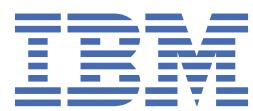*IBM SPSS Modeler 18.6 User's Guide*

IBM

Note

Before you use this information and the product it supports, read the information in "Notices" on page 255.

Contents

Chapter 1. About IBM SPSS Modeler

IBM SPSS Modeler is a set of data mining tools that enable you to quickly develop predictive models using business expertise and deploy them into business operations to improve decision making. Designed around the industry-standard CRISP-DM model, IBM SPSS Modeler supports the entire data mining process, from data to better business results.

IBM SPSS Modeler offers a variety of modeling methods taken from machine learning, artificial intelligence, and statistics. The methods available on the Modeling palette allow you to derive new information from your data and to develop predictive models. Each method has certain strengths and is best suited for particular types of problems.

SPSS Modeler can be purchased as a standalone product, or used as a client in combination with SPSS Modeler Server. A number of additional options are also available, as summarized in the following sections. For more information, see https://www.ibm.com/analytics/us/en/technology/spss/.

IBM SPSS Modeler Products

The IBM SPSS Modeler family of products and associated software comprises the following.

- IBM SPSS Modeler
- IBM SPSS Modeler Server
- IBM SPSS Modeler Administration Console (included with IBM SPSS Deployment Manager)
- IBM SPSS Modeler Batch
- IBM SPSS Modeler Solution Publisher
- IBM SPSS Modeler Server adapters for IBM SPSS Collaboration and Deployment Services

IBM SPSS Modeler

SPSS Modeler is a functionally complete version of the product that you install and run on your personal computer. You can run SPSS Modeler in local mode as a standalone product, or use it in distributed mode along with IBM SPSS Modeler Server for improved performance on large data sets.

With SPSS Modeler, you can build accurate predictive models quickly and intuitively, without programming. Using the unique visual interface, you can easily visualize the data mining process. With the support of the advanced analytics embedded in the product, you can discover previously hidden patterns and trends in your data. You can model outcomes and understand the factors that influence them, enabling you to take advantage of business opportunities and mitigate risks.

SPSS Modeler is available in two editions: SPSS Modeler Professional and SPSS Modeler Premium. See the topic " IBM SPSS Modeler Editions" on page 2 for more information.

IBM SPSS Modeler Server

SPSS Modeler uses a client/server architecture to distribute requests for resource-intensive operations to powerful server software, resulting in faster performance on larger data sets.

SPSS Modeler Server is a separately-licensed product that runs continually in distributed analysis mode on a server host in conjunction with one or more IBM SPSS Modeler installations. In this way, SPSS Modeler Server provides superior performance on large data sets because memory-intensive operations can be done on the server without downloading data to the client computer. IBM SPSS Modeler Server also provides support for SQL optimization and in-database modeling capabilities, delivering further benefits in performance and automation.

IBM SPSS Modeler Administration Console

The Modeler Administration Console is a graphical user interface for managing many of the SPSS Modeler Server configuration options, which are also configurable by means of an options file. The console is included in IBM SPSS Deployment Manager, can be used to monitor and configure your SPSS Modeler Server installations, and is available free-of-charge to current SPSS Modeler Server customers. The application can be installed only on Windows computers; however, it can administer a server installed on any supported platform.

IBM SPSS Modeler Batch

While data mining is usually an interactive process, it is also possible to run SPSS Modeler from a command line, without the need for the graphical user interface. For example, you might have long-running or repetitive tasks that you want to perform with no user intervention. SPSS Modeler Batch is a special version of the product that provides support for the complete analytical capabilities of SPSS Modeler without access to the regular user interface. SPSS Modeler Server is required to use SPSS Modeler Batch.

IBM SPSS Modeler Solution Publisher

SPSS Modeler Solution Publisher is a tool that enables you to create a packaged version of an SPSS Modeler stream that can be run by an external runtime engine or embedded in an external application. In this way, you can publish and deploy complete SPSS Modeler streams for use in environments that do not have SPSS Modeler installed. SPSS Modeler Solution Publisher is distributed as part of the IBM SPSS Collaboration and Deployment Services - Scoring service, for which a separate license is required. With this license, you receive SPSS Modeler Solution Publisher Runtime, which enables you to execute the published streams.

For more information about SPSS Modeler Solution Publisher, see the IBM SPSS Collaboration and Deployment Services documentation. The IBM SPSS Collaboration and Deployment Services IBM Documentation contains sections called "IBM SPSS Modeler Solution Publisher" and "IBM SPSS Analytics Toolkit."

IBM SPSS Modeler Server Adapters for IBM SPSS Collaboration and Deployment Services

A number of adapters for IBM SPSS Collaboration and Deployment Services are available that enable SPSS Modeler and SPSS Modeler Server to interact with an IBM SPSS Collaboration and Deployment Services repository. In this way, an SPSS Modeler stream deployed to the repository can be shared by multiple users, or accessed from the thin-client application IBM SPSS Modeler Advantage. You install the adapter on the system that hosts the repository.

IBM SPSS Modeler Editions

SPSS Modeler is available in the following editions.

SPSS Modeler Professional

SPSS Modeler Professional provides all the tools you need to work with most types of structured data, such as behaviors and interactions tracked in CRM systems, demographics, purchasing behavior and sales data.

SPSS Modeler Premium

SPSS Modeler Premium is a separately-licensed product that extends SPSS Modeler Professional to work with specialized data and with unstructured text data. SPSS Modeler Premium includes IBM SPSS Modeler Text Analytics:

IBM SPSS Modeler Text Analytics uses advanced linguistic technologies and Natural Language Processing (NLP) to rapidly process a large variety of unstructured text data, extract and organize the key concepts, and group these concepts into categories. Extracted concepts and categories can be combined with existing structured data, such as demographics, and applied to modeling using the full suite of IBM SPSS Modeler data mining tools to yield better and more focused decisions.

IBM SPSS Modeler Subscription

IBM SPSS Modeler Subscription provides all the same predictive analytics capabilities as the traditional IBM SPSS Modeler client. With the Subscription edition, you can download product updates regularly.

Documentation

Documentation is available from the **Help** menu in SPSS Modeler. This opens the online IBM Documentation, which is always available outside the product.

The complete documentation for each product (including installation instructions) is also available in PDF format. See the following support page: **SPSS Modeler 18.6 documentation**.

SPSS Modeler Professional Documentation

The SPSS Modeler Professional documentation suite (excluding installation instructions) is as follows.

- **IBM SPSS Modeler User's Guide.** General introduction to using SPSS Modeler, including how to build data streams, handle missing values, build CLEM expressions, work with projects and reports, and package streams for deployment to IBM SPSS Collaboration and Deployment Services or IBM SPSS Modeler Advantage.
- **IBM SPSS Modeler Source, Process, and Output Nodes.** Descriptions of all the nodes used to read, process, and output data in different formats. Effectively this means all nodes other than modeling nodes.
- **IBM SPSS Modeler Modeling Nodes.** Descriptions of all the nodes used to create data mining models. IBM SPSS Modeler offers a variety of modeling methods taken from machine learning, artificial intelligence, and statistics.
- **IBM SPSS Modeler Applications Guide.** The examples in this guide provide brief, targeted introductions to specific modeling methods and techniques. An online version of this guide is also available from the Help menu. See the topic "Application examples" on page 4 for more information.
- **IBM SPSS Modeler Python Scripting and Automation.** Information on automating the system through Python scripting, including the properties that can be used to manipulate nodes and streams.
- **IBM SPSS Modeler Deployment Guide.** Information on running IBM SPSS Modeler streams as steps in processing jobs under IBM SPSS Deployment Manager.
- **IBM SPSS Modeler In-Database Mining Guide.** Information on how to use the power of your database to improve performance and extend the range of analytical capabilities through third-party algorithms.
- **IBM SPSS Modeler Server Administration and Performance Guide.** Information on how to configure and administer IBM SPSS Modeler Server.
- **IBM SPSS Deployment Manager User Guide.** Information on using the administration console user interface included in the Deployment Manager application for monitoring and configuring IBM SPSS Modeler Server.
- **IBM SPSS Modeler CRISP-DM Guide.** Step-by-step guide to using the CRISP-DM methodology for data mining with SPSS Modeler.
- **IBM SPSS Modeler Batch User's Guide.** Complete guide to using IBM SPSS Modeler in batch mode, including details of batch mode execution and command-line arguments. This guide is available in PDF format only.

SPSS Modeler Premium Documentation

The SPSS Modeler Premium documentation suite (excluding installation instructions) is as follows.

• **SPSS Modeler Text Analytics User's Guide.** Information on using text analytics with SPSS Modeler, covering the text mining nodes, interactive workbench, templates, and other resources.

Application examples

While the data mining tools in SPSS Modeler can help solve a wide variety of business and organizational problems, the application examples provide brief, targeted introductions to specific modeling methods and techniques. The data sets used here are much smaller than the enormous data stores managed by some data miners, but the concepts and methods that are involved are scalable to real-world applications.

To access the examples, click **Application Examples** on the Help menu in SPSS Modeler.

The data files and sample streams are installed in the Demos folder under the product installation directory. For more information, see "Demos Folder" on page 4.

Database modeling examples. See the examples in the *IBM SPSS Modeler In-Database Mining Guide*.

Scripting examples. See the examples in the *IBM SPSS Modeler Scripting and Automation Guide*.

Demos Folder

The data files and sample streams that are used with the application examples are installed in the Demos folder under the product installation directory (for example: `C:\Program Files\IBM\SPSS\Modeler\<version>\Demos`). This folder can also be accessed from the IBM SPSS Modeler program group on the Windows Start menu, or by clicking Demos on the list of recent directories in the **File** > **Open Stream** dialog box.

License tracking

When you use SPSS Modeler, license usage is tracked and logged at regular intervals. The license metrics that are logged are *AUTHORIZED_USER* and *CONCURRENT_USER*, and the type of metric that is logged depends on the type of license that you have for SPSS Modeler.

The log files that are produced can be processed by the IBM License Metric Tool, from which you can generate license usage reports.

The license log files are created in the same directory where SPSS Modeler Client log files are recorded (by default, `%ALLUSERSPROFILE%/IBM/SPSS/Modeler/<version>/log`).

Chapter 2. What's new in version 18.6.0?

IBM SPSS Modeler adds the following new features and enhancements in this release.

- Connectivity to watsonx.data through Presto SQL query engine with SQL pushback support.
- Support for R 4.4.0 capabilities in Extension Import and Extension Export nodes.
- Java runtime is now upgraded to IBM Semeru JRE 17.
- Improved performance for Azure Synapse connectivity with SQL pushback..
- The following new OS versions are supported: Ubuntu 24 and RHEL 9.4.
- Text Analytics is now supported in Japanese.

Release notes

See the Release notes for system requirements and known issues and limitations for using IBM SPSS Modeler.

Chapter 3. Product overview

Getting started

As a data mining application, IBM SPSS Modeler offers a strategic approach to finding useful relationships in large data sets. In contrast to more traditional statistical methods, you do not necessarily need to know what you are looking for when you start. You can explore your data, fitting different models and investigating different relationships, until you find useful information.

Starting IBM SPSS Modeler

To start the application, click:

Start > **[All] Programs** > **IBM SPSS Modeler <version>** > **IBM SPSS Modeler <version>**

The main window is displayed after a few seconds.
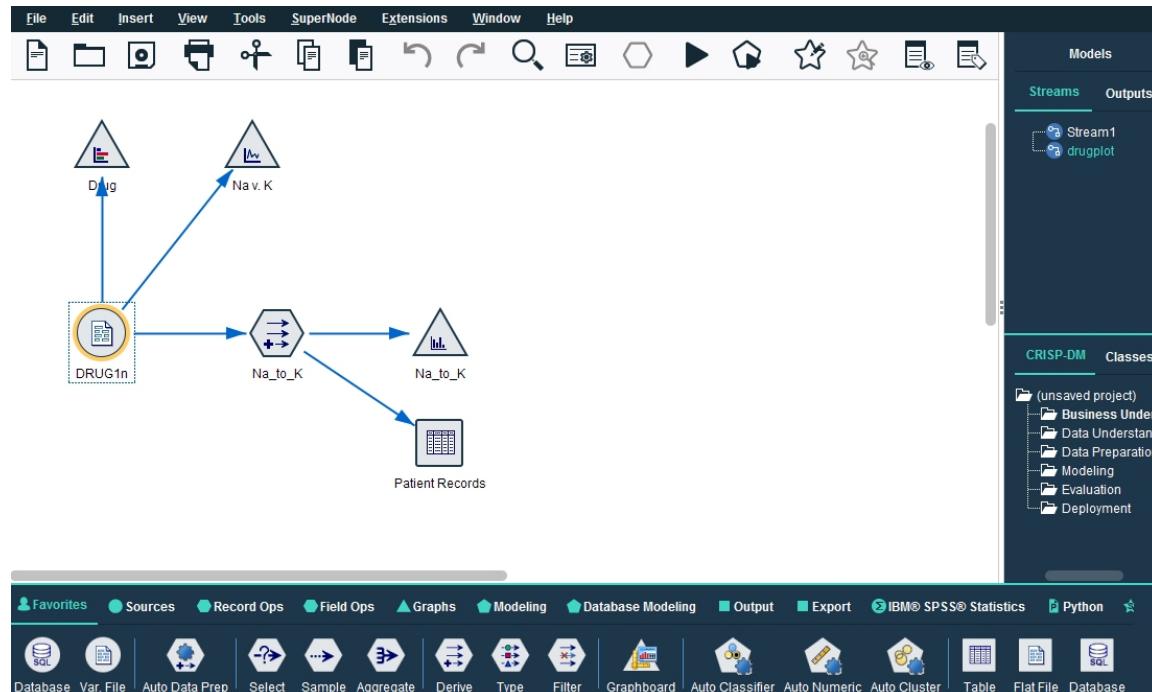
Figure 1. IBM SPSS Modeler main application window

Launching from the Command Line

You can use the command line of your operating system to start IBM SPSS Modeler as follows.

Microsoft Windows

1. On a computer where IBM SPSS Modeler is installed, open a DOS, or command-prompt, window.
2. Switch to the installation path for IBM SPSS Modeler (for example, `[Installpath]\Program Files\IBM\SPSS\Modeler\18.6\bin`).
3. To start the IBM SPSS Modeler interface in interactive mode, type the `modelerclient` command followed by the required arguments; for example:

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

You can use the available arguments (flags) to connect to a server, load streams, run scripts, or specify other parameters as needed.

Mac OS

1. Locate the Mac OS command path for IBM SPSS Modeler (for example, `[Installpath]/Applications/IBM/SPSS/Modeler/18.6/IBM SPSS Modeler.app/Contents/MacOS`).
2. To start the IBM SPSS Modeler interface in interactive mode, run the modeler command followed by the required arguments; for example:

```
./modeler -stream report.str -execute
```

Connecting to IBM SPSS Modeler Server

IBM SPSS Modeler can be run as a standalone application, or as a client connected to IBM SPSS Modeler Server directly or to an IBM SPSS Modeler Server or server cluster through the Coordinator of Processes plug-in from IBM SPSS Collaboration and Deployment Services. The current connection status is displayed at the bottom left of the IBM SPSS Modeler window.

Whenever you want to connect to a server, you can manually enter the server name to which you want to connect or select a name that you have previously defined. However, if you have IBM SPSS Collaboration and Deployment Services, you can search through a list of servers or server clusters from the Server Login dialog box. The ability to browse through the Statistics services running on a network is made available through the Coordinator of Processes.

To Connect to a Server

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens. Alternatively, double-click the connection status area of the IBM SPSS Modeler window.
2. Using the dialog box, specify options to connect to the local server computer or select a connection from the table.

   - Click **Add** or **Edit** to add or edit a connection. See the topic "Adding and Editing the IBM SPSS Modeler Server Connection" on page 9 for more information.
   - Click **Search** to access a server or server cluster in the Coordinator of Processes. See the topic "Searching for Servers in IBM SPSS Collaboration and Deployment Services " on page 9 for more information.

   Server table. This table contains the set of defined server connections. The table displays the default connection, server name, description, and port number. You can manually add a new connection, as well as select or search for an existing connection. To set a particular server as the default connection, select the check box in the Default column in the table for the connection.

   Default data path. Specify a path used for data on the server computer. Click the ellipsis button **(...)** to browse to the required location.

   Set Credentials. Leave this box unchecked to enable the **single sign-on** feature, which attempts to log you in to the server using your local computer username and password details. If single sign-on is not possible, or if you check this box to disable single sign-on (for example, to log in to an administrator account), the following fields are enabled for you to enter your credentials.

   User ID. Enter the user name with which to log on to the server.

   Password. Enter the password associated with the specified user name.

   Domain. Specify the domain used to log on to the server. A domain name is required only when the server computer is in a different Windows domain than the client computer.
3. Click **OK** to complete the connection.

To Disconnect from a Server

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens. Alternatively, double-click the connection status area of the IBM SPSS Modeler window.
2. In the dialog box, select the Local Server and click **OK**.

Adding and Editing the IBM SPSS Modeler Server Connection

You can manually edit or add a server connection in the Server Login dialog box. By clicking Add, you can access an empty Add/Edit Server dialog box in which you can enter server connection details. By selecting an existing connection and clicking Edit in the Server Login dialog box, the Add/Edit Server dialog box opens with the details for that connection so that you can make any changes.

Note: You cannot edit a server connection that was added from IBM SPSS Collaboration and Deployment Services, since the name, port, and other details are defined in IBM SPSS Collaboration and Deployment Services. Best practice dictates that the same ports should be used to communicate with both IBM SPSS Collaboration and Deployment Services and SPSS Modeler Client. These can be set as `max_server_port` and `min_server_port` in the options.cfg file.

To Add Server Connections

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
2. In this dialog box, click **Add**. The Server Login Add/Edit Server dialog box opens.
3. Enter the server connection details and click **OK** to save the connection and return to the Server Login dialog box.

- **Server.** Specify an available server or select one from the list. The server computer can be identified by an alphanumeric name (for example, *myserver*) or an IP address assigned to the server computer (for example, 202.123.456.78).
- **Port.** Give the port number on which the server is listening. If the default does not work, ask your system administrator for the correct port number.
- **Description.** Enter an optional description for this server connection.
- **Ensure secure connection (use SSL).** Specifies whether an SSL (**Secure Sockets Layer**) connection should be used. SSL is a commonly used protocol for securing data sent over a network. To use this feature, SSL must be enabled on the server hosting IBM SPSS Modeler Server. If necessary, contact your local administrator for details.

To Edit Server Connections

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
2. In this dialog box, select the connection you want to edit and then click **Edit**. The Server Login Add/Edit Server dialog box opens.
3. Change the server connection details and click **OK** to save the changes and return to the Server Login dialog box.

Searching for Servers in IBM SPSS Collaboration and Deployment Services

Instead of entering a server connection manually, you can select a server or server cluster available on the network through the Coordinator of Processes, available in IBM SPSS Collaboration and Deployment Services. A server cluster is a group of servers from which the Coordinator of Processes determines the server best suited to respond to a processing request.

Although you can manually add servers in the Server Login dialog box, searching for available servers lets you connect to servers without requiring that you know the correct server name and port number. This information is automatically provided. However, you still need the correct logon information, such as username, domain, and password.

Note: If you do not have access to the Coordinator of Processes capability, you can still manually enter the server name to which you want to connect or select a name that you have previously defined. See the topic "Adding and Editing the IBM SPSS Modeler Server Connection" on page 9 for more information.

To search for servers and clusters

1. On the Tools menu, click **Server Login**. The Server Login dialog box opens.
2. In this dialog box, click **Search** to open the Search for Servers dialog box. If you are not logged on to IBM SPSS Collaboration and Deployment Services when you attempt to browse the Coordinator of Processes, you will be prompted to do so.
3. Select the server or server cluster from the list.
4. Click **OK** to close the dialog box and add this connection to the table in the Server Login dialog box.

Connecting to Analytic Server

If you have multiple Analytic Servers available, you can use the Analytic Server Connection dialog to define more than one server for use in IBM SPSS Modeler. Your administrator may have already set up a default Analytic Server in the `<Modeler_install_path>/config/options.cfg` file. But you can also use other available servers after defining them. For example, when using the Analytic Server Source and Export nodes, you may want to use different Analytic Server connections in different branches of a stream so that when each branch runs it uses its own Analytic Server and no data will be pulled to the IBM SPSS Modeler Server. Note that if a branch contains more than one Analytic Server connection, the data will be pulled from the Analytic Servers to the IBM SPSS Modeler Server.

To create a new Analytic Server connection, go to **Tools** > **Analytic Server Connections** and provide the required information in the following sections of the dialog.

Connection

URL. Type the URL for the Analytic Server in the format `https://hostname:port/contextroot`, where `hostname` is the IP address or host name of the Analytic Server, `port` is its port number, and `contextroot` is the context root of the Analytic Server.

Tenant. Type the name of the tenant that the IBM SPSS Modeler Server is a member of. Contact your administrator if you don't know the tenant.

Authentication

Mode. Select from the following authentication modes.

- **Username and password** requires you to enter the username and password.
- **Stored credential** requires you to select a credential from the IBM SPSS Collaboration and Deployment Services Repository.
- **Kerberos** requires you to enter the service principal name and the config file path. Contact your administrator if you don't know this information.

Username. Type the Analytic Server username.

Reams. Select the realm to use for the Analytic Server connection.

Password. Type the Analytic Server password.

Connect. Click **Connect** to test the new connection.

Connections

After specifying the information above and clicking **Connect**, the connection will be added to this Connections table. If you need to remove a connection, select it and click **Remove**.

If your administrator defined a default Analytic Server connection in the `options.cfg` file, you can click **Add default connection** to add it to your available connections also. You will be prompted for the username and password.

Changing the temp directory

Some operations that are done by IBM SPSS Modeler Server might require temporary files to be created. By default, the IBM SPSS Modeler uses the system temporary directory to create temp files. You can alter the location of the temporary directory using the following steps.

1. Create a new directory called `spss` and subdirectory called `servertemp`.
2. Edit `options.cfg`, located in the `/config` directory of your IBM SPSS Modeler installation directory. Edit the `temp_directory` parameter in this file to read: `temp_directory, "C:/spss/servertemp"`.
3. Restart the IBM SPSS Modeler Server service. You can do this by clicking the **Services** tab on your Windows Control Panel. Stop the service and then start it to activate the changes you made. Restarting the machine also restarts the service.

All temp files now get written to this new directory.

Note: Forward slashes must be used.

Temp directory for data view

For data view service, complete the following steps to configure the temporary directory.

1. Create the temp directory `D:/SPSSTemp`.
2. The data view temp files are controlled by `-Djava.io.tmpdir=D:/SPSSTemp`. Add this option to the start script for data view in the following file: `${ModelerInstallDir}/dataview/start_graph_micro_service.sh`.
3. Start the IBM SPSS Modeler.

Files now get written to `D:/SPSSTemp`.

Starting Multiple IBM SPSS Modeler Sessions

If you need to launch more than one IBM SPSS Modeler session at a time, you must make some changes to your IBM SPSS Modeler and Windows settings. For example, you may need to do this if you have two separate server licenses and want to run two streams against two different servers from the same client machine.

To enable multiple IBM SPSS Modeler sessions:

1. Click:

   Start > **[All] Programs** > **IBM SPSS Modeler**
2. On the IBM SPSS Modeler shortcut (the one with the icon), right-click and select **Properties**.
3. In the **Target** text box, add `-noshare` to the end of the string.
4. In Windows Explorer, select:

   Tools > **Folder Options...**
5. On the File Types tab, select the IBM SPSS Modeler Stream option and click **Advanced**.
6. In the Edit File Type dialog box, select Open with IBM SPSS Modeler and click **Edit**.
7. In the **Application used to perform action** text box, add `-noshare` before the **-stream** argument.

IBM SPSS Modeler Interface at a Glance

At each point in the data mining process, the easy-to-use IBM SPSS Modeler interface invites your specific business expertise. Modeling algorithms, such as prediction, classification, segmentation, and association detection, ensure powerful and accurate models. Model results can easily be deployed and read into databases, IBM SPSS Statistics, and a wide variety of other applications.

Working with IBM SPSS Modeler is a three-step process of working with data.

- First, you read data into IBM SPSS Modeler.
- Next, you run the data through a series of manipulations.
- Finally, you send the data to a destination.

This sequence of operations is known as a **data stream** because the data flows record by record from the source through each manipulation and, finally, to the destination--either a model or type of data output.
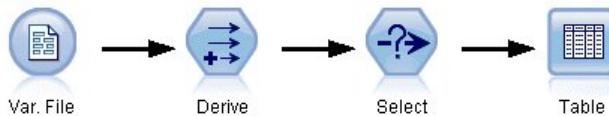
Figure 2. A simple stream

IBM SPSS Modeler Stream Canvas

The stream canvas is the largest area of the IBM SPSS Modeler window and is where you will build and manipulate data streams.
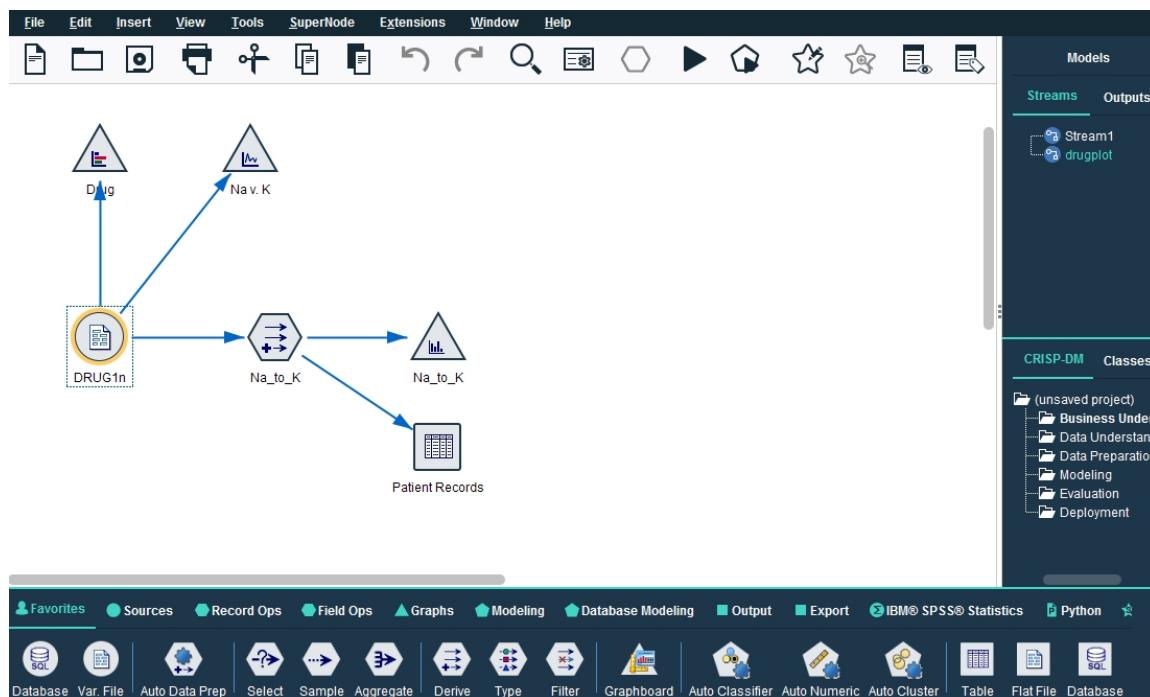
Figure 3. IBM SPSS Modeler workspace (default view)

Streams are created by drawing diagrams of data operations relevant to your business on the main canvas in the interface. Each operation is represented by an icon or **node**, and the nodes are linked together in a **stream** representing the flow of data through each operation.

You can work with multiple streams at one time in IBM SPSS Modeler, either in the same stream canvas or by opening a new stream canvas. During a session, streams are stored in the Streams manager, at the upper right of the IBM SPSS Modeler window.

Note: If using a MacBook with the built-in trackpad's **Force Click and haptic feedback** setting enabled, dragging and dropping nodes from the nodes palette to the stream canvas can result in duplicate nodes being added to the canvas. To avoid this issue, we recommend disabling the **Force Click and haptic feedback** trackpad system preference.

Nodes palette

Most of the data and modeling tools in SPSS Modeler are available from the *Nodes Palette*, across the bottom of the window below the stream canvas.

For example, the **Record Ops** palette tab contains nodes that you can use to perform operations on the data *records*, such as selecting, merging, and appending.

To add nodes to the canvas, double-click icons from the Nodes Palette or drag them onto the canvas. You then connect them to create a *stream*, representing the flow of data.

Figure 4. Record Ops tab on the nodes palette

Each palette tab contains a collection of related nodes used for different phases of stream operations, such as:

- **Sources** nodes bring data into SPSS Modeler.
- **Record Ops** nodes perform operations on data *records*, such as selecting, merging, and appending.
- **Field Ops** nodes perform operations on data *fields*, such as filtering, deriving new fields, and determining the measurement level for given fields.
- **Graphs** nodes graphically display data before and after modeling. Graphs include plots, histograms, web nodes, and evaluation charts.
- **Modeling** nodes use the modeling algorithms available in SPSS Modeler, such as neural nets, decision trees, clustering algorithms, and data sequencing.
- **Database Modeling** nodes use the modeling algorithms available in Microsoft SQL Server, IBM Db2, and Oracle and Netezza databases.
- **Output** nodes produce various output for data, charts, and model results that can be viewed in SPSS Modeler.
- **Export** nodes produce various output that can be viewed in external applications, such as IBM SPSS Data Collection or Excel.
- **IBM SPSS Statistics** nodes import data from, or export data to, IBM SPSS Statistics, as well as running IBM SPSS Statistics procedures.
- **Python** nodes can be used to run Python algorithms.
- **Spark** nodes can be used to run Spark algorithms.

As you become more familiar with SPSS Modeler, you can customize the palette contents for your own use.

On the left side of the Nodes Palette, you can filter the nodes that display by selecting Supervised, Association, or Segmentation.

Located below the Nodes Palette, a report pane provides feedback on the progress of various operations, such as when data is being read into the data stream. Also located below the Nodes Palette, a status pane provides information on what the application is currently doing, as well as indications of when user feedback is required.

Note: If using a MacBook with the built-in trackpad's **Force Click and haptic feedback** setting enabled, dragging and dropping nodes from the nodes palette to the stream canvas can result in duplicate nodes being added to the canvas. To avoid this issue, we recommend disabling the **Force Click and haptic feedback** trackpad system preference.

IBM SPSS Modeler Managers

At the top right of the window is the managers pane. This has three tabs, which are used to manage streams, output and models.

You can use the Streams tab to open, rename, save, and delete the streams created in a session.

Figure 5. Streams tab

Figure 6. Outputs tab

The Outputs tab contains a variety of files, such as graphs and tables, produced by stream operations in IBM SPSS Modeler. You can display, save, rename, and close the tables, graphs, and reports listed on this tab.
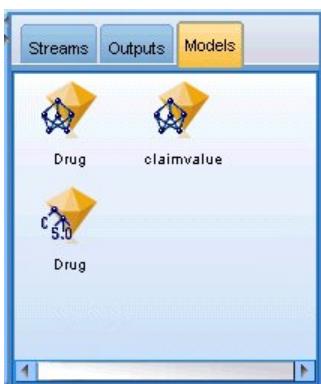
Figure 7. Models tab containing model nuggets

The Models tab is the most powerful of the manager tabs. This tab contains all model **nuggets**, which contain the models generated in IBM SPSS Modeler, for the current session. These models can be browsed directly from the Models tab or added to the stream in the canvas.

IBM SPSS Modeler Projects

On the lower right side of the window is the project pane, used to create and manage data mining **projects** (groups of files related to a data mining task). There are two ways to view projects you create in IBM SPSS Modeler—in the Classes view and the CRISP-DM view.
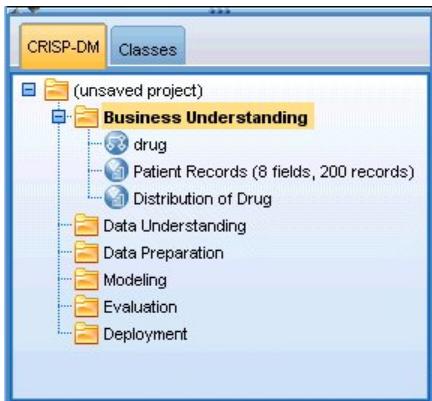
Figure 8. CRISP-DM view

The CRISP-DM tab provides a way to organize projects according to the Cross-Industry Standard Process for Data Mining, an industry-proven, nonproprietary methodology. For both experienced and first-time data miners, using the CRISP-DM tool will help you to better organize and communicate your efforts.
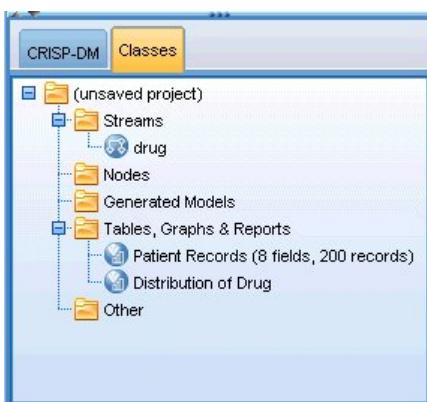
Figure 9. Classes view

The Classes tab provides a way to organize your work in IBM SPSS Modeler categorically—by the types of objects you create. This view is useful when taking inventory of data, streams, and models.

IBM SPSS Modeler Toolbar

At the top of the IBM SPSS Modeler window, you will find a toolbar of icons that provides a number of useful functions. Following are the toolbar buttons and their functions.

	Create new stream		Open stream
	Save stream		Print current stream
	Cut & move to clipboard		Copy to clipboard
	Paste selection		Undo last action

	Redo		Search for nodes
	Edit stream properties		Preview SQL generation
	Run current stream		Run stream selection
	Stop stream (Active only while stream is running)		Add SuperNode
	Zoom in (SuperNodes only)		Zoom out (SuperNodes only)
	No markup in stream		Insert comment
	Hide stream markup (if any)		Show hidden stream markup
	Open stream in IBM SPSS Modeler Advantage		

Stream markup consists of stream comments, model links, and scoring branch indications.

Model links are described in the *IBM SPSS Modeling Nodes* guide.

Customizing the Toolbar

You can change various aspects of the toolbar, such as:

- Whether it is displayed
- Whether the icons have tooltips available
- Whether it uses large or small icons

To turn the toolbar display on and off:

1. On the main menu, click:

    View > **Toolbar** > **Display**

To change the tooltip or icon size settings:

1. On the main menu, click:

    View > **Toolbar** > **Customize**

Click **Show ToolTips** or **Large Buttons** as required.

Customizing the IBM SPSS Modeler window

Using the dividers between various portions of the SPSS Modeler interface, you can resize or close tools to meet your preferences. For example, if you are working with a large stream, you can use the small arrows located on each divider to close the nodes palette, managers pane, and project pane. This maximizes the stream canvas, providing enough work space for large or multiple streams.

Alternatively, on the View menu, click **Nodes Palette**, **Managers**, or **Project** to turn the display of these items on or off.
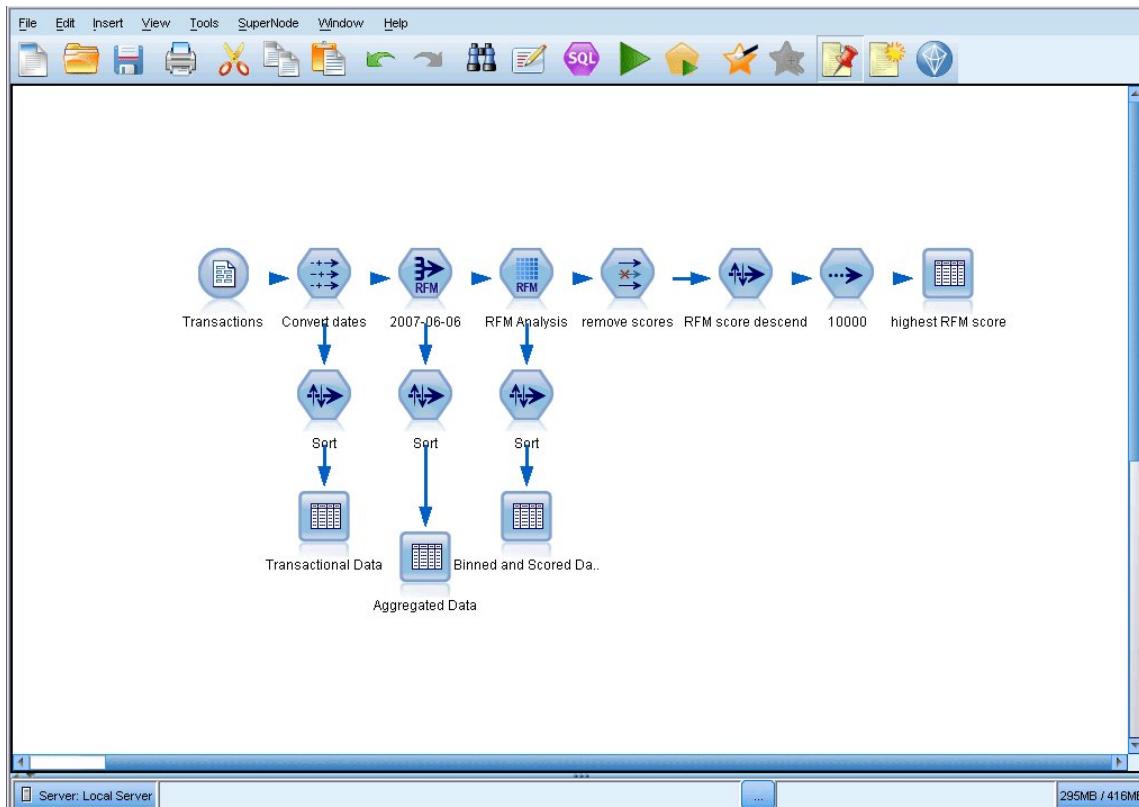
Figure 10. Maximized stream canvas

As an alternative to closing the nodes palette, and the managers and project panes, you can use the stream canvas as a scrollable page by moving vertically and horizontally with the scrollbars at the side and bottom of the SPSS Modeler window.

You can also control the display of screen markup, which consists of stream comments, model links, and scoring branch indications. To turn this display on or off, click:

View > **Stream Markup**

Changing the icon size for a stream

You can change the size of the stream icons in the following ways.

- Through a stream property setting
- Through a pop-up menu in the stream
- Using the keyboard

You can scale the entire stream view to one of a number of sizes between 8% and 200% of the standard icon size.

To scale the entire stream (stream properties method)

1. From the main menu, choose

   Tools > **Stream Properties** > **Options** > **Layout.**
2. Choose the size you want from the Icon Size menu.
3. Click **Apply** to see the result.
4. Click **OK** to save the change.

To scale the entire stream (menu method)

1. Right-click the stream background on the canvas.
2. Choose **Icon Size** and select the size you want.

To scale the entire stream (keyboard method)

1. Press Ctrl + [-] on the main keyboard to zoom out to the next smaller size.
2. Press Ctrl + Shift + [+] on the main keyboard to zoom in to the next larger size.

   Note that this method of zooming in may not work depending on your operating system and keyboard used.

This feature is particularly useful for gaining an overall view of a complex stream. You can also use it to minimize the number of pages needed to print a stream.

Using the Mouse in IBM SPSS Modeler

The most common uses of the mouse in IBM SPSS Modeler include the following:

- **Single-click.** Use either the right or left mouse button to select options from menus, open pop-up menus, and access various other standard controls and options. Click and hold the button to move and drag nodes.
- **Double-click.** Double-click using the left mouse button to place nodes on the stream canvas and edit existing nodes.
- **Middle-click.** Click the middle mouse button and drag the cursor to connect nodes on the stream canvas. Double-click the middle mouse button to disconnect a node. If you do not have a three-button mouse, you can simulate this feature by pressing the Alt key while clicking and dragging the mouse.

Using shortcut keys

Many visual programming operations in IBM SPSS Modeler have shortcut keys associated with them. For example, you can delete a node by clicking the node and pressing the Delete key on your keyboard. Likewise, you can quickly save a stream by pressing the S key while holding down the Ctrl key. Control commands like this one are indicated by a combination of Ctrl and another key--for example, Ctrl+S.

There are a number of shortcut keys used in standard Windows operations, such as Ctrl+X to cut. These shortcuts are supported in IBM SPSS Modeler along with the following application-specific shortcuts.

Note: In some cases, old shortcut keys used in IBM SPSS Modeler conflict with standard Windows shortcut keys. These old shortcuts are supported with the addition of the Alt key. For example, Ctrl+Alt+C can be used to toggle the cache on and off.

Table 1. Supported shortcut keys	
Shortcut Key	**Function**
Ctrl+A	Select all
Ctrl+X	Cut

Table 1. Supported shortcut keys (continued)	
Shortcut Key	**Function**
Ctrl+N	New stream
Ctrl+O	Open stream
Ctrl+P	Print
Ctrl+C	Copy
Ctrl+V	Paste
Ctrl+Z	Undo
Ctrl+Q	Select all nodes downstream of the selected node
Ctrl+W	Deselect all downstream nodes (toggles with Ctrl+Q)
Ctrl+E	Run from selected node
Ctrl+S	Save current stream
Alt+Arrow keys	Move selected nodes on the stream canvas in the direction of the arrow used
Shift+F10	Open the pop-up menu for the selected node

Table 2. Supported shortcuts for old hot keys	
Shortcut Key	**Function**
Ctrl+Alt+D	Duplicate node
Ctrl+Alt+L	Load node
Ctrl+Alt+R	Rename node
Ctrl+Alt+U	Create User Input node
Ctrl+Alt+C	Toggle cache on/off
Ctrl+Alt+F	Flush cache
Ctrl+Alt+X	Expand SuperNode
Ctrl+Alt+Z	Zoom in/zoom out
Delete	Delete node or connection

Printing

The following objects can be printed in IBM SPSS Modeler:

- Stream diagrams
- Graphs
- Tables
- Reports (from the Report node and Project Reports)
- Scripts (from the stream properties, Standalone Script, or SuperNode script dialog boxes)
- Models (Model browsers, dialog box tabs with current focus, tree viewers)
- Annotations (using the Annotations tab for output)

To print an object:

- To print without previewing, click the Print button on the toolbar.

- To set up the page before printing, select **Page Setup** from the File menu.
- To preview before printing, select **Print Preview** from the File menu.
- To view the standard print dialog box with options for selecting printers, and specifying appearance options, select **Print** from the File menu.

Automating IBM SPSS Modeler

Since advanced data mining can be a complex and sometimes lengthy process, IBM SPSS Modeler includes several types of coding and automation support.

- **Control Language for Expression Manipulation** (CLEM) is a language for analyzing and manipulating the data that flows along IBM SPSS Modeler streams. Data miners use CLEM extensively in stream operations to perform tasks as simple as deriving profit from cost and revenue data or as complex as transforming web log data into a set of fields and records with usable information.
- **Scripting** is a powerful tool for automating processes in the user interface. Scripts can perform the same kinds of actions that users perform with a mouse or a keyboard. You can also specify output and manipulate generated models.

Chapter 4. Understanding data mining

Data Mining Overview

Through a variety of techniques, **data mining** identifies nuggets of information in bodies of data. Data mining extracts information in such a way that it can be used in areas such as decision support, prediction, forecasts, and estimation. Data is often voluminous but of low value and with little direct usefulness in its raw form. It is the hidden information in the data that has value.

In data mining, success comes from combining your (or your expert's) knowledge of the data with advanced, active analysis techniques in which the computer identifies the underlying relationships and features in the data. The process of data mining generates models from historical data that are later used for predictions, pattern detection, and more. The technique for building these models is called **machine learning** or **modeling**.

Modeling Techniques

IBM SPSS Modeler includes a number of machine-learning and modeling technologies, which can be roughly grouped according to the types of problems they are intended to solve.

- Predictive modeling methods include decision trees, neural networks, and statistical models.
- Clustering models focus on identifying groups of similar records and labeling the records according to the group to which they belong. Clustering methods include Kohonen, k-means, and TwoStep.
- Association rules associate a particular conclusion (such as the purchase of a particular product) with a set of conditions (the purchase of several other products).
- Screening models can be used to screen data to locate fields and records that are most likely to be of interest in modeling and identify outliers that may not fit known patterns. Available methods include feature selection and anomaly detection.

Data Manipulation and Discovery

IBM SPSS Modeler also includes many facilities that let you apply your expertise to the data:

- **Data manipulation.** Constructs new data items derived from existing ones and breaks down the data into meaningful subsets. Data from a variety of sources can be merged and filtered.
- **Browsing and visualization.** Displays aspects of the data using the Data Audit node to perform an initial audit including graphs and statistics. Advanced visualization includes interactive graphics, which can be exported for inclusion in project reports.
- **Statistics.** Confirms suspected relationships between variables in the data. Statistics from IBM SPSS Statistics can also be used within IBM SPSS Modeler.
- **Hypothesis testing.** Constructs models of how the data behaves and verifies these models.

Typically, you will use these facilities to identify a promising set of attributes in the data. These attributes can then be fed to the modeling techniques, which will attempt to identify underlying rules and relationships.

Typical Applications

Typical applications of data mining techniques include the following:

Direct mail. Determine which demographic groups have the highest response rate. Use this information to maximize the response to future mailings.

Credit scoring. Use an individual's credit history to make credit decisions.

Human resources. Understand past hiring practices and create decision rules to streamline the hiring process.

Medical research. Create decision rules that suggest appropriate procedures based on medical evidence.

Market analysis. Determine which variables, such as geography, price, and customer characteristics, are associated with sales.

Quality control. Analyze data from product manufacturing and identify variables determining product defects.

Policy studies. Use survey data to formulate policy by applying decision rules to select the most important variables.

Health care. User surveys and clinical data can be combined to discover variables that contribute to health.

Terminology

The terms **attribute**, **field**, and **variable** refer to a single data item common to all cases under consideration. A collection of attribute values that refers to a specific case is called a **record**, an **example**, or a **case**.

Assessing the Data

Data mining is not likely to be fruitful unless the data you want to use meets certain criteria. The following sections present some of the aspects of the data and its application that you should consider.

Ensure that the data is available

This may seem obvious, but be aware that although data might be available, it may not be in a form that can be used easily. IBM SPSS Modeler can import data from databases (through ODBC) or from files. The data, however, might be held in some other form on a machine that cannot be directly accessed. It will need to be downloaded or dumped in a suitable form before it can be used. It might be scattered among different databases and sources and need to be pulled together. It may not even be online. If it exists only on paper, data entry will be required before you can begin data mining.

Check whether the data covers the relevant attributes

The object of data mining is to identify relevant attributes, so including this check may seem odd at first. It is very useful, however, to look at what data is available and to try to identify the likely relevant factors that are not recorded. In trying to predict ice cream sales, for example, you may have a lot of information about retail outlets or sales history, but you may not have weather and temperature information, which is likely to play a significant role. Missing attributes do not necessarily mean that data mining will not produce useful results, but they can limit the accuracy of resulting predictions.

A quick way of assessing the situation is to perform a comprehensive audit of your data. Before moving on, consider attaching a Data Audit node to your data source and running it to generate a full report.

Beware of noisy data

Data often contains errors or may contain subjective, and therefore variable, judgments. These phenomena are collectively referred to as **noise**. Sometimes noise in data is normal. There may well be underlying rules, but they may not hold for 100% of the cases.

Typically, the more noise there is in data, the more difficult it is to get accurate results. However, 's The machine-learning methods in IBM SPSS Modeler are able to handle noisy data and have been used successfully on data sets containing almost 50% noise.

Ensure that there is sufficient data

In data mining, it is not necessarily the size of a data set that is important. The representativeness of the data set is far more significant, together with its coverage of possible outcomes and combinations of variables.

Typically, the more attributes that are considered, the more records that will be needed to give representative coverage.

If the data is representative and there are general underlying rules, it may well be that a data sample of a few thousand (or even a few hundred) records will give equally good results as a million, and you will get the results more quickly.

Seek out the experts on the data

In many cases, you will be working on your own data and will therefore be highly familiar with its content and meaning. However, if you are working on data for another department of your organization or for a client, it is highly desirable that you have access to experts who know the data. They can guide you in the identification of relevant attributes and can help to interpret the results of data mining, distinguishing the true nuggets of information from "fool's gold," or artifacts caused by anomalies in the data sets.

A Strategy for Data Mining

As with most business endeavors, data mining is much more effective if done in a planned, systematic way. Even with cutting-edge data mining tools, such as IBM SPSS Modeler, the majority of the work in data mining requires a knowledgeable business analyst to keep the process on track. To guide your planning, answer the following questions:

- What substantive problem do you want to solve?
- What data sources are available, and what parts of the data are relevant to the current problem?
- What kind of preprocessing and data cleaning do you need to do before you start mining the data?
- What data mining technique(s) will you use?
- How will you evaluate the results of the data mining analysis?
- How will you get the most out of the information you obtained from data mining?

The typical data mining process can become complicated very quickly. There is a lot to keep track of--complex business problems, multiple data sources, varying data quality across data sources, an array of data mining techniques, different ways of measuring data mining success, and so on.

To stay on track, it helps to have an explicitly defined process model for data mining. The process model helps you answer the questions listed earlier in this section, and makes sure the important points are addressed. It serves as a data mining road map so that you will not lose your way as you dig into the complexities of your data.

The data mining process suggested for use with SPSS Modeler is the Cross-Industry Standard Process for Data Mining (CRISP-DM). As you can tell from the name, this model is designed as a general model that can be applied to a wide variety of industries and business problems.

The CRISP-DM Process Model

The general CRISP-DM process model includes six phases that address the main issues in data mining. The six phases fit together in a cyclical process designed to incorporate data mining into your larger business practices.

