SAS® Enterprise Miner™ High-Performance Data Mining Node Reference for SAS® 9.3

Third Edition

SAS® Documentation

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2012. *SAS® Enterprise Miner™ High-Performance Data Mining Node Reference for SAS® 9.3, Third Edition*. Cary, NC: SAS Institute Inc.

SAS® Enterprise Miner™ High-Performance Data Mining Node Reference for SAS® 9.3

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

Electronic book 1, December 2012

Contents

Chapter 1
The HP Data Partition Node

Overview of the HP Data Partition Node

Most data mining projects use large volumes of sampled data. After sampling, the data is usually partitioned before modeling.

Use the HP Data Partition node to partition your input data into one of the following data sets:

Train	used for preliminary model fitting. The analyst tries to find the best model weights by using this data set.
Validation	used to assess the adequacy of the model in the **Model Comparison** node. The validation data set is also used for fine-tuning models in the following nodes: • **HP Forest node** — to create a decision tree model. • **HP Neural node** — to choose among network architectures or for the early-stopping of the training algorithm. • **HP Regression node** — to choose a final subset of predictors from all the subsets that are computed during stepwise regression.

Partitioning provides mutually exclusive data sets. Two or more mutually exclusive data sets share no observations with each other. Partitioning the input data reduces the computation time of preliminary modeling runs. However, you should be aware that for small data sets, partitioning might be inefficient because the reduced sample size can degrade the fit of the model and its ability to generalize.

What are the steps to partition a data set? First, you specify a sampling method: simple random sampling or stratified random sampling. Then you specify the proportion of sampled observations to write to each output data set (Train or Validation).

The HP Data Partition node creates a permanent table on the HPA appliance to store only the relevant partition information. By storing this information you avoid rerunning the diagrams and provide consistent results from run to run because the partitions remain static.

HP Data Partition Node Requirements

The **HP Data Partition** node must be preceded by a node that exports at least one raw or train data table. If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

The input data set must also contain a variable with the role Key. The key variable contains a unique identifier for each observation in the input data set. Note that the key variable is required for Teradata and Greenplum data sets, but not for Hadoop data sets.

HP Data Partition Node Properties

HP Data Partition Node General Properties

The following general properties are associated with the **HP Data Partition** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Data Partition** node that is added to a diagram has a Node ID of HPPart. The second **HP Data Partition** node that is added to a diagram has a Node ID of HPPart2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Data Partition window. The Imported Data — HP Data Partition window contains a list of the ports that provide data sources to the **HP Data Partition** node. Click ... on the right of the **Imported Data** property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and click one of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and variables.

- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Data Partition window. The Exported Data — HP Data Partition window contains a list of the output data ports for which the **HP Data Partition** node creates data it runs. Click ... on the right of the **Exported Data** property to open a table that lists the exported data sets.

  If data exists for an exported data set, you can select the row in the table and click one of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data set. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and variables.

- **Notes** — Click ... on the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Data Partition Node Train Properties

The following train properties are associated with the **HP Data Partition** node:

- **Variables** — Use the Variables property to specify the status for individual variables that are imported into the **HP Data Partition** node. Click ... to open a window containing the variables table. You can specify the **Partitioning Role** for individual variables, view column metadata, or open an Explore window to view a variable's sampling information, observation values, or a plot of variable distribution.

- **Partitioning Method** — Use the Partitioning Method property to specify the sampling method that you want to use when you are partitioning your data.

  You can choose from the following:

  - **Default** — When you select **Default**, if a class target variable or variables is specified, then the partitioning is stratified on the class target variables. Otherwise, simple random partitioning is performed. Note that if additional stratification variables are specified in the variables editor, they are used in addition to the target variables.
  - **Simple Random** — When you select **Simple Random**, every observation in the data set has the same probability of being written to one of the partitioned data sets.
  - **Stratified** — Use this option to specify variables to form strata (or subgroups) of the total population. Within each stratum, all observations have an equal probability of being written to one of the partitioned data sets. You perform stratified partitioning to preserve the strata proportions of the population within each partition data set. This might improve the classification precision of fitted models. If you select Stratified partitioning as your method and no class target variables are defined, then you must use the Variables window to set the partition role and specify a stratification variable.

- **Random Seed** — Use the Random Seed property to specify an integer value to use as a random seed for the pseudorandom number generator that chooses observations for sampling. The default value for the Random Seed property is 12345.

HP Data Partition Node Train Properties: Data Set Allocations

You can specify the percentages, or proportions of the source data that you want to allocate as Train, Validation, and Test data sets.

- **Training** — Use the Training property to specify the percentage of observations that you want to allocate to the training data set. Permissible values are real numbers between 0 and 100. The SAS default training percentage is 70%.
- **Validation** — Use the Validation property to specify the percentage of observations that you want to allocate to the validation data set. Permissible values are real numbers between 0 and 100. The SAS default validation percentage is 30%.

HP Data Partition Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Data Partition Node Variables Table

Use the table in the Variables — HP Data Partition window to identify cluster and stratification variables and to specify data partitioning methods for individual variables on a one-by-one basis. To open the Variables — HP Data Partition window, click ... (on the right of the **Variables** property in the properties panel) when the **Data Partition** node is selected in the diagram workspace.

Use this table to specify the partitioning role of specific variables. You can choose from the following roles: Cluster, Stratification, None, and Default. If a partition role is dimmed and unavailable, then that partition role is not applicable for that particular variable.

The following are read-only attributes: Role, Level, Type, Order, Label, Format, Informat, Length, Lower Limit, Upper Limit, Comment, Report, Creator, Family, and Distribution.

You can highlight a variable in the table and click **Explore** to get sampling, distribution, and metadata information in the Explore Variable window.

The table in the Variables — HP Data Partition window contains the following columns:

- **Name** — displays the name of the variable.
- **Partition Role** — Click the cell to specify the partitioning method that you want to use on an individual variable in your imported data set.
  - **Default** — uses the partitioning method that is specified for all variables in the **Data Partition** node properties panel. If the partitioning method that is specified for all variables in the properties panel is **Default**, then class variables are stratified and interval variables use simple random sampling.
  - **None** — Do not partition the specified class or interval variable.
  - **Stratification** — You specify variables from the input data set to form strata (or subsets) of the total population. Within each stratum, all observations have an equal probability of being selected for the sample. Across all strata, however, the observations in the input data set generally do not have equal probabilities of being selected for the sample. You perform stratified sampling to preserve the strata proportions of the population within the sample. This might improve the classification precision of fitted models.

The following are read-only columns in the Variables — HP Data Partition table. You must use a **Metadata** node if you want to modify any of the following information about Data Source variables that are imported into the **HP Data Partition** node.

- **Role** — displays the variable role
- **Level** — displays the level for class variables
- **Type** — displays the variable type
- **Order** — displays the variable order
- **Label** — displays the text label to be used for variables in graphs and output
- **Format** — displays the variable format
- **Informat** — displays the SAS Informat for variables
- **Length** — displays the variable length
- **Lower Limit** — displays the minimum value for variables
- **Upper Limit** — displays the maximum value for variables
- **Distribution** — the expected univariate distribution
- **Family** — identifies the general source of a variable (such as demographic, financial, or historic) for record keeping. Such values can be useful in constructing the data.
- **Report** — displays the Boolean setting for creating reports
- **Creator** — displays the user who created the process flow diagram
- **Comment** — displays the user-supplied comments about a variable

HP Data Partition Node Results

After a successful node run, you can open the Results window of the HP Data Partition node by clicking **Results** in the Run Status window. Or in the diagram workspace, right-click the HP Data Partition node and click **Results** on the pop-up menu.

Select **View** from the main menu to view the following results in the Results window:

- **Properties**
  - **Settings** — displays a window that shows how the **HP Data Partition** node properties were configured when the node last ran. Use the **Show Advanced Properties** check box at the bottom of the window to see all of the available properties.
  - **Run Status** — indicates the status of the **HP Data Partition** node run. The Run Start Time, Run Duration, and information about whether the run completed successfully are displayed in this window.
  - **Variables** — a table of the properties of the variables that are used in this node.
  - **Train Code** — the code that SAS Enterprise Miner used to train the node.
  - **Notes** — enables users to read notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Data Partition** node run.
  - **Output** — the SAS output of the **HP Data Partition** node run. The output displays a variable summary and summary statistics for class or interval targets or both in the original data and in the various partition data sets.
  - **Flow Code** — Flow Code is not available for the HP Data Partition node.
- **Scoring**
  - **SAS Code** — The **HP Data Partition** node does not generate SAS code.
  - **PMML Code** — The **HP Data Partition** node does not generate PMML code.
- **Summary Statistics**
  - **Partition** — The Partition window describes how many observations are contained in each data set allocation, displayed by target level.
- **Table** — displays a table that contains the underlying data that is used to produce the selected chart. The **Table** menu item is dimmed and unavailable unless a results chart is open and selected.
- **Plot** — displays the Graph wizard that you can use to create ad hoc plots that are based on the underlying data that produced the selected table.

Chapter 2
The HP Explore Node

Overview of the HP Explore Node

The **HP Explore** node enables you to obtain descriptive information about a training data set. Statistics such as mean, standard deviation, minimum value, maximum value, and percentage of missing values are generated for each input variable and displayed in tabular and graphical form. These descriptive statistics are important tools in the pre-model building process. For example, if the **HP Explore** node identifies variables with missing values, you can impute these values using the **HP Impute** node.

HP Explore Node Requirements

If your input data set contains a frequency variable, then the frequency variable must be an interval variable, and all observations must be positive integers.

If you are running the **HP Explore** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Explore Node Properties

HP Explore Node General Properties

The following general properties are associated with the **HP Explore** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Explore** node that is added to a diagram has a Node ID of HPExpl. The second **HP Explore** node that is added to a diagram has a Node ID of HPExpl2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Explore window. The Imported Data — HP Explore window contains a list of the ports that provide data sources to the **HP Explore** node. Click ... (to the right of the Imported Data property) to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.
- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Explore window. The Exported Data — HP Explore window contains a list of the output data ports for which the **HP Explore** node creates data when it runs. Click ... (to the right of the Exported Data property) to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.
- **Notes** — Click ... (to the right of the Notes property) to open a window that you can use to store notes, such as data or configuration information.

HP Explore Node Train Properties

The following train properties are associated with the **HP Explore** node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Click ... (to the right of the Variables property) to open a variables table. You can set the variable status to either **Use** or **Don't Use** in the table. You can set a variable's report status to **Yes** or **No**.

- **Maximum Number of Levels Cutoff** — specifies the maximum number of variable measurement levels. Variables with more levels than the value specified here are rejected. You can specify only positive integers in this property.
- **Quantiles** — specify **Yes** to compute quantile statistics.

HP Explore Node Train Properties: Robust Statistics

- **Winsorized Statistics** — specifies whether Winsorized statistics are computed for the input data set. Winsorization is a data manipulation technique that replaces all values beyond a given threshold with the values at that threshold. For example, a 95% Winsorization sets all data below the 2.5th to the 2.5th percentile and also sets all data above the 97.5th percentile to the 97.5th percentile. Specify **Yes** to compute statistics on the Winsorized data set.
- **Trimmed Statistics** — specifies whether trimmed statistics are computed for the input data set. Trimming is a data manipulation technique that removes all values beyond a given threshold. For example, a 95% trimming removes all data below the 2.5th percentile and all data above the 97.5th percentile. Specify **Yes** to compute statistics on the trimmed data set.
- **Cutoff** — specifies the percentage of data that is removed from the tails of the distribution. For example, set this value to 2.5 to create a 95% Winsorized or trimmed data set. Valid values for this property are real numbers between 0 and 50.

HP Explore Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Explore Node Results

After the **HP Explore** node runs successfully, you can open the Results — Explore window by right-clicking the node in the diagram workspace and clicking **Results** on the pop-up menu. The Results — Explore window contains an imputation summary table and a window that displays the node's output.

Select **View** from the main menu of the Explore Results window to view the following information:

- **Properties**
  - **Settings** — displays a window with a Read-Only table of the HP Explore node properties configuration when the node was last run. Use the **Show Advanced Properties** check box at the bottom of the window to see all of the available properties.
  - **Run Status** — indicates the status of the **HP Explore** node run. The Run Start Time, Run Duration, and information about whether the run completed successfully are displayed in this window.
  - **Variables** — a table of variables properties in the node.
  - **Train Code** — the code that SAS Enterprise Miner used to train the node.
  - **Notes** — enables users to read or create notes.
- **SAS Results**
  - **Log** — the SAS log of the **HP Explore** node run.
  - **Output** — the SAS output of the **HP Explore** node run. The SAS output displays how many and which types of variables are in the training data set. If you run in a grid environment, the output displays limited information about the connection to the grid.
  - **Flow Code** — the SAS code that is used to produce the output that the **HP Explore** node passes on to the next node in the process flow diagram.
- **Scoring**
  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.
  - **PMML Code** — The **HP Explore** node does not generate PMML code.
- **Plots**
  - **Class Variables** — created for each input class variable. The Class Plot displays the frequency of each measurement level. The **Subset** menu enables you to select the variable that is plotted.
  - **Interval Variables** — displays the descriptive statistics for each of the interval inputs. The **Subset** menu enables you to select the following statistics:
    - **Coefficient of Variation**
    - **Kurtosis**
    - **Maximum**
    - **Mean**
    - **Minimum**
    - **Number of Missing Values**
    - **Number of Non-Missing Values**
    - **Percentage of Missing Values**
    - **Skewness**
    - **Standard Deviation**

  - **Missing Values** — displays the total number of missing observations for each input variable.
- **Summary Statistics**
  - **Statistics Table** — provides a summary of the generated statistics for each variable. Note that if you are in a grid environment and partition the training data, the **HP Explore** node ignores the partitions and summarizes the entire data set. The information available in this table is as follows:
    - **Data Role** — the role of the data.
    - **Scale** — the level of the variable. Interval variables are assigned the label `VAR`, and class variables are assigned the label `CLASS`.
    - **Variable** — the name of the input variable.
    - **Missing** — the number of missing observations.
    - **Percent Missing** — the percentage of missing observations.
    - **Non Missing** — the number of nonmissing observations.
    - **Minimum** — the minimum value of the input variable.
    - **Mean** — the mean value of the input variable.
    - **Maximum** — the maximum value of the input variable.
    - **Standard Deviation** — the standard deviation of the input variable.
    - **Skewness** — the measure of skewness of the input variable.
    - **Kurtosis** — the kurtosis, also called steepness, of the input variable.
    - **Coefficient of Variation** — the standard deviation divided by the mean, a unit-less measure.
    - **Number of Levels** — the number of distinct levels for each class variable.
    - **Mode** — the modal value for each class variable.
    - **Mode Percent** — the percentage of observations that equal the mode.
  - **Winsorized Statistics** — displays summary statistics for each input variable in the Winsorized data set.
  - **Trimmed Statistics** — displays summary statistics for each input variable in the trimmed data set.
  - **Quantiles** — displays quantile information for each input variable.
- **Table** — opens the data table that corresponds to the graph that you have in focus.
- **Plot** — opens the Select a Chart plotting wizard so that you can plot the data in the table that you have in focus.

Chapter 3

The HP Forest Node

Overview of the HP Forest Node

The HP Forest node creates a predictive model called a forest. A forest consists of several decision trees that differ from each other in two ways. First, the training data for a tree is a sample without replacement from all available observations. Second, the input variables that are considered for splitting a node are randomly selected from all available inputs. In other respects, trees in a forest are trained like standard trees.

