito. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo flag. |
| set_ensemble_method | Voting ConfidenceWeightedVoting HighestConfidence | Specifica il metodo utilizzato per determinare il punteggio del classificatore binario. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo insieme. |
| set_voting_tie_selection | Random HighestConfidence | Se è selezionato un metodo di confronto, specifica le modalità di risoluzione delle situazioni di pari merito. Questa impostazione è valida solamente se l'obiettivo selezionato è un campo nominale. |

Proprietà *applyautoclusternode*

I nodi Modelli Cluster automatico si possono utilizzare per generare un insieme di modelli Cluster automatico. Il nome di script di questo insieme di modelli è *applyautoclusternode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà autoclusternode nel il capitolo 16 a pag. 195.](#)

Proprietà *applyautonumericnode*

I nodi Modelli Numerico automatico si possono utilizzare per generare un insieme di modelli Numerico automatico. Il nome di script di questo insieme di modelli è *applyautonumericnode*. Per ulteriori informazioni sugli script per il nodo Modelli, [vedere Proprietà autonumericnode in il capitolo 16 a pag. 197.](#)

| Proprietà <i>applyautonumericnode</i> | Valori | Descrizione proprietà |
|---------------------------------------|-------------|-----------------------|
| calculate_standard_error | <i>flag</i> | |

Proprietà *applybayesnetnode*

I nodi Modelli Rete bayesiana si possono utilizzare per generare un insieme di modelli Rete bayesiana. Il nome di script di questo insieme di modelli è *applybayesnetnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà bayesnetnode nel il capitolo 16 a pag. 198.](#)

| Proprietà <i>applybayesnetnode</i> | Valori | Descrizione proprietà |
|------------------------------------|-------------|-----------------------|
| all_probabilities | <i>flag</i> | |
| raw_propensity | <i>flag</i> | |
| adjusted_propensity | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applyc50node*

I nodi Modelli C5.0 si possono utilizzare per generare un insieme di modelli C5.0. Il nome di script di questo insieme di modelli è *applyc50node*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà c50node nel il capitolo 16 a pag. 200.](#)

| Proprietà <i>applyc50node</i> | Valori | Descrizione proprietà |
|---------------------------------|--------------------------|--|
| sql_generate | Never NoMissingValues | Consente di impostare le opzioni di generazione SQL durante l'esecuzione dell'insieme di regole. |
| calculate_conf | <i>flag</i> | Disponibile quando è attivata la generazione SQL, questa proprietà include i calcoli di confidenza nell'albero generato. |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applycarmanode*

I nodi Modelli CARMA si possono utilizzare per generare un insieme di modelli CARMA. Il nome di script di questo insieme di modelli è *applycarmanode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà carmanode nel il capitolo 16 a pag. 201.](#)

Proprietà *applycartnode*

I nodi Modelli C&R Tree si possono utilizzare per generare un insieme di modelli C&R Tree. Il nome di script di questo insieme di modelli è *applycartnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà cartnode nel il capitolo 16 a pag. 202.](#)

| Proprietà <i>applycartnode</i> | Valori | Descrizione proprietà |
|---------------------------------|---|---|
| sql_generate | Never MissingValues NoMissingValues | Consente di impostare le opzioni di generazione SQL durante l'esecuzione dell'insieme di regole. |
| calculate_conf | <i>flag</i> | Disponibile quando è attivata la generazione SQL, questa proprietà include i calcoli di confidenza nell'albero generato. |
| display_rule_id | <i>flag</i> | Aggiunge un campo all'output del calcolo del punteggio che indica l'ID del nodo terminale al quale è assegnato ogni record. |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applychaidnode*

I nodi Modelli CHAID si possono utilizzare per generare un insieme di modelli CHAID. Il nome di script di questo insieme di modelli è *applychaidnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà chaidnode nel il capitolo 16 a pag. 205.](#)

| Proprietà <i>applychaidnode</i> | Valori | Descrizione proprietà |
|---------------------------------|------------------------|---|
| sql_generate | Never MissingValues | |
| calculate_conf | <i>flag</i> | |
| display_rule_id | <i>flag</i> | Aggiunge un campo all'output del calcolo del punteggio che indica l'ID del nodo terminale al quale è assegnato ogni record. |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applycoxregnode*

I nodi Modelli Cox si possono utilizzare per generare un insieme di modelli Cox. Il nome di script di questo insieme di modelli è *applycoxregnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà coxregnode nel il capitolo 16 a pag. 207](#).

| Proprietà <i>applycoxregnode</i> | Valori | Descrizione proprietà |
|----------------------------------|---------------------|-----------------------|
| future_time_as | Intervals Fields | |
| time_interval | <i>numero</i> | |
| num_future_times | <i>intero</i> | |
| time_field | <i>campo</i> | |
| past_survival_time | <i>campo</i> | |
| all_probabilities | <i>flag</i> | |
| cumulative_hazard | <i>flag</i> | |

Proprietà *applydecisionlistnode*

I nodi Modelli Elenco decisionale si possono utilizzare per generare un insieme di modelli Elenco decisionale. Il nome di script di questo insieme di modelli è *applydecisionlistnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà decisionlistnode nel il capitolo 16 a pag. 209](#).

| Proprietà <i>applydecisionlistnode</i> | Valori | Descrizione proprietà |
|--|-------------|---|
| enable_sql_generation | <i>flag</i> | Se questa proprietà è vera, IBM® SPSS® Modeler cerca di rinviare il modello Elenco decisionale a SQL. |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applydiscriminantnode*

I nodi Modelli Discriminante si possono utilizzare per generare un insieme di modelli Discriminante. Il nome di script di questo insieme di modelli è *applydiscriminantnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà discriminantnode nel il capitolo 16 a pag. 210](#).

| Proprietà <i>applydiscriminantnode</i> | Valori | Descrizione proprietà |
|--|-------------|-----------------------|
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applyfactornode*

I nodi Modelli Fattoriale si possono utilizzare per generare un insieme di modelli Fattoriale. Il nome di script di questo insieme di modelli è *applyfactornode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà factornode nel il capitolo 16 a pag. 212.](#)

Proprietà *applyfeatureselectionnode*

I nodi Modelli Selezione funzioni si possono utilizzare per generare un insieme di modelli Selezione funzioni. Il nome di script di questo insieme di modelli è *applyfeatureselectionnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà featureselectionnode nel il capitolo 16 a pag. 214.](#)

| Proprietà <i>applyfeatureselectionnode</i> | Valori | Descrizione proprietà |
|--|--------|--|
| selected_ranked_fields | | Specifica quali campi classificati sono selezionati nel browser dei modelli. |
| selected_screened_fields | | Specifica quali campi sottoposti a screening sono selezionati nel browser dei modelli. |

Proprietà *applygeneralizedlinearnode*

I nodi Modelli lineari generalizzati (GenLin) si possono utilizzare per generare un insieme di modelli Lineari generalizzati. Il nome di script di questo insieme di modelli è *applygeneralizedlinearnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà genlinnode nel il capitolo 16 a pag. 216.](#)

| Proprietà <i>applygeneralizedlinearnode</i> | Valori | Descrizione proprietà |
|---|-------------|-----------------------|
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applykmeansnode*

I nodi Modelli K-Means si possono utilizzare per generare un insieme di modelli K-Means. Il nome di script di questo insieme di modelli è *applykmeansnode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà kmeansnode nel il capitolo 16 a pag. 219.](#)

Proprietà *applyknnnode*

I nodi Modelli KNN possono essere utilizzati per generare un insieme di modelli KNN. Il nome di script di questo insieme di modelli è *applyknnnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà knnnode nel il capitolo 16 a pag. 220](#).

| Proprietà <i>applyknnnode</i> | Valori | Descrizione proprietà |
|-------------------------------|-------------|-----------------------|
| all_probabilities | <i>flag</i> | |
| save_distances | <i>flag</i> | |

Proprietà *applykohonennode*

I nodi Modelli Kohonen si possono utilizzare per generare un insieme di modelli Kohonen. Il nome di script di questo insieme di modelli è *applykohonennode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà c50node nel il capitolo 16 a pag. 200](#).

Proprietà *applylinearnode*

I nodi Modelli lineari si possono utilizzare per generare un insieme di modelli lineari. Il nome di script di questo insieme di modelli è *applylinearnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà linearnode nel il capitolo 16 a pag. 222](#).

| Proprietà <i>linear</i> | Valori | Descrizione proprietà |
|-------------------------|----------------|-----------------------|
| use_custom_name | <i>flag</i> | |
| custom_name | <i>stringa</i> | |
| enable_sql_generation | <i>flag</i> | |

Proprietà *applylogregnode*

I nodi Modelli Regressione logistica si possono utilizzare per generare un insieme di modelli Regressione logistica. Il nome di script di questo insieme di modelli è *applylogregnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà logregnode nel il capitolo 16 a pag. 224](#).

| Proprietà <i>applylogregnode</i> | Valori | Descrizione proprietà |
|----------------------------------|-------------|-----------------------|
| calculate_raw_propensities | <i>flag</i> | |

Proprietà *applyneuralnetnode*

I nodi Modelli Rete neurale si possono utilizzare per generare un insieme di modelli Rete neurale. Il nome di script di questo insieme di modelli è *applyneuralnetnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà neuralnetnode nel il capitolo 16 a pag. 228](#).

Attenzione: in questa versione è disponibile una nuova versione dell'insieme Rete neurale con funzionalità avanzate, descritta nella sezione che segue (*applyneuralnetwork*). Sebbene la versione precedente sia ancora disponibile, si consiglia di aggiornare gli script in modo da utilizzare la nuova versione. I dettagli della versione precedente vengono mantenuti in questa sezione per riferimento, ma nelle versioni future non sarà più supportata.

| Proprietà <i>applyneuralnetnode</i> | Valori | Descrizione proprietà |
|-------------------------------------|-----------------------|--|
| calculate_conf | <i>flag</i> | Disponibile quando è attivata la generazione SQL, questa proprietà include i calcoli di confidenza nell'albero generato. |
| enable_sql_generation | <i>flag</i> | |
| nn_score_method | Difference SoftMax | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà *applyneuralnetworknode*

I nodi Modelli Rete neurale si possono utilizzare per generare un insieme di modelli Rete neurale. Il nome di script di questo insieme di modelli è *applyneuralnetworknode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà neuralnetworknode nel capitolo 16 a pag. 230](#).

| Proprietà <i>applyneuralnetworknode</i> | Valori | Descrizione proprietà |
|---|-----------------------------|-----------------------|
| use_custom_name | <i>flag</i> | |
| custom_name | <i>stringa</i> | |
| confidence | onProbability onIncrease | |
| score_category_probabilities | <i>flag</i> | |
| max_categories | <i>numero</i> | |
| score_propensity | <i>flag</i> | |

Proprietà *applyquestnode*

I nodi Modelli QUEST si possono utilizzare per generare un insieme di modelli QUEST. Il nome di script di questo insieme di modelli è *applyquestnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà questnode nel capitolo 16 a pag. 232](#).

| Proprietà <i>applyquestnode</i> | Valori | Descrizione proprietà |
|---------------------------------|---|---|
| sql_generate | Never MissingValues NoMissingValues | |
| calculate_conf | <i>flag</i> | |
| display_rule_id | <i>flag</i> | Aggiunge un campo all'output del calcolo del punteggio che indica l'ID del nodo terminale al quale è assegnato ogni record. |

| Proprietà applyquestnode | Valori | Descrizione proprietà |
|---------------------------------|-------------|-----------------------|
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà **applyregressionnode**

I nodi Modelli Regressione lineare si possono utilizzare per generare un insieme di modelli Regressione lineare. Il nome di script di questo insieme di modelli è *applyregressionnode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà regressionnode nel capitolo 16 a pag. 234](#).

Proprietà **applyselflearningnode**

I nodi Modelli SLRM (Risposta autoapprendimento) si possono utilizzare per generare un insieme di modelli SLRM. Il nome di script di questo insieme di modelli è *applyselflearningnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà slrmnode nel capitolo 16 a pag. 237](#).

| Proprietà applyselflearningnode | Valori | Descrizione proprietà |
|--|-------------------------|--|
| max_predictions | <i>numero</i> | |
| randomization | <i>numero</i> | |
| scoring_random_seed | <i>numero</i> | |
| sort | ascending descending | Specifica se verranno visualizzate per prime le offerte con i punteggi più alti o più bassi. |
| model_reliability | <i>flag</i> | Tiene conto dell'opzione di affidabilità del modello inclusa nella scheda Impostazioni. |

Proprietà **applysequencenode**

I nodi Modelli Sequenza si possono utilizzare per generare un insieme di modelli Sequenza. Il nome di script di questo insieme di modelli è *applysequencenode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà sequencenode nel capitolo 16 a pag. 236](#).

Proprietà **applysvmnode**

I nodi Modelli SVM si possono utilizzare per generare un insieme di modelli SVM. Il nome di script di questo insieme di modelli è *applysvmnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà svmnode nel capitolo 16 a pag. 238](#).

| Proprietà applysvmnode | Valori | Descrizione proprietà |
|---------------------------------|-------------|-----------------------|
| all_probabilities | <i>flag</i> | |
| calculate_raw_propensities | <i>flag</i> | |
| calculate_adjusted_propensities | <i>flag</i> | |

Proprietà applytimeseriesnode

I nodi Modelli Serie storica si possono utilizzare per generare un insieme di modelli Serie storica. Il nome di script di questo insieme di modelli è *applytimeseriesnode*. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà timeseriesnode nel il capitolo 16 a pag. 239](#).

| Proprietà applytimeseriesnode | Valori | Descrizione proprietà |
|--------------------------------------|---------------|------------------------------|
| calculate_conf | <i>flag</i> | |
| calculate_residuals | <i>flag</i> | |

Proprietà applytwostepnode

I nodi Modelli TwoStep si possono utilizzare per generare un insieme di modelli TwoStep. Il nome di script di questo insieme di modelli è *applytwostepnode*. Per questo insieme di modelli non esistono altre proprietà. Per ulteriori informazioni sugli script del nodo Modelli specifico, [vedere Proprietà twostepnode nel il capitolo 16 a pag. 241](#).

Proprietà dei nodi Modelli database

IBM® SPSS® Modeler supporta l'integrazione con gli strumenti di modellazione e di data mining offerti da diversi fornitori di database, quali Microsoft SQL Server Analysis Services, Oracle Data Mining, IBM® DB2® InfoSphere Warehouse e IBM® Netezza® Analytics. [Per ulteriori informazioni, vedere l'argomento Panoramica sulla modellazione di database in il capitolo 2 in IBM SPSS Modeler 14.2 Guida alla modellazione in-database.](#) Operando all'interno dell'applicazione SPSS Modeler è infatti possibile creare modelli e calcolarne il punteggio mediante algoritmi nativi del database. I modelli di database possono essere creati e manipolati anche tramite script utilizzando le proprietà descritte in questa sezione.

Per esempio, lo script seguente illustra come creare un modello di alberi decisionali Microsoft tramite l'interfaccia script di SPSS Modeler:

```
create mstreenode
rename :mstreenode as msbuilder
set msbuilder.analysis_server_name = 'hostlocale'
set msbuilder.analysis_database_name = 'TESTDB'
set msbuilder.mode = 'Expert'
set msbuilder.datasource = 'ServerLocale'
set msbuilder.target = 'Cura'
set msbuilder.inputs = ['Età' 'Sesso']
set msbuilder.unique_field = 'IDX'
set msbuilder.custom_fields = true
set msbuilder.model_name = 'CURAMS'

connect :typenode to msbuilder
execute msbuilder

insert model CURAMS connected between :typenode and :tablenode
set CURAMS.sql_generate = true
execute :tablenode
```

Proprietà dei nodi Modelli Microsoft

Proprietà dei nodi Modelli Microsoft

Proprietà comuni

Le seguenti proprietà sono comuni ai nodi Modelli database Microsoft.

| Proprietà comuni dei nodi Microsoft | Valori | Descrizione proprietà |
|--|----------------|---|
| analysis_database_name | <i>stringa</i> | Nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Services. |

| Proprietà comuni dei nodi Microsoft | Valori | Descrizione proprietà |
|-------------------------------------|----------------------------|---|
| use_transactional_data | <i>flag</i> | Specifica se i dati di input sono in formato tabulare o transazionale. |
| inputs | <i>[campo campo campo]</i> | Campi di input per dati tabulari. |
| target | <i>campo</i> | Campo predittore (non applicabile al nodo Raggruppamento cluster MS o Cluster di sequenze MS). |
| unique_field | <i>campo</i> | Campo chiave. |
| msas_parameters | <i>strutturato</i> | Parametri degli algoritmi. Per ulteriori informazioni, vedere l'argomento Parametri degli algoritmi a pag. 255. |
| with_drillthrough | <i>flag</i> | Opzione Con funzione drill-through. |

Albero decisionale MS

Non vi sono proprietà specifiche definite per i nodi di tipo *mstreenode*. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Raggruppamento cluster MS

Non vi sono proprietà specifiche definite per i nodi di tipo *msclusternode*. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Regole di associazione Microsoft

Per i nodi di tipo *msassocnode* sono disponibili le seguenti proprietà specifiche:

| Proprietà <i>msassocnode</i> | Valori | Descrizione proprietà |
|------------------------------|----------------------------|---|
| id_field | <i>campo</i> | Identifica le singole transazioni nei dati. |
| trans_inputs | <i>[campo campo campo]</i> | Campi di input per dati transazionali. |
| transactional_target | <i>campo</i> | Campo predittore (dati transazionali). |

Naive Bayes Microsoft

Non vi sono proprietà specifiche definite per i nodi di tipo *msbayesnode*. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Regressione lineare MS

Non vi sono proprietà specifiche definite per i nodi di tipo *msregressionnode*. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Rete neurale MS

Non vi sono proprietà specifiche definite per i nodi di tipo *msneuralnetworknode*. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Regressione logistica MS

Non vi sono proprietà specifiche definite per i nodi di tipo `mslogisticnode`. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Serie storica Microsoft

Non vi sono proprietà specifiche definite per i nodi di tipo `mstimeseriesnode`. Vedere le proprietà comuni di Microsoft all'inizio di questa sezione.