The six phases include:

- **Business understanding.** This is perhaps the most important phase of data mining. Business understanding includes determining business objectives, assessing the situation, determining data mining goals, and producing a project plan.
- **Data understanding.** Data provides the "raw materials" of data mining. This phase addresses the need to understand what your data resources are and the characteristics of those resources. It includes collecting initial data, describing data, exploring data, and verifying data quality. The Data Audit node available from the Output nodes palette is an indispensable tool for data understanding.
- **Data preparation.** After cataloging your data resources, you will need to prepare your data for mining. Preparations include selecting, cleaning, constructing, integrating, and formatting data.

- **Modeling.** This is, of course, the flashy part of data mining, where sophisticated analysis methods are used to extract information from the data. This phase involves selecting modeling techniques, generating test designs, and building and assessing models.
- **Evaluation.** Once you have chosen your models, you are ready to evaluate how the data mining results can help you to achieve your business objectives. Elements of this phase include evaluating results, reviewing the data mining process, and determining the next steps.
- **Deployment.** Now that you have invested all of this effort, it is time to reap the benefits. This phase focuses on integrating your new knowledge into your everyday business processes to solve your original business problem. This phase includes plan deployment, monitoring and maintenance, producing a final report, and reviewing the project.

There are some key points in this process model. First, while there is a general tendency for the process to flow through the steps in the order outlined in the previous paragraphs, there are also a number of places where the phases influence each other in a nonlinear way. For example, data preparation usually precedes modeling. However, decisions made and information gathered during the modeling phase can often lead you to rethink parts of the data preparation phase, which can then present new modeling issues. The two phases feed back on each other until both phases have been resolved adequately. Similarly, the evaluation phase can lead you to reevaluate your original business understanding, and you may decide that you have been trying to answer the wrong question. At this point, you can revise your business understanding and proceed through the rest of the process again with a better target in mind.

The second key point is the iterative nature of data mining. You will rarely, if ever, simply plan a data mining project, complete it, and then pack up your data and go home. Data mining to address your customers' demands is an ongoing endeavor. The knowledge gained from one cycle of data mining will almost invariably lead to new questions, new issues, and new opportunities to identify and meet your customers' needs. Those new questions, issues, and opportunities can usually be addressed by mining your data once again. This process of mining and identifying new opportunities should become part of the way you think about your business and a cornerstone of your overall business strategy.

This introduction provides only a brief overview of the CRISP-DM process model. For complete details on the model, consult the following resources:

- The *CRISP-DM Guide*, which can be accessed along with other documentation from the *Documentation* folder on the installation disk.
- The CRISP-DM Help system, available from the Start menu or by clicking **CRISP-DM Help** on the Help menu in IBM SPSS Modeler.

Types of models

IBM SPSS Modeler offers a variety of modeling methods taken from machine learning, artificial intelligence, and statistics. The methods available on the Modeling palette allow you to derive new information from your data and to develop predictive models. Each method has certain strengths and is best suited for particular types of problems.

The *IBM SPSS Modeler Applications Guide* provides examples for many of these methods, along with a general introduction to the modeling process. This guide is available as an online tutorial, and also in PDF format. See the topic "Application examples" on page 4 for more information.

Modeling methods are divided into these categories:

- Supervised
- Association
- Segmentation

Supervised Models

Supervised models use the values of one or more **input** fields to predict the value of one or more output, or **target**, fields. Some examples of these techniques are: decision trees (C&R Tree, QUEST, CHAID and C5.0 algorithms), regression (linear, logistic, generalized linear, and Cox regression algorithms), neural networks, support vector machines, and Bayesian networks.

Supervised models help organizations to predict a known result, such as whether a customer will buy or leave or whether a transaction fits a known pattern of fraud. Modeling techniques include machine learning, rule induction, subgroup identification, statistical methods, and multiple model generation.

Supervised nodes

The Auto Classifier node creates and compares a number of different models for binary outcomes (yes or no, churn or do not churn, and so on), allowing you to choose the best approach for a given analysis. A number of modeling algorithms are supported, making it possible to select the methods you want to use, the specific options for each, and the criteria for comparing the results. The node generates a set of models based on the specified options and ranks the best candidates according to the criteria you specify.

The Auto Numeric node estimates and compares models for continuous numeric range outcomes using a number of different methods. The node works in the same manner as the Auto Classifier node, allowing you to choose the algorithms to use and to experiment with multiple combinations of options in a single modeling pass. Supported algorithms include neural networks, C&R Tree, CHAID, linear regression, generalized linear regression, and support vector machines (SVM). Models can be compared based on correlation, relative error, or number of variables used.

The Classification and Regression (C&R) Tree node generates a decision tree that allows you to predict or classify future observations. The method uses recursive partitioning to split the training records into segments by minimizing the impurity at each step, where a node in the tree is considered "pure" if 100% of cases in the node fall into a specific category of the target field. Target and input fields can be numeric ranges or categorical (nominal, ordinal, or flags); all splits are binary (only two subgroups).

The QUEST node provides a binary classification method for building decision trees, designed to reduce the processing time required for large C&R Tree analyses while also reducing the tendency found in classification tree methods to favor inputs that allow more splits. Input fields can be numeric ranges (continuous), but the target field must be categorical. All splits are binary.

The CHAID node generates decision trees using chi-square statistics to identify optimal splits. Unlike the C&R Tree and QUEST nodes, CHAID can generate nonbinary trees, meaning that some splits have more than two branches. Target and input fields can be numeric range (continuous) or categorical. Exhaustive CHAID is a modification of CHAID that does a more thorough job of examining all possible splits but takes longer to compute.

The C5.0 node builds either a decision tree or a rule set. The model works by splitting the sample based on the field that provides the maximum information gain at each level. The target field must be categorical. Multiple splits into more than two subgroups are allowed.

The Decision List node identifies subgroups, or segments, that show a higher or lower likelihood of a given binary outcome relative to the overall population. For example, you might look for customers who are unlikely to churn or are most likely to respond favorably to a campaign. You can incorporate your business knowledge into the model by adding your own custom segments and previewing alternative models side by side to compare the results. Decision List models consist of a list of rules in which each rule has a condition and an outcome. Rules are applied in order, and the first rule that matches determines the outcome.

Linear regression models predict a continuous target based on linear relationships between the target and one or more predictors.

The PCA/Factor node provides powerful data-reduction techniques to reduce the complexity of your data. Principal components analysis (PCA) finds linear combinations of the input fields that do the best job of capturing the variance in the entire set of fields, where the components are orthogonal (perpendicular) to each other. Factor analysis attempts to identify underlying factors that explain the pattern of correlations within a set of observed fields. For both approaches, the goal is to find a small number of derived fields that effectively summarizes the information in the original set of fields.

The Feature Selection node screens input fields for removal based on a set of criteria (such as the percentage of missing values); it then ranks the importance of remaining inputs relative to a specified target. For example, given a data set with hundreds of potential inputs, which are most likely to be useful in modeling patient outcomes?

Discriminant analysis makes more stringent assumptions than logistic regression but can be a valuable alternative or supplement to a logistic regression analysis when those assumptions are met.

Logistic regression is a statistical technique for classifying records based on values of input fields. It is analogous to linear regression but takes a categorical target field instead of a numeric range.

The Generalized Linear model expands the general linear model so that the dependent variable is linearly related to the factors and covariates through a specified link function. Moreover, the model allows for the dependent variable to have a non-normal distribution. It covers the functionality of a wide number of statistical models, including linear regression, logistic regression, loglinear models for count data, and interval-censored survival models.

A generalized linear mixed model (GLMM) extends the linear model so that the target can have a non-normal distribution, is linearly related to the factors and covariates via a specified link function, and so that the observations can be correlated. Generalized linear mixed models cover a wide variety of models, from simple linear regression to complex multilevel models for non-normal longitudinal data.

The Cox regression node enables you to build a survival model for time-to-event data in the presence of censored records. The model produces a survival function that predicts the probability that the event of interest has occurred at a given time (t) for given values of the input variables.

The Support Vector Machine (SVM) node enables you to classify data into one of two groups without overfitting. SVM works well with wide data sets, such as those with a very large number of input fields.

The Bayesian Network node enables you to build a probability model by combining observed and recorded evidence with real-world knowledge to establish the likelihood of occurrences. The node focuses on Tree Augmented Naïve Bayes (TAN) and Markov Blanket networks that are primarily used for classification.

The Self-Learning Response Model (SLRM) node enables you to build a model in which a single new case, or small number of new cases, can be used to reestimate the model without having to retrain the model using all data.

The Time Series node estimates exponential smoothing, univariate Autoregressive Integrated Moving Average (ARIMA), and multivariate ARIMA (or transfer function) models for time series data and produces forecasts of future performance. This Time Series node is similar to the previous Time Series node that was deprecated in SPSS Modeler version 18. However, this newer Time Series node is designed to harness the power of IBM SPSS Analytic Server to process big data, and display the resulting model in the output viewer that was added in SPSS Modeler version 17.

The *k*-Nearest Neighbor (KNN) node associates a new case with the category or value of the *k* objects nearest to it in the predictor space, where *k* is an integer. Similar cases are near each other and dissimilar cases are distant from each other.

The Spatio-Temporal Prediction (STP) node uses data that contains location data, input fields for prediction (predictors), a time field, and a target field. Each location has numerous rows in the data that represent the values of each predictor at each time of measurement. After the data is analyzed, it can be used to predict target values at any location within the shape data that is used in the analysis.

Association Models

Association models find patterns in your data where one or more entities (such as events, purchases, or attributes) are associated with one or more other entities. The models construct rule sets that define these relationships. Here the fields within the data can act as both inputs and targets. You could find these associations manually, but association rule algorithms do so much more quickly, and can explore more complex patterns. Apriori and Carma models are examples of the use of such algorithms. One other type of association model is a sequence detection model, which finds sequential patterns in time-structured data.

Association models are most useful when predicting multiple outcomes—for example, customers who bought product X also bought Y and Z. Association models associate a particular conclusion (such as the decision to buy something) with a set of conditions. The advantage of association rule algorithms over the more standard decision tree algorithms (C5.0 and C&RT) is that associations can exist between any of the attributes. A decision tree algorithm will build rules with only a single conclusion, whereas association algorithms attempt to find many rules, each of which may have a different conclusion.

Association nodes

The Apriori node extracts a set of rules from the data, pulling out the rules with the highest information content. Apriori offers five different methods of selecting rules and uses a sophisticated indexing scheme to process large data sets efficiently. For large problems, Apriori is generally faster to train; it has no arbitrary limit on the number of rules that can be retained, and it can handle rules with up to 32 preconditions. Apriori requires that input and output fields all be categorical but delivers better performance because it is optimized for this type of data.

The CARMA model extracts a set of rules from the data without requiring you to specify input or target fields. In contrast to Apriori the CARMA node offers build settings for rule support (support for both antecedent and consequent) rather than just antecedent support. This means that the rules generated can be used for a wider variety of applications—for example, to find a list of products or services (antecedents) whose consequent is the item that you want to promote this holiday season.

The Sequence node discovers association rules in sequential or time-oriented data. A sequence is a list of item sets that tends to occur in a predictable order. For example, a customer who purchases a razor and aftershave lotion may purchase shaving cream the next time he shops. The Sequence node is based on the CARMA association rules algorithm, which uses an efficient two-pass method for finding sequences.

The Association Rules Node is similar to the Apriori Node; however, unlike Apriori, the Association Rules Node can process list data. In addition, the Association Rules Node can be used with IBM SPSS Analytic Server to process big data and take advantage of faster parallel processing.

Segmentation Models

Segmentation models divide the data into segments, or clusters, of records that have similar patterns of input fields. As they are only interested in the input fields, segmentation models have no concept of output or target fields. Examples of segmentation models are Kohonen networks, K-Means clustering, two-step clustering and anomaly detection.

Segmentation models (also known as "clustering models") are useful in cases where the specific result is unknown (for example, when identifying new patterns of fraud, or when identifying groups of interest in your customer base). Clustering models focus on identifying groups of similar records and labeling the records according to the group to which they belong. This is done without the benefit of prior knowledge about the groups and their characteristics, and it distinguishes clustering models from the other modeling techniques in that there is no predefined output or target field for the model to predict. There are no right or wrong answers for these models. Their value is determined by their ability to capture interesting groupings in the data and provide useful descriptions of those groupings. Clustering models are often used to create clusters or segments that are then used as inputs in subsequent analyses (for example, by segmenting potential customers into homogeneous subgroups).

Segmentation nodes

The Auto Cluster node estimates and compares clustering models, which identify groups of records that have similar characteristics. The node works in the same manner as other automated modeling nodes, allowing you to experiment with multiple combinations of options in a single modeling pass. Models can be compared using basic measures with which to attempt to filter and rank the usefulness of the cluster models, and provide a measure based on the importance of particular fields.

The K-Means node clusters the data set into distinct groups (or clusters). The method defines a fixed number of clusters, iteratively assigns records to clusters, and adjusts the cluster centers until further refinement can no longer improve the model. Instead of trying to predict an outcome, *k*-means uses a process known as unsupervised learning to uncover patterns in the set of input fields.

The Kohonen node generates a type of neural network that can be used to cluster the data set into distinct groups. When the network is fully trained, records that are similar should be close together on the output map, while records that are different will be far apart. You can look at the number of observations captured by each unit in the model nugget to identify the strong units. This may give you a sense of the appropriate number of clusters.

The TwoStep node uses a two-step clustering method. The first step makes a single pass through the data to compress the raw input data into a manageable set of subclusters. The second step uses a hierarchical clustering method to progressively merge the subclusters into larger and larger clusters. TwoStep has the advantage of automatically estimating the optimal number of clusters for the training data. It can handle mixed field types and large data sets efficiently.

The Anomaly Detection node identifies unusual cases, or outliers, that do not conform to patterns of "normal" data. With this node, it is possible to identify outliers even if they do not fit any previously known patterns and even if you are not exactly sure what you are looking for.

In-Database Mining Models

IBM SPSS Modeler supports integration with data mining and modeling tools that are available from database vendors, including Oracle Data Miner and Microsoft Analysis Services. You can build, score, and store models inside the database—all from within the IBM SPSS Modeler application. For full details, see the *IBM SPSS Modeler In-Database Mining Guide*.

IBM SPSS Statistics Models

If you have a copy of IBM SPSS Statistics installed and licensed on your computer, you can access and run certain IBM SPSS Statistics routines from within IBM SPSS Modeler to build and score models.

Data Mining Examples

The best way to learn about data mining in practice is to start with an example. A number of application examples are available in the *IBM SPSS Modeler Applications Guide*, which provides brief, targeted introductions to specific modeling methods and techniques. See the topic "Application examples" on page 4 for more information.

Chapter 5. Building streams

Stream-building overview

Data mining using IBM SPSS Modeler focuses on the process of running data through a series of nodes, referred to as a **stream.** This series of nodes represents operations to be performed on the data, while links between the nodes indicate the direction of data flow. Typically, you use a data stream to read data into IBM SPSS Modeler, run it through a series of manipulations, and then send it to a destination, such as a table or a viewer.

For example, suppose that you want to open a data source, add a new field, select records based on values in the new field, and then display the results in a table. In this case, your data stream would consist of four nodes:

A Variable File node, which you set up to read the data from the data source.

A Derive node, which you use to add the new, calculated field to the data set.

A Select node, which you use to set up selection criteria to exclude records from the data stream.

A Table node, which you use to display the results of your manipulations onscreen.

Building data streams

The unique SPSS Modeler interface lets you mine your data visually by working with diagrams of data streams. At the most basic level, you can build a data stream using the following steps:

- Add nodes to the stream canvas.
- Connect the nodes to form a stream.
- Specify any node or stream options.
- Run the stream.

This section contains more detailed information on working with nodes to create more complex data streams. It also discusses options and settings for nodes and streams. For step-by-step examples of stream building using the data shipped with SPSS Modeler (in the Demos folder of your program installation), see "Application examples" on page 4.

Working with nodes

Nodes are used in IBM SPSS Modeler to help you explore data. Various nodes in the workspace represent different objects and actions. The palette at the bottom of the IBM SPSS Modeler window contains all of the possible nodes used in stream building.

There are several types of nodes. **Source nodes** bring data into the stream, and are located on the Sources tab of the nodes palette. **Process nodes** perform operations on individual data records and fields, and can be found in the Record Ops and Field Ops tabs of the palette. **Output nodes** produce a variety

of output for data, charts and model results, and are included on the Graphs, Output and Export tabs of the nodes palette. **Modeling nodes** use statistical algorithms to create model nuggets, and are located on the Modeling tab, and (if activated) the Database Modeling tab, of the nodes palette. See the topic "Nodes palette" on page 13 for more information.

You connect the nodes to form streams which, when run, let you visualize relationships and draw conclusions. Streams are like scripts--you can save them and reuse them with different data files.

A runnable node that processes stream data is known as a **terminal node**. A modeling or output node is a terminal node if it is located at the end of a stream or stream branch. You cannot connect further nodes to a terminal node.

Note: You can customize the Nodes palette. See the topic "Customizing the Nodes Palette" on page 233 for more information.

Adding Nodes to a Stream

There are several ways to add nodes to a stream from the nodes palette:

- Double-click a node on the palette. *Note*: Double-clicking a node automatically connects it to the current stream. See the topic "Connecting Nodes in a Stream" on page 32 for more information.
- Drag and drop a node from the palette to the stream canvas.
- Click a node on the palette, and then click the stream canvas.
- Select an appropriate option from the Insert menu of IBM SPSS Modeler.

Once you have added a node to the stream canvas, double-click the node to display its dialog box. The available options depend on the type of node that you are adding. For information about specific controls within the dialog box, click its **Help** button.

Removing Nodes

To remove a node from the data stream, click it and either press the Delete key, or right-click and select **Delete** from the menu.

Connecting Nodes in a Stream

Nodes added to the stream canvas do not form a data stream until they have been connected. Connections between the nodes indicate the direction of the data as it flows from one operation to the next. There are a number of ways to connect nodes to form a stream: double-clicking, using the middle mouse button, or manually.

To Add and Connect Nodes by Double-Clicking

The simplest way to form a stream is to double-click nodes on the palette. This method automatically connects the new node to the selected node on the stream canvas. For example, if the canvas contains a Database node, you can select this node and then double-click the next node from the palette, such as a Derive node. This action automatically connects the Derive node to the existing Database node. You can repeat this process until you have reached a terminal node, such as a Histogram or Table node, at which point any new nodes will be connected to the last non-terminal node upstream.

To Connect Nodes Using the Middle Mouse Button

On the stream canvas, you can click and drag from one node to another using the middle mouse button. (If your mouse does not have a middle button, you can simulate this by pressing the Alt key while dragging with the mouse from one node to another.)

To Manually Connect Nodes

If you do not have a middle mouse button and prefer to manually connect nodes, you can use the pop-up menu for a node to connect it to another node already on the canvas.

1. Right-click the node from which you want to start the connection. Doing so opens the node menu.
2. On the menu, click **Connect**.

3. A connection icon is displayed both on the start node and the cursor. Click a second node on the canvas to connect the two nodes.

When connecting nodes, there are several guidelines to follow. You will receive an error message if you attempt to make any of the following types of connections:

- A connection leading to a source node
- A connection leading from a terminal node
- A node having more than its maximum number of input connections
- Connecting two nodes that are already connected
- Circularity (data returns to a node from which it has already flowed)

Bypassing Nodes in a Stream

When you bypass a node in the data stream, all of its input and output connections are replaced by connections that lead directly from its input nodes to its output nodes. If the node does not have both input and output connections, then all of its connections are deleted rather than rerouted.

For example, you might have a stream that derives a new field, filters fields, and then explores the results in a histogram and table. If you want to also view the same graph and table for data *before* fields are filtered, you can add either new Histogram and Table nodes to the stream, or you can bypass the Filter node. When you bypass the Filter node, the connections to the graph and table pass directly from the Derive node. The Filter node is disconnected from the stream.

To Bypass a Node

1. On the stream canvas, use the middle mouse button to double-click the node that you want to bypass. Alternatively, you can use Alt+double-click.

Note: You can undo this action clicking **Undo** on the Edit menu or by pressing Ctrl+Z.

Disabling Nodes in a Stream

Process nodes with a single input within streams can be disabled, with the result that the node is ignored during running of the stream. This saves you from having to remove or bypass the node and means you can leave it connected to the remaining nodes. You can still open and edit the node settings; however, any changes will not take effect until you enable the node again.

For example, you might have a stream that filters several fields, and then builds models with the reduced data set. If you want to also build the same models *without* fields being filtered, to see if they improve the model results, you can disable the Filter node. When you disable the Filter node, the connections to the modeling nodes pass directly through from the Derive node to the Type node.

To Disable a Node

1. On the stream canvas, right-click the node that you want to disable.
2. Click **Disable Node** on the pop-up menu.

Alternatively, you can click **Node** > **Disable Node** on the Edit menu. When you want to include the node back in the stream, click **Enable Node** in the same way.

Note: You can undo this action clicking **Undo** on the Edit menu or by pressing Ctrl+Z.

Adding Nodes in Existing Connections

You can add a new node between two connected nodes by dragging the arrow that connects the two nodes.

1. With the middle mouse button, click and drag the connection arrow into which you want to insert the node. Alternatively, you can hold down the Alt key while clicking and dragging to simulate a middle mouse button.
2. Drag the connection to the node that you want to include and release the mouse button.

Note: You can remove new connections from the node and restore the original by **bypassing** the node.

Deleting Connections between Nodes

To delete the connection between two nodes:

1. Right-click the connection arrow.
2. On the menu, click **Delete Connection**.

To delete all connections to and from a node, do one of the following:

- Select the node and press F3.
- Select the node and, on the main menu, click:

Edit > **Node** > **Disconnect**

Setting options for nodes

Once you creat and connect nodes, there are several options for customizing nodes. Right-click a node and select one of the menu options.

- Click **Edit** to open the dialog box for the selected node.
- Click **Connect** to manually connect one node to another.
- Click **Disconnect** to delete all links to and from the node.
- Click **Rename and Annotate** to open the Annotations tab of the editing dialog box.
- Click **New Comment** to add a comment related to the node. See the topic "Adding Comments and Annotations to Nodes and Streams" on page 49 for more information.
- Click **Disable Node** to "hide" the node during processing. To make the node visible again for processing, click **Enable Node**. See the topic "Disabling Nodes in a Stream" on page 33 for more information.
- Click **Cut** or **Delete** to remove the selected node(s) from the stream canvas. *Note*: Clicking **Cut** allows you to paste nodes, while **Delete** does not.
- Click **Copy Node** to make a copy of the node with no connections. This can be added to a new or existing stream.
- Click **Load Node** to open a previously saved node and load its options into the currently selected node. The nodes must be of identical types.
- Click **Retrieve Node** to retrieve a node from a connected IBM SPSS Collaboration and Deployment Services Repository.
- Click **Save Node** to save the node's details in a file. You can load node details only into another node of the same type.
- Click **Store Node** to store the selected node in a connected IBM SPSS Collaboration and Deployment Services Repository.
- Click **Cache** to expand the menu, with options for caching the selected node.
- Click **Data Mapping** to expand the menu, with options for mapping data to a new source or specifying mandatory fields.
- Click **Create SuperNode** to expand the menu, with options for creating a SuperNode in the current stream.
- Click **Generate User Input Node** to replace the selected node. Examples generated by this node will have the same fields as the current node.
- Click **Run From Here** to run all terminal nodes downstream from the selected node.

Caching options for nodes

To optimize stream running, you can set up a *cache* on any nonterminal node. When you set up a cache on a node, the cache is filled with the data that passes through the node the next time you run the data

stream. From then on, the data is read from the cache (which is stored on disk in a temporary directory) rather than from the data source.

Caching is most useful following a time-consuming operation such as a sort, merge, or aggregation. For example, suppose that you have a source node set to read sales data from a database and an Aggregate node that summarizes sales by location. You can set up a cache on the Aggregate node rather than on the source node because you want the cache to store the aggregated data rather than the entire data set.

Note: Caching at source nodes, which simply stores a copy of the original data as it is read into IBM SPSS Modeler, will not improve performance in most circumstances.

Nodes with caching enabled are displayed with a small document icon at the top right corner. When the data is cached at the node, the document icon is green.

To Enable a Cache

1. On the stream canvas, right-click the node and click **Cache** on the menu.
2. On the caching submenu, click **Enable**.
3. You can turn the cache off by right-clicking the node and clicking **Disable** on the caching submenu.

Caching Nodes in a Database

For streams run in a database, data can be cached midstream to a temporary table in the database rather than the file system. When combined with SQL optimization, this may result in significant gains in performance. For example, the output from a stream that merges multiple tables to create a data mining view may be cached and reused as needed. By automatically generating SQL for all downstream nodes, performance can be further improved.

To take advantage of database caching, both SQL optimization and database caching must be enabled. Note that Server optimization settings override those on the Client. See the topic "Setting optimization options for streams" on page 39 for more information.

With database caching enabled, simply right-click any nonterminal node to cache data at that point, and the cache will be created automatically directly in the database the next time the stream is run. If database caching or SQL optimization is not enabled, the cache will be written to the file system instead.

Note: The following databases support temporary tables for the purpose of caching: Db2, Oracle, SQL Server, and Teradata. Other databases, such as Netezza, will use a normal table for database caching. The SQL code can be customized for specific databases - contact Services for assistance.

To Flush a Cache

A white document icon on a node indicates that its cache is empty. When the cache is full, the document icon becomes solid green. If you want to replace the contents of the cache, you must first flush the cache and then re-run the data stream to refill it.

1. On the stream canvas, right-click the node and click **Cache** on the menu.
2. On the caching submenu, click **Flush**.

To Save a Cache

You can save the contents of a cache as an IBM SPSS Statistics data file (*.sav). You can then either reload the file as a cache, or you can set up a node that uses the cache file as its data source. You can also load a cache that you saved from another project.

1. On the stream canvas, right-click the node and click **Cache** on the menu.
2. On the caching submenu, click **Save Cache**.
3. In the Save Cache dialog box, browse to the location where you want to save the cache file.
4. Enter a name in the File Name text box.
5. Be sure that ***.sav** is selected in the Files of Type list, and click **Save**.

To Load a Cache

If you have saved a cache file before removing it from the node, you can reload it.

1. On the stream canvas, right-click the node and click **Cache** on the menu.
2. On the caching submenu, click **Load Cache**.
3. In the Load Cache dialog box, browse to the location of the cache file, select it, and click **Load**.

Previewing data in nodes

To ensure that data is being changed in the way you expect as you build a stream, you could run your data through a Table node at each significant step. To save you from having to do this, you can generate a preview from each node that displays a sample of the data that will be created, thereby reducing the time it takes to build each node.

For nodes upstream of a model nugget, the preview shows the input fields; for a model nugget or nodes downstream of the nugget (except terminal nodes), the preview shows input and generated fields.

The default number of rows displayed is 10; however, you can change this in the stream properties. See the topic "Setting general options for streams" on page 37 for more information.

From the **Generate** menu, you can create several types of nodes.

Note: When previewing data generated by this node, all property changes will be applied to this node and cannot be cancelled (the same behavior as clicking **Apply**).

Locking nodes

To prevent other users from amending the settings of one or more nodes in a stream, you can encapsulate the node or nodes in a special type of node called a SuperNode, and then lock the SuperNode by applying password protection.

Working with streams

Once you have connected source, process, and terminal nodes on the stream canvas, you have created a stream. As a collection of nodes, streams can be saved, annotated, and added to projects. You can also set numerous options for streams, such as optimization, date and time settings, parameters, and scripts. These properties are discussed in the topics that follow.

In IBM SPSS Modeler, you can use and modify more than one data stream in the same IBM SPSS Modeler session. The right side of the main window contains the managers pane, which helps you to navigate the streams, outputs, and models that are currently open. If you cannot see the managers pane, click **Managers** on the View menu, then click the **Streams** tab.

From this tab, you can:

- Access streams.
- Save streams.
- Save streams to the current project.
- Close streams.
- Open new streams.
- Store and retrieve streams from an IBM SPSS Collaboration and Deployment Services repository (if available at your site). See the topic "About the IBM SPSS Collaboration and Deployment Services Repository " on page 195 for more information.

Right-click a stream on the Streams tab to access these options.

Setting options for streams

You can specify a number of options to apply to the current stream. You can also save these options as defaults to apply to all your streams. The options are as follows.

- **General.** Miscellaneous options such as symbols and text encoding to use in the stream. See "Setting general options for streams" on page 37 for more information.
- **Date/Time.** Options relating to the format of date and time expressions. See "Setting date and time options for streams" on page 39 for more information.
- **Number formats.** Options controlling the format of numeric expressions. See "Setting number format options for streams" on page 39 for more information.
- **Optimization.** Options for optimizing stream performance. See "Setting optimization options for streams" on page 39 for more information.
- **Logging and Status.** Options controlling SQL logging and record status. See "Setting SQL logging and record status options for streams" on page 41 for more information.
- **Layout.** Options relating to the layout of the stream on the canvas. See "Setting layout options for streams" on page 41 for more information.
- **Analytic Server.** Options relating to the use of Analytic Server with SPSS Modeler. See "Analytic Server stream properties" on page 41 for more information.
- **Geospatial.** Options relating to the formatting of geospatial data for use in the stream. See "Setting geospatial options for streams" on page 42 for more information.

To set stream options

1. On the File menu, click **Stream Properties** (or select the stream from the Streams tab in the managers pane, right-click and then click **Stream Properties** on the pop-up menu).
2. Click the **Options** tab.

Alternatively, on the Tools menu, click:

Stream Properties > **Options**

Setting general options for streams

The general options are a set of miscellaneous options that apply to various aspects of the current stream.

The **Basic** section includes the following basic options:

- **Decimal symbol.** Select either a comma (,) or a period (.) as a decimal separator.
- **Grouping symbol.** For number display formats, select the symbol used to group values (for example, the comma in 3,000.00). Options include none, period, comma, space, and locale-defined (in which case the default for the current locale is used).
- **Encoding.** Specify the stream default method for text encoding. (*Note*: Applies to Var. File source node and Flat File export node only. No other nodes use this setting; most data files have embedded encoding information.) You can choose either the system default or UTF-8. The system default is specified in the Windows Control Panel or, if running in distributed mode, on the server computer. See the topic "Unicode Support in IBM SPSS Modeler " on page 253 for more information.
- **Ruleset Evaluation.** Determines how rule set models are evaluated. By default, rule sets use **Voting** to combine predictions from individual rules and determine the final prediction. To ensure that rule sets use the first hit rule by default, select **First Hit**. Note that this option does not apply to Decision List models, which always use the first hit as defined by the algorithm.

Maximum number of rows to show in Data Preview. Specify the number of rows to be shown when a preview of the data is requested for a node. See the topic "Previewing data in nodes" on page 36 for more information.

Maximum members for nominal fields. Select to specify a maximum number of members for nominal (set) fields after which the data type of the field becomes **Typeless**. This option is useful when working with large nominal fields. *Note*: When the measurement level of a field is set to **Typeless**, its role is automatically set to **None**. This means that the fields are not available for modeling.

Limit set size for Kohonen, and K-Means modeling. Select to specify a maximum number of members for nominal fields used in Kohonen nets and *K*-Means modeling. The default set size is 20, after which the field is ignored and a warning is raised, providing information on the field in question.

Note that, for compatibility, this option also applies to the old Neural Network node that was replaced in version 14 of IBM SPSS Modeler; some legacy streams may still contain this node.

Refresh source nodes on execution. Select to automatically refresh all source nodes when running the current stream. This action is analogous to clicking the **Refresh** button on a source node, except that this option automatically refreshes all source nodes (except User Input nodes) for the current stream.

Note: Selecting this option flushes the caches of downstream nodes even if the data has not changed. If you use the **Run the current stream** option from the toolbar, flushing occurs only once per running of the stream, though, which means that you can still use downstream caches as temporary storage for a single running. For example, say that you have set a cache midstream after a complex derive operation and that you have several graphs and reports attached downstream of this Derive node. When running the stream, the cache at the Derive node will be flushed and refilled but only for the first graph or report. Subsequent terminal nodes will read data from the Derive node cache. Note that if you choose to execute each terminal node individually (when you have more than one terminal node), instead of using the **Run the current stream**option, cache flushing occurs every time you execute a terminal node.

Display field and value labels in output. Displays field and value labels in tables, charts, and other output. If labels do not exist, the field names and data values will be displayed instead. Labels are turned off by default; however, you can toggle labels on an individual basis elsewhere in IBM SPSS Modeler. You can also choose to display labels on the output window using a toggle button available on the toolbar.

Figure 11. Toolbar icon used to toggle field and value labels

Display execution times. Displays individual execution times for stream nodes on the Execution Times tab after the stream is run. See the topic "Viewing node execution times" on page 43 for more information.

The **Automatic Node Creation** section includes the following options for creating nodes automatically in individual streams. These options control whether or not to insert the modeling nuggets onto the stream canvas when generating new nuggets. By default, these options only apply to streams created in version 16 or later. In IBM SPSS Modeler 16 or later, if you open a stream created in version 15 or earlier and execute a modeling node, the nugget will not be placed onto the stream canvas as it used to be in previous releases. If you create a new stream using IBM SPSS Modeler 16 or later and execute a modeling node, the nugget generated is placed onto the stream canvas. This is as designed because, for example, the **Create model apply nodes for new model output** option would likely break pre-16 streams that run in batch, in IBM SPSS Collaboration and Deployment Services, and in other environments where the IBM SPSS Modeler Server client user interface is not present.

- **Create model apply nodes for new model output.** Automatically creates model apply nodes for the new model output. If you select this option, you can also choose from the **Create model update links** whether to set the links as enabled, disabled, or not to create them.

  When a new model applier or source node is created, the link options in the drop-downs control whether the update links between the builder node and the new node are created and, if so, what mode they are in. If links are created, chances are you want them enabled, but these options provide the user with complete control.

- **Create source nodes from source builders.** Automatically creates source nodes from the source builders. Similar to the previous option, if you select this option you can also choose from the **Create source refresh links** drop-down whether to set the refresh links as enabled, disabled, or not to create them.

Save As Default. The options specified apply only to the current stream. Click this button to set these options as the default for all streams.

Setting date and time options for streams

These options specify the format to use for various date and time expressions in the current stream.

Import date/time as Select whether to use date/time storage for date/time fields or whether to import them as string variables.

Date format Select a date format to be used for date storage fields or when strings are interpreted as dates by CLEM date functions.

Time format Select a time format to be used for time storage fields or when strings are interpreted as times by CLEM time functions.

Rollover days/mins For time formats, select whether negative time differences are interpreted as referring to the previous day or hour.

Date baseline (1st Jan) Select the baseline years (always 1 January) to be used by CLEM date functions that work with a single date.

2-digit dates start from Specify the cutoff year to add century digits for years that are denoted with only 2 digits. For example, specifying 1930 as the cutoff year assumes that 05/11/02 is in the year 2002. The same setting will use the 20th century for dates after 30; thus 05/11/73 is assumed to be in 1973.

Time zone Select how the time zone is chosen for use with the datetime_now CLEM expression.

- If you select **Server**, the time zone depends on the following items:

    - If the current stream uses an Analytic Server data source, the datetime_now expression uses the time from the Analytic Server; by default the server uses Coordinated Universal Time time.

    - If the current stream uses a Database source node, the supported databases use SQL pushback, and the datetime_now expression uses the time of the database.

    - For all other streams, the time zone uses the time from SPSS Modeler Server.

- If you select **Modeler client** the time zone reflects the time zone details from the machine on which SPSS Modeler is installed.

- Alternatively, you can select any of the Coordinated Universal Time values for the time zone.

Save As Default. The options that are specified apply only to the current stream. Click this button to set these options as the default for all streams.

Setting number format options for streams

These options specify the format to use for various numeric expressions in the current stream.

Number display format. You can choose from standard (**####.###**), scientific (**#.###E+##**), or currency display formats (**$###.##**).

Decimal places (standard, scientific, currency). For number display formats, specifies the number of decimal places to be used when displaying or printing real numbers. This option is specified separately for each display format.

Calculations in. Select **Radians** or **Degrees** as the unit of measurement to be used in trigonometric CLEM expressions. See the topic "Trigonometric Functions" on page 170 for more information.

Save As Default. The options specified apply only to the current stream. Click this button to set these options as the default for all streams.

Setting optimization options for streams

You can use the Optimization settings to optimize stream performance. Note that the performance and optimization settings on IBM SPSS Modeler Server (if used) override any equivalent settings in the client. If these settings are disabled in the server, then the client cannot enable them. But if they are enabled in the server, the client can choose to disable them.

Note: Database modeling and SQL optimization require that IBM SPSS Modeler Server connectivity be enabled on the IBM SPSS Modeler computer. With this setting enabled, you can access database

algorithms, push back SQL directly from IBM SPSS Modeler, and access IBM SPSS Modeler Server. To verify the current license status, choose the following from the IBM SPSS Modeler menu.

Help > **About** > **Additional Details**

If connectivity is enabled, you see the option **Server Enablement** in the License Status tab.

See "Connecting to IBM SPSS Modeler Server " on page 8 for more information.

Note: Whether SQL pushback and optimization are supported depends on the type of database in use. For the latest information on which databases and ODBC drivers are supported and tested for use with IBM SPSS Modeler, see the corporate Support site at http://www.ibm.com/support.

Enable stream rewriting. Select this option to enable stream rewriting in IBM SPSS Modeler. Four types of rewriting are available, and you can select one or more of them. Stream rewriting reorders the nodes in a stream behind the scenes for more efficient operation, without altering stream semantics.

- **Optimize SQL generation.** This option enables nodes to be reordered within the stream so that more operations can be pushed back using SQL generation for execution in the database. When it finds a node that cannot be rendered into SQL, the optimizer will look ahead to see if there are any downstream nodes that can be rendered into SQL and safely moved in front of the problem node without affecting the stream semantics. Not only can the database perform operations more efficiently than IBM SPSS Modeler, but such pushbacks act to reduce the size of the data set that is returned to IBM SPSS Modeler for processing. This, in turn, can reduce network traffic and speed stream operations. Note that the **Generate SQL** check box must be selected for SQL optimization to have any effect.

- **Optimize CLEM expression.** This option enables the optimizer to search for CLEM expressions that can be preprocessed before the stream is run, in order to increase the processing speed. As a simple example, if you have an expression such as *log(salary)*, the optimizer would calculate the actual salary value and pass that on for processing. This can be used both to improve SQL pushback and IBM SPSS Modeler Server performance.

- **Optimize syntax execution.** This method of stream rewriting increases the efficiency of operations that incorporate more than one node containing IBM SPSS Statistics syntax. Optimization is achieved by combining the syntax commands into a single operation, instead of running each as a separate operation.

- **Optimize other execution.** This method of stream rewriting increases the efficiency of operations that cannot be delegated to the database. Optimization is achieved by reducing the amount of data in the stream as early as possible. While maintaining data integrity, the stream is rewritten to push operations closer to the data source, thus reducing data downstream for costly operations, such as joins.

Enable parallel processing. When running on a computer with multiple processors, this option allows the system to balance the load across those processors, which may result in faster performance. Use of multiple nodes or use of the following individual nodes may benefit from parallel processing: C5.0, Merge (by key), Sort, Bin (rank and tile methods), and Aggregate (using one or more key fields).

Generate SQL. Select this option to enable SQL generation, allowing stream operations to be pushed back to the database by using SQL code to generate execution processes, which may improve performance. To further improve performance, **Optimize SQL generation** can also be selected to maximize the number of operations pushed back to the database. When operations for a node have been pushed back to the database, the node will be highlighted in purple when the stream is run.

- **Database caching.** For streams that generate SQL to be executed in the database, data can be cached midstream to a temporary table in the database rather than to the file system. When combined with SQL optimization, this may result in significant gains in performance. For example, the output from a stream that merges multiple tables to create a data mining view may be cached and reused as needed. With database caching enabled, simply right-click any nonterminal node to cache data at that point, and the cache is automatically created directly in the database the next time the stream is run. This allows SQL to be generated for downstream nodes, further improving performance. Alternatively, this option can be disabled if needed, such as when policies or permissions preclude data being written to the database. If database caching or SQL optimization is not enabled, the cache will be written to the file system instead. See the topic "Caching options for nodes" on page 34 for more information.

- **Use relaxed conversion.** This option enables the conversion of data from either strings to numbers, or numbers to strings, if stored in a suitable format. For example, if the data is kept in the database as a string, but actually contains a meaningful number, the data can be converted for use when the pushback occurs.

Note: Due to minor differences in SQL implementation, streams run in a database may return slightly different results from those returned when run in IBM SPSS Modeler. For similar reasons, these differences may also vary depending on the database vendor.

Save As Default. The options specified apply only to the current stream. Click this button to set these options as the default for all streams.

Setting SQL logging and record status options for streams

These settings include various options controlling the display of SQL statements generated by the stream, and the display of the number of records processed by the stream.

Display SQL in the messages log during stream execution. Specifies whether SQL generated while running the stream is passed to the message log.

Display SQL generation details in the messages log during stream preparation. During stream preview, specifies whether a preview of the SQL that would be generated is passed to the messages log.

Display SQL. Specifies whether any SQL that is displayed in the log should contain native SQL functions or standard ODBC functions of the form `{fn FUNC(…)}`, as generated by SPSS Modeler. The former relies on ODBC driver functionality that may not be implemented.

Reformat SQL for improved readability. Specifies whether SQL displayed in the log should be formatted for readability.

Show status for records. Specifies when records should be reported as they arrive at terminal nodes. Specify a number that is used for updating the status every *N* records.

Save As Default. The options specified apply only to the current stream. Click this button to set these options as the default for all streams.

Setting layout options for streams

These settings provide a number of options relating to the display and use of the stream canvas.

Minimum stream canvas width. Specify the minimum width of the stream canvas in pixels.

Minimum stream canvas height. Specify the minimum height of the stream canvas in pixels.

Stream scroll rate. Specify the scrolling rate for the stream canvas to control how quickly the stream canvas pane scrolls when a node is being dragged from one place to another on the canvas. Higher numbers specify a faster scroll rate.

Icon name maximum. Specify a limit in characters for the names of nodes on the stream canvas.

Icon size. Select an option to scale the entire stream view to one of a number of sizes between 8% and 200% of the standard icon size.

Grid cell size. Select a grid cell size from the list. This number is used for aligning nodes on the stream canvas using an invisible grid. The default grid cell size is 0.25.

Snap to Grid. Select to align icons to an invisible grid pattern (selected by default).

Generated icon placement. Choose where on the canvas to place icons for nodes generated from model nuggets. Default is top left.

Save As Default. The options specified apply only to the current stream. Click this button to set these options as the default for all streams.

Analytic Server stream properties

These settings provide a number of options for working with Analytic Server.

Maximum number of records to process outside of Analytic Server
Specify the maximum number of records to be imported into SPSS Modeler server from an Analytic Server data source.

Notification when a node can't be processed in Analytic Server
This setting determines what happens when a stream that would be submitted to Analytic Server contains a node that can't be processed in Analytic Server. Specify whether to issue a warning and continue processing the stream, or throw an error and stop processing.

Split Model Storage Settings

Store split models by reference on Analytic Server when model size (MB) exceeds
Model nuggets are typically stored as part of the stream. Split models with many splits can produce large nuggets, and moving the nugget back and forth between the stream and the Analytic Server can impact performance. As a solution, when a split model exceeds the specified size, it is stored on the Analytic Server, and the nugget in the SPSS Modeler contains a reference to the model.

Default folder to store models by reference on Analytic Server once execution is complete
Specify the default path where you want to store split models on Analytic Server. The path should start with a valid Analytic Server project name.

Folder to store promoted models
Specify the default path where you want to store "promoted" models. A promoted model is not cleaned up when the SPSS Modeler session is over.

Setting geospatial options for streams

Any geospatial field, whether it is a shape, coordinate, or single axis value (such as x or y, or latitude and longitude) has an associated coordinate system. This coordinate system sets attributes such as the origin point (0,0) and the units associated with the values.

There are a number of coordinate systems and there are two types: Geographic and Projected. All of the spatial functions in SPSS Modeler can only be used with a Projected coordinate system.

Due to the nature of coordinate systems, merging or appending data from two separate geospatial data sources requires that the sources use the same coordinate system. Because of this you must specify a coordinate setting for any geospatial data used in the stream.

Data is automatically reprojected to use the chosen stream coordinate system in the following situations:

- For spatial functions (such as area, closeto, within), the parameter passed to the function is automatically reprojected; however, the original row data is left unchanged.
- When using either the build or scoring (nugget) nodes in Spatio-Temporal Prediction (STP), the location field is automatically reprojected. When scoring, the location that comes out of the nugget is the original location.
- When using the Map Visualization node.

Stream coordinate system. Only available if you select the check box. Click **Change** to display a list of available Projected Coordinate Systems and select the one you want to use for the current stream.

Save As Default. The coordinate system you select only applies to the current stream. To select the system as the default for all streams, click this button.

Selecting geospatial coordinate systems

All of the spatial functions in SPSS Modeler can be used only with a Projected coordinate system.

The **Select Stream Coordinate System** dialog box contains a list of all the projected coordinate systems that you can select for any geospatial data that is used in a stream.

The following information is listed for each coordinate system.

- **WKID** The Well Known ID that is unique to each coordinate system.
- **Name** The name of the coordinate system.

- **Units** The unit of measurement that is associated with the coordinate system.

In addition to the list of all coordinate systems, the dialog box has a **Filtering** control. If you know all or part of the name of the coordinate system you require, type it in the **Name** field at the bottom of the dialog box. The list of coordinate systems from which you can choose is automatically filtered to show only the systems with names that contain the text you entered.

Viewing stream operation messages

Messages regarding stream operations, such as running, optimization, and time elapsed for model building and evaluation, can easily be viewed using the Messages tab in the stream properties dialog box. Error messages are also reported in this table.

To View Stream Messages

1. On the File menu, click **Stream Properties** (or select the stream from the Streams tab in the managers pane, right-click and then click **Stream Properties** on the pop-up menu).
2. Click the **Messages** tab.

Alternatively, on the Tools menu, click:

Stream Properties > **Messages**

In addition to messages regarding stream operations, error messages are reported here. When stream running is terminated because of an error, this dialog box will open to the Messages tab with the error message visible. Additionally, the node with errors is highlighted in red on the stream canvas.

If SQL optimization and logging options are enabled in the User Options dialog box, then information on generated SQL is also displayed. See the topic "Setting optimization options for streams" on page 39 for more information.

You can save messages reported here for a stream by clicking **Save Messages** on the Save button drop-down list (on the left, just below the Messages tab).

You can clear all messages for a given stream by selecting **Clear All Messages** from the drop-down.

Note that CPU time is the amount of time the server process is utilizing CPU. Elapsed time is the total time between execution start and execution end, so also includes things like transferring files and rendering outputs. CPU time can be more than elapsed time when a stream is leveraging multiple CPUs (parallel execution). When a stream fully pushes back to the database being used as a data source, the CPU time will be zero.

Viewing node execution times

On the Messages tab you can choose to display Execution Times, where you can see the individual execution times for nodes in the stream that run on the IBM SPSS Modeler Server. Note that the times may not be accurate for streams run in other areas, such as R or Analytic Server. And the execution time of some nodes cannot be calculated.

Note: For this feature to work, the **Display execution times** option must be selected on the **General** setting of the **Options** tab.

In the table of node execution times, the columns are as follows. Click a column heading to sort the entries into ascending or descending order (for example, to see which nodes have the longest execution times).

Terminal Node. The identifier of the branch to which the node belongs. The identifier is the name of the terminal node at the end of the branch.

Node Label. The name of the node to which the execution time refers.

Node Id. The unique identifier of the node to which the execution time refers. This identifier is generated by the system when the node is created.

Execution Time(s). The time in seconds taken to execute this node. Note that execution time can often vary from the time you see in general messages because time is spent on preparing data and retrieving data in the output when running a stream, and this type of time cannot be calculated.

Setting Stream and Session Parameters

Parameters can be defined for use in CLEM expressions and in scripting. They are, in effect, user-defined variables that are saved and persisted with the current stream, session, or SuperNode and can be accessed from the user interface as well as through scripting. If you save a stream, for example, any parameters set for that stream are also saved. (This distinguishes them from local script variables, which can be used only in the script in which they are declared.) Parameters are often used in scripting to control the behavior of the script, by providing information about fields and values that do not need to be hard coded in the script.

The scope of a parameter depends on where it is set:

- Stream parameters can be set in a stream script or in the stream properties dialog box, and they are available to all nodes in the stream. They are displayed on the Parameters list in the Expression Builder.
- Session parameters can be set in a stand-alone script or in the session parameters dialog box. They are available to all streams used in the current session (all streams listed on the Streams tab in the managers pane).

Parameters can also be set for SuperNodes, in which case they are visible only to nodes encapsulated within that SuperNode.

To Set Stream and Session Parameters through the User Interface

1. To set stream parameters, on the main menu, click:

   Tools > **Stream Properties** > **Parameters**

2. To set session parameters, click **Set Session Parameters** on the Tools menu.

Prompt?. Check this box if you want the user to be prompted at runtime to enter a value for this parameter.

Name. Parameter names are listed here. You can create a new parameter by entering a name in this field. For example, to create a parameter for the minimum temperature, you could type `minvalue`. Do not include the `$P-` prefix that denotes a parameter in CLEM expressions. This name is also used for display in the CLEM Expression Builder.

Long name. Lists the descriptive name for each parameter created.

Storage. Select a storage type from the list. Storage indicates how the data values are stored in the parameter. For example, when working with values containing leading zeros that you want to preserve (such as 008), you should select **String** as the storage type. Otherwise, the zeros will be stripped from the value. Available storage types are string, integer, real, time, date, and timestamp. For date parameters, note that values must be specified using ISO standard notation as shown in the next paragraph.