The HP Forest node accepts interval and nominal target variables. For an interval target, the procedure averages the predictions of the individual trees to predict an observation. For a categorical target, the posterior probabilities in the forest are the averages of the posterior probabilities of the individual trees. The node makes a second prediction by voting: the forest predicts the target category that the individual trees predict most often.

The training data for an individual tree excludes some of the available data. The data that is withheld from training is called the *out-of-bag sample*. An individual tree uses only the out-of-bag sample to form predictions. These are more reliable than predictions from training data.

Averaging over trees with different training samples reduces the dependence of the predictions on a particular training sample. Increasing the number of trees does not increase the risk of overfitting the data and can decrease it. However, if the predictions from different trees are correlated, then increasing the number of trees makes little or no improvement.

The HP Forest node overfits the training data when every tree overfits the training data. One way to mitigate overfitting in a tree without pruning is to require each leaf to contain many observations. The HP Forest node initializes the minimum leaf size to 0.1% of the available data and limits the number of leaves to one thousand. Use the **Smallest percentage of obs in node** property to adjust this value.

The three main training options for tree forests are the number of trees, the number of inputs for a node, and the sampling strategy. In the HP Forest node, the properties **Maximum Number of Trees**, **Number vars to consider in split search**, and **Proportion of obs in each sample** control these three parameters, respectively.

In each node of a decision tree, the HP Forest node randomly selects which input variables to consider for splitting the node, and ignores the rest of the available inputs. The selection ignores the predictive quality of the inputs. The intent is to insert random variation in the trees to reduce their correlation. If most inputs are predictive, then limiting the selection to a few of these can make sense. However, if most inputs are useless for prediction, then many inputs should be considered in a node in order to include at least one that it is predictive. Unfortunately, the number of predictive inputs is rarely known before you perform the analysis. Creating several small forests with different values of the **Number vars to consider in split search** option might suggest a value for this option.

The HP Forest node uses normalized, formatted values of categorical variables. It considers two categorical values the same if the normalized values are identical. Normalization removes any leading blank spaces from a value, converts lowercase characters to uppercase, and truncates all values to 32 bytes.

HP Forest Node Requirements

The HP Forest node must be preceded by a node that exports at least one raw or train data table. If your input data set contains a frequency variable, then the frequency variable must be an interval variable. Frequency variable observations that are non-positive are ignored.

HP Forest Node Properties

HP Forest Node General Properties

The following general properties are associated with the **HP Forest** node:

- **Node ID** — The **Node ID** property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Forest** node that is added to a diagram has a Node ID of HPDMForest. The second **HP Forest** node added to a diagram has a Node ID of HPDMForest2, and so on.

- **Imported Data** — The **Imported Data** property provides access to the Imported Data — HP Forest window. The Imported Data — HP Forest window contains a list of the ports that provide data sources to the **HP Forest** node. Select the ... button to the right of the **Imported Data** property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and click one of the following buttons:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and variables.

- **Exported Data** — The **Exported Data** property provides access to the Exported Data — HP Forest window. The Exported Data — HP Forest window contains a list of the output data ports that the **HP Forest** node creates data for when it runs. Select the ... button to the right of the **Exported Data** property to open a table that lists the exported data sets.

  If data exists for an exported data set, you can select the row in the table and click one of the following buttons:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data set. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and variables.

- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Forest Node Train Properties

The following train property is associated with the HP Forest node:

- **Variables** — Use the **Variables** property to specify the status for individual variables that are imported into the **HP Forest** node. Select the ... button to open a window containing the variables table. You can open an Explore window to view a variable's sampling information, observation values, or a plot of variable distribution.

HP Forest Node Train Properties: Tree Options

- **Maximum Number of Trees** — specifies the number of trees in the forest. The number of trees in the resulting forest can be less than the value specified here when the HP Forest node fails to split the training data for a tree. The HP Forest node attempts to create up to twice the number of trees specified. The default value is 50.

  The **Significance Level**, **Smallest number of obs in node**, and **Minimum Category Size** options constrain the split search to form trees that are more likely to predict well using new data. Setting all of these options to 1 generally frees the search algorithm to find a split and train a tree. However, a tree generated with such constraints might not help the forest predict well.

- **Seed** — Use the Seed property to specify an integer value to use as a random seed for the pseudorandom number generator that chooses observations for sampling. The default value for the Random Seed property is 12345.
- **Type of Sample** — specifies whether the number of observations or the percentage of observations is used to determine the training sample. Specify **Count** to use number of observations and **Proportion** to use percentage of observations.
- **Proportion of obs in each sample** — specifies what percentage of observations is used for each tree when the **Type of Sample** property is set to **Proportion**.
- **Number of obs in each sample** — specifies the number of observations that are used for each tree when the **Type of Sample** property is set to **Count**.

HP Forest Node Train Properties: Splitting Rule Options

- **Maximum Depth** — specifies the maximum depth of a node in any tree that the HP Forest node creates. The root node has depth 0. The children of the root have depth 1, and so on. The smallest acceptable value is 1. The default value is 50.
- **Missing Values** — specifies how the training procedure handles an observation with missing values. If you specify **Use in Search** and enough training observations in the node are missing the value of the candidate variable, then the missing value is used as a separate value in the test of association and the split search. If you specify **Distribute**, observations with a missing value of the candidate variable are omitted from the test of association and split search in that node. A splitting rule distributes such an observation to all branches. See Missing Values on page 19 for a more complete explanation. The default value of policy is **Use in Search**.
- **Minimum Use in Search** — specifies the minimum number of observations with a missing value in a node to initiate the **Use in Search** option for missing values. See Missing Values on page 19 for a more complete explanation. The default value is 1.
- **Number vars to consider in split search** — specifies the number of input variables to consider splitting on in a node. Valid values are integers between 1 and the number of variables in the training data, inclusive.
- **Significance Level** — specifies a threshold p-value for the significance level of a test of association of a candidate variable with the target. If no association meets this threshold, the node is not split. The default value is 0.05.
- **Max Categories in Split Search** — specifies the maximum number of categories of a nominal candidate variable to use in the association test. This value refers only to the categories that are present in the training data in the node and that satisfy the **Minimum Category Size** option. The categories are counted independently in each node. If more categories are present than the value specified here, then the least frequent categories are removed from the association test. Many infrequent categories can dilute a strong predictive ability of common categories. The search for a splitting rule uses all categories that satisfy the **Minimum Category Size** option. The value specified here must be a positive integer. The default value is 30.
- **Minimum Category Size** — specifies the minimum number of observations that a given nominal input category must have in order to use the category in a split search. Categorical values that appear in fewer observations than the value specified here are handled as if they were missing. The policy for assigning such observations to a

branch is the same as the policy for assigning missing values to a branch. The default value is 5.

- **Exhaustive** — specifies the maximum number of splits to examine in a complete enumeration of all possible splits when the input variable is nominal and the target has more than two nominal categories. The exhaustive method of searching for a split examines all possible splits. If the number of possible splits is greater than the value specified here, then a heuristic search is done instead of an exhaustive search. The default value is 5,000.

HP Forest Node Train Properties: Node Options

- **Method for Leaf Size** — specifies the method used to determine the leaf size value. Specify **Default** to let the HP Forest node determine the method that is used for each leaf. Specify **Count** to use number of observations and **Proportion** to use percentage of observations.
- **Smallest percentage of obs in node** — specifies the smallest number of training observations that a new branch can have, expressed as the fraction of the number of available observations. The number of available observations does not include observations with missing target values and observations with a nonpositive value for the frequency variable. The value specified here must be between 0 and 1, exclusive. The default value is 0.001.
- **Smallest number of obs in node** — specifies the smallest number of training observations a new branch can have. The default value is 5.
- **Split Size** — specifies the requisite number of training observations a node must have for the HP Forest node to consider splitting it. The default value is either twice the value specified by the **Smallest number of obs in node** property or twice the number of observations implied by the **Smallest percentage of obs in node** property. The HP Forest node counts the number of observations in a node without adjusting for the frequency variable.

HP Forest Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Forest Node Details

Bagging the Data

A decision tree in a forest trains on new training data that is derived from the original training data presented to the HP Forest node. Training different trees with different training data reduces the correlation of the predictions of the trees, which in turn should improve the predictions of the forest.

The HP Forest node samples the original data without replacement to create the training data for an individual tree. The word *bagging* stems from "bootstrap aggregating," where "bootstrap" refers to a procedure that uses sampling with replacement. Observations that are excluded from the sample as called *out-of-bag* (OOB) observations. Therefore, observations in the training sample are called the *bagged* observations, and the training data for a specific decision tree is called the bagged data.

The **Number of obs in each sample** and **Proportion of obs in each sample** properties specify the number of observations to sample without replacement into a bagged data set. Estimating the goodness-of-fit of the model by using the training data is usually too optimistic; the fit of the model to new data is usually worse than the fit to the training data. Estimating the goodness-of-fit by using the out-of-bag data is usually too pessimistic at first. With enough trees, the out-of-bag estimates are an unbiased estimate of the generalization fit.

Training a Decision Tree

The HP Forest node trains a decision tree by forming a binary split of the bagged data, and then forming a binary split of each of the segments. This process is repeated recursively until some constraint is met.

Creating a binary split involves the following tasks:

- selecting candidate inputs
- reducing the number of nominal input categories
- computing the association of each input with the target
- searching for the best split using the most highly associated input

The HP Forest node selects candidate inputs independently in every node. The purpose of preselecting candidate inputs is to increase the differences among the trees, thereby decreasing the correlation and theoretically increasing the quality of the forest predictions. The selection is random, and each input has the same chance to be selected. The **Number vars to consider in split search** property specifies the number of candidates to select. The quality of the forest often depends on the number of candidates. Unfortunately, a good value for **Number vars to consider in split search** is usually not known in advance. Data with more irrelevant variables generally warrants a larger value.

The reason for searching only one input variable for a splitting rule instead of searching all inputs and choosing the best split is to improve prediction on new data. An input that offers more splitting possibilities provides the search routine more chances to find a spurious split. Preselecting the input variable and then searching only on that one input reduces the likelihood of producing a spurious split. The HP Forest node preselects the input with the largest p-value of an asymptotic permutation distribution of an association statistic.

The HP Forest node sometimes reduces the number of categories of a nominal input. Nominal inputs with fewer categories in the node than the number specified in the **Max Categories in Split Search** property are not modified. For nominal inputs with more categories, the procedure ignores observations with the least frequent category values. Limiting the number of categories in a nominal input can strengthen the association of that input with the target by eliminating categories of less predictive potential. The HP Forest node reduces the categories independently in every node.

The split search seeks to maximize the reduction in the Gini index for a nominal target and the reduction in variance of an interval target.

Predicting an Observation

To predict an observation, the HP Forest node assigns the observation to a single leaf in each decision tree in the forest. That leaf is used to make a prediction based on the tree that contains the leaf. Finally, the HP Forest node averages the predictions over all the trees. For an interval target, the prediction in a leaf equals the average of the target values among the bagged training observations in that leaf. For a nominal target, the posterior probability of a target category equals the proportion of that category among the bagged training observations in that leaf. The predicted nominal target category is the category with the largest posterior probability. In case of a tie, the first category that occurs in the training data is the prediction.

The HP Forest node also computes out-of-bag predictions. The out-of-bag prediction of an observation uses only trees for which the observation is out-of-bag. (That is, the observation is not selected as part of the training data for that tree.)

A model is worthless if its predictions are no better than predictions without a model. For an interval target, the no-model prediction of an observation is the average of the target among training observations. For a nominal target, the no-model posterior probabilities are the class proportions in the training data. The no-model predictions are the same for every observation.

Missing Values

When the value of **Missing Values** is set to **Use in Search**, missing values are treated as separate, legitimate values. However, this is true provided that the number of observations that contain missing values is greater than or equal to the value specified in the **Minimum Use in Search** property. By default, the HP Forest node uses missing values in the split search, and the value of **Minimum Use in Search** is set to 1.

Assuming that the split search uses the missing values, the search will find a rule that associates missing values with a branch that maximizes the worth of the split. For a nominal input variable, a new nominal category that represents missing values is created for the duration of the split search. For an ordinal or interval input variable, a rule preserves the ordering of the nonmissing values when assigning them to branches, but might assign missing values to any single branch. It is possible to produce a branch exclusively for missing values. This is desirable when the existence of a missing value is predictive of a target value.

If the split search does not use missing values, the resulting rule will distribute observations with a missing value to the branches. The observation is in effect copied, one copy for each branch. The copy assigned to a branch is given a fractional frequency proportional to the number of training observations assigned to the branch. The prediction of an observation distributed to the branches is the same as the prediction of the observation in the parent of the branches.

Specify **Distribute** for the **Missing Values** property to force every splitting rule to distribute an observation to the branches when the value of the splitting variable is missing. Specify **Use in Search** for the **Missing Values** property. This produces rules that distribute observations if the splitting variable has no missing values in the training data, or when the number of observations with missing values is less than the value specified in the **Minimum Use in Search** property.

Observations that are distributed into multiple branches might slow down training noticeably. Values of distributed observations in a leaf are stored in a linked list and passed to the association and split-search routines individually. Values of non-distributed observations (observations that reside entirely within one leaf) are passed as a single vector. Processing a single vector of values is much faster than processing a linked list and calling an accumulation routine separately for each value.

The logic described in this section applies independently to each candidate variable and each node. For example, the test of association using Variable A might use all observations, and the test using input Variable B might ignore some observations because of missing values. Or, the test using Variable A might use all observations in one node but not all in another node.

Unseen Categorical Values

A splitting rule that uses a categorical variable might not recognize all possible values of the variable. Some categories might not be in the training data. Others might be so infrequent in the within-node training sample that the HP Forest Node excluded them. The **Minimum Category Size** property specifies the minimum number of occurrences that are required for a categorical value to participate in the search for a splitting rule. Splitting rules treat observations with unseen categorical values exactly as they treat observations with missing values.

HP Forest Node Results

After a successful node run, you can open the Results window of the HP Forest node by right-clicking the node and selecting **Results** from the pop-up menu.

Select **View** from the main menu in the Results window to view the following:

- **Properties**
  - **Settings** — displays a window with a read-only table of the HP Forest node properties configuration when the node was last run.
  - **Run Status** — displays the status of the HP Forest node run. Information about the run start time, run duration, and completion status are displayed in this window.
  - **Variables** — displays a table of the variables in the training data set.
  - **Train Code** — displays the code that SAS Enterprise Miner used to train the node.
  - **Notes** — displays notes of interest, such as data or configuration information.
- **SAS Results**
  - **Log** — the SAS log of the HP Forest run.
  - **Output** — the SAS output of the HP Forest run.

- **Flow Code** — the HP Forest node does not generate flow code.

- **Scoring**
  - **SAS Code** — the HP Forest node does not generate SAS code.
  - **PMML Code** — the HP Forest node does not generate PMML code.

- **Assessment**
  - **Fit Statistics** — displays a table of fit statistics from the model.
  - **Residual Statistics** — The Residual Statistics plot displays a box plot for the residual measurements when the target is interval.
  - **Classification Chart** — displays a stacked bar chart of the classification results for a categorical target variable. The horizontal axis displays the target levels that observations actually belong to. The color of the stacked bars identifies the target levels that observations are classified into. The height of the stacked bars represents the percentage of total observations.
  - **Score Rankings Overlay** — several statistics for each decile (group) of observations are plotted on the vertical axis. For a binary target, all observations in the scored data set are sorted by the posterior probabilities of the event level in descending order. For a nominal or ordinal target, observations are sorted from highest expected profit to lowest expected profit (or from lowest expected loss to highest expected loss). Then the sorted observations are grouped into deciles and observations in a decile are used to calculate the statistics that are plotted in deciles charts. The Score Rankings Overlay plot displays both train and validate statistics on the same axis.
  - **Score Distribution** — The Score Distribution chart plots the proportions of events (by default), nonevents, and other values on the vertical axis. The values on the horizontal axis represent the model score of a bin. The model score depends on the prediction of the target and the number of buckets used.