Cluster di sequenze MS

Per i nodi di tipo `mssequenceclusternode` sono disponibili le seguenti proprietà specifiche:

| Proprietà <code>mssequenceclusternode</code> | Valori | Descrizione proprietà |
|--|----------------------------|---|
| <code>id_field</code> | <i>campo</i> | Identifica le singole transazioni nei dati. |
| <code>input_fields</code> | <i>[campo campo campo]</i> | Campi di input per dati transazionali. |
| <code>sequence_field</code> | <i>campo</i> | Identificativo sequenza. |
| <code>target_field</code> | <i>campo</i> | Campo predittore (dati tabulari). |

Parametri degli algoritmi

Ogni tipo di modello di database Microsoft possiede parametri specifici che è possibile impostare mediante la proprietà `msas_parameters`, per esempio:

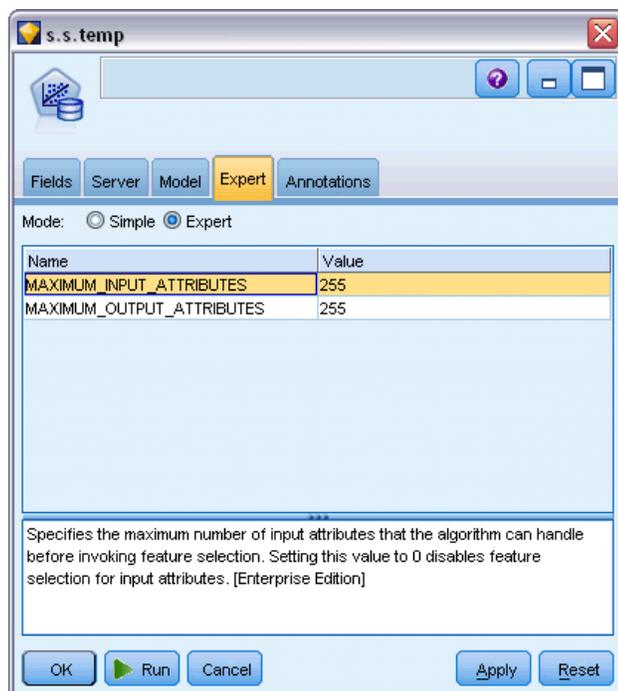
```
set :msregressionnode.msas_parameters =
[{"MAXIMUM_INPUT_ATTRIBUTES" 255}{ "MAXIMUM_OUTPUT_ATTRIBUTES" 255}]
```

Tali parametri sono derivati da SQL Server. Per visualizzare i parametri relativi ai singoli nodi:

- ▶ Collocare un nodo di input di database nell'area di disegno.
- ▶ Aprire il nodo di input di database.
- ▶ Selezionare un'origine valida dall'elenco a discesa Sorgente dati.
- ▶ Selezionare una tabella valida dall'elenco Nome tabella.
- ▶ Fare clic su OK per chiudere il nodo di input di database.
- ▶ Collegare il nodo Modelli database Microsoft di cui si desiderano elencare le proprietà.
- ▶ Aprire il nodo Modelli database.
- ▶ Selezionare la scheda Livello avanzato.

Vengono visualizzate le proprietà `msas_parameters` disponibili per quel nodo.

Figura 18-1
Esempio di visualizzazione dei parametri di un algoritmo



Proprietà degli insiemi di modelli Microsoft

Le seguenti proprietà sono relative agli insiemi di modelli creati mediante i nodi Modelli database Microsoft.

Albero decisionale MS

| Proprietà applymstreenode | Valori | Descrizione |
|----------------------------------|----------------|--|
| analysis_database_name | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Server. |
| datasource | <i>stringa</i> | Nome del DSN (nome sorgente dati, Data Source Name) ODBC SQL Server. |
| sql_generate | <i>flag</i> | Attiva la generazione SQL. |

Regressione lineare MS

| Proprietà appliesregressionnode | Valori | Descrizione |
|--|----------------|--|
| analysis_database_name | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Server. |

Rete neurale MS

| Proprietà appliesneuralnetworknode | Valori | Descrizione |
|---|----------------|--|
| analysis_database_name | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Server. |

Regressione logistica MS

| Proprietà applieslogisticnode | Valori | Descrizione |
|--------------------------------------|----------------|--|
| analysis_database_name | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Server. |

Serie storica Microsoft

| Proprietà appliestime-seriesnode | Valori | Descrizione |
|---|---|--|
| analysis_database_name | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| analysis_server_name | <i>stringa</i> | Nome dell'host di Analysis Server. |
| start_from | <i>new_prediction</i> <i>historical_prediction</i> | Specifica se effettuare previsioni future o storiche. |
| new_step | <i>numero</i> | Definisce il periodo di tempo iniziale per le previsioni future. |
| historical_step | <i>numero</i> | Definisce il periodo di tempo iniziale per le previsioni storiche. |
| end_step | <i>numero</i> | Definisce il periodo di tempo finale per le previsioni. |

Cluster di sequenze MS

| Proprietà <code>applsse- quenceclusternode</code> | Valori | Descrizione |
|---|----------------|--|
| <code>analysis_database_name</code> | <i>stringa</i> | Il calcolo del punteggio di questo nodo può essere eseguito direttamente in uno stream. Questa proprietà consente di identificare il nome del database di Analysis Services. |
| <code>analysis_server_name</code> | <i>stringa</i> | Nome dell'host di Analysis Server. |

Proprietà dei nodi Modelli Oracle**Proprietà dei nodi Modelli Oracle**

Le seguenti proprietà sono comuni ai nodi Modelli database Oracle.

| Proprietà comuni dei nodi Oracle | Valori | Descrizione proprietà |
|---|------------------------|---|
| <code>target</code> | <i>campo</i> | |
| <code>inputs</code> | <i>Elenco di campi</i> | |
| <code>partition</code> | <i>campo</i> | Campo utilizzato per partizionare i dati in campioni distinti per le fasi di addestramento, di test e di validazione della creazione del modello. |
| <code>datasource</code> | | |
| <code>username</code> | | |
| <code>password</code> | | |
| <code>epassword</code> | | |
| <code>use_model_name</code> | <i>flag</i> | |
| <code>model_name</code> | <i>stringa</i> | Nome personalizzato per il nuovo modello. |
| <code>use_partitioned_data</code> | <i>flag</i> | Se è definito un campo di partizione, questa opzione garantisce che per la creazione del modello verranno utilizzati solo i dati della partizione di addestramento. |
| <code>unique_field</code> | <i>campo</i> | |
| <code>auto_data_prep</code> | <i>flag</i> | Attiva o disattiva la funzione di preparazione dei dati automatici di Oracle (solo database 11g). |
| <code>costs</code> | <i>matrice costi</i> | Proprietà strutturata nel formato: <code>{{drugA drugB 1.5} {drugA drugC 2.1}}</code> , dove gli argomenti racchiusi tra <code>{}</code> sono i costi previsti effettivi. |
| <code>mode</code> | Simple Expert | Consente di ignorare determinate proprietà se impostate su <code>Simple</code> , come illustrato nelle proprietà dei singoli nodi. |
| <code>use_prediction_probability</code> | <i>flag</i> | |
| <code>prediction_probability</code> | <i>stringa</i> | |
| <code>use_prediction_set</code> | <i>flag</i> | |

Bayes naïve Oracle

Per i nodi di tipo oranbnode sono disponibili le seguenti proprietà.

| Proprietà oranbnode | Valori | Descrizione proprietà |
|---------------------|-------------------------|--|
| singleton_threshold | numero | 0.0–1.0.* |
| pairwise_threshold | numero | 0.0–1.0.* |
| priors | Data Equal Custom | |
| custom_priors | strutturato | Proprietà strutturata nel formato: set :oranbnode.custom_priors = [{drugA 1}{drugB 2}{drugC 3}{drugX 4}{drugY 5}] |

* Proprietà ignorata se mode è impostata su Simple.

Bayes adattivi Oracle

Per i nodi di tipo oraabnnode sono disponibili le seguenti proprietà.

| Proprietà oraabnnode | Valori | Descrizione proprietà |
|----------------------------|---|---|
| model_type | SingleFeature MultiFeature NaiveBayes | |
| use_execution_time_limit | flag | * |
| execution_time_limit | intero | Il valore deve essere maggiore di 0.* |
| max_naive_bayes_predictors | intero | Il valore deve essere maggiore di 0.* |
| max_predictors | intero | Il valore deve essere maggiore di 0.* |
| priors | Data Equal Custom | |
| custom_priors | strutturato | Proprietà strutturata nel formato: set :oraabnnode.custom_priors = [{drugA 1}{drugB 2}{drugC 3}{drugX 4}{drugY 5}] |

* Proprietà ignorata se mode è impostata su Simple.

SVM Oracle

Per i nodi di tipo orasvmnode sono disponibili le seguenti proprietà.

| Proprietà orasvmnode | Valori | Descrizione proprietà |
|----------------------|------------------------------|-----------------------|
| active_learning | Enable Disable | |
| kernel_function | Linear Gaussian System | |
| normalization_method | zscore minmax none | |

| Proprietà orasvmnode | Valori | Descrizione proprietà |
|------------------------|-------------------------|--|
| kernel_cache_size | <i>intero</i> | Solo kernel gaussiano. Il valore deve essere maggiore di 0.* |
| convergence_tolerance | <i>numero</i> | Il valore deve essere maggiore di 0.* |
| use_standard_deviation | <i>flag</i> | Solo kernel gaussiano.* |
| standard_deviation | <i>numero</i> | Il valore deve essere maggiore di 0.* |
| use_epsilon | <i>flag</i> | Solo modelli di regressione.* |
| epsilon | <i>numero</i> | Il valore deve essere maggiore di 0.* |
| use_complexity_factor | <i>flag</i> | * |
| complexity_factor | <i>numero</i> | * |
| use_outlier_rate | <i>flag</i> | Solo variante a una classe.* |
| outlier_rate | <i>numero</i> | Solo variante a una classe. 0.0–1.0.* |
| weights | Data Equal Custom | |
| custom_weights | <i>strutturato</i> | Proprietà strutturata nel formato: set :orasvmnode.custom_weights = [{drugA 1}{drugB 2}{drugC 3}{drugX 4}{drugY 5}] |

* Proprietà ignorata se mode è impostata su Simple.

Modelli lineari generalizzati Oracle

Per i nodi di tipo oraglmnode sono disponibili le seguenti proprietà.

| Proprietà oraglmnode | Valori | Descrizione proprietà |
|------------------------|--|-----------------------|
| normalization_method | zscore minmax none | |
| missing_value_handling | ReplaceWith-Mean UseCompleteRecords | |
| use_row_weights | <i>flag</i> | * |
| row_weights_field | <i>campo</i> | * |
| save_row_diagnostics | <i>flag</i> | * |
| row_diagnostics_table | <i>stringa</i> | * |
| coefficient_confidence | <i>numero</i> | * |
| use_reference_category | <i>flag</i> | * |
| reference_category | <i>stringa</i> | * |
| ridge_regression | Auto Off On | * |
| parameter_value | <i>numero</i> | * |
| vif_for_ridge | <i>flag</i> | * |

* Proprietà ignorata se mode è impostata su Simple.

Albero decisionale Oracle

Per i nodi di tipo `oradecisiontreenode` sono disponibili le seguenti proprietà.

| Proprietà <code>oradecisiontreenode</code> | Valori | Descrizione proprietà |
|--|-----------------|---------------------------------------|
| <code>use_costs</code> | <i>flag</i> | |
| <code>impurity_metric</code> | Entropy Gini | |
| <code>term_max_depth</code> | <i>intero</i> | 2–20.* |
| <code>term_minpct_node</code> | <i>numero</i> | 0.0–10.0.* |
| <code>term_minpct_split</code> | <i>numero</i> | 0.0–20.0.* |
| <code>term_minrec_node</code> | <i>intero</i> | Il valore deve essere maggiore di 0.* |
| <code>term_minrec_split</code> | <i>intero</i> | Il valore deve essere maggiore di 0.* |
| <code>display_rule_ids</code> | <i>flag</i> | * |

* Proprietà ignorata se `mode` è impostata su `Simple`.

O-Cluster Oracle

Per i nodi di tipo `oraoclusternode` sono disponibili le seguenti proprietà.

| Proprietà <code>oraoclusternode</code> | Valori | Descrizione proprietà |
|--|---------------|---------------------------------------|
| <code>max_num_clusters</code> | <i>intero</i> | Il valore deve essere maggiore di 0. |
| <code>max_buffer</code> | <i>intero</i> | Il valore deve essere maggiore di 0.* |
| <code>sensitivity</code> | <i>numero</i> | 0.0–1.0.* |

* Proprietà ignorata se `mode` è impostata su `Simple`.

K-Means Oracle

Per i nodi di tipo `orakmeansnode` sono disponibili le seguenti proprietà.

| Proprietà <code>orakmeansnode</code> | Valori | Descrizione proprietà |
|--------------------------------------|--------------------------|--|
| <code>num_clusters</code> | <i>intero</i> | Il valore deve essere maggiore di 0. |
| <code>normalization_method</code> | zscore minmax none | |
| <code>distance_function</code> | Euclidean Cosine | |
| <code>iterations</code> | <i>intero</i> | 0–20.* |
| <code>conv_tolerance</code> | <i>numero</i> | 0.0–0.5.* |
| <code>split_criterion</code> | Variance Size | L'impostazione di default è <code>Variance</code> .* |
| <code>num_bins</code> | <i>intero</i> | Il valore deve essere maggiore di 0.* |
| <code>block_growth</code> | <i>intero</i> | 1–5.* |
| <code>min_pct_attr_support</code> | <i>numero</i> | 0.0–1.0.* |

* Proprietà ignorata se `mode` è impostata su `Simple`.

NMF Oracle

Per i nodi di tipo `orannfnode` sono disponibili le seguenti proprietà.

| Proprietà <code>orannfnode</code> | Valori | Descrizione proprietà |
|-----------------------------------|----------------|--|
| <code>normalization_method</code> | minmax none | |
| <code>use_num_features</code> | <i>flag</i> | * |
| <code>num_features</code> | <i>intero</i> | 0–1. Il valore di default viene stimato dall'algoritmo in base ai dati.* |
| <code>random_seed</code> | <i>numero</i> | * |
| <code>num_iterations</code> | <i>intero</i> | 0–500.* |
| <code>conv_tolerance</code> | <i>numero</i> | 0.0–0.5.* |
| <code>display_all_features</code> | <i>flag</i> | * |

* Proprietà ignorata se `mode` è impostata su `Simple`.

Apriori Oracle

Per i nodi di tipo `oraapriorinode` sono disponibili le seguenti proprietà.

| Proprietà <code>oraapriorinode</code> | Valori | Descrizione proprietà |
|---------------------------------------|---------------|-----------------------|
| <code>content_field</code> | <i>campo</i> | |
| <code>id_field</code> | <i>campo</i> | |
| <code>max_rule_length</code> | <i>intero</i> | 2–20. |
| <code>min_confidence</code> | <i>numero</i> | 0.0–1.0. |
| <code>min_support</code> | <i>numero</i> | 0.0–1.0. |
| <code>use_transactional_data</code> | <i>flag</i> | |

Oracle MDL (Lunghezza descrizione minima)

Non vi sono proprietà specifiche definite per i nodi di tipo `oramdlnode`. Vedere le proprietà comuni di Oracle all'inizio di questa sezione.

Importanza attributo Oracle (AI)

Per i nodi di tipo `oraainode` sono disponibili le seguenti proprietà.

| Proprietà <code>oraainode</code> | Valori | Descrizione proprietà |
|----------------------------------|---|---|
| <code>custom_fields</code> | <i>flag</i> | Se vera, consente di specificare i campi obiettivo, di input e di altro tipo per il nodo corrente. Se falsa, vengono utilizzate le impostazioni correnti di un nodo Tipo a monte. |
| <code>selection_mode</code> | ImportanceLevel Importance-Value TopN | |

| Proprietà oraainode | Valori | Descrizione proprietà |
|---------------------|----------------|--|
| select_important | <i>flag</i> | Quando <i>selection_mode</i> è impostata su <i>ImportanceLevel</i> , specifica se selezionare i campi importanti. |
| important_label | <i>stringa</i> | Specifica l'etichetta per la classificazione "importante". |
| select_marginal | <i>flag</i> | Quando <i>selection_mode</i> è impostata su <i>ImportanceLevel</i> , specifica se selezionare i campi marginali. |
| marginal_label | <i>stringa</i> | Specifica l'etichetta per la classificazione "marginale". |
| important_above | <i>numero</i> | 0.0–1.0. |
| select_unimportant | <i>flag</i> | Quando <i>selection_mode</i> è impostata su <i>ImportanceLevel</i> , specifica se selezionare i campi non importanti. |
| unimportant_label | <i>stringa</i> | Specifica l'etichetta per la classificazione "non importante". |
| unimportant_below | <i>numero</i> | 0.0–1.0. |
| importance_value | <i>numero</i> | Quando <i>selection_mode</i> è impostata su <i>ImportanceValue</i> , specifica il valore di interruzione da utilizzare. Accetta i valori compresi fra 0 e 100. |
| top_n | <i>numero</i> | Quando <i>selection_mode</i> è impostata su <i>TopN</i> , specifica il valore di interruzione da utilizzare. Accetta i valori compresi fra 0 e 1000. |

Proprietà degli insiemi di modelli Oracle

Le seguenti proprietà sono relative agli insiemi di modelli creati mediante i modelli Oracle.