Value. Lists the current value for each parameter. Adjust the parameter as required. Note that for date parameters, values must be specified in ISO standard notation (that is, `YYYY-MM-DD`). Dates specified in other formats are not accepted.

Type (optional). If you plan to deploy the stream to an external application, select a measurement level from the list. Otherwise, it is advisable to leave the *Type* column as is. If you want to specify value constraints for the parameter, such as upper and lower bounds for a numeric range, select **Specify** from the list.

Note that long name, storage, and type options can be set for parameters through the user interface only. These options cannot be set using scripts.

Click the arrows at the right to move the selected parameter further up or down the list of available parameters. Use the delete button (marked with an *X*) to remove the selected parameter.

Specifying Runtime Prompts for Parameter Values

If you have streams where you might need to enter different values for the same parameter on different occasions, you can specify runtime prompts for one or more stream or session parameter values.

Parameters. (Optional) Enter a value for the parameter, or leave the default value if there is one.

Turn off these prompts. Select this box if you do not want these prompts to be displayed when you run the stream. You can cause them to be redisplayed by selecting the **Prompt?** check box on the stream properties or session properties dialog box where the parameters were defined. See the topic "Setting Stream and Session Parameters" on page 44 for more information.

Specifying Value Constraints for a Parameter Type

You can make value constraints for a parameter available during stream deployment to an external application that reads data modeling streams. This dialog box allows you to specify the values available to an external user running the stream. Depending on the data type, value constraints vary dynamically in the dialog box. The options shown here are identical to the options available for values from the Type node.

Type. Displays the currently selected measurement level. You can change this value to reflect the way that you intend to use the parameter in IBM SPSS Modeler.

Storage. Displays the storage type if known. Storage types are unaffected by the measurement level (continuous, nominal or flag) that you choose for work in IBM SPSS Modeler. You can alter the storage type on the main Parameters tab.

The bottom half of the dialog box dynamically changes depending on the measurement level selected in the **Type** field.

Continuous Measurement Levels

Lower. Specify a lower limit for the parameter values.

Upper. Specify an upper limit for the parameter values.

Labels. You can specify labels for any value of a range field. Click the **Labels** button to open a separate dialog box for specifying value labels.

Nominal Measurement Levels

Values. This option allows you to specify values for a parameter that will be used as a nominal field. Values will not be coerced in the IBM SPSS Modeler stream but will be used in a drop-down list for external deployment applications. Using the arrow and delete buttons, you can modify existing values as well as reorder or delete values.

Flag Measurement Levels

True. Specify a flag value for the parameter when the condition is met.

False. Specify a flag value for the parameter when the condition is not met.

Labels. You can specify labels for the values of a flag field.

Stream deployment options

The Deployment tab of the stream properties dialog box enables you to specify options for deploying the stream within IBM SPSS Collaboration and Deployment Services for the purposes of model refresh or automated job scheduling. All streams require a designated scoring branch before they can be deployed. See "Storing and deploying repository objects" on page 195 for more information.

Looping Execution for Streams

Using the Execution tab in the stream properties dialog box, you can set up looping conditions to automate repetitive tasks in the current stream.

Once you set these conditions you can use it as an introduction to scripting as it populates the script window with basic scripting for your stream which you can then modify - perhaps to use as a base from which to build better scripts. See the topic "Global Functions" on page 190 for more information.

To set Looping for a Stream

1. On the File menu, click **Stream Properties** (or select the stream from the Streams tab in the managers pane, right-click and then click **Stream Properties** on the pop-up menu).
2. Click the **Execution** tab.
3. Select the **Looping / Conditional Execution** execution mode.
4. Click the **Looping** tab.

Alternatively, on the Tools menu, click:

Stream Properties > **Execution**

As a further alternative, right click on the node and from the context menu, click:

Looping / Conditional Execution > **Edit Looping Settings**

Iteration. You cannot edit this row number value, but you can add, delete, or move an iteration up or down using the buttons to the right of the table.

Table headers. These reflect the iteration key and any iteration variables you created when setting up the loop.

Viewing Global Values for Streams

Using the Globals tab in the stream properties dialog box, you can view the global values set for the current stream. Global values are created using a Set Globals node to determine statistics such as mean, sum, or standard deviation for selected fields.

Once the Set Globals node is run, these values are then available for a variety of uses in stream operations. See the topic "Global Functions" on page 190 for more information.

To View Global Values for a Stream

1. On the File menu, click **Stream Properties** (or select the stream from the Streams tab in the managers pane, right-click and then click **Stream Properties** on the pop-up menu).
2. Click the **Globals** tab.

Alternatively, on the Tools menu, click:

Stream Properties > **Globals**

Globals available. Available globals are listed in this table. You cannot edit global values here, but you can clear all global values for a stream using the Clear All Values button to the right of the table.

Searching for Nodes in a Stream

You can search for nodes in a stream by specifying a number of search criteria, such as node name, category and identifier. This feature can be especially useful for complex streams containing a large number of nodes.

To Search for Nodes in a Stream

1. On the File menu, click **Stream Properties** (or select the stream from the Streams tab in the managers pane, right-click and then click **Stream Properties** on the pop-up menu).
2. Click the **Search** tab.

Alternatively, on the Tools menu, click:

Stream Properties > **Search**

You can specify more than one option to limit the search, except that searching by node ID (using the **ID equals** field) excludes the other options.

Node label contains. Check this box and enter all or part of a node label to search for a particular node. Searches are not case-sensitive, and multiple words are treated as a single piece of text.

Node category. Check this box and click a category on the list to search for a particular type of node. **Process Node** means a node from the Record Ops or Field Ops tab of the nodes palette; **Apply Model Node** refers to a model nugget.

Keywords include. Check this box and enter one or more complete keywords to search for nodes having that text in the Keywords field on the Annotations tab of the node dialog box. Keyword text that you enter must be an exact match. Separate multiple keywords with semicolons to search for alternatives (for example, entering `proton;neutron` will find all nodes with either of these keywords. See the topic "Annotations" on page 53 for more information.

Annotation contains. Check this box and enter one or more words to search for nodes that contain this text in the main text area on the Annotations tab of the node dialog box. Searches are not case-sensitive, and multiple words are treated as a single piece of text. See the topic "Annotations" on page 53 for more information.

Generates field called. Check this box and enter the name of a generated field (for example, $C-Drug). You can use this option to search for modeling nodes that generate a particular field. Enter only one field name, which must be an exact match.

ID equals. Check this box and enter a node ID to search for a particular node with that identifier (selecting this option disables all the preceding options). Node IDs are assigned by the system when the node is created, and can be used to reference the node for the purposes of scripting or automation. Enter only one node ID, which must be an exact match. See the topic "Annotations" on page 53 for more information.

Search in SuperNodes. This box is checked by default, meaning that the search is performed on nodes both inside and outside SuperNodes. Clear the box if you want to perform the search only on nodes outside SuperNodes, at the top level of the stream.

Find. When you have specified all the options you want, click this button to start the search.

Nodes that match the specified options are listed in the lower part of the dialog box. Select a node in the list to highlight it on the stream canvas.

Renaming streams

Using the Annotations tab in the stream properties dialog box, you can add descriptive annotations for a stream and create a custom name for the stream. These options are useful especially when generating reports for streams added to the project pane. See the topic "Annotations" on page 53 for more information.

Stream descriptions

For each stream that you create, IBM SPSS Modeler produces a stream description containing information on the contents of the stream. This can be useful if you are trying to see what a stream does but you do not have IBM SPSS Modeler installed, for example when accessing a stream through IBM SPSS Collaboration and Deployment Services.

The stream description is displayed in the form of an HTML document consisting of a number of sections.

General Stream Information

This section contains the stream name, together with details of when the stream was created and last saved.

Description and Comments

This section includes any:

• Stream annotations (see "Annotations" on page 53)

- Comments not connected to specific nodes
- Comments connected to nodes in both the modeling and scoring branches of the stream

Scoring Information

This section contains information under various headings relating to the scoring branch of the stream.

- **Comments.** Includes comments linked only to nodes in the scoring branch.
- **Inputs.** Lists the input fields together with their storage types (for example, string, integer, real and so on).
- **Outputs.** Lists the output fields, including the additional fields generated by the modeling node, together with their storage types.
- **Parameters.** Lists any parameters relating to the scoring branch of the stream and which can be viewed or edited each time the model is scored. These parameters are identified when you click the **Scoring Parameters** button on the **Deployment** tab of the stream properties dialog box.
- **Model Node.** Shows the model name and type (for example, Neural Net, C&R Tree and so on). This is the model nugget selected for the **Model node** field on the **Deployment** tab of the stream properties dialog box.
- **Model Details.** Shows details of the model nugget identified under the previous heading. Where possible, predictor importance and evaluation charts for the model are included.

Modeling Information

Contains information relating to the modeling branch of the stream.

- **Comments.** Lists any comments or annotations that are connected to nodes in the modeling branch.
- **Inputs.** Lists the input fields together with their role in the modeling branch (in the form of the field role value, for example, Input, Target, Split and so on).
- **Parameters.** Lists any parameters relating to the modeling branch of the stream and which can be viewed or edited each time the model is updated. These parameters are identified when you click the **Model Build Parameters** button on the **Deployment** tab of the stream properties dialog box.
- **Modeling node.** Shows the name and type of the modeling node used to generate or update the model.

Previewing stream descriptions

You can view the contents of a stream description in a web browser by clicking an option on the stream properties dialog box. The contents of the description depend on the options you specify on the Deployment tab of the dialog box. See the topic "Stream Deployment Options" on page 207 for more information.

To view a stream description:

1. On the main IBM SPSS Modeler menu, click:

   Tools > **Stream Properties** > **Deployment**
2. Set the deployment type, the designated scoring node and any scoring parameters.
3. If the deployment type is Model Refresh, you can optionally select a:

   - Modeling node and any model build parameters
   - Model nugget on the scoring branch of the stream
4. Click the **Preview Stream Description** button.

Exporting Stream Descriptions

You can export the contents of the stream description to an HTML file.

To export a stream description:

1. On the main menu, click:

   File > **Export Stream Description**
2. Enter a name for the HTML file and click **Save**.

Running streams

Once you specify the required options for streams and connect the required nodes, you can run the stream by running the data through nodes in the stream. There are several ways to run a stream within IBM SPSS Modeler. You can:

- Click **Run** on the Tools menu.
- Click one of the **Run...** buttons on the toolbar. These buttons allow you to run the entire stream or simply the selected terminal node. See the topic " IBM SPSS Modeler Toolbar" on page 15 for more information.
- Run a single data stream by right-clicking a terminal node and clicking **Run** on the pop-up menu.
- Run part of a data stream by right-clicking any non-terminal node and clicking **Run From Here** on the pop-up menu. Doing so causes only those operations after the selected node to be performed.

To halt the running of a stream in progress, you can click the red **Stop** button on the toolbar, or select **Stop Execution** from the Tools menu. Note that clicking the Stop button one time tells Modeler to have Modeler Server stop the execution. In some cases execution will stop immediately, but in other cases it must finish the current step before it can stop the entire execution. So time will vary. If you double-click the button, the server connection is dropped and a new connection is established. In most cases, this will close all server processes (and it may take some time). But in some cases, some server process won't be stopped.

If any stream takes longer than three seconds to run, the Execution Feedback dialog box is displayed to indicate the progress.

Some nodes have further displays giving additional information about stream execution. Select the corresponding row in the dialog box to see these displays. The first row is selected automatically.

Working with Models

If a stream includes a modeling node (that is, one from the Modeling or Database Modeling tab of the nodes palette), a **model nugget** is created when the stream is run. A model nugget is a container for a **model**, that is, the set of rules, formulas or equations that enables you to generate predictions against your source data, and which lies at the heart of predictive analytics.

Figure 12. Model nugget

When you successfully run a modeling node, a corresponding model nugget is placed on the stream canvas, where it is represented by a gold diamond-shaped icon (hence the name "nugget"). You can open the nugget and browse its contents to view details about the model. To view the predictions, you attach and run one or more terminal nodes, the output from which presents the predictions in a readable form.

A typical modeling stream consists of two branches. The **modeling branch** contains the modeling node, together with the source and processing nodes that precede it. The **scoring branch** is created when you run the modeling node, and contains the model nugget and the terminal node or nodes that you use to view the predictions.

For more information, see the *IBM SPSS Modeler Modeling Nodes* guide.

Adding Comments and Annotations to Nodes and Streams

You may need to describe a stream to others in your organization. To help you do this, you can attach explanatory comments to streams, nodes and model nuggets.

Others can then view these comments on-screen, or you can print out an image of the stream that includes the comments.

You can list all the comments for a stream or SuperNode, change the order of comments in the list, edit the comment text, and change the foreground or background color of a comment. See the topic "Listing Stream Comments" on page 53 for more information.
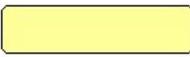
You can also add notes in the form of text annotations to streams, nodes and nuggets by means of the Annotations tab of a stream properties dialog box, a node dialog box, or a model nugget window. These notes are visible only when the Annotations tab is open, except that stream annotations can also be shown as on-screen comments. See the topic "Annotations" on page 53 for more information.

Comments

Comments take the form of text boxes in which you can enter any amount of text, and you can add as many comments as you like. A comment can be freestanding (not attached to any stream objects), or it can be connected to one or more nodes or model nuggets in the stream. Freestanding comments are typically used to describe the overall purpose of the stream; connected comments describe the node or nugget to which they are attached. Nodes and nuggets can have more than one comment attached, and the stream can have any number of freestanding comments.

Note: You can also show stream annotations as on-screen comments, though these cannot be attached to nodes or nuggets. See the topic "Converting Annotations to Comments" on page 53 for more information.

The appearance of the text box changes to indicate the current mode of the comment (or annotation shown as a comment), as the following table shows.

Table 3. Comment and annotation text box modes				
Comment text box	**Annotation text box**	**Mode**	**Indicates**	**Obtained by...**
		Edit	Comment is open for editing.	Creating a new comment or annotation, or double-clicking an existing one.
		Last selected	Comment can be moved, resized or deleted.	Clicking the stream background after editing, or single-clicking an existing comment or annotation.
		View	Editing is complete.	Clicking on another node, comment or annotation after editing.

When you create a new freestanding comment, it is initially displayed in the top left corner of the stream canvas.

If you are attaching a comment to a node or nugget, the comment is initially displayed above the stream object to which it is attached.

The text box is colored white to show that text can be entered. When you have entered the text, click outside the text box. The comment background changes to yellow to show that text entry is complete. The comment remains selected, enabling you to move, resize, or delete it.

When you click again, the border changes to solid lines to show that editing is complete.

Double-clicking a comment changes the text box to edit mode--the background changes to white and the comment text can be edited.

You can also attach comments to SuperNodes.

Operations Involving Comments

You can perform a number of operations on comments. You can:

- Add a freestanding comment
- Attach a comment to a node or nugget
- Edit a comment
- Resize a comment
- Move a comment
- Disconnect a comment
- Delete a comment
- Show or hide all comments for a stream

To add a freestanding comment

1. Ensure that nothing is selected on the stream.
2. Do one of the following:

   - On the main menu, click:

     Insert > **New Comment**
   - Right-click the stream background and click **New Comment** on the pop-up menu.
   - Click the **New Comment** button in the toolbar.
3. Enter the comment text (or paste in text from the clipboard).
4. Click a node in the stream to save the comment.

To attach a comment to a node or nugget

1. Select one or more nodes or nuggets on the stream canvas.
2. Do one of the following:

   - On the main menu, click:

     Insert > **New Comment**
   - Right-click the stream background and click **New Comment** on the pop-up menu.
   - Click the **New Comment** button in the toolbar.
3. Enter the comment text.
4. Click another node in the stream to save the comment.

   Alternatively, you can:
5. Insert a freestanding comment (see previous section).
6. Do one of the following:

- Select the comment, press F2, then select the node or nugget.
- Select the node or nugget, press F2, then select the comment.
- (Three-button mice only) Move the mouse pointer over the comment, hold down the middle button, drag the mouse pointer over the node or nugget, and release the mouse button.

To attach a comment to an additional node or nugget

If a comment is already attached to a node or nugget, or if it is currently at stream level, and you want to attach it to an additional node or nugget, do one of the following:

- Select the comment, press F2, then select the node or nugget.
- Select the node or nugget, press F2, then select the comment.
- (Three-button mice only) Move the mouse pointer over the comment, hold down the middle button, drag the mouse pointer over the node or nugget, and release the mouse button.

To edit an existing comment

1. Do one of the following:

    - Double-click the comment text box.
    - Select the text box and press Enter.
    - Right-click the text box to display its menu, and click Edit.

2. Edit the comment text. You can use standard Windows shortcut keys when editing, for example Ctrl+C to copy text. Other options during editing are listed in the pop-up menu for the comment.

3. Click outside the text box once to display the resizing controls, then again to complete the comment.

To resize a comment text box

1. Select the comment to display the resizing controls.

2. Click and drag a control to resize the box.

3. Click outside the text box to save the change.

To move an existing comment

If you want to move a comment but not its attached objects (if any), do one of the following:

- Move the mouse pointer over the comment, hold down the left mouse button, and drag the comment to the new position.
- Select the comment, hold down the Alt key, and move the comment using the arrow keys.

If you want to move a comment together with any nodes or nuggets to which the comment is attached:

1. Select all the objects you want to move.

2. Do one of the following:

- Move the mouse pointer over one of the objects, hold down the left mouse button, and drag the objects to the new position.
- Select one of the objects, hold down the Alt key, and move the objects using the arrow keys.

To disconnect a comment from a node or nugget

1. Select one or more comments to be disconnected.

2. Do one of the following:

- Press F3.
- Right-click a selected comment and click Disconnect on its menu.

To delete a comment

1. Select one or more comments to be deleted.

2. Do one of the following:

- Press the Delete key.
- Right-click a selected comment and click Delete on its menu.

If the comment was attached to a node or nugget, the connection line is deleted as well.

If the comment was originally a stream or SuperNode annotation that had been converted to a freestanding comment, the comment is deleted from the canvas but its text is retained on the Annotations tab for the stream or SuperNode.

To show or hide comments for a stream

1. Do one of the following:

- On the main menu, click:

    View > **Comments**

- Click the **Show/hide comments** button in the toolbar.

Listing Stream Comments

You can view a list of all the comments that have been made for a particular stream or SuperNode.

On this list, you can

- Change the order of comments
- Edit the comment text
- Change the foreground or background color of a comment

Listing Comments

To list the comments made for a stream, do one of the following:

- On the main menu, click:

  Tools > **Stream Properties** > **Comments**
- Right-click a stream in the managers pane and click **Stream Properties**, then **Comments**.
- Right-click a stream background on the canvas and click **Stream Properties**, then **Comments**.

Text. The text of the comment. Double-click the text to change the field to an editable text box.

Links. The name of the node to which the comment is attached. If this field is empty, the comment applies to the stream.

Positioning buttons. These move a selected comment up or down in the list.

Comment Colors. To change the foreground or background color of a comment, select the comment, select the **Custom colors** check box, then select a color from the **Background** or **Foreground** list (or both). Click **Apply**, then click the stream background, to see the effect of the change. Click **OK** to save the change.

Converting Annotations to Comments

Annotations made to streams or SuperNodes can be converted into comments.

In the case of streams, the annotation is converted to a freestanding comment (that is, it is not attached to any nodes) on the stream canvas.

When a SuperNode annotation is converted to a comment, the comment is not attached to the SuperNode on the stream canvas, but is visible when you zoom in to the SuperNode.

To convert a stream annotation to a comment

1. Click **Stream Properties** on the Tools menu. (Alternatively, you can right-click a stream in the managers pane and click **Stream Properties**.)
2. Click the **Annotations** tab.
3. Select the **Show annotation as comment** check box.
4. Click **OK**.

To convert a SuperNode annotation to a comment

1. Double-click the SuperNode icon on the canvas.
2. Click the **Annotations** tab.
3. Select the **Show annotation as comment** check box.
4. Click **OK**.

Annotations

Nodes, streams, and models can be annotated in a number of ways. You can add descriptive annotations and specify a custom name. These options are useful especially when generating reports for streams added to the project pane. For nodes and model nuggets, you can also add ToolTip text to help distinguish between similar nodes on the stream canvas.

Adding annotations

Editing a node or model nugget opens a tabbed dialog box containing an Annotations tab used to set a variety of annotation options. You can also open the Annotations tab directly.

1. To annotate a node or nugget, right-click the node or nugget on the stream canvas and click **Rename and Annotate**. The editing dialog box opens with the Annotations tab visible.
2. To annotate a stream, click **Stream Properties** on the Tools menu. (Alternatively, you can right-click a stream in the managers pane and click **Stream Properties**.) Click the Annotations tab.

Name. Select **Custom** to adjust the autogenerated name or to create a unique name for the node as displayed on the stream canvas.

Tooltip text. (For nodes and model nuggets only) Enter text used as a tooltip on the stream canvas. This is particularly useful when working with a large number of similar nodes.

Keywords. Specify keywords to be used in project reports and when searching for nodes in a stream, or tracking objects stored in the repository (see "About the IBM SPSS Collaboration and Deployment Services Repository " on page 195). Multiple keywords can be separated by semicolons--for example, `income; crop type; claim value`. White spaces at the beginning and end of each keyword are trimmed--for example, `income ; crop type` will produce the same results as `income;crop type`. (White spaces within keywords are not trimmed, however. For example, `crop type` with one space and `crop  type` with two spaces are not the same.)

The main text area can be used to enter lengthy annotations regarding the operations of the node or decisions made in the node. For example, when you are sharing and reusing streams, it is helpful to take notes on decisions such as discarding a field with numerous blanks using a Filter node. Annotating the node stores this information with the node. You can also choose to include these annotations in a project report created from the project pane. See the topic "Introduction to Projects" on page 217 for more information.

Show annotation as comment. (For stream and SuperNode annotations only) Check this box to convert the annotation to a freestanding comment that will be visible on the stream canvas. See the topic "Adding Comments and Annotations to Nodes and Streams" on page 49 for more information.

ID. Displays a unique ID that can be used to reference the node for the purpose of scripting or automation. This value is automatically generated when the node is created and will not change. Also note that to avoid confusion with the letter "O," zeros are not used in node IDs. Use the copy button at the right to copy and paste the ID into scripts or elsewhere as needed.

Saving data streams

After you create a stream, you can save it for future reuse.

To save a stream

1. On the File menu, click **Save Stream** or **Save Stream As**.
2. In the Save dialog box, browse to the folder in which you want to save the stream file.
3. Enter a name for the stream in the File Name text box.
4. Select **Add to project** if you would like to add the saved stream to the current project.

Clicking **Save** stores the stream with the extension `*.str` in the specified directory.

Automatic backup files. Each time a stream is saved, the previously saved version of the file is automatically preserved as a backup, with a hyphen appended to the filename (for example `mystream.str-`). To restore the backed-up version, simply delete the hyphen and reopen the file.

Saving States

In addition to streams, you can save **states**, which include the currently displayed stream diagram and any model nuggets that you have created (listed on the Models tab in the managers pane).

To Save a State

1. On the File menu, click:

   State > **Save State or Save State As**

2. In the Save dialog box, browse to the folder in which you want to save the state file.

Clicking **Save** stores the state with the extension *.cst* in the specified directory.

Saving Nodes

You can also save an individual node by right-clicking the node on the stream canvas and clicking **Save Node** on the pop-up menu. Use the file extension *.nod*.

Saving multiple stream objects

When you exit IBM SPSS Modeler with multiple unsaved objects, such as streams, projects, or model nuggets, you will be prompted to save before completely closing the software. If you choose to save items, a dialog box displays options for saving each object.

1. Simply select the check boxes for the objects that you want to save.

2. Click **OK** to save each object in the required location.

You will then be prompted with a standard Save dialog box for each object. After you finish saving, the application will close.

Saving Output

Tables, graphs, and reports generated from IBM SPSS Modeler output nodes can be saved in output object (*.cou*) format.

1. When viewing the output you want to save, on the output window menus click:

   File > **Save**

2. Specify a name and location for the output file.

3. Optionally, select **Add file to project** in the Save dialog box to include the file in the current project. See the topic "Introduction to Projects" on page 217 for more information.

Alternatively, you can right-click any output object listed in the managers pane and select **Save** from the pop-up menu.

Encrypting and Decrypting Information

When you save a stream, node, project, output file, or model nugget, you can encrypt it to prevent its unauthorized use. To do this, you select an extra option when saving, and add a password to the item being saved. This encryption can be set for any of the items that you save and adds extra security to them; it is not the same as the SSL encryption used if you are passing files between IBM SPSS Modeler and IBM SPSS Modeler Server.

When you try to open an encrypted item, you are prompted to enter the password. After you enter the correct password, the item is decrypted automatically and opens as usual.

To Encrypt an Item

1. In the Save dialog box, for the item to be encrypted, click **Options**. The Encryption Options dialog box opens.

2. Select **Encrypt this file**.

3. Optionally, for further security, select **Mask password**. This displays anything you enter as a series of dots.

4. Enter the password. *Warning*: If you forget the password, the file or model cannot be opened.

5. If you selected **Mask password**, re-enter the password to confirm that you entered it correctly.

6. Click **OK** to return to the Save dialog box.

Note: If you save a copy of any encryption-protected item, the new item is automatically saved in an encrypted format using the original password unless you change the settings in the Encryption Options dialog box.

Loading files

You can reload a number of saved objects in IBM SPSS Modeler:

- Streams (`.str`)
- States (`.cst`)
- Models (.gm)
- Models palette (.gen)
- Nodes (.nod)
- Output (.cou)
- Projects (`.cpj`)

Opening new files

Streams can be loaded directly from the File menu.

- On the File menu, click **Open Stream**.

All other file types can be opened using the submenu items available on the File menu. For example, to load a model, on the File menu click:

Models > **Open Model or Load Models Palette**

Opening recently used files

For quick loading of recently used files, you can use the options at the bottom of the File menu.

Select **Recent Streams**, **Recent Projects**, or **Recent States** to expand a list of recently used files.

Mapping Data Streams

Using the mapping tool, you can connect a new data source to a preexisting stream. The mapping tool will not only set up the connection but it will also help you to specify how fields in the new source will replace those in the existing stream. Instead of re-creating an entire data stream for a new data source, you can simply connect to an existing stream.

The data mapping tool allows you to join together two stream fragments and be sure that all of the (essential) field names match up properly. In essence, mapping data results simply in the creation of a new Filter node, which matches up the appropriate fields by renaming them.

There are two equivalent ways to map data:

Select replacement node. This method starts with the node to be replaced. First, you right-click the node to replace; then, using the **Data Mapping** > **Select Replacement Node** option from the pop-up menu, select the node with which to replace it.

Map to. This method starts with the node to be introduced to the stream. First, right-click the node to introduce; then, using the **Data Mapping** > **Map To** option from the pop-up menu, select the node to which it should join. This method is particularly useful for mapping to a terminal node. *Note*: You cannot map to Merge or Append nodes. Instead, you should simply connect the stream to the Merge node in the normal manner.

Data mapping is tightly integrated into stream building. If you try to connect to a node that already has a connection, you will be offered the option of replacing the connection or mapping to that node.

Mapping Data to a Template

To replace the data source for a template stream with a new source node bringing your own data into IBM SPSS Modeler, you should use the **Select Replacement Node** option from the Data Mapping pop-up menu. This option is available for all nodes except Merge, Aggregate, and all terminal nodes. Using the data mapping tool to perform this action helps ensure that fields are matched properly between the existing stream operations and the new data source. The following steps provide an overview of the data mapping process.

Step 1: Specify essential fields in the original source node. In order for stream operations to run properly, essential fields should be specified. See the topic "Specifying Essential Fields" on page 57 for more information.

Step 2: Add new data source to the stream canvas. Using one of the source nodes, bring in the new replacement data.

Step 3: Replace the template source node. Using the Data Mapping option on the pop-up menu for the template source node, click **Select Replacement Node**, then select the source node for the replacement data.

Step 4: Check mapped fields. In the dialog box that opens, check that the software is mapping fields properly from the replacement data source to the stream. Any unmapped essential fields are displayed in red. These fields are used in stream operations and must be replaced with a similar field in the new data source in order for downstream operations to function properly. See the topic "Examining Mapped Fields" on page 58 for more information.

After using the dialog box to ensure that all essential fields are properly mapped, the old data source is disconnected and the new data source is connected to the stream using a Filter node called *Map*. This Filter node directs the actual mapping of fields in the stream. An *Unmap* Filter node is also included on the stream canvas. The *Unmap* Filter node can be used to reverse field name mapping by adding it to the stream. It will undo the mapped fields, but note that you will have to edit any downstream terminal nodes to reselect the fields and overlays.

Mapping between Streams

Similar to connecting nodes, this method of data mapping does not require you to set essential fields beforehand. With this method, you simply connect from one stream to another using **Map to** from the Data Mapping pop-up menu. This type of data mapping is useful for mapping to terminal nodes and copying and pasting between streams. *Note*: Using the **Map to** option, you cannot map to Merge, Append, and all types of source nodes.

To Map Data between Streams

1. Right-click the node that you want to use for connecting to the new stream.
2. On the menu, click:

    Data Mapping > **Map to**
3. Use the cursor to select a destination node on the target stream.
4. In the dialog box that opens, ensure that fields are properly matched and click **OK**.

Specifying Essential Fields

When mapping to an existing stream, essential fields will typically be specified by the stream author. These essential fields indicate whether a particular field is used in downstream operations. For example, the existing stream may build a model that uses a field called *Churn*. In this stream, *Churn* is an essential field because you could not build the model without it. Likewise, fields used in manipulation nodes, such as a Derive node, are necessary to derive the new field. Explicitly setting such fields as essential helps to ensure that the proper fields in the new source node are mapped to them. If mandatory fields are not mapped, you will receive an error message. If you decide that certain manipulations or output nodes are unnecessary, you can delete the nodes from the stream and remove the appropriate fields from the Essential Fields list.

To Set Essential Fields

1. Right-click the source node of the template stream that will be replaced.
2. On the menu, click:

    Data Mapping > **Specify Essential Fields**

3. Using the Field Chooser, you can add or remove fields from the list. To open the Field Chooser, click the icon to the right of the fields list.

Examining Mapped Fields

Once you have selected the point at which one data stream or data source will be mapped to another, a dialog box is displayed for you to select fields for mapping or to ensure that the system default mapping is correct. If essential fields have been set for the stream or data source and they are unmatched, these fields are displayed in red. Any unmapped fields from the data source will pass through the Filter node unaltered, but note that you can map non-essential fields as well.

Original. Lists all fields in the template or existing stream—all of the fields that are present further downstream. Fields from the new data source will be mapped to these fields.

Mapped. Lists the fields selected for mapping to template fields. These are the fields whose names may have to change to match the original fields used in stream operations. Click in the table cell for a field to activate a list of available fields.

If you are unsure of which fields to map, it may be useful to examine the source data closely before mapping. For example, you can use the Types tab in the source node to review a summary of the source data.

Tips and Shortcuts

Work quickly and easily by familiarizing yourself with the following shortcuts and tips:

- **Build streams quickly by double-clicking.** Simply double-click a node on the palette to add and connect it to the current stream.
- **Use key combinations to select downstream nodes.** Press Ctrl+Q and Ctrl+W to toggle the selection of all nodes downstream.
- **Use shortcut keys to connect and disconnect nodes.** When a node is selected in the canvas, press F2 to begin a connection, press Tab to move to the required node, and press Shift+Spacebar to complete the connection. Press F3 to disconnect all inputs and outputs to the selected node.
- **Customize the Nodes Palette tab with your favorite nodes.** On the Tools menu, click **Manage Palettes** to open a dialog box for adding, removing, or moving the nodes shown on the Nodes Palette.
- **Rename nodes and add ToolTips.** Each node dialog box includes an Annotations tab on which you can specify a custom name for nodes on the canvas as well as add ToolTips to help organize your stream. You can also include lengthy annotations to track progress, save process details, and denote any business decisions required or achieved.
- **Insert values automatically into a CLEM expression.** Using the Expression Builder, accessible from a variety of dialog boxes (such as those for Derive and Filler nodes), you can automatically insert field values into a CLEM expression. Click the values button on the Expression Builder to choose from existing field values.

Figure 13. Values button

- **Browse for files quickly.** When browsing for files on an Open dialog box, use the File list (click the yellow diamond button at the top of the dialog box, next to the Look In field) to access previously used directories as well as IBM SPSS Modeler default directories. Use the forward and back buttons to scroll through accessed directories.

- **Minimize output window clutter.** You can close and delete output quickly using the red *X* button at the top right corner of all output windows. This enables you to keep only promising or interesting results on the Outputs tab of the managers pane.

A full range of keyboard shortcuts is available for the software. See the topic "Keyboard Accessibility" on page 242 for more information.

Did you know that you can...

- Drag and select a group of nodes on the stream canvas using your mouse.
- Copy and paste nodes from one stream to another.
- Access Help from every dialog box and output window.
- Get Help on CRISP-DM, the Cross-Industry Standard Process for Data Mining. (On the Help menu, click **CRISP-DM Help**.)

Chapter 6. Building charts

Required service
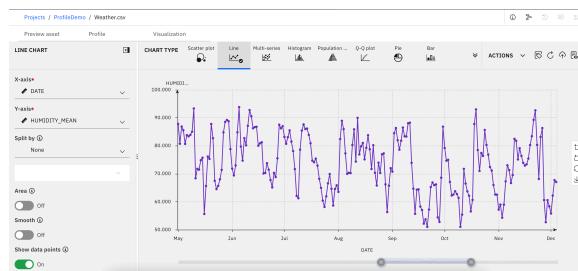> Watson Studio

Data format
> Tabular: Avro, CSV, JSON, Parquet, TSV, SAV, Microsoft Excel .xls and .xlsx files, SAS, delimited text files, and connected data.
>
> For more information about supported data sources, see Connectors.

Data size
> No limit

You can create graphics similar to the following example that shows how humidity values vary over time.

Starting the Chart builder

- Right-click the data node that you want to work with and then select **View Data**.

Chart catalog

The Chart catalog provides options for finding suitable charts by browsing chart categories, using keyword search, or filtering by chart characteristics. The Chart catalog can be accessed from any active dataset tab.

From any active dataset tab click the **Chart builder** control that is available in the toolbar. Alternately, access the catalog through **Visualize** > **Chart builder...**. There are various methods you can use to search for suitable chart types.

- The **Find chart type** field can be used to search for specific charts and keywords. As you enter a chart name or keywords, the number of available chart tiles filters to provide results for only the specified chart names or keywords.

  Note: You can use the tile view and list view icons next to the **Find chart type** field to control how chart types are presented in the Chart catalog.

- The Chart catalog tiles are grouped according to the their corresponding categories. You can scroll through the available tiles until you locate the desired chart type.

- The **Filter** pane provides options for filtering charts according to their corresponding categories, purpose, dependent variables, and statistics.

  Notes:

  - The number next to each **Filter** pane heading identifies the number of filters that are currently selected for each section.

  - Click **Reset filter** to clear the **Filter** pane settings.

Layout and terms

Canvas
The canvas is the area of the Chart Builder dialog where you build the chart.

Chart type
Lists the available chart types. The graphic elements are the items in the chart that represent data (bars, points, lines, and so on).

Details pane
The Details pane provides the basic chart building blocks.

Chart settings
Provides options for selecting which variables are used to build the chart, distribution method, title and subtitle fields, and so on. Depending on the selected chart type, the Details pane options might vary. For more information, see "Chart types" on page 62.

Actions
Provides options for downloading chart configuration files, downloading charts as image files, resetting charts, and setting the global chart preferences.

Building a chart from the chart type gallery

Use chart type gallery for building charts. Following are general steps for building a chart from the gallery.

1. In the **Chart Type** section, select a chart category. A preview version of the selected chart type is shown on the chart canvas.

2. If the canvas already displays a chart, the new chart replaces the chart's axis set and graphic elements.

   a. Depending on the selected chart type, the available variables are presented under a number of different headings in the Details pane (for example, **Category** for bar charts, **X-axis** and **Y-axis** for line charts). Select the appropriate variables for the selected chart type.

Chart types

The gallery contains a collection of the most commonly used charts.

The **Visualization** tab contains a collection of the most commonly used charts.

3D charts

3D charts are commonly used to represent multiple-variable functions and include a z-axis variable that is a function of both the x and y-axis variables.

Creating a simple 3D chart

1. In the **Chart Type** section, click the **3D** icon.

   The canvas updates to display a 3D chart template.

2. Select the chart **Type** from the drop-down list.

3. Select an **X-axis** variable from the drop-down list.

4. Select an **Y-axis** variable from the drop-down list.

5. Select an **Z-axis** variable from the drop-down list.

Options

Type
Lists the chart types that are available to represent the data.

X-axis

Lists variables that are available for the chart's X-axis.

Y-axis

Lists variables that are available for the chart's Y-axis.

Z-axis

Lists variables that are available for the chart's Z-axis.

Tooltip info

Lists the variables that can be used the generate tooltip information when the cursor hovers over a data point.

Color map

Lists available color map variables. These variables use color progression, based on the range of values in the specified column, to represent themselves in the plot points. Color maps are also known as choropleth maps.

Size map

Lists available size map variables. These variables use differing sizes to represent themselves in the plot points.

Z ratio

Sets the scale of the Z-axis data values, relative to the X and Y axes.

Rotate

Enables and disables chart rotation.

Data point tooltips

Controls where the data point tooltips display (right of data points, top-right of chart, or hide).

Primary title

The chart title.

Subtitle

The chart subtitle.

Footnote

The chart footnote.

Bar charts

Bar charts are useful for summarizing categorical variables. For example, you can use a bar chart to show the number of men and the number of women who participated in a survey. You can also use a bar chart to show the mean salary for men and the mean salary for women.

Creating a simple bar chart

1. In the **Chart Type** section, click the **Bar** icon.

   Bar ▪
   ▌▃▐▤

   The canvas updates to display a bar chart template.

2. Select a categorical (nominal or ordinal) variable as the **Category** variable. You can use a scale variable, but the results are useful in only a few special cases. A bar chart looks best with a limited number of distinct values. If you create a bar chart with a scale **Category** axis, the bars are thin because each bar is drawn at an exact value, and the bar cannot overlap other continuous values.

3. Select a statistic from the **Summary** list. The result of any statistic determines the height of the bars. If the statistic you want does not appear in the **Summary** list, it might require a variable. Select a variable from the **Value** list and check if the statistic is now available. Other chart type limitations might exist. For example, error bar charts can be calculated only for specific statistics.

Options

Category

Lists variables that are available for the chart's X-axis.

Order based on
Select a sorting option for the categories within the variable.

Category name
Use the category labels for sorting the variable's categories. The labels appear in the chart, usually as tick or legend labels.

Category value
Use the value that is stored in the data set for sorting the variable's categories. The category's value is what identifies the category in the data set. It often differs from its label and is not necessarily descriptive. For example, the value might be a number (for example, *1*), while the label is a text description of the category (for example, *Female*).

Category order
Select the order in which variable categories are sorted.

As read
Variable categories are presented as they appear in the data set.

Ascending
Sort variable categories in ascending order.

Descending
Sort variable categories in descending order.

Summary
Select a statistical summary function for the graphic element. The result of the statistic determines the position of the graphic elements on the Y-axis. In a 2-D chart, the statistic is calculated for each value on the X-axis. In a 3-D chart, it is calculated for the intersection of values on the X-axis and Z-axis.

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

- **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.

- **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Split type
When a **Split by** variable is selected, you can choose to display the resulting category bars as either stacked or clustered. Clustering and stacking add dimensionality within the chart. Clustering splits one bar into multiple bars, and stacking creates segments in each bar. Be careful that you choose the right statistic for stacking. When the values are added (stacked), the result must make sense. For example, adding and stacking mean (averaged) values is not usually meaningful.

Bar type
Select the bar chart type from the provided options.

- X-axis

- Y-axis

- X-axis inverse

- Y-axis inverse

- Polar-angle axis
- Polar-radius axis
- Polar-rainbow

Label position

Select the chart's label position from the drop-down menu.

- none
- top
- left
- right
- bottom
- inside
- insideLeft
- insideRight
- insideTop
- insideBottom
- insideTopLeft
- insideBottomLeft
- insideTopRight
- insideBottomRight

Show reference line

The toggle control enables and disables the display of reference lines in the chart. Available options are **Min**, **Max**, and **Average**, which display reference lines at the chart's minimum, maximum and average values.

Enter a reference line value

When **Show reference line** is enabled, this setting provides the option of specifying a reference line value. Click **Add another column** to specify more reference line values.

Transpose

When enabled, the chart's X and Y axes are transposed.

Primary title

The chart title.

Subtitle

The chart subtitle.

Footnote

The chart footnote.

XAxis label

The x-axis label.

YAxis label

The y-axis label.

Box plots

A box plot chart shows the five statistics (minimum, first quartile, median, third quartile, and maximum). It is useful for displaying the distribution of a scale variable and pinpointing outliers.

Creating a simple box plot

1. In the **Chart Type** section, click the **Box plot** icon.

Box plot

The canvas updates to display a box plot chart template.

2. Select one or more scale variables as the **Columns** variable.

   Note: The statistic for a dot plot is Box plot. You cannot change this setting.

Options

Columns
Lists variables that are available for the chart's X-axis.

Click **Add another column** to add more columns.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Category order
Select the order in which variable categories are sorted.

As read
Variable categories are presented as they appear in the data set.

Ascending
Sort variable categories in ascending order.

Descending
Sort variable categories in descending order.

Strength holder of IQR
The strength holder of the inter-quartile range (N*IQR). The N default value is 1.5.

Normalize data
When enabled, this setting transforms data into a normal distribution compares data from multiple data sets or multiple columns. This setting creates 100% stacking for counts and converts statistics to percents.

Transpose
When enabled, the chart's X and Y axes are transposed.

Show the outlier
When enabled, outliers display on the chart.

Show extreme outlier style
When enabled, extreme outlier style displays on the chart.

Show label
When enabled, column labels display on the chart. Only scatter series data is supported.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Bubble charts

Bubble charts display categories in your groups as nonhierarchical packed circles. The size of each circle (bubble) is proportional to its value. Bubble charts are useful for comparing relationships in your data.

Creating a simple bubble chart

1. In the **Chart Type** section, click the **Bubble** icon.

Bubble

The canvas updates to display a bubble chart template.

2. Select a **Columns** variable from the drop-down list.

   Note: Click **Add another column** to include more column variables.

Options

Columns
Lists variables that are available for the chart.

Click **Add another column** to add more columns.

Group color
Turn on or off color groupings.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Candlestick charts

Candlestick charts are a style of financial charts that are used to describe price movements of a security, derivative, or currency. Each candlestick element typically shows one day. A one-month chart might show the 20 trading days as 20 candlesticks elements. Candlestick charts are most often used in the analysis of equity and currency price patterns and are similar to box plots.

The data set that is used to create a candlestick chart must contain open, high, low, and close values for each time period you want to display.

Creating a simple Candlestick chart

1. In the **Chart Type** section, click the **Candlestick** icon.

Candlestick

The canvas updates to display a Candlestick chart template.

2. Select a variable as the **X-axis** variable.

3. Select a variable as the **High** variable.

4. Select a variable as the **Low** variable.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Summary
When enabled, summary calculations for available for the following options.

High
> Lists variables that are available for the chart's high price value.

High field summary
> Select a statistical summary function for the selected high variable.

Low
> Lists variables that are available for the chart's low price value.

Open
> Lists variables that are available for the chart's opening price value.

Close
> Lists variables that are available for the chart's closing price value.

Volume
> Lists variables that are available for the chart's volume bars.

Category order
> Select the order in which variable categories are sorted.
>
> > **As read**
> > > Variable categories are presented as they appear in the data set.
> >
> > **Ascending**
> > > Sort variable categories in ascending order.
> >
> > **Descending**
> > > Sort variable categories in descending order.

Candlestick
> Toggles the chart data to display as either candlestick or line.

Primary title
> The chart title.

Subtitle
> The chart subtitle.

Footnote
> The chart footnote.

XAxis label
> The x-axis label.

YAxis label
> The y-axis label.

Circle packing charts

Circle packing charts display hierarchical data as a set of nested areas to visualize a large amount of hierarchically structured data. It's similar to a treemap, but uses circles instead of rectangles. Circle packing charts use containment (nesting) to display hierarchy data.

Creating a simple circle packing chart

1. In the **Chart Type** section, click the **Time plot** icon.

Circle packing

   The canvas updates to display a circle packing chart template.
2. Select a **Columns** variable from the drop-down list.

   Note: Click **Add another column** to include more column variables.

Options

Columns
> Lists variables that are available for the chart.
>
> Click **Add another column** to add more columns.

Group color
> Turn on or off color groupings.

Summary
> Select a statistical summary function (the method that is used for summarizing each category).
>
> Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.
>
> - **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.
> - **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
> This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Primary title
> The chart title.

Subtitle
> The chart subtitle.

Footnote
> The chart footnote.

Custom charts

The custom charts option provides options for pasting or editing JSON code to create the desired chart.

Creating a custom chart

1. In the **Chart Type** section, click the **Customized** icon

   .

2. Paste the JSON code that contains the chart specifications into the provided **JSON script** field in the Details**CUSTOMIZED CHART** pane.

Dendrogram charts

Dendrogram charts are similar to tree charts and are typically used to illustrate a network structure (for example, a hierarchical structure). Dendrogram charts consist of a root node that is connected to subordinate nodes through edges or branches. The last nodes in the hierarchy are called leaves.

Creating a simple dendrogram chart

1. In the **Chart Type** section, click the **Dendrogram** icon.
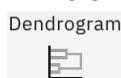
   The canvas updates to display a Dendrogram chart template.

2. Select at least two variables from the **Field name** list. You can select **SELECT ALL** to select all available variables. A maximum number 300 variables are recommended.

Options

Field name
The list provides all available variables that represent the leave nodes. You must select at least two variables.

Show axis
The toggle control enables or disables the display of the distance value axis.

Linkage
The linkage criterion determines the distance between sets of observations as a function of the pairwise distances between observations. Select **Average distance**, **Min distance**, or **Max distance**.

Tree layout

> **Left to right**
> The root node displays on the left and the leaf nodes display on the right.
>
> **Right to left**
> The root node displays on the right and the leaf nodes display on the left.
>
> **Top to bottom**
> The root node displays on the top and the leaf nodes display on the bottom.
>
> **Bottom to top**
> The root node displays on the bottom and the leaf nodes display on the top.
>
> **Radial**
> The root node displays in the middle and the leaf nodes radiate from the root.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Dual Y-axes charts

A dual Y-axes chart summarizes or plots two Y-axes variables that have different domains. For example, you can plot the number of cases on one axis and the mean salary on another. This chart can also be a mix of different graphic elements so that the dual Y-axes chart encompasses several of the different chart types. Dual Y-axes charts can display the counts as a line and the mean of each category as a bar.

Creating a simple Dual Y-axes chart

1. In the **Chart Type** section, click the **Dual Y-axes** icon.

   Dual Y-axes

   The canvas updates to display a Dual Y-axes chart template.
2. Select a variable as the **X-axis** variable.
3. Select variable for the first **Y-axis** variable and then select a chart type to represent the variable (**Bar**, **Line**, or **Scatter plot**).
4. Select variable for the second **Y-axis** variable and then select a chart type to represent the variable (**Bar**, **Line**, or **Scatter plot**).

   Note: You can use the up and down arrow controls to change the Y-axes order.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Y-axis
Lists variables that are available for the chart's dual Y-axes.

Summary
When enabled, options for selecting the method that is used for summarizing each category are displayed.

> **Left Y-axis summary**
> Sets the summary method for the Y-axis that displays on the left side of the chart. Options include **Sum**, **Mean**, **Maximum**, and **Minimum**.

> **Right Y-axis summary**
> Sets the summary method for the Y-axis that displays on the right side of the chart. Options include **Sum**, **Mean**, **Maximum**, and **Minimum**.

Normalize data
When enabled, data is transformed into a normal distribution, which allows data from multiple data sets or columns to be easily compared.

Second axis lines
When enabled, the chart's second axis line is shown.

Reorder
When enabled, the chart's data is reordered based on the X and Y axis values.

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Error bar charts

Error bar charts represent the variability of data and indicate the error (or uncertainty) in a reported measurement. Error bars help determine whether differences are statistically significant. Error bars can also suggest goodness of fit for a specific function.

Creating a simple error bar chart

1. In the **Chart Type** section, click the **Error bar** icon.

   Error bar

   The canvas updates to display an error bar chart template.

2. Select a scale variable as the **Category** variable (the variable whose data is represented on the X-axis).

3. Select a variable as the **Y-axis** variable (the variable whose data is represented on the Y-axis).

Options

Category
Lists variables that are available for the chart's X-axis.

Y-axis
Lists variables that are available for the chart's Y-axis.

Category order
Select the order in which variable categories are sorted.

> **As read**
> Variable categories are presented as they appear in the data set.
>
> **Ascending**
> Sort variable categories in ascending order.
>
> **Descending**
> Sort variable categories in descending order.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Reference line
When enabled, displays a reference line on the chart. The reference line correlates with the selected **Statistical method**.

Error bars
When enabled, the lines that represent the range of error are displayed in the chart.

Measure
Select the measure type that is represented by the error bars:

> **Confidence intervals**
> Sets the confidence intervals for the selected variables. The default value is 0.95 (95%), as reflected in the **Represent value** field.
>
> **Standard error**
> Measures the standard error of the selected variables.
>
> **Standard deviation**
> Measures the standard deviations of the selected variables.