    For categorical targets, observations are grouped into bins, based on the posterior probabilities of the event level and the number of buckets. The Score Distribution chart of a useful model shows a higher percentage of events for higher model score and a higher percentage of nonevents for lower model scores.

    For interval targets, observations are grouped into bins, based on the actual predicted values of the target. The default chart choice is Percentage of Events. Multiple chart choices are available for the Score Distribution Chart.

- **Model** — graphs and tables with information about the variables in the model. The available graphs and tables are as follows:
  - **Baseline Fit Statistics** — a read-only table with information about the baseline model.
  - **Iteration History** — a read-only table with information about each iteration of the model-building process.
  - **Iteration Plot** — displays goodness-of-fit statistics plotted against the number of trees in the forest.
  - **Leaf Plot** — displays the total number of leaves in the forest plotted against the number of trees in the forest. The bar chart indicates how many leaves were added with each tree in addition to the total number of leaves.
  - **Leaf Statistics** — a histogram that displays the distribution for the number of leaves in each tree.

  - **Variable Importance** — a read-only table with information about each variable’s worth to the model.
- **Table** — displays a table that contains the underlying data that is used to produce a chart. The Table menu item is dimmed and unavailable unless a results chart is open and selected.
- **Plot** — use the Graph Wizard to modify an existing Results plot or create a Results plot of your own. The Plot menu item is dimmed and unavailable unless a Results chart or table is open and selected.

Chapter 4

The HP Impute Node

Overview of the HP Impute Node

Use the **HP Impute** node to replace missing values in data sets that are used for data mining. The **HP Impute** node is typically used during the modification phase of the Sample, Explore, Modify, Model, and Assess (SEMMA) SAS data mining methodology.

Data mining databases often contain observations that have missing values for one or more variables. Missing values can result from the following: data collection errors; incomplete customer responses; actual system and measurement failures; or from a revision of the data collection scope over time (such as tracking new variables that were not included in the previous data collection schema).

If an observation contains a missing value, then by default that observation is not used for modeling by nodes such as **HP Neural** or **HP Regression**. However, rejecting all incomplete observations might ignore useful or important information that is still contained in the nonmissing variables. Rejecting all incomplete observations might also bias the sample, since observations that are missing values might have other things in common as well.

Choosing the "best" missing value replacement technique inherently requires the researcher to make assumptions about the true (missing) data. For example, researchers often replace a missing value with the mean of the variable. This approach assumes that the variable's data distribution follows a normal population response. Replacing missing values with the mean, median, or another measure of central tendency is simple. But it can greatly affect a variable's sample distribution. You should use these replacement statistics carefully and only when the effect is minimal.

Another imputation technique replaces missing values with the mean of all other responses given by that data source. This assumes that the input from that specific data source conforms to a normal distribution. Another technique studies the data to see whether the missing values occur in only a few variables. If those variables are determined to be insignificant, the variables can be rejected from the analysis. The observations can still be used by the modeling nodes.

The **HP Impute** node provides the following imputations for missing interval variables:

- Default Constant Value
- Mean
- Maximum
- Minimum
- Midrange
- None

The **HP Impute** node provides the following imputations for missing class variables:

- Count
- Default Constant Value
- Distribution
- None

You can customize the default imputation statistics by specifying your own replacement values for missing and nonmissing data. You replace missing values for the training, validation, test, and score data sets by using imputation statistics that are calculated from the active training predecessor data set.

The **HP Impute** node must follow an **Input Data** node, **SAS Code** node, or other high-performance node.

HP Impute Node Requirements

If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

If you are running the **HP Impute** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Impute Node Properties

HP Impute Node General Properties

The following general properties are associated with the **HP Impute** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Impute** node that is added to a diagram has a Node ID of HPImp. The second **HP Impute** node that is added to a diagram has a Node ID of HPImp2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Impute window. The Imported Data — HP Impute window contains a list of the ports that provide data sources to the **HP Impute** node. Click ... (to the right of the Imported Data property) to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Impute window. The Exported Data — HP Impute window contains a list of the output data ports for which the **HP Impute** node creates data when it runs. Click ... (to the right of the Exported Data property) to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Notes** — Click ... (to the right of the Notes property) to open a window that you can use to store notes, such as data or configuration information.

HP Impute Node Train Properties

The following train properties are associated with the **HP Impute** node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Click ... (to the right of the Variables property) to open a variables table. You can set the variable status to either **Use** or **Don't Use** in the table. You can set a variable's report status to **Yes** or **No**.

- **Non Missing Variables** — Use the Non Missing Variables property of the **HP Impute** node to specify if you want to impute variables with no missing values. The default setting for the Non Missing Variables property is **No**.
- **Missing Cutoff** — Use the Missing Cutoff property of the **HP Impute** node to specify the maximum percent of missing values that are allowed for a variable to be imputed. Variables whose percentage of missing exceeds this cutoff are ignored. The default setting for the Missing Cutoff property is 50%.

HP Impute Node Train Properties: Class Variables

- **Default Input Method** — Use the Default Input Method property of the **HP Impute** node to specify the imputation statistic that you want to use to replace missing class variables. The choices are as follows:
  - **Count** — Use the Count setting to replace missing class variable values with the most frequently occurring class variable value.
  - **Default Constant Value** — Use the Default Constant setting to replace missing class variable values with the value that you enter in the Default Character Value property.
  - **Distribution** — Use the Distribution setting to replace missing class variable values with replacement values that are calculated based on the random percentiles of the variable's distribution. In this case, the assignment of values is based on the probability distribution of the nonmissing observations. The distribution imputation method typically does not change the distribution of the data very much.
  - **None** — Missing class variable values are not imputed under the None setting.
- **Default Target Method** — Use the Default Target Method property of the **HP Impute** node to specify the imputation statistic that you want to use to replace missing class target variables. The choices are as follows:
  - **Count** — Use the Count setting to replace missing target variable values with the most frequently occurring target variable value.
  - **Default Constant Value** — Use the Default Constant setting to replace missing target variable values with the value that you enter in the Default Character Value property.
  - **Distribution** — Use the Distribution setting to replace missing target variable values with replacement values that are calculated based on the random percentiles of the variable's distribution. In this case, the assignment of values is based on the probability distribution of the nonmissing observations. The distribution imputation method typically does not change the distribution of the data very much.
  - **None** — Missing target variable values are not imputed under the None setting.

HP Impute Node Train Properties: Interval Variables

- **Statistics** — specifies which statistics are used during the imputation process. Select **Data** to base the statistics on the entire data set, **Winsorized** to base the statistics on the Winsorized data set, and **Trimmed** to base the statistics on the trimmed data set. If you select **Winsorized** or **Trimmed**, you must specify a cutoff value by using the **Cutoff** property. If the **Cutoff** property is not available, choose **Data**.

- **Cutoff** — specifies the percentage of data that is removed from the tails of the distribution. For example, setting this value to 2.5 creates a 95% Winsorized or trimmed data set. Valid values for this property are real numbers between 0 and 50.
- **Default Input Method** — Use the Method Interval property of the **HP Impute** node to specify the imputation statistic that you want to use to replace missing interval variables. The choices are as follows:
  - **Mean** — Use the Mean setting to replace missing interval variable values with the arithmetic average, which is calculated as the sum of all values divided by the number of observations. The mean is the most common measure of a variable's central tendency. It is an unbiased estimate of the population mean. The mean is the preferred statistic to use to replace missing values if the variable values are at least approximately symmetric (for example, a bell-shaped normal distribution). Mean is the default setting for the Default Input Method for interval variables.
  - **Maximum** — Use the Maximum setting to replace missing interval variable values with the maximum value for the variable.
  - **Minimum** — Use the Minimum setting to replace missing interval variable values with the minimum value for the variable.
  - **Midrange** — Use the Midrange setting to replace missing interval variable values with the maximum value for the variable plus the minimum value for the variable divided by two. The midrange is a rough measure of central tendency and is easy to calculate.
  - **Pseudo-Median** — Use the Pseudo-Median setting to replace missing interval variable values with the pseudo-median. The pseudo-median is the median of all possible midpoints of pairs of observations.
  - **Default Constant Value** — Use the Default Constant Value setting to replace missing interval variable values with the value that you enter in the Default Character Value property.
  - **None** — Specify the None setting if you do not want to replace missing interval variable values.
- **Default Target Method** — Use the Default Target Method property of the **HP Impute** node to specify the imputation statistic that you want to use to replace missing target variables. The choices are as follows:
  - **Mean** — Use the Mean setting to replace missing target variable values with the arithmetic average, which is calculated as the sum of all values divided by the number of observations. The mean is the most common measure of a variable's central tendency. It is an unbiased estimate of the population mean. The mean is the preferred statistic to use to replace missing values if the variable values are at least approximately symmetric (for example, a bell-shaped normal distribution). Mean is the default setting for the Default Target Method for interval variables.
  - **Maximum** — Use the Maximum setting to replace missing interval variable values with the maximum value for the variable.
  - **Minimum** — Use the Minimum setting to replace missing interval variable values with the minimum value for the variable.
  - **Midrange** — Use the Midrange setting to replace missing target variable values with the maximum value for the variable plus the minimum value for the variable divided by two. The midrange is a rough measure of central tendency that is easy to calculate.

- **Pseudo-Median** — Use the Pseudo-Median setting to replace missing interval variable values with the pseudo-median. The pseudo-median is the median of all possible midpoints of pairs of observations.
- **Midrange** — Use the Midrange setting to replace missing interval variable values with the maximum value for the variable plus the minimum value for the variable divided by two. The midrange is a rough measure of central tendency and is easy to calculate.
- **Default Constant Value** — Use the Default Constant Value setting to replace missing target variable values with the value that you enter in the Default Character Value property.
- **None** — Specify the None setting if you do not want to replace missing target variable values.

HP Impute Node Train Properties: Default Constant Value

Use the Default Constant properties to specify how you want to manage numeric and character constant values to be used during imputation.

- **Default Character Value** — Use the Default Character Value property of the **HP Impute** node to specify the character string or value that you want to use during constant character imputation.
- **Default Number Value** — Use the Default Number Value property of the **HP Impute** node to specify the numeric value that you want to use as a constant value for numeric imputation. The Default Number Value property defaults to a setting of 0.0.

HP Impute Node Train Properties: Method Options

- **Random Seed** — Use the Random Seed property of the **HP Impute** node to specify the initial Random Seed value that you want to use for random number generation. The default Random Seed value is 12345.

HP Impute Node Score Properties

The following score properties are associated with the **HP Impute** node:

- **Hide Original Variables** — Set the Hide Original Variables property of the **HP Impute** node to **No** if you want to keep the original variables in the exported metadata from your imputed data set. In that case, the variables are exported with a role of Rejected. The default setting for the Hide Original Variables is **Yes**. When Set to **Yes**, the original variables are removed only from the exported metadata, and not removed from the exported data sets and data views.
- **Reject Based on Missing Values** — Specify **Yes** to reject all variables with a percentage of missing values that exceeds the value specified in the **Missing Cutoff** property. In many cases, variables with a large percentage of missing values should not be imputed. Instead, they should be rejected. If you specify **No**, variables with a percentage of missing values that exceeds the value specified in the **Missing Cutoff** property are imputed, but are not rejected.

HP Impute Node Score Properties: Indicator Variables

- **Type** — Use the Type property of the **HP Impute** node to specify the type of indicator variables used to flag the imputed observations for each variable.

  You can choose from the following settings:

  - **None** — (default setting) Do not create an indicator variable.
  - **Single** — A single indicator variable is created to indicate that one or more variables were imputed.
  - **Unique** — Unique binary indicator variables are created for every imputed variable.
- **Source** — Use the Indicator Variable Source property of the **HP Impute** node to specify the role that you want to assign to the created indicator variables. You can choose between **Missing Variables** and **Imputed Variables**. The default setting is **Imputed Variables**.
- **Role** — Use the Indicator Variable Role property of the **HP Impute** node to specify the role that you want to assign to the created indicator variables. You can choose between the roles **Rejected** and **Input**. The default setting is **Rejected**.

HP Impute Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

Using Indicator Variables with the HP Impute Node

Use the Indicator Variables property to create flag variables that identify each variable's imputed observations. Select the check box to create flag variables for each imputed input variable. Flag variables are named by concatenating the prefix M_ with the input variable's name. Each flag variable contains an indicator value of 1 if the observation was imputed and an indicator value of 0 if it was not. For example, assume that you have two input variables named AGE and HEIGHT that contain the following values prior to imputation:

Age	Height
25	73
.	66
30	.

When you run the node, one or more imputed flag variables are created, depending on the setting that is configured for the Indicator Variable property:

Table 4.1 Indicator Variable Setting Is Set to UNIQUE

Age	Height	M_AGE	M_HEIGHT
25	73	0	0
34	66	1	0
30	62	0	1

Table 4.2 Indicator Variable Setting Is Set to SINGLE

Age	Height	M_VARIABLE
25	73	0
34	66	1
30	62	1

Imputation Methods

Interval Imputation Methods

To set the default interval imputation statistic, click the **Default Input Method** or **Default Target Method** property drop-down arrow and select one of the following imputation methods:

- Mean — (default) or the arithmetic average, calculated as the sum of all values divided by the number of observations. The mean is the most common measure of a variable's central tendency. It is an unbiased estimate of the population mean. The mean is the preferred statistic to use to replace missing values if the variable values are at least approximately symmetric (for example, a bell-shaped normal distribution).
- Maximum — All replacements are equal to the maximum value for the variable.
- Minimum — All replacements are equal to the minimum value for the variable.

- Mid-range — the maximum plus the minimum divided by two. The midrange is a rough measure of central tendency and is easy to calculate.
- Default Constant — All replacements are equal to a fixed constant value. You configure the fixed constant value using the **Default Character Value** and the **Default Number Value** properties.
- None — Do not impute the missing values.

Class Imputation Statistics

Missing values for class variables can be replaced with one of the following statistics:

- Count — Missing values are replaced with the modal value.
- Default Constant Value — All replacements are equal to a fixed constant value. You configure the fixed constant value using the Default Character Value and the Default Number Value properties.
- Distribution — Replacement values are calculated based on the random percentiles of the variable's distribution. In this case, the assignment of values is based on the probability distribution of the nonmissing observations. This imputation method typically does not change the distribution of the data very much.
- None — Replacement values are not imputed, and are left as missing.

If several values occur with the same frequency and Count is the class imputation statistic, the smallest value is used as the Count. If the Count is a missing value, then the next most frequently occurring nonmissing value is used for data replacement.

HP Impute Node Results

After the **HP Impute** node successfully runs, you can open the Results — Impute window by right-clicking the node in the Diagram Workspace and selecting **Results** from the pop-up menu. The Results — Impute window contains an imputation summary table and a window that displays the node's output.

Select **View** from the main menu of the Impute Results window to view the following information:

- **Properties**
  - **Settings** — displays a window with a read-only table of the **HP Impute** node properties configuration when the node was last run. Use the **Show Advanced Properties** check box at the bottom of the window to see all of the available properties.
  - **Run Status** — indicates the status of the **HP Impute** node run. The Run Start Time, Run Duration, and information about whether the run completed successfully are displayed in this window.
  - **Variables** — a table of variable properties of the node.
  - **Train Code** — the code that SAS Enterprise Miner used to train the node.
  - **Notes** — enables users to read notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Impute** node run.

