MineSet™ Enterprise Edition
User's Guide for the Windows® Client

Document Number 007-4005-002

CONTRIBUTORS

Written by Sandra Motroni and Helen Vanderberg
Illustrated by Dany Galgiani
Production by Diane Ciardelli
Engineering contributions by Barry Becker, Amit Bleiweiss, Jeff Brainerd, Cliff Brunk, Eben Haber, Ara Jerahian, Andy Kar, Ed Karrels, Eser Kandogan, Alex Kozlov, Alan Norton, Peter Rathmann, Mario Schkolnick, Dan Sommerfield, Peter Welch, and Brett Zane-Ulman.

MineSet™ Enterprise Edition User's Guide for the Windows® Client
Document Number 007-4005-002

Contents

List of Figures

List of Tables

About This Guide

The *MineSet Enterprise Edition User's Guide* describes the features and capabilities of the MineSet mining and visualization tools. Current information about the MineSet product can also be found on the World Wide Web at http://www.sgi.com/software/mineset/.

Use this book with MineSet version 3.1 and later.

Audience for This Guide

You do not need to be an expert in data mining to use this guide or to use MineSet, but understanding your data and what it represents can help you more easily interpret the results. If you have experience with data mining techniques, this guide will still help you learn how the MineSet algorithms work and how they can be applied and visualized.

If you are using Tool Manager to import data from a database into the MineSet tools, you should refer primarily to the *MineSet Enterprise Edition Interface Guide* to perform the associated tasks. Once the data has been loaded into the various visualization tools, you do not need a database or programming background.

Finding MineSet Information

This guide deals with the tasks involved with running MineSet. Most of the book is oriented towards telling you how to run the various tools. A chapter-by-chapter summary can be found in "Structure of This Document".

For technical details and more complete explanations, refer to the *MineSet Enterprise Edition Reference Guide.* For very technical information about specific MineSet algorithms, see the list of white papers at http:www.sgi.com/software/mineset/mineset_data.html

For information on exporting MineSet to other locations or other applications, working on the command-line, or anything to do with system administration, turn to the *MineSet Enterprise Edition Interface Guide*.

MineSet also provides a means for third-party products such as AcPro to plug in to the application.

Structure of This Document

The first three chapters of this guide introduce data mining and the MineSet product. Subsequent chapters concentrate on specific tools and processes as shown:

Chapter 1, “Overview of Data Mining and MineSet Tools”
This chapter provides a brief overview of the principles of data mining, explains terminology, and introduces the processes involved in analytical and visual data mining with the MineSet tool suite.

Chapter 2, “Accessing Data with MineSet”
This chapter describes how to start MineSet and how to use some basic tools to look at data.

Chapter 3, “Shaping the Data”
This chapter describes both the need for, and the process of, transforming original data using the Tool Manager. The Column Importance and Clustering tools for exploring unseen characteristics of the data are explained.

Chapter 4, “Examining Data with the Scatter and Splat Visualizers”
This chapter provides a description of the Scatter and Splat Visualizer interfaces. These tools are valuable for visualizing multidimensional data, statically or by animation.

Chapter 5, “Examining Data with the Tree Visualizer”
This chapter provides a description of the Tree Visualizer tool interface. This tool is valuable for visualizing hierarchical data.

Chapter 6, “Examining Data with the Map Visualizer”
This chapter describes the Map Visualizer tool interface. This tool is useful for data with a geographical or spatial context.

Chapter 7, “Understanding Predictive Modeling”
This chapter describes predictive modeling, as opposed to descriptive modeling, and the varieties of classifiers offered by the MineSet tools.

Chapter 8, “Modeling and Predicting with Decision, Option, and Regression Trees”
This chapter describes how to generate and use the Decision, Option, and Regression tools. These tools are valuable for classifying data according to a set of attributes by making a series of decisions based on those attributes. Option trees can show the influence of splitting on multiple attributes simultaneously. Regression Trees are useful for predicting attributes based on continuous values, such as occur in real life.

Chapter 9, “Modeling and Predicting with the Decision Table Classifier and Visualizer”
This chapter provides a complete description of the Decision Table interface. This tool is valuable for visualizing decisions made in classifying data and for creating a classifier.

Chapter 10, “Modeling and Predicting with the Evidence Classifier and Visualizer”
This chapter describes how to generate and use the Evidence Classifier. This tool is valuable for classifying data by examining the probabilities of a specified result occurring based on a given attribute.

Chapter 11, “Refining Predictive Modeling”
This chapter describes the purpose and use of the Confusion Matrix, Loss Matrix, and ROI and Lift Curves to relate predictive modeling to real life situations.

Chapter 12, “Segmenting the Data with Clustering,”
This chapter provides a description of the Cluster Visualizer interface. This tool is valuable for examining your data and determining clustering patterns.

Chapter 13, “Analyzing Data with Association Rules”
This chapter provides a description of the Association Rules Visualizer. This tool is valuable for mining large datasets, and discovering correlations in that data.

Glossary, “MineSet User’s Glossary”
The glossary explains technical words and phrases used frequently in this guide.

Illustrations in This Guide

The hard copy of this guide shows screen shots and illustrations in black and white. The online version provides these visuals in full color. If you find a particular graphic or screen shot difficult to see in hard copy, refer to that page online for greater clarity.

Further Reading

For general information on data mining and related techniques, you may wish to consult the following books:

- *Data Mining Techniques for Marketing, Sales, and Customer Support*, by Michael Berry and Gordon Linoff, published by John Wiley & Sons.
- *Data Mining Solutions: Methods and Tools for Solving Real World Problems*, by Christopher Westphal and Teresa Braxton, published by John Wiley & Sons.

Appendix A in the *MineSet Enterprise Edition Interface Guide* lists more reading if you are interested in more technical information about data mining.

Typographical Conventions

The following type conventions and symbols are used in this guide:

Italics	Executable names, filenames, program variables, tools, utilities, variable command-line arguments, and variables to be supplied by the user in examples, code, and syntax statements
Bold	Keywords
`Fixed-width type`	On-screen command-line text and prompts
`Bold fixed-width type`	User input, including keyboard keys (printing and nonprinting); literals supplied by the user in examples, code, and syntax statements

Reader Comments

If you have comments about the technical accuracy, content, or organization of this document, please tell us. Be sure to include the title and document number of the manual with your comments. (Online, the document number is located in the front matter of the manual. In printed manuals, the document number can be found on the back cover.)

You can contact us in any of the following ways:

- Send e-mail to the following address:

  `techpubs@sgi.com`

- Use the Feedback option on the Technical Publications Library World Wide Web page:

  `http://techpubs.sgi.com`

- Contact your customer service representative and ask that an incident be filed in the SGI incident tracking system.

- Send mail to the following address:

  Technical Publications
  SGI
  1600 Amphitheater Pkwy., M/S 535
  Mountain View, California 94043-1351

- Send a fax to the attention of "Technical Publications" at +1 650 932 0801.

We value your comments and will respond to them promptly.

Overview of Data Mining and MineSet Tools

This chapter introduces the subject of data mining and how MineSet tools work in that context. If you already know about MineSet, you can skip this chapter. The subjects covered here are:

MineSet is an integrated tool suite for data analysis. MineSet's graphical user interface can be used by marketing and sales specialists, financial analysts, insurers, indeed anyone wanting to analyze data. MineSet also provides interfaces for application developers wishing to write decision support applications for business and scientific purposes.

Data Mining Terminology

The use of basic data mining terms is outlined here and in the glossary at the end of the book. For instance, in this context, data is a collection of records, often initially gathered when a business or scientific process is performed, such as customer billing, pharmaceutical testing, or point-of-sale transactions. These records are organized for MineSet in table form with rows and columns. The table could be given a descriptive name like "Customers," representing all records about customers of a particular enterprise. In the context of data mining, columns are often referred to as attributes, and rows may be referred to as instances or records.

Among the algorithms or mathematical formulas MineSet uses, some are called inducers because they induce the model to be built. The models themselves are called classifiers simply because they classify the rows and columns of data. Hence, you can take the record contents of rows and columns and sort or classify them in many ways.

About Data Mining and Data Mining Methods

The purpose of data mining is to discover patterns in data so that this knowledge can be applied to solving problems. Typical problems solved by data mining include:

- Fraud detection
- Churn analysis
- Calling pattern analysis
- Target marketing
- Determining market segmentation
- Improving operational procedures
- Improving medical service
- Market basket analysis

When you use queries or online analytic processing (OLAP) to gather information about data from a database, you must specify directly any relationships between data elements. For example, you might query for all the sales by region. This method presupposes you have an idea that sales vary by region, and so you test that hypothesis against data to confirm or reject its validity.

It is at this juncture that OLAP moves in the direction of discovery-based data mining, where relationships may be uncovered that you did not know existed. This method allows the data itself to suggest conclusions to the investigator. It is this ability to discover the previously unknown that distinguishes data mining from OLAP and other approaches to data mining such as queries.

Visual data mining presents data in a revealing visual form allowing you to see trends and tendencies just by looking at the visualization. This human ability to visualize and survey complex data patterns can prove invaluable in making decisions. Such visual data mining is descriptive, that is, it describes existing data that has been measured or quantified.

Analytical data mining uses algorithms to automatically develop models derived from your data. These algorithms then run through the data in various ways, depending on the algorithmic method chosen. You can display the results of your data analysis in visual form, test them against untried data, or fit models on a completely new set of data. Analytical data mining uses the model to predict the characteristics of the next piece of data presented.

Analytical data mining results in models about the data. It does this by using algorithms that fall into two broad categories—supervised and unsupervised modeling methods.

In MineSet, the combination of visualization with mining models provides the user with greater insight than either technique applied independently.

Analytical Data Mining

In this discussion, data appears in a table consisting of columns and rows. Data records are represented by rows: the data attributes are represented in the columns of the table (see Figure 1-1).

	Descriptive Attributes				Label
	Age	**Marital Status**	**Sex**	**Yrs Employed**	**Credit Risk**
Record 1	35	D	M	3	> 60
Record 2	45	M	F	19	< 60
Record 3	29	S	M	45	> 60

Figure 1-1 Sample Data Table

Supervised Modeling

Supervised modeling is a task in which the goal is to predict the content or value of one column (attribute) in a record, based on the values of other columns (attributes) in that same record. You intend to predict a special attribute or characteristic. This attribute is called the *label*. By encoding the relation between the label and the other attributes, the model built by a supervised modeling algorithm can be used to make predictions about new, unlabeled data.

You can then visualize the model to gain insight into the relationship between labels and other attributes. For example, if customers have left your company (typically called attrition or churn), you can build a model that not only predicts which customers are likely to churn, but also helps you understand the reasons and patterns that lead to this behavior.

The two most common supervised modeling tasks are called *classification* and *regression*. If the label is discrete (that is, it contains a fixed set of values—for example, "yes" or "no"), the task is called classification; if the label is a continuous value (that is, it can take a value in a continuous range—for example, income or stock price), the task is called regression.

Classification

Classification is the task of assigning a discrete label value to an unlabeled record. In doing so, records are divided into predefined groups. For example, a simple classification might group customer billing records into two specific classes: those who pay their bills within 60 days and those who take longer than 60 days to pay. Further data classification examples might divide customers by sex or income. The process of classification can also predict the probability that the label will take on a specific value. For example, you can compute the probability that a person will pay the bill within 60 days.

In creating these classifications, MineSet utilizes algorithms that induce (build) models based on the data you give it. MineSet can induce a classifier automatically from a training set. A training set is a random subset of the original data. For example, you may decide to use just 40% of the data to build the model. Once a model is generated, it can be used to classify or predict class probabilities for records with unknown values of the label. For instance, when new records are added to the table, the value of the label may not be known. When this happens, the model can predict the value of the label column. If the value of the label is later found to be the same as the value predicted, the model is said to be correct. The part of the original dataset not used for training may be used to test the accuracy of the induced model.

MineSet can induce four classification models, each one of which can be viewed using a visualizer. These models are different, based on the way the algorithm makes decisions in sorting and classifying the data. For example, a Decision Tree might segment records of telephone calls into two groups based on customers who talk for more than or less than 256 minutes. An Option Tree might give you three choices of ways to search for customers likely to churn. These differences are detailed in "MineSet Tools for Data Mining" on page 7, and examples are shown in later chapters.

Regression

Regression is a supervised modeling task similar to classification, except that the values that the label can take are continuous, not discrete. For example, predicting salary or the price of a stock is a regression, whereas predicting whether a salary is one of many given ranges (for instance, less than $60K, between $60K and $120K, more than $120K), or whether a stock will go up or down, is a classification task.

MineSet can also induce regressors automatically from a training set or sample of the data. Once a regressor is generated, it can be used to predict values for the continuous attribute in question, such as a salary of $60,420. MineSet currently has one regression model: the Regression Tree. Like all other models built by MineSet, a Regression Tree model can be visualized with a MineSet visualization.

Assessing the Accuracy of Models

Estimating the accuracy of supervised models is an important part of the data mining process. Classifiers are usually evaluated according to their error rate, or proportion of misclassified records. Regressors are evaluated according to their absolute error of the standard deviation in their predictive error. When assessing the accuracy of a model, it is important to test it on data that was not used in building the model.

Unsupervised Modeling

Unsupervised modeling discovers rules and segments of the data that behave similarly. There is no notion of a correct answer, or any obvious agreed-upon measure of performance. Unsupervised modeling provides insight into the data as a whole by showing patterns and segments that are similar. The models cannot be used directly to make predictions; therefore it is not necessary to set aside part of the data as a training set from which to build the classifier. MineSet provides algorithms for the two most common unsupervised modeling: associations and clustering.

Associations

In generating associations, the task is to describe by what rules A implies B. The classic association grouping is market basket analysis, reporting the frequency with which certain items are purchased at the same time. For example, observing that purchasing baby food implies a higher probability that a customer will buy low-tar cigarettes rather than regular cigarettes might help stores arrange their shelves differently.

Clustering

Clustering algorithms segment the data into groups of records, or clusters, that have similar characteristics. For instance, a health-insurance company may discover that these characteristics define a segment: 20-to-45 years old, technical worker, fewer than two children, television science-fiction fan, and a disposable income of $5,000 to $10,000 per year. The segment can then be targeted more effectively with a health-insurance package well suited for these people, by using television ads in new science-fiction episodes.

Visual Data Mining

A data mining algorithm can be complemented with data visualization techniques, taking advantage of the human brain's amazing pattern recognition capability. The following MineSet visualizers are working examples:

- Map Visualizer—Data is displayed on a map, commonly a geographical map.
- Scatter Visualizer—Data points are shown in one, two, or three dimensions. Additional attributes can be mapped to color and size. Two other attributes may be mapped to sliders, allowing animation and fly-throughs, and an eighth attribute can be mapped to a color code to guide animations through potentially interesting combinations of values of the animation variables. The Column Importance operation in MineSet can help you identify the important dimensions to map for a given task.
- Splat Visualizer—Similar to the Scatter Visualizer, with the distinction that data density is shown by using varying opacity. The result approximates the effect of rendering each data point individually, and is particularly useful for datasets that may contain too many points to display in the Scatter Visualizer.
- Tree Visualizer—Data is mapped to nodes in order to see hierarchical breakdowns of the data. Decision Trees, Option Trees, and Regression Trees all show data in a variety of branching tree-like visualizations.

MineSet Tools for Data Mining

If you have datasets that require classification, regression, association, and clustering, you will find these MineSet tools useful:

- Decision Tree Inducer and Visualizer—Induces a classifier resulting in a branching format decision tree visualization.
- Option Tree Inducer and Visualizer—Induces a classifier similar to a decision tree inducer and classifier. However, it builds alternative trees and averages them during classification, usually leading to improved accuracy.
- Evidence Inducer and Visualizer—Creates its own classifier and produces a visualization to display evidence based on the data provided.
- Decision Table Inducer and Visualizer—Creates a hierarchical visualization displaying pairs of dimensions at every level. You can drill up to get an overview, and drill down to see detail quickly, while maintaining context.
- Regression Tree Inducer and Visualizer—Induces a regressor that predicts real valued attributes, that is, results with gradations of value rather than specific predetermined limits. The structure of the regressor is viewed using the Tree Visualizer.
- Association Rules—Encodes rules that govern associations in which A implies B, commonly called market basket analysis. The rules are shown using the Scatter Visualizer, typically with bars and disks.
- Clustering Algorithm—Groups data according to similarity of characteristics, then displays it as a series of box plots and histograms, similar to the Statistics Visualizer. The clustering algorithm displays results using the Cluster Visualizer by default, but other visual tools may be used as an alternative.
- Column Importance—Determines the importance of specific columns in discriminating one label value from another. Used to observe the varying effects of changing variables, or to suggest columns to map to the axes of the Scatter and Splat Visualizers.

MineSet contains additional tools to aid the knowledge discovery process:

- Statistics Visualizer—Displays data in the form of box plots and histograms, one per column. Continuous columns are shown as box plots; discrete columns are shown as histograms.

- Histogram Visualizer—Displays data in the form of histograms, one histogram per column of data, binning continuous columns if necessary.
- Record Viewer—Displays original data as a spreadsheet.

Surveying the Data Mining Process

This section introduces the specific tasks involved in the knowledge discovery process. The process is iterative, commonly going back to earlier stages once you discover new patterns and improve your understanding of the data, as shown in Figure 1-2.

Figure 1-2 Data Mining Process

The process follows these steps:

1. Identify the source of the data—see "Identifying the Data" on page 9.
2. Prepare the data—see "Preparing the Data" on page 9.
3. Build a model—see "Building a Model" on page 10.
4. Evaluate the model—see "Evaluating a Model" on page 11.
5. Deploy the model—see "Deploying a Model" on page 11.

Identifying the Data

The task of identifying the data begins by deciding what data is needed to solve a problem. For example, predictability about customer behavior is often a necessary goal. Recast in terms of a problem, the investigator must identify the data needed for a solution and explore other possible sources of data.

Data may be in a difficult location or in an obscure form. Sometimes there are several initial databases that may be incompatible with each other. Further, if data is scanty or incomplete, more data may be needed. The form in which new data is to be collected depends on the form of existing data. MineSet supports native interfaces to several commercial databases (Oracle, Informix, SQL), ODBC interface, as well as reading data from different file formats (tab-separated flat file, MineSet binary file, Excel, SPSS, MATLAB, etc.). Tools such as DBMS/COPY from Conceptual Software, Inc., allow you to convert data from over 100 formats to a format used by MineSet, and the MineSet Import Data capability (available from the Tool Manager File menu) performs a similar conversion (see the *MineSet Enterprise Edition Reference Guide* and *MineSet Enterprise Edition Interface Guide* for details).

Preparing the Data

Data may need modification before loading into MineSet (a step often called cleaning). Specifically, the following problems are common:

- Data may be in a format incompatible with MineSet representation (for example, binary, encoded, or EBCDIC strings from old mainframe computers).
- Data may be misspelled or erroneous, or have incomplete, or erroneous values.
- Field descriptions may be unclear or confusing, or may mean different things depending on the source. For example, order date may mean the date that the order was sent, postmarked, received, or keyed in.
- Data may be out of date; for example, customers may have moved, changed households, or changed spending patterns.

Even clean data may need to be transformed before it is suitable for mining and visualization.

Transforming the Data

Transformations can greatly improve model performance. If you were analyzing telephone company data, for instance, you may find that long distance rate (sales divided by total minutes used) is a better predictor of customer behavior than either element given separately. Data transformations are at the heart of developing a sound model. As you progress, you may even go back and transform the data differently. You can transform the data by:

- Adding columns, usually by applying a mathematical formula to existing data to create a new field.
- Removing columns that are not pertinent, are redundant, or contain obvious, uninteresting predictors.
- Filtering data based on a boolean expression of column values to influence the model or visualization. For example, you may want to see only the strongest rules or the most profitable customer segments.
- Binning data—breaking up a continuous range of data into discrete segments (for instance, [1 - 10], [11 - 20], and so on).
- Aggregating data—grouping records together, and finding the sum, maximum, minimum, or average values.
- Sampling the data to get a random subset of the data (by percentage or count).
- Applying a classifier, regressor, or clustering model that you have previously created, to label new records with a class label, or to estimate the probability of a given label value.

In MineSet, most of these transformations take place using the Data Transformation pane in Tool Manager. Plug-in modules may be available to create transformations or a particular kind of model.

Building a Model

At the core of the knowledge discovery process is model building, automatically done by analytical data mining algorithms. You can build the model using all the data you have, or you may hold back part of it with which to test the accuracy of the model. Building a model offers you several options in the way you induce the model. These choices influence not only the way the visualization is presented but also the decisions the algorithm makes in building the classifier. These choices are discussed further in Chapter 7, “Understanding Predictive Modeling.”

Evaluating a Model

Evaluating the accuracy of a model refines your understanding of that model and its usefulness. Some models, notably the Decision Tree classifier and the Option Tree classifier, evaluate different parts of the model and display them directly through visualization.

MineSet implements four model-assessment methods: error estimation, confusion matrix, lift curve, and ROI (return-on-investment) curve. These methods are explained in Chapter 11, "Refining Predictive Modeling."

Deploying a Model

A model can be deployed by applying it to new data. New data can give rise to further questions, which may require further refinements.

In the telecommunications example in the *MineSet Enterprise Edition Tutorial*, a model was created to determine which customers were likely churn (that is, leave their phone carrier). Customer records could then be evaluated through the model to identify the specific customers most likely to churn. These customers could then be offered incentives to stay.

You can step through the knowledge discovery process on the churn dataset—a prepared dataset of telecommunication customers that is included with the MineSet software. As you work through the examples in this guide, think how your business or scientific operations progress forward and loop back to establish a fully tested analytical data mining model.

In the next chapter, Chapter 2, "Accessing Data with MineSet," you can begin working directly with MineSet and the sample files that have been provided.

Accessing Data with MineSet

This chapter describes how to start out with MineSet, and shows how to view your data using procedures from the graphical user interface called Tool Manager. The topics discussed are:

Also see the *MineSet Enterprise Edition Tutorial* to practice using the various tools. When you want to use data from your own database, refer to *MineSet Enterprise Edition Interface Guide*, "Setting Up MineSet."

Where to Find MineSet

You can install MineSet from the CD or over the Internet. Once you have installed MineSet, usually on your own system, you can start using it immediately by practicing with the preformatted data files supplied with the MineSet release (found in the *data* directory in the directory where MineSet was installed. The next section contains a brief explanation of the operation of MineSet. If you want start immediately, see "Starting MineSet" on page 15.

How the MineSet Software Works

MineSet works in a client/server manner, with the server process existing either on the same system as the client or on a different, usually more powerful, system. The server must be installed somewhere accessible to the client. The DataMover is a process that runs on the server, and is responsible for accessing database files, performing data transformations, running mining operations, and generating the visualization files. These visualization files are then transferred to the client.

The Tool Manager is a client process providing the graphical user interface (GUI) in which you do most of your interactions with MineSet. These interactions compose the bulk of this chapter. You use Tool Manager to specify the source of your data, the set of transformations to be applied to the data, which mining and visualization tools to use, and how to save the results of your work.

Once these specifications are complete, you can direct MineSet to access the required data and perform the actions you specified through Tool Manager. Tool Manager then sends this information to the server DataMover process. If the data to be analyzed exists in a flat file, MineSet reads a configuration (*.schema*) file containing meta-information about the table and the name of a *.data* file where the real data is located. If the data to be analyzed is in an Oracle, Sybase, or Informix database, DataMover executes a query against the database to get the desired dataset.

The *.schema* file describes the format of an input data file. These files are explained in detail in the *MineSet Enterprise Edition Interface Guide*.

Note: Before using any of the MineSet tools, follow the installation and licensing instructions in the MineSet release notes. Your system administrator may need to set up the DataMover configuration file. The setup details are described in the *MineSet Enterprise Edition Interface Guide*.

MineSet and Your Application

MineSet features an application programming interface (API) so that you can integrate MineSet's visualizers and data mining engines into your own applications. See the *MineSet Enterprise Edition Interface Guide* and the MineSet website for more information.

System Settings for Optimum Visual Displays

In order for the examples in this book to display properly, the color palette on your display must be set to True Color. To set this, choose Start > Settings > Control Panel. When the Control Panel directory displays, double-click on the Display icon. In the resulting dialog box, choose True Color from the Color Palette drop down menu.

Starting MineSet

The easiest way to start MineSet is to create a shortcut for *mineset.exe* on the desktop. To do this, from the directory in which MineSet is installed, go to the *\bin* directory and click the *mineset.exe* file. From the File menu choose Create Shortcut.

You can also reach MineSet from the desktop Start > Programs > MineSet Enterprise Edition menu. Either select with the left mouse button to launch it, or press and hold the right mouse button to drag to the desktop and release the right mouse button.

1. Click on the MineSet shortcut icon and the MineSet Tool Manager is displayed.
2. If the "Log in to Server" dialog box (Figure 2-1) does not come up by default, choose File > Connect to Server. In the resulting dialog box, click "This machine as current user," if you wish to use your current system as both client and server. If you wish to use another system as a server, type in the server name, your login name, and your password (if any).

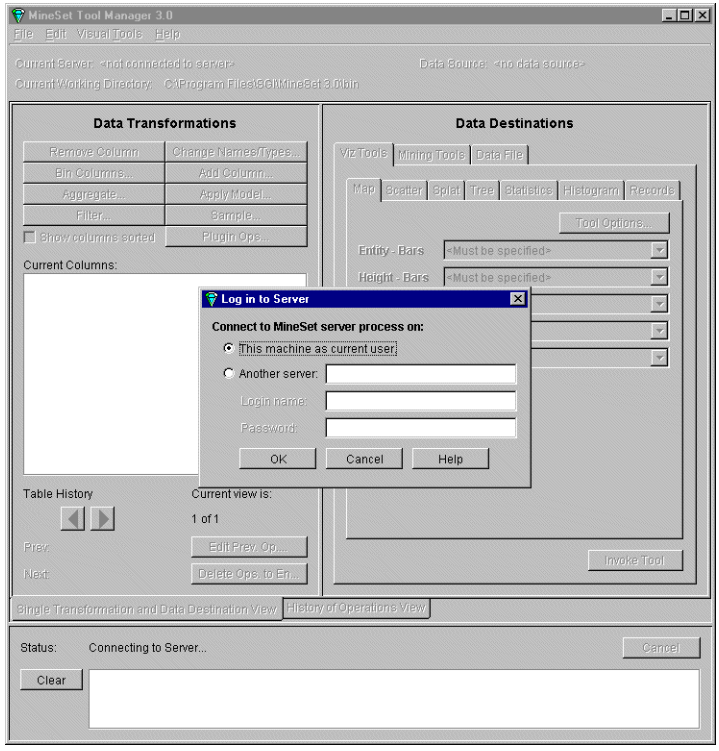


Figure 2-1 Tool Manager Login

3. In the Tool Manager window, choose File > Open New Data File. In the Open Data File window that appears, enter the path to the data you plan to use. If you wish to experiment with some of the sample files shipped with MineSet, they are in the *MineSet\data* directory, in the location where MineSet was initially installed.

   To work with a sample file, select *churn.schema* from the Open Data File window. This file is a sample table of data provided with MineSet. The entries shown in the right-hand Preview Columns pane are columns of data, as shown in Figure 2-2. Click *OK*.

The *churn.schema* file gives you access to a dataset of telecommunications customers. The next time you run MineSet, you will be automatically returned to the state you were in when you last exited MineSet, and any option selections you made will be restored.

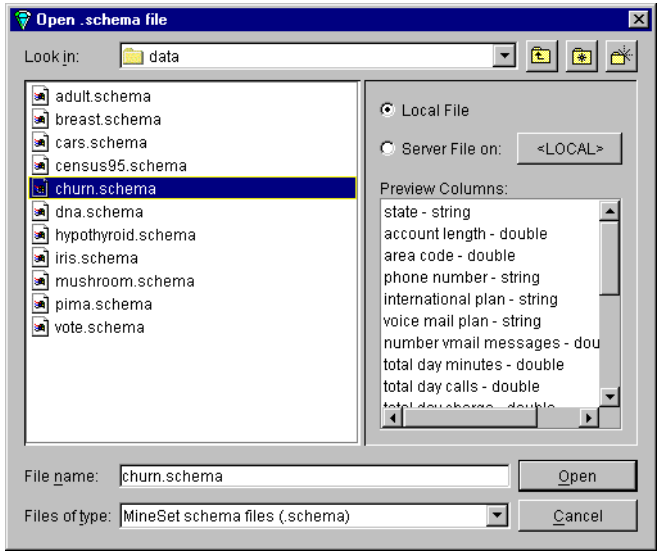


Figure 2-2 Open Data File Window

The *MineSet Enterprise Edition Reference Guide* contains a list of other sample files you can use to practice with; see the entry "Sample Configuration and Data Files" for the contents of each.

Note: You can also run MineSet in batch mode. When running in batch mode, the Tool Manager performs the actions described in a session file without bringing up any visualizations. Batch mode can be particularly useful in projects requiring lengthy computations that need to be done frequently. For instance, the computations can be run at night so the data will be ready the next morning.

Using the Tool Manager Window

The Tool Manager is your main point of entry to all actions, so it is helpful to make a quick tour of the parts.

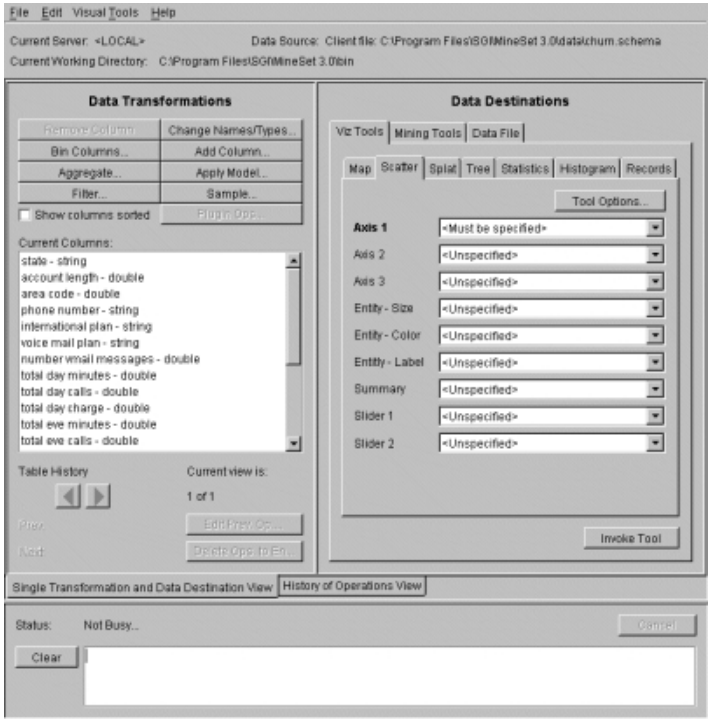


Figure 2-3 Tool Manager Window

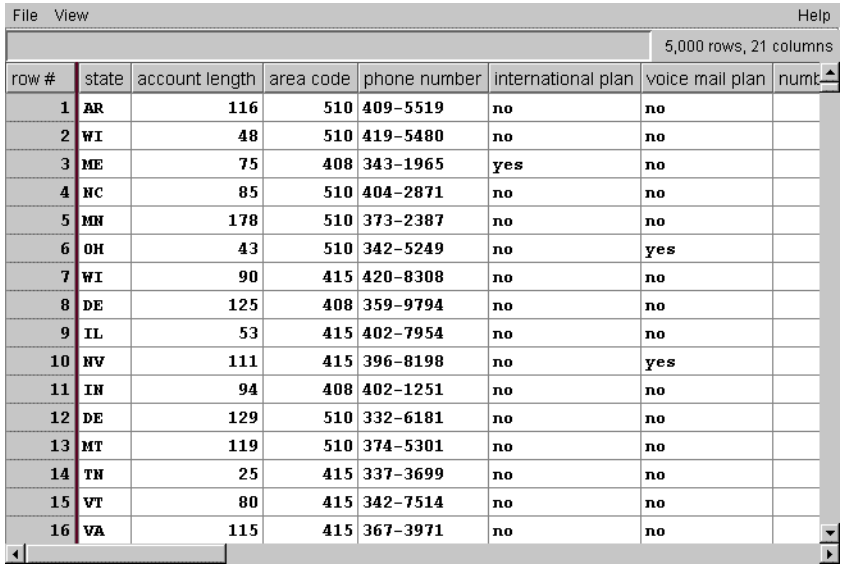
The Data Transformation pane on the left is where you transform all the data. The actions you take using the buttons at the top affect the Current Columns listed beneath. The buttons below allow you to go forward and back in the history of your transformations; the Status window across the bottom shows the action currently taking place on the dataset. The Data Destination pane on the right allows you to choose between Visualization Tools and Mining Tools or to save your current state to a Data File. Each of these functions is covered in later sections.

Viewing the Raw Records with Record Viewer

An easy way to become familiar with your selected dataset is to use the Record Viewer to see the database records and the data values within the columns. The records appear in spreadsheet form.

Bring up the Tool Manager as shown in “Starting MineSet”and follow these steps:

1. From the Tool Manager Data Destinations pane, click the Viz Tools tab.
2. From the lower row of tabs, click Records to access the Record Viewer.
3. Click *Invoke Tool* in the lower right corner of the pane, and the Record Viewer appears (Figure 2-4).



row #	state	account length	area code	phone number	international plan	voice mail plan	numb
1	AR	116	510	409-5519	no	no	
2	WI	48	510	419-5480	no	no	
3	ME	75	408	343-1965	yes	no	
4	NC	85	510	404-2871	no	no	
5	MN	178	510	373-2387	no	no	
6	OH	43	510	342-5249	no	yes	
7	WI	90	415	420-8308	no	no	
8	DE	125	408	359-9794	no	no	
9	IL	53	415	402-7954	no	no	
10	NV	111	415	396-8198	no	yes	
11	IN	94	408	402-1251	no	no	
12	DE	129	510	332-6181	no	no	
13	MT	119	510	374-5301	no	no	
14	TN	25	415	337-3699	no	no	
15	VT	80	415	342-7514	no	no	
16	VA	115	415	367-3971	no	no	

Figure 2-4 Record Viewer Display

If you prefer to see the results differently, the Record Viewer can make various sorting, renumbering, and resizing adjustments (see Table 2-1 below.)

Changing Record Viewer Columns

You can change the appearance of the Record Viewer columns using the methods shown in Table 2-1:

Table 2-1 Manipulating Record View Columns

To:	Do This:
Resize columns	Click on right column divider and drag in desired direction.
Rearrange columns	Click and drag title cell to desired location.
Hide columns	Choose Hide/Show columns from View menu, deselect columns to hide.
Sort record order by column values	Click on column title cell.
Sort record order by multiple columns	Hold down the Control key and click on each column title that you wish to sort by, in the order in which you wish them to sort.
Reverse the sort	Click column title cell again.
Return to original order	Click the row # column title.
Renumber rows	Choose Renumber rows from the View menu (cannot undo this function).
Search for a number or value	Choose Search panel from View pulldown menu, enter a value in the Find field, highlight the columns you want to search in, click *Find Next* or *Find Previous*. To select multiple columns, use Shift-click; for noncontiguous columns, use Ctrl-click.

Filtering Data in Record Viewer

You can filter data so that you see only a certain range of values.

1. From the Record Viewer window, choose View > Filter Panel.
2. Select a column and expression from which to build a filtering expression. For example, in the cars dataset, you build the expression `'cubicinches' > 400` to filter those records with cubic-inch capacity greater than 400 ci. (See Figure 2-5).

   You can build the expression by selecting elements on the left and clicking the right-pointing arrow; or you can simply type directly in the text field.
3. Click *Apply* and the filtering starts at once. The panel remains up until you dismiss it.

To remove the filtering, clear the expression in the expression window and click *Apply* or choose View > Remove Filter from the Record Viewer window.

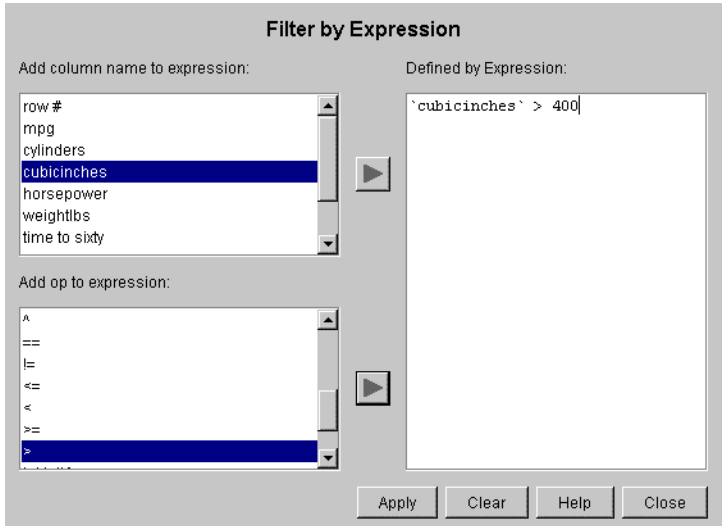


Figure 2-5 Record Viewer Filter Panel

You may have as many filter panels open as you wish. To apply more than one filter at a time, first apply one, renumber the rows, then apply the next.

Note: Renumbering the rows cannot be undone. To return to your original data, you must reopen the file.

Saving Data in Record Viewer

You can save your file in Record Viewer, using either File > Save or File > Save As.

With Save As, you can save your data in four formats: binary, ASCII, HTML, or text. When you save in binary or ASCII format, both the data file (the table) and a schema file (describing the contents of the data file to Tool Manager and Data Mover) are saved. HTML format saves the file as an HTML table. Text format saves the file in tab-delimited form, with the column titles as the first row.

Seeing Record Statistics with the Statistics Visualizer

You can find out more about the records in the dataset with the Statistics Visualizer. Certain statistics are calculated, based on the number of records in the dataset. Depending on whether the type of data in the column is *numeric* or *discrete,* statistics are shown as box plots and histograms, respectively.

Understanding Box Plots

The box plots show the minimum, maximum, mean, median, and two quartiles (25th and 75th percentiles) of numeric values in a column, as lines across a vertical colored bar. The standard deviation of the dataset population is shown as a +/- value. The quartiles are shown whenever there are fewer than 50,000 distinct values (see Figure 2-6). If there are more than 50,000 distinct values in the column, the statistics are shown as a gray vertical bar. For definitions of the terms *Mean*, *Median* and *Standard Deviation*, refer to the glossary.

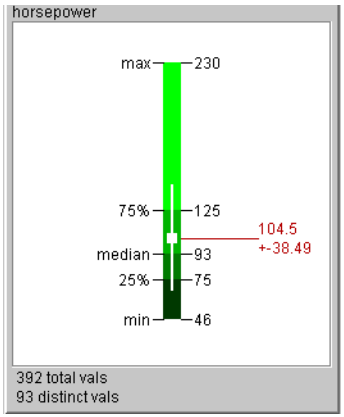


Figure 2-6 Numeric Column Displayed by the Statistics Visualizer

Understanding Histograms

The histograms show results from columns of non-numeric data, such as string, or bin. (See Figure 2-7.) This means that the columns in the data table can contain strings such as “yes” or “spore_color,” or binned values such as “10-70.” There can be up to 100 distinct entries like this. The default ordering of the string-valued (nominal) attributes is by decreasing count, but you can use the View pulldown menu to choose an alternative sorting. The two ways of sorting strings are by count (or weight if weighting is selected), or alphabetically by name. If there are 100 or fewer distinct categories, the column panel also contains the count of distinct values. If the values are binned, the bin ordering is maintained regardless of the sorting method.

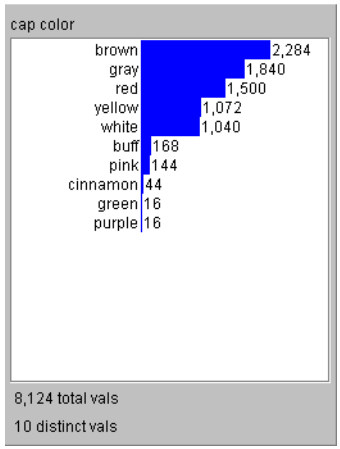


Figure 2-7 Discrete Column Displayed by the Statistics Visualizer

After creating a visualization of your data, you can see any truncated textual information in the histograms when you pass the mouse across a field without clicking.

Starting the Statistics Visualizer

The simplest way to start the Statistics Visualizer is to follow these steps, after you have selected a dataset:

1. From the Tool Manager Data Destinations pane, click the Viz Tools tab.
2. From the lower row of tabs, click the Statistics tab to access the Statistics Visualizer (Figure 2-8).
3. Click *Invoke Tool* in the lower right of the pane.

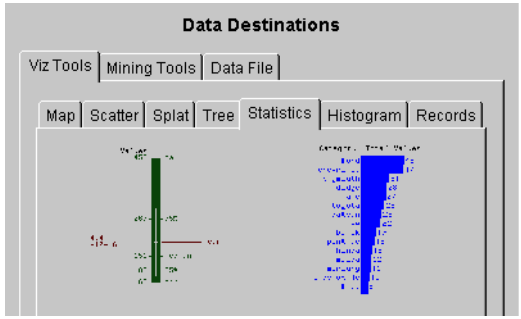


Figure 2-8 Data Destination Panel with the Statistics Visualizer Selected

See the "Statistics Visualizer" entry in the *MineSet Enterprise Edition Reference Guide* for other methods of starting the tool.

Working with Sample Data Files

You can work with some sample data files to familiarize yourself with the Statistics Visualizer's features and capabilities. These files are in the location where MineSet was initially installed in the *MineSet* directory, under *\examples*.

Start the Statistics Visualizer from the Visual Tools menu on Tool Manager's menu bar, then use the File > Open pulldown menu and open any *.statviz* file.

Starting the Histogram Visualizer

The Histogram Visualizer automatically bins all of the columns containing continuous, that is, numeric values and sends the result to the Statistics Visualizer for display. See the glossary entry for *Binning*.

The easiest way to start the Histogram Visualizer is to follow these steps:

1. From the Tool Manager Data Destinations pane, click the Viz Tools tab.
2. From the Tool choice menu, choose the Histogram Visualizer (Figure 2-9).

3. You can set the following options or simply go on to Step 4:
   - Pick the number of bins, or allow MineSet to do it for you.
   - Set the trimming fraction, which indicates the fraction of extreme values to be excluded before the bins are generated. The default is 0.05. This excludes the 5% of instances with most extreme values (2.5% at the lowest end and 2.5% at the highest). Trimming tends to reduce the influence of outliers on the generation of thresholds. Outliers are single instances that lie outside the bulk of the data.

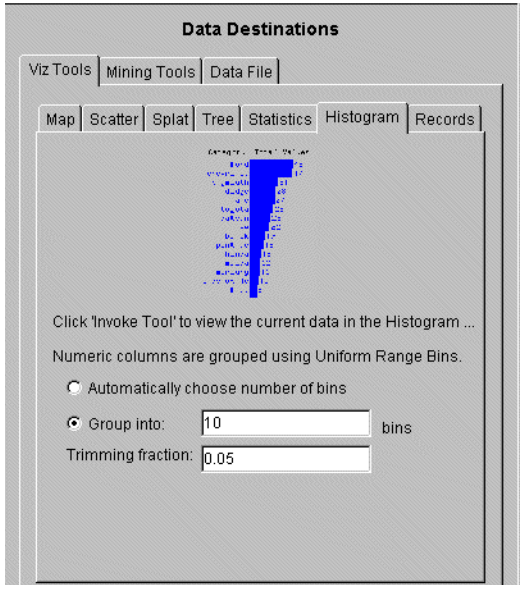


Figure 2-9 Data Destination Panel with the Histogram Visualizer

4. Click *Invoke Tool* in the lower right of the pane.

   The Status window at the bottom of the Tool Manager window shows the progress of the algorithm through the data. The current data is then displayed as histograms, from which you can gain a swift overview of where the data falls. See "Understanding Histograms" on page 23.

Navigating in the MineSet 3D Visualizers

This section describes the navigation controls in the various MineSet visualizers. The tree visualizers have one set of controls, described in "Navigating in the Tree Visualizers," below. The non-tree visualizers have another set of controls, described in "Navigating in Non-Tree Visualizers" on page 29.

Navigating in the Tree Visualizers

The Tree Visualizer display is best thought of as though you are viewing the scene through a camera. To change the view, you change the position of the camera (the viewpoint). This section consists of two tables that serve as a quick reference for the Tree, Decision Tree, Option Tree, and Regression Tree Visualizer controls. Table 2-2 describes the navigation buttons.

Table 2-2 Navigation Icons in the Tree Visualizers

icon	Action
	Returns the chart to the size and position designated as the home view. By default this is the size and position of the chart when the visualizer is first invoked. You can change the home position by using the next icon.
	Sets a new home view for the chart. Use this to save a certain view or position.
	Moves the chart to a position where it is centered and all of it is visible in the window.
	Undoes the previous move (like the Back button on a Web browser).
	Redoes a move that has been undone (like the Forward button on a Web browser).
	Moves one node closer to root of tree.
	Moves one node or bar to left.
	Moves one node or bar to right.
	Moves one node down the tree on the left path.
	Moves one node down the tree on the right path.
	Pops up a menu of possible paths from the current node.

Table 2-3 lists several manipulations you can perform on the scene in the Tree Visualizer. Most of these manipulations can be done either with one of the controls on the visualizer window or with a mouse action.

Table 2-3 Manipulating the Tree Visualizer Scene

Action	Slider or Wheel	Mouse Equivalent
Fly over surface of the scene	N/A	Hold down left and right mouse buttons (or middle mouse button) and move mouse.
Raise or lower bar heights to emphasize differences	Height slider (upper left)	N/A
Move viewpoint up and down	H wheel	Hold down right mouse button, and move mouse up and down.
Move the viewpoint from side-to-side	Side-to-side wheel (<-->)	Hold down left and right mouse buttons (or middle mouse button) and move mouse side to side.
Move the viewpoint backwards and forwards	Dolly wheel	Hold down left and right mouse buttons (or middle mouse button) and move mouse up and down.
Change the up and down tilt of the camera	Tilt wheel	N/A
Move forward in direction you are pointing	N/A	Hold down Alt key and left and right mouse buttons (or middle mouse button) and move mouse. While moving forward, the viewpoint also moves down, based on the current tilt. Similarly, while moving backward, the viewpoint moves up, based on the tilt.
Select a child of a node	N/A	Hold down Ctrl key and click right mouse button on parent node, then click on child to move there (or use branching navigation icon).

Navigating in Non-Tree Visualizers

This section consists of two tables that serve as a quick reference for the Evidence, Decision Table, Map, Scatter, and Splat Visualizer navigation controls. Table 2-4 describes the navigation buttons.

Table 2-4 Navigation Buttons in Non-Tree Visualizers

Button	Name	Action
	Pick	Changes the program to pick mode (an arrow). In pick mode (also called select mode), you can highlight (brush over) or select (click) elements of the chart.
	Grasp	Changes the program to grasp mode (a hand). In grasp mode, you can move the chart around in the window: — To move the chart in the window, hold down the right mouse button while moving the mouse. — To rotate the chart, hold down the left mouse button and move the mouse. —To dolly the chart in and out, hold down the left and right mouse buttons (or the middle mouse button) and move the mouse.
	Home	Returns the chart to the size and position designated as the home view. By default this is the size and position of the chart when the visualizer is first invoked. You can change the home position by using the set home icon.
	Set home	Sets a new home view for the chart. Use this when you want to save a certain view or position.
	View All	Moves the chart to a position where it is centered and all of it is visible in the window.
	Zoom	Moves the point you select to the middle of the pane and zooms to it. When the mouse cursor becomes a targeting sight, move it to the spot you want to see more clearly, then click left mouse button.
	3D	Toggles the 3D perspective.
	Top View	Changes the chart to a top view (Scatter and Splat Visualizers only).
	Front View	Changes the chart to a front view (Scatter and Splat Visualizers only).
	Side View	Changes the chart to a side view (Scatter and Splat Visualizers only).