Confidence level
This value represents the confidence intervals for the selected **Measure**. The default value is 0.95 (95%).

Statistical method
Select the method for describing the central tendency:

> **Mean**
> The result of summing the ratios and dividing the result by the total number ratios.
>
> **Median**
> The value such that number of ratios less than this value and the number of ratios greater than this value are the same.

Display mode
Select how the **Statistical method** selection displays (bar, line, or circle).

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
    The chart footnote.

XAxis label
    The x-axis label.

YAxis label
    The y-axis label.

Evaluation charts

Evaluation charts are similar to histograms or collection graphs. Evaluation charts show how accurate models are in predicting particular outcomes. They work by sorting records based on the predicted value and confidence of the prediction, splitting the records into groups of equal size (quantiles), and then plotting the value of the criterion for each quantile, from highest to lowest. Multiple models are shown as separate lines in the plot.

Outcomes are handled by defining a specific value or range of values as a "hit". Hits usually indicate success of some sort (such as a sale to a customer) or an event of interest (such as a specific medical diagnosis).

Flag
    Output fields are straightforward; hits correspond to `true` values.

Nominal
    For nominal output fields, the first value in the set defines a hit.

Continuous
    For continuous output fields, hits equal values greater than the midpoint of the field's range.

Evaluation charts can also be cumulative so that each point equals the value for the corresponding quantile plus all higher quantiles. Cumulative charts usually convey the overall performance of models better, whereas noncumulative charts often excel at indicating particular problem areas for models.

Creating a simple Evaluation chart

1. In the **Chart Type** section, click the **Evaluation** icon.

    Evaluation

    The canvas updates to display an Evaluation chart template.

2. Set the **Target field**, **Predict field** and **Confidence field** variables. The target field can be any instantiated flag or nominal field with two or more values. The predict field defines the variable that is used as the predicted value. The confidence field defines the variable that is used to establish the confidence of the prediction.

    Note: The **Predict field** variable type must match the variable type that is selected for the **Target field**.

3. Specify a custom condition used to indicate the **User defined hit**. This option is useful for defining the outcome of interest rather than deducing it from the type of target field and the order of values.

    You must specify a CLEM expression for a hit condition. For example, @TARGET = "YES" is a valid condition that indicates a value of Yes for the target field is counted as a hit in the evaluation. The specified condition is used for all target fields.

Options

Target field
    Lists instantiated flag or nominal field variables with two or more values.

User defined hit
    Specify a hit value. Hits indicate events of interest (for example, a specific medical diagnosis).

Predict field
    Lists variables that can be used as the predicted value.

Confidence field
Lists variables that can establish the confidence of the prediction.

Cumulative plot
Create a cumulative chart when enabled. Values in cumulative charts are plotted for each quantile plus all higher quantiles.

Display mode
The settings control which charts display in preview mode and in the output.

> **Single mode**
> When selected, the Model Classification Tuning chart is in the only chart that displays in preview mode and in the output.

> **Classical mode**
> When selected, the Model Classification Tuning, Cutoff, Matrix Bar, ROC, Gains, ROI, and Profit charts display in preview mode and in the output.

> **Full mode**
> When selected, the Model Classification Tuning, Cutoff, Matrix Bar, ROC, Gains, ROI, Profit, GINI, Lift, and Response charts display in preview mode and in the output.

Evaluation charts

> **Cutoff**
> The cutoff chart shows the predicted versus actual values for selected variables for a specified cutoff value.

> **Matrix Bar**
> Matrix Bar charts are a good way to determine whether linear correlations exist between multiple variables.

> **ROC**
> ROC (Receiver Operating Characteristic) evaluates the performance of classification schemes where subjects are classified for one variable with two categories.

> **Gains**
> Gains are defined as the proportion of total hits that occurs in each quantile. Gains are computed as `(number of hits in quantile / total number of hits) × 100%`.

> **ROI**
> ROI (return on investment) is similar to profit in that it involves defining revenues and costs. ROI compares profits to costs for the quantile. ROI is computed as `(profits for quantile / costs for quantile) × 100%`.

> **Profit**
> Profit equals the revenue for each record minus the cost for the record. Profits for a quantile are the sum of profits for all records in the quantile. Revenues are assumed to apply only to hits, but costs apply to all records. Profits and costs can be fixed or can be defined by fields in the data. Profits are computed as (sum of revenue for records in quantile – sum of costs for records in quantile).

> **Kolmogorov-Smirnov**
> Compares the observed cumulative distribution function for a variable with a specified theoretical distribution, which can be normal, uniform, exponential, or Poisson.

> **GINI**
> GINI measures statistical dispersion and is intended to represent the income or wealth distribution. It is the most commonly used measurement of inequality.

> **Lift**
> Lift compares the percentage of records in each quantile that are hits with the overall percentage of hits in the training data. It is computed as `(hits in quantile / records in quantile) / (total hits / total records)`.

> **Response**
> Response is the percentage of records in the quantile that are hits. Response is computed as `(hits in quantile / records in quantile) × 100%`.

Evaluation chart settings

The following settings apply only to profit and ROI charts.

Costs

Specify the fixed cost associated with each record.

Revenue

Specify the fixed revenue associated with each record that represents a hit.

Weight

If the records in your data represent more than one unit, you can use frequency weights to adjust the results. Specify the fixed weight associated with each record.

Heat map charts

Heat map charts present data where the individual values that are contained in a matrix are represented as colors.

Creating a simple heat map chart

1. In the **Chart Type** section, click the **Heat map** icon.

Heat map

The canvas updates to display a heat map chart template.

2. Select a variable as the **Column** variable. Each variable category is represented as an individual chart column.

3. Select a variable as the **Row** variable. Each variable category is represented as an individual chart row.

Options

Column

Lists variables that are available for the chart's columns. Each variable category is represented as an individual chart column.

Row

Lists variables that are available for the chart's rows. Each variable category is represented as an individual chart row.

Category order

Select the order in which variable categories are sorted.

As read

Variable categories are presented as they appear in the data set.

Ascending

Sort variable categories in ascending order.

Descending

Sort variable categories in descending order.

Summary

Select a statistical summary function for the graphic element. The result of the statistic determines the position of the graphic elements on the Y-axis. In a 2-D chart, the statistic is calculated for each value on the X-axis. In a 3-D chart, it is calculated for the intersection of values on the X-axis and Z-axis.

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

- **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.

- **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Histogram charts

A histogram is similar in appearance to a bar chart, but instead of comparing categories or looking for trends over time, each bar represents how data is distributed in a single category. Each bar represents a continuous range of data or the number of frequencies for a specific data point.

Histograms are useful for showing the distribution of a single scale variable. Data are binned and summarized by using a count or percentage statistic. A variation of a histogram is a frequency polygon, which is like a typical histogram except that the area graphic element is used instead of the bar graphic element.

Another variation of the histogram is the population pyramid. Its name is derived from its most common use: summarizing population data. When used with population data, it is split by gender to provide two back-to-back, horizontal histograms of age data. In countries with a young population, the shape of the resulting graph resembles a pyramid.

Creating a histogram chart

1. In the **Chart Type** section, click the **Histogram** icon.

   Histogram ·

   The canvas updates to display a histogram chart template.

2. Select a scale variable as the **X-axis** variable.

   Note: The statistic for a histogram is Histogram or Histogram Percent. These statistics bin the data and calculate a count for each bin.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Bin method
Specify a bin method that is used to create the chart bars. Available option include **Auto bin**, **By bin width**, and **By bin num**.

Show kde curve
    When enabled, the kernel density estimate curve is shown on the chart.

Show distribution curve
    When enabled, the distribution fitting curve is shown on the chart.

Distribution
    The drop-down list provides to the following distribution options.

    Auto fit distribution
        Automatically fits the distribution (the default setting).

    Beta
        Returns the value from a Beta distribution with specified shape parameters.

    Exponential
        Returns the value from an exponential distribution.

    Gamma
        Returns the value from the Gamma distribution, with the specified shape and scale parameters.

    Log-normal
        Returns the value from a log-normal distribution with specified parameters.

    Normal
        Returns the value from a normal distribution with specified mean and standard deviation.

    Triangular
        Returns the value from a triangular distribution with specified parameters.

    Uniform
        Returns the value from the uniform distribution between the minimum and maximum.

    Weibull
        Returns the value from a Weibull distribution with specified parameters.

Bin width
    The slider controls the size of the interval that is used to split the data into groups.

Primary title
    The chart title.

Subtitle
    The chart subtitle.

Footnote
    The chart footnote.

XAxis label
    The x-axis label.

YAxis label
    The y-axis label.

Line charts

A line chart plots a series of data points on a graph and connects them with lines. A line chart is useful for showing trend lines with subtle differences, or with data lines that cross one another. You can use a line chart to summarize categorical variables, in which case it is similar to a bar chart (see "Bar charts" on page 63). Line charts are also useful for time-series data.

Creating a simple time-series line chart

1. In the **Chart Type** section, click the **Line** icon.

    Line ·

    The canvas updates to display a line chart template.

2. Select a date variable as the **X-axis** variable.

3. Select a scale variable as the **Y-axis** variable (the variable whose values were recorded over time).

Options

X-axis
Lists variables that are available for the chart's X-axis.

Y-axis
Lists variables that are available for the chart's Y-axis.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Area
When enabled, the area beneath the line is shown in a different color.

Smooth
When enabled, the chart shows a smooth curve.

Show data points
When enabled, the data point is shown in the chart.

Reorder
The toggle control reorders data based on X and Y-axis values.

Fit line
In a fit line, the data points are fitted to a line that usually does not pass through all of the data points. The fit line represents the trend of the data. Some fits lines are regression-based. Others are based on iterative weighted least squares. Select a fit line option from the drop-down list.

Show reference line
When enabled, the option shows a reference line on the chart that is based on the specified **xAxis** and **yAxis** values.

   Enter a reference line value on xAxis
   When **Show reference line** is enabled, this setting provides the option of specifying a specific reference line value for the X-axis. Click **Add another column** to specify more reference line values.

   Enter a reference line value on yAxis
   When **Show reference line** is enabled, this setting provides the option of specifying a specific reference line value for the Y-axis. Click **Add another column** to specify more reference line values.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Map charts

Map charts are commonly used to compare values and show categories across geographical regions. Map charts are most beneficial when the data contains geographic information (countries, regions, states, counties, postal codes, and so on).

Creating a simple map chart

1. In the **Chart Type** section, click the **Map** icon.

   Map

   The canvas updates to display a map chart template.

2. Select a service to use for serving map images from the **Map service** drop-down list. The list provides options that cover specific global regions.

3. Select a map chart **Type** from the drop-down menu. The following options are available, dependent on the selected chart type:

   Longitude
   Select a variable to serve as the longitudinal value from the drop-down list.

   Latitude
   Select a variable to serve as the latitudinal value from the drop-down list.

   Group
   Select a variable that groups the data point locations from the drop-down menu.

   Category
   Select a column variable that you want to visualize.

   Summary
   Select a statistical summary function (the method that is used for summarizing each category).

   Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

   - **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.

   - **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

   Value
   This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Options

Map service
Lists the services that are available for providing map images. See "Map service options" on page 80.

Type
Lists the chart types that are available to represent the data.

Longitude
Lists the variables that are available to serve as the longitudinal value.

Latitude
Lists the variables that are available to serve as the latitudinal value.

Group
Lists the variables that can be used to group the data point locations.

Category
Lists column variables.

Summary
Select a statistical summary function (the method that is used for summarizing each category).

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

- **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.
- **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Tooltip info
Lists the variables that can be used the generate tooltip information when the cursor hovers over a data point.

Size map
Lists available size map variables. These variables use differing sizes to represent themselves in the plot points.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Map service options

On the Details pane for Map charts, you can select the map service to use for providing map images.

Configuring map service options

1. Stop the IBM SPSS Modeler Server.
2. If you're going to apply local geojson formatted map files, put them in the directory `<Modeler_installation_directory>/dataview/conf/public/mapfiles`. If the directory doesn't exist, create it.
3. Open the file `<Modeler_installation_directory>/dataview/conf/application.conf` in a text editor and add the following options to the `map.resources` section. In this example, two new maps named `World - Local Map` and `World - Online Map` are added.

```
map.resources {
  "mapservices" :
    [

      {
        "type"      : "geojson",
        "name"      : "World (Bulit-in)",
        "location"  : "built_in_world",
        "id"        : "built_in_world"
      },
      {
        "type"      : "geojson",
        "name"      : "Blank (Built-in)",
        "location"  : "built_in_blank",
        "id"        : "built_in_blank"
      },
      {
        "type"      : "geojson",
        "name"      : "World - Local Map",
        "location"  : "/mapfiles/world.json",
        "useProxy"  : "false",
        "id"        : "world_local"
      },
      {
        "type"      : "geojson",
        "name"      : "World - Online Map",
        "location"  : "http://map.example.com/map/world.json",
```

```
        "useProxy"  : "true",
        "id"        : "world_online"
    }    ]
}
```

The following parameters are available for the map options:

Option	Value	Type	Required?	Description
`type`	`geojson`	String	Yes	Currently, only the value `geojson` is supported for this option.
`name`	Map name	String	Yes	Enter a name for the map. This name is shown in the **Map service** drop-down on the **Details** pane for map charts.
`location`	Map location	String	Yes	Enter the URL or local file location of the map. For URL, it must begin with `http://` or `https://`. For local file, it must begin with the path `/mapfiles/`.
`useProxy`	`true` or `false`	String	Yes	Use `true` if the map location is a URL. Use `false` if the map location is a local file.
`id`	Map ID	String	Yes	Use an unique ID for each map.

Table 4. Map service parameters

Note:

- Don't remove the default **World (Built-in)** and **Blank (Built-in)** options.
- Remember to use a comma between the content blocks as shown in the example, and be sure your new `map.resources` section is valid JSON formatted content.
- The IBM SPSS Modeler Server (or the standalone SPSS Modeler Client without Server) must be able to access the map location.

4. Start the IBM SPSS Modeler Server.

Now while working with map charts, on the **Details** pane, you'll see the new map service you added.

Math curve charts

A math curve chart plots mathematical equation curves that are based on user-entered expressions.

Creating a simple math curve chart

1. In the **Chart Type** section, click the **Math curve** icon.

Math curve

The canvas updates to display a math curve chart template.

2. Enter a starting value for the X-axis in the **X value starts from** field.
3. Enter an ending value for the X-axis in the **X value ends with** field.
4. Enter an equation that plots the graph curve in the **equations** field. Click **Add another column** to include more equations.

Each equation treats x as an independent variable. The following equations are allowed.

- `+, -, *, /, %, and ^`
- `abs(x)`
- `ceil(x)`
- `floor(x)`
- `log(x)`
- `max(a,b,c...)`
- `min(a,b,c...)`
- `random()`
- `round(x)`
- `sqrt(x)`
- `sin`
- `cos`
- `exp`
- `tan`
- `atan`
- `atan2`
- `asin`
- `acos`

Options

X value starts from
   The X-axis starting value.

X value ends with
   The X-axis ending value.

equations
   User-entered equations that plot the graph curve.

Primary title
   The chart title.

Subtitle
   The chart subtitle.

Footnote
   The chart footnote.

XAxis label
   The x-axis label.

YAxis label
   The y-axis label.

Multi-chart charts

Multi-chart charts provide options for creating multiple charts. The charts can be of the same or different types, and can include different variables from the same data set.

Creating a simple multi-chart chart

1. In the **Chart Type** section, click the **Multi-chart** icon.

   Multi-chart

   The canvas updates to display a multi-chart chart template.
2. Click **Add another sub-chart** to add a subchart.
3. Select a chart type from the **Type** drop-down list.
4. Depending on the selected chart type, the following options are available:

   Bar and Pie charts
   Select a **Category** variable from the drop-down list.

   Line and Scatter plot charts
   Select an **X-axis** variable from the drop-down list.

   Select an **Y-axis** variable from the drop-down list.
5. Click **Add another sub-chart** to include more charts types.

Options

Title
The subchart title.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Multiple series charts

Multiple series charts are similar to line charts, with the exception that you can chart multiple variables on the Y-axis.

Creating a simple multiple series chart

1. In the **Chart Type** section, click the **Multi-series** icon.

   Multi-series

   The canvas updates to display a multiple series chart template.
2. Select a variable as the **X-axis** variable.
3. Select at least two scale variables as the **Y-axis** variables.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Y-axis
Lists variables that are available for the chart's Y-axes.

Select the chart type (bar, line, or scatter) from the drop-down list.

Click **Add another column** to include more columns on the chart.

Series style
Provides options for defining the Y-axes orientation. The following options are available.

- Default
- Separate Y axes
- Show a secondary Y axis
- Dual Y axes

Secondary Y-axis
Lists variables that are available for the chart's secondary Y-axes.

Select the chart type (bar, line, or scatter) from the drop-down list.

Click **Add another column** to include more columns on the chart.

Normalize data
When enabled, this setting transforms data into a normal distribution compares data from multiple data sets or multiple columns. This setting creates 100% stacking for counts and converts statistics to percents.

Reorder
When enabled, the chart's data is reordered based on the X and Y axis values.

Show label
When enabled, column labels display on the chart. Only scatter series data is supported.

Label field
The field menu provides variables to display as chart labels.

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Parallel charts

Parallel charts are useful for visualizing high dimensional geometry and for analyzing multivariate data. Parallel charts resemble line charts for time-series data, but the axes do not correspond to points in time (a natural order is not present).

Creating a simple parallel chart

1. In the **Chart Type** section, click the **Parallel** icon.

   Parallel

   The canvas updates to display a parallel chart template.
2. Select at least two variables as the **Columns** variables. Each column represents a vertical, parallel axis in the chart.

Note: The column order is important for finding features. In a typical data analysis, you might need to reorder the columns numerous times.

Options

Columns
Lists variables that are available for the chart's Y-axes.

Click **Add another column** to add more columns.

Color map
Lists available color map variables. These variables use color progression, based on the range of values in the specified column, to represent themselves in the plot points. Color maps are also known as choropleth maps.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Pareto charts

Pareto charts contain both bars and a line graph. The bars represent individual variable categories and the line graph represents the cumulative total.

Creating a simple Pareto chart

1. In the **Chart Type** section, click the **Pareto** icon.

   The canvas updates to display a Pareto chart template.
2. Select a variable as the **Category** variable. The selected variable categories are drawn on the chart's X-axis.

Options

Category
Lists variables that are available for the chart's X-axis.

YAxis align
The toggle control enables and disables the alignment of the two Y-axes on the chart.

Gradient bar
The toggle control enables and disables the display of color gradients in the chart bars.

Highlight vital few
The toggle control enables and disables the highlighting of variable categories that are considered vital.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Pie charts

A pie chart is useful for comparing proportions. For example, you can use a pie chart to demonstrate that a greater proportion of Europeans is enrolled in a certain class.

Creating a simple pie chart

1. In the **Chart Type** section, click the **Pie** icon.

   Pie •
   ◔

   The canvas updates to display a pie chart template.
2. Select a categorical (nominal or ordinal) variable from the **Category** list. The categories in this variable determine the number of slices in the pie chart.
3. Select a statistical summary function for the graphic element. For pie charts, you typically want a count-based statistic or a sum. The result of the statistic determines the size of each slice.

Options

Category
   Select a categorical (nominal or ordinal) variable that determines the number of slices in the pie chart.

Summary
   Select a statistical summary function for the graphic element. For pie charts, you typically want a count-based statistic or a sum. The result of the statistic determines the size of each slice.

   Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

   - **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.
   - **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
   This field displays when a **Summary** function that requires a scale variable, is selected. Select a variable to serve as the scale variable.

Pie type
   The following styles are available.

   Normal
      The pie segments display as normal slices.

   Ring
      The pie segments display as a ring. This style is also known as a doughnut chart.

   Rose
      Unlike the normal pie chart, which uses a common radius, the pie segment sizes vary depending on their value.

   Rose area
      Unlike the normal pie chart, which uses a common radius, the pie segment sizes vary depending on their area.

   Rose ring
      Unlike the normal pie chart, which uses a common radius, the pie segment sizes vary depending on their value and the segments display as a ring.

   Half rose
      Same as Rose, except the chart is represented as one-half of a pie.

Show value
When enabled, the pie slice values display in the legend.

Outer field
The list provides variables that can be used as the count value of the outer ring.

Outer label angle
The value species the location of the outer field variable value on the outer ring.

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Population pyramid charts

Population pyramid charts (also known as "age-sex pyramids") are commonly used to present and analyze population information based on age and gender.

Creating a simple Population pyramid chart

1. In the **Chart Type** section, click the **Population pyramid** icon.

   Population ...

   The canvas updates to display a Population pyramid chart template.
2. Select a **Y-axis** variable from the drop-down list.
3. Select a **Split by** variable from the drop-down list.

Options

Y-axis
Lists variables that are available for the chart's Y-axis.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Bin width
The slider controls the size of the interval that is used to split the data into groups.

Show distribution curve
When enabled, the distribution fitting curve is shown on the chart.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Q-Q plots

Q-Q (quantile-quantile) plots compare two probability distributions by plotting their quantiles against each other. A Q–Q plot is used to compare the shapes of distributions, providing a graphical view of how properties such as location, scale, and skewness are similar or different in the two distributions.

Creating a simple Q-Q plot chart

1. In the **Chart Type** section, click the **Q-Q plot** icon.

   Q-Q plot

2. Select a variable as the **X-axis** variable.

   The canvas updates to display a Q-Q plot chart template.

2. Select a variable as the **X-axis** variable.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Distribution
The drop-down list provides all available distribution methods.

> **Auto fit distribution**
> Automatically fits the distribution (the default setting).

> **Beta**
> Returns the value from a Beta distribution with specified shape parameters.

> **Exponential**
> Returns the value from an exponential distribution.

> **Gamma**
> Returns the value from the Gamma distribution, with the specified shape and scale parameters.

> **Log-normal**
> Returns the value from a log-normal distribution with specified parameters.

> **Normal**
> Returns the value from a normal distribution with specified mean and standard deviation.

> **Uniform**
> Returns the value from the uniform distribution between the minimum and maximum.

> **Student t**
> Returns the value from Student's t distribution, with specified degrees of freedom.

Plot type
Select either a Q-Q (quantile-quantile) plot or a P-P (percent-percent) plot.

Auto fit
When enabled, data parameters for the selected **Distribution** are automatically estimated. When not selected, the **Shape** and **Scale** distribution values display.

Shape1
Sets the shape1 value for the **Beta** distribution. This setting is available only when **Auto fit** is not enabled and **Beta** is selected as the **Distribution**.

Shape2
Sets the shape2 value for the **Beta** distribution. This setting is available only when **Auto fit** is not enabled and **Beta** is selected as the **Distribution**.

Shape
Sets the shape value for the selected distribution. This setting is available only when **Auto fit** is not enabled and **Gamma** or **Log-normal** is selected as the **Distribution**.

Scale
Sets the scale value for the selected distribution. This setting is available only when **Auto fit** is not enabled and **Exponential**, **Gamma**, or **Log-normal** is selected as the **Distribution**.

Mean
Sets the means value for the **Normal** distribution. This setting is available only when **Auto fit** is not enabled and **Normal** is selected as the **Distribution**.

Std. dev.
> Sets the standard deviation value for the **Normal** distribution. This setting is available only when **Normal** is selected as the **Distribution**.

Min
> Sets the minimum value for the **Uniform** distribution. This setting is available only when **Auto fit** is not enabled and **Uniform** is selected as the **Distribution**.

Max
> Sets the maximum value for the **Uniform** distribution. This setting is available only when **Auto fit** is not enabled and **Uniform** is selected as the **Distribution**.

Degrees of freedom(df)
> Set the degrees of freedom value for the **Student t** distribution. This setting is available only when **Auto fit** is not enabled and **Student t** is selected as the **Distribution**.

Detrend
> When enabled, detrended plots display on the chart.

Primary title
> The chart title.

Subtitle
> The chart subtitle.

Footnote
> The chart footnote.

XAxis label
> The x-axis label.

YAxis label
> The y-axis label.

Radar charts

Radar charts compare multiple quantitative variables and are useful for visualizing which variables have similar values, or if outliers exist among the variables. Radar charts consists of a sequence of spokes, with each spoke representing a single variable. Radar Charts are also useful for determining which variables are scoring high or low within a data set.

Creating a simple radar chart

1. In the **Chart Type** section, click the **Radar** icon.

   Radar

   The canvas updates to display a radar chart template.

2. Select a **Columns** variable from the drop-down list.

   Note: Click **Add another column** to include more columns. At least three columns variables must be defined.

Options

Category
> Select a categorical (nominal or ordinal) variable. If you select **None** as the category values, all values are shown separately, and no summary method is applied.

Summary
> When a categorical variable is selected, the following summary statistics are available.

> **Count**
>> Total number of cases.

Sum
> Sum of the values.

Mean
> Arithmetic average; the sum divided by the number of cases.

Maximum
> Largest (highest) value.

Minimum
> Smallest (lowest) value.

Radar Layout
> Determines the background image layout for the radar chart:

> **Circle**
>> When selected, the radar chart is drawn over a circular layout.

> **Polygon**
>> When selected, the radar chart is drawn over a polygonal layout.

Split by
> Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Primary title
> The chart title.

Subtitle
> The chart subtitle.

Footnote
> The chart footnote.

Relationship charts

A relationship chart is useful for determining how variables relate to each other.

Creating a simple relationship chart

1. In the **Chart Type** section, click the **Relationship** icon.

   Relationship

   The canvas updates to display a relational chart template.

2. Select at least two variables as **Columns** variables.

   Note: Click **Add another column** to include more column variables.

Options

Columns
> Lists the available variables.

> Click **Add another column** to add more columns.

Line style
> Controls the line style between related data points.

> **Curve**
>> When selected, curved lines are drawn between related data points.

> **Straight**
>> When selected, straight lines are drawn between related data points.

Label threshold
> Displays labels for data points whose values exceed the defined value.

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Scatter plots and dot plots

Several broad categories of charts are created with the point graphic element.

Scatter plots
Scatter plots are useful for plotting multivariate data. They can help you determine potential relationships among scale variables. A simple scatter plot uses a 2-D coordinate system to plot two variables. A 3-D scatter plot uses a 3-D coordinate system to plot three variables. When you need to plot more variables, you can try overlay scatter plots and scatter plot matrices (SPLOMs). An overlay scatter plot displays overlaid pairs of X-Y variables, with each pair distinguished by color or shape. A SPLOM creates a matrix of 2-D scatter plots, with each variable plotted against every other variable in the SPLOM.

Dot plots
Like histograms, dot plots are useful for showing the distribution of a single scale variable. The data are binned, but, instead of one value for each bin (like a count), all of the points in each bin are displayed and stacked. These graphs are sometimes called density plots.

Summary point plots
Summary point plots are similar to bar charts, except that points are drawn in place of the top of the bars. For more information, see "Bar charts" on page 63.

Drop-line charts
Drop-line charts are a special type of summary point plot. The points are grouped and a line is drawn through the points in each category. The drop-line chart is useful for comparing a statistic across categorical variables.

Creating a simple scatter plot

1. In the **Chart Type** section, click the **Scatter plot** icon.

   Scatter plot

   The canvas updates to display a scatter plot chart template.

2. Select a scale variable as the **X-axis** variable.

3. Select a scale variable as the **Y-axis** variable. You do not need to specify a statistic because scatter plots typically show raw values.

Options

X-axis
Lists variables that are available for the chart's X-axis.

Y-axis
Lists variables that are available for the chart's Y-axis.

Color map
Lists available color map variables. These variables use color progression, based on the range of values in the specified column, to represent themselves in the plot points. Color maps are also known as choropleth maps.

Size map
Lists available size map variables. These variables use differing sizes to represent themselves in the plot points.

Shape map
Lists available shape map variables. These variables use differing shapes to represent themselves in the plot points.

Fit line
In a fit line, the data points are fitted to a line that usually does not pass through all of the data points. The fit line represents the trend of the data. Some fits lines are regression-based. Others are based on iterative weighted least squares. Select a fit line option from the drop-down list.

Gradient bubble
The toggle control enables and disables the display of color gradients and 3D effects in the chart bubbles. The setting is not available when a **Color map** variable is selected.

Minimum bubble size
Sets the minimum bubble size. Enter a value the range 5 - 20.

Maximum bubble size
Sets the maximum bubble size. Enter a value in the range 20 - 80.

Show reference line
When enabled, the option shows a reference line on the chart that is based on the specified **xAxis** and **yAxis** values.

> **Enter a reference line value on xAxis**
> When **Show reference line** is enabled, this setting provides the option of specifying a specific reference line value for the X-axis. Click **Add another column** to specify more reference line values.

> **Enter a reference line value on yAxis**
> When **Show reference line** is enabled, this setting provides the option of specifying a specific reference line value for the Y-axis. Click **Add another column** to specify more reference line values.

Show label
When enabled, column labels display on the chart. Only scatter series data is supported.

Label field
The field menu provides variables to display as chart labels.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Scatter matrix charts

Scatter plot matrices are a good way to determine whether linear correlations exist between multiple variables.

Creating a scatter matrix chart

1. In the **Chart Type** section, click the **Scatter matrix** icon.

Scatterplot matrix

The canvas updates to display a scatter matrix chart template.

2. Select at least two scale **Columns** variables.

Note: Click **Add another column** to include more column variables.

Each selected variable is plotted against every other variable to create a matrix of individual scatter plots.

Options

Columns
Select at least two matrix variables. The variables must be numeric (but not date format).

Click **Add another column** to add more columns.

Color map
Lists available color map variables. These variables use color progression, based on the range of values in the specified column, to represent themselves in the plot points. Color maps are also known as choropleth maps.

Correlation
When enabled, linear correlation information (Strong, Medium, Weak) is shown for the selected variables.

Show kde curve
When enabled, the kernel density estimate curve is shown on the chart.

Show histogram
When enabled, histogram charts display for the selected column variables.

Gradient bubble
The toggle control enables and disables the display of color gradients and 3D effects in the chart bubbles. The setting is not available when a **Color map** variable is selected.

Minimum bubble size
Sets the minimum bubble size. Enter a value the range 5 - 20.

Maximum bubble size
Sets the maximum bubble size. Enter a value in the range 20 - 80.

Show label
When enabled, column labels display on the chart. Only scatter series data is supported.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Series array charts

Series array charts include individual sub charts and display the Y-axis for all sub charts in the legend.

Creating a series array chart

1. In the **Chart Type** section, click the **Series array** icon.

Series array

The canvas updates to display a series array chart template.

2. Click **Add another sub-chart** to define the first series array sub chart.
3. Select a variable as the **X-axis** variable.
4. Select a variable as the **Y-axis** variable.

5. Click **Add another sub-chart** to define more series array sub charts.

Options

X-axis
Lists variables that are available for the sub chart's x-axis.

Y-axis
Lists variables that are available for the sub chart's y-axis.

Split by
Select a categorical variable that creates a table of charts, with a cell for each category in the Split by variable. Like grouping, split by variables essentially add more dimensions to your chart by displaying information for each variable category.

Legend label
Specify a sub-chart legend title.

Title
Specify a sub-chart title.

Type
Select a sub-chart type. Available options are **line**, **bar**, and **scatter**.

Smooth
When enabled, the chart shows a smooth curve.

Show data points
When enabled, the data point is shown in the chart.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Sunburst charts

A sunburst chart is useful for visualizing hierarchical data structures. A sunburst chart consists of an inner circle that is surrounded by rings of deeper hierarchy levels. The angle of each segment proportional to either a value or divided equally under its inner segment. The chart segments are colored based on the category or hierarchical level to which they belong.

Creating a simple sunburst chart

1. In the **Chart Type** section, click the **Sunburst** icon.

Sunburst

◎

The canvas updates to display a sunburst chart template.

2. Select a categorical (nominal or ordinal) variable from the **Columns** list. The categories in this variable determine the number of segments in the chart.

3. Click **Add another column** and select another categorical (nominal or ordinal) variable from the **Columns** list. The categories in this variable determine the number of segments in the chart's second ring and represent a hierarchical level.

4. Select a statistical summary function for the graphic element (count-based statistic or a sum). The result of the statistic determines the size of each segment. When **Sum** is selected, choose a scale variable from the **Value** list to represent the value in the data set to summarize.

5. Select a **Sunburst layout** option (either **Traditional** or **Divergent**).

Options

Columns
Select a categorical (nominal or ordinal) variable that determines the number of segments in the chart.

Summary
Select a statistical summary function for the graphic element (count-based statistic or a sum). The result of the statistic determines the size of each slice.

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

Functions that do not require a value variable
All count and percentage statistics are in this category. These statistics are available when there is no defined **Value** variable.

Functions that do require a value variable
For example, the **Sum** function requires a variable on which the summary is calculated.

Value
This field displays when a **Summary** function that requires a scale variable, is selected. Select a variable to serve as the scale variable.

Sunburst layout
Available options are **Traditional** and **Divergent**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

t-SNE charts

T-distributed Stochastic Neighbor Embedding (t-SNE) is a machine learning algorithm for visualization. t-SNE charts model each high-dimensional object by a two-or-three dimensional point in such a way that similar objects are modeled by nearby points and dissimilar objects are modeled by distant points with high probability.

Creating a simple t-SNE chart

1. In the **Chart Type** section, click the **t-SNE** icon.

t-SNE

The canvas updates to display a t-SNE chart template.

2. Set the **Perplexity**, **Learning rate**, and **Maximum iterations** values.

3. Optional: Select a **Color map** variable.

Options

Perplexity
Sets a number that establishes an educated guess as to the number of close neighbors for each data point. The purpose is to balance the local and global aspects for your data.

Learning rate
This value affects the speed of learning by specifying the weight size changes at each iteration.

Maximum iterations
The maximum number of iterations to run.

Color map
Lists available color map variables. These variables use color progression, based on the range of values in the specified column, to represent themselves in the plot points. Color maps are also known as choropleth maps.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Time plots

Time plots illustrate data points at successive intervals of time. The time series you plot must contain numeric values and are assumed to occur over a range of time in which the periods are uniform. Time plots provide a preliminary analysis of the characteristics of time series data on basic statistics and test, and thus generate useful insights about your data before modeling. Time plots include analysis methods such as decomposition, augmented Dickey-Fuller test (ADF), correlations (ACF/PACF), and spectral analysis.

Creating a simple time plot

1. In the **Chart Type** section, click the **Time plot** icon.

Time plot

The canvas updates to display a time plot chart template.

2. From the **Values** drop-down, select a value for the y-axis.

Options

Values
Lists variables that are available for the time plot values.

Date
Select a date from the drop-down, if applicable. Each observation is separated by the same time interval. If you select a date variable, a resample option is shown. You can use this option to aggregate value fields to match the specified interval.

Time plot algorithm
The time plot algorithm to use for analyzing the time series data.

Decomposition
Decompose a time series as three components (trend-cycle, seasonal, and irregular). Decomposition is run in an additive fashion.

ADF test
Augmented Dickey–Fuller (ADF) tests the null hypothesis that a unit root exists in the series and the series is not stationary. If the test result rejects the null hypothesis, the series is stationary, or can be represented to be stationary with a difference model.

ACF/PACF
Correlations of series.

Spectral analysis
An analytic tool in the frequency domain. The highest frequency is marked as diamond.

Swap chart position
Reverses the positions of the charts on the planimetric rectangular coordinate system and the polar coordinate system.

Show the turning point
Shows or hides the turning points in the charts. The series is explored to determine whether it has an overall trend or has some turning points that change the direction of the trend pattern.

Show the outlier
Shows or hides any outliers. The outliers of the time series are analyzed from the irregular component of time series decomposition.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

XAxis label
The x-axis label.

YAxis label
The y-axis label.

Theme River charts

A theme river is a specialized flow graph that shows changes over time.

Creating a simple theme river chart

1. In the **Chart Type** section, click the **Theme River** icon.

The canvas updates to display a theme river chart template.

2. Select an **X-axis** variable.

3. Select a **Category** variable.

   Restriction: If a category field has more than 50 distinct categories, only the first 50 maximum categories are used as events in the chart.

Options

X-axis
Lists variables that are available for the X-axis.

Order based on
Depending on the X-axis value you select, you might be able to specify whether the category order is based on category name or category value.

Category order
Depending on the X-axis value you select, you might be able to specify whether the category order is ascending, descending, or as read from the data set.

Category
Lists variables that are available as categories.

Summary
Select a statistical summary function (the method that is used for summarizing each category).

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

- **Functions that do not require a value variable.** Functions that do not require a variable. All count and percentage statistics are in this category. These statistics are available when the **Value** variable is not defined.

- **Functions that do require a value variable.** Functions that do require a **Value** variable. For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when the **Value** variable is not defined.

Value
This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the value.

Legend orient
Sets the chart legend orientation. Available options are **Horizontal**, **Vertical**, and **Vertical bottom**.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Tree charts

Tree charts represent hierarchy in a tree-like structure. The structure of a Tree chart consists of a root node (has no parent node), line connections (named branches), and leaf nodes (have no child nodes). Line connections represent the relationships and connections between the members.

Creating a simple tree chart

1. In the **Chart Type** section, click the **Tree** icon.

   Tree

   ⸜⅃⸝

   The canvas updates to display a Tree chart template.

2. Select a **Columns** variable from the drop-down list.

   Note: Click **Add another column** to include more column variables.

Options

Columns
Lists variables that are available to represent chart columns.

Summary
Select a statistical summary function for the graphic element. The result of the statistic determines the position of the graphic elements.

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

Functions that do not require a value variable
All count and percentage statistics are in this category. These statistics are available when there is no defined **Value** variable.

Functions that do require a value variable
For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when there is a defined **Value** variable.

Value
This field displays when a **Summary** function that requires a value variable, is selected. Select a variable to serve as the basis for the summary value.

Tree layout

Left to right
The root node displays on the left and the leaf nodes display on the right.

Right to left
The root node displays on the right and the leaf nodes display on the left.

Top to bottom
The root node displays on the top and the leaf nodes display on the bottom.

Bottom to top
The root node displays on the bottom and the leaf nodes display on the top.

Radial
The root node displays in the middle and the leaf nodes radiate from the root.

Leaf depth
Sets the drill-down level value for the leaf nodes.

Show leaves label
When enabled, labels display for each leaf node.

Primary title
The chart title.

Subtitle
The chart subtitle.

Footnote
The chart footnote.

Treemap charts

Treemap charts are an alternative method for visualizing the hierarchical structure of tree diagrams while also displaying quantities for each category. Treemap charts are useful for identifying patterns in data. Tree branches are represented by rectangles, with each sub branch represented by smaller rectangles.

Creating a simple Treemap chart

1. In the **Chart Type** section, click the **Treemap** icon.

   Treemap

   The canvas updates to display a Treemap chart template.

2. Select a **Columns** variable from the drop-down list.

   Note: Click **Add another column** to include more column variables.

Options

Columns
Lists variables that are available to represent chart columns.

Summary
Select a statistical summary function for the graphic element. The result of the statistic determines the position of the graphic elements.

Two types of statistical summary functions are available. The distinction is important because it determines whether you need to specify a **Value** variable.

Functions that do not require a value variable

All count and percentage statistics are in this category. These statistics are available when there is no defined **Value** variable.

Functions that do require a value variable

For example, the **Mean** function requires a variable on which the mean is calculated. These statistics are available when there is a defined **Value** variable.

Value

This field displays when a **Summary** function that requires a value variable is selected. Select a variable to serve as the basis for the summary value.

Leaf depth

Sets the drill-down level value for the leaf nodes.

Primary title

The chart title.

Subtitle

The chart subtitle.

Footnote

The chart footnote.

Word cloud charts

Word cloud charts present data as words, where the size and placement of any individual word is determined by how it is weighted.

Creating a simple word cloud chart

1. In the **Chart Type** section, click the **Word cloud** icon.

Word cloud

The canvas updates to display a word cloud chart template.

2. Select a variable as the **Source** variable. Each variable category is represented in the chart based on its weight value.

3. Select a **Shape** value for the chart. The chart data is presented in the selected shape.

Options

Source

Lists variables that are available as the chart's source. Each variable category is represented in the chart based on its weight value.

Shape

Lists the available chart shapes. The chart data is presented in the selected shape.

Primary title

The chart title.

Subtitle

The chart subtitle.

Footnote

The chart footnote.

Global visualization preferences

You can override the default settings for titles, range slider, grid lines, and mouse tracking for all visualizations of a data asset. You can also specify a different color scheme template. Color theme templates are useful for data with groupings or stacking variables.

With the **Tools** options, you can use a **Range Slider** to control how much data you want displayed in a chart or to add options such as zoom, save as image, restore, select data, and clear selection. The **Aria** option when enabled, makes web content and web applications more accessible.

To change the default settings for all visualizations for a data asset:

1. On the **Visualization** tab, After right-clicking a data node and selecting **View Data**, click the **Global visualization preferences** control in the **Actions** section.

   ⚙

   Figure 14. Global visualization preferences control

   a. The **Global visualization preferences** dialog provides the following settings. These settings are applied to all visualizations for the current asset.

      Titles
      Provides global chart title settings.

      Global titles
      Enables or disables the global titles for all charts.

      Global primary title
      Enables or disables the display of global, primary chart titles. When enabled, the top-level chart title that you enter here is applied to all chart's, effectively overriding each chart's individual **Primary title** setting.

      Global subtitle
      Enables or disables the display of global chart subtitles. When enabled, the chart subtitle that you enter here is applied to all chart's, effectively overriding each chart's individual **Subtitle** setting.

      Default titles
      Enables or disables the default titles for all charts.

      Title alignment
      Provides the title alignment options **Left**, **Center** (the default setting), and **Right**.

      Tools
      Provides options that control chart behavior.

      Range slider
      Enables or disables the range slider for each chart. When enabled, you can control the amount of chart data that displays with a range slider that is provided for each chart.

      Grid lines
      Controls the display of X axis (vertical) and Y axis (horizontal) grid lines.

      Mouse tracker
      When enabled, the mouse cursor location, in relation to the chart data, is tracked and displayed when placed anywhere over the chart.

      Toolbox
      Enables or disables the toolbox for each chart. Depending on the chart type, the toolbox on the right of the screen provides tools such as zoom, save as image, restore, select data, and clear selection.

      ARIA
      When enabled, web content and web applications are more accessible to users with disabilities.

      Filter out null
      Enables or disables the filtering of null chart data.

      X axis on zero
      When enabled, the X axis lies on the other's origin position. When not enabled, the X axis always starts at 0.

Y axis on zero
> When enabled, the Y axis lies on the other's origin position. When not enabled, the Y axis always starts at 0.

Show xAxis Label
> Enables or disables the xAxis label.

Show yAxis Label
> Enables or disables the yAxis label.

Show xAxis Line
> Enables or disables the xAxis line.

Show yAxis Line
> Enables or disables the yAxis line.

Show xAxis Name
> Enables or disables the xAxis name.

Show yAxis Name
> Enables or disables the yAxis name.

yAxis Name Location
> The drop-down list provides options for specifying the yAxis name location. Options include **Start**, **Middle**, and **End**.

Truncation length
> The specified value sets the string length. Strings that are longer than the specified length are truncated. The default value is 10. When 0 is specified, truncation is turned off.

xAxis tick label decimal
> Sets the tick label decimal value for the xAxis. The default value is 3.

yAxis tick label decimal
> Sets the tick label decimal value for the yAxis. The default value is 3.

xAxis tick label rotate
> Sets the xAxis tick label rotation value. The default value is 0 (no rotation). You can specify value in the range -90 to 90 degrees.

Theme
> Select a template to change the colors that are used in charts that have a grouping or stacking variable. Any element attributes defined in the selected template file override the default template settings for those element attributes.

2. Choose the settings that you require. The settings that you select are applied to all charts.

3. Click **Apply** to save your settings or **Cancel** to disregard the changes.

The **Global visualization preferences** are applied to all visualizations for the current data asset.

For a list of chart types, see "Chart types" on page 62.

Chapter 7. Dashboard

You can create a chart dashboard layout for viewing several charts at a time. Save layouts as templates and drag-and-drop your saved charts to the positions in your layout.

1. After right-clicking a data node and selecting **View Data**, click the Dashboard control in the **Actions** section.

   Figure 15. Dashboard control

   The Dashboard displays and provides the following settings.

   Template
   In this section, you can create new layout templates or choose from predefined templates or from your saved templates.

   Choose a layout template
   Choose from available layout templates or start with a new template.

   Actions
   Click the **Edit the layout** icon to edit the layout template you selected. You can also import a dashboard file. When finished, save your template.

   Click the **Leave layout edit mode** icon when you're finished.

   Content
   Use this section to drag-and-drop items to your dashboard layout.

   Choose a saved chart
   This section provides a list of your saved charts. Drag-and-drop charts to the desired position on your dashboard layout.

   Choose an object
   You can also drag HTML text or images to your dashboard layout.

Chapter 8. Working with output

When you run some streams, the results are available in the Viewer via the **Model** tab or the **Advanced** tab of model nugget nodes. In the Viewer, you can easily navigate to the output that you want to see. You can also manipulate the output and create a document that contains precisely the output that you want. Some graph output also uses the Viewer.

The Viewer is used for the following output in IBM SPSS Modeler:

- TCM model nuggets
- STP model nuggets
- TwoStep-AS Cluster model nuggets
- GSAR model nuggets
- Map Visualization graph node

Viewer

Results are displayed in the Viewer. You can use the Viewer to:

- Browse results
- Show or hide selected tables and charts
- Change the display order of results by moving selected items
- Move items between the Viewer and other applications

The Viewer is divided into two panes:

- The left pane contains an outline view of the contents.
- The right pane contains statistical tables, charts, and text output.

You can click an item in the outline to go directly to the corresponding table or chart. You can click and drag the right border of the outline pane to change the width of the outline pane.

Showing and hiding results

In the Viewer, you can selectively show and hide individual tables or results from an entire procedure. This process is useful when you want to shorten the amount of visible output in the contents pane.

To hide tables and charts

1. Double-click the item's book icon in the outline pane of the Viewer.

   or
2. Click the item to select it.
3. From the menus choose:

   View > **Hide**

   or
4. Click the closed book (Hide) icon on the Outlining toolbar.

The open book (Show) icon becomes the active icon, indicating that the item is now hidden.

To hide procedure results

1. Click the box to the left of the procedure name in the outline pane.

This hides all results from the procedure and collapses the outline view.

Moving, deleting, and copying output

You can rearrange the results by copying, moving, or deleting an item or a group of items.

Moving output in the Viewer

1. Select the items in the outline or contents pane.
2. Drag and drop the selected items into a different location.

The item you cut in step 1, is moved to the selected location.

Deleting output in the Viewer

1. Select the items in the outline or contents pane.
2. Press the **Delete** key.

   or

3. From the menus choose:

   Edit > **Delete**

Changing initial alignment

By default, all results are initially left-aligned. To change the initial alignment of new output items:

1. From the menus choose:

   Edit > **Options**
2. Click the **Viewer** tab.
3. In the Initial Output State group, select the item type (for example, pivot table, chart, text output).
4. Select the alignment option you want.

Changing alignment of output items

1. In the outline or contents pane, select the items that you want to align.
2. From the menus choose:

   Format > **Align Left**

   or

   Format > **Center**

   or

   Format > **Align Right**

Viewer outline

The outline pane provides a table of contents of the Viewer document. You can use the outline pane to navigate through your results and control the display. Most actions in the outline pane have a corresponding effect on the contents pane.

- Selecting an item in the outline pane displays the corresponding item in the contents pane.
- Moving an item in the outline pane moves the corresponding item in the contents pane.
- Collapsing the outline view hides the results from all items in the collapsed levels.

Controlling the outline display

To control the outline display, you can:

- Expand and collapse the outline view
- Change the outline level for selected items
- Change the size of items in the outline display
- Change the font that is used in the outline display

Collapsing and expanding the outline view

1. Click the box to the left of the outline item that you want to collapse or expand.

   or
2. Click the item in the outline.
3. From the menus choose:

   View > **Collapse**

   or

View > **Expand**

Changing the outline level

1. Click the item in the outline pane.
2. From the menus choose:

   Edit > **Outline** > **Promote**

   or

   Edit > **Outline** > **Demote**

To change the size of outline items

1. From the menus choose:

   View > **Outline Size**
2. Select the outline size (**Small**, **Medium**, or **Large**).

To change the font in the outline

1. From the menus choose:

   View > **Outline Font...**
2. Select a font.

Editing and adding items to the Viewer

In the Viewer, you can edit and add items such as titles, new text, charts, or material from other applications.

Adding a title or text

Text items that are not connected to a table or chart can be added to the Viewer.

1. Click the table, chart, or other object that will precede the title or text.
2. From the menus choose:

   Insert > **New Title**

   or

   Insert > **New Text**
3. Double-click the new object.

4. Enter the text.

To add a text file

1. In the outline pane or contents pane of the Viewer, click the table, chart, or other object that will precede the text.
2. From the menus choose:

   Insert > **Text File...**

3. Select a text file.

To edit the text, double-click it.

Pasting objects into the Viewer

Objects from other applications can be pasted into the Viewer. You can use either **Paste After** or **Paste Special**. Either type of pasting puts the new object after the currently selected object in the Viewer. Use **Paste Special** when you want to choose the format of the pasted object.

Finding and replacing information in the Viewer

1. To find or replace information in the Viewer, from the menus choose:

   Edit > **Find**

   or

   Edit > **Replace**

You can use Find and Replace to:

- Search the entire document or just the selected items.
- Search down or up from the current location.
- Search both panes or restrict the search to the contents or outline pane.
- Search for hidden items. These include any items hidden in the contents pane (for example, Notes tables, which are hidden by default) and hidden rows and columns in pivot tables.
- Restrict the search criteria to case-sensitive matches.
- Restrict the search criteria in pivot tables to matches of the entire cell contents.
- Restrict the search criteria in pivot tables to footnote markers only. This option is not available if the selection in the Viewer includes anything other than pivot tables.