  - **Output** — the SAS output of the **HP Impute** node run. The SAS output includes a variable summary, a distribution of missing observations in training data table, and an imputation summary by variable.
  - **Flow Code** — the SAS code that is used to produce the output that the **HP Impute** node passes on to the next node in the process flow diagram.
- **Scoring**
  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.
  - **PMML Code** — The **HP Impute** node does not generate PMML code.
- **Model**
  - **Imputation Summary** — a table that shows a summary of the imputation run. The Imputation Summary table displays the following: the variable name, the imputation method, the imputed variable name, the variable role, the variable level, the variable type (numeric or character), the variable label (if any), and the number of missing values for the train data set, the validation data set, and the test data set. The Imputation Summary table is a subset of the SAS output from the **HP Impute** node run.
- **Table** — opens the data table that corresponds to the graph that you have in focus.
- **Plot** — opens the Select a Chart plotting wizard so that you can plot the data in the table that you have in focus.

Chapter 5
The HP Neural Node

Overview of the HP Neural Node

The **HP Neural** node creates multilayer neural networks that pass information from one layer to the next in order to map an input to a specific category or predicted value. The **HP Neural** node enables this mapping to take place in a distributed computing environment, which enables you to build neural networks on massive data sets in a relatively short amount of time.

A neural network consists of units (neurons) and connections between those units. There are three types of units:

- Input Units — obtain the values of input variables and standardize those values.
- Hidden Units — perform internal computations and provide the nonlinearity that makes neural networks powerful.
- Output Units — compute predicted values and compare those values with the values of the target variables.

Units pass information to other units through connections. Connections are directional and indicate the flow of computation within the network. Connections cannot form loops, because the **HP Neural** node permits only feed-forward networks. The following restrictions apply to feed-forward networks:

- Input units can be connected to hidden units or to output units.
- Hidden units can be connected to other hidden units or to output units.
- Output units cannot be connected to other units.

Each unit produces a single computed value. For input and hidden units, this computed value is passed along the connections to other hidden or output units. For output units, the computed value is the predicted value. The predicted value is compared with the target value to compute the error function, which the training method attempts to minimize.

The **HP Neural** node was designed with two goals. First, the **HP Neural** node aims to perform efficient, high-speed training of neural networks. Second, the **HP Neural** node attempts to create accurate, generalizable models in an easy to use manner. With these goals in mind, most parameters for the neural network are selected automatically. This includes standardization of input and target variables, activation and error functions, and termination of model training.

HP Neural Node Requirements

The **HP Neural** node requires one or more input variables and one or more target variables. The inputs and targets can be binary, nominal, or interval. All ordinal variables are ignored during training. If an observation has missing values for any of the specified target variables, the observation is not used for training or for computing validation error.

If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

If you are running the **HP Neural** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Neural Node Properties

HP Neural Node General Properties

The following general properties are associated with the **HP Neural** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Neural** node that is added to a diagram has a Node ID of HPDMNeural. The second **HP Neural** node added to a diagram has a Node ID of HPDMNeural2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Neural window. The Imported Data — HP Neural window contains a list of

the ports that provide data sources to the **HP Neural** node. Select the ... button to the right of the Imported Data property to open a table of the imported data.

If data exists for an imported data source, you can select the row in the imported data table and click for the desired option:

- **Browse** to open a window where you can browse the data set.
- **Explore** to open the Explore window, where you can sample and plot the data.
- **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.

- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Neural window. The Exported Data — HP Neural window contains a list of the output data ports that the **HP Neural** node creates data for when it runs. Select the ... button to the right of the Exported Data property to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and click as follows:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.

- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Neural Node Train Properties

The following train properties are associated with the **HP Neural** node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Select the ... button to the right of the Variables property to open a variables table. You can set the variable status to either **Use** or **Don't Use** in the table, and you can set a variable's report status to **Yes** or **No**.

- **Use Inverse Priors** — specifies whether inverse priors are used to weight training observations in the presence of a class target. This is especially helpful when you have rare events. When you specify **Yes**, a weight is calculated as the inverse of the fraction of time that the target class occurs in the input data set. It is applied to the prediction error of each nominal target variable. The default value for this property is **No**.

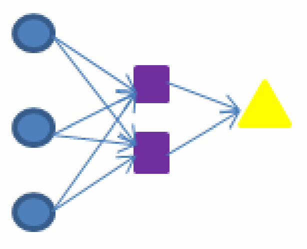
- **Create Validation** — specifies whether a validation data set is created from the incoming training data set. When you specify **Yes**, every fourth observation is used to generate the validation data set. When you specify **No**, the entire data set is used for training. Training continues until the optimizer can no longer improve the total training error or the number of iterations specified in **Maximum Iterations** is reached. Note that if a validation data set was created with the HP Data Partition node, validation is performed with that data set, and this property is ignored.

  There are cases when it is useful to determine how well a neural network can fit a set of data with a particular number of neurons. This is accomplished by setting **Create**

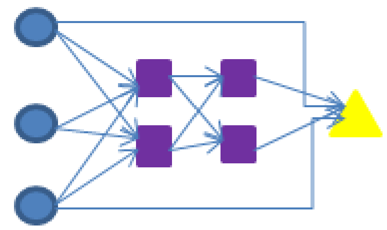
Validation to **No**. In this case, training does not stop when the validation error no longer improves. It continues until the maximum number of iterations have been met, or until no further improvement in total training error is seen.

- **Number of Hidden Neurons** — specifies the number of hidden neurons within the network. The number of neurons is split equally across the hidden layers. A good strategy is to start with a small number of hidden neurons and slowly increase the number until the validation error stops improving.
- **Architecture** — specifies the network architecture that you want to use to train the neural network. You can specify **Logistic**, **One Layer**, **One Layer with Direct**, **Two Layers**, or **Two Layers with Direct**. The options **One Layer with Direct** and **Two Layers with Direct** enable direct connections from the input units to the output units.

  The simplest network that can be modeled is the One Layer network. Each input unit is connected to each hidden unit and each hidden unit is connected to the output unit, as demonstrated in the image below.



  The most complex network available is the Two Layers with Direct. This adds an additional hidden layer to the network as well as direct connections between inputs and targets, as demonstrated in the image below.



- **Number of Tries** — specifies the number of times, or tries, that the network is to be retrained, using a different set of initial weights with each try. Because training involves optimizing a nonlinear objective function, this provides one way to be reasonably sure that a good set of weights is found. The default value is **2**.
- **Maximum Iterations** — specifies the maximum number of iterations allowed within each try. The default value is **50**.
- **Use Missing as Level** — specifies if missing values should be considered as their own classification level.

HP Neural Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.

- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node ran.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies whether the node was created by a user as a SAS Enterprise Miner extension node.

HP Neural Node Results

After the **HP Neural** node successfully runs, you can open the Results — Neural window. Right-click the node in the Diagram Workspace and select **Results** from the pop-up menu. The Results — Neural window contains an imputation summary table and a window that displays the node's output.

Select **View** from the main menu of the Neural Results window to view the following information:

- **Properties**
  - **Settings** — displays a window with a read-only table of the **HP Neural** node properties configuration when the node was last run. Use the **Show Advanced Properties** check box at the bottom of the window to see all of the available properties.
  - **Run Status** — indicates the status of the **HP Neural** node run. The Run Start Time, Run Duration, and information about whether the run completed successfully are displayed in this window.
  - **Variables** — a table of variable properties of the node.
  - **Train Code** — the code that SAS Enterprise Miner used to train the node.
  - **Notes** — enables users to read notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Neural** node run.
  - **Output** — the SAS output of the **HP Neural** node run. The SAS output includes a variable summary, a distribution of missing observations in the training data table, and an imputation summary by variable.
  - **Flow Code** — the SAS code that is used to produce the output that the **HP Neural** node passes on to the next node in the process flow diagram.
- **Scoring**
  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.
  - **PMML Code** — the **HP Neural** node does not generate PMML code.
- **Assessment**
  - **Fit Statistics** — opens the Fit Statistics window, which contains fit statistics from the model that are calculated for interval and non-interval targets.

- **Classification Chart** — displays a stacked bar chart of the classification results for a categorical target variable. The horizontal axis displays the target levels that observations actually belong to. The color of the stacked bars identifies the target levels that observations are classified into. The height of the stacked bars represents the percentage of total observations.
- **Score Rankings Overlay** — In a score rankings chart, several statistics for each decile (group) of observations are plotted on the vertical axis.

  The Score Rankings Overlay plot displays both train and validate statistics on the same axis. By default, the horizontal axis of a score rankings chart displays the deciles (groups) of the observations. The vertical axis displays the following values, and their mean, minimum, and maximum (if any):

  - posterior probability of target event
  - number of events
  - cumulative and noncumulative lift values
  - cumulative and noncumulative % response
  - cumulative and noncumulative % captured response
  - gain
  - actual profit or loss
  - expected profit or loss.
- **Score Distribution** — The Score Distribution chart plots the proportions of events (by default), nonevents, and other values on the vertical axis. The values on the horizontal axis represent the model score of a bin. The model score depends on the prediction of the target and the number of buckets used.

  For categorical targets, observations are grouped into bins, based on the posterior probabilities of the event level and the number of buckets.

  The Score Distribution chart of a useful model shows a higher percentage of events for higher model scores and a higher percentage of nonevents for lower model scores. For interval targets, observations are grouped into bins, based on the actual predicted values of the target. The default chart choice is Percentage of Events. Multiple chart choices are available for the Score Distribution Chart. The chart choices are as follows:

  - Percentage of Events — for categorical targets
  - Number of Events — for categorical targets
  - Cumulative Percentage of Events — for categorical targets
  - Mean for Predicted — for interval targets
  - Max. for Predicted — for interval targets
  - Min. for Predicted — for interval targets

- **Model**
  - **Link Graph** — graphically displays the neural network architecture. All input variables are listed on the far left of the Link Graph. The target variable is placed at the far right of the Link Graph. Hidden units are found between the input variables and the target variable. The width and color of the lines indicate the magnitude of the weight for that particular connection. The thinner blue values indicate a smaller magnitude link weight, and the thicker red values indicate a larger magnitude link weight.

  - **Weights** — displays the value of the weights for each connection. As in the Link Graph, the colors represent the magnitude of the weight.
  - **Training History** — summarizes each try made during retraining. For each try, the report contains the number of completed iterations, the root mean square error for training and validation, the reason for stopping, and which try was identified as the best.
  - **Iteration History Plot** — shows the distribution of the root mean square training and validation error across all iterations for the best try that was identified in the Training History table.
- **Table** — opens the data table that corresponds to the graph that you have in focus.
- **Plot** — opens the Select a Chart plotting wizard so that you can plot the data in the table that you have in focus.

HPDM Assessment

Overview

The High-Performance Data Mining (HPDM) assessment plots display a range of rank order statistics for model assessment of the HPDM models. The HPDM assessment statistics are different from the regular SAS Enterprise Miner assessment statistics when you run HPDM models using data sets in a grid mode. The HPDM assessment statistics are computed on the whole data set, and the regular SAS Enterprise Miner assessment statistics are computed on a sample of the data set. When you are running HPDM models in solo mode, only the regular SAS Enterprise Miner assessment plots are displayed in the result window. More details about HPDM assessment can be found in “The %EM_new_assess Macro” in *SAS Enterprise Miner High-Performance Data Mining Procedures and Macro Reference for SAS 9.3* and the %HPDM_node_assess macro documentation.

HPDM Assessment Plots for Binary Target Variables

In the HPDM assessment for binary target variables, the data is binned by descending value of the nonmissing estimated probability of the event level for the binary target variable. For each bin (group) of observations, several statistics are computed and plotted against the depth of bins in the HPDM assessment plots. By default, the horizontal axis of an HPDM assessment plot displays the depth (binning) of the observations, and the vertical axis of an HPDM assessment plot displays one or several statistics. The available plots are as follows:

- Lift and Cumulative Lift
- Event and Non-Event Rate
- Classification Rates — CR
- Separation Curve — KS
- Cumulative Captured Events
- Receiver-Operator Characteristic

HPDM Assessment Plots for Interval Target Variables

In the HPDM assessment for interval target variables, the data is binned by descending order of the nonmissing predicted values of the interval target variable. For each bin (group) of observations, several statistics are computed and plotted against the depth of bins in the HPDM assessment plots. By default, the horizontal axis of an HPDM assessment plot displays the depth (binning) of the observations. The vertical axis of an HPDM assessment plot displays the actual target means, the predicted target means, and the residual means in each bin.

HPDM Bin Statistics Table

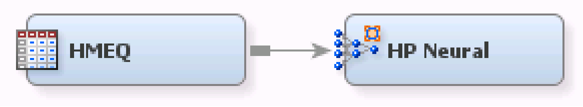
This table displays the summary and fit statistics for the binning of the target variable.

HP Neural Node Example

This example uses the sample SAS data set called Sampsio.Hmeq. You must use the data set to create a SAS Enterprise Miner Data Source. Right-click the **Data Sources** icon in the Project Navigator, and select **Create Data Source** to launch the Data Source wizard.

1. Choose **SAS Table** as your metadata source and click **Next**.
2. Enter `Sampsio.Hmeq` in the **Table** field and click **Next**.
3. Continue to the Metadata Advisor step, and choose the **Basic Metadata Advisor**.
4. In the Column Metadata window, set the role of the variable Bad to **Target** and set the level of the variable Bad to **Binary**. Click **Next**.
5. There is no decision processing. Click **Next**.
6. In the Create Sample window, you are asked if you want to create a sample data set. Select **No**. Click **Next**.
7. Set the role of the HMEQ data set to **Train**, and then click **Finish**.

Drag the HMEQ data set and the **HP Neural** node to your diagram workspace. Connect them as shown in the diagram below.



Select the **HP Neural** node and make the following changes:

1. Set the **Number of Hidden Neurons** to `6`.
2. Select **Two Layers with Direct** for the **Architecture** property.
3. Set the **Number of Tries** to `5`.
4. Set the **Maximum Iterations** property to `10`.

Right-click the **HP Neural** node and select **Run** from the drop-down menu. In the Confirmation window, select **Yes**. After the **HP Neural** node has successfully run, select **Results** from the Run Status window. Notice the following results:

- The Model Information table indicates that the limited memory BFGS algorithm was used, and that there were 12 input variables, 1 output variable, 6 hidden neurons, and 87 weights.
- The Number of Observations table indicates that 2512 observations were used for training and 852 were used for validation.
- The Classification Chart shows a large number of observations were incorrectly assigned a target value of `1`, especially when compared to the number incorrectly assigned to `0`.
- The Link Graph table provides a visual representation of the neural network. Note that the thinner blue values indicate a smaller link weight, and the thicker red values indicate a larger link weight. In this neural network, there are six hidden neurons, three in each of the two hidden layers.
- The Weights graph provides an alternative view of the weights for each connection.
- The Training History table provides the root mean square error for both the training and validation data sets, in addition to the reason each try was stopped. In this example, the second try produced the best neural network.
- The Iteration History Plot graphs the root mean square error for the training and validation data against the iteration number. This plot corresponds to the second try. Notice that this graph ends at the tenth iteration because you specified **10** in the **Maximum Iterations** property, which is also the best iteration.

Chapter 6
The HP Regression Node

Overview of the HP Regression Node

The **HP Regression** node fits a linear regression or a logistic regression for an interval or binary target variable that is specified in the training data set. Linear regression attempts to predict the value of an interval target as a linear function of one or more independent inputs. Logistic regression attempts to predict the probability that a binary target will acquire the event of interest based on a specified link function of one or more independent inputs.