Bayes naive Oracle

Non vi sono proprietà specifiche definite per i nodi di tipo *applyoranbnode*.

Bayes adattivi Oracle

Non vi sono proprietà specifiche definite per i nodi di tipo *applyoraabnode*.

SVM Oracle

Non vi sono proprietà specifiche definite per i nodi di tipo *applyorasvmnode*.

Albero decisionale Oracle

Per i nodi di tipo *applyradecisiontreenode* sono disponibili le seguenti proprietà.

| Proprietà <i>applyradecisiontreenode</i> | Valori | Descrizione proprietà |
|--|-------------|-----------------------|
| <i>use_costs</i> | <i>flag</i> | |
| <i>display_rule_ids</i> | <i>flag</i> | |

O-Cluster Oracle

Non vi sono proprietà specifiche definite per i nodi di tipo `applyoraoclusternode`.

K-Means Oracle

Non vi sono proprietà specifiche definite per i nodi di tipo `applyorakmeansnode`.

NMF Oracle

Per i nodi di tipo `applyoranmfnode` è disponibile la seguente proprietà:

| Proprietà <code>applyoranmfnode</code> | Valori | Descrizione proprietà |
|--|-------------|-----------------------|
| <code>display_all_features</code> | <i>flag</i> | |

Apriori Oracle

Questo insieme di modelli non può essere applicato negli script.

MDL Oracle

Questo insieme di modelli non può essere applicato negli script.

Proprietà dei nodi Modelli IBM DB2**Proprietà dei nodi Modelli IBM DB2**

Le seguenti proprietà sono comuni ai nodi Modelli database IBM InfoSphere Warehouse (ISW).

| Proprietà comuni dei nodi ISW | Valori | Descrizione proprietà |
|---|------------------------|---------------------------------------|
| <code>inputs</code> | <i>Elenco di campi</i> | |
| <code>datasource</code> | | |
| <code>username</code> | | |
| <code>password</code> | | |
| <code>epassword</code> | | |
| <code>enable_power_options</code> | <i>flag</i> | |
| <code>power_options_max_memory</code> | <i>intero</i> | Il valore deve essere maggiore di 32. |
| <code>power_options_cmdline</code> | <i>stringa</i> | |
| <code>mining_data_custom_sql</code> | <i>stringa</i> | |
| <code>logical_data_custom_sql</code> | <i>stringa</i> | |
| <code>mining_settings_custom_sql</code> | | |

Albero decisionale ISW

Per i nodi di tipo db2imtreenode sono disponibili le seguenti proprietà.

| Proprietà db2imtreenode | Valori | Descrizione proprietà |
|----------------------------|--------------------|--|
| target | <i>campo</i> | |
| perform_test_run | <i>flag</i> | |
| use_max_tree_depth | <i>flag</i> | |
| max_tree_depth | <i>intero</i> | Valore maggiore di 0. |
| use_maximum_purity | <i>flag</i> | |
| maximum_purity | <i>numero</i> | Numero compreso tra 0 e 100. |
| use_minimum_internal_cases | <i>flag</i> | |
| minimum_internal_cases | <i>intero</i> | Valore maggiore di 1. |
| use_costs | <i>flag</i> | |
| costs | <i>strutturato</i> | Proprietà strutturata nel formato: {{drugA drugB 1.5} {drugA drugC 2.1}}, dove gli argomenti racchiusi tra {} sono i costi previsti effettivi. |

Associazione ISW

Per i nodi di tipo db2imassocnode sono disponibili le seguenti proprietà.

| Proprietà db2imassocnode | Valori | Descrizione proprietà |
|----------------------------|--------------------|--|
| use_transactional_data | <i>flag</i> | |
| id_field | <i>campo</i> | |
| content_field | <i>campo</i> | |
| max_rule_size | <i>intero</i> | Il valore deve essere maggiore di 2. |
| min_rule_support | <i>numero</i> | 0–100% |
| min_rule_confidence | <i>numero</i> | 0–100% |
| use_item_constraints | <i>flag</i> | |
| item_constraints_type | Include Exclude | |
| use_taxonomy | <i>flag</i> | |
| taxonomy_table_name | <i>stringa</i> | Nome della tabella DB2 in cui archiviare i dettagli relativi alla tassonomia. |
| taxonomy_child_column_name | <i>stringa</i> | Nome della colonna figlio nella tabella di tassonomia. Tale colonna contiene i nomi di elemento o i nomi di categoria. |

| Proprietà db2imassocnode | Valori | Descrizione proprietà |
|-----------------------------|----------------|---|
| taxonomy_parent_column_name | <i>stringa</i> | Nome della colonna padre nella tabella di tassonomia. Tale colonna contiene i nomi di categoria. |
| load_taxonomy_to_table | <i>flag</i> | Stabilisce se le informazioni di tassonomia memorizzate in IBM® SPSS® Modeler devono essere caricate nella tabella di tassonomia nella fase di creazione del modello. Si noti che se già esiste una tabella di tassonomia, tale tabella verrà eliminata. Le informazioni relative alla tassonomia vengono archiviate con il nodo di creazione modello e si possono modificare mediante i pulsanti Modifica categorie e Modifica tassonomia. |

Sequenza ISW

Per i nodi di tipo db2imsequencenode sono disponibili le seguenti proprietà.

| Proprietà db2imsequencenode | Valori | Descrizione proprietà |
|-----------------------------|--------------------|---|
| id_field | <i>campo</i> | |
| group_field | <i>campo</i> | |
| content_field | <i>campo</i> | |
| max_rule_size | <i>intero</i> | Il valore deve essere maggiore di 2. |
| min_rule_support | <i>numero</i> | 0–100% |
| min_rule_confidence | <i>numero</i> | 0–100% |
| use_item_constraints | <i>flag</i> | |
| item_constraints_type | Include Exclude | |
| use_taxonomy | <i>flag</i> | |
| taxonomy_table_name | <i>stringa</i> | Nome della tabella DB2 in cui archiviare i dettagli relativi alla tassonomia. |
| taxonomy_child_column_name | <i>stringa</i> | Nome della colonna figlio nella tabella di tassonomia. Tale colonna contiene i nomi di elemento o i nomi di categoria. |
| taxonomy_parent_column_name | <i>stringa</i> | Nome della colonna padre nella tabella di tassonomia. Tale colonna contiene i nomi di categoria. |
| load_taxonomy_to_table | <i>flag</i> | Stabilisce se le informazioni di tassonomia memorizzate in SPSS Modeler devono essere caricate nella tabella di tassonomia nella fase di creazione del modello. Si noti che se già esiste una tabella di tassonomia, tale tabella verrà eliminata. Le informazioni relative alla tassonomia vengono archiviate con il nodo di creazione modello e si possono modificare mediante i pulsanti Modifica categorie e Modifica tassonomia. |

Regressione ISW

Per i nodi di tipo db2imregnode sono disponibili le seguenti proprietà.

| Proprietà db2imregnode | Valori | Descrizione proprietà |
|---|--|--|
| target | <i>campo</i> | |
| regression_method | transform linear polynomial rbf | |
| perform_test_run | <i>campo</i> | |
| limit_rsquared_value | <i>flag</i> | |
| max_rsquared_value | <i>numero</i> | Valore compreso tra 0,0 e 1,0. |
| use_execution_time_limit | <i>flag</i> | |
| execution_time_limit_mins | <i>intero</i> | Valore maggiore di 0. |
| use_max_degree_polynomial | <i>flag</i> | |
| max_degree_polynomial | <i>intero</i> | |
| use_intercept | <i>flag</i> | |
| use_auto_feature_selection_method | <i>flag</i> | |
| auto_feature_selection_method | normal adjusted | |
| use_min_significance_level | <i>flag</i> | |
| min_significance_level | <i>numero</i> | |
| use_min_significance_level | <i>flag</i> | |
| Le seguenti proprietà sono valide solo se regression_method = rbf | | |
| use_output_sample_size | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| output_sample_size | <i>intero</i> | L'impostazione di default è 2. Il minimo è 1. |
| use_input_sample_size | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| input_sample_size | <i>intero</i> | L'impostazione di default è 2. Il minimo è 1. |
| use_max_num_centers | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| max_num_centers | <i>intero</i> | L'impostazione di default è 20. Il minimo è 1. |
| use_min_region_size | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| min_region_size | <i>intero</i> | L'impostazione di default è 15. Il minimo è 1. |
| use_max_data_passes | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| max_data_passes | <i>intero</i> | L'impostazione di default è 5. Il minimo è 2. |
| use_min_data_passes | <i>flag</i> | Se vera, imposta automaticamente il valore di default. |
| min_data_passes | <i>intero</i> | L'impostazione di default è 5. Il minimo è 2. |

Raggruppamento cluster ISW

Per i nodi di tipo `db2imclusternode` sono disponibili le seguenti proprietà.

| Proprietà <code>db2imclusternode</code> | Valori | Descrizione proprietà |
|---|------------------------|--------------------------------|
| <code>cluster_method</code> | demographic kohonen | |
| <code>kohonen_num_rows</code> | <i>intero</i> | |
| <code>kohonen_num_columns</code> | <i>intero</i> | |
| <code>kohonen_passes</code> | <i>intero</i> | |
| <code>use_num_passes_limit</code> | <i>flag</i> | |
| <code>use_num_clusters_limit</code> | <i>flag</i> | |
| <code>max_num_clusters</code> | <i>intero</i> | Valore maggiore di 1. |
| <code>use_execution_time_limit</code> | <i>flag</i> | |
| <code>execution_time_limit_mins</code> | <i>intero</i> | Valore maggiore di 0. |
| <code>min_data_percentage</code> | <i>numero</i> | 0–100% |
| <code>maximum_CF_leaf_nodes</code> | <i>intero</i> | Il valore di default è 1000. |
| <code>use_similarity_threshold</code> | <i>flag</i> | |
| <code>similarity_threshold</code> | <i>numero</i> | Valore compreso tra 0,0 e 1,0. |

Bayes naive ISW

Per i nodi di tipo `db2imnbsnode` sono disponibili le seguenti proprietà.

| Proprietà <code>db2imnbsnode</code> | Valori | Descrizione proprietà |
|-------------------------------------|--------------------|---|
| <code>perform_test_run</code> | <i>flag</i> | |
| <code>probability_threshold</code> | <i>numero</i> | L'impostazione di default è 0,001. Il valore minimo è 0; il valore massimo è 1,000 |
| <code>use_costs</code> | <i>flag</i> | |
| <code>costs</code> | <i>strutturato</i> | Proprietà strutturata nel formato: {{drugA drugB 1.5} {drugA drugC 2.1}}, dove gli argomenti racchiusi tra {} sono i costi previsti effettivi. |

Regressione logistica ISW

Per i nodi di tipo `db2imlognode` sono disponibili le seguenti proprietà.

| Proprietà <code>db2imlognode</code> | Valori | Descrizione proprietà |
|-------------------------------------|--------------------|---|
| <code>perform_test_run</code> | <i>flag</i> | |
| <code>use_costs</code> | <i>flag</i> | |
| <code>costs</code> | <i>strutturato</i> | Proprietà strutturata nel formato: {{drugA drugB 1.5} {drugA drugC 2.1}}, dove gli argomenti racchiusi tra {} sono i costi previsti effettivi. |

Serie storica ISW

Nota: per questo nodo non viene utilizzato il parametro dei campi di input. Se questo parametro viene rilevato nello script, un messaggio di avvertenza segnala che il nodo dispone dei campi in entrata *time* e *targets*, ma non dispone di campi di input.

Per i nodi di tipo *db2imtimeseriesnode* sono disponibili le seguenti proprietà.

| Proprietà <i>db2imtime-seriesnode</i> | Valori | Descrizione proprietà |
|---------------------------------------|---|--|
| <i>time</i> | <i>campo</i> | Sono consentiti <i>integer</i> , <i>time</i> o <i>date</i> . |
| <i>targets</i> | <i>elenco di campi</i> | |
| <i>forecasting_algorithm</i> | <i>arima</i> <i>exponential_smoothing</i> <i>seasonal_trend_decomposition</i> | |
| <i>forecasting_end_time</i> | <i>auto</i> <i>integer</i> <i>date</i> <i>time</i> | |
| <i>use_records_all</i> | <i>boolean</i> | Se <i>falsa</i> , è necessario impostare <i>use_records_start</i> e <i>use_records_end</i> . |
| <i>use_records_start</i> | <i>integer / time / date</i> | Dipende dal tipo di campo <i>time</i> |
| <i>use_records_end</i> | <i>integer / time / date</i> | Dipende dal tipo di campo <i>time</i> |
| <i>interpolation_method</i> | <i>none</i> <i>linear</i> <i>exponential_splines</i> <i>cubic_splines</i> | |

Proprietà degli insiemi di modelli IBM DB2

Le seguenti proprietà sono relative agli insiemi di modelli creati mediante i modelli IBM DB2 ISW.

Albero decisionale ISW

Non vi sono proprietà specifiche definite per i nodi di tipo *applydb2imtreenode*.

Associazione ISW

Questo insieme di modelli non può essere applicato negli script.

Sequenza ISW

Questo insieme di modelli non può essere applicato negli script.

Regressione ISW

Non vi sono proprietà specifiche definite per i nodi di tipo `applydb2imregnode`.

Raggruppamento cluster ISW

Non vi sono proprietà specifiche definite per i nodi di tipo `applydb2imclusternode`.

Bayes naive ISW

Non vi sono proprietà specifiche definite per i nodi di tipo `applydb2imnbnnode`.

Regressione logistica ISW

Non vi sono proprietà specifiche definite per i nodi di tipo `applydb2imlognode`.

Serie storica ISW

Questo insieme di modelli non può essere applicato negli script.

Proprietà dei nodi Modelli IBM Netezza Analytics**Proprietà dei nodi Modelli Netezza**

Le seguenti proprietà sono comuni ai nodi Modelli database IBM Netezza.

| Proprietà comuni dei nodi Netezza | Valori | Descrizione proprietà |
|--------------------------------------|----------------------------|---|
| <code>custom_fields</code> | <i>flag</i> | Se vera, consente di specificare i campi obiettivo, di input e di altro tipo per il nodo corrente. Se falsa, vengono utilizzate le impostazioni correnti di un nodo Tipo a monte. |
| <code>inputs</code> | <i>[campo1 ... campoN]</i> | I campi di input o predittore utilizzati dal modello. |
| <code>use_upstream_connection</code> | <i>flag</i> | Se vera (default) usa i dettagli di connessione specificati in un nodo a monte. Non utilizzato se è specificato <code>move_data_to_connection</code> . |
| <code>move_data_to_connection</code> | <i>flag</i> | Se vera, sposta i dati nel database specificato. Non utilizzato se è specificato <code>use_upstream_connection</code> . |
| <code>record_id</code> | <i>campo</i> | Campo da utilizzare come identificatore univoco del record. |
| <code>table_name</code> | <i>stringa</i> | Nome della tabella di database in cui sarà archiviato il modello. |
| <code>use_model_name</code> | <i>flag</i> | |
| <code>model_name</code> | <i>stringa</i> | Nome personalizzato per il nuovo modello. |
| <code>include_input_fields</code> | <i>flag</i> | |

Albero decisionale di Netezza

Per i nodi di tipo `netezzadectreenode` sono disponibili le seguenti proprietà.

| Proprietà <code>netezzadectreenode</code> | Valori | Descrizione proprietà |
|---|---|--|
| <code>target</code> | <i>campo</i> | Campo obiettivo (continuo o categoriale). |
| <code>impurity_measure</code> | Entropy Gini | |
| <code>max_tree_depth</code> | <i>intero</i> | Numero massimo di livelli di cui un albero può crescere. Il valore di default è 62 (il massimo possibile). |
| <code>min_improvement_splits</code> | <i>numero</i> | Miglioramento minimo in impurità perché si verifichi una suddivisione. L'impostazione di default è 0,01. |
| <code>min_instances_split</code> | <i>intero</i> | Numero minimo di record ancora da suddividere perché sia possibile effettuare una suddivisione. Il valore di default è 2 (il minimo possibile). |
| <code>weights</code> | <i>strutturato</i> | Ponderazioni relative per le classi. Proprietà strutturata nel formato: set :netezza_dectree.weights = [{drugA 0.3}{drugB 0.6}] L'impostazione di default è il peso 1 per tutte le classi. |
| <code>pruning_measure</code> | Acc wAcc | L'impostazione di default è Acc (precisione). In alternativa, wAcc (precisione ponderata) tiene conto dei pesi delle classi durante l'applicazione del taglio. |
| <code>prune_tree_options</code> | AllTrainingData partitionTrainingData useOtherTable | L'impostazione di default prevede l'utilizzo di AllTrainingData per stimare la precisione del modello. Specificare una percentuale dei dati di addestramento da utilizzare con partitionTrainingData, o specificare useOtherTable per utilizzare un insieme di dati di addestramento di una tabella di database specifica. |
| <code>compute_probabilities</code> | <i>flag</i> | |

Netezza K-Means

Per i nodi di tipo `netezzakmeansnode` sono disponibili le seguenti proprietà.

| Proprietà <code>netezzakmeansnode</code> | Valori | Descrizione proprietà |
|--|---|---|
| <code>distance_measure</code> | Euclidean Manhattan Canberra maximum | Metodo da utilizzare per misurare la distanza fra punti dei dati. |
| <code>num_clusters</code> | <i>intero</i> | Numero di cluster da creare; l'impostazione di default è 3. |
| <code>max_iterations</code> | <i>intero</i> | Numero di iterazioni dell'algoritmo dopo cui interrompere l'addestramento del modello; l'impostazione di default è 5. |
| <code>rand_seed</code> | <i>intero</i> | Seme aleatorio da utilizzare per replicare i risultati delle analisi; l'impostazione di default è 12345. |

Proprietà degli insiemi di modelli Netezza

Le seguenti proprietà sono comuni agli insiemi di modelli di database Netezza.

| Proprietà comuni degli insiemi di modelli Netezza | Valori | Descrizione proprietà |
|--|----------------|---|
| connection | <i>stringa</i> | La stringa di connessione per il database Netezza in cui è archiviato il modello. |
| model_name | <i>stringa</i> | Nome del modello. |
| table_name | <i>stringa</i> | Nome della tabella di database in cui è archiviato il modello. |

Le seguenti proprietà sono relative agli insiemi di modelli creati mediante i modelli Netezza.