Hidden Items and Pivot Table Layers

- Layers beneath the currently visible layer of a multidimensional pivot table are not considered hidden and will be included in the search area even when hidden items are not included in the search.
- Hidden items include hidden items in the contents pane (items with closed book icons in the outline pane or included within collapsed blocks of the outline pane) and rows and columns in pivot tables either hidden by default (for example, empty rows and columns are hidden by default) or manually hidden by editing the table and selectively hiding specific rows or columns. Hidden items are only included in the search if you explicitly select **Include hidden items**.
- In both cases, the hidden or nonvisible element that contains the search text or value is displayed when it is found, but the item is returned to its original state afterward.

Finding a range of values in pivot tables

To find values that fall within a specified range of values in pivot tables:

1. Activate a pivot table or select one or more pivot tables in the Viewer. Make sure that only pivot tables are selected. If any other objects are selected, the Range option is not available.

2. From the menus choose:

> **Edit** > **Find**

3. Click the **Range** tab.

4. Select the type of range: Between, Greater than or equal to, or Less than or equal to.

5. Select the value or values that define the range.

- If either value contains non-numeric characters, both values are treated as strings.

- If both values are numbers, only numeric values are searched.

- You cannot use the Range tab to replace values.

This feature is not available for legacy tables. See the topic "Legacy tables" on page 131 for more information.

Copying output into other applications

Output objects can be copied and pasted into other applications, such as a word-processing program or a spreadsheet. You can paste output in a variety of formats. Depending on the target application and the selected output objects, some or all of the following formats may be available:

Metafile. WMF and EMF metafile format. These formats are available only on Windows operating systems.

RTF (rich text format). Multiple selected objects, text output, and pivot tables can be copied and pasted in RTF format. For pivot tables, in most applications this means that the tables are pasted as tables that can then be edited in the other application. Pivot tables that are too wide for the document width will either be wrapped, scaled down to fit the document width, or left unchanged, depending on the pivot table options settings. See the topic "Pivot table options" on page 133 for more information.

Note: Microsoft Word may not display extremely wide tables properly.

Image. JPG and PNG image formats.

BIFF. Pivot tables and text output can be pasted into a spreadsheet in BIFF format. Numbers in pivot tables retain numeric precision. This format is available only on Windows operating systems.

Text. Pivot tables and text output can be copied and pasted as text. This process can be useful for applications such as e-mail, where the application can accept or transmit only text.

Microsoft Office Graphic Object. Charts that support this format can be copied to Microsoft Office applications and edited in those applications as native Microsoft Office charts. Because of differences between SPSS Statistics/SPSS Modeler charts and Microsoft Office charts, some features of SPSS Statistics/SPSS Modeler charts are not retained in the copied version. Copying multiple selected charts in Microsoft Office Graphic Object format is not supported.

If the target application supports multiple available formats, it may have a Paste Special menu item that allows you to select the format, or it may automatically display a list of available formats.

Note: Microsoft Office version 16 (or higher) is required when copying and pasting Boxplots and Histograms.

Copying and pasting multiple output objects

The following limitations apply when pasting multiple output objects into other applications:

- **RTF format**. In most applications, pivot tables are pasted as tables that can be edited in that application. Charts, trees, and model views are pasted as images.
- **Metafile and image formats**. All the selected output objects are pasted as a single object in the other application.
- **BIFF format**. Charts, trees, and model views are excluded.

Copy special

When copying and pasting large amounts of output, particularly very large pivot tables, you can improve the speed of the operation by using **Edit** > **Copy Special** to limit the number of formats copied to the clipboard.

You can also save the selected formats as the default set of formats to copy to the clipboard. This setting persists across sessions.

Copy as

You can right-click a selected object in the Output Viewer and select **Edit** > **Copy as** to copy to the most popular formats (for example, **All**, **Image**, or **Microsoft Office Graphic Object**). Selecting **Edit** > **Copy** copies **All**. Note that if **Copy as** is grayed out or not present for a selected object, this copy format is not available for that particular object.

Interactive output

Interactive output objects contain multiple, related output objects. The selection in one object can change what is displayed or highlighted in the other object. For example, selecting a row in a table might highlight an area in a map or display a chart for a different category.

Interactive output objects do not support editing features, such as changing text, colors, fonts, or table borders. The individual objects can be copied from the interactive object to the Viewer. Tables copied from interactive output can be edited in the pivot table editor.

Copying objects from interactive output

File>**Copy to Viewer** copies individual output objects to the **Viewer** window.

- The available options depend on the contents of the interactive output.
- **Chart** and **Map** create chart objects.
- **Table** creates a pivot table that can be edited in the pivot table editor.
- **Snapshot** creates an image of the current view.
- **Model** creates a copy of the current interactive output object.

Edit>**Copy Object** copies individual output objects to the clipboard.

- Pasting the copied object into the Viewer is equivalent to **File**>**Copy to Viewer**.
- Pasting the object into another application pastes the object as an image.

Zoom and Pan

For maps, you can use **View**>**Zoom** to zoom the view of the map. Within a zoomed map view, you can use **View**>**Pan** to move the view.

Print settings

File>**Print Settings** controls how interactive objects are printed.

- **Print visible view only**. Prints only the view that is currently displayed. This option is the default setting.
- **Print all views**. Prints all views contained in the interactive output.
- The selected option also determines the default action for exporting the output object.

Export output

Export Output saves Viewer output in HTML, text, Word/RTF, Excel, PowerPoint (requires PowerPoint 97 or later), and PDF formats. Charts can also be exported in a number of different graphics formats.

Note: Export to PowerPoint is available only on Windows operating systems.

Exporting Output

1. Make the Viewer the active dialog (click anywhere in the dialog).
2. Click the **Export** button on the toolbar or right-click in the output window and select **Export**.
3. Enter a file name (or prefix for charts) and select an export format.

Objects to Export. You can export all objects in the Viewer, all visible objects, or only selected objects.

Document Type. The available options are:

- **Word/RTF (*.doc).** Pivot tables are exported as Word tables with all formatting attributes intact (for example, cell borders, font styles, and background colors). Text output is exported as formatted RTF. Charts, tree diagrams, and model views are included in PNG format. Note that Microsoft Word might not display extremely wide tables properly.

- **Excel 97-2004 (*.xls)/Excel 2007 and Higher (*.xlsx).** Pivot table rows, columns, and cells are exported as Excel rows, columns, and cells, with all formatting attributes intact (for example, cell borders, font styles, and background colors). Text output is exported with all font attributes intact. Each line in the text output is a row in the Excel file, with the entire contents of the line in a single cell. Charts, tree diagrams, and model views are included in PNG format. Output can be exported as *Excel 97-2004* or *Excel 2007 and higher*.

- **HTML (*.htm).** Pivot tables are exported as HTML tables. Text output is exported as preformatted HTML. Charts, tree diagrams, and model views are embedded in the document in the selected graphic format. A browser compatible with HTML 5 is required for viewing output that is exported in HTML format.

- **Portable Document Format (*.pdf).** All output is exported as it appears in Print Preview, with all formatting attributes intact.

- **Text - Plain/UTF8/UTF16 (*.txt).** Text output formats include plain text, UTF-8, and UTF-16. Pivot tables can be exported in tab-separated or space-separated format. All text output is exported in space-separated format. For charts, tree diagrams, and model views, a line is inserted in the text file for each graphic, indicating the image file name.

- **None (Graphics Only).** Available export formats include: EPS, JPEG, TIFF, PNG, and BMP. On Windows operating systems, EMF (enhanced metafile) format is also available.

Open the containing folder. Opens the folder that contains the files that are created by the export.

HTML options

HTML export requires a browser that is compatible with HTML 5.

The following options are available for exporting output in HTML format:

Layers in pivot tables. By default, inclusion or exclusion of pivot table layers is controlled by the table properties for each pivot table. You can override this setting and include all layers or exclude all but the currently visible layer. See the topic "Table properties: printing" on page 127 for more information.

Export layered tables as interactive. Layered tables are displayed as they appear in the Viewer, and you can interactively change the displayed layer in the browser. If this option is not selected, each table layer is displayed as a separate table.

Tables as HTML. This controls style information included for exported pivot tables.

- **Export with styles and fixed column width.** All pivot table style information (font styles, background colors, etc.) and column widths are preserved.

- **Export without styles.** Pivot tables are converted to default HTML tables. No style attributes are preserved. Column width is determined automatically.

Include footnotes and captions. Controls the inclusion or exclusion of all pivot table footnotes and captions.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

Note: For HTML, you can also control the image file format for exported charts. See the topic "Graphics format options" on page 116 for more information.

To set HTML export options

1. Select **HTML** as the export format.
2. Click **Change Options**.

Web report options

A web report is an interactive document that is compatible with most browsers. Many of the interactive features of pivot tables available in the Viewer are also available in web reports.

Report Title. The title that is displayed in the header of the report. By default, the file name is used. You can specify a custom title to use instead of the file name.

Format. There are two options for report format:

- **SPSS Web Report (HTML 5)**. This format requires a browser that is compatible with HTML 5.
- **Cognos Active Report (mht)**. This format requires a browser that supports MHT format files or the Cognos Active Report application.

Exclude Objects. You can exclude selected object types from the report:

- **Text**. Text objects that are not logs. This option includes text objects that contain information about the active dataset.
- **Logs**. Text objects that contain a listing of the command syntax that was run. Log items also include warnings and error messages that are encountered by commands that do not produce any Viewer output.
- **Notes Tables**. Output from statistical and charting procedures includes a Notes table. This table contains information about the dataset that was used, missing values, and the command syntax that was used to run the procedure.
- **Warnings and Error Messages**. Warnings and error messages from statistical and charting procedures.

Restyle the tables and charts to match the Web Report. This option applies the standard web report style to all tables and charts. This setting overrides any fonts, colors, or other styles in the output as displayed in the Viewer. You cannot modify the standard web report style.

Web Server Connection. You can include the URL location of one or more application servers that are running the IBM SPSS Statistics Web Report Application Server. The web application server provides features to pivot tables, edit charts, and save modified web reports.

- Select **Use** for each application server that you want to include in the web report.
- If a web report contains a URL specification, the web report connects to that application server to provide the additional editing features.
- If you specify multiple URLs, the web report attempts to connect to each server in the order in which they are specified.

The IBM SPSS Statistics Web Report Application Server can be downloaded from http://www.ibm.com/developerworks/spssdevcentral.

Word options

The following options are available for exporting output in Word format:

Preserve break points. If you have defined break points, these settings will be preserved in the Word tables.

Include footnotes and captions. Controls the inclusion or exclusion of all pivot table footnotes and captions.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

Page Setup for Export. This opens a dialog where you can define the page size and margins for the exported document. The document width used to determine wrapping and shrinking behavior is the page width minus the left and right margins.

To set Word export options

1. Select **Word/RTF** as the export format.
2. Click **Change Options**.

Excel options

The following options are available for exporting output in Excel format:

Create a worksheet or workbook or modify an existing worksheet. By default, a new workbook is created. If a file with the specified name already exists, it will be overwritten. If you select the option to create a worksheet, if a worksheet with the specified name already exists in the specified file, it will be overwritten. If you select the option to modify an existing worksheet, you must also specify the worksheet name. (This is optional for creating a worksheet.) Worksheet names cannot exceed 31 characters and cannot contain forward or back slashes, square brackets, question marks, or asterisks.

When exporting to Excel 97-2004, if you modify an existing worksheet, charts, model views, and tree diagrams are not included in the exported output.

Location in worksheet. Controls the location within the worksheet for the exported output. By default, exported output will be added after the last column that has any content, starting in the first row, without modifying any existing contents. This is a good choice for adding new columns to an existing worksheet. Adding exported output after the last row is a good choice for adding new rows to an existing worksheet. Adding exported output starting at a specific cell location will overwrite any existing content in the area where the exported output is added.

Layers in pivot tables. By default, inclusion or exclusion of pivot table layers is controlled by the table properties for each pivot table. You can override this setting and include all layers or exclude all but the currently visible layer. See the topic "Table properties: printing" on page 127 for more information.

Include footnotes and captions. Controls the inclusion or exclusion of all pivot table footnotes and captions.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

To set Excel export options

1. Select **Excel** as the export format.
2. Click **Change Options**.

PowerPoint options

The following options are available for PowerPoint:

Layers in pivot tables. By default, inclusion or exclusion of pivot table layers is controlled by the table properties for each pivot table. You can override this setting and include all layers or exclude all but the currently visible layer. See the topic "Table properties: printing" on page 127 for more information.

Wide Pivot Tables. Controls the treatment of tables that are too wide for the defined document width. By default, the table is wrapped to fit. The table is divided into sections, and row labels are repeated for each section of the table. Alternatively, you can shrink wide tables or make no changes to wide tables and allow them to extend beyond the defined document width.

Include footnotes and captions. Controls the inclusion or exclusion of all pivot table footnotes and captions.

Use Viewer outline entries as slide titles. Includes a title on each slide that is created by the export. Each slide contains a single item that is exported from the Viewer. The title is formed from the outline entry for the item in the outline pane of the Viewer.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

Page Setup for Export. This opens a dialog where you can define the page size and margins for the exported document. The document width used to determine wrapping and shrinking behavior is the page width minus the left and right margins.

To set PowerPoint export options

1. Select **PowerPoint** as the export format.
2. Click **Change Options**.

Note: Export to PowerPoint is available only on Windows operating systems.

PDF options

The following options are available for PDF:

Embed bookmarks. This option includes bookmarks in the PDF document that correspond to the Viewer outline entries. Like the Viewer outline pane, bookmarks can make it much easier to navigate documents with a large number of output objects.

Embed fonts. Embedding fonts ensures that the PDF document will look the same on all computers. Otherwise, if some fonts used in the document are not available on the computer being used to view (or print) the PDF document, font substitution may yield suboptimal results.

Layers in pivot tables. By default, inclusion or exclusion of pivot table layers is controlled by the table properties for each pivot table. You can override this setting and include all layers or exclude all but the currently visible layer. See the topic "Table properties: printing" on page 127 for more information.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

To set PDF export options

1. Select **Portable Document Format** as the export format.
2. Click **Change Options**.

Other Settings That Affect PDF Output

Page Setup/Page Attributes. Page size, orientation, margins, content and display of page headers and footers, and printed chart size in PDF documents are controlled by page setup and page attribute options.

Table Properties/TableLooks. Scaling of wide and/or long tables and printing of table layers are controlled by table properties for each table. These properties can also be saved in TableLooks.

Default/Current Printer. The resolution (DPI) of the PDF document is the current resolution setting for the default or currently selected printer (which can be changed using Page Setup). The maximum resolution is 1200 DPI. If the printer setting is higher, the PDF document resolution will be 1200 DPI.

Note: High-resolution documents may yield poor results when printed on lower-resolution printers.

Text options

Setting text output options

The following options are available for text export:

Pivot Table Format. Pivot tables can be exported in tab-separated or space-separated format. For space-separated format, you can also control:

- **Column Width. Autofit** does not wrap any column contents, and each column is as wide as the widest label or value in that column. **Custom** sets a maximum column width that is applied to all columns in the table, and values that exceed that width wrap onto the next line in that column.
- **Row/Column Border Character.** Controls the characters used to create row and column borders. To suppress display of row and column borders, enter blank spaces for the values.

Layers in pivot tables. By default, inclusion or exclusion of pivot table layers is controlled by the table properties for each pivot table. You can override this setting and include all layers or exclude all but the currently visible layer. See the topic "Table properties: printing" on page 127 for more information.

Include footnotes and captions. Controls the inclusion or exclusion of all pivot table footnotes and captions.

Views of Models. By default, inclusion or exclusion of model views is controlled by the model properties for each model. You can override this setting and include all views or exclude all but the currently visible view. (Note: all model views, including tables, are exported as graphics.)

To set text export options

1. Select **Text** as the export format.
2. Click **Change Options**.

Images only options

The following options are available for exporting images only:

1. Make the Workbook tab the active tab (click anywhere in the tab).
2. Select the items that you want to export, click the vertical ellipsis control, and then select **Export** from the menu. You can also click the **Export** button on the toolbar. The **Export workbook** dialog displays.

   Document setting

   Type
   Select **Images only** as the document **Type**.

   Image format
   Select an image format for exported images.

   PNG
   When selected, images are exported in Portable Network Graphics (`*.png`) format. The setting provides options for selecting the image **Scale (%)** and **Color depth**.

   JPG
   When selected, images are exported in Joint Photographic Experts Group (`*.jpg`) format. The setting provides options for selecting the image **Scale (%)** and **Convert to grayscale**.

   TIFF
   When selected, images are exported in Tag Image File Format (`*.tiff`) format. The setting provides options for selecting the image **Scale (%)**.

BMP

When selected, images are exported in bitmap (`*.bmp`) format. The setting provides options for selecting the image **Scale (%)** and **Compress image to reduce file size**.

Open the containing folder

Optionally, select the setting to open the export save file location folder after export.

Objects to export

Objects to include

Select to export **All** or **Selected** (the default setting) output objects.

3. Click **Export** after specifying the image only export settings.

Graphics format options

For HTML and text documents and for exporting charts only, you can select the graphic format, and for each graphic format you can control various optional settings.

To select the graphic format and options for exported charts:

1. Select **HTML**, **Text**, or **None (Graphics only)** as the document type.
2. Select the graphic file format from the drop-down list.
3. Click **Change Options** to change the options for the selected graphic file format.

JPEG chart export options

Image size(%)
Percentage of original chart size, up to 200 percent.

Convert to grayscale
Converts colors to shades of gray.

BMP chart export options

Image size(%)
Percentage of original chart size, up to 200 percent.

Compress image to reduce file size
A lossless compression technique that creates smaller files without affecting image quality.

PNG chart export options

Image size(%)
Percentage of original chart size, up to 200 percent.

Color Depth
Determines the number of colors in the exported chart. A chart that is saved under any depth will have a minimum of the number of colors that are actually used and a maximum of the number of colors that are allowed by the depth. For example, if the chart contains three colors (red, white, and black) and you save it as 16 colors, the chart will remain as three colors.

Note: If the number of colors in the chart exceeds the number of colors for that depth, the colors will be dithered to replicate the colors in the chart.

EMF and TIFF chart export options

Image size. Percentage of original chart size, up to 200 percent.

Note: EMF (enhanced metafile) format is available only on Windows operating systems.

EPS chart export options

Image size. You can specify the size as a percentage of the original image size (up to 200 percent), or you can specify an image width in pixels (with height determined by the width value and the aspect ratio). The exported image is always proportional to the original.

Include TIFF preview image. Saves a preview with the EPS image in TIFF format for display in applications that cannot display EPS images on screen.

Fonts. Controls the treatment of fonts in EPS images.

- **Use font references.** If the fonts that are used in the chart are available on the output device, the fonts are used. Otherwise, the output device uses alternate fonts.
- **Replace fonts with curves.** Turns fonts into PostScript curve data. The text itself is no longer editable as text in applications that can edit EPS graphics. This option is useful if the fonts that are used in the chart are not available on the output device.

Viewer printing

There are two options for printing the contents of the Viewer window:

All visible output. Prints only items that are currently displayed in the contents pane. Hidden items (items with a closed book icon in the outline pane or hidden in collapsed outline layers) are not printed.

Selection. Prints only items that are currently selected in the outline and/or contents panes.

To print output and charts

1. Make the Viewer the active window (click anywhere in the window).
2. From the menus choose:

   File > **Print...**
3. Select the print settings that you want.
4. Click **OK** to print.

Print Preview

Print Preview shows you what will print on each page for Viewer documents. It is a good idea to check Print Preview before actually printing a Viewer document, because Print Preview shows you items that may not be visible by looking at the contents pane of the Viewer, including:

- Page breaks
- Hidden layers of pivot tables
- Breaks in wide tables
- Headers and footers that are printed on each page

If any output is currently selected in the Viewer, the preview displays only the selected output. To view a preview for all output, make sure nothing is selected in the Viewer.

Page Attributes: Headers and Footers

Headers and footers are the information that is printed at the top and bottom of each page. You can enter any text that you want to use as headers and footers. You can also use the toolbar in the middle of the dialog box to insert:

- Date and time
- Page numbers
- Viewer filename

- Outline heading labels
- Page titles and subtitles
- **Make Default** uses the settings specified here as the default settings for new Viewer documents. (Note: this makes the current settings on both the Header/Footer tab and the Options tab the default settings.)
- Outline heading labels indicate the first-, second-, third-, and/or fourth-level outline heading for the first item on each page.
- Page titles and subtitles print the current page titles and subtitles. These can be created with New Page Title on the Viewer Insert menu or with the TITLE and SUBTITLE commands. If you have not specified any page titles or subtitles, this setting is ignored.

  Note: Font characteristics for new page titles and subtitles are controlled on the Viewer tab of the Options dialog box (accessed by choosing Options on the Edit menu). Font characteristics for existing page titles and subtitles can be changed by editing the titles in the Viewer.

To see how your headers and footers will look on the printed page, choose Print Preview from the File menu.

To insert page headers and footers

1. Make the Viewer the active window (click anywhere in the window).
2. From the menus, choose:

   File > **Header and Footer...**
3. Enter the header and/or footer that you want to appear on each page.

Page Attributes: Options

This dialog box controls the printed chart size, the space between printed output items, and page numbering.

- **Printed Chart Size.** Controls the size of the printed chart relative to the defined page size. The chart's aspect ratio (width-to-height ratio) is not affected by the printed chart size. The overall printed size of a chart is limited by both its height and width. When the outer borders of a chart reach the left and right borders of the page, the chart size cannot increase further to fill more page height.
- **Space between items.** Controls the space between printed items. Each pivot table, chart, and text object is a separate item. This setting does not affect the display of items in the Viewer.
- **Number pages starting with.** Numbers pages sequentially, starting with the specified number.
- **Make Default.** This option uses the settings that are specified here as the default settings for new Viewer documents. (Note that this option makes the current Header/Footer settings and Options settings the default.)

To change printed chart size, page numbering, and space between printed Items

1. Make the Viewer the active window (click anywhere in the window).
2. From the menus choose:

   File > **Page Attributes...**
3. Click the **Options** tab.
4. Change the settings and click **OK**.

Saving output

The contents of the Viewer can be saved.

- **Output object (*.cou)**. This format saves the entire output container, including the graph, tabs, annotations, and so on. This format can be opened and viewed in IBM SPSS Modeler, added to projects,

and published and tracked using the IBM SPSS Collaboration and Deployment Services Repository. This format is not compatible with IBM SPSS Statistics.

- **Viewer files (*.spv)**. The format that is used to display files in the **Viewer** window. When you save to this format from a model nugget in IBM SPSS Modeler, only the content of the Viewer from the **Model** tab is saved.

To control options for saving web reports or save results in other formats (for example, text, Word, Excel), use **Export** on the **File** menu.

Saving a Viewer document

1. From the Viewer window menus choose:

    File > **Save**

2. Enter the name of the document, and then click **Save**.

    Optionally, you can do the following:

    Lock files to prevent editing in IBM SPSS Smartreader
    If a Viewer document is locked, you can manipulate pivot tables (swap rows and columns, change the displayed layer, etc.) but you cannot edit any output or save any changes to the Viewer document in IBM SPSS Smartreader (a separate product for working with Viewer documents). This setting has no effect on Viewer documents opened in IBM SPSS Statistics or IBM SPSS Modeler.

    Encrypt files with a password
    You can protect confidential information stored in a Viewer document by encrypting the document with a password. Once encrypted, the document can only be opened by providing the password. IBM SPSS Smartreader users will also be required to provide the password in order to open the file.

    To encrypt a Viewer document:

    a. Select **Encrypt file with password** in the Save Output As dialog box.

    b. Click **Save**.

    c. In the Encrypt File dialog box, provide a password and re-enter it in the Confirm password text box. Passwords are limited to 10 characters and are case-sensitive.

    Warning: Passwords cannot be recovered if they are lost. If the password is lost the file cannot be opened.

    Creating strong passwords

    - Use eight or more characters.
    - Include numbers, symbols and even punctuation in your password.
    - Avoid sequences of numbers or characters, such as "123" and "abc", and avoid repetition, such as "111aaa".
    - Do not create passwords that use personal information such as birthdays or nicknames.
    - Periodically change the password.

    Note: Storing encrypted files to an IBM SPSS Collaboration and Deployment Services Repository is not supported.

    Modifying encrypted files

    - If you open an encrypted file, make modifications to it and choose File > Save, the modified file will be saved with the same password.
    - You can change the password on an encrypted file by opening the file, repeating the steps for encrypting it, and specifying a different password in the Encrypt File dialog box.
    - You can save an unencrypted version of an encrypted file by opening the file, choosing File > Save As and deselecting **Encrypt file with password** in the Save Output As dialog box.

Note: Encrypted data files and output documents cannot be opened in versions of IBM SPSS Statistics prior to version 21. Encrypted syntax files cannot be opened in versions prior to version 22.

Store required model information with the output document
This option applies only when there are model viewer items in the output document that require auxiliary information to enable some of the interactive features. Click **More Info** to display a list of these model viewer items and the interactive features that require auxiliary information. Storing this information with the output document might substantially increase the document size. If you choose not to store this information, you can still open these output items but the specified interactive features will not be available.

Pivot tables

Pivot tables

Many results are presented in tables that can be pivoted interactively. That is, you can rearrange the rows, columns, and layers.

Manipulating a pivot table

Options for manipulating a pivot table include:

- Transposing rows and columns
- Moving rows and columns
- Creating multidimensional layers
- Grouping and ungrouping rows and columns
- Showing and hiding rows, columns, and other information
- Rotating row and column labels
- Finding definitions of terms

Activating a pivot table

Before you can manipulate or modify a pivot table, you need to **activate** the table. To activate a table:

1. Double-click the table.

   or

2. Right-click the table and from the pop-up menu choose **Edit Content**.
3. From the sub-menu choose either **In Viewer** or **In Separate Window.**

Pivoting a table

A table has three dimensions: rows, columns, and layers. A dimension can contain multiple elements (or none at all). You can change the organization of the table by moving elements between or within dimensions. To move an element, just drag and drop it where you want it.

Changing display order of elements within a dimension

To change the display order of elements within a table dimension (row, column, or layer):

1. If pivoting trays are not already on, from the Pivot Table menu choose:

   Pivot > **Pivoting Trays**

2. Drag and drop the elements within the dimension in the pivoting tray.

Moving rows and columns within a dimension element

1. In the table itself (not the pivoting trays), click the label for the row or column you want to move.
2. Drag the label to the new position.

Transposing rows and columns

If you just want to flip the rows and columns, there's a simple alternative to using the pivoting trays:

1. From the menus choose:

   Pivot > **Transpose Rows and Columns**

This has the same effect as dragging all of the row elements into the column dimension and dragging all of the column elements into the row dimension.

Grouping rows or columns

1. Select the labels for the rows or columns that you want to group together (click and drag or Shift+click to select multiple labels).
2. From the menus choose:

   Edit > **Group**

A group label is automatically inserted. Double-click the group label to edit the label text.

Note: To add rows or columns to an existing group, you must first ungroup the items that are currently in the group. Then you can create a new group that includes the additional items.

Ungrouping rows or columns

Ungrouping automatically deletes the group label.

Rotating row or column labels

You can rotate labels between horizontal and vertical display for the innermost column labels and the outermost row labels in a table.

1. From the menus choose:

   Format > **Rotate Inner Column Labels**

   or

   Format > **Rotate Outer Row Labels**

Only the innermost column labels and the outermost row labels can be rotated.

Sorting rows

To sort the rows of a pivot table:

1. Activate the table.
2. Select any cell in the column you want to use to sort on. To sort just a selected group of rows, select two or more contiguous cells in the column you want to use to sort on.
3. From the menus, choose:

   Edit > **Sort rows**

4. Select **Ascending** or **Descending** from the submenu.

- If the row dimension contains groups, sorting affects only the group that contains the selection.
- You cannot sort across group boundaries.
- You cannot sort tables with more than one item in the row dimension.

Inserting rows and columns

To insert a row or column in a pivot table:

1. Activate the table.
2. Select any cell in the table.
3. From the menus, choose:

   Insert Before

   or

   Insert After

   From the submenu, choose:

   Row

   or

   Column

- A plus sign (+) is inserted in each cell of the new row or column to prevent the new row or column from being automatically hidden because it is empty.
- In a table with nested or layered dimensions, a column or row is inserted at every corresponding dimension level.

Controlling display of variable and value labels

If variables contain descriptive variable or value labels, you can control the display of variable names and labels and data values and value labels in pivot tables.

1. Activate the pivot table.
2. From the menus, choose:

   View > **Variable labels**

   or

   View > **Value labels**

3. Select one of the follow options from the submenu:

- **Name** or **Value**. Only variable names (or values) are displayed. Descriptive labels are not displayed.
- **Label**. Only descriptive labels are displayed. Variable names (or values) are not displayed.
- **Both**. Both names (or values) and descriptive labels are displayed.

Changing the output language

To change the output language in a pivot table:

1. Activate the table
2. From the menus, choose:

   View > **Language**

3. Select one of the available languages.

Changing the language affects only text that is generated by the application, such as table titles, row and column labels, and footnote text. Variable names and descriptive variable and value labels are not affected.

Navigating large tables

To use the navigation window to navigate large tables:

1. Activate the table.
2. From the menus choose:

    View > **Navigation**

Undoing changes

You can undo the most recent change or all changes to an activated pivot table. Both actions apply only to changes made since the most recent activation of the table.

To undo the most recent change:

1. From the menus, choose:

    Edit > **Undo**

    To undo all changes:

2. From the menus, choose:

    Edit > **Restore**

Working with layers

You can display a separate two-dimensional table for each category or combination of categories. The table can be thought of as stacked in layers, with only the top layer visible.

Creating and displaying layers

To create layers:

1. If pivoting trays are not already on, from the **Pivot Table** menu, choose:

    Pivot > **Pivoting Trays**

2. Drag an element from the row or column dimension into the layer dimension.

Moving elements to the layer dimension creates a multidimensional table, but only a single two-dimensional "slice" is displayed. The visible table is the table for the top layer. For example, if a yes/no categorical variable is in the layer dimension, then the multidimensional table has two layers: one for the "yes" category and one for the "no" category.

Changing the displayed layer

1. Choose a category from the drop-down list of layers (in the pivot table itself, not the pivoting tray).

Go to layer category

Go to Layer Category allows you to change layers in a pivot table. This dialog box is particularly useful when there are many layers or the selected layer has many categories.

Showing and hiding items

Many types of cells can be hidden, including:

- Dimension labels
- Categories, including the label cell and data cells in a row or column
- Category labels (without hiding the data cells)
- Footnotes, titles, and captions

Hiding rows and columns in a table

Showing hidden rows and columns in a table

1. From the menus choose:

   View > **Show All Categories**

This displays all hidden rows and columns in the table. (If **Hide empty rows and columns** is selected in Table Properties for this table, a completely empty row or column remains hidden.)

Hiding and showing dimension labels

1. Select the dimension label or any category label within the dimension.
2. From the View menu or the pop-up menu choose **Hide Dimension Label** or **Show Dimension Label**.

Hiding and showing table titles

To hide a title:

1. Activate the pivot table.
2. Select the title.
3. From the View menu choose **Hide**.

   To show hidden titles:

4. From the View menu choose **Show All**.

TableLooks

A TableLook is a set of properties that define the appearance of a table. You can select a previously defined TableLook or create your own TableLook.

- Before or after a TableLook is applied, you can change cell formats for individual cells or groups of cells by using cell properties. The edited cell formats remain intact, even when you apply a new TableLook.
- Optionally, you can reset all cells to the cell formats that are defined by the current TableLook. This option resets any cells that were edited. If **As Displayed** is selected in the TableLook Files list, any edited cells are reset to the current table properties.
- Only table properties that are defined in the Table Properties dialog are saved in TableLooks. TableLooks do not include individual cell modifications.

To apply a TableLook

1. Activate a pivot table.
2. From the menus, choose:

   Format > **TableLooks...**

3. Select a TableLook from the list of files. To select a file from another directory, click **Browse**.
4. Click **OK** to apply the TableLook to the selected pivot table.

To edit or create a TableLook

1. In the TableLooks dialog box, select a TableLook from the list of files.
2. Click **Edit Look**.
3. Adjust the table properties for the attributes that you want, and then click **OK**.
4. Click **Save Look** to save the edited TableLook, or click **Save As** to save it as a new TableLook.

- Editing a TableLook affects only the selected pivot table. An edited TableLook is not applied to any other tables that use that TableLook unless you select those tables and reapply the TableLook.
- Only table properties that are defined in the Table Properties dialog are saved in TableLooks. TableLooks do not include individual cell modifications.

Table properties

Table Properties allows you to set general properties of a table, and set cell styles for various parts of a table. You can:

- Control general properties, such as hiding empty rows or columns and adjusting printing properties.
- Control the format and position of footnote markers.
- Determine specific formats for cells in the data area, for row and column labels, and for other areas of the table.
- Control the width and color of the lines that form the borders of each area of the table.

To change pivot table properties

1. From the menus choose:

   Format > **Table Properties...**
2. Select a tab (**General**, **Footnotes**, **Cell Formats**, **Borders**, or **Printing**).
3. Select the options that you want.
4. Click **OK** or **Apply**.

The new properties are applied to the selected pivot table.

Table properties: general

Several properties apply to the table as a whole. You can:

- Show or hide empty rows and columns. (An empty row or column has nothing in any of the data cells.)
- Control the placement of row labels, which can be in the upper left corner or nested.
- Control maximum and minimum column width (expressed in points).

To change general table properties:

1. Click the **General** tab.
2. Select the options that you want.
3. Click **OK** or **Apply**.

Set rows to display

Note: This feature only applies to legacy tables.

By default, tables with many rows are displayed in sections of 100 rows. To control the number of rows displayed in a table:

1. Select **Display table by rows**.
2. Click **Set Rows to Display**.

   or
3. From the View menu of an activated pivot table, choose **Display table by rows** and **Set Rows to Display**.

Rows to display. Controls the maximum number of rows to display at one time. Navigation controls allow you move to different sections of the table. The minimum value is 10. The default is 100.

Widow/orphan tolerance. Controls the maximum number of rows of the inner most row dimension of the table to split across displayed views of the table. For example, if there are six categories in each

group of the inner most row dimension, specifying a value of six would prevent any group from splitting across displayed views. This setting can cause the total number of rows in a displayed view to exceed the specified maximum number of rows to display.

Table properties: notes

The Notes tab of the Table Properties dialog controls footnote formatting and table comment text.

Footnotes. The properties of footnote markers include style and position in relation to text.

- The style of footnote markers is either numbers (1, 2, 3, ...) or letters (a, b, c, ...).
- The footnote markers can be attached to text as superscripts or subscripts.

Comment Text. You can add comment text to each table.

- Comment text is displayed in a tooltip when you hover over a table in the Viewer.
- Screen readers read the comment text when the table has focus.
- The tooltip in the Viewer displays only the first 200 characters of the comment, but screen readers read the entire text.
- When you export output to HTML, the comment text is used as alt text.

Table properties: cell formats

For formatting, a table is divided into areas: title, layers, corner labels, row labels, column labels, data, caption, and footnotes. For each area of a table, you can modify the associated cell formats. Cell formats include text characteristics (such as font, size, color, and style), horizontal and vertical alignment, background colors, and inner cell margins.

Cell formats are applied to areas (categories of information). They are not characteristics of individual cells. This distinction is an important consideration when pivoting a table.

For example,

- If you specify a bold font as a cell format of column labels, the column labels will appear bold no matter what information is currently displayed in the column dimension. If you move an item from the column dimension to another dimension, it does not retain the bold characteristic of the column labels.
- If you make column labels bold simply by highlighting the cells in an activated pivot table and clicking the Bold button on the toolbar, the contents of those cells will remain bold no matter what dimension you move them to, and the column labels will not retain the bold characteristic for other items moved into the column dimension.

To change cell formats:

1. Select the **Cell Formats** tab.
2. Select an Area from the drop-down list or click an area of the sample.
3. Select characteristics for the area. Your selections are reflected in the sample.
4. Click **OK** or **Apply**.

Alternating row colors

To apply a different background and/or text color to alternate rows in the Data area of the table:

1. Select **Data** from the Area drop-down list.
2. Select (check) **Alternate row color** in the Background Color group.
3. Select the colors to use for the alternate row background and text.

Alternate row colors affect only the Data area of the table. They do not affect row or column label areas.

Table properties: borders

For each border location in a table, you can select a line style and a color. If you select **None** as the style, there will be no line at the selected location.

To change table borders:

1. Click the **Borders** tab.
2. Select a border location, either by clicking its name in the list or by clicking a line in the Sample area.
3. Select a line style or select **None**.
4. Select a color.
5. Click **OK** or **Apply**.

Table properties: printing

You can control the following properties for printed pivot tables:

- Print all layers or only the top layer of the table, and print each layer on a separate page.
- Shrink a table horizontally or vertically to fit the page for printing.
- Control widow/orphan lines by controlling the minimum number of rows and columns that will be contained in any printed section of a table if the table is too wide and/or too long for the defined page size.

  Note: If a table is too long to fit on the current page because there is other output above it, but it will fit within the defined page length, the table is automatically printed on a new page, regardless of the widow/orphan setting.

- Include continuation text for tables that don't fit on a single page. You can display continuation text at the bottom of each page and at the top of each page. If neither option is selected, the continuation text will not be displayed.

To control pivot table printing properties:

1. Click the **Printing** tab.
2. Select the printing options that you want.
3. Click **OK** or **Apply**.

Cell properties

Cell properties are applied to a selected cell. You can change the font, value format, alignment, margins, and colors. Cell properties override table properties; therefore, if you change table properties, you do not change any individually applied cell properties.

To change cell properties:

1. Select the cells in the table.
2. From the Format menu or the pop-up menu choose **Cell Properties**.

Font and background

The Font and Background tab controls the font style and color and background color for the selected cells in the table.

Format value

The Format Value tab controls value formats for the selected cells. You can select formats for numbers, dates, time, or currencies, and you can adjust the number of decimal digits that are displayed.

Alignment and margins

The Alignment and Margins tab controls horizontal and vertical alignment of values and top, bottom, left, and right margins for the selected cells. **Mixed** horizontal alignment aligns the content of each cell according to its type. For example, dates are right-aligned and text values are left-aligned.

Footnotes and captions

You can add footnotes and captions to a table. You can also hide footnotes or captions, change footnote markers, and renumber footnotes.

Adding footnotes and captions

To add a caption to a table:

1. From the Insert menu choose **Caption**.

A footnote can be attached to any item in a table. To add a footnote:

1. Click a title, cell, or caption within an activated pivot table.
2. From the Insert menu choose **Footnote**.
3. Insert the footnote text in the provided area.

To hide or show a caption

To hide a caption:

1. Select the caption.
2. From the View menu choose **Hide**.

To show hidden captions:

1. From the View menu choose **Show All**.

To hide or show a footnote in a table

To hide a footnote:

1. Right-click the cell that contains the footnote reference and select **Hide Footnotes** from the pop-up menu

   or
2. Select the footnote in the footnote area of the table and select **Hide** from the pop-up menu.

   Note: For legacy tables, select the footnote area of the table, select **Edit Footnote** from the pop-up menu, and then deselect (clear) the Visible property for any footnotes you want to hide.

   If a cell contains multiple footnotes, use the latter method to selectively hide footnotes.

To hide all footnotes in the table:

1. Select all of the footnotes in the footnote area of the table (use click and drag or Shift+click to select the footnotes) and select **Hide** from the View menu.

To show hidden footnotes:

1. Select **Show All Footnotes** from the View menu.

Footnote marker

Footnote Marker changes the characters that can be used to mark a footnote. By default, standard footnote markers are sequential letters or numbers, depending on the table properties settings. You can also assign a special marker. Special markers are not affected when you renumber footnotes or switch between numbers and letters for standard markers. The display of numbers or letters for standard

markers and the subscript or superscript position of footnote markers are controlled by the Footnotes tab of the Table Properties dialog.

To change footnote markers:

1. Select a footnote.
2. From the **Format** menu, choose **Footnote Marker**.

Special markers are limited to 2 characters. Footnotes with special markers precede footnotes with sequential letters or numbers in the footnote area of the table; so changing to a special marker can reorder the footnote list.

Renumbering footnotes

When you have pivoted a table by switching rows, columns, and layers, the footnotes may be out of order. To renumber the footnotes:

1. From the Format menu choose **Renumber Footnotes**.

Editing footnotes in legacy tables

For legacy tables, you can use the Edit Footnotes dialog to enter and modify footnote text and font settings, change footnote markers, and selectively hide or delete footnotes.

When you insert a new footnote in a legacy table, the Edit Footnotes dialog automatically opens. To use the Edit Footnotes dialog to edit existing footnotes (without creating a new footnote):

Marker. By default, standard footnote markers are sequential letters or numbers, depending on the table properties settings. To assign a special marker, simply enter the new marker value in the Marker column. Special markers are not affected when you renumber footnotes or switch between numbers and letters for standard markers. The display of numbers or letters for standard markers and the subscript or superscript position of footnote markers are controlled by the Footnotes tab of the Table Properties dialog. See the topic "Table properties: notes" on page 126 for more information.

To change a special marker back to a standard marker, right-click on the marker in the Edit Footnotes dialog, select **Footnote Marker** from the pop-up menu, and select Standard marker in the Footnote Marker dialog box.

Footnote. The content of the footnote. The display reflects the current font and background settings. The font settings can be changed for individual footnotes using the Format subdialog. See the topic "Footnote font and color settings" on page 129 for more information. A single background color is applied to all footnotes and can be changed in the Font and Background tab of the Cell Properties dialog. See the topic " Font and background" on page 127 for more information.

Visible. All footnotes are visible by default. Deselect (clear) the Visible checkbox to hide a footnote.

Footnote font and color settings

For legacy tables, you can use the Format dialog to change the font family, style, size and color for one or more selected footnotes:

1. In the Edit Footnotes dialog, select (click) one or more footnotes in the Footnotes grid.
2. Click the **Format** button.

The selected font family, style, size, and colors are applied to all the selected footnotes.

Background color, alignment, and margins can be set in the Cell Properties dialog and apply to all footnotes. You cannot change these settings for individual footnotes. See the topic " Font and background" on page 127 for more information.

Data cell widths

Set Data Cell Width is used to set all data cells to the same width.

To set the width for all data cells:

1. From the menus choose:

   Format > **Set Data Cell Widths...**

2. Enter a value for the cell width.

Changing column width

1. Click and drag the column border.

Displaying hidden borders in a pivot table

For tables without many visible borders, you can display the hidden borders. This can simplify tasks like changing column widths.

1. From the View menu choose **Gridlines**.

Selecting rows, columns, and cells in a pivot table

You can select an entire row or column or a specified set of data and label cells.

To select multiple cells:

Select > **Data and Label Cells**

Printing pivot tables

Several factors can affect the way that printed pivot tables look, and these factors can be controlled by changing pivot table attributes.

- For multidimensional pivot tables (tables with layers), you can either print all layers or print only the top (visible) layer. See the topic "Table properties: printing" on page 127 for more information.
- For long or wide pivot tables, you can automatically resize the table to fit the page or control the location of table breaks and page breaks. See the topic "Table properties: printing" on page 127 for more information.

Use Print Preview on the File menu to see how printed pivot tables will look.

Controlling table breaks for wide and long tables

Pivot tables that are either too wide or too long to print within the defined page size are automatically split and printed in multiple sections. You can:

- Control the row and column locations where large tables are split.
- Specify rows and columns that should be kept together when tables are split.
- Rescale large tables to fit the defined page size.

To specify row and column breaks for printed pivot tables:

1. Activate the pivot table.
2. Click any cell in the column to the left of where you want to insert the break, or click any cell in the row before the row where you want to insert the break.
3. From the menus, choose:

   Format > **Breakpoints** > **Vertical Breakpoint**

   or

   Format > **Breakpoints** > **Horizontal Breakpoint**

1. Activate the pivot table.

2. Click any cell in the column to the left of where you want to insert the break, or click any cell in the row before the row where you want to insert the break.

3. From the menus, choose:

Format > **Breakpoints** > **Vertical Breakpoint**

or

Format > **Breakpoints** > **Horizontal Breakpoint**

To specify rows or columns to keep together:

1. Select the labels of the rows or columns that you want to keep together. Click and drag or Shift+click to select multiple row or column labels.

2. From the menus, choose:

Format > **Breakpoints** > **Keep Together**

To view breakpoints and keep together groups:

1. From the menus, choose:

Format > **Breakpoints** > **Display Breakpoints**

Breakpoints are shown as vertical or horizontal lines. Keep together groups appear as grayed out rectangular regions that are enclosed by a darker border.

Note: Displaying breakpoints and keep together groups is not supported for legacy tables.

To clear breakpoints and keep together groups

To clear a breakpoint:

1. Click any cell in the column to the left of a vertical breakpoint, or click any cell in the row above a horizontal breakpoint.

2. From the menus, choose:

Format > **Breakpoints** > **Clear Breakpoint or Group**

To clear a keep together group:

3. Select the column or row labels that specify the group.

4. From the menus, choose:

Format > **Breakpoints** > **Clear Breakpoint or Group**

All breakpoints and keep together groups are automatically cleared when you pivot or reorder any row or column. This behavior does not apply to legacy tables.

Creating a chart from a pivot table

1. Double-click the pivot table to activate it.
2. Select the rows, columns, or cells you want to display in the chart.
3. Right-click anywhere in the selected area.
4. Choose **Create Graph** from the pop-up menu and select a chart type.

Legacy tables

You can choose to render tables as legacy tables (referred to as full-featured tables in release 19) which are then fully compatible with IBM SPSS Statistics releases prior to 20. Legacy tables may render slowly and are only recommended if compatibility with releases prior to 20 is required. For information on how to create legacy tables, see "Pivot table options" on page 133.

Options

Options

Options control various settings.

To change options settings

1. From the menus, choose:

   Edit > **Options...**
2. Click the tabs for the settings that you want to change.
3. Change the settings.
4. Click **OK** or **Apply**.

General options

Maximum Number of Threads

The number of threads that multithreaded procedures use when calculating results. The **Automatic** setting is based on the number of available processing cores. Specify a lower value if you want to make more processing resources available to other applications while multithreaded procedures are running. This option is disabled in distributed analysis mode.

Output

Display a leading zero for decimal values. Displays leading zeros for numeric values that consist only of a decimal part. For example, when leading zeros are displayed, the value .123 is displayed as 0.123. This setting does not apply to numeric values that have a currency or percent format. Except for fixed ASCII files (*.dat), leading zeros are not included when the data are saved to an external file.

Measurement System. The measurement system used (points, inches, or centimeters) for specifying attributes such as pivot table cell margins, cell widths, and space between tables for printing.

Viewer options

Viewer output display options affect only new output that is produced after you change the settings. Output that is already displayed in the Viewer is not affected by changes in these settings.

Initial Output State
Controls which items are automatically displayed or hidden each time that you run a procedure and how items are initially aligned. You can control the display of the following items: log, warnings, notes, titles, pivot tables, charts, tree diagrams, and text output. You can also turn the display of commands in the log on or off. You can copy command syntax from the log and save it in a syntax file.

Note: All output items are displayed left-aligned in the Viewer. Only the alignment of printed output is affected by the justification settings. Centered and right-aligned items are identified by a small symbol.

Title
Controls the font style, size, and color for new output titles. The font **Size** list provides a set of predefined sizes, but you can manually enter other, supported size values.

Page Title
Controls the font style, size, and color for new page titles and page titles that are generated by TITLE and SUBTITLE command syntax or created by **New Page Title** on the **Insert** menu. The font **Size** list provides a set of predefined sizes, but you can manually enter other, supported size values.

Text Output
> Font that is used for text output. Text output is designed for use with a monospaced (fixed-pitch) font. If you select a proportional font, tabular output does not align properly. The font **Size** list provides a set of predefined sizes, but you can manually enter other, supported size values.

Default Page Setup
> Controls the default options for orientation and margins for printing.

Pivot table options

Pivot Table options set various options for the display of pivot tables.

TableLook

Select a TableLook from the list of files and click **OK** or **Apply**. You can use one of the TableLooks provided with IBM SPSS Statistics, or you can create your own in the Pivot Table Editor (choose **TableLooks** from the Format menu).

- **Browse.** Allows you to select a TableLook from another directory.
- **Set TableLook Directory.** Allows you to change the default TableLook directory. Use **Browse** to navigate to the directory you want to use, select a TableLook in that directory, and then select **Set TableLook Directory**.

Note: TableLooks created in earlier versions of IBM SPSS Statistics cannot be used in version 16.0 or later.

Column Widths

These options control the automatic adjustment of column widths in pivot tables.

- **Adjust for labels only.** Adjusts column width to the width of the column label. This produces more compact tables, but data values wider than the label may be truncated.
- **Adjust for labels and data for all tables.** Adjusts column width to whichever is larger: the column label or the largest data value. This produces wider tables, but it ensures that all values will be displayed.

Default Editing Mode

This option controls activation of pivot tables in the Viewer window or in a separate window. By default, double-clicking a pivot table activates all but very large tables in the Viewer window. You can choose to activate pivot tables in a separate window or select a size setting that will open smaller pivot tables in the Viewer window and larger pivot tables in a separate window.

Copying wide tables to the clipboard in rich text format

When pivot tables are pasted in Word/RTF format, tables that are too wide for the document width will either be wrapped, scaled down to fit the document width, or left unchanged.

Output options

Output options control the default setting for a number of output options.

Screen Reader Accessibility. Controls how pivot table row and column labels are read by screen readers. You can read full row and column labels for each data cell or read only the labels that change as you move between data cells in the table.