The **HP Regression** node supports binary and interval target variables. For example, when modeling customer profiles, a variable named Purchase that indicates whether a customer made a purchase can be modeled as a binary target variable. If your customer profiles also contain a variable named Value that indicates the amount spent, this variable can be modeled as an interval target variable. The **HP Regression** node does not supports the modeling of more than one target variable.

The **HP Regression** node supports forward, backward and stepwise selection methods for interval targets, and forward, backward, stepwise, LAR, and LASSO selection methods.

HP Regression Node Requirements

If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

If you are running the **HP Regression** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Regression Node Properties

HP Regression Node General Properties

The following general properties are associated with the **HP Regression** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Regression** node that is added to a diagram has a Node ID of HPReg. The second **HP Regression** node that is added to a diagram has a Node ID of HPReg2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Regression window. The Imported Data — HP Regression window contains a list of the ports that provide data sources to the **HP Regression** node. Select the ... button to the right of the Imported Data property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following tasks:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Regression window. The Exported Data — HP Regression window contains a list of the output data ports that the HP Regression node creates data for when it runs. Select the ... button to the right of the Exported Data property to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and do any of the following:

  - **Browse** to open a window where you can browse the data set.

    - **Explore** to open the Explore window, where you can sample and plot the data.
    - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.
- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes, such as data or configuration information.

HP Regression Node Train Properties

The following train properties are associated with the **HP Regression** node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Select the ... button to the right of the Variables property to open a variables table. You can set the variable status to either **Use** or **Don't Use** in the table, and you can set a variable's report status to **Yes** or **No**.

HP Regression Node Train Properties: Equation

- **Main Effects** — Set the Main Effects property to **No** if you want to suppress the input and rejected variables with status of Use in the regression analysis. The default setting for this property is **Yes**.
- **Two-Factor Interactions** — Set the Two-Factor Interactions property to **Yes** if you want to include all two-factor interactions for class variables that have a status of Use. The default setting for this property is **No**.
- **Polynomial Terms** — Set the Polynomial Terms property to **Yes** if you want to include polynomial terms for interval variables with status of Use in the regression analysis. When this property is set to **Yes**, you must specify an integer value for the Polynomial Degree property. The default setting for Polynomial Terms is **No**.
- **Polynomial Degree** — When the Polynomial Terms property of the Regression node is set to **Yes**, use the Polynomial Degree property to specify the highest degree of polynomial terms (for interval variables with status set to **Use**) to be included in the regression analysis. The Polynomial Degree property can be set to 2 or 3.
- **Suppress Intercept** — Set the Suppress Intercept property to **Yes** to suppress intercepts when you are coding class variables. The Suppress Intercept property is ignored for ordinal targets. The default setting for the Suppress Intercept property is **No**.
- **Use Missing as Level** — specifies if missing values should be considered as their own classification level.

Note: If **Main Effects**, **Two-Factor Interactions**, and **Polynomial Terms** are all set to **No**, then SAS Enterprise Miner will report an exception message. When all three properties are set to **No**, there are no effects specified for the regression analysis.

HP Regression Node Train Properties: Modeling

- **Regression Type** — Use the Regression Type property to specify the type of regression that you want to run.
    - **Logistic Regression** — the default regression type for binary targets. For logistic regression, the event level of the binary target is determined by the sorting order

    of this variable in the preceding input data set node. The default sorting order of the binary target is **Descending**. For example, if the binary target has two levels, 1 and 0, then the **HP Regression** node chooses 1 as the event level and 0 as the non-event level.

  - **Linear Regression** — the default regression type for interval targets.

- **Link Function** — Use the Link Function property of the **HP Regression** node to specify the link function that you want to use in your regression analysis. Link functions link the response mean to the linear predictor. In a linear regression, the identity link function g(M) = Xβ is used.

  In a logistic regression, you can select one of the link functions:

  - **Complementary log-log**
  - **Logit** (default)
  - **Log-Log**
  - **Probit**

- **Optimization Options** — specifies the optimization options for the regression model. Select the ... button to the right of the Optimization Options property to open the Optimization Options window.

  The following properties are available in the Optimization Options window:

  - **Optimization Technique** — specifies the optimization technique used by the **HP Regression** node.
    - **Conjugate-Gradient**
    - **Double-Dogleg**
    - **Newton-Raphson**
    - **Nelder-Mead Simplex**
    - **Newton-Raphson with Ridging** — default
    - **Qual Quasi-Newton**
    - **Trust-Region**
  - **Maximum Number of Iterations** — Use the Maximum Number of Iterations property to specify the maximum number of iterations to be used in the optimization technique. To use the default value, leave the value as blank or a dot. The default value for the Maximum Number of Iterations property varies according to the selected optimization technique:

Optimization Technique	Default Max Iterations
Conjugate-Gradient	400
Double-Dogleg	200
Newton-Raphson	50
Nelder-Mead Simplex	1000
Newton-Raphson with Ridging	50
Qual Quasi-Newton	200

Optimization Technique	Default Max Iterations
Trust-Region	50

  - **Maximum Number of Function Evaluations** — Use the Maximum Number of Function Evaluations property of the **HP Regression** node to specify the maximum number of function evaluations to allow in the optimization technique. To use the default value, leave the value as blank or a dot.

Optimization Technique	Default Max Function Evaluations
Conjugate-Gradient	1000
Double-Dogleg	500
Newton-Raphson	125
Nelder-Mead Simplex	3000
Newton-Raphson with Ridging	125
Qual Quasi-Newton	500
Trust-Region	125

  - **Maximum CPU Time in seconds** — specifies an upper limit of CPU time (in seconds) for the optimization process. The default value is the largest floating-point double representation of your computer. The time specified by this property is checked only once at the end of each iteration. Therefore, the actual run time can be longer than that which the property specifies. To use the default value, leave the value as blank or a dot.

    Note: The **Maximum CPU Time in seconds** property governs only the optimization process time for the HPLOGISTIC procedure. It does not govern the maximum overall execution time for the **HP Regression** node.

  - **Minimum Number of Iterations** — specifies the minimum number of iterations. The default value is 1. If you request more iterations than are actually needed for convergence, the optimization algorithms can behave unpredictably. To use the default value, leave the value as blank or a dot.
  - **Normalize Objective Function** — Use the Normalize Objective Function property to determine whether the objective function should be normalized during the optimization. The reciprocal of the used frequency count is used for normalization. The default value is **Yes**.

- **Convergence Options** — specifies the convergence options for the regression model. Select the ... button to the right of the Convergence Options property to open the Convergence Options window.

  The following properties are available in the Convergence Options window:

  - **Absolute Function Convergence** — specifies the threshold for absolute function convergence. The default value is the negative square root of the largest double precision value that is available on your computer. To use the default value, leave the value as blank or a dot.

- **Absolute Function Difference Convergence** — specifies the threshold for absolute function difference convergence. The default value is 0. To use the default value, leave the value as blank or a dot.
- **Absolute Gradient Convergence** — specifies the threshold for absolute gradient convergence. The default value is 1E-5. To use the default value, leave the value as blank or a dot.
- **Relative Function Difference Convergence** — specifies the threshold for relative function difference convergence. The default value is twice the machine precision. To use the default value, leave the value as blank or a dot.
- **Relative Gradient Convergence** — specifies the threshold for relative gradient convergence. The default value is 1E-8. To use the default value, leave the value as blank or a dot.

HP Regression Node Train Properties: Model Selection

This section details the model selection options that are available in the **HP Regression** node. The Model Selection Options Table on page 51 provides a quick reference to determine which options are available for each model selection method.

- **Selection Model** — Use the Selection Model property to specify the model selection method that you want to use during training.

  You can choose from the following effect selection methods:

  - **None** — (default setting) uses all inputs to fit the model. The **Selection Criterion**, **Stop Criterion**, and **Selection Options** properties are not available when **None** is selected.
  - **Backward** — begins with all candidate effects in the model and removes effects until the Stay Significance Level or the Stop Criterion is met.
  - **Forward** — begins with no candidate effects in the model and adds effects until the Entry Significance Level or the Stop Criterion is met.
  - **Stepwise** — begins as in the forward model but might remove effects already in the model. Continues until Stay Significance Level or Stepwise Stopping Criteria are met.
  - **LAR** — performs least angle regression selection. This method, like forward selection, starts with no effects in the model and adds effects. The parameter estimates at any step are shrunk when compared to the corresponding least squares estimates. If the model contains classification variables, then these classification variables are split. The **LAR** selection method is supported only for interval target variables.
  - **Lasso** — adds and deletes parameters based on a version of ordinary least squares, where the sum of the absolute regression coefficients is constrained. If the model contains classification variables, then these classification variables are split. The **Lasso** selection method is supported only for interval target variables.

  Note: The **LAR** and **Lasso** methods are available only for interval target variables. If you specify either **LAR** or **Lasso** for a training set with a binary target variable, then **None** is used instead.

- **Selection Criterion** — Use this property to determine the order in which effects enter or leave at each step of the selection method. If you use either **LAR** or **Lasso** as the selection method, then you can specify only **Default** or **Significance Level** here.

The following section criteria are available:

- **Default** — uses the significance levels of the effects.
- **Adjusted R-square** — available only for linear regressions. If you have a binary target variable and specify this option, then Default is used instead.
- **AIC** — Akaike's Information Criterion
- **AICC** — Corrected Akaike's Information Criterion
- **SBC** — Schwarz Bayesian Information Criterion
- **Mallow's CP$_p$** — Mallow's C$_p$ statistic, available only for linear regressions. If you have a binary target variable and specify this option, then **Default** is used instead.
- **Significance Level** — uses the significance levels of the effects.

Note: If you select **Logistic Regression** as the **Regression Type**, then **Significance Level** is always used as the **Selection Criterion**. Similarly, if you select **LAR** or **Lasso** as the **Selection Model**, then **Significance Level** is always used as the **Selection Criterion**.

- **Stop Criterion** — specifies the criterion that is used to stop the selection process.

  The following section criteria are available:

  - **Default** — uses the significance levels of the effects.
  - **Adjusted R-square** — available only for linear regressions.
  - **AIC** — Akaike's Information Criterion
  - **AICC** — Corrected Akaike's Information Criterion
  - **SBC** — Schwarz Bayesian Information Criterion
  - **Mallow's CP$_p$** — Mallow's C$_p$ statistic.
  - **Significance Level** — uses the significance levels of the effects.
  - **No Criterion for Stopping Selection** — No criterion for stopping selection is used. The selection process stops when no suitable addition or removal of candidate effects is found or if a size-based limit, such as **Maximum Number of Effects**, is reached.

  Note: If you specify a criterion other than **Significance Level** or **None**, then the selection process stops when a local extremum is found or if a size-based limit is reached. The determination of whether a local minimum is achieved is made on the basis of a stop horizon at the next three steps.

- **Selection Options** — specifies the selection options for the regression model. Select the ... button to the right of the Convergence Options property to open the Selection Options window.

  The following selection options are available:

  - **Entry Significance Level** — significance level for adding variables in forward, stepwise, LAR, or Lasso regression. The default value for the Entry Significance Level is 0.05. Values must be between 0 and 1.
  - **Stay Significance Level** — significance level for removing variables in backward, stepwise, or LASSO regression. The default value for the Entry Significance Level is 0.05. Values must be between 0 and 1.

- **Max Number of Effects** — the maximum number of effects in any model that is considered during the selection process. This option is ignored with the backward regression. If a model at some step of the selection process contains the specified maximum number of effects, then no candidate effects are considered for addition. The default value is zero, which indicates that this option should be ignored.
- **Min Number of Effects** — the minimum number of effects in any model that is considered during the backward or stepwise selection process. For backward regression, the selection process terminates if a model at some step of the selection process contains the specified minimum number of effects. The default value is zero, which indicates that this option should be ignored.
- **Hierarchy** — specifies whether no variable, only class variables, or both class and interval variables are subject to hierarchy rules. You can specify **None** for no hierarchy rules, **Class Variables** for just class variables, and **All Variables** for class and interval variables.
- **Max Number of Steps** — Maximum number of selection steps that are performed. The default value is zero, which indicates that this option is ignored.

HP Regression Node Score Properties

The following score properties are associated with the **HP Regression** node:

- **Excluded Variables** — specifies how the **HP Regression** node handles variables that are excluded from the final model. This option is active only when you specify a variable selection method. When set to **None**, the role of these variables remains unchanged. When set to **Hide**, these variables are dropped from the metadata exported by the node. When set to **Reject**, the role of these variables is set to Rejected.

HP Regression Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Regression Node Model Selection Options Tables

The two tables below provide a quick reference to determine which options are available for each model selection method. The available model selection methods are given in the first row and the model selection options are given in the first column. A Y indicates that the corresponding model selection option is available for that method, and an N indicates that it is not available.

Table 6.1 Linear Regression Model Selection Options

	Forward	Backward	Stepwise	LAR	Lasso
Selection Criterion	Y	Y	Y	Significance Level	Significance Level
Stop Criterion	Y	Y	Y	Y	Y
Entry Significance Level	Y	N	Y	Y	Y
Stay Significance Level	N	Y	Y	N	Y
Maximum Number of Effects	Y	N	Y	Y	Y
Minimum Number of Effects	N	Y	N	N	N
Hierarchy	Y	Y	Y	N	N
Maximum Number of Steps	Y	Y	Y	Y	Y

Table 6.2 Logistic Regression Model Selection Options

	Forward	Backward	Stepwise
Selection Criterion	Significance Level	Significance Level	Significance Level
Stop Criterion	Y	Y	Y
Entry Significance Level	Y	N	Y

	Forward	Backward	Stepwise
Stay Significance Level	N	Y	Y
Maximum Number of Effects	Y	N	Y
Minimum Number of Effects	N	Y	N
Hierarchy	Y	Y	Y
Maximum Number of Steps	Y	Y	Y

HP Regression Node Results

After a successful node run, you can open the Results window of the **HP Regression** node by right-clicking the node and selecting **Results** from the pop-up menu.

Select **View** from the main menu in the Results window to view the following:

- **Properties**
  - **Settings** — displays a window with a read-only table of the **HP Regression** node properties configuration when the node was last run.
  - **Run Status** — displays the status of the **HP Regression** node run. Information about the run start time, run duration, and completion status are displayed in this window.
  - **Variables** — displays a table of the variables in the training data set.
  - **Train Code** — displays the code that SAS Enterprise Miner used to train the node.
  - **Notes** — displays notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Regression** node run.
  - **Output** — the SAS output of the **HP Regression** node run.
  - **Flow Code** — the SAS code used to produce the output that the **HP Regression** node passes on to the next node in the process flow diagram.
- **Scoring**
  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.
  - **PMML Code** — The **HP Regression** node does not generate PMML code.
- **Assessment** — The selections listed here are available when the **HP Regression** node is run in solo mode or on the grid. Additional assessment plots are available

when the **HP Regression** node runs on the grid. See HPDM Assessment on page 54 for more information about these assessment plots.

- **Fit Statistics** — a table of the fit statistics from the model.
- **Classification Chart** — displays a stacked bar chart of the classification results for a categorical target variable. The horizontal axis displays the target levels that observations actually belong to. The color of the stacked bars identifies the target levels that observations are classified into. The height of the stacked bars represents the percentage of total observations. This selection is available only for logistic regressions.
- **Score Rankings Overlay** — In a score rankings chart, several statistics for each decile (group) of observations are plotted on the vertical axis. For a binary target, all observations in the scored data set are sorted by the posterior probabilities of the event level in descending order. Then the sorted observations are grouped into deciles. Observations in a decile are used to calculate the statistics that are plotted in deciles charts. The Score Rankings Overlay plot displays both train and validation statistics on the same axis.

  By default, the horizontal axis of a score rankings chart displays the deciles (groups) of the observations.