Albero decisionale di Netezza

Non vi sono proprietà specifiche definite per i nodi di tipo `applynetezzadectreenode`.

Netezza K-Means

Non vi sono proprietà specifiche definite per i nodi di tipo `applynetezzakmeansnode`.

Proprietà dei nodi Output

Le proprietà del nodo Output sono leggermente diverse da quelle di altri tipi di nodi. Anziché fare riferimento all'opzione di un nodo specifico, le proprietà dei nodi Output consentono di memorizzare un riferimento all'oggetto di output. Ciò risulta utile per recuperare un valore da una tabella e impostarlo come un parametro dello stream.

In questa sezione vengono illustrate le proprietà degli script disponibili per i nodi Output.

Proprietà analysisnode



Il nodo Analisi valuta la capacità dei modelli predittivi di generare previsioni accurate. I nodi Analisi eseguono diversi confronti tra i valori previsti e i valori effettivi per uno o più insiemi di modelli. Possono inoltre confrontare i modelli predittivi fra loro. [Per ulteriori informazioni, vedere l'argomento nodo Analisi in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create analysisnode
#"Analysis" tab
set :analysisnode.coincidence = True
set :analysisnode.performance = True
set :analysisnode.confidence = True
set :analysisnode.threshold = 75
set :analysisnode.improve_accuracy = 3
set :analysisnode.inc_user_measure = True
#"Define User Measure..."
set :analysisnode.user_if = "@TARGET = @PREDICTED"
set :analysisnode.user_then = "101"
set :analysisnode.user_else = "1"
set :analysisnode.user_compute = [Mean Sum]
set :analysisnode.by_fields = ['Drug']
#"Output" tab
set :analysisnode.output_format = HTML
set :analysisnode.full_filename = "C:/output/analysis_out.html"
```

| proprietà analysisnode | Tipo di dati | Descrizione proprietà |
|------------------------|--------------|--|
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| use_output_name | flag | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | stringa | Se use_output_name è impostata su true (vero), specifica il nome da utilizzare. |

| proprietà analysisnode | Tipo di dati | Descrizione proprietà |
|------------------------|--|--|
| output_format | Text (.txt) HTML (.html) Output (.cou) | Utilizzata per specificare il tipo di output. |
| by_fields | [campo campo campo] | |
| full_filename | stringa | Se si tratta di output su disco, di dati o HTML, rappresenta il nome del file di output. |
| coincidence | flag | |
| performance | flag | |
| confidence | flag | |
| threshold | numero | |
| improve_accuracy | numero | |
| inc_user_measure | flag | |
| user_if | espr | |
| user_then | espr | |
| user_else | espr | |
| user_compute | [Mean Sum Min Max SDev] | |

Proprietà dataauditnode



Il nodo Esplora offre una prima panoramica completa dei dati, incluse statistiche riassuntive, istogrammi e distribuzione per ciascun campo, nonché informazioni su valori anomali, mancanti ed estremi. I risultati vengono visualizzati in una matrice di semplice lettura che può essere ordinata e utilizzata per generare grafici a schermo intero e nodi di preparazione dei dati. [Per ulteriori informazioni, vedere l'argomento Nodo Esplora in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create dataauditnode
connect :variablefilenode to :dataauditnode
set :dataauditnode.custom_fields = True
set :dataauditnode.fields = [Age Na K]
set :dataauditnode.display_graphs = True
set :dataauditnode.basic_stats = True
set :dataauditnode.advanced_stats = True
set :dataauditnode.median_stats = False
set :dataauditnode.calculate = [Count Breakdown]
set :dataauditnode.outlier_detection_method = std
set :dataauditnode.outlier_detection_std_outlier = 1.0
set :dataauditnode.outlier_detection_std_extreme = 3.0
set :dataauditnode.output_mode = Screen
```

| proprietà dataauditnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---------------------|-----------------------|
| custom_fields | flag | |
| fields | [campo1 ... campoN] | |

| proprietà dataauditnode | Tipo di dati | Descrizione proprietà |
|-------------------------------|---|---|
| overlay | <i>campo</i> | |
| display_graphs | <i>flag</i> | Utilizzato per attivare o disattivare la visualizzazione di grafici nella matrice di output. |
| basic_stats | <i>flag</i> | |
| advanced_stats | <i>flag</i> | |
| median_stats | <i>flag</i> | |
| calculate | Count Breakdown | Utilizzato per calcolare valori mancanti. Selezionare uno, entrambi o nessun metodo di calcolo. |
| outlier_detection_method | std iqr | Utilizzato per specificare il metodo di rilevamento dei valori anomali ed estremi. |
| outlier_detection_std_outlier | <i>numero</i> | Se outlier_detection_method è std, specifica il numero da utilizzare per definire i valori anomali. |
| outlier_detection_std_extreme | <i>numero</i> | Se outlier_detection_method è std, specifica il numero da utilizzare per definire i valori estremi. |
| outlier_detection_iqr_outlier | <i>numero</i> | Se outlier_detection_method è iqr, specifica il numero da utilizzare per definire i valori anomali. |
| outlier_detection_iqr_extreme | <i>numero</i> | Se outlier_detection_method è iqr, specifica il numero da utilizzare per definire i valori estremi. |
| use_output_name | <i>flag</i> | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | <i>stringa</i> | Se use_output_name è impostata su true (vero), specifica il nome da utilizzare. |
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | Formatted (.tab) Delimited (.csv) HTML (.html) Output (.cou) | Utilizzata per specificare il tipo di output. |
| paginate_output | <i>flag</i> | Quando output_format è HTML, l'output viene separato in pagine. |
| lines_per_page | <i>numero</i> | Se utilizzato con paginate_output, specifica le righe per pagina di output. |
| full_filename | <i>stringa</i> | |

Proprietà matrixnode



Il nodo Matrice crea una tabella che mostra le relazioni tra i campi. In genere viene utilizzato per mostrare le relazioni tra due campi simbolici, ma è possibile avvalersene anche per mostrare le relazioni tra campi flag o numerici. [Per ulteriori informazioni, vedere l'argomento Nodo Matrice in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create matrixnode
# "Settings" tab
set :matrixnode.fields = Numerics
set :matrixnode.row = 'K'
set :matrixnode.column = 'Na'
set :matrixnode.cell_contents = Function
set :matrixnode.function_field = 'Age'
set :matrixnode.function = Sum
# "Appearance" tab
set :matrixnode.sort_mode = Ascending
set :matrixnode.highlight_top = 1
set :matrixnode.highlight_bottom = 5
set :matrixnode.display = [Counts Expected Residuals]
set :matrixnode.include_totals = True
# "Output" tab
set :matrixnode.full_filename = "C:/output/matrix_output.html"
set :matrixnode.output_format = HTML
set :matrixnode.paginate_output = true
set :matrixnode.lines_per_page = 50
```

| proprietà matrixnode | Tipo di dati | Descrizione proprietà |
|------------------------|-------------------------------------|--|
| fields | Selected Flags Numerics | |
| row | <i>campo</i> | |
| column | <i>campo</i> | |
| include_missing_values | <i>flag</i> | Specifica se i valori mancanti definiti dall'utente (vuoti) e i valori mancanti di sistema (nulli) sono inclusi nell'output delle righe e delle colonne. |
| cell_contents | CrossTabs Function | |
| function_field | <i>stringa</i> | |
| function | Sum Mean Min Max SDev | |
| sort_mode | Unsorted Ascending Descending | |

| proprietà matrixnode | Tipo di dati | Descrizione proprietà |
|----------------------|---|---|
| highlight_top | numero | Se diversa da zero, è vera. |
| highlight_bottom | numero | Se diversa da zero, è vera. |
| display | [Counts Expected Residuals RowPct ColumnPct TotalPct] | |
| include_totals | flag | |
| use_output_name | flag | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | stringa | Se use_output_name è impostata su true (vero), specifica il nome da utilizzare. |
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | Formatted (.tab) Delimited (.csv) HTML (.html) Output (.cou) | Utilizzata per specificare il tipo di output. Sia il formato Formatted che il formato Delimited possono recare il modificatore transposed, che traspone le righe e le colonne nella tabella, per esempio: NODE.output_format=transposed Delimited |
| paginate_output | flag | Quando output_format è HTML, l'output viene separato in pagine. |
| lines_per_page | numero | Se utilizzato con paginate_output, specifica le righe per pagina di output. |
| full_filename | stringa | |

Proprietà meansnode



Il nodo Medie confronta le medie tra gruppi indipendenti o coppie di campi correlati per verificare se esiste una differenza significativa. Per esempio, è possibile confrontare le entrate medie prima e dopo il lancio di una promozione, oppure confrontare le entrate determinate da clienti che non hanno ricevuto la promozione con quelli che l'hanno ricevuta. [Per ulteriori informazioni, vedere l'argomento Nodo Medie in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create meansnode
set :meansnode.means_mode = BetweenFields
set :meansnode.paired_fields = [{'OPEN_BAL' 'CURR_BAL'}]
set :meansnode.label_correlations = true
set :meansnode.output_view = Advanced
set :meansnode.output_mode = File
```

```
set :meansnode.output_format = HTML
set :meansnode.full_filename = "C:/output/means_output.html"
```

| proprietà meansnode | Tipo di dati | Descrizione proprietà |
|--------------------------|---|---|
| means_mode | BetweenGroups BetweenFields | Specifica il tipo di medie statistiche da eseguire sui dati. |
| test_fields | [field1 ... fieldn] | Specifica il campo di verifica quando means_mode è impostato su BetweenGroups. |
| grouping_field | campo | Specifica il campo di raggruppamento. |
| paired_fields | {{field1 field2} {field3 field4} ...} | Specifica le coppie di campi da utilizzare quando means_mode è impostato su BetweenFields. |
| label_correlations | flag | Specifica se le etichette di correlazione devono essere visualizzate nell'output. Questa impostazione si applica solo quando means_mode è impostato su BetweenFields. |
| correlation_mode | Probability Absolute | Specifica se etichettare le correlazioni per probabilità o valore assoluto. |
| weak_label | stringa | |
| medium_label | stringa | |
| strong_label | stringa | |
| weak_below_probability | numero | Quando correlation_mode è impostata su Probability, specifica il valore di interruzione per le correlazioni deboli. Deve essere un valore compreso tra 0 e 1, per esempio 0.90. |
| strong_above_probability | numero | Valore di interruzione per correlazioni forti. |
| weak_below_absolute | numero | Quando correlation_mode è impostata su Absolute, specifica il valore di interruzione per le correlazioni deboli. Deve essere un valore compreso tra 0 e 1, per esempio 0.90. |
| strong_above_absolute | numero | Valore di interruzione per correlazioni forti. |
| unimportant_label | stringa | |
| marginal_label | stringa | |
| important_label | stringa | |
| unimportant_below | numero | Valore di interruzione per importanza di campo bassa. Deve essere un valore compreso tra 0 e 1, per esempio 0.90. |
| important_above | numero | |
| use_output_name | flag | Specifica se viene utilizzato un nome di output personalizzato. |

| proprietà meansnode | Tipo di dati | Descrizione proprietà |
|---------------------|---|---|
| output_name | stringa | Nome da utilizzare. |
| output_mode | Screen File | Specifica la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | Formatted (.tab) Delimited (.csv) HTML (.html) Output (.cou) | Specifica il tipo di output. |
| full_filename | stringa | |
| output_view | Simple Advanced | Specifica se l'output deve presentare la visualizzazione di base o avanzata. |

Proprietà reportnode



Il nodo Report crea report formattati che contengono sia testo fisso sia dati e altre espressioni derivate dai dati. Il formato del report viene specificato utilizzando modelli di testo per definire il testo fisso e costruzioni di output dei dati. È possibile fornire una formattazione personalizzata del testo utilizzando tag HTML nel modello e impostando apposite opzioni nella scheda Output. È possibile includere valori di dati e altro output condizionale utilizzando espressioni CLEM nel modello. [Per ulteriori informazioni, vedere l'argomento Nodo Report in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create reportnode
set :reportnode.output_format = HTML
set :reportnode.full_filename = "C:/report_output.html"
set :reportnode.lines_per_page = 50
set :reportnode.title = "Report node created by a script"
set :reportnode.highlights = False
```

| proprietà reportnode | Tipo di dati | Descrizione proprietà |
|----------------------|--|--|
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | HTML (.html) Text (.txt) Output (.cou) | Utilizzata per specificare il tipo di output. |
| use_output_name | flag | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | stringa | Se use_output_name è impostata su true (vero), specifica il nome da utilizzare. |
| text | stringa | |
| full_filename | stringa | |
| highlights | flag | |
| title | stringa | |
| lines_per_page | numero | |

Proprietà setglobalsnode



Il nodo Globali analizza i dati e calcola i valori di riepilogo che possono essere utilizzati nelle espressioni CLEM. Per esempio, è possibile utilizzare questo nodo per calcolare le statistiche di un campo denominato *età* e utilizzare quindi la media globale dell'*età* nelle espressioni CLEM inserendo la funzione @GLOBAL_MEAN(*età*). Per ulteriori informazioni, vedere l'argomento [Nodo Globali in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create setglobalsnode
connect :typenode to :setglobalsnode
set :setglobalsnode.globals.Na = [Max Sum Mean]
set :setglobalsnode.globals.K = [Max Sum Mean]
set :setglobalsnode.globals.Age = [Max Sum Mean SDev]
set :setglobalsnode.clear_first = False
set :setglobalsnode.show_preview = True
```

| proprietà setglobalsnode | Tipo di dati | Descrizione proprietà |
|--------------------------|-------------------------|---|
| globals | [Sum Mean Min Max SDev] | Proprietà strutturata nella quale per fare riferimento ai campi da impostare è necessario utilizzare la sintassi seguente: set :setglobalsnode.globals.Age = [Sum Mean Min Max SDev] |
| clear_first | flag | |
| show_preview | flag | |

Proprietà statisticsnode



Il nodo Statistiche fornisce informazioni riassuntive di base su campi numerici. Calcola statistiche riassuntive per singoli campi e per correlazioni tra campi. Per ulteriori informazioni, vedere l'argomento [Nodo Statistiche in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create statisticsnode
# "Settings" tab
set :statisticsnode.examine = ['Age' 'BP' 'Drug']
set :statisticsnode.statistics = [Mean Sum SDev]
set :statisticsnode.correlate = ['BP' 'Drug']
# "Correlation Labels..." section
set :statisticsnode.label_correlations = True
set :statisticsnode.weak_below_absolute = 0.25
set :statisticsnode.weak_label = "lower quartile"
set :statisticsnode.strong_above_absolute = 0.75
set :statisticsnode.medium_label = "middle quartiles"
set :statisticsnode.strong_label = "upper quartile"
# "Output" tab
```