Chapter 9. Handling missing values

Overview of Missing Values

During the Data Preparation phase of data mining, you will often want to replace missing values in the data. **Missing values** are values in the data set that are unknown, uncollected, or incorrectly entered. Usually, such values are invalid for their fields. For example, the field *Sex* should contain the values *M* and *F*. If you discover the values *Y* or *Z* in the field, you can safely assume that such values are invalid and should therefore be interpreted as blanks. Likewise, a negative value for the field *Age* is meaningless and should also be interpreted as a blank. Frequently, such obviously wrong values are purposely entered, or fields left blank, during a questionnaire to indicate a nonresponse. At times, you may want to examine these blanks more closely to determine whether a nonresponse, such as the refusal to give one's age, is a factor in predicting a specific outcome.

Some modeling techniques handle missing data better than others. For example, C5.0 and Apriori cope well with values that are explicitly declared as "missing" in a Type node. Other modeling techniques have trouble dealing with missing values and experience longer training times, resulting in less-accurate models.

There are several types of missing values recognized by IBM SPSS Modeler:

- **Null or system-missing values.** These are nonstring values that have been left blank in the database or source file and have not been specifically defined as "missing" in a source or Type node. System-missing values are displayed as **$null$**. Note that empty strings are not considered nulls in IBM SPSS Modeler, although they may be treated as nulls by certain databases.

- **Empty strings and white space.** Empty string values and white space (strings with no visible characters) are treated as distinct from null values. Empty strings are treated as equivalent to white space for most purposes. For example, if you select the option to treat white space as blanks in a source or Type node, this setting applies to empty strings as well.

- **Blank or user-defined missing values.** These are values such as unknown, 99, or −1 that are explicitly defined in a source node or Type node as missing. Optionally, you can also choose to treat nulls and white space as blanks, which allows them to be flagged for special treatment and to be excluded from most calculations. For example, you can use the @BLANK function to treat these values, along with other types of missing values, as blanks.

Reading in mixed data. Note that when you are reading in fields with numeric storage (either integer, real, time, timestamp, or date), any non-numeric values are set to *null* or *system missing*. This is because, unlike some applications, does not allow mixed storage types within a field. To avoid this, any fields with mixed data should be read in as strings by changing the storage type in the source node or external application as necessary.

Reading empty strings from Oracle. When reading from or writing to an Oracle database, be aware that, unlike IBM SPSS Modeler and unlike most other databases, Oracle treats and stores empty string values as equivalent to null values. This means that the same data extracted from an Oracle database may behave differently than when extracted from a file or another database, and the data may return different results.

Handling Missing Values

You should decide how to treat missing values in light of your business or domain knowledge. To ease training time and increase accuracy, you may want to remove blanks from your data set. On the other hand, the presence of blank values may lead to new business opportunities or additional insights. In choosing the best technique, you should consider the following aspects of your data:

- Size of the data set
- Number of fields containing blanks

- Amount of missing information

In general terms, there are two approaches you can follow:

- You can exclude fields or records with missing values

- You can impute, replace, or coerce missing values using a variety of methods

Both of these approaches can be largely automated using the Data Audit node. For example, you can generate a Filter node that excludes fields with too many missing values to be useful in modeling, and generate a Supernode that imputes missing values for any or all of the fields that remain. This is where the real power of the audit comes in, allowing you not only to assess the current state of your data, but to take action based on the assessment.

Handling Records with Missing Values

If the majority of missing values is concentrated in a small number of records, you can just exclude those records. For example, a bank usually keeps detailed and complete records on its loan customers. If, however, the bank is less restrictive in approving loans for its own staff members, data gathered for staff loans is likely to have several blank fields. In such a case, there are two options for handling these missing values:

- You can use a Select node to remove the staff records.

- If the data set is large, you can discard all records with blanks.

Handling Fields with Missing Values

If the majority of missing values is concentrated in a small number of fields, you can address them at the field level rather than at the record level. This approach also allows you to experiment with the relative importance of particular fields before deciding on an approach for handling missing values. If a field is unimportant in modeling, it probably is not worth keeping, regardless of how many missing values it has.

For example, a market research company may collect data from a general questionnaire containing 50 questions. Two of the questions address age and political persuasion, information that many people are reluctant to give. In this case, *Age* and *Political_persuasion* have many missing values.

Field Measurement Level

In determining which method to use, you should also consider the measurement level of fields with missing values.

Numeric fields. For numeric field types, such as *Continuous*, you should always eliminate any non-numeric values before building a model, because many models will not function if blanks are included in numeric fields.

Categorical fields. For categorical fields, such as *Nominal* and *Flag*, altering missing values is not necessary but will increase the accuracy of the model. For example, a model that uses the field *Sex* will still function with meaningless values, such as *Y* and *Z*, but removing all values other than *M* and *F* will increase the accuracy of the model.

Screening or Removing Fields

To screen out fields with too many missing values, you have several options:

- You can use a Data Audit node to filter fields based on quality.

- You can use a Feature Selection node to screen out fields with more than a specified percentage of missing values and to rank fields based on importance relative to a specified target.

- Instead of removing the fields, you can use a Type node to set the field role to **None**. This will keep the fields in the data set but exclude them from the modeling processes.

Handling Records with System Missing Values

What are system missing values?

System missing values represent data values that are not known or not applicable. In databases, these values are often referred to as *NULL* values.

System missing values are different from blank values. Blank values are typically defined in the Type node as particular values, or ranges of values, which can be regarded as user-defined-missing. Blank values are handled differently in the context of modeling.

Constructing system missing values

System missing values might be present in data that is read from a data source (for example, database tables might contain *NULL* values).

System missing values can be constructed by using the value **undef** in expressions. For example, the following CLEM expression returns the Age, if less than or equal to 30, or a missing value if greater than 30:

```
if Age > 30 then undef else Age endif
```

Missing values can also be created when an outer join is carried out, when a number is divided by zero, when the square root of a negative number is computed, and in other situations.

Displaying system missing values

System missing values are displayed in tables and other output as $null$.

Testing for system missing values

Use the special function **@NULL** to return `true` if the argument value is a system missing value, for example:

```
if @NULL(MyFieldName) then 'It is null' else 'It is not null' endif
```

System missing values passed to functions

System missing values that are passed to functions usually propagate missing values to the output. For example, if the value of field `f1` is a system missing value in a particular row, then the expression `log(f1)` also evaluates to a system missing value for that row. An exception is the **@NULL** function.

System missing values in expressions that involve arithmetic operators

Applying arithmetic operators to values that include a system missing value results in a system missing value. For example, if the value of field `f1` is a system missing value in a particular row, then the expression `f1 + 10` also evaluates to a system missing value for that row.

System missing values in expressions that involve logical operators

When you work with system missing values in expressions that involve logical operators, the rules of three valued logic (*true*, *false*, and *missing*) apply and can be described in truth tables. The truth tables for the common logical operators of *not*, *and*, and *or* are shown in the following tables.

Table 5. Truth table for NOT	
Operand	**NOT Operand**
true	false

Table 5. Truth table for NOT (continued)	
Operand	**NOT Operand**
false	true
missing	missing

Table 6. Truth table for AND		
Operand1	**Operand2**	**Operand1 AND Operand2**
true	true	true
true	false	false
true	missing	missing
false	true	false
false	false	false
false	missing	false
missing	true	missing
missing	false	false
missing	missing	missing

Table 7. Truth table for OR		
Operand1	**Operand2**	**Operand1 OR Operand2**
true	true	true
true	false	true
true	missing	true
false	true	true
false	false	false
false	missing	missing
missing	true	true
missing	false	missing
missing	missing	missing

System missing values in expressions that involve comparison operators

When you compare a system missing value and a non-system-missing value, the outcome evaluates to a system missing value rather than a true or false result. System missing values can be compared with each other; two system missing values are considered to be equal.

System missing values in if/then/else/endif expressions

When you use conditional expressions, and the conditional expression returns a system missing value, the value from the else clause is returned from the conditional expression.

System missing values in the Select node

When, for a particular record, the selection expression evaluates to a missing value, the record is not output from the Select node (this action applies to both Include and Discard modes).

System missing values in the Merge node

When you merge by using a key, any records that have system missing values in a key field are not merged.

System missing values in aggregation

When aggregating data on columns, missing values are not included in the calculation. For example, in a column with three values { 1, 2, and undef }, the sum of the values in the column is computed as 3; the mean value is computed as 1.5.

Imputing or Filling Missing Values

In cases where there are only a few missing values, it may be useful to insert values to replace the blanks. You can do this from the Data Audit report, which allows you to specify options for specific fields as appropriate and then generate a SuperNode that imputes values using a number of methods. This is the most flexible method, and it also allows you to specify handling for large numbers of fields in a single node.

The following methods are available for imputing missing values:

Fixed. Substitutes a fixed value (either the field mean, midpoint of the range, or a constant that you specify).

Random. Substitutes a random value based on a normal or uniform distribution.

Expression. Allows you to specify a custom expression. For example, you could replace values with a global variable created by the Set Globals node.

Algorithm. Substitutes a value predicted by a model based on the C&RT algorithm. For each field imputed using this method, there will be a separate C&RT model, along with a Filler node that replaces blanks and nulls with the value predicted by the model. A Filter node is then used to remove the prediction fields generated by the model.

Alternatively, to coerce values for specific fields, you can use a Type node to ensure that the field types cover only legal values and then set the *Check* column to **Coerce** for the fields whose blank values need replacing.

CLEM Functions for Missing Values

There are several functions used to handle missing values. The following functions are often used in Select and Filler nodes to discard or fill missing values:

- `count_nulls(LIST)`
- `@BLANK(FIELD)`
- `@NULL(FIELD)`
- `undef`

The @ functions can be used in conjunction with the @FIELD function to identify the presence of blank or null values in one or more fields. The fields can simply be flagged when blank or null values are present, or they can be filled with replacement values or used in a variety of other operations.

You can count nulls across a list of fields, as follows:

```
count_nulls(['cardtenure' 'card2tenure' 'card3tenure'])
```

When using any of the functions that accept a list of fields as input, the special functions @FIELDS_BETWEEN and @FIELDS_MATCHING can be used, as shown in the following example:

```
count_nulls(@FIELDS_MATCHING('card*'))
```

You can use the undef function to fill fields with the system-missing value, displayed as **$null$**. For example, to replace any numeric value, you could use a conditional statement, such as:

```
if not(Age > 17) or not(Age < 66) then undef else Age endif
```

This replaces anything that is not in the range with a system-missing value, displayed as **$null$**. By using the not() function, you can catch all other numeric values, including any negatives. See the topic "Functions Handling Blanks and Null Values" on page 191 for more information.

Note on Discarding Records

When using a Select node to discard records, note that syntax uses three-valued logic and automatically includes null values in select statements. To exclude null values (system-missing) in a select expression, you must explicitly specify this by using and not in the expression. For example, to select and include all records where the type of prescription drug is Drug C, you would use the following select statement:

```
Drug = 'drugC' and not(@NULL(Drug))
```

Earlier versions of excluded null values in such situations.

Chapter 10. Building CLEM expressions

About CLEM

The Control Language for Expression Manipulation (CLEM) is a powerful language for analyzing and manipulating the data that flows along IBM SPSS Modeler streams. Data miners use CLEM extensively in stream operations to perform tasks as simple as deriving profit from cost and revenue data or as complex as transforming web log data into a set of fields and records with usable information.

CLEM is used within IBM SPSS Modeler to:

- Compare and evaluate conditions on record fields.
- Derive values for new fields.
- Derive new values for existing fields.
- Reason about the sequence of records.
- Insert data from records into reports.

CLEM expressions are indispensable for data preparation in IBM SPSS Modeler and can be used in a wide range of nodes—from record and field operations (Select, Balance, Filler) to plots and output (Analysis, Report, Table). For example, you can use CLEM in a Derive node to create a new field based on a formula such as ratio.

CLEM expressions can also be used for global search and replace operations. For example, the expression @NULL(@FIELD) can be used in a Filler node to replace **system-missing values** with the integer value 0. (To replace **user-missing values**, also called blanks, use the @BLANK function.)

More complex CLEM expressions can also be created. For example, you can derive new fields based on a conditional set of rules, such as a new value category created by using the following expressions: If: CardID = @OFFSET(CardID,1), Then: @OFFSET(ValueCategory,1), Else: 'exclude'.

This example uses the @OFFSET function to say, "If the value of the field *CardID* for a given record is the same as for the previous record, then return the value of the field named *ValueCategory* for the previous record. Otherwise, assign the string "exclude." In other words, if the *CardID*s for adjacent records are the same, they should be assigned the same value category. (Records with the exclude string can later be culled using a Select node.)

CLEM Examples

To illustrate correct syntax as well as the types of expressions possible with CLEM, example expressions follow.

Simple Expressions

Formulas can be as simple as this one, which derives a new field based on the values of the fields *After* and *Before*:

```
(After - Before) / Before * 100.0
```

Notice that field names are unquoted when referring to the values of the field.

Similarly, the following expression simply returns the log of each value for the field *salary*.

```
log(salary)
```

Complex Expressions

Expressions can also be lengthy and more complex. The following expression returns *true* if the value of two fields (*$KX-Kohonen* and *$KY-Kohonen*) fall within the specified ranges. Notice that here the field names are single-quoted because the field names contain special characters.

```
('$KX-Kohonen' >= -0.2635771036148072 and '$KX-Kohonen' <=
0.3146203637123107
and '$KY-Kohonen' >= -0.18975617885589602 and
'$KY-Kohonen' <= 0.17674794197082522) -> T
```

Several functions, such as string functions, require you to enter several parameters using correct syntax. In the following example, the function `subscrs` is used to return the first character of a *produce_ID* field, indicating whether an item is organic, genetically modified, or conventional. The results of an expression are described by `-> `result``.

```
subscrs(1,produce_ID) -> `c`
```

Similarly, the following expression is:

```
stripchar(`3`,`123`) -> `12`
```

It is important to note that characters are always encapsulated within single backquotes.

Combining Functions in an Expression

Frequently, CLEM expressions consist of a combination of functions. The following function combines `subscr` and `lowertoupper` to return the first character of *produce_ID* and convert it to upper case.

```
lowertoupper(subscr(1,produce_ID)) -> `C`
```

This same expression can be written in shorthand as:

```
lowertoupper(produce_ID(1)) -> `C`
```

Another commonly used combination of functions is:

```
locchar_back(`n`, (length(web_page)), web_page)
```

This expression locates the character `n` within the values of the field *web_page* reading backward from the last character of the field value. By including the `length` function as well, the expression dynamically calculates the length of the current value rather than using a static number, such as 7, which will be invalid for values with less than seven characters.

Special Functions

Numerous special functions (preceded with an @ symbol) are available. Commonly used functions include:

```
@BLANK('referrer ID') -> T
```

Frequently, special functions are used in combination, which is a commonly used method of flagging blanks in more than one field at a time.

```
@BLANK(@FIELD)-> T
```

Additional examples are discussed throughout the CLEM documentation. See the topic for more information.

Values and Data Types

CLEM expressions are similar to formulas constructed from values, field names, operators, and functions. The simplest valid CLEM expression is a value or a field name. Examples of valid values are:

```
3
1.79
'banana'
```

Examples of field names are:

```
Product_ID
'$P-NextField'
```

where *Product* is the name of a field from a market basket data set, *'$P-NextField'* is the name of a parameter, and the value of the expression is the value of the named field. Typically, field names start with a letter and may also contain digits and underscores (_). You can use names that do not follow these rules if you place the name within quotation marks. CLEM values can be any of the following:

- Strings--for example, `"c1"`, `"Type 2"`, `"a piece of free text"`
- Integers--for example, `12`, `0`, `−189`
- Real numbers--for example, `12.34`, `0.0`, `−0.0045`
- Date/time fields--for example, `05/12/2002`, `12/05/2002`, `12/05/02`

It is also possible to use the following elements:

- Character codes--for example, `` `a` `` or `3`
- Lists of items--for example, `[1 2 3]`, `['Type 1' 'Type 2']`

Character codes and lists do not usually occur as field values. Typically, they are used as arguments of CLEM functions.

Quoting Rules

Although the software is flexible when determining the fields, values, parameters, and strings used in a CLEM expression, the following general rules provide a list of "best practices" to use when creating expressions:

- **Strings**—Always use double quotes when writing strings (`"Type 2"` or `"value"`). Single quotes can be used instead but at the risk of confusion with quoted fields.
- **Characters**—Always use single backquotes like this `` ` ``. For example, note the character d in the function `stripchar(`d`,"drugA")`. The only exception to this is when you are using an integer to refer to a specific character in a string. For example, note the character 5 in the function `lowertoupper("druga"(5)) —> "A"`. *Note*: On a standard U.K. and U.S. keyboard, the key for the backquote character (grave accent, Unicode 0060) can be found just below the Esc key.
- **Fields**—Fields are typically unquoted when used in CLEM expressions (`subscr(2,arrayID)) —> CHAR`). You can use single quotes when necessary to enclose spaces or other special characters (`'Order Number'`). Fields that are quoted but undefined in the data set will be misread as strings.
- **Parameters**—Always use single quotes (`'$P-threshold'`).

Expressions and Conditions

CLEM expressions can return a result (used when deriving new values)--for example:

```
Weight * 2.2
Age + 1
sqrt(Signal-Echo)
```

Or, they can evaluate *true* or *false* (used when selecting on a condition)--for example:

```
Drug = "drugA"
Age < 16
not(PowerFlux) and Power > 2000
```

You can combine operators and functions arbitrarily in CLEM expressions--for example:

```
sqrt(abs(Signal)) * max(T1, T2) + Baseline
```

Brackets and operator precedence determine the order in which the expression is evaluated. In this example, the order of evaluation is:

- `abs(Signal)` is evaluated, and `sqrt` is applied to its result.
- `max(T1, T2)` is evaluated.
- The two results are multiplied: **x** has higher precedence than +.
- Finally, `Baseline` is added to the result.

The descending order of precedence (that is, operations that are performed first to operations that are performed last) is as follows:

- Function arguments
- Function calls
- **xx**
- **x / mod div rem**
- + –
- > < >= <= /== == = /=

If you want to override precedence, or if you are in any doubt of the order of evaluation, you can use parentheses to make it explicit--for example,

```
sqrt(abs(Signal)) * (max(T1, T2) + Baseline)
```

Stream, Session, and SuperNode Parameters

Parameters can be defined for use in CLEM expressions and in scripting. They are, in effect, user-defined variables that are saved and persisted with the current stream, session, or SuperNode and can be accessed from the user interface as well as through scripting. If you save a stream, for example, any parameters set for that stream are also saved. (This distinguishes them from local script variables, which can be used only in the script in which they are declared.) Parameters are often used in scripting to control the behavior of the script, by providing information about fields and values that do not need to be hard coded in the script.

The scope of a parameter depends on where it is set:

- Stream parameters can be set in a stream script or in the stream properties dialog box, and they are available to all nodes in the stream. They are displayed on the Parameters list in the Expression Builder.
- Session parameters can be set in a stand-alone script or in the session parameters dialog box. They are available to all streams used in the current session (all streams listed on the Streams tab in the managers pane).

Parameters can also be set for SuperNodes, in which case they are visible only to nodes encapsulated within that SuperNode.

Using Parameters in CLEM Expressions

Parameters are represented in CLEM expressions by $P-pname, where pname is the name of the parameter. When used in CLEM expressions, parameters must be placed within single quotes––for example, '$P-scale'.

Available parameters are easily viewed using the Expression Builder. To view current parameters:

1. In any dialog box accepting CLEM expressions, click the Expression Builder button.
2. From the Fields list, select **Parameters**.

You can select parameters from the list for insertion into the CLEM expression. See the topic "Selecting fields, parameters, and global variables" on page 152 for more information.

Working with Strings

There are a number of operations available for strings, including:

- Converting a string to upper case or lower case—`uppertolower(CHAR)`.
- Removing specified characters, such as `` `ID_` `` or `` `$` ``, from a string variable—`stripchar(CHAR,STRING)`.
- Determining the length (number of characters) for a string variable—`length(STRING)`.
- Checking the alphabetical ordering of string values—`alphabefore(STRING1, STRING2)`.
- Removing leading or trailing white space from values—`trim(STRING)`, `trim_start(STRING)`, or `trimend(STRING)`.
- Extract the first or last *n* characters from a string—`startstring(LENGTH, STRING)` or `endstring(LENGTH, STRING)`. For example, suppose you have a field named *item* that combines a product name with a four-digit ID code (`ACME CAMERA-D109`). To create a new field that contains only the four-digit code, specify the following formula in a Derive node:

  ```
  endstring(4, item)
  ```

- Matching a specific pattern—`STRING matches PATTERN`. For example, to select persons with "market" anywhere in their job title, you could specify the following in a Select node:

  ```
  job_title matches "*market*"
  ```

- Replacing all instances of a substring within a string—`replace(SUBSTRING, NEWSUBSTRING, STRING)`. For example, to replace all instances of an unsupported character, such as a vertical pipe (|), with a semicolon prior to text mining, use the `replace` function in a Filler node. Under **Fill in fields:**, select all fields where the character may occur. For the **Replace:** condition, select **Always**, and specify the following condition under **Replace with:**

  ```
  replace('|',';',@FIELD)
  ```

- Deriving a flag field based on the presence of a specific substring. For example, you could use a string function in a Derive node to generate a separate flag field for each response with an expression such as:

  ```
  hassubstring(museums,"museum_of_design")
  ```

See the topic "String Functions" on page 173 for more information.

Handling Blanks and Missing Values

Replacing blanks or missing values is a common data preparation task for data miners. CLEM provides you with a number of tools to automate blank handling. The Filler node is the most common place to work with blanks; however, the following functions can be used in any node that accepts CLEM expressions:

- `@BLANK(FIELD)` can be used to determine records whose values are blank for a particular field, such as *Age*.
- `@NULL(FIELD)` can be used to determine records whose values are system-missing for the specified field(s). In IBM SPSS Modeler, system-missing values are displayed as **$null$** values.

See the topic "Functions Handling Blanks and Null Values" on page 191 for more information.

Working with Numbers

Numerous standard operations on numeric values are available in IBM SPSS Modeler, such as:

- Calculating the sine of the specified angle—`sin(NUM)`
- Calculating the natural log of numeric fields—`log(NUM)`
- Calculating the sum of two numbers—NUM1 + NUM2

See the topic "Numeric Functions" on page 168 for more information.

Working with Times and Dates

Time and date formats may vary depending on your data source and locale. The formats of date and time are specific to each stream and are set in the stream properties dialog box. The following examples are commonly used functions for working with date/time fields.

Calculating Time Passed

You can easily calculate the time passed from a baseline date using a family of functions similar to the following one. This function returns the time in months from the baseline date to the date represented by the date string DATE as a real number. This is an approximate figure, based on a month of 30.0 days.

```
date_in_months(Date)
```

Comparing Date/Time Values

Values of date/time fields can be compared across records using functions similar to the following one. This function returns a value of *true* if the date string DATE1 represents a date prior to that represented by the date string DATE2. Otherwise, this function returns a value of 0.

```
date_before(Date1, Date2)
```

Calculating Differences

You can also calculate the difference between two times and two dates using functions, such as:

```
date_weeks_difference(Date1, Date2)
```

This function returns the time in weeks from the date represented by the date string DATE1 to the date represented by the date string DATE2 as a real number. This is based on a week of 7.0 days. If DATE2 is prior to DATE1, this function returns a negative number.

Today's Date

The current date can be added to the data set using the function @TODAY. Today's date is added as a string to the specified field or new field using the date format selected in the stream properties dialog box. See the topic "Date and Time Functions" on page 179 for more information.

Summarizing Multiple Fields

The CLEM language includes a number of functions that return summary statistics across multiple fields. These functions may be particularly useful in analyzing survey data, where multiple responses to a question may be stored in multiple fields. See the topic "Working with Multiple-Response Data" on page 147 for more information.

Comparison Functions

You can compare values across multiple fields using the `min_n` and `max_n` functions—for example:

```
max_n(['card1fee' 'card2fee''card3fee''card4fee'])
```

You can also use a number of counting functions to obtain counts of values that meet specific criteria, even when those values are stored in multiple fields. For example, to count the number of cards that have been held for more than five years:

```
count_greater_than(5, ['cardtenure' 'card2tenure' 'card3tenure'])
```

To count null values across the same set of fields:

```
count_nulls(['cardtenure' 'card2tenure' 'card3tenure'])
```

Note that this example counts the number of cards being held, not the number of people holding them. See the topic "Comparison Functions" on page 166 for more information.

To count the number of times a specified value occurs across multiple fields, you can use the count_equal function. The following example counts the number of fields in the list that contain the value Y.

```
count_equal("Y",[Answer1, Answer2, Answer3])
```

Given the following values for the fields in the list, the function returns the results for the value Y as shown.

Table 8. Function values

Answer1	Answer2	Answer3	Count
Y	N	Y	2
Y	N	N	1

Numeric Functions

You can obtain statistics across multiple fields using the sum_n, mean_n, and sdev_n functions—for example:

```
sum_n(['card1bal' 'card2bal''card3bal'])
```

```
mean_n(['card1bal' 'card2bal''card3bal'])
```

See the topic "Numeric Functions" on page 168 for more information.

Generating Lists of Fields

When using any of the functions that accept a list of fields as input, the special functions @FIELDS_BETWEEN(start, end) and @FIELDS_MATCHING(pattern) can be used as input. For example, assuming the order of fields is as shown in the sum_n example earlier, the following would be equivalent:

```
sum_n(@FIELDS_BETWEEN(card1bal, card3bal))
```

Alternatively, to count the number of null values across all fields beginning with "*card*":

```
count_nulls(@FIELDS_MATCHING('card*'))
```

See the topic "Special Fields" on page 192 for more information.

Working with Multiple-Response Data

A number of comparison functions can be used to analyze multiple-response data, including:

- value_at

- `first_index / last_index`
- `first_non_null / last_non_null`
- `first_non_null_index / last_non_null_index`
- `min_index / max_index`

For example, suppose a multiple-response question asked for the first, second, and third most important reasons for deciding on a particular purchase (for example, price, personal recommendation, review, local supplier, other). In this case, you might determine the importance of price by deriving the index of the field in which it was first included:

```
first_index("price", [Reason1 Reason2 Reason3])
```

Similarly, suppose you have asked customers to rank three cars in order of likelihood to purchase and coded the responses in three separate fields, as follows:

Table 9. Car ranking example			
customer id	car1	car2	car3
101	1	3	2
102	3	2	1
103	2	3	1

In this case, you could determine the index of the field for the car they like most (ranked #1, or the lowest rank) using the `min_index` function:

```
min_index(['car1' 'car2' 'car3'])
```

See the topic "Comparison Functions" on page 166 for more information.

Referencing Multiple-Response Sets

The special `@MULTI_RESPONSE_SET` function can be used to reference all of the fields in a multiple-response set. For example, if the three *car* fields in the previous example are included in a multiple-response set named *car_rankings*, the following would return the same result:

```
max_index(@MULTI_RESPONSE_SET("car_rankings"))
```

The Expression Builder

You can type CLEM expressions manually or use the Expression Builder, which displays a complete list of CLEM functions and operators as well as data fields from the current stream, allowing you to quickly build expressions without memorizing the exact names of fields or functions. In addition, the Builder controls automatically add the proper quotes for fields and values, making it easier to create syntactically correct expressions.

Note: The Expression Builder is not supported in scripting or parameter settings.

Note: If you want to change your datasource, before changing the source you should check that the Expression Builder can still support the functions you have selected. Because not all databases support all functions, you may encounter an error if you run against a new datasource.

Accessing the Expression Builder

The Expression Builder is available in all nodes where CLEM expressions are used, including Select, Balance, Derive, Filler, Analysis, Report, and Table nodes. You can open it by clicking the calculator button just to the right of the formula field.

Creating Expressions

The Expression Builder provides not only complete lists of fields, functions, and operators but also access to data values if your data is instantiated.

To Create an Expression Using the Expression Builder

1. Type in the expression field, using the function and field lists as references.

   or

2. Select the required fields and functions from the scrolling lists.
3. Double-click or click the yellow arrow button to add the field or function to the expression field.
4. Use the operand buttons in the center of the dialog box to insert the operations into the expression.

Selecting functions

The function list displays all available CLEM functions and operators. Scroll to select a function from the list, or, for easier searching, use the drop-down list to display a subset of functions or operators. Available functions are grouped into categories for easier searching.

Most of these categories are described in the Reference section of the CLEM language description. For more information, see "Functions reference" on page 162.

The other categories are as follows.

- **General Functions** contains a selection of some of the most commonly-used functions.
- **Recently Used** contains a list of CLEM functions used within the current session.
- **@ Functions** contains a list of all the special functions, which have their names preceded by an "@" sign.

  Note: The @DIFF1(FIELD1,FIELD2) and @DIFF2(FIELD1,FIELD2) functions require that the two field types are the same (for example, both Integer or both Long or both Real).

- **Database Functions**. If the stream includes a database connection (by means of a Database source node), this selection lists the functions available from within that database, including user-defined functions (UDFs). For more information, see "Database functions" on page 150.
- **Database Aggregates**. If the stream includes a database connection (by means of a Database source node), this selection lists the aggregation options available from within that database. These options are available in the Expression Builder of the Aggregate node.
- **Database Window Aggregates**. If the stream includes a database connection (by means of a Database source node), this selection lists the window aggregation options that you can use within that database. These options are available in the Expression Builder within nodes in the **Field Operations** palette only.

  Note: Because SPSS Modeler obtains the **Window Aggregate Functions** from the Database System View, the available options are dependent on Database behavior.

  Although called "aggregates" these options are not designed for use in the Aggregate node; they are more applicable to nodes such as Derive or Select. This is because their output is scalar instead of a true aggregate; that is, they do not reduce the amount of data shown in the output in the same way that the Aggregate node does. For example, you could use this sort of aggregation to provide a moving average down through rows of data, such as "average of the current row and all previous rows".

- **Built-In Aggregates**. Contains a list of the possible modes of aggregation that can be used.
- **Operators** lists all the operators you can use when building expressions. Operators are also available from the buttons in the center of the dialog box.
- **All Functions** contains a complete list of available CLEM functions.

After you have selected a group of functions, double-click to insert the functions into the expression field at the point indicated by the position of the cursor.

Database functions

Database functions can be listed in many different locations; the following table shows the locations that SPSS Modeler searches when looking for function details. This table can be used by database administrators to ensure that users have access privileges to the required areas to be able to use the different functions.

In addition, the table lists the conditions that are used to filter when a function is available for use, based on the database and function type.

Note: If using database functions from Amazon Redshift, your database administrator may need to grant you permissions to the following six database objects. The first four are system catalog tables, and the last two are schemas.

- pg_type
- pg_proc
- pg_namespace
- pg_aggregate
- information_schema
- pg_catalog

Table 10. Database functions in the Expression Builder

Database	Function type	Where to find functions	Conditions used to filter functions
Db2 LUW	UDF	SYSCAT.ROUTINES SYSCAT.ROUTINEPARMS	ROUTINETYPE is F and FUNCTIONTYPE is S
Db2 LUW	UDA	SYSCAT.ROUTINES SYSCAT.ROUTINEPARMS	ROUTINETYPE is F and FUNCTIONTYPE is C
Db2 iSeries	UDF	QSYS2.SYSROUTINES QSYS2.SYSPARMS	ROUTINE_TYPE is F and FUNCTION_TYPE is S
Db2 iSeries	UDA	QSYS2.SYSROUTINES QSYS2.SYSPARMS	ROUTINE_TYPE is F and FUNCTION_TYPE is C
Db2 z/OS	UDF	SYSIBM.SYSROUTINES SYSIBM.SYSPARMS	ROUTINETYPE is F and FUNCTIONTYPE is S
Db2 z/OS	UDA	SYSIBM.SYSROUTINES SYSIBM.SYSPARMS	ROUTINETYPE is F and FUNCTIONTYPE is C
SQL Server	UDF	SYS.ALL_OBJECTS SYS.ALL_PARAMETERS SYS.TYPES	TYPE is either FN or FS
SQL Server	UDA	SYS.ALL_OBJECTS SYS.ALL_PARAMETERS SYS.TYPES	TYPE is AF
Oracle	UDF	ALL_ARGUMENTS ALL_PROCEDURES	All of the following conditions are satisfied: • OBJECT_TYPE is FUNCTION • AGGREGATE is NO • PLS_TYPE is not NULL

Table 10. Database functions in the Expression Builder (continued)			
Database	**Function type**	**Where to find functions**	**Conditions used to filter functions**
Oracle	UDA	`ALL_ARGUMENTS` `ALL_PROCEDURES`	All of the following conditions are satisfied: • `ARGUMENT_NAME` is NULL • `AGGREGATE` is YES • `PLS_TYPE` is not NULL
Teradata	UDF	`DBC.FUNCTIONS` `DBC.ALLRIGHTS`	All of the following conditions are satisfied: • `FUNCTIONTYPE` is F • `COLUMNNAME` is RETURN0 • `SPPARAMETERTYPE` is O • `ACCESSRIGHT` is EF
Teradata	UDA	`DBC.FUNCTIONS` `DBC.ALLRIGHTS`	All of the following conditions are satisfied: • `FUNCTIONTYPE` is A • `COLUMNNAME` is RETURN0 • `SPPARAMETERTYPE` is O • `ACCESSRIGHT` is EF

Netezza	UDF	`####.._V_FUNCTION`
		`NZA.._V_FUNCTION`
		`INZA.._V_FUNCTION`
			For `####.._V_FUNCTION`, the following conditions apply:
			• `RESULT` does not contain a string with values such as: TABLE%
			• `FUNCTION` does not contain a string with values such as: `'/_%' escape '/'`
			• `VARARGS` is FALSE
			
			For both `NZA.._V_FUNCTION` and `INZA.._V_FUNCTION`, the following conditions apply:
			• `RESULT` does not contain a string with values such as: TABLE%
			• `FUNCTION` does not contain a string with values such as: `'/_%' escape '/'`
			• `BUILTIN` is f
			• `VARARGS` is FALSE

Netezza	UDA	`####.._V_AGGREGATE`
		`NZA.._V_FUNCTION`
		`INZA.._V_FUNCTION`
			Both of the following conditions are satisfied:
			• `AGGTYPE` is ANY or GROUPED
			• `VARARGS` is FALSE

Table 10. Database functions in the Expression Builder (continued)			
Database	**Function type**	**Where to find functions**	**Conditions used to filter functions**
Netezza	WUDA	*####*.._V_AGGREGA TE NZA.._V_FUNCTION INZA.._V_FUNCTIO N	For *####*.._V_AGGREGATE, the following conditions apply: • AGGTYPE is ANY or ANALYTIC • AGGREGATE is not MAX_LABEL • VARARGS is FALSE For both NZA.._V_FUNCTION and INZA.._V_FUNCTION, the following conditions apply: • AGGTYPE is ANY or ANALYTIC • BUILTIN is f • VARARGS is FALSE

Key to terms used in the table

• UDF User Defined Function
• UDA User Defined Aggregate Function
• WUDA User Defined Window Aggregate Function
• #### the database that you are currently connected to.

Selecting fields, parameters, and global variables

The field list displays all fields available at this point in the data stream. Scroll to select a field from the list. Double-click or click the yellow arrow button to add a field to the expression.

See the topic "Stream, Session, and SuperNode Parameters" on page 144 for more information.

In addition to fields, you can also choose from the following items:

Multiple-response sets. For more information, see the *IBM SPSS Modeler Source, Process, and Output Nodes* guide.

Recently used contains a list of fields, multiple-response sets, parameters, and global values used within the current session.

Parameters. See the topic "Stream, Session, and SuperNode Parameters" on page 144 for more information.

Global values. For more information, see the *IBM SPSS Modeler Source, Process, and Output Nodes* guide.

Viewing or selecting values

Field values can be viewed from a number of places in the system, including the Expression Builder, data audit reports, and when editing future values in a Time Intervals node. Note that data must be fully instantiated in a source or Type node to use this feature, so that storage, types, and values are known.

To view values for a field from the Expression Builder or a Time Intervals node, select the required field and click the value picker button to open a dialog box listing values for the selected field. You can then select a value and click **Insert** to paste the value into the current expression or list.

Figure 16. Value picker button

For flag and nominal fields, all defined values are listed. For continuous (numeric range) fields, the minimum and maximum values are displayed.

Checking CLEM expressions

Click **Check** in the Expression Builder (lower right corner) to validate the expression. Expressions that have not been checked are displayed in red. If errors are found, a message indicating the cause is displayed.

The following items are checked:

- Correct quoting of values and field names
- Correct usage of parameters and global variables
- Valid usage of operators
- Existence of referenced fields
- Existence and definition of referenced globals

If you encounter errors in syntax, try creating the expression using the lists and operator buttons rather than typing the expression manually. This method automatically adds the proper quotes for fields and values.

Note the following limitations when building expressions in IBM Analytical Decision Management. Expressions cannot contain any of the following items:

- A reference to an IBM SPSS Modeler stream parameter
- A reference to an IBM SPSS Modeler stream global
- A reference to a database function
- A reference to one of the following special field or field value @ functions:
  - @TARGET
  - @PREDICTED
  - @FIELD
  - @PARTITION_FIELD
  - @TRAINING_PARTITION
  - @TESTING_PARTITION
  - @VALIDATION_PARTITION

Note: Field names that contain separators must be surrounded by single quotes. To automatically add quotes, you can create expressions using the lists and operator buttons rather than typing expressions manually. The following characters in field names may cause errors: • ! "# $% & '() = ~ |-^ ¥ @" "+ *" "<>? . ,/ :; → (arrow mark), □ △ (graphic mark, etc.)

Find and Replace

The Find/Replace dialog box is available in places where you edit script or expression text, including the script editor, CLEM expression builder, or when defining a template in the Report node. When editing text in any of these areas, press `Ctrl+F` to access the dialog box, making sure cursor has focus in a text area. If working in a Filler node, for example, you can access the dialog box from any of the text areas on the Settings tab, or from the text field in the Expression Builder.

1. With the cursor in a text area, press Ctrl+F to access the Find/Replace dialog box.
2. Enter the text you want to search for, or choose from the drop-down list of recently searched items.

3. Enter the replacement text, if any.
4. Click **Find Next** to start the search.
5. Click **Replace** to replace the current selection, or **Replace All** to update all or selected instances.
6. The dialog box closes after each operation. Press F3 from any text area to repeat the last find operation, or press Ctrl+F to access the dialog box again.

Search Options

Match case. Specifies whether the find operation is case-sensitive; for example, whether *myvar* matches *myVar*. Replacement text is always inserted exactly as entered, regardless of this setting.

Whole words only. Specifies whether the find operation matches text embedded within words. If selected, for example, a search on *spider* will not match *spiderman* or *spider-man*.

Regular expressions. Specifies whether regular expression syntax is used (see next section). When selected, the **Whole words only** option is disabled and its value is ignored.

Selected text only. Controls the scope of the search when using the **Replace All** option.

Regular Expression Syntax

Regular expressions allow you to search on special characters such as tabs or newline characters, classes or ranges of characters such as *a* through *d*, any digit or non-digit, and boundaries such as the beginning or end of a line. The following types of expressions are supported.

Table 11. Character matches

Characters	Matches
x	The character x
\\	The backslash character
\0n	The character with octal value 0n (0 <= n <= 7)
\0nn	The character with octal value 0nn (0 <= n <= 7)
\0mnn	The character with octal value 0mnn (0 <= m <= 3, 0 <= n <= 7)
\xhh	The character with hexadecimal value 0xhh
\uhhhh	The character with hexadecimal value 0xhhhh
\t	The tab character ('\u0009')
\n	The newline (line feed) character ('\u000A')
\r	The carriage-return character ('\u000D')
\f	The form-feed character ('\u000C')
\a	The alert (bell) character ('\u0007')
\e	The escape character ('\u001B')
\cx	The control character corresponding to x

Table 12. Matching character classes

Character classes	Matches
[abc]	a, b, or c (simple class)
[^abc]	Any character except a, b, or c (subtraction)
[a-zA-Z]	a through z or A through Z, inclusive (range)

Table 12. Matching character classes (continued)	
Character classes	**Matches**
[a-d[m-p]]	a through d, or m through p (union). Alternatively this could be specified as [a-dm-p]
[a-z&&[def]]	a through z, and d, e, or f (intersection)
[a-z&&[^bc]]	a through z, except for b and c (subtraction). Alternatively this could be specified as [ad-z]
[a-z&&[^m-p]]	a through z, and not m through p (subtraction). Alternatively this could be specified as [a-lq-z]

Table 13. Predefined character classes	
Predefined character classes	**Matches**
.	Any character (may or may not match line terminators)
\d	Any digit: [0-9]
\D	A non-digit: [^0-9]
\s	A white space character: [\t\n\x0B\f\r]
\S	A non-white space character: [^\s]
\w	A word character: [a-zA-Z_0-9]
\W	A non-word character: [^\w]

Table 14. Boundary matches	
Boundary matchers	**Matches**
^	The beginning of a line
$	The end of a line
\b	A word boundary
\B	A non-word boundary
\A	The beginning of the input
\Z	The end of the input but for the final terminator, if any
\z	The end of the input

Chapter 11. CLEM language reference

CLEM Reference Overview

This section describes the Control Language for Expression Manipulation (CLEM), which is a powerful tool used to analyze and manipulate the data used in IBM SPSS Modeler streams. You can use CLEM within nodes to perform tasks ranging from evaluating conditions or deriving values to inserting data into reports.

CLEM expressions consist of values, field names, operators, and functions. Using the correct syntax, you can create a wide variety of powerful data operations.

CLEM Datatypes

CLEM datatypes can be made up of any of the following:

- Integers
- Reals
- Characters
- Strings
- Lists
- Fields
- Date/Time

Rules for Quoting

Although IBM SPSS Modeler is flexible when you are determining the fields, values, parameters, and strings used in a CLEM expression, the following general rules provide a list of "good practices" to use in creating expressions:

- Strings—Always use double quotes when writing strings, such as `"Type 2"`. Single quotes can be used instead but at the risk of confusion with quoted fields.
- Fields—Use single quotes only where necessary to enclose spaces or other special characters, such as `'Order Number'`. Fields that are quoted but undefined in the data set will be misread as strings.
- Parameters—Always use single quotes when using parameters, such as `'$P-threshold'`.
- Characters—Always use single backquotes (`` ` ``), such as `stripchar(`d`, "drugA")`.

These rules are covered in more detail in the following topics.

Integers

Integers are represented as a sequence of decimal digits. Optionally, you can place a minus sign (–) before the integer to denote a negative number—for example, 1234, 999, –77.

The CLEM language handles integers of arbitrary precision. The maximum integer size depends on your platform. If the values are too large to be displayed in an integer field, changing the field type to `Real` usually restores the value.

Reals

Real refers to a floating-point number. Reals are represented by one or more digits followed by a decimal point followed by one or more digits. CLEM reals are held in double precision.

Optionally, you can place a minus sign (–) before the real to denote a negative number——for example, `1.234`, `0.999`, `–77.001`. Use the form *<number>* e *<exponent>* to express a real number in exponential

notation—for example, 1234.0e5, 1.7e-2. When the IBM SPSS Modeler application reads number strings from files and converts them automatically to numbers, numbers with no leading digit before the decimal point or with no digit after the point are accepted——for example, 999. or .11. However, these forms are illegal in CLEM expressions.

Note: When referencing real numbers in CLEM expressions, a period must be used as the decimal separator, regardless of any settings for the current stream or locale. For example, specify

```
Na > 0.6
```

rather than

```
Na > 0,6
```

This applies even if a comma is selected as the decimal symbol in the stream properties dialog box and is consistent with the general guideline that code syntax should be independent of any specific locale or convention.

Characters

Characters (usually shown as CHAR) are typically used within a CLEM expression to perform tests on strings. For example, you can use the function `isuppercode` to determine whether the first character of a string is upper case. The following CLEM expression uses a character to indicate that the test should be performed on the first character of the string:

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

To express the code (in contrast to the location) of a particular character in a CLEM expression, use single backquotes of the form `` `<character>` ``——for example, `` `A` ``, `` `Z` ``.

Note: There is no CHAR storage type for a field, so if a field is derived or filled with an expression that results in a CHAR, then that result will be converted to a string.

Strings

Generally, you should enclose strings in double quotation marks. Examples of strings are `"c35product2"` and `"referrerID"`. To indicate special characters in a string, use a backslash——for example, `"\$65443"`. (To indicate a backslash character, use a double backslash, `\\`.) You can use single quotes around a string, but the result is indistinguishable from a quoted field (`'referrerID'`). See the topic "String Functions" on page 173 for more information.

Lists

A list is an ordered sequence of elements, which may be of mixed type. Lists are enclosed in square brackets ([]). Examples of lists are `[1 2 4 16]` and `["abc" "def"]`. Lists are not used as the value of IBM SPSS Modeler fields. They are used to provide arguments to functions, such as `member` and `oneof`.

Note: Lists can be composed only of static objects (for example, a string, number, or field name) and not calls to functions.

Fields

Names in CLEM expressions that are not names of functions are assumed to be field names. You can write these simply as `Power`, `val27`, `state_flag`, and so on, but if the name begins with a digit or includes non-alphabetic characters, such as spaces (with the exception of the underscore), place the name within single quotation marks——for example, `'Power Increase'`, `'2nd answer'`, `'#101'`, `'$P-NextField'`.

Note: Fields that are quoted but undefined in the data set will be misread as strings.

Dates

Date calculations are based on a "baseline" date, which is specified in the stream properties dialog box. The default baseline date is 1 January 1900.

The CLEM language supports the following date formats.

Format	Examples
Table 15. CLEM language date formats	
Format	**Examples**
DDMMYY	150163
MMDDYY	011563
YYMMDD	630115
YYYYMMDD	19630115
YYYYDDD	Four-digit year followed by a three-digit number representing the day of the year—for example, 2000032 represents the 32nd day of 2000, or 1 February 2000.
DAY	Day of the week in the current locale—for example, Monday, Tuesday, ..., in English.
MONTH	Month in the current locale—for example, January, February, ...
DD/MM/YY	15/01/63
DD/MM/YYYY	15/01/1963
MM/DD/YY	01/15/63
MM/DD/YYYY	01/15/1963
DD-MM-YY	15-01-63
DD-MM-YYYY	15-01-1963
MM-DD-YY	01-15-63
MM-DD-YYYY	01-15-1963
DD.MM.YY	15.01.63
DD.MM.YYYY	15.01.1963
MM.DD.YY	01.15.63
MM.DD.YYYY	01.15.1963
DD-MON-YY	15-JAN-63, 15-jan-63, 15-Jan-63
DD/MON/YY	15/JAN/63, 15/jan/63, 15/Jan/63
DD.MON.YY	15.JAN.63, 15.jan.63, 15.Jan.63
DD-MON-YYYY	15-JAN-1963, 15-jan-1963, 15-Jan-1963
DD/MON/YYYY	15/JAN/1963, 15/jan/1963, 15/Jan/1963
DD.MON.YYYY	15.JAN.1963, 15.jan.1963, 15.Jan.1963
MON YYYY	Jan 2004
q Q YYYY	Date represented as a digit (1–4) representing the quarter followed by the letter *Q* and a four-digit year—for example, 25 December 2004 would be represented as 4 Q 2004.

Table 15. CLEM language date formats (continued)	
Format	**Examples**
`ww WK YYYY`	Two-digit number representing the week of the year followed by the letters *WK* and then a four-digit year. The week of the year is calculated assuming that the first day of the week is Monday and there is at least one day in the first week.

Time

The CLEM language supports the following time formats.

Table 16. CLEM language time formats	
Format	**Examples**
`HHMMSS`	`120112, 010101, 221212`
`HHMM`	`1223, 0745, 2207`
`MMSS`	`5558, 0100`
`HH:MM:SS`	`12:01:12, 01:01:01, 22:12:12`
`HH:MM`	`12:23, 07:45, 22:07`
`MM:SS`	`55:58, 01:00`
`(H)H:(M)M:(S)S`	`12:1:12, 1:1:1, 22:12:12`
`(H)H:(M)M`	`12:23, 7:45, 22:7`
`(M)M:(S)S`	`55:58, 1:0`
`HH.MM.SS`	`12.01.12, 01.01.01, 22.12.12`
`HH.MM`	`12.23, 07.45, 22.07`
`MM.SS`	`55.58, 01.00`
`(H)H.(M)M.(S)S`	`12.1.12, 1.1.1, 22.12.12`
`(H)H.(M)M`	`12.23, 7.45, 22.7`
`(M)M.(S)S`	`55.58, 1.0`

CLEM Operators

The following operators are available.

