Getting Started with

SAS® Enterprise Miner™ 5.2

The Power to Know.®

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2006. *Getting Started with SAS® Enterprise Miner™ 5.2*. Cary, NC: SAS Institute Inc.

Getting Started with SAS® Enterprise Miner™ 5.2

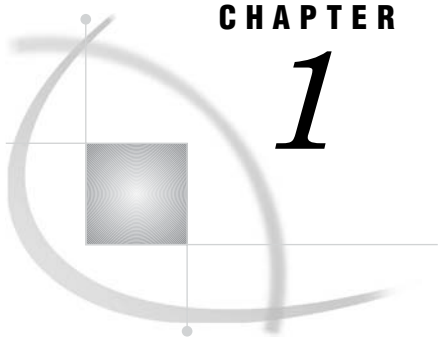
ISBN-13: 978-1-59994-002-1
ISBN-10: 1-59994-002-7

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

1st printing, April 2006

SAS Publishing provides a complete selection of books and electronic products to help customers use SAS software to its fullest potential. For more information about our e-books, e-learning products, CDs, and hard-copy books, visit the SAS Publishing Web site at **support.sas.com/pubs** or call 1-800-727-3228.

Contents



CHAPTER

1

Introduction to SAS Enterprise Miner 5.2 Software

Data Mining Overview

SAS defines *data mining* as the process of uncovering hidden patterns in large amounts of data. Many industries use data mining to address business problems and opportunities such as fraud detection, risk and affinity analyses, database marketing, householding, customer churn, bankruptcy prediction, and portfolio analysis.The SAS data mining process is summarized in the acronym SEMMA, which stands for sampling, exploring, modifying, modeling, and assessing data.

- *Sample* the data by creating one or more data tables. The sample should be large enough to contain the significant information, yet small enough to process.
- *Explore* the data by searching for anticipated relationships, unanticipated trends, and anomalies in order to gain understanding and ideas.
- *Modify* the data by creating, selecting, and transforming the variables to focus the model selection process.
- *Model* the data by using the analytical tools to search for a combination of the data that reliably predicts a desired outcome.

- *Assess* the data by evaluating the usefulness and reliability of the findings from the data mining process.

You might not include all of these steps in your analysis, and it might be necessary to repeat one or more of the steps several times before you are satisfied with the results. After you have completed the assessment phase of the SEMMA process, you apply the scoring formula from one or more champion models to new data that might or might not contain the target. The goal of most data mining tasks is to apply models that are constructed using training and validation data in order to make accurate predictions about observations of new, raw data.

The SEMMA data mining process is driven by a process flow diagram, which you can modify and save. The GUI is designed in such a way that the business analyst who has little statistical expertise can navigate through the data mining methodology, while the quantitative expert can go "behind the scenes" to fine-tune the analytical process.

SAS Enterprise Miner 5.2 contains a collection of sophisticated analysis tools that have a common user-friendly interface that you can use to create and compare multiple models. Statistical tools include clustering, self-organizing maps / Kohonen, variable selection, trees, linear and logistic regression, and neural networking. Data preparation tools include outlier detection, variable transformations, data imputation, random sampling, and the partitioning of data sets (into train, test, and validate data sets). Advanced visualization tools enable you to quickly and easily examine large amounts of data in multidimensional histograms and to graphically compare modeling results.

Enterprise Miner is designed for PCs or servers that are running under Windows XP, UNIX, Linux, or subsequent releases of those operating environments. The figures and screen captures that are presented in this document were taken on a PC that was running under Windows XP.

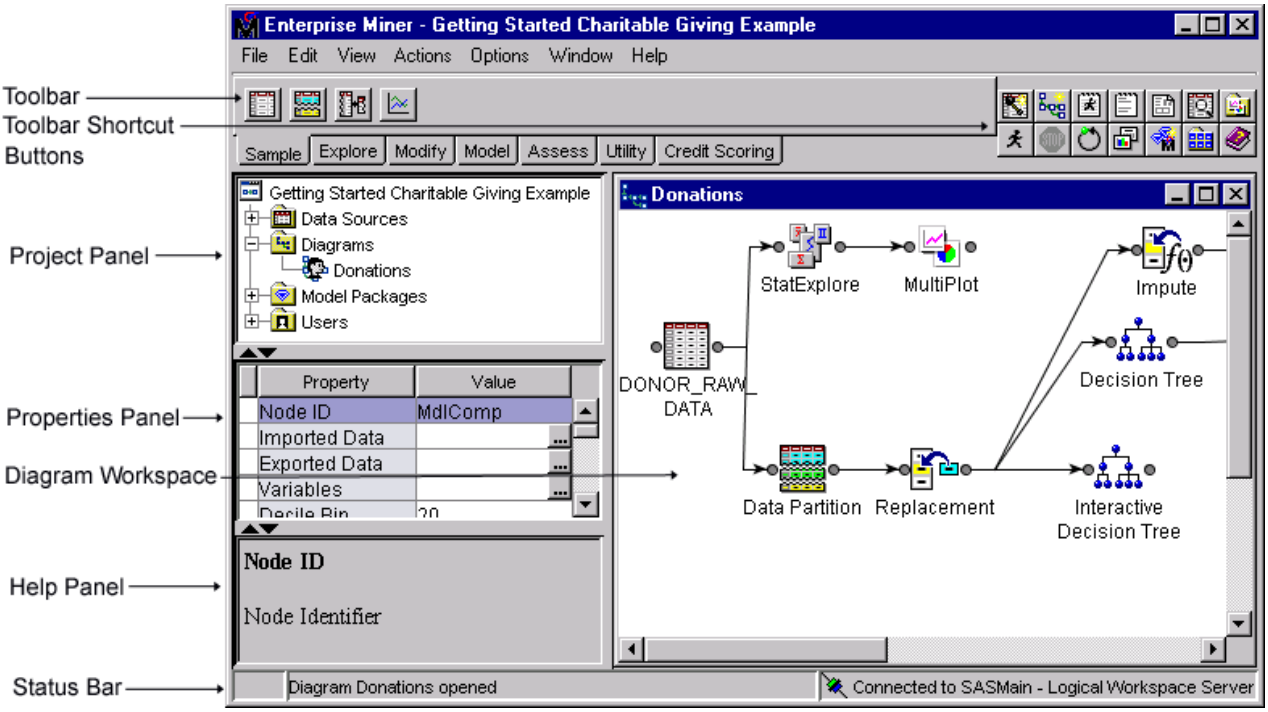
Layout of the Enterprise Miner Window

About the Graphical Interface

You use the Enterprise Miner graphical interface to build a process flow diagram that controls your data mining project.

Figure 1.1 shows the components of the Enterprise Miner window.

Figure 1.1 The Enterprise Miner Window



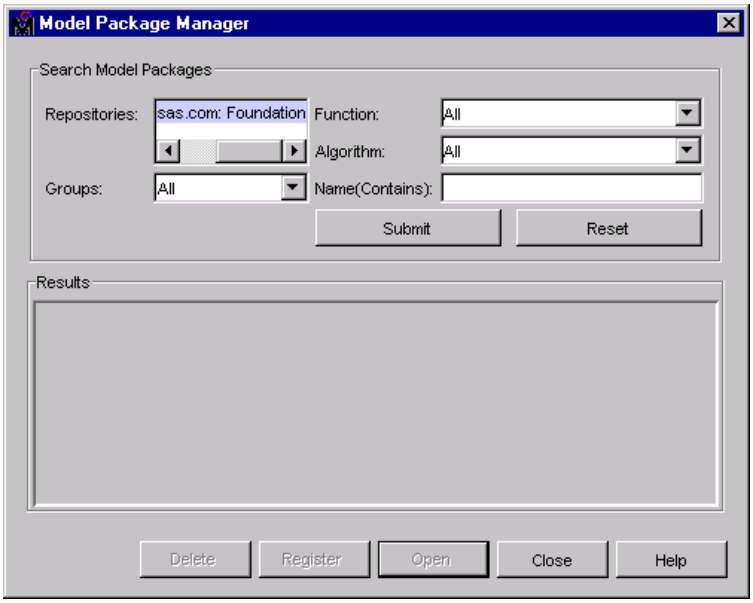
The Enterprise Miner window contains the following interface components:

- Toolbar and Toolbar shortcut buttons — The Enterprise Miner Toolbar is a graphic set of node icons that are organized by SEMMA categories. To the right side of the toolbar is a collection of Toolbar shortcut buttons that are commonly used to build process flow diagrams in the Diagram Workspace. Move the mouse pointer over any node, or shortcut button to see the text name. Drag a node or tool into the Diagram Workspace to use it. The Toolbar icon remains in place and the node in the Diagram Workspace is ready to be connected and configured for use in your process flow diagram. Click on a shortcut button to use it.
- Project Panel — Use the Project Panel to manage and view data sources, diagrams, model packages, and project users.
- Properties Panel — Use the Properties Panel to view and edit the settings of data sources, diagrams, nodes, model packages, and users.
- Diagram Workspace — Use the Diagram Workspace to build, edit, run, and save process flow diagrams. This is where you graphically build, order, sequence and connect the nodes that you use to mine your data and generate reports.
- Help Panel — The Help Panel displays a short description of the property that you select in the Properties Panel. Extended help can be found in the Help Topics selection from the Help main menu or from the Help button on many windows.
- Status Bar — The Status Bar is a single pane at the bottom of the window that indicates the execution status of a SAS Enterprise Miner task.

Enterprise Miner Menus

Here is a summary of the Enterprise Miner menus:

- File
  - New
    - Project — creates a new project.
    - Diagram — creates a new diagram.
    - Data Source — creates a new data source using the Data Source wizard.
  - Open Project — opens an existing project. You can also create a new project from the Open Project window.
  - Recent Projects — lists the projects on which you were most recently working.
  - Open Model Package — opens a model package SAS Package (SPK) file that you have previously created.
  - Explore Model Packages — opens the Model Package Manager window, in which you can view and compare model packages.



  - Open Diagram — opens the diagram that you select in the Project Panel.
  - Close Diagram — closes the open diagram that you select in the Project Panel.
  - Close this Project — closes the current project.
  - Delete this Project — deletes the current project.
  - Import Diagram from XML — imports a diagram that has been defined by an XML file.
  - Save Diagram As — saves a diagram as an image (BMP or GIF) or as an XML file.
  - Print Diagram — prints the contents of the window that is open in the Diagram Workspace.
  - Exit — ends the Enterprise Miner session and closes the window.
- Edit
  - Cut — deletes the selected item and copies it to the clipboard.

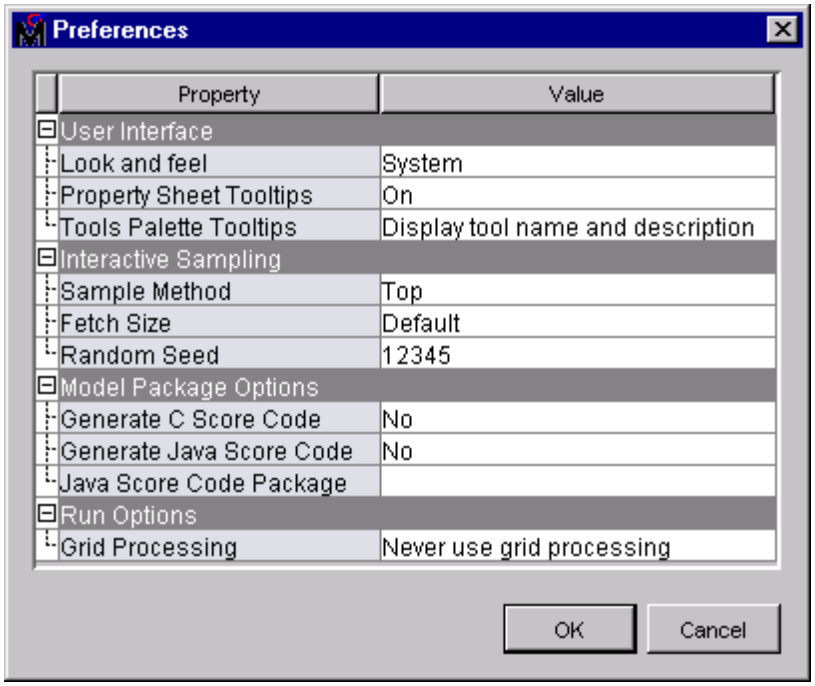
- Copy — copies the selected node to the clipboard.
- Paste — pastes a copied object from the clipboard.
- Delete — deletes the selected diagram, data source, or node.
- Rename — renames the selected diagram, data source, or node.
- Duplicate — creates a copy of the selected data source.
- Select All — selects all of the nodes in the open diagram, selects all texts in the Program Editor, Log, or Output windows.
- Clear All — clears text from the Program Editor, Log, or Output windows.
- Find/Replace — opens the Find/Replace window so that you can search for and replace text in the Program Editor, Log, and Results windows.
- Go To Line — opens the Go To Line window. Enter the line number on which you want to enter or view text.
- Layout
  - Horizontally — creates an orderly horizontal arrangement of the layout of nodes that you have placed in the Diagram Workspace.
  - Vertically — creates an orderly vertical arrangement of the layout of nodes that you have placed in the Diagram Workspace.
- Zoom — increases or decreases the size of the process flow diagram within the diagram window.

- View
  - Property Sheet
    - Basic — displays the basic properties in the Properties Panel.
    - Advanced — displays the basic and advanced properties in the Properties Panel.
    - Hide — removes the Properties Panel and the Help Panel from the user interface.
  - Program Editor — opens a SAS Program Editor window in which you can enter SAS code.
  - Log — opens a SAS Log window.
  - Output — opens a SAS Output window.
  - Graphs — opens the Graphs window. Graphs that you create with SAS code in the Program Editor are displayed in this window.
  - Table — opens a table from the libraries that you have defined. You select a table from the Select a SAS Table window.
  - Refresh Project — updates the project tree to incorporate any changes that were made to the project from outside the Enterprise Miner user interface.

- Actions
  - Add Node — adds a node that you have selected to the Diagram Workspace.
  - Select Nodes — opens the Select Nodes window.
  - Connect nodes — opens the Connect Nodes window. You must select a node in the Diagram Workspace to make this menu item available. You can connect the node that you select to any nodes that have been placed in your Diagram Workspace.
  - Update — updates the selected node to incorporate any changes that you have made.
  - Run — runs the selected node and any predecessor nodes in the process flow that have not been executed, or submits any code that you type in the Program Editor window.

- Stop Run — interrupts a currently running process flow.
- View Results — opens the Results window for the selected node.
- Create Model Package — generates a mining model package.
- Export Path as SAS Program — saves the path that you select as a SAS program. In the window that opens, you can specify the location to which you want to save the file. You also specify whether you want the code to run the path or create a model package.

- Options
  - Preferences — opens the Preferences window. Use the following options to change the user interface:



    - Look and Feel — you can select **`Cross Platform`**, which uses a standard appearance scheme that is the same on all platforms, or **`System`** which uses the appearance scheme that you have chosen for your platform.
    - Property Sheet Tooltips — controls whether tooltips are displayed on various property sheets appearing throughout the user interface.
    - Tools Palette Tooltips — controls how much tooltip information you want displayed for the tool icons in the tools palette.
    - Sample Methods — generates a sample that will be used for graphical displays. You can specify either **`Top`** or **`Random`**.
    - Fetch Size — specifies the number of observations to download for graphical displays.
    - Random Seed — specifies the value you want to use to randomly sample observations from your input data.
    - Generate C Score Code — creates C score code when you create a report. By default, this option is selected.
    - Generate Java Score Code — creates Java score code when you create a report. By default, this option is selected. If you select **`Generate Java`**

Score Code, then enter a filename for the score code package in the Java Score Code Package box.
- Java Score Code Package — identifies the filename of the Java Score Code package.
- Grid Processing — enables you to use grid processing when you are running data mining flows on grid-enabled servers.

- Window
  - Tile — displays windows in the Diagram Workspace so that all windows are visible at the same time.
  - Cascade — displays windows in the Diagram Workspace so that windows overlap.
- Help
  - Contents — opens the Enterprise Miner Help window, which enables you to view all the Enterprise Miner Reference Help.
  - Component Properties — opens a table that displays the component properties of each tool.
  - Generate Sample Data Sources — creates sample data sources that you can access from the Data Sources folder.
  - Configuration — displays the current system configuration of your Enterprise Miner session.
  - About — displays information about the version of Enterprise Miner that you are using.

Diagram Workspace Pop-up Menus

You can use the Diagram Workspace pop-up menus to perform many tasks. To open the pop-up menu, right-click in an open area of the Diagram Workspace. (Note that you can also perform many of these tasks by using the pull-down menus.) The pop-up menu contains the following items:

- **Add node** — accesses the Add Node window.
- **Paste** — pastes a node from the clipboard to the Diagram Workspace.
- **Select All** — selects all nodes in the process flow diagram.
- **Select Nodes** — opens a window that displays all the nodes that are on your diagram. You can select as many as you want.
- **Layout Nodes** — creates an orderly arrangement of the nodes in the Diagram Workspace.
- **Zoom** — increases or decreases the size of the process flow diagram within the diagram window by the amount that you choose.

Organization and Uses of Enterprise Miner Nodes

About Nodes

The nodes of Enterprise Miner are organized according to the Sample, Explore, Modify, Model, and Assess (SEMMA) data mining methodology. In addition, there are

also Credit Scoring and Utility node tools. You use the Credit Scoring node tools to score your data models and to create freestanding code. You use the Utility node tools to submit SAS programming statements, and to define control points in the process flow diagram.

All of the Enterprise Miner nodes are listed in a set of folders that are located on the **Tools** tab of the Enterprise Miner Project Navigator. The nodes are listed under the folder that corresponds to their data mining functions.

Note: The **Credit Scoring** tab does not appear in all installed versions of Enterprise Miner. △

Remember that in a data mining project, it can be an advantage to repeat parts of the data mining process. For example, you might want to explore and plot the data at several intervals throughout your project. It might be advantageous to fit models, assess the models, and then refit the models and then assess them again.

The following tables list the nodes, give each node's primary purpose, and supply examples and illustrations.

Sample Nodes

Node Name	Description
Input Data Source	Use the Input Data Source node to access SAS data sets and other types of data. This node introduces a predefined Enterprise Miner Data Source and metadata into a Diagram Workspace for processing. You can view metadata information about your data in the Input Data Source node, such as initial values for measurement levels and model roles of each variable. Summary statistics are displayed for interval and class variables. See Chapter 3.
Data Partition	Use the Data Partition node to partition data sets into training, test, and validation data sets. The training data set is used for preliminary model fitting. The validation data set is used to monitor and tune the model weights during estimation and is also used for model assessment. The test data set is an additional hold-out data set that you can use for model assessment. This node uses simple random sampling, stratified random sampling, or user defined partitions to create partitioned data sets. See Chapter 3.

Node Name	Description
Sample	Use the Sample node to take random, stratified random samples, and to take cluster samples of data sets. Sampling is recommended for extremely large databases because it can significantly decrease model training time. If the random sample sufficiently represents the source data set, then data relationships that Enterprise Miner finds in the sample can be extrapolated upon the complete source data set. The Sample node writes the sampled observations to an output data set and saves the seed values that are used to generate the random numbers for the samples so that you can replicate the samples.
Time Series	Use the Time Series node to convert transactional data to time series data to perform seasonal and trend analysis. This node enables you to understand trends and seasonal variations in the transaction data that you collect from your customers and suppliers over the time, by converting transactional data into time series data. Transactional data is time-stamped data that is collected over time at no particular frequency. By contrast, time series data is time-stamped data that is collected over time at a specific frequency. The size of transaction data can be very large, which makes traditional data mining tasks difficult. By condensing the information into a time series, you can discover trends and seasonal variations in customer and supplier habits that might not be visible in transactional data.

Explore Nodes

Node Name	Description
Association	Use the Association node to identify association relationships within the data. For example, if a customer buys a loaf of bread, how likely is the customer to also buy a gallon of milk? You use the Association node to perform sequence discovery if a time-stamped variable (a sequence variable) is present in the data set. Binary sequences are constructed automatically, but you can use the Event Chain Handler to construct longer sequences that are based on the patterns that the algorithm discovered.
Cluster	Use the Cluster node to segment your data so that you can identify data observations that are similar in some way. When displayed in a plot, observations that are similar tend to be in the same cluster, and observations that are different tend to be in different clusters. The cluster identifier for each observation can be passed to other nodes for use as an input, ID, or target variable. This identifier can also be passed as a group variable that enables you to automatically construct separate models for each group.

Node Name	Description
MultiPlot	Use the MultiPlot node to explore larger volumes of data graphically. The MultiPlot node automatically creates bar charts and scatter plots for the input and target variables without requiring you to make several menu or window item selections. The code that is created by this node can be used to create graphs in a batch environment. See Chapter 3.
Path Analysis	Use the Path Analysis node to analyze Web log data and to determine the paths that visitors take as they navigate through a Web site. You can also use the node to perform sequence analysis.
SOM/Kohonen	Use the SOM/Kohonen node to perform unsupervised learning by using Kohonen vector quantization (VQ), Kohonen self-organizing maps (SOMs), or batch SOMs with Nadaraya-Watson or local-linear smoothing. Kohonen VQ is a clustering method, whereas SOMs are primarily dimension-reduction methods.
StatExplore	Use the StatExplore node to examine variable distributions and statistics in your data sets. You can use the StatExplore node to compute standard univariate distribution statistics, to compute standard bivariate statistics by class target and class segment, and to compute correlation statistics for interval variables by interval input and target. You can also combine the StatExplore node with other Enterprise Miner tools to perform data mining tasks such as using the StatExplore node with the Metadata node to reject variables, using the StatExplore node with the Transform Variables node to suggest transformations, or even using the StatExplore node with the Regression node to create interactions terms. See Chapter 3.
Variable Selection	Use the Variable Selection node to evaluate the importance of input variables in predicting or classifying the target variable. To preselect the important inputs, the Variable Selection node uses either an R-Square or a Chi-Square selection (tree-based) criterion. You can use the R-Square criterion to remove variables in hierarchies, remove variables that have large percentages of missing values, and remove class variables that are based on the number of unique values. The variables that are not related to the target are set to a status of rejected. Although rejected variables are passed to subsequent nodes in the process flow diagram, these variables are not used as model inputs by a more detailed modeling node, such as the Neural Network and Decision Tree nodes. You can reassign the status of the input model variables to rejected in the Variable Selection node. See Chapter 5.

Modify Nodes

Node Name	Description
Drop	Use the Drop node to drop certain variables from your scored Enterprise Miner data sets. You can drop variables that have roles of Assess, Classification, Frequency, Hidden, Input, Rejected, Residual, and Target from your scored data sets.
Filter	Use the Filter node to apply a filter to the training data set in order to exclude outliers or other observations that you do not want to include in your data mining analysis. The Filter node does not filter observations in the validation, test, or score data sets. Checking for outliers is recommended as outliers can greatly affect modeling results and, subsequently, the classification and prediction precision of fitted models.
Impute	Use the Impute node to impute (fill in) values for observations that have missing values. You can replace missing values for interval variables with the mean, median, midrange, mid-minimum spacing, distribution-based replacement. Alternatively, you can use a replacement M-estimator such as Tukey's biweight, Hubers, or Andrew's Wave. You can also estimate the replacement values for each interval input by using a tree-based imputation method. Missing values for class variables can be replaced with the most frequently occurring value, distribution-based replacement, tree-based imputation, or a constant. See Chapter 5.
Principal Components	Use the Principal Components node to perform a principal components analysis for data interpretation and dimension reduction. The node generates principal components that are uncorrelated linear combinations of the original input variables and that depend on the covariance matrix or correlation matrix of the input variables. In data mining, principal components are usually used as the new set of input variables for subsequent analysis by modeling nodes.

Node Name	Description
Replacement	Use the Replacement node to impute (fill in) values for observations that have missing values and to replace specified non-missing values for class variables in data sets. You can replace missing values for interval variables with the mean, median, midrange, or mid-minimum spacing, or with a distribution-based replacement. Alternatively, you can use a replacement M-estimator such as Tukey's biweight, Huber's, or Andrew's Wave. You can also estimate the replacement values for each interval input by using a tree-based imputation method. Missing values for class variables can be replaced with the most frequently occurring value, distribution-based replacement, tree-based imputation, or a constant. See Chapters 3, 4, and 5.
Transform Variables	Use the Transform Variables node to create new variables that are transformations of existing variables in your data. Transformations are useful when you want to improve the fit of a model to the data. For example, transformations can be used to stabilize variances, remove nonlinearity, improve additivity, and correct nonnormality in variables. In Enterprise Miner, the Transform Variables node also enables you to transform class variables and to create interaction variables. See Chapter 5.

Model Nodes

Node Name	Description
AutoNeural	Use the AutoNeural node to automatically configure a neural network. It conducts limited searches for a better network configuration. See Chapters 5 and 6.
Decision Tree	Use the Decision Tree node to fit decision tree models to your data. The implementation includes features that are found in a variety of popular decision tree algorithms such as CHAID, CART, and C4.5. The node supports both automatic and interactive training. When you run the Decision Tree node in automatic mode, it automatically ranks the input variables, based on the strength of their contribution to the tree. This ranking can be used to select variables for use in subsequent modeling. You can override any automatic step with the option to define a splitting rule and prune explicit tools or subtrees. Interactive training enables you to explore and evaluate a large set of trees as you develop them. See Chapters 4 and 6.
Dmine Regression	Use the Dmine Regression node to compute a forward stepwise least-squares regression model. In each step, an independent variable is selected that contributes maximally to the model R-square value.
DMNeural	Use DMNeural node to fit an additive nonlinear model. The additive nonlinear model uses bucketed principal components as inputs to predict a binary or an interval target variable.

Node Name	Description
Ensemble	Use the Ensemble node to create new models by combining the posterior probabilities (for class targets) or the predicted values (for interval targets) from multiple predecessor models.
MBR (Memory-Based Reasoning)	Use the MBR (Memory-Based Reasoning) node to identify similar cases and to apply information that is obtained from these cases to a new record. The MBR node uses *k*-nearest neighbor algorithms to categorize or predict observations.
Neural Network	Use the Neural Network node to construct, train, and validate multilayer feedforward neural networks. By default, the Neural Network node automatically constructs a multilayer feedforward network that has one hidden layer consisting of three neurons. In general, each input is fully connected to the first hidden layer, each hidden layer is fully connected to the next hidden layer, and the last hidden layer is fully connected to the output. The Neural Network node supports many variations of this general form. See Chapters 5 and 6.
Regression	Use the Regression node to fit both linear and logistic regression models to your data. You can use continuous, ordinal, and binary target variables. You can use both continuous and discrete variables as inputs. The node supports the stepwise, forward, and backward selection methods. A point-and-click term editor enables you to customize your model by specifying interaction terms and the ordering of the model terms. See Chapters 5 and 6.
Rule Induction	Use the Rule Induction node to improve the classification of rare events in your modeling data. The Rule Induction node creates a Rule Induction model that uses split techniques to remove the largest pure split node from the data. Rule Induction also creates binary models for each level of a target variable and ranks the levels from the most rare event to the most common. After all levels of the target variable are modeled, the score code is combined into a SAS DATA step.
TwoStage	Use the TwoStage node to compute a two-stage model for predicting a class and an interval target variables at the same time. The interval target variable is usually a value that is associated with a level of the class target.

Note: These modeling nodes use a directory table facility, called the Model Manager, in which you can store and access models on demand. The modeling nodes also enable you to modify the target profile or profiles for a target variable. △

Assess Nodes

Node Name	Description
Decisions	Use the Decisions node to define target profiles for a target that produces optimal decisions. The decisions are made using a user-specified decision matrix and output from a subsequent modeling procedure.
Model Comparison	Use the Model Comparison node to use a common framework for comparing models and predictions from any of the modeling tools (such as Regression, Decision Tree, and Neural Network tools). The comparison is based on the expected and actual profits or losses that would result from implementing the model. The node produces the following charts that help to describe the usefulness of the model: lift, profit, return on investment, receiver operating curves, diagnostic charts, and threshold-based charts. See Chapter 6.
Segment Profile	Use the Segment Profile node to assess and explore segmented data sets. Segmented data is created from data BY-values, clustering, or applied business rules. The Segment Profile node facilitates data exploration to identify factors that differentiate individual segments from the population, and to compare the distribution of key factors between individual segments and the population. The Segment Profile node outputs a Profile plot of variable distributions across segments and population, a Segment Size pie chart, a Variable Worth plot that ranks factor importance within each segment, and summary statistics for the segmentation results. The Segment Profile node does not generate score code or modify metadata.
Score	Use the Score node to manage, edit, export, and execute scoring code that is generated from a trained model. Scoring is the generation of predicted values for a data set that might not contain a target variable. The Score node generates and manages scoring formulas in the form of a single SAS DATA step, which can be used in most SAS environments even without the presence of Enterprise Miner. See Chapter 6.

Utility Nodes

Node Name	Description
Control Point	Use the Control Point node to establish a control point to reduce the number of connections that are made in process flow diagrams. For example, suppose three Input Data nodes are to be connected to three modeling nodes. If no Control Point node is used, then nine connections are required to connect all of the Input Data nodes to all of the modeling nodes. However, if a Control Point node is used, only six connections are required.
Merge	Use the Merge node to merge observations from two or more data sets or more into a single observation in a new data set. The Merge node supports both one-to-one and match merging. In addition, you can choose not to overwrite certain variables (such predicted values and posterior probabilities), depending on the settings of the node.
Metadata	Use the Metadata node to modify the columns metadata information at some point in your process flow diagram. You can modify attributes such as roles, measurement levels, and order.
SAS Code	Use the SAS Code node to incorporate new or existing SAS code into process flows that you develop using Enterprise Miner. The SAS Code node extends the functionality of Enterprise Miner by making other SAS procedures available in your data mining analysis. You can also write a SAS DATA step to create customized scoring code, to conditionally process data, and to concatenate or to merge existing data sets. See Chapter 6.

Usage Rules for Nodes

Here are some general rules that govern the placement of nodes in a process flow diagram:

- The Input Data Source node cannot be preceded by any other nodes.
- All nodes except the Input Data Source and SAS Code nodes must be preceded by a node that exports a data set.
- The SAS Code node can be defined in any stage of the process flow diagram. It does not require an input data set that is defined in the Input Data Source node.
- The Assessment node must be preceded by one or more modeling nodes.
- The Score node must be preceded by a node that produces score code. For example, the modeling nodes produce score code.
- The Ensemble node must be preceded by a modeling node.
- The Replacement node must follow a node that exports a data set, such as a Data Source, Sample, or Data Partition node.

Overview of the SAS Enterprise Miner 5.2 Getting Started Example

This book uses an extended example that is intended to familiarize you with the many features of Enterprise Miner. Several key components of the Enterprise Miner process flow diagram are covered.

In this step-by-step example you learn to do basic tasks in Enterprise Miner: you create a project and build a process flow diagram. In your diagram you perform tasks such as accessing data, preparing the data, building multiple predictive models, comparing the models, selecting the best model, and applying the chosen model to new data (known as scoring data). You also perform tasks such as filtering data, exploring data, and transforming variables. The example is designed to be used in conjunction with Enterprise Miner software. For details see "Configure SAS Enterprise Miner 5.2 for the Example" on page 17.

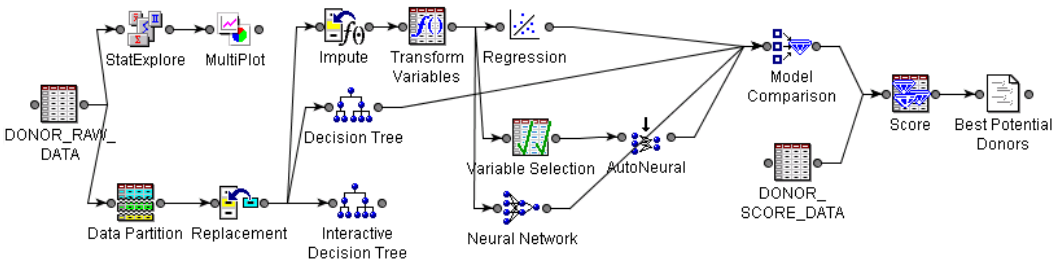
Example Problem Description

A national charitable organization seeks to better target its solicitations for donations. By only soliciting the most likely donors, less money will be spent on solicitation efforts and more money will be available for charitable concerns. Solicitations involve sending a small gift to an individual along with a request for a donation. Gifts include mailing labels and greeting cards.

The organization has more than 3.5 million individuals in its mailing database. These individuals have been classified by their response to previous solicitation efforts. Of particular interest is the class of individuals who are identified as lapsing donors. These individuals have made their most recent donation between 12 and 24 months ago. The organization has found that by predicting the response of this group, they can use the model to rank all 3.5 million individuals in their database. The campaign refers to a greeting card mailing sent in June of 1997. It is identified in the raw data as the 97NK campaign.

When the most appropriate model for maximizing solicitation profit by screening the most likely donors is determined, the scoring code will be used to create a new score data set that is named DONOR.ScoreData. Scoring new data that does not contain the target is the end result of most data mining applications.

When you are finished with this example, your process flow diagram will resemble the one shown below.



Here is a preview of topics and tasks in this example:

Chapter	Task
2	Create your project, define the data source, configure the metadata, define prior probabilities and profit matrix, and create an empty process flow diagram.
3	Define the input data, explore your data by generating descriptive statistics and creating exploratory plots. You will also partition the raw data and replace missing data.
4	Create a decision tree and interactive decision tree models.
5	Impute missing values and create variable transformations. You will also develop regression, neural, and auto neural models. Finally, you will use the preliminary variable selection node.
6	Assess and compare the models. Also, you will score new data using the models.
7	Create model results packages, register your models, save and import the process flow diagram in XML.

Note: The complete process flow diagram is provided in XML format at **`http://support.sas.com/documentation/onlinedoc/miner`** under the Tutorials and Samples heading. In order to use the provided XML, you must do the following:

- Complete all the instructions in "Create a New Project" on page 21.
- Complete all the instructions in "Define the Donor Data Source" on page 23.
- Complete all the instructions in importing XML diagrams in "Save and Import Diagrams in XML" on page 132.

This example provides an introduction to using Enterprise Miner in order to familiarize you with the interface and the capabilities of the software. The example is not meant to provide a comprehensive analysis of the sample data. △

Example Data Description

See Appendix 2, "Example Data Description," on page 137 for a list of variables that are used in this example.

Configure SAS Enterprise Miner 5.2 for the Example

Software Requirements

In order to re-create this example, you must have access to SAS Enterprise Miner 5.2 software, either as client/server application, or as a complete client on your local machine. In order to complete all the portions of the example, you also must have the Enterprise Miner Tree Desktop Application installed.

Locate and Install the Example Data

Download the **donor_raw_data.sas7bdat** and **donor_score_data.sas7bdat** data sets from **http://support.sas.com/documentation/onlinedoc/miner** under the Tutorials and Samples heading.

See “Configure Example Data on a Metadata Server” on page 18 for details about how to define and set up your data sets.

Configure Example Data on a Metadata Server

This example is designed to be performed on a two-tier Enterprise Miner 5.2 client/server installation, the most common customer configuration. Ask your system administrator to create a library in your Enterprise Miner server environment to contain the example data. You and other example users will also need access to the example data library.

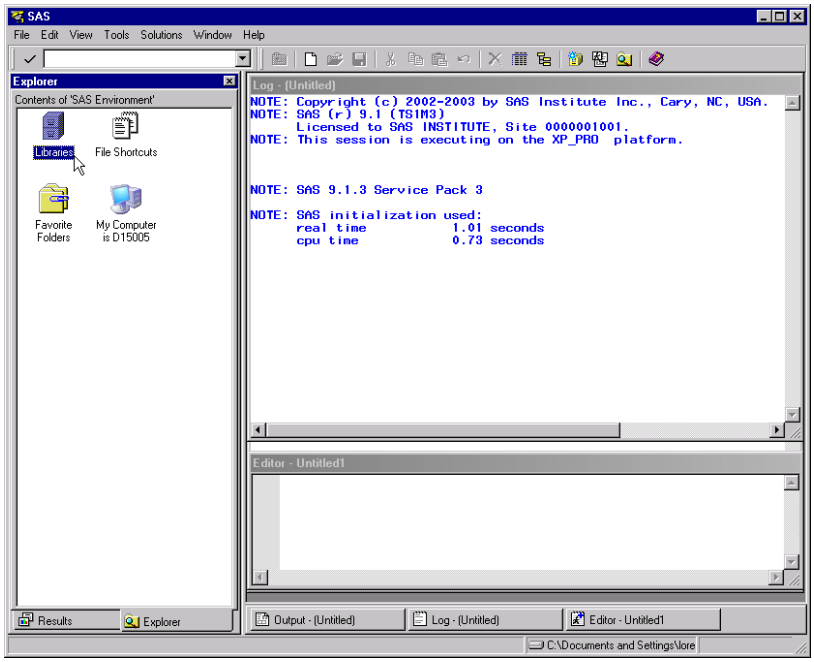
Configure Your Data on an Enterprise Miner Complete Client

If you access Enterprise Miner 5.2 as a complete client, define the donor sample data source in your local machine.

When you create a library, you give SAS a shortcut name or pointer to a storage location in your operating environment where you store SAS files.

To create a new SAS library for your sample donor data using SAS 9.1.3, complete the following steps:

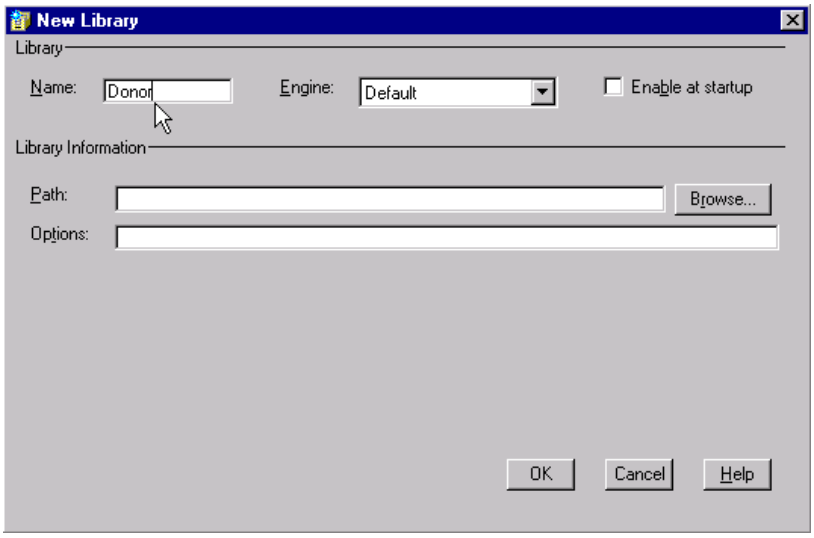
1 From the Explorer window, select the Libraries folder.



2 Select **File** ▶ **New** .

3 In the **Name** box of the New Library window, enter a library reference. The library name is **Donor** in this example.

Note: Library names are limited to eight characters. △



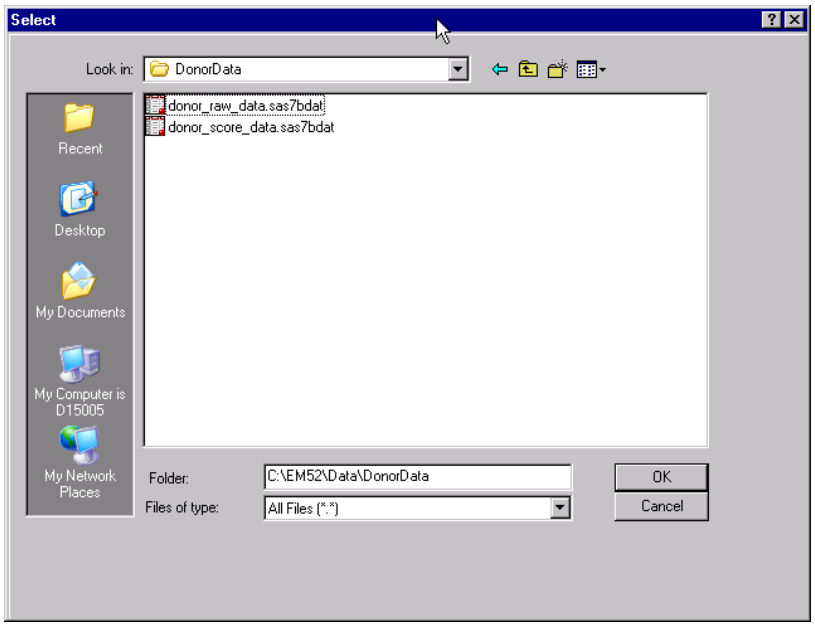
4 Select an engine type. The engine type determines what fields are available in the Library Information area. If you are not sure which engine to choose, use the Default engine (which is selected automatically). The Default engine enables SAS to choose which engine to use for any data sets that exist in your new library. If no data sets exist in your new library, then the Base SAS engine is assigned.

5 Select the **`Enable at startup`** check box in the New Library window.

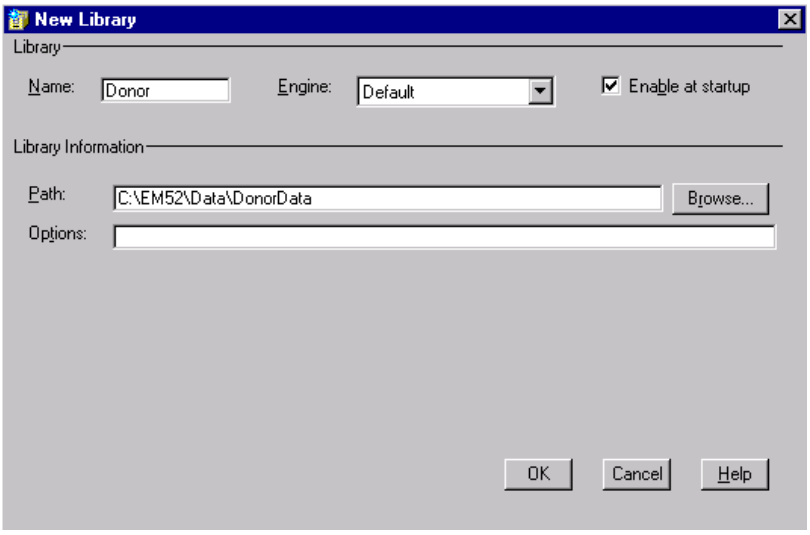
6 Type the appropriate information in the fields of the Library Information area. The fields that are available in this area depend on the engine that you select.

7 For this example, click Browse.

8 In the Select window, navigate to the folder where you downloaded the sample data sets **`donor_raw_data.sas7bdat`** and **`donor_score_data.sas7bdat`**.