Table 2-5 describes the adjustment sliders and wheels in the non-tree visualizers.

Table 2-5 Manipulating the Non-Tree Visualizer Scene

Action	Slider or Wheel	Mouse or Keyboard Action
Toggle between Select and Grasp mode	N/A	Press Esc key or the navigation buttons.
Move the scene	N/A	In grasp mode, click and hold the right mouse button. Move the cursor in the direction you want to move the chart.
Raise or lower cake, pie, or bar heights to emphasize differences	Height slider (upper left)	N/A
Rotate the scene around X axis	Rotx wheel	In grasp mode, click and hold the left mouse button. Move cursor in the direction you want to rotate the chart.
Rotate the scene around Y axis	Roty wheel	In grasp mode, click and hold the left mouse button. Move the cursor in the direction you want to rotate the chart.
Zoom the scene in and out	Dolly wheel	In grasp mode, click and hold the left and right mouse buttons (or middle mouse button). Move the mouse down to zoom in and up to zoom out.
Drill down through levels of detail (Decision Table and Map Visualizers only)	N/A	Put the mouse arrow over a specific chart (or the background for all charts) and click the right mouse button.
Drill up through levels of detail (Decision Table and Map Visualizers only)	N/A	Put the mouse arrow over a specific chart (or the background for all charts) and Ctrl-click the right mouse button (or click the middle mouse button).

Brushing in the 3DVisualizer Window

Brushing is a feature that allows you to select the same data points in more than one visualization at a time. When you select something in a visualization that supports the sending of brushing events, it will also become selected in any other visualization that is currently open and that supports the receiving of brushing events. Scatter Visualizer and Evidence visualizer support both send and receive brushing events. Tree Visualizer, Map Visualizer and Decision Table Visualizer support only the sending of brushing events.

Viewing Multiple Documents in the 3DVisualizer Window

All of the Viz Tools allow you to view multiple documents in one window. A document can be a MineSet visualization, a Word document, or any other viewable file. Documents can be opened from the File menu, or dragged and dropped into the 3DVisualizer window. When a document that is not a MineSet visualization is displayed in the 3DVisualizer window, some menu items are disabled. By default, the 3DVisualizer window only allows one document at a time. To change to multiple documents:

1. Choose File > Preferences. The preferences Dialog box opens (Figure 2-10).

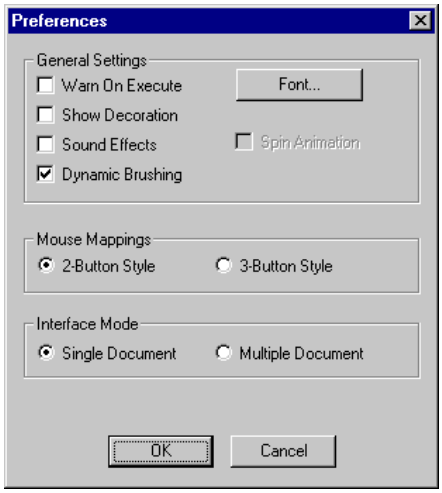


Figure 2-10 3D Visualizer Preferences Dialog

2. In the Preferences dialog box, check the Multiple Documents radio button.

Note: This change will not take effect until the next time you open the Viz window.

Figure 2-11 shows a 3D Visualizer window with both an Option Tree and a Decision Tree visualization.

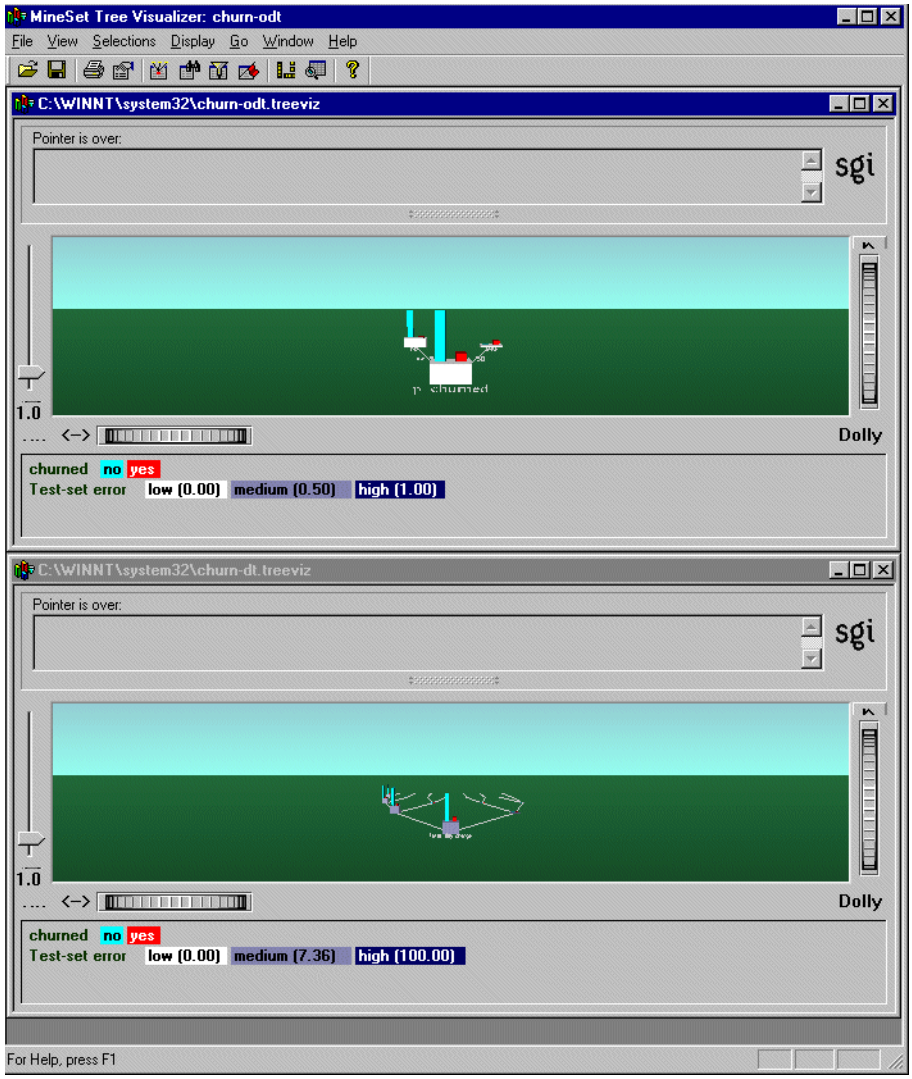


Figure 2-11 Multiple Visualizations in Viz Window

Getting Help

Besides the sources listed in “Finding MineSet Information” on page xvii, other sources of information about MineSet are:

- *MineSet Enterprise Edition Tutorial* for a quick introduction to the use of the tools.
- Context-sensitive help is available within each tool: press F1 over an area of interest.
- Online demo movies give a quick overview of some of the tools from http://mineset.engr.sgi.com/movies.

Shaping the Data

This chapter introduces you to the concepts of shaping data using MineSet Tools. These subjects are presented as preparation for the task of creating a classifier, which is covered in the next chapter. This chapter includes the following sections:

Why Shape the Data?

Confronted with an unfamiliar dataset, your first move might be to explore with the Record Viewer and Statistics Visualizer. (See Chapter 2, "Accessing Data with MineSet.") In this chapter the process continues by shaping or transforming the raw data. For example, you may find you have too much information to make sense of. One the easiest ways of simplifying a visualization is by adding, removing, or combining columns. This manipulation does not change the underlying data, merely the session file on which you are currently working. The sequence of transformations that you apply will be referred to as the history for that Tool Manager session.

The Tool Manager is the graphical user interface (GUI) you use for most of your interactions with the MineSet components. With Tool Manager you can select an existing data source, transform or analyze that data, and visualize the results using any of the MineSet individual tools.

For detailed information on how to interact with MineSet visualizers, see the specific tool entry in the *MineSet Enterprise Edition Reference Guide*.

Note: The Tool Manager may require some manual work to make data files not created by MineSet compatible; see the *MineSet Enterprise Edition Interface Guide*.

Transforming Data with the Tool Manager

The functions of the buttons shown in Figure 3-1 are indicated in Table 3-1. To transform data, select the column from the Current Columns pane, then click the appropriate button.

1. Start Mineset (refer to “Starting MineSet” in Chapter 2).
2. From the Tool Manager window select a dataset using the File > Open New Data File pulldown menu. Click on a file in the Open Data File dialog, and the dataset’s column headings appear in the *Current Columns* window of the Data Transformations pane (Figure 3-1).

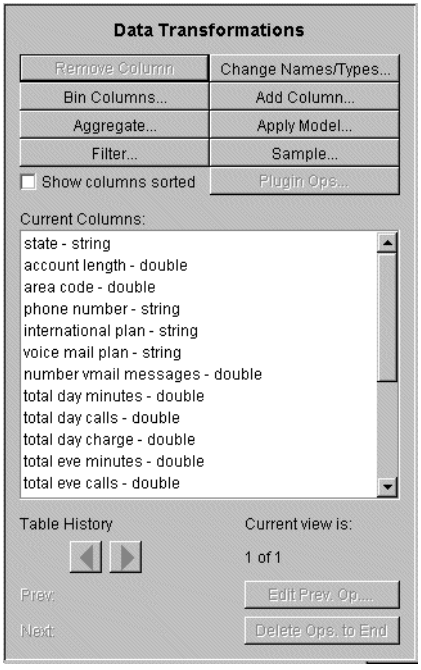


Figure 3-1 Data Transformations Pane

Table 3-1 Button Functions on Tool Manager's Data Transformations Pane

Button Title:	Action:
Remove Column	Deletes all currently selected columns.
Bin Columns	Assigns each record to a certain range (bin).
Aggregate	Performs sum, average, min, max, or count aggregation of some columns, while grouping by others. It is also possible to array index the aggregated columns.
Filter	Creates a subset of the data based on a boolean expression. Only records for which the expression evaluates to true are kept.
Change Types	Changes a column's name as well as type, for example from float to int. (See Table 3-3 for an explanation.)
Add Column	Adds a new column based on a mathematical expression, which may include other columns.
Apply Model	Labels new records with an existing model or backfits data.
Sample	Selects a random subset of the data in large datasets.
Show Columns Sorted	Sorts the columns of the dataset alphabetically.

Removing and Adding Columns

To remove or add a column, do the following from the Tool Manager Data Transformations pane:

Remove Column—Select a column that is not relevant to your visualization or mining, then click the Remove Column button. Deleting extraneous columns may save computational time. To choose multiple non-contiguous columns for simultaneous removal, hold down the Ctrl key while selecting the additional columns. Use the Shift key for contiguous columns.

Add Column—Click the *Add Column* button to get a dialog box to specify a new column name and mathematical expression (Figure 3-2). For example, you could add a new column called "minor_age" based on the column "age," using the expression: "if age is less than or equal to 18 then minor_age is true; else minor_age is false." Such and expression is written "if ('age' <= 18) then ('minor_age' = 1) else ('minor_age' = 0)."

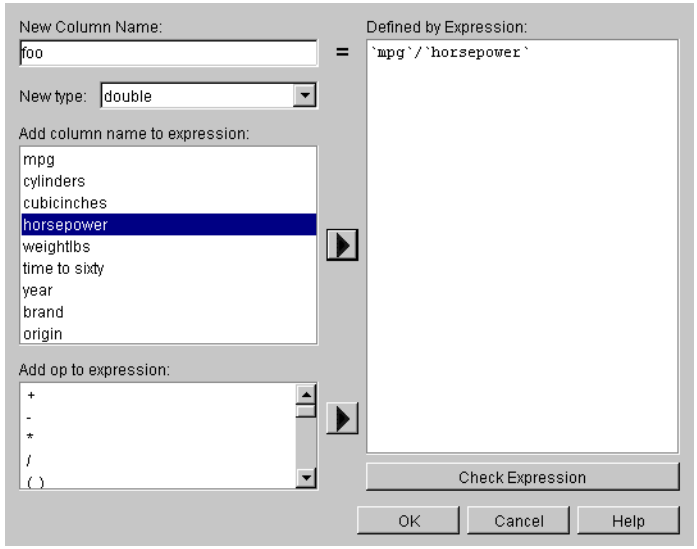


Figure 3-2 Add Column Dialog Box

To accomplish this, when you click the Add Column button, a dialog box appears (Figure 3-2).

1. Enter the new column's name in the dialog box, and use the popup menu below it to specify the column type (integer, string, floating point, and so on). Column names should contain no spaces.
2. Fill in the definition of the expression in the large text entry area:
   - Use the scrolled lists in the lower left to quickly select from available columns and operators.
   - To insert a column name or operator into the expression, either double-click it in the scrolled list, or select it and click the arrow to the right of the list.
3. Click the *Check Expression* button to check that the expression syntax is correct. Click OK to dismiss the status box. Click OK again to dismiss the Add Column dialog box.

The Add Column dialog box checks for type compatibility: if you have assigned a numerical expression to a string column (or vice versa), a warning message appears, and the type of the new column is automatically changed to be correct.

Changing or Creating New Bins for Columns

To organize a model or visualization, records must often be grouped into bins, especially to create the sliders that are needed for an animation. You can change bin boundaries, and experiment with the number and range of columns to make a visualization less complex.

For example, if you wanted to break up a range of ages into: 0-18, 19-30, 31-50, 51-60, 61+. those ranges are expressed as (...18], (18 ... 30], (30 ... 50], (50 ...60], (60 ...]. The open parenthesis next to a number indicates it is not included in the range. A square bracket next to number indicates it is included. See also the glossary entry for *Binning*, and the "Binning" entry in the *MineSet Enterprise Edition Reference Guide.*

From the Tool Manager Data Transformations pane, click *Bin Columns* to get a dialog box to specify the binning options (Figure 3-3).

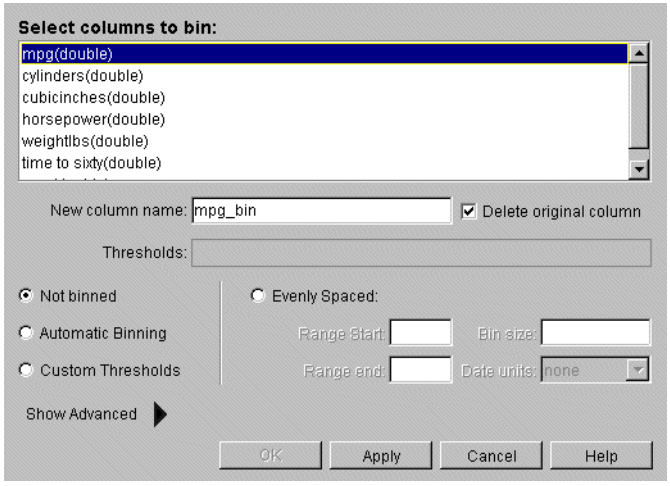


Figure 3-3 Bin Columns Dialog Box

1. In the top pane of the dialog, click the column that is to be divided into bins. The default is "Not binned." Choose a binning method and the name of the binned column appears in the "New column name" field, or you can type in a new name. If you select more than one column for binning, "New column name" stays inactive.

   To delete the original column, ensure that the box labeled "Delete original column" is checked.

2. Choose Automatic Binning, Custom Thresholds, or Evenly Spaced (meaning evenly spaced bins).
   - Automatic Binning lets MineSet suggest the bin thresholds using machine learning.
   - Custom Thresholds allows you to specify bin thresholds in the Thresholds text field.
   - Evenly Spaced lets you specify the Range start and end, and the size of each bin.

   When you click the "Show Advanced" arrow, the Advanced Options Binning panel is displayed (Figure 3-4).

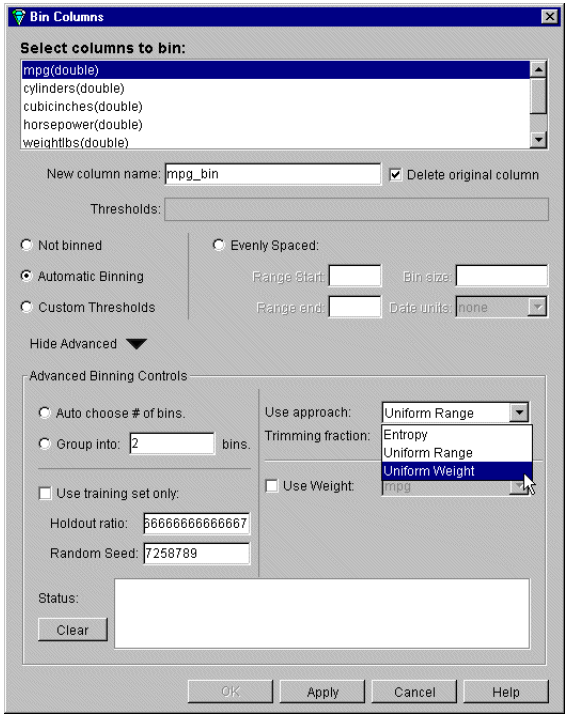


Figure 3-4 Advanced Binning Options Panel

3. From the Advanced Binning Options panel you can make any of the following choices:

Table 3-2 Advanced Binning Options

Selection	Resulting Action
Auto choose # of bins	MineSet automatically decides the best number of bins.
Group into ___bins	The dataset is divided into the number of bins you specify.
Use Approach	This menu gives a choice of three approaches to creating bins.
Entropy	Picks bins so that the label values within each bin are similar, thus minimizing entropy or disorder among binning labels.
Uniform Range	Divides the dataset into bins evenly distributed across the range.
Uniform Weight	Divides the dataset into bins each with the same weight (or number of instances) of data per bin.
Trimming fraction	(Only available from Uniform Range and Uniform Weight). Lets you eliminate outliers from bin thresholds. Outliers are single instances that lie outside the bulk of the data.
Use training set	Divides the dataset into subsets and uses only the holdout ratio.
Holdout ratio	Specifies the percentage of the dataset to be used to create the binning.
Random Seed	Specifies where to start dividing the dataset. The random seed number changes with the time. If you want to get exactly the same data on your next run, you must specify the same random seed.

When you choose *Custom Thresholds* to set your own binning thresholds you can also enter:

- *Thresholds*

  In the text field, enter threshold ranges separated by commas, with a parenthesis next to the number that is not included in the range, and a square bracket next to the number that is included. For example, all ages above 18, not including 18 itself, up to and including 30 is expressed as (18 ... 30].

  A date entry in the Use custom thresholds field requires double quotations around it. If you enter dates without quotation marks they are added automatically.

  `“1/1/99”, “2/1/99”, “3/1/99”, “4/1/99”, “5/1/99”, “6/1/99”`

  You can also use the *mm/dd/yyyyy* format (for example: “05/22/1999”).

- *Evenly spaced*

  When you click this button the text fields below it become active. Enter the start of the binning range, the end of the range, and the size of the bins into the appropriate fields.

  Do not put quotation marks around dates used with *Evenly spaced* thresholds. You can use both date formats (1/1/96 or 01/01/1999) in this field.

Further details can be found in the "Binning" entry in the *MineSet Enterprise Edition Reference Guide*.

Creating New Columns by Aggregation

You can create new columns out of the contents of existing columns by aggregation. For example, using the churn dataset you can create a subset of the data in a column containing the average churn per state. You can also distribute the contents of the columns using various criteria. Because aggregation reshapes the data, it is important to refer to the *MineSet Enterprise Edition Reference Guide* for background on arrays and distribution for a fuller understanding of this feature. See the glossary definition for *Aggregate*.

The process of aggregation will reduce the overall number of rows in a dataset by grouping sets of rows in the original dataset into new aggregate rows.

To create aggregations, follow these steps:

1. From the Tool Manager Data Transformations pane, click *Aggregate* to get a dialog box to create simple aggregations, make arrays, or distribute columns (Figure 3-5). The columns of the current dataset initially appear in the central "Group By Columns" text field.

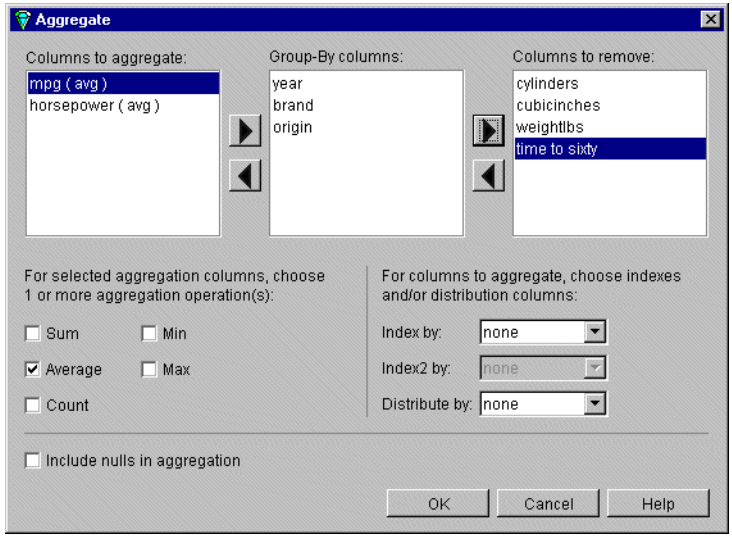


Figure 3-5 Aggregate Dialog Box

2. Select a column to aggregate, and click the left arrow button to move the column left. Leave the "Group By Columns" entries in the center (the columns by which you want to index), and move all the rest to the right column. (Hold down the Ctrl key while selecting to gather multiple columns.)

   Columns to aggregate—move columns here to create an aggregate. Values in the aggregated columns are summed, averaged, given a minimum or maximum, or counted, depending on the boxes checked at the bottom of the panel.

   Group-By columns—columns remaining here are unchanged throughout the operation. For each set of records with the same combination of values in the Group-By columns, only one record is output in the resulting table.

   Note: If you have many columns or numeric columns in the group-by list, you may get more rows in the result than you expect. Typically, a few columns having few values each are used in the group-by list.

   Columns to remove—columns removed are not used in the aggregation.

3. Click the check boxes under the columns to specify how values are to be combined in the aggregation: summed, averaged, the minimum or maximum value, or counted. Minimum and maximum are rarely used.

   For numeric columns choose any combination. For other types, only Count is permitted. Each option gives a separate result; selecting average and max gives you one column with average values, and another one with maximum values. The default column name identifies the process used to create it—`count_state`, `avg_weightlbs` and so on.

4. Use the popup menus in the lower right to specify indexes (if the result is to be an array) and a distribution column (if the result is to be distributed).

   Arrays are useful for creating sliders, which are needed for the animations available in several of the visualizers. Columns to be indexed in an array must have been previously binned in order to show how one characteristic varies according to (indexed by) another. See the "Aggregate" entry in the *MineSet Enterprise Edition Reference Guide* for further explanation of arrays and distribution.

5. Click OK.

6. Make sure the Current Columns text box in the Data Transformations window shows the new column names that result from applying your criteria.

An example of using aggregation is detailed in the *MineSet Enterprise Edition Tutorial*, distinguishing the average churn by state.

Restricting Column Contents by Filtering

You can select a subset of the data based on an expression involving column values, for example, you can leave only those records in which the age is less than 20, or miles per gallon is less than 25. Once you have loaded a dataset using the Tool Manager, you can begin filtering by following these steps:

1. From the Tool Manager Data Transformations pane, click *Filter* to get a dialog box to filter data using a mathematical expression (Figure 3-6).

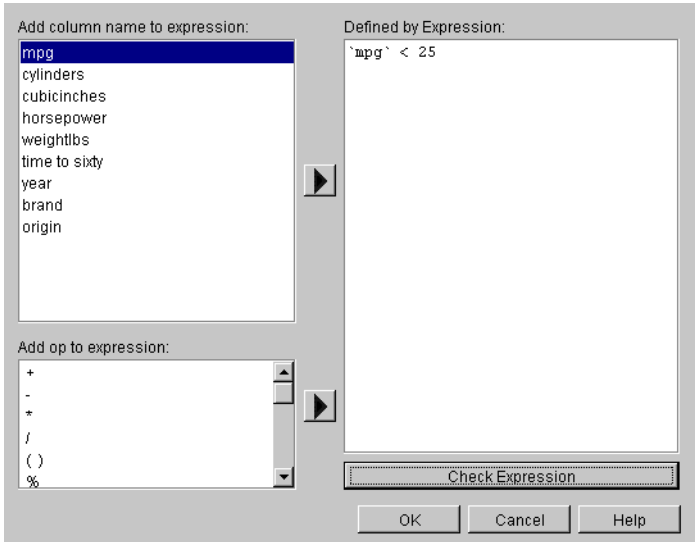


Figure 3-6 Filter Dialog Box

2. Select a column name and operators from the left side to build an expression on the right, or you can type it directly in the Defined by Expression text field. The resulting table includes only records for which the expression is true (or, if numerical, non-zero). See also the "Filter Panel" entry in the *MineSet Enterprise Edition Reference Guide*. For a complete description of the expression definition language, see the *MineSet Enterprise Edition Interface Guide*.

Changing a Column Type or Name

Some databases store numerical values as strings. Oracle databases store all numbers (both integers and real numbers) in a single format, which defaults to the data type *double* in the Tool Manager. Use the *Change Types* button to ensure that these values are processed correctly. Use the same button to change column names.

Changing a Column Type

1. From the Tool Manager Data Transformations pane, click *Change Names/Types*. A dialog box appears showing column headings and their types (Figure 3-7).
2. Select a column that you can convert, and click the *New type* menu. The list shows possible column types (invalid types are shown in gray). Explanations of the choices are shown in Table 3-3.

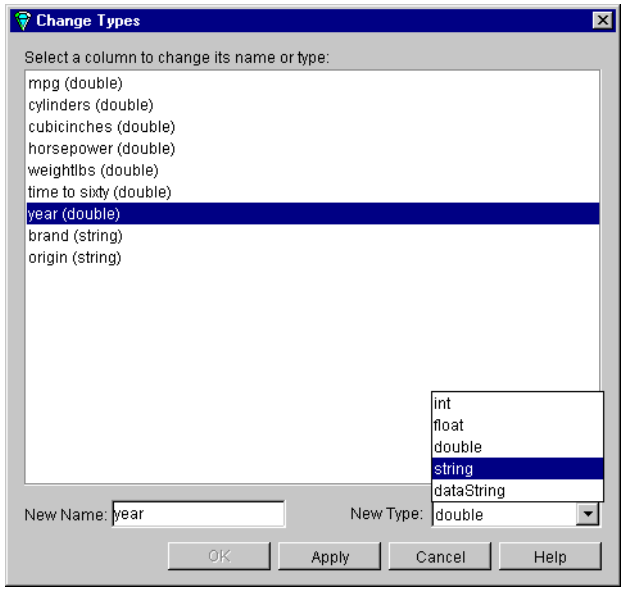


Figure 3-7 Types Popup List

Table 3-3 Column Type Meanings

Type	Meaning
int	A 32-bit signed integer.
float	A single-precision floating-point number. The decimal point is optional.
double	A double-precision floating-point number. The decimal point is optional.
dataString	A *dataString* is a string whose value is not kept in a common string table, thus every data string value is stored independently, even when multiple values are identical. Seldom used.
string	A string of characters that is stored in a common string table, which saves memory. If you are unsure whether to use a *string* or a *dataString*, use a string.
date	Specified as the date type from the database.
bin	A column created by a binning operation.

3. After selecting a new type, click *Apply* to have the change take effect.
4. Click *Close* to exit the dialog box.

If you try to convert an inappropriate field (such as a name) to a number, the resulting values are all zeroes.

Changing a Column Name

Changing a column's name is done from the same dialog box as changing column type.

1. From the Change Types dialog box, select the column you want to rename, type a new name in the New Name text field, and click *Apply.*
2. Then click *Close* to exit the dialog box.

Applying a Model

If you have previously created a model, you can use the *Apply Model* button to label new records in the current table, to estimate probabilities for a label value, to test the performance of the model on the current table, or to backfit the current table onto an existing model. See Chapter 7, "Understanding Predictive Modeling" for examples.

Sampling Data

If your dataset is too large to work with efficiently, you may want to work with a randomly sampled subset. After you have loaded your data file, from the Tool Manager Data Transformation pane:

1. Click the *Sample* button to bring up the Sampling dialog box (Figure 3-8).
2. Choose one option:

   Click the Percentage check box and type in the text field the percentage of the current dataset you want to be sampled. The percentage is approximate.

   Click the Count check box and establish the number of records you want in the sample.

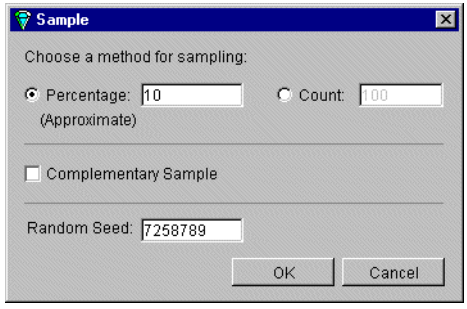


Figure 3-8 Sampling Dialog Box

3. Click the Complementary Sample check box to get all records except those that fall in the random sample. For example if the Complementary Sample box is not checked when you request a 10% sample, when you check the box, you get the remaining 90% of the data. Click OK.
4. (optional) Enter a seed number in the Random Seed field on which to base the random sample. If no seed is specified, a number based on the current time is used as the seed. If you want a different random sample, specify a different random seed. Using the same seed for different mining explorations of a dataset allows you to work with the same random sample each time. Change the seed when you want to test how well a discovery holds up across the dataset. See also the glossary entry for *Random Seed*.

Retracing Your Steps with Table History

To see the history of previous operations, and go back if you change your mind or made a mistake, there are two *Table History* buttons at the bottom of the Data Transformations pane. Click the left arrow button to show the table at an earlier step. Click the right arrow button to return the table to its current state. See Table 3-4 for an explanation of the button meanings.

Table 3-4 Table History Meanings

Field	Meaning
Current view is	Counts changes made and indicates which step you are viewing.
Prev: and Next:	Notes the actions taken, helps you keep track.
Edit Prev. Op	Brings up dialog box for editing the operation shown in the *Prev:* field. Be careful about deleting operations used in subsequent operations.
Delete Ops. to End	Removes all operations after the current step. If you are not already at the end of the history, the Data Destination panel becomes active again after clicking the button.
History of Operations View tab	Brings up a complete history of the Data Transformation table (Figure 3-9).
Single Transformation and Data Destination View tab	Takes you to a single point in the history, presenting you with Tool Manager at that state of affairs.

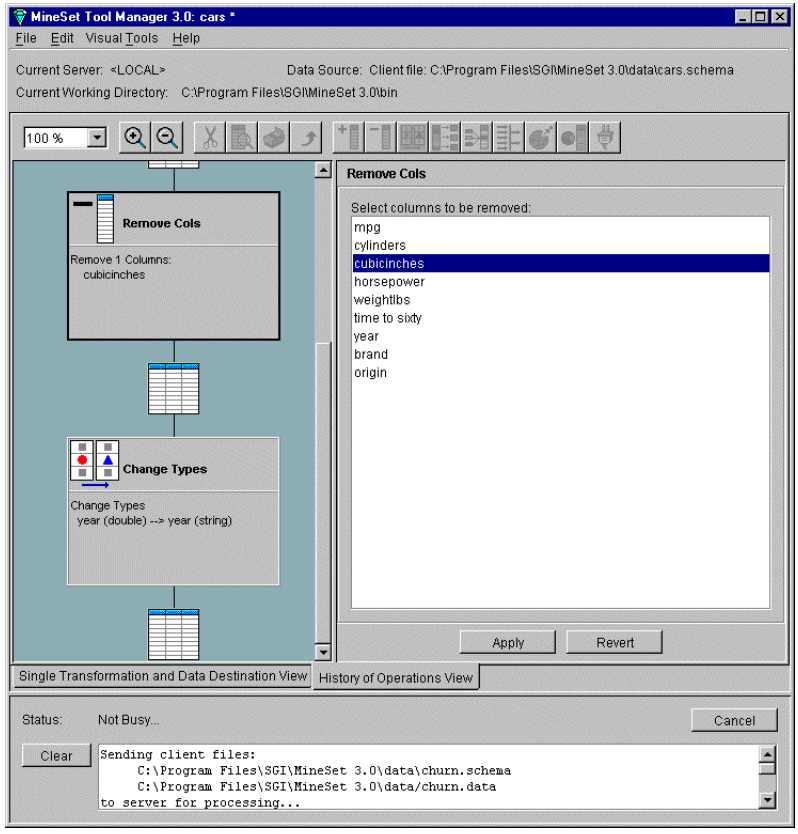


Figure 3-9 View History Dialog Box

As with *Edit Prev. Op*, changing one operation usually affects (sometimes invalidates) subsequent operations in the history. You can select a specific operation to edit, add, or view. The History of Operations view warns you when changes affect the history, and shows you the new history as it occurs. Click on the diagram to return to a previous operation, or click on the table between operations to get to the table between states.

To return to your previous position in the Tool Manager, click the Single Transformation and Data Destination View tab.

Weighting Records

Among other ways of transforming data, weighting records lets you make some records more important than others, or less important, if the need arises. As an example, a phone company stores all fraudulent phone calls in the dataset, while storing only a small fraction of non-fraudulent calls. By using record weighting, it is possible to give each record its true proportion of the total population of customers.

Some datasets are already aggregated, and the records have a natural "count" associated with them (for example, statistics about cities in the U.S. usually have an associated count of the population). This count attribute can be mapped to weight, which is equivalent to replicating each record by the number of counts.

The semantics of record weighting is that a record weight of 2 is equivalent to two records with a record weight of 1. Floating-point weights are allowed.

The weighting of records can be done using the *Advanced Options* dialog for all inducers (see the glossary definition for *Inducer*). After you have loaded your dataset, weight records by following these steps:

1. In the Tool Manager Data Destinations pane, choose Mining Tools, and click the Classify tab.
2. From the Mode popup menu, choose any inducer.
3. Click the *Advanced Options* button, and from the resulting dialog box, click the Use Weight check box. This means the records will be weighted by the selected attribute chosen in step 4.
4. Choose the column for the weight.
5. Click the "Weight is Attribute" check box if the attribute used in step 4 is also to appear as a normal attribute. If the box is unchecked, the weight attribute will have no influence over the induction process.
6. Click *OK* to accept these options, and *Go* to run your chosen inducer.

Finding Important Columns

Column importance helps you discover which are the most important columns in predicting different values for a label column you choose. The difference between column importance and clustering (refer to Chapter 12, “Segmenting the Data with Clustering”) is that with column importance, you decide which label you will use to determine the importance of columns. In clustering, the data itself shows you which are the discriminating factors, but does not supply labels.

Use column importance for example, to find the best three columns for mapping to axes in the Scatter Visualizer. You simply select the label and run the tool, and a window appears with the three columns that are the best three discriminators. A measure called “purity” (a number from 0 to 100) informs you how well the columns discriminate the different labels. Adding more columns increases purity.

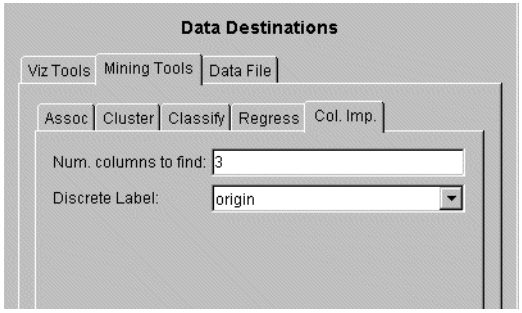


Figure 3-10 Column Importance Tab

1. From the Tool Manager Data Destinations pane, choose the Mining Tools tab and click the Col. Imp. (Column Importance) tab shown in Figure 3-10.
2. To run in simple mode, choose a discrete label from the popup menu, specify a number in the “Num. columns to find” field, then click *Go.* The bottom status window gives the result.

   To control the choice of columns, click the *Advanced Mode* button and the Column Importance dialog box appears, (see Figure 3-10).

3. In the dialog box, you may click the Use Weight check box and select a column from the “Importance with respect to column” popup menu. This helps decide whether the column behaves as a regular attribute for determining importance. See also “Weighting Records” on page 51. For details of tool operation see the “Column Importance” entry in the *MineSet Enterprise Edition Reference Guide.*
4. The dialog box contains two lists of column names: the left list contains the available attributes and the right list contains attributes chosen as important (either by you or the Column Importance algorithm). You move a selected column from left to right by clicking the right-pointing arrow.

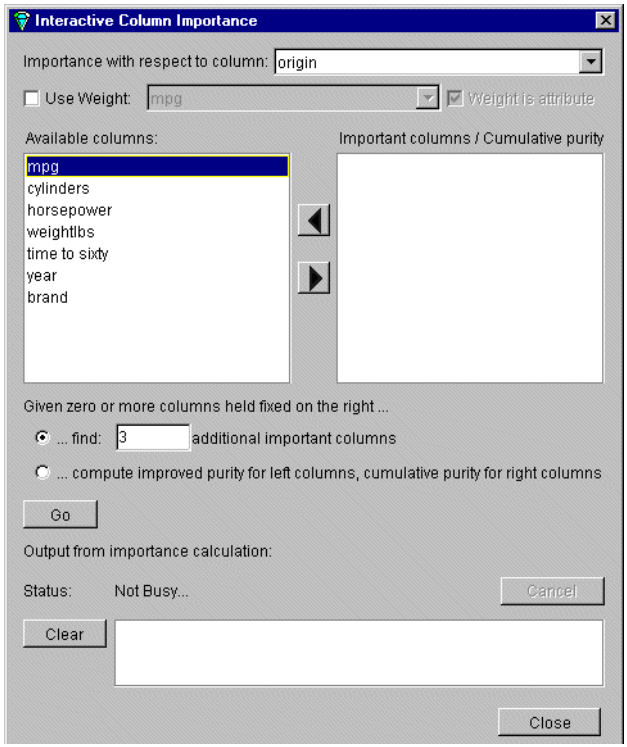


Figure 3-11 Advanced Mode of Column Importance

5. In the middle section of the dialog box are two alternatives in advanced mode:

   - Find Several Important Attributes

     Click the button at *...find [number] additional important columns*. If you click *Go* with no further changes, the effect is the same as in simple mode. The specified number of important columns are found and moved to the right column and cumulative purity is given.

     However, if you move column names from the left list to the right by clicking on the right-pointing arrow, you can specify columns to be included and let the system add more.

     Click *Go* to see the cumulative purity of each column, together with the previous ones in the list. A purity of 100 means that using the given columns, you can perfectly discriminate the different label values in the dataset.

   - Rank Available Attributes

     You can test the gain in purity that each column adds to those already in the list on the right. If you use the cars dataset, as shown in Figure 3-11, you can move the column *cylinders* to the list on the right, and then ask the system to compute any incremental improvement in purity that each column remaining in the left column yields. The cumulative purity is computed for columns on the right (already marked important). The output is displayed in the right column.

     Click the button at *...compute improved purity for left columns, cumulative purity for right columns.* This submode permits fine control over the process. If two columns are ranked very closely, you might prefer one over the other (for example, because it is cheaper to gather, more reliable, or easier to understand).

Column importance is useful for finding the best three axes for the Scatter and Splat Visualizers. It is also useful when using the Tree Visualizer, to find a good discriminatory hierarchy (hierarchy that separates different label values), when you select the label to be the key for the Tree Visualizer.

See the “Column Importance” entry in the *MineSet Enterprise Edition Reference Guide* for details on using a sample file with Column Importance.

Examining Data with the Scatter and Splat Visualizers

The Scatter and Splat Visualizers look and feel similar, with the distinction that the Scatter Visualizer shows data as a series of individual entities, while the Splat Visualizer shows aggregates of data points appearing as clouds of varying opacity and color in a three-dimensional landscape. To learn more about the finer points of each, see the *MineSet Enterprise Edition Reference Guide*. This chapter covers the following topics:

Sample configuration and data files are provided to demonstrate the Scatter and Splat Visualizer's features and capabilities. These files are in the *\examples* directory under *MineSet*, in the location where MineSet was initially installed.

Overview of the Scatter and Splat Visualizers

This section explains the use of the Scatter and Splat Visualizers and some of the reasons for choosing one over the other.

Overview of the Scatter Visualizer

The Scatter Visualizer's individual datapoints correspond to rows in the data file. This visualization works well when the number of datapoints is less than 50,000, or when some processing has been performed so that the data is reduced to a small set of aggregates. The Scatter Visualizer produces scatterplots that can be animated to show relationships more clearly (Figure 4-1).

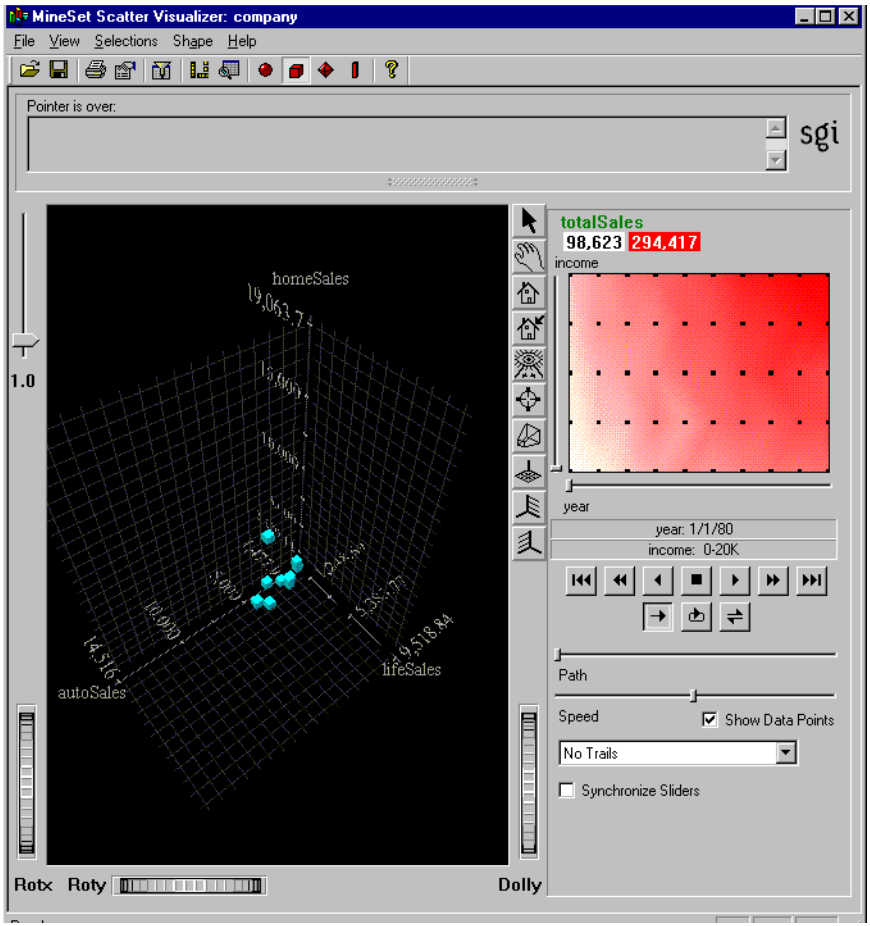


Figure 4-1 Sample Scatter Visualizer Screen

The Scatter Visualizer displays a three-dimensional landscape with columns of data mapped to entities, and to elements such as the axes, size and color. If you map one or two numeric variables to the sliders, you can animate the size, color, or position of the entities. In the example in Figure 4-1, the data represents the sales of several companies over time. If the time variable is mapped to a slider and the sales variable is mapped to size, then the entities grow or shrink as the time slider is animated. This example is *MineSet\examples\company.scatterviz*.

As you play back the animation path, you can watch the changing size, color, and motion of the datapoints for trends or anomalies. You can travel through the three-dimensional landscape to find an interesting point of view, or scale the values of variables for greater emphasis. To clarify the scene, you can filter the display to show only those entities meeting certain criteria.

Overview of the Splat Visualizer

With the Splat Visualizer you can visually analyze relationships among several variables, with some relationships seen even more clearly when you use the animation feature. The Splat Visualizer uses graphical objects, called *splats*, which represent aggregates of data points. The color and opacity, but not the position, of the splats can change during animation.

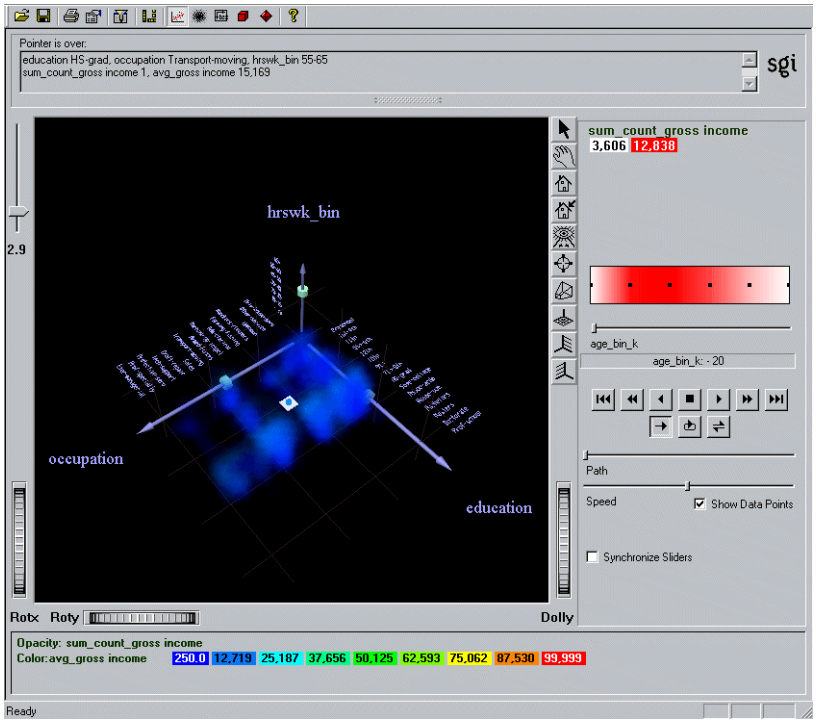


Figure 4-2 Sample Splat Visualizer with One-dimensional Summary Slider

Figure 4-2 shows a Splat Visualizer view of a three-dimensional landscape with columns from the *adult94* sample dataset mapped to axes, sliders, color, and opacity. It is similar to a scatterplot, except a scatterplot draws every datapoint separately, and the Splat Visualizer aggregates data points that are close together (fall in the same bin) and draws them as a single splat. The result approximates the image obtained if you rendered each individual point in a scatterplot. The resulting image can be thought of as a 3D color histogram.

From the Splat Visualizer results you can:

- See global shifts and trends in the data, using the animation panel. Changing color and opacity give the illusion of actual movement (see the online version of this book for the color version of the figure).
- Emphasize particular dimensions or a point of view, by flying over the three-dimensional landscape.
- Enhance visibility using the scale slider (top left of the Main Window) to lower or increase splat opacity. The regions with dense data are likely to show less color variation, because the color is based on the average of many values.
- Filter the display to show only those splats meeting certain criteria. You can filter on the columns corresponding to axes, sliders, weight, and color.
- Pick out textual information about individual splats in the volume.
- Define a selected region with a box selector for drilling through to the original data or for sending to the Tool Manager.

For example, the left axis in Figure 4-2 shows each occupation sorted by average income along an axis. The occupation executive-managerial, listed at the end of the axis, has the highest average income, and provides a natural progression for the values. On the other hand, the ordering for the values of *education* (the right axis in Figure 4-2) is generally from low to high; but in a few cases, there are anomalies in the order. This unexpected ordering might be interesting because it points out places where the data does not agree with expectations.

Transforming the Data for the Scatter and Splat Visualizers

Considerable planning and knowledge of your data go into data transformation decisions. The columns that make up the dataset often require manipulation or transforming in some way before they can be mapped to an axis, color, or slider. You may often need to go back and transform the data differently after viewing a result.