```
set :statisticsnode.full_filename = "c:/output/statistics_output.html"
set :statisticsnode.output_format = HTML
```

| proprietà statisticsnode | Tipo di dati | Descrizione proprietà |
|--------------------------|--|---|
| use_output_name | <i>flag</i> | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | <i>stringa</i> | Se use_output_name è impostata su true (vero), specifica il nome da utilizzare. |
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | Text (.txt) HTML (.html) Output (.cou) | Utilizzata per specificare il tipo di output. |
| full_filename | <i>stringa</i> | |
| examine | [<i>campo campo campo</i>] | |
| correlate | [<i>campo campo campo</i>] | |
| statistics | [Count Mean Sum Min Max Range Variance SDev SErr Median Mode] | |
| correlation_mode | Probability Absolute | Specifica se etichettare le correlazioni per probabilità o valore assoluto. |
| label_correlations | <i>flag</i> | |
| weak_label | <i>stringa</i> | |
| medium_label | <i>stringa</i> | |
| strong_label | <i>stringa</i> | |
| weak_below_probability | <i>numero</i> | Quando correlation_mode è impostata su Probability, specifica il valore di interruzione per le correlazioni deboli. Deve essere un valore compreso tra 0 e 1, per esempio 0.90. |
| strong_above_probability | <i>numero</i> | Valore di interruzione per correlazioni forti. |
| weak_below_absolute | <i>numero</i> | Quando correlation_mode è impostata su Absolute, specifica il valore di interruzione per le correlazioni deboli. Deve essere un valore compreso tra 0 e 1, per esempio 0.90. |
| strong_above_absolute | <i>numero</i> | Valore di interruzione per correlazioni forti. |

Proprietà *statisticsoutputnode*



Il nodo Output Statistics consente di chiamare una procedura IBM® SPSS® Statistics per analizzare i dati di IBM® SPSS® Modeler. È disponibile una vasta gamma di procedure analitiche di SPSS Statistics. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. [Per ulteriori informazioni, vedere l'argomento Nodo Output Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Le proprietà di questo nodo sono descritte in [Proprietà statisticsoutputnode a pag. 297.](#)

Proprietà *tablenode*



Il nodo Tabella visualizza i dati in formato tabella, che è inoltre possibile scrivere su un file. Questa funzione è utile tutte le volte che si desidera controllare i valori dei dati o esportarli in un formato di facile lettura. [Per ulteriori informazioni, vedere l'argomento Nodo Tabella in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create tablenode
set :tablenode.highlight_expr = "Age > 30"
set :tablenode.output_format = HTML
set :tablenode.transpose_data = true
set :tablenode.full_filename = "C:/output/table_output.htm"
set :tablenode.paginate_output = true
set :tablenode.lines_per_page = 50
```

| proprietà <i>tablenode</i> | Tipo di dati | Descrizione proprietà |
|----------------------------|---|---|
| full_filename | <i>stringa</i> | Se si tratta di output su disco, di dati o HTML, rappresenta il nome del file di output. |
| use_output_name | <i>flag</i> | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | <i>stringa</i> | Se <i>use_output_name</i> è impostata su true (vero), specifica il nome da utilizzare. |
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | Formatted (.tab) Delimited (.csv) HTML (.html) Output (.cou) | Utilizzata per specificare il tipo di output. |
| transpose_data | <i>flag</i> | Traspone i dati prima dell'esportazione in modo che le righe rappresentino i campi e le colonne rappresentino i record. |
| paginate_output | <i>flag</i> | Quando <i>output_format</i> è HTML, l'output viene separato in pagine. |

| proprietà tablenode | Tipo di dati | Descrizione proprietà |
|---------------------|---|--|
| lines_per_page | numero | Se utilizzato con paginate_output, specifica le righe per pagina di output. |
| highlight_expr | stringa | |
| output | stringa | Proprietà di sola lettura che restituisce un riferimento all'ultima tabella creata dal nodo. |
| value_labels | <i>[[Valore EtichettaStringa] {Valore EtichettaStringa} ...]</i> | Utilizzata per specificare etichette per coppie di valori. Per esempio, set :typenode.value_labels. 'Drug'={{drugA label1} {drugB label2}} |
| display_places | intero | Imposta il numero di decimali del campo per la visualizzazione (valida solo per campi con archiviazione di tipo Reale). Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.display_places. FIELDNAME |
| export_places | intero | Imposta il numero di decimali del campo per l'esportazione (valida solo per campi con archiviazione di tipo Reale). Se viene specificato il valore -1, verrà utilizzata l'impostazione di default dello stream. Formato di utilizzo: NODE.export_places.FIELDNAME |
| decimal_separator | DEFAULT PERIOD COMMA | Imposta il separatore decimale per il campo (valido solo per campi con archiviazione di tipo Reale). Formato di utilizzo: NODE.decimal_separator. FIELDNAME |
| date_format | "GGMMAA" "MMGGAA" "AAMMGG" "AAAAMMGG" "AAAAGGG" GIORNO MESE "GG-MM-AA" "GG-MM-AAAA" "MM-GG-AA" "MM-GG-AAAA" "GG-MES-AA" "GG-MES-AAAA" "AAAA-MM-GG" "GG.MM.AA" "GG.MM.AAAA" "MM.GG.AA" "MM.GG.AAAA" "GG.MES.AA" "GG.MES.AAAA" | Imposta il formato della data per il campo (valida solo per i campi con archiviazione di tipo DATE o TIMESTAMP). Formato di utilizzo: NODE.date_format.FIELDNAME Per esempio, set :tablenode.date_format. 'LaunchDate' = "DDMMYY" |

| proprietà tablenode | Tipo di dati | Descrizione proprietà |
|---------------------|--|--|
| | "GG/MM/AA" "GG/MM/AAAA" "MM/GG/AA" "MM/GG/AAAA" "GG/MES/AA" "GG/MES/AAAA" MES AAAA t T AAAA ss ST AAAA | |
| time_format | "HHMMSS" "HHMM" "MMSS" "HH:MM:SS" "HH:MM" "MM:SS" "(H)H:(M)M:(S)S" "(H)H:(M)M" "(M)M:(S)S" "HH.MM.SS" "HH.MM" "MM.SS" "(H)H.(M)M.(S)S" "(H)H.(M)M" "(M)M.(S)S" | Imposta il formato dell'ora per il campo (valida solo per i campi con archiviazione di tipo TIME o TIMESTAMP). Formato di utilizzo: NODE.time_format.FIELDNAME Per esempio, set :tablenode.time_format. set 'BOF_enter' = "HHMMSS" |
| column_width | <i>intero</i> | Imposta la larghezza delle colonne per il campo. Se viene specificato il valore -1, la larghezza delle colonne verrà impostata su Auto. Formato di utilizzo: NODE.column_width.FIELDNAME |
| justify | AUTO CENTER LEFT RIGHT | Imposta la giustificazione delle colonne per il campo. Formato di utilizzo: NODE.justify.FIELDNAME |

Proprietà transformnode



Il nodo Trasformazioni consente di selezionare e visualizzare in anteprima i risultati di trasformazioni prima di applicarli ai campi selezionati. [Per ulteriori informazioni, vedere l'argomento Nodo Trasformazioni in il capitolo 6 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create transformnode
set :transformnode.fields = [AGE INCOME]
set :transformnode.formula = Select
set :transformnode.formula_log_n = true
```

```
set :transformnode.formula_log_n_offset = 1
```

| proprietà transformnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---|---|
| fields | [<i>campo1</i> ... <i>campon</i>] | I campi da utilizzare nella trasformazione. |
| formula | All Select | Indica se devono essere calcolate tutte le trasformazioni o solo quelle selezionate. |
| formula_inverse | <i>flag</i> | Indica se deve essere utilizzata la trasformazione inversa. |
| formula_inverse_offset | <i>numero</i> | Indica la distanza tra i dati da utilizzare per la formula. Se non è specificata dall'utente, è impostata su 0 per default. |
| formula_log_n | <i>flag</i> | Indica se deve essere utilizzata la trasformazione \log_n . |
| formula_log_n_offset | <i>numero</i> | |
| formula_log_10 | <i>flag</i> | Indica se deve essere utilizzata la trasformazione \log_{10} . |
| formula_log_10_offset | <i>numero</i> | |
| formula_exponential | <i>flag</i> | Indica se deve essere utilizzata la trasformazione esponenziale (e^x). |
| formula_square_root | <i>flag</i> | Indica se deve essere utilizzata la trasformazione radice quadrata. |
| use_output_name | <i>flag</i> | Specifica se viene utilizzato un nome di output personalizzato. |
| output_name | <i>stringa</i> | Se <i>use_output_name</i> è impostata su vero, specifica il nome da utilizzare. |
| output_mode | Screen File | Utilizzata per specificare la posizione di destinazione per l'output generato dal nodo Output. |
| output_format | HTML (<i>.html</i>) Output (<i>.cou</i>) | Utilizzata per specificare il tipo di output. |
| paginate_output | <i>flag</i> | Quando <i>output_format</i> è HTML, l'output viene separato in pagine. |
| lines_per_page | <i>numero</i> | Se utilizzato con <i>paginate_output</i> , specifica le righe per pagina di output. |
| full_filename | <i>stringa</i> | Indica il nome di file da utilizzare per l'output su file. |

Proprietà dei nodi di esportazione

Proprietà comuni dei nodi di esportazione

Le seguenti proprietà sono valide per tutti i nodi di esportazione:

| Proprietà | Valori | Descrizione proprietà |
|------------------------|------------------------|--|
| publish_path | stringa | Specificare il nome di base da utilizzare per i file immagine e dei parametri pubblicati. |
| publish_metadata | flag | Specifica se viene generato un file di metadati che descrive gli input e gli output dell'immagine e dei rispettivi modelli di dati. |
| publish_use_parameters | flag | Specifica se i parametri dello stream sono contenuti nel file *.par. |
| publish_parameters | elenco di stringhe | Specifica i parametri da includere. |
| execute_mode | export_data publish | Specifica se il nodo viene eseguito senza pubblicare lo stream o se lo stream viene pubblicato automaticamente quando si esegue il nodo. |

Proprietà del nodo di esportazione Cognos



Il nodo di esportazione IBM Cognos BI esporta i dati in un formato leggibile dai database Cognos BI. [Per ulteriori informazioni, vedere l'argomento Nodo di esportazione IBM Cognos BI in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Nota: Per questo nodo è necessario definire una connessione Cognos e una connessione ODBC.

Connessione Cognos

Le proprietà della connessione Cognos sono le seguenti.

| proprietà cognosexportnode | Tipo di dati | Descrizione proprietà |
|----------------------------|--|---|
| cognos_connection | { "field", "field", ... , "field" } | Una proprietà elenco contenente i dettagli di connessione per il server Cognos. Il formato è: { "URL_server_Cognos", modalità_accesso, "spaziodeinomi", "nomeutente", "password*" } dove: URL_server_Cognos è l'URL del server Cognos in cui si sta esportando modalità_accesso indica se viene utilizzato un accesso anonimo e può essere true o |

| proprietà cognosexportnode | Tipo di dati | Descrizione proprietà |
|----------------------------|-----------------------|---|
| | | false; se impostato su true, i seguenti campi devono essere impostati su "" <i>spaziodeinomi</i> specifica il provider di protezione per l'autenticazione utilizzato per accedere al server. <i>nomeutente</i> e <i>password</i> sono i dati utilizzati per accedere al server Cognos |
| cognos_package_name | stringa | Il percorso e il nome del pacchetto Cognos in cui esportare i dati, per esempio: /Public Folders/MyPackage |
| cognos_datasource | stringa | |
| cognos_export_mode | Publish ExportFile | |
| cognos_filename | stringa | |

Connessione ODBC

Le proprietà della connessione ODBC sono identiche a quelle riportate per `databaseexportnode` nella sezione che segue, con la differenza che la proprietà `datasource` non è valida.

Proprietà `databaseexportnode`



Il nodo Database scrive dati in una sorgente dati relazionale compatibile con ODBC. Per scrivere in una sorgente dati ODBC, è necessario utilizzare una sorgente dati esistente e disporre dell'autorizzazione in scrittura per tale sorgente. [Per ulteriori informazioni, vedere l'argomento Nodo di esportazione Database in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
/*
Use this sample with fraud.str from demo folder
Assumes a datasource named "MyDatasource" has been configured
*/
create databaseexport
connect claimvalue:applyneuralnetwork to :databaseexport
# Export tab
set :databaseexport.username = "user"
set :databaseexport.datasource = "MyDatasource"
set :databaseexport.password = "password"
set :databaseexport.table_name = "predictions"
set :databaseexport.write_mode = Create
set :databaseexport.generate_import = true
set :databaseexport.drop_existing_table = true
set :databaseexport.delete_existing_rows = true
set :databaseexport.default_string_size = 32

# Schema dialog
set :databaseexport.type.region = "VARCHAR(10)"
set :databaseexport.export_db_primarykey.id = true
```

```
set :databaseexportnode.use_custom_create_table_command = true
set :databaseexportnode.custom_create_table_command = "My SQL Code"
```

```
# Indexes dialog
```

```
set :databaseexport.use_custom_create_index_command = true
set :databaseexport.custom_create_index_command = \
  "CREATE BITMAP INDEX <index-name> ON <table-name> <{index-columns}>"
set :databaseexport.indexes.MYINDEX.fields = [id region]
```

| proprietà databaseexportnode | Tipo di dati | Descrizione proprietà |
|------------------------------|---------------------------|--|
| datasource | stringa | |
| username | stringa | |
| password | stringa | |
| epassword | stringa | Questa configurazione è di sola lettura durante l'esecuzione. Per generare una password codificata, utilizzare lo strumento Password disponibile dal menu Strumenti. Per ulteriori informazioni, vedere l'argomento Generazione di una password codificata in il capitolo 5 a pag. 60. |
| table_name | stringa | |
| write_mode | Create Append Merge | |
| map | stringa | Mappa il nome campo di uno stream al nome di una colonna di database (valido solo se write_mode è Merge). Esempio: set :databaseexportnode.map.streamBP = 'databaseBP' La mappatura multipla è supportata, in base alla posizione del campo, per esempio: set :databaseexportnode.map.[streamfield1 streamfield2 streamfield3] = [field1 field2 field3] In caso di unione è necessario che tutti i campi siano mappati per poter essere esportati. I nomi di campi che non esistono nel database vengono aggiunti come nuove colonne. |
| key_fields | [campo campo... campo] | Specifica il campo dello stream utilizzato come chiave; la proprietà map mostra a che cosa corrisponde nel database. |

| proprietà databaseexportnode | Tipo di dati | Descrizione proprietà |
|---------------------------------|-------------------------|---|
| join | Database Add | Esempio: set : databaseexportnode..join = Database |
| drop_existing_table | flag | |
| delete_existing_rows | flag | |
| default_string_size | intero | |
| type | | Proprietà strutturata utilizzata per impostare il tipo di schema. Formato di utilizzo: set :databaseexportnode. type.BP = 'VARCHAR(10)' |
| generate_import | flag | |
| use_custom_create_table_command | flag | Utilizzare la configurazione <i>custom_create_table</i> per modificare il comando SQL standard CREATE TABLE. |
| custom_create_table_command | stringa | Specifica un comando stringa da utilizzare al posto del comando SQL standard CREATE TABLE. |
| use_batch | flag | Le seguenti proprietà sono opzioni avanzate per il caricamento di massa di database. Un valore vero per Use_batch disattiva riga per riga i commit al database. |
| batch_size | numero | Specifica il numero di record da inviare al database prima del commit nella memoria. |
| bulk_loading | Off ODBC External | Specifica il tipo di caricamento di massa. Di seguito vengono elencate opzioni aggiuntive per ODBC ed External. |
| odbc_binding | Row Column | Specificare l'associazione in base a righe o colonne per il caricamento di massa tramite ODBC. |
| loader_delimit_mode | Tab Space Other | Specificare il tipo di delimitatore per il caricamento di massa tramite un programma esterno. Selezionare Other insieme alla proprietà loader_other_delimiter per specificare i delimitatori, quali la virgola (,). |
| loader_other_delimiter | stringa | |
| specify_data_file | flag | Un flag true attiva la proprietà data_file seguente, che consente di specificare il nome e il percorso del file in cui scrivere durante il caricamento di massa nel database. |
| data_file | stringa | |

| proprietà databaseexportnode | Tipo di dati | Descrizione proprietà |
|---|---------------------|--|
| specify_loader_program | <i>flag</i> | Un flag vero attiva la proprietà loader_program illustrata di seguito, che consente di specificare il nome e la posizione di un programma o di uno script di caricamento esterno. |
| loader_program | <i>stringa</i> | |
| gen_logfile | <i>flag</i> | Un flag vero attiva la proprietà logfile_name illustrata di seguito, che consente di specificare il nome di un file sul server per generare un registro errori. |
| logfile_name | <i>stringa</i> | |
| check_table_size | <i>flag</i> | Un flag vero consente il controllo della tabella che assicura che l'incremento nelle dimensioni della tabella del database corrisponda al numero di righe esportate da IBM® SPSS® Modeler. |
| loader_options | <i>stringa</i> | Specificare argomenti aggiuntivi, quali -comment e -specialdir, al programma di caricamento. |
| export_db_primarykey | <i>flag</i> | Specifica se un determinato campo è una chiave principale. |
| use_custom_create_index_command | <i>flag</i> | Se true, (vero), attiva l'SQL personalizzato per tutti gli indici. |
| custom_create_index_command | <i>stringa</i> | Specifica il comando SQL utilizzato per creare gli indici quando è attivato l'SQL personalizzato. (Questo valore può essere sovrascritto per indici specifici come indicato di seguito). |
| indexes.INDEXNAME.fields | | Crea l'indice specificato, se necessario, ed elenca i nomi dei campi da includere in tale indice. |
| indexes.INDEXNAME.use_custom_create_index_command | <i>flag</i> | Utilizzato per attivare o disattivare l'SQL personalizzato per un indice specifico. |
| indexes.INDEXNAME.custom_create_command | | Specifica l'SQL personalizzato utilizzato per l'indice specificato. |
| indexes.INDEXNAME.remove | <i>flag</i> | Se true, (vero), rimuove l'indice specificato dall'insieme di indici. |

Proprietà *datacollectionexportnode*



Il nodo Esportazione di IBM® SPSS® Data Collection esegue l'output di dati nel formato utilizzato dal software di ricerche di mercato Data Collection. Per utilizzare questo nodo, è necessario che sia installata Data Collection Data Library. [Per ulteriori informazioni, vedere l'argomento Nodo Esporta IBM SPSS Data Collection in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create datacollectionexportnode
set :datacollectionexportnode.metadata_file = "c:\museums.mdd"
set :datacollectionexportnode.merge_metadata = Overwrite
set :datacollectionexportnode.casedata_file = "c:\museumdata.sav"
set :datacollectionexportnode.generate_import = true
set :datacollectionexportnode.enable_system_variables = true
```

| proprietà <i>datacollectionexportnode</i> | Tipo di dati | Descrizione proprietà |
|---|---------------------------|---|
| metadata_file | <i>stringa</i> | Nome del file di metadati da esportare. |
| merge_metadata | Overwrite MergeCurrent | |
| enable_system_variables | <i>flag</i> | Specifica se il file <i>.mdd</i> esportato deve includere le variabili di sistema di Data Collection. |
| casedata_file | <i>stringa</i> | Il nome del file <i>.sav</i> in cui vengono esportati i dati del caso. |
| generate_import | <i>flag</i> | |

Proprietà *excelexportnode*



Il nodo Esportazione da Excel esegue l'output di dati in formato Microsoft Excel (*.xls*). Se lo si desidera, è possibile scegliere di avviare Excel automaticamente e aprire il file esportato quando si esegue il nodo. [Per ulteriori informazioni, vedere l'argomento Nodo Esportazione da Excel in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create excelexportnode
set :excelexportnode.full_filename = "C:/output/myexport.xls"
set :excelexportnode.excel_file_type = Excel2007
set :excelexportnode.inc_field_names = True
set :excelexportnode.inc_labels_as_cell_notes = False
set :excelexportnode.launch_application = True
```

```
set :excelexportnode.generate_import = True
```

| proprietà excelexportnode | Tipo di dati | Descrizione proprietà |
|---------------------------|------------------------|---|
| full_filename | stringa | |
| excel_file_type | Excel2003 Excel2007 | |
| export_mode | Create Append | |
| inc_field_names | flag | Specifica se i nomi dei campi devono essere inclusi nella prima riga del foglio di lavoro. |
| start_cell | stringa | Specifica la cella di partenza per l'esportazione. |
| worksheet_name | stringa | Nome del foglio di lavoro da scrivere. |
| launch_application | flag | Specifica se sul file risultante deve essere richiamato Excel. Si noti che il percorso per avviare Excel deve essere specificato nella finestra di dialogo Applicazioni di supporto (menu Strumenti, Applicazioni di supporto). |
| generate_import | flag | Specifica se deve essere generato un nodo Importazione da Excel che legga il file di dati esportato. |