  The vertical axis displays the following values:

  - Cumulative Lift
  - Lift
  - Gain
  - % Response
  - Cumulative % Response
  - % Captured Response
  - Mean for Predicted — for interval targets
  - Maximum for Predicted — for interval targets
  - Minimum for Predicted — for interval targets
- **Score Distribution** — The Score Distribution chart plots the proportions of events (by default), nonevents, and other values on the vertical axis. The values on the horizontal axis represent the model score of a bin. The model score depends on the prediction of the target and the number of buckets used. For categorical targets, observations are grouped into bins, based on the posterior probabilities of the event level and the number of buckets. The Score Distribution chart of a useful model shows a higher percentage of events for higher model scores and a higher percentage of nonevents for lower model scores. For interval targets, observations are grouped into bins, based on the actual predicted values of the target. The default chart choice is Percentage of Events. Multiple chart choices are available for the Score Distribution Chart. This selection is available only for logistic regressions. The chart choices are as follows:

  - Percentage of Events — for categorical targets.
  - Number of Events — for categorical targets.
  - Cumulative Percentage of Events — for categorical targets.
  - Expected Profit — for categorical targets.
  - Report Variables — for categorical targets.

        - Mean for Predicted — for interval targets.
        - Max. for Predicted — for interval targets.
        - Min. for Predicted. — for interval targets.
    - **HPDM Assessment** — Additional assessment plots are available when the **HP Regression** node runs on the grid. See HPDM Assessment on page 54 for more information about these assessment plots.
    - **Residual Statistics** — displays a box plot for the residual variable VALUE measurements when the target is interval. This selection is available only for linear regressions.
- **Model** — graphs and tables with information about the variables in the model. The available graphs and tables are as follows:
    - **Parameter Estimates** — displays bar charts for the coefficients in the final model. The bars are color-coded to indicate the algebraic signs of the coefficients. The chart choices are as follows:
        - T-values
        - Estimates
        - Standard Errors
        - P-Values
    - **Odds Ratio Plot** — displays the odds ratio, $e^{\hat{\beta}}$, for changing either one unit of an interval input variable or between the specified level and the reference level of a categorical input variable. Formally, let y be the binary outcome variable where 0 indicates failure and 1 indicates success. Let p be the probability that y is a success. Let x_1, x_2, ..., x_k be a set of predictor variables. The logistic regression of y on x_1, x_2, ..., x_k (without the interaction term) estimates the parameter values for β_0, β_1, ..., β_k. This estimate is made via the maximum likelihood method, given as logit(p) = log(p / (1 — p)) = $\beta_0 + \beta_1 * x_1 + ... + \beta_k * x_k$. This plot is displayed as $e^{\hat{\beta}}$ for each variable.
- **Table** — displays a table that contains the underlying data that is used to produce a chart. The **Table** menu item is dimmed and unavailable unless a results chart is open and selected.
- **Plot** — use the Graph wizard to modify an existing Results plot or create a Results plot of your own. The **Plot** menu item is dimmed and unavailable unless a Results chart or table is open and selected.

HPDM Assessment

Overview

The High-Performance Data Mining (HPDM) assessment plots display a range of rank order statistics for model assessment of the HPDM models. The HPDM assessment statistics are different from the regular SAS Enterprise Miner assessment statistics when you run HPDM models using data sets in a grid mode. The HPDM assessment statistics are computed on the whole data set, and the regular SAS Enterprise Miner assessment statistics are computed on a sample of the data set. When running HPDM models in solo mode, only the regular SAS Enterprise Miner assessment plots are displayed in the result window. More details about HPDM assessment can be found in “The %EM_new_assess Macro” in *SAS Enterprise Miner High-Performance Data Mining Procedures and Macro Reference for SAS 9.3* and the %HPDM_node_assess macro documentation.

HPDM Assessment Plots for Binary Target Variables

In the HPDM assessment for binary target variables, the data is binned by descending value of the nonmissing estimated probability of the event level for the binary target variable. For each bin (group) of observations, several statistics are computed and plotted against the depth of bins in the HPDM assessment plots. By default, the horizontal axis of an HPDM assessment plot displays the depth (binning) of the observations, and the vertical axis of an HPDM assessment plot displays one or several statistics. The available plots are as follows:

- Lift and Cumulative Lift
- Event and Non-Event Rate
- Classification Rates — CR
- Separation Curve — KS
- Cumulative Captured Events
- Receiver-Operator Characteristic

HPDM Assessment Plots for Interval Target Variables

In the HPDM assessment for interval target variables, the data is binned by descending order of the nonmissing predicted values of the interval target variable. For each bin (group) of observations, several statistics are computed and plotted against the depth of bins in the HPDM assessment plots. By default, the horizontal axis of an HPDM assessment plot displays the depth (binning) of the observations. The vertical axis of an HPDM assessment plot displays the actual target means, the predicted target means, and the residual means in each bin.

HPDM Bin Statistics Table

This table displays the summary and fit statistics for the binning of the target variable.

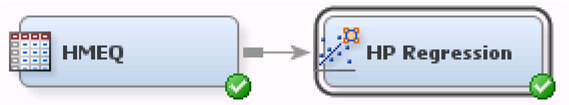
HP Regression Node Example

This example uses the sample SAS data set called Sampsio.Hmeq. You must use the data set to create a SAS Enterprise Miner Data Source. Right-click the **Data Sources** folder in the Project Navigator and select **Create Data Source** to launch the Data Source wizard.

- Select **SAS Table** as your metadata source and click **Next**.
- Enter `Sampsio.Hmeq` in the **Table** field and click **Next**.
- Continue to the Metadata Advisor step and select the **Basic Metadata Advisor**.
- In the Column Metadata window, set the role of the variable Bad to **Target** and set the level of the variable Bad to **Binary**. Click **Next**.
- There is no decision processing. Click **Next**.

- In the Create Sample window, you are asked if you want to create a sample data set. Select **No**. Click **Next**.
- Set the role of the HMEQ data set to **Train**, and then click **Finish**.

Drag the HMEQ data set and the HP Regression node to your diagram workspace. Connect them as shown in the diagram below.



Run the **HP Regression** node with the default settings by right-clicking on the **HP Regression** node and selecting **Run**. In the Confirmation window, select **Yes**. After a successful run of the **HP Regression** node, select **Results** in the Run Status window.

Notice the following information:

- Model/Performance Information — The Model/Performance Information window contains information about the input data set, the regression model, and the high-performance computing environment. In this example, a logistic regression and a Logit link function were used to model the target variable BAD. The **Procedure Task Timing** group provides the timing information for various computational stages of the **HP Regression** node.
- Odds Ratio Plot — In the Logit model, the logarithm of the odds of the outcome is modeled as a linear combination of the input variables. Displays the odds ratio, $e^{\hat{\beta}}$, for changing either one unit of an interval input variable or between the specified level and the reference level of a categorical input variable.
- Parameter Estimation — The Parameter Estimation plot displays the parameter estimates and associated p-value, t-value, and standard error for each input variable that is in the model. The drop-down menu in the upper left corner of the window enables you to select different parameters. Select **Estimates** from this menu. In this plot, both Delinq and Derog have strong positive coefficients. Also, the class level Sales for the Job variable has a strong positive coefficient. All other levels of Job have negative coefficients.
- Classification Chart — The Classification Chart displays the percentage of correct and incorrect classifications for the different levels of the class target variable. This example displays a large percentage of incorrect classification for target level 1 compared to target level 0.

Chapter 7

The HP Text Miner Node

Overview of the HP Text Miner Node

The HP Text Miner node enables you to build predictive models for a document collection in a distributed computing environment. Data is processed in two phases: text parsing and transformation. Text parsing processes textual data into a term-by-document frequency matrix. Transformations such as singular value decomposition (SVD) alter this matrix into a data set that is suitable for data mining purposes. A document collection of millions of documents and hundreds of thousands of terms can be represented in a compact and efficient form.

You can connect the HP Text Miner node with other HP nodes to perform modeling.

English is the only parsing language available for the HP Text Miner node.

Note: A valid Text Miner for SAS Enterprise Miner license is required to use the HP Text Miner node.

HP Text Miner Node Requirements

The HP Text Miner node requires an input data set that contains a variable with the role **Text** or **Textloc**. This variable cannot be an interval variable. The input data set must also contain a variable with the role Key. The key variable contains a unique identifier for each observation in the input data set.

HP Text Miner Node Properties

HP Text Miner Node General Properties

The following general properties are associated with the HP Text Miner node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first HP Text Miner node that is added to a diagram has a Node ID of HPTM. The second HP Text Miner node added to a diagram has a Node ID of HPTM2, and so on.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Text Miner window. The Imported Data — HP Text Miner window contains a list of the ports that provide data sources to the HP Text Miner node. Select the ... button to the right of the Imported Data property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and click for the desired option:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.
- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Text Miner window. The Exported Data — HP Text Miner window contains a list of the output data ports that the HP Text Miner node creates data for when it runs. Select the ... button to the right of the Exported Data property to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and choose an action:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contain summary information (metadata) about the table and the variables.

- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Text Miner Node Train Properties

The following train properties are associated with the HP Text Miner node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Select the ... button to the right of the Variables property to open a variables table. You can set the variable status to either **Default**, **Yes**, or **No** in the table, and you can set a variable's report status to **Yes** or **No**.

HP Text Miner Node Train Properties: Detect

- **Different Parts of Speech** — specifies whether to identify the parts of speech of parsed terms. If the value of this property is **Yes**, then same terms with different parts of speech are treated as different terms. For more information, see Parts of Speech on page 61 .

- **Find Entities** — specifies whether to identify the entities contained in the documents. Entity detection relies on linguistic rules and lists that are provided for many entity types; these are known as standard entities. Specify **Yes** to detect entities. For more information, see Entities on page 62 .
- **Multi-word Terms** — specifies a SAS data set that contains multi-word terms. For more information, see Multi-Term Lists on page 63 .

  Click the ellipsis button to open a window in which you can do the following:

  - import a multi-word term data set
  - (if a multi-word term data set is selected) add, delete, and edit terms in the multi-term list
- **Synonyms** — specifies a SAS data set that contains synonyms to be treated as equivalent. For more information, see Synonym Lists on page 64 .

  Click the ellipsis button to open a window in which you can do the following:

  - import a synonym data set
  - (if a synonym data set is selected) add, delete, and edit terms in the synonym list

HP Text Miner Node Train Properties: Filter

- **Stop List** — specifies a SAS data set that contains terms to exclude from parsing. If you include a stop list, then the terms that are included in the stop list appear in the results Term table with a Keep status of `N`. For more information, see Stop Lists on page 66 .

Click the ellipsis button to open a window in which you can do the following:

- import a stop list data set
- (if a stop list is selected) add, delete, and edit terms in the stop list

- **Minimum Number of Documents** — specifies the minimum number of documents that a term must appear in to be considered for analysis. In other words, terms that appear in fewer documents than the number specified here will be excluded from analysis.

HP Text Miner Node Train Properties: Transform

- **SVD Resolution** — specifies the resolution to use to generate the SVD dimensions.
- **Max SVD Dimensions** — specifies the maximum number of SVD dimensions to generate. The minimum value that you can specify is 2, and the maximum value that you can specify is 500.

A high number of SVD dimensions usually summarize the data better, but the higher the number, the more computing resources are required. The **HP Text Miner** node determines the number of SVD dimensions based on the values of the **SVD Resolution** and **Max SVD Dimensions** properties. The value of the **SVD Resolution** property can be set to **Low** (default), **Medium**, or **High**. High resolution yields more SVD dimensions. The default value of the **Max SVD Dimensions** property is 100, and the value must be between 2 and 500.

Suppose that the maximum number of SVD dimensions that you specify for the **Max SVD Dimensions** property is maxdim, and these maxdim SVD dimensions account for p % of the total variance. High resolution always generates the maximum number of SVD dimensions (maxdim). For medium resolution, the recommended number of SVD dimensions accounts for 5/6*(p% of the total variance). For low resolution, the recommended number of SVD dimensions accounts for 2/3*(p% of the total variance).

For more information about singular value decompositions, see Singular Value Decomposition in the SAS Text Miner Help.

HP Text Miner Node Report Properties

The following report property is associated with the HP Text Miner node:

- **Number of Terms to Display** — indicates the maximum number of terms to be displayed in the Results viewer. Terms are first sorted by the number of documents in which they appear, and then the list is truncated to the maximum number.

HP Text Miner Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.

- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

Term Roles and Attributes for the HP Text Miner Node

Parts of Speech

The HP Text Miner node can identify the part of speech for each term in a document based on the context of that term. Terms are identified as one of the following parts of speech:

- Abbr — abbreviation
- Adj — adjective
- Adv — adverb
- Aux — auxiliary or modal
- Conj — conjunction
- Det — determiner
- Interj — interjection
- Noun — noun
- Num — number or numeric expression
- Part — infinitive marker, negative participle, or possessive marker
- Prep — preposition
- Pron — pronoun
- Prop — proper noun
- Verb — verb
- VerbAdj — verb adjective

Noun Groups

The HP Text Miner node automatically identifies noun groups, like “clinical trial” and “data set," in a document collection. Noun groups are identified based on linguistic relationships that exist within sentences. Syntactically, these noun groups act as single units and are parsed as single terms.

The HP Text Miner node automatically performs noun group stemming. For example, the text “number of defects” is parsed as “number of defect.” Frequently, shorter noun groups are contained within larger noun groups. Both the shorter and larger noun groups appear in parsing results.

Entities

An *entity* is any of several types of information that HP Text Miner node can distinguish from general text. If you enable the HP Text Miner node to identify them, entities are analyzed as a unit, and they are sometimes normalized. When the HP Text Miner node extracts entities that consist of two or more words, the individual words of the entity are also used in the analysis.

The HP Text Miner node identifies the following standard entities:

- ADDRESS — postal address or number and street name
- COMPANY — company name
- CURRENCY — currency or currency expression
- DATE — date, day, month, or year
- INTERNET — e-mail address or URL
- LOCATION — city, country, state, geographical place or region, or political place or region
- MEASURE — measurement or measurement expression
- ORGANIZATION — government, legal, or service agency
- PERCENT — percentage or percentage expression
- PERSON — person's name
- PHONE — phone number
- PROP_MISC — proper noun with an ambiguous classification
- SSN — Social Security number
- TIME — time or time expression
- TIME_PERIOD — measure of time expressions
- TITLE — person's title or position
- VEHICLE — motor vehicle including color, year, make, and model

Attributes

When a document collection is parsed, the HP Text Miner node categorizes each term as one of the following attributes. The attributes give an indication of the characters that compose that term:

- Abbr — if the term is an abbreviation
- Alpha — if characters are all letters
- Entity — if the term is an entity
- Mixed — if term characters include a mix of letters, punctuation, and white space
- Num — if term characters include a number
- Punct — if the term is a punctuation character

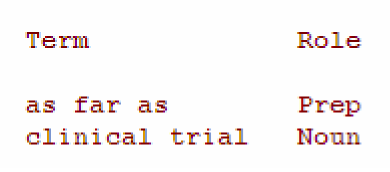
Note: Any term with the attribute Num or Punct is automatically dropped from the output terms table.

Multi-Term Lists

Multi-term lists enable you to specify groups of words that should be processed together as single terms. Multi-term data sets have a required format. You must include the variables “Term,” which contains a multi-word term, and “Role,” which contains an associated role.

Note: A role is either a part of speech, an entity classification, or the value **Noun Group**. For more information about roles, see Term Roles and Attributes for the HP Text Miner Node on page 61 .