In MineSet most of these transformations take place using the Data Transformations pane in the Tool Manager, (see "Transforming Data with the Tool Manager" on page 36). Some transformations you might want to consider are:

- Adding columns, usually applying a mathematical formula to existing data.
- Removing columns that are not pertinent, are redundant, or contain obvious, uninteresting predictors. This is useful to reduce the size of the dataset and speed processing.
- Filtering the data. For example, you may want to use only records containing a certain range of values for a particular attribute.
- Changing a column's name or type.
- Binning data—breaking up a continuous range of data into discrete segments.
- For the Splat Visualizer only, aggregating data—if you do not aggregate data manually with the Tool Manager, it will be done automatically in the tool. However, it is much better to do it in the Tool Manager so that the processing is done on the server, and the aggregated dataset is not brought to the client system (see "Techniques of Processing for the Splat Visualizer" on page 61).
- For the Scatter Visualizer only, aggregating data—aggregation can be used to reduce a large dataset to a small set of aggregates. Each row in the table after aggregation will correspond to an entity in the scatterplot. Array columns for animation may be created during the aggregation step.

The interactivity of the visualization in the Splat Visualizer is independent of the number of data points represented; it depends only on the number of bins in the axis dimensions. If your dataset is very large, aggregate explicitly in the Tool Manager. This lets the server perform the processing using a streaming operation, rather than having the entire dataset sent to the client and aggregating there (see "Techniques of Processing for the Splat Visualizer" on page 61).

Table 4-1 lists the column types that you can map to visual entities in the Splat Visualizer.

Table 4-1 Allowable Column Types for Mappings in the Splat Visualizer

Entity	Column Type	Requirements
Axis	Numeric	Will be binned, either manually or automatically by Tool Manager.
	Binned	Can be mapped directly.
	String	Can be mapped directly.
Color	Numeric	Can be mapped directly.
Opacity	Numeric	Defaults to record counts if nothing is mapped.
Slider	Numeric	Must be binned, either manually or automatically by Tool Manager.
Summary	Numeric	Can be mapped directly.

Table 4-2 lists the column types that you can map to visual entities in the Scatter Visualizer.

Table 4-2 Allowable Column Types for Mappings in the Scatter Visualizer

Entity	Column Type	Requirements
Axis	Any, including array	Can be mapped directly.
Entity - size	Numeric or array	Can be mapped directly.
Entity - color	Any, including array	Can be mapped directly.
Entity - label	Any, including array	Can be mapped directly.
Sliders	Binned	Can be mapped implicitly during aggregation.
Summary	Numeric	Can be mapped implicitly during aggregation.

If a string column is mapped to an axis, binning is defined to be the distinct values of that column. Those values are ordered so they correlate with the attribute mapped to color (if nothing is mapped to color, then they correlate to opacity). Looking at the color changes along a string-valued axis lets you see how well that column correlates with the column mapped to color.

Techniques of Processing for the Splat Visualizer

You can avoid processing on the client by aggregating in the Tool Manager. This means most of the work is done on the server. The server is typically a more powerful machine than the client, and performs a streaming operation so that the full dataset does not need to be loaded into memory. Furthermore, aggregating in the Tool Manager avoids having to transfer a large dataset to the client. This can be done by:

1. Use Tool Manager to bin the numeric columns to be used for axes and sliders.
2. Aggregating the column to be mapped to color by count and average, while grouping by the axis and slider columns.
3. Mapping the resulting count aggregation to opacity.
4. Mapping the resulting average aggregation to color.

Using the *adult94* dataset (provided with the distribution) as an example, the steps of the process shown above are listed below.

1. Bin `age` and `hours_per_week`. You will be using the former on a slider, which needs binned values, and the latter as an example of binned numerical values on an axis.
2. Aggregate `gross_income` using Count and Average. Keep `education`, `occupation`, `age_bin` and `hours_per_week_bin` in the Group-by Columns pane while removing all the other columns. This gives a visualization showing how `gross_income` varies according to the three other criteria.
3. Map `education`, `occupation`, and `hours_per_week_bin` to the axes.
4. Map `avg_gross_income` to color, `count_gross_income` to opacity, and `age_bin` to a slider. This mapping allows color and opacity to show variations in income (average and count) according to age, depending on the criteria mapped to the axes.

When you invoke the tool, note that all the processing is done on the server, and that the datafile, *adult94.splatviz.data*, contains rows that are aggregates of rows in the original data. This produces the same visualization as seen in Figure 4-2.

Starting the Scatter and Splat Visualizers

The simplest way to start the Scatter or Splat Visualizer is from the Tool Manager:

1. Bring up Tool Manager, and choose a server and data source. This is explained in "Starting MineSet" on page 15. In the first example, the *churn* dataset is used.
2. In the Data Destinations pane of Tool Manager, choose the Viz Tools tab; from the lower row of tabs, choose Scatter Visualizer or Splat Visualizer, (Figure 4-3).

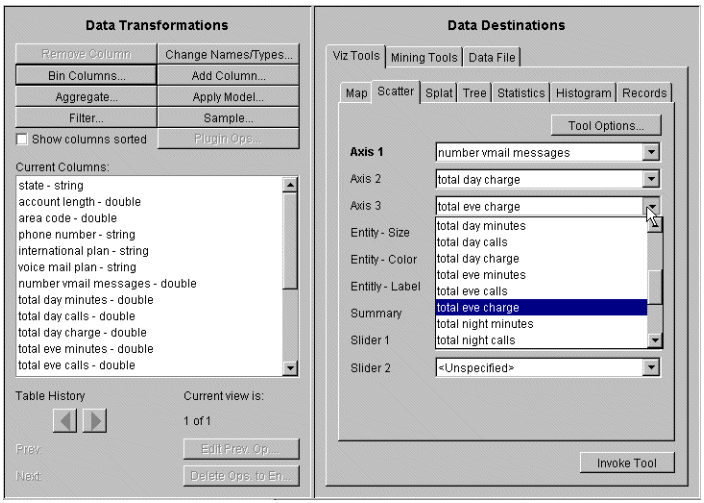


Figure 4-3 Data Destination Panel with the Scatter Visualizer Selected

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Mapping the Scatter Visualizer Visual Elements

1. In the Data Transformations pane of Tool Manager, map Current Columns to Visual Elements by selecting from the scrolling menus. For example, for the Scatter Visualizer, if you are using the *churn* dataset, for Axis 1 choose `number of vmail message`, Axis 2 choose `total day charge`, and Axis 3 `total eve charge`

   Table 4-3 identifies the mapping effects of the Scatter Visualizer's Visual Elements.

   Table 4-3 Mapping Visual Elements in the Scatter Visualizer

Visual Element	Action:
Axis 1 Axis 2 Axis 3	Assigning data to the first axis alone is usually not useful. Assign data to all three axes to produce a three-dimensional chart.
Entity-Size Entity-Color Entity-Label	Assigns a size, color, and label to entities. Specify options for these mappings with the ScatterViz Options dialog box, available when you click the Scatter Visualizer Tool Options button.
Summary	Determines the color of the summary slider's background.
Slider 1 Slider 2	Maps columns directly to one or two animation sliders.

2. To undo a mapping, select the element, then select `<unspecified>` from the list of available columns. If you are using your own dataset, consider running the Column Importance tool to help determine possible mappings (see "Finding Important Columns.")
3. Click *Invoke Tool* in the lower right of the Data Destination pane.

Mapping the Splat Visualizer Elements

Like the Scatter Visualizer, the Splat Visualizer requires that columns be linked to visual elements before being launched. Some columns may require manipulation before they fit the visual elements criteria. This example uses the mushroom dataset from the provided examples to show how to deal with manipulating columns to fulfill mapping requirements. Begin from the Tool Manager as described in "Starting the Scatter and Splat Visualizers" on page 62, choosing the mushroom dataset and selecting the Splat Visualizer.

1. In the Data Transformations pane of Tool Manager, map items in Current Columns to items in Visual Elements by choosing from the scrolling menus.

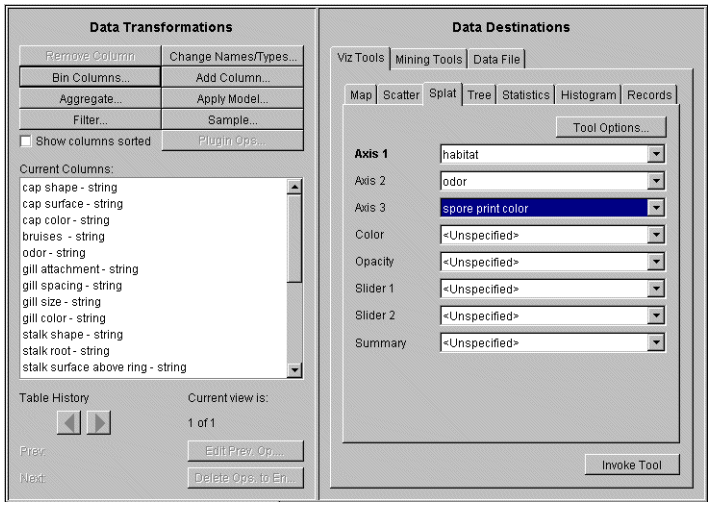


Figure 4-4 Mapping Columns to Visual Elements for the Splat Visualizer

For this example, for Axis 1 choose `habitat`, for Axis 2 `odor`, for Axis 3 `spore print color`.

Table 4-4 identifies the mapping effects of the Splat Visualizer Visual Elements.

Table 4-4 Mapping Visual Elements in the Splat Visualizer

Visual Element	Action:
Axis 1 Axis 2 Axis 3	Assigning data to the first axis alone is usually not useful. Assign data to all three axes to produce a three-dimensional chart.
Color	Assign a string or numeric value to Color. If nothing is mapped to color, the effect is monochromatic.
Opacity	Count of records by default. If you do a count aggregation (or sum of weighting columns) in Tool Manager, or a column in the dataset is already based on counts or weights, use that column for this element.

Table 4-4 (continued) Mapping Visual Elements in the Splat Visualizer

Visual Element	Action:
Slider 1 Slider 2	Maps columns directly to one or two sliders. The columns must be numeric or binned.
Summary	Determines the value of the summary slider's background. If no summary column is mapped, the default is count. If a summary column is mapped, a weighted average for that column is shown in the summary.

In mapping columns, if you need a numeric column, you can turn a two-valued string column into a newly created numeric column using an expression such as: ('stringCol'==”value1”)? 1:0. See “Removing and Adding Columns” in Chapter 3 for more details.

The current example adds an integer column to represent the probability of edibility. This step is included in order to demonstrate how to add a new column, but it is actually not necessary. Color can be mapped to a string value directly.

2. In the Tool Manager Data Transformations pane, click the *Add Column* button. In the resulting dialog box, create `p_edible` as a new column name of type `int`. The expression is `('edibility' == "edible")?100:0`. Notice the disposition of the quotes.

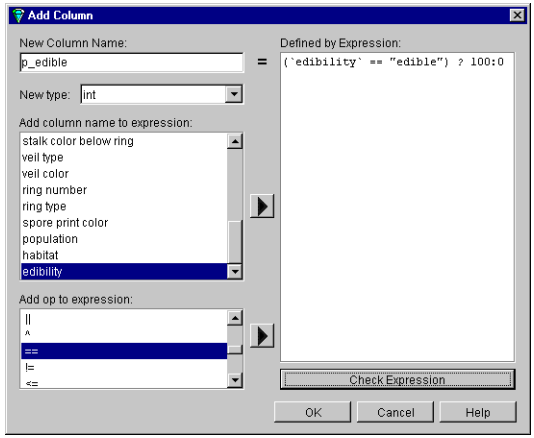


Figure 4-5 Adding an Integer Type Column for Mapping

3. Map `p_edible` to color to define whether a mushroom is edible or not according to the color displayed.

   To undo a mapping, choose `<Unspecified>` from the pulldown menu.

4. Click *Invoke Tool* in the lower right of Tool Manager's Data Destination pane. The resulting visualization provides interesting insights into mushrooms.

Viewing Categorical Data in the Splat Visualizer

The Splat Visualizer also allows you to map a categorical column to the entity color. For example, in the *cars* sample database, the `origin` column contains a string type with the possible values "Japan," "USA," and "Europe." The Splat Visualizer displays this information using categorical splats. If you map the column `origin` to color, each possible value of `origin` is assigned a unique color. The Splat Visualizer then breaks down each individual splat to show the different origins contained in that splat.

Figure 4-6 shows categorical splats for the *cars* database. This visualization was generated by mapping `mpg` to Axis 1, `weightlbs` to Axis 2, `time-to-60` to Axis 3, and `origin` to Color.

When a categorical column is mapped to Color in the Splat Visualizer, the menu and toolbar are slightly different. The normal options under the shape menu are replaced with five different settings for categorical splats. Also, toolbar buttons are provided for the different modes of categorical splats.

The five shape options are as follows:

nominal Hex96	Renders categorical splats using 96 transparent triangles in a hexagonal layout.
nominal Hex384	Renders categorical splats using 384 transparent triangles in a hexagonal layout.
nominal Tri96	Renders categorical splats using 96 opaque triangles in a hexagonal layout.
nominal Tri384	Renders categorical splats using 384 opaque triangles in a hexagonal layout.
nominal pointcloud	Renders categorical splats using a cloud of colored points.

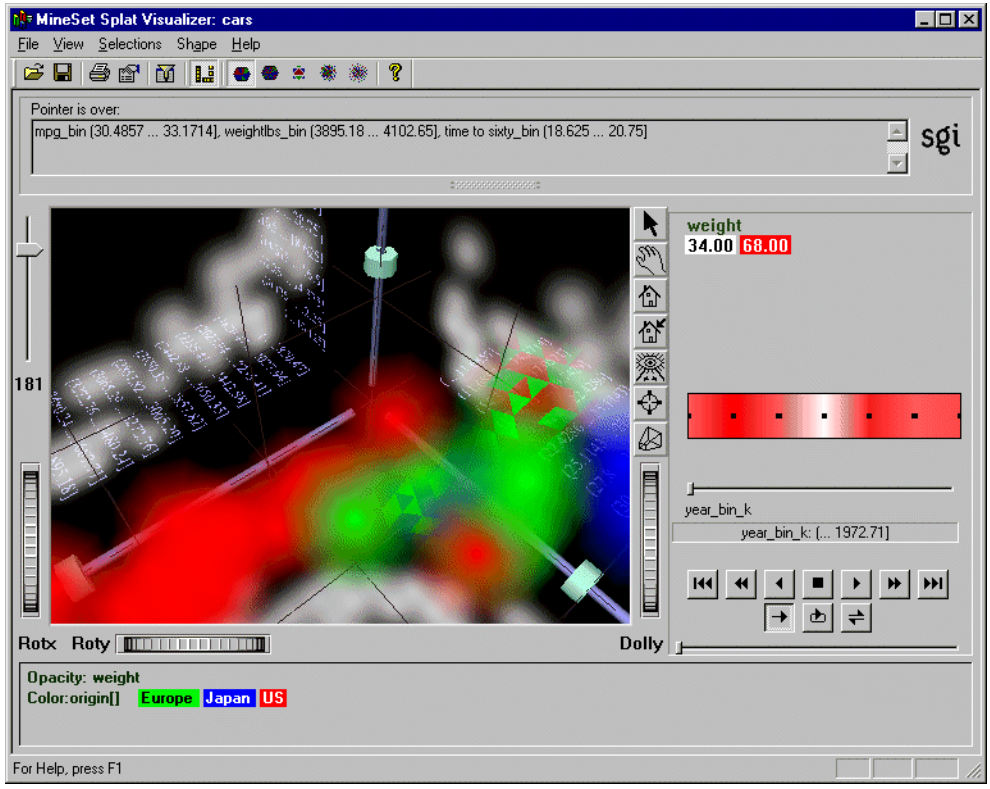


Figure 4-6 Categorical Data in Splat Visualizer

Examining Results in the Scatter and Splat Visualizers

As with any of the visualizers, if you start either the Scatter or Splat Visualizer without specifying a configuration file, (that is, a file with a name ending in *.scatterviz* or *.splatviz*) only the File and Help pulldown menus can be used. For the Visualizer main window to show all menus and controls, open a configuration file. Use File > Open to see a list of configuration files. These files are in the *MineSet\examples* directory, in the location where MineSet was initially installed.

When you open a valid configuration file, the three-dimensional landscape is visible. The *company-total.scatterviz* is shown in Figure 4-1, showing the sales of life insurance, auto insurance, and home insurance with respect to income brackets over time.

Viewing Modes

The two modes of viewing in the Scatter and Splat Visualizers are *grasp* and *select*. To toggle between these modes, move the cursor into the main window, and press the Esc key. You can also change from one mode to the other by clicking on either the arrow or the hand on the border of the main pane of the visualizer window.

Select Mode in the Scatter Visualizer

When in select mode in the Scatter Visualizer:

- To show information about an entity in the Scatter Visualizer upper text field, move the mouse cursor over an object.
- To select an entity, press the left mouse button over an object. To select multiple entities, hold down the Ctrl key while left-clicking the mouse button.
- To clear the selection, click the black background. For full details of navigation see "Navigating in the MineSet 3D Visualizers" on page 27.

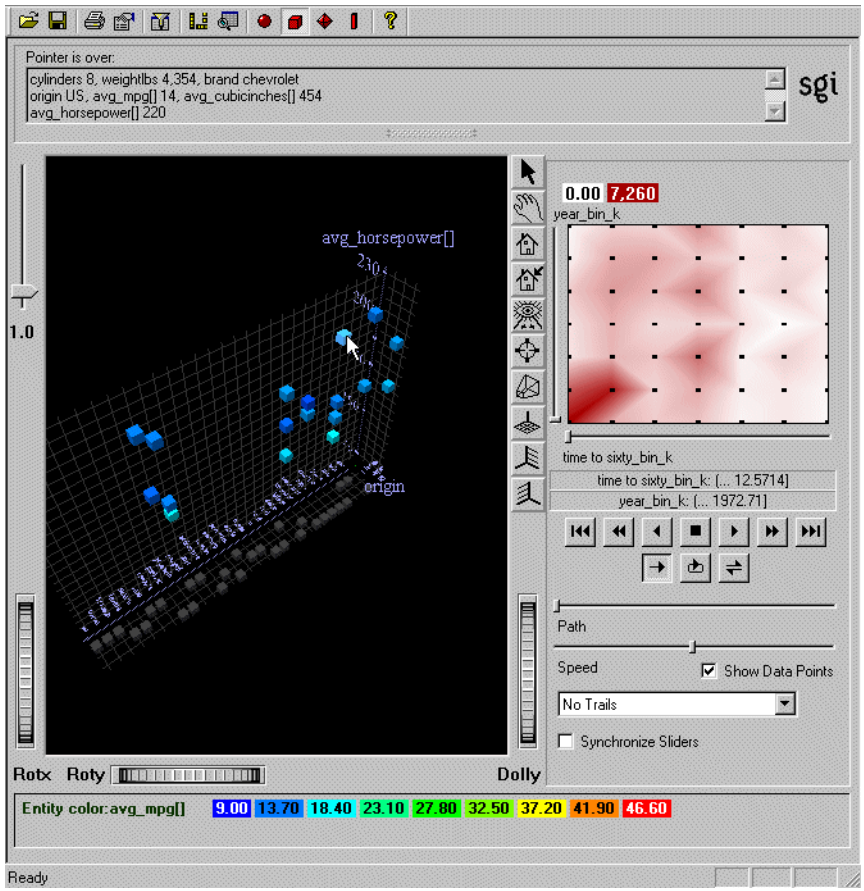


Figure 4-7 Displayed Information about a Selected Entity

For information on specifying the Scatter and Splat Visualizers from the command line, refer to their respective entries in the *MineSet Enterprise Edition Reference Guide*.

Select Mode in the Splat Visualizer

In *select* mode in the Splat Visualizer, you can move a 3-D pick dragger through the dense cloud of points to find information about regions in the scene. This pick dragger is composed of a cylinder and a square.

- To move parallel to the cylinder's axis, click the cylinder and drag the mouse cursor in the desired direction.
- For movement constrained by a selected plane of the square, click the square and drag. Use the Shift key to constrain the motion along one of the two axes within the plane.
- To change between parallel and constrained modes, press the Control key with the cursor over the dragger. (You need not press the mouse button.)
- Alternatively, each axis has a disk that aligns with the pick dragger position. Moving the disk on an axis moves the dragger, and vice versa.

When the pick dragger is over data, the cylinder changes color to that of the splat under it, and information about that region shows at the top of the view area (Figure 4-8). If there is no data, the cylinder remains light gray, and information about its position is displayed at the top of the area to help in navigation.

When you are finished dragging and have released the mouse button, the message for the splat you are currently over is shown in the selection window at the top. This pick information is updated if the animation slider is moved. Using the mouse, you can cut and paste this selection information into other applications, such as reports or databases.

The pick dragger may be removed from the scene by unchecking Selection > Show Pick Dragger.

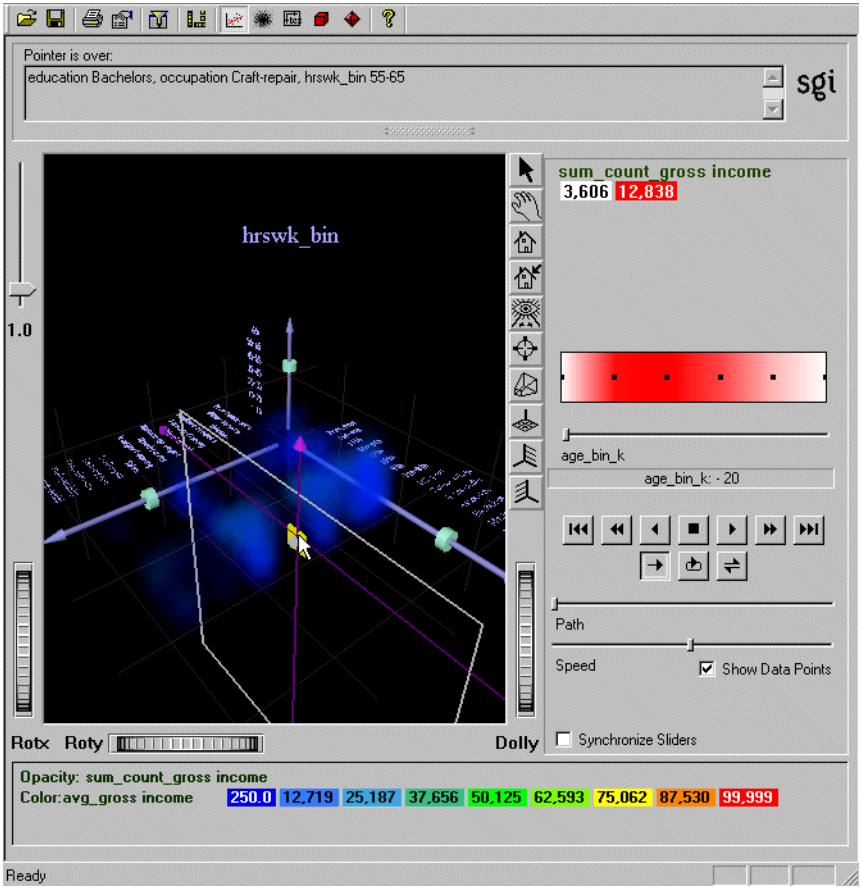


Figure 4-8 Manipulating the Pick Dragger Over Data

An explanation of the window icons can be found in “Navigating in the MineSet 3D Visualizers” on page 27. See the “Scatter Visualizer” and “Splat Visualizer” entries in *MineSet Enterprise Edition Reference Guide* for expanded descriptions of visualizer options, and the “Color Browser” entry for details of the changing colors.

Creating Sliders for the Scatter Visualizer

When you suspect that a value changes according to the value of another column, you can map a column to a slider. A column can be mapped to a slider if that column is numeric (of the type int, float, double) or binned. If the column is already binned, it has `_bin` after the name. The column type is noted after the name of the column in the Current Columns list, for example `total day calls - double`. In most cases, simply mapping a column to a slider in the Tool Manager automatically creates the slider.

Automatic Slider Creation

If the sliders are not specified implicitly during the aggregation step, the Tool Manager creates them for you through automatic binning and aggregation. These automatic operations occur after clicking *Invoke Tool*. Every column that is not deleted or mapped to a visual entity is used to determine the number of unique entities (in other words these will be the Group-by columns in the automatic aggregation). The current slider indices are listed in the Tool Options dialog box when you click *Tool Options* in the Tool Manager.

Every numeric column that is mapped to a visual element will be aggregated. You can choose the type of aggregation in the Tool Options panel. If you want to aggregate differently for different entities, you must use manual slider creation.

Manual Slider Creation

To guarantee the results you expect, it is preferable to explicitly bin and aggregate with Tool Manager yourself. Do this by performing an aggregation in which you aggregate columns (using average, count, or sum) which you plan to map to visual entities, group by other string or binned columns, and finally "array index" by those binned columns which you wish to become animation sliders. The result will have array columns for all the aggregated columns that are indexed by the slider variables. When the sliders are specified this way, they cannot be mapped directly to the slider elements. Refer to the "Aggregate" entry in the *MineSet Enterprise Edition Reference Guide* for details about aggregation.

If there is an array column in the current data table, the Scatter Visualizer will have a slider corresponding to the indices of that column. For one index, there will be a one-dimensional X slider; for two indices, both an X and Y slider are created. All array columns in the current table must have the same indices; otherwise, no slider is created. See "Creating New Columns by Aggregation" in Chapter 3, and the *MineSet Enterprise Edition Reference Guide* for more information on creating arrayed columns.

Creating Animations in the Scatter and Splat Visualizers

Animations are created using the animation control panel to the right of the main visualizer window. The animation window appears only if the sliders have been mapped. For example, *company.scatterviz* offers such a dataset. See the "Animation" entry in the *MineSet Enterprise Edition Reference Guide*. When you suspect a value changes according to a specific criterion, for instance population density changing over the years, you can map that column to a slider. A Splat Visualizer example that uses an animation slider is *adultJobs.splatviz.*

It is most useful to choose attributes which you feel are independent attributes for sliders or axes, and dependent attributes for axes, size and color. If there is only one slider mapped (such as *adultJobs.splatviz* or *adult94.scatterviz*), the summary window is compressed. This slider's dimension is identified by a label below it. If the dataset has no columns mapped to sliders (such as *brand.scatterviz*), no slider control appears.

Animating Using the Visualizer Summary Window

You create an animation in the summary window portion of the animation control panel of either the Scatter or Splat Visualizer. The summary window shows in one or two dimensions an array of summary values based on an aggregation of the data appearing in the main window. There is a summary value for each black dot in the summary slider window. The summary color is assigned to high values, and white is assigned to low values (interpolating between these two colors for intermediate values). If there is no data, or only null values, for the summary variable at a particular position, then it is colored gray to indicate an unknown. The summary value is computed using the specified aggregation of all the data shown in the plot on the left of that slider position. You can turn off these black dots using the "Show data points" check box on the animation panel.

This example shows you how to create an animation, using the prepared *company.scatterviz* file.

1. From the Tool Manager menu bar choose Visual Tools and select 3D Visualizer.
2. From the menu, select File > Open menu.
3. Open the *company.scatterviz* file. This file is under the *MineSet\examples* directory, in the location where MineSet was initially installed.

4. In the summary window, select a black dot to be the animation path starting point. Click and hold the left mouse button on the dot and drag the cursor over the window. End the path by releasing the left mouse button.
   - You can also define a starting point by left-clicking on a dot with the mouse. Then drag one of the independent dimension sliders, drawing a straight line along this dimension. If there are two sliders, you can then use the second slider to draw a straight line along the second axis.
   - Another way is to define a path by left-clicking on a dot, then middle-clicking on another dot to make successive path segments. This option is available only if you have a three-button mouse and your preferences are set to three-button mouse.
5. Use the VCR-like buttons below the summary window to control the animation. Click the play button in the top row before clicking the path button in the second row. For details on the controls, see the "Animation" entry in the *MineSet Enterprise Edition Reference Guide.*

Showing Animation Trails in the Scatter Visualizer

With the Scatter Visualizer you can show motion trails to demonstrate the changing animation path of an entity. When you create an animation, the trail shows behind each selected entity in the form you have selected. The motion option menu, located at the bottom right of the Scatter Visualizer animation control panel (Figure 4-9), allows you to select from:

- No trails—the default
- Line trails—a thin colored line
- Fade-out trails—a transparent colored line similar to the line trail, most opaque at its most recent position
- Tube trails—trails in three-dimensional tubular form, the thickness of which varies with the entity's changing size as it moves through the animation path. Too many tube trails may slow animation noticeably.

All trails are color-coded according to the originating entity. When an entity changes color, say from red to blue, as the slider beneath the summary window moves, the corresponding trail changes color gradually between the two positions. Trails are made between points whose unmapped attributes stay the same over the course of the path. As an example, you can show how income changes over the years, and leave behind a visual trail to show the rate of change between one point and another.

Aggregated data grouped by a small number of columns tends to be an excellent candidate for displaying motion trails. Initially, motion trails are displayed for all points in the scatterplot affected by the path. Selecting any entity by clicking it causes only the selected point to display a trail. This can be used to reduce visual clutter. Entities with null positions appear as breaks in the trails.

Figure 4-9 shows an example of the Scatter Visualizer with tube motion trails.

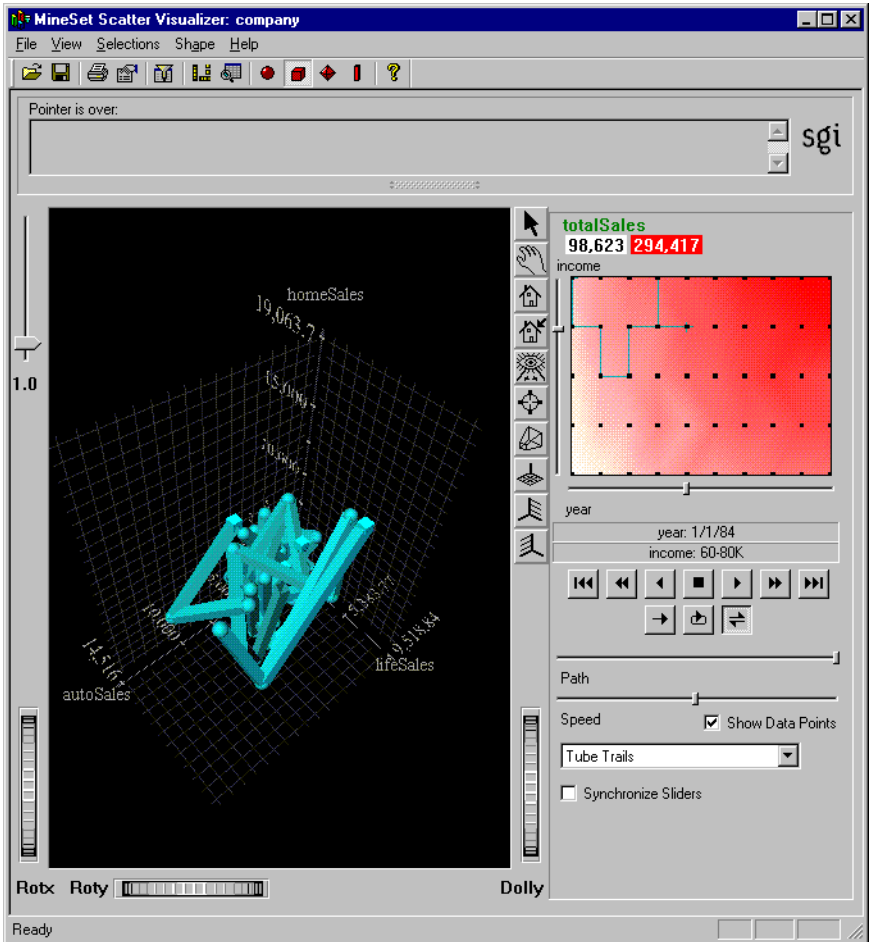


Figure 4-9 Example of Scatter Visualizer Tube Motion Trails

Animating in the Splat Visualizer

Animating in the Splat Visualizer is substantially the same as in the Scatter Visualizer, except that the Splat Visualizer does not have the trails options. An example for *adultJobs.splatviz* is shown in Figure 4-10.

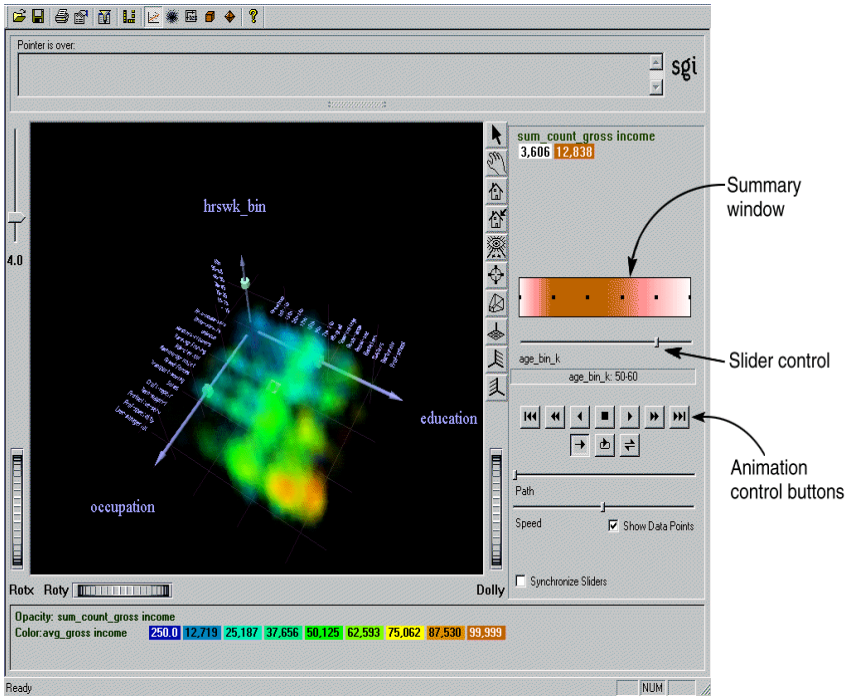


Figure 4-10 Splat Visualizer Animation Control Panel with Summary Window and Slider Control

If the configuration file specifies no slider mappings, then no slider control appears.

Interpreting the Summary Window in the Splat Visualizer

The summary window (see Figure 4-10) shows the aggregation of the summary attributes in the slider dimension. Color density indicates high summary values represented by the splats in the main window. If no column is explicitly mapped to summary, count is used to show which positions on the slider represent the most data.

In the example shown in Figure 4-10, the black dot in the pure white region of the summary slider indicates that the splat plot on the left is composed of 3,606 records when the slider is at that position. The plot has 12,838 records for the reddest point on the slider.

The evenly spaced black dots indicate the precise positions of the discrete data points. You can turn off these black dots by unchecking the “Show data points” box at the bottom of the animation panel. Slider positions between these positions use interpolation of the underlying data to produce an image.

Color Density in the Summary Window

You can interpret how color density reflects values by using the *adultJobs.splatviz* file. The summary window shows a color range from white (on the left) to red (in the middle) to white (on the right). Red represents more records, while white represents fewer records. In this example, the greater density of red in the middle of the slider means the highest concentration of people are in the 20-50 age range.

Creating a Path in the Summary Window

You create a path in exactly the same manner as you do in the Scatter Visualizer, and the controls are the same. It is worth noting, however, that as animation proceeds, the size and color of the splats change smoothly. The information displayed in the status field shows the interpolated data values. When the slider motion stops, the slider position continues to the nearest discrete data position where interpolated data values are not used.

There is a table for each binned position on the summary slider. Each row in one of these tables (which is an aggregate of original data) defines a splat in the scene. Tables corresponding to adjacent bins on the summary need not have the same number of rows because of the differences in data distribution from one position to the next. For example, if the visualization in Figure 4-2 on page 57 changes from showing 40-50 year-olds to one showing 50-60 year-olds by moving the slider one notch to the right (see Figure 4-11), some positions might show splats where there were none before, and vice versa.

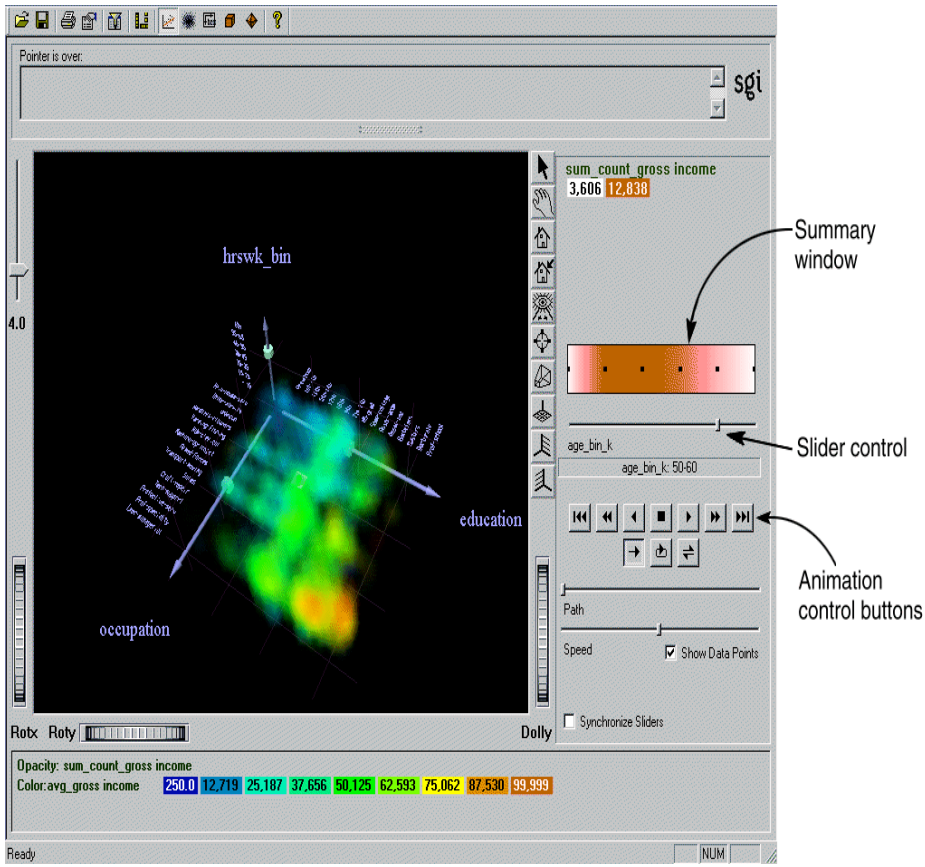


Figure 4-11 Changed Visualization as a Result of Moving the Slider (Compare to Figure 4-2)

For interpolation on a one-dimensional slider, two adjacent tables are merged, then aggregated using the spatial columns as unique keys. The weight is simply interpolated (0 weight is assumed if one of the tables lacks a particular row). The average value used for color is also interpolated, but weighted by the weight.

For an example of the technical details of interpolation, see the “Splat Visualizer” entry in the *MineSet Enterprise Edition Reference Guide*.

Manipulating Scatter and Splat Visualizer Results

Five pulldown menus let you access additional Scatter and Splat Visualizer functions. These are labeled File, View, Selections, Shape, and Help. If you start the visualizers without specifying a configuration file, only the File and the Help menus are available. Once you have opened a file, other menus become available. The File and Help menus are standard and are detailed in the *MineSet Enterprise Edition Reference Guide*.

Changing the Displays

The View menu lets you control certain aspects of what is shown in the visualizer window. Figure 4-12 shows the Scatter and Splat Visualizer View menu.

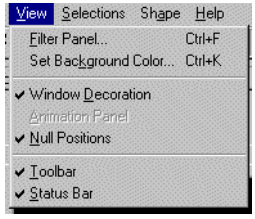


Figure 4-12 Scatter and Splat Visualizer View Menu

The following table explains the choices:

Table 4-5 View Menu Choices for Scatter and Splat Visualizers

Option	Description
Filter Panel	Brings up Filter Panel (Figure 4-13) to manipulate the content of the display.
Set Background Color	Brings up a color chooser to specify new display background color.
Window Decoration	Shows or hides external controls around the main window.
Animation Panel	Shows or hides any animation control panel.
Null positions	Shows or hides entities with null or unknown positions along the axes.

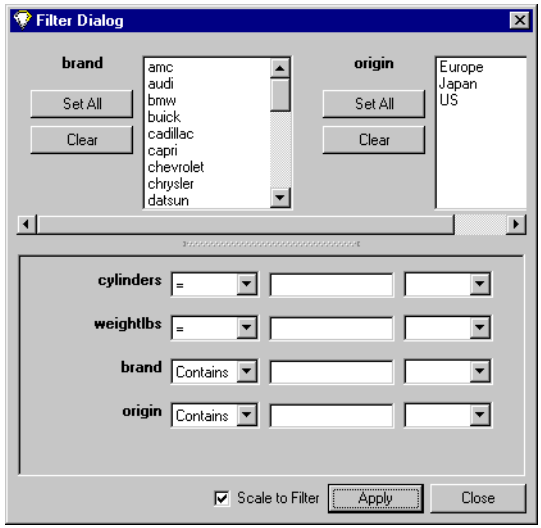


Figure 4-13 Scatter and Spat Visualizer Filter Panel

If you want to filter the amount of data shown, based on certain criteria, use the Filter Dialog shown in Figure 4-13. From the View menu select Filter Panel. The upper pane lets you filter based on string columns. To select a value, click it. To deselect a value, simply click it again. To select all values of a column, click the *Set All* button. To clear all the current selections, click the *Clear* button.

The bottom pane lets you filter based on the values of both string and numeric columns. See the "Filter Panel" entry in *MineSet Enterprise Edition Reference Guide* for detailed information.

With the "Scale to Filter" check box in the lower right, you can specify whether the landscape in the main window covers the domain of the entire dataset or just the filtered data.

Click the *Apply* button to start filtering. If you press the Enter key while the panel is active, filtering starts automatically.

Selecting and Drilling Through in the Scatter Visualizer

The Scatter Visualizer Selections menu lets you drill through to the underlying data.

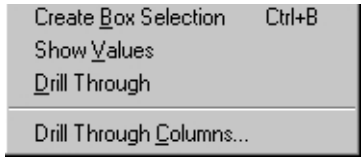


Figure 4-14 Scatter Visualizer Selection Menu

- *Create Box Selection* creates a three-dimensional box selector that can be stretched and moved to select regions of the volume. While this is active, a Record Viewer table opens showing information on the aggregated data about the entities within it. Make sure the cursor is in Select mode. To move the selection box, click on one of the faces with the left mouse button, and drag it in the desired direction. Holding the Shift key while dragging constrains the motion to the axis to which the drag motion is closest. To change the extent of the selection box, drag one of the gray scale tabs in the desired direction. You cannot resize or move beyond the bounds of the volume. The gray scale tabs constantly resize to maintain constant screen size. If at any time they appear too big, you can zoom in closer, and they reduce their size relative to the box.
- *Show Values* displays the values of selected entities in a table.
- *Drill Through* opens the Drill Through Dialog box (Figure 4-15). This dialog box gives you the following choices for viewing the selected records:
  - Show original data in recordviewer.
  - Send to ToolManager as Filter.
  - Send to ToolManager as new column.
  - Send to ToolManager as SQL.

  You can also choose one of the following:
  - Send the records to a new copy of ToolManager.
  - Send all non-selected records to the ToolManager (Complement drill through).

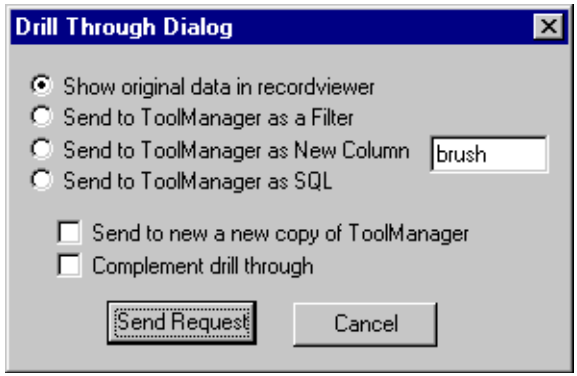


Figure 4-15 Scatter Visualizer Drill Through Dialog

- *Drill Through Columns* brings up a panel that lets you select which columns are used in drill through. Unlike other visual tools, there are no specific columns in the data that are designated as the key to the data. It is impossible for the Scatter Visualizer to determine which columns the user desires in the drill-through expression. For example, you might have cars data with brand, model, and weight. Perhaps you want to drill through to the original data, and specify that brand and model should be considered, but weight should not. By default, all columns that have been mapped to graphical requirements are considered significant on drill-through. The others are not, but may be made so by highlighting them in the Preferences dialog box.

For further details see the "Drill Through" entry *MineSet Enterprise Edition Reference Guide*.

Selecting and Drilling Through in the Splat Visualizer

The Splat Visualizer Selections menu differs slightly from the Scatter Visualizer in the options available to drill through to the underlying data.

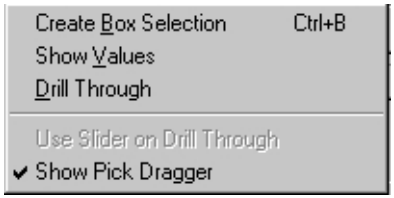


Figure 4-16 Splat Visualizer Selection Menu

- *Create Box Selection* creates a 3-D box selector that can be stretched and moved to select regions of the volume. While the box selector is active, a table is open and showing information about all of the aggregated data that is represented by the entities within it (Figure 4-17). Closing this window clears all current selections. Any entities within the selection box or selected using shift-click are shown in the table window.

  To move the selection box, click on one of the faces with the left mouse button, and drag it in the desired direction. Holding the Shift key while dragging constrains the motion to the axis to which the drag motion is closest. To change the extent of the selection box, drag one of the gray scale tabs in the desired direction. Trying to resize or translate beyond the bounds of the volume is not permitted. The gray scale tabs constantly resize to maintain constant screen size. If at any time they appear too big, you can zoom in closer, and they reduce their size relative to the box.

  Within the box selection, you will also see a 3D rectangle or rectangles resembling wire-frame boxes (Figure 4-17). These boxes show where the actual data points are.

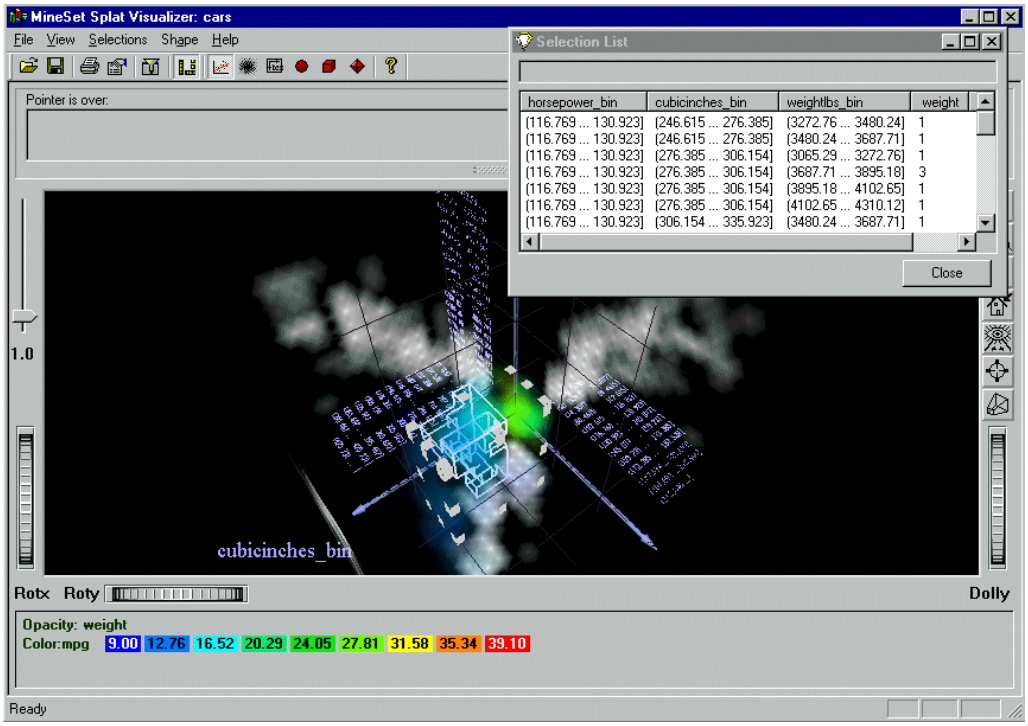


Figure 4-17 Wire-frame Boxes in Splat Visualization

- *Show Values* retrieves and displays the records corresponding to the data points which you have selected in the Splat visualization. The resulting records are shown in a table.
- *Drill Through* opens the Drill Through Dialog box (Figure 4-18). This dialog box gives you the following choices for viewing the selected records:
  - Show original data in recordviewer.
  - Send to ToolManager as Filter.
  - Send to ToolManager as new column.
  - Send to ToolManager as SQL.