Proprietà outputfilenode



Il nodo di esportazione File piatto restituisce dati in un file di testo delimitato. È utile per esportare i dati che possono essere letti da altri software di analisi o fogli di calcolo. [Per ulteriori informazioni, vedere l'argomento Nodo di esportazione File piatto in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create outputfile
set :outputfile.full_filename = "c:/output/flatfile_output.txt"
set :outputfile.write_mode = Append
set :outputfile.inc_field_names = False
set :outputfile.use_newline_after_records = False
set :outputfile.delimit_mode = Tab
set :outputfile.other_delimiter = ";"
set :outputfile.quote_mode = Double
set :outputfile.other_quote = "*"
set :outputfile.decimal_symbol = Period
```

```
set :outputfile.generate_import = True
```

| proprietà outputfilenode | Tipo di dati | Descrizione proprietà |
|---------------------------|---|--------------------------|
| full_filename | <i>stringa</i> | Nome del file di output. |
| write_mode | Overwrite Append | |
| inc_field_names | <i>flag</i> | |
| use_newline_after_records | <i>flag</i> | |
| delimit_mode | Comma Tab Space Other | |
| other_delimiter | <i>car</i> | |
| quote_mode | None Single Double Other | |
| other_quote | <i>flag</i> | |
| generate_import | <i>flag</i> | |
| encoding | StreamDefault SystemDefault "UTF-8" | |

Proprietà sasexportnode



Il nodo Esporta SAS restituisce nel formato SAS i dati che devono essere letti in SAS o in un pacchetto software compatibile con SAS. I formati di file SAS disponibili sono tre: SAS per Windows/OS2, SAS per UNIX o SAS Versione 7/8. [Per ulteriori informazioni, vedere l'argomento Nodo Esporta SAS in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create sasexportnode
set :sasexportnode.full_filename = "c:/output/SAS_output.sas7bdat"
set :sasexportnode.format = SAS8
set :sasexportnode.export_names = NamesAndLabels
set :sasexportnode.generate_import = True
```

| proprietà sasexportnode | Tipo di dati | Descrizione proprietà |
|-------------------------|---------------------------------|---|
| format | Windows UNIX SAS7 SAS8 | Campi delle etichette delle proprietà Variant. |
| full_filename | <i>stringa</i> | |
| export_names | NamesAndLabels NamesAsLabels | Utilizzata per mappare nomi di campi IBM® SPSS® Modeler a nomi di variabili IBM® SPSS® Statistics o SAS durante l'esportazione. |
| generate_import | <i>flag</i> | |

Proprietà *statisticsexportnode*



Il nodo Esporta Statistics restituisce i dati nel formato *.sav* di IBM® SPSS® Statistics. I file *.sav* possono essere letti da SPSS Statistics Base e altri prodotti. Questo formato viene inoltre utilizzato per i file cache di IBM® SPSS® Modeler. [Per ulteriori informazioni, vedere l'argomento Nodo Esporta Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Le proprietà di questo nodo sono descritte in [Proprietà *statisticsexportnode* a pag. 297.](#)

Proprietà *xmlexportnode*



Il nodo Esporta XML restituisce i dati in un file in formato XML. Se lo si desidera, è possibile creare un nodo di input XML per leggere nuovamente i dati esportati nello stream. [Per ulteriori informazioni, vedere l'argomento Nodo Esporta XML in il capitolo 7 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create xmlexportnode
set :xmlexportnode.full_filename = "c:\export\data.xml"
set :xmlexportnode.map = [{"/catalog/book/genre" genre} {"/catalog/book/title" title}]
```

| proprietà <i>xmlexportnode</i> | Tipo di dati | Descrizione proprietà |
|-----------------------------------|----------------|---|
| <code>full_filename</code> | <i>stringa</i> | (obbligatorio) Percorso e nome file completi del file di esportazione XML. |
| <code>use_xml_schema</code> | <i>flag</i> | Specifica se utilizzare uno schema XML (file XSD o DTD) per controllare la struttura dei dati esportati. |
| <code>full_schema_filename</code> | <i>stringa</i> | Percorso e nome file completi del file XSD o DTD da utilizzare. Necessario solo se <code>use_xml_schema</code> è impostato su <code>true</code> (vero). |
| <code>generate_import</code> | <i>flag</i> | Genera un nodo di input XML che rileggerà il file di dati esportato nello stream. |
| <code>records</code> | <i>stringa</i> | Espressione XPath che indica i limiti dei record. |
| <code>map</code> | <i>stringa</i> | Mappa i nomi dei campi alla struttura XML. Esempio: <code>set :xmlexportnode.map = [{"/top/node1" field1} {"/top/node2" field2}]</code> Il campo dello stream <code>field1</code> viene mappato all'elemento XML <code>/top/node1</code> e così via. |

Proprietà dei nodi IBM SPSS Statistics

Proprietà *statisticsimportnode*



Il nodo File Statistics legge i dati dal formato di file *.sav* utilizzato da IBM® SPSS® Statistics, nonché da file della cache salvati in IBM® SPSS® Modeler, che utilizzano lo stesso formato. [Per ulteriori informazioni, vedere l'argomento Nodo File Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create statisticsimportnode
set :statisticsimportnode.full_filename = "C:/data/drug1n.sav"
set :statisticsimportnode.import_names = true
set :statisticsimportnode.import_data = true
```

| Proprietà <i>statisticsimportnode</i> | Tipo di dati | Descrizione proprietà |
|---------------------------------------|---------------------------------|---|
| full_filename | <i>stringa</i> | Il nome del file completo compreso il percorso. |
| import_names | NamesAndLabels LabelsAsNames | Metodo per gestire nomi ed etichette di variabili. |
| import_data | DataAndLabels LabelsAsData | Metodo per gestire valori ed etichette. |
| use_field_format_for_storage | <i>Booleana</i> | Specifica se utilizzare le informazioni relative al formato dei campi di SPSS Statistics durante le importazioni. |

Proprietà *statisticstransformnode*



Il nodo Trasformazioni Statistics esegue una selezione di comandi di sintassi IBM® SPSS® Statistics rispetto alle sorgenti dati in IBM® SPSS® Modeler. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. [Per ulteriori informazioni, vedere l'argomento Nodo Trasformazioni Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create statisticstransformnode
set :statisticstransformnode.syntax = "COMPUTE NewVar = Na + K."
set :statisticstransformnode.new_name.NewVar = "Mixed Drugs"
```

```
set :statisticstransformnode.check_before_saving = true
```

| Proprietà statisticstransformnode | Tipo di dati | Descrizione proprietà |
|-----------------------------------|--------------|---|
| syntax | stringa | |
| check_before_saving | flag | Convalida la sintassi inserita prima di salvare le voci. Visualizza un messaggio di errore se la sintassi non è valida. |
| default_include | flag | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |
| include | flag | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |
| new_name | stringa | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |

Proprietà statisticsmodelnode



Il nodo Modello Statistics consente di analizzare e operare con i dati eseguendo le procedure IBM® SPSS® Statistics che generano PMML. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. Per ulteriori informazioni, vedere l'argomento Nodo Modello Statistics in il capitolo 8 in *IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output*.

Esempio

```
create statisticsmodelnode
set :statisticsmodelnode.syntax = "COMPUTE NewVar = Na + K."
set :statisticsmodelnode.new_name.NewVar = "Mixed Drugs"
```

| Proprietà statisticsmodelnode | Tipo di dati | Descrizione proprietà |
|-------------------------------|--------------|---|
| syntax | stringa | |
| default_include | flag | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |
| include | flag | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |
| new_name | stringa | Per ulteriori informazioni, vedere l'argomento Proprietà filternode in il capitolo 14 a pag. 156. |

Proprietà *statisticsoutputnode*



Il nodo Output Statistics consente di chiamare una procedura IBM® SPSS® Statistics per analizzare i dati di IBM® SPSS® Modeler. È disponibile una vasta gamma di procedure analitiche di SPSS Statistics. Questo nodo richiede una copia di SPSS Statistics con regolare licenza. [Per ulteriori informazioni, vedere l'argomento Nodo Output Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create statisticsoutputnode
set :statisticsoutputnode.syntax = "SORT CASES BY Age(A) Sex(A) BP(A) Cholesterol(A)"
set :statisticsoutputnode.use_output_name = False
set :statisticsoutputnode.output_mode = File
set :statisticsoutputnode.full_filename = "Cases by Age, Sex and Medical History"
set :statisticsoutputnode.file_type = HTML
```

| Proprietà <i>statisticsoutputnode</i> | Tipo di dati | Descrizione proprietà |
|---------------------------------------|--------------------|--|
| mode | Dialog Syntax | Selezionare l'opzione "Finestra di dialogo SPSS Statistics" o Editor di sintassi |
| syntax | <i>stringa</i> | |
| use_output_name | <i>flag</i> | |
| output_name | <i>stringa</i> | |
| output_mode | Screen File | |
| full_filename | <i>stringa</i> | |
| file_type | HTML SPV SPW | |

Proprietà *statisticsexportnode*



Il nodo Esporta Statistics restituisce i dati nel formato *.sav* di IBM® SPSS® Statistics. I file *.sav* possono essere letti da SPSS Statistics Base e altri prodotti. Questo formato viene inoltre utilizzato per i file cache di IBM® SPSS® Modeler. [Per ulteriori informazioni, vedere l'argomento Nodo Esporta Statistics in il capitolo 8 in IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output.](#)

Esempio

```
create statisticsexportnode
set :statisticsexportnode.full_filename = "c:/output/SPSS_Statistics_out.sav"
set :statisticsexportnode.field_names = Names
set :statisticsexportnode.launch_application = True
set :statisticsexportnode.generate_import = True
```

| Proprietà <i>statisticsexportnode</i> | Tipo di dati | Descrizione proprietà |
|---------------------------------------|----------------|-----------------------|
| full_filename | <i>stringa</i> | |
| launch_application | <i>flag</i> | |

| Proprietà statisticsexportnode | Tipo di dati | Descrizione proprietà |
|---------------------------------------|---------------------------------|---|
| export_names | NamesAndLabels NamesAsLabels | Utilizzata per mappare nomi di campi SPSS Modeler a nomi di variabili SPSS Statistics o SAS durante l'esportazione. |
| generate_import | <i>flag</i> | |

Proprietà dei Supernodi

Nelle tabelle seguenti vengono illustrate le proprietà specifiche dei Supernodi. Si noti che le proprietà comuni dei nodi si applicano anche ai Supernodi.

Tabella 22-1
source_supernode

| Nome proprietà | Tipo di proprietà/Elenco di valori | Descrizione proprietà |
|----------------|------------------------------------|---|
| parameters | <i>qualsiasi</i> | Questa proprietà consente di creare e accedere ai parametri specificati nella tabella dei parametri del Supernodo. Per informazioni dettagliate, vedere più avanti. |

Tabella 22-2
process_supernode

| Nome proprietà | Tipo di proprietà/Elenco di valori | Descrizione proprietà |
|----------------|------------------------------------|---|
| parameters | <i>qualsiasi</i> | Questa proprietà consente di creare e accedere ai parametri specificati nella tabella dei parametri del Supernodo. Per informazioni dettagliate, vedere più avanti. |

Tabella 22-3
terminal_supernode

| Nome proprietà | Tipo di proprietà/Elenco di valori | Descrizione proprietà |
|----------------|------------------------------------|---|
| parameters | <i>qualsiasi</i> | Questa proprietà consente di creare e accedere ai parametri specificati nella tabella dei parametri del Supernodo. Per informazioni dettagliate, vedere più avanti. |
| execute_method | Script Normal | |
| script | <i>stringa</i> | |

Parametri dei Supernodi

Per creare o impostare i parametri dei Supernodi è possibile utilizzare gli script, con il seguente formato generale:

```
set mySuperNode.parameters.minvalue = 30
```

In alternativa, oltre al nome (o al posto del nome) è possibile specificare il tipo di Supernodo:

```
set :process_supernode.parameters.minvalue = 30
```

```
set mySuperNode:process_supernode.parameters.minvalue = 30
```

Il valore del parametro può essere impostato anche mediante un'espressione CLEM:

```
set :process_supernode.parameters.minvalue = "<expression>"
```

Impostazione delle proprietà dei nodi incapsulati

Per impostare le proprietà di nodi specifici incapsulati all'interno di un Supernodo è possibile creare un parametro del Supernodo corrispondente al nome letterale del nodo e della proprietà che si desidera impostare. Si supponga per esempio di disporre di un Supernodo di input con un nodo Testo variabile incapsulato per la lettura dei dati. È possibile passare il nome del file da leggere (specificato mediante la proprietà `full_filename`) nel modo seguente:

```
set :source_supernode.parameters.':variablefilenode.full_filename' = "c:/dati.txt"
```

In questo modo viene creato un parametro del Supernodo denominato `:variablefilenode.full_filename` con valore `c:/dati.txt`. Presupponendo che nel Supernodo sia presente un nodo del tipo specificato, il relativo valore della proprietà denominata sarà impostato di conseguenza. Si noti che questa operazione viene eseguita nello script dello stream (ovvero nello script relativo allo stream che *comprende* il Supernodo), anziché nello script del Supernodo. Per specificare il nome del parametro, utilizzare le virgolette singole.

Questa soluzione può essere utilizzata con qualsiasi nodo incapsulato, purché il risultato sia un riferimento a un nodo e a una proprietà validi. Per esempio, per impostare la proprietà `rand_pct` per un nodo Campione incapsulato si potrebbe utilizzare una qualsiasi delle seguenti righe di codice:

```
set mySuperNode.parameters.':samplenode.rand_pct' = 50
```

o

```
set mySuperNode.parameters.'Sample.rand_pct'= 50
```

o

```
set mySuperNode.parameters.'Sample:samplenode.rand_pct'= 50
```

Il primo dei riferimenti riportati sopra presuppone che nello stream sia presente un solo nodo Campione; il secondo, invece, che sia presente un solo nodo denominato "Sample", indipendentemente dal tipo. Il terzo riferimento è il più esplicito, in quanto specifica sia il nome che il tipo di nodo.

[Per ulteriori informazioni, vedere l'argomento Parametri dei Supernodi in il capitolo 9 in *IBM SPSS Modeler 14.2 Nodi di input, elaborazione e output*.](#)

Limitazioni degli script dei Supernodi. I Supernodi non possono manipolare altri stream, né modificare lo stream corrente. Pertanto, i comandi validi per gli stream, quali `open stream`, `get stream`, `execute_script` e simili non si possono utilizzare negli script dei Supernodi.

Note

This information was developed for products and services offered worldwide.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, Armonk, NY 10504-1785, U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing, Legal and Intellectual Property Law, IBM Japan Ltd., 1623-14, Shimotsuruma, Yamato-shi, Kanagawa 242-8502 Japan.

Il seguente paragrafo non si applica per il Regno Unito o altri paesi in cui le presenti disposizioni non sono conformi alle leggi locali: SPSS INC., UNA SOCIETÀ IBM, FORNISCE IL PRESENTE DOCUMENTO “COSÌ COM'È” SENZA GARANZIA DI ALCUN TIPO, ESPRESSA O IMPLICITA, COMPRESE, TRA LE ALTRE, LE GARANZIE IMPLICITE DI NON VIOLAZIONE DEI DIRITTI ALTRUI, COMMERCIALIZZABILITÀ O IDONEITÀ PER UNO SCOPO SPECIFICO. Alcuni stati non consentono limitazioni di garanzie espresse o implicite in determinate transazioni, pertanto quanto sopra potrebbe non essere applicabile.

Le presenti informazioni possono includere imprecisioni tecniche o errori tipografici. Le modifiche periodiche apportate alle informazioni contenute in questa pubblicazione verranno inserite nelle nuove edizioni della pubblicazione. SPSS Inc. può apportare miglioramenti e/o modifiche al/ai prodotto/i e/o al/ai programma/i descritti nella presente pubblicazione in qualsiasi momento senza preavviso.

Eventuali riferimenti contenuti nelle presenti informazioni a siti Web non SPSS e non IBM vengono forniti per comodità dell'utente e non implicano in alcun modo l'approvazione di tali siti Web. I materiali presenti in tali siti Web non sono parte dei materiali di questo prodotto SPSS Inc. e l'utilizzo di tali siti Web è a rischio esclusivo dell'utente.