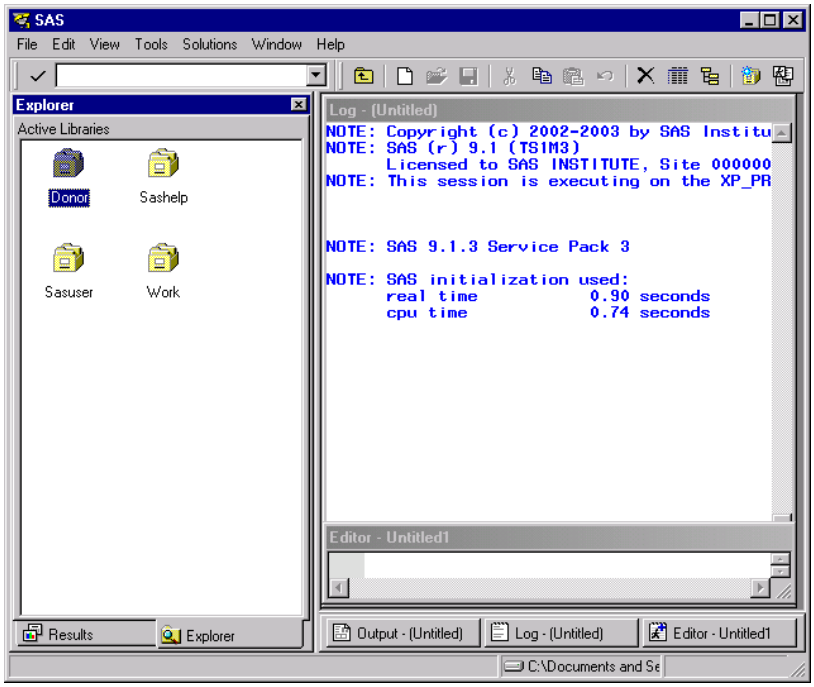


9 Click OK. This selected path will appear in the **`Path`** box of the New Library window.



10 Enter any options that you want to specify. For this example, leave the **`Options`** box blank.

11 Click OK. The new library will appear under Libraries in the Explorer window.



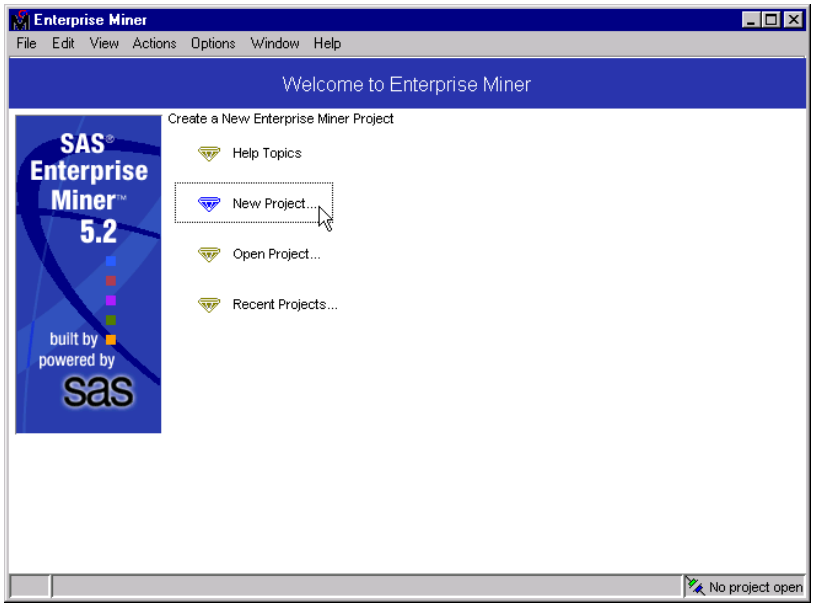
CHAPTER

2 Setting Up Your Project

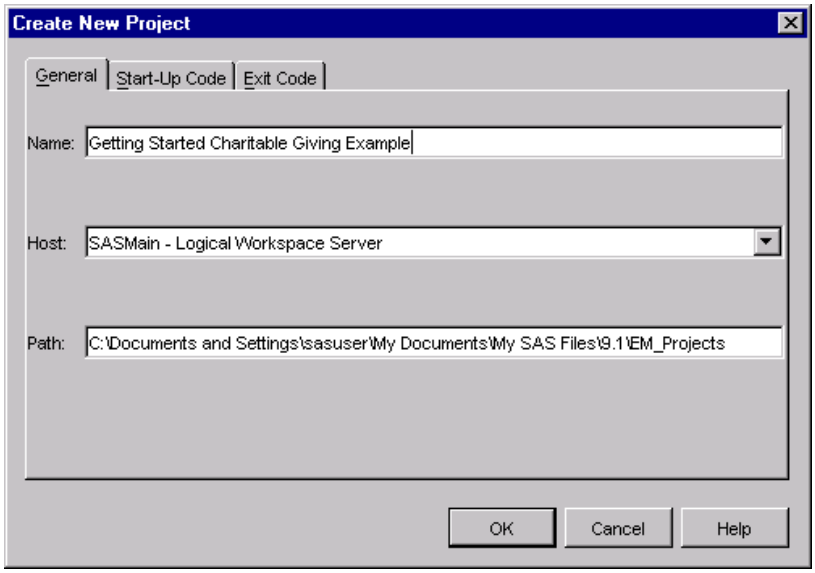
Create a New Project

In Enterprise Miner, you store your work in projects. A project can contain multiple process flow diagrams and information that pertains to them. It is a good idea to create a separate project for each major data mining problem that you want to investigate. This task creates a new project that you will use for this example.

1 To create a new project, click **`New Project`** in the Welcome to Enterprise Miner window.



2 The Create New Project window opens. In the **Name** box, type a name for the project, such as **Getting Started Charitable Giving Example**.



3 In the **Host** box, connect to the main SAS application (or workspace) server, named SASMain by default. Contact your system administrator if you are unsure of your site's configuration.

4 In the **Path** box, type the path to the location on the server where you want to store the data that is associated with the example project. Your project path depends on whether you are running Enterprise Miner as a complete client on your local machine or as a client/server application.

If you are running Enterprise Miner as a complete client, your local machine acts as its own server. Your Enterprise Miner projects are stored on your local machine, in a location that you specify, such as **C:\EMProjects**.

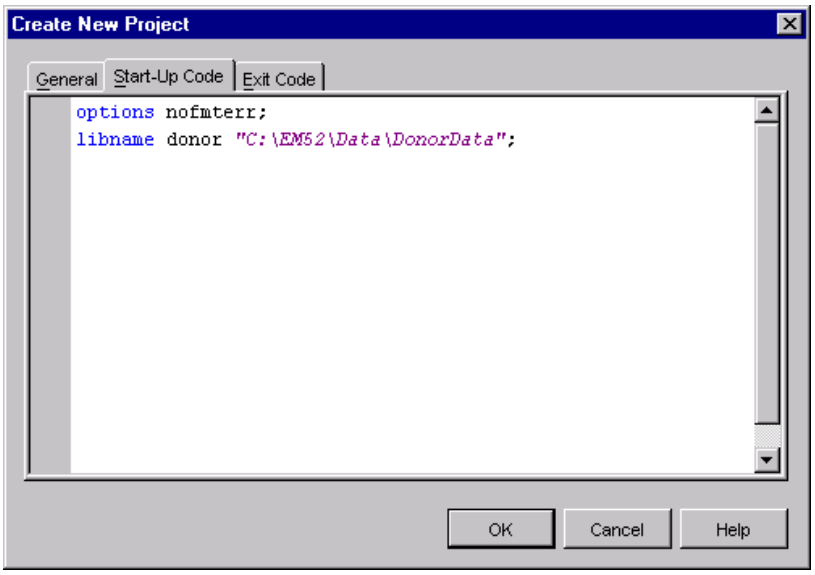
If you are running Enterprise Miner as a client/server application, all projects are stored on the Enterprise Miner server. Ask your system administrator to configure the library location and access permission to the data source for this example.

If you see a default path in the **Path** box, you can accept the default project path, or you can specify your own project path. This example uses **C:\EM52\Projects**.

5 On the **Start-Up Code** tab, you can enter SAS code that you want SAS Enterprise Miner to run each time you open the project. Enter the following statement.

```
options nofmterr;
libname donor ''<path-to-your-example-library>'';
```

Note: You should replace *<path-to-your-example-library>* with the path specification that points to your example data files, either on an Enterprise Miner server, or on your complete client's local machine. The example example uses the local path specification, **C:\EM52\Data\DonorData**. In SAS code, remember to enclose your path specification in double quotation marks. △



Similarly, you can use the **Exit Code** tab to enter SAS code that you want Enterprise Miner to run each time you exit the project. This example does not use the SAS exit code.

6 Click OK. The new project will be created and it opens automatically.

Note: Example results might differ from your results. Enterprise Miner nodes and their statistical methods might incrementally change between releases. Your process flow diagram results might differ slightly from the results that are shown in this example. However, the overall scope of the analysis will be the same. △

Define the Donor Data Source

Overview of the Enterprise Miner Data Source

In order to access the example data in Enterprise Miner, you need to define the imported data as an Enterprise Miner data source. An Enterprise Miner data source stores all of the data set's metadata. Enterprise Miner metadata includes the data set's name, location, library path, as well as variable role assignments measurement levels, and other attributes that guide the data mining process. The metadata is necessary in order to start data mining. Note that Enterprise Miner data sources are not the actual training data, but are the metadata that defines the data source for Enterprise Miner.

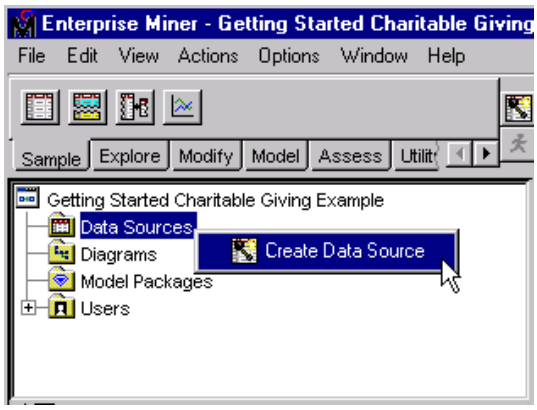
The data source must reside in an allocated library. You assigned the libname Donor to the data that is found in **C:\EM52\Data\DonorData** when you created the start-up code for this example.

The following tasks use the Data Source wizard in order to define the data source that you will use for this example.

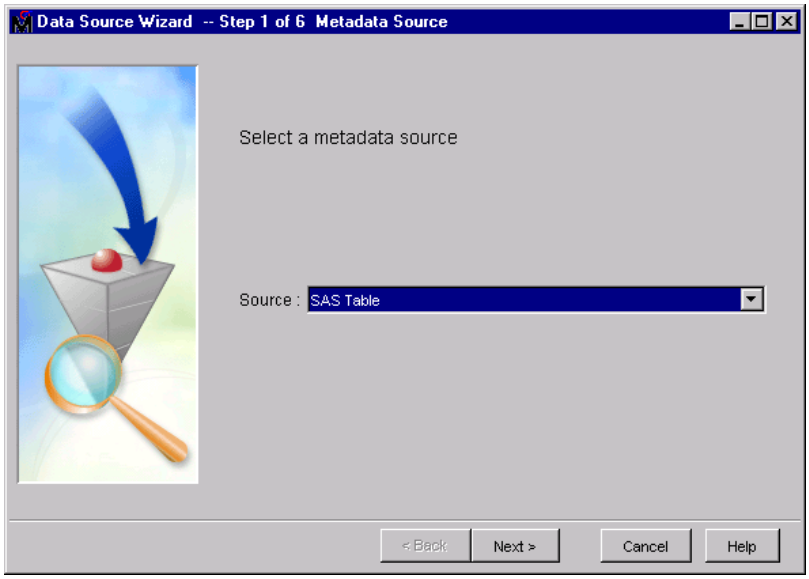
Specify the Data Type

In this task you open the Data Source wizard and identify the type of data that you will use.

1 Right-click the Data Sources folder in the Project Navigator and select **`Create Data Source`** to open the Data Source wizard. Alternatively, you can select **File ▶ New ▶ Data Source** from the main menu, or you can click the Create Data Source on the Shortcut Toolbar.



2 In the **`Source`** box of the Data Source Wizard Metadata Source window, select **`SAS Table`** to tell SAS Enterprise Miner that the data is formatted as a SAS table.

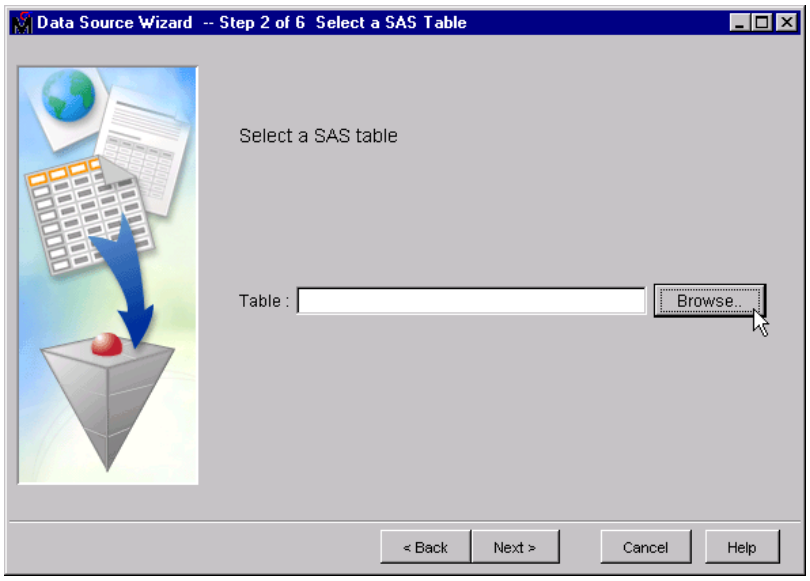


3 Click Next. The Data Source Wizard Select a SAS Table window opens.

Select a SAS Table

In this task, you specify the data set that you will use, and view the table metadata.

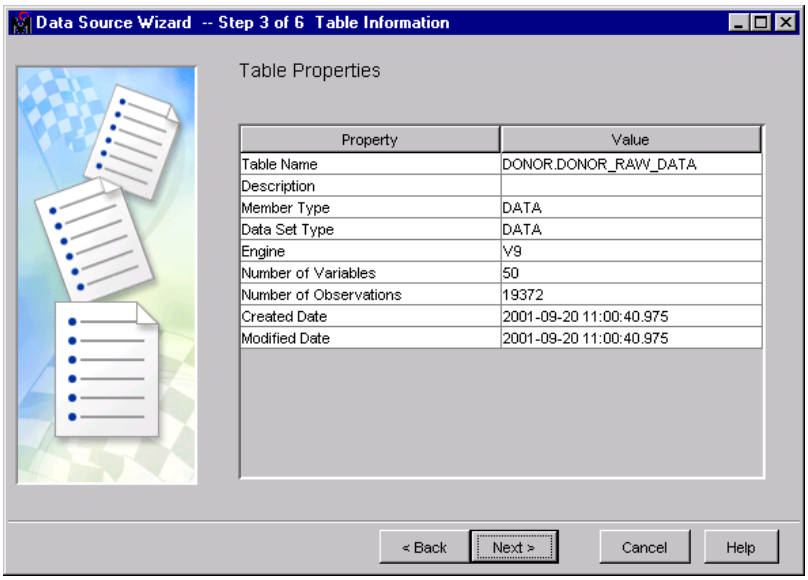
1 Click Browse in the Data Source Wizard – Select a SAS Table window. The Select a SAS Table window opens.



2 Double-click the SAS library named DONOR. It is the library that you or your system administrator assigned in the start-up code. The DONOR library folder expands to show all the data sets that are in the library.

3 Select the **`DONOR_RAW_DATA`** table and click OK. The two-level name **`DONOR.DONOR_RAW_DATA`** appears in the **`Table`** box of the Select a SAS Table window.

4 Click Next. The Table Information window opens. Examine the metadata in the Table Properties section. Notice that the DONOR_RAW_DATA data set has 50 variables and 19,372 observations.



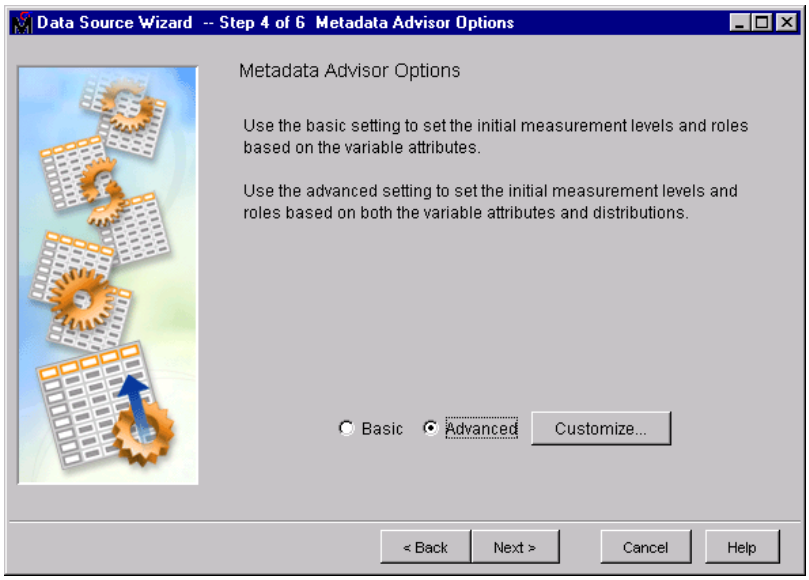
5 After you finish examining the table metadata, click Next. The Data Source Wizard Metadata Advisor Options window opens.

Configure the Metadata

The Metadata Configuration step activates the Metadata Advisor, which you can use to control how Enterprise Miner organizes metadata for the variables in your data source.

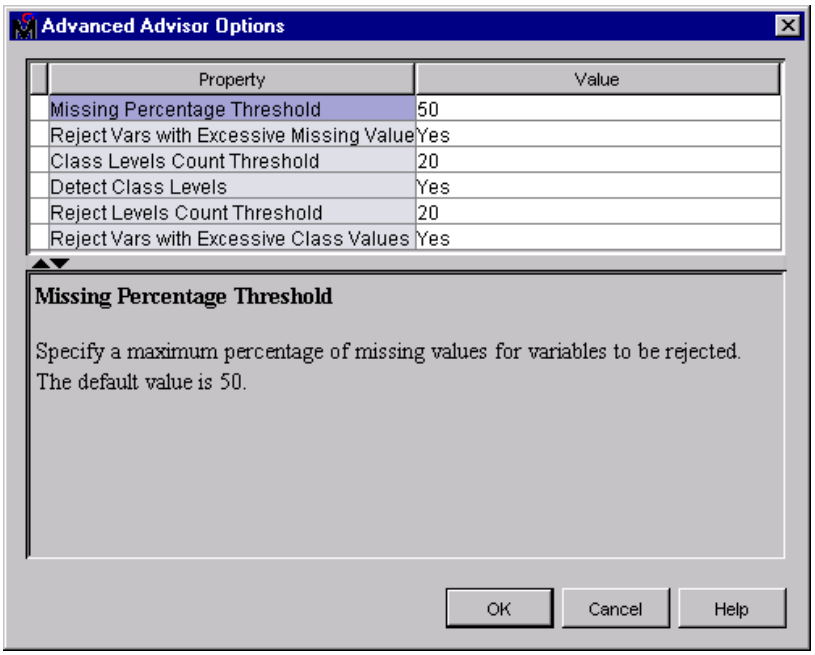
In this task, you generate and examine metadata about the variables in your data set.

1 Select **Advanced** and click Customize.



The Advanced Advisor Options window opens.

In the Advanced Advisor Options window, you can view or set additional metadata properties. When you select a property, the property description appears in the bottom half of the window.

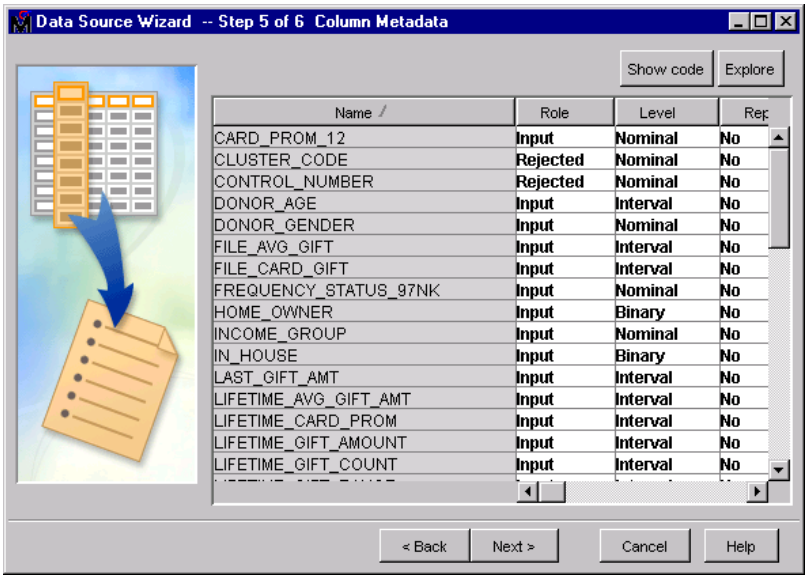


Notice that the threshold value for class variables is 20 levels. You will see the effects of this setting when you view the Column Metadata window in the next step. Click OK to use the defaults for this example.

2 Click Next in the Data Source Wizard Metadata Advisor Options window to generate the metadata for the table. The Data Source Wizard Column Metadata window opens.

Note: In the Column Metadata window, you can view and, if necessary, adjust the metadata that has been defined for the variables in your SAS table. Scroll through the table and examine the metadata. In this window, columns that have a white background are editable, and columns that have a gray background are not editable. △

3 Select the **Names** column header to sort the variables alphabetically.

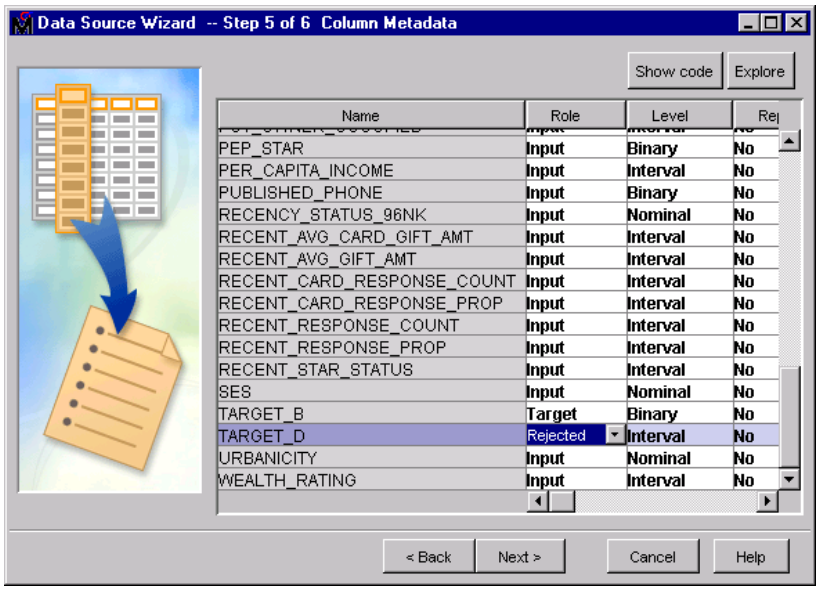


Note that the roles for the variables CLUSTER_CODE and CONTROL_NUMBER are set to **Rejected** because the variables exceed the maximum class count threshold of 20. This is a direct result of the threshold values that were set in the Data Source Wizard Metadata Advisory Options window in the previous step. To see all of the levels of data, select the columns of interest and then click Explore in the upper right-hand corner of the window.

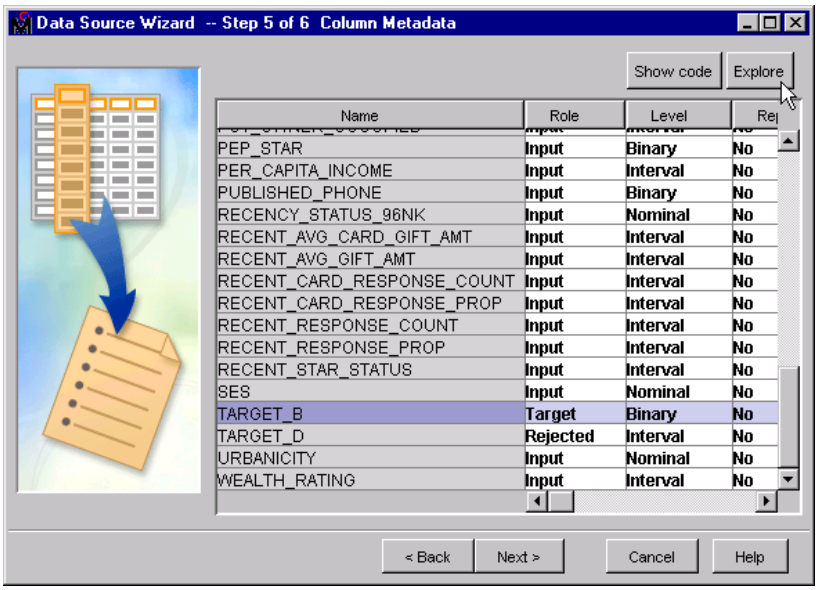
4 Redefine these variable roles and measurement levels:

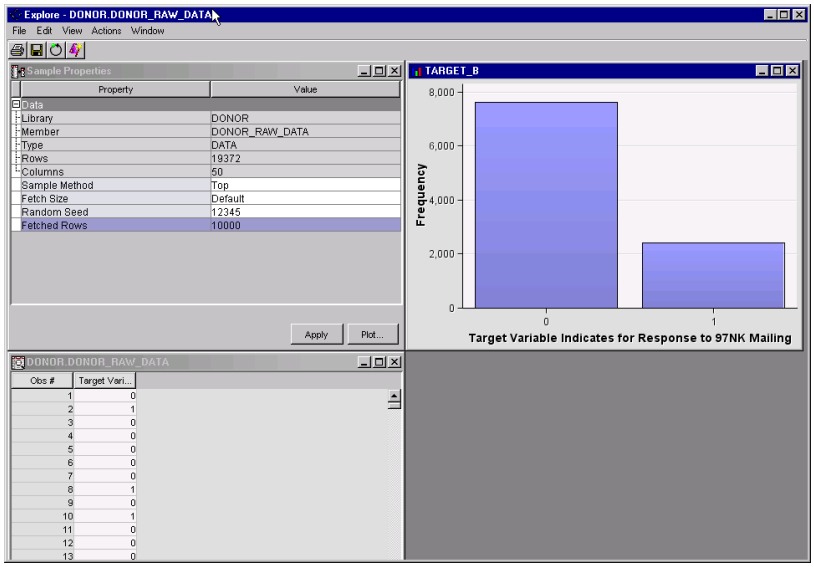
- Set the role for the CONTROL_NUMBER variable to **ID**.
- Set these variables to the **Interval** measurement level:
  - CARD_PROM_12
  - INCOME_GROUP
  - RECENT_CARD_RESPONSE_COUNT
  - RECENT_RESPONSE_COUNT
  - WEALTH_RATING

5 Set the role for the variable TARGET_D to **`Rejected`**, since you will not model this variable. Note that Enterprise Miner correctly identified TARGET_D and TARGET_B as targets since they start with the prefix **`TARGET`**.

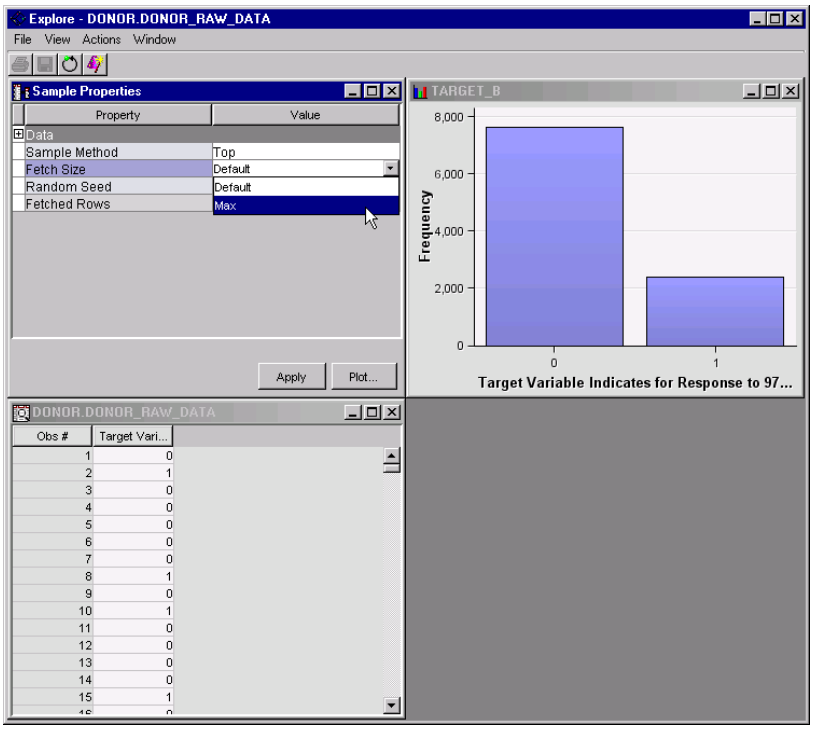


6 Select the TARGET_B variable and click Explore to view the distribution of TARGET_B. As an exercise, select additional variables and explore their distributions.

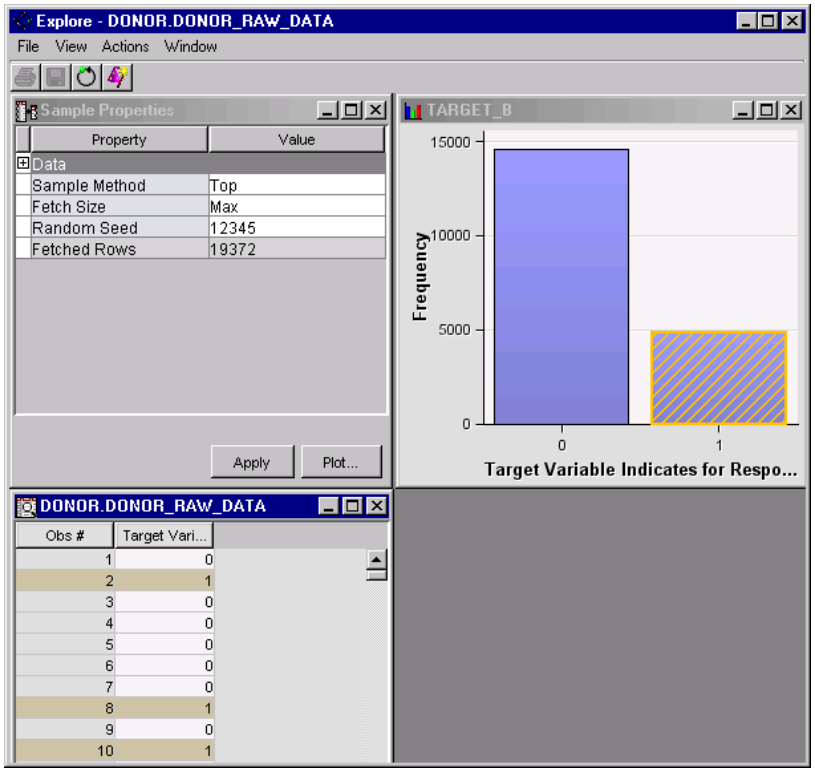




7 In the Sample Properties window, set **`Fetch Size`** to **`Max`** and then click Apply.

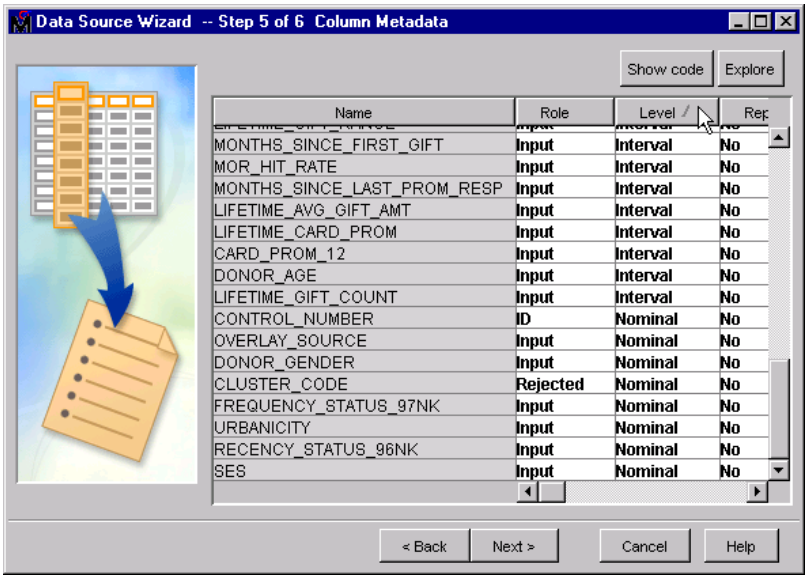


8 Select the bar that corresponds to donors (TARGET_B = '1') on the TARGET_B histogram and note that the donors are highlighted in the DONOR.DONOR_RAW_DATA table.

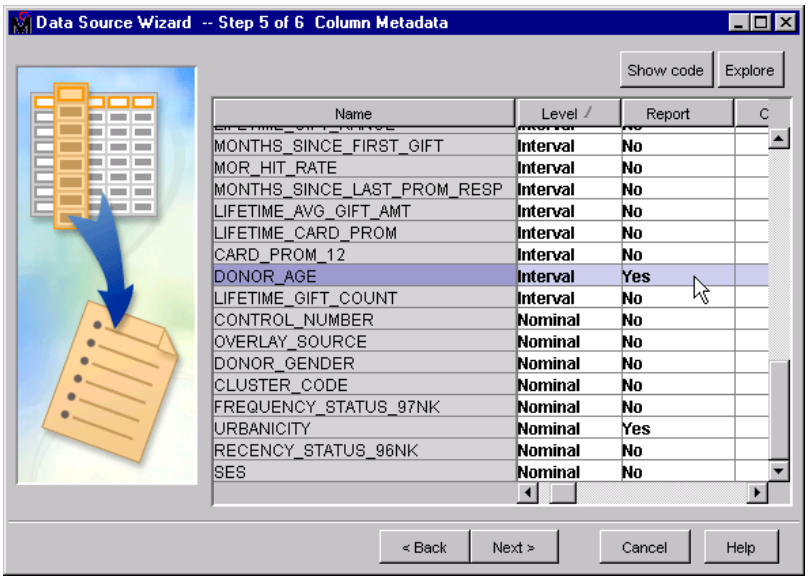


9 Close the Explore window.

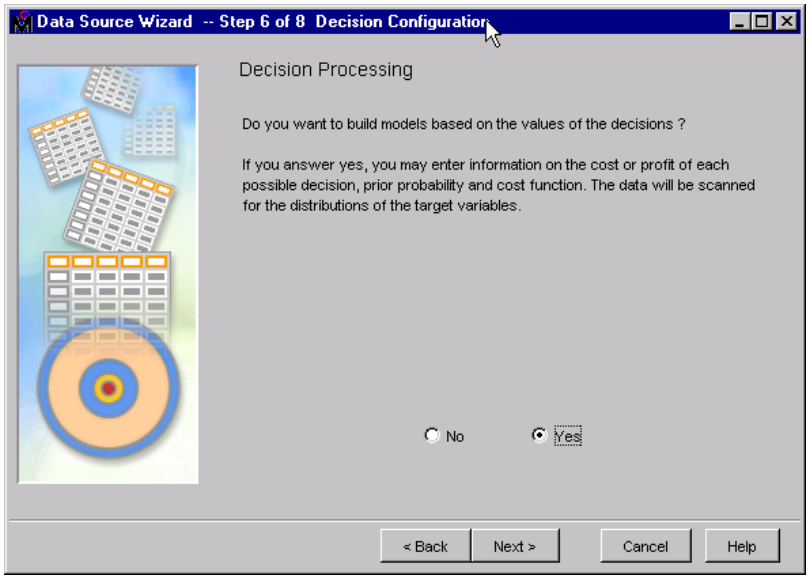
10 Sort the Metadata table by **`Level`** and check your customized metadata assignments.



11 Select the **Report** column and select **Yes** for URBANICITY and DONOR_AGE to define them as report variables. These variables will be used as additional profiling variables in results such as assessment tables and cluster profiles plots.



12 Click Next to open the Data Source Wizard Decision Configuration window.



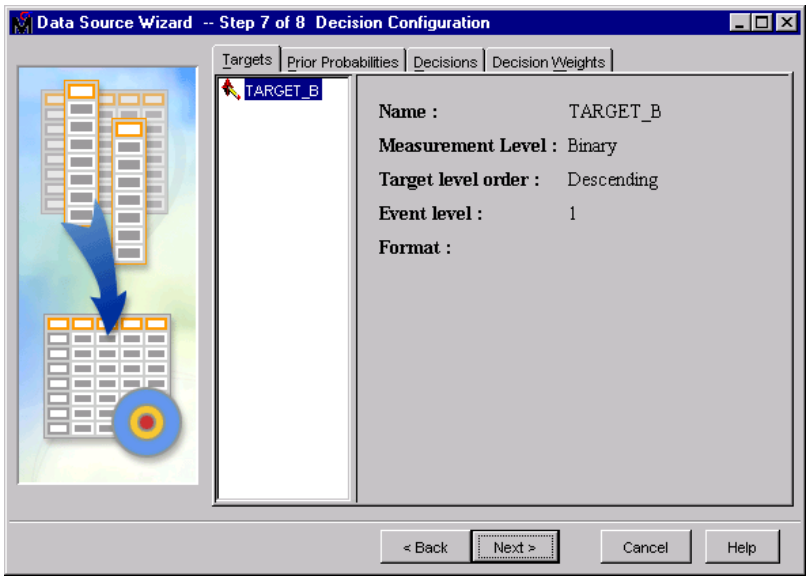
To end this task, select **Yes** and click Next in order to open the Decision Configuration window.

Define Prior Probabilities and a Profit Matrix

The Data Source Wizard Decision Configuration window enables you to define a target profile that produces optimal decisions from a model. You can specify target profile information such as the profit or loss of each possible decision, prior probabilities, and cost functions. In order to create a target profile in the Decision

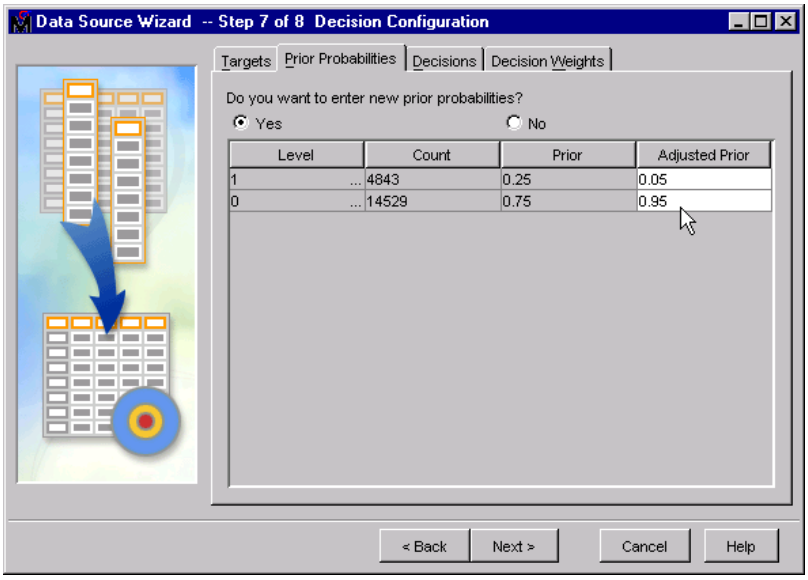
Configuration window, you must have a variable that has a role of Target in your data source. You cannot define decisions for an interval level target variable.

In this task, you specify whether to implement decision processing when you build your models.



1 Select the **Prior Probabilities** tab. Click Yes to reveal the **Adjusted Prior** column and enter the following adjusted probabilities, which are representative of the underlying population of donors.

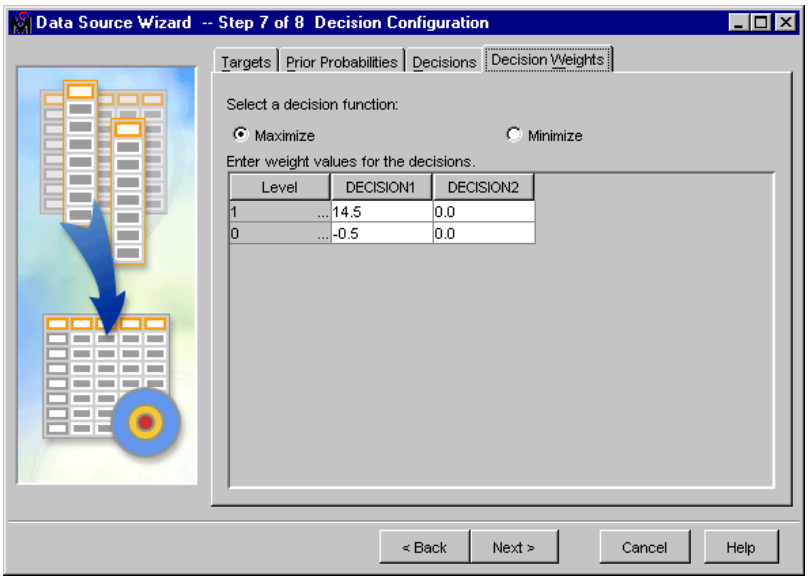
- Level 1 = 0.05
- Level 0 = 0.95



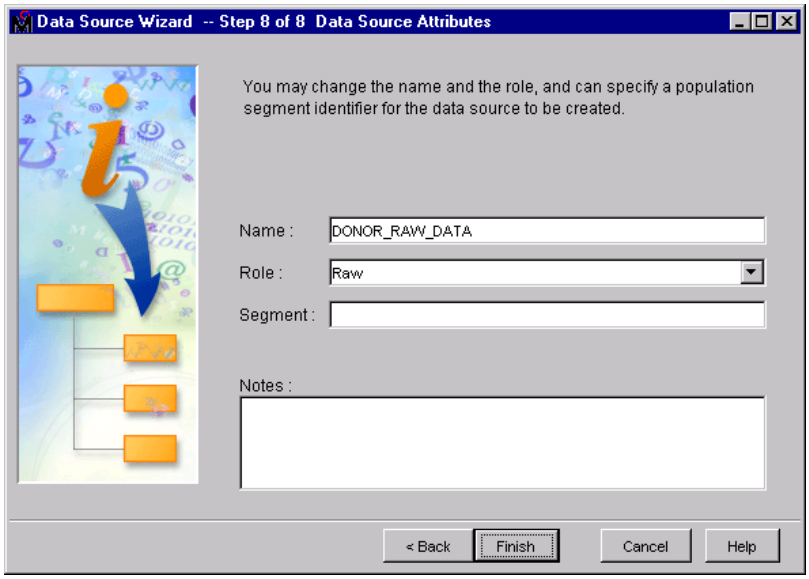
2 Select the **Decision Weights** tab and specify the following weight values:

Level	Decision 1	Decision 2
1	14.5	0
0	-0.5	0

A profit value of $14.50 is obtained after accounting for a 50–cent mailing cost. The focus of this example will be to develop models that maximize profit.



3 Click Next to open the Data Source Attributes window. In this window, you can specify a name, role, and segment for your data source.

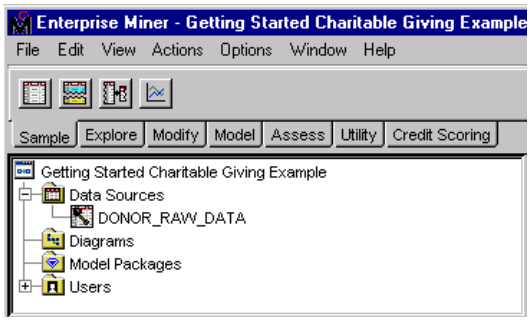


4 Click Finish to add the donor table to the Data Sources folder of the Project Navigator.

Note: You can also define global data sources that can be used across multiple projects. △

Optional Steps

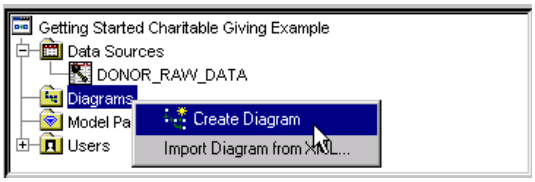
□ The data source can be used in other diagrams. Expand the Data Sources folder. Select the DONOR_RAW_DATA data source and notice that the Property panel now shows properties for this data source.



Create a Diagram

Now that you have created a project and defined your data source, you are ready to begin building your process flow diagram. This task creates a new process flow diagram in your project.

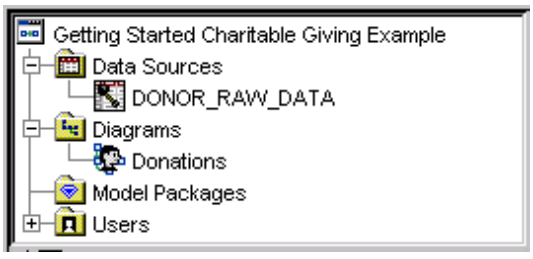
1 Right-click the Diagrams folder of the Project Navigator and select **`Create Diagram`**.



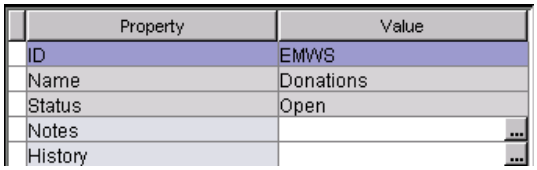
Alternatively, you can select **File ▶ New Diagram** from the main menu, or you can click Create Diagram in the toolbar. The Input window opens.

2 Enter **`Donations`** in the **`Diagram Name`** box and click OK. The empty Donations diagram opens in the Diagram Workspace area.

3 Expand the Diagrams folder to see the newly created Donations diagram.



4 Click the diagram icon next to your newly created diagram and notice that the Properties panel now shows properties for the diagram.



Property	Value
ID	EMWS
Name	Donations
Status	Open
Notes	...
History	...

Other Useful Tasks and Tips

- Explore the node tools that are organized by the SEMMA process on the toolbar. When you move your mouse pointer over a toolbar icon, a tooltip displays the name of each node tool.
- Explore the Toolbar Shortcut buttons that are located to the right of the node tool icons.
- Note that the Properties panel displays the properties that are associated with the project that you just created.
- From the main menu, select **Help ► Contents** or, alternatively, press the F1 key. Browse the Help topics.
- To specify model results package options or to customize the appearance of your Enterprise Miner GUI, select **Options ► Preferences** from the main menu.
- You can also use the **`View`** menu items to open the Program Editor, Log, Output, and Graph windows as well as to open SAS tables and to set the Properties panel to show the advanced options.