  You can also choose one of the following:
  - Send the records to a new copy of ToolManager.
  - Send all non-selected records to the ToolManager (Complement drill through).

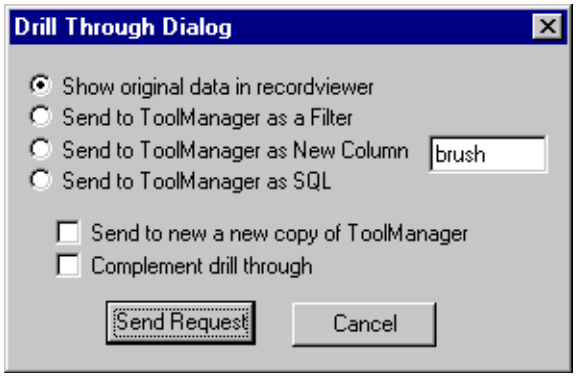


Figure 4-18 Splat Visualizer Drill Through Dialog

- *Use Slider on Drill Through* uses the slider position when drilling through.
- *Show Pick Dragger* toggles the visibility of the pick dragger (on by default). The pick dragger is removed when a box selection is started, but it can be made active at the same time that a box selection is active.

For further details see the "Drill Through" entry in the *MineSet Enterprise Edition Reference Guide*.

Changing the Scatter Visualizer Display with the Shape Menu

The Shape menu lets you change the method for drawing the scatter points in the Scatter Visualizer. The following opaque primitives can be used to represent scatter points.

- *Cube* draws an opaque cube, whose volume is proportional to the attribute mapped to the size visual element.
- *Diamond* draws a wire frame triangle. Its orientation varies with its color, and its size is proportionate to the value of the attribute mapped to the size visual element.
- *Sphere* draws an opaque sphere, whose volume is proportional to the value of the attribute mapped to the size visual element.
- *Bar* draws elongated rods, the height of which is determined by the value of the attribute mapped to the size visual element.

Changing the Splat Visualizer Display with the Shape Menu

The Splat Visualizer Shape menu lets you change the method for drawing the splats. You can choose to exchange accuracy for interactivity. Texture splats are the most accurate representation of ideal Gaussian density distribution that is approximated in every approach. Because most computers support hardware-assisted texturing well, the texture splat is usually the best choice. The three splat types are:

- *Linear* draws a small set of triangles to give a linear approximation to a Gaussian splat.
- *Gaussian* draws a large set of triangles to approximate a Gaussian splat.
- *Texture* uses a texture-mapped rectangle to give the most accurate representation. This can be very slow on systems that do not support hardware-assisted texture mapping.

Alternatively, you can choose one of the following opaque primitives to represent the splats:

- *Cube* draws an opaque cube whose volume is proportional to the attribute mapped to the opacity visual element, if any, or to count.
- *Diamond* draws a wire frame triangle. Its orientation varies with its color, and its size is proportional to the attribute mapped to the opacity visual element, if any, or to count.
- *Sphere* draws an opaque sphere whose volume is proportional to the attribute mapped to the opacity visual element, if any, or to count.

Examining Data with the Tree Visualizer

This chapter contains the following sections:

Sample configuration and data files have been provided to demonstrate the Tree Visualizer's features and capabilities. These files are in the *examples* directory under *MineSet*, in the location where MineSet was initially installed.

Overview of the Tree Visualizer

The Tree Visualizer is a graphical interface that displays data as a three-dimensional landscape. It presents your data hierarchically in the form of a tree. Each level of the tree branches on the values of a different attribute. Each node in the tree shows a chart representing all the data in the subtree below it. The chart is composed of a base block with height and color depending on the data attributes. On each base are bars and/or disks whose number, label, height, and color are also determined by the attributes you have specified.

As shown in Figure 5-1, the Tree Visualizer displays quantitative and relational characteristics of your data by showing them as hierarchically connected nodes. Each node contains bars and disks whose height and color correspond to aggregations of data values (usually sums, averages, or counts). The lines connecting nodes, called edges, show the relationship of one set of data to its subsets.

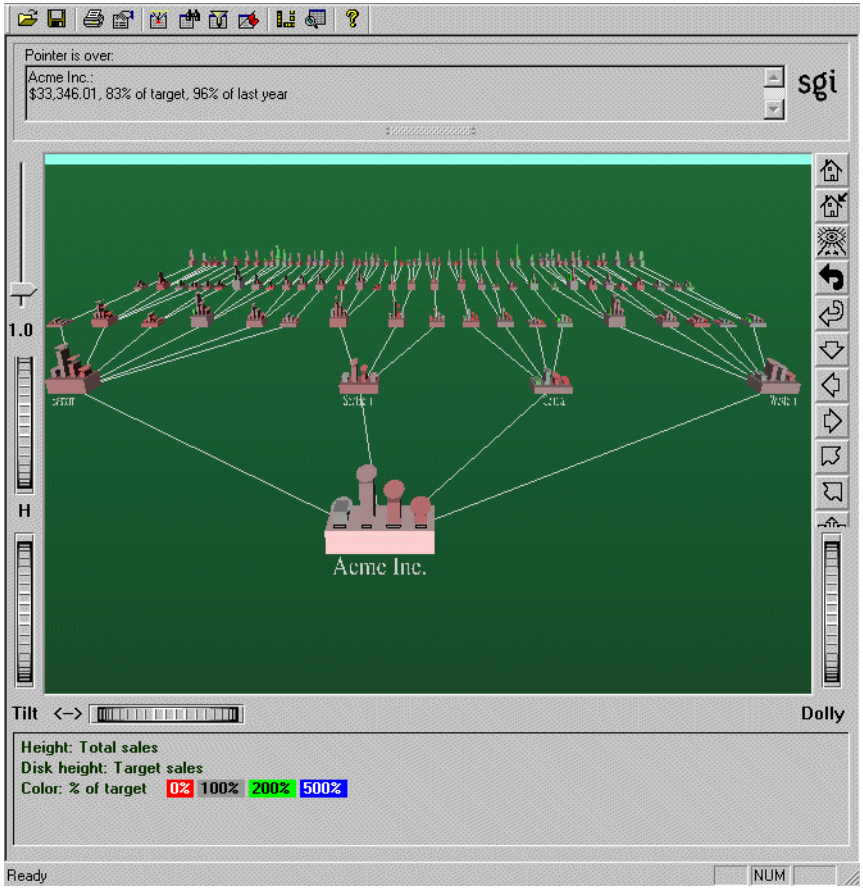


Figure 5-1 Example Display in the Tree Visualizer's Main Window

Starting the Tree Visualizer

This section describes how the Tree Visualizer can be configured using the Tool Manager. Although the Tool Manager greatly simplifies the task of configuring the Tree Visualizer, you can construct a configuration file manually using an editor. There are also several other ways to start the Tree Visualizer, see the *MineSet Enterprise Edition Reference Guide* for details.

To start the Tree Visualizer from the Tool Manager:

1. Choose File > Connect to Server and log into a server.
2. Choose File > Open New Data File and select or type in the desired filename.
3. In the Data Destination panel, click the Viz Tools tab (Figure 5-2).
4. From the tools tabs, choose the Tree Visualizer tab.
5. Map specific attributes from your dataset to the appropriate Visual Elements in the Data Destinations Pane using the pulldown menus beside the Visual Elements. See "Mapping the Tree Visualizer Visual Elements" on page 90 for more information about mapping.
6. Click the Tool Options button and specify Height and Color Aggregations in the Tree Visualizer Configuration Options panel. This step can be crucial for getting a meaningful visualization.

   By default, the height (or color) of the bars of the parent node represents the sum of the values of all the bars of the children; however, for some datasets, average, max, min, or count may be better choices. Aggregations can be specified for the heights and colors of bars, bases, and disks. See the *MineSet Enterprise Edition Reference Guide* for more details about aggregation and other, more advanced options.
7. Click *Invoke Tool* to start the visualizer.

The Status box at the bottom of the Tool Manager window shows the progress and resulting statistics.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Mapping the Tree Visualizer Visual Elements

When the Tree Visualizer is selected, the Data Destinations panel shows visual elements to which you can map the attributes from your data (Figure 5-2). Select columns from the pulldown menu next to each visual element. Table 5-1 lists the visual elements along with comments on their use. Often the best way to use the Tree Visualizer is to experiment with the element mapping until you come up with the best combination of attributes and visual elements.

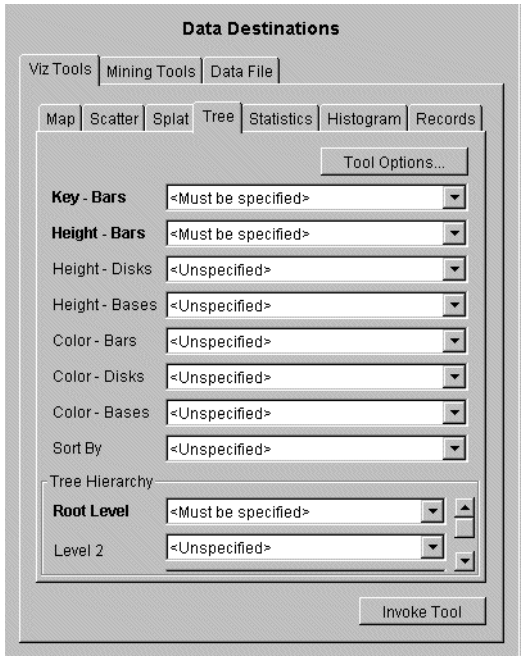


Figure 5-2 Data Destination Panel of Tool Manager with Tree Visualizer Selected

Table 5-1 lists the visual elements.

Table 5-1 Tree Visualizer Visual Elements

Visual Element	Comments
Key - Bars	Use this to specify what the bars represent. If the column is numeric, it is often best to bin the column. Otherwise, MineSet creates a bar for every value of the attribute, which can make the visualization difficult to work with.
Height - Bars	Use this to specify what the bar heights represent. Typically, the higher the bar, the greater the value represented. See the tool options panel for options on aggregating the attribute at each level.
Height - Disks	Use this to specify what the heights of the disks represent. The optional disks are placed at the same locations as the bars. Disks are most useful if the disk attribute is in the same unit of measurement as bar height attribute (for instance, this year's sales and last year's sales). If no mapping is specified, no disks are displayed.
Height - Bases	Use this to specify what the base heights represent. If no mapping is specified, the bar height mapping is used. See the tool options panel for options on aggregating the attribute at each level.
Color - Bars	Use this to specify what the bar colors represent. You can assign the specific colors via the Tool Options panel (See the "Color Selection" entry in the *MineSet Enterprise Edition Reference Guide*, or allow MineSet to automatically assign colors. See the tool options panel for options on aggregating the attribute at each level.
Color - Disks	Use this to specify what the disk colors represent. You can assign the specific colors via the Tool Options panel (See the "Color Selection" entry in the *MineSet Enterprise Edition Reference Guide*), or allow MineSet to automatically assign colors. This option has an effect only if the disk height is specified.

Table 5-1 (continued) Tree Visualizer Visual Elements

Visual Element	Comments
Color - Bases	Use this to specify what the base colors represent. You can assign the specific colors via the Tool Options panel (See the "Color Selection" section in the *MineSet Enterprise Edition Reference Guide*), or allow MineSet to automatically assign colors. See the tool options panel for options on aggregating the attribute at each level. If no mapping is specified, the bar color mapping is used.
Sort By	Use this to sort the layout of the nodes by the values of the selected attribute. The sort order defaults to ascending from left to right.
Tree Hierarchy Root Level	Use this to specify how the table from your data source is converted into a hierarchy. The second level of the tree will have as many nodes as there are values for the selected attribute. If the attribute has many possible values, the tree will be large and difficult to analyze. Binning the column can alleviate this.
Tree Hierarchy Level 2, 3, ...	Use these to split your data into more hierarchies. The Visual Elements list defaults to three hierarchical levels. If you specify a third hierarchy level, the Tree Visualizer automatically adds a fourth. With every extra level you specify, another one is added. You can specify as many hierarchy levels as necessary. If the attribute has many possible values, the tree will be large and difficult to analyze. Binning the column can alleviate this.

Examining Your Results with the Tree Visualizer

When the Tree Visualizer first launches, the root node of the hierarchy is at the front of the scene, near the bottom of the main window. In back of the root node are its descendents. Each node consists of a base with bars on it. You can change what the heights and colors of the bars represent via the Tool Manager or by manually changing the *.treeviz* configuration file (see the "Tree Visualizer" entry in the *MineSet Enterprise Edition Interface Guide*); usually, the base represents the aggregate of all the bars. Bases are connected with edges representing the connection of the nodes to their descendents.

Values in subgroups can be summed and displayed automatically in the next higher level. The base under the bars can provide information about the aggregate value of all the bars. Bars representing negative values are shown below the top of the base. You can

see negative value bars more clearly by disabling the base height (see “Fine-Tuning the Tree Visualizer Display” on page 102, or the “Tree Visualizer” entry in the *MineSet Enterprise Edition Reference Guide*).

Tree Visualizer Example

Figure 5-3 shows the opening scene for the stores.treeviz file. This file is in the *MineSet 3.1\examples* directory, in the location where MineSet was initially installed. To see it from the Tool Manager Visual Tools menu, open the 3D Visualizer; then, from the Tree Visualizer File menu, choose Open. Each record in the data file contains the following columns: region (eastern, southern, central, or western), state, city, store number, product type (appliances, clothing, electronics, or furniture), this year’s sales, last year’s sales, and target sales. Two new columns have also been added: the percentage of the target sales attained this year, and the percentage of last year’s sales attained this year. Table 5-2 shows the mapping of columns to the Tree Visualizer’s visual elements.

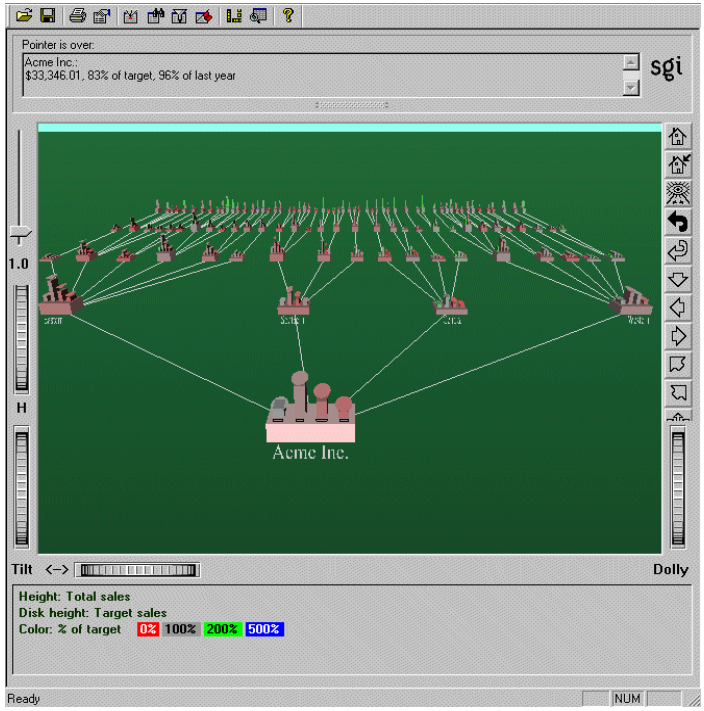


Figure 5-3 Tree Visualizer’s Initial View of Store Dataset

The bars in the top node of the tree represent total sales for each product. The disks represent the target sales. The next level down splits the sales by region, the next by state, and so forth, down to the individual stores.

In examining the Tree Visualizer scene, you can see at a glance how well each region, state, city, or store met the target sales for each product type. You can also compare sales across regions, states, product types, and so forth.

Table 5-2 shows the mapping of columns to visual elements for this example.

Table 5-2 Element Mapping for Stores Data

Column	Aggregation	Mapping
Region	None	Hierarchy level 1
State	None	Hierarchy level 2
City	None	Hierarchy level 3
Store ID	None	Hierarchy level 4
Product type	None	Key - bars
This year's sales	Sum	Bar height
Last year's sales	Sum	None
Target sales	Sum	Disk height
Percent of target	None	Bar color
Percent of last year's sales	None	None

Mapping is the key to obtaining a meaningful visualization. Until you are familiar with the tool, experimentation is the best way to find the best mapping scheme.

Getting a Closer Look at Your Data in the Tree Visualizer

There are two ways to see the details at each node:

- Highlighting a node or object (a bar or a base) lets you view the underlying data. (See "Highlighting an Object or Node.")
- Selecting a node or object zooms the viewpoint to that location, and lets you view underlying data. (See "Selecting an Object.") You can select more than one location at a time. Selecting a node or object also spotlights it. Spotlighting lets you keep track of that location, even if it is far in the distance of the scene.

Highlighting an Object or Node

To highlight an object (a bar or a base), place the mouse over that object. Information about that object appears over the top left of the view area, under the "Pointer is over" label (Figure 5-4). To highlight a node and obtain information about that node, place the pointer over a line leading to that node.

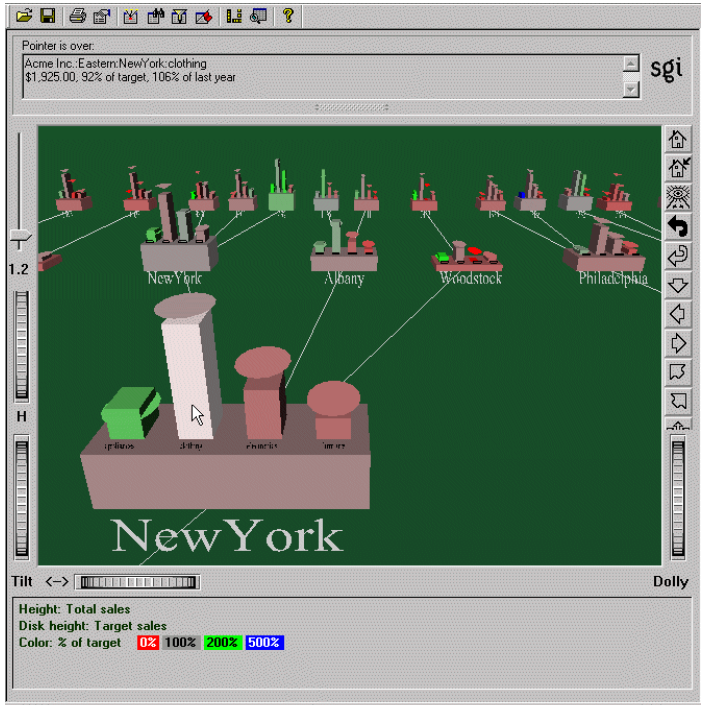


Figure 5-4 Highlighted Object and Its Underlying Information

Selecting an Object

To select an object and zoom to it, click that object. Press the Shift key while clicking to select the object without zooming to it. As long as the object is selected, the information is displayed.

If you press the Ctrl key while clicking an object, the selection of that object is toggled. If the object is currently not selected, clicking selects it, and vice versa. Using this technique, it is possible to select multiple objects simultaneously. While the information in the "Pointer is over" field shows only the information on the last object selected, it is possible to see the values for all selections by choosing Selections > Show Values or by drilling through to the original data behind the selections.

When you select an object, a white spotlight appears on it (Figure 5-5). A yellow spotlight appears when you are searching (see "Finding Specific Objects with the Search Panel" on page 98). Spotlights are visible even if the selected object is a descendent node in the far background.

The edges of spotlights are surrogates for an object: when you move the pointer over the edge of a spotlight, the associated object is highlighted, and information about that object appears above the top left corner of the view. Click the edge of a spotlight to select the associated object and (if the Shift key is not held down) to zoom to it. The spotlight is active only on the solid lines along the edges, not the translucent section in the center. Clicking in the translucent area selects objects behind the spotlight.

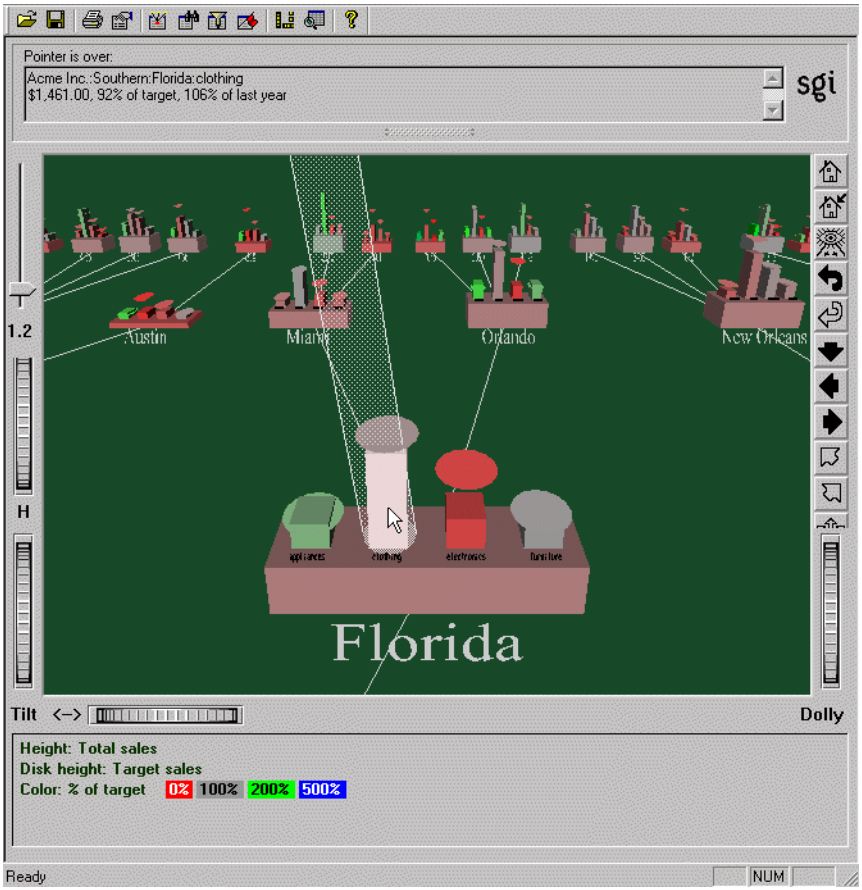


Figure 5-5 Example of a Selected (Spotlighted) Object

Seeing the Whole Picture with Overview

To open a window with an overhead view of the complete hierarchy (Figure 5-6), choose View> Overview. If you want the Overview to display each time you view the scene, set the Overview option in the configuration file (see the "Overview" section in the *MineSet Enterprise Edition Interface Guide*).

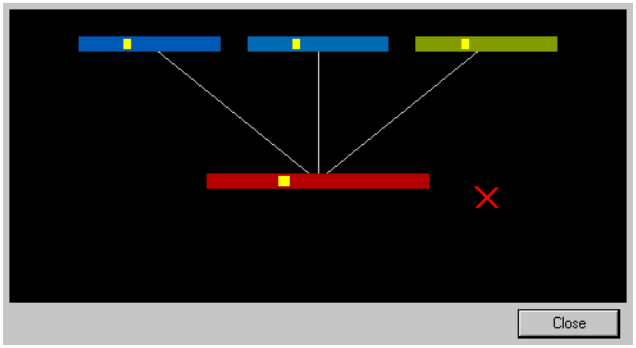


Figure 5-6 Tree Visualizer's Overview Window

The "X" in the Overview window shows your current location. The Overview helps you keep track of your location and viewpoint in the entire scene. It can also help you quickly go to a specific location:

- To select a node in the Overview and have the main view zoom to it and spotlight it, click that node. This is similar to clicking the node in the main view.
- To zoom your viewpoint to a new location, even if there is no node at that location, click the right and left mouse buttons (or the middle mouse button) at the location.

Finding Specific Objects with the Search Panel

To search for specific objects in the Tree Visualizer window, choose View > Search Panel. Fill in the search criteria and click Search. For more information about the Search panel, see the *MineSet Enterprise Edition Reference Guide*.

Once the search is complete, yellow spotlights highlight objects matching the search criteria (see Figure 5-7). To display information about an object under a yellow spotlight, move the pointer over that spotlight; the information appears in the upper left corner, under the label "Pointer is over." To select and zoom to an object under a yellow spotlight, click the spotlight; if you press the Shift key while clicking, zooming does not occur.

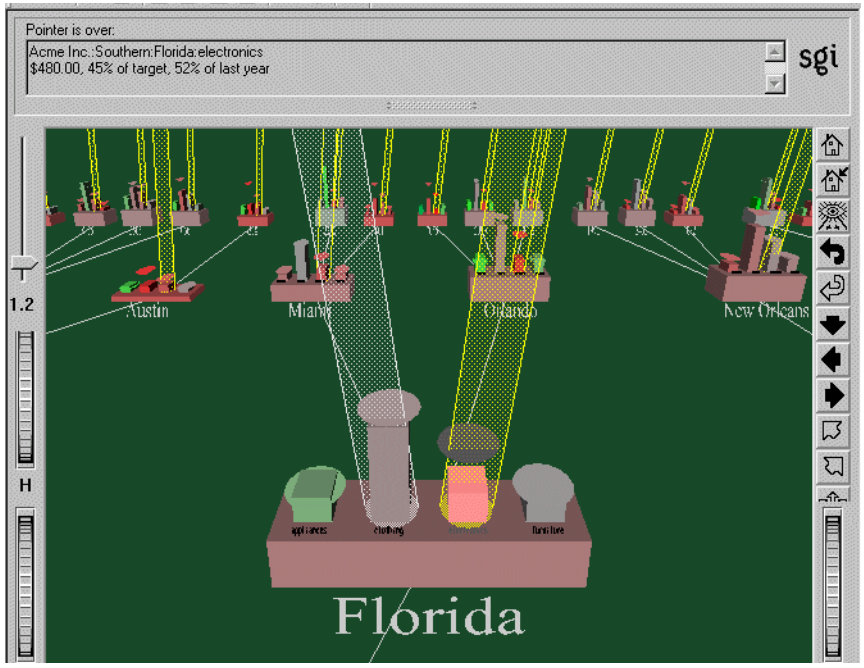


Figure 5-7 Sample Results of a Search in the Tree Visualizer

Flagging Important Locations with the Marks Panel

The Marks panel lets you name and store important locations (or viewpoints) for later reference. All marks can be indicated by colored flags in the main view. If the mark represents a selected object, the flag is placed on that object. If it represents a viewpoint position, the flag is placed at that position. To go to the mark, click the flag. All flags can be turned on and off by choosing Mark Flags from the Display menu (see *Mark Flags* in "Fine-Tuning the Tree Visualizer Display" on page 102).

To save a location and place a flag to mark it (optional):

1. From the Tree Visualizer's View Menu, choose the Marks panel. This opens the window shown in Figure 5-8. If you have not yet placed any marks, the list will be empty.

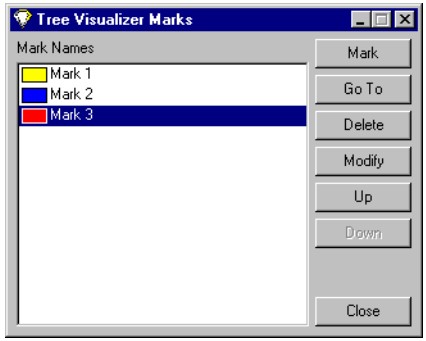


Figure 5-8 Tree Visualizer Marks Panel

2. Click the *Mark* button to mark the current location in the main window. This displays the dialog box shown in Figure 5-9.
3. Enter the name you wish to use for the mark and choose a color. The default name is that of the currently selected object. The color you choose controls the color of the flag appearing in the main window and represents the mark.

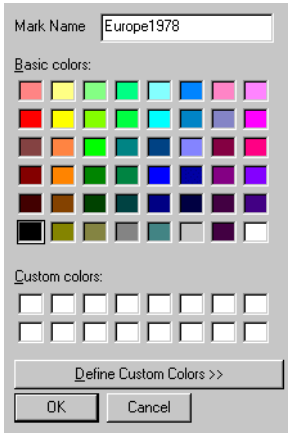


Figure 5-9 Choose Mark Dialog Box

Figure 5-10 shows a sample main window with flags representing the created marks.

Figure 5-10 Main Window with Flags Representing Marks

The buttons along the side of the Marks window allow you to perform the following tasks:

- To go to the location represented with the selected mark in the panel, click the *Go to* button in the Marks panel. Double-clicking a mark in the panel has the same effect. If the object selected by that mark no longer exists (because it was filtered out, or the data was changed since the mark was created), the location shown is close to where the object would have been.
- To delete the selected mark in the panel, click the *Delete* button.

- To change the name or color of the selected mark in the panel, click the *Modify* button.
- To move the selected mark in the panel up the listing order, click the *Up* button.
- To move the selected mark in the panel down the listing order, click the *Down* button.
- To exit the Marks panel, click the *Close* button.

The file storing the marks information has the same name as the configuration file, with a *.marks* suffix appended. Whenever a mark is changed, it is saved to that file. If all marks are deleted, the *.marks* file is removed. If mark changes cannot be saved (because of a permission error, for instance), a warning appears; this warning is not repeated when subsequent mark changes are attempted.

Filtering the Data with the Filter Panel

The Filter panel removes selected information, thus fine-tuning the displayed hierarchy. You can use the Filter panel to emphasize specific information, or to shrink the amount of data for better performance. To access the Filter panel, choose it from the Tree Visualizer View menu. See the "Tree Visualizer" entry in the *MineSet Enterprise Edition Reference Guide* for more information about using the Filter panel.

Fine-Tuning the Tree Visualizer Display

Table 5-3 describes several options for fine-tuning the Tree Visualizer display. These options from the Tree Visualizer Display menu let you control some of the display parameters.

Table 5-3 Tree Visualizer Display Parameters

Option	Description
Base Heights (toggle)	Turns the heights of the bases on and off. To see negative numbers, or to make it easier to compare the bar heights, turn this option off. Turning it on provides summary information about all the bars. The initial value of this toggle can be changed with the "base height" statement in the configuration file.
Marks Flags (toggle)	Turns the flags representing marks on or off (see "Flagging Important Locations with the Marks Panel").

Table 5-3 (continued) Tree Visualizer Display Parameters

Option	Description
Zeros (submenu)	Controls how objects with zero height are displayed. By default, they are shown like other objects: a solid cube of height zero (a plane). The submenu lets you specify that they be displayed as outlines (appearing as a hollow square), or be hidden completely (not drawn). The initial value of this can be changed using the "zero" option in the configuration file (see the *MineSet Enterprise Edition Reference Guide*).
Nulls (submenu)	Controls how objects of null height are displayed. It has the same options as the zero menu; however, the default for null options is to display the objects as an outline. The initial value can be changed using the "null" option in the configuration file (see the *MineSet Enterprise Edition Reference Guide*).

Examining Data with the Map Visualizer

The Map Visualizer shows data displayed over a three-dimensional landscape of bar chart shapes or recognizable geographic shapes. It is useful when your data has a geographical context or some specific topographical layout. This section covers the following topics:

Sample configuration and data files have been provided to demonstrate the Map Visualizer's features and capabilities. These files are in the *MineSet\examples* directory, in the location where MineSet was initially installed.

Overview of the Map Visualizer

The Map Visualizer helps you look at spatially related data. Besides dynamically navigating through this geographically based landscape, you can drill up and down to get an overview or to see increased granularity, as well as use animation to see how the data changes across one or two independent dimensions. An independent dimension is any attribute such as age or year, that can vary independently of another column. The animation panel to the right of the main window appears only when the dataset contains independent dimensions mapped to sliders (as shown in Figure 6-1.)

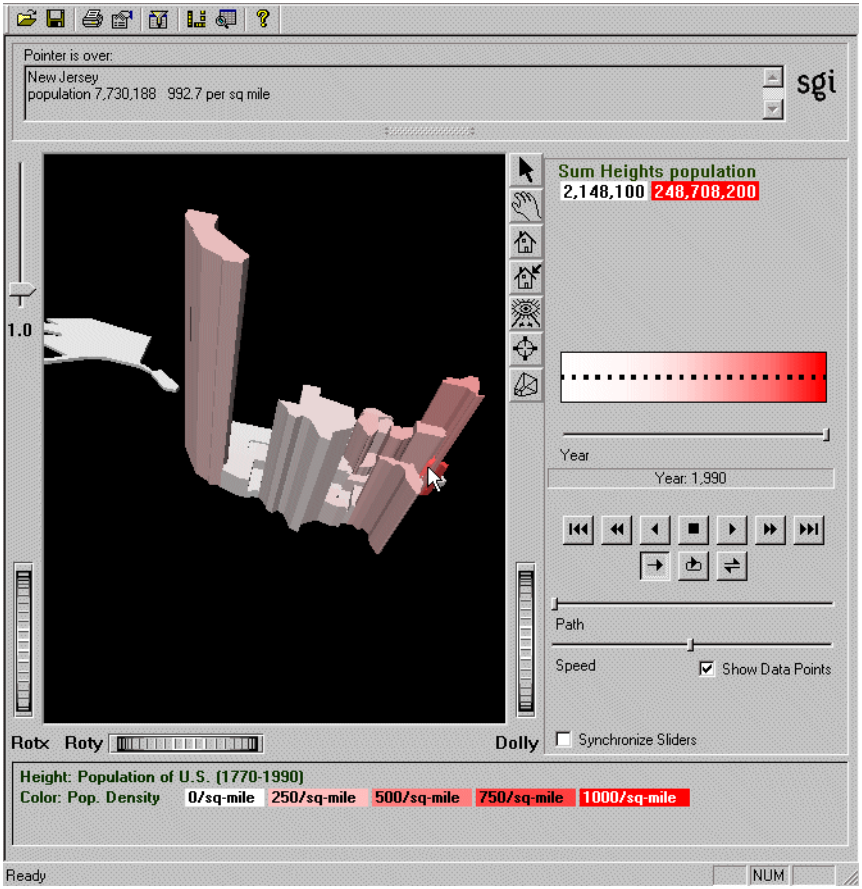


Figure 6-1 Sample Map Visualizer with Geographic Shapes

The landscape can also consist of a flat plane of geographical objects drawn as simple outlines, with “bar chart” cylinders placed at specific locations (see Figure 6-2).

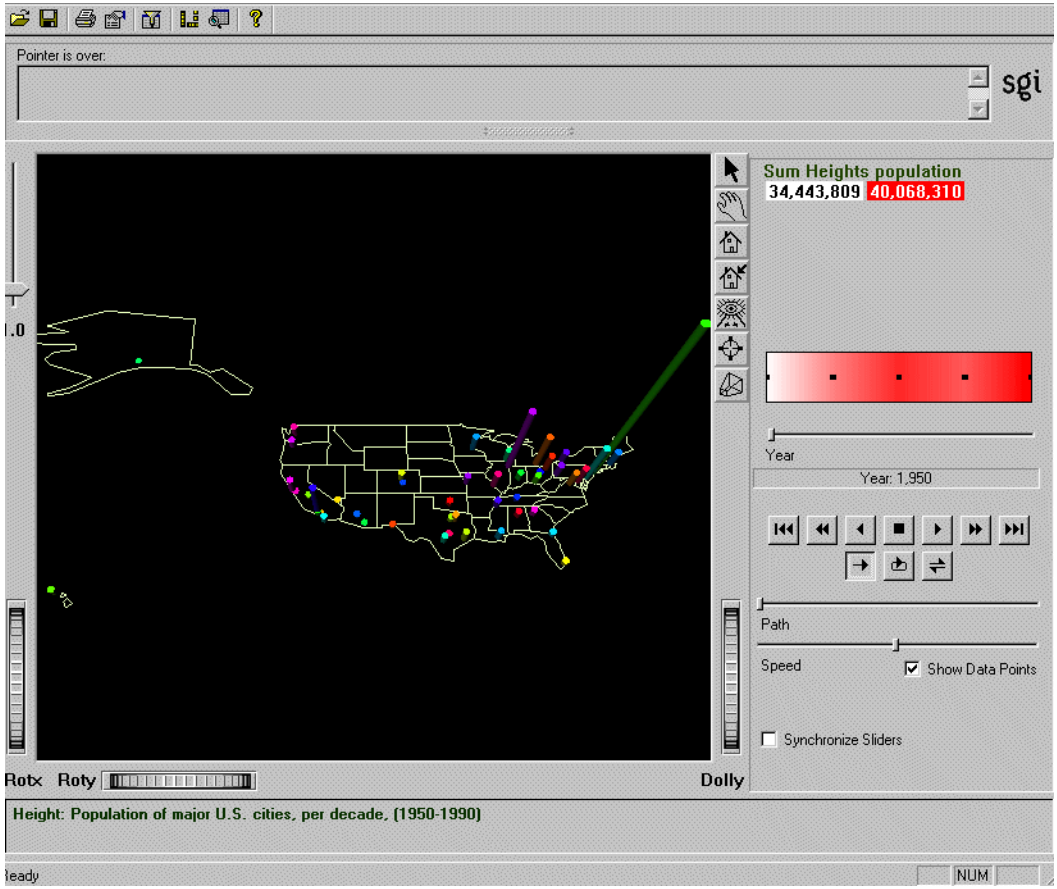


Figure 6-2 Map Visualizer with Bar Charts of Relative Population on Geographic Outline

Another landscape possibility is lines with endpoints at specific locations, each line showing an individual width and color (see Figure 6-3). The individuality of these lines corresponds to height and color properties of the usual objects and cylinders.

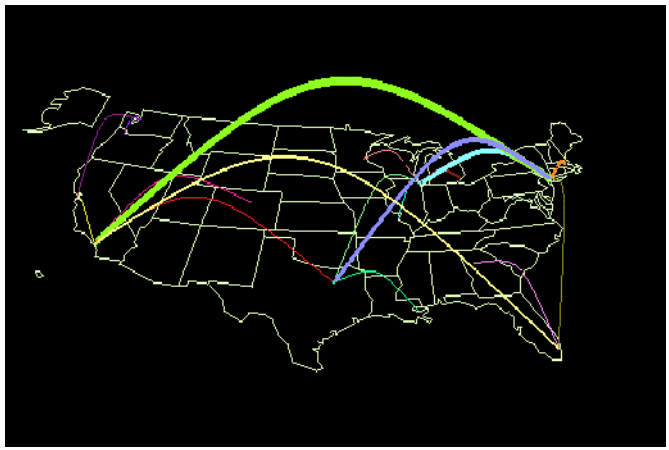


Figure 6-3 Sample Map Visualizer Showing the United States with Specific Endpoints

Starting the Map Visualizer

This is the most common way to start the Map Visualizer:

1. Bring up the Tool Manager, and select a server and data source. Refer to Chapter 2, "Accessing Data with MineSet," if you need help. Use the *churn* dataset for this example.
2. In the Tool Manager Data Destinations pane, click the Viz Tools tab; from the lower row of tabs, click Map Visualizer.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Preparing the Data

This sections walks you through the steps of transforming geographic data so that it can be displayed by the Map Visualizer.

Aggregating the Data

You must now transform the data into a smaller dataset before distributing it across the geographical map. Such a transformation is called aggregation.

1. In the Tool Manager Data Transformations pane, click *Aggregate.*

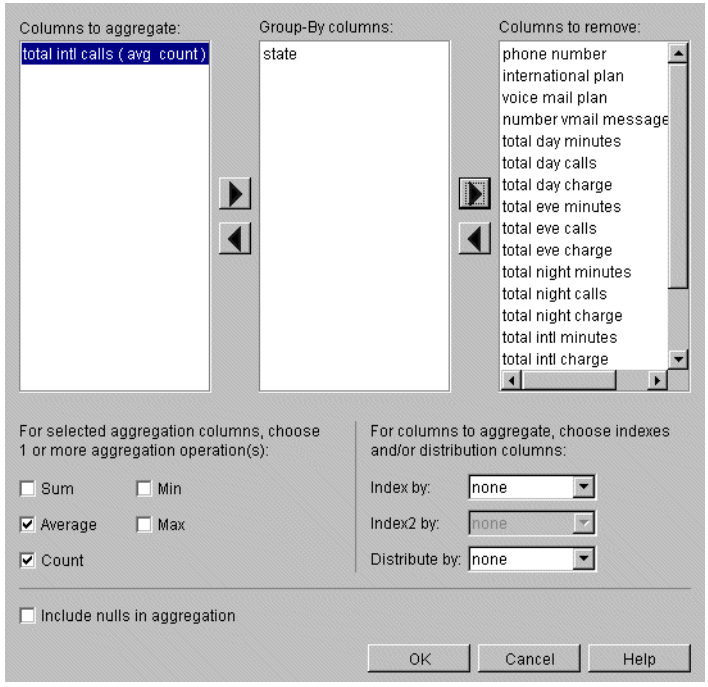


Figure 6-4 Aggregating in Map Visualizer

2. Move the column of interest to the “Columns to aggregate” list, and identify the mathematical action using the bottom check boxes. Leave the spatial grouping in the central “Group-By columns” field, and remove all other columns from the visualization. See Figure 3-5 on page 43 for an explanation of the action.

For example, using the *churn* dataset, in the Aggregate dialog box, move `total intl calls` into the left list by clicking on the column, then the left-pointing arrow. Check *Average* and *Count* on, and turn off *Sum*. Leave `state` in the central list and move all the rest to the right list. (Hold down the Ctrl key to gather multiple columns.) Click *OK* to apply your choices.

The next step is to find the appropriate spatial shapes.

Choosing the Map Shapes

1. Click the *Tool Options* button on the Viz Tools tab of the Data Destinations pane to access the geographic files. Tool Manager displays the Map Visualizer Options dialog box.
2. Click the button to the right of the Entities text field and select one of the geographic entities files. These files are in the location where MineSet was initially installed, under *MineSet\config\mapviz\gfx_files*. This example uses *usa.state.hierarchy.*
3. Click *OK* to retrieve the selected file and *OK* again to dismiss the Map Visualizer Options panel. For more details about *.gfx files*, see the "Map Visualizer" entry in the *MineSet Enterprise Edition Reference Guide*.

   The next step is to link the visual elements to the columns.

Linking the Map Visualizer Elements

When linking the map elements to visual entities, some columns may require manipulation before they fit the visual elements criteria:

- If the column offers too many discrete values, see "Changing or Creating New Bins for Columns" in Chapter 3.
- If the column is the incorrect type, see "Changing a Column Type or Name" in Chapter 3.

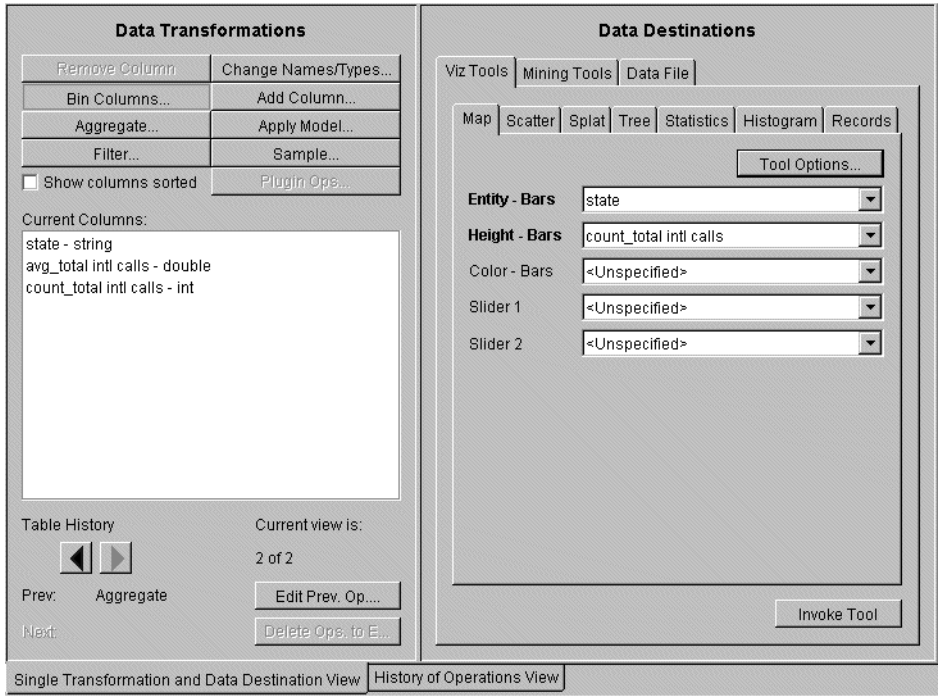


Figure 6-5 Mapping Columns to Visual Elements in Map Visualizer

1. From the Tool Manager Data Transformations pane, map the columns to entities in the Data Destinations pane. Map items by making a choice from the popup menus next to each entity.

   For this example, for Entity-Bars choose `state`, and for Height-Bars choose `count_total intl calls`.

   For further mapping possibilities see “Mapping to Sliders” and “Mapping to Maps.”

2. Click *Invoke Tool* to view the geographic distribution of your dataset.

Mapping to Sliders

When you suspect a value changes according to a specific criteria, such as income varying by age, you can map a column to a slider when you are mapping other columns to other visual elements. For example, Figure 6-1 has a slider mapped to the column Year. This allows you to show how the data varies year by year by moving the slider.

Columns can be mapped to sliders if that column is numeric (of the type int, float, double) or binned. If the column is already binned it has a `_bin` suffix. Column type is noted after the name of the column in the Current Columns field, for example `total day calls - double`.

Sliders can be created automatically or manually. The Tool Manager automatically generates sliders when you map columns to them. Sliders are created through a combination of automatic binning and aggregation. See the glossary entries: *Binning* and *Aggregation*. These automatic operations occur after clicking *Invoke Tool*.

Whenever the current data table has an array column, the Tool Manager can create a slider corresponding to the indices of the array column. The current slider indices are indicated in the Tool Options dialog box. To display them, from the Tool Manager Data Destination pane, click the Viz Tools tab, then click the *Tool Options* button. All array columns in the current table must have the same indices; otherwise, no slider is created.

See "Creating New Columns by Aggregation" in Chapter 3 and the "Aggregate" entry in the *MineSet Enterprise Edition Reference Guide* for more information on creating arrayed columns.

For details of automatic and manual sliders, see the "Slider Creation for Mapviz, Scatterviz, Splatviz" entry in the *MineSet Enterprise Edition Reference Guide.*

Mapping to Maps

MineSet supplies a variety of *.gfx* and *.hierarchy* files consisting of recognizable geographic shapes and other spatial entities. Unless you map Entity-Bars to a graphic element, you will not get a recognizable map.

If you want to show another visual entity or geographical map, replace "Choosing the Map Shapes" on page 110 with these steps:

1. From Tool Manager Viz Tools tab, click the *Tool Options* button.

2. Choose the options you want from the dialog box. The MapViz Options dialog box appears as shown in Figure 6-6.

   - For map-shaped entities, click the *Find File* button to the right of the Entities File text field and select one of the *.hierarchy* geographic entities files to be used for the representation of objects in the Map Visualizer's main window. These can be found in */config/mapviz/gfx_files*.
   - For rectangles click the *Find File* button to the right of the Outlines File text field to specify outline objects to draw. These appear as a flat plane on which the 3D entity objects are placed.