For example, if you use the following multi-term list, then any instance of the phrase “as far as” is processed as one term, a preposition, and any instance of the phrase “clinical trial” is processed as one term, a noun:



```
Term              Role

as far as         Prep
clinical trial    Noun
```

You can similarly define multi-word terms using a synonym list. In this case, the groups of words that you specify will be processed together as single terms. For more information, see Defining Multi-Word Terms Using a Synonym List on page 65 .

Term Stemming

Stemming is the process of finding the stem or root form of a term. The HP Text Miner node uses dictionary-based stemming, which unlike tail-chopping stemmers, produces only valid words as stems. When part-of-speech tagging is on, the stem selection process restricts the stem to be of the same part-of-speech as the original term.

Table 7.1 *Examples of Stemming*

Stem	Terms
aller (French)	vais, vas, va, allons, allez, vont
reach	reaches, reached, reaching
big	bigger, biggest
balloon	balloons
go	goes

Stemming can be very important for text mining because text mining is based on the co-occurrence relationships of terms throughout the collection. By treating the variations of a term as the term itself, document relationships can be clarified. For example, if “grinds,” “grinding,” and “ground” each occur independently in three separate documents, the individual terms do not contribute to the similarity of these three documents. However, if the terms are all stemmed to “grind” and the documents are treated as if they contain “grind” rather than the original variants, the documents will be related by this common stem.

Because the HP Text Miner node uses the same equivalent term concept to manage stems as it does to manage synonyms, you can customize the stem by editing the synonym list.

Synonym Lists

Overview

A synonym list enables you to specify different words that should be processed equivalently, as the same representative parent term. A default synonym list is provided for the English language. Synonym data sets have a required format.

You must include the following variables:

- TERM — contains a term to treat as a synonym of the PARENT.
- PARENT — contains the representative term to which the TERM should be assigned.
- TERMROLE — enables you to specify that the synonym is assigned only when the TERM occurs with a specific role.
- PARENTROLE — enables you to specify the role of the PARENT.

Note: If a synonym list includes multiple entries that assign the same terms to different parents, then the parsing results will reflect only the first entry.

TERMROLE enables you to specify that the same word, when it has different roles, can be processed either as different synonyms or as itself. For example, you can specify that the term “SAS,” when tagged as a noun, is parsed as “SAS Institute”, but, when tagged as a verb, is processed as “sass.” In order for the variable TERMROLE to be used when the **HP Text Miner** node processes a synonym list, the **Different Parts of Speech** property must be set to **Yes**, and the **Find Entities** property should not be set to **No**, if these roles are used in the synonym list.

For example, use of the following synonym list causes any instance of “SAS,” when identified as a company, to be processed as “SAS Institute.” Also, any instance of “employees” is processed as “employees.” In fact, if part-of-speech tagging is on, then this entry disables stemming for only the term “employees.”

TERM	PARENT	TERMROLE	PARENTROLE
SAS	SAS Institute	Company	
employees	employees		

Synonyms and Part-of-Speech Tagging

The following examples demonstrate how synonym lists are handled when part-of-speech tagging is on.

TERM	PARENT	TERMROLE	PARENTROLE
well	water		

In this example, TERM and PARENT (but not TERMROLE or PARENTROLE) are defined. When part-of-speech tagging is either on or off, every occurrence of “well,” regardless of part of speech, is assigned to the parent “water.”

TERM	PARENT	TERMROLE	PARENTROLE
well		noun	
data mining		noun	

In this example, TERM and TERMROLE (but not PARENT or PARENTROLE) are defined. When part-of-speech tagging is either on or off, the entry for the single word term, “well,” has no effect on the parsing results. However, when part-of-speech tagging is on, the multi-word term, “data mining” is treated as a single term only when identified as a noun. When part-of-speech tagging is off, any instance of “data mining” is treated as a single term.

TERM	PARENT	TERMROLE	PARENTROLE
well	water	noun	

In this example, TERM, PARENT, and TERMROLE (but not PARENTROLE) are defined. When part-of-speech tagging is on, “well” is assigned to the parent “water” only when it is identified as a noun. When part-of-speech tagging is off, all instances of “well” are assigned to the parent “water.”

Defining Multi-Word Terms Using a Synonym List

You can use a synonym to specify groups of words that should be processed together as single terms. To define a multi-word term, include it as a term in a synonym list; do not assign it to a parent. For more information, see Multi-Term Lists on page 63 .

Unlike other entries in a synonym list, multi-word terms are case-sensitive.

Appearances of the multi-word term that have the following casings are identified and treated as a single term:

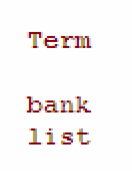
- same casing as the multi-word term entry in the synonym list
- all uppercase version of the multi-word term
- all lowercase version of the multi-word term
- a version that capitalizes the first letter of each term in the multi-word term and lowercases the remaining characters

Stop Lists

Stop lists enable you to control which terms are not used in a text mining analysis. A “stop list” is a data set that contains a list of terms to exclude from the parsing results. Stop lists are often used to exclude terms that contain little information or that are extraneous to your text mining tasks.

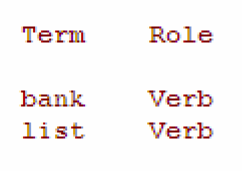
Stop lists must include the variable Term, which contains the terms to exclude from analysis. Also, you can include the variable Role, which contains an associated role. If you include Role and you have set the **Different Parts of Speech** property to `Yes`, then terms are excluded or included based on the `(Term, Role)` pair.

For example, if you use the following stop list, then any instance of the terms “bank” and “list” are excluded from parsing results, regardless of their roles:



Term
bank
list

However, if you use the following stop list and the **Different Parts of Speech** property has the value `Yes`, then the terms “bank” and “list” are excluded from parsing results only if they are used as verbs:



Term	Role
bank	Verb
list	Verb

HP Text Miner Node Results

After a successful node run, you can open the Results window of the HP Text Miner node by right-clicking the node and selecting **Results** from the pop-up menu.

Select **View** from the main menu in the Results window to view the following:

- **Properties**
  - **Settings** — displays a window with a read-only table of the HP Text Miner node properties configuration when the node was last run.
  - **Run Status** — displays the status of the HP Text Miner node run. Information about the run start time, run duration, and completion status are displayed in this window.
  - **Variables** — displays a table of the variables in the training data set.
  - **Train Code** — displays the code that SAS Enterprise Miner used to train the node.

  - **Notes** — displays notes of interest, such as data or configuration information.

- **SAS Results**

  - **Log** — the SAS log of the HP Text Miner run.

  - **Output** — the SAS output of the HP Text Miner run.

  - **Flow Code** — the SAS code used to produce the output that the HP Text Miner node passes on to the next node in the process flow diagram.

- **Scoring**

  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.

  - **PMML Code** — the HP Text Miner node does not generate PMML code.

- **Terms** — graphs and tables with information about the variables in the model. The available graphs and tables are as follows:

  - **Terms** — a table that lists every term in the document collection, up to the number specified in the **Number of Terms to Display** property. This table includes the term, term role, attribute, frequency of occurrence, number of documents that contain the term, keep status, and term weight.

  - **Terms: Freq by Weight** — a scatter plot that displays the frequency of occurrence of each term in the entire document collection versus its term weight. Each data point represents a parsed term. If you position the mouse pointer over a plotted point, then a tooltip indicates the term name, the number of documents in which that term appears, and the weight of that term.

  - **Terms: # of Docs by Freq: Scatter Plot** — a scatter plot that displays the number of documents in which a term appears versus the frequency of occurrence of that term in the entire document collection. Each data point represents a parsed term. If you position the mouse pointer over a plotted point, then a tooltip indicates the term name, the number of documents in which that term appears, and the number of times that term appears in the entire document collection.

  - **Terms: # of Doc by Freq: Histogram** — a histogram that displays the number of documents in which a term appears versus the frequency of occurrence of that term in the entire document collection.

  - **Terms: Role by Freq** — a bar chart that displays the total frequency of occurrence of parsed terms in the document collection, broken down by term role. Each bar represents a role. If you position the mouse pointer over a bar, then a tooltip indicates the role name and the number of times a parsed term with that role appears in the entire document collection.

  - **Terms: Attribute by Freq** — a bar chart that displays the total frequency of occurrence of terms in the document collection, analyzed by attribute. If you position the mouse pointer over a bar, then a tooltip indicates the attribute name and the number of times a term with that attribute appears in the entire document collection.

- **Table** — displays a table that contains the underlying data that is used to produce a chart. The **Table** menu item is dimmed and unavailable unless a results chart is open and selected.

- **Plot** — use the Graph Wizard to modify an existing Results plot or create a Results plot of your own. The **Plot** menu item is dimmed and unavailable unless a Results chart or table is open and selected.

Chapter 8

The HP Transform Node

Overview of the HP Transform Node

The **HP Transform** node enables you to make transformations to your interval input variables. Interval input transformations are important for improving the fit of your model. Transformation of variables can be used to stabilize variances, remove nonlinearity, and correct non-normality in variables.

HP Transform Node Requirements

If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

If you are running the **HP Transform** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Transform Node Properties

HP Transform Node General Properties

The following general properties are associated with the **HP Transform** node:

- **Node ID** — The Node ID property displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Transform** node that is added to a diagram has a Node ID of HPTrans. The second **HP Transform** node that is added to the diagram has a Node ID of HPTrans2.
- **Imported Data** — The Imported Data property provides access to the Imported Data — HP Transform window. The Imported Data — HP Transform window contains a list of the ports that provide data sources to the **HP Transform** node. Select the ... button to the right of the Imported Data property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and click as follows:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Exported Data** — The Exported Data property provides access to the Exported Data — HP Transform window. The Exported Data — HP Transform window contains a list of the output data ports that the **HP Transform** node creates data for when it runs. Select the ... button to the right of the Exported Data property to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and click as follows:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Transform Node Train Properties

The following train properties are associated with the **HP Transform** node:

- **Variables** — Use the Variables property to specify the properties of each variable that you want to use in the data source. Select the ... button to open a variables table. You set the transformation method in the Variables table.

- **Interval Inputs** — Use the Interval Inputs property to specify the default transformation method that you want to apply to interval input variables. Each variable in the Variables table that uses the Default method uses the method that you specify in this property. The available methods are as follows:
  - **Log** — transformed using the logarithm of the variable.
  - **Log 10** — transformed using the base-10 logarithm of the variable.
  - **Square Root** — transformed using the square root of the variable.
  - **Inverse** — transformed using the inverse of the variable.
  - **Square** — transformed using the square of the variable.
  - **Exponential** — transformed using the exponential of the variable.
  - **Centering** — centers variable values by subtracting the mean from each variable.
  - **Standardize** — standardizes the variable by subtracting the mean and dividing by the standard deviation.
  - **Range** — transformed with a scaled value of a variable equal to (x - min) / (max - min), where x is current variable value, min is the minimum value for that variable, and max is the maximum value for that variable.
  - **Bucket** — Buckets are created by dividing the data into evenly spaced intervals, based on the difference between the maximum and minimum values.
  - **Pseudo-Quantile** — groups the input variables into pseudo-quantile bins.
  - **None** — (default setting) No transformation is performed.
- **Interval Targets** — Use the Interval Targets property to specify the default transformation method that you want to use for interval target variables. Each interval target variable in the Variables table that uses the Default method uses the method that you specify in this property. The available methods are as follows:
  - **Log** — transformed using the logarithm of the variable.
  - **Log 10** — transformed using the base-10 logarithm of the variable.
  - **Square Root** — transformed using the square root of the variable.
  - **Inverse** — transformed using the inverse of the variable.
  - **Square** — transformed using the square of the variable.
  - **Exponential** — transformed using the exponential of the variable.
  - **Centering** — centers variable values by subtracting the mean from each variable.
  - **Standardize** — standardizes the variable by subtracting the mean and dividing by the standard deviation.
  - **Range** — transformed with a scaled value of a variable equal to (x - min) / (max - min), where x is current variable value, min is the minimum value for that variable, and max is the maximum value for that variable.
  - **Bucket** — Buckets are created by dividing the data into evenly spaced intervals, based on the difference between the maximum and minimum values.
  - **Pseudo-Quantile** — groups the target variables into pseudo-quantile bins.
  - **None** — (default setting) No transformation is performed.

- **SAS Code** — Select the ... button to open the SAS Code window. You enter SAS code statements to create your own custom variable transformation in the SAS Code window. If you want to use code from a SAS catalog or external file, use the SAS Code window to submit a filename statement and a %include statement.

HP Transform Node Train Properties: Binning

- **Number of Bins** — specifies the number of bins to use when performing bucket or quantile transformations. When **Variables** is selected, the **Number of Bins** property specified in the Variables Editor is used.
- **Missing Values** — Use the Missing property to specify how to handle missing values when you use an optimal binning transformation. Select from any of the available missing value policies.
  - **Separate** — assigns missing values to its own separate branch.
  - **Ignore** — assigns all missing values to a missing value.
  - **First** — assigns the observations that contain missing values to the first bin.

HP Transform Node Score Properties

The following score properties are associated with the **HP Transform** node:

- **Hide** — Use the Hide property to specify how to handle the original variables after a transformation is performed. Setting the Hide property to **No** if you want to keep the original variables in the exported metadata from your transformed data set. The default setting for the Hide property is **Yes**. When this property is set to **Yes**, the original variables are removed only from the exported metadata, and not removed from the exported data sets and data views.
- **Reject** — Use the Reject property to specify whether the model role of the original variables should be changed to Rejected or not. The default value for this property is **Yes**. To change the Reject value from **Yes** to **No**, you must set the Hide property value to **No**.

HP Transform Node Report Properties

The following report property is associated with the **HP Transform** node:

- **Summary Statistics** — Use the Summary Variables property to specify which variables have summary statistics computed. The default setting of **Yes** generates summary statistics on the transformed and new variables in the data set. The **No** setting does not generate any summary statistics.

HP Transform Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.

- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Transform Node Results

After the **HP Transform** node successfully runs, you can open the Results — Transform window by right-clicking the node in the Diagram Workspace and selecting **Results** from the pop-up menu. The Results — Transform window contains an imputation summary table and a window displaying the node's output.

Select **View** from the main menu of the Transform Results window to view the following information:

- **Properties**
  - **Settings** — displays a window with a read-only table of the **HP Transform** node properties configuration when the node was last run. Use the **Show Advanced Properties** check box at the bottom of the window to see all of the available properties.
  - **Run Status** — indicates the status of the **HP Transform** node run. The Run Start Time, Run Duration, and information about whether the run completed successfully are displayed in this window.
  - **Variables** — a table of variables property of the node.
  - **Train Code** — the code that SAS Enterprise Miner used to train the node.
  - **Notes** — enables users to read notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Transform** node run.
  - **Output** — the SAS output of the **HP Transform** node run. The SAS output displays how many and which types of variables are in the training data set. If you run in a grid environment, the output displays limited information about the connection to the grid.
  - **Flow Code** — the SAS code used to produce the output that the **HP Transform** node passes on to the next node in the process flow diagram.
- **Scoring**
  - **SAS Code** — the SAS score code that was created by the node. The SAS score code can be used outside of the SAS Enterprise Miner environment in custom user applications.
  - **PMML Code** — The **HP Transform** node does not generate PMML code.
- **Summary Statistics**
  - **Statistics Table** — The Statistics Table provides a summary of the generated statistics for each variable. The information available in this table is as follows:

  - **Missing** — the number of missing observations
  - **Non Missing** — the number of nonmissing observations
  - **Minimum** — the minimum value of the input variable
  - **Mean** — the mean value of the input variable
  - **Maximum** — the maximum value of the input variable
  - **Standard Deviation** — the standard deviation of the input variable
  - **Skewness** — the measure of skewness of the input variable
  - **Kurtosis** — the kurtosis, also called steepness, of the input variable
- **Table** — opens the data table that corresponds to the graph that you have in focus.
- **Plot** — opens the Select a Chart plotting wizard so that you can plot the data in the table that you have in focus.