CHAPTER

3

Working with Nodes That Sample, Explore, and Modify

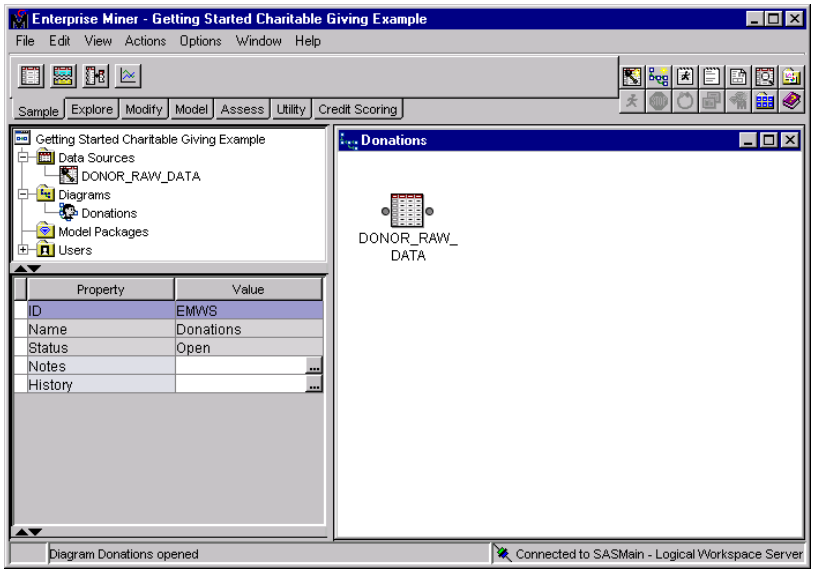
Overview of This Group of Tasks

These tasks develop the process flow diagram that you created in "Create a Diagram" on page 35. The Input Data node is typically the first node that you use when you create a process flow diagram. The node represents the data source that you choose for your data mining analysis and provides metadata about the variables. The other nodes that you use in this chapter show you some typical techniques of exploring and modifying your data.

Identify Input Data

In this task, you add an Input Data node to your process flow diagram.

1. Select the DONOR_RAW_DATA data source from the Data Sources list in the Project panel.
2. Select the **`Sample`** tab and drag the DONOR_RAW_DATA data source into the Diagram Workspace.



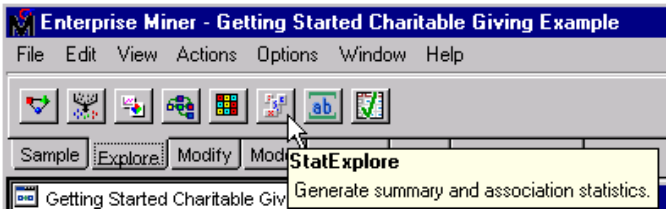
Note: Although this task develops one process flow diagram, Enterprise Miner enables you to open multiple diagrams at one time. You can also disconnect from and reconnect to a diagram if you have also configured the Enterprise Miner application server. Other users can also access the same project. However, only one user can open a diagram at a time. △

Generate Descriptive Statistics

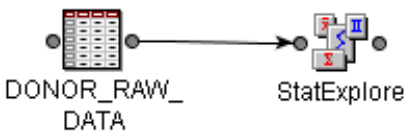
As you begin a project, you should consider creating summary statistics for each of the variables, including their relationship with the target, using tools like the StatExplore node.

In this task, you add a StatExplore node to your diagram.

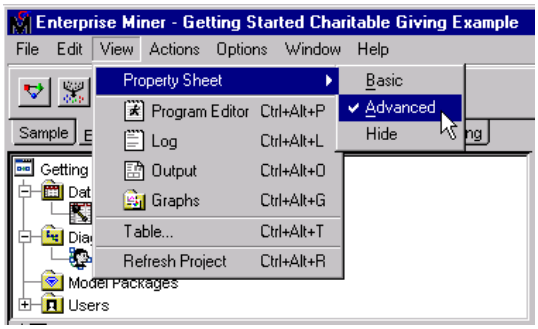
1 Select the **Explore** tab on the toolbar at the top left and select the StatExplore node. Drag this node into the Diagram Workspace. Alternatively, you can also right-click the Diagram Workspace and use the pop-up menus to add nodes to the workspace.



2 Connect the DONOR_RAW_DATA Data Source node to the StatExplore node.



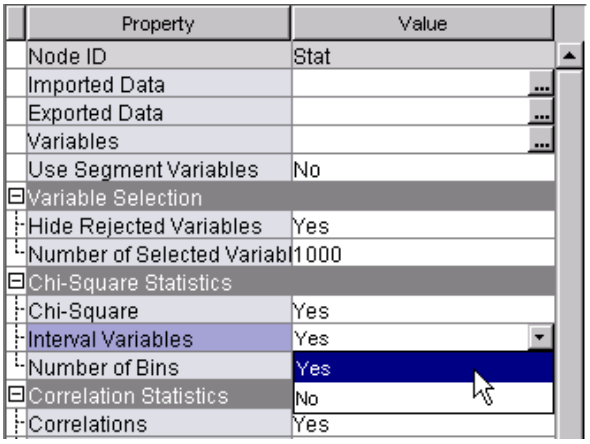
3 Select **View** ▶ **Property Sheet** ▶ **Advanced**.



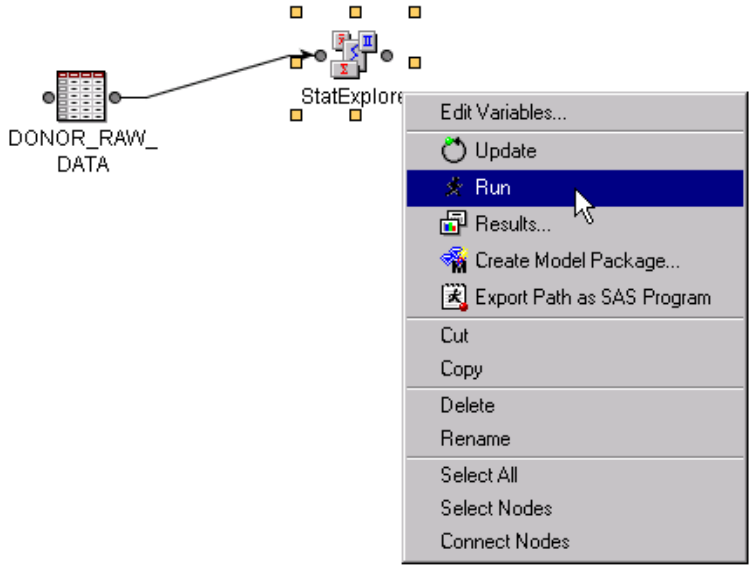
4 Select the StatExplore node to view its properties. Details about the node appear in the Properties panel. By default, the StatExplore node creates Chi-Square statistics and correlation statistics.

Note: An alternate way to see all of the properties for a node is to double-click the node in the toolbar at the top left of the application. △

5 To create Chi-Square statistics for the binned interval variables in addition to the class variables, set the **`Interval Variables`** property to **`Yes`**.



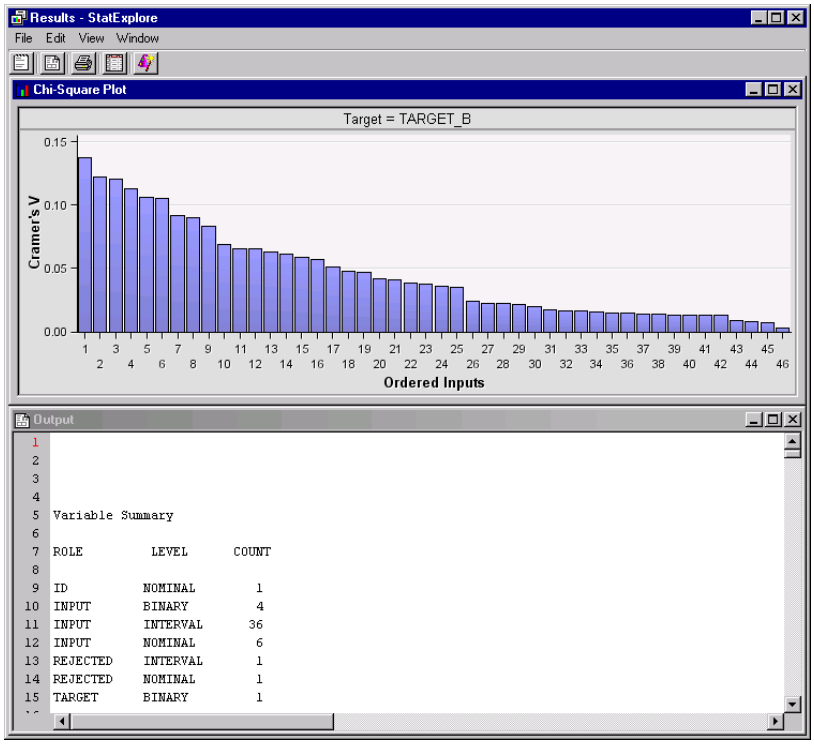
6 Right-click the StatExplore node and select **Run**. A Confirmation window appears. Click Yes. A green border appears around each successive node in the diagram as Enterprise Miner runs the path to the StatExplore node.



Note: An alternate way to run a node is to select the **Run** icon on the upper right-hand side of the toolbar to run the path from the Input Data node to the selected node on the diagram.

If there are any errors in the path that you ran, the border around the node that contains the error will be red rather than green, and an Error window will appear. The Error window tells you that the run has failed and provides information about what is wrong. △

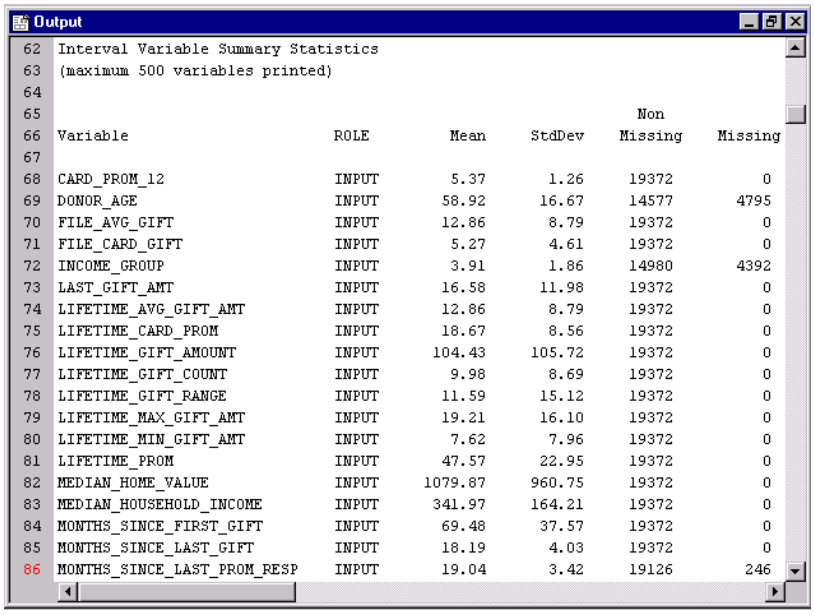
7 A Run Status window opens when the path has run. Click Results. The Results window opens.



The Chi-Square plot highlights inputs that are associated with the target. Many of the binned continuous inputs have the largest Cramer's V values. The Pearson's correlation coefficients are displayed if the target is a continuous variable.

Note: An alternate way to view results is to select the **Results** icon on the upper right-hand side of the toolbar. △

8 Maximize the Output window. The Output window provides distribution and summary statistics for the class and interval inputs, including summaries that are relative to the target.



9 Scroll down to the **Interval Variables Summary Statistics** section. The **Non-Missing** column lists the number of observations that have valid values for each interval variable. The **Missing** column lists the number of observations that have missing values for each interval variable.

Several variables such as DONOR_AGE, INCOME_GROUP, WEALTH_RATING, and MONTHS_SINCE_LAST_PROM_RESP have missing values. The entire customer case is excluded from a regression or neural network analysis when a variable attribute about a customer is missing. Later, you will impute some of these variables using the Replacement node.

Notice that many variables have very large standard deviations. You should plot these variables in order to decide whether transformations are warranted.

10 Close the Results window.

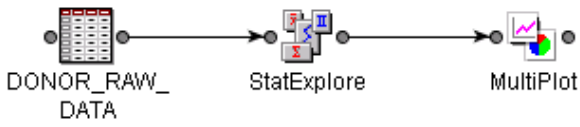
Note: If you make changes to any of the nodes in your process flow diagram after you have run a path, you need to rerun the path in order for the changes to affect later nodes. △

Create Exploratory Plots

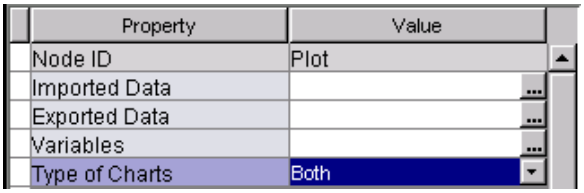
Enterprise Miner enables you to generate numerous data visualization graphics in order to reveal extreme values in the data and to discover patterns and trends. You use the MultiPlot node to visualize your data from a wide range of perspectives. With MultiPlot you can graphically explore large volumes of data, observe data distributions, and examine relationships among the variables. The MultiPlot node uses all of the observations for plotting.

In this task, you add a MultiPlot node to your diagram.

1 Select the **Explore** tab from the node toolbar and drag a MultiPlot node into the Diagram Workspace. Connect the StatExplore node to the MultiPlot node.



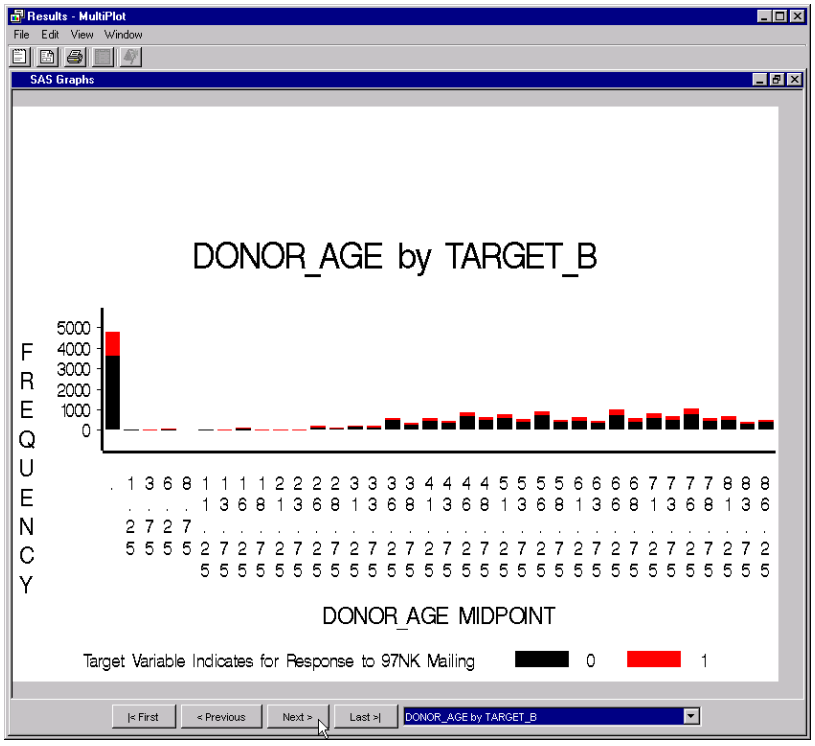
2 Select the MultiPlot node in the Diagram Workspace. In the Properties panel, set the **Type of Charts** property to **Both** in order to generate both scatter and bar charts.



3 In the Diagram Workspace, right-click the MultiPlot node, and select **Run**.

4 After the run is complete, select **Results** from the Run Status window.

5 In the Results window, maximize the SAS Graphs window.



Click First, Previous, or Next at the bottom of the window to scroll through the graphs. You can also view a specific graph by selecting the variable on the selection box to the right of Last.

You will notice several results in the graphs.

- One value for the variable DONOR_GENDER is incorrectly recorded as an **A**.
- There are several heavily skewed variables, such as FILE_AVG_GIFT, LAST_GIFT_AMT, LIFETIME_AVG_GIFT_AMT, LIFETIME_GIFT_AMOUNT, MOR_HIT_RATE, PCT_ATTRIBUTE1, and PCT_OWNER_OCCUPIED. You might want to consider a log transformation later.
- Increasing values of LIFTIME_CARD_PROM, RECENT_RESPONSE_PROP, LIFETIME_GIFT_AMOUNT, LIFETIME_GIFT_COUNT , MEDIAN_HOME_VALUE, MEDIAN_HOUSEHOLD_INCOME, PER_CAPITA_INCOME, and RECENT_STAR_STATUS tend to be more associated with donors and are also heavily skewed. You might want to consider a bucket transformation that will be relative to the relationship with target.
- Other variables, such as MONTHS_SINCE_LAST_PROM_RESP and NUMBER_PROM_12, show some good separation of the target values at both tails of the distribution.

6 Close the Results window.

Partition the Raw Data

In data mining, one strategy for assessing model generalization is to partition the data source. A portion of the data, called the training data, is used for preliminary

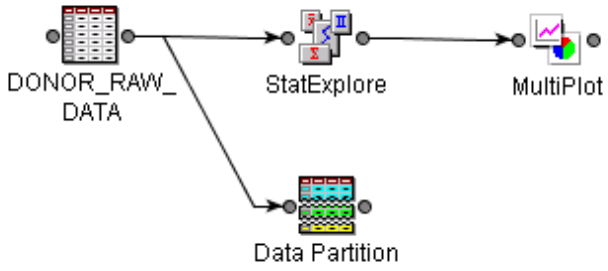
model fitting. The rest is reserved for empirical validation. The hold-out sample itself is often split into two parts: validation data and test data. The validation data is used to prevent a modeling node from over-fitting the training data (model fine-tuning), and to compare prediction models. The test data set is used for a final assessment of the chosen model.

Enterprise Miner can partition your data in several ways. Choose one of the following methods.

- By default, Enterprise Miner uses either simple random sampling or stratified sampling, depending on your target. If your target is a class variable, then SAS Enterprise Miner stratifies the sample on the class target. Otherwise, simple random sampling is used.
- If you specify simple random sampling, every observation in the data set has the same probability of being included in the sample.
- If you specify simple cluster sampling, SAS Enterprise Miner samples from a cluster of observations that are similar in some way.
- If you specify stratified sampling, you identify variables in your data set to form strata of the total population. SAS Enterprise Miner samples from each stratum so that the strata proportions of the total population are preserved in each sample.

In this task, you use the Data Partition node to partition your data.

1. Select the **Sample** tab from the node toolbar at the top left of the application. Drag a Data Partition node from the toolbar into the Diagram Workspace.
2. Connect the DONOR_RAW_DATA Data Source node to the Data Partition node.



3. Select the Data Partition node in the Diagram Workspace. Details about data partitioning appear in the Properties panel.

   Note: If the target variable is a class variable, the default partitioning method that Enterprise Miner uses is stratification. Otherwise, the default partitioning method is simple random. △

4. In the Properties panel under the Data Set Percentages section, set the following values:
   - set **Training** to **55**
   - set **Validation** to **45**
   - set **Test** to **0**

Property	Value
Node ID	Part
Imported Data	
Exported Data	
Variables	
Partitioning Method	Default
Random Seed	12345
Data Set Percentages	
Training	55.0
Validation	45.0
Test	0.0

In the Data Set Percentages section of the Properties panel, the values for the **`Training`**, **`Validation`**, and **`Test`** properties specify how you want to proportionally allocate the original data set into the three partitions. You can allocate the percentages for each partition by using any real number between 0 and 100, as long as the sum of the three partitions equals 100.

Note: By default, the Data Partition node partitions the data by stratifying on the target variable. This is a good idea in this case, because there are few donors relative to non-donors. △

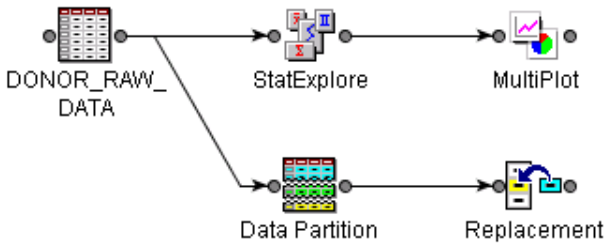
5 Run the Data Partition node.

Replace Missing Data

You use the Replacement node to generate score code to process unknown variable levels when you are scoring data, and to interactively specify replacement values for class levels.

In this task, you add and configure a Replacement node in your process flow diagram.

1 From the **`Modify`** tab of the node toolbar, drag a Replacement node into the Diagram Workspace and connect it to the Data Partition node.

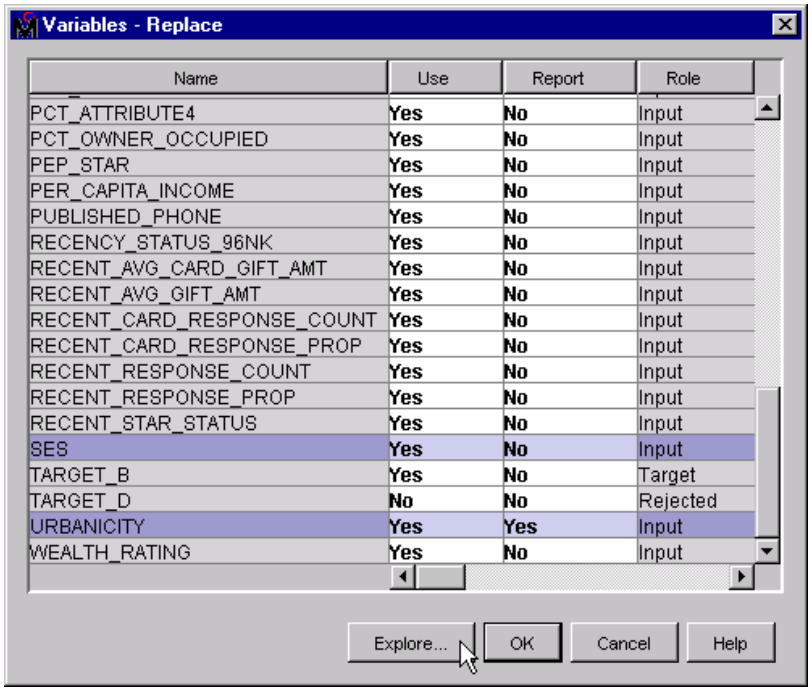


2 Select the Replacement node. On the Properties panel, select the ellipsis button to the right of the **`Variables`** property to explore any of the variables in the input data set. The Variables - Replace window opens.

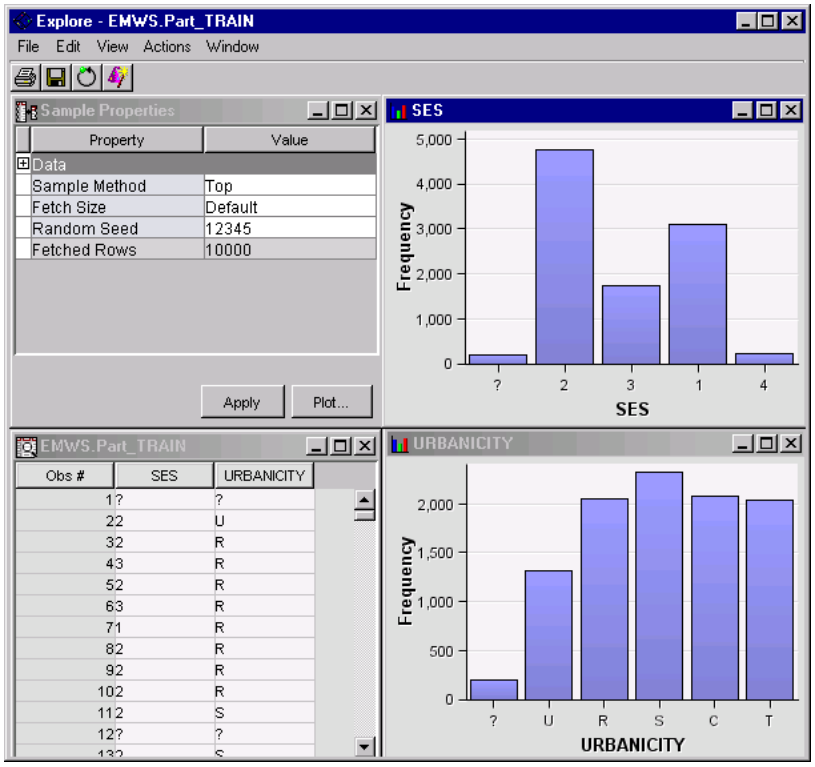
Property	Value
Node ID	Replace
Imported Data	
Exported Data	
Variables	
Replacement Editor	
Unknown Levels	Ignore
Hide	No

3 In the Variables - Replace window, select the variables SES and URBANICITY, and then click Explore. The Explore window opens.

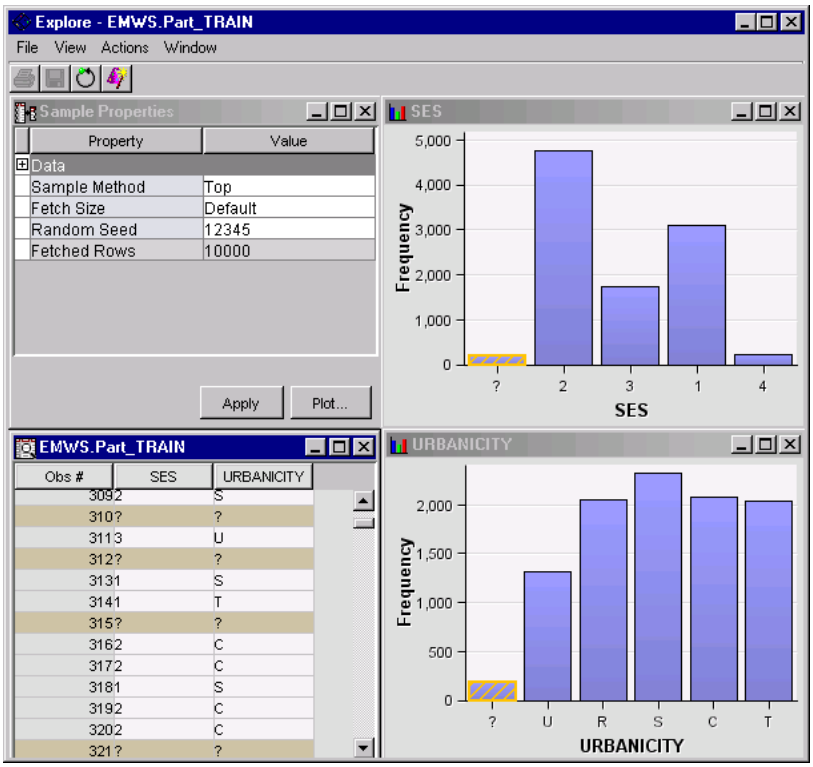
Note: If Explore is dimmed and unavailable, right-click the Data Partition node and select **Run**. △



4 In the Explore window, notice that both the SES and URBANICITY variables contain observations that have missing values. The observations are represented by question marks. Later, you will use the Impute node to replace the missing values with imputed values that have more predictive power.



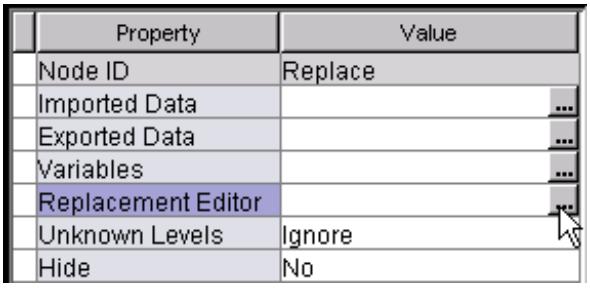
5 Select the bar that corresponds to missing values (SES = "?") in the SES histogram. Notice that when observations display missing values for the variable SES, the observations also display missing values for the variable URBANICITY. The graphs interact with one another.



6 Close the Explore window.

7 Click OK to close the Variables window.

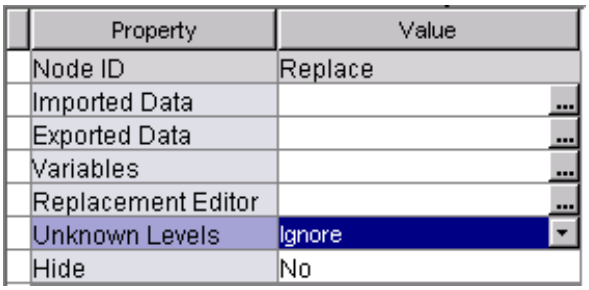
8 In the Replacement node Properties panel, select the ellipsis button to the right of the **`Replacement Editor`** property.



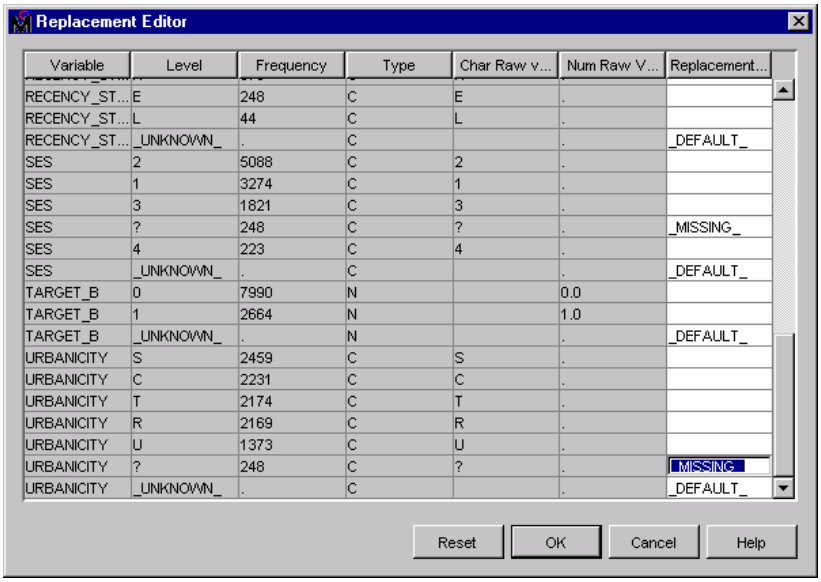
9 When you are prompted to create a replacement data table, click Yes.

The Replacement Editor window opens.

Note: By default, Enterprise Miner replaces unknown levels using the **`Unknown Levels`** property in the Properties panel. The choices are Ignore, Missing and Mode (the most frequent value). Ensure that the **`Unknown Level`** property is set to **`Ignore`**. △

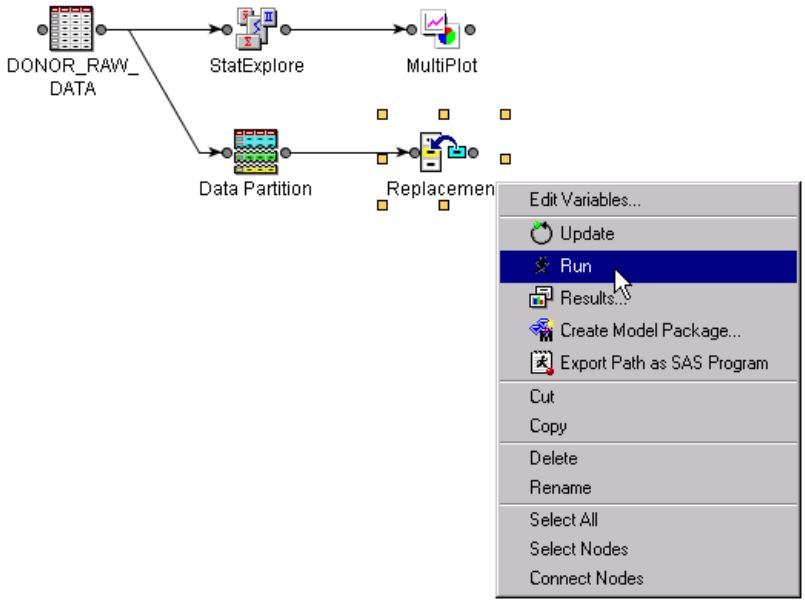


10 Scroll through the data table in the Replacement Editor window. Observe the values for the variable levels of SES and URBANICITY. When one of these variable levels displays a question mark (?) in the **`Char Raw value`** column, enter **`_MISSING_`** in the **`Replacement Value`** column for that row.



11 Click OK.

12 Right-click the Replacement node and select **Run**.



CHAPTER

4

Working with Nodes That Model

Overview of This Group of Tasks

These tasks introduce you to the Decision Tree node and to the Tree Desktop Application. You use the Decision Tree node to model the data. You use the Tree Desktop Application to explore and evaluate the decision tree as you develop the tree. You also learn to perform typical interactive tasks.

Basic Decision Tree Terms and Results

An *empirical tree* is a segmentation of the data. Enterprise Miner creates an empirical tree by applying a series of simple rules that you specify. Each rule assigns an observation to a segment, based on the value of one input. One rule is applied after another, resulting in a hierarchy of segments within segments. The hierarchy is called a *tree*, and each segment is called a *node*. The original segment contains the entire data set and is called the *root node* of the tree. A node and all its successors form a *branch* of the node that created it. The final nodes are called *leaves*. For each leaf, a decision is made and applied to all observations in the leaf. The type of decision depends on the context of the data mining problem. In this example, the decision is simply the predicted value. The path from the root to the target leaf is the rule that classifies the target.

Tree models readily accommodate nonlinear associations between the input variables and the target. They offer easy interpretability, accept different data types, and handle missing values without using imputation.

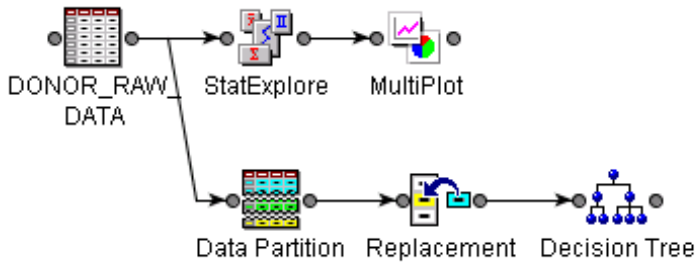
In Enterprise Miner, you use the plots and tables of the Results window to assess how well the tree model fits the training and validation data. You can benchmark the accuracy, profitability, and stability of your model. The Decision Tree node displays the following results:

- A standard Cumulative Lift Chart for the training and validation data. This chart provides not only lift values but also provides a quick check as to whether the tree model is reliable. If the tree is unreliable, then none of the numbers or splits is valuable. Trees can be unreliable when they are applied to new data, so you should always evaluate the tree using both the validation and test data.
- A Leaf Statistics bar chart in which the height of each bar equals the percentage of donors in the leaf for both the training and validation data. The order of the bars is based on the percentage of donors (1's) in the training data. Use the scroll bar at the top to show additional leaves. You should also look for consistency in each leaf with regards to the training and validation data.
- The Tree Diagram (in a window labeled Tree) that shows how the data splits into subgroups.
- Fit Statistics for both the training and validation data.
- The Tree Map represents a compact graphical representation of the tree. The nodes have the same top-to-bottom, left-to-right relationship as the traditional tree diagram. The width of a node is proportional to the number of training cases in the node. Colored rectangles represent individual nodes of the tree: larger rectangles represent the nodes that contain more cases. The nodes are colored according to the values of a statistic. By default, a node's color reflects the percentage of the target event in the node. For categorical targets, color represents the proportion of the target value in the training data set that is assigned to this node. For an interval target, color represents the average target value in the training data that is assigned to this node.
- The Output window contains information such as variable importance, tree leaf report, fit statistics, and a classification matrix.

Create a Decision Tree

In this task, you use the Decision Tree node to build a decision tree using your partitioned data.

1. Drag the Decision Tree icon from the **Model** tab of the toolbar into the Diagram Workspace.
2. Connect the Replacement node to the Decision Tree node.



3 Select the Decision Tree node in the Diagram Workspace. The Properties panel indicates how each Decision Tree node property is configured.

4 Set the Decision Tree node properties as follows:

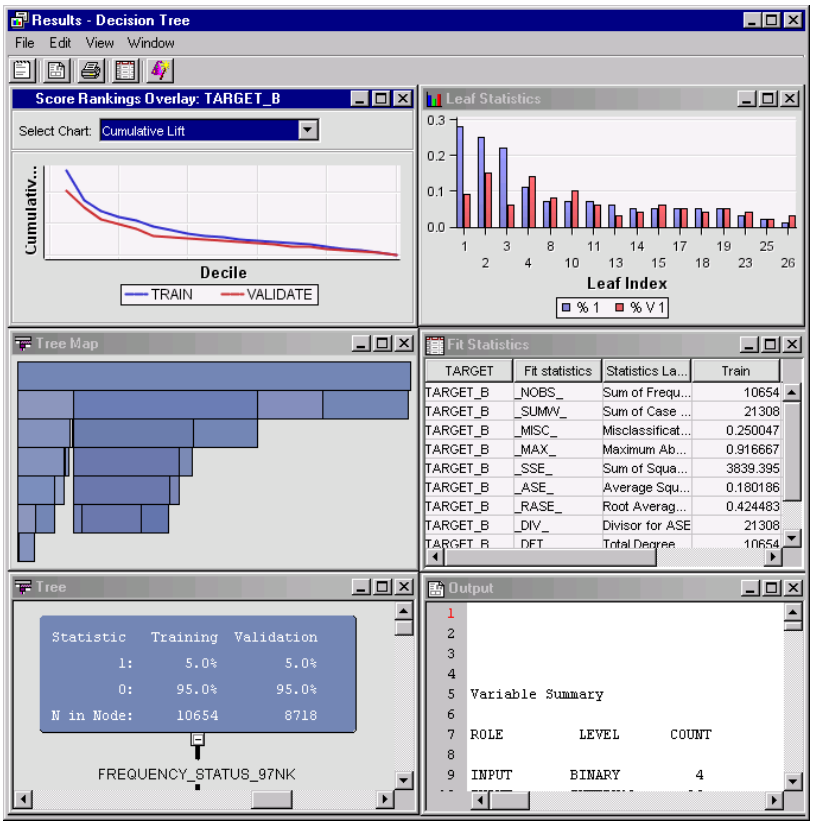
- Set the **Maximum Branch** to **4** in order to allow the Tree node to create up to four-way rules. The Decision Tree node creates binary splits by default.
- Set the **Leaf Size** to **8** in order to ensure that each leaf contains at least 8 observations.
- Set the **Maximum Depth** to **10** in order to potentially grow a bushier tree.
- Set the **Number of Surrogate Rules** to **4** in order to handle missing values in the data.
- Keep the Splitting Rule Criterion property in its Default (Chi-Square) setting.

Note: The **Assessment Measure** property is automatically set to **Decision** by default because you have defined a profit matrix. The Decision Tree node will choose the tree that maximizes profit in the validation data. △

Property	Value
Node ID	Tree
Imported Data	
Exported Data	
Variables	
Interactive	
Splitting Rule	
Criterion	Default
Significance Level	0.2
Missing Values	Use in search
Use Input Once	No
Maximum Branch	4
Maximum Depth	10
Minimum Categorical Size	5
Node	
Leaf Size	8
Number of Rules	5
Number of Surrogate Rule	4
Split Size	
Split Search	
Exhaustive	5000
Node Sample	5000
Subtree	
Method	Assessment
Number of Leaves	1
Assessment Measure	Decision

5 Right-click the Decision Tree node in the Diagram Workspace and select **Run**.

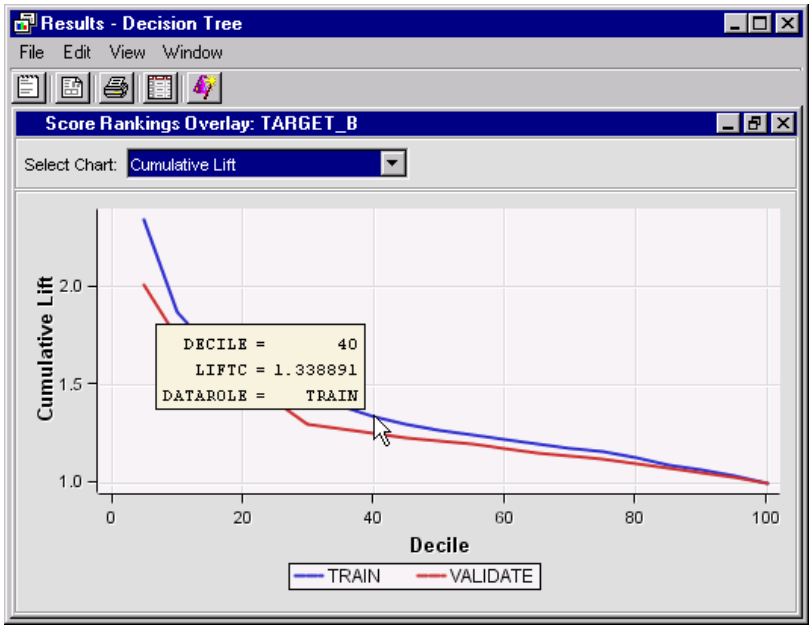
6 A Run Status window appears when the Decision Tree run has been completed. Click Results. The Results window opens.



The Score Rankings Overlay: TARGET_B chart shows that a consistent trend is achieved on both the training and validation data. Although the lift values decrease quickly, the tree does seem to be stable.

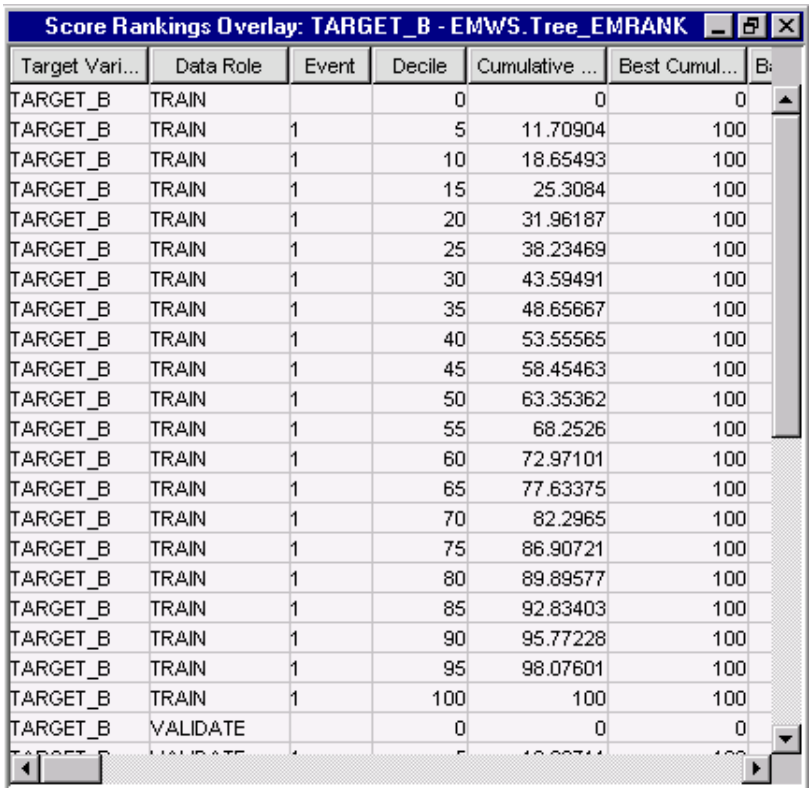
The Fit Statistics table shows that the average profit of the training and validation data is about 27.7 and 25.4 cents, respectively.

7 Move your mouse over the different points of either the training or validation line in order to reveal the various lift values on the Score Rankings Overlay: TARGET_B chart.