   Specifying Entities File and Outlines File fields are optional. If the Entities File is not supplied, you see entity objects consisting of rectangles that are arbitrarily arranged in the scene.

   For description of other options, see the "Map Visualizer" entry in the *MineSet Enterprise Edition Reference Guide*.

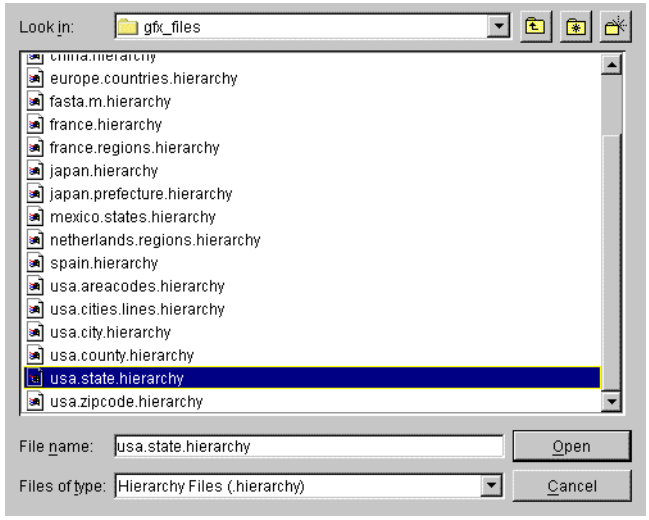


Figure 6-6 Map Visualizer's Options Dialog Box

3. Once you have finished making changes to the Tool Options dialog box, click *OK* to return the Tool Manager's main screen. You can then click *Invoke Tool*.

Examining the Map Visualizer

As with any of the visualizers, if you started the Map Visualizer without specifying a configuration file, perhaps using the Visual Tools pulldown menu from the Tool Manager, the main window appears empty, and only the File, View, and Help pulldown menus can be used. To show all menus and controls, use File > Open to see a list of configuration files. These files are in the *MineSet \examples* directory, in the location where MineSet was initially installed.

When you have specified a valid configuration file, its geographical landscape is visible. For example, Figure 6-7 shows the results of specifying *population.usa.mapviz* and moving the *Year* slider to the far right.

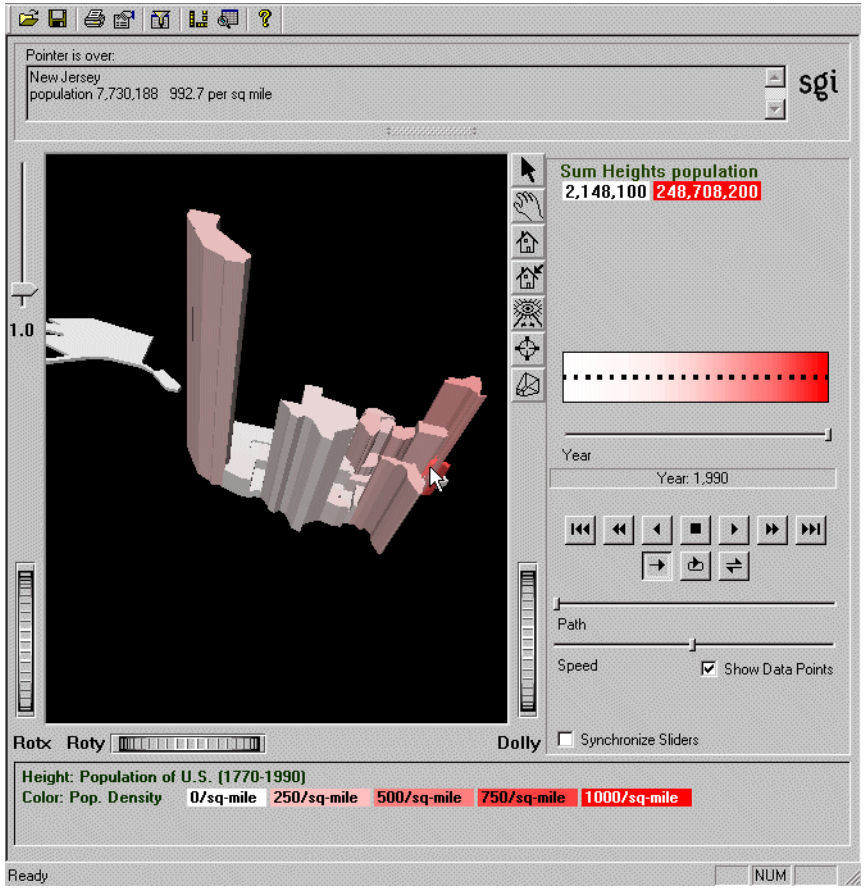


Figure 6-7 Population.usa.mapviz Example with the Year Slider at 1990

This shows the population and population density for each state of the United States. The population of each state is represented by the height of the state's graphical shape. Heights are relative to one another across the entire range of the animation controls, which are in the area to the right of the main window.

Viewing Modes

The two modes of viewing are *grasp* and *select*. To toggle between these modes, move the cursor into the main window, and press the Esc key. You can also change from one mode to the other by clicking on either the arrow or the hand on the window border. For more details see page 68.

Drilling Up and Drilling Down

You can drill down in select mode to get finer detail, or drill up to see a more general view only if the *.data* and *.hierarchy* files support drilling down and up. The Map Visualizer sample files support drilling down and up.

Table 6-1 Drilling Techniques

To:	Do this:
Drill down on object	Click the right mouse button over the object.
Drill down on all objects	Click the right mouse button over the background.
Drill up on object	Hold the Ctrl key and click the right mouse button (or click the middle mouse button) over the object.
Drill up on all objects	Hold the Ctrl key and click the right mouse button (or click the middle mouse button) over the background.

You can repeat drilling down to the finest level of granularity supported by the data. The greater the number of objects, the longer the Map Visualizer takes to reconstruct the scene after a drilling operation, and the slower the performance when moving the animation controls.

By default, the Map Visualizer initially displays objects at the greatest level of detail, so that, initially, only drill-up (to coarser granularity) is active.

Visualizing Null Data

When a lack of data, or nulls, occurs in a particular position, a question mark is displayed. In the Map Visualizer, nulls occur, for example, when the database or data file contains a null, and other specific causes (see "Nulls" in the *MineSet Enterprise Edition Reference Guide*).

Creating an Animation in the Map Visualizer

You can create an animation using the animation control panel to the right of the main window. The animation window appears only if the dataset has one or two sliders mapped. See the "Animation" entry in the *MineSet Enterprise Edition Reference Guide*.

Manipulating the Map Visualizer Results

You can access additional Map Visualizer functions using the pulldown menus View and Selections.

Changing the Map Visualizer Display

The View menu gives you the options summarized in Table 6-2. For greater detail, refer to the View menu entry in the *MineSet Enterprise Edition Reference Guide*.

Table 6-2 View Menu Choices for Map Visualizer

Option	Description
Filter Panel	Brings up the Filter Panel to filter the number of entities displayed in the main window based on selected criteria. See the Filter entry in the *MineSet Enterprise Edition Reference Guide*.
Window Decoration	Shows or hides external controls around the main window.
Animation Panel	Shows or hides any animation control panel to the right of the main window.
Data Points	Causes a grid of black dots indicating precise data values to appear or disappear in the 2D summary window.

Table 6-2 (continued) View Menu Choices for Map Visualizer

Option	Description
Use Random Colors	Activates or deactivates the configuration file's color mapping specifications.
Display X-Y Coordinates	Used for developing and refining *.gfx* files, not for data analysis. For details see the Map Visualizer entry in the *MineSet Enterprise Edition Reference Guide*.

Selecting and Drilling Through

The Selections menu lets you perform the actions outlined in Table 6-3 on your selected data in Map Visualizer. For further details see the "Map Visualizer" entry in the *MineSet Enterprise Edition Reference Guide*.

Table 6-3 Selections Menu Choices for the Map Visualizer

Option	Description
Show Values	Displays a Record Viewer table of values for the selected objects.
Drill Through	Opens the Drill Through Dialog, which gives you the option to view the selected data in the Record Viewer or send the data to the Tool Manager as a filter, a new column, or as SQL. You can send the records to a new copy of Tool Manager or you can send all non-selected records to the Tool Manager (complement drill through).
Use Slider on Drill Through	Determines if the slider position is used in creating the drill-through expression. The default limits the drill-through to those records defined by the slider's position.

For further details on drill-through, see the "Drill-Through" entry in the *MineSet Enterprise Edition Reference Guide*.

For a description of additional sample files for Map Visualizer, see the "Sample Files" entry in the *MineSet Enterprise Edition Reference Guide*.

Understanding Predictive Modeling

This chapter introduces predictive modeling and explains how to use MineSet to generate predictive models from datasets. The visualizations generated by MineSet then help you understand how the model operates. Drill through techniques allow you to examine the underlying data. The topics discussed are:

Subsequent chapters show the visualizations and models in greater depth. For greater technical details, refer to the *MineSet Enterprise Edition Reference Guide.*

Overview of Predictive Modeling

MineSet contains a number of analytical data mining algorithms that generate predictive models. A predictive model predicts the value of an attribute called the label, given the value of several other attributes in the record. If there is enough information in the attributes, the model can accurately predict the label. The predicted label refers to the unknown characteristic in a given record. For example, in a credit history dataset, given the values for attributes "age", "sex", and "occupation", the task of a predictive model might be to predict the value of the attribute "credit_risk."

If you have data where the label is unknown, use predictive modeling, but if you want to visualize data where the label is known, descriptive modeling is more appropriate. MineSet's two types of predictive modeling are classification and regression. Classification predicts discrete values and regression predicts values in a continuous range. Each of these is detailed in subsequent chapters. This chapter discusses processes common to all predictive models.

Generating a Model

The MineSet algorithms for generating predictive models depend on the type of attribute that will be predicted. For classification tasks, MineSet's algorithms generate an Evidence model, a Decision Tree, an Option Tree, or a Decision Table. For regression tasks, MineSet generates a Regression Tree.

The following five subsections provide a short introduction to the various types of predictive models that MineSet supports, and illustrate how they may be generated. You can tailor the way the model-generating algorithm works from Tool Manager's Data Destinations pane using the *Advanced Options* button. For a more detailed technical description of how these options effect the model generators, see the "Inducer Modes in Tool Manager" entry in the *MineSet Enterprise Edition Reference Guide.*

See Appendix A in the *MineSet Enterprise Edition Interface Guide* for a list of further readings about individual inducers.

Evidence Model

This section shows you how to generate a predictive model called the Evidence Classifier, for more details see Chapter 10, "Modeling and Predicting with the Evidence Classifier and Visualizer."

To build a predictive model, begin by selecting a dataset using the Tool Manager. See "Starting MineSet" in Chapter 2 for details. This example uses the churn dataset to build models that predict which customers are likely to change telecommunications providers.

1. In the Tool Manager Data Destinations pane, click the Mining Tools tab.
2. Click the Classify tab, and make these selections from the popup menus:

   Mode: Classifier Only

   Inducer: Evidence

   Discrete Label: churned (for example)

   You are about to induce an evidence classifier to help characterize the customers who are likely to churn. In this case you use all the data to build the model, and don't hold any in reserve for testing the accuracy.

3. Click *Go.*

   The Status window on the bottom of Tool Manager shows progress and summary information about the induction process. When the induction step is done, the Evidence Visualizer is automatically invoked, showing a visual representation of the model as in Figure 7-1.

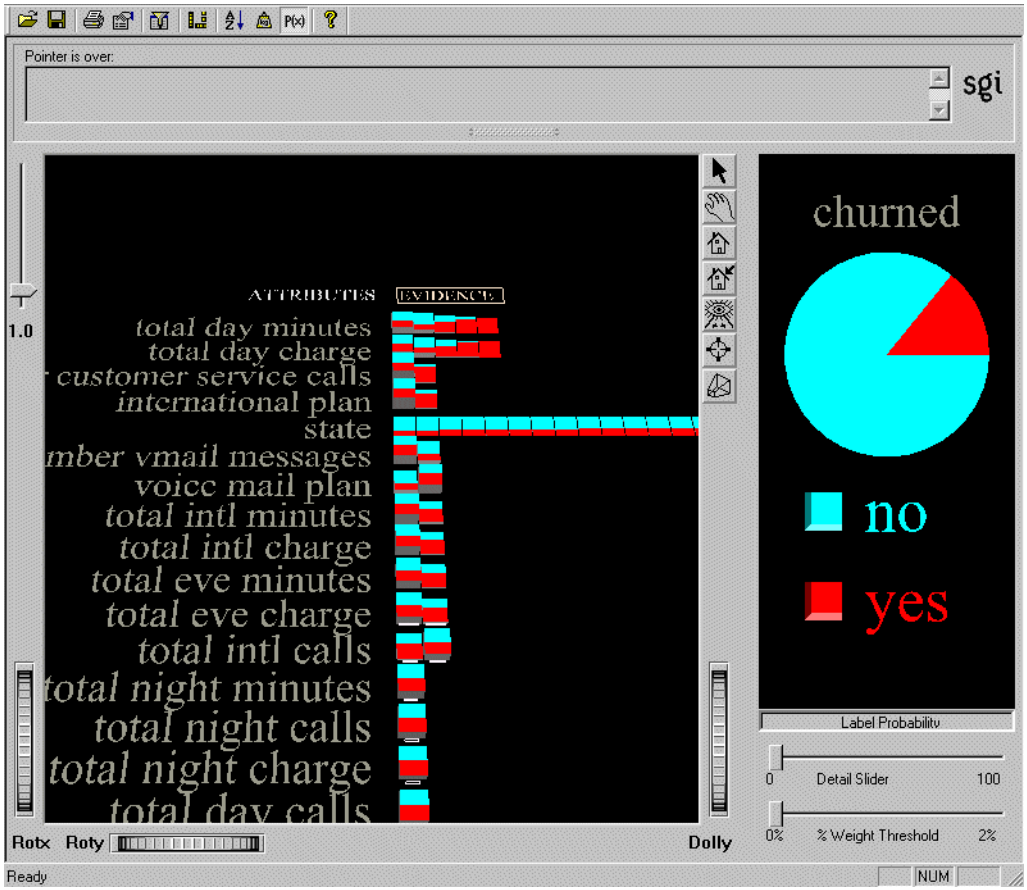


Figure 7-1 Evidence Visualization Generated by the Evidence Inducer for the Churn Dataset

This model shows how well each attribute predicts the value of churn. For an explanation of how to use this model for predictive behavior, see Chapter 10, “Modeling and Predicting with the Evidence Classifier and Visualizer.”

Decision Tree Model

In the previous example you used the Evidence method to induce and visualize the classifier. If you choose the Decision Tree method you can show how various attributes interact, that is, how combinations of attribute values affect the predicted label. With the Decision Tree, how the data is distributed at subsequent nodes (decision points) depends on the decision made at the previous node. In the example shown in Figure 7-2, the root decision point splits on whether the total day minutes attribute is greater than or less than 264.45. This affects the distribution of values at subsequent branches; clearly, higher bars appear on the left of the visualization.

To generate a Decision Tree Model follow the same steps as for the Evidence Model, except when you choose the Inducer on the Classify tab, choose Decision Tree. The task is explained in full in Chapter 8, "Modeling and Predicting with Decision, Option, and Regression Trees." The result of this method is shown in Figure 7-2. The fact that the distribution of classes in subtrees branching out from the root are different shows that interactions between attributes are significant.

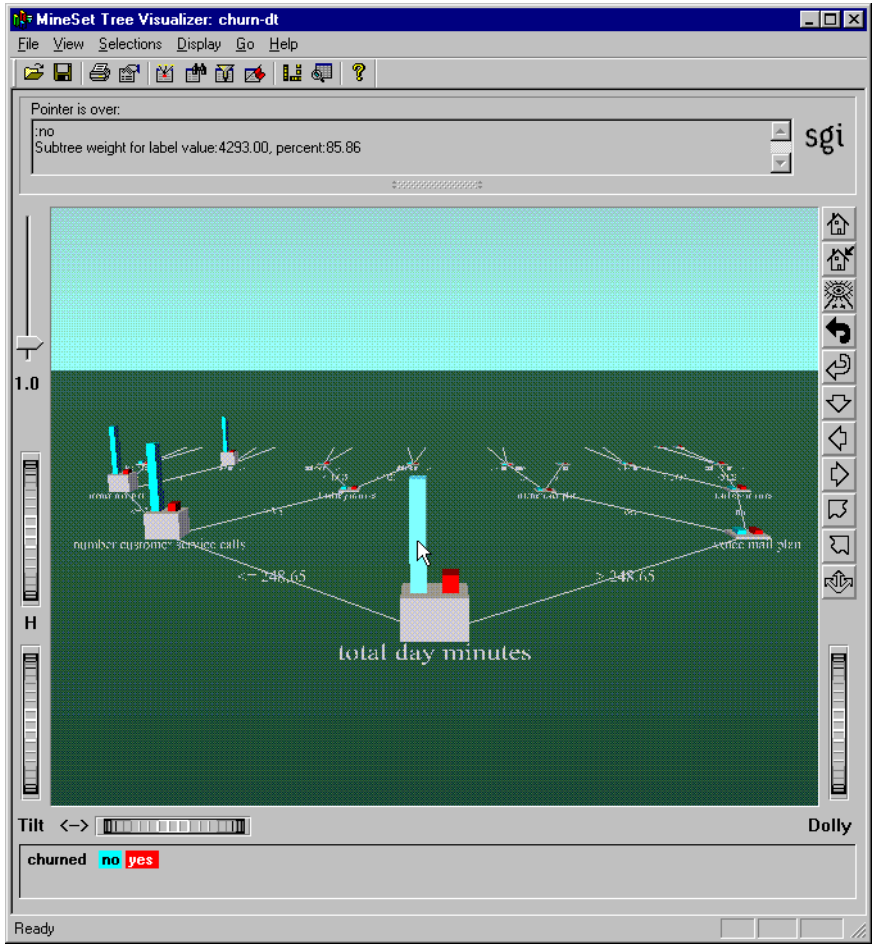


Figure 7-2 Decision Tree Generated by the Decision Tree Inducer for Churn Dataset

The bars at each node of this decision tree represent the distribution of label values (classes). Placing the cursor over a bar shows the record weight and percentage for that label value in the status area above the display. The height of the base of each node represents the weight (record count if no weighting is used) of the records.

In this example the block that is the root of the decision tree is marked with the label "total day minutes," indicating that this is the single most important factor in predicting churn—how long these customers talked, with a dividing threshold of 264.45 minutes.

The Decision Tree is explained in further detail in Chapter 8, “Modeling and Predicting with Decision, Option, and Regression Trees.”

Option Tree Model

A Decision Tree Inducer picks the single “best” attribute for each subtree; however, there might be several good attributes on which to split. In such cases, an Option Tree can create option nodes. An Option Tree appears similar to a Decision Tree but is generated by the Option Tree inducer.

The top node is an “Option node,” showing that several good attributes can be chosen at the root. You can set the number of options generated at each option node, representing the best attributes at that point. A branch is then generated for each option. The example shown in Figure 7-3 uses the cars dataset. The Option Tree is explained in further detail in Chapter 8, “Modeling and Predicting with Decision, Option, and Regression Trees.”

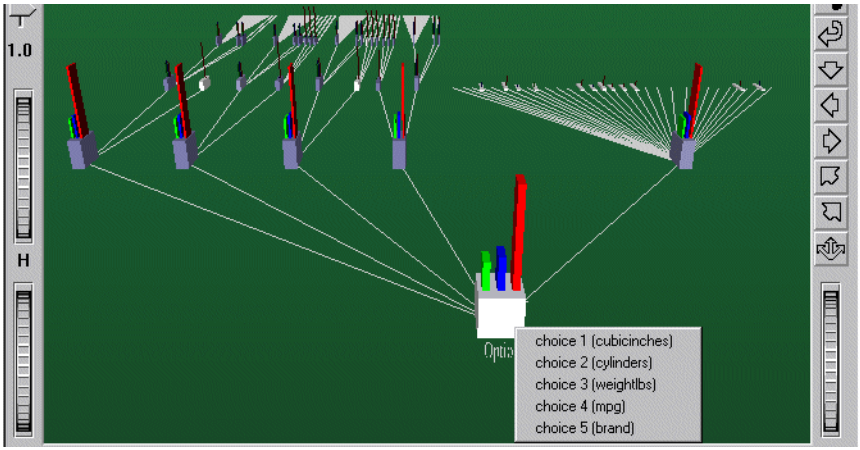


Figure 7-3 Option Tree Generated by the Option Tree Inducer for the Cars Dataset

In the example dataset (Figure 7-3), the task is to predict whether a car is manufactured in Europe, Japan, or the U.S. The Decision Tree Inducer picks cubic inches for the root. By contrast, the Option Tree inducer chooses several options: cubic inches, cylinders, weight, mpg, and brand as alternative good choices for the root. You can see these choices by clicking the Ctrl key and right mouse button on the root.

Option nodes can appear elsewhere besides the root. With the default settings, however, they appear only at the root or one level below the root.

Option Trees usually take 10 to 15 times longer to build, and are much larger than, Decision Trees, but they provide two significant advantages:

- Flexibility—Option nodes let you see several likely options. Instead of settling for a single attribute, option nodes let you choose from several. When you fly over the tree, you can follow an option that is easier to understand or fits your background knowledge of the problem.
- Accuracy—Often, Option Trees have lower error rates than Decision Trees. Option Trees classify by letting each option "vote" for each label value, then averaging the votes. This is similar to having a panel of experts, each one attempting to predict the label based on a different main criterion. The option node averages all these experts' votes, resulting in a more stable, less risky classifier.

Decision Table Model

The Decision Table induces a model that presents correlations between pairs of attributes in a layered hierarchy. Decisions are made by the inducer in the same way as the Decision Tree, but attributes are evaluated across the entire level of the tree rather than on a specific sub-tree. The result is then presented as a hierarchical table rather than a tree. Clicking on the blocks in the visualization shows the details of how the rectangular block called a *cake* is composed, as shown in Figure 7-4.

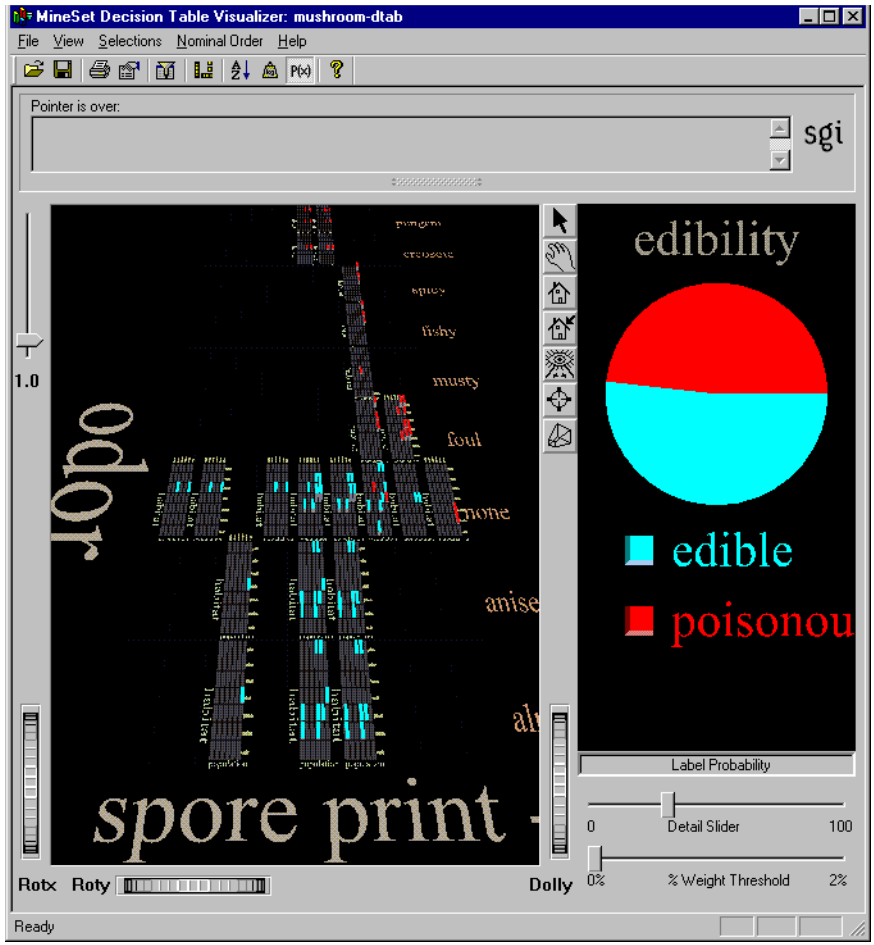


Figure 7-4 Decision Table Generated by the Decision Table Inducer for the Mushroom Dataset

This model shows the label probability in predicting the edibility of mushrooms.

See Chapter 9, “Modeling and Predicting with the Decision Table Classifier and Visualizer,” for a description of how to use the Decision Table.

Regression Tree Model

The Regression Tree Inducer generates a regressor, which is a predictive model similar to the classifier. The difference is that regressors predict labels with values in the continuous range, such as an individual's annual salary. Classifiers predict only discrete values, such as yes or no, or a specific range such as 20-100.

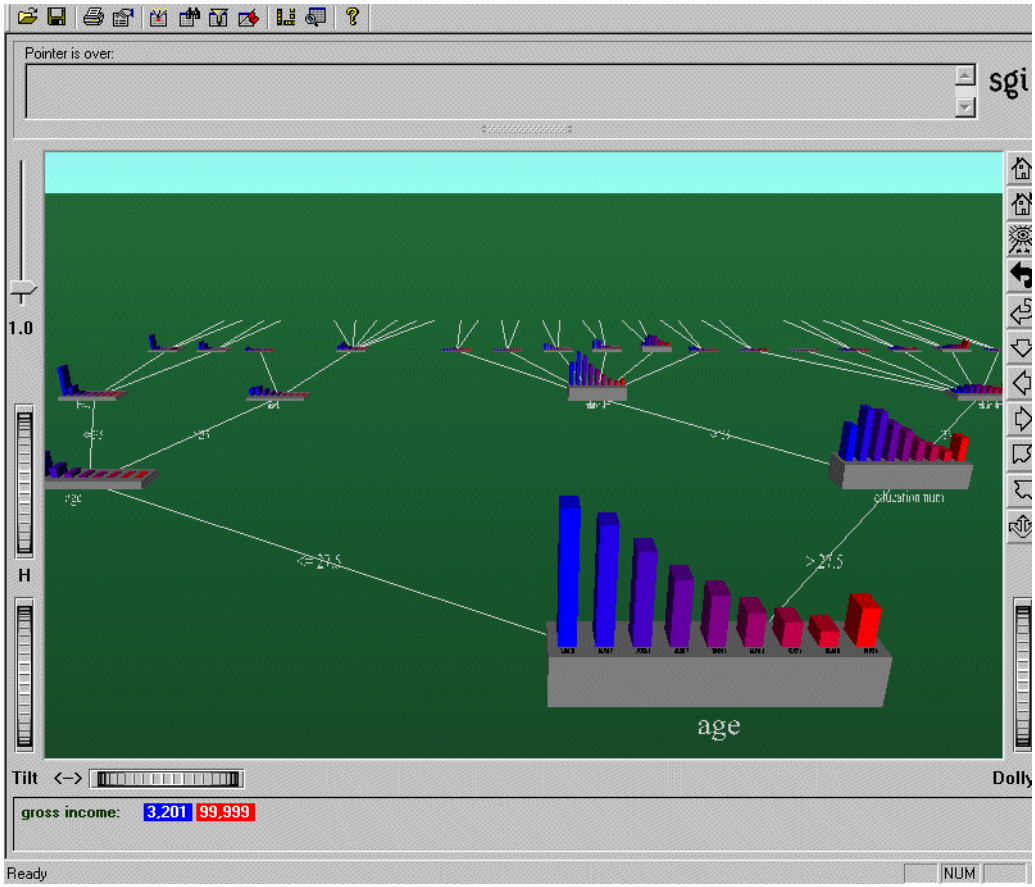


Figure 7-5 Regression Tree Generated by the Regressor on the Adult Dataset

The following example shows you how to generate a regressor, a task that is expanded in Chapter 8, "Modeling and Predicting with Decision, Option, and Regression Trees."

To build a predictive regressor, begin by selecting a dataset using the Tool Manager. See Chapter 2, "Accessing Data with MineSet," for details. This example uses the *adult* dataset.

1. In the Tool Manager Data Destinations pane, click the Mining Tools tab.
2. Click the Regress tab, and make these selections from the popup menus:

   Mode: Regressor Only

   Inducer: Regression Tree

   Continuous Label: gross income (for example)

   You are about to induce a regressor to help characterize the gross income of adults represented by records in the dataset. In this case you use all the data to build the model.
3. Click *Go.*

   The Status window on the bottom of Tool Manager shows progress and summary information about the induction process, or you can watch the progress indicator on the dialog box. When the induction step is done, the Regression Tree Visualizer is automatically invoked, showing the model visually as in Figure 7-5. Some datasets take a longer time to generate than others.

You can generate a regressor only for datasets using continuous attributes, that is, with a column containing data in a continuous range. If there are none, you can add a new continuous column from the Tool Manager Data Transformations pane, see "Removing and Adding Columns" in Chapter 3.

The Learning Curve classifier mode is discussed in "Creating a Learning Curve" on page 133.

Evaluating a Predictive Model

The goal of predictive models is to make predictions, therefore the usefulness of the model is clearly dependent upon the accuracy of its predictions. The accuracy of a model is measured by the error rate. This section discusses the range of MineSet's modes of modeling: Classifier Only using all the data and with no error estimation; and the two common methods for estimating error rate—holdout and cross-validation.

In the holdout method of estimating errors, a section of the dataset (commonly two-thirds) is held out to generate the model. The inducer uses the labels in this training set of data and constructs the model, the rest of the data is used to test the model and estimate the error-rate.

Cross-validation is a method of estimating classifier error that splits the dataset into a certain number of folds, or subsets, (commonly 10), and builds that same number classifiers. The process can be repeated multiple times to increase the reliability of the estimate.

Both holdout and cross-validation methods expect future records to be sampled from the same data distribution as the training set. A more complete discussion may be found in "Error Estimation" and "Cross-Validation" entries in the *MineSet Enterprise Edition Reference Guide*.

Classifying Using All Your Data

The *Classifier Only* mode of building the model uses all available data to build the classifier. It is useful when you are not concerned about estimating the error rate of the predictive model, such as when you just want to see the visualization.

Suppose you want a Decision Tree for the iris dataset induced using the *Classifier Only* mode. This example goes through the entire sequence of building and applying the model. You could find out when the model misclassifies by following these steps:

1. From the Tool Manager window select a dataset using the File > Open New Data File pulldown menu. Choose the *iris.schema* dataset.
2. From the Tool Manager Data Destinations pane, click the Mining Tools tab, then click the Classify tab and choose:
   - *Mode:* Classifier Only
   - *Inducer:* Decision Tree
   - *Discrete label:* iris type

   You are using all available data to build a Decision Tree model that determines what attributes distinguish between types of irises.
3. Click *Go* to run the Decision Tree. Examine then dismiss the visualization. The created model used all the data. The following steps show you how to evaluate correct and incorrect predictions in such a case.

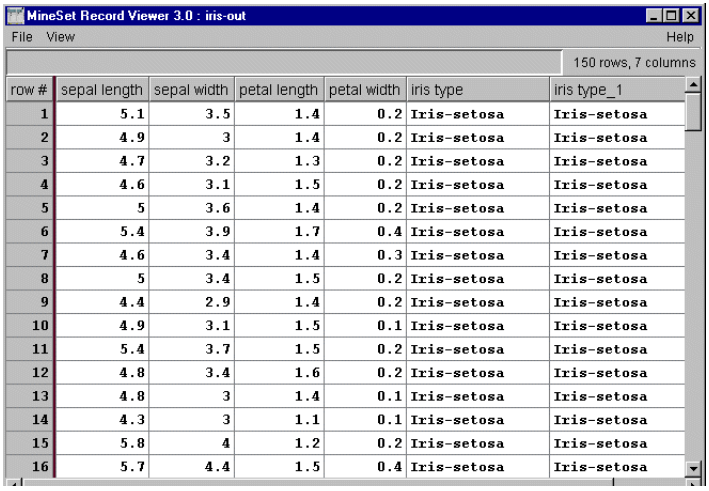
4. In the Tool Manager Data Transformations pane click *Apply Model*. Select *iris-dt.class* (the iris decision table classifier), and click *OK* to accept the default New column name `iris type_1`, which now contains the predicted labels.

5. Click the *Add Column* button to create another new column that will identify errors (call the column `iris_fault`). You can then add a column that is defined as type **int** with the expression (`'iris type' != 'iris type_1'`).

   To build the expression, select from the left list, and press the right-pointing arrow, or type directly in the text field.

6. Click *Check Expression*, and then click OK. This adds the new column `iris_fault`, based on `iris type_1`. The new column has a 1 whenever the classifier misclassifies, and a zero when it correctly classifies.

7. Verify your results. In the Tool Manager Data Destination pane, click Viz Tools, choose Record Viewer from the Tool popup menu and click *Invoke Tool*. An example of the result is shown in Figure 7-6.



row #	sepal length	sepal width	petal length	petal width	iris type	iris type_1
1	5.1	3.5	1.4	0.2	Iris-setosa	Iris-setosa
2	4.9	3	1.4	0.2	Iris-setosa	Iris-setosa
3	4.7	3.2	1.3	0.2	Iris-setosa	Iris-setosa
4	4.6	3.1	1.5	0.2	Iris-setosa	Iris-setosa
5	5	3.6	1.4	0.2	Iris-setosa	Iris-setosa
6	5.4	3.9	1.7	0.4	Iris-setosa	Iris-setosa
7	4.6	3.4	1.4	0.3	Iris-setosa	Iris-setosa
8	5	3.4	1.5	0.2	Iris-setosa	Iris-setosa
9	4.4	2.9	1.4	0.2	Iris-setosa	Iris-setosa
10	4.9	3.1	1.5	0.1	Iris-setosa	Iris-setosa
11	5.4	3.7	1.5	0.2	Iris-setosa	Iris-setosa
12	4.8	3.4	1.6	0.2	Iris-setosa	Iris-setosa
13	4.8	3	1.4	0.1	Iris-setosa	Iris-setosa
14	4.3	3	1.1	0.1	Iris-setosa	Iris-setosa
15	5.8	4	1.2	0.2	Iris-setosa	Iris-setosa
16	5.7	4.4	1.5	0.4	Iris-setosa	Iris-setosa

Figure 7-6 Iris Misclassification Example

You can then apply your results to a scatterplot so that the new column is mapped to color, with the colors set such that green is 0 (no error) and 1 is red (error). See Chapter 4, "Examining Data with the Scatter and Splat Visualizers," for examples of the Scatter Visualizer.

Holdout Error Estimation

Instead of using all the data to build the model, you can hold out part of the data as a training set to induce the classifier. The Classifier and Error mode (MineSet's default mode) automatically partitions the dataset into independent training and test subsets. The proportion of data used as the training set can be changed to fit your requirements.

To see an example of how holdout error estimation applies to a selected dataset, follow these steps:

1. Start with the *iris.schema* dataset (see page 129).
2. From the Tool Manager Data Destinations pane click the Mining Tools tab, then click the Classify tab and choose:
   - *Mode:* Classifier and Error
   - *Inducer:* Decision Tree
   - *Discrete label:* iris type

   You are partitioning the iris dataset into two sets. The training set containing two-thirds, the default holdout ratio of records, is used to build the Decision Tree model. The remaining one-third of the records are used to estimate the error rate of the model.
3. Click the *Advanced Options* button to vary the holdout ratio of the training set using the text panel shown in Figure 7-7. You may also change the random seed to get different records in the training and test sets. See the glossary entry *Random Seed*. Dismiss the panel by clicking *OK*.

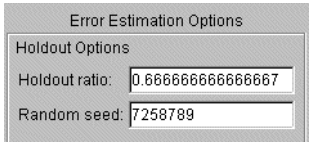


Figure 7-7 Error Estimation Options with Holdout

4. Run the Decision Tree inducer by clicking *Go.*

The error rate of the inducer estimated using the test set (the remaining one-third of the data) is displayed in the status area at the bottom of the main window. You can scroll down to see all the information.

Cross-Validation Error Estimation

Use cross-validation for building the final classifier or for small datasets. The process used should explain the reason. Cross-validation is a method for getting a more precise estimate of error. In *n-fold* cross-validation (where *n* represents any number you care to name and *fold* is number of subsets into which you divide the data), the dataset is partitioned into *n* independent subsets. In turn each of these subsets is held out and the remaining *n*-1 subsets (one less than the number you originally specified) are combined to form a training set. The resulting model is evaluated using the held-out subset. These *n* independent estimates are then averaged and the data is combined to build the final model. *N*-fold cross validation takes approximately *n*+1 times longer than Classifier and Error, or Classifier Only methods.

To apply cross-validation to a selected dataset and set the number of Folds:

1. From the Tool Manager Data Destination pane, choose the Mining Tools tab and click Classify.
2. From the Mode popup menu choose *Estimate Error* to assess the error if all the data were used (as with Classifier Only mode). Running time is likely to be long, since cross-validation is used. Use *Estimate Error* when there is little data. The induced classifier is exactly the same as the one induced by the *Classifier Only* mode.
3. Click the *Advanced Options* button to change the way the cross-validation operates. In Error Estimation Options (see Figure 7-8), you can set the number of Folds and the number of Times to repeat the error estimation process. Accept any changes by clicking *OK*.

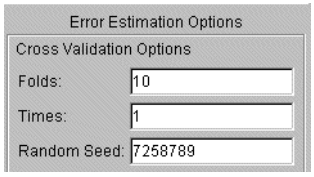


Figure 7-8 Error Estimation Options With Cross-Validation

4. Run the algorithm by clicking *Go.*

For a more detailed discussion of Error Estimation, see the its entry in the *MineSet Enterprise Edition Reference Guide.*

Creating a Learning Curve

The Learning Curve shows the error of the classifier generated by an inducer in proportion to the number of records used to create the classifier. Typically, the more records used to generate the classifier, the lower its error.

You create a Learning Curve by indicating the range of training set sizes you want to investigate, and the number of times you want to build an inducer at each training set size. To indicate the range of training set size you want to investigate, enter the start (minimum) and end (maximum) number of records and the number of points to examine. Each classifier is generated using a random sample of the records, and its error is estimated using the remaining records (those not used for training).

1. Start with the *churn.schema* dataset used on page 122.
2. From the Tool Manager Data Destinations pane click Mining Tools, then click the Classify tab and choose:
   - *Mode:* Learning Curve
   - *Inducer:* Decision Tree
   - *Discrete Label:* churned (for example)

   You will be prompted that the phone number column will not be computed because it contains too many unique values. You can either remove the column or simply click OK.
3. Click *Go*.

   The process takes a long time because the inducer is generating a classifier for each point on the curve. The status window displays the training size, the number of times the training is being done, and the mean error in each pass.

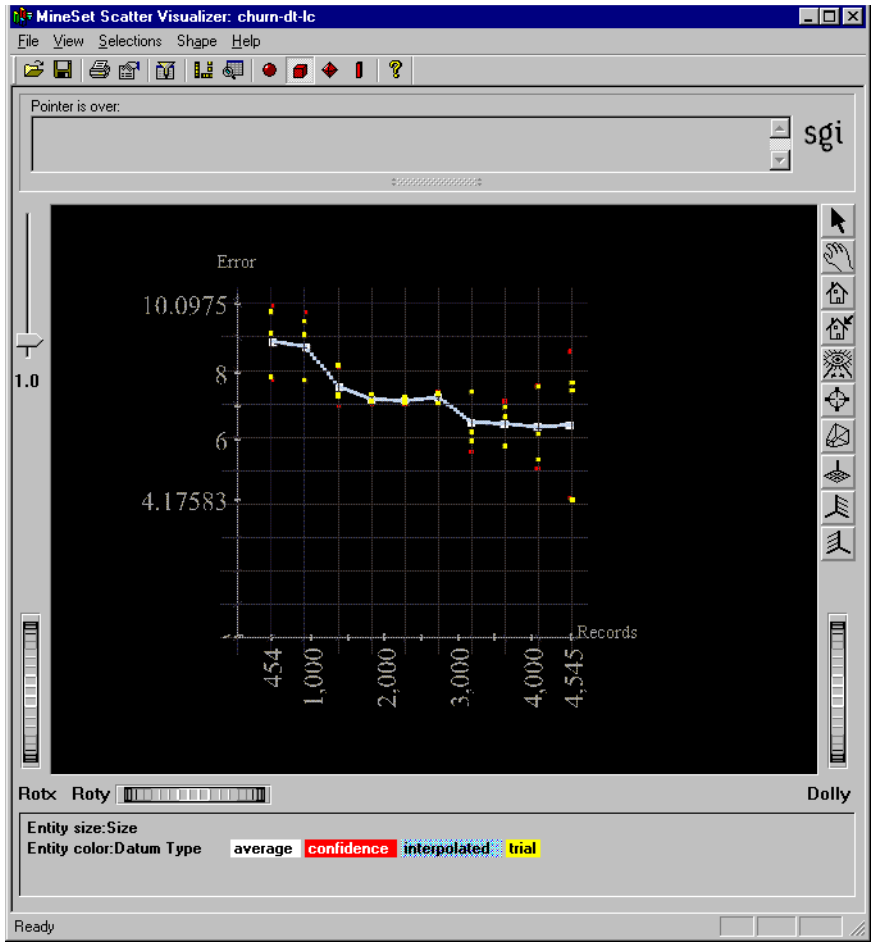


Figure 7-9 Learning Curve for Churn Dataset with Label Set to Churned

Figure 7-9 indicates the error rate for a given number of records in the churn dataset. If you train on 454 records, there is an error of 9.41%, if you train on almost all the records there is an error of 5.27%. By identifying the point at which the error rate no longer decreases, you can determine the number of training records needed to produce a classifier with the desired error rate. In this case it appears that training on 3,000 records is sufficient. You can pick a sample size based on a point on the learning curve with reasonable accuracy.

The graph shows four types of points:

- The yellow points are the actual error estimates taken from the runs (number of times the algorithm runs).
- The white points are averages.
- The blue points (sometimes resembling a blue line) interpolate between the white points.
- The red points show a 95% confidence interval about the average, based on actual error estimates for each run.

By default the algorithm uses all the records to build the learning curve, but you can specify other modes:

1. Using the previous Learning Curve, Decision Tree, and churned selections from the Tool Manager Data Destination pane, click *Advanced Options.*
2. From the *Advanced Options* dialog box, you can specify these Learning Curve Options (Figure 7-10)
   - Number of points in the learning curve—must be greater than one.
   - Number of runs per point.
   - Number of records to use at the start and end points.

   The number of records to use at each intermediate point is calculated automatically.

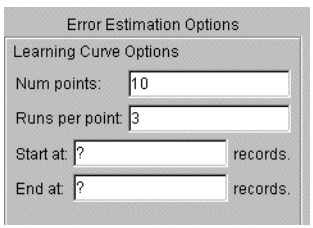


Figure 7-10 Learning Curve Options

If either the range or the number of points is left blank, it is calculated automatically based on the number of points in the learning curve and the total number of records in the training set. This default covers the entire range of the training set. For instance, assume a file containing 80,000 records. If you specify 3 points in the learning curve, the algorithm generates points at 20,000, 40,000 and 60,000 records. Often it is useful to

"zoom in" on a smaller range. For example, a learning curve might be generated only for a range of 1,000 to 10,000 records.

Increasing the number of runs per point increases the running time proportionally, but improves the estimate of the error. The default value of the number of runs is 3.

A learning curve may be generated for any inducer. You can run with the defaults or quantify the points in the curve, number of runs to make, or the range of records. Table 7-1 shows the effect of certain choices on a learning curve.

Table 7-1 Actions for Learning Curve Results

To achieve this result:	Do this:
Reduced confidence interval (greater accuracy)	Increase number of runs. Increase size of test set.
Speed processing	Use small sample (see "Sampling Data" on page 48).

For detailed information on the operation of inducers, refer to the "Inducer" entry in *MineSet Enterprise Edition Reference Guide*.

Applying a Predictive Model

After building a predictive model, you can apply it to other records to predict their label. For example, if you built a classifier for predicting iris type, you can apply the classifier to records containing only the descriptive attributes, and a new column will be added with the predicted iris type.

Selecting a Model

Models generated by MineSet are saved in your designated server directory. For example, *churn-dt.class* is a decision tree model generated from the churn dataset. You can apply this classifier to another data table you specify. After you have built the model using the Decision Tree for this example, follow these steps:

1. In the Tool Manager Data Transformations pane click the *Apply Model* button.
2. From the Test and Apply Model dialog box (Figure 7-11), select a model in the "Available models" pane. The right-hand pane then lists the column names and types required by that classifier.

   If these requirements match the current table that is loaded in memory, the buttons on the bottom of each tab (*OK*, *Run Test*, or *Fit Data*) are activated. If the requirements do not match, the columns that are missing are selected in the list on the right, and the bottom buttons are deactivated.

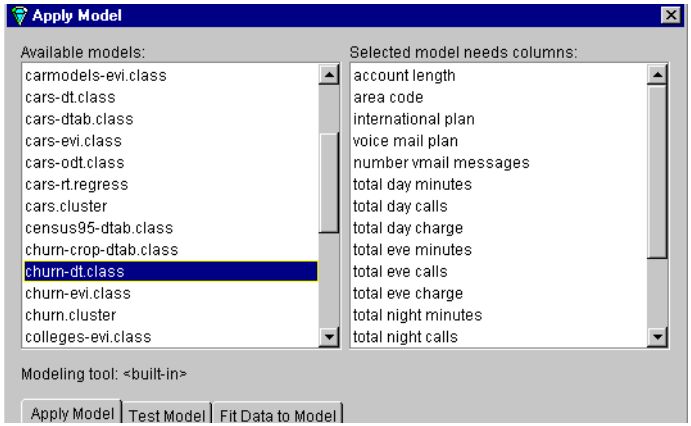


Figure 7-11 Test and Apply Model Window: Selecting a Classifier

Applying a Model

After you have selected the model you wish to apply:

1. In the Test and Apply Model dialog box, click the Apply Model tab (Figure 7-12) and select one of the following:

   Predicted label values: adds a column to predict, for example, if a customer will churn. The new column lets you see how well the model predicts when used on new data.

   Estimated probability values: estimates the probability that each record has a given label value. Select the available values from the popup menu. This adds a column that estimates the probability of correctness as a numerical value.
2. Enter a name for the new column in the text field, for example, `p_churned` to represent the probability of a customer churning.

3. Click OK.
4. To graphically display the model as a table of columns, from the Tool Manager Data Destination pane, click Viz Tools and from the Tool popup menu choose Record Viewer, and click *Invoke Tool*.

   The labeled columns `churn` and `p_churned` are displayed, allowing you to match predictions and results.

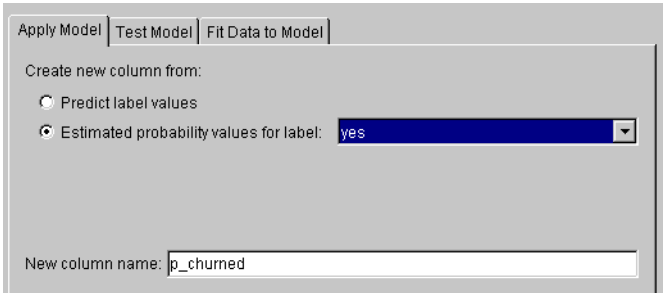


Figure 7-12 Apply Model Panel

Applying a model might be used in a marketing campaign, in which a model was generated from running the campaign in one city and generating a column recording (label) values according to the responses in that city. The model might be applied and campaign mail can then be sent only to people in another city who are labeled by the classifier as likely to respond, thereby saving mailing costs.