Chapter 9
The HP Variable Selection Node

Overview of the HP Variable Selection Node

Many data mining databases have hundreds of potential model inputs (independent or explanatory variables) that can be used to predict the target (dependent or response variable). The **HP Variable Selection** node reduces the number of inputs by identifying the input variables that are not related to the target variable, and rejecting them. Although rejected variables are passed to subsequent nodes in the process flow, these variables are not used as model inputs by a successor modeling nodes.

The **HP Variable Selection** node quickly identifies input variables that are useful for predicting the target variables. The use status **Input** is assigned to these variables. You can override the automatic selection process by assigning the status **Input** to a rejected variable or the status **Rejected** to an input variable. The information-rich inputs are then evaluated in more detail by one of the modeling nodes.

The **HP Variable Selection** node provides both unsupervised and supervised variable selection. The input interval variables that have more missing data than desired or input class variables that have more levels than desired are excluded in the pre-selection stage. The unsupervised model performs variable selection by identifying a set of variables that jointly explain the maximal amount of data variance. Supervised variable selection includes LASSO, LARS, and stepwise regression analysis. The node can be run prior to any other analysis and the results passed to any SAS Enterprise Miner node or any procedure in the SAS System.

HP Variable Selection Node Requirements

One or more input variables are required for the **HP Variable Selection** node. The data set can contain at most one target variable. If a target variable is missing, then only unsupervised selection is available. The **HP Variable Selection** node does not support multiple target variables.

If your input data set contains a frequency variable, then the frequency variable must be an interval variable and all observations must be positive integers.

If you are running the **HP Variable Selection** node in a grid environment and using group processing, you cannot specify **Bagging** or **Boosting** for the Mode property of the **Start Groups** node.

HP Variable Selection Node Properties

HP Variable Selection Node General Properties

The following general properties are associated with the **HP Variable Selection** Node:

- **Node ID** — displays the ID that SAS Enterprise Miner assigns to a node in a process flow diagram. Node IDs are important when a process flow diagram contains two or more nodes of the same type. The first **HP Variable Selection** node that is added to a diagram has a Node ID of HPVS. The second **HP Variable Selection** node that is added to a diagram has a Node ID of HPVS2, and so on.
- **Imported Data** — provides access to the Imported Data — HP Variable Selection window. The Imported Data — HP Variable Selection window contains a list of the ports that provide data sources to the HP Variable Selection Node. Select the ... button to the right of the Imported Data property to open a table of the imported data.

  If data exists for an imported data source, you can select the row in the imported data table and click as follows:

  - **Browse** to open a window where you can browse the data set.
  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Exported Data** — provides access to the Exported Data — HP Variable Selection window. The Exported Data — HP Variable Selection window contains a list of the output data ports that the **HP Variable Selection** node creates data for when it runs. Select the ... button to the right of the Exported Data property to open a table that lists the exported data sets.

  If data exists for an imported data source, you can select the row in the imported data table and click as follows:

  - **Browse** to open a window where you can browse the data set.

  - **Explore** to open the Explore window, where you can sample and plot the data.
  - **Properties** to open the Properties window for the data source. The Properties window contains a **Table** tab and a **Variables** tab. The tabs contains summary information (metadata) about the table and the variables.
- **Notes** — Select the ... button to the right of the Notes property to open a window that you can use to store notes of interest, such as data or configuration information.

HP Variable Selection Node Train Properties

The following train properties are associated with the **HP Variable Selection** node:

- **Variables** — specifies the properties of each variable in the data source that you want to use. Select the ... button to the right of the Variables property to open a variables table. You can set the variable status to either **Use** or **Don't Use** in the table. You can set a variable's report status to **Yes** or **No**.
- **Pre-screening** — Set the Pre-screening property to **Yes** if you want to exclude variables that meet either of the following criteria:
  - The number of measurement levels is larger than the value specified in the Maximum Level property.
  - The percentage of missing data is larger than the value specified in the Maximum Missing Percent property.
- **Maximum Level** — Class variables with more measurement levels than specified here have their use status set to **Rejected**. This property is available only when the Pre-screening property is set to **Yes**.
- **Maximum Missing Percent** — Variables with a greater percentage of missing data than the value specified here have their use status set to **Rejected**. Valid values for this property are real numbers between 0 and 100. This property is available only when the Pre-screening property is set to **Yes**.
- **Target Model** — specifies the variable selection method.
  - **Unsupervised Selection** — identifies a set of variables that jointly explains the maximal amount of data variance. This method does not require a target variable. The HPREDUCE procedure is used in unsupervised selection.
  - **Supervised Selection** — requires exactly one target variable, and includes LASSO, LARS, and stepwise regression analysis. The HPREG procedure is used for binary or interval target variables. The HPLOGISTIC procedure is used for nominal or ordinal target variables.
  - **Sequential Selection** — runs unsupervised selection and supervised selection, sequentially.

HP Variable Selection Node Train Properties: Unsupervised Selection

- **Maximum Steps** — specifies the maximum number of steps to take for variable selection.
- **Maximum Effects** — specifies the maximum number of effects to select.
- **Correlation Statistics** — specifies the statistics that determine variable selection.

  - **Covariance** — selects variables based on the covariance matrix.
  - **Correlation** — selects variables based on the correlation matrix.
  - **Sum of Squares and Crossproducts** — selects variables based on the sum of squares and cross-product matrices.
- **Cumulative Cutoff** — specifies the fraction of the total variance to be explained by the selected variables.
- **Increment** — specifies the minimal increment of explained variance allowed after the cumulative cutoff value is reached.

HP Variable Selection Node Train Properties: Supervised Selection

- **Intercept** — Set the Intercept property to **Yes** if you want to include the intercept. The intercept is required for nominal and ordinal target variables. The default setting is **No**.
- **Selection Method** — specifies the method that is used to select effects.
  - **Fast Selection** — For binary or interval target variables, fast selection starts with no effects in the model and adds effects until the entry significance level is met. For nominal or ordinal target variables, fast selection starts with all effects in the model and deletes effects until the exit significance level is met.
  - **LAR** — The least angle regression (LAR) method starts with no effects in the model and adds effects. The estimates at any step are reduced when compared to the corresponding least squares estimates. The **LAR** option is available only for binary and interval target variables.
  - **LASSO** — The LASSO method adds and deletes variables based on a version of ordinary least squares where the sum of the absolute regression coefficients is constrained. The **LASSO** option is available only for binary and interval target variables.
- **Choose Criteria** — During the selection process, variables are chosen based on the selection criterion specified here.
  - **SBC** — chooses the model that has the smallest Schwarz Bayesian Criterion value.
  - **AIC** — chooses the model that has the smallest Akaike Information Criterion value.
  - **AICC** — chooses the model that has the smallest Corrected Akaike Information Criterion value.
- **Stop Criteria** — specifies the criterion that is used to stop the selection process.
  - **Max Steps** — the maximum number of steps to take for variable selection
  - **SBC** — Schwarz Bayesian Criterion value
  - **AIC** — Akaike Information Criterion value
  - **AICC** — Corrected Akaike Information Criterion value
  - **Significance Level** — Significance Level
- **Maximum Steps** — specifies the maximum number of selection steps that are performed.
- **Collinearity Diagnostics** — Set the Collinearity Diagnostics property to **Yes** if you want the model to produce variance inflation factors with the parameter estimates.

The Collinearity Diagnostics property is available only for binary and interval target variables.

HP Variable Selection Node Status Properties

The following status properties are associated with this node:

- **Create Time** — displays the time that the node was created.
- **Run ID** — displays the identifier of the node run. A new identifier is created every time the node runs.
- **Last Error** — displays the error message from the last run.
- **Last Status** — displays the last reported status of the node.
- **Last Run Time** — displays the time at which the node was last run.
- **Run Duration** — displays the length of time of the last node run.
- **Grid Host** — displays the grid server that was used during the node run.
- **User-Added Node** — specifies if the node was created by a user as a SAS Enterprise Miner extension node.

HP Variable Selection Node Results

After a successful node run, you can open the Results window of the **HP Variable Selection** node by right-clicking the node and selecting **Results** from the pop-up menu.

Select **View** from the main menu in the Results window to view the following:

- **Properties**
  - **Settings** — displays a window with a read-only table of the **HP Variable Selection** node properties configuration when the node was last run.
  - **Run Status** — displays the status of the **HP Variable Selection** node run. Information about the run start time, run duration, and completion status are displayed in this window.
  - **Variables** — displays a table of the variables in the training data set.
  - **Train Code** — displays the code that SAS Enterprise Miner used to train the node.
  - **Notes** — displays the notes that are associated with this node.
- **SAS Results**
  - **Log** — the SAS log of the **HP Variable Selection** node run.
  - **Output** — the SAS output of the **HP Variable Selection** node run.
  - **Flow Code** — The **HP Variable Selection** node does not generate flow code.
- **Scoring**
  - **SAS Code** — The **HP Variable Selection** node does not generate SAS code.
  - **PMML Code** — The **HP Variable Selection** node does not generate PMML code.

- **Model** — graphs and tables with information about the variables in the model. The available graphs and tables are as follows:
  - **Variable Explained by HPReduce** — A selection summary table displays what variable (or effects for class variables) is selected in each step. Also, the total variance that is explained by the variables selected is given. This table is available for both the unsupervised selection model and sequential selection model.
  - **Variable Selection by HPReduce** — This table provides the use status, role, measurement level, and reason for rejection or inclusion for the variables in the input data set. This table is available for both the unsupervised selection model and sequential selection model.
  - **Parameter Estimation by HPReg** — This table displays the parameters (or effects for class variables) in the selected model. Also, the estimates, degrees of freedom (DF), standard error, standardized estimates, t-value, and two-tailed significance probability (Pr > |t|) are provided. This table is available for both the supervised selection model and sequential selection model, but only for binary or interval target variables.
  - **Variable Selection by HPReg** — This table provides the use status, role, measurement level, and reason for rejection or inclusion for the variables in the input data set. This table is available for both the supervised selection model and sequential selection model, but only for binary or interval target variables.
  - **Fit Statistics by HPReg** — A table of fit statistics for the selected model, such as root mean square error, R-square, Adjusted R-square, AIC, AICC, SBC, and ASE. This table is available for both the supervised selection model and sequential selection model, but only for binary or interval target variables.
  - **ANOVA by HPReg** — This table displays an analysis of variance for the selected model. This table is available for both the supervised selection model and sequential selection model, but only for binary or interval target variables.
  - **Parameter Estimation by HPLgistic** — This table displays the parameters (or effects for class variables) in the selected model. Also, the estimates, degrees of freedom (DF), standard error, standardized estimates, t-value, and two-tailed significance probability (Pr > |t|) are provided. This table is available for both the supervised selection model and sequential selection model, but only for nominal or ordinal target variables.
  - **Variable Selection by HPLogistic** — This table provides the use status, role, measurement level, and reason for rejection or inclusion for the variables in the input data set. This table is available for both the supervised selection model and sequential selection model, but only for nominal or ordinal target variables.
  - **Global Test by HPLogistic** — This table provides a statistical test for the hypothesis of whether the final model provides a better fit than a model without effects (an “intercept-only” model). This table is available for both the supervised selection model and sequential selection model, but only for nominal or ordinal target variables.
  - **Fit Statistics by HPLogistic** — a table of fit statistics for the selection model, such as the log-likelihood, AIC, AICC, and BIC. This table is available for both the supervised selection model and sequential selection model, but only for nominal or ordinal target variables.
- **Plots**

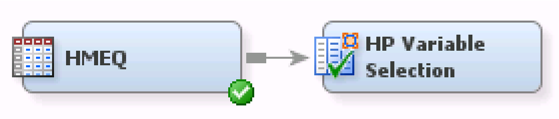
  - **Parameter Estimates** — A bar chart that displays the absolute values of the parameter estimates. The color of the bar indicates the sign of the parameter estimate.
  - **Variance Explained** — A bar chart that displays the percentage of variance that is explained by the variables selected. This plot is available for both the unsupervised selection model and sequential selection model.
  - **Solution Path** — This plot displays the standardized coefficients for all of the effects selected at each step in the stepwise selection method. This plot is available for both the supervised selection model and sequential selection model, but only for binary or interval target variables.
  - **Iteration Plot** — This plot displays the change in selection criterion as effects enter the model. This plot is available for both the supervised selection model and sequential selection model.
- **Table** — displays a table that contains the underlying data used to produce a chart. The **Table** menu item is dimmed and unavailable unless a results chart is open and selected.
- **Plot** — Use the Graph wizard to modify an existing Results plot or create a Results plot of your own. The **Plot** menu item is dimmed and unavailable unless a Results chart or table is open and selected.

HP Variable Selection Node Example

This example uses the sample SAS data set called Sampsio.Hmeq. You must use the data set to create a SAS Enterprise Miner data source. Right-click the **Data Sources** folder in the Project Navigator and select **Create Data Source** to launch the Data Source wizard.

- Choose **SAS Table** as your metadata source and click **Next**.
- Enter **`Sampsio.Hmeq`** in the **Table** field and click **Next**.
- Continue to the Metadata Advisor step and choose the **Basic Metadata Advisor**.
- In the Column Metadata window, set the role of the variable Bad to **Target** and set the level of the variable Bad to **Binary**. Click **Next**.
- There is no decision processing. Click **Next**.
- In the Create Sample window, you are asked if you want to create a sample data set. Select **No**. Click **Next**.
- Set the role of the HMEQ data set to **Train**, and then click **Finish**.

Drag the HMEQ data set and the **HP Variable Selection** node to your diagram workspace. Connect them as shown in the diagram below.



Select the **HP Variable Selection** node and change the following properties:

- Set the value for **Correlation Statistics** to **Correlation**.

- Set the value for **Selection Method** to **LAR**.

Right-click the **HP Variable Selection** node and select **Run**. In the Confirmation window, select **Yes**. After a successful run of the **HP Variable Selection** node, select **Results** in the Run Status window.

Notice the following results:

- Variable Selection by HPReg — In this example, the variables Mortdue, Reason, and Value are rejected. In this example, the HPREDUCE procedure and the HPREG procedure are responsible for identifying the role of each variable.
- Variance Explained — The variance-explained plot displays the percentage of variance that is explained by the effects in the model at each step. Each bar indicates the base variance explained and the incremental variance explained. The base variance explained is the total variance explained by the effects in the model before a new effect enters the model. The incremental variance explained is the total variance explained by the new effect that entered the model in that iteration.
- Parameter Estimates — The parameter estimates bar chart is color-coded to indicate the sign of the parameter estimate. Notice that Job_Sales has a relatively strong, positive parameter estimate and Job_Office has a relatively strong, negative parameter estimate.
- Iteration Plot — The iteration plot shows how the criterion used to choose the selected model changes as the effects enter the model. Notice that the graph achieves a minimum value at the 12th step, which is where model selection terminates.
- Solution Path — The solution path plot enables you to assess the relative importance of the effects selected at any step in the selection process. Also, it provides information as to when effects entered the model. The blue vertical line appearing at step 12 indicates that the model was selected at that step because the optimal value of the Schwarz-Bayesian Coefficient was reached.

Appendix 1

Nodes Available for Connection to the High Performance Data Mining Nodes

The High-Performance Data Mining nodes support connections with the following nodes:

- Control Point
- End Groups
- Ext Demo
- Metadata
- Model Comparison
- Reporter
- SAS Code
- Score
- Start Groups