Al momento dell'invio di informazioni a IBM o SPSS, l'utente concede a IBM e SPSS un diritto non esclusivo a utilizzare o distribuire tali informazioni nel modo che ritiene appropriato senza incorrere in alcun obbligo nei confronti dell'utente.

Le informazioni relative ai prodotti non SPSS provengono dai fornitori di tali prodotti, dagli annunci pubblicati o da altre fonti pubbliche disponibili. SPSS non ha testato tali prodotti e non può confermare l'accuratezza delle dichiarazioni relative alle prestazioni, alla compatibilità o ad altre caratteristiche relative ai prodotti non SPSS. Le domande relative alle funzionalità dei prodotti non SPSS devono essere indirizzate ai fornitori di tali prodotti.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Software Group, Attention: Licensing, 233 S. Wacker Dr., Chicago, IL 60606, USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Le presenti informazioni includono esempi di dati e report utilizzati in operazioni aziendali quotidiane. Per fornire una descrizione il più possibile esaustiva, gli esempi includono nomi di persone, società, marchi e prodotti. Tutti questi nomi sono fittizi e ogni somiglianza a nomi e indirizzi utilizzati da aziende reali è puramente casuale.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Marchi

IBM, il logo IBM e [ibm.com](http://www.ibm.com) sono marchi di IBM Corporation, registrati in numerose giurisdizioni nel mondo. Un elenco aggiornato dei marchi IBM è disponibile sul Web all'indirizzo <http://www.ibm.com/legal/copytrade.shtml>.

SPSS è un marchio di SPSS Inc., una società IBM, registrati in numerose giurisdizioni nel mondo.

Adobe, il logo Adobe, PostScript e il logo PostScript sono marchi o marchi registrati di Adobe Systems Incorporated negli Stati Uniti e/o in altri paesi.

IT Infrastructure Library è un marchio registrato della Central Computer and Telecommunications Agency ora parte dell'Office of Government Commerce.

Intel, il logo Intel, Intel Inside, il logo Intel Inside, Intel Centrino, il logo Intel Centrino, Celeron, Intel Xeon, Intel SpeedStep, Itanium e Pentium sono marchi o marchi registrati di Intel Corporation o delle sue consociate negli Stati Uniti e in altri paesi.

Linux è un marchio registrato di Linus Torvalds negli Stati Uniti e/o negli altri paesi.

Microsoft, Windows, Windows NT e il logo Windows sono marchi di Microsoft Corporation negli Stati Uniti e/o negli altri paesi.

ITIL è un marchio registrato e un marchio comunitario registrato dell'Office of Government Commerce ed è registrato presso l'Ufficio Brevetti degli Stati Uniti.

UNIX è un marchio registrato di The Open Group negli Stati Uniti e in altri paesi.

Cell Broadband Engine è un marchio di Sony Computer Entertainment, Inc. negli Stati Uniti e/o in altri paesi ed è utilizzato su licenza.

Java e tutti i marchi e i logo basati su Java sono marchi di Sun Microsystems, Inc. negli Stati Uniti e/o negli altri paesi.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

I nomi di altri prodotti o servizi possono essere marchi di IBM, SPSS o altre società.

- Albero decisionale MS
 - proprietà script dei nodi, 253, 256
- and, operatore, 87
- annotazioni
 - accesso negli script, 67
- argomenti
 - connessione a IBM SPSS Collaboration and Deployment Services Repository, 72
 - connessione al server, 70
 - file di comando, 70
 - modulo, 72

- barra rovesciata nelle espressioni CLEM, 77

- campi, 75, 77
- campi continui
 - proprietà values, 65
- campi flag
 - proprietà values, 65
- campi nominali
 - proprietà values, 65
- campi ora
 - conversione , 102
- caratteri, 75, 77
- cicli
 - uso negli script, 57, 65–66
- CLEM
 - espressioni, 75
 - lingua, 75
 - script, 7, 18
 - tipi di dati, 76–77
- CLEM espressioni
 - parametri, 24
- CLEM funzioni
 - bitwise, 90
 - casuali, 92
 - confronto, 85
 - datetime, 98
 - funzioni speciali, 109
 - globali, 108
 - informazione, 83
 - logiche, 87
 - numerico, 88
 - probability, 90
 - sequence, 102, 104
 - stringa, 92
 - trigonometriche, 89
 - vuoti e nulli, 109
- CLEMfunzioni
 - conversione, 84
 - elenco funzioni disponibili, 81
- Cluster di sequenze MS
 - proprietà script dei nodi, 258
- comando clear generated palette, 44, 61
- comando clear stream, 48
- comando close FILE, 53
- comando close STREAM, 47
- comando connect NODO, 37
- comando create NODO, 36
- comando create stream, 47
- comando delete model, 44
- comando delete NODO, 37
- comando delete output, 55
- comando di impostazione multipla, 114
- comando disable NODO, 37
- comando disconnect NODO, 38
- comando duplicate NODO, 38
- comando enable NODO, 38
- comando execute NODO, 38
- comando execute_all, 30
- comando execute_project, 51
- comando execute_script, 30
- comando exit, 26, 30
- comando export model, 45
- comando export NODO, 39
- comando export output, 55
- comando flush NODO, 39
- comando for, 23, 26, 57, 63, 66
- comando for...endfor, 31
- comando get, 21
- comando get node, 40
- comando get output, 55
- comando get stream, 48
- comando if, 26, 63
- comando if...then...else ..., 32
- comando insert model, 46
- comando load model, 46
- comando load node, 40
- comando load output, 55
- comando load project, 51
- comando load state, 52
- comando load stream, 48
- comando open FILE, 53
- comando open stream, 23, 49
- comando position NODO, 40
- comando rename NODE, 23, 41
- comando retrieve model, 47
- comando retrieve node, 41
- comando retrieve output, 55
- comando retrieve project, 51
- comando retrieve stream, 49
- comando save, 21
- comando save model, 47
- comando save node, 42
- comando save output, 56
- comando save project, 52
- comando save STREAM, 49
- comando set, 19, 23–24, 32
- comando store model, 47
- comando store node, 42
- comando store output, 56

- comando store project, 52
- comando store stream, 50
- comando value, 52
- comando var, 19, 23, 35
- comando with stream, 23, 51
- comando write FILE, 54
- comando writeln FILE, 54, 63, 66
- commenti
 - script, 27
- concatenazione di stringhe, 84
- continuazioni
 - script, 27
- controllo degli errori
 - script, 60
- convenzioni, 82

- date
 - conversione, 102
 - gestione, 102
- direttive di ingrandimento dell'albero
 - incorporazione negli script, 28
- distribuzione chi-quadrato
 - funzioni di probabilità, 90
- distribuzione *f*
 - funzioni di probabilità, 90
- distribuzione normale
 - funzioni di probabilità, 90
- distribuzione *t*
 - funzioni di probabilità, 90
- diverso da, operatore, 85
- documentazione, 2

- elenchi, 75, 77
- esecuzione degli script, 14
- esempi
 - cenni generali, 4
 - Guida alle applicazioni, 2
- esempi di applicazioni, 2
- esportazione
 - modelli, 45
 - nodi, 39
 - PMML, 39, 45
 - SQL, 39, 45
- espressioni, 75
- espressioni CLEM
 - ricerca e sostituzione di testo, 15
 - script, 27, 32

- fields
 - disattivazione negli script, 175
- flag
 - argomenti della riga di comando, 69
 - combinazione di più flag, 70
- formati di data, 78–79
- formati di ora, 78–79

- formato HTML
 - esportazione di modelli, 45
 - esportazione di nodi, 39
- formato PMML
 - esportazione di modelli, 45
 - esportazione di nodi, 39
- formato SQL
 - esportazione di nodi, 39, 45
- formato testo
 - esportazione di modelli, 45
 - esportazione di nodi, 39
- funzione @BLANK, 83, 109
- funzione @DIFF, 102, 104
- funzione @FIELD, 109
- funzione @FIELDS_BETWEEN, 109
- funzione @FIELDS_MATCHING, 109
- funzione @INDEX, 102, 104
- funzione @LAST_NON_BLANK, 102, 104, 109
- funzione @MAX, 102, 104
- funzione @MEAN, 102, 104
- funzione @MIN, 102, 104
- funzione @MULTI_RESPONSE_SET, 109
- funzione @NULL, 83, 109
- funzione @OFFSET, 102, 104
- funzione @PARTITION_FIELD, 109
- funzione @PREDICTED, 109
- funzione @SDEV, 102, 104
- funzione @SINCE, 102, 104
- funzione @SUM, 102, 104
- funzione @TARGET, 109
- funzione @TESTING_PARTITION, 109
- funzione @THIS, 102, 104
- funzione @TODAY, 98
- funzione @TRAINING_PARTITION, 109
- funzione @VALIDATION_PARTITION, 109
- funzione abs, 88
- funzione allbutfirst, 92
- funzione allbutlast, 92
- funzione alphabefore, 92
- funzione arccos, 89
- funzione arccosh, 89
- funzione arcsin, 89
- funzione arcsinh, 89
- funzione arctan, 89
- funzione arctan2, 89
- funzione arctanh, 89
- funzione cdf_chisq, 90
- funzione cdf_f, 90
- funzione cdf_normal, 90
- funzione cdf_t, 90
- funzione cos, 89
- funzione cosh, 89
- funzione count_equal, 85
- funzione count_greater_than, 85
- funzione count_less_than, 85
- funzione count_non_nulls, 85
- funzione count_not_equal, 85

- funzione count_nulls, 85
funzione count_substring, 92
funzione date_before, 85
funzione datetime_date, 84
funzione DIFF, 104
funzione div, 88
funzione endstring, 92
funzione exp, 88
funzione first_index, 85
funzione first_non_null, 85
funzione first_non_null_index, 85
funzione fracof, 88
funzione hasendstring, 92
funzione hasmidstring, 92
funzione hasstartstring, 92
funzione INDEX, 104
funzione integer_bitcount, 90
funzione integer_leastbit, 90
funzione integer_length, 90
funzione intof, 88
funzione is_date, 83
funzione is_datetime, 83
funzione is_integer, 83
funzione is_number, 83
funzione is_real, 83
funzione is_string, 83
funzione is_time, 83
funzione is_timestamp, 83
funzione isalphacode, 92
funzione isendstring, 92
funzione islowercode, 92
funzione ismidstring, 92
funzione isnumbercode, 92
funzione isstartstring, 92
funzione issubstring, 92
funzione issubstring_count, 92
funzione issubstring_lim, 92
funzione isuppercode, 92
funzione last_index, 85
funzione LAST_NON_BLANK, 104
funzione last_non_null, 85
funzione last_non_null_index, 85
funzione length, 92
funzione locchar, 92
funzione locchar_back, 92
funzione log, 88
funzione log10, 88
funzione lowertoupper, 57, 92
funzione matches, 92
funzione max, 85
funzione MAX, 104
funzione max_index, 85
funzione max_n, 85
funzione MEAN, 102, 104
funzione mean_n, 88
funzione member, 85
funzione min, 85
funzione MIN, 104
funzione min_index, 85
funzione min_n, 85
funzione mod, 88
funzione negate, 88
funzione OFFSET, 104
funzione oneof, 92
funzione pi, 89
funzione potenza (esponenziale), 88
funzione random, 92
funzione random0, 92
funzione rem, 88
funzione replace, 92
funzione replicate, 92
funzione round, 88
funzione SDEV, 104
funzione sdev_n, 88
funzione sign, 88
funzione sin, 89
funzione SINCE, 104
funzione sinh, 89
funzione skipchar, 92
funzione skipchar_back, 92
funzione soundex, 97
funzione soundex_difference, 97
funzione sqrt, 88
funzione startstring, 92
funzione stripchar, 92
funzione strmember, 92
funzione subscrs, 92
funzione substring, 92
funzione substring_between, 92
funzione SUM, 104
funzione sum_n, 88
funzione tan, 89
funzione tanh, 89
funzione testbit, 90
funzione THIS, 104
funzione time_before, 85
funzione to_date, 84, 98
funzione to_dateline, 98
funzione to_datetime, 84
funzione to_integer, 84
funzione to_number, 84
funzione to_string, 84
funzione to_time, 84, 98
funzione to_timestamp, 84, 98
funzione trim, 92
funzione trim_start, 92
funzione trimend, 92
funzione undef, 109
funzione unicode_char, 92
funzione unicode_value, 92
funzione uppertolower, 92
funzione value_at, 85
funzioni, 78–79, 82–83, 102
@FIELD, 109

- @GLOBAL_MAX, 108
- @GLOBAL_MEAN, 108
- @GLOBAL_MIN, 108
- @GLOBAL_SDEV, 108
- @GLOBAL_SUM, 108
- @PARTITION, 109
- @PREDICTED, 109
- @TARGET, 109
- funzioni bitwise, 90
- funzioni datetime
 - datetime_date, 98
 - datetime_day, 98
 - datetime_day_name, 98
 - datetime_day_short_name, 98
 - datetime_hour, 98
 - datetime_in_seconds, 98
 - datetime_minute, 98
 - datetime_month, 98
 - datetime_month_name, 98
 - datetime_month_short_name, 98
 - datetime_now datetime_second, 98
 - datetime_time, 98
 - datetime_timestamp, 98
 - datetime_weekday, 98
 - datetime_year, 98
- funzioni di confronto, 85
- funzioni di conversione, 84
- funzioni di data, 78–79
 - date_before, 85, 98
 - date_days_difference, 98
 - date_in_days, 98
 - date_in_months, 98
 - date_in_weeks, 98
 - date_in_years, 98
 - date_months_difference, 98
 - date_weeks_difference, 98
 - date_years_difference, 98
 - funzione @TODAY, 98
- funzioni di data e ora, 78–79
- funzioni di distribuzione, 90
- funzioni di informazione, 83
- funzioni di ora, 78–79
 - time_before, 85, 98
 - time_hours_difference, 98
 - time_in_hours, 98
 - time_in_mins, 98
 - time_in_secs, 98
 - time_mins_difference, 98
 - time_secs_difference, 98
- funzioni di probabilità, 90
- funzioni di sequenza, 102, 104
- funzioni globali, 108
- funzioni logiche, 87
- funzioni numeriche, 88
- funzioni speciali, 109
- funzioni stringa, 57, 92
- funzioni trigonometriche, 89
- funzione to_real, 84
- Generatore espressioni
 - ricerca e sostituzione di testo, 15
- gestione dei vuoti
 - CLEMfunzioni, 109
- hassubstring, funzione, 92
- IBM SPSS Collaboration and Deployment Services Repository
 - argomenti della riga di comando, 72
 - script, 58
- IBM SPSS Modeler, 1
 - documentazione, 2
 - esecuzione dalla riga di comando, 69
- IBM SPSS Text Analytics, 2
- ID nodi
 - riferimenti negli script, 19
- if then else, funzioni, 87
- insiemi di modelli
 - nomi di script, 42, 44
 - proprietà script dei nodi, 243
- interi, 75–76
- interruzione degli script, 14
- introduzione, 75
- maggiore di, operatore, 85
- marchi, 302
- minore di, operatore, 85
- modellazione di database, 253
- modelli
 - esportazione, 45
 - nomi di script, 42, 44
 - script, 45
- modelli Albero decisionale Oracle
 - proprietà script dei nodi, 261, 263
- modelli Apriori
 - proprietà script dei nodi, 191, 243
- modelli Apriori Oracle
 - proprietà script dei nodi, 262, 264
- modelli Bayes adattivi Oracle
 - proprietà script dei nodi, 259, 263
- Modelli Bayes Naive IBM ISW
 - proprietà script dei nodi, 268, 270
- modelli Bayes naive Oracle
 - proprietà script dei nodi, 259, 263
- modelli C&R Tree
 - proprietà script dei nodi, 202, 246
- modelli C5.0
 - proprietà script dei nodi, 200, 245
- modelli CARMA
 - proprietà script dei nodi, 201, 246