Where to Go from Here

The next several chapters describe various ways of looking at data, and include an indication of the particular strengths of each visualizer. Accordingly, you may find yourself beginning with Record Viewer to look at raw untransformed data, progressing through the Statistics or Histogram Visualizer, and then searching for important columns using the Column Importance tool in preparation for achieving a useful visualization of known data with the Scatter, Splat, or Map Visualizer.

If you have no hypotheses from which to start, you may want to start by clustering and let that method drive your knowledge discovery.

If you are ready to test and apply a model, and to determine model accuracy, several approaches can be found in Chapter 11, “Refining Predictive Modeling.”

Modeling and Predicting with Decision, Option, and Regression Trees

The Decision Tree, Option Tree, and Regression Tree visualizers look and feel similar, even though the algorithms behind them vary considerably. This chapter covers the similarities and the differences in the use of these three tree-based models. To learn more about the algorithms used by the inducers, and possible options, see the *MineSet Enterprise Edition Reference Guide*.

This chapter contains the following sections:

Sample configuration and data files have been provided to demonstrate the tree visualizers' features and capabilities. These files are in the *\examples* directory under *MineSet*, in the location where MineSet is installed.

Overview of Decision, Option, and Regression Trees

This section explains the use of the three types of tree, and some of the reasons for choosing one over the other. Each tree tool consists of an inducer, which builds the model, and a visualizer, which displays the resulting tree. The inducer is either a classifier (Decision Tree and Option Tree) or a regressor (Regression Tree).

The Decision Tree

The Decision Tree is a predictive model. It makes predictions by using the dependent, or known, attribute values to help determine the value of the label, or unknown attribute. The task of predicting the value of a nominal value (usually character strings such as "yes" and "no"), or an attribute that can only take on a small number of values, is referred to as classification. A decision tree classifies data by predicting the label for each record. The underlying structure used for classification is a decision tree, such as the one shown in Figure 8-1. Once the Decision Tree Inducer has built a classifier the data, the Tree Visualizer displays its structure.

The first element of the tree is the root node, representing all of the data. From there, the tree spits into two or more branches, each representing data with different values for a specific attribute (column). For example, Figure 8-1 is a decision tree visualization of the *cars* dataset, where the records are classified by cubic inches. The first split is on 169.5 inches or less on the right branch, and those over 169.5 cubic inches on the left. The tree can split on the same attribute at more than one node. The node can also split into more than two branches. For instance, for the *mushroom* dataset, with the records being classified into edible and non-edible, the first split is into nine branches, corresponding to the nine odor types.

The object is to reach nodes at the ends of the branches (leaf nodes) where the records all, or nearly all, have the same class (label). In Figure 8-1, the classes are the different ranges of cubic inches values listed at the bottom of the Decision Tree window.

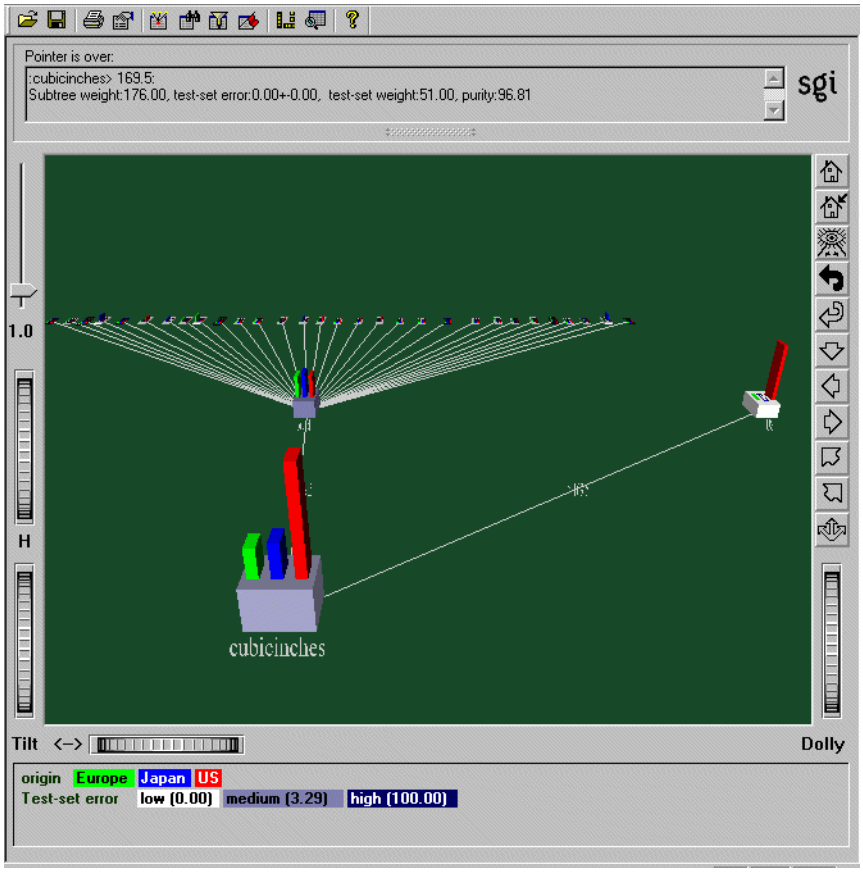


Figure 8-1 Decision Tree for the Cars Dataset

Option Tree

Like Decision Tree classifiers, Option Tree classifiers also assign each record to a class. The underlying structure used for classification is a decision tree, as described in the previous section. Once the Option Tree Inducer has built the classifier, the Tree Visualizer displays its structure. An option tree actually consists of several decision trees. Instead of picking an attribute to split on for the root node, it picks several, and makes a decision tree for each.

Figure 8-2 shows an option tree for the *cars* dataset, where the goal is to predict the origin of a car (U.S., Japan, or Europe). The Option Tree extends the Decision Tree Classifier by allowing *Option Nodes*. An Option Node gives several options that can be chosen at a decision node in the tree. For example, in Figure 8-2, the root is an option node with five options: cubic inches, number of cylinders, weight (in pounds), miles per gallon, and brand name.

Option nodes serve two purposes:

- They enhance comprehensibility of the factors affecting the determination of class label by showing several choices that can be made. Instead of using a single attribute at a node, an option node provides you with several options. When examining the tree, you can choose to follow an option that you believe is easier to understand or is better for predictions based upon your previous experience, or one you select based on the error estimate.

  In the *cars* dataset shown in Figure 8-2, you can follow the *cylinders* subtree because it has few values or the *weightlbs* subtree because its estimated error is lower (1.53). Error estimates are only estimates; generally, if the error difference between two options is less than twice their mean standard deviation, then statistically the errors are not different.

- They reduce the risk of creating an inaccurate classifier by averaging the votes of each option subtree. Every option leads to a subtree that can be thought of as an "expert." The option node averages these experts' votes. Such averaging can lead to a better classifier with a lower error rate.

  In the cars dataset, shown in Figure 8-2, the root node has an estimated error rate of 0.76%, which is lower than any of its children. Sometimes, the obvious choice for an option is not necessarily the best one to use. For instance, while brand might seem like an obvious attribute for this task, the training set might not contain all brands (and in fact, it does not contain all of them). For an unseen brand, Decision Tree guesses the majority class (U.S.) and makes two errors. However, when there are other options, they are averaged, and the error is reduced.

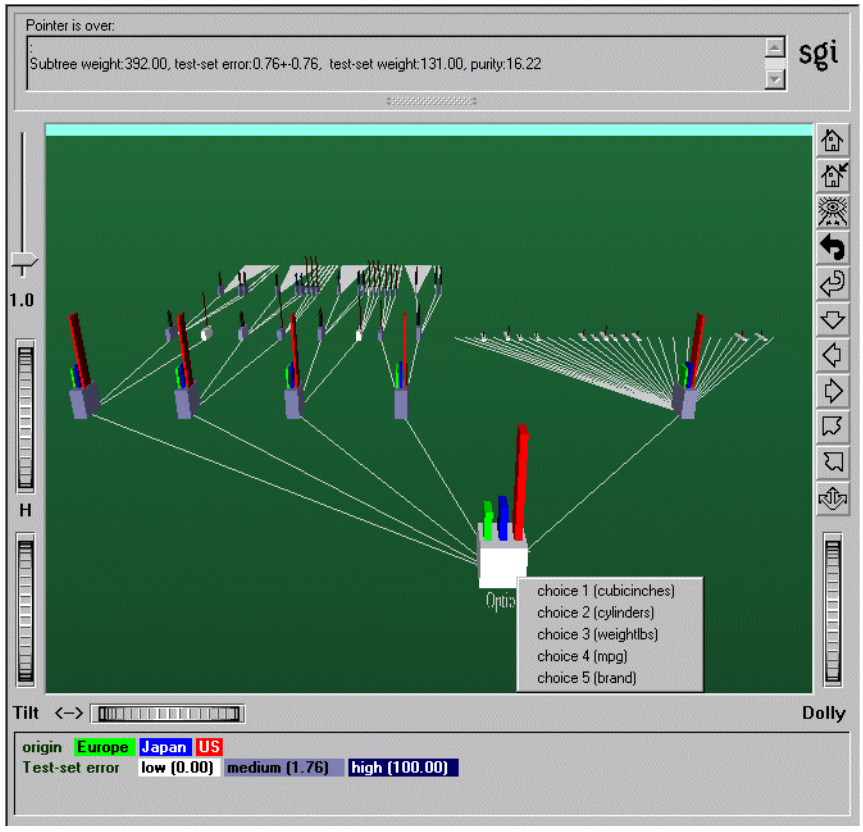


Figure 8-2 Option Tree for the Cars Dataset

Option Trees, however, have two disadvantages:

- The time necessary to build an option tree under the default setting is about 10 to 15 times longer than that needed to build a decision tree.
- The Tree Visualizer file that is created is very large, containing 10 to 15 times as many nodes as a regular decision tree.

Run the Option Tree inducer on your dataset to determine whether the advantages in comprehensibility and error rates justify the longer induction time. You might gain additional insight as to which attributes to remove or use when building a decision tree.

Regression Tree

Regression is the task of predicting a continuous label value, given a set of descriptive attributes. Regression and classification are similar, except that in classification the predicted label can take on only a small number of discrete values, for example, an iris may be classified by type as iris-setosa, iris-versicolor, or iris-virginica. In regression, the predicted label can be any value in a continuous range, for example, an individual's annual income may be any amount greater than zero.

When a regressor is generated, the Option Tree Visualizer displays the tree. This visualization can help you understand the regressor and how it makes predictions. In addition, it can provide valuable insight into the data itself. Once generated, a regressor can be used to predict the label value for unlabeled records.

The underlying structure used for regression is a regression tree, such as the one shown in Figure 8-3.

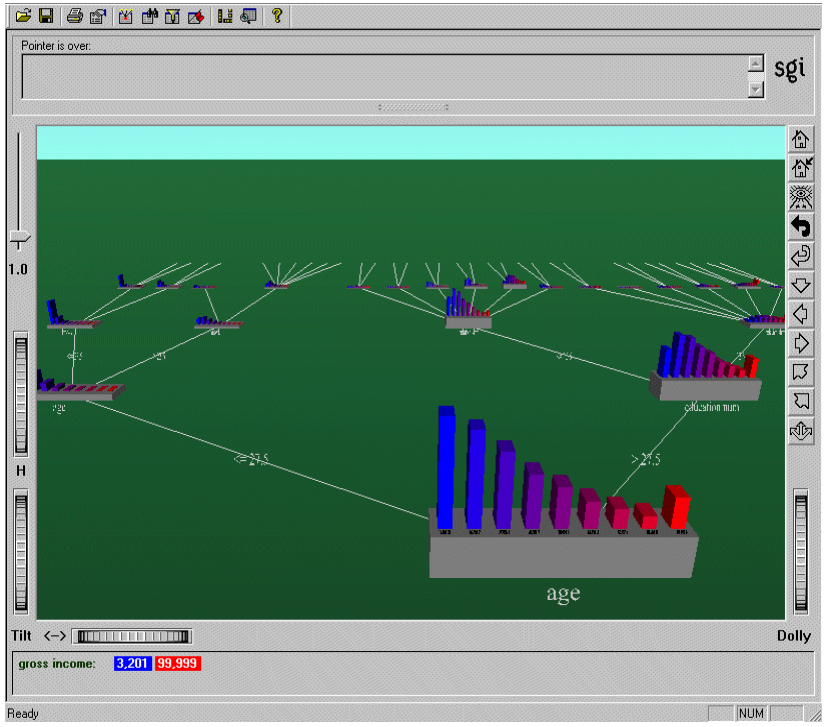


Figure 8-3 Regression Tree for the Adult Dataset

Starting the Decision, Option, and Regression Trees

The easiest way to start the tree tools is from the Tool Manager:

1. From the Tool Manager File menu, connect and log in to a server; from the same menu choose Open New Data File and select or type in the desired filename.
2. In the Data Destinations pane, click Mining Tools (Figure 8-4) and click either Classify or Regress in the lower row of tabs:
   - For Decision Tree and Option Tree, click the Classify tab.
   - For Regression Tree, click the Regress tab.

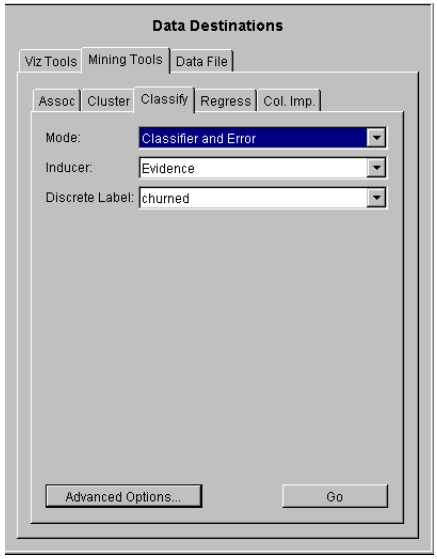


Figure 8-4 Tool Manager Data Destination Panel, Classify Tab

3. From the popup Mode menu, choose a mode. See the “Error Estimation” entry in the *MineSet Enterprise Edition Reference Guide* for more information about the four available modes.
4. From the popup Inducer menu, choose the inducer.

5. From the Discrete Label (Continuous Label for Regression Tree) popup menu, choose the attribute (column) you wish to use for the label. For instance, for the *mushroom* dataset, a logical choice is *edibility*. For the *churn* dataset, a logical choice is *churned*.
6. Click *Go* to start the inducer. The Status pane at the bottom of the Tool Manger window shows the progress and resulting statistics. Once the MineSet finishes it's calculations, the visualizer window appears.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Examining Your Results with the Decision Tree Visualizer

Decision Tree and Option Tree visualizations are similar. They both consist of two types of nodes connected by edges (the lines): decision nodes and leaf nodes.

Labels on the decision nodes specify the attribute that is tested at that node. The values (or ranges of values) against which the attributes are tested are shown at the edges. Each possible value for the attribute matches exactly one edge. For example, the root of the decision tree in Figure 8-1 tests the attribute *weightlbs*; the two edges emanating from the node specify the ranges of values for that attribute (less than or equal to 3018 and greater than 3018) so that every possible value matches either the right branch or the left branch.

Leaf nodes in a decision tree specify a class. If you follow the right branch in Figure 8-1 from the root to a leaf labeled (... 16.5], you see that Decision Tree classifies records with *weightlbs* greater than 3018 pounds, *horsepower* greater than 141, and *cubicinches* greater than 311.5, as getting 16.5 miles per gallon or less.

The color of the base indicates the error estimate of the subtree: indigo shows high error, grey indicates medium, white indicates low error. If no test set records reached a node (which means there is no error estimate), the color of the base is black.

The vertical bars atop each node show the distribution of the classes at the node. The base of each node has a height and a color. The height corresponds to the weight of the training set records that have reached this node (this is the number of records if weight was not set). In general, the higher the weight, the more reliable the class distribution at every node (see "Weighting Records" in Chapter 3 for more details about weighting records).

Placing your mouse arrow over a node displays the following information:

- *Subtree weight*—The weight of the training set records in the subtree below the node pointed to. This value is mapped to the height of the base.
- *Test set error/loss*—An estimate of the subtree error (or loss if a loss matrix was given). The number after the +/- is the standard deviation of the estimate. The higher the standard deviation, the less accurate the error estimate. The error/loss estimate and the standard deviation are less reliable for leaves with few records or when the test set error is close to 0% or 100%.
- *Test set weight*—The weight of records from the test set that reached the node (number of records if weight was not set).
- *Purity*—A number from 0 to 100 indicating the skewness (see the *Skewness* entry in the glossary) of the label value distribution at the node. If a node has records from a single class, the purity is 100. If the label values have the same weight, the purity is 0. The purity is computed after backfitting (see the *Backfitting* entry in the glossary).

Only Classify and Error mode yields the test set error/loss and weight. You can use the Test Classifier option to generate a visualization based on an existing classifier and a test set.

Classifying a Record Using the Decision Tree Main Window

To classify a record with an unknown label, start at the root of the tree, and follow the branches dictated by the attribute values for that record. By following the appropriate edges based on the record's attribute values, you reach a leaf node. The label, or class, associated with the leaf node is the predicted classification of the record.

Some decisions are made quickly and take a shorter path. Other decisions can take a longer path. In general, every leaf corresponds to a rule that is the conjunction of all tests at the decision nodes and all the values (or ranges of values) on the edges leading to it from the root.

The decision tree produced for the *iris* dataset (Figure 8-5) is very simple, and works well for illustration purposes. This file can be opened from the Tool Manager Visual Tools menu; select the 3D Visualizer, then *\examples\ iris-dt.treeviz*. At the root of this tree, the error rate is 6%, with a standard deviation of 3.39%. The standard deviation is high because the file is small, and the test set has only 50 records. The purity is 0.0, indicating that the distribution is uniform.

The left child of the root has 0 test set error and a purity of 100 because all records with petal_length less than or equal to 2.6 inches are of the iris-setosa class; thus, the prediction of iris-setosa is likely to be very accurate for all records with petal_length less than or equal to 2.6 inches. The right child of the root has an estimated error of 8.57%. In this child, which matches records whose petal lengths are greater than 2.6 inches, there are no records belonging to the iris-setosa class; thus, the class is more likely to be iris-versicolor or iris-virginica. Because only two possibilities exist at this node, there is a higher purity than at the root (36.91).

The Decision Tree leaf nodes segment the data into clusters sharing the same classification rule (path that leads to each leaf). By looking at the leaf nodes, it is possible to see clusters that share the same set of properties.

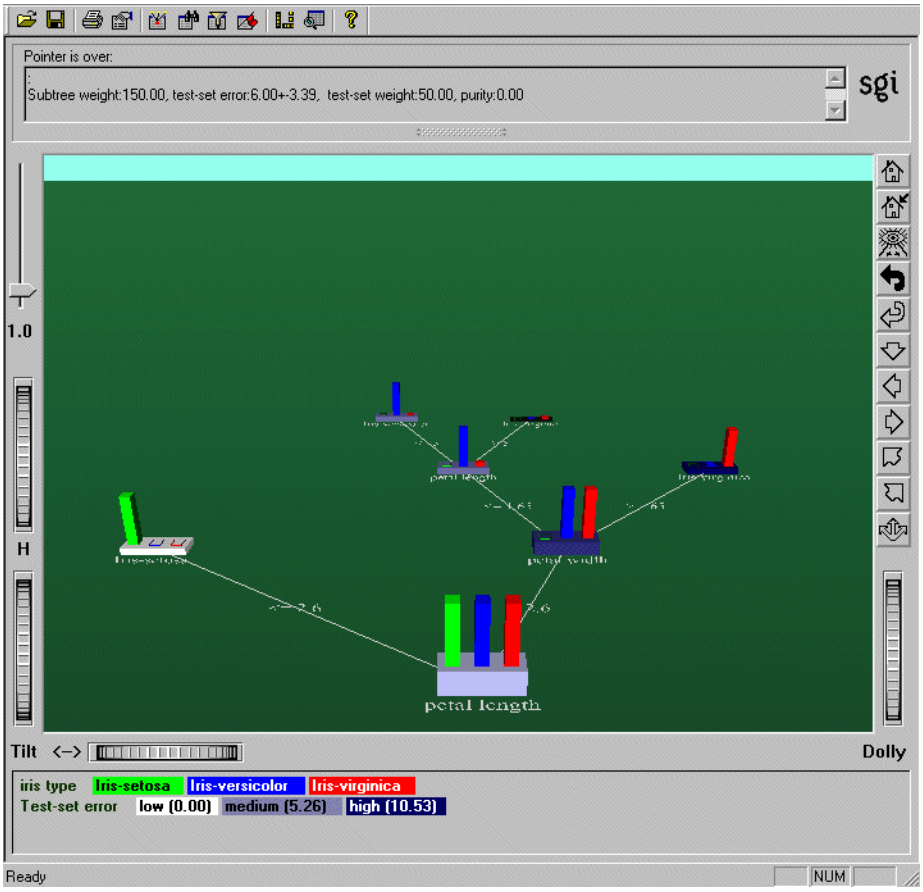


Figure 8-5 Decision Tree for Iris Dataset

Other Useful Options

The Decision Tree Visualizer uses the Tree Visualizer for its display. The Tree Visualizer features several useful devices, such as a search panel, a filter panel, and an overview window, accessible from the View menu of the Tool Manager. These are described in Chapter 5, “Examining Data with the Tree Visualizer,” and the “Tree Visualizer” entry in the *MineSet Enterprise Edition Reference Guide*.

Decision Tree also has various options available to fine-tune the induction algorithms, pruning, and evaluation metrics. These are covered in the "Decision Tree" entry in the *MineSet Enterprise Edition Reference Guide*.

Examining Your Results with the Option Tree Visualizer

The Decision and Option Tree visualizations are fundamentally similar. However, there are a few differences between the two types of tree:

- The leftmost option in an option tree would have been the only option chosen by the Decision Tree inducer. As you go right, the options are ranked in decreasing order by the fitness scoring. The fitness scores do not necessarily match the test-set error shown. This is expected, as the inducer is using a non-perfect scoring function. The test-set estimate also has natural variability: the larger the test-set, the more accurate the estimate.
- The option node can have a different error rate from that of every one of its children. Because the option node averages the children's predictions, its error rate can be different. In some cases, its error is lower than that of every child, showing that averaging helps.
- The distribution of instances (shown in bars) at every child of an option node is exactly the same as that of the option node itself. This is because there was no decision made by the option node: options are being presented as children.
- One feature of the Option Tree visualization that is very useful for navigating through the tree is clicking the right mouse button on an option node. This presents the list of children, which are the options.

A simple Option Tree worth examining can be found in *MineSet 3.1\examples\cars-odt.treeviz*. The Option Tree Visualizer uses the Tree Visualizer for its display. The Tree Visualizer features several useful devices, such as a search panel, a filter panel, and an overview window. These are described in Chapter 5, "Examining Data with the Tree Visualizer."

Option Tree also has various options available to fine-tune the induction algorithms, pruning, and evaluation metrics. These are covered in the "Option Tree" entry in the *MineSet Enterprise Edition Reference Guide*.

Examining Your Results with the Regression Tree Visualizer

Regression Tree and Decision Tree visualizations are similar. They both consist of two types of nodes connected by edges: decision nodes and leaf nodes.

Decision nodes specify the attribute that is tested at the node. Values (or ranges of values) against which the attributes are tested are shown at the edges. Each possible value for the attribute matches exactly one edge. For example, the root of the regression tree in Figure 8-3 tests the attribute age; the two edges emanating from the node partition values for that attribute (less than 27.5 and greater than or equal to 27.5) so that every possible value matches either the right branch or the left branch. If the value is unknown and there is no edge labeled with a question mark, the mean or median label value at the current node is predicted.

Leaf nodes in a regression tree predict a value. A useful example can be found in *\examples\adult-rt.treeviz*, discussed here. If you follow the leftmost branch in Figure 8-3 from the root to the leaf labeled 4002.300, you find that the Regression Tree predicts that people under 19.5 years old who work less than 35.5 hours per week average a gross income of $4002.30.

Each bar on a node in a Regression Tree corresponds to a subrange of continuous label values. The range of continuous label values covered by each node can be different. The bars at each node form a histogram indicating how the weight (number) of records is distributed over this range. The number of bars is determined by the weight of records at the root. The leftmost bar always corresponds to the lowest value. The size and midpoint, therefore, may be different at every node. The color of a bar indicates the midpoint of the subrange that bar covers. The maximum range is indicated by blue on the left and red on the right. A node that covers only records with a limited range of label values has a histogram that does not range from blue to red.

The base of each node has a height. The height corresponds to the weight of the training set records that have reached this node (this is the number of records if weight was not set). In general, the higher the weight, the more reliable the distribution at a node.

Placing your mouse arrow over a node displays the following information (see the glossary for definitions of the terms):

- *Subtree weight*—The weight of the training set records in the subtree below the node pointed to. This value is mapped to the height of the base.
- *Mean*: the mean of the continuous label.

- *Standard deviation*: the standard deviation of the continuous label. The higher the standard deviation, the less reliable the model.
- *Median*: the median of continuous label.
- *Absolute deviation*: absolute deviation of the continuous label. The higher the absolute deviation, the less reliable the model.

Predicting Values Using the Regression Tree Main Window

To predict a value for a record, start at the root of the tree, and follow the branches dictated by the attribute values for that record. By following the appropriate edges based on the record's attribute values, you reach a leaf node. The prediction associated with the leaf node is the predicted value of the record.

Error Estimation in Regression Tree

When evaluating a classifier, the natural metric is error (the number of examples for which the classifier predicts the wrong label). When a loss matrix is supplied, different types of misclassification errors may have different associated costs. In this situation loss is a natural measure. For more information, see "Defining a Loss Matrix" in Chapter 11.

For regression, where the task is to predict a real value, there is no single natural evaluation metric. The two measures that are frequently used are *standard deviation* and *absolute deviation*. The standard deviation is the square root of the mean squared error. The absolute deviation is the mean of the absolute value of the difference between the predicted label value and the actual label value.

Other Useful Options

The Regression Tree Visualizer utilizes the Tree Visualizer for its display. The Tree Visualizer features several useful devices, such as a search panel, a filter panel, and an overview window. These are described in Chapter 5, "Examining Data with the Tree Visualizer,"and the "Tree Visualizer" entry in the *MineSet Enterprise Edition Reference Guide*.

The Regression Tree also has various options available to fine-tune the induction algorithms, pruning, and evaluation metrics. These are covered in the "Regression Trees" entry in the *MineSet Enterprise Edition Reference Guide*.

Predicting with Decision, Option, and Regression Trees

Up to this point, this chapter has discussed classifying data with known outcomes (for instance, churned or not churned and edible or poisonous), but Decision, Option, and Regression Trees can be most useful for predicting unknown or future outcomes. You can build a model based on data where the class is known, and then use that model to classify new data where the class is unknown. See Chapter 7, "Understanding Predictive Modeling," and Chapter 11, "Refining Predictive Modeling," for more information about building and applying models.

Modeling and Predicting with the Decision Table Classifier and Visualizer

This chapter discuss the features and capabilities of Decision Table, and contains the following sections:

Sample configuration and data files have been provided to demonstrate Decision Table features and capabilities. These files are in the *\examples* directory under *MineSet*, in the location where MineSet was initially installed.

Overview of the Decision Table Classifier

A decision table is a predictive modeling tool that performs classification (see Chapter 1, "Overview of Data Mining and MineSet Tools," and Chapter 7, "Understanding Predictive Modeling," for more information on classifiers and predictive modeling). It incorporates an inducer (an algorithm for generating decision table models), and a visualizer. Unlike the evidence model, the Decision Table model does not assume that the attributes are independent.

A decision table is a hierarchical breakdown of the data, with two attributes at each level of the hierarchy. The Decision Table inducer identifies the most important attributes (columns) for classifying the data, and the accompanying visualizer displays the resulting model graphically as a series of cake charts. Each cake in the visualization can in turn be divided into smaller cakes representing the next pair of most important attributes. Each visualization can contain several levels representing decreasingly important attributes. Figure 9-1 shows the top level of the Decision Table visualization for the *mushroom* dataset, where the two most important attributes for determining edibility are odor and spore print color.

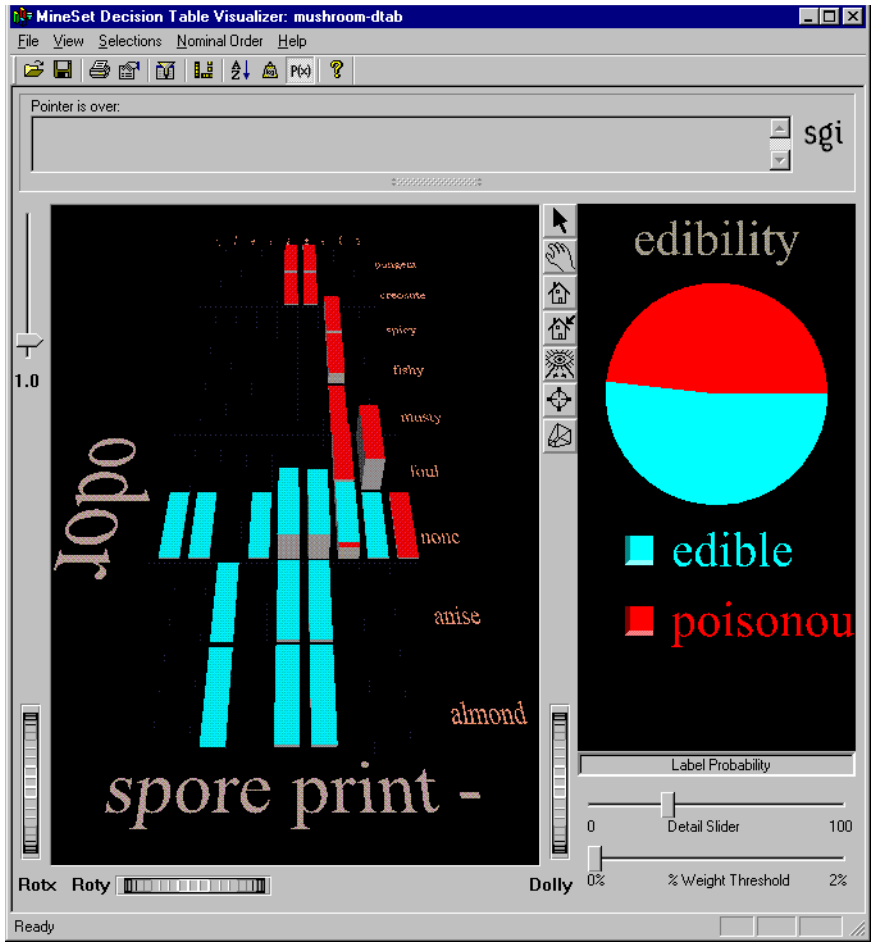


Figure 9-1 Decision Table for the Mushroom Dataset

Starting the Decision Table

The easiest way to create a Decision Table classifier is from the Tool Manager (see the *MineSet Enterprise Edition Reference Guide* for more options):

1. From the Tool Manager File menu, connect and log in to a server;
2. From the same menu choose Open New Data File and select or type in the desired filename.
3. In the Data Destinations pane, click the Mining Tools tab (Figure 9-2), and choose Classify from the lower row of tabs.

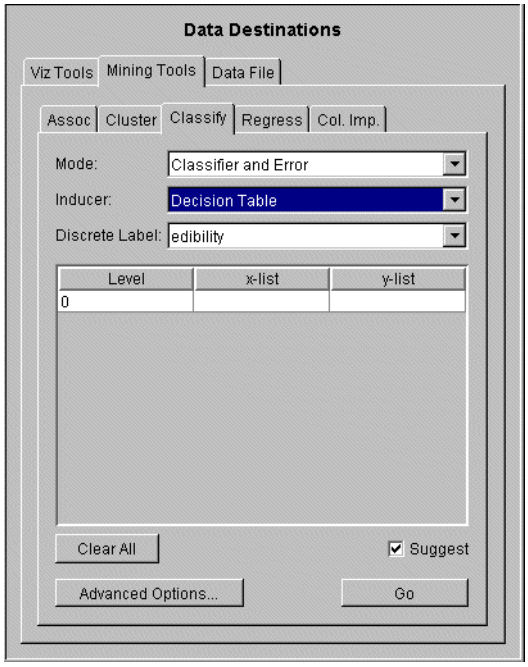


Figure 9-2 Data Destination Panel in Tool Manager Showing Classifiers

4. From the popup Mode menu, choose a mode. See Chapter 7, “Understanding Predictive Modeling,” for more information about the four modes.
5. From the popup Inducer menu, choose Decision Table Inducer.

6. From the popup Discrete Label menu, choose the attribute you wish to use for the label.
7. In the x-list and y-list pulldown menus (see Figure 9-2), specify the attributes you want to view. You can do this in three different ways:
   - You can let MineSet select the attributes automatically by checking the Suggest box. MineSet identifies the most useful attributes for predicting the class, and displays them.
   - You can manually select the attributes from the pulldown menus in the Data Destination pane. If you want a place to start, use Column Importance to find the most important attributes (see "Finding Important Columns" in Chapter 3).
   - You can select some of the attributes and let MineSet choose the rest. Select the attributes as described above, and check the Suggest box. MineSet maps the rest of the relevant attributes.
8. Click *Go* to start the Inducer.

   The Status pane at the bottom of the Tool Manager window shows the progress and resulting statistics. You can also interrupt the automatic attribute suggestion process by clicking *Cancel* or *Show Viz Now* in the progress dialog. Clicking *Show Viz Now* stops the current server computation and uses the intermediate result to construct the decision table for the columns mapped so far.

Note: See the "Decision Table" entry in the *MineSet Enterprise Edition Reference Guide* for information about other Decision Table inducer options.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Examining Your Results with the Decision Table Visualizer

The Decision Table visualizer has two panes, the Decision Table pane on the left and the Label Probability pane on the right. Figure 9-3 shows a Decision Table visualization of the *mushroom* dataset with the first two levels of detail revealed.

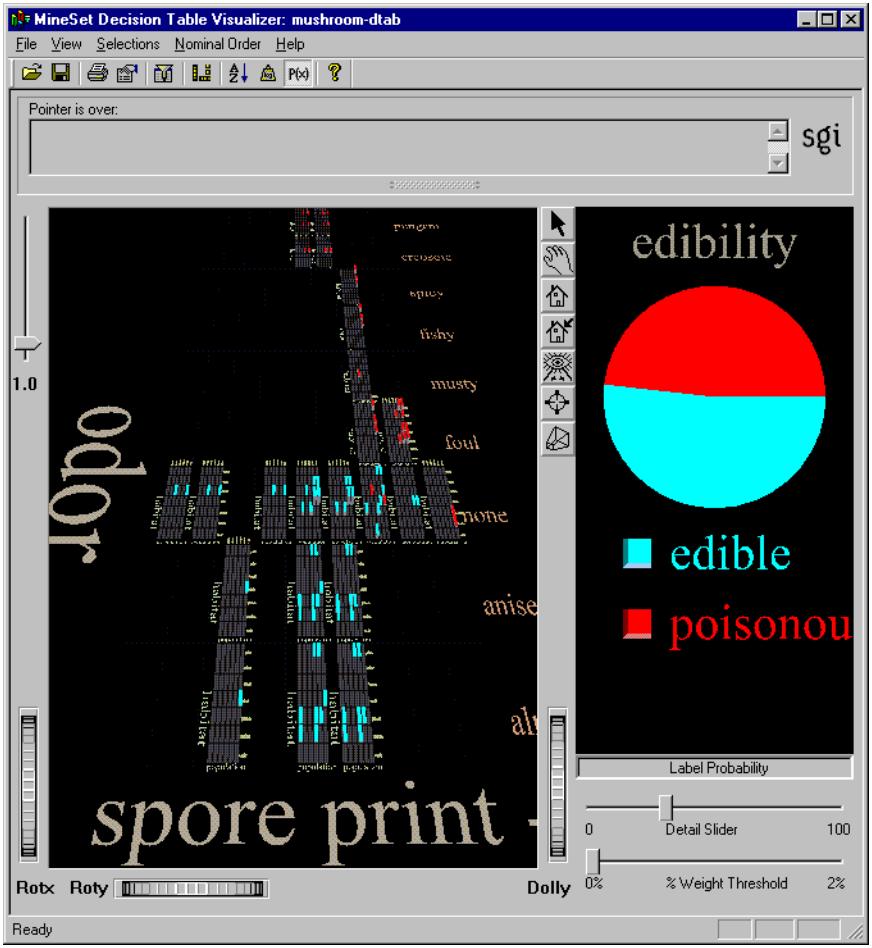


Figure 9-3 Decision Table Visualization of the Mushroom Dataset

Examining the Decision Table Pane

The Decision Table pane on the left consists of cake charts which are square charts with colored slices representing the label probabilities for records with certain attribute values. The label probabilities represent the likelihood that a record with those values for the specified attributes will be in a certain class. For instance, in Figure 9-3, the Decision Table was run on the *mushroom* dataset. The resulting chart shows that the probability that a mushroom with a white spore print color and a fishy odor is poisonous is 100%. If the mushroom has a white spore print color, and no odor, however, the probability of it being poisonous is only 7.69%, and the probability of it being edible is 92.31%. To see these percentage values, select a cake on the left then place your mouse arrow over the colored box next to the label (edible or poisonous). The percentages are displayed in the area between the menu bar and the main window.

The elements in the Decision Table pane can be further subdivided into smaller and smaller cake charts by clicking with the right mouse button, in a process called drill-down. To examine the cake charts more closely:

- To see the values of the two attributes at the current level of detail, place the mouse arrow (in select mode) over the desired cake chart. The attribute values and the weight of records represented are displayed between the menu bar and the main window. The height of the cake chart is proportional to the weight (see the *Weight* entry in the glossary).
- To drill down to the next level of detail, place the mouse arrow over the desired cake chart and click the right mouse button, or click the background to drill down globally on all cakes. Figure 9-5 shows a close-up view of a subdivided cake for the *mushroom* dataset.
- To drill back up through the levels, hold the Ctrl key while clicking the right mouse button (or use the middle mouse button). You can drill up for a single region, or globally by holding the Ctrl key while clicking the right mouse button (or use the middle mouse button) on the background.
- To see the values for the two attributes for the cake one level higher in the hierarchy, as well as the weight of records, place the mouse arrow over the base (the gray block under the cake).
- To see the values that define a particular cake chart, start by highlighting the base at the coarsest level of detail, and continue through the next most detailed base until you reach the base immediately below the cake chart of interest. The relevant value pair is displayed in the selection pane below the menu bar.

At each level of detail, one attribute's name is shown to the left of the array of cake charts, and it's values are shown to the right; the other attribute's name is shown at the bottom of the array, and it's values are shown at the top. If there is an odd total number of attributes, the lowest level shows only one attribute.

The Label Probability Pane

The Label Probability pane (on the right side of the Tool Manager window) shows a pie chart of the label probabilities for the entire dataset. A list of all class labels appears under the pie chart.

If you want to examine the label probabilities more closely do the following:

- To see the label probabilities for a specific set of attributes, click the desired cake chart. The pie chart in the Label Probability pane shows the label probabilities for records with the set of attributes represented by that cake.
- To see the label probabilities for a combination of attributes, Ctrl-click the desired cake charts. By Ctrl-clicking, you can select a number of cakes at different levels of detail in the scene. The Label Probability pane at the right of the visualizer window shows the label probabilities for the selected set of records.
- To see the specific percentages and confidence levels for each label, place the mouse arrow over the colored box next to the desired label. The numbers are displayed between the menu bar and the Decision Table pane.

Decision Table Example

As an example of how to use Decision Table, say you want to figure out which mushrooms are safe to eat, and which are poisonous. To start, you might create a decision table from the *mushroom* dataset (see Figure 9-1). The left pane shows the top level of detail for the resulting decision table. Odor and spore print color were chosen for the top level because they were the top two attributes determined by the Column Importance algorithm. Only one top-level block has more than one class present, the block with odor equal to none and spore print color equal to white. When you drill down to the next level (by clicking the right mouse button over the chart), the attributes habitat and population are shown (see Figure 9-4). A close-up of this drilled-down region is shown in Figure 9-5. At this level, there are no mixed cakes. Therefore, if you find a mushroom that has a white spore print color and no odor, growing with several other mushrooms in the woods, you can be pretty sure it is poisonous.

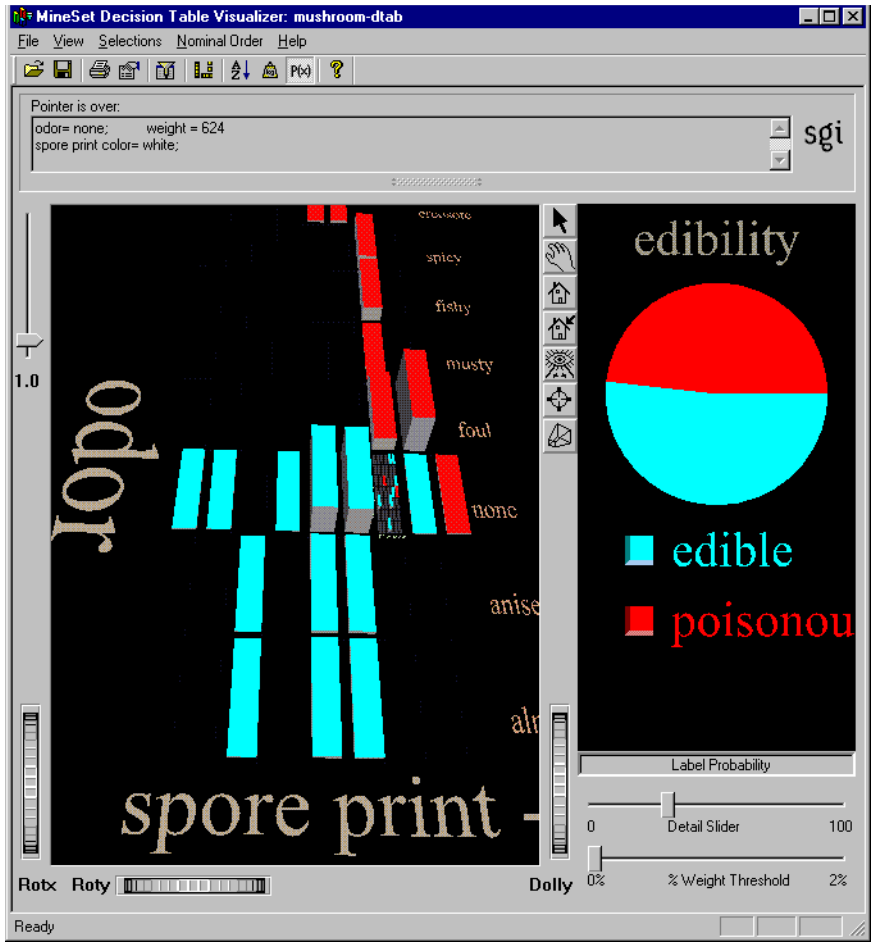


Figure 9-4 Decision Table with Drill-Down on Mushroom Dataset

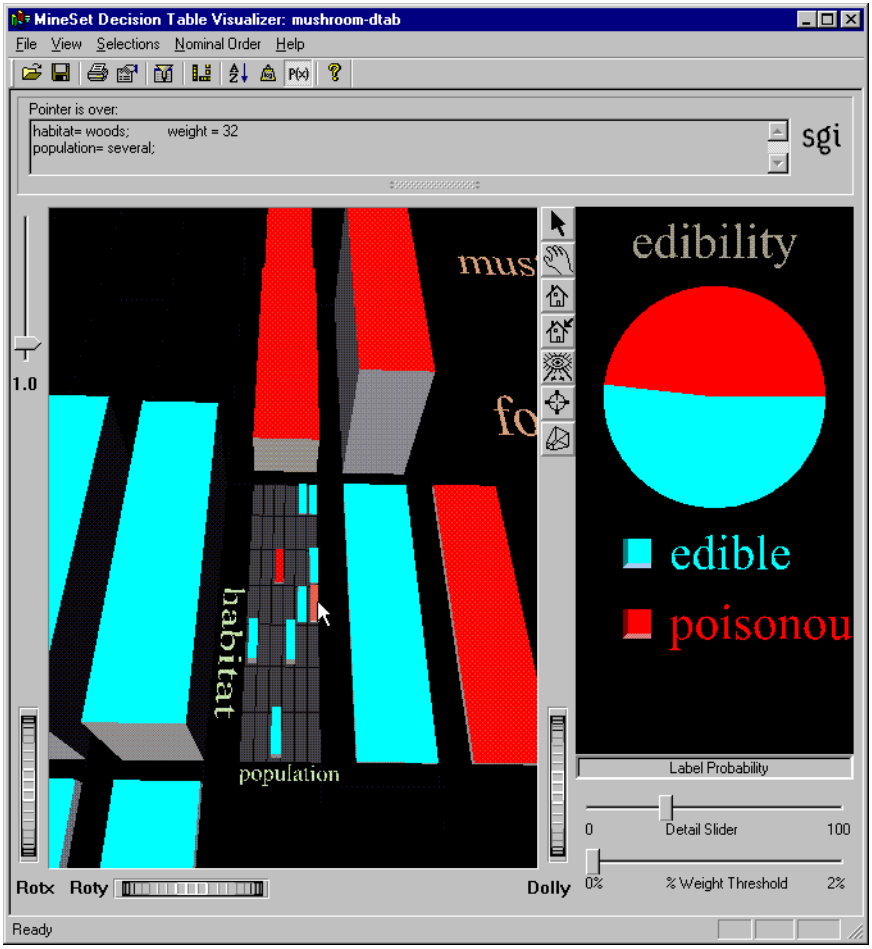


Figure 9-5 Close-up of Decision Table with Drill-Down

Predicting with Decision Table

Up to this point, this chapter has discussed classifying data with known outcomes (for instance, edible or poisonous), but Decision Table can be most useful for predicting unknown or future outcomes. You can build a model based on data where the class is known, and then use that model to classify new data where the class is unknown. See Chapter 7, "Understanding Predictive Modeling," and Chapter 11, "Refining Predictive Modeling," for more information about building and applying models.

For details of the menus, see the "Decision Table" entry in the *MineSet Enterprise Edition Reference Guide*.

Modeling and Predicting with the Evidence Classifier and Visualizer

This chapter discusses the features and capabilities of the Evidence Classifier and Visualizer, and contains the following sections:

Sample configuration and data files have been provided to demonstrate the Evidence Visualizer's features and capabilities. These files are in the *\examples* directory under *MineSet*, in the location where MineSet was initially installed.

Overview of the Evidence Classifier and Visualizer

The Evidence tool is a predictive modeling tool that performs classification (see Chapter 1, "Overview of Data Mining and MineSet Tools," and Chapter 7, "Understanding Predictive Modeling," for more information on classifiers and predictive modeling). It incorporates an inducer (an algorithm for generating Naive Bayes models) and a visualizer. Unlike the Decision Table model, the Evidence model assumes that the attributes are independent, although it still produces reasonable results even if they are not.

The Evidence Visualizer displays the information from the classifier as shown in Figure 10-1. The visualizer can help you understand the importance of specific attributes (columns) for classification.

The Evidence Visualizer Window

Once the classifier is built, the results are displayed in the Evidence Visualizer window. Initially, the left pane contains rows of cake charts for each attribute used by the classifier (Figure 10-1). A cake chart resembles a pie chart in that it shows proportions, except that it is square with rectangular slices. You can toggle between the cake charts, which represent evidence, pie charts, which represent probabilities, and bar charts, which give more information on a selected label (see "Examining Your Results with the Evidence Visualizer" on page 169 for more information).

Figure 10-1 shows the evidence view, and Figure 10-2 shows the probability view.