8 Select the Cumulative Lift chart and then click Table at the top left of the Results window in order to display a table of the lift values,

Note: You can highlight rows in the table and then use **Copy** to paste the contents to another application such as Microsoft Word or Excel. You can also copy graphs the same way. This feature is common to most of the Enterprise Miner tools. △



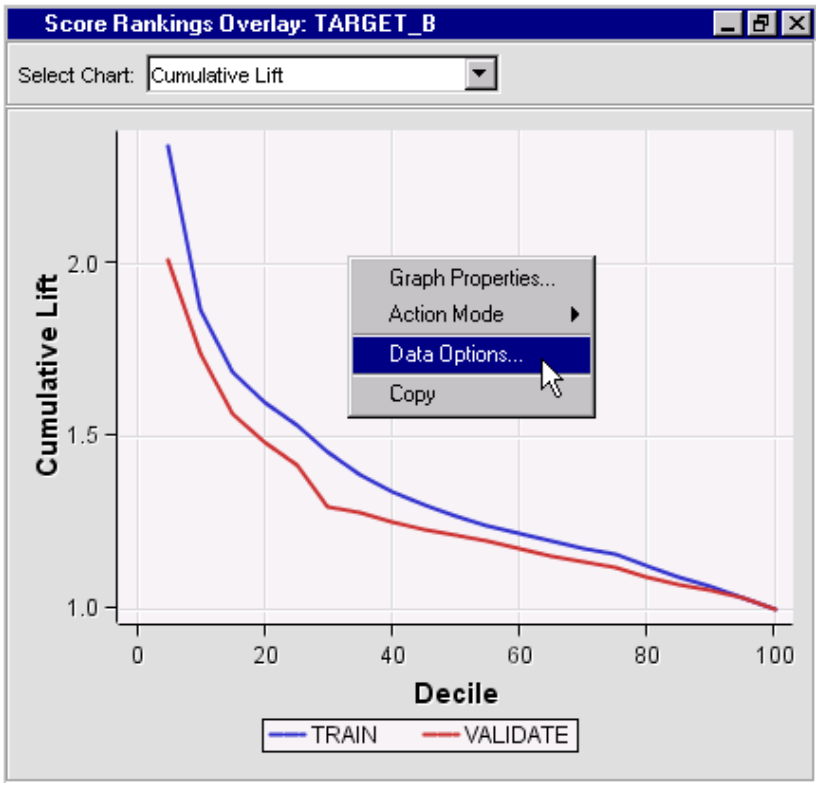
Score Rankings Overlay: TARGET_B - EMWS.Tree_EMRANK

Target Vari...	Data Role	Event	Decile	Cumulative ...	Best Cumul...	B
TARGET_B	TRAIN		0	0	0	
TARGET_B	TRAIN	1	5	11.70904	100	
TARGET_B	TRAIN	1	10	18.65493	100	
TARGET_B	TRAIN	1	15	25.3084	100	
TARGET_B	TRAIN	1	20	31.96187	100	
TARGET_B	TRAIN	1	25	38.23469	100	
TARGET_B	TRAIN	1	30	43.59491	100	
TARGET_B	TRAIN	1	35	48.65667	100	
TARGET_B	TRAIN	1	40	53.55565	100	
TARGET_B	TRAIN	1	45	58.45463	100	
TARGET_B	TRAIN	1	50	63.35362	100	
TARGET_B	TRAIN	1	55	68.2526	100	
TARGET_B	TRAIN	1	60	72.97101	100	
TARGET_B	TRAIN	1	65	77.63375	100	
TARGET_B	TRAIN	1	70	82.2965	100	
TARGET_B	TRAIN	1	75	86.90721	100	
TARGET_B	TRAIN	1	80	89.89577	100	
TARGET_B	TRAIN	1	85	92.83403	100	
TARGET_B	TRAIN	1	90	95.77228	100	
TARGET_B	TRAIN	1	95	98.07601	100	
TARGET_B	TRAIN	1	100	100	100	
TARGET_B	VALIDATE		0	0	0	

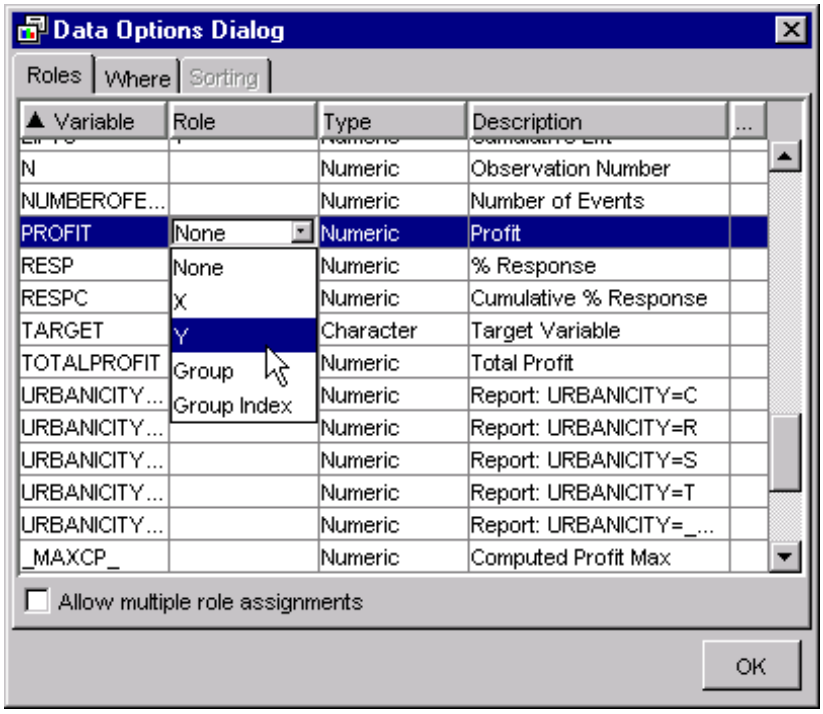
9 Close the Score Rankings Overlay table.

10 Because you defined a profit matrix for the Donor data, you should base your evaluation of the model on profit. To display profit, rather than lift, on the Score Rankings plot, follow these steps:

   a Maximize the Score Rankings Overlay: TARGET_B chart.
   b Right-click the background of the plot and select **`Data Options`**.

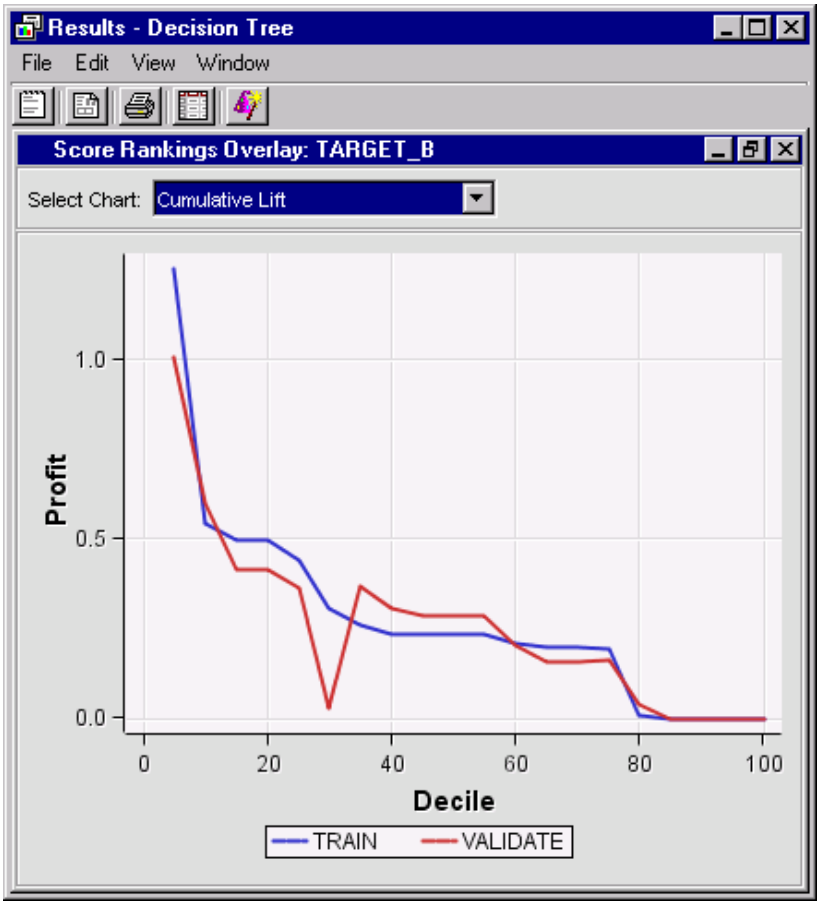


c Scroll down the list of variables and set the Role for PROFIT to **Y**.

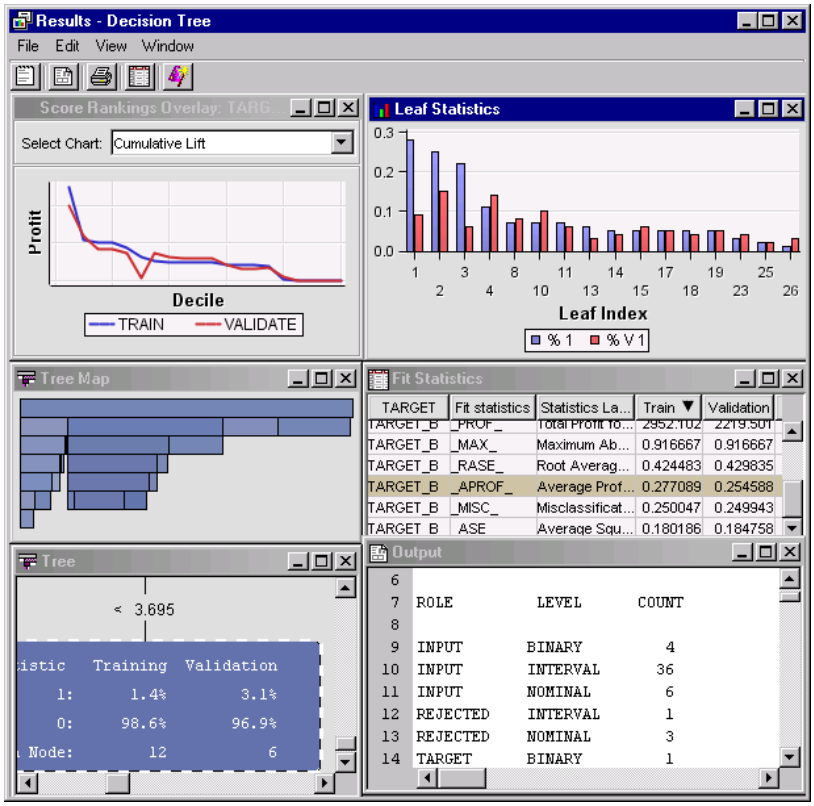


d Click OK.

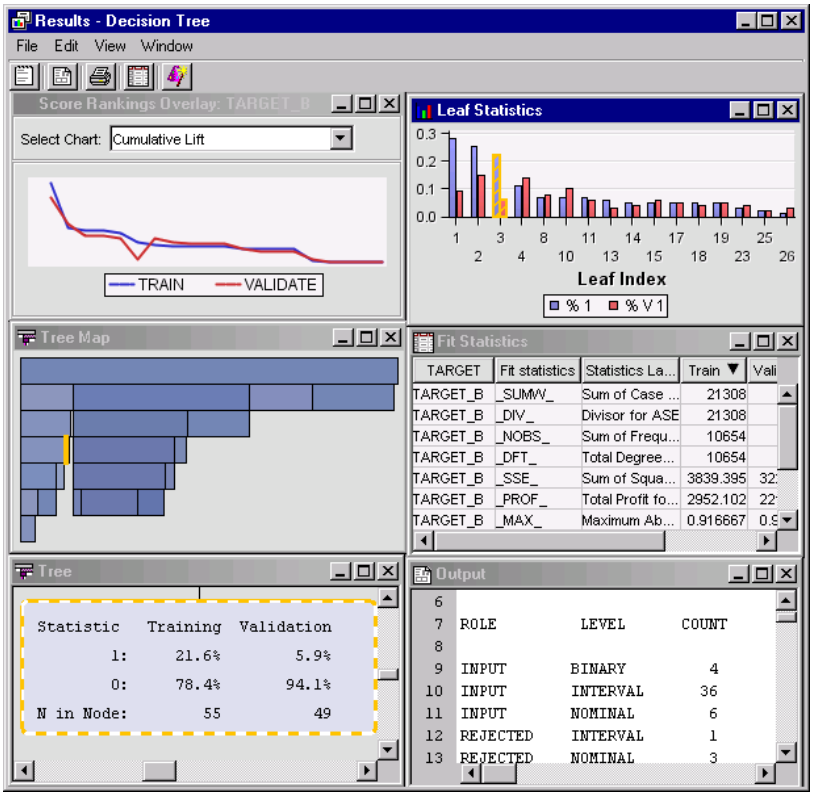
11 Restore the Score Rankings chart to its original size.



12 Select the Leaf Statistics plot and select the bar that corresponds to Leaf Index = 3. When you select the bar, note that the corresponding node is highlighted in both the Tree Map and Tree Diagram. The Leaf Statistics plot, Tree Map, and Tree Diagram are interactive, dynamically linked plots. This feature is especially useful when you are using the Tree Map to isolate interesting nodes in a large tree.

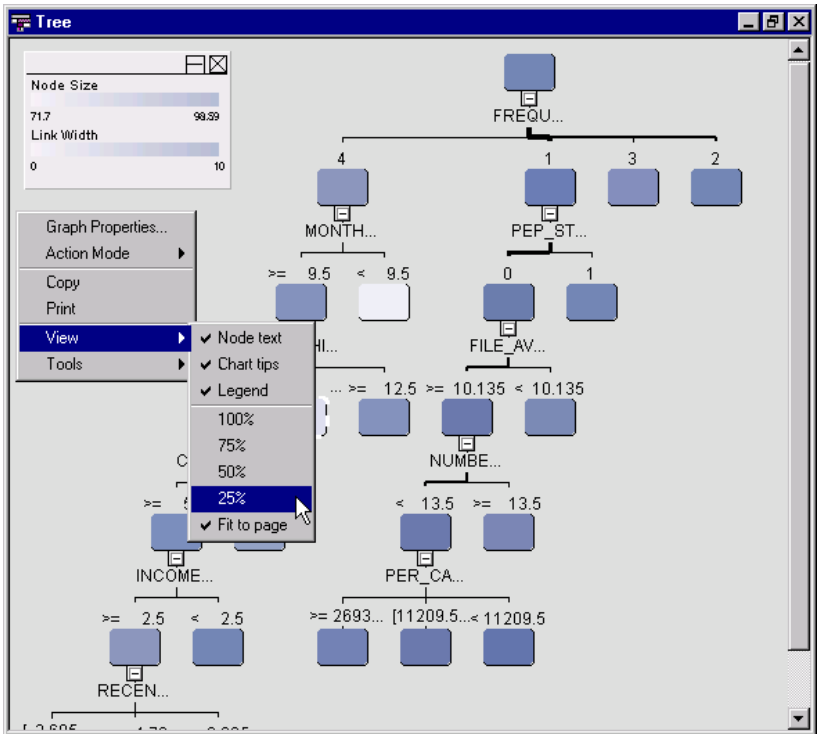


The largest nodes with a high percentage of donors are in the lower left quadrant. Select a node from the lower left quadrant and examine the corresponding node in the Tree view.

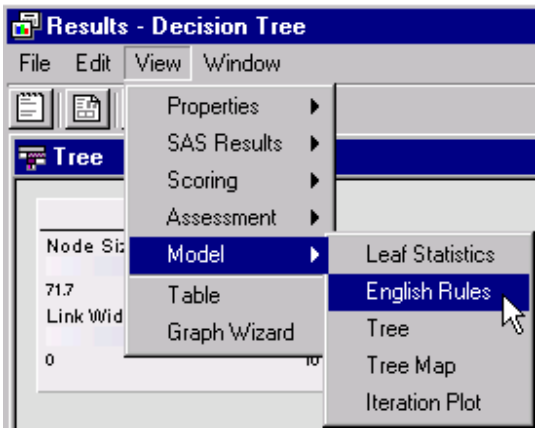


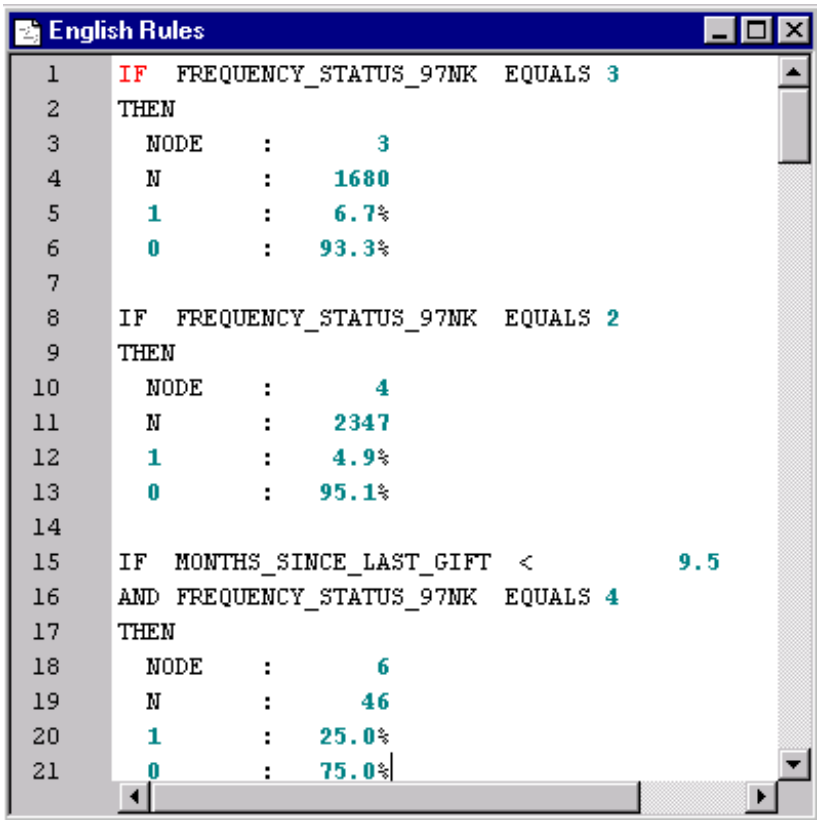
13 Move your mouse pointer over the node to display node statistics for both donors (1) and non-donors (0). By default, each node in the Tree Diagram displays the predicted target value percentages and counts, which in this leaf is 0.

14 Maximize the Tree window and explore the Tree diagram. Note that the line thickness for each split indicates the number of cases in each underlying node. Right-click the plot background and examine the different View menu item settings.

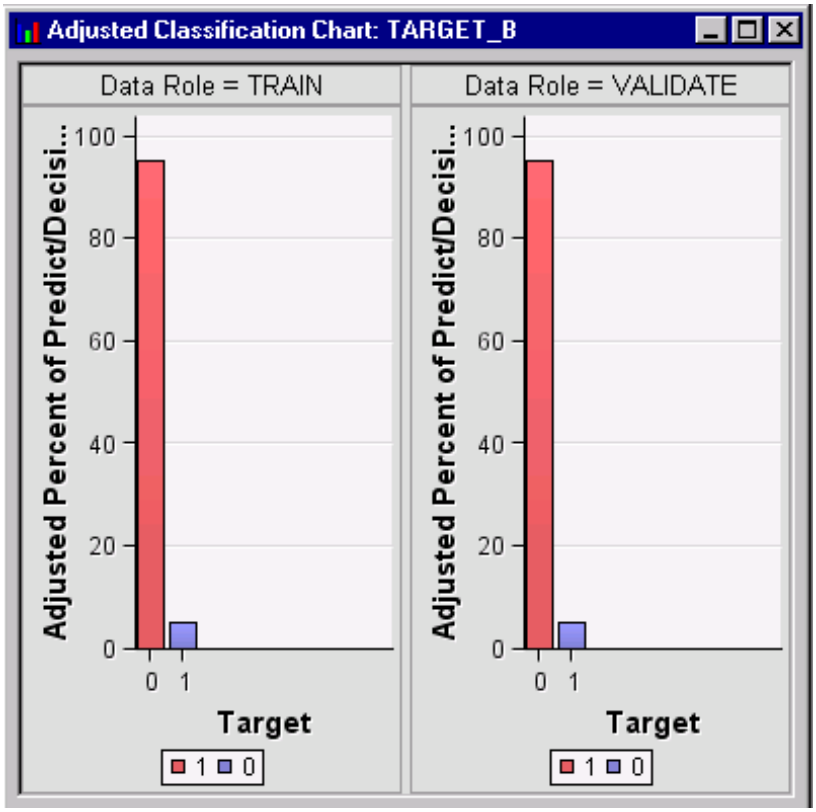


15 Select **View** ▶ **Model** ▶ **English Rules** from the Results window menu in order to view the English Rules.





16 Select **View** ▶ **Assessment** ▶ **Adjusted Classification Chart: TARGET_B** from the Results window menu to view the Adjusted Classification chart.



Notice that none of the donors has been correctly classified in either partitioned data set. However, the goal is centered on isolating the set of candidate donors that will maximize profit. Even small average profit values of about 26 cents will result in a significant total profit, especially when applied to a large customer base.

17 Examine the Score Code. From the main menu, select **View ► Scoring ► SAS Code**. You will notice these entries:

- Publish Score Code is the SAS score code that you can use to score data in applications that run outside the Enterprise Miner environment.
- Flow Score Code is the SAS score code that contains potential warnings and error messages about the targets in the process flow diagram.
- Enterprise Miner also produces PMML, an XML representation of a data mining model. SAS PMML is based on the Data Mining Group PMML Version 2.1, that has significant extensions to support the data types, transformations, and model definitions that SAS requires. These files can be used with PMML scoring engines that support PMML Version 2.1.

18 Close the Tree Results window.

Create an Interactive Decision Tree

About the Tree Desktop Application

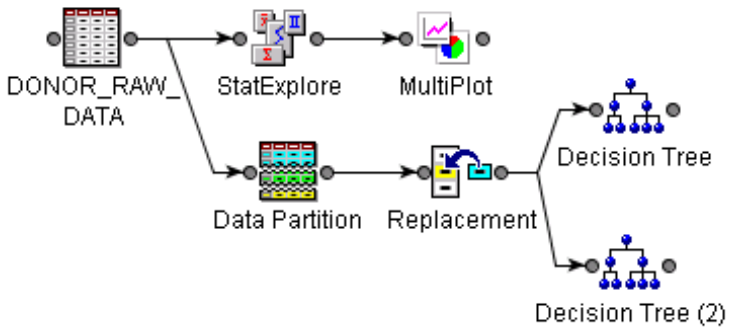
SAS Enterprise Miner Tree Desktop Application is a Microsoft Windows application that implements decision tree methodology for data mining. The application functions

in either viewer mode or SAS mode. Viewer mode enables you to interactively browse decision trees that are created with the Enterprise Miner Tree node. SAS mode enables you not only to browse the results, but also to use the software as a complete application by providing automatic and interactive training modes. After you train the tree interactively with the application, you can also view the Java tree results.

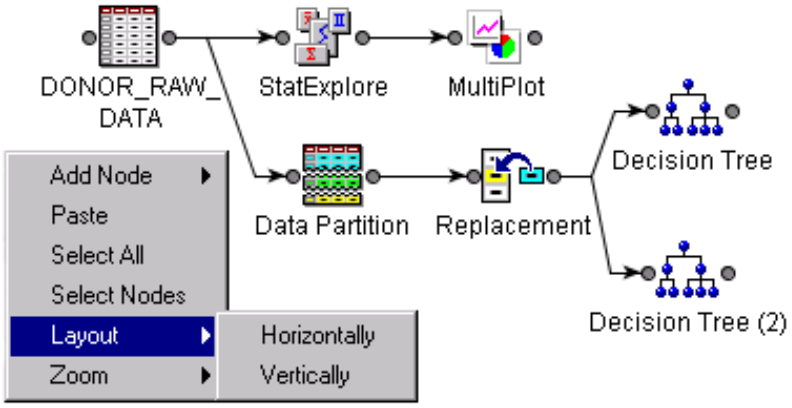
Invoke the Application

In this task, you will use Tree Desktop Application to assess the decision tree model.

1 Drag a second Decision Tree node from the **`Model`** tab on the toolbar into the Diagram Workspace and connect it to the Replacement node.

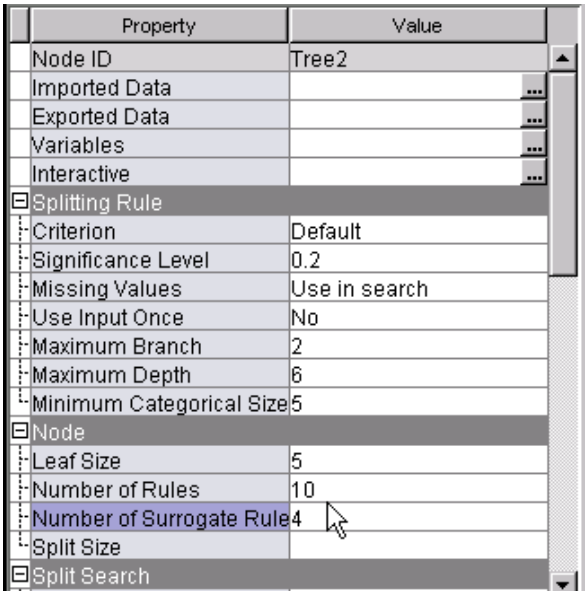


Note: To organize the diagram layout, right-click the background of the Diagram Workspace and select **Layout ► Horizontally** as shown below. Continue to use this feature to organize the layout as the diagram becomes more complex. △

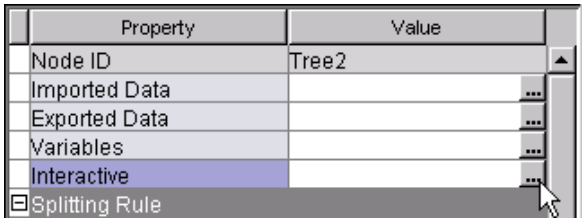


2 Select the second Decision Tree node and set the following **`Tree node`** properties in the Properties panel:

- Set the **`Number of Rules`** to **`10`**. This property controls how many candidate splitting rules are shown during interactive training.
- Set the **`Number of Surrogate Rules`** to **`4`**.

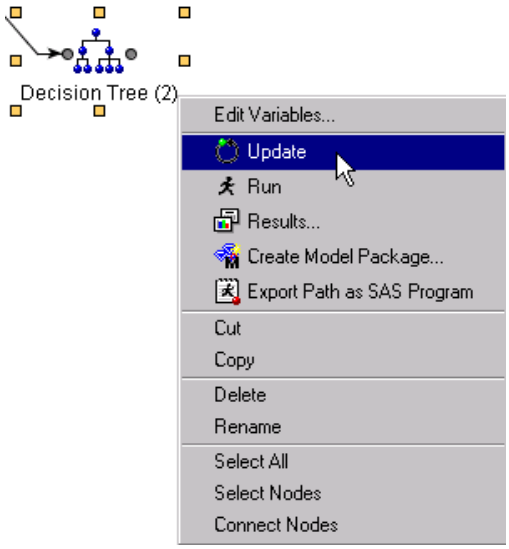


3 Click the ellipses button to the right of the **`Interactive`** property to invoke theTree Desktop Application.



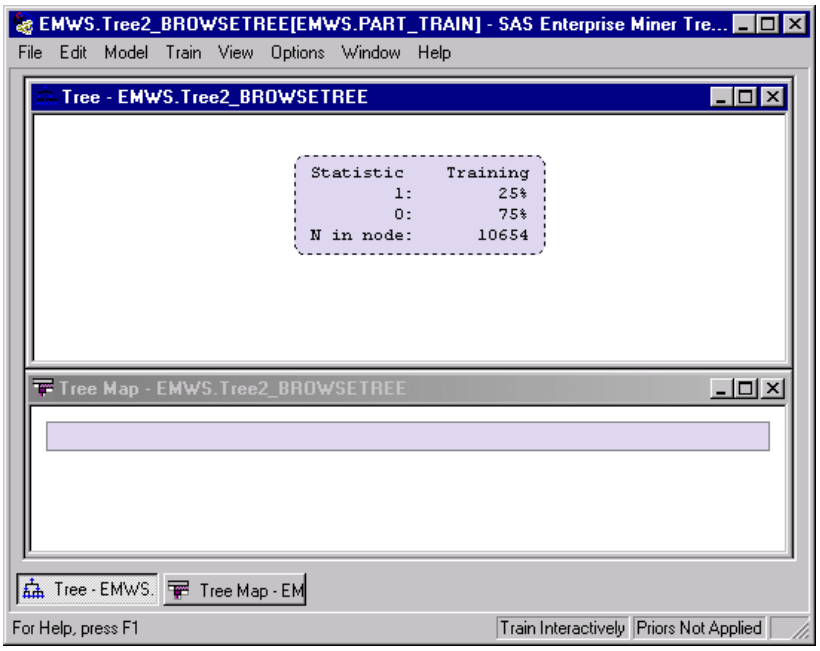
Note: If you are asked to update the path before the application is invoked, click OK. △

4 Right-click the Decision Tree node and select **`Update`** in order to ensure that all predecessor nodes have been run.



5 Click OK from the Status window when the path is updated.

By default, the root node of the tree diagram and the tree map are displayed.

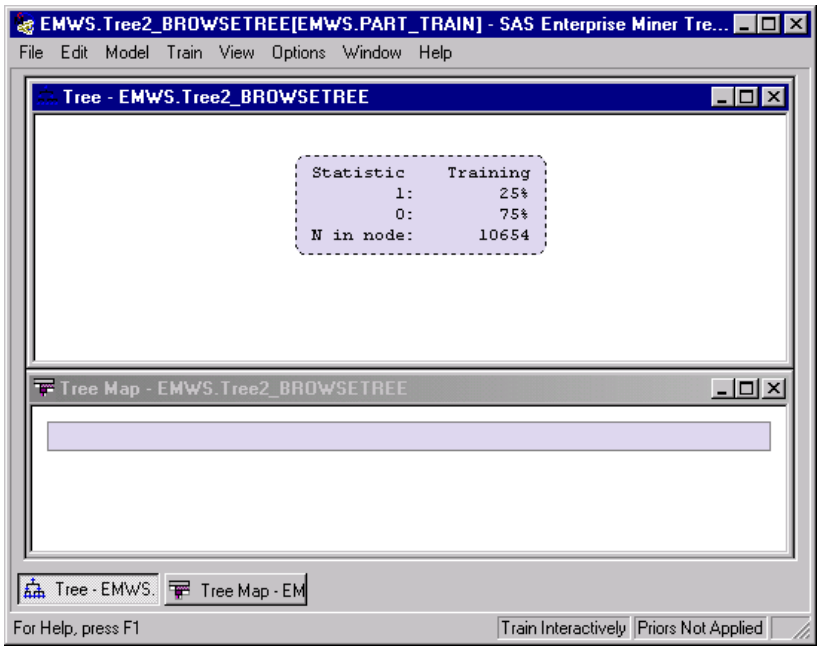


Note: If the tree map is not displayed, select **View** ► **Tree Map** from the window menu. Select **Windows** ► **Tile** window menu to arrange your windows.

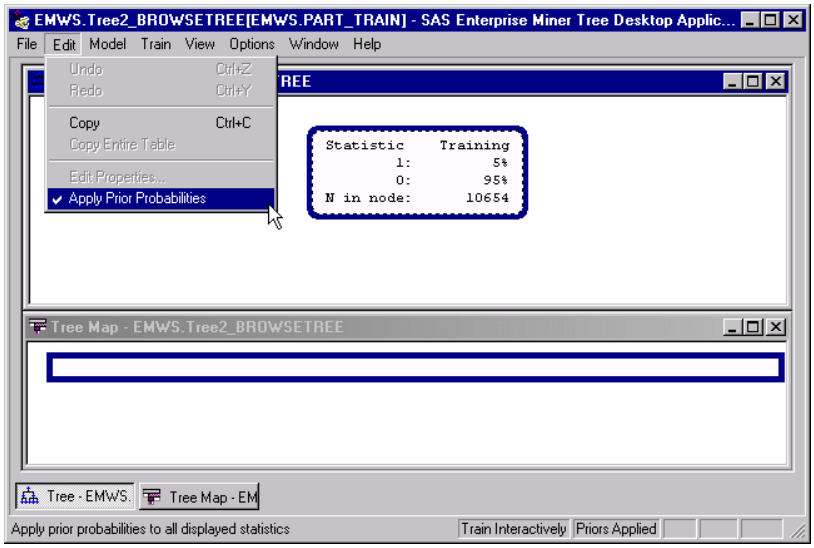
You can also press the F1 key to open the TDA Help. △

Assign Prior Probabilities

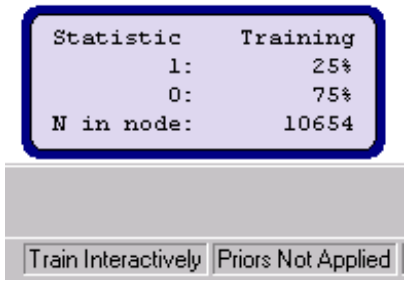
1 Examine the messages in the lower right-hand corner of the window. Note the message **`Priors Not Applied`**. The message indicates that the probabilities that were defined earlier have not been applied to this view of the data.

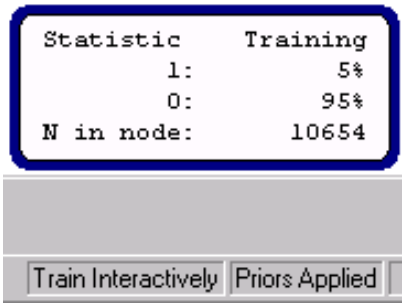


2 Select **Edit** ► **Apply Prior Probabilities** from the menu in order to apply the prior probabilities that you defined earlier.



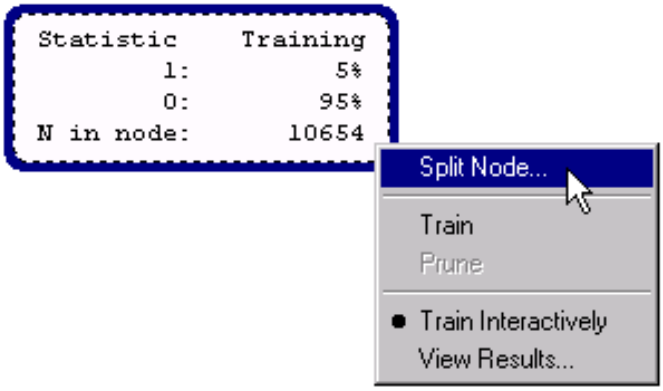
Note that the node counts are now adjusted for the priors. The message panel at the bottom right verifies that the prior probabilities have been applied.





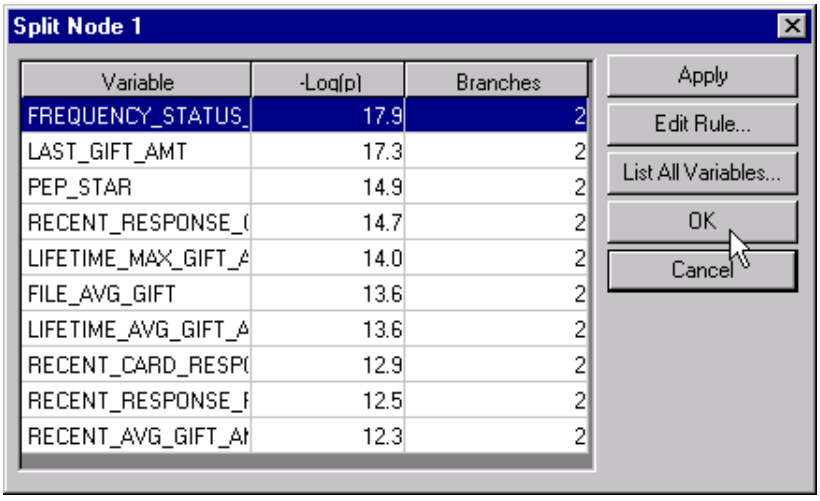
Create the First Interactive Split

1 Right-click the root node of either the tree diagram or the tree map and select **`Split Node`**.



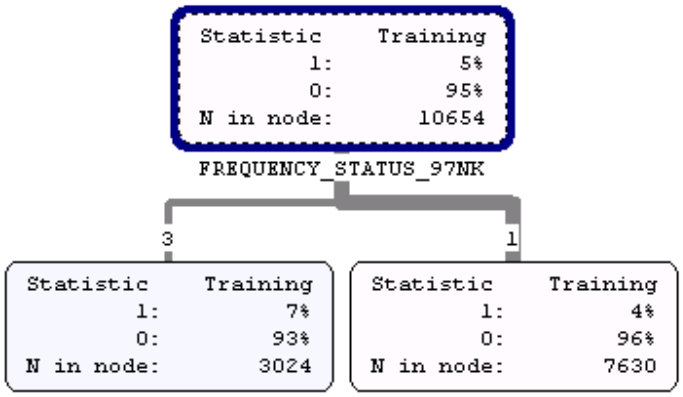
The Candidate Splitting Rules window opens, displaying the top ten variables, which have been sorted by logworth. In this case, logworth is the negative log of the *p*-value for the Chi-Square test. Good predictors have higher logworth values.

2 Select the variable FREQUENCY_STATUS_97NK in the Split Node window.



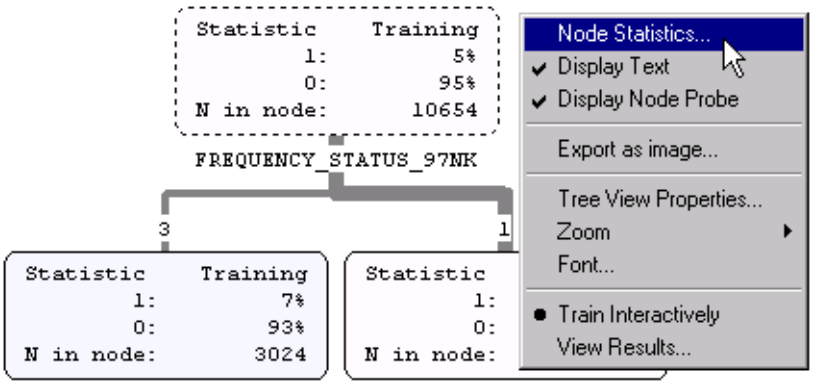
3 Click OK to define the first split.

The Tree Diagram shows the first split.

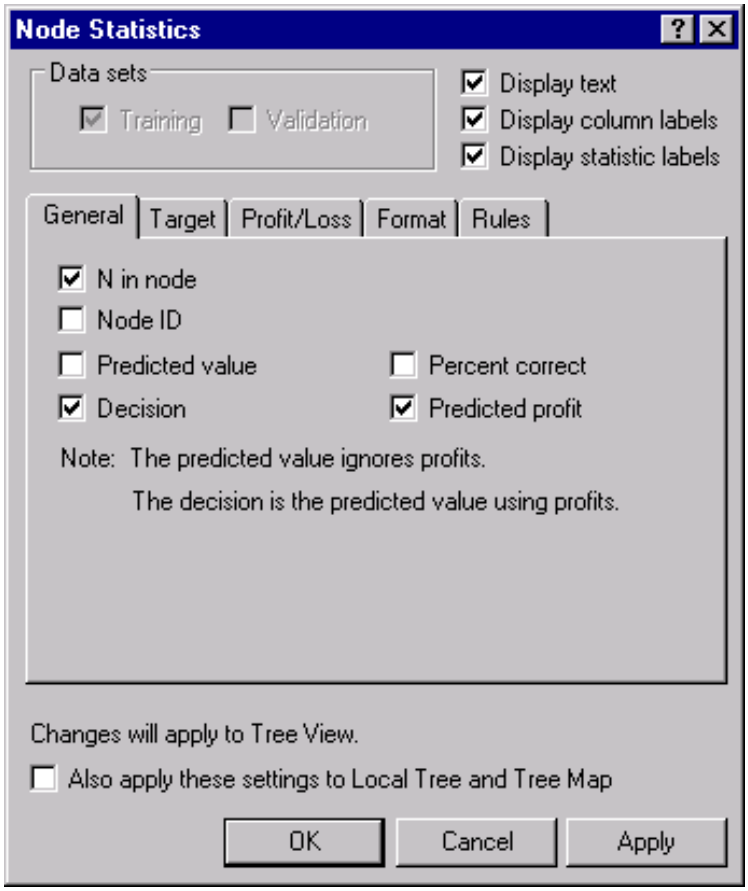


Add Additional Node Statistics

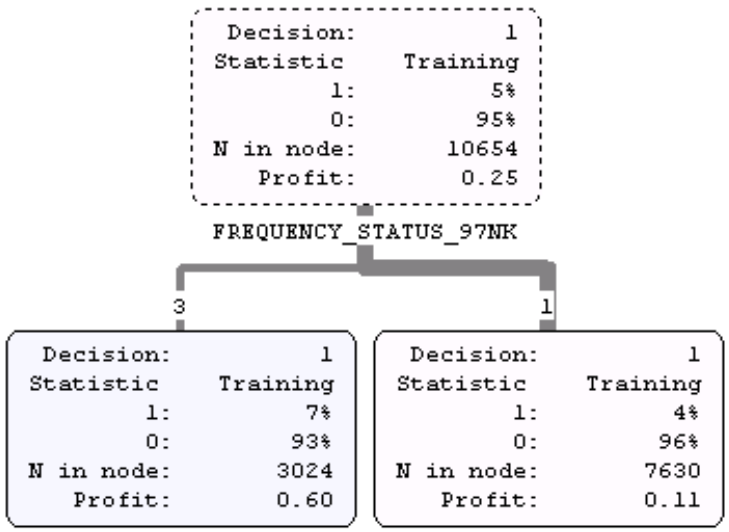
1 Right-click the background of the Tree Diagram, and then select **Node Statistics**.



2 In the **General** tab of the Node Statistics window, select the **Decision** and **Predicted Profit** boxes and click OK.



The Decision Tree displays the additional statistics.

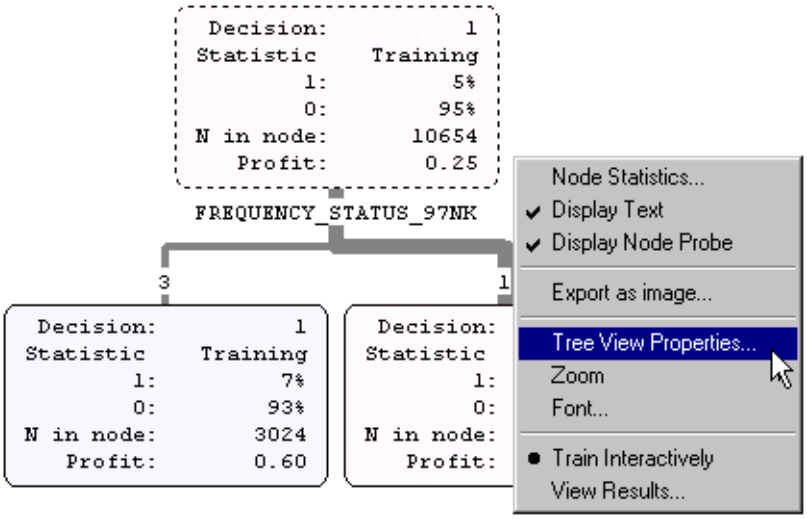


Note: By default, the width of a line from a parent node to a child node depends on the ratio of the number of cases in the child node compared to the number of cases in the parent node. This distinction is useful when you are examining a much larger tree and you hide the node statistics. △

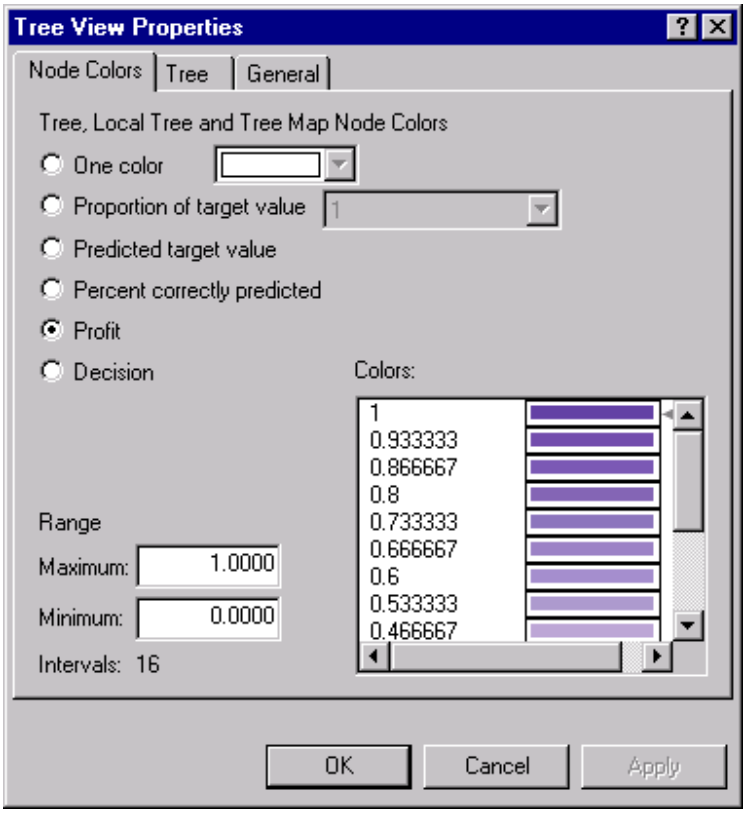
Shade the Nodes by Profit

You can shade the nodes according to the expected profit value. By default, both the Tree Map and the Tree nodes are shaded according to the proportion of the target event in each node. Lighter shaded nodes indicate a greater frequency of non-donors.