- modelli CHAID
 - proprietà script dei nodi, 205, 246
- Modelli Classificatore automatico
 - proprietà script dei nodi, 244
- Modelli Cluster automatico
 - proprietà script dei nodi, 244
- modelli del vicino più vicino
 - proprietà script dei nodi, 220
- modelli di albero decisionale IBM ISW
 - proprietà script dei nodi, 265, 269
- Modelli di albero decisionale Netezza
 - proprietà script dei nodi, 271–272
- modelli di associazione IBM ISW
 - proprietà script dei nodi, 265, 269
- modelli di raggruppamento cluster IBM ISW
 - proprietà script dei nodi, 268, 270
- modelli di regressione di Cox
 - proprietà script dei nodi, 207, 247
- modelli di regressione IBM ISW
 - proprietà script dei nodi, 267, 270
- modelli di regressione lineare
 - proprietà script dei nodi, 234, 251
- modelli di regressione logistica
 - proprietà script dei nodi, 224, 249
- modelli di regressione logistica IBM ISW
 - proprietà script dei nodi, 268, 270
- modelli di rete bayesiana
 - proprietà script dei nodi, 198
- modelli di selezione funzioni
 - applicazione, 12
 - proprietà script dei nodi, 214, 248
 - script, 12
- modelli di sequenza IBM ISW
 - proprietà script dei nodi, 266, 269
- modelli di serie storica
 - proprietà script dei nodi, 239, 252
- Modelli di serie storica IBM ISW
 - proprietà script dei nodi, 269
- modelli discriminanti
 - proprietà script dei nodi, 210, 247
- modelli Elenco decisionale
 - proprietà script dei nodi, 209, 247
- modelli Fattoriale
 - proprietà script dei nodi, 212, 248
- modelli generati
 - nomi di script, 42, 44
- modelli IBM DB2
 - proprietà script dei nodi, 264
- modelli IBM SPSS Statistics
 - proprietà script dei nodi, 296
- modelli K-Means
 - proprietà script dei nodi, 219, 248
- modelli K-Means Netezza
 - proprietà script dei nodi, 271–272
- modelli K-Means Oracle
 - proprietà script dei nodi, 261, 264
- modelli KNN
 - proprietà script dei nodi, 249
- modelli Kohonen
 - proprietà script dei nodi, 221, 249
- modelli lineari
 - proprietà script dei nodi, 222, 249
- modelli lineari generalizzati
 - proprietà script dei nodi, 216, 248
- modelli lineari generalizzati Oracle
 - proprietà script dei nodi, 260
- modelli MDL Oracle
 - proprietà script dei nodi, 262, 264
- modelli Microsoft
 - proprietà script dei nodi, 253, 256
- Modelli Netezza
 - proprietà script dei nodi, 270
- modelli NMF Oracle
 - proprietà script dei nodi, 262, 264
- modelli numerici automatici
 - proprietà script dei nodi, 197
- Modelli Numerici automatici
 - proprietà script dei nodi, 245
- modelli Oracle
 - proprietà script dei nodi, 258
- Modelli Oracle AI
 - proprietà script dei nodi, 262
- modelli PCA
 - proprietà script dei nodi, 212, 248
- modelli QUEST
 - proprietà script dei nodi, 232, 250
- modelli Rete neurale
 - proprietà script dei nodi, 228, 249
- modelli Rilevamento anomalie
 - proprietà script dei nodi, 190, 243
- modelli Risposta autoapprendimento
 - proprietà script dei nodi, 237, 251
- modelli Sequenza
 - proprietà script dei nodi, 236, 251
- modelli SLRM
 - proprietà script dei nodi, 237, 251
- modelli support vector machine
 - proprietà script dei nodi, 251
- modelli SVM
 - proprietà script dei nodi, 238
- modelli SVM Oracle
 - proprietà script dei nodi, 259, 263
- modelli TwoStep
 - proprietà script dei nodi, 241, 252
- nodi
 - esecuzione di cicli sugli script, 57
- nodi di esportazione
 - proprietà script dei nodi, 286
- nodi di input
 - proprietà, 121
- nodi Grafici
 - proprietà script, 175

- nodì Modelli
 - proprietà script dei nodi, 189
- nodì output
 - proprietà script, 273
- nodò Accodamento
 - proprietà, 138
- nodò Aggregazione
 - proprietà, 138
- nodò Aggregazione RFM
 - proprietà, 142
- nodò Analisi
 - proprietà, 273
- nodò Analisi RFM
 - proprietà, 162
- nodò Anonimizza
 - proprietà, 147
- nodò Bilanciamento
 - proprietà, 139
- nodò Campione
 - proprietà, 144
- nodò Classificatore automatico
 - proprietà script dei nodi, 192
- Nodò Cluster automatico
 - proprietà script dei nodi, 195
- nodò Crea flag
 - proprietà, 163
- nodò Cronologia
 - proprietà, 158
- nodò Database
 - proprietà, 124, 287
- nodò di input Excel
 - proprietà, 127
- Nodò di input IBM Cognos BI
 - proprietà, 123
- nodò di input SAS
 - proprietà, 131
- Nodò di input XML
 - proprietà, 136
- nodò Discretizza
 - proprietà, 151
- nodò Distribuzione
 - proprietà, 177
- nodò Elimina duplicati
 - proprietà, 140
- nodò Enterprise View
 - proprietà, 129
- nodò Esplora
 - proprietà, 274
- nodò Esporta IBM SPSS Data Collection
 - proprietà, 291
- nodò Esporta IBM SPSS Statistics
 - proprietà, 297
- nodò Esporta SAS
 - proprietà, 293
- nodò Esporta XML
 - proprietà, 294
- nodò Esportazione da Excel
 - proprietà, 291
- nodò File piatto
 - proprietà, 292
- nodò Filtro
 - proprietà, 156
- nodò Globali
 - proprietà, 280
- nodò Input IBM SPSS Data Collection
 - proprietà, 125
- nodò Input IBM SPSS Statistics
 - proprietà, 295
- nodò Input utente
 - proprietà, 132
- nodò Intervalli di tempo
 - proprietà, 164
- nodò Istogramma
 - proprietà, 181
- nodò Lavagna grafica
 - proprietà, 180
- nodò Matrice
 - proprietà, 276
- nodò Medie
 - proprietà, 277
- nodò Nuovo campo
 - proprietà, 153
- nodò Ordina
 - proprietà, 146
- nodò Output IBM SPSS Statistics
 - proprietà, 297
- nodò Partizione
 - proprietà, 158
- nodò Plot
 - proprietà, 183
- nodò Plot multiplo
 - proprietà, 182
- nodò Plot tempo
 - proprietà, 185
- nodò Raccolta
 - proprietà, 176
- nodò Report, 63, 66
 - proprietà, 279
- nodò Ricodifica
 - proprietà, 160
- nodò Riempimento
 - proprietà, 156
- nodò Riordina
 - proprietà, 161
- nodò Riordina campi
 - proprietà, 161
- nodò Riorganizza
 - proprietà, 161
- nodò Risultati classificatore binario
 - proprietà, 155
- nodò Seleziona
 - proprietà, 146

- nodo Statistiche
 - proprietà, 280
- nodo Tabella
 - proprietà, 282
- nodo Testo fisso
 - proprietà, 129
- nodo Testo variabile
 - proprietà, 133
- nodo Tipo
 - proprietà, 170
- nodo Trasformazioni
 - proprietà, 284
- nodo Trasformazioni IBM SPSS Statistics
 - proprietà, 295
- nodo Trasponi
 - proprietà, 169
- nodo Unione
 - proprietà, 141
- nodo Valutazione
 - proprietà, 178
- nodo Web
 - proprietà, 187
- nodo Web diretto
 - proprietà, 187
- nomi di campo
 - modifica di maiuscole/minuscole, 57
- nomi di stream
 - accesso negli script, 67
- not, operatore, 87
- note legali, 301
- numeri, 76
- numeri reali, 75–76

- O-Cluster Oracle
 - proprietà script dei nodi, 261, 264
- oggetti di output
 - comandi di script, 54
 - nomi di script, 54
- oggetti file
 - comandi di script, 53
- oggetti modello
 - comandi di script, 42
 - nomi di script, 42, 44
- oggetti nodo
 - comandi di script, 36
 - script, 19
- oggetti risultato
 - comando di script, 52
- oggetti stato
 - comandi di script, 52
- oggetti stream
 - apertura, 22–23
 - comandi di script, 47
 - riferimento, 23
- oggetto corrente
 - riferimenti negli script, 21

- Operatori
 - script, 26
 - unione di stringhe, 67, 84
- or, operatore, 87
- ordine di esecuzione
 - modifica con script, 57
- output HTML
 - creazione mediante script, 63, 66

- parameters, 32
- parametri, 13, 113–115, 117
 - script, 18, 27
 - sessione, 24
 - stream, 24
 - Supernodi, 299
- parametri di configurazione, 13, 32, 113, 116
- parametri elenco
 - modifica negli script, 26
- parametri sessione, 24, 32
- parametri stream, 24, 32
- parola chiave generated, 61
- password
 - aggiunta a script, 60
 - codifica, 70
- password codificata
 - aggiunta a script, 60
- precedenza, 79
- precedenza degli operatori, 79
- preparazione automatica dati
 - proprietà, 148
- progetti
 - proprietà, 120
 - proprietà   restructurenode, 161
 - proprietà, 32
 - nodi Filtro, 114
 - nodi Modelli database, 253
 - progetti, 120
 - script, 113–116, 189, 243, 286
 - script comuni, 116
 - stream, 117
 - Supernodi, 299
- proprietà aggregatenode, 138
- proprietà analysisnode, 273
- proprietà anomalydetectionnode, 190
- proprietà anonymizenode, 147
- proprietà appendnode, 138
- proprietà applyanomalydetectionnode, 243
- proprietà applyapriorinode, 243
- proprietà applyautoclassifiernode, 244
- proprietà applyautoclusternode, 244
- proprietà applyautonumericnode, 245
- proprietà applybayesnetnode, 245
- proprietà applyc50node, 245
- proprietà applycarmanode, 246
- proprietà applycartnode, 246
- proprietà applychaidnode, 246
- proprietà applycoxregnode, 247

- proprietà applydb2imclusternode, 270
- proprietà applydb2imlognode, 270
- proprietà applydb2imnbnode, 270
- proprietà applydb2imregnode, 270
- proprietà applydb2imtreenode, 269
- proprietà applydecisionlistnode, 247
- proprietà applydiscriminantnode, 247
- proprietà applyfactornode, 248
- proprietà applyfeatureselectionnode, 248
- proprietà applygeneralizedlinearnode, 248
- proprietà applykmeansnode, 248
- proprietà applyknnnode, 249
- proprietà applykohonenode, 249
- proprietà applylinearnode, 249
- proprietà applylogregnode, 249
- proprietà applymslogisticnode, 257
- proprietà applymsneuralnetworknode, 257
- proprietà applymsregressionnode, 257
- proprietà applymssequenceclusternode, 258
- proprietà applymstimeseriesnode, 257
- proprietà applymstreenode, 256
- proprietà applynetezzadectreenode, 272
- proprietà applynetezzakmeansnode, 272
- proprietà applyneuralnetnode, 249
- proprietà applyneuralnetworknode, 250
- proprietà applyoraabnnode, 263
- proprietà applyoradecisiontreenode, 263
- proprietà applyorakmeansnode, 264
- proprietà applyoranbnnode, 263
- proprietà applyoranmfnode, 264
- proprietà applyoraoclusternode, 264
- proprietà applyorasvmnode, 263
- proprietà applyquestnode, 250
- proprietà applyregressionnode, 251
- proprietà applyselflearningnode, 251
- proprietà applysequencenode, 251
- proprietà applysvmnode, 251
- proprietà applytimeseriesnode, 252
- proprietà applytwestepnode, 252
- proprietà apriorinode, 191
- proprietà autotoclassifiernode, 192
- proprietà autotoclusternode, 195
- proprietà autodataprepnode, 148
- proprietà autonumericnode, 197
- proprietà balancenode, 139
- proprietà bayesnet, 198
- proprietà binningnode, 151
- proprietà c50node, 200
- proprietà carmanode, 201
- proprietà cartnode, 202
- proprietà chaidnode, 205
- proprietà cognosimportnode, 123
- proprietà collectionnode, 176
- proprietà column_count, 53
- proprietà coxregnode, 207
- proprietà dataauditnode, 274
- proprietà databaseexportnode, 287
- proprietà databasenode, 124
- proprietà datacollectionexportnode, 291
- proprietà datacollectionimportnode, 125
- proprietà db2imassocnode, 265
- proprietà db2imclusternode, 268
- proprietà db2imlognode, 268
- proprietà db2imnbnode, 268
- proprietà db2imregnode, 267
- proprietà db2imsequencenode, 266
- proprietà db2imtimeseriesnode, 269
- proprietà db2imtreenode, 265
- proprietà decisionlist, 209
- proprietà dei nodi
 - accesso negli script, 67
- proprietà del nodo Ricodifica, 160
- proprietà derivenode, 153
- proprietà directedwebnode, 187
- proprietà discriminantnode, 210
- proprietà distinctnode, 140
- proprietà distributionnode, 177
- proprietà ensemblenode, 155
- proprietà evaluationnode, 178
- proprietà evimportnode, 129
- proprietà excelexportnode, 291
- proprietà excelimportnode, 127
- proprietà factornode, 212
- proprietà featureselectionnode, 12, 214
- proprietà fillernode, 156
- proprietà filternode, 156
- proprietà fixedfilenode, 129
- proprietà flatfilenode, 292
- proprietà genlinnode, 216
- proprietà graphboardnode, 180
- proprietà histogramnode, 181
- proprietà historynode, 158
- proprietà kmeansnode, 219
- proprietà knnnode, 220
- proprietà kohonenode, 221
- proprietà lineari, 222
- proprietà logregnode, 224
- proprietà matrixnode, 276
- proprietà meansnode, 277
- proprietà mergenode, 141
- proprietà msassocnode, 253
- proprietà msbayesnode, 253
- proprietà msclusternode, 253
- proprietà mslogisticnode, 253
- proprietà msneuralnetworknode, 253
- proprietà msregressionnode, 253
- proprietà mssequenceclusternode, 253
- proprietà mstimeseriesnode, 253
- proprietà mstreenode, 253
- proprietà multiplotnode, 182
- proprietà netezzadectreenode, 271
- proprietà netezzakmeansnode, 271
- proprietà neuralnetnode, 228
- proprietà neuralnetworknode, 230

- proprietà numericpredictornode, 197
- proprietà oraabnnode, 259
- proprietà oraainode, 262
- proprietà oraapriorinode, 262
- proprietà oradecisiontreenode, 261
- proprietà oraglmnode, 260
- proprietà orakmeansnode, 261
- proprietà oramdlnode, 262
- proprietà oranbnode, 259
- proprietà oranmfnode, 262
- proprietà oraoclusternode, 261
- proprietà orasvmnode, 259
- proprietà outputfilenode, 292
- proprietà partitionnode, 158
- proprietà plotnode, 183
- proprietà questnode, 232
- proprietà regressionnode, 234
- proprietà reordernode, 161
- proprietà reportnode, 279
- proprietà rfmaggregatenode, 142
- proprietà rfanalysisnode, 162
- proprietà row_count, 53
- proprietà samplenode, 144
- proprietà sasexportnode, 293
- proprietà sasimportnode, 131
- proprietà script dei nodi, 253
 - insiemi di modelli, 243
 - nodi di esportazione, 286
 - nodi Modelli, 189
- proprietà selectnode, 146
- proprietà sequencenode, 236
- proprietà setglobalsnode, 280
- proprietà settoflagnode, 163
- proprietà slrmnode, 237
- proprietà sortnode, 146
- proprietà statisticsexportnode, 297
- proprietà statisticsimportnode, 12, 295
- proprietà statisticsmodelnode, 296
- proprietà statisticsnode, 280
- proprietà statisticsoutputnode, 297
- proprietà statisticstransformnode, 295
- proprietà stream, 67
- proprietà stream.nodes, 57
- proprietà strutturate, 114
- proprietà svmnode, 238
- proprietà tablenode, 282
- proprietà timeintervalnode, 164
- proprietà timeplotnode, 185
- proprietà timeseriesnode, 239
- proprietà transformnode, 284
- proprietà transposenode, 169
- proprietà twostepnode, 241
- proprietà typenode, 12, 64, 170
- proprietà userinputnode, 132
- proprietà values, 65
- proprietà variablefilenode, 133
- proprietà webnode, 187
- proprietà xmlexportnode, 294
- proprietà xmlimportnode, 136
- Regressione lineare MS
 - proprietà script dei nodi, 253, 257
- Regressione logistica MS
 - proprietà script dei nodi, 253, 257
- report
 - creazione mediante script, 63, 66
- Rete bayesiana, modelli
 - proprietà script dei nodi, 245
- Rete neurale MS
 - proprietà script dei nodi, 253, 257
- reti neurali
 - proprietà script dei nodi, 230, 250
- retrieve, comando, 58
- ricerca di testo, 15
- riga di comando
 - avvio di IBM SPSS Modeler, 69
 - elenco di argomenti, 70, 72
 - esecuzione di IBM SPSS Modeler, 69
 - parametri, 74
 - più argomenti, 70
 - script, 61
- script
 - abbreviazioni utilizzate, 115
 - cenni generali, 7, 18
 - commenti, 27
 - compatibilità con versioni precedenti, 61
 - continuazioni, 27
 - controllo degli errori, 60
 - dalla riga di comando, 61
 - esecuzione, 14
 - esecuzione degli script, 26
 - esempi, 63, 66
 - espressioni CLEM, 27
 - importazione da file di testo, 8
 - interfaccia utente, 8, 10, 13
 - interruzione, 14
 - modelli di selezione funzioni, 12
 - nei Supernodi, 13
 - nodi, 19
 - nodi Grafici, 175
 - nodi output, 273
 - oggetto corrente, 21
 - Operatori, 26
 - ordine di esecuzione, 57
 - proprietà comuni, 116
 - ricerca e sostituzione di testo, 15
 - salvataggio, 8
 - script del Supernodo, 7
 - script locali, 7
 - stream, 7
 - syntax, 18
- script locali, 7, 10

-
- Serie storica Microsoft
 - proprietà script dei nodi, 257
 - server
 - argomenti della riga di comando, 70
 - sicurezza
 - password codificata, 60, 70
 - sintassi dell'accento circonflesso
 - riferimenti a variabili, 19, 23
 - sistema
 - argomenti della riga di comando, 72
 - sostituzione di testo, 15
 - sovrapposte, 23, 32
 - riferimenti a nodi, 19
 - script, 18, 21
 - spazi
 - rimozione da stringhe, 92
 - spazio bianco
 - rimozione da stringhe, 92
 - SPSS Modeler Server, 1
 - store, comando, 58
 - stream
 - comando di impostazione multipla, 113
 - proprietà, 117
 - script, 7–8
 - stringhe, 75, 77
 - modifica di maiuscole/minuscole, 57
 - script, 19
 - stringhe di testo
 - incorporazione negli script, 28
 - stringhe letterali
 - incorporazione negli script, 28
 - Supernodi
 - impostazione delle proprietà, 299
 - parametri, 299
 - proprietà, 299
 - script, 7, 13–14, 299
 - Supernodo, 113
 - parameters, 32
 - parametri, 24

 - uguale a, operatore, 85

 - valori letterali
 - script, 18, 28
 - variabili locali, 23, 32
 - variabili speciali, 21