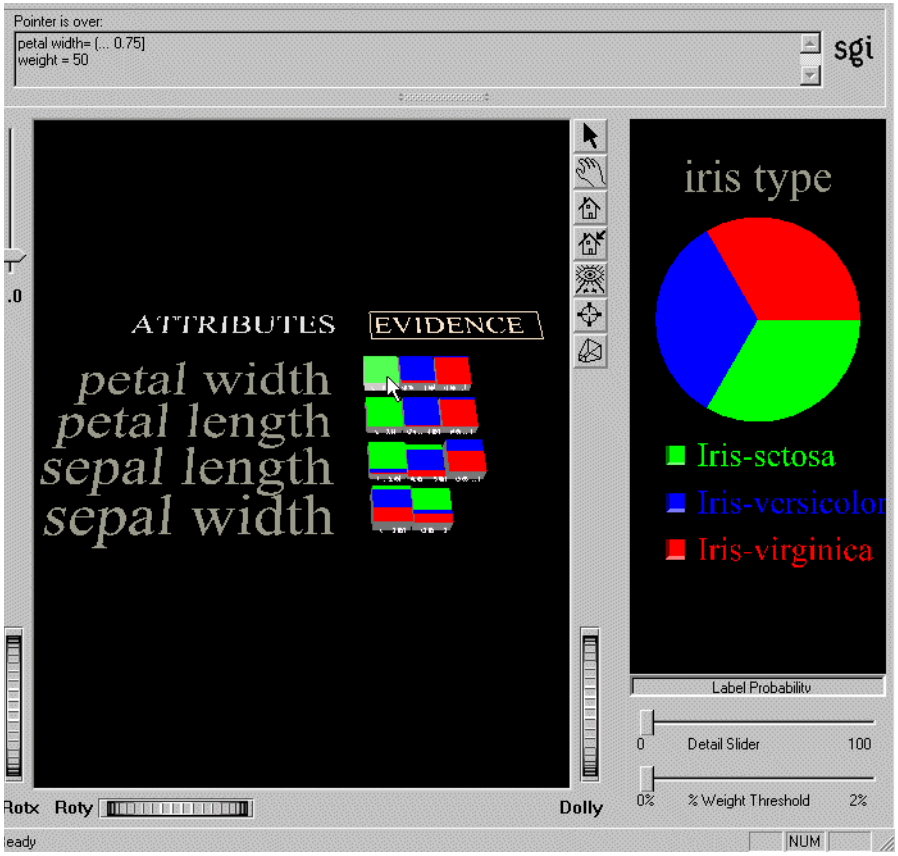


Figure 10-1 Evidence Visualizer Applied to the Iris Dataset

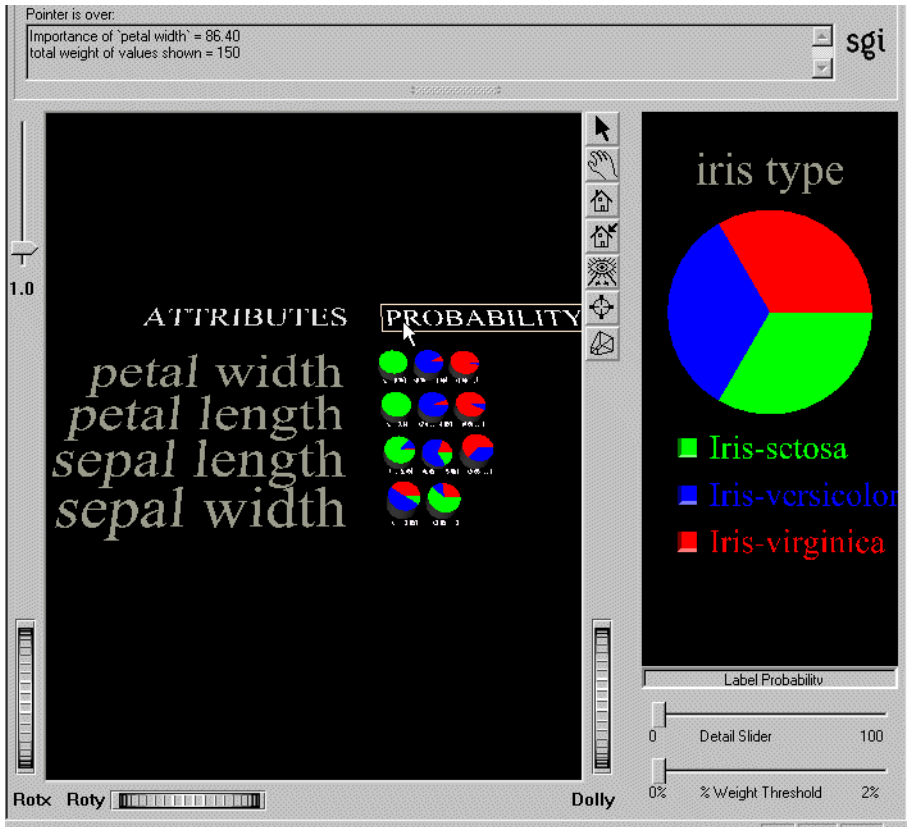


Figure 10-2 Evidence Visualizer Showing Probabilities

Starting the Evidence Tool

The easiest way to start the Evidence tool is from the Tool Manager (see the *MineSet Enterprise Edition Reference Guide* for more options):

1. From the Tool Manager File menu, connect and log in to a server.
2. From the same menu choose Open New Data File and select or type in the desired filename.
3. In the Data Destinations pane, click the Mining Tools tab (Figure 10-3) and choose Classify from the lower row of tabs.

4. From the popup Mode menu, choose a mode. See Chapter 7, "Understanding Predictive Modeling," for more information about the four modes.
5. From the popup Inducer menu, choose Evidence.
6. From the popup Discrete Label menu, choose the attribute you wish to use for the label. Edibility is a good one for the mushroom dataset.
7. Click *Go* to start the Inducer.

The Status pane at the bottom of the Tool Manager window shows the progress and resulting statistics.

Note: See the "Evidence Inducer Options" entry in the *MineSet Enterprise Edition Reference Guide* for information about other Evidence inducer options.

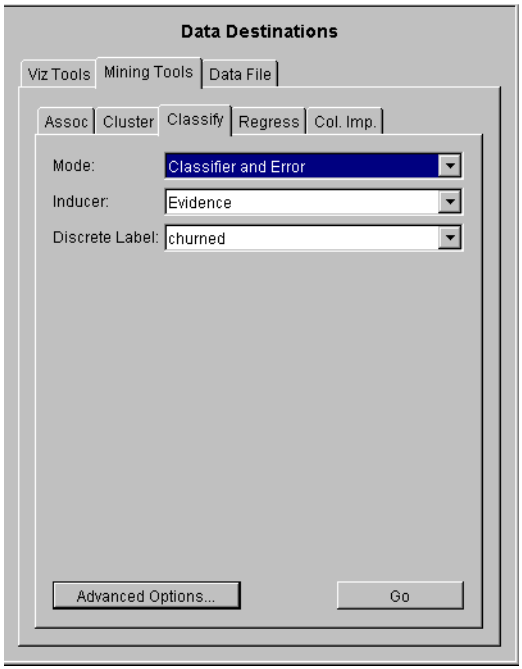


Figure 10-3 Tool Manager Data Destination Panel

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Examining Your Results with the Evidence Visualizer

Once you have started the Evidence Visualizer, you can examine your results in several ways. In the left pane of the visualizer window, you can switch between evidence view, probability view, and bar view:

- Evidence view shows cake charts, representing evidence (see "Evidence View" on page 170).
- Probability view shows pie charts, representing probabilities (see "Probability View" on page 173).
- Bar view shows bar charts (see "Bar View" on page 174), representing the evidence for and against a particular label.

The Label Probability pane (on the right side of the visualizer window) shows a pie chart representing the distribution of the label values across the entire database (see "Examining Probabilities in the Evidence View"). This pane is displayed for all three views in the left pane (evidence, probability, and bar).

Evidence View

When the Evidence Visualizer first launches, the main pane shows the evidence view (Figure 10-4).

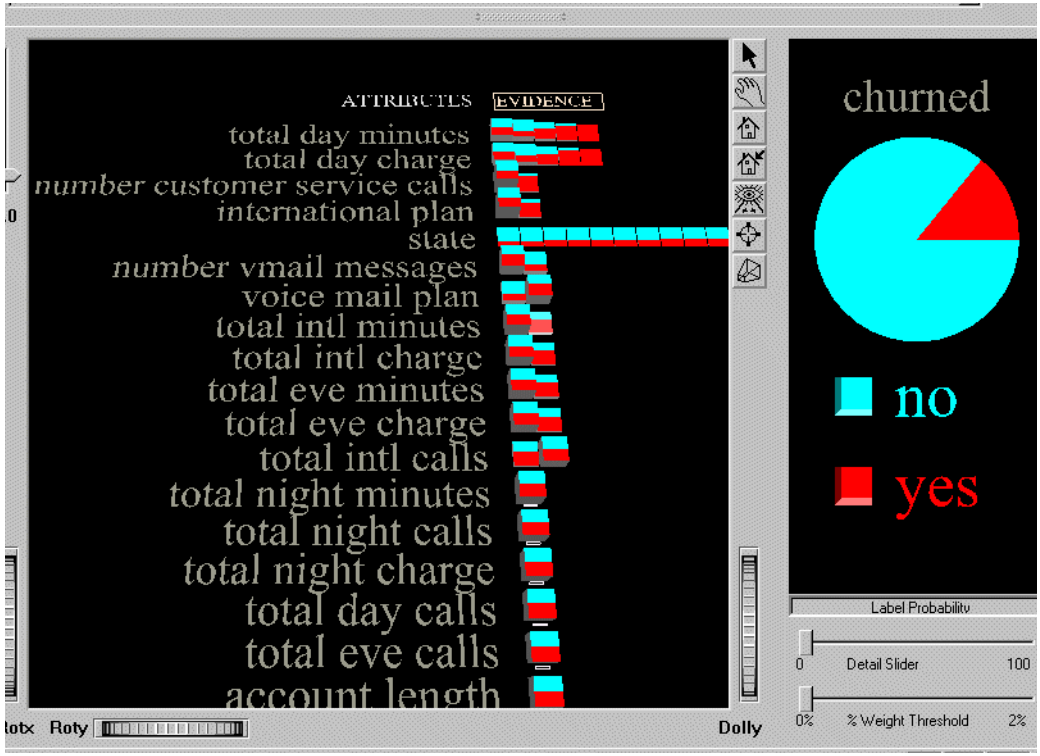


Figure 10-4 Evidence Visualizer Cake Chart for the Churn Dataset

Interpreting the Evidence View

The Evidence View shows one row of cake charts for each attribute or column in the dataset. Each row contains one cake for each discrete value of the attribute. If the values of the attributes are continuous rather than discrete, MineSet automatically breaks the values into ranges or bins (for instance 0 to 5, 5 to 10, and so on) in a way that maximizes the differences between adjacent cakes. A cake's height is proportional to the weight of records having that attribute value (see the entry for *Weight* in the glossary). If no weight attribute is set, the height represents the number of records.

In the cake charts, the size of a slice represents conditional probability (evidence), not straight probability. For instance, in the *churn* dataset, if a cake is half "yes" and half "no," that means that a customer having the particular value of the attribute represented by that cake has the same probability of churning as the overall population. In other words, that value of the attribute has no predictive value whatsoever. In contrast, if a cake is 90% "yes" and 10% "no," that means that a customer having that particular value of the attribute is considerably more likely to churn than the overall population. It does *not* mean, however, that the customer has a 90% chance of churning, because the overall probability can depend on many variables, not just this one. See the "Evidence Inducer" entry in the *MineSet Enterprise Edition Reference Guide* for a more detailed explanation.

The order of the rows indicates the relative importance of the attributes in predicting the label. Each attribute is assigned an importance value between 1 and 100. You can see this value by placing the mouse arrow over the attribute name in the scene. The information is displayed in the area between the menu bar and the visualizer pane. For instance, in the *mushroom* dataset shown in Figure 10-5, a mushroom's odor is extremely indicative of edibility, so the odor attribute is ranked at the top of the list, with an importance value of 90.69. In contrast, stalk shape is not at all indicative of edibility, so the stalk shape attribute is ranked at the bottom, with an importance value of .75.

The veil-color attribute presents an interesting case, too. It is ranked near the bottom in terms of importance because most mushrooms have white veils. Mushrooms with white veils are split almost evenly between edible and poisonous, which is very close to the prior probability (that is, the overall probability of any mushroom in the dataset being poisonous). Statistically, therefore, this attribute does not add much evidence to either class and is ranked near the bottom in importance. However, for veil colors of brown, orange, or yellow (the first, second, and fourth cakes in the veil color row), the picture is quite different. Of mushrooms with brown or orange veils, 100% are edible. Of mushrooms with yellow veils, 100% are poisonous. While this sounds persuasive, the number of mushrooms with non-white veils is so small that veil color does not make a good distinguishing characteristic.

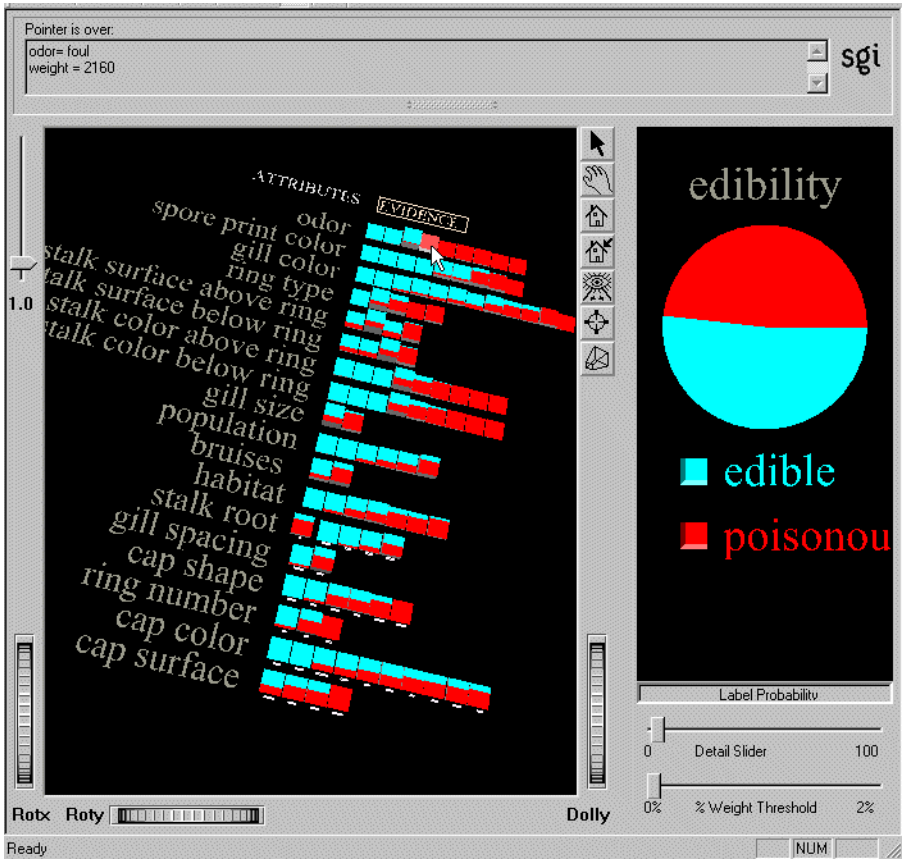


Figure 10-5 Mushroom Dataset in the Evidence Visualizer

Examining Probabilities in the Evidence View

When the Evidence Visualizer first opens, the Label Probability pane shows the label probabilities for the entire dataset. For instance, in Figure 10-4, using the *churn* dataset, the probability of a random record having a churned label of "yes" is 14.14%. If you place the mouse arrow over the colored box next to one of the labels, the probability and weight of that label are displayed in the upper left of the window.

If you wish to see the probability pie chart for records with a particular value for an attribute, click the cake for that value (the mouse must be in select mode to select attributes). The Probability Pane now shows the probability pie chart for just those records matching the selected value. To deselect the chart, click anywhere on the background of the main pane.

To see the probabilities for records with a certain combination of attributes, you can select several cakes by Ctrl-clicking them. The Probability Pane now shows the probability pie chart for the records with that combination of attributes. Because the probabilities for each attribute are computed as if that attribute were independent of the others, the combined probabilities may not be entirely accurate. A warning appears reminding you of this. Use them as an estimate. They are only completely accurate if the attributes are truly independent, which is rarely the case.

Sometimes while selecting a combination of attributes, you might select some that are mutually exclusive. In this case, the Label Probability pane shows a gray pie. This indicates that there are few or no records that have that particular combination of values.

To reset the Label Probability pane to show the prior probability, click anywhere in the background of the main pane.

Probability View

Probability view (Figure 10-6) shows pie charts, representing probabilities, instead of cake charts. To access the probability view from the evidence view, click the word Evidence in the white-outlined box at the top of the chart, or toggle the “Show as evidence” check box in the visualizer View menu. To return to evidence view, click the word Probability or toggle the check box again.

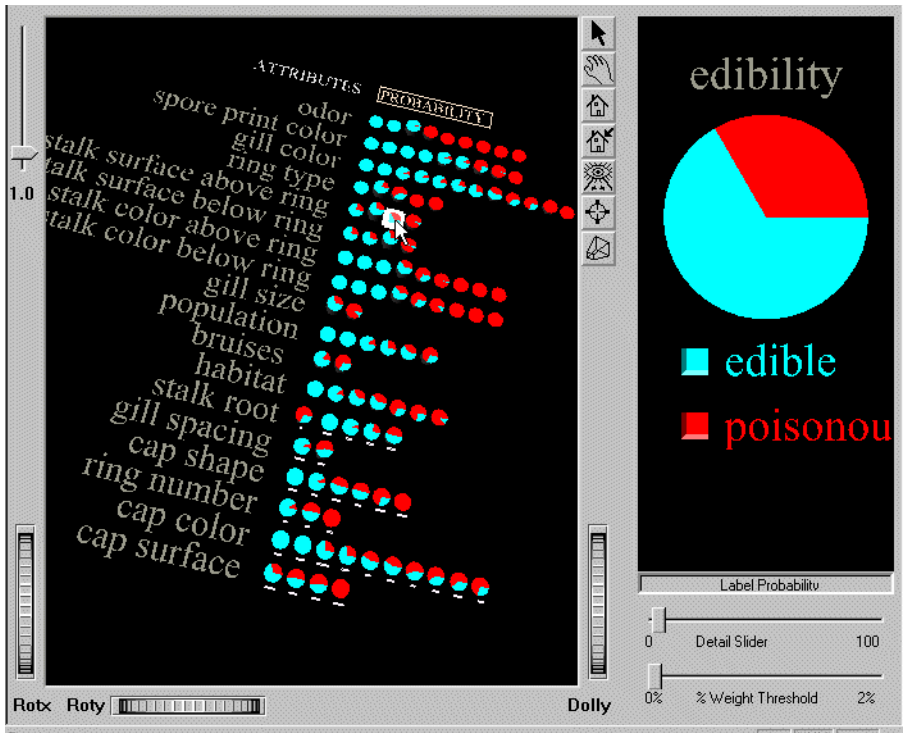


Figure 10-6 Evidence Visualizer Pie Charts

Bar View

Bar view shows the evidence in favor of (or against) a selected label value. To access the bar view, click the colored box next to the desired label in the Label Probability pane. The main window then displays bar charts (Figure 10-7). The height of each bar shows the evidence in favor of the selected label value. To exit bar view, click the colored box again.

By clicking the word “For” in the white-outlined box at the top of the bar view chart, you can switch to “evidence against” mode. To return to “evidence for” click the word “Against.”

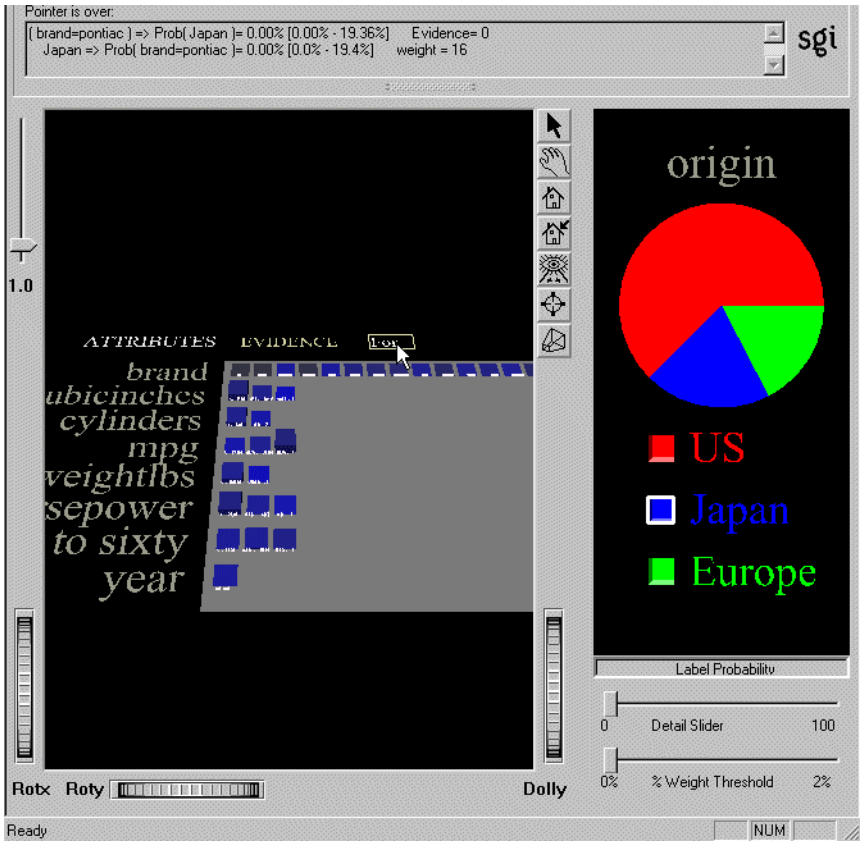


Figure 10-7 Label Value “Japan” Selected Using the Cars Dataset

“Evidence for” can be useful in determining which values are the most helpful in predicting a particular label value. “Evidence against” can be useful in determining the opposite, that is, which values are the most helpful in predicting when a label will *not* occur. For a more technical description of how these values are computed, see the “Evidence Inducer” entry in the *MineSet Enterprise Edition Reference Guide*.

The color of the bars can range from a fully saturated color to completely gray. The grayness of the bars is based on the 95% confidence interval. This, in turn, depends on the weight for that value. Therefore, bars that are nearly gray have low weight and a large confidence interval. The height of these bars is not likely to be accurate. Conversely, the height and corresponding evidence value for a fully saturated bar can be relied on because it is based on large weight, representing many records. The exact number of records (*weight*) is reflected in the text output line when that bar is highlighted.

Changing the Evidence Visualizer Views

Table 10-1 shows various actions you can take to better examine your results in the different views.

Table 10-1 Working with the Evidence Visualizer Views

To:	Do:
Switch between cake charts (evidence) and pie charts (probability)	Click the white-outlined word "Evidence" or "Probability" or toggle View > Show As Evidence.
Switch between cake or pie charts and bar charts	Click the colored square next to the desired label in the Label Probability pane.
Switch between bar charts and cake or pie charts	Click the colored button next to the desired label in the Label Probability pane.
Switch between "For" and "Against" in bar view	Click the white-outlined box around the word "For" or "Against."
Display attribute value and record weight for a particular cake or pie	Put the mouse arrow over the cake or pie.
Display the label probability	Put the mouse arrow over the box next to the label in the Label Probability pane.
Display (in the right pane) the expected distribution of labels for the new records having a certain attribute value	Click the cake for that value.
Display attribute importance and total weight	Put the mouse arrow over the attribute name on the left.
Get an estimated label probability distribution for records with a certain combination of attribute values	Select two or more cakes by Ctrl-clicking them and examine the pie chart in the Label Probability pane.

Table 10-1 (continued) Working with the Evidence Visualizer Views

To:	Do:
Clear a selection	Click anywhere in the background area of the left pane.
See all the records with the selected attribute values	Click on the cakes representing the desired combination of attribute values and choose Selection > Drill Through > Show original data in recordviewer.
See all the records *except* those that have the selected attribute values	Choose Selection > Drill Through > Complement drill through.
Exaggerate or diminish the height differences between the charts	Use the height slider on the left side of the main window.
Reduce (or increase) the number of attribute rows shown, starting with the least (or most) predictive attributes	Use the detail slider. Removes attributes with importance less than the slider value.
Filter out attribute values with low counts or weights	Use the weight threshold slider. Removes values whose weight is less than the percentage indicated by the slider.
Predict a class label for a new record for which you know some of the attribute values	Click the charts corresponding to those attribute values. The largest slice in the pie chart on the right corresponds to the predicted class.

Predicting with the Evidence Classifier

So far, discussion has focused on using the Evidence Classifier and Visualizer to examine data where the label is already known. In actuality, this tool can be most useful in predicting the label in data where it is not known. In this case, you create a model using known data (see the "Training Set" entry in the *MineSet Enterprise Edition Reference Guide*), save it, and then apply it to data with unknown labels. See the "Apply Model" entry in the *MineSet Enterprise Edition Reference Guide*.

For details about the visualizer menus, see the "Evidence Visualizer Menus" entry in the *MineSet Enterprise Edition Reference Guide*.

Refining Predictive Modeling

This chapter builds on concepts developed in previous chapters, using the Tool Manager to examine the cost of making errors in classification, and to fit data mining operations to real-life needs. MineSet's advanced options for inducers allow you to improve accuracy, to compensate for the cost of making mistakes, and to determine the stage at which it is pointless to continue investing in a project. The sections in this chapter are:

The screen illustrations may be viewed in color using the online version of this guide (available from the Help menu).

Ensuring the Accuracy of Models

The model you have previously developed can be checked for accuracy in different ways. A few methods have been discussed in "Evaluating a Predictive Model" on page 128. This section continues the discussion with methods of verifying the model using different sources. If you do not presently have a model, return to Chapter 7, "Understanding Predictive Modeling," for help in creating one.

Testing the Model

If you have a model created at an earlier time, MineSet allows you test it on data you are currently working on, using the Test Model facility. The columns in the dataset must contain the same names and types as those used when creating the model.

Unlike Apply Model (described in "Applying a Predictive Model" in Chapter 7) Test Model requires that the table also contain a label column with the same name and type as that used when building the model. The reason for this is that the test must work from data for which you have the correct answer. The test file is assigned a default name with the suffix *-test*, such as *iris-test*.

The following example creates a test file from the *iris* dataset. Once you have logged in to the server, select your dataset (the example file is *MineSet\data\iris.schema*, in the directory where MineSet was originally installed).

To test a model on the current dataset, follow these steps:

1. In the Tool Manager Data Transformations pane, click the *Apply Model* button.
2. From the Test and Apply Model dialog box, click the Test Model tab.
3. Select the name of the model from the list of available models that appears at the top of the dialog box (for example *iris-dt.class*) and the default test filename appears (Figure 11-1).

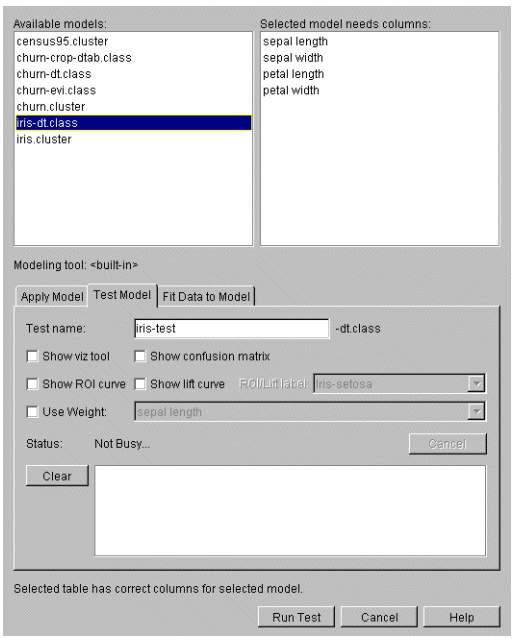


Figure 11-1 Test Model Panel

If your current table is missing any columns, those column names are highlighted in the "Selected model needs columns" list on the right. Choose another model or restore the missing columns. The Test Model panel gives you these options:

Table 11-1 Test Model Panel Options

Selection	Action
Test name	Shows the name of the test file, which you can change.
Show viz tool	Shows a visualization of the classifier with the table used as the test set.
Show confusion matrix	Shows confusion matrix of model based on table records. See "Displaying a Confusion Matrix" on page 186.
Show ROI curve	Shows ROI curve of model using the specified label value. See "Finding Marketing Benefits Using a Return on Investment Curve" on page 193.
Show lift curve	Shows lift curve of model using the specified label value. See also "Assessing Models with the Lift Curve and ROI Curve" on page 191.
ROI/Lift label	Displays popup menu of available label values.
Use weight	Select an available attribute from the popup menu by which to weight the records. See also "Weighting Records" in Chapter 3.

4. Click *Run Test.* The text field at the bottom of the Test Model panel shows the results, and the visualizer options you selected are displayed.

In this case, the entire dataset of *iris.schema* is run on a previously created model, *iris-dt.class*, and four errors are noted on 150 test instances.

Fitting Data to a Model

When you have fresh data, use the Fit Data to Model panel (Figure 11-2) to fit the data in the current table to a previously created model. This produces a new model with the same structure as the original one; however, the new one uses the data from the table to update the probability estimates (see "Backfitting in Error Estimation"). Because all of the data from the table is being fit into the structure of the model, there is no error estimation. Fit Data to Model cannot be used on models that were built using boosting. See "Improving Accuracy with Boosting" below. If you want to evaluate the performance of the new model on a separate test set (disjoint from the fit data), use the Test Model facility.

To fit data to an existing model, follow these steps:

1. In the Tool Manager Data Transformations pane, click the *Apply Model* button.
2. From the Test and Apply Model dialog box click the Fit Data to Model tab. (Figure 11-2). Fit Data to Model options appear in Table 11-2.

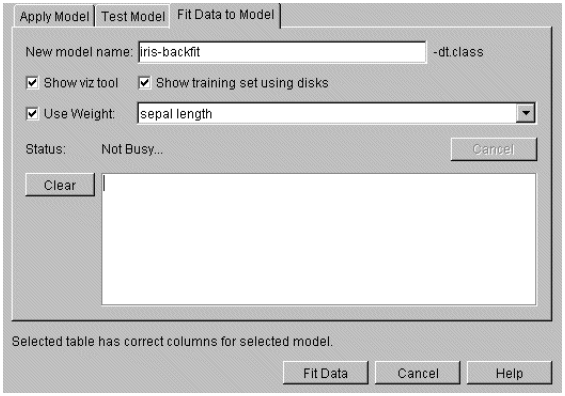


Figure 11-2 Fit Data to Model Panel

The Fit Data to Model panel (Figure 11-2) gives you these options:

Table 11-2 Fit Data to Model Options

Selection	Action
Show viz tool	Shows a visualization of the new model (only for tree-based models).
Show training set using disks	Sets disks to indicate proportion of training set data in visualization.
Use Weight	Select an attribute from the popup menu to use as record weight.

Backfitting in Error Estimation

When you backfit a model with a set of records, you update the probability estimates, you do not alter the structure of the model. Backfitting is similar to fitting data to a model, a mode that defaults to using backfitting. Backfitting is useful when:

- You build a model structure from a small training set and backfit it with a large dataset. This is often quicker than inducing the larger model structure.
- You need greater accuracy. When counts, weights, and probabilities are shown in the model's structure, they reflect all the data, not just the training set portion.

To enable backfitting:

1. In the Data Destinations pane of the Tool Manager window, click the Mining Tools tab.
2. Click the Classify tab.
3. Choose the Classifier and Error mode, and click the *Advanced Options* button to bring up the Advanced Options dialog. Click the Backfit test set check box under Error Estimation Options. The check mark is disabled when Boosting is enabled.

Improving Accuracy with Boosting

In some cases error rate is the most important criterion in choosing a model, so you might want to enable boosting. Boosting is an algorithm that creates several different models and combines their predictions using a weighted voting scheme. Boosting improves model accuracy by focusing the induction process on examples in the data that are harder to model than others. However, boosted models cannot be visualized.

To activate boosting on any selected model:

1. In the Tool Manager Data Destinations pane, click Mining Tools.
2. Click the Classify tab.
3. From any mode click the *Advanced Options* button to bring up the Advanced Options dialog. Click the Boost (no viz) check box in the Inducer options section.

Boosting does not always increase accuracy, but it often does. Boosted models cannot be visualized, though you can still see confusion matrices, lift curves, learning curves and ROI curves for boosted models. Boosting is a computationally intensive process, often taking 25 times longer to run than the corresponding inducer without boosting. Boosted models may not use backfitting.

Note: Although the theory behind boosting has not been generalized past two-class problems, MineSet allows you to use boosting with labels that have any number of values. Boosting is less likely to improve the error rate in these cases.

Boosting works by repeatedly assigning new weight distributions to the training set and inducing models on the reweighted sets. See the *MineSet Enterprise Edition Reference Guide* for further information on boosting.

Fine-tuning Models with the Confusion Matrix and Loss Matrix

When you find errors in classification, it is helpful to know in what way the classifier is becoming confused, and the extent of loss this confusion entails. MineSet provides two complementary matrices to answer your questions: the Confusion Matrix and the Loss Matrix.

Using a Confusion Matrix to Investigate Errors

Suppose you are trying to classify mushrooms as poisonous or edible. Classifying a mushroom as poisonous might cost you $2 (the cost of the mushroom). Classifying a poisonous mushroom as edible however, might land you in the hospital with a $10,000 hospital bill (optimistically). The MineSet Confusion Matrix and Loss Matrix features are designed for handling such situations, and can give a detailed picture of errors and incorrect predictions.

Figure 11-3 shows a Confusion Matrix for the *mushroom* dataset with a Decision Tree trained on 10% of the dataset, considerably less than the recommended training set. See Chapter 7, “Understanding Predictive Modeling,” for details.

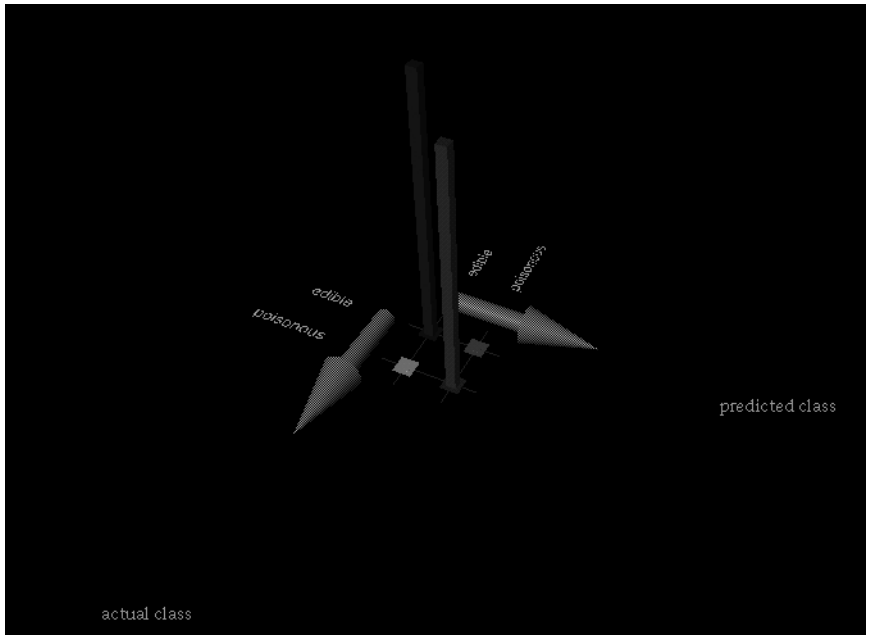


Figure 11-3 Confusion Matrix for the Mushroom Dataset

Eight records representing poisonous mushrooms were classified as edible (0.1%); 15 records representing edible mushrooms were classified as poisonous (0.2%). The remaining 3793 edible mushrooms and 3496 poisonous mushrooms were correctly classified. While the error rate for the model is only 0.31% (less than one percent), the estimated loss based on the example in previous paragraph is $10000*8 + $2*15 = $80,030.

Confusion Matrices give a more detailed picture of errors made by a model, by displaying a visualization of the type of errors being made. The two axes represent the *predicted* class values and the *actual* class values given in the test set (holdout set). The Confusion Matrix is computed prior to backfitting. The following section shows how to use a Confusion Matrix that follows the above scenario.

Displaying a Confusion Matrix

For this example use the Tool Manager window to open the data file *mushroom.schema*, and explore confusions in classifying mushrooms as edible or poisonous. This file is in the *MineSet\data* directory, in the location where MineSet was initially installed.

1. From the Tool Manager Data Destinations pane, click the Mining Tools tab; then click the Classify tab and make selections from these popup menus:
   - *Mode:* Classifier and Error
   - *Inducer:* Decision Tree
   - *Discrete Label:* edibility
2. Click *Advanced Options* and the Classifier Advanced Options pane shown in Figure 11-4 appears.

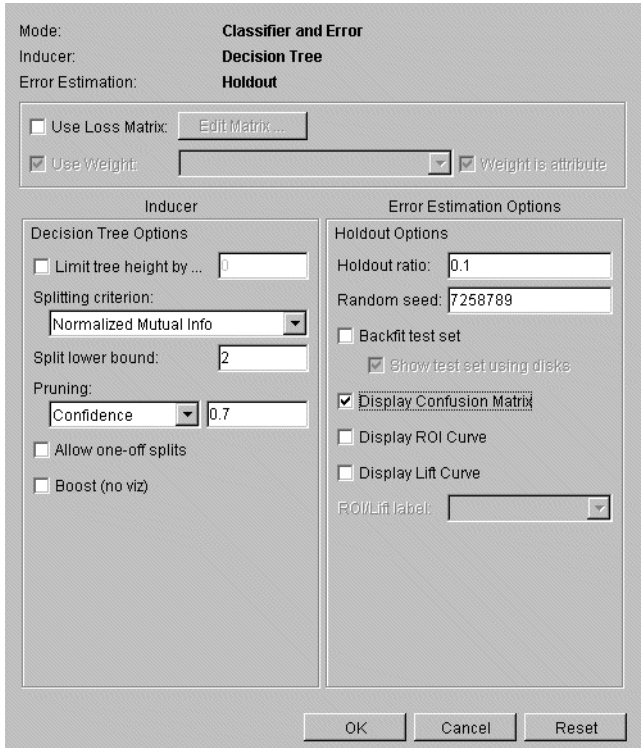


Figure 11-4 Classifier Options Pane Showing Confusion Matrix Checked

3. In the Error Estimation Options pane, change Holdout ratio to 0.1, turn off Backfit test set, click the check box at Display Confusion Matrix, then click OK.

   You are reducing the size of the training set in order to cause the classifier to make mistakes for demonstration purposes. A typical training set would be much larger.

4. Click *Go* on the Tool Manager Data Destinations pane.

   When the Decision Tree visualization appears, dismiss it and examine the Confusion Matrix shown in Figure 11-5. The Confusion Matrix displays where the classifier makes mistakes in classifying. From this you can construct a Loss Matrix based on what you now know about the data, to make certain kinds of errors less tolerable than others.

   The display shown in Figure 11-5 indicates that using this reduced training set, eight poisonous mushrooms were misclassified as edible, or 0.1% of total. Click on a selection in the visualization, then choose Show Values from the Selections menu of the Tool Manager to see the Selection List table.

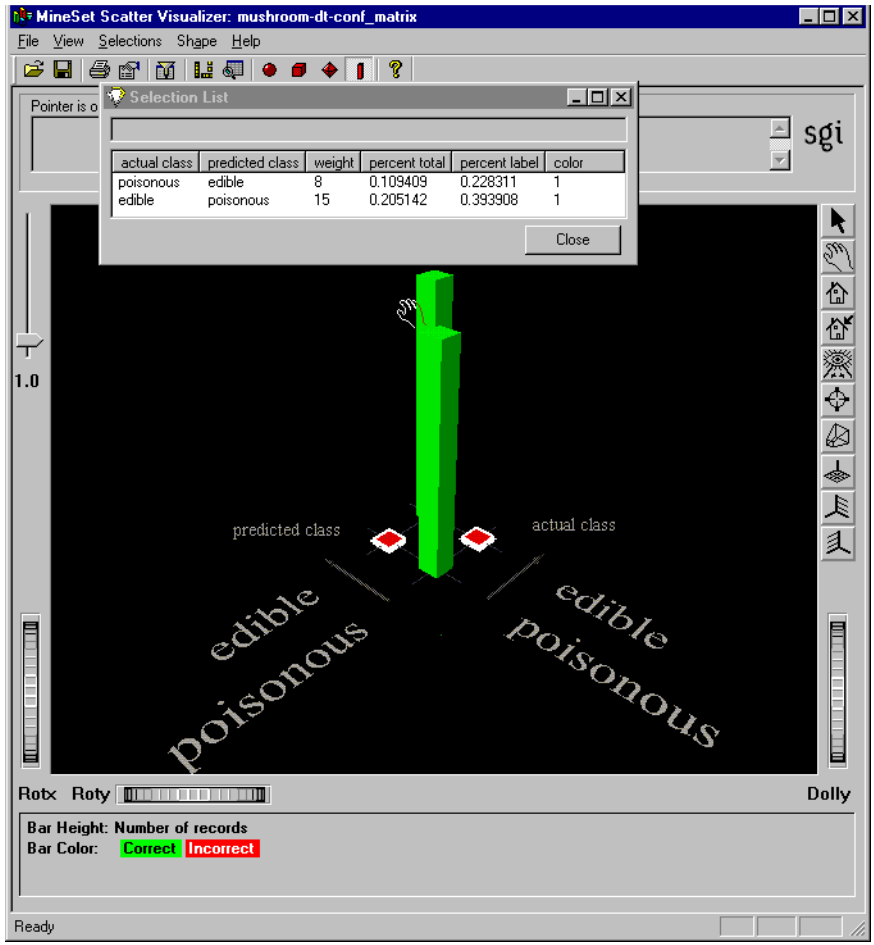


Figure 11-5 Confusion Matrix for Mushroom Dataset Showing Misclassifications

Defining a Loss Matrix

This section explains how a Loss Matrix may be used to reduce the more costly types of classification errors. The example continues the scenario from the previous section, using the mushroom dataset, a decision tree inducer, and classifying for the label edibility.

1. Dismiss the Confusion Matrix display with File > Exit and return to the Tool Manager window.
2. On the Mining Tools tab in the Data Destinations pane, click *Advanced Options* to open to the Classifier options pane.
3. In the second section of the pane, click *Use Loss Matrix;* wait briefly.
4. Click *Edit Matrix* to weight the cost of making errors. A Loss Matrix window similar to that shown in Figure 11-6 appears.

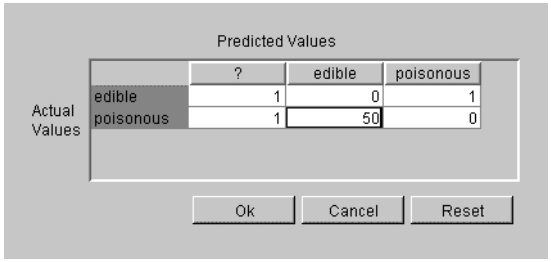


Figure 11-6 Loss Matrix Edit Pane

5. Set these values across the Actual Values rows of the Loss Matrix, reading from left to right as shown in Figure 11-6:

   edible: 1—0—1

   poisonous: 1—50—0

   This weights the cost of classifying poisonous mushrooms as edible 50 times more expensive as the reverse.

6. Click *OK* to set the Loss Matrix values settings, and *Go* to run the classifier.

   The new classifier is very conservative; no mistakes are made in classifying a poisonous mushroom as edible. On the other hand, 1558 mistakes are made in classifying an edible mushroom as poisonous. Using the figures of the cost of mushrooms ($2 each) versus hospital stay($10,000), the new estimated loss incurred is $10,000*0 + $2*1,558 = $3,116, only 3% of the cost of the classifier that did not take losses into account. Figure 11-7 shows the resulting Confusion Matrix.

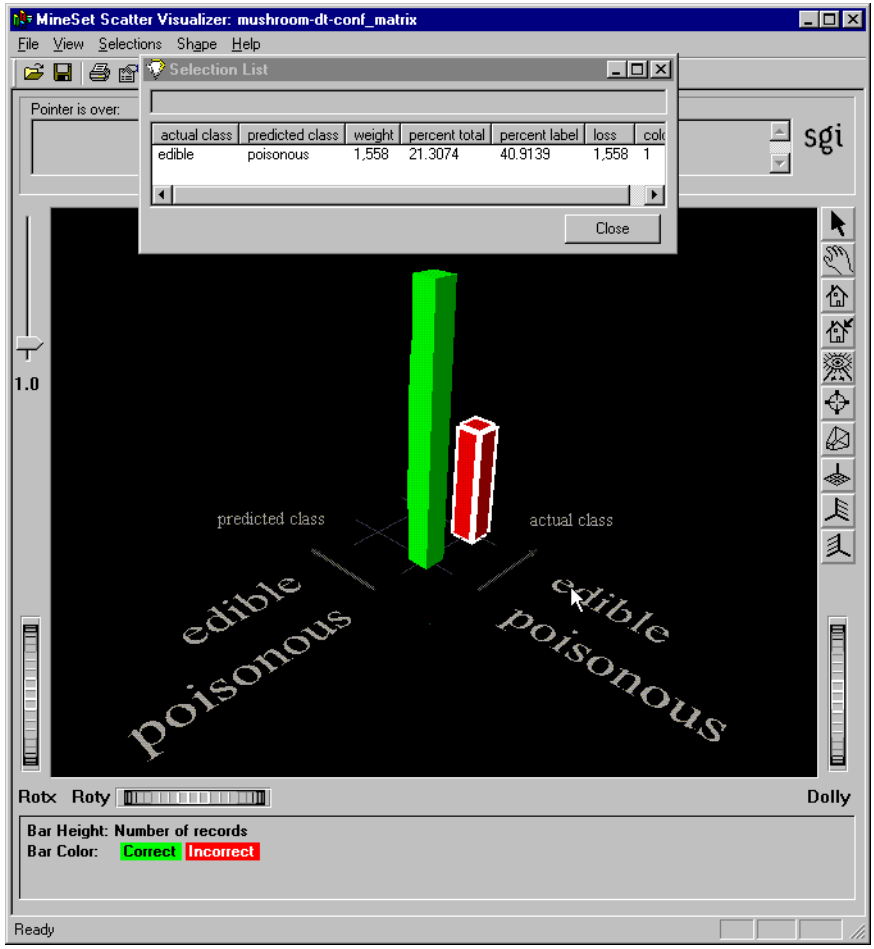


Figure 11-7 Confusion Matrix for the Mushroom Dataset with Loss Matrix

For more information on Error Estimation, see the *MineSet Enterprise Edition Reference Guide*.

Assessing Models with the Lift Curve and ROI Curve

When a business embarks on a course of action, it is often helpful to anticipate gains or losses before changing course or abandoning a project. MineSet provides an opportunity to gauge a probable outcome, and to make changes before events have significant impact. Two methods of doing this are with the Lift Curve and the Return-on-Investment (ROI) Curve.

Examining Prediction with a Lift Curve

A lift curve is a graph that shows the difference between a random ordering of records and an ordering produced by a model in predicting a particular label value. For example, when you create a model to predict which customers are likely to churn, you may want to target customers who are likely to churn *before* they actually churn. The lift curve helps accomplish this goal. This example uses the churn dataset.

A lift curve plots the number of records from 0 to 100% on the X axis and the number of records corresponding to customers who have a given label value (`Churn=yes` in this case) on the Y axis.