1 Right-click the background of the Tree and select **`Tree View Properties`**.



2 In the Tree View Properties window, select **`Profit`**.

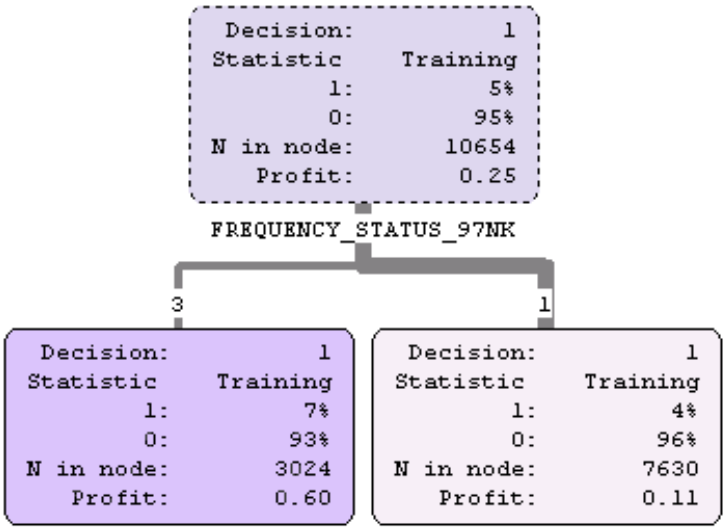


3 Set **`Range Maximum`** to **`1`**.

4 Set **`Range Minimum`** to **`0`**.

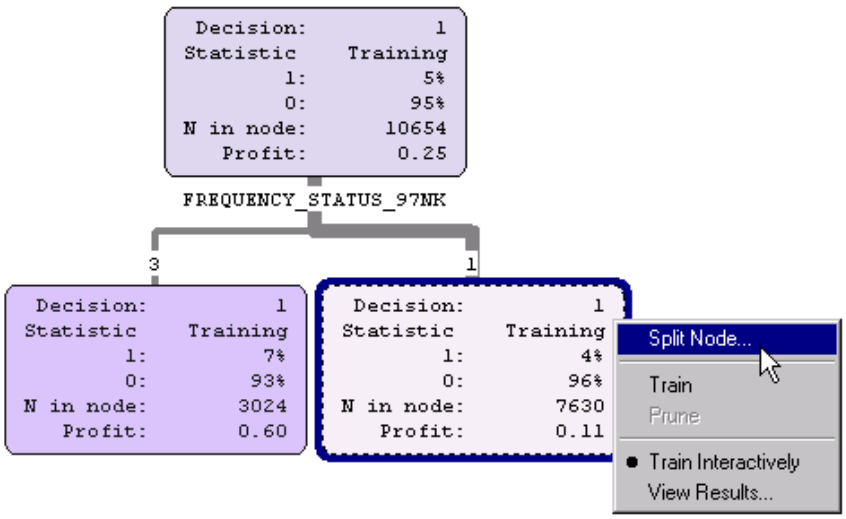
5 Click OK.

In the Tree window, note that nodes that have higher expected **`Profit`** values are shaded darker than nodes that have lower expected **`Profit`** values.



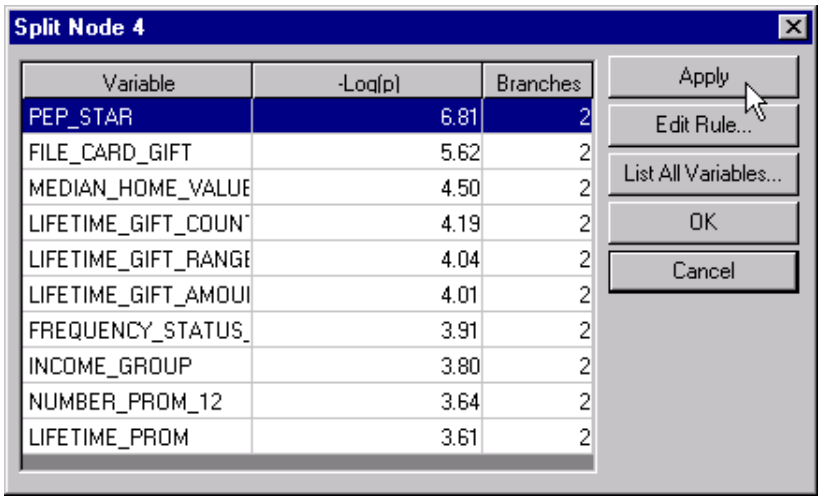
Define the Second Split

1 Right-click the node that has 7,630 cases and select **`Split Node`**.

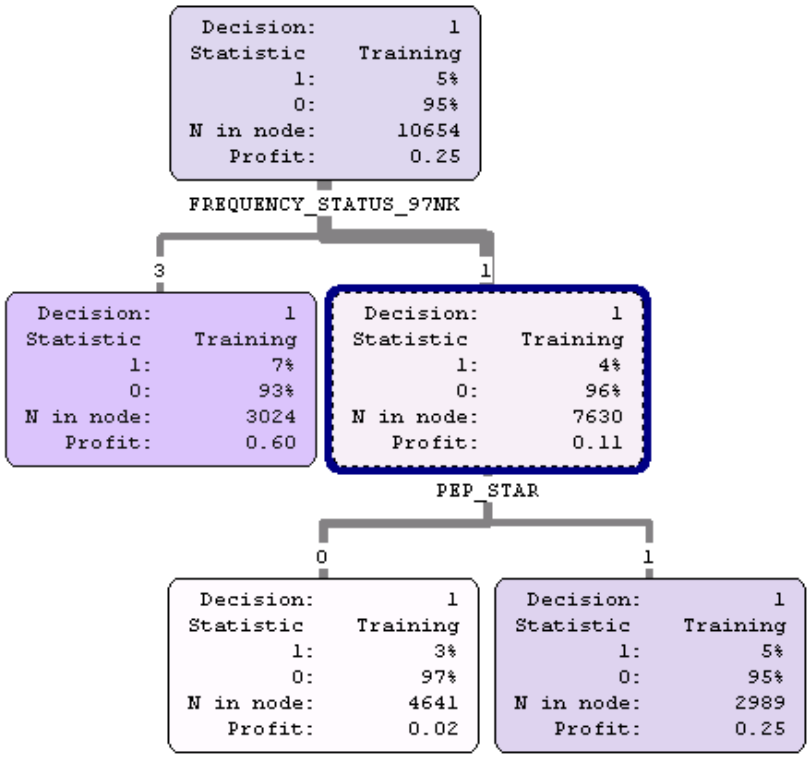


The Split Node window opens. Candidate rules are always shown for the node that you choose. You control how many rules are displayed prior to invoking the application.

2 Select PEP_STAR as the splitting variable and then click Apply.

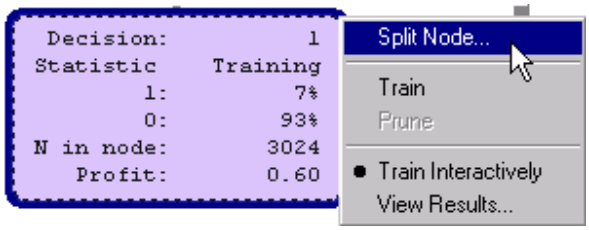


3 Click OK. The Tree appears as follows:

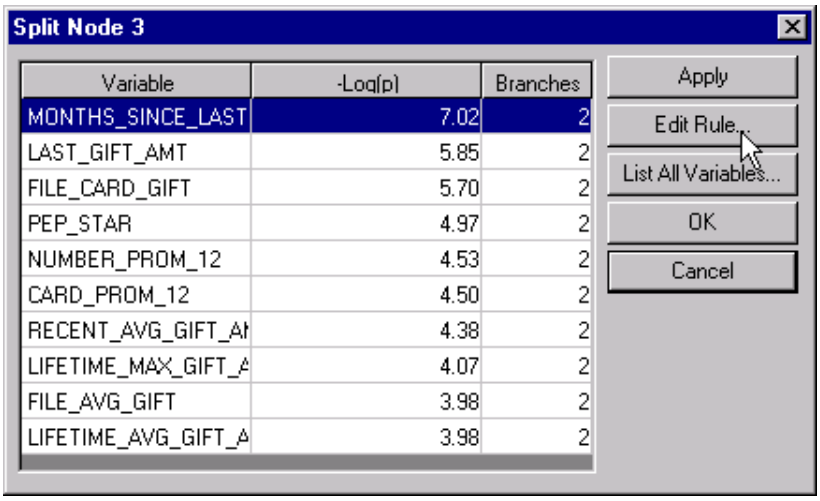


Create a Multi-Way Split

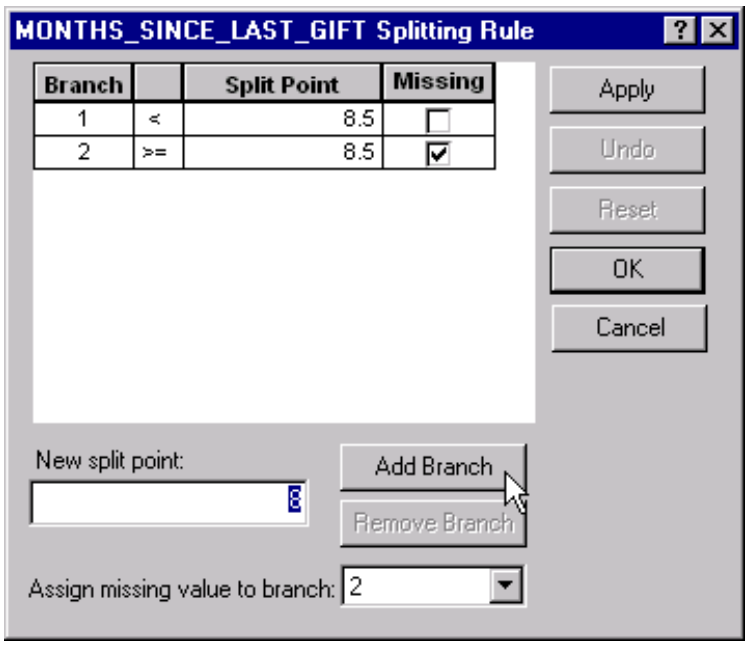
1 To open the candidate splitting rules for the dark-shaded node (the node that has 3,024 observations), right-click the node and select **`Split Node`**.



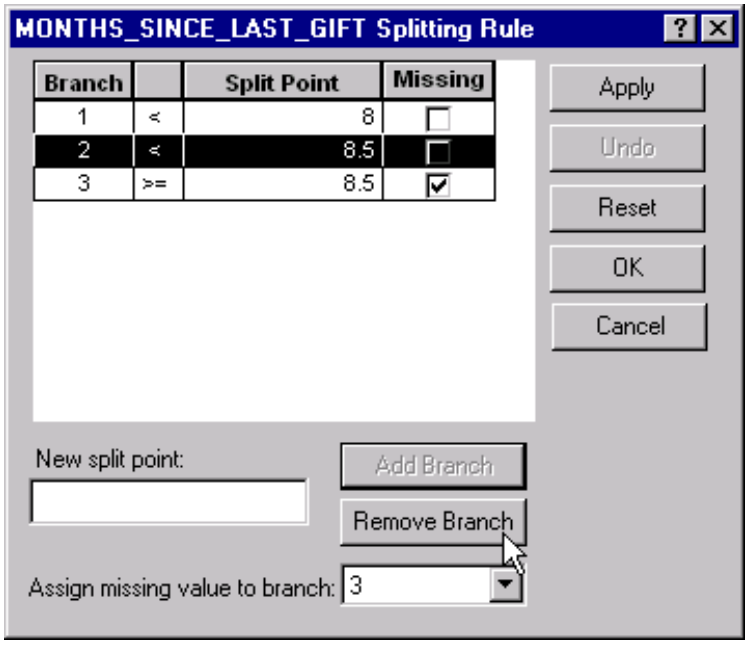
2 In the Split Node window, select the variable MONTHS_SINCE_LAST_GIFT and then click Edit Rule.



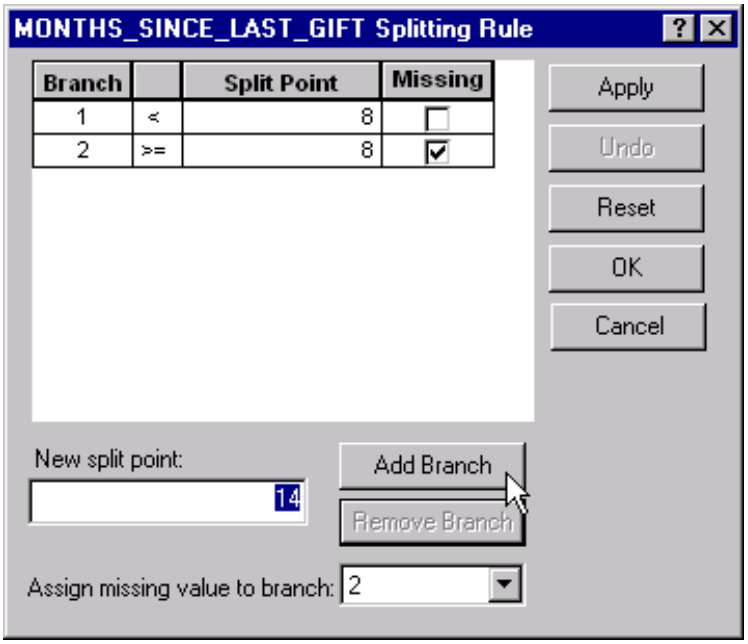
3 In the MONTHS_SINCE_LAST_GIFT Splitting Rule window, enter **8** in the **New split point** box and then click Add Branch.



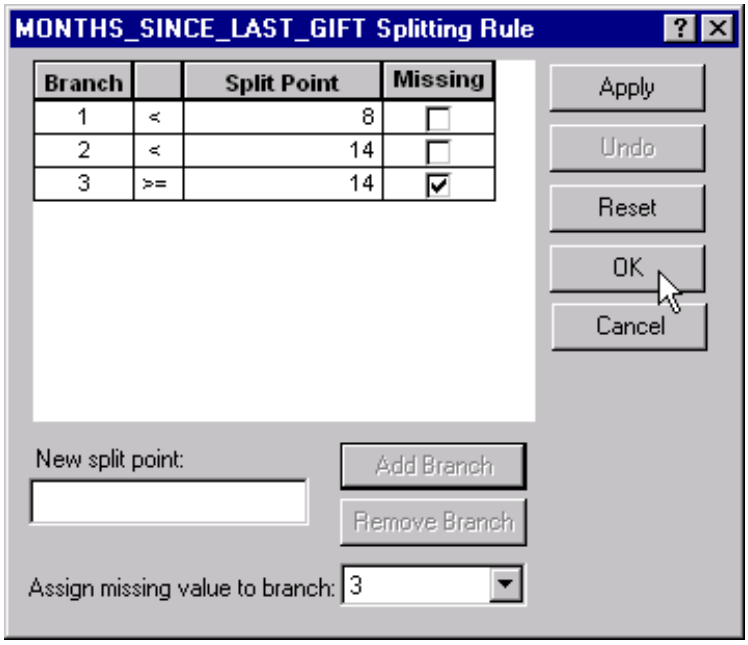
4 Select Branch 2 (<8.5) and then click Remove Branch.



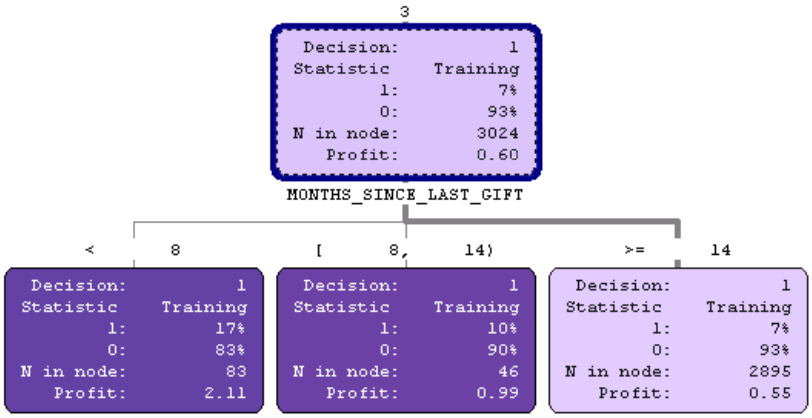
5 In the **`New split point`** box, enter **`14`** as the third split point and then click Add Branch.



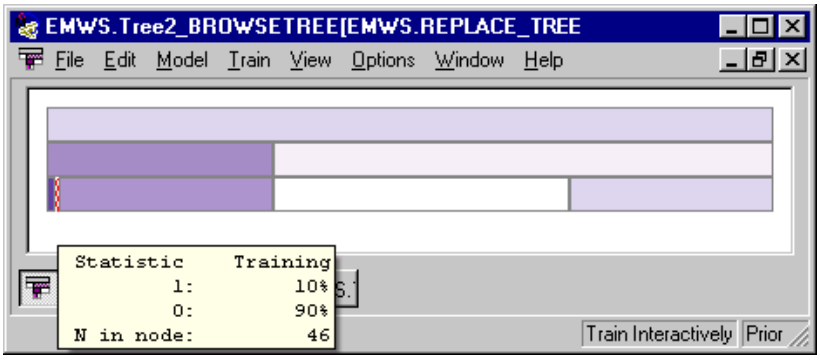
6 Click OK in order to create the modified three-way split.



The node that has 3,024 observations is split three ways.



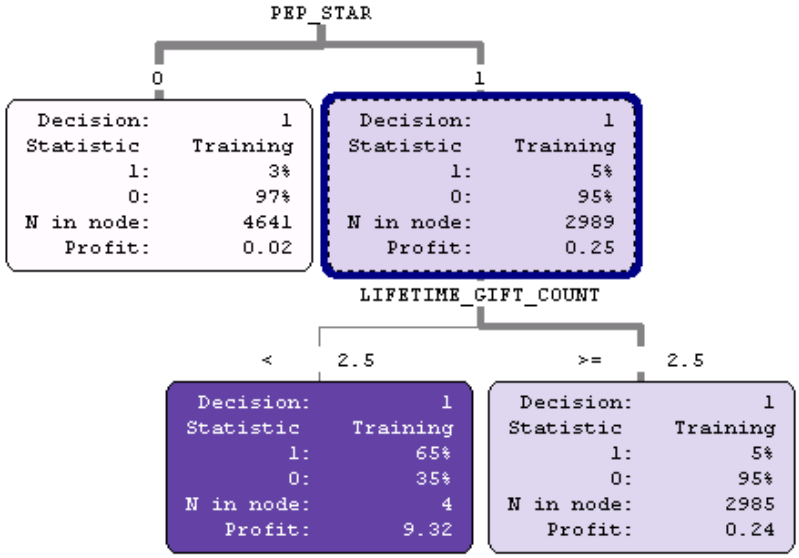
7 Select **View ▶ Tree Map** from the main menu. The node that has only 46 observations is shaded red in the Tree Map. Recall that the width of the node is proportional to the number of training cases in the node. Nodes that contain few observations cannot be drawn accurately in the space that is allocated to the view. Depending on how your windows are arranged, you might or might not see a red node. Try reducing the size of the Tree Map window in order to see a node that has fewer observations.



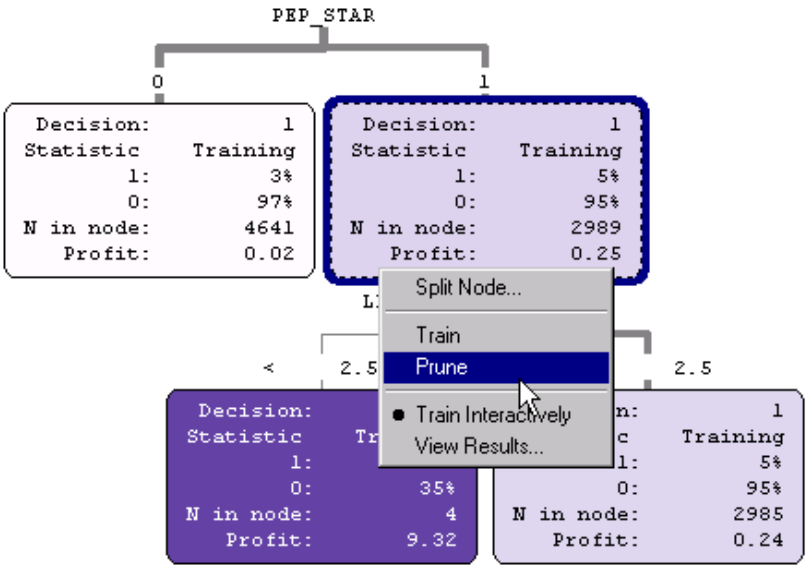
Prune a Node from the Tree

Use interactive mode to prune a tree.

1 To define a split for any current terminal node, right-click the node. Select **Split Node**, then select a splitting rule, and click OK. This example uses the top right node, the node that has 2,989 observations.

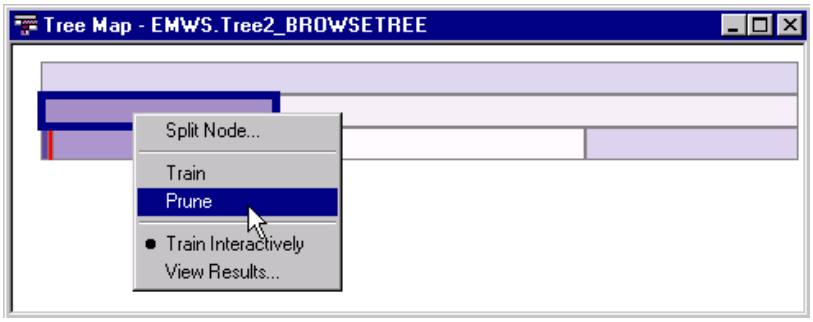


2 Right-click the parent node for the split that you just defined and select **Prune**.



Note: You can also define splits and prune nodes from the Tree Map view.

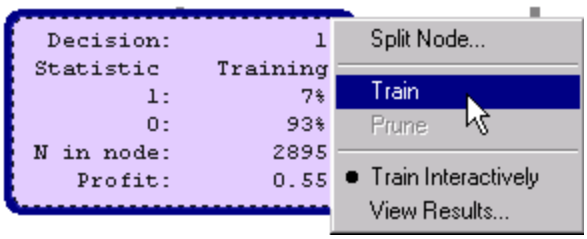
△



Train the Tree in Automatic Mode

At this point you can continue to split and prune nodes interactively until you obtain a satisfactory tree, or you can use the algorithm to train the rest of the tree in automatic mode.

1 To train the tree in automatic mode, right-click a terminal node that has several observations, then select **`Train`**. Repeat the process as desired for other nodes.



Other Tree Control Features

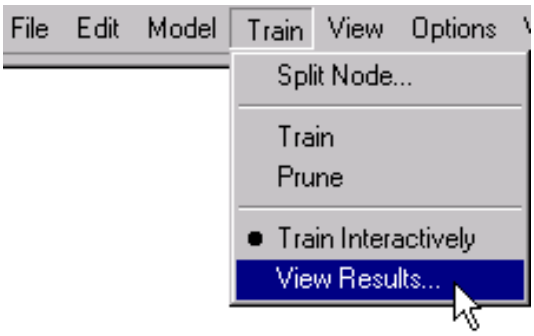
Here are some additional Tree features that you might want to explore:

- Use the zoom in/out feature by right-clicking on the background of the Tree and selecting **Zoom**. You might also want to change the node statistics.
- Follow a similar menu path to change the font.
- To print the tree on one page or across multiple pages, select **File** ▶ **Print**.

View the Tree Results

At this point you are still working in interactive mode.

1 From the main menu, select **Train** ▶ **View Results** to change to the results viewer mode. In this task you incorporate validation data into the evaluation of the tree.



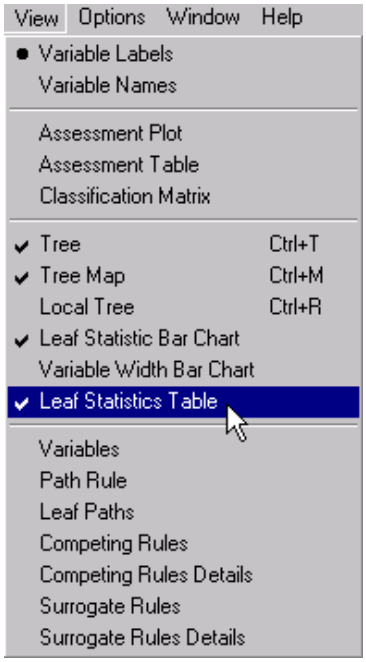
2 From the main menu, select

View ▶ **Leaf Statistics Bar Chart**

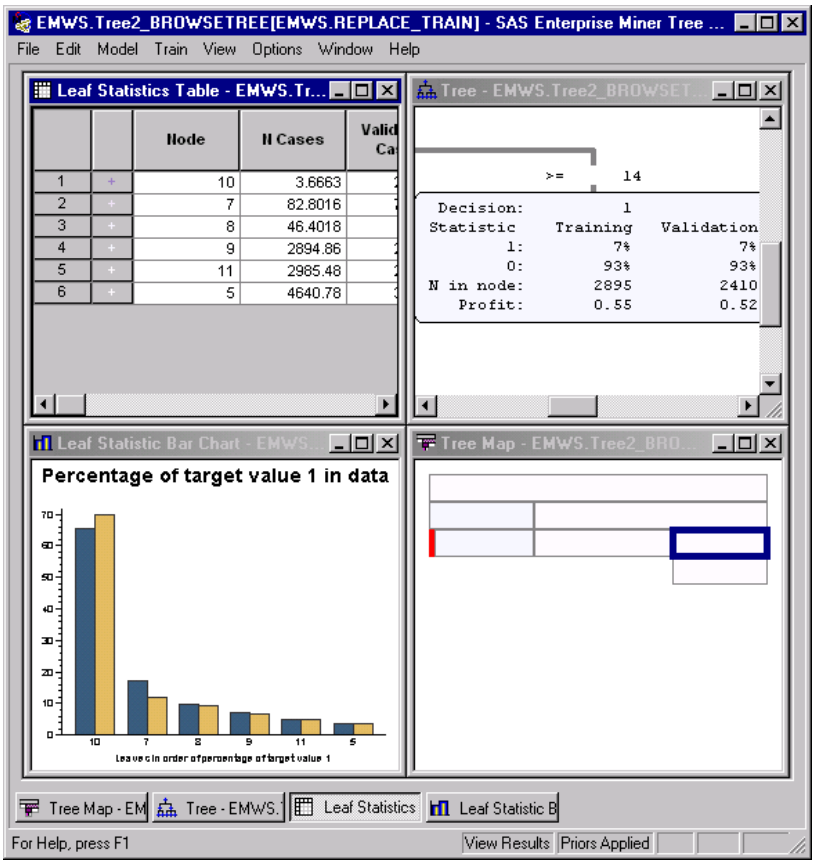
and

View ▶ **Leaf Statistics Table**

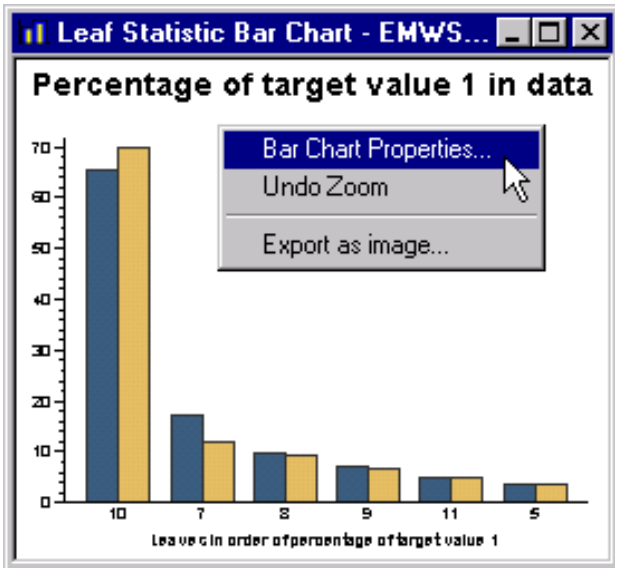
to open the Leaf Statistics Bar Chart and Table.



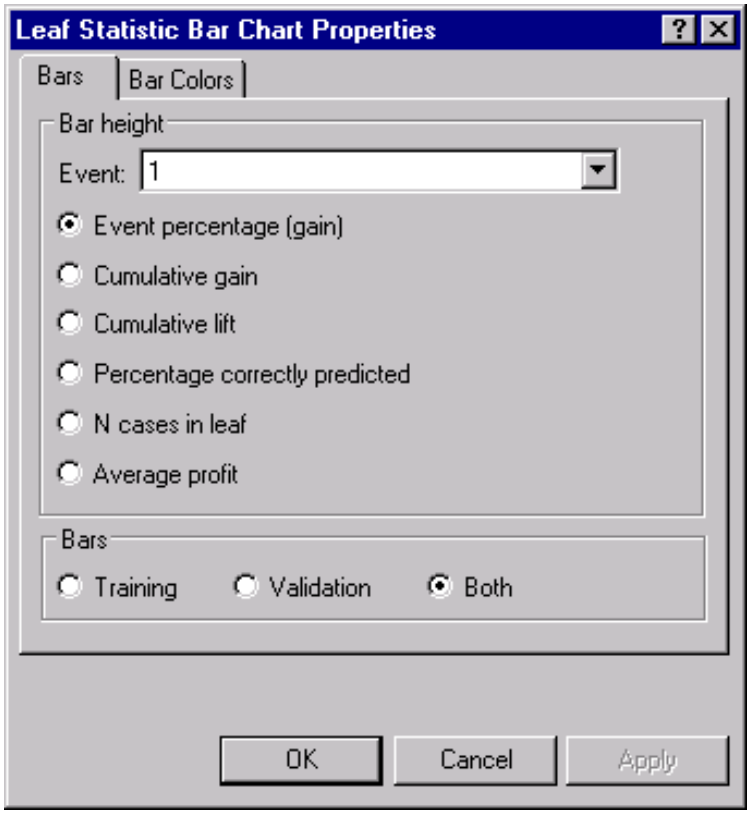
The Leaf Statistics Bar Chart and Leaf Statistics Table windows open.



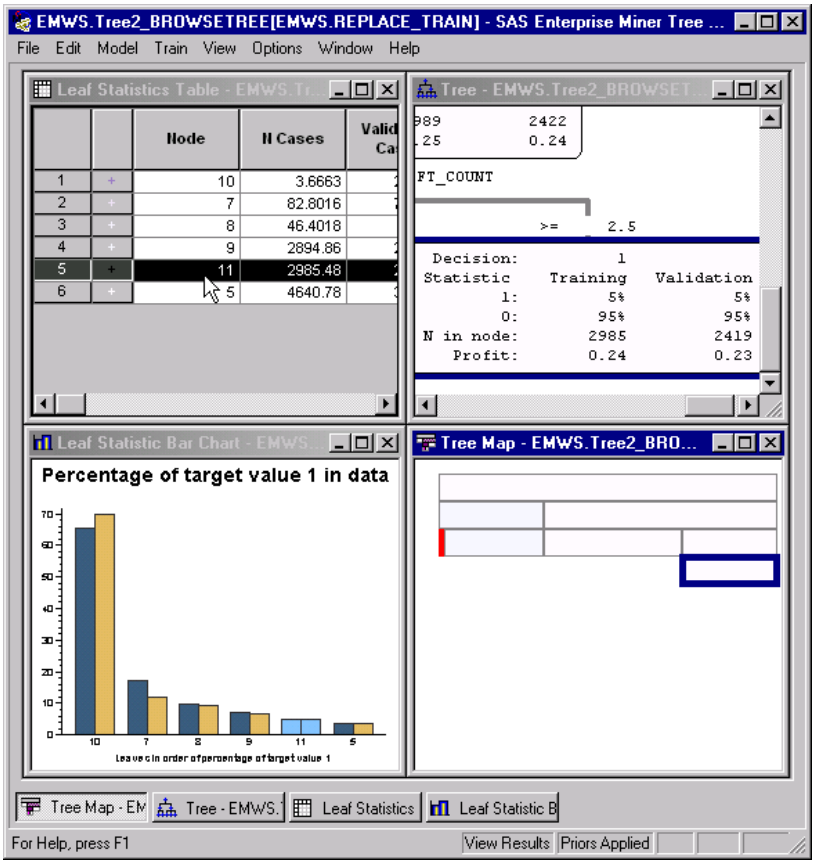
3 Right-click inside the Leaf Statistics Bar Chart and select **`Bar Chart Properties`**.



4 Examine the various Bar Chart settings.



5 Select a node, a subtree, or a variable in one window. Note that the other windows are automatically updated to select the corresponding items.

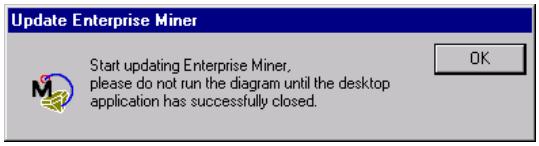


6 From the main menu, select **View ► Assessment Plot** to display the Assessment plot.

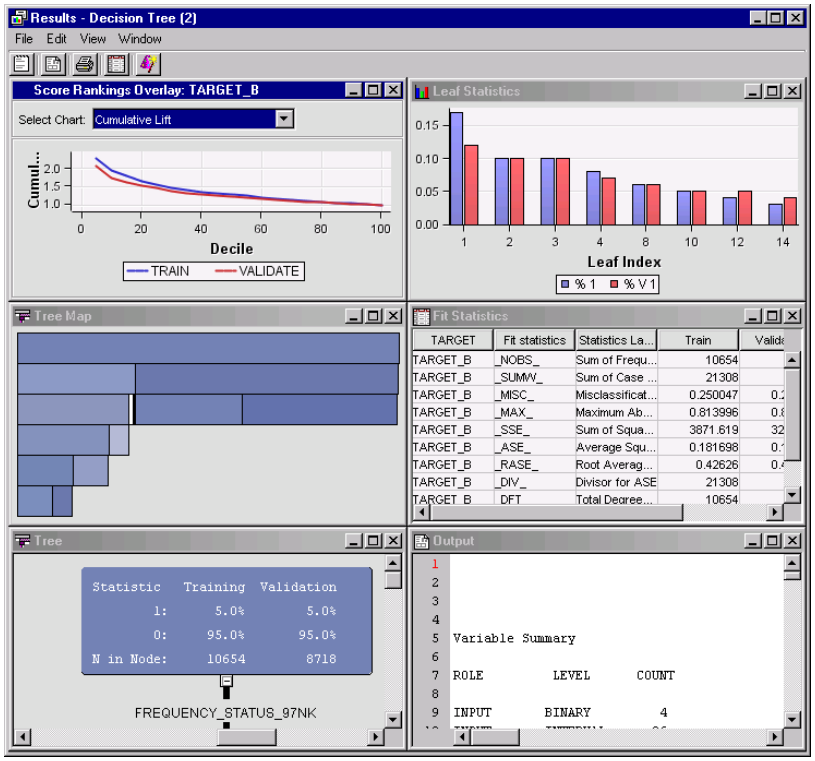
Note: You can select a smaller tree interactively by selecting a smaller number of leaves in this plot. This feature allows you to choose a smaller tree that performs well using both the training and validation data. △

View the Tree in the Java Tree Results Viewer of Enterprise Miner

1 Close the application and save the changes to the model. The following message is displayed when you close the model in the Tree Desktop Application.

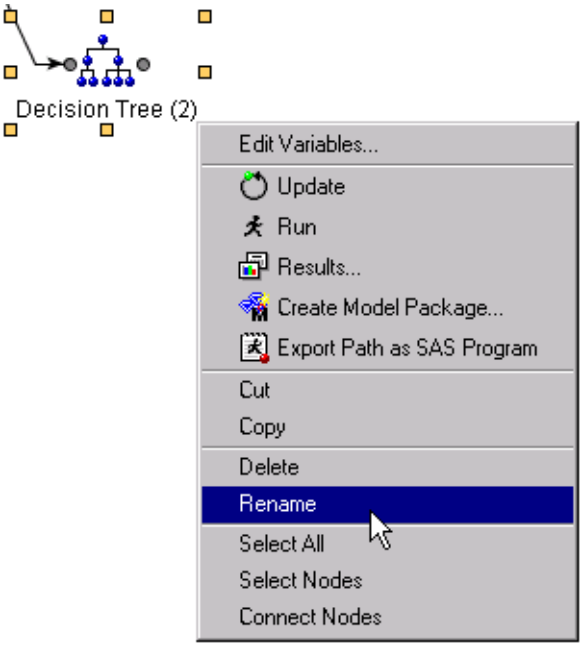


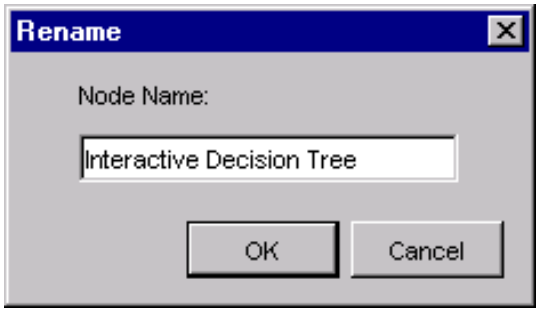
2 Run the Decision Tree node path from the Diagram Workspace and follow the messages that direct you to view the results.



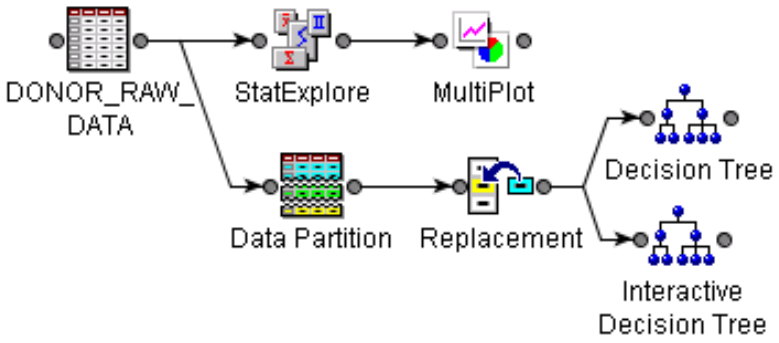
3 Close the Tree Results window.

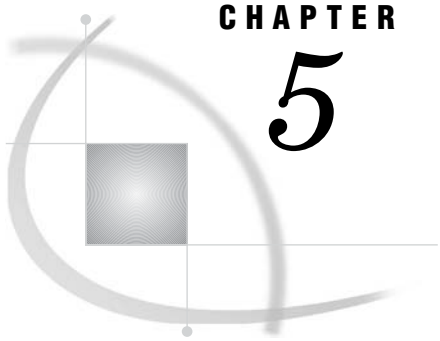
4 Rename the second Decision Tree in your diagram to indicate that it was the interactive training tree. Right-click the second Decision Tree node, select **Rename**, and rename the node, **Interactive Decision Tree**




Rename
Node Name:
Interactive Decision Tree
OK
Cancel

5 Click OK.


DONOR_RAW_
DATA
StatExplore
MultiPlot
Data Partition
Replacement
Decision Tree
Interactive
Decision Tree



CHAPTER

5

Working with Nodes That Modify, Model, and Explore

Overview of This Group of Tasks

These tasks show you several ways to fill in missing values across observations. You also create new variables from existing variables and reduce the number of input variables.

About Missing Values

Many of the input variables in the Donor data set that you have been using have missing values. If an observation contains a missing value, then by default that observation is not used for modeling by nodes such as Variable Selection, Neural Network, or Regression.

Depending on the type of predictive model that you build, missing values can cause problems. If your model is based on a decision tree, missing values cause no problems because decision trees handle missing values directly.

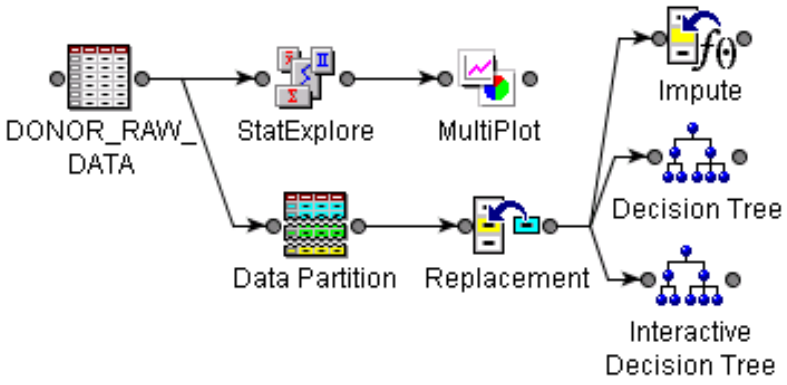
However, in Enterprise Miner, regression and neural network models ignore observations that contain missing values. Substantially reducing the size of the training data set can weaken these predictive models. It is wise to impute missing values before you fit a regression model or neural network model. When you replace missing observations with imputed data, regression and neural algorithms are able to perform

whole-case analysis on the entire training data set. If you do not impute missing values for these models, the missing values might result in the creation of an inferior model. Additionally, it is important to impute missing values if you are planning to compare a regression model or neural network model with a decision tree model, because it is more appropriate to compare models that are built on the same set of observations.

Impute Missing Values

In this task, you use the Impute node to replace the missing values in the Donor data set.

1 Drag an Impute node from the **Modify** tab of the toolbar into the Diagram Workspace, and connect it to the Replacement node.



2 Select the Impute node in the Diagram Workspace. The Impute node property settings are displayed in the Properties panel.

For class variables, the Default Input Method property uses Count. The Count method replaces missing values with the value that is most common among all observations. The Mean method replaces missing values with the average of all the non-missing values. By default, Enterprise Miner does not impute replacement values for target variables.

3 Set the following properties in the Impute node Properties panel:

- In the **Class Variables** section, set the **Default Input Method** property to **Tree Surrogate**.
- In the **Interval Variables** section, set the **Default Input Method** property to **Median**.

Property	Value
Node ID	Impt
Imported Data	
Exported Data	
Variables	
Non Missing Variables	No
Class Variables	
Default Input Method	Tree Surrogate
Default Target Method	None
Interval Variables	
Default Input Method	Median
Default Target Method	None

4 Right-click the Impute node and select **Run**. When the run is complete, click OK in the Run Status window. You do not need to view the Results window now. The Impute node must run before you can use some of the features in the Transform Variables node.

Create Variable Transformations

Overview

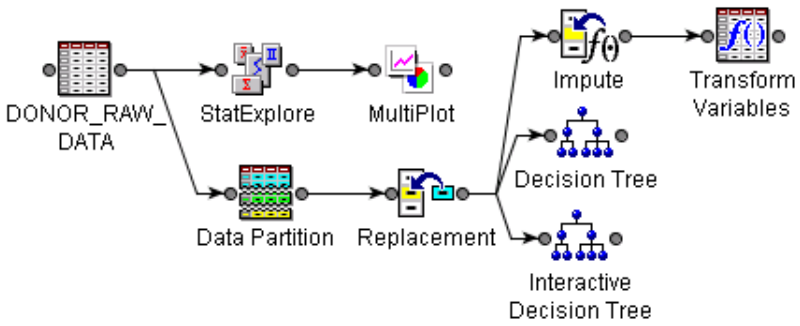
Some data can be better mined by modifying the variable values with some transformation function. The data is often useful in its original form, but transforming the data might help maximize the information content that you can retrieve. Transformations are useful when you want to improve the fit of a model to the data. For example, transformations can be used to stabilize variance, remove nonlinearity, improve additivity, and correct non-normality.

You can use the Formula Builder and Expression Builder windows in the Transform Variable node to create variable transformations. You can also view distribution plots of variables before and after the transformation to assess how effective the data transformation is.

View Variable Distribution Plots

1 Drag a Transform Variables node from the **Modify** tab of the node toolbar into the Diagram Workspace.

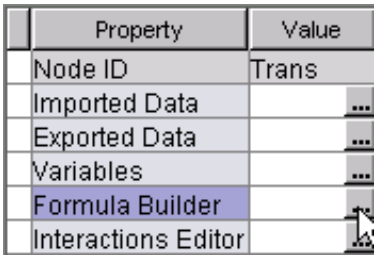
2 Connect the Impute node to the Transform Variables node.



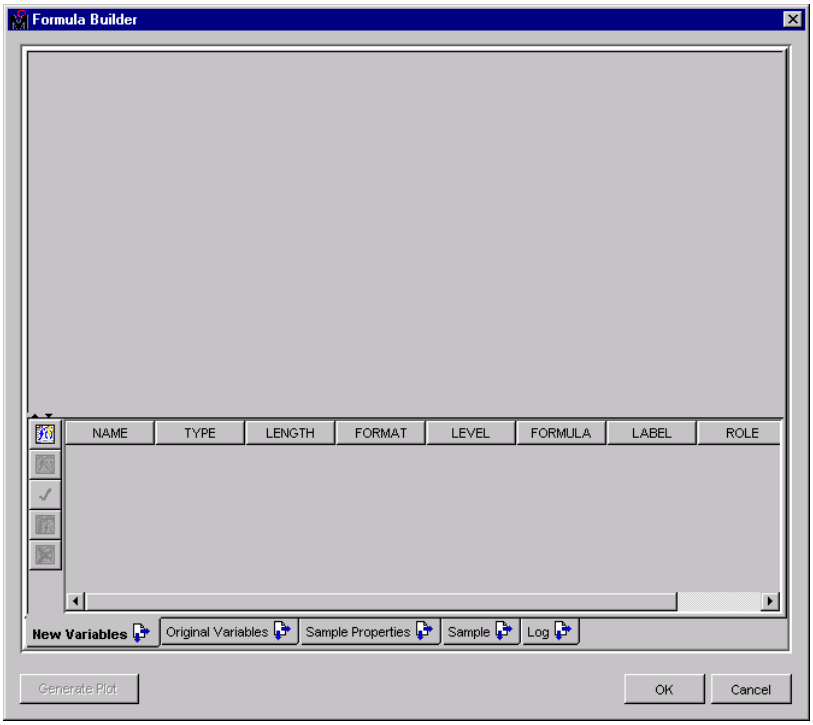
3 Select the Transform Variables node in the Diagram Workspace to view its settings in the Properties panel. The default transformation method for all variables is None. You can use the Variables property to configure variable transformation on a case-by-case basis, or you can use the Default Methods section of the Properties panel to set transformation methods according to variable type.

The variable distribution plots that you view in the Transform Variables node are generated using sampled data. You can configure how the data is sampled in the Sample Properties section of the Transform Variables node Properties panel.