Table 17. CLEM language operators		
Operation	**Comments**	**Precedence (see next section)**
`or`	Used between two CLEM expressions. Returns a value of true if either is true or if both are true.	10
`and`	Used between two CLEM expressions. Returns a value of true if both are true.	9
`=`	Used between any two comparable items. Returns true if ITEM1 is equal to ITEM2.	7

Operation	Comments	Precedence (see next section)
==	Identical to =.	7
/=	Used between any two comparable items. Returns true if ITEM1 is *not* equal to ITEM2.	7
/==	Identical to /=.	7
>	Used between any two comparable items. Returns true if ITEM1 is strictly greater than ITEM2.	6
>=	Used between any two comparable items. Returns true if ITEM1 is greater than or equal to ITEM2.	6
<	Used between any two comparable items. Returns true if ITEM1 is strictly less than ITEM2	6
<=	Used between any two comparable items. Returns true if ITEM1 is less than or equal to ITEM2.	6
&&=_0	Used between two integers. Equivalent to the Boolean expression INT1 && INT2 = 0.	6
&&/=_0	Used between two integers. Equivalent to the Boolean expression INT1 && INT2 /= 0.	6
+	Adds two numbers: NUM1 + NUM2.	5
><	Concatenates two strings; for example, STRING1 >< STRING2.	5
-	Subtracts one number from another: NUM1 - NUM2. Can also be used in front of a number: - NUM.	5
*	Used to multiply two numbers: NUM1 * NUM2.	4
&&	Used between two integers. The result is the bitwise 'and' of the integers INT1 and INT2.	4
&&~~	Used between two integers. The result is the bitwise 'and' of INT1 and the bitwise complement of INT2.	4
||	Used between two integers. The result is the bitwise 'inclusive or' of INT1 and INT2.	4
~~	Used in front of an integer. Produces the bitwise complement of INT.	4
||/&	Used between two integers. The result is the bitwise 'exclusive or' of INT1 and INT2.	4

Table 17. CLEM language operators (continued)

Table 17. CLEM language operators (continued)		
Operation	**Comments**	**Precedence (see next section)**
`INT1 << N`	Used between two integers. Produces the bit pattern of INT shifted left by N positions.	4
`INT1 >> N`	Used between two integers. Produces the bit pattern of INT shifted right by N positions.	4
`/`	Used to divide one number by another: NUM1 / NUM2.	4
`**`	Used between two numbers: BASE ** POWER. Returns BASE raised to the power POWER.	3
`rem`	Used between two integers: INT1 rem INT2. Returns the remainder, INT1 - (INT1 div INT2) * INT2.	2
`div`	Used between two integers: INT1 div INT2. Performs integer division.	2

Operator Precedence

Precedences determine the parsing of complex expressions, especially unbracketed expressions with more than one infix operator. For example,

```
3 + 4 * 5
```

parses as $3 + (4 * 5)$ rather than $(3 + 4) * 5$ because the relative precedences dictate that * is to be parsed before +. Every operator in the CLEM language has a precedence value associated with it; the lower this value, the more important it is on the parsing list, meaning that it will be processed sooner than other operators with higher precedence values.

Functions reference

The following CLEM functions are available for working with data in IBM SPSS Modeler. You can enter these functions as code in various dialog boxes, such as Derive and Set To Flag nodes, or you can use the Expression Builder to create valid CLEM expressions without memorizing function lists or field names.

Table 18. CLEM functions for use with IBM SPSS Modeler data	
Function Type	**Description**
Information	Used to gain insight into field values. For example, the function `is_string` returns true for all records whose type is a string.
Conversion	Used to construct new fields or convert storage type. For example, the function `to_timestamp` converts the selected field to a timestamp.
Comparison	Used to compare field values to each other or to a specified string. For example, `<=` is used to compare whether the values of two fields are lesser or equal.
Logical	Used to perform logical operations, such as `if`, `then`, `else` operations.

Table 18. CLEM functions for use with IBM SPSS Modeler data (continued)	
Function Type	**Description**
Numeric	Used to perform numeric calculations, such as the natural log of field values.
Trigonometric	Used to perform trigonometric calculations, such as the arccosine of a specified angle.
Probability	Returns probabilities that are based on various distributions, such as probability that a value from Student's *t* distribution is less than a specific value.
Spatial	Used to perform spatial calculations on geospatial data.
Bitwise	Used to manipulate integers as bit patterns.
Random	Used to randomly select items or generate numbers.
String	Used to perform various operations on strings, such as `stripchar`, which allows you to remove a specified character.
SoundEx	Used to find strings when the precise spelling is not known; based on phonetic assumptions about how certain letters are pronounced.
Date and time	Used to perform various operations on date, time, and timestamp fields.
Sequence	Used to gain insight into the record sequence of a data set or perform operations that are based on that sequence.
Global	Used to access global values that are created by a Set Globals node. For example, @MEAN is used to refer to the mean average of all values for a field across the entire data set.
Blanks and null	Used to access, flag, and frequently fill user-specified blanks or system-missing values. For example, @BLANK(FIELD) is used to raise a true flag for records where blanks are present.
Special fields	Used to denote the specific fields under examination. For example, @FIELD is used when deriving multiple fields.

Conventions in Function Descriptions

The following conventions are used throughout this guide when referring to items in a function.

Table 19. Conventions in function descriptions	
Convention	**Description**
BOOL	A Boolean, or flag, such as true or false.
NUM, NUM1, NUM2	Any number.
REAL, REAL1, REAL2	Any real number, such as `1.234` or `−77.01`.
INT, INT1, INT2	Any integer, such as 1 or −77.
CHAR	A character code, such as `` `A` ``.
STRING	A string, such as `"referrerID"`.
LIST	A list of items, such as `["abc" "def"]`.
ITEM	A field, such as `Customer` or `extract_concept`.

Table 19. Conventions in function descriptions (continued)	
Convention	**Description**
DATE	A date field, such as `start_date`, where values are in a format such as `DD-MON-YYYY`.
TIME	A time field, such as `power_flux`, where values are in a format such as `HHMMSS`.

Functions in this guide are listed with the function in one column, the result type (integer, string, and so on) in another, and a description (where available) in a third column. For example, the following is the description of the `rem` function.

Table 20. rem function description		
Function	**Result**	**Description**
`INT1 rem INT2`	*Number*	Returns the remainder of *INT1* divided by *INT2*. For example, `INT1 – (INT1 div INT2) * INT2`.

Details on usage conventions, such as how to list items or specify characters in a function, are described elsewhere. See the topic for more information.

Information Functions

Information functions are used to gain insight into the values of a particular field. They are typically used to derive flag fields. For example, you can use the @BLANK function to create a flag field indicating records whose values are blank for the selected field. Similarly, you can check the storage type for a field using any of the storage type functions, such as `is_string`.

Table 21. CLEM information functions		
Function	**Result**	**Description**
`@BLANK(FIELD)`	*Boolean*	Returns true for all records whose values are blank according to the blank-handling rules set in an upstream Type node or source node (Types tab).
`@NULL(ITEM)`	*Boolean*	Returns true for all records whose values are undefined. Undefined values are system null values, displayed in IBM SPSS Modeler as $null$.
`is_date(ITEM)`	*Boolean*	Returns true for all records whose type is a date.
`is_datetime(ITEM)`	*Boolean*	Returns true for all records whose type is a date, time, or timestamp.
`is_integer(ITEM)`	*Boolean*	Returns true for all records whose type is an integer.
`is_number(ITEM)`	*Boolean*	Returns true for all records whose type is a number.
`is_real(ITEM)`	*Boolean*	Returns true for all records whose type is a real.
`is_string(ITEM)`	*Boolean*	Returns true for all records whose type is a string.
`is_time(ITEM)`	*Boolean*	Returns true for all records whose type is a time.
`is_timestamp(ITEM)`	*Boolean*	Returns true for all records whose type is a timestamp.

Conversion Functions

Conversion functions allow you to construct new fields and convert the storage type of existing files. For example, you can form new strings by joining strings together or by taking strings apart. To join two

strings, use the operator ><. For example, if the field `Site` has the value `"BRAMLEY"`, then `"xx" ><`
`Site` returns `"xxBRAMLEY"`. The result of `><` is always a string, even if the arguments are not strings.
Thus, if field V1 is 3 and field V2 is 5, then V1 `><` V2 returns `"35"` (a string, not a number).

Conversion functions (and any other functions that require a specific type of input, such as a date or time
value) depend on the current formats specified in the Stream Options dialog box. For example, if you want
to convert a string field with values *Jan 2003*, *Feb 2003*, and so on, select the matching date format **MON
YYYY** as the default date format for the stream.

Table 22. CLEM conversion functions		
Function	**Result**	**Description**
`ITEM1 >< ITEM2`	*String*	Concatenates values for two fields and returns the resulting string as *ITEM1ITEM2*.
`to_integer(ITEM)`	*Integer*	Converts the storage of the specified field to an integer.
`to_real(ITEM)`	*Real*	Converts the storage of the specified field to a real.
`to_number(ITEM)`	*Number*	Converts the storage of the specified field to a number.
`to_string(ITEM)`	*String*	Converts the storage of the specified field to a string. When a real is converted to string using this function, it returns a value with 6 digits after the radix point.
`to_time(ITEM)`	*Time*	Converts the storage of the specified field to a time.
`to_date(ITEM)`	*Date*	Converts the storage of the specified field to a date.
`to_timestamp(ITEM)`	*Timestamp*	Converts the storage of the specified field to a timestamp.
`to_datetime(ITEM)`	*Datetime*	Converts the storage of the specified field to a date, time, or timestamp value.
`datetime_date(ITEM)`	*Date*	Returns the date value for a *number*, *string*, or *timestamp*. Note this is the only function that allows you to convert a number (in seconds) back to a date. If ITEM is a string, creates a date by parsing a string in the current date format. The date format specified in the stream properties dialog box must be correct for this function to be successful. If ITEM is a number, it is interpreted as a number of seconds since the base date (or epoch). Fractions of a day are truncated. If ITEM is a timestamp, the date part of the timestamp is returned. If ITEM is a date, it is returned unchanged.
`stb_centroid_latitude(ITEM)`	*Integer*	Returns an integer value for latitude corresponding to centroid of the geohash argument.
`stb_centroid_longitude(ITEM)`	*Integer*	Returns an integer value for longitude corresponding to centroid of the geohash argument.

Table 22. CLEM conversion functions (continued)		
Function	**Result**	**Description**
`to_geohash(ITEM)`	*String*	Returns the geohashed string corresponding to the latitude and longitude using the specified number of bits for the density. A geohash is a code used to identify a set of geographic coordinates based on the latitude and longitude details. The three parameters for `to_geohash` are: • *latitude*: Range (-180, 180), and units are degrees in the WGS84 coordinate system • *longitude*: Range (-90, 90), and units are degrees in the WGS84 coordinate system • *bits*: The number of bits to use to store the hash. Range [1,75]. This affects both the length of the returned string (1 character is used for every 5 bits), and the accuracy of the hash. For example, 5 bits (1 character) represents approximately 2500 kilometers, or 45 bits (9 characters), represents approximately 2.3 meters.

Comparison Functions

Comparison functions are used to compare field values to each other or to a specified string. For example, you can check strings for equality using =. An example of string equality verification is: `Class = "class 1"`.

For purposes of numeric comparison, *greater* means closer to positive infinity, and *lesser* means closer to negative infinity. That is, all negative numbers are less than any positive number.

Table 23. CLEM comparison functions		
Function	**Result**	**Description**
`count_equal(ITEM1, LIST)`	*Integer*	Returns the number of values from a list of fields that are equal to *ITEM1* or null if *ITEM1* is null.
`count_greater_than( ITEM1, LIST)`	*Integer*	Returns the number of values from a list of fields that are greater than *ITEM1* or null if *ITEM1* is null.
`count_less_than(ITEM1, LIST)`	*Integer*	Returns the number of values from a list of fields that are less than *ITEM1* or null if *ITEM1* is null.
`count_not_equal(ITEM1, LIST)`	*Integer*	Returns the number of values from a list of fields that are not equal to *ITEM1* or null if *ITEM1* is null.
`count_nulls(LIST)`	*Integer*	Returns the number of null values from a list of fields.
`count_non_nulls(LIST)`	*Integer*	Returns the number of non-null values from a list of fields.
`date_before(DATE1, DATE2)`	*Boolean*	Used to check the ordering of date values. Returns a true value if *DATE1* is before *DATE2*.
`first_index(ITEM, LIST)`	*Integer*	Returns the index of the first field containing ITEM from a LIST of fields or 0 if the value is not found. Supported for string, integer, and real types only.
`first_non_null(LIST)`	*Any*	Returns the first non-null value in the supplied list of fields. All storage types supported.

Function	Result	Description
`first_non_null_index(LIST)`	*Integer*	Returns the index of the first field in the specified LIST containing a non-null value or 0 if all values are null. All storage types are supported.
`ITEM1 = ITEM2`	*Boolean*	Returns true for records where *ITEM1* is equal to *ITEM2*.
`ITEM1 /= ITEM2`	*Boolean*	Returns true if the two strings are not identical or 0 if they are identical.
`ITEM1 < ITEM2`	*Boolean*	Returns true for records where *ITEM1* is less than *ITEM2*.
`ITEM1 <= ITEM2`	*Boolean*	Returns true for records where *ITEM1* is less than or equal to *ITEM2*.
`ITEM1 > ITEM2`	*Boolean*	Returns true for records where *ITEM1* is greater than *ITEM2*.
`ITEM1 >= ITEM2`	*Boolean*	Returns true for records where *ITEM1* is greater than or equal to *ITEM2*.
`last_index(ITEM, LIST)`	*Integer*	Returns the index of the last field containing ITEM from a LIST of fields or 0 if the value is not found. Supported for string, integer, and real types only.
`last_non_null(LIST)`	*Any*	Returns the last non-null value in the supplied list of fields. All storage types supported.
`last_non_null_index(LIST)`	*Integer*	Returns the index of the last field in the specified LIST containing a non-null value or 0 if all values are null. All storage types are supported.
`max(ITEM1, ITEM2)`	*Any*	Returns the greater of the two items--*ITEM1* or *ITEM2*.
`max_index(LIST)`	*Integer*	Returns the index of the field containing the maximum value from a list of numeric fields or 0 if all values are null. For example, if the third field listed contains the maximum, the index value 3 is returned. If multiple fields contain the maximum value, the one listed first (leftmost) is returned.
`max_n(LIST)`	*Number*	Returns the maximum value from a list of numeric fields or null if all of the field values are null.
`member(ITEM, LIST)`	*Boolean*	Returns true if *ITEM* is a member of the specified *LIST*. Otherwise, a false value is returned. A list of field names can also be specified.
`min(ITEM1, ITEM2)`	*Any*	Returns the lesser of the two items--*ITEM1* or *ITEM2*.
`min_index(LIST)`	*Integer*	Returns the index of the field containing the minimum value from a list of numeric fields or 0 if all values are null. For example, if the third field listed contains the minimum, the index value 3 is returned. If multiple fields contain the minimum value, the one listed first (leftmost) is returned.
`min_n(LIST)`	*Number*	Returns the minimum value from a list of numeric fields or null if all of the field values are null.
`time_before(TIME1, TIME2)`	*Boolean*	Used to check the ordering of time values. Returns a true value if *TIME1* is before *TIME2*.

Table 23. CLEM comparison functions (continued)

Table 23. CLEM comparison functions (continued)		
Function	**Result**	**Description**
`value_at(INT, LIST)`		Returns the value of each listed field at offset INT or NULL if the offset is outside the range of valid values (that is, less than 1 or greater than the number of listed fields). All storage types supported.

Logical Functions

CLEM expressions can be used to perform logical operations.

Table 24. CLEM logical functions		
Function	**Result**	**Description**
`COND1 and COND2`	*Boolean*	This operation is a logical conjunction and returns a true value if both *COND1* and *COND2* are true. If *COND1* is false, then *COND2* is not evaluated; this makes it possible to have conjunctions where *COND1* first tests that an operation in *COND2* is legal. For example, `length(Label) >=6` and `Label(6) = 'x'`.
`COND1 or COND2`	*Boolean*	This operation is a logical (inclusive) disjunction and returns a true value if either *COND1* or *COND2* is true or if both are true. If *COND1* is true, *COND2* is not evaluated.
`not(COND)`	*Boolean*	This operation is a logical negation and returns a true value if *COND* is false. Otherwise, this operation returns a value of 0.
`if COND then EXPR1 else EXPR2 endif`	*Any*	This operation is a conditional evaluation. If *COND* is true, this operation returns the result of *EXPR1*. Otherwise, the result of evaluating *EXPR2* is returned.
`if COND1 then EXPR1 elseif COND2 then EXPR2 else EXPR_N endif`	*Any*	This operation is a multibranch conditional evaluation. If *COND1* is true, this operation returns the result of *EXPR1*. Otherwise, if *COND2* is true, this operation returns the result of evaluating *EXPR2*. Otherwise, the result of evaluating *EXPR_N* is returned.

Numeric Functions

CLEM contains a number of commonly used numeric functions.

Table 25. CLEM numeric functions		
Function	**Result**	**Description**
`–NUM`	*Number*	Used to negate *NUM*. Returns the corresponding number with the opposite sign.
`NUM1 + NUM2`	*Number*	Returns the sum of *NUM1* and *NUM2*.
`NUM1 –NUM2`	*Number*	Returns the value of *NUM2* subtracted from *NUM1*.
`NUM1 * NUM2`	*Number*	Returns the value of *NUM1* multiplied by *NUM2*.
`NUM1 / NUM2`	*Number*	Returns the value of *NUM1* divided by *NUM2*.
`INT1 div INT2`	*Number*	Used to perform integer division. Returns the value of *INT1* divided by *INT2*.

Function	Result	Description
`INT1 rem INT2`	*Number*	Returns the remainder of *INT1* divided by *INT2*. For example, `INT1 – (INT1 div INT2) * INT2`.
`INT1 mod INT2`	*Number*	This function has been deprecated. Use the `rem` function instead.
`BASE ** POWER`	*Number*	Returns *BASE* raised to the power *POWER*, where either may be any number (except that *BASE* must not be zero if *POWER* is zero of any type other than integer 0). If *POWER* is an integer, the computation is performed by successively multiplying powers of *BASE*. Thus, if *BASE* is an integer, the result will be an integer. If *POWER* is integer 0, the result is always a 1 of the same type as *BASE*. Otherwise, if *POWER* is not an integer, the result is computed as `exp(POWER * log(BASE))`.
`abs(NUM)`	*Number*	Returns the absolute value of *NUM*, which is always a number of the same type.
`exp(NUM)`	*Real*	Returns *e* raised to the power *NUM*, where *e* is the base of natural logarithms.
`fracof(NUM)`	*Real*	Returns the fractional part of *NUM*, defined as NUM–`intof(NUM)`.
`intof(NUM)`	*Integer*	Truncates its argument to an integer. It returns the integer of the same sign as *NUM* and with the largest magnitude such that `abs(INT) <= abs(NUM)`.
`log(NUM)`	*Real*	Returns the natural (base *e*) logarithm of *NUM*, which must not be a zero of any kind.
`log10(NUM)`	*Real*	Returns the base 10 logarithm of *NUM*, which must not be a zero of any kind. This function is defined as `log(NUM) / log(10)`.
`negate(NUM)`	*Number*	Used to negate *NUM*. Returns the corresponding number with the opposite sign.
`round(NUM)`	*Integer*	Used to round *NUM* to an integer by taking `intof(NUM+0.5)` if *NUM* is positive or `intof(NUM–0.5)` if *NUM* is negative.
`sign(NUM)`	*Number*	Used to determine the sign of *NUM*. This operation returns –1, 0, or 1 if *NUM* is an integer. If *NUM* is a real, it returns –1.0, 0.0, or 1.0, depending on whether *NUM* is negative, zero, or positive.
`sqrt(NUM)`	*Real*	Returns the square root of *NUM*. *NUM* must be positive.
`sum_n(LIST)`	*Number*	Returns the sum of values from a list of numeric fields or null if all of the field values are null.
`mean_n(LIST)`	*Number*	Returns the mean value from a list of numeric fields or null if all of the field values are null.
`sdev_n(LIST)`	*Number*	Returns the standard deviation from a list of numeric fields or null if all of the field values are null.

Table 25. CLEM numeric functions (continued)

Trigonometric Functions

All of the functions in this section either take an angle as an argument or return one as a result. In both cases, the units of the angle (radians or degrees) are controlled by the setting of the relevant stream option.

Table 26. CLEM trigonometric functions		
Function	**Result**	**Description**
`arccos(NUM)`	*Real*	Computes the arccosine of the specified angle.
`arccosh(NUM)`	*Real*	Computes the hyperbolic arccosine of the specified angle.
`arcsin(NUM)`	*Real*	Computes the arcsine of the specified angle.
`arcsinh(NUM)`	*Real*	Computes the hyperbolic arcsine of the specified angle.
`arctan(NUM)`	*Real*	Computes the arctangent of the specified angle.
`arctan2(NUM_Y, NUM_X)`	*Real*	Computes the arctangent of NUM_Y / NUM_X and uses the signs of the two numbers to derive quadrant information. The result is a real in the range - `pi < ANGLE <= pi` (radians) − 180 < ANGLE <= 180 (degrees)
`arctanh(NUM)`	*Real*	Computes the hyperbolic arctangent of the specified angle.
`cos(NUM)`	*Real*	Computes the cosine of the specified angle.
`cosh(NUM)`	*Real*	Computes the hyperbolic cosine of the specified angle.
`pi`	*Real*	This constant is the best real approximation to pi.
`sin(NUM)`	*Real*	Computes the sine of the specified angle.
`sinh(NUM)`	*Real*	Computes the hyperbolic sine of the specified angle.
`tan(NUM)`	*Real*	Computes the tangent of the specified angle.
`tanh(NUM)`	*Real*	Computes the hyperbolic tangent of the specified angle.

Probability Functions

Probability functions return probabilities based on various distributions, such as the probability that a value from Student's *t* distribution will be less than a specific value.

Table 27. CLEM probability functions		
Function	**Result**	**Description**
`cdf_chisq(NUM, DF)`	*Real*	Returns the probability that a value from the chi-square distribution with the specified degrees of freedom will be less than the specified number.
`cdf_f(NUM, DF1, DF2)`	*Real*	Returns the probability that a value from the *F* distribution, with degrees of freedom *DF1* and *DF2*, will be less than the specified number.
`cdf_normal(NUM, MEAN, STDDEV)`	*Real*	Returns the probability that a value from the normal distribution with the specified mean and standard deviation will be less than the specified number.
`cdf_t(NUM, DF)`	*Real*	Returns the probability that a value from Student's *t* distribution with the specified degrees of freedom will be less than the specified number.

Spatial functions

Spatial functions can be used with geospatial data. For example, they allow you to calculate the distances between two points, the area of a polygon, and so on. There can also be situations that require a merge of multiple geospatial data sets that are based on a spatial predicate (within, close to, and so on), which can be done through a merge condition.

These spatial functions work in conjunction with the coordinate system specified in **Tools** > **Stream Properties** > **Options** > **Geospatial**.

Note: These spatial functions do not apply to three-dimensional data. If three-dimensional data is imported into the stream, only the first two dimensions are used by these functions. The z-axis values are ignored.

Table 28. CLEM spatial functions

Function	Result	Description
`close_to(SHAPE,SHAPE,NUM)`	*Boolean*	Tests whether 2 shapes are within a certain DISTANCE of each other. If a projected coordinate system is used, DISTANCE is in meters. If no coordinate system is used, it is an arbitrary unit.
`crosses(SHAPE,SHAPE)`	*Boolean*	Tests whether 2 shapes cross each other. This function is suitable for 2 linestring shapes, or 1 linestring and 1 polygon.
`overlap(SHAPE,SHAPE)`	*Boolean*	Tests whether there is an intersection between 2 polygons and that the intersection is interior to both shapes.
`within(SHAPE,SHAPE)`	*Boolean*	Tests whether the entirety of SHAPE1 is contained within a POLYGON.
`area(SHAPE)`	*Real*	Returns the area of the specified POLYGON. If a projected system is used, the function returns meters squared. If no coordinate system is used, it is an arbitrary unit. The shape must be a POLYGON or a MULTIPOLYGON.
`num_points(SHAPE,LIST)`	*Integer*	Returns the number of points from a point field (MULTIPOINT) which are contained within the bounds of a POLYGON. SHAPE1 must be a POLYGON or a MULTIPOLYGON.
`distance(SHAPE,SHAPE)`	*Real*	Returns the distance between SHAPE1 and SHAPE2. If a projected coordinate system is used, the function returns meters. If no coordinate system is used, it is an arbitrary unit. SHAPE1 and SHAPE2 can be any geo measurement type.

Bitwise Integer Operations

These functions enable integers to be manipulated as bit patterns representing two's-complement values, where bit position N has weight $2**N$. Bits are numbered from 0 upward. These operations act as though the sign bit of an integer is extended indefinitely to the left. Thus, everywhere above its most significant bit, a positive integer has 0 bits and a negative integer has 1 bit.

Table 29. CLEM bitwise integer operations		
Function	**Result**	**Description**
`~~ INT1`	*Integer*	Produces the bitwise complement of the integer *INT1*. That is, there is a 1 in the result for each bit position for which *INT1* has 0. It is always true that `~~ INT = -(INT + 1)`.
`INT1 || INT2`	*Integer*	The result of this operation is the bitwise "inclusive or" of *INT1* and *INT2*. That is, there is a 1 in the result for each bit position for which there is a 1 in either *INT1* or *INT2* or both.
`INT1 ||/& INT2`	*Integer*	The result of this operation is the bitwise "exclusive or" of *INT1* and *INT2*. That is, there is a 1 in the result for each bit position for which there is a 1 in either *INT1* or *INT2* but not in both.
`INT1 && INT2`	*Integer*	Produces the bitwise "and" of the integers *INT1* and *INT2*. That is, there is a 1 in the result for each bit position for which there is a 1 in both *INT1* and *INT2*.
`INT1 &&~~ INT2`	*Integer*	Produces the bitwise "and" of *INT1* and the bitwise complement of *INT2*. That is, there is a 1 in the result for each bit position for which there is a 1 in *INT1* and a 0 in *INT2*. This is the same as `INT1 && (~~INT2)` and is useful for clearing bits of *INT1* set in *INT2*.
`INT << N`	*Integer*	Produces the bit pattern of *INT1* shifted left by *N* positions. A negative value for *N* produces a right shift.
`INT >> N`	*Integer*	Produces the bit pattern of *INT1* shifted right by *N* positions. A negative value for *N* produces a left shift.
`INT1 &&=_0 INT2`	*Boolean*	Equivalent to the Boolean expression `INT1 && INT2 /== 0` but is more efficient.
`INT1 &&/=_0 INT2`	*Boolean*	Equivalent to the Boolean expression `INT1 && INT2 == 0` but is more efficient.
`integer_bitcount(INT)`	*Integer*	Counts the number of 1 or 0 bits in the two's-complement representation of *INT*. If *INT* is non-negative, *N* is the number of 1 bits. If *INT* is negative, it is the number of 0 bits. Owing to the sign extension, there are an infinite number of 0 bits in a non-negative integer or 1 bits in a negative integer. It is always the case that `integer_bitcount(INT) = integer_bitcount(-(INT+1))`.
`integer_leastbit(INT)`	*Integer*	Returns the bit position *N* of the least-significant bit set in the integer *INT*. *N* is the highest power of 2 by which *INT* divides exactly.
`integer_length(INT)`	*Integer*	Returns the length in bits of *INT* as a two's-complement integer. That is, *N* is the smallest integer such that `INT < (1 << N) if INT >= 0 INT >= (-1 << N) if INT < 0`. If *INT* is non-negative, then the representation of *INT* as an unsigned integer requires a field of at least *N* bits. Alternatively, a minimum of *N+1* bits is required to represent *INT* as a signed integer, regardless of its sign.

Table 29. CLEM bitwise integer operations (continued)		
Function	**Result**	**Description**
testbit(INT, N)	*Boolean*	Tests the bit at position *N* in the integer *INT* and returns the state of bit *N* as a Boolean value, which is true for 1 and false for 0.

Random Functions

The following functions are used to randomly select items or randomly generate numbers.

Table 30. CLEM random functions		
Function	**Result**	**Description**
oneof(LIST)	*Any*	Returns a randomly chosen element of *LIST*. List items should be entered as [ITEM1,ITEM2,...,ITEM_N]. Note that a list of field names can also be specified.
random(NUM)	*Number*	Returns a uniformly distributed random number of the same type (*INT* or *REAL*), starting from 1 to *NUM*. If you use an integer, then only integers are returned. If you use a real (decimal) number, then real numbers are returned (decimal precision determined by the stream options). The largest random number returned by the function could equal *NUM*.
random0(NUM)	*Number*	This has the same properties as random(NUM), but starting from 0. The largest random number returned by the function will never equal *NUM*.

String Functions

In CLEM, you can perform the following operations with strings:

- Compare strings
- Create strings
- Access characters

In CLEM, a string is any sequence of characters between matching double quotation marks ("string quotes"). Characters (CHAR) can be any single alphanumeric character. They are declared in CLEM expressions using single backquotes in the form of `` `<character>` ``, such as `` `z` ``, `` `A` ``, or `` `2` ``. Characters that are out-of-bounds or negative indices to a string will result in undefined behavior.

Note: Comparisons between strings that do and do not use SQL pushback may generate different results where trailing spaces exist.

Table 31. CLEM string functions		
Function	**Result**	**Description**
allbutfirst(N, STRING)	*String*	Returns a string, which is *STRING* with the first *N* characters removed.
allbutlast(N, STRING)	*String*	Returns a string, which is *STRING* with the last characters removed.
alphabefore(STRING1, STRING2)	*Boolean*	Used to check the alphabetical ordering of strings. Returns true if *STRING1* precedes *STRING2*.

Table 31. CLEM string functions (continued)		
Function	**Result**	**Description**
`endstring(LENGTH, STRING)`	*String*	Extracts the last *N* characters from the specified string. If the string length is less than or equal to the specified length, then it is unchanged.
`hasendstring(STRING, SUBSTRING)`	*Integer*	This function is the same as `isendstring(SUBSTRING, STRING)`.
`hasmidstring(STRING, SUBSTRING)`	*Integer*	This function is the same as `ismidstring(SUBSTRING, STRING)` (embedded substring).
`hasstartstring(STRING, SUBSTRING)`	*Integer*	This function is the same as `isstartstring(SUBSTRING, STRING)`.
`hassubstring(STRING, N, SUBSTRING)`	*Integer*	This function is the same as `issubstring(SUBSTRING, N, STRING)`, where *N* defaults to 1.
`count_substring(STRING, SUBSTRING)`	*Integer*	Returns the number of times the specified substring occurs within the string. For example, `count_substring("foooo.txt", "oo")` returns 3.
`hassubstring(STRING, SUBSTRING)`	*Integer*	This function is the same as `issubstring(SUBSTRING, 1, STRING)`, where *N* defaults to 1.
`isalphacode(CHAR)`	*Boolean*	Returns a value of true if *CHAR* is a character in the specified string (often a field name) whose character code is a letter. Otherwise, this function returns a value of 0. For example, `isalphacode(produce_num(1))`.
`isendstring(SUBSTRING, STRING)`	*Integer*	If the string *STRING* ends with the substring *SUBSTRING*, then this function returns the integer subscript of *SUBSTRING* in *STRING*. Otherwise, this function returns a value of 0.
`islowercode(CHAR)`	*Boolean*	Returns a value of true if *CHAR* is a lowercase letter character for the specified string (often a field name). Otherwise, this function returns a value of 0. For example, both `islowercode(``)` and `islowercode(country_name(2))` are valid expressions.
`ismidstring(SUBSTRING, STRING)`	*Integer*	If *SUBSTRING* is a substring of *STRING* but does not start on the first character of *STRING* or end on the last, then this function returns the subscript at which the substring starts. Otherwise, this function returns a value of 0.

Table 31. CLEM string functions (continued)		
Function	**Result**	**Description**
`isnumbercode(CHAR)`	*Boolean*	Returns a value of true if *CHAR* for the specified string (often a field name) is a character whose character code is a digit. Otherwise, this function returns a value of 0. For example, `isnumbercode(product_id(2))`.
`isstartstring(SUBSTRING, STRING)`	*Integer*	If the string *STRING* starts with the substring *SUBSTRING*, then this function returns the subscript 1. Otherwise, this function returns a value of 0.
`issubstring(SUBSTRING, N, STRING)`	*Integer*	Searches the string *STRING*, starting from its *N*th character, for a substring equal to the string *SUBSTRING*. If found, this function returns the integer subscript at which the matching substring begins. Otherwise, this function returns a value of 0. If *N* is not given, this function defaults to 1.
`issubstring(SUBSTRING, STRING)`	*Integer*	Searches the string *STRING*, starting from its *N*th character, for a substring equal to the string *SUBSTRING*. If found, this function returns the integer subscript at which the matching substring begins. Otherwise, this function returns a value of 0. If *N* is not given, this function defaults to 1.
`issubstring_count(SUBSTRING, N, STRING):`	*Integer*	Returns the index of the *N*th occurrence of *SUBSTRING* within the specified *STRING*. If there are fewer than *N* occurrences of *SUBSTRING*, 0 is returned.
`issubstring_lim(SUBSTRING, N, STARTLIM, ENDLIM, STRING)`	*Integer*	This function is the same as `issubstring`, but the match is constrained to start on or before the subscript *STARTLIM* and to end on or before the subscript *ENDLIM*. The *STARTLIM* or *ENDLIM* constraints may be disabled by supplying a value of false for either argument—for example, `issubstring_lim(SUBSTRING, N, false, false, STRING)` is the same as `issubstring`.
`isuppercode(CHAR)`	*Boolean*	Returns a value of true if *CHAR* is an uppercase letter character. Otherwise, this function returns a value of 0. For example, both `isuppercode(``)` and `isuppercode(country_name(2))` are valid expressions.
`last(CHAR)`	*String*	Returns the last character *CHAR* of *STRING* (which must be at least one character long).
`length(STRING)`	*Integer*	Returns the length of the string *STRING*-- that is, the number of characters in it.

Table 31. CLEM string functions (continued)		
Function	**Result**	**Description**
locchar(CHAR, N, STRING)	*Integer*	Used to identify the location of characters in symbolic fields. The function searches the string *STRING* for the character *CHAR*, starting the search at the *N*th character of *STRING*. This function returns a value indicating the location (starting at *N*) where the character is found. If the character is not found, this function returns a value of 0. If the function has an invalid offset *(N)* (for example, an offset that is beyond the length of the string), this function returns $null$. For example, locchar(`n`, 2, web_page) searches the field called *web_page* for the `n` character beginning at the second character in the field value. *Note*: Be sure to use single backquotes to encapsulate the specified character.
locchar_back(CHAR, N, STRING)	*Integer*	Similar to locchar, except that the search is performed backward starting from the *N*th character. For example, locchar_back(`n`, 9, web_page) searches the field *web_page* starting from the ninth character and moving backward toward the start of the string. If the function has an invalid offset (for example, an offset that is beyond the length of the string), this function returns $null$. Ideally, you should use locchar_back in conjunction with the function length(<field>) to dynamically use the length of the current value of the field. For example, locchar_back(`n`, (length(web_page)), web_page).
lowertoupper(CHAR) lowertoupper (STRING)	*CHAR* or *String*	Input can be either a string or character, which is used in this function to return a new item of the same type, with any lowercase characters converted to their uppercase equivalents. For example, lowertoupper(`a`), lowertoupper("My string"), and lowertoupper(field_name(2)) are all valid expressions.

Table 31. CLEM string functions (continued)		
Function	**Result**	**Description**
`matches`	*Boolean*	Returns true if a string matches a specified pattern. The pattern must be a string literal; it cannot be a field name containing a pattern. A question mark (?) can be included in the pattern to match exactly one character; an asterisk (*) matches zero or more characters. To match a literal question mark or asterisk (rather than using these as wildcards), a backslash (\) can be used as an escape character.
`replace(SUBSTRING, NEWSUBSTRING, STRING)`	*String*	Within the specified *STRING*, replace all instances of *SUBSTRING* with *NEWSUBSTRING*.
`replicate(COUNT, STRING)`	*String*	Returns a string that consists of the original string copied the specified number of times.
`stripchar(CHAR,STRING)`	*String*	Enables you to remove specified characters from a string or field. You can use this function, for example, to remove extra symbols, such as currency notations, from data to achieve a simple number or name. For example, using the syntax `stripchar(` `` `$` ``, 'Cost')` returns a new field with the dollar sign removed from all values. *Note*: Be sure to use single backquotes to encapsulate the specified character.
`skipchar(CHAR, N, STRING)`	*Integer*	Searches the string *STRING* for any character other than *CHAR*, starting at the *N*th character. This function returns an integer substring indicating the point at which one is found or 0 if every character from the *N*th onward is a *CHAR*. If the function has an invalid offset (for example, an offset that is beyond the length of the string), this function returns $null$. `locchar` is often used in conjunction with the `skipchar` functions to determine the value of *N* (the point at which to start searching the string). For example, `skipchar(` `` `s` ``, (locchar(` `` `s` ``, 1, "MyString")), "MyString")`.
`skipchar_back(CHAR, N, STRING)`	*Integer*	Similar to `skipchar`, except that the search is performed **backward**, starting from the *N*th character.
`startstring(LENGTH, STRING)`	*String*	Extracts the first *N* characters from the specified string. If the string length is less than or equal to the specified length, then it is unchanged.

Function	Result	Description
strmember(CHAR, STRING)	*Integer*	Equivalent to locchar(CHAR, 1, STRING). It returns an integer substring indicating the point at which *CHAR* first occurs, or 0. If the function has an invalid offset (for example, an offset that is beyond the length of the string), this function returns $null$.
subscrs(N, STRING)	*CHAR*	Returns the *N*th character *CHAR* of the input string *STRING*. This function can also be written in a shorthand form as STRING(N). For example, lowertoupper("name"(1)) is a valid expression.
substring(N, LEN, STRING)	*String*	Returns a string *SUBSTRING*, which consists of the *LEN* characters of the string *STRING*, starting from the character at subscript *N*.
substring_between(N1, N2, STRING)	*String*	Returns the substring of *STRING*, which begins at subscript *N1* and ends at subscript *N2*.
trim(STRING)	*String*	Removes leading and trailing white space characters from the specified string.
trim_start(STRING)	*String*	Removes leading white space characters from the specified string.
trimend(STRING)	*String*	Removes trailing white space characters from the specified string.
unicode_char(NUM)	*CHAR*	Input must be decimal, not hexadecimal values. Returns the character with Unicode value *NUM.*
unicode_value(CHAR)	*NUM*	Returns the Unicode value of *CHAR*
uppertolower(CHAR) uppertolower (STRING)	*CHAR* or *String*	Input can be either a string or character and is used in this function to return a new item of the same type with any uppercase characters converted to their lowercase equivalents. *Note*: Remember to specify strings with double quotes and characters with single backquotes. Simple field names should be specified without quotes.

SoundEx Functions

SoundEx is a method used to find strings when the sound is known but the precise spelling is not. Developed in 1918, it searches out words with similar sounds based on phonetic assumptions about how certain letters are pronounced. It can be used to search names in a database, for example, where spellings and pronunciations for similar names may vary. The basic SoundEx algorithm is documented in a number of sources and, despite known limitations (for example, leading letter combinations such as ph and f will not match even though they sound the same), is supported in some form by most databases.

Table 32. CLEM soundex functions		
Function	**Result**	**Description**
soundex(STRING)	*Integer*	Returns the four-character SoundEx code for the specified *STRING*.
soundex_difference(STRING1, STRING2)	*Integer*	Returns an integer between 0 and 4 that indicates the number of characters that are the same in the SoundEx encoding for the two strings, where 0 indicates no similarity and 4 indicates strong similarity or identical strings.

Date and Time Functions

CLEM includes a family of functions for handling fields with datetime storage of string variables representing dates and times. The formats of date and time used are specific to each stream and are specified in the stream properties dialog box. The date and time functions parse date and time strings according to the currently selected format.

When you specify a year in a date that uses only two digits (that is, the century is not specified), IBM SPSS Modeler uses the default century that is specified in the stream properties dialog box.

Note: If the data function is pushed back to SQL or IBM SPSS Analytic Server, in a branch that follows an Analytic Server data source, any date format strings (to_date) within that data must match the date format specified in the SPSS Modeler stream.

Table 33. CLEM date and time functions		
Function	**Result**	**Description**
@TODAY	*String*	If you select **Rollover days/mins** in the stream properties dialog box, this function returns the current date as a string in the current date format. If you use a two-digit date format and do not select **Rollover days/mins**, this function returns $null$ on the current server.
to_time(ITEM)	*Time*	Converts the storage of the specified field to a time.
to_date(ITEM)	*Date*	Converts the storage of the specified field to a date.
to_timestamp(ITEM)	*Timestamp*	Converts the storage of the specified field to a timestamp.
to_datetime(ITEM)	*Datetime*	Converts the storage of the specified field to a date, time, or timestamp value.
datetime_date(ITEM)	*Date*	Returns the date value for a *number*, *string*, or *timestamp*. Note this is the only function that allows you to convert a number (in seconds) back to a date. If ITEM is a string, creates a date by parsing a string in the current date format. The date format specified in the stream properties dialog box must be correct for this function to be successful. If ITEM is a number, it is interpreted as a number of seconds since the base date (or epoch). Fractions of a day are truncated. If ITEM is timestamp, the date part of the timestamp is returned. If ITEM is a date, it is returned unchanged.

Table 33. CLEM date and time functions (continued)

Function	Result	Description
`date_add_years(NUM, DATE)`	*Date*	Returns the new DATE or timestamp after adding the specified value for the specified date part YEAR to a DATE or timestamp.
`date_add_years(NUM,TIMES TAMP)`	*Timestamp*	Returns the new DATE or timestamp after adding the specified value for the specified date part YEAR to a DATE or timestamp.
`date_add_months(NUM, DATE)`	*Date*	Returns the new DATE or timestamp after adding the specified value for the specified date part MONTH to a DATE or timestamp.
`date_add_months(NUM,TIME STAMP)`	*Timestamp*	Returns the new DATE or timestamp after adding the specified value for the specified date part MONTH to a DATE or timestamp.
`date_add_weeks(NUM, DATE)`	*Date*	Returns the new DATE or timestamp after adding the specified value for the specified date part WEEK to a DATE or timestamp.
`date_add_weeks(NUM,TIMES TAMP)`	*Timestamp*	Returns the new DATE or timestamp after adding the specified value for the specified date part WEEK to a DATE or timestamp.
`date_add_days(NUM, DATE)`	*Date*	Returns the new DATE or timestamp after adding the specified value for the specified date part DAY to a DATE or timestamp.
`date_add_days(NUM,TIMEST AMP)`	*Timestamp*	Returns the new DATE or timestamp after adding the specified value for the specified date part DAY to a DATE or timestamp.
`date_before(DATE1, DATE2)`	*Boolean*	Returns a value of true if *DATE1* represents a date or timestamp before that represented by *DATE2*. Otherwise, this function returns a value of 0.
`date_days_difference(DAT E1, DATE2)`	*Integer*	Returns the time in days from the date or timestamp represented by *DATE1* to that represented by *DATE2*, as an integer. If *DATE2* is before *DATE1*, this function returns a negative number.
`date_in_days(DATE)`	*Integer*	Returns the time in days from the baseline date to the date or timestamp represented by *DATE*, as an integer. If *DATE* is before the baseline date, this function returns a negative number. You must include a valid date for the calculation to work appropriately. For example, you should not specify 29 February 2001 as the date. Because 2001 is a not a leap year, this date does not exist.

Function	Result	Description
`date_in_months(DATE)`	*Real*	Returns the time in months from the baseline date to the date or timestamp represented by *DATE*, as a real number. This is an approximate figure based on a month of 30.4375 days. If *DATE* is before the baseline date, this function returns a negative number. You must include a valid date for the calculation to work appropriately. For example, you should not specify 29 February 2001 as the date. Because 2001 is a not a leap year, this date does not exist.
`date_in_weeks(DATE)`	*Real*	Returns the time in weeks from the baseline date to the date or timestamp represented by *DATE*, as a real number. This is based on a week of 7.0 days. If *DATE* is before the baseline date, this function returns a negative number. You must include a valid date for the calculation to work appropriately. For example, you should not specify 29 February 2001 as the date. Because 2001 is a not a leap year, this date does not exist.
`date_in_years(DATE)`	*Real*	Returns the time in years from the baseline date to the date or timestamp represented by *DATE*, as a real number. This is an approximate figure based on a year of 365.25 days. If *DATE* is before the baseline date, this function returns a negative number. You must include a valid date for the calculation to work appropriately. For example, you should not specify 29 February 2001 as the date. Because 2001 is a not a leap year, this date does not exist.
`date_months_difference (DATE1, DATE2)`	*Real*	Returns the time in months from the date or timestamp represented by *DATE1* to that represented by *DATE2*, as a real number. This is an approximate figure based on a month of 30.4375 days. If *DATE2* is before *DATE1*, this function returns a negative number.
`datetime_date(YEAR, MONTH, DAY)`	*Date*	Creates a date value for the given *YEAR*, *MONTH*, and *DAY*. The arguments must be integers.
`datetime_day(DATE)`	*Integer*	Returns the day of the month from a given *DATE* or timestamp. The result is an integer in the range 1 to 31.
`datetime_day_name(DAY)`	*String*	Returns the full name of the given *DAY*. The argument must be an integer in the range 1 (Sunday) to 7 (Saturday).
`datetime_hour(TIME)`	*Integer*	Returns the hour from a *TIME* or timestamp. The result is an integer in the range 0 to 23.
`datetime_in_seconds(TIME)`	*Real*	Returns the seconds portion stored in *TIME*.
`datetime_in_seconds(DATE), datetime_in_seconds(DATETIME)`	*Real*	Returns the accumulated number, converted into seconds, from the difference between the current *DATE* or *DATETIME* and the baseline date (1900-01-01).

Function	Result	Description
`datetime_minute(TIME)`	*Integer*	Returns the minute from a *TIME* or timestamp. The result is an integer in the range 0 to 59.
`datetime_month(DATE)`	*Integer*	Returns the month from a *DATE* or timestamp. The result is an integer in the range 1 to 12.
`datetime_month_name (MONTH)`	*String*	Returns the full name of the given *MONTH*. The argument must be an integer in the range 1 to 12.
`datetime_now`	*Timestamp*	Returns the current time as a timestamp.
`datetime_second(TIME)`	*Integer*	Returns the second from a *TIME* or timestamp. The result is an integer in the range 0 to 59.
`datetime_day_short_name (DAY)`	*String*	Returns the abbreviated name of the given *DAY*. The argument must be an integer in the range 1 (Sunday) to 7 (Saturday).
`datetime_month_short_nam e (MONTH)`	*String*	Returns the abbreviated name of the given *MONTH*. The argument must be an integer in the range 1 to 12.
`datetime_time(HOUR, MINUTE, SECOND)`	*Time*	Returns the time value for the specified *HOUR*, *MINUTE*, and *SECOND*. The arguments must be integers.
`datetime_time(ITEM)`	*Time*	Returns the time value of the given *ITEM*.
`datetime_timestamp(YEAR, MONTH, DAY, HOUR, MINUTE, SECOND)`	*Timestamp*	Returns the timestamp value for the given *YEAR*, *MONTH*, *DAY*, *HOUR*, *MINUTE*, and *SECOND*.
`datetime_timestamp(DATE, TIME)`	*Timestamp*	Returns the timestamp value for the given *DATE* and *TIME*.
`datetime_timestamp (NUMBER)`	*Timestamp*	Returns the timestamp value of the given number of seconds.
`datetime_weekday(DATE)`	*Integer*	Returns the day of the week from the given *DATE* or timestamp.
`datetime_year(DATE)`	*Integer*	Returns the year from a *DATE* or timestamp. The result is an integer such as 2002.
`date_weeks_difference (DATE1, DATE2)`	*Real*	Returns the time in weeks from the date or timestamp represented by *DATE1* to that represented by *DATE2*, as a real number. This is based on a week of 7.0 days. If *DATE2* is before *DATE1*, this function returns a negative number.
`date_years_difference (DATE1, DATE2)`	*Real*	Returns the time in years from the date or timestamp represented by *DATE1* to that represented by *DATE2*, as a real number. This is an approximate figure based on a year of 365.25 days. If *DATE2* is before *DATE1*, this function returns a negative number.
`date_from_ywd(YEAR, WEEK, DAY)`	*Integer*	Converts the year, week in year, and day in week, to a date using the ISO 8601 standard.
`date_iso_day(DATE)`	*Integer*	Returns the day in the week from the date using the ISO 8601 standard.

Table 33. CLEM date and time functions (continued)		
Function	**Result**	**Description**
`date_iso_week(DATE)`	*Integer*	Returns the week in the year from the date using the ISO 8601 standard.
`date_iso_year(DATE)`	*Integer*	Returns the year from the date using the ISO 8601 standard.
`time_add_hours(NUM, DATE)`	*Date*	Returns the new TIME or timestamp after adding the specified value for the specified time part HOUR to a TIME or timestamp.
`time_add_hours(NUM,TIMES TAMP)`	*Timestamp*	Returns the new TIME or timestamp after adding the specified value for the specified time part HOUR to a TIME or timestamp.
`time_add_minutes(NUM, DATE)`	*Date*	Returns the new TIME or timestamp after adding the specified value for the specified time part MINUTE to a TIME or timestamp.
`time_add_minutes(NUM,TIM ESTAMP)`	*Timestamp*	Returns the new TIME or timestamp after adding the specified value for the specified time part MINUTE to a TIME or timestamp.
`time_add_seconds(NUM, DATE)`	*Date*	Returns the new TIME or timestamp after adding the specified value for the specified time part SECOND to a TIME or timestamp.
`time_add_seconds(NUM,TIM ESTAMP)`	*Timestamp*	Returns the new TIME or timestamp after adding the specified value for the specified time part SECOND to a TIME or timestamp.
`time_before(TIME1, TIME2)`	*Boolean*	Returns a value of true if *TIME1* represents a time or timestamp before that represented by *TIME2*. Otherwise, this function returns a value of 0.
`time_hours_difference (TIME1, TIME2)`	*Real*	Returns the time difference in hours between the times or timestamps represented by *TIME1* and *TIME2*, as a real number. If you select **Rollover days/ mins** in the stream properties dialog box, a higher value of *TIME1* is taken to refer to the previous day. If you do not select the rollover option, a higher value of *TIME1* causes the returned value to be negative.
`time_in_hours(TIME)`	*Real*	Returns the time in hours represented by *TIME*, as a real number. For example, under time format HHMM, the expression `time_in_hours('0130')` evaluates to 1.5. *TIME* can represent a time or a timestamp.
`time_in_mins(TIME)`	*Real*	Returns the time in minutes represented by *TIME*, as a real number. *TIME* can represent a time or a timestamp.
`time_in_secs(TIME)`	*Integer*	Returns the time in seconds represented by *TIME*, as an integer. *TIME* can represent a time or a timestamp.