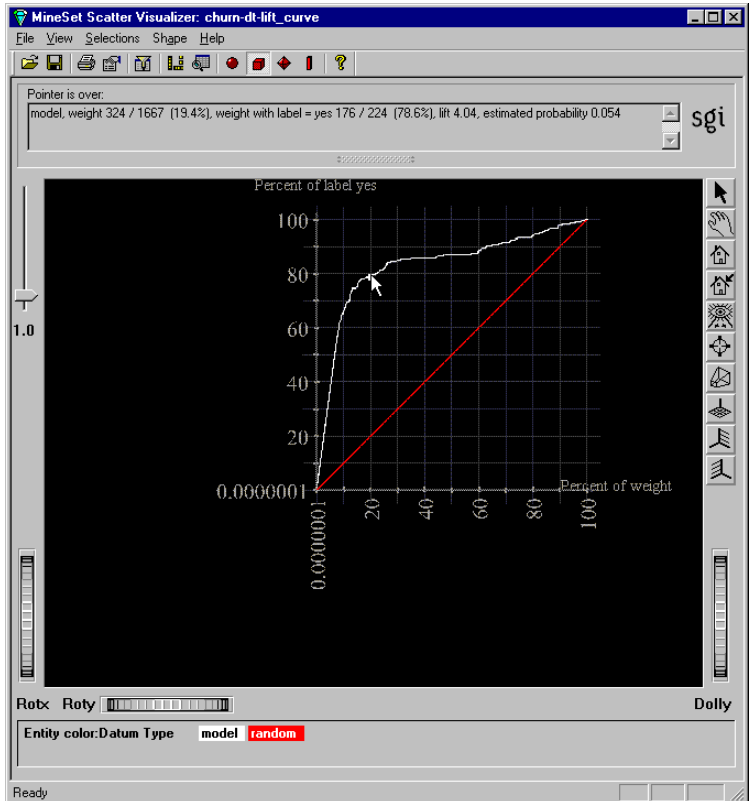


Figure 11-8 Lift Curve

Two curves are shown on the graph in Figure 11-8. The lower curve (red) shows the number of customers expected to churn given a random ordering of the records. The upper curve (white) shows the percentage of customers who churn when placed in order according to the model's score (probability estimate) for each record. Records representing customers whom the model identifies as most likely to churn appear first; those less likely to churn appear last. The advantage that the model ordering provides can be seen by the difference between the model curve and the random curve.

In building this lift curve, a selected model is applied to the test set. In the example, a specified segment of the dataset is used for training. Then the induced model is run on the remainder of the dataset.

To generate a lift curve on any selected model:

1. In the Tool Manager Data Destinations pane, click the Mining Tools tab.
2. Click the Classify tab.
3. Choose the Classifier and Error mode and Decision Tree inducer.
4. Click *Advanced Options*, and from the Classifier options popup menu, click the *Display Lift Curve* check box under Error Estimation Options. Choose a label value (yes) from the ROI/Lift label popup menu.
5. Click *OK* to accept the inducer options and dismiss the dialog box.
6. Click *Go.*

The process may take some time. Dismiss the Decision Tree. The resulting lift curve shows the details of any selected point in the selection window under the task bar. Move the pointer along the white (model) line, and click at various points to see the lift and percentage of customers with `churn=yes`. Look for the knee of the curve, in this example, where the estimated probability of the classifier is 0.056. This is the point at which the return on investment in sending incentives to customers that may churn diminishes rapidly. The next step is to apply the classifier to the full dataset (see "Classifying Using All Your Data" in Chapter 7).

Finding Marketing Benefits Using a Return on Investment Curve

A Return-on-Investment (ROI) curve is similar to a Lift Curve, but displays accuracy in terms of loss rather than in terms of error; depending on how you weight your Loss Matrix (see "Defining a Loss Matrix" on page 188). If the model is sure about its prediction, the expected loss will be low, and the record appears near the left side of the ROI curve.

The ROI curve expects you to act on the result of each individual record in the dataset. That action is the one associated with the chosen label value. For example, in the *churn* dataset, the action associated with the label Yes might be to send a person some marketing material. This might stop that person from churning, but the action is costly if done indiscriminately. The peak of the ROI curve shows approximately how much money would have been saved on the test set, if the model was used to predict whether or not to send the mailing to a particular person.

The Return on Investment curve lets you see the cost of making certain kinds of errors, and indicates to you the point at which it is no longer fruitful to continue taking action. To generate an ROI curve on the *churn* dataset using Classifier and Error mode:

1. In the Tool Manager Data Destinations pane, click the Mining Tools tab.
2. Click the Classify tab then click the *Advanced Options* button.
3. Check that Backfit test set, and Display Confusion Matrix, and Use Loss Matrix are clicked on in the Classifier options pane.
4. In the upper pane click *Edit Matrix* to weight the cost of making errors as shown in "Defining a Loss Matrix" on page 188. In this case, fill the rows this way:

   Actual Values: no: 1 0 2

   Actual Values: yes: 1 0 -10

   Click *OK* to set the Loss Matrix.
5. Set *Display ROI Curve* on.
6. Choose yes (for the churn example) from the ROI/Lift label pulldown menu.
7. Click *OK*.
8. Click *Go* in the Tool Manager window.

Three display windows appear: the Decision Tree, the Confusion Matrix, and the ROI Curve. The Confusion Matrix shows that the classifier is more conservative in making churn=no predictions, thus reducing false negatives. The Confusion Matrix displays a weighting of 4% which takes loss into account. The errors on one side have been increased, but those on the other have been decreased. Dismiss both the Confusion Matrix and the Decision Tree display using File > Exit, and examine the ROI curve window Figure 11-9.

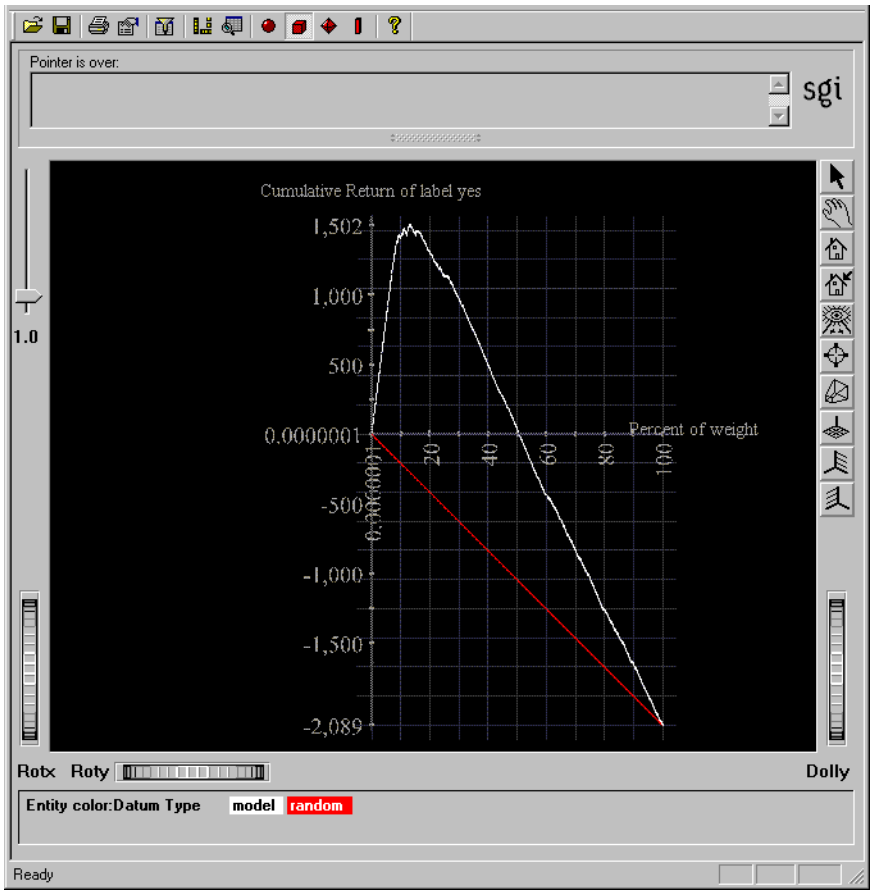


Figure 11-9 Return on Investment Curve

The ROI curve bears a marked resemblance to a lift curve. The horizontal line across the middle represents zero profit and loss. The red line represents the expected performance if you were to take a random sample of the population and send them mail. You expect a loss if you mail to everyone, because of the cost of mailing. However, there is a point of optimum return on investment, represented by the apex of the curve, at 1448, or 15.2 percent of the population.

Take care when filling out a Loss Matrix for use with an ROI curve. The column under a certain predicted label determines the resulting ROI curve for that label value. If you were to fill out a Loss Matrix for the churn dataset using the label churn, the entries in this column need to represent the expected gain or loss for taking the action associated with that label value, on all of the possible classes. For example, the entry under the column "prediction yes" in churn, under the row "actual value no," may contain the value 2 to indicate that the cost of mailing a brochure (the action associated with "yes") to someone who was not going to churn is 2 dollars. On the other hand, the entry under the column yes, row yes, may have a value of -10 to indicate that a customer was prevented from churning, saving the company 10 dollars over the cost of the mailing. These costs can be difficult to estimate, and small variations may significantly affect the position of the apex in the resulting ROI curve.

For further details, see the *MineSet Enterprise Edition Reference Guide* and check the MineSet website at http://mineset.sgi.com for the latest updates.

Segmenting the Data with Clustering

The purpose of clustering is to determine what elements or characteristics in the dataset are similar. This is useful for exploring a fresh dataset. For example, the iris dataset shows three clearly discriminated clusters around iris type—*iris-setosa*, *iris-versicolor,* and *iris-virginica*. This chapter includes the following topics:

- "Overview of Clustering"
- "Starting Clustering with Tool Manager"
- "Working with a Sample Clustering File"
- "Understanding the Cluster Visualizer Main Window"
- "Alternative Visualization of Clustering"

Overview of Clustering

Clustering is a descriptive data mining task, so you do not specify a column as a label. For this reason, Clustering is categorized as an unsupervised learning algorithm. Clustering models are always built from and evaluated on the full dataset.

MineSet forms clusters by grouping similar records together, aiming to maximize the overall similarity within each group. MineSet provides two distinct modes of clustering, single and iterative k-means, both of which are explained in depth in the "Clustering" entry in the *MineSet Enterprise Edition Reference Guide*.

Once the clustering has been run, you can view the Cluster centers directly using the Cluster Visualizer, which shows the independent attributes in a dataset. You can examine the most prominent, and see how each differs. However, to see how attributes relate to each other between clusters, the Scatter Visualizer and Decision Table provide a clearer view.

To view the clustered model using the Scatter Visualizer, you need to determine which columns should be mapped to the various axes by using the Column Importance tool. First use the Apply Model facility (see "Applying a Predictive Model" on page 136), and select the <*datasetname*>*.cluster* from the list of available models. The provides you with a new "Cluster" column. Map that column to color in the Scatter Visualizer.

Starting Clustering with Tool Manager

Clustering has no prerequisites and no required options. By default, MineSet uses the single k-means clustering method to discover three clusters in the data. Once the clustering is complete, you see an evaluation of the clustering, and then the Cluster Visualizer appears.

1. From the Tool Manager Data Destinations pane, click Mining Tools, and click the Cluster tab, as shown in Figure 12-1.

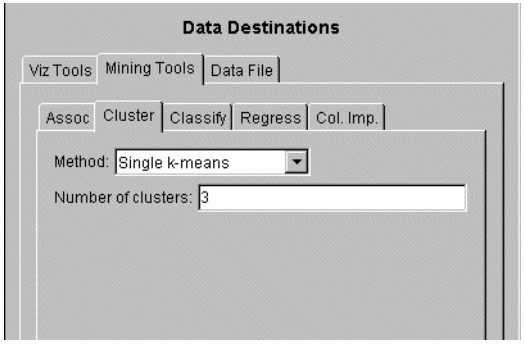


Figure 12-1 Cluster Tab

2. To control the choice of clustering method, from the Method popup menu choose either single or iterative k-means, as shown in Table 12-1.

Table 12-1 Clustering Method Options

Method	Description
Single k-means	Default method. Specify number of clusters to find (default 3).
Iterative k-means	Requires upper and lower boundary and Choice Point (a number between 0 and 1—default 0.5).

See the glossary definition for *Choice Point*.

3. Click *Go.*

Note: Clustering is a computationally intensive operation and can take some time to complete on larger datasets, especially when running in iterative k-means mode. If your dataset has more than 10,000 records, try clustering a sample of the data.

Clusters are assigned sequential numerical names starting from 1. Although cluster names are represented by numbers, there is no ordering to the clusters.

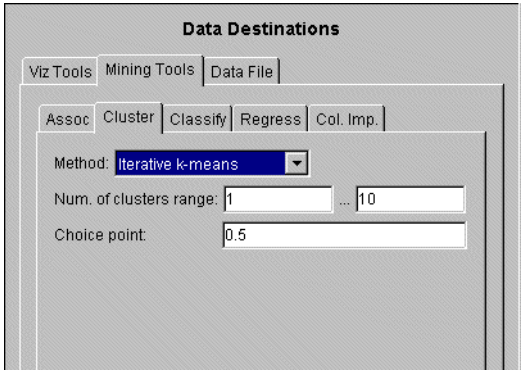


Figure 12-2 Clustering Using Iterative K-Means

In iterative k-means, clusters are named based on their derivation during the splitting process. The initial clustering (based on the minimum number of clusters) is named using sequential numbers, just as in single k-means. When a cluster is split, the two new clusters are named after the split cluster, with an "A" or a "B" appended. For example, the cluster named "2-B-A" was derived from cluster 2 in the initial clustering, and was split twice.

For details on the operation of the clustering algorithm, see the "Clustering" entry in the *MineSet Enterprise Edition Reference Guide*.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Working with a Sample Clustering File

A cars sample dataset provided with MineSet is relatively simple, dealing with familiar concepts of horsepower, vehicle weight, and time required to reach 60 mph.

Using the Tool Manager File > Open New Data File pulldown menu, select the *cars.schema* file. Look in the directory in which MineSet is installed under *\data.*

1. In the Tool Manager Data Destinations pane, click the Mining Tools tab.
2. Click the Cluster tab, leave the Method as Single k-means, and the number of clusters as 3, then click *Go* in the lower right.

Figure 12-3 Cluster Visualizer Main Window

3. In the resulting Cluster Visualizer window (Figure 12-3), click the green bar showing cluster 1, which controls the priority ordering of attributes represented by the bar charts and histograms. When you click on cluster 1, the attribute importance (that is, the order in which the histograms are presented, reading from top to bottom) is cylinders, cubic inches, horsepower, weight lbs, then miles per gallon. This order of importance is only controlled by this cluster for the visualization.
4. Compare the same row across the other clusters to see how that attribute differs from cluster to cluster.
5. Select cluster 2, and notice a difference in the order of attributes. In this case, origin is more important, then brand, then cubic inches, cylinders, weight lbs, then horsepower, then miles per gallon.

Understanding the Cluster Visualizer Main Window

The Cluster Visualizer shows box plots arranged in rows and columns. One row is shown for each attribute in the data, and each column represents a cluster which is named in the top pane. The Population column displays statistics for the dataset as a whole.

By following the procedure in "Starting Clustering with Tool Manager" on page 198, you achieve a display similar to that shown in Figure 12-3, with the clusters numbered sequentially. The algorithm cannot detect what the clusters represent, and human interpretation is required. The order in which attributes are displayed in the Cluster Visualizer window is significant; by default, attributes are displayed in order of importance for discriminating between the clusters.

To change the order of the columns:

1. Click the pane containing the name of a cluster to redisplay the attributes in order of importance for discriminating that cluster from all others.
2. Click on the pane containing "Population" to restore the default ordering.

Alternative Visualization of Clustering

The Cluster Visualizer provides an easy-to-use basic visualization of the clustering. However, it is not the only way to visualize clustering. An example of transferring the results of clustering into a display using the Scatter Visualizer is shown in the *MineSet Enterprise Edition Tutorial.* The main idea is to create a new column by applying the just-built clustering model, then using the visualizer of choice. The Scatter, Evidence, and Decision Table Visualizers are particularly useful for viewing clustering results.

Analyzing Data with Association Rules

Association Rules mine data correspondences, that is, which data instances are likely to occur with other specific instances. This is known as market-basket analysis, based on the metaphor of what products are likely to be purchased together in the same market basket. This chapter covers the following topics:

Sample configuration and data files have been provided to demonstrate the Association Rules features and capabilities. These files are in the *\examples* directory under *MineSet*, in the location where MineSet was initially installed.

Overview of Association Rules Generation and Visualization

If you have ever wondered why egg substitutes and waffles are displayed next to one another in the supermarket frozen food display case, you may have seen association rules in action. Association Rules discover relationships and patterns in data—a classic data-driven knowledge discovery process in which correlations that were not immediately apparent become clear, and can be displayed in visual form. Association Rules are particularly useful in discovering patterns in large datasets, with the rules of association indicating the frequency of items occurring together in a dataset. This is an unsupervised learning algorithm because it does not focus on any one particular attribute. Instead it treats all attributes equivalently, and does a global search for interesting rules.

The development of Association Rules proceeds in two phases: the data file is first processed by the Association Rules Generator, which creates a file usable by the visualizer. Then the file is displayed by the visualizer, which in this case is a Scatter Visualizer.

Association Rules Generation

The Association Rules Generator can generate both simple (one-to-one) and multiway association rules. This section describes simple association rules. For a description of multiway rules, see "Multiway Association Rules" on page 214.

A simple association rule states that given that X is true, there is a certain probability that Y is also true. MineSet refers to X as the left-hand side (LHS) of the rule and Y as the right-hand side of the rule (RHS).

Consider a set of items bought by a customer on a single visit to a store. In this context the rule might be: "80% of the people who buy diapers also buy baby powder." This percentage is known as the *confidence* of the rule.

In this example, "diapers" is the item on the left-hand side (LHS) of the rule, and "baby powder" is the item on the right-hand side (RHS) of the rule. In this case:

- If baby powder appears on the RHS, the LHS can help determine what should be done to boost sales of this item.
- If diapers appears on the LHS, the RHS can help determine what products might be affected if the store were to discontinue selling diapers.

The Association Rules Generator processes an input file, then generates an output file consisting of the rules. If X and Y are items in a record, then a rule such as:

X=>*Y*

indicates that whenever *X* occurs in a record, expect *Y* to occur with some frequency.

The strength of the association is quantified by four numbers, summarized in Table 13-1

Table 13-1 Association Rules Components

Measure	Description	Statistical Description
Support	Frequency of LHS and RHS occurring together.	P(LHS ∩ RHS)
Confidence	Of all occurrences of LHS, the fraction where RHS is also seen, or the support divided by the frequency of occurrence of LHS items.	P(RHS | LHS)
Expected confidence	Frequency of occurrence of RHS items.	P(RHS)
Lift	Ratio of confidence to expected confidence.	P(RHS | LHS):P(RHS)

See the glossary definitions *Support*, *Confidence*, *Expected Confidence*, and *Lift*. For technical details see the "Association Rules" entry in the *MineSet Enterprise Edition Reference Guide*.

Note: Given only Y and a rule of the form X=>Y, nothing is known about X. Rules specify implications only from the LHS to the RHS.

Rules Visualization

You can see the rules of association in graphic form when it is displayed on a grid landscape. The left-hand side (LHS) items are on one axis, and right-hand side (RHS) items are on the other. As shown in Figure 13-1, attributes of a rule are displayed at the junction of its LHS and RHS item. The display can include bars, disks, and labels.

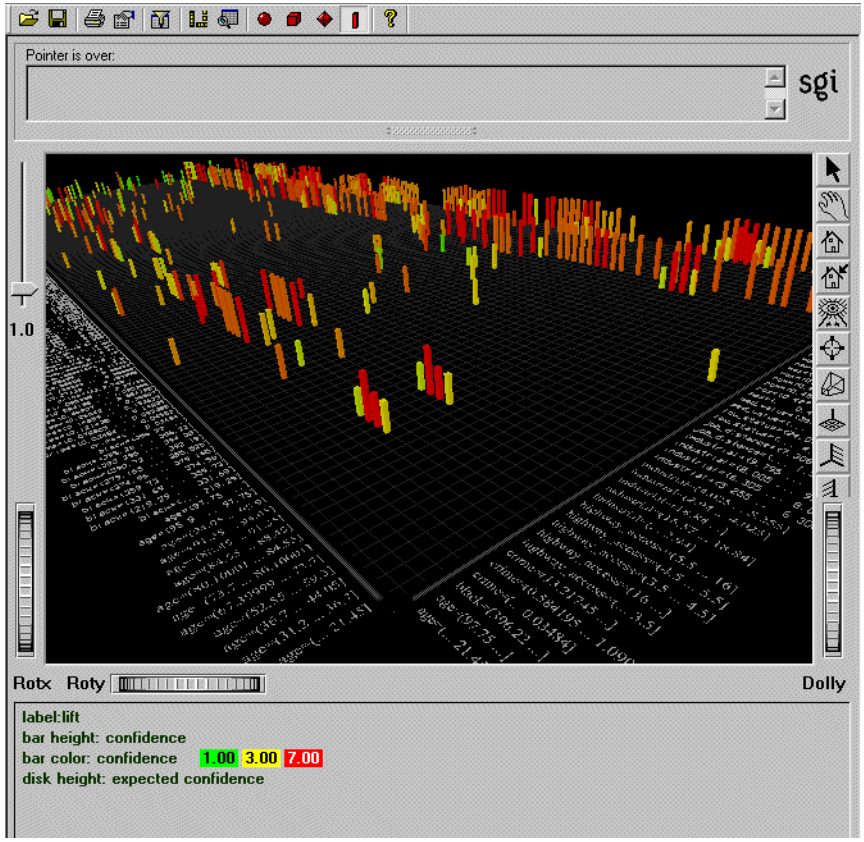


Figure 13-1 Detail View of the Association Rules Visualizer's Main Window

A legend indicating the mapping between displayed attributes (such as bar heights and colors) and the values associated with the underlying rules (such as confidence and support) is displayed at the bottom of the main window.

Starting the Association Rules

This section describes how to use the Tool Manager to simplify the task of configuring the Association Rules tool.

Setting Up Associations

This example uses the cars dataset to set up simple associations. You may discover any association between miles per gallon, horsepower, acceleration, weight, engine size, origin, brand, and the year the car was built. For example, did mileage improve over time? Did engines become less powerful?

The simplest way to set up the associations is to follow these steps:

1. Bring up Tool Manager, and choose a server and choose *cars.schema* as the data source (see "Starting MineSet" on page 15.)
2. In the Data Destinations pane of Tool Manager, click the Mining Tools tab; from the lower row of tabs, choose *Assoc* (for Association Rules). Figure 13-2 is displayed.

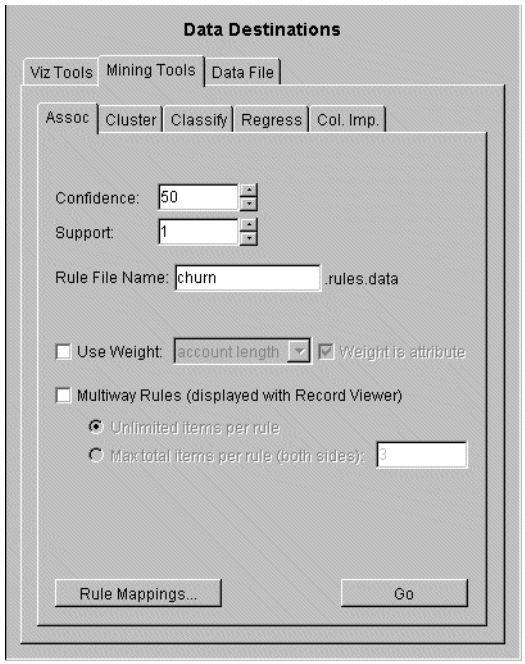


Figure 13-2 Initial Tool Manager Window for Association Generation

3. (Optional step) The algorithm for Association Rules works only with discrete values; therefore continuous-type columns are binned automatically. If you prefer different bins, use the *Bin Columns* button on the Data Transformations pane. (See "Changing or Creating New Bins for Columns" on page 39 for details.)
4. You can run the Association Rules Generator by clicking *Go.* Or you can choose settings:

   Confidence—lets you specify the minimum confidence threshold for rules. Rules with a confidence below this value are not generated. The default is 50%. The possible values are 1–100.

   Support—lets you specify the minimum support threshold as a percentage of the total number of records. Rules with a support below this value are not generated. The default is 1%. The possible values are 1–100.
5. (Optional) Once you have made your association rule options selections, click the *Rule Mappings* button to map rule attributes such as support, confidence, expected confidence, and lift to visual elements. See "Mapping Rule Attributes to Visual Elements" below.

Dragging and Dropping Existing Visualization Files

If you have the Viz Tool icon on your desktop, you can drag an existing MineSet visualization file icon and drop it on the Viz Tool icon. The visualization will then display in the Viz window.

You can also drag a visualization file into an open Viz Tool window. If you have your preferences set to Single Document mode, you must drop the file icon onto the title bar of the Viz window. If you have your preferences set to Multiple Document mode, you can drop the file icon anywhere within the window, and it will display.

Record Weighting

The Association Rules allow for record weighting for cases in which you want to specify that certain records are more important than others, or when you want to compensate for uneven sampling. If you do not check the box for *Weight by Column*, then each record has a weight of one. The *Weight by Column* box is only active when the dataset contains columns available for weighting.

To enable record weighting, click the *Weight by Column* checkbox. When the box is checked, a popup menu appears that allows you to choose the column that contains the weight for each record. The *Weight is attribute* box, if checked, includes the weight column in the rules found by the Association Rule Generator. If the box is unchecked, all records are given equal weight in determining rules found by the Generator. See "Weighting Records" in Chapter 3 for a further explanation of record weighting.

Mapping Rule Attributes to Visual Elements

The Association Rules tool lets you map attributes of the rules to visual elements of the display. Clicking on the Rule Mappings button brings up the Mapping Rules to RuleViz Elements panel shown in Figure 13-3.

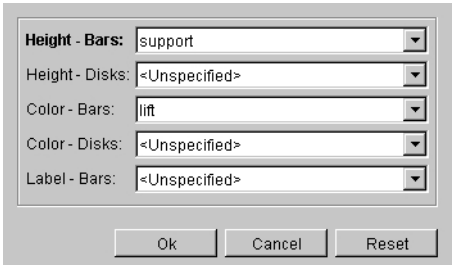


Figure 13-3 Association Rules Mappings

The visual elements that can be mapped are listed below; the items preceded by an asterisk are optional:

Table 13-2 Association Rules Mapping to Visual Elements

Visual Element	Mapping
Height - Bars	Specifies what the bar heights represent. Default is Support.
*Height - Disks	Specifies what the disk heights represent.
*Color - Bars	Specifies what the bar colors represent. Default is Lift.
*Color - Disks	Specifies what the disk colors represent.
*Label - Bars	Specifies what the bar labels represent.

Interpreting Association Rules in the Scatter Visualizer

The Association Rules tool displays the data from a rules file in the Scatter Visualizer using the specifications of a valid configuration file. To see a sample file, from the MineSet 3D Visualizer use File > Open to see a list of configuration files. These files are in the *MineSet\examples* directory, in the location where MineSet was initially installed. For example, specifying *brand.rules.scatterviz* results in the image shown in Figure 13-4. Files of association rules can be identified by having names ending with.*rules.scatterviz.*

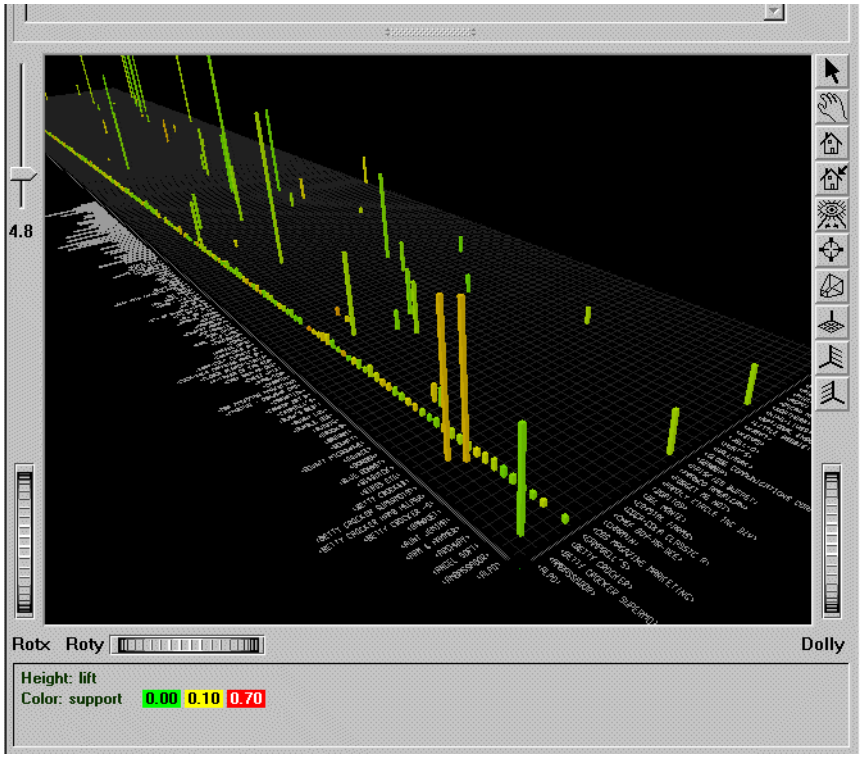


Figure 13-4 Initial Association Rules View when Specifying brand.rules.scatterviz

The rules are presented on a grid landscape in the Scatter Visualizer. The left-hand side (LHS) items are on one axis, and right-hand side (RHS) items are on the other. As shown in Figure 13-4, attributes of a rule are displayed at the junction of its LHS and RHS item. For example bar heights correspond to support and bar colors correspond to lift. If the displayed view is too small, item labels do not appear on the sides of the axes. You can zoom in on the view until the item labels appear using the Dolly thumbwheel at the

lower right of the window. You can also view the labels for a particular rule by placing the mouse pointer over an individual bar when the mouse is in select mode (see Figure 13-5). All of the details for that particular rule will be displayed in the upper left-hand corner of the view area.

For example, in Figure 13-4, bar heights correspond to support and bar colors correspond to lift.

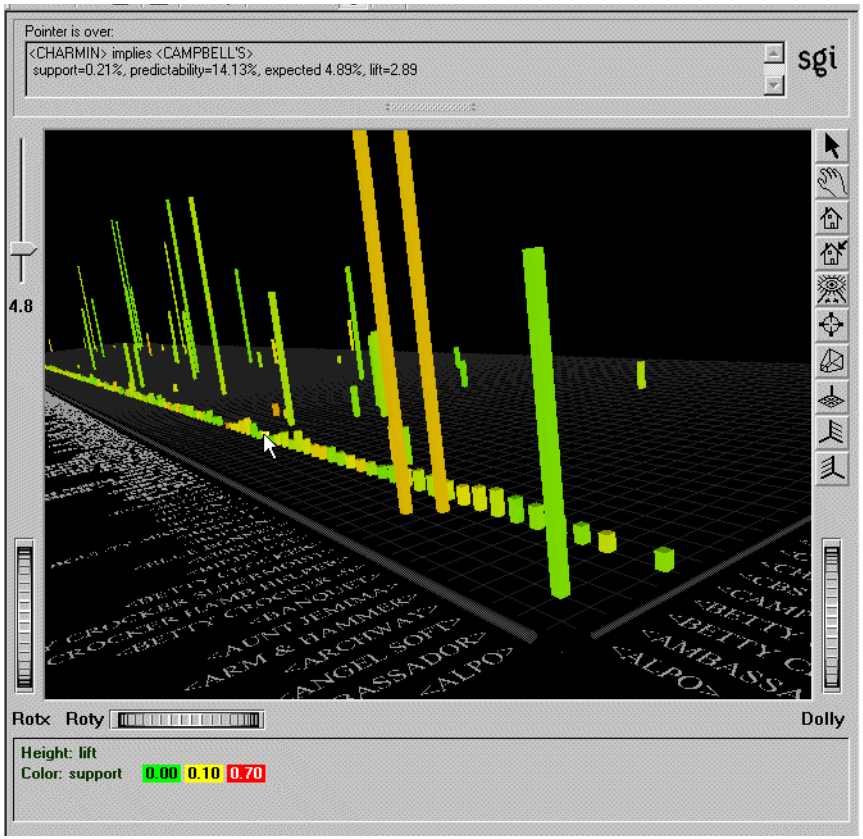


Figure 13-5 Cursor Over a Bar Representing a Rule

Placing the mouse pointer over an Association Rules object as shown in Figure 13-5 displays that object's information. The information is displayed as long as the pointer remains over the object. If you position the pointer over an object and click the left mouse button, that same information appears in the selection area above the main window. In addition, the bar is selected and appears in a separate window containing all selected rules. Multiple rules may be selected by holding down the Shift key while clicking.

This information remains visible until another object is selected, or until no object is selected (if you click the black background). Using the mouse, you can cut and paste text from the selection window into other applications, such as reports or databases.

Drilling Through

The drill-through expression is determined by doing a logical "and" of selected rules. Because the columns in the original table do not match the columns in the *.rules.data* file, the rules generator produces a special column to help construct the filter expression when a drill-through is performed. This means that changing the drill-through preferences panel has no effect, because a special string-valued column has already been mapped to drill-through in the *.rules.scatterviz* file.

When you drill through on a rule (or rules), MineSet shows all the records that satisfy the rule (or rules).

Multiway Association Rules

In some cases, it is useful to have more complex rules that have multiple items on the LHS and/or the RHS. These are multiway association rules. Figure 13-6 illustrates the Tool Manager Association panel configured for multiway rules generation.

If you check the Multiway Rules check box, the Association Rules Generator generates all rules that satisfy the minimum support and confidence thresholds, including those that have more than one item in the LHS and RHS. An example of such a rule might be "beer and linguini implies potato chips and salsa and wine."

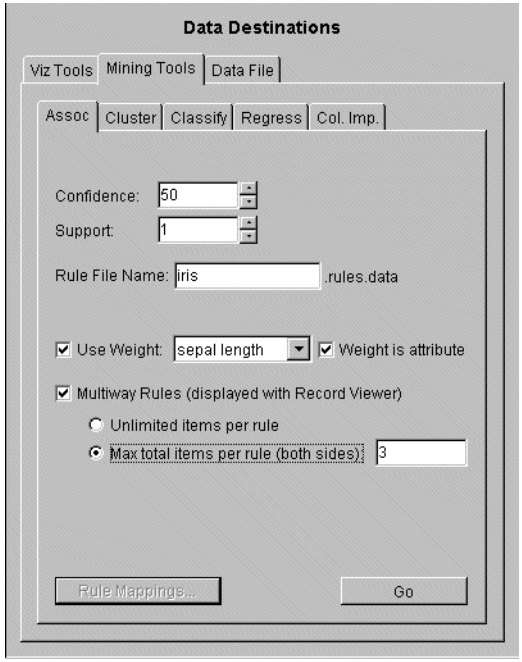


Figure 13-6 Initial Tool Manager Window Set Up for Multiway Association Generation

Multiway rules are displayed using the Record Viewer because there is no obvious way to show the more complex rules in the Scatter Visualizer. They are displayed with one rule per row. The first two columns of the table contain the number of items in the LHS and RHS. The next four columns contain the support, confidence, expected confidence, and lift values. The last two columns contain the LHS and RHS items. In the LHS and RHS columns, the items are separated by the word "and." In the example rule above, the LHS contains two items and is represented as "beer and linguini." The RHS contains three items and is represented as "potato chips and salsa and wine."

You can limit the size of the rules generated by entering a number in the "Max total items per rule" field. This number indicates the maximum number of items that are allowed in any rule. The number of items in a rule is the sum of the number of items in the LHS and RHS. The example rule above has three items; simple rules have two items.

Note: Generating multiway rules can take a long time. Watch the Tool Manager status window for an indication of the number of rules generated at each iteration. If too many rules are being generated, cancel the operation and increase the minimum support or confidence thresholds, or decrease the maximum allowable number of items per rule.

MineSet User's Glossary

Aggregate

The act of aggregation (see below) in which data is combined in various ways.

Aggregation

The process of summarizing multiple rows of data using one or more operations over some specified set of columns. Aggregation operations are capable of producing summary statistics, transposing a table, or even producing arrays based on the original data.

Algorithm

A formal procedure for accomplishing some task. For example, an inducer algorithm is a set of machine instructions that tell the computer how to build a classifier or regressor.

Array

A data type capable of storing multiple items. Array types are produced by aggregation and are used by the visualizers to store data which animates over time. MineSet supports one-dimensional arrays (also known as vectors), and two-dimensional arrays (also known as matrices).

Attribute

A column used as input by a predictive model. When performing predictive modeling, columns in the data are divided into Attributes (which provide the inputs) and the Label (which represents the output of the model).

Automatic Binning

An algorithm which chooses bin thresholds (see binning) so that the distribution of labels within different bins are as different as possible. Formally, this algorithm minimizes the entropy (see Entropy) within each bin.

Backfitting

Backfitting applies the full dataset to a model built from a smaller sample. The resulting model retains its original structure. The purpose of backfitting is to allow a model's internal distributions to accurately reflect the full data, while maintaining the model's error estimate.

Binning

Binning partitions continuous data into discrete groups, transforming real data into categorical data. For example, a continuous range of ages can be binned as 0-18, 19-25, 26-35, and so on. These groups or bins are defined as regions of a continuum that do not overlap.

Cake Charts

Segmented blocks displayed in a visualization showing a distribution of data. Similar to pie charts, but square.

Choice Point

The choice point is a value between zero and one which guides, for instance, the selection of the number of clusters; higher choice points suggest a larger number of clusters while lower choice points suggest smaller. A choice point of 1.0 will always pick the upper boundary. In clustering, if your boundaries are one and five clusters, a choice point of 0.4 might pick two clusters while a choice point of 0.8 will pick four. A choice point of 1.0 always picks five.

Classifier

A predictive model that attempts to describe one column (the label) in terms of others (the attributes). A classifier is constructed from data where the label is known, and may be later applied to predict label values for new data where the label is unknown. Internally, a classifier is an algorithm or mathematical formula that predicts one discrete value for each input row. For example, a classifier built from a dataset of iris flowers could predict the type of a presented iris given the length and width of its petals and stamen). Classifiers may also produce probability estimates for each value of the label. For example, a classifier built from a dataset of cars could predict the probability that a specific car was built in the United States. Classifiers only work with discrete labels. A Regressor is the analog of a Classifier for continuous labels.

Clustering

Clustering determines which elements in a dataset are similar. It works to group records together according to an algorithm or mathematical formula that attempts to find centroids, or centers, around which similar records gravitate. It is the process of dividing a dataset into mutually exclusive subgroups, without relying on predefined classes.

Conditional Probability

The probability of some event (A) given that some other event (B) has occurred. Written P(A | B) and read "probability of A given B. Conditional probability is shown by the rectangular charts in the Evidence Visualizer's left-hand window, which show the relative probability of each attribute value given (conditioned on) each label value. Conditional probability can be thought of as evidence for a given label value.

Continuous-valued Attribute

An attribute which can take on a full range of continuous values. In MineSet, columns of types int, double, and float may be considered continuous.

Confidence

Confidence of an Association Rule (X --> Y), quantifies how often *X* and *Y* occur together as a fraction of the number of records in which *X* occurs. For example, if the confidence is 50%, *X* and *Y* occur together in 50% of the records in which *X* occurs. Thus, knowing that *X* occurs in a record, the probability that *Y* also occurs in that record is 50%.

Configuration File

A file that identifies how the data from the *.data* file is to be mapped to visualizer requirements. MineSet configuration files have the following possible suffixes: .treeviz, .scatterviz, .splatviz, .mapviz, .eviviz, .dtableviz, .statviz, and .clusterviz

Cross-Validation

A method of estimating predictive error. Cross validation splits that dataset into k equal-sized pieces called folds (typically 10). k predictive models are built, each tested on a distinct fold after being trained on the remaining folds. The process can be repeated multiple times to increase the reliability of the estimate.

Data Cleaning

Data cleaning is the process of improving the quality of the data by modifying its form or content, for example, by removing or correcting erroneous data values.

Data File

A data file for MineSet is a tab-separated flat file of original data from which the model or visualization is created. MineSet's data files use the *.data* suffix. They may have been in binary (smaller and faster) or ASCII (human readable) format.

Dataset

A dataset is a table having rows and columns of data. The rows are sometimes referred to as records or instances. The MineSet CD includes several sample datasets, such as *churn* for people interested in telecommunications habits, *cars* which is of general interest, *adult* which is a dataset based on information culled from the US Census Bureau. A dataset can also be described as a schema and set of instances matching the schema. Generally, no ordering on instances is assumed.

Decision Tree

A predictive model based on a branching series of tests. Each test examines the value of a single column in the data and uses it to determine the next test to apply. The results of all tests determine which label to predict.

Discrete Attribute

A column with a finite number of distinct values. In MineSet, strings, binned values, and dates are considered discrete. Ints may also be considered discrete if used as the label for a classifier.

Drill Through

The act of retrieving the portion of the original data upon which a visualization is based.

Entropy

A measure from Information Theory which refers to the degree of disorder of data. The higher the entropy of a dataset, the more diverse and mixed its values. Many of the mining algorithms in MineSet operate by dividing data so as to minimize entropy.

Error Rate

The rate of correct or incorrect predictions made by the model over the dataset. Error rate is usually determined by using an independent test set that was not used at any time during the learning process. More complex accuracy estimation techniques, such as cross-validation, are sometimes used, especially with datasets containing a small number of instances.

Evidence

The degree to which a probability estimate affects an eventual prediction. The Evidence classifier predicts its label values based on the sum total of evidence from each attribute. Formally, evidence is the negative log of the normalized conditional probability.

Expected Confidence

Expected confidence is the frequency of occurrence of the RHS item in the dataset. So the difference between expected confidence and confidence is a measure of the change in predictive power due to the presence of the LHS item. Expected confidence gives an indication of what the confidence would be if there were no relationship between the items.

Filter

In MineSet, a filter usually a boolean expression used to select or manipulate column contents. For example, 'm.p.g. < 25' means to select only those entries showing less than 25 miles per gallon (that is, those records for which this expression evaluates to true). A filter is a saved set of chosen criteria that specify a subset of records in a dataset.

Holdout

The fraction of the dataset (commonly two-thirds) which is used to train a model so that the remainder may be set aside for testing. The holdout method is the simplest form of error estimation supported by MineSet.

Hypothesis Testing

An hypothesis is a proposed explanation whose validity can be tested. Hypothesis testing is a top-down approach that attempts to validate or disprove preconceived ideas. In creating an hypothesis, one thinks up a possible explanation for a remarked behavior. By extension, the hypothesis dictates the data selected to be analyzed.

Inducer

An algorithm which builds a model from training data. MineSet supports four classification inducers (Decision Tree, Option Tree, Evidence, Decision Table), and one regression inducer (Regression Tree).

Instance

A term for a record or row.

Knowledge Discovery

Knowledge discovery is the process of finding novel, interesting, and useful patterns in data. Data mining is a subset of knowledge discovery. It lets the data suggest new hypotheses to test.

Label

The single column which a predictive model is attempting to predict. The label must be selected in advance by the user. For example, "iris type" is a good candidate for the label in the iris dataset.

Lift

The ratio of confidence to expected confidence of a rule created by the Association Rules Generator. Generally, the greater this number, the more interesting the rule.

Lift Curve

A method for assessing predictive accuracy of a model. The lift curve visually displays the advantage a classifier provides over random determination of the label.

Mean

The mean may be thought of as the number found by adding the values of data in a sample, then dividing by the number of pieces of data. The mean is usually denoted by x-bar, and is popularly called "average." (For example, the mean of (1 +2 +3) /3 is 2.)

Median

The middle number when numbers are arranged in order of size.

Model

A description based on observed phenomena, often omitting certain details. A model may imply prediction. For example, if a direct mail model is formed it implies "This is what we think a direct mail customer looks like." The learner can change values and observe the effects of the change on the operation of the system. See also Classifier.

Predictability

An alternative term for Confidence.

Prevalence

An alternative term for Support.

Prior Probability

The prior probability for a class label is the probability of seeing this label in the data for a randomly chosen record, ignoring all attribute values. Mathematically, this is the number of records with the class label divided by the total number of records. (See also Conditional Probability. Prevalence was called support in previous versions of MineSet.)

Purity

Purity in the Column Importance context is a measure of correctness of the label value distribution. The cumulative purity is a measure of how well the data is partitioned in reference to the label values. The data is partitioned using columns found as important in the same way data is partitioned in a Decision Tree. Each set in the partition has its own purity measure, and the purity measure within the partition is a combination of these individual measures. For a given set in the partition, the purity is 0 if each class has equal representation, and 100 if every record is of the same class. Similarly, the cumulative purity will be 0 if each set in the partition has an equal representation of classes, and 100 if each set in the partition contains record that all have the same class. In MineSet, purity is based on Entropy.

Random Seed

Random seed is a method of selecting a particular starting point for selecting a random sample data. When you want a different random sample, specify a different random seed. Using the same seed for different mining explorations of a dataset allows you to work with the same random sample each time. Change the seed when you want to test the stability of a particular mode discovered during exploration.

Range

The difference between the largest and smallest possible value of a column.

Regressor

A predictive model where the label takes on a continuous value. Regressors are very similar to classifiers.

Relational Database

Relational databases are at the heart of the data warehouse. Data and relations between them are organized into tables—collections of records with each entry containing the same fields. Some fields are designated as keys, so that searches indexed by specific values for key fields retrieve data rapidly. Records in different tables may be linked if they have the same value in a certain field. Examples are Sybase, Informix, OLEDB, SQL server, and Oracle.

Rules

Rules are prescribed patterns of behavior applied to the handling of data, such a rules of association, or rules of classification. It is also the unique path followed in a decision tree to get from root to leaf in order to classify the records. A rules-based system performs inferences by applying a set of "if-then" rules to a set of facts following a given procedure.

Return on Investment (ROI)

A financial term which measures the worth of a project by measuring what benefits (return) accrue from an investment. Also known as ROI.

Row

A record in a relational table.

Schema

A description of all columns in a dataset. The schema includes a name and type information for each column. Schemas may be stored in files with a .schema suffix.

Skewness

The lack of symmetry in a frequency distribution.

Standard Deviation

A measure of dispersion of data. It is defined as the square root of variance.

Support

Given an Association Rule *X* --> *Y* (X implies Y), support quantifies how often X and Y occur together in the file as a fraction of the total number of records. For example, if the support is 1%, *X* and *Y* occur together in 1% of the total number of records. Support is the same as Prevalence.

Table

A normalized relational dataset.

Test Set

A test set consists of records in the dataset that have been set aside to test the error rate of a classifier after it has been induced. See also Training Set.

Training Set

A training set in MineSet is a subset of the data, segmented off before major data manipulation, used to build a classifier or model. It consists of records in the database for which the label has been determined, based on descriptive attributes. It is used by the inducer to learn how to construct the classifier. See also Test Set.

Trimming Fraction

The trimming fraction indicates the fraction of extreme values of the dataset to be excluded from the value range prior to binning. The default trimming fraction is 0.05. This excludes the 5% of the instances with the most extreme values (2.5% with the lowest values in the range, and 2.5% with the highest values in the range). Trimming tends to reduce the influence of outliers on the generation of thresholds.

Uniform Range

Uniform range is a selection used in automatic binning of data in which the value range is divided into uniformly sized subintervals.

Uniform Weight

Uniform weight is a selection used in automatic binning of data in which the value range is divided into the specified number of bins so that each bin contains the same number of records. If record weighting is enabled, the range is divided so that each bin contains the same total weight.

Weight

MineSet supports two types of weights:

A record weight is a numerical value applied to each row in the data. The mining tools treats a row with a weight k as if there were k copies of that row. Non-integral weights are supported as well. The weight values are supplied by a user-selected column in the data.

An attribute weight is a user-specified measure of importance for a column. Attribute weights are used by the clustering algorithm to affect the distance metric.

Index

B

C

D

P

Q

R

S

T

U

V

W