4 In the Properties panel for the Transform Variables node, click the ellipsis button to the right of the **`Formula Builder`** property. This action opens the Formula Builder window.



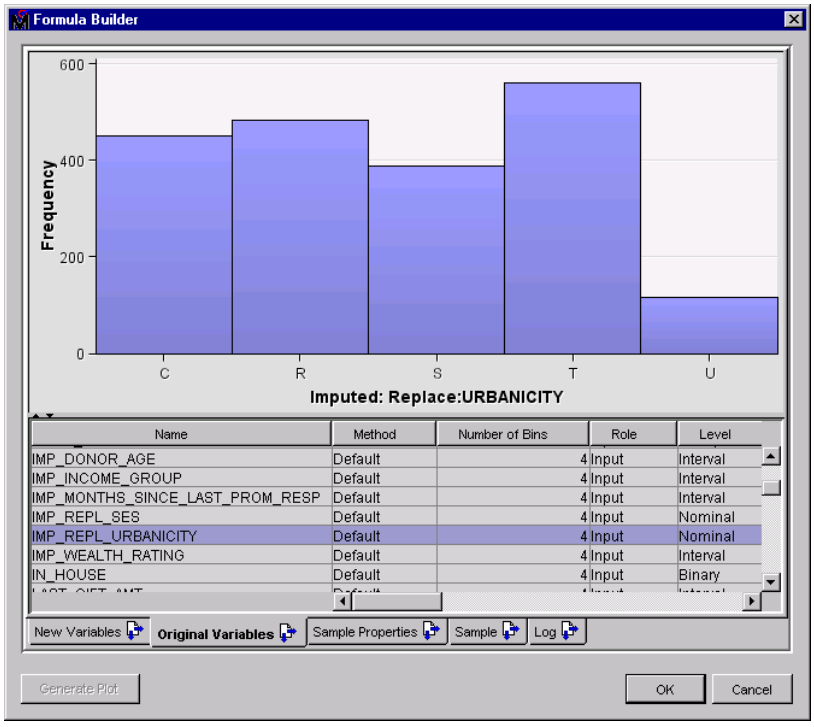
In the Formula Builder window, the New Variables table is empty, because you have not created any variables yet.



5 Select the **`Original Variables`** tab to see a list of the current variables and a distribution plot for the selected variable.

6 Examine the distributions of the current variables, and note which variables might benefit from transformation. A good variable transformation modifies the distribution of the variable so that it more closely resembles a normal distribution (a bell-shaped curve).

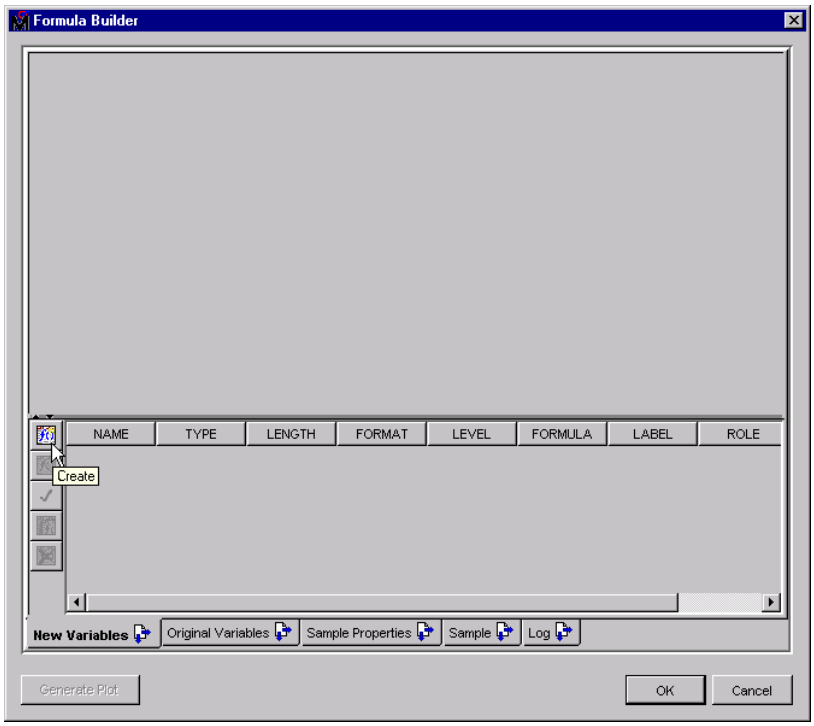
7 View the distribution plots for the variables SES and URBANICITY to see the data before the missing values were replaced with imputed values. Distribution plots for the variables IMP_REPL_SES and IMP_REPL_URBANICITY show the data after the missing values were imputed and replaced.



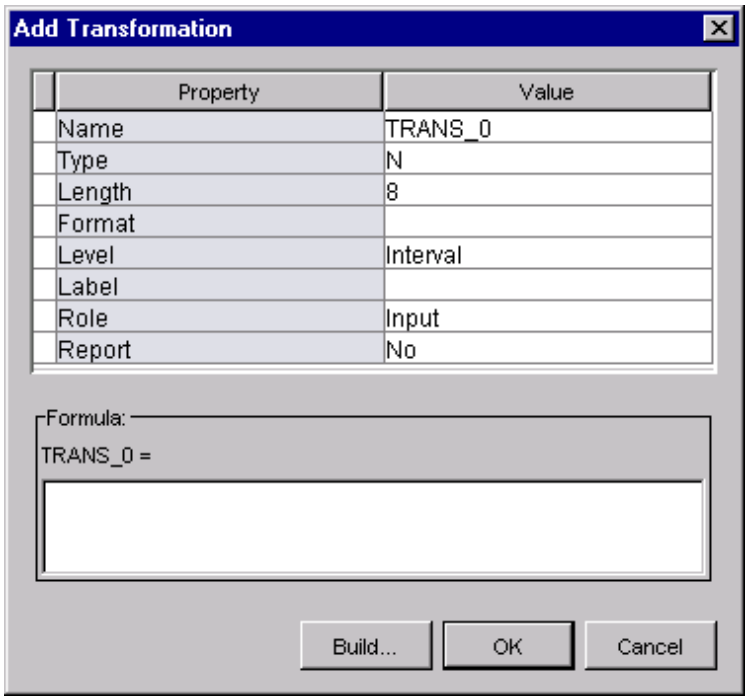
Add a Variable Transformation

1 Select the **`New Variables`** tab to create a new variable by using the Formula Builder.

2 Select the **`Transform Variables`** icon  on the left side of the Formula Builder to start creating a variable transformation.

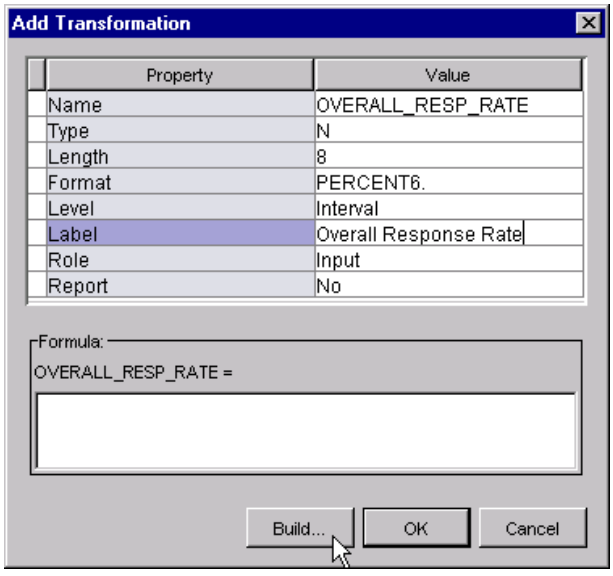


The Add Transformation window opens.

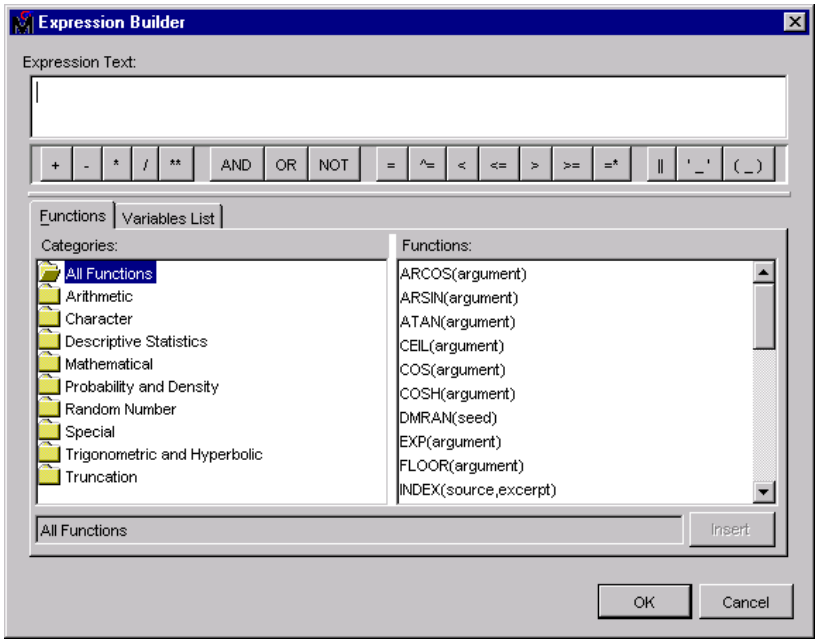


3 Edit the following **Value** columns to configure the new variable that you are creating:

- Change the **Name** from **TRANS_0** to **OVERALL_RESP_RATE**.
- Set **Format** to **PERCENT6.**.
- Set **Label** to **Overall Response Rate**.

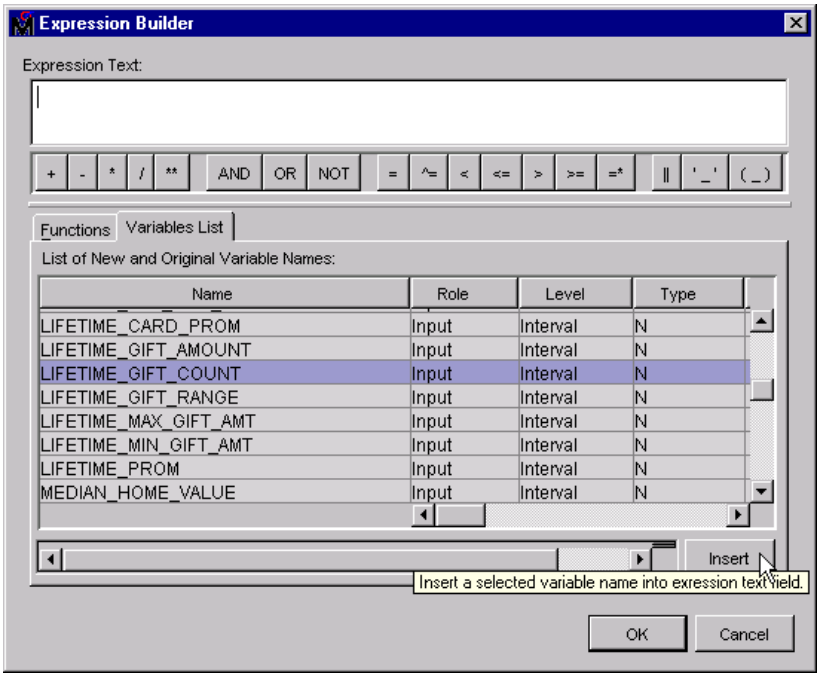


4 Click Build in the Add Transformation window. The Expression Builder window opens.

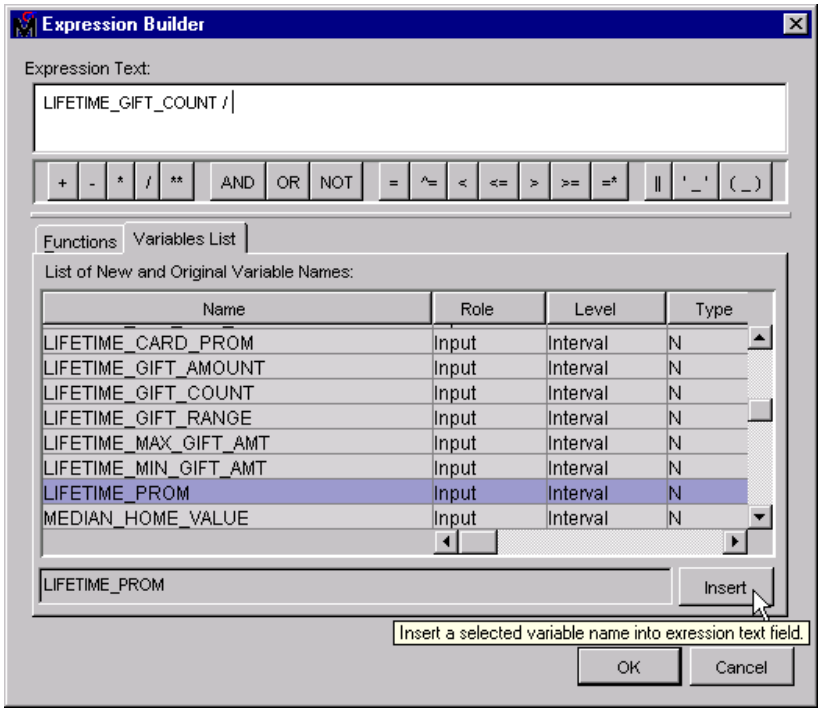


5 Note the comprehensive list of pre-built SAS functions that are available for variable transformations.

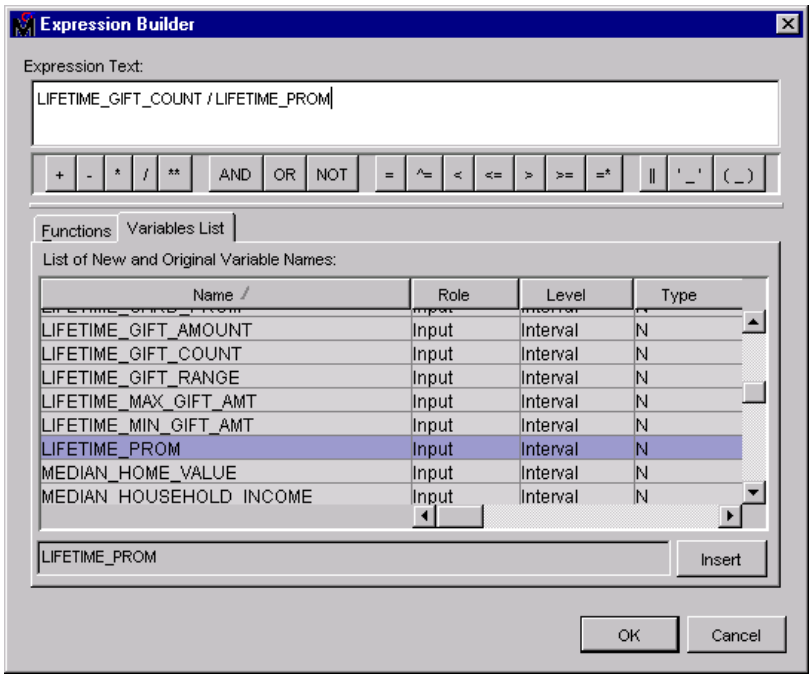
6 Select the **`Variables List`** tab in the Expression Builder window. Scroll down the list of variables to LIFETIME_GIFT_COUNT, select it, and click Insert. The LIFETIME_GIFT_COUNT variable appears in the **`Expression Text`** box.



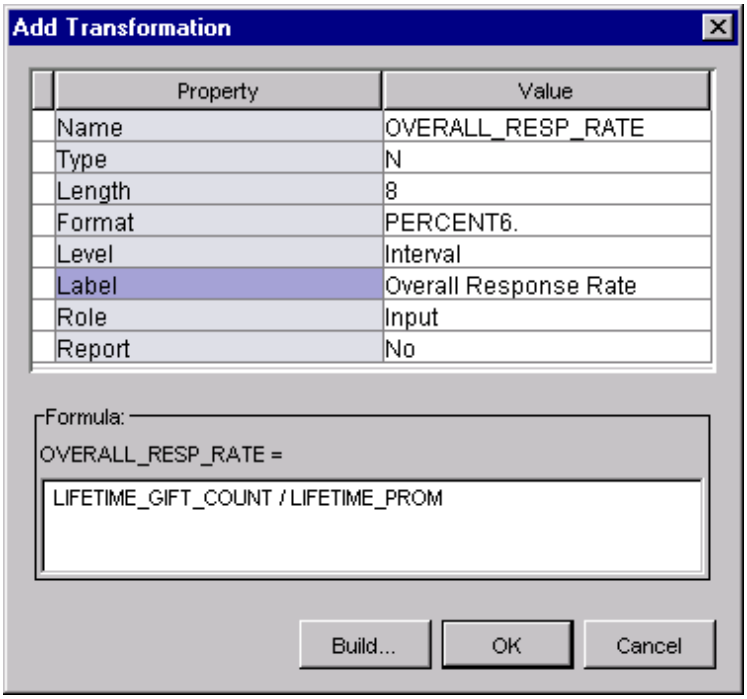
7 Click the division operator button **/** . Return to the **`Variables List`** tab and select the variable LIFETIME_PROM.



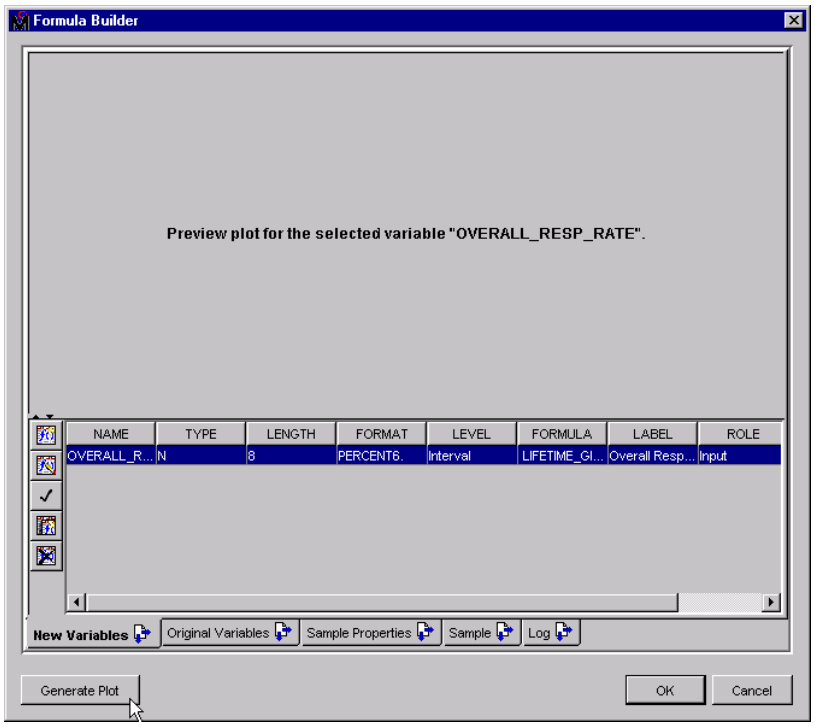
8 Click Insert. The LIFETIME_GIFT_COUNT/LIFETIME_PROM expression appears in the **`Expression Text`** box.



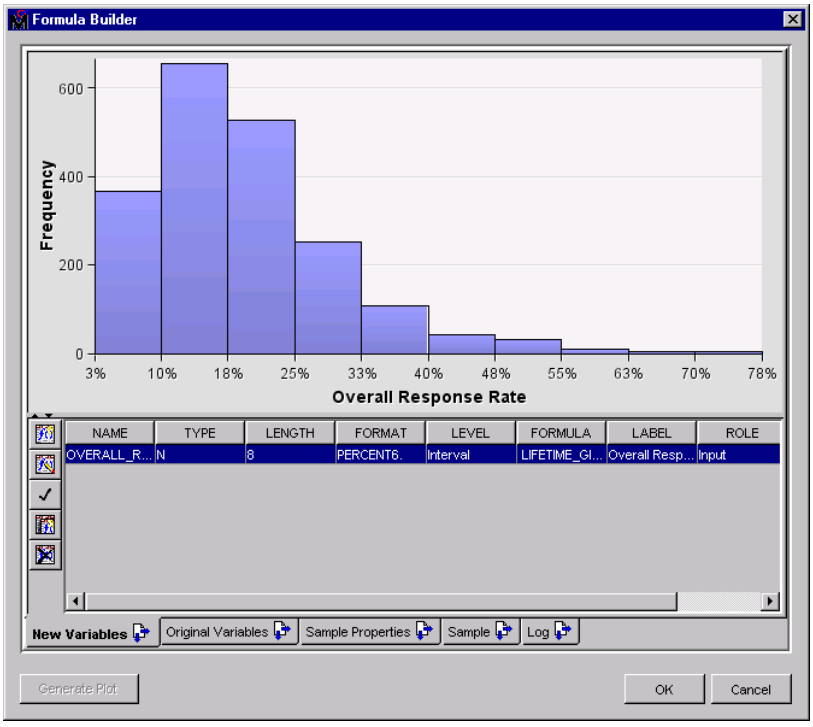
9 Click OK in the Add Transformation window.



10 In the Formula Builder window, click Generate Plot to see a plot of the new variable.



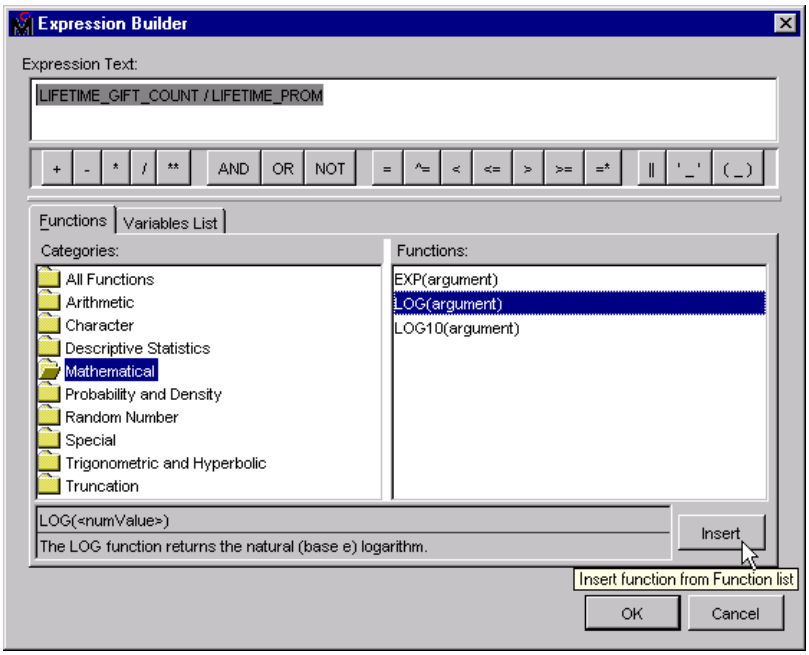
11 Note that because the distribution of OVERALL_RESP_RATE is skewed, you should return to the Formula Builder window to transform it further.



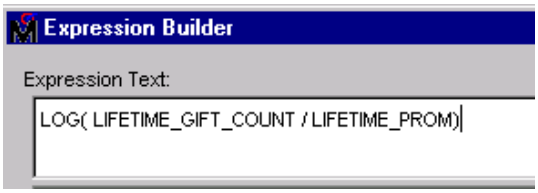
12 Click the Edit Expression button  on the left side of the Formula Builder.

13 Select the LIFETIME_GIFT_COUNT/LIFETIME_PROM expression in the **`Expression Text`** box.

14 On the **`Functions`** tab, select the **`Mathematical`** folder and then select **`LOG(argument)`** from the panel on the right.

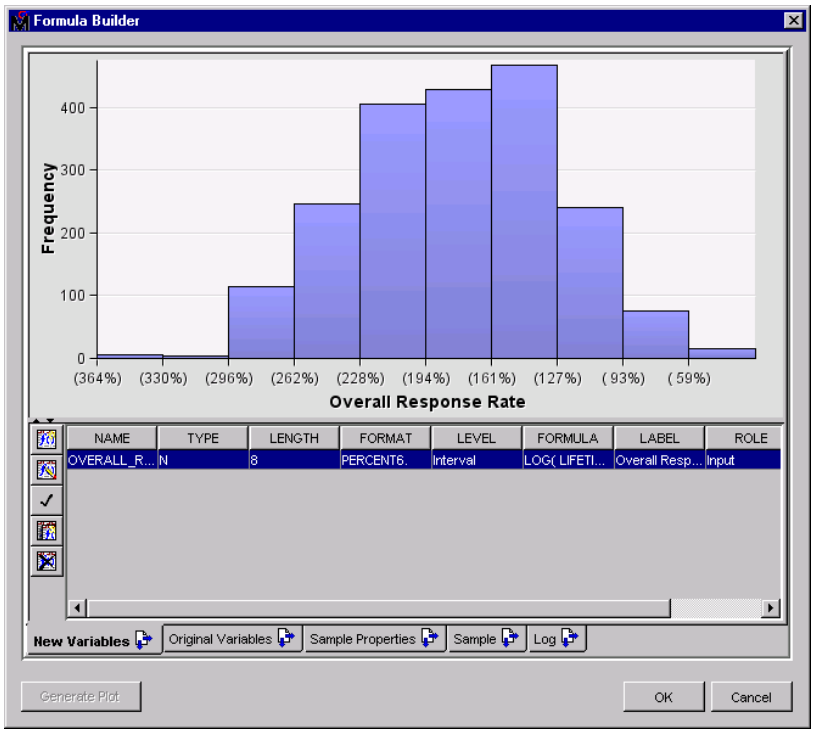


15 Click Insert. The expression text is updated as follows:



16 Click OK in the Expression Builder window.

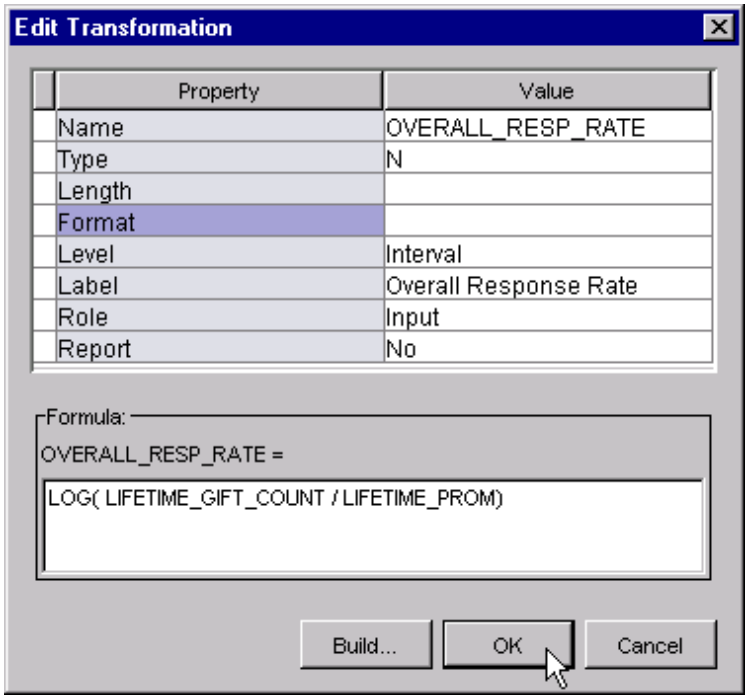
17 Click Refresh Plot at the bottom left of the Formula Builder window.



The distribution is now much closer to a normal distribution.

18 Because the Overall Response Rate variable has been mathematically transformed, the variable's format (PERCENT) is no longer accurate. The variable format requires updating. To change the variable format, click the Edit Properties icon  on the left side of the Formula Builder.

19 In the Edit Transformation window, select **`Format`** and then press the Backspace key to clear the text box. Leave the **`Format`** value blank in order to use the default format for numeric values.



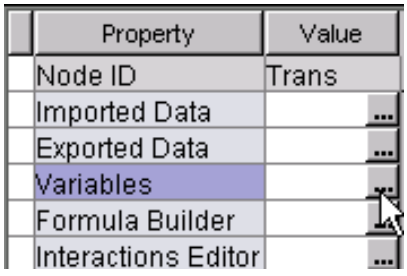
20 Click OK in the Edit Transformation window.

21 Click OK to exit the Formula Builder.

Apply Standard Variable Transformations

You can now apply standard transformations to some of the original variables to modify the distributions so that they more closely resemble a normal distribution. Typical transformations include functions such as logarithmic functions, binning, square root, and inverse functions. The default method for variable transformations for all target and input measurement levels is none, as noted in the Properties panel.

1 To apply transformations to selected variables, click the ellipsis button to the right of the **`Variables`** property in the Transform Variables Properties panel.

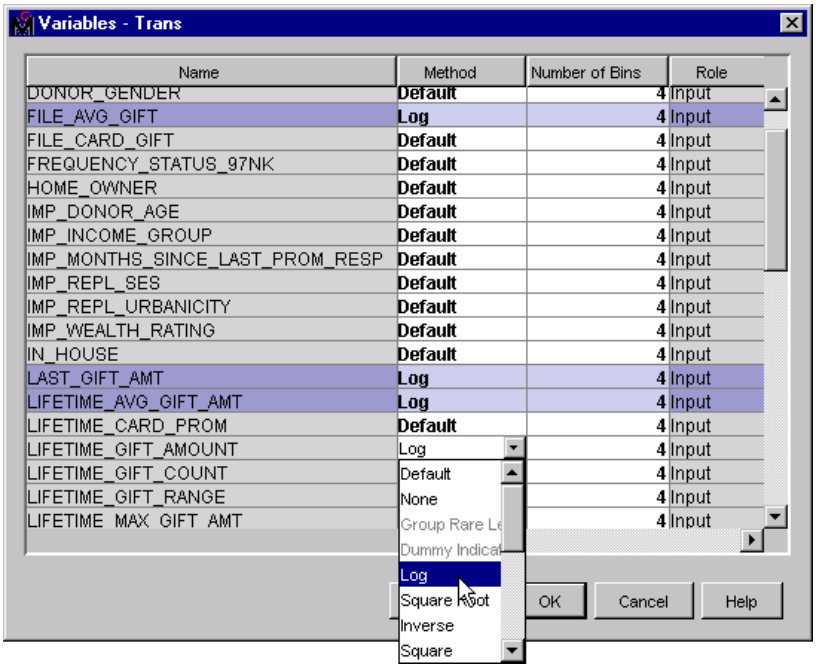


The Variables - Trans window opens.

2 You can transform individual variables in the Variables - Trans window. Apply the **`Log Method`** transformation to each of the following variables:

- FILE_AVG_GIFT
- LAST_GIFT_AMT
- LIFETIME_AVG_GIFT_AMT
- LIFETIME_GIFT_AMOUNT

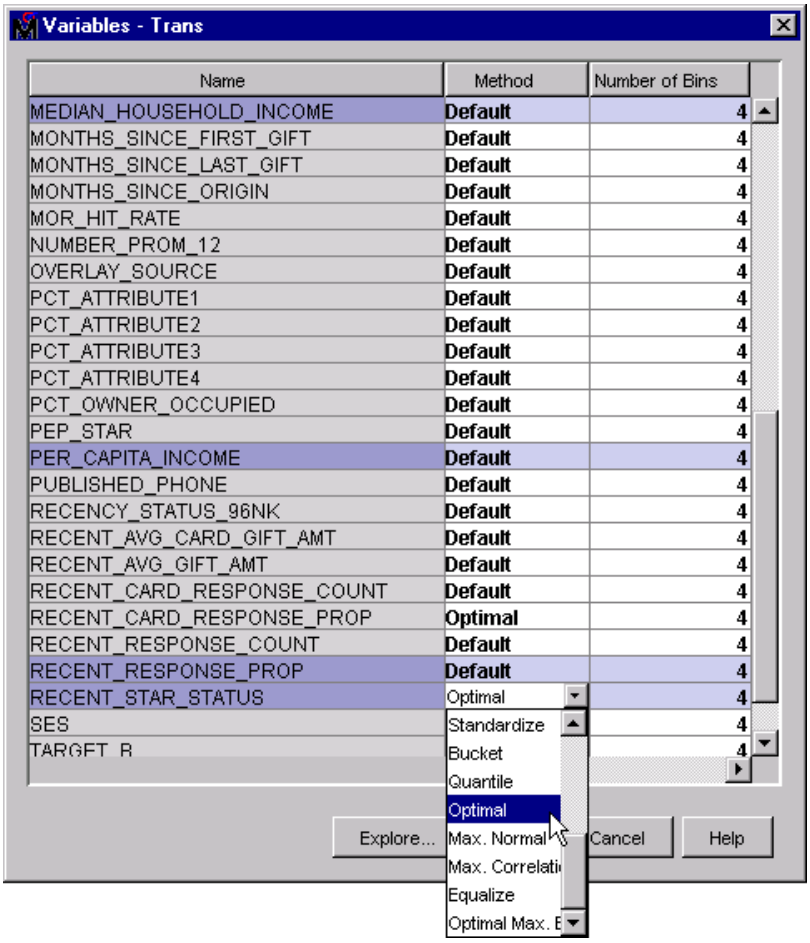
You can highlight adjacent variable rows, or you can hold down the CTRL-key and selecting non-contiguous variables and apply the same transformation to these highlighted variables.



3 Apply the Optimal method to the following variables:

- LIFTIME_CARD_PROM
- LIFETIME_GIFT_COUNT
- MEDIAN_HOME_VALUE
- MEDIAN_HOUSEHOLD_INCOME
- PER_CAPITA_INCOME
- RECENT_RESPONSE_PROP
- RECENT_STAR_STATUS

Note that you can hold down the CTRL key and select multiple variables to change their settings at one time instead of changing each one individually.



4 Select the **Method** column heading to sort the variable rows by the transformation method.

5 Click OK to close the Variables - Trans window.

Note: When Enterprise Miner creates imputed variable values in a data set, the original data set variables remain, but are automatically assigned a Rejected variable status. Rejected variables are not included in data mining algorithms that follow the data imputation step. △

6 Run the Transform Variables node.

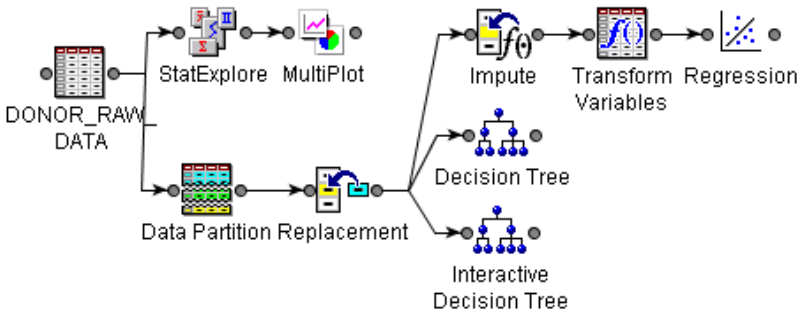
Develop a Stepwise Logistic Regression

Overview

SAS Enterprise Miner provides numerous predictive modeling tools. The Regression node automatically performs either a logistic or ordinary least squares regression, depending on the target measurement level. Like the Decision Tree and Neural Network nodes, the Regression node supports binary, nominal, ordinal, and continuous targets.

This task builds a regression model that uses the partitioned, imputed and transformed DONOR_RAW_DATA data set.

1 Drag a Regression node from the **Model** tab of the toolbar into the Diagram Workspace, and connect it to the Transform Variables node.



Create Histograms of Transformed Variables

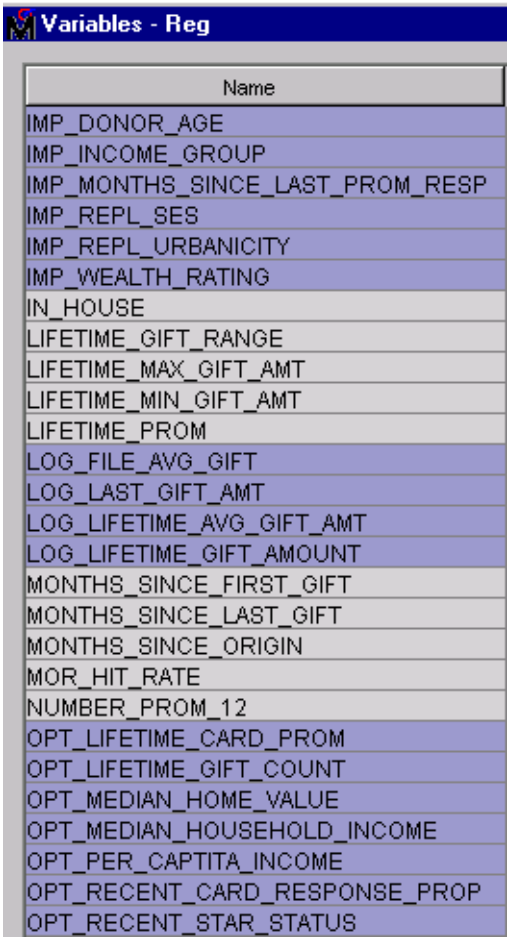
It might be useful to view the distributions of newly transformed variables before you set the properties in the Regression node Properties panel.

1 Select the Regression node in the Diagram Workspace to view the node settings in the Properties panel.

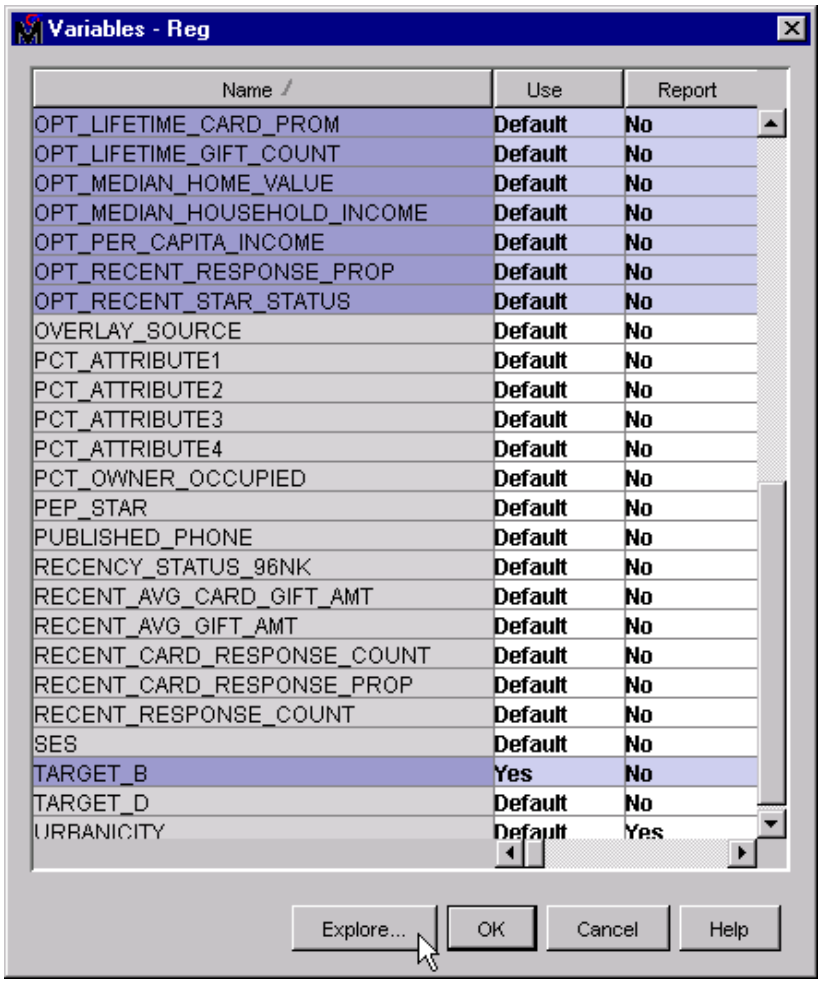
2 Click the ellipsis button to the right of the **Variables** property to open the Variables - Reg window.

The transformed variables that you created begin with variable prefixes LOG_ and OPT_. The imputed variables that you created begin with an IMP_ prefix.

Note: If you do not see these variables, close the Variables - Reg window, right-click the Regression node and select **Update**. △

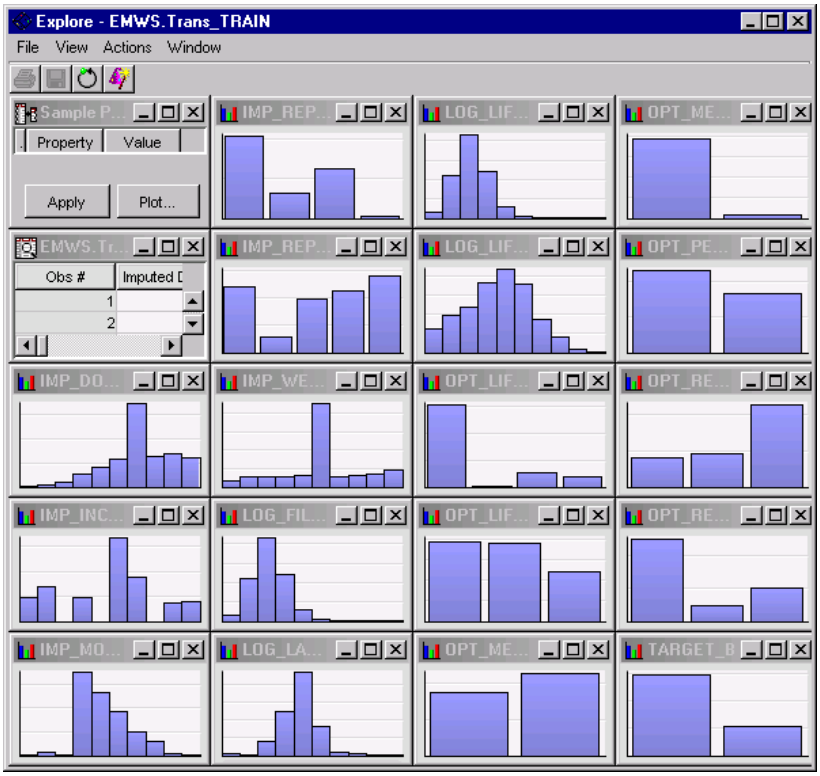


3 Select the TARGET_B variable, as well as variables that have the prefixes IMP_, LOG_, and OPT in order to create a histogram or bar chart of all the transformed variables.



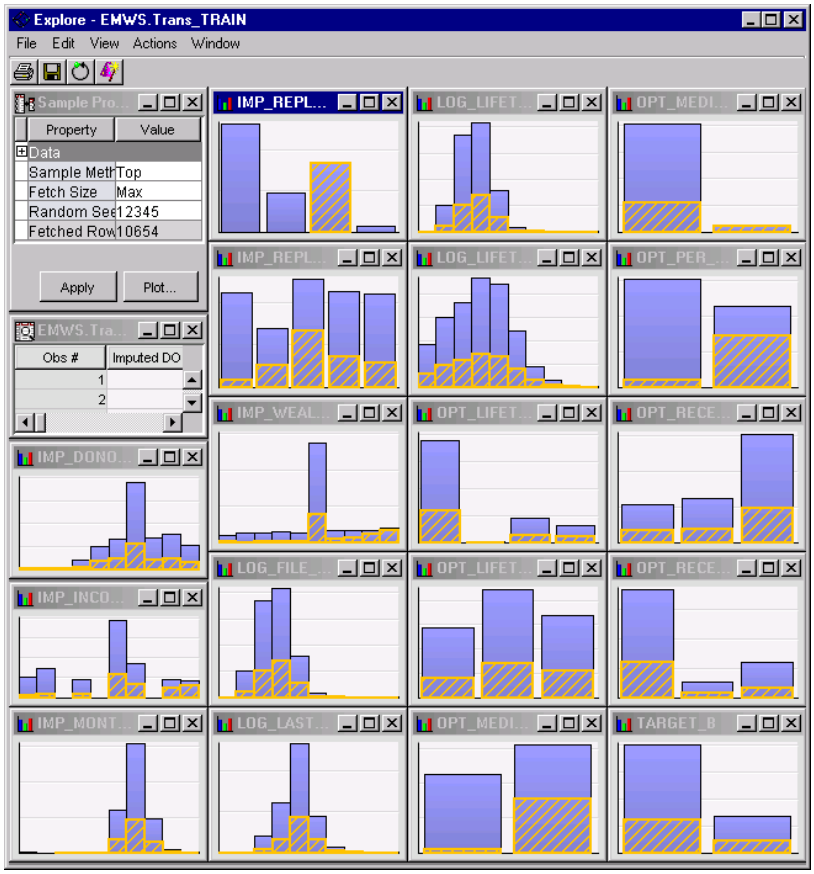
4 Click Explore.

5 Maximize the Explore window. Then from the menu, select **Window** ▶ **Tile** in order to improve the visual layout of the plots.



6 The plots for each variable are created using a sample size of 2,000 observations. In the Sample Properties window, set the **`Fetch Size`** to **`Max`** and then click Apply to plot all 10,654 observations on the client.

7 Select each level of the target variable in order to view how the donors and non-donors are distributed across each of the transformed variables.



Note that some of the heavily skewed variables are more normally distributed after you apply the logarithmic transformation.

8 Close the Explore window and then close the Variables window.

Set Regression Properties

The Regression node can select a model from a set of candidate terms by using one of several methods and criteria. In this task, you specify a model selection criterion that you use during training.

1 Select the Regression node in the Diagram Workspace. In the Regression node Properties panel, set the Regression node **`Selection Model`** property to **`Stepwise`**. This setting systematically adds and deletes variables from the model, based on the **`Entry Significance Level`** and **`Stay Significance Level`**s (defaults of 0.05).

2 Run the Regression node and then view the results. Examine the average profit of the validation data

By default the Regression node displays the following:

- A table of fit statistics for both the training and validation data. Examine the average profit in the validation data.
- A cumulative lift plot (score rankings) across the various deciles for the training and validation data sources. The plot lift values are very consistent for both the training and validation data. You can change the plotting variables on this chart as you did when viewing the lift plot for the decision tree. You can change the vertical axis (Y) to display profit.

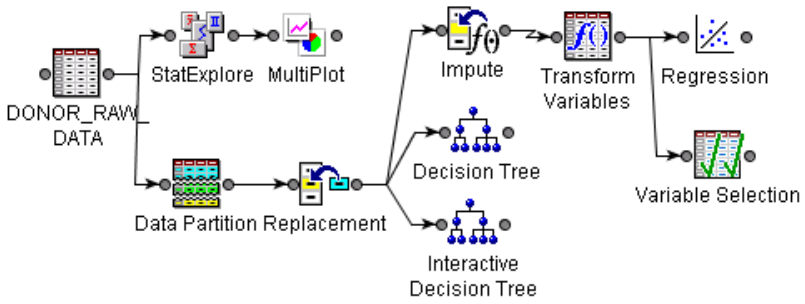
- An effects plot that shows the model effects in order by size. The model effects are sized according to their absolute coefficients. The color of the model effect indicates the sign of the coefficient. When you hold your mouse pointer over the effect bars, you will see that some of the transformed inputs and one of the imputed variables have significant effects in the stepwise selection.
- A detailed output window. The detailed output window provides several statistics in addition to a summary of the stepwise selection process.

3 Close the Results window.

Preliminary Variable Selection

The Variable Selection node can help you reduce the number of inputs to your models by rejecting the input variables that are not related to the target. The remaining inputs can then be modeled by using a more intensive algorithm such as a neural network.

1 Drag a Variable Selection node from the **Explore** tab of the node toolbar into the Diagram Workspace, and connect it to the Transform Variables node.



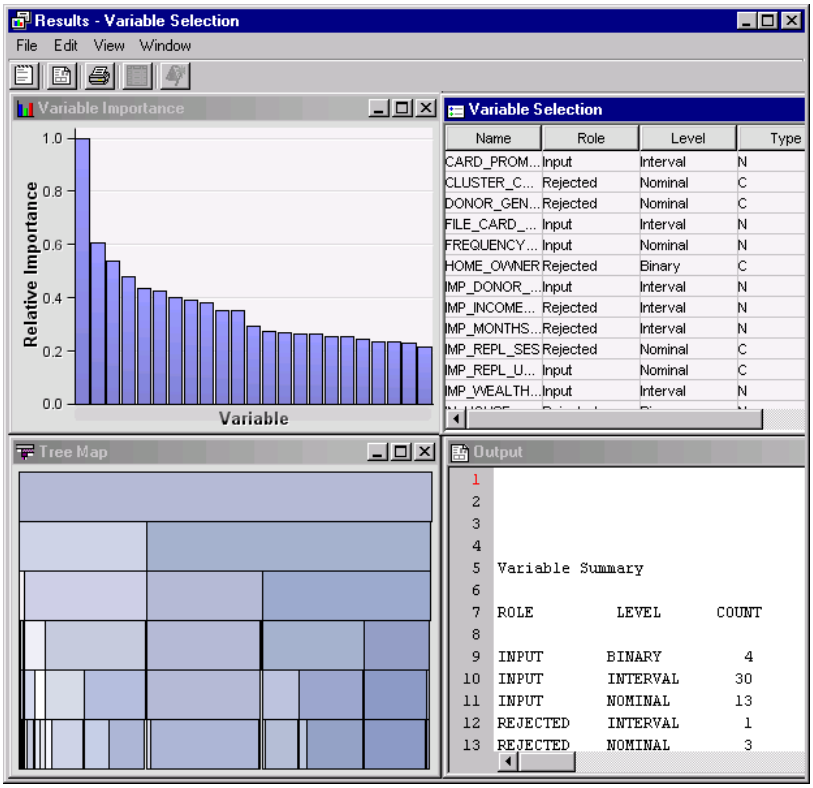
2 Select the Variable Selection node in the Diagram Workspace and set the following **Variable Selection** properties in the Properties panel:

- Set the **Target Model** property to **Chi-Square**.
- Set the **Hide Rejected Variables** property to **No**.

Property	Value
Node ID	Varsel
Imported Data	
Exported Data	
Variables	
Max Class Level	100
Max Missing Percentage	50
Target Model	Chi-Square
Hide Rejected Variables	No
Reject Unused Variables	Yes

3 Run the Variable Selection node and view the Results window.

The Variable Selection table provides a list of the selected variables. The Results window also displays a histogram that is called Variable Importance. The histogram shows each variable's contribution toward a prediction, based on Chi-squared scores. Hold your mouse pointer over a bar in the Relative Importance plot to view its variable name and relative importance score.



4 Close the Results window.

Develop Other Competitor Models

Overview

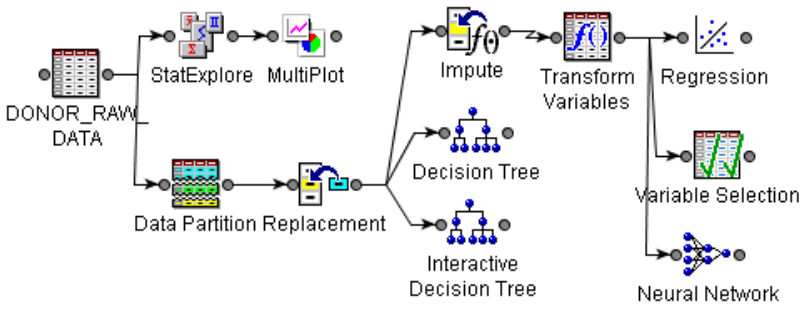
Enterprise Miner enables you to try several different models and then compare the results in one common framework. You can then choose the best model for scoring new data.

One method of arriving at a model is the artificial neural network. The artificial neural network attempts to mimic the actions of the human brain, which is a large organic neural network. Experienced data miners know that artificial neural networks, when carefully tuned, are often very useful in showing nonlinear associations between inputs and the target. Common applications for neural network models include credit risk assessment, direct marketing, and sales predictions.

Add a Neural Network

In this task, you use the Neural Network node to build a neural network model that is based on your data.

1 Drag a Neural Network node from the **Model** tab of the node toolbar into the Diagram Workspace, and connect it to the Transform Variables node.



2 Select the Neural Network node in the Diagram Workspace, then set the following Neural Network properties in the Properties panel:

- Set the **Direct Connection** property to **Yes**.
- Set the **Number of Hidden Units** property to **5**.
- In the **Preliminary Training Options** section, set the **Preliminary Training** property to **Yes**.

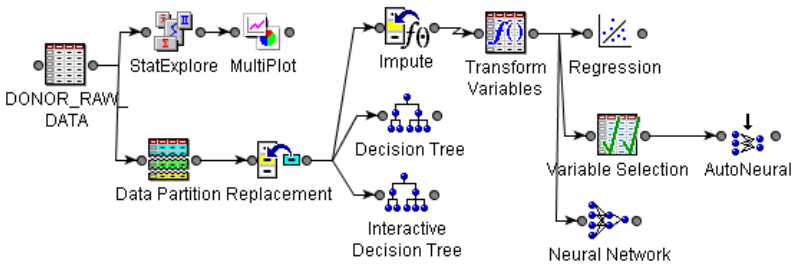
Property	Value
Node ID	Neural
Imported Data	
Exported Data	
Variables	
Use Current Estimates	No
Architecture	MLP
Direct Connection	Yes
Model Selection Criterior	Profit/Loss
Number of Hidden Units	5
⊞ Training Options	
⊞ User Defined Network O	
⊟ Preliminary Training Opti	
Preliminary Training	Yes
Maximum Iterations	10

3 Run the Neural Network node.

Add an AutoNeural Model

The AutoNeural node enables you to find optimal configurations for a neural network model.

1 Drag an AutoNeural node from the **Model** tab of the node toolbar into the Diagram Workspace, and connect it to the Variable Selection node. The combinatorial search that the AutoNeural node performs can be computationally expensive, so you should reduce the input set for training.



2 Select the AutoNeural node in the Diagram Workspace, and then set the following properties in the AutoNeural Properties panel:

- Set the **Architecture** property to **Cascade**, in order to add nodes in a cascade fashion.
- Set the **Train Action** property to **Search**, in order to add nodes according to the Cascade architecture and to find the best topology for the network.

Property	Value
Node ID	AutoNeural
Imported Data	
Exported Data	
Variables	
Architecture	Cascade
Termination	Overfitting
Train Action	Search

3 Run the AutoNeural node.

CHAPTER

6

Working with Nodes That Assess

Overview of This Group of Tasks

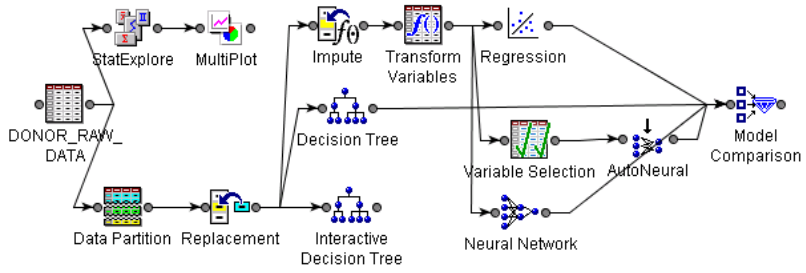
Now that you have created four predictive models and briefly examined assessment statistics for each of them, you can compare the models in order to select the model that best predicts your target.

You use the Model Comparison node to assess how well the Regression, Neural, AutoNeural, and Decision Tree models performed. The comparison is based on the expected profit that would result from implementing the model. In the last set of tasks, you learn how to produce and score new data using the Score node. The scored data identifies the candidates who are most likely to donate money.

Compare Models

In this task, you use the Model Comparison node to benchmark model performance and find a champion model among the Regression, Neural Network, AutoNeural, and Decision Tree nodes in your process flow diagram. The Model Comparison node enables you to judge the generalization properties of each predictive model based on their predictive power, lift, sensitivity, profit or loss, and so on.

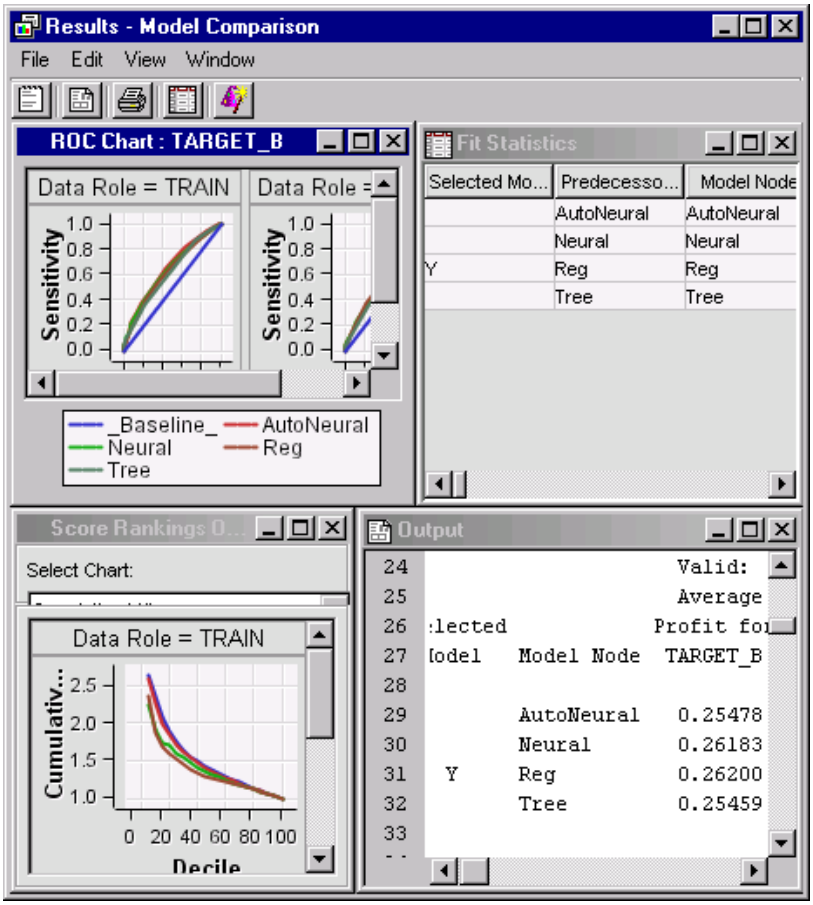
1 Drag a Model Comparison node from the **Assess** tab of the node toolbar into the Diagram Workspace. Connect the Model Comparison node to the Regression, Decision Tree, AutoNeural, and Neural Network nodes as shown below.



2 Right-click the Model Comparison node and select **Run**. A Confirmation window appears. Click Yes.

Note: Running the process flow diagram might take several minutes. △

3 Click Results when the process flow diagram run is complete. The Results window opens.



The Results window displays the following information for a binary target:

- Receiver Operating Characteristics (ROC) charts. The charts overlay the competing models for both the training and validation data (this example does not create a test data set). Each point on the ROC curve represents a cutoff probability. Points closer to the upper-right corner correspond to low cutoff probabilities. Points closer to the lower-left corner correspond to higher cutoff probabilities. The performance quality of a model is indicated by the degree that the ROC curve pushes upward and to the left. This degree can be quantified as

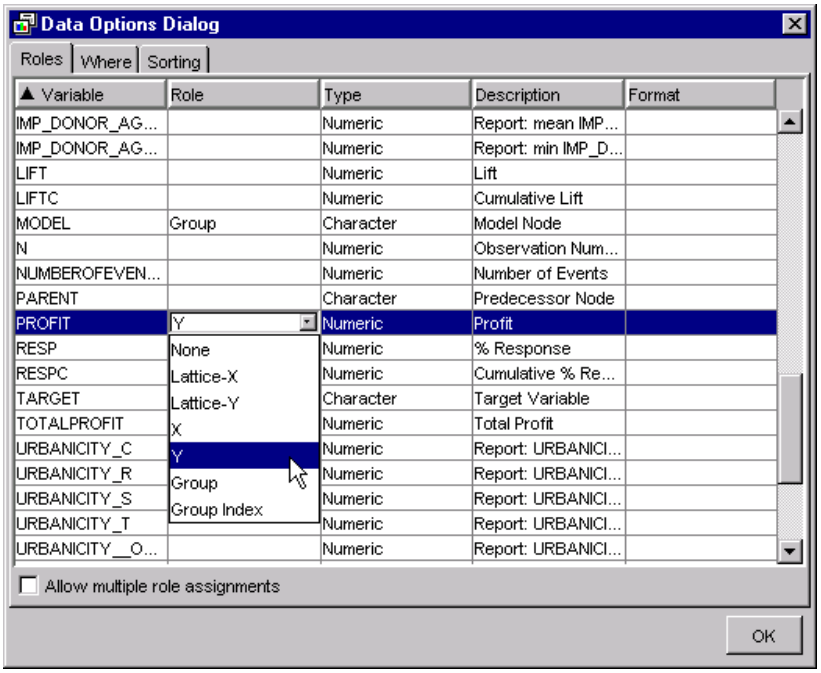
the area under the ROC curve. The area under the ROC curve, or ROC Index, is summarized in the Output window of the Model Comparison node.

- A Score Rankings chart. 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 each model.
- A detailed listing of model diagnostics. The list is provided in the Output window. In this example, the Regression model is marked **`Y`** as the selected model, because the Regression model maximizes the average profit when applied to the validation data. Maximizing average profit is the default criterion for choosing the best model when a profit matrix is defined and when validation data is available. The scoring formula for this model will automatically be passed to the successor Score node for scoring new data.

  Note: You can also use the Fit Statistics window to determine the champion model. The champion model displays a **`Y`** in the **`Selected Model`** column of the Fit Statistics window. △

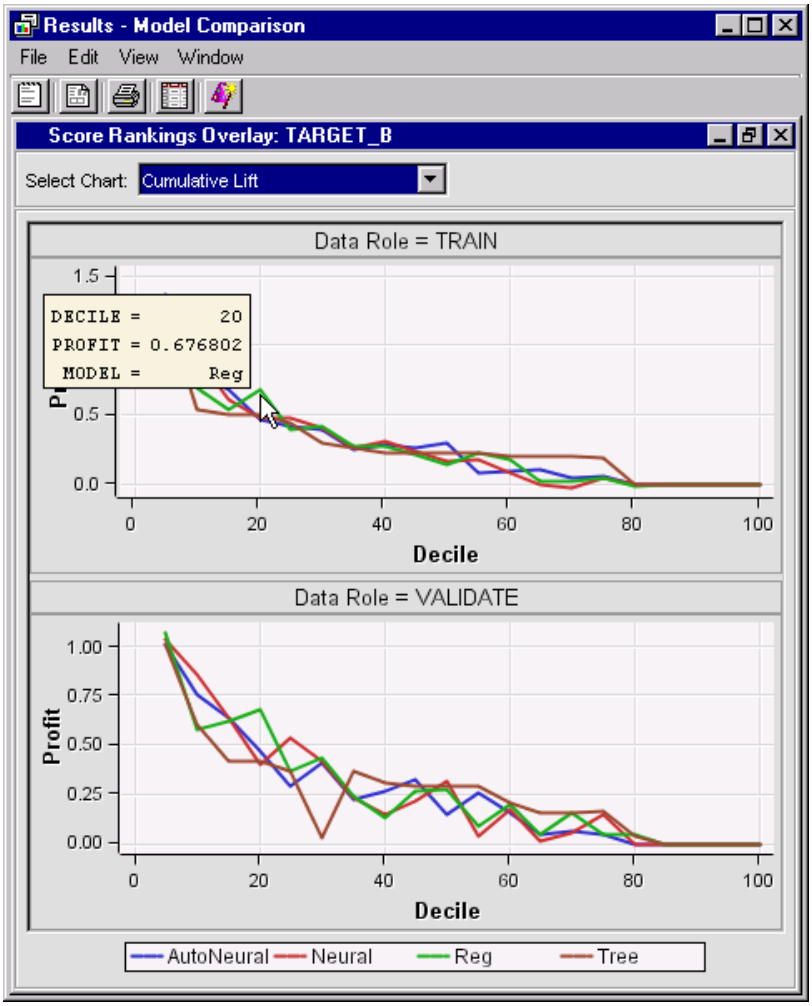
4 Change the vertical axis statistic on the Score Rankings Plot to display the predicted profit. Right-click the background of the Score Rankings plot and select **`Data Options`**.

5 In the Data Options Dialog window, scroll down the list of variables until you see the variable Profit. Change the **`Role`** of Profit to **`Y`**.



6 Click OK.

7 Move your mouse pointer over the different points for each model (line) in the Score Rankings plot to see the predicted profit values.



When you examine the Score Ranking Overlay charts, the Autoneural model captures the most profit in the first decile of the training data, but the same is not true for the validation data. This is also not true for the other deciles. Note that the maximum profit for the models varies across the deciles, but overall the Regression model maximizes total and average profit. The Results window provides the same assessment charts and reports as the Results windows of the individual modeling nodes.

8 Close the Results window.

Score New Data

Overview

The final step in most data mining problems is to create scoring code that you can use to score new data. For example, now that you have a good predictive model for profitable donations, you can apply that model to raw data that does not include the

target variable TARGET_B. Thus you can automate the model scoring process of deciding which individuals are likely to donate.

There are several types of scoring including interactive, batch, and on-demand. Typically, interactive scoring is performed on smaller tables, while batch scoring is performed on larger tables. On-demand scoring includes single transactions that can be performed in a real-time setting. There are also different types of score code. By default, the score code that SAS Enterprise Miner creates is SAS code. It is also possible to create Java or C code for integration into environments other than SAS. Additionally, SAS Enterprise Miner can create PMML code, which is an XML representation of the model for scoring in other databases. In this topic, you will perform interactive scoring that produces SAS code.

You use the Score node to manage, edit, export, and execute scoring code that is generated from a trained model or models. The Score node generates and manages scoring formulas in the form of a single SAS DATA step, which can be used in most SAS environments even without the presence of SAS Enterprise Miner.

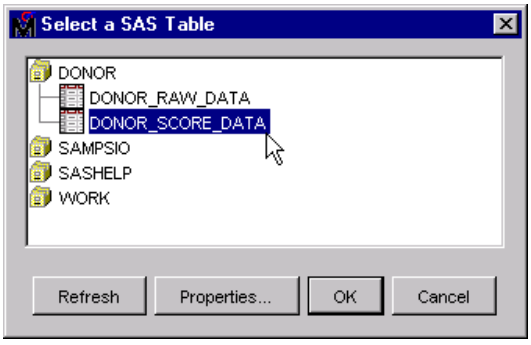
In this example you use the Score node to score a score data set within the process flow.

Define Data Source for Scoring

In order to add a data set to an Enterprise Miner process flow diagram, you must define it as a data source first.

In this task, you define the DONOR_SCORE_DATA as a data source.

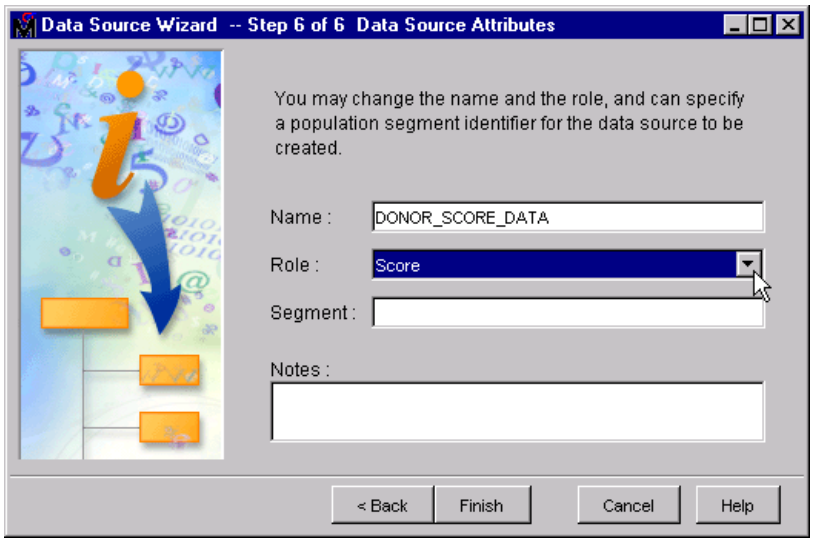
1. Right-click the Data Sources folder in the Project Panel and select **`Create Data Source`**. The Data Source wizard opens.
2. In the **`Source`** box, select SAS Table. Click Next.
3. In the Data Source Wizard – Select a SAS Table window, click Browse.
4. In the Select a SAS Table window, double-click the DONOR library folder to expand it. Select the DONOR_SCORE_DATA table and click OK.



DONOR.DONOR_SCORE_DATA appears in the **`Table`** box of the Select a SAS Table window. Click Next.

5. Click Next in the Table Information window.
6. Select **`Basic`** in the Metadata Advisor Options window. Click Next. The Column Metadata window opens.
7. Redefine the metadata by setting the **`Role`** for the CONTROL_NUMBER variable to **`ID`**.
8. Click Next. The Data Source Attributes window opens.

9 In the **Role** box, select **Score** from the list to indicate that this data set contains Score data.

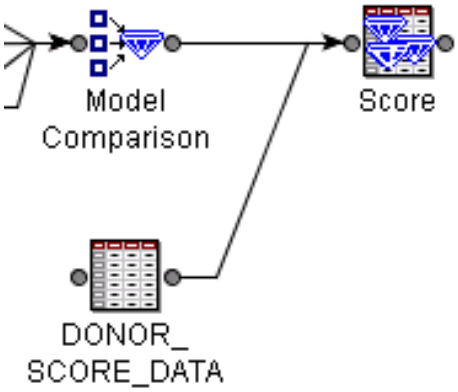


10 Click Finish. The DONOR_SCORE_DATA data source appears in the Data Sources folder in the Project panel.

Add Score Data and Score Node to Diagram

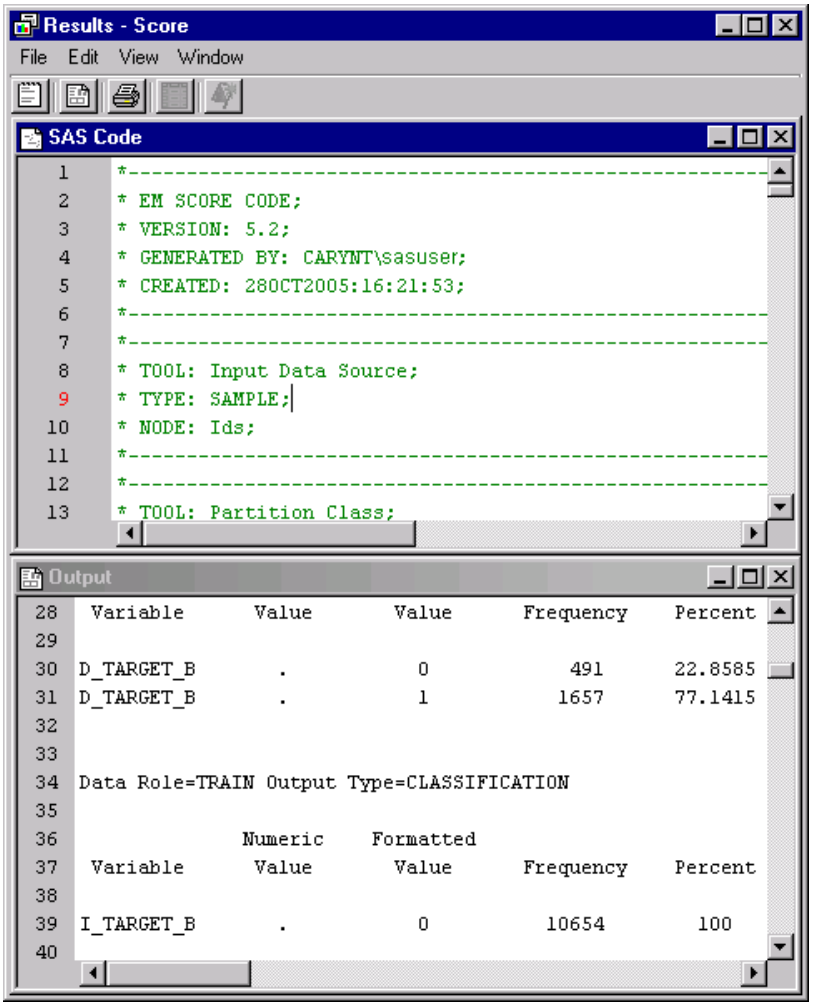
1 Drag the DONOR_SCORE_DATA data source from the DONOR library folder in the Project panel into the Diagram Workspace. Place it near the Model Comparison node.

2 Drag a Score node from the **Assess** tab of the node toolbar into the Diagram Workspace. Connect both the Model Comparison node and the DONOR_SCORE_DATA data source to the Score node.



3 Run the Score node in order to apply the SAS scoring code to the new data source.

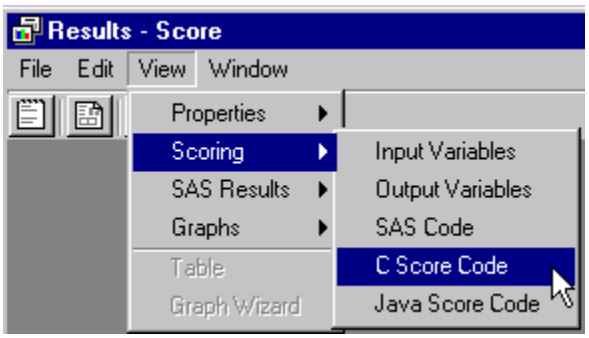
4 View the Score node results when the node has finished running.



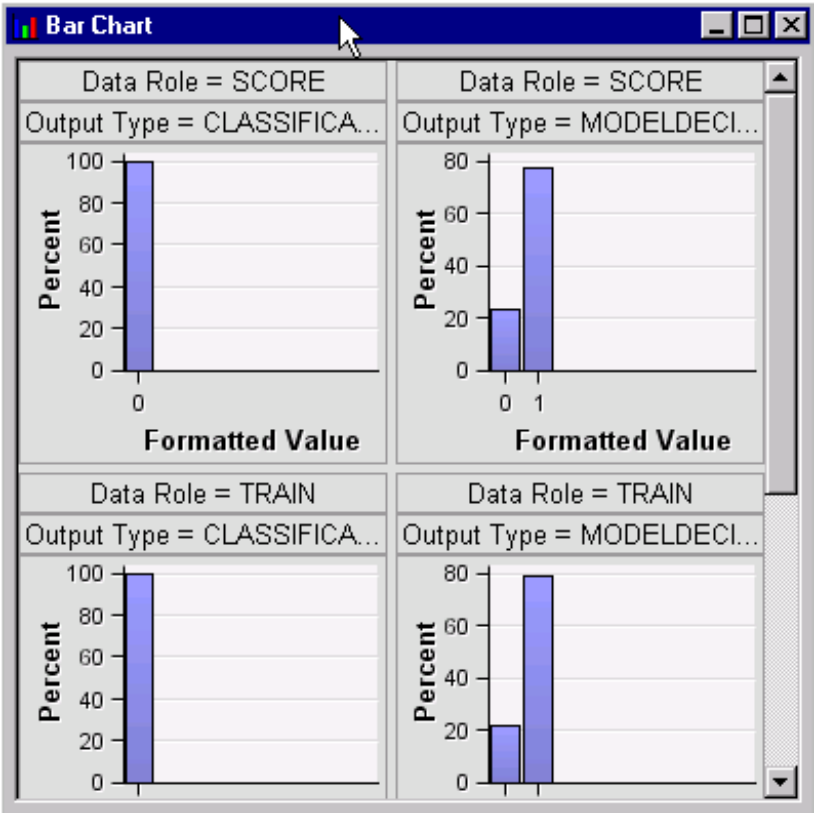
As you examine the results, notice these details:

- The SAS Code window displays code that was generated by the entire process flow diagram. The SAS score code can be used outside the Enterprise Miner environment for custom applications. The results also contain C and Java translations of the score code that can be used for external deployment.
- The Output window displays summary statistics for class and interval variables. You can also view lists of the score input and output variables.

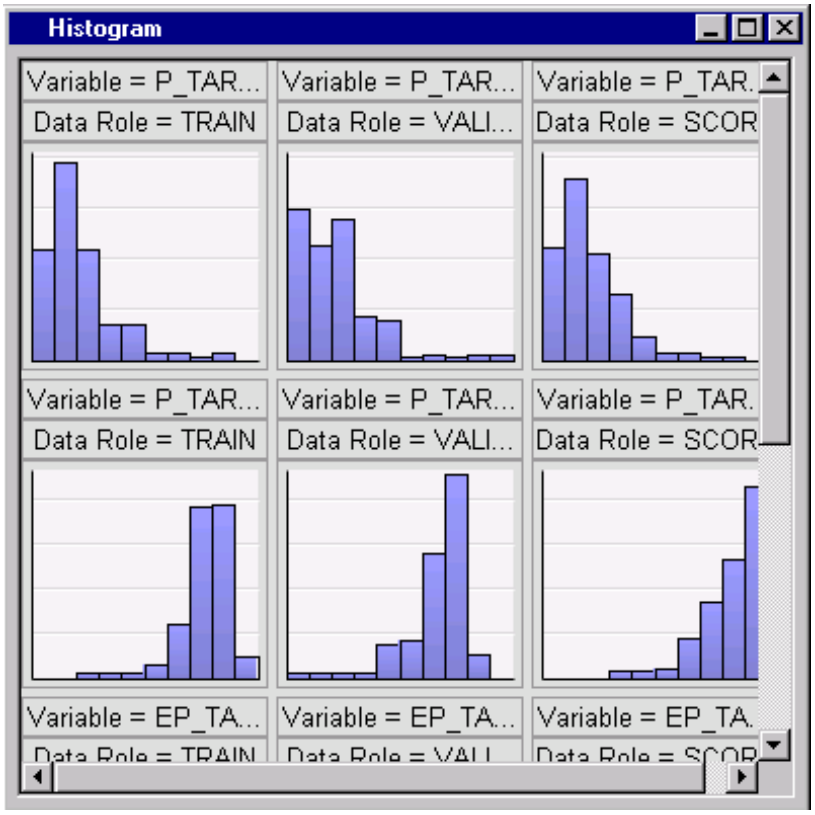
5 Select **View** ▶ **Scoring** in order to view the SAS, C, and Java score code.



6 Select **View** ▶ **Graphs** ▶ **Bar Chart** in order to display a bar chart of the values of the target variable for classification, decision, and segment output types, if applicable, for each data set.

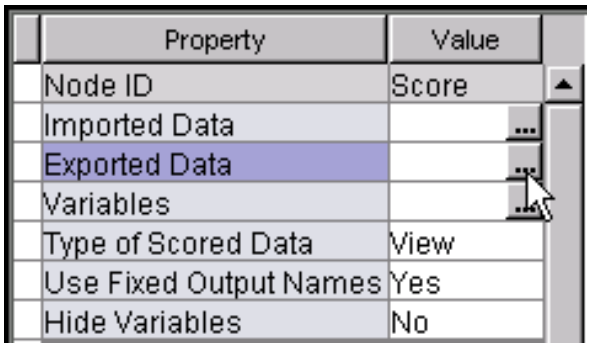


7 Select **View** ▶ **Graphs** ▶ **Histogram** in order to display a histogram of the values of the predicted, probability, and profit output types for each of the data sets.

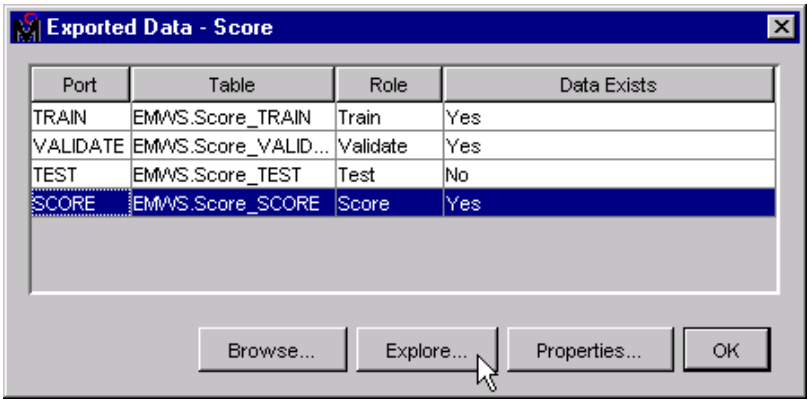


8 Close the Results window.

9 Click the ellipsis button to the right of the **`Exported Data`** property in the Score node Properties panel in order to view the scored data.

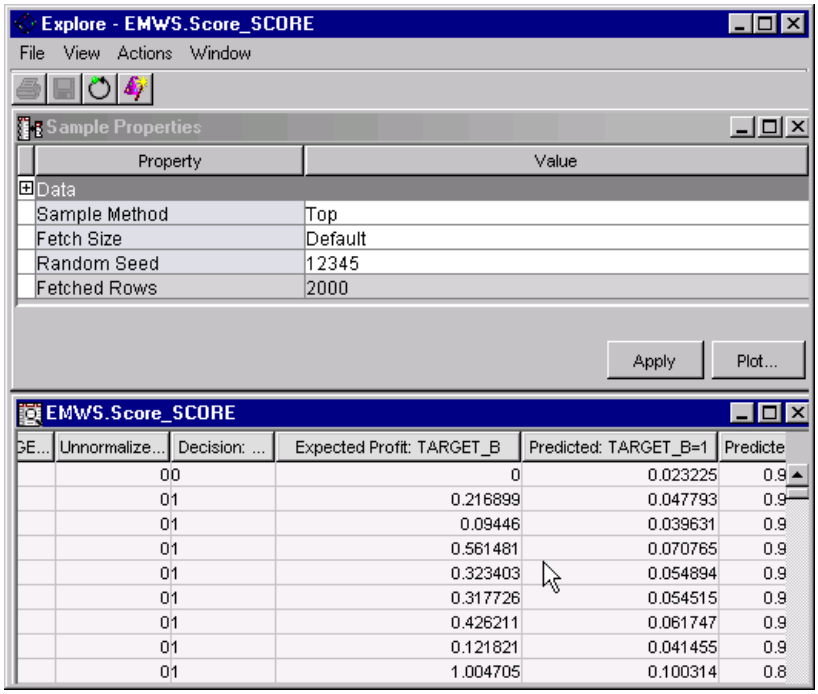


10 Select the SCORE port table in the Exported Data - Score window.



11 Click Explore.

12 Examine the SCORE table. Values for predicted profit, expected profit, and other variables were generated by the Score node for export.



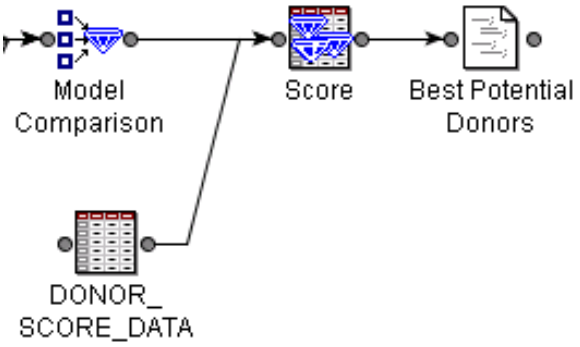
13 Close the Score table. Click OK to close the Exported Data window.

Add a SAS Code Node

1 Drag a SAS Code node from the **Utility** tab of the nodes toolbar into the Diagram Workspace, and connect it to the Score node.

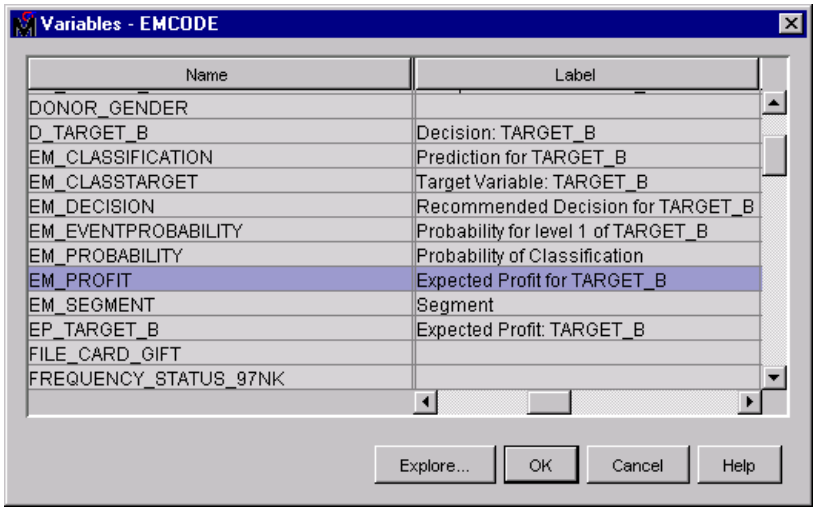
2 Right-click the SAS Code node and select **Rename**.

3 Type **Best Potential Donors** on the **Node name** box.



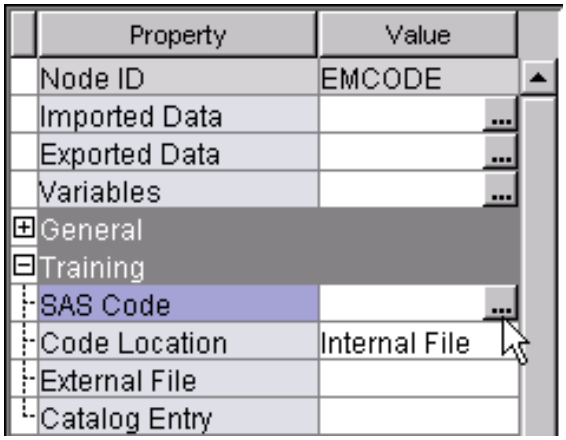
4 In the SAS Code node Properties panel, click the ellipsis button to the right of the **`Variables`** property to open the Variables table. The name of the average profit variable is EM_PROFIT and the name of the decision variable is EM_DECISION.

5 Scroll to the right to see the **`Label`** column. You can widen the column to view its entire contents.

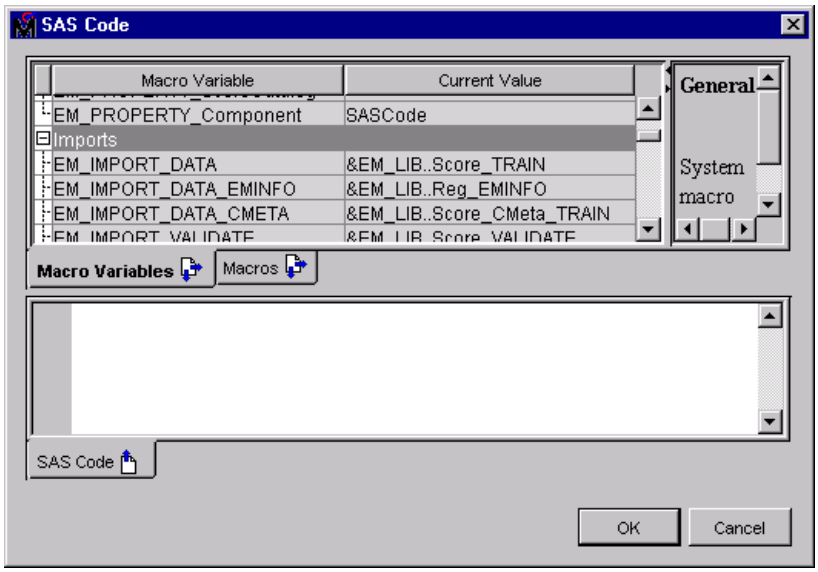


6 Close the Variables window.

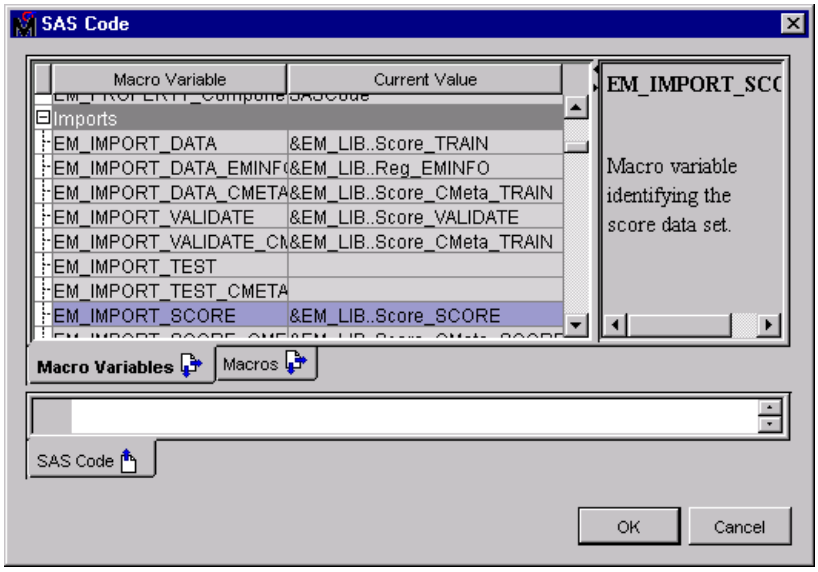
7 In the SAS Code node Properties panel, click the ellipsis button to the right of the **`SAS Code`** property to open the SAS Code window.



The SAS Code window has two tabs: **Macro Variables** and **Macros**. These tabs contain the system-defined macro variables and their value if these are already assigned, and a list of macros provided by SAS.



8 Select the **Macro Variables** tab. The tab holds a list of macros that you can reference in your SAS code. Examine the Imports section of the **Macro Variable** list. A macro variable named EM_IMPORT_SCORE appears in this section. You can use the EM_IMPORT_SCORE macro variable to reference the score code that you import into the SAS Code node.



9 On the **SAS Code** tab, enter the following code:

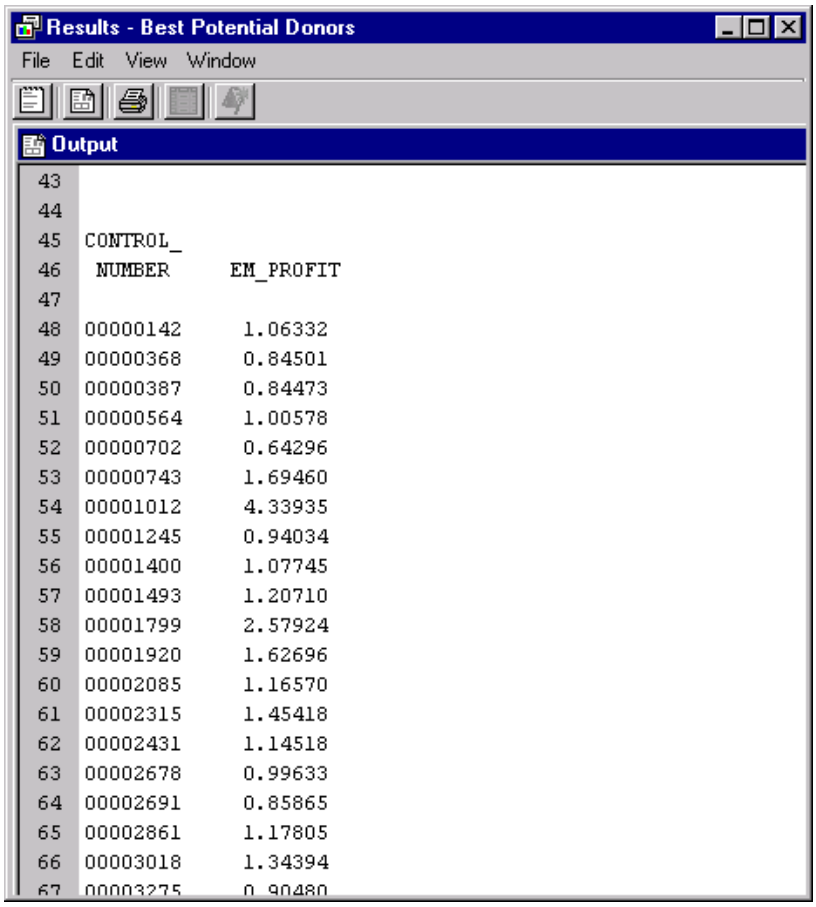
```
proc means data=&em_import_score n min mean median max;
class em_decision;
var em_profit;
run;
```

```
proc print data=&em_import_score noobs;
var control_number em_profit;
where em_profit gt .60;
run;
```

Note: The PROC MEANS step calculates descriptive statistics for expected profit, and the PROC PRINT step generates a list of donors that exceed an expected profit threshold. △

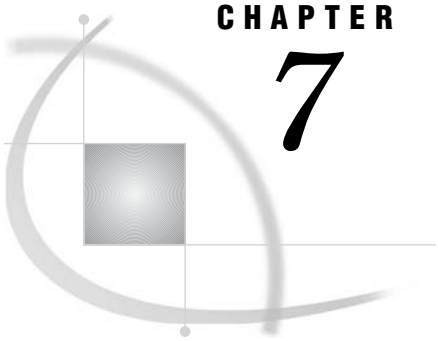
10 Click OK to close the SAS Code window.

11 Run the SAS Code node named Best Potential Donors and view the results.



12 Select **View** ▶ **SAS Results** ▶ **Log** in order to view the SAS log.

13 Close the Results window.



CHAPTER

7

Sharing Models and Projects

Overview of This Group of Tasks

You have fit and built a predictive model using training and validation data. You have used the model to score new data. What are some potential next steps for the scored data?

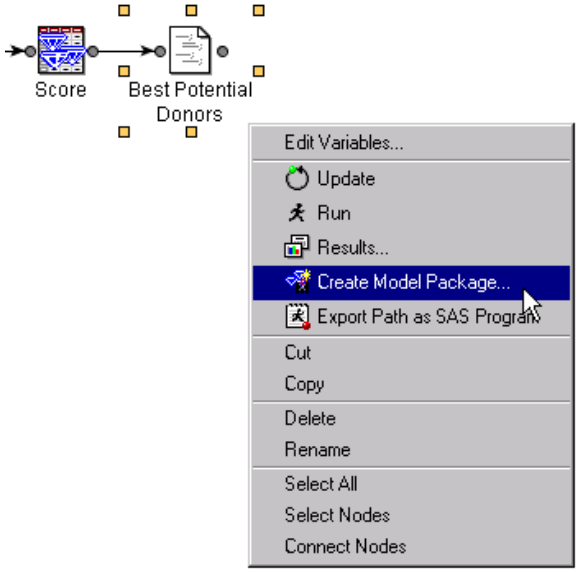
You can place your scored data in a cube for Online Analytical Processing (OLAP) exploration, or use it as input for reports or further analysis. You might want to save the model in an archive so you can share it with other users. There are a number of ways you can save your data mining model for future use.

One of the ways that you might save your model is with model packages. With Enterprise Miner, you can create model packages and share them with fellow data miners, business managers, and data managers throughout the organization. Besides enabling you to share models, model packages also provide an audit trail of the underlying data mining processes. With packages, process flow diagrams can be archived, stored, and re-used. The Enterprise Miner model package also includes an XML file that completely defines the process flow diagram layout and configuration for each component node.

Create Model Packages

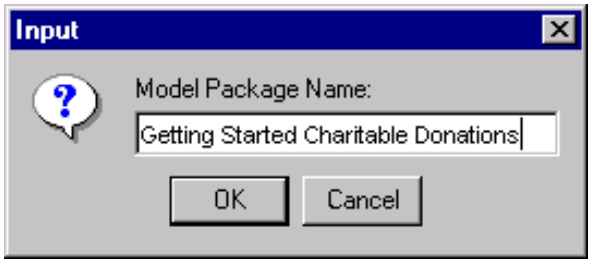
To create a model package:

1 Right-click the SAS Code node named Best Potential Donors, in the Diagram Workspace, and select **`Create Model Package`**.



The model package that you create contains the output and results for each node that precedes the node you select. Use the terminal node in a process flow diagram to capture the entire diagram in a model package.

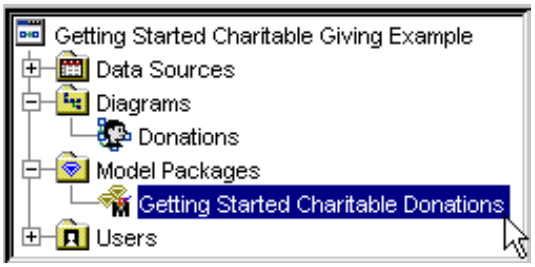
2 Type a model package name in the Input window.



3 Click OK to create the package. In the Diagram Workspace, the SAS Code node is highlighted in blue while the model package is being created.

4 Click OK in the Run Status window.

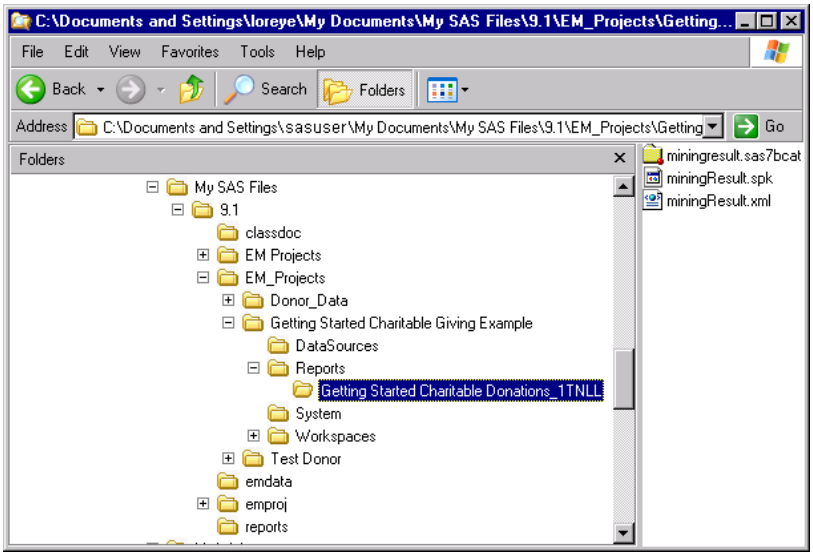
The report is stored inside the Model Packages Folder of the Project panel.



To store a model package:

a Find your EM Projects folder on your local machine or on your allotted server space. Each Enterprise Miner project that you create is stored in a folder under the EM Projects folder.

b Open your Getting Started Charitable Donations project folder and locate a Reports folder. The Reports folder is the default location for the model packages that you create.



c Examine the model package folders. They contain the following files:

- miningresult.sas7bcat — the SAS catalog that contains a single SLIST file and metadata from the process flow diagram.
- miningResult.spk — the model package SPK file.
- miningResult.xml — an XML file that contains metadata about the process flow diagram. This file contains the same information as miningresult.sas7bcat.

d Right-click a model package in the Project panel of the workspace to open it, register it to the Web-based model repository, re-create the diagram and save it as another package, or to delete it.

About SAS Package (SPK) Files

A SAS Package (SPK) file is a container file to which modeling results are exported. SPK files are compressed and can contain any of the following file types:

- SAS files
- SAS catalogs
- SAS data sets
- SAS databases (such as DMDB, FDB, and MDDB)
- SAS SQL views
- Binary files (such as Excel, GIF, JPG, PDF, PowerPoint, and Word)
- HTML files (including ODS output)
- References strings (such as a URL)
- Text files (such as a SAS program)

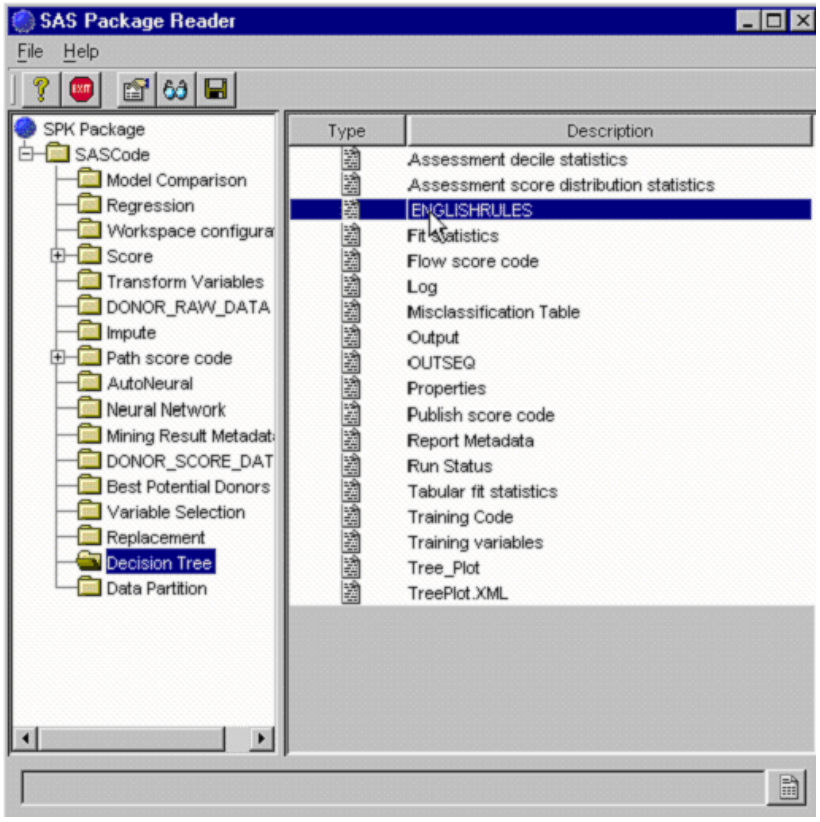
You use the SAS Package Reader to view the SPK files. For more information about the SAS Package Reader, see **`http://support.sas.com/rnd/itech/doc/reader/index.html`**.

Use the SAS Package Reader to View Model Results

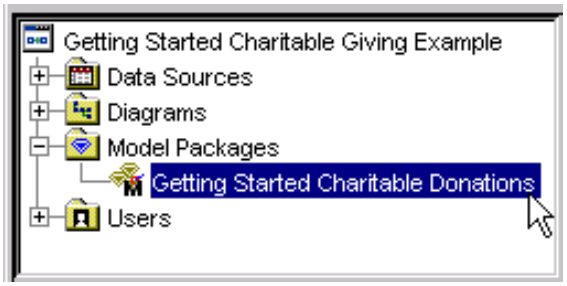
You can download the SAS Package Reader, from the SAS Web site. To download the software:

1. Direct your Web browser to **`http://support.sas.com`**.
2. Select **`Software Downloads`**.
3. Select **`SAS Integration Technologies`**.
4. In the **`For SAS®9`** category, select **`SAS Package Reader`**.

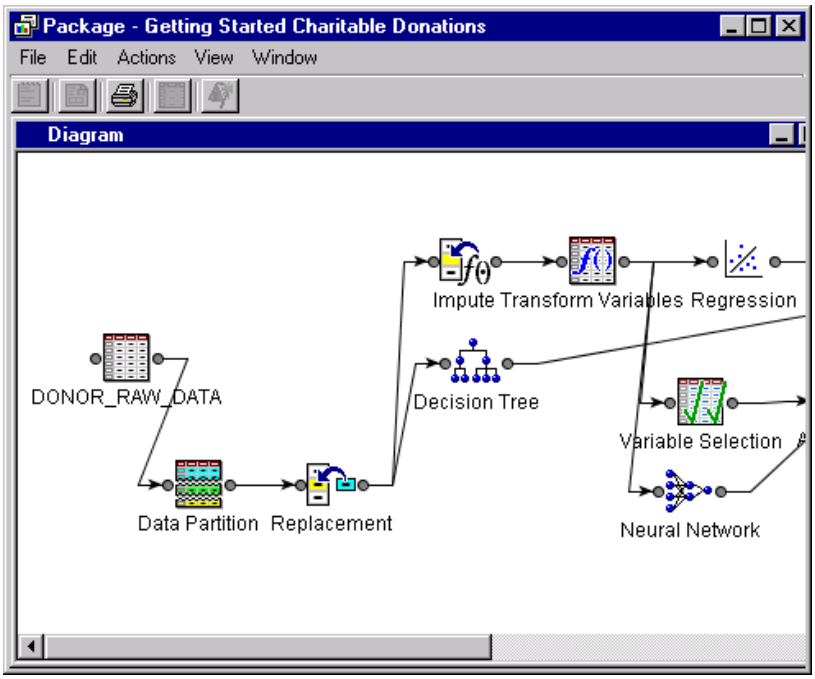
   The SAS Package Reader displays a list of the entries within a package. It contains a built-in viewer for SAS data set entries and it launches a Web browser for all other entry types. You can use the SAS Package Reader and SPK files to share projects and reports with those who do not have SAS installed on their machines.



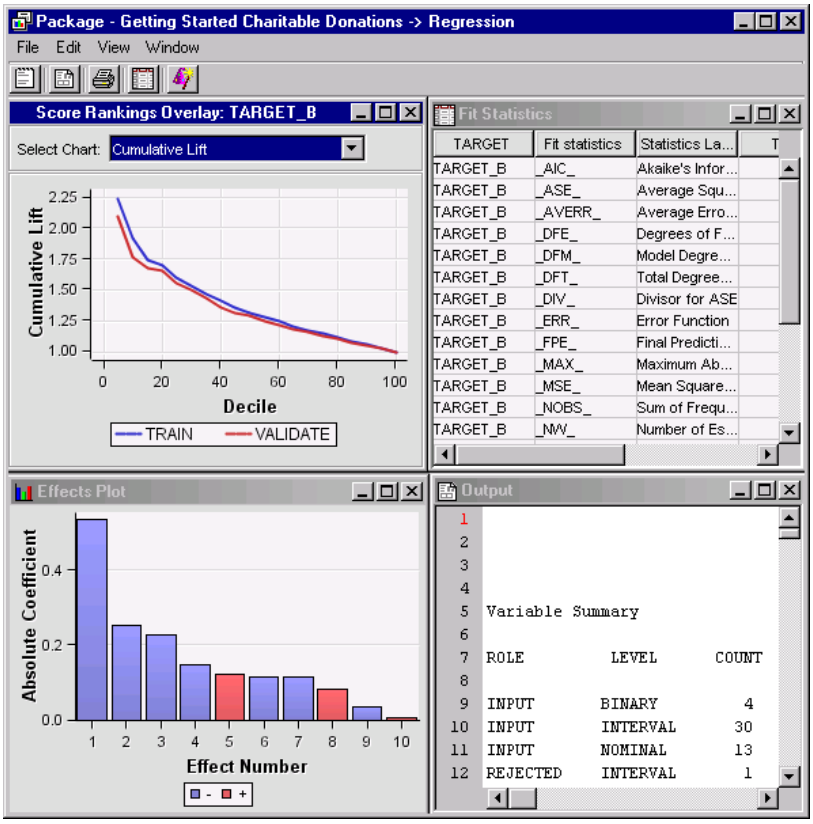
5 To open a package directly from Enterprise Miner, expand the Model Packages folder and double-click the model package that you want.



The package opens and displays the process flow diagram.



6 Double-click the Regression node to open the Results window.

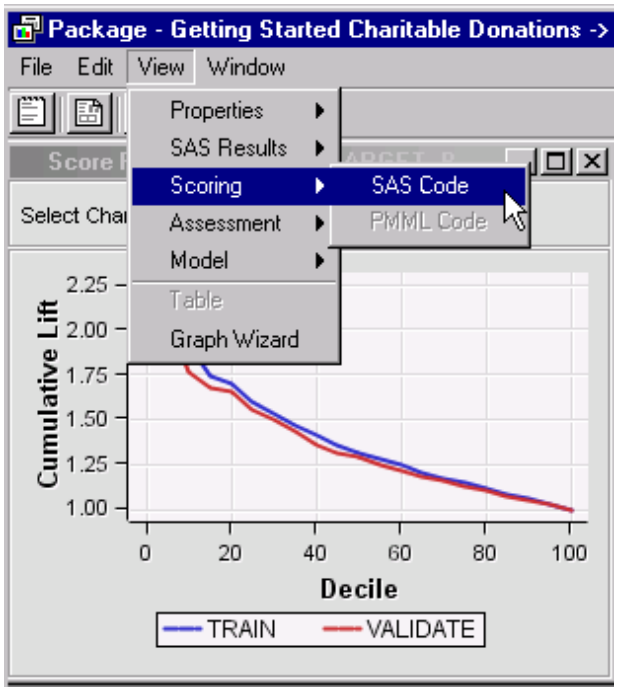


7 Close the Results window.

View the Score Code

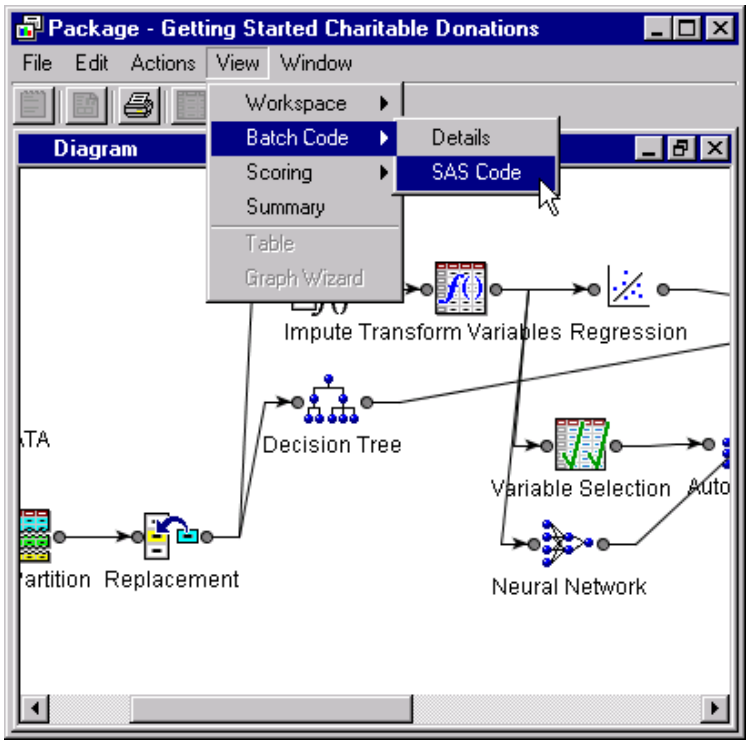
If you ran a node that generated SAS, C, Java, or PMML score code, you can view the code from the node's Results window.

1 From the Results window main menu, select **View** ▶ **Scoring** ▶ **SAS Code** .



Note: Enterprise Miner includes a set of SAS macros that you can use to run process flow diagrams during off-peak computing hours, in batch mode. Batch processing code that recreates the process flow diagram and its settings is generated when you create a model package. You can use the Enterprise Miner graphical user interface to view and interact with results that were generated by process flow diagrams that were executed in batch mode. △

2 Select **View** ▶ **Batch Code** from the window's main menu in order to view the batch code that has been saved in a process flow diagram's model package,



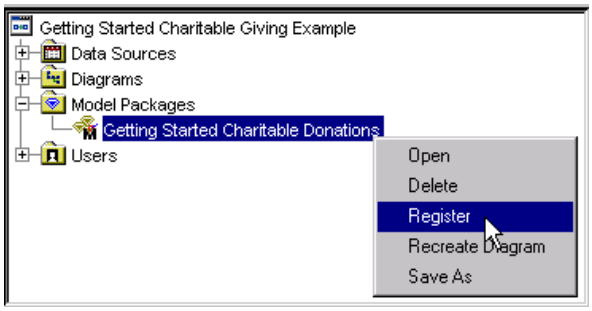
3 Close the Results window.

Register Models

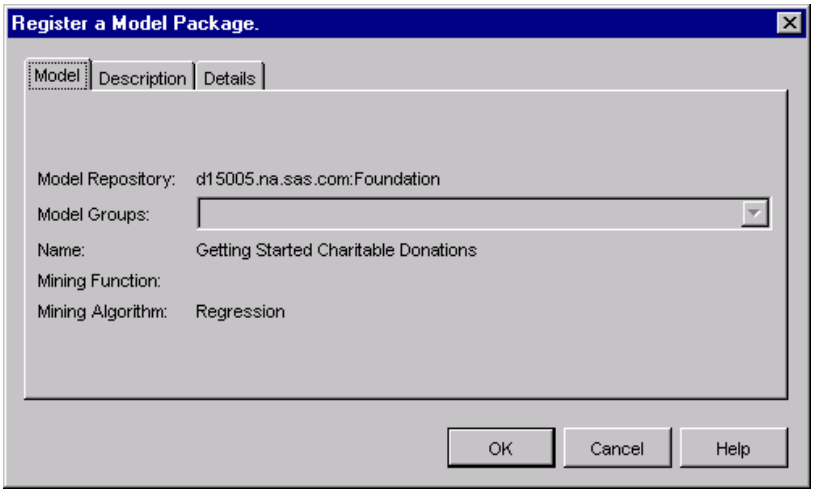
You can archive and register saved models in the Enterprise Miner Model Repository. The repository resides on the SAS Metadata Server. When you register a model, the model details and metadata are stored on the Server. The Server allows data mining models to be shared. The SAS Metadata Sever can be used to distribute models to those who are not Enterprise Miner users.

The first step is to create a model package, as you have just done above. Since you created the model package from the SAS Score node, the champion regression model will actually be registered for retrieval via other SAS interfaces like the Model Manager. To register other models, such as the Decision Tree and Neural Network models, you first need to create a model package for each of them.

1 In the Project panel, open the Model Packages folder and right-click the package that you created. Select **`Register`**.

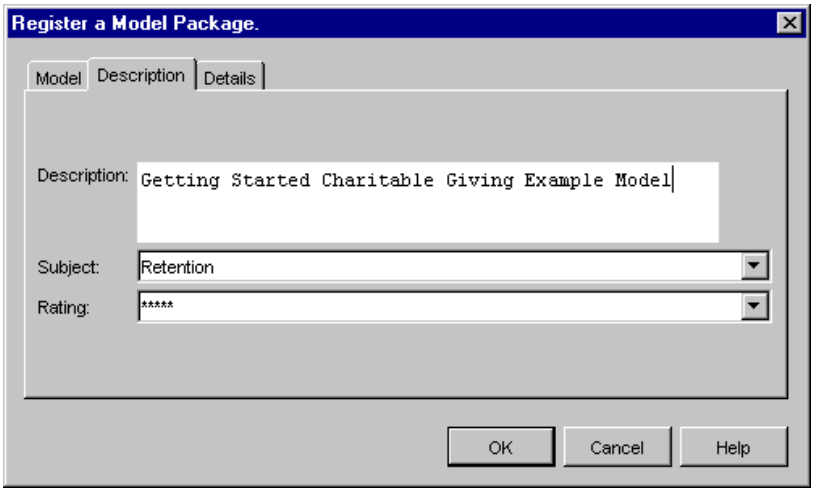


The Register a Model Package window opens.

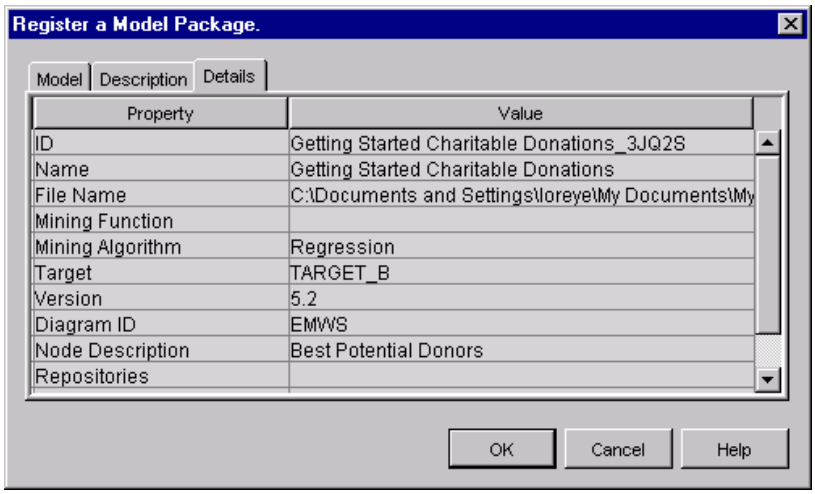


On the **Model** tab, you can select a model group if groups have been defined. None have been defined in this case.

2 On the **Description** tab, you can type descriptive text such as **Getting Started Charitable Giving Example Model**. You can also assign a model subject, and assign a model rating.



3 Select the **Details** tab to see metadata about the registered model.

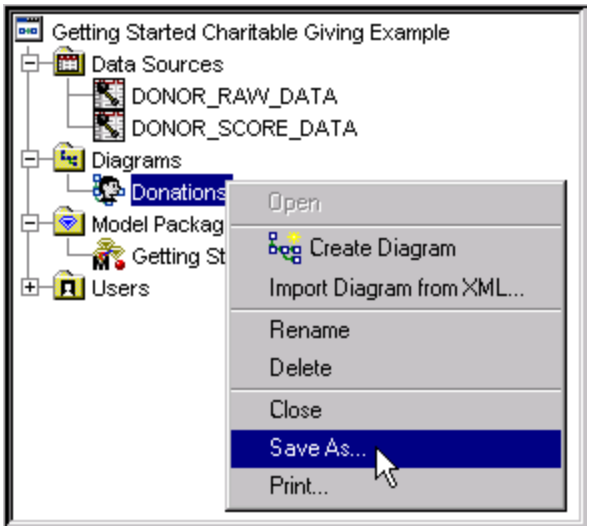


4 Click OK to register the model.

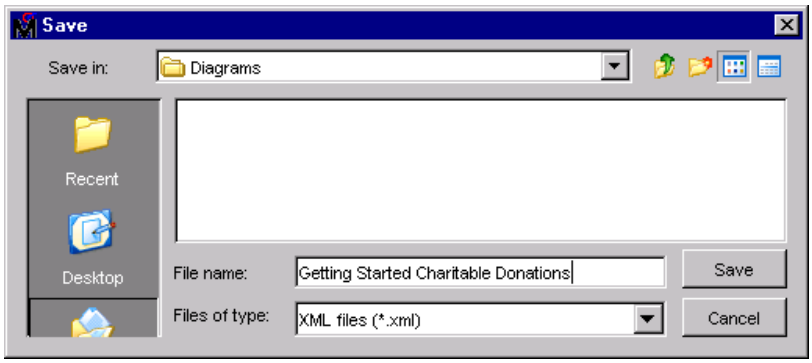
Save and Import Diagrams in XML

Enterprise Miner process flow diagrams can be encapsulated as XML files and shared with other users. In the Project Navigator, open the Diagrams folder and right-click any diagram in order to create an XML wrapper for a process flow diagram, or to import a process flow diagram that was saved in XML format.

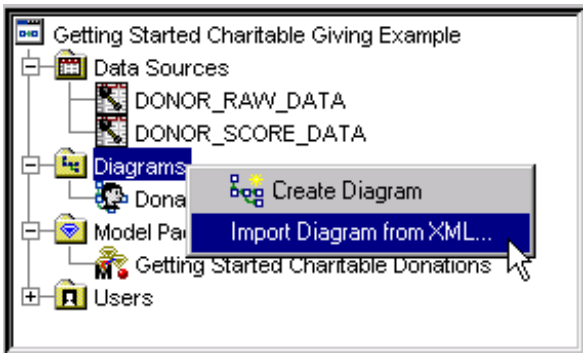
1 Right-click the process flow diagram's folder from the Diagrams subfolder of the Project panel. Select **Save As**.



2 Specify the folder where you want to save your process flow diagram, and type the name in the **File name** box. Click Save to store your diagram as an XML file.

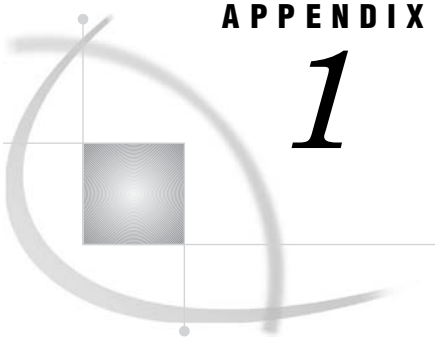


3 To import a diagram, from the main menu, select **File ▶ Import Diagram from XML**, or, from the Project panel, right-click the **`Diagrams`** folder and select **`Import Diagram from XML`**.



4 Navigate to the location of the saved XML file and then click OK to import the diagram.

Note: After you open an imported process flow diagram in the diagram workspace, you will need to run the flow to generate results. If you import the diagram into a project where the data sources do not reside, you will also need to define these data sources. △



APPENDIX

1

Recommended Reading

Recommended Reading

Here is the recommended reading list for this title:

- *Data Mining Using SAS Enterprise Miner: A Case Study Approach*
- *Getting Started with SAS Enterprise Miner 4.3*

For a complete list of SAS publications, see the current *SAS Publishing Catalog*. To order the most current publications or to receive a free copy of the catalog, contact a SAS representative at

SAS Publishing Sales
SAS Campus Drive
Cary, NC 27513
Telephone: (800) 727-3228*
Fax: (919) 677-8166
E-mail: sasbook@sas.com
Web address: **support.sas.com/pubs**
* For other SAS Institute business, call (919) 677-8000.

Customers outside the United States should contact their local SAS office.

APPENDIX

2

Example Data Description

The following table describes the variables that are used in this example.

Table A2.1 Variables That Are Used in the Donor Data set

Variable	Description
CARD_PROM_12	Number of card promotions received in the last 12 months
CLUSTER_CODE	54 socio-economic cluster codes
CONTROL_NUMBER	The control number uniquely identifies each member of the analysis population
DONOR_AGE	Age as of June 1997
DONOR_GENDER	Actual or inferred gender
FILE_AVG_GIFT	Average gift from raw data
FILE_CARD_GIFT	Average card gift from raw data
FREQUENCY_STATUS-97NK	Frequency status as of June 1997
HOME_OWNER	H=Homeowner U=Unknown
INCOME_GROUP	7 income group levels
IN_HOUSE	A final field identifies donors who are part of the organization's In House program
LAST_GIFT_AMT	Amount of most recent donation
LIFETIME_AVG_GIFT_AMT	Overall average gift amount
LIFETIME_CARD_PROM	Total number of card promotions received
LIFETIME_GIFT_AMOUNT	Total gift amount given
LIFETIME_GIFT_COUNT	Total number donations given
LIFETIME_GIFT_RANGE	Maximum less minimum gift amount
LIFETIME_MAX_GIFT_AMT	Maximum gift amount
LIFETIME_MIN_GIFT_AMT	Minimum gift amount
LIFETIME_PROM	Total number of promotions received
MEDIAN_HOME_VALUE	Median home value in $100's
MEDIAN_HOUSEHOLD_INCOME	Median household income in $100's

MONTHS_SINCE_FIRST_GIFT	First donation date from June 1997
MONTHS_SINCE_LAST_GIFT	Last donation date from June 1997
MONTHS_SINCE_LAST_PROM_RESP	Number of months since donor has responded to a promotion date from June 1997
MONTHS_SINCE_ORIGIN	This number is derived from MONTHS_SINCE_FIRST
MOR_HIT_RATE	Total number of known times the donor has responded to a mail order offer other than the national charitable organization's.
NUMBER_PROM_12	Number of promotions received in the last 12 months
OVERLAY_SOURCE	M=Metromail P=Polk B=Both
PCT_ATTRIBUTE1	Percent with attribute1 in the block
PCT_ATTRIBUTE2	Percent with attribute2 in the block
PCT_ATTRIBUTE3	Percent with attribute3 in the block
PCT_ATTRIBUTE4	Percent with attribute4 in the block
PCT_OWNER_OCCUPIED	Percent of owner-occupied housing
PEP_STAR	STAR-status ever (1=yes, 0=no)
PER_CAPITA_INCOME	Per capita income in dollars
PUBLISHED_PHONE	Indicator of presence of published telephone listing
RECENCY_STATUS_96NK	Recency status as of June 1996
RECENT_AVG_CARD_GIFT_AMT	Average gift amount to card promotions since June 1994
RECENT_AVG_GIFT_AMT	Average gift amount since June 1994
RECENT_CARD_RESPONSE_COUNT	Response count since June 1994
RECENT_CARD_RESPONSE_PROP	Response proportion since June 1994
RECENT_RESPONSE_COUNT	Response count since June 1994
RECENT_RESPONSE_PROP	Response proportion since June 1994
RECENT_STAR_STATUS	STAR (1,0) status since June 1994
SES	5 socio-economic cluster codes
TARGET_B	Response to 97NK solicitation (1=yes, 0=no)
TARGET_D	Response amount to 97NK solicitation (missing if no response)
URBANICITY	U=Urban C=City S=Suburban T=Town R=Rural ?=Unknown
WEALTH_RATING	10 wealth rating groups

Glossary

assessment
the process of determining how well a model computes good outputs from input data that is not used during training. Assessment statistics are automatically computed when you train a model with a modeling node. By default, assessment statistics are calculated from the validation data set.

association discovery
the process of identifying items that occur together in a particular event or record. This technique is also known as market basket analysis. Association discovery rules are based on frequency counts of the number of times items occur alone and in combination in the database.

binary variable
a variable that contains two discrete values (for example, PURCHASE: Yes and No).

branch
a subtree that is rooted in one of the initial divisions of a segment of a tree. For example, if a rule splits a segment into seven subsets, then seven branches grow from the segment.

CART (classification and regression trees)
a decision tree technique that is used for classifying or segmenting a data set. The technique provides a set of rules that can be applied to new data sets in order to predict which records will have a particular outcome. It also segments a data set by creating 2-way splits. The CART technique requires less data preparation than CHAID.

case
a collection of information about one of many entities that are represented in a data set. A case is an observation in the data set.

CHAID (chi-squared automatic interaction detection)
a technique for building decision trees. The CHAID technique specifies a significance level of a chi-square test to stop tree growth.

champion model
the best predictive model that is chosen from a pool of candidate models in a data mining environment. Candidate models are developed using various data mining heuristics and algorithm configurations. Competing models are compared and

assessed using criteria such as training, validation, and test data fit and model score comparisons.

clustering
the process of dividing a data set into mutually exclusive groups such that the observations for each group are as close as possible to one another, and different groups are as far as possible from one another.

cost variable
a variable that is used to track cost in a data mining analysis.

data mining database (DMDB)
a SAS data set that is designed to optimize the performance of the modeling nodes. DMDBs enhance performance by reducing the number of passes that the analytical engine needs to make through the data. Each DMDB contains a meta catalog, which includes summary statistics for numeric variables and factor-level information for categorical variables.

data source
a data object that represents a SAS data set in the Java-based Enterprise Miner GUI. A data source contains all the metadata for a SAS data set that Enterprise Miner needs in order to use the data set in a data mining process flow diagram. The SAS data set metadata that is required to create an Enterprise Miner data source includes the name and location of the data set, the SAS code that is used to define its library path, and the variable roles, measurement levels, and associated attributes that are used in the data mining process.

data subdirectory
a subdirectory within the Enterprise Miner project location. The data subdirectory contains files that are created when you run process flow diagrams in an Enterprise Miner project.

decile
any of the nine points that divide the values of a variable into ten groups of equal frequency, or any of those groups.

dependent variable
a variable whose value is determined by the value of another variable or by the values of a set of variables.

depth
the number of successive hierarchical partitions of the data in a tree. The initial, undivided segment has a depth of 0.

diagram
See process flow diagram.

format
a pattern or set of instructions that SAS uses to determine how the values of a variable (or column) should be written or displayed. SAS provides a set of standard formats and also enables you to define your own formats.

generalization
the computation of accurate outputs, using input data that was not used during training.

hidden layer
in a neural network, a layer between input and output to which one or more activation functions are applied. Hidden layers are typically used to introduce nonlinearity.

hidden neuron

in a feed-forward, multilayer neural network, a neuron that is in one or more of the hidden layers that exist between the input and output neuron layers. The size of a neural network depends largely on the number of layers and on the number of hidden units per layer. See also hidden layer.

hold-out data
a portion of the historical data that is set aside during model development. Hold-out data can be used as test data to benchmark the fit and accuracy of the emerging predictive model. See also model.

imputation
the computation of replacement values for missing input values.

input variable
a variable that is used in a data mining process to predict the value of one or more target variables.

interval variable
a continuous variable that contains values across a range. For example, a continuous variable called Temperature could have values such as 0, 32, 34, 36, 43.5, 44, 56, 80, 99, 99.9, and 100.

leaf
in a tree diagram, any segment that is not further segmented. The final leaves in a tree are called terminal nodes.

level
a successive hierarchical partition of data in a tree. The first level represents the entire unpartitioned data set. The second level represents the first partition of the data into segments, and so on.

libref (library reference)
a name that is temporarily associated with a SAS library. The complete name of a SAS file consists of two words, separated by a period. The libref, which is the first word, indicates the library. The second word is the name of the specific SAS file. For example, in VLIB.NEWBDAY, the libref VLIB tells SAS which library contains the file NEWBDAY. You assign a libref with a LIBNAME statement or with an operating system command.

lift
in association analyses and sequence analyses, a calculation that is equal to the confidence factor divided by the expected confidence. See also confidence, expected confidence.

logistic regression
a form of regression analysis in which the target variable (response variable) represents a binary-level or ordinal-level response.

macro variable
a variable that is part of the SAS macro programming language. The value of a macro variable is a string that remains constant until you change it. Macro variables are sometimes referred to as symbolic variables.

measurement
the process of assigning numbers to an object in order to quantify, rank, or scale an attribute of the object.

measurement level
a classification that describes the type of data that a variable contains. The most common measurement levels for variables are nominal, ordinal, interval, log-interval, ratio, and absolute. See also interval variable, nominal variable, ordinal variable.

metadata
a description or definition of data or information.

metadata sample
a sample of the input data source that is downloaded to the client and that is used throughout SAS Enterprise Miner to determine meta information about the data, such as number of variables, variable roles, variable status, variable level, variable type, and variable label.

model
a formula or algorithm that computes outputs from inputs. A data mining model includes information about the conditional distribution of the target variables, given the input variables.

multilayer perceptron (MLP)
a neural network that has one or more hidden layers, each of which has a linear combination function and executes a nonlinear activation function on the input to that layer. See also hidden layer.

neural networks
a class of flexible nonlinear regression models, discriminant models, data reduction models, and nonlinear dynamic systems that often consist of a large number of neurons. These neurons are usually interconnected in complex ways and are often organized into layers. See also neuron.

node
(1) in the SAS Enterprise Miner user interface, a graphical object that represents a data mining task in a process flow diagram. The statistical tools that perform the data mining tasks are called nodes when they are placed on a data mining process flow diagram. Each node performs a mathematical or graphical operation as a component of an analytical and predictive data model. (2) in a neural network, a linear or nonlinear computing element that accepts one or more inputs, computes a function of the inputs, and optionally directs the result to one or more other neurons. Nodes are also known as neurons or units. (3) a leaf in a tree diagram. The terms leaf, node, and segment are closely related and sometimes refer to the same part of a tree. See also process flow diagram, internal node.

nominal variable
a variable that contains discrete values that do not have a logical order. For example, a nominal variable called Vehicle could have values such as car, truck, bus, and train.

numeric variable
a variable that contains only numeric values and related symbols, such as decimal points, plus signs, and minus signs.

observation
a row in a SAS data set. All of the data values in an observation are associated with a single entity such as a customer or a state. Each observation contains either one data value or a missing-value indicator for each variable.

partition
to divide available data into training, validation, and test data sets.

perceptron
a linear or nonlinear neural network with or without one or more hidden layers.

predicted value
in a regression model, the value of a dependent variable that is calculated by evaluating the estimated regression equation for a specified set of values of the explanatory variables.

process flow diagram
a graphical representation of the various data mining tasks that are performed by individual Enterprise Miner nodes during a data mining analysis. A process flow diagram consists of two or more individual nodes that are connected in the order in which the data miner wants the corresponding statistical operations to be performed.

profit matrix
a table of expected revenues and expected costs for each decision alternative for each level of a target variable.

project
a collection of Enterprise Miner process flow diagrams. See also process flow diagram.

root node
the initial segment of a tree. The root node represents the entire data set that is submitted to the tree, before any splits are made.

rule
See association analysis rule, sequence analysis rule, tree splitting rule.

sampling
the process of subsetting a population into *n* cases. The reason for sampling is to decrease the time required for fitting a model.

SAS data set
a file whose contents are in one of the native SAS file formats. There are two types of SAS data sets: SAS data files and SAS data views. SAS data files contain data values in addition to descriptor information that is associated with the data. SAS data views contain only the descriptor information plus other information that is required for retrieving data values from other SAS data sets or from files whose contents are in other software vendors' file formats.

scoring
the process of applying a model to new data in order to compute outputs. Scoring is the last process that is performed in data mining.

seed
an initial value from which a random number function or CALL routine calculates a random value.

segmentation
the process of dividing a population into sub-populations of similar individuals. Segmentation can be done in a supervisory mode (using a target variable and various techniques, including decision trees) or without supervision (using clustering or a Kohonen network). See also Kohonen network.

self-organizing map
See SOM (self-organizing map).

SEMMA
the data mining process that is used by Enterprise Miner. SEMMA stands for Sample, Explore, Modify, Model, and Assess.

sequence variable
a variable whose value is a time stamp that is used to determine the sequence in which two or more events occurred.

SOM (self-organizing map)
a competitive learning neural network that is used for clustering, visualization, and abstraction. A SOM classifies the parameter space into multiple clusters, while at the same time organizing the clusters into a map that is based on the relative distances between clusters. See also Kohonen network.

target variable
a variable whose values are known in one or more data sets that are available (in training data, for example) but whose values are unknown in one or more future data sets (in a score data set, for example). Data mining models use data from known variables to predict the values of target variables.

test data
currently available data that contains input values and target values that are not used during training, but which instead are used for generalization and to compare models.

training
the process of computing good values for the weights in a model.

training data
currently available data that contains input values and target values that are used for model training.

transformation
the process of applying a function to a variable in order to adjust the variable's range, variability, or both.

tree
the complete set of rules that are used to split data into a hierarchy of successive segments. A tree consists of branches and leaves, in which each set of leaves represents an optimal segmentation of the branches above them according to a statistical measure.

validation data
data that is used to validate the suitability of a data model that was developed using training data. Both training data sets and validation data sets contain target variable values. Target variable values in the training data are used to train the model. Target variable values in the validation data set are used to compare the training model's predictions to the known target values, assessing the model's fit before using the model to score new data.

variable
a column in a SAS data set or in a SAS data view. The data values for each variable describe a single characteristic for all observations. Each SAS variable can have the following attributes: name, data type (character or numeric), length, format, informat, and label.

variable attribute
any of the following characteristics that are associated with a particular variable: name, label, format, informat, data type, and length.

variable level
the set of data dimensions for binary, interval, or class variables. Binary variables have two levels. A binary variable CREDIT could have levels of 1 and 0, Yes and No, or Accept and Reject. Interval variables have levels that correspond to the number of interval variable partitions. For example, an interval variable PURCHASE_AGE might have levels of 0-18, 19-39, 40-65, and >65. Class variables have levels that correspond to the class members. For example, a class variable HOMEHEAT might have four variable levels: Coal/Wood, FuelOil, Gas, and Electric. Data mining decision and profit matrixes are composed of variable levels.

Index

L

M

N

O

P

R

S

T

U

V

W

X

Your Turn

If you have comments or suggestions about *Getting Started with SAS Enterprise Miner™ 5.2*, please send them to us on a photocopy of this page, or send us electronic mail.

For comments about this book, please return the photocopy to

SAS Publishing
SAS Campus Drive
Cary, NC 27513
E-mail: **yourturn@sas.com**

For suggestions about the software, please return the photocopy to

SAS Institute, Inc.
Technical Support Division
SAS Campus Drive
Cary, NC 27513
E-mail: **suggest@sas.com**