Function	Result	Description
`time_mins_difference(TIME1, TIME2)`	*Real*	Returns the time difference in minutes between the times or timestamps represented by *TIME1* and *TIME2*, as a real number. If you select **Rollover days/mins** in the stream properties dialog box, a higher value of *TIME1* is taken to refer to the previous day (or the previous hour, if only minutes and seconds are specified in the current format). If you do not select the rollover option, a higher value of *TIME1* will cause the returned value to be negative.
`time_secs_difference(TIME1, TIME2)`	*Integer*	Returns the time difference in seconds between the times or timestamps represented by *TIME1* and *TIME2*, as an integer. If you select **Rollover days/mins** in the stream properties dialog box, a higher value of *TIME1* is taken to refer to the previous day (or the previous hour, if only minutes and seconds are specified in the current format). If you do not select the rollover option, a higher value of *TIME1* causes the returned value to be negative.

Table 33. CLEM date and time functions (continued)

Converting Date and Time Values

Note that conversion functions (and any other functions that require a specific type of input, such as a date or time value) depend on the current formats specified in the Stream Options dialog box. For example, if you have a field named *DATE* that is stored as a string with values *Jan 2003*, *Feb 2003*, and so on, you could convert it to date storage as follows:

```
to_date(DATE)
```

For this conversion to work, select the matching date format **MON YYYY** as the default date format for the stream.

For an example that converts string values to dates using a Filler node, see the stream *broadband_create_models.str*, installed in the *\Demos* folder under the *streams* subfolder.

Dates stored as numbers. Note that *DATE* in the previous example is the name of a field, while `to_date` is a CLEM function. If you have dates stored as numbers, you can convert them using the `datetime_date` function, where the number is interpreted as a number of seconds since the base date (or epoch).

```
datetime_date(DATE)
```

By converting a date to a number of seconds (and back), you can perform calculations such as computing the current date plus or minus a fixed number of days, for example:

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

Sequence functions

For some operations, the sequence of events is important. The application allows you to work with the following record sequences:

- Sequences and time series
- Sequence functions
- Record indexing

- Averaging, summing, and comparing values
- Monitoring change--differentiation
- @SINCE
- Offset values
- Additional sequence facilities

For many applications, each record passing through a stream can be considered as an individual case, independent of all others. In such situations, the order of records is usually unimportant.

For some classes of problems, however, the record sequence is very important. These are typically time series situations, in which the sequence of records represents an ordered sequence of events or occurrences. Each record represents a snapshot at a particular instant in time; much of the richest information, however, might be contained not in instantaneous values but in the way in which such values are changing and behaving over time.

Of course, the relevant parameter may be something other than time. For example, the records could represent analyses performed at distances along a line, but the same principles would apply.

Sequence and special functions are immediately recognizable by the following characteristics:

- They are all prefixed by @.
- Their names are given in upper case.

Sequence functions can refer to the record currently being processed by a node, the records that have already passed through a node, and even, in one case, records that have yet to pass through a node. Sequence functions can be mixed freely with other components of CLEM expressions, although some have restrictions on what can be used as their arguments.

Examples

You may find it useful to know how long it has been since a certain event occurred or a condition was true. Use the function @SINCE to do this—for example:

```
@SINCE(Income > Outgoings)
```

This function returns the offset of the last record where this condition was true--that is, the number of records before this one in which the condition was true. If the condition has never been true, @SINCE returns @INDEX + 1.

Sometimes you may want to refer to a value of the current record in the expression used by @SINCE. You can do this using the function @THIS, which specifies that a field name always applies to the current record. To find the offset of the last record that had a `Concentration` field value more than twice that of the current record, you could use:

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

In some cases the condition given to @SINCE is true of the current record by definition—for example:

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

For this reason, @SINCE does not evaluate its condition for the current record. Use a similar function, @SINCE0, if you want to evaluate the condition for the current record as well as previous ones; if the condition is true in the current record, @SINCE0 returns 0.

Table 34. CLEM sequence functions		
Function	**Result**	**Description**
MEAN(FIELD)	*Real*	Returns the mean average of values for the specified *FIELD* or *FIELDS*.

Table 34. CLEM sequence functions (continued)		
Function	**Result**	**Description**
@MEAN(FIELD, EXPR)	*Real*	Returns the mean average of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted or if it exceeds the number of records received so far, the average over all of the records received so far is returned.
@MEAN(FIELD, EXPR, INT)	*Real*	Returns the mean average of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted or if it exceeds the number of records received so far, the average over all of the records received so far is returned. *INT* specifies the maximum number of values to look back. This is far more efficient than using just two arguments.
@DIFF1(FIELD)	*Real*	Returns the first differential of *FIELD*. The single-argument form thus simply returns the difference between the current value and the previous value of the field. Returns **$null$** if the relevant previous records do not exist.
@DIFF1(FIELD1, FIELD2)	*Real*	The two-argument form gives the first differential of *FIELD1* with respect to *FIELD2*. Returns **$null$** if the relevant previous records do not exist. It is calculated as @DIFF1(FIELD1)/@DIFF1(FIELD2).
@DIFF2(FIELD)	*Real*	Returns the second differential of *FIELD*. The single-argument form thus simply returns the difference between the current value and the previous value of the field. Returns **$null$** if the relevant previous records do not exist. @DIFF2 is calculated as @DIFF(@DIFF(FIELD)).
@DIFF2(FIELD1, FIELD2)	*Real*	The two-argument form gives the second differential of *FIELD1* with respect to *FIELD2*. Returns **$null$** if the relevant previous records do not exist. This is a complex calculation -- @DIFF1(FIELD1)/@DIFF1(FIELD2) - @OFFSET(@DIFF1(FIELD1),1)/ @OFFSET(@DIFF1(FIELD2))) / @DIFF1(FIELD2).
@INDEX	*Integer*	Returns the index of the current record. Indices are allocated to records as they arrive at the current node. The first record is given index 1, and the index is incremented by 1 for each subsequent record.

Table 34. CLEM sequence functions (continued)		
Function	**Result**	**Description**
@LAST_NON_BLANK(FIELD)	*Any*	Returns the last value for *FIELD* that was not blank, as defined in an upstream source or Type node. If there are no nonblank values for *FIELD* in the records read so far, $null$ is returned. Note that blank values, also called user-missing values, can be defined separately for each field.
@MAX(FIELD)	*Number*	Returns the maximum value for the specified *FIELD*.
@MAX(FIELD, EXPR)	*Number*	Returns the maximum value for *FIELD* over the last *EXPR* records received so far, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0.
@MAX(FIELD, EXPR, INT)	*Number*	Returns the maximum value for *FIELD* over the last *EXPR* records received so far, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the maximum value over all of the records received so far is returned. *INT* specifies the maximum number of values to look back. This is far more efficient than using just two arguments.
@MIN(FIELD)	*Number*	Returns the minimum value for the specified *FIELD*.
@MIN(FIELD, EXPR)	*Number*	Returns the minimum value for *FIELD* over the last *EXPR* records received so far, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0.
@MIN(FIELD, EXPR, INT)	*Number*	Returns the minimum value for *FIELD* over the last *EXPR* records received so far, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the minimum value over all of the records received so far is returned. *INT* specifies the maximum number of values to look back. This is far more efficient than using just two arguments.

Table 34. CLEM sequence functions (continued)		
Function	**Result**	**Description**
@OFFSET(FIELD, EXPR)	*Any*	Returns the value of *FIELD* in the record offset from the current record by the value of *EXPR*. A positive offset refers to a record that has already passed (a "lookback"), while a negative one specifies a "lookahead" to a record that has yet to arrive. For example, @OFFSET(Status, 1) returns the value of the Status field in the previous record, while @OFFSET(Status, −4) "looks ahead" four records in the sequence (that is, to records that have not yet passed through this node) to obtain the value. *Note that a negative (look ahead) offset must be specified as a constant.* For positive offsets only, *EXPR* may also be an arbitrary CLEM expression, which is evaluated for the current record to give the offset. In this case, the three-argument version of this function should improve performance (see next function). If the expression returns anything other than a non-negative integer, this causes an error— that is, it is not legal to have calculated lookahead offsets.

Note: A self-referential @OFFSET function cannot use literal lookahead. For example, in a Filler node, you cannot replace the value of field1 using an expression such as @OFFSET(field1,-2).

Note: In the Filler node, when filling a field, there are effectively two different values of that field, namely the pre-filled value and the post-filled value. When @OFFSET refers to itself it refers to the post-filled value. This post-filled value only exists for past rows so self referential @OFFSET can only refer to past rows. Since self referential @OFFSET cannot refer to the future it carries out the following checks of the offset:

• If the offset is literal, and into the future, an error is reported before execution begins.
• If the offset is an expression and evaluates to the future at runtime then @OFFSET returns $null$.

Note: Using both "lookahead" and "lookback" within one node is not supported. |

Function	Result	Description
@OFFSET(FIELD, EXPR, INT)	*Any*	Performs the same operation as the @OFFSET function with the addition of a third argument, *INT*, which specifies the maximum number of values to look back. In cases where the offset is computed from an expression, this third argument should improve performance. For example, in an expression such as@OFFSET(Foo, Month, 12), the system knows to keep only the last twelve values of Foo; otherwise, it has to store every value just in case. In cases where the offset value is a constant—including negative "lookahead" offsets, which must be constant—the third argument is pointless and the two-argument version of this function should be used. See also the note about self-referential functions in the two-argument version described earlier. *Note*: Using both "lookahead" and "lookback" within one node is not supported.
@SDEV(FIELD)	*Real*	Returns the standard deviation of values for the specified *FIELD* or *FIELDS*.
@SDEV(FIELD, EXPR)	*Real*	Returns the standard deviation of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the standard deviation over all of the records received so far is returned.
@SDEV(FIELD, EXPR, INT)	*Real*	Returns the standard deviation of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the standard deviation over all of the records received so far is returned. *INT* specifies the maximum number of values to look back. This is far more efficient than using just two arguments.
@SINCE(EXPR)	*Any*	Returns the number of records that have passed since *EXPR*, an arbitrary CLEM expression, was true.
@SINCE(EXPR, INT)	*Any*	Adding the second argument, *INT*, specifies the maximum number of records to look back. If *EXPR* has never been true, *INT* is @INDEX+1.
@SINCE0(EXPR)	*Any*	Considers the current record, while @SINCE does not; @SINCE0 returns 0 if *EXPR* is true for the current record.

Table 34. CLEM sequence functions (continued)

Table 34. CLEM sequence functions (continued)		
Function	**Result**	**Description**
`@SINCE0(EXPR, INT)`	*Any*	Adding the second argument, *INT* specifies the maximum number of records to look back.
`@SUM(FIELD)`	*Number*	Returns the sum of values for the specified *FIELD* or *FIELDS*.
`@SUM(FIELD, EXPR)`	*Number*	Returns the sum of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the sum over all of the records received so far is returned.
`@SUM(FIELD, EXPR, INT)`	*Number*	Returns the sum of values for *FIELD* over the last *EXPR* records received by the current node, including the current record. *FIELD* must be the name of a numeric field. *EXPR* may be any expression evaluating to an integer greater than 0. If *EXPR* is omitted, or if it exceeds the number of records received so far, the sum over all of the records received so far is returned. *INT* specifies the maximum number of values to look back. This is far more efficient than using just two arguments.
`@THIS(FIELD)`	*Any*	Returns the value of the field named *FIELD* in the current record. Used only in @SINCE expressions.

Global Functions

The functions @MEAN, @SUM, @MIN, @MAX, and @SDEV work on, at most, all of the records read up to and including the current one. In some cases, however, it is useful to be able to work out how values in the current record compare with values seen in the entire data set. Using a Set Globals node to generate values across the entire data set, you can access these values in a CLEM expression using the global functions.

For example,

```
@GLOBAL_MAX(Age)
```

returns the highest value of Age in the data set, while the expression

```
(Value - @GLOBAL_MEAN(Value)) / @GLOBAL_SDEV(Value)
```

expresses the difference between this record's Value and the global mean as a number of standard deviations. You can use global values only after they have been calculated by a Set Globals node. All current global values can be canceled by clicking the **Clear Global Values** button on the Globals tab in the stream properties dialog box.

Table 35. CLEM global functions		
Function	**Result**	**Description**
@GLOBAL_MAX(FIELD)	*Number*	Returns the maximum value for *FIELD* over the whole data set, as previously generated by a Set Globals node. *FIELD* must be the name of a numeric, date/time/ datetime, or string field. If the corresponding global value has not been set, an error occurs.
@GLOBAL_MIN(FIELD)	*Number*	Returns the minimum value for *FIELD* over the whole data set, as previously generated by a Set Globals node. *FIELD* must be the name of a numeric, date/time/ datetime, or string field. If the corresponding global value has not been set, an error occurs.
@GLOBAL_SDEV(FIELD)	*Number*	Returns the standard deviation of values for *FIELD* over the whole data set, as previously generated by a Set Globals node. *FIELD* must be the name of a numeric field. If the corresponding global value has not been set, an error occurs.
@GLOBAL_MEAN(FIELD)	*Number*	Returns the mean average of values for *FIELD* over the whole data set, as previously generated by a Set Globals node. *FIELD* must be the name of a numeric field. If the corresponding global value has not been set, an error occurs.
@GLOBAL_SUM(FIELD)	*Number*	Returns the sum of values for *FIELD* over the whole data set, as previously generated by a Set Globals node. *FIELD* must be the name of a numeric field. If the corresponding global value has not been set, an error occurs.

Functions Handling Blanks and Null Values

Using CLEM, you can specify that certain values in a field are to be regarded as "blanks," or missing values. The following functions work with blanks.

Table 36. CLEM blank and null value functions		
Function	**Result**	**Description**
@BLANK(FIELD)	*Boolean*	Returns true for all records whose values are blank according to the blank-handling rules set in an upstream Type node or source node (Types tab).
@LAST_NON_BLANK(FIELD)	*Any*	Returns the last value for *FIELD* that was not blank, as defined in an upstream source or Type node. If there are no nonblank values for *FIELD* in the records read so far, $null$ is returned. Note that blank values, also called user-missing values, can be defined separately for each field.
@NULL(FIELD)	*Boolean*	Returns true if the value of *FIELD* is the system-missing $null$. Returns false for all other values, including user-defined blanks. If you want to check for both, use @BLANK(FIELD) and@NULL(FIELD).
undef	*Any*	Used generally in CLEM to enter a $null$ value— for example, to fill blank values with nulls in the Filler node.

Blank fields may be "filled in" with the Filler node. In both Filler and Derive nodes (multiple mode only), the special CLEM function @FIELD refers to the current field(s) being examined.

Special Fields

Special functions are used to denote the specific fields under examination, or to generate a list of fields as input. For example, when deriving multiple fields at once, you should use @FIELD to denote "perform this derive action on the selected fields." Using the expression log(@FIELD) derives a new log field for each selected field.

Function	Result	Description
Table 37. CLEM special fields		
@FIELD	*Any*	Performs an action on all fields specified in the expression context.
@TARGET	*Any*	When a CLEM expression is used in a user-defined analysis function, @TARGET represents the target field or "correct value" for the target/predicted pair being analyzed. This function is commonly used in an Analysis node.
@PREDICTED	*Any*	When a CLEM expression is used in a user-defined analysis function, @PREDICTED represents the predicted value for the target/predicted pair being analyzed. This function is commonly used in an Analysis node.
@PARTITION_FIELD	*Any*	Substitutes the name of the current partition field.
@TRAINING_PARTITION	*Any*	Returns the value of the current training partition. For example, to select training records using a Select node, use the CLEM expression: @PARTITION_FIELD = @TRAINING_PARTITION This ensures that the Select node will always work regardless of which values are used to represent each partition in the data.
@TESTING_PARTITION	*Any*	Returns the value of the current testing partition.
@VALIDATION_PARTITION	*Any*	Returns the value of the current validation partition.
@FIELDS_BETWEEN(start, end)	*Any*	Returns the list of field names between the specified start and end fields (inclusive) based on the natural (that is, insert) order of the fields in the data.

Table 37. CLEM special fields (continued)		
Function	**Result**	**Description**
`@FIELDS_MATCHING(pattern)`	*Any*	Returns a list a field names matching a specified pattern. A question mark (?) can be included in the pattern to match exactly one character; an asterisk (*) matches zero or more characters. To match a literal question mark or asterisk (rather than using these as wildcards), a backslash (\) can be used as an escape character. **Note:** This requires a string literal as an argument; it cannot use a nested expression to generate the argument.
`@MULTI_RESPONSE_SET`	*Any*	Returns the list of fields in the named multiple response set.

Chapter 12. Using IBM SPSS Modeler with a repository

About the IBM SPSS Collaboration and Deployment Services Repository

SPSS Modeler can be used in conjunction with an IBM SPSS Collaboration and Deployment Services repository, enabling you to manage the life cycle of data mining models and related predictive objects, and enabling these objects to be used by enterprise applications, tools, and solutions. IBM SPSS Modeler objects that can be shared in this way include streams, nodes, stream outputs, projects, and models. Objects are stored in the central repository, from where they can be shared with other applications and tracked using extended versioning, metadata, and search capabilities.

Before you can use SPSS Modeler with the repository, you need to install an adapter at the repository host. Without this adapter, you may see the following message when attempting to access repository objects from certain SPSS Modeler nodes or models:

```
The repository may need updating to support new node, model and output types.
```

For instructions on installing the adapter, see the *SPSS Modeler Deployment Installation* guide, available as a PDF file as part of your product download. Details of how to access IBM SPSS Modeler repository objects from IBM SPSS Deployment Manager are given in the *SPSS Modeler Deployment Guide.*

The following sections provide information on accessing the repository from within SPSS Modeler.

Extensive Versioning and Search Support

The repository provides comprehensive object versioning and search capabilities. For example, suppose that you create a stream and store it in the repository where it can be shared with researchers from other divisions. If you later update the stream in SPSS Modeler, you can add the updated version to the repository without overwriting the previous version. All versions remain accessible and can be searched by name, label, fields used, or other attributes. You could, for example, search for all model versions that use net revenue as an input, or all models created by a particular author. (To do this with a traditional file system, you would have to save each version under a different filename, and the relationships between versions would be unknown to the software.)

Single Sign-On

The single sign-on feature enables users to connect to the repository without having to enter username and password details each time. The user's existing local network login details provide the necessary authentication to IBM SPSS Collaboration and Deployment Services. This feature depends on the following:

- IBM SPSS Collaboration and Deployment Services must be configured to use a single sign-on provider.
- The user must be logged in to a host that is compatible with the provider.

For more information, see .

Storing and deploying repository objects

Streams created in IBM SPSS Modeler can be **stored** in the repository just as they are, as files with the extension `.str`. In this way, a single stream can be accessed by multiple users throughout the enterprise. See the topic for more information.

It is also possible to deploy a stream in the repository. A deployed stream is stored as a file with additional metadata. A deployed stream can take full advantage of the enterprise-level features of IBM SPSS Collaboration and Deployment Services, such as automated scoring and model refresh. For example, a model can be automatically updated at regularly-scheduled intervals as new data becomes available. Alternatively, a set of streams can be deployed for Champion Challenger analysis, in which streams are compared to determine which one contains the most effective predictive model.

Note: The Association Rules, STP, and TCM modeling nodes do not support the Model Evaluation or Champion Challenger steps in IBM SPSS Collaboration and Deployment Services.

You can deploy a stream (with the extension `.str`). Deployment as a stream enables the stream to be used by the thin-client application IBM SPSS Modeler Advantage. See the topic "Opening a Stream in IBM SPSS Modeler Advantage " on page 213 for more information.

For more information, see "Stream Deployment Options" on page 207.

Other Deployment Options

While IBM SPSS Collaboration and Deployment Services offers the most extensive features for managing enterprise content, a number of other mechanisms for deploying or exporting streams are also available, including:

- Export the stream and model for later use with IBM SPSS Modeler Solution Publisher Runtime.
- Export one or more models in PMML, an XML-based format for encoding model information. See the topic "Importing and exporting models as PMML" on page 213 for more information.

Connecting to the Repository

1. To connect to the repository, on the IBM SPSS Modeler main menu, click:

   Tools > **Repository** > **Options...**

2. In the **RepositoryURL.** field, enter or select the directory path to, or URL of, the repository installation you want to access. You can connect to only one repository at a time.

   Settings are specific to each site or installation. For specific login details, contact your local system administrator.

Set Credentials. Leave this box unchecked to enable the *single sign-on* feature, which attempts to log you in using your local computer username and password details. If single sign-on is not possible, or if you select this option to disable single sign-on (for example, to log in to an administrator account), a screen is displayed for you to enter your credentials.

Entering Credentials for the Repository

Depending on your settings, the following fields may be required in the Repository: Credentials dialog box:

User ID and password. Specify a valid user name and password for logging on. If necessary, contact your local administrator for more information.

Provider. Choose a security provider for authentication. The repository can be configured to use different security providers; if necessary, contact your local administrator for more information.

Remember repository and user ID. Saves the current settings as the default so that you do not have to reenter them each time you want to connect.

Browse for repository credentials

When you connect to a repository from an Analytic Server, Cognos, ODBC, or TM1 source node, you can select previously recorded credentials to connect to a repository. These credentials are listed in the **Select Repository Credential** dialog box. To select this dialog box, click Browse next to the **Credential** field.

In the **Select Repository Credential** dialog box, highlight the credentials in the list supplied and click OK. If the list is too large, use the **Filter** field to enter the name, or part of the name, to find the credentials you require.

Browsing the Repository Contents

The repository allows you to browse stored content in a manner similar to Windows Explorer; you can also browse *versions* of each stored object.

1. To open the IBM SPSS Collaboration and Deployment Services Repository window, on the SPSS Modeler menus click:

Tools > **Repository** > **Explore...**

1. Specify connection settings to the repository if necessary. See the topic <u>"Connecting to the Repository" on page 196</u> for more information. For specific port, password, and other connection details, contact your local system administrator.

The explorer window initially displays a tree view of the folder hierarchy. Click a folder name to display its contents.

Objects that match the current selection or search criteria are listed in the right pane, with detailed information on the selected version displayed in the lower right pane. The attributes displayed apply to the most recent version.

Storing Objects in the Repository

You can store streams, nodes, models, model palettes, projects, and output objects in the repository, from where they can be accessed by other users and applications.

You can also publish stream output to the repository in a format that enables other users to view it over the Internet using the IBM SPSS Collaboration and Deployment Services Deployment Portal.

Setting Object Properties

When you store an object, the Repository: Store dialog box is displayed, enabling you to set the values of a number of properties for the object. You can:

- Choose the name and repository folder under which the object is to be stored
- Add information about the object such as the version label and other searchable properties
- Assign one or more classification topics to the object
- Set security options for the object

The following sections describe the properties you can set.

Choosing the Location for Storing Objects

In the Repository: Store dialog box, enter the following.

Save in. Shows the current folder--the location where the object will be stored. Double-click a folder name in the list to set that folder as the current folder. Use the Up Folder button to navigate to the parent folder. Use the New Folder button to create a folder at the current level.

File name. The name under which the object will be stored.

Store. Stores the object at the current location.

Adding Information About Stored Objects

All of the fields on the Information tab of the Repository: Store dialog box are optional.

Author. The username of the user creating the object in the repository. By default, this shows the username used for the repository connection, but you can change this name here.

Version Label. Select a label from the list to indicate the object version, or click **Add** to create a new label. Avoid using the "[" character in the label. Ensure that no boxes are checked if you do not want to assign a label to this object version. See the topic "Viewing and Editing Object Properties" on page 205 for more information.

Description. A description of the object. Users can search for objects by description (see note).

Keywords. One or more keywords that relate to the object and which can be used for search purposes (see note).

Expiration. A date after which the object is no longer visible to general users, although it can still be seen by its owner and by the repository administrator. To set an expiration date, select the **Date** option and enter the date, or choose one using the calendar button.

Store. Stores the object at the current location.

Note: Information in the **Description** and **Keywords** fields is treated as distinct from anything entered in SPSS Modeler on the Annotations tab of the object. A repository search by description or keyword does not return information from the Annotations tab. See the topic "Searching for objects in the repository " on page 202 for more information.

Assigning Topics to a Stored Object

Topics are a hierarchical classification system for the content stored in the repository. You can choose from the available topics when storing objects, and users can also search for objects by topic. The list of available topics is set by repository users with the appropriate privileges (for more information, see the *Deployment Manager User's Guide*).

To assign a topic to the object, on the Topics tab of the Repository: Store dialog box:

1. Click the **Add** button.
2. Click a topic name from the list of available topics.
3. Click **OK**.

   To remove a topic assignment:
4. Select the topic in the list of assigned topics.
5. Click **Delete**.

Setting Security Options for Stored Objects

You can set or change a number of security options for a stored object on the Security tab of the Repository: Store dialog box. For one or more **principals** (that is, users or groups of users), you can:

- Assign access rights to the object
- Modify access rights to the object
- Remove access rights to the object

Principal. The repository username of the user or group who has access rights on this object.

Permissions. The access rights that this user or group has for the object.

Add. Enables you to add one or more users or groups to the list of those with access rights on this object. See the topic "Adding a User to the Permissions List" on page 199 for more information.

Modify. Enables you to modify the access rights of the selected user or group for this object. Read access is granted by default. This option enables you to grant additional access rights, namely Owner, Write, Delete, and Modify Permissions.

Delete. Deletes the selected user or group from the permissions list for this object.

Adding a User to the Permissions List

The following fields are available when you select **Add** on the Security tab of the Repository: Store dialog box:

Select provider. Choose a security provider for authentication. The repository can be configured to use different security providers; if necessary, contact your local administrator for more information.

Find. Enter the repository username of the user or group you want to add, and click **Search** to display that name in the user list. To add more than one username at a time, leave this field blank and just click **Search** to display a list of all the repository usernames.

User list. Select one or more usernames from the list and click OK to add them to the permissions list.

Modifying Access Rights for an Object

The following fields are available when you select **Modify** on the Security tab of the Repository: Store dialog box:

Owner. Select this option to give this user or group owner access rights to the object. The owner has full control over the object, including Delete and Modify access rights.

Read. By default, a user or group that is not the object owner has only Read access rights to the object. Select the appropriate check boxes to add Write, Delete, and Modify Permissions access rights for this user or group.

Storing Streams

You can store a stream as a *.str* file in the repository, from where it can be accessed by other users.

Note: For information on deploying a stream, to take advantage of additional repository features, see "Deploying streams" on page 206.

To store the current stream:

1. On the main menu, click:

   File > **Store** > **Store as Stream...**

2. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

3. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Storing Projects

You can store a complete IBM SPSS Modeler project as a *.cpj* file in the repository so that it can be accessed by other users.

Because a project file is a container for other IBM SPSS Modeler objects, you need to tell IBM SPSS Modeler to store the project's objects in the repository. You do this using a setting in the Project Properties dialog box. See the topic "Setting Project Properties" on page 219 for more information.

Once you configure a project to store objects in the repository, whenever you add a new object to the project, IBM SPSS Modeler automatically prompts you to store the object.

When you have finished your IBM SPSS Modeler session, you must store a new version of the project file so that it remembers your additions. The project file automatically contains (and retrieves) the latest versions of its objects. If you did not add any objects to a project during an IBM SPSS Modeler session, then you do not have to re-store the project file. You must, however, store new versions for the project objects (streams, output, and so forth) that you changed.

To store a project

1. Select the project on the CRISP-DM or Classes tab in the managers pane in IBM SPSS Modeler, and on the main menu click:

   File > **Project** > **Store Project...**

2. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

3. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Storing Nodes

You can store an individual node definition from the current stream as a *.nod* file in the repository, from where it can be accessed by other users.

To store a node:

1. Right-click the node in the stream canvas and click **Store Node**.

2. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

3. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Storing Output Objects

You can store an output object from the current stream as a *.cou* file in the repository, from where it can be accessed by other users.

To store an output object:

1. Click the object on the Outputs tab of the managers pane in SPSS Modeler, and on the main menu click:

   File > **Outputs** > **Store Output...**

2. Alternatively, right-click an object in the Outputs tab and click **Store**.

3. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

4. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Storing Models and Model Palettes

You can store an individual model as a *.gm* file in the repository, from where it can be accessed by other users. You can also store the complete contents of the Models palette as a *.gen* file in the repository.

Storing a model:

1. Click the object on the Models palette in SPSS Modeler, and on the main menu click:

   File > **Models** > **Store Model...**

2. Alternatively, right-click an object in the Models palette and click **Store Model**.

3. Continue from "Completing the storage procedure".

Storing a Models palette:

1. Right-click the background of the Models palette.

2. On the pop-up menu, click **Store Palette**.

3. Continue from "Completing the storage procedure".

Completing the storage procedure:

1. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

2. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Retrieving Objects from the Repository

You can retrieve streams, models, model palettes, nodes, projects, and output objects that have been stored in the repository.

Note: Besides using the menu options as described here, you can also retrieve streams, output objects, models and model palettes by right-clicking in the appropriate tab of the managers pane at the top right of the SPSS Modeler window.

1. To retrieve a stream, on the IBM SPSS Modeler main menu click:

    File > **Retrieve Stream...**

2. To retrieve a model, model palette, project, or output object, on the IBM SPSS Modeler main menu click:

    File > **Models** > **Retrieve Model...**

    or

    File > **Models** > **Retrieve Models Palette...**

    or

    File > **Projects** > **Retrieve Project...**

    or

    File > **Outputs** > **Retrieve Output...**

3. Alternatively, right-click in the managers or project pane and click **Retrieve** on the pop-up menu.

4. To retrieve a node, on the IBM SPSS Modeler main menu click:

    Insert > **Node (or SuperNode) from Repository...**

    a. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

5. In the Repository: Retrieve dialog box, browse to the object, select it and click the **Retrieve** button. See the topic for more information.

Choosing an Object to Retrieve

The following fields are available in the Repository: Retrieve/Search dialog box:

Look in. Shows the folder hierarchy for the current folder. To navigate to a different folder, select one from this list to navigate there directly, or navigate using the object list below this field.

Up Folder button. Navigates to one level above the current folder in the hierarchy.

New Folder button. Creates a new folder at the current level in the hierarchy.

File name. The repository file name of the selected object. To retrieve that object, click **Retrieve**.

Files of type. The type of object that you have chosen to retrieve. Only objects of this type, together with folders, are shown in the object list. To display objects of a different type for retrieval, select the object type from the list.

Open as locked. By default, when an object is retrieved, it is locked in the repository so that others cannot update it. If you do not want the object to be locked on retrieval, uncheck this box.

Description, Keywords. If additional details about the object were defined when the object was stored, those details are displayed here. See the topic "Adding Information About Stored Objects" on page 197 for more information.

Version. To retrieve a version of the object other than the latest, click this button. Detailed information for all versions is displayed, allowing you to choose the version you want.

Selecting an Object Version

To select a specific version of a repository object, in the Repository: Select Version dialog box:

1. (Optional) Sort the list by version, label, size, creation date or creating user, by double-clicking on the header of the appropriate column.
2. Select the object version you want to work with.
3. Click Continue.

Searching for objects in the repository

You can search for objects by name, folder, type, label, date, or other criteria.

Searching for objects by name

1. On the IBM SPSS Modeler main menu click:

   Tools > **Repository** > **Explore...**

   a. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.
2. Click the **Search** tab.
3. In the **Search for objects named** field, specify the name of the object you want to find.

When searching for objects by name, an asterisk (*) can be used as a wildcard character to match any string of characters, and a question mark (?) matches any single character. For example, *cluster* matches all objects that include the string cluster anywhere in the name. The search string m0?_* matches M01_cluster.str and M02_cluster.str but not M01a_cluster.str. Searches are not case sensitive (cluster matches Cluster matches CLUSTER).

Note: If the number of objects is large, searches may take a few moments.

Searching by other criteria

You can perform a search based on title, label, dates, author, keywords, indexed content, or description. Only objects that match *all* specified search criteria will be found. For example, you could locate all streams containing one or more clustering models that also have a specific label applied, and which were modified after a specific date.

Object Types. You can restrict the search to models, streams, outputs, nodes, SuperNodes, projects, model palettes, or other types of objects.

- **Models.** You can search for models by category (classification, approximation, clustering, etc.) or by a specific modeling algorithm, such as Kohonen.

  You can also search by fields used—for example, all models that use a field named *income* as an input or output (target) field.

- **Streams.** For streams, you can restrict the search by fields used, or model type (either category or algorithm) contained in the stream.

Topics. You can search on models associated with specific topics from a list set by repository users with the appropriate privileges (for more information, see the *Deployment Manager User's Guide*). To obtain the list, check this box, then click the Add Topics button that is displayed, select one or more topics from the list and click OK.

Label. Restricts the search to specific object version labels.

Dates. You can specify a creation or modification date and search on objects before, after, or between the specified date range.

Author. Restricts the search to objects created by a specific user.

Keywords. Search on specific keywords. In IBM SPSS Modeler, keywords are specified on the Annotation tab for a stream, model, or output object.

Description. Search on specific terms in the description field. In IBM SPSS Modeler, the description is specified on the Annotation tab for a stream, model, or output object. Multiple search phrases can be separated by semicolons—for example, `income; crop type; claim value`. (Note that within a search phrase, spaces matter. For example, `crop type` with one space and `crop  type` with two spaces are not the same.)

Modifying Repository Objects

You can modify existing objects in the repository directly from SPSS Modeler. You can:

- Create, rename, or delete folders
- Lock or unlock objects
- Delete objects

Creating, Renaming, and Deleting Folders

1. To perform operations on folders in the repository, on the SPSS Modeler main menu click:

   Tools > **Repository** > **Explore...**

   a. Specify connection settings to the repository if necessary. See the topic for more information. For specific port, password, and other connection details, contact your local system administrator.

2. Ensure that the **Folders** tab is active.
3. To create a new folder, right-click the parent folder and click **New Folder**.
4. To rename a folder, right-click it and click **Rename Folder**.
5. To delete a folder, right-click it and click **Delete Folder**.

Locking and Unlocking Repository Objects

You can lock an object to prevent other users from updating any of its existing versions or creating new versions. A locked object is indicated by a padlock symbol over the object icon.

Figure 17. Locked object

To lock an object

1. In the repository explorer window, right-click the required object.
2. Click **Lock**.

To unlock an object

1. In the repository explorer window, right-click the required object.
2. Click **Unlock**.

Deleting Repository Objects

Before deleting an object from the repository, you must decide if you want to delete all versions of the object, or just a particular version.

To Delete All Versions of an Object

1. In the repository explorer window, right-click the required object.
2. Click **Delete Objects**.

To Delete the Most Recent Version of an Object

1. In the repository explorer window, right-click the required object.
2. Click **Delete**.

To Delete a Previous Version of an Object

1. In the repository explorer window, right-click the required object.
2. Click **Delete Versions**.
3. Select the version(s) to delete and click **OK**.

Managing Properties of Repository Objects

You can control various object properties from SPSS Modeler. You can:

- View the properties of a folder
- View and edit the properties of an object
- Create, apply and delete version labels for an object

Viewing Folder Properties

To view properties for any folder in the repository window, right-click the required folder. Click **Folder Properties**.

General tab

This tab displays the folder name, creation, and modification dates.

Permissions tab

In this tab you specify read and write permissions for the folder. All users and groups with access to the parent folder are listed. Permissions follow a hierarchy. For example, if you do not have read permission, you cannot have write permission. If you do not have write permission, you cannot have delete permission.

Users And Groups. Lists the repository users and groups that have at least Read access to this folder. Select the Write and Delete check boxes to add those access rights for this folder to a particular user or group. Click the **Add Users/Groups** icon on the right side of the Permissions tab to assign access to additional users and groups. The list of available users and groups is controlled by the administrator.

Cascade Permissions. Choose an option to control how changes made to the current folder are applied to its child folders, if any.

- **Cascade all permissions.** Cascades permission settings from the current folder to all child and descendant folders. This is a quick way to set permissions for several folders at once. Set permissions as required for the parent folder, and then cascade as required.
- **Cascade changes only.** Cascades only changes made since the last time changes were applied. For example, if a new group has been added and you want to give it access to all folders under the Sales

branch, you can give the group access to the root Sales folder and cascade the change to all subfolders. All other permissions to existing subfolders remain as before.

- **Do not cascade.** Any changes made apply to the current folder only and do not cascade to child folders.

Viewing and Editing Object Properties

In the Object Properties dialog box you can view and edit properties. Although some properties cannot be changed, you can always update an object by adding a new version.

1. In the repository window, right-click the required object.
2. Click **Object Properties**.

General Tab

Name. The name of the object as viewed in the repository.

Created on. Date the object (not the version) was created.

Last modified. Date the most recent version was modified.

Author. The user's login name.

Description. By default, this contains the description specified on the object's Annotation tab in SPSS Modeler.

Linked topics. The repository allows models and related objects to be organized by topics if required. The list of available topics is set by repository users with the appropriate privileges (for more information, see the *Deployment Manager User's Guide*).

Keywords. You specify keywords on the Annotation tab for a stream, model, or output object. Multiple keywords should be separated by spaces, up to a maximum of 255 characters. (If keywords contain spaces, use quotation marks to separate them.)

Versions Tab

Objects stored in the repository may have multiple versions. The Versions tab displays information about each version.

The following properties can be specified or modified for specific versions of a stored object:

Version. Unique identifier for the version generated based on the time when the version was stored.

Label. Current label for the version, if any. Unlike the version identifier, labels can be moved from one version of an object to another.

The file size, creation date, and author are also displayed for each version.

Edit Labels. Click the **Edit Labels** icon at the top right of the Versions tab to define, apply or remove labels for stored objects. See the topic "Managing Object Version Labels" on page 205 for more information.

Permissions Tab

On the Permissions tab you can set read and write permissions for the object. All users and groups with access to the current object are listed. Permissions follow a hierarchy. For example, if you do not have read permission, you cannot have write permission. If you do not have write permission, you cannot have delete permission.

Users And Groups. Lists the repository users and groups that have at least Read access to this object. Select the Write and Delete check boxes to add those access rights for this object to a particular user or group. Click the **Add Users/Groups** icon on the right side of the Permissions tab to assign access to additional users and groups. The list of available users and groups is controlled by the administrator.

Managing Object Version Labels

The Edit Version Labels dialog box enables you to:

- Apply labels to the selected object
- Remove labels from the selected object
- Define a new label and apply it to the object

To apply labels to the object

1. Select one or more labels in the **Available Labels** list.
2. Click the right-arrow button to move the selected labels to the **Applied Labels** list.
3. Click **OK**.

To remove labels from the object

1. Select one or more labels in the **Applied Labels** list.
2. Click the left-arrow button to move the selected labels to the **Available Labels** list.
3. Click **OK**.

To define a new label and apply it to the object

1. Type the label name in the **New Label** field.
2. Click the right-arrow button to move the new label to the **Applied Labels** list.
3. Click **OK**.

Deploying streams

To enable a stream to be used with the thin-client application IBM SPSS Modeler Advantage, it must be deployed as a stream (`.str` file) in the repository.

Note: You cannot deploy a stream that has more than one source node in the scoring branch.

The stream can take full advantage of the enterprise-level features of IBM SPSS Collaboration and Deployment Services. See the topic "Storing and deploying repository objects" on page 195 for more information.

To deploy the current stream (File menu method)

1. On the main menu, click:

   File > **Store** > **Deploy**
2. Choose the deployment type and complete the rest of the dialog box as necessary.
3. Click **Deploy as stream** to deploy the stream for use with IBM SPSS Modeler Advantage or IBM SPSS Collaboration and Deployment Services.
4. Click **Store**. For more information, click **Help**.
5. Continue from "Completing the deployment process."

To deploy the current stream (Tools menu method)

1. On the main menu, click:

   Tools > **Stream Properties** > **Deployment**
2. Choose the deployment type, complete the rest of the Deployment tab as necessary, and click **Store**. See the topic "Stream Deployment Options" on page 207 for more information.

Completing the deployment process

1. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

2. In the Repository: Store dialog box, choose the folder where you want to store the object, specify any other information you want to record, and click **Store**. See the topic "Setting Object Properties" on page 197 for more information.

Stream Deployment Options

The Deployment tab in the Stream Options dialog box allows you to specify options for deploying the stream.

When you deploy a stream, it's stored in the repository as a file with the extension `.str`.

Deploying a stream allows you to take advantage of the additional functionality available with IBM SPSS Collaboration and Deployment Services, such as multi-user access, automated scoring, model refresh, and Champion Challenger analysis.

Note: The Association Rules, STP, and TCM modeling nodes do not support the Model Evaluation or Champion Challenger steps in IBM SPSS Collaboration and Deployment Services.

From the Deployment tab, you can also preview the stream description that IBM SPSS Modeler creates for the stream. See the topic "Stream descriptions" on page 47 for more information.

Deployment type. Choose how you want to deploy the stream. All streams require a designated scoring node before they can be deployed; additional requirements and options depend on the deployment type.

Note: The Association Rules, STP, and TCM modeling nodes do not support the Model Evaluation or Champion Challenger steps in IBM SPSS Collaboration and Deployment Services.

- **<none>**. The stream will not be deployed to the repository. All options are disabled except stream description preview.
- **Scoring Only**. The stream is deployed to the repository when you click the **Store** button. Data can be scored using the node that you designate in the **Scoring node** field.
- **Model Refresh**. Same as for Scoring Only but in addition, the model can be updated in the repository using the objects that you designate in the **Modeling node** and **Model nugget** fields. Note that automatic model refresh is not supported by default in IBM SPSS Collaboration and Deployment Services, so you must choose this deployment type if you want to use this feature when running a stream from the repository. See "Model Refresh" on page 209 for more information.

Scoring node. Select a graph, output or export node to identify the stream branch to be used for scoring the data. While the stream can actually contain any number of valid branches, models, and terminal nodes, one and only one scoring branch must be designated for purposes of deployment. This is the most basic requirement to deploy any stream.

Scoring Parameters. Allows you to specify parameters that can be modified when the scoring branch is run. See "Scoring and modeling parameters" on page 208 for more information.

Modeling node. For model refresh, specifies the modeling node used to regenerate or update the model in the repository. Must be a modeling node of the same type as that specified for **Model nugget**.

Model Build Parameters. Allows you to specify parameters that can be modified when the modeling node is run. See "Scoring and modeling parameters" on page 208 for more information.

Model nugget. For model refresh, specifies the model nugget that will be updated or regenerated each time the stream is updated in the repository (typically as part of a scheduled job). The model must be located on the scoring branch. While multiple models may exist on the scoring branch, only one can be designated. Note that when the stream is initially created this may effectively be a placeholder model that is updated or regenerated as new data is available.

Check. Click this button to check whether this is a valid stream for deployment. All streams must have a designated scoring node before they can be deployed. Error messages are displayed if these conditions are not satisfied.

Store. Deploys the stream if it is valid. If not, an error message is displayed. Click the **Fix** button, correct the error, and try again.

Preview Stream Description. Enables you to view the contents of the stream description that IBM SPSS Modeler creates for the stream. See "Stream descriptions" on page 47 for more information.

Note: The Association Rules, STP, and TCM modeling nodes do not support the Model Evaluation or Champion Challenger steps in IBM SPSS Collaboration and Deployment Services.

Scoring and modeling parameters

When deploying a stream to IBM SPSS Collaboration and Deployment Services, you can choose which parameters can be viewed or edited each time the model is updated or scored. For example, you might specify maximum and minimum values, or some other value that may be subject to change each time a job is run.

1. To make a parameter visible so it can be viewed or edited after the stream is deployed, select it from the list in the Scoring Parameters dialog box.

The list of available parameters is defined on the Parameters tab in the stream properties dialog box. See the topic "Setting Stream and Session Parameters" on page 44 for more information.

The Scoring Branch

If you are deploying a stream, one branch of the stream must be designated as the **scoring branch** (that is, the one containing the scoring node). When you designate a branch as the scoring branch, that branch is highlighted on the stream canvas, as is the model link to the nugget on the scoring branch. This visual representation is particularly useful in complex streams with multiple branches, where the scoring branch might not be immediately obvious.

Note: Only one stream branch can be designated as the scoring branch.

If the stream already had a scoring branch defined, the newly-designated branch replaces it as the scoring branch. You can set the color of the scoring branch indication by means of a Custom Color option. See the topic "Setting display options" on page 228 for more information.

You can show or hide the scoring branch indication by means of the Show/hide stream markup toolbar button.

Figure 18. Show/hide stream markup toolbar button

Identifying the Scoring Branch for Deployment

You can designate the scoring branch either from the pop-up menu of a terminal node, or from the Tools menu. If you use the pop-up menu, the scoring node is set automatically in the Deployment tab of the stream properties.

To designate a branch as the scoring branch (pop-up menu)

1. Connect the model nugget to a terminal node (a processing or output node downstream from the nugget).

2. Right-click the terminal node.

3. On the menu, click **Use as Scoring Branch**.

To designate a branch as the scoring branch (Tools menu)

1. Connect the model nugget to a terminal node (a processing or output node downstream from the nugget).

2. On the main menu, click:

   Tools > **Stream Properties** > **Deployment**

3. On the **Deployment type** list, click **Scoring Only** or **Model Refresh** as required. See the topic "Stream Deployment Options" on page 207 for more information.

4. Click the **Scoring node** field and select a terminal node from the list.

5. Click **OK**.

Model Refresh

Model refresh is the process of rebuilding an existing model in a stream using newer data. The stream itself does not change in the repository. For example, the algorithm type and stream-specific settings remain the same, but the model is retrained on new data, and updated if the new version of the model works better than the old one.

Only one model nugget in a stream can be set to refresh--this is known as the **refresh model**. If you click the **Model Refresh** option on the Deployment tab of the stream properties (see "Stream Deployment Options" on page 207), the model nugget that you designate at that time becomes the refresh model. You can also designate a model as the refresh model from the pop-up menu of a model nugget. The nugget must already be on the scoring branch for this to be possible.

If you turn off the "refresh model" status of a nugget, this is equivalent to setting the deployment type of the stream to Scoring Only, and the Deployment tab of the stream properties dialog box is updated accordingly. You can turn this status on and off by means of the **Use as Refresh Model** option on the pop-up menu of the nugget on the current scoring branch.

Removing the model link of a nugget on the scoring branch also removes the "refresh model" status of the nugget. You can undo removal of the model link by means of the Edit menu or the toolbar; doing so also reinstates the "refresh model" status of the nugget.

How the Refresh Model is Selected

As well as the scoring branch, the link to the refresh model is also highlighted in the stream. The model nugget chosen as the refresh model, and therefore the link that is highlighted, depends on how many nuggets are in the stream.

Single Model in Stream

If a single linked model nugget is on the scoring branch when it is identified as such, that nugget becomes the refresh model for the stream.

Multiple Models in Stream

If there is more than one linked nugget in the stream, the refresh model is chosen as follows.

If a model nugget has been defined in the Deployment tab of the stream properties dialog box and is also in the stream, then that nugget becomes the refresh model.

If no nugget has been defined in the Deployment tab, or if one has been defined but is not on the scoring branch, then the nugget closest to the terminal node becomes the refresh model.

If you subsequently deselect all model links as refresh links, only the scoring branch is highlighted, not the links. The deployment type is set to Scoring Only.

Note: You can choose to set one of the links to Replace status, but not the other one. In this case, the model nugget chosen as the refresh model is the one that has a refresh link and which is closest to the terminal node when the scoring branch is designated.

No Models in Stream

If there are no models in the stream, or only models with no model links, the deployment type is set to Scoring Only.

Checking a scoring branch for errors

When you designate the scoring branch, it is checked for errors.

If an error is found, the scoring branch is highlighted in the scoring branch error color, and an error message is displayed. You can set the error color by means of a Custom Color option. See the topic "Setting display options" on page 228 for more information.

If an error is found, proceed as follows:

1. Correct the error according to the contents of the error message.
2. On the main menu, click:

   Tools > **Stream Properties** > **Deployment**

   and click **Check**.
3. If necessary, repeat this process until no errors are found.

Chapter 13. Saving streams to IBM Cloud Pak for Data

You can save streams to an IBM Cloud Pak for Data server where you can open them as flows and take advantage of the many features available in Cloud Pak for Data.

Cloud Pak for Data simplifies how you collect, organize, and analyze data and infuse AI across your business. Seamlessly integrate a broad range of data and AI software, leading governance and security, and a unified experience for collaboration. For more information, see https://www.ibm.com/products/cloud-pak-for-data. Contact your IBM representative if you're interested in Cloud Pak for Data.

To configure the server connection

These instructions are based on Firefox. Steps may vary based on which web browser you're using.

1. Open Firefox, paste your Cloud Pak for Data URL in the address bar, and press Enter.

   You can obtain the URL by opening the Cloud Pak for Data web client and looking at the URL in your browser address bar (it's everything up to and not including /zen). By default, the URL takes the form `https://<namespace>-cpd-<namespace>.apps.<cluster-subdomain>`

   where:

   - `<namespace>` is your namespace (such as ab123).
   - `<cluster-subdomain>` is the subdomain name of your cluster (such as `ocp454x86-aqt-mycompany.com`)

   In this example, the URL is `https://ab123-cpd-ab123.apps.ocp454x86-aqt-mycompany.com`

   For more information about the Cloud Pak for Data URL, see the Cloud Pak for Data documentation.

2. Click the yellow padlock icon to the left of the address bar, then click the arrow beside **Connection not secure**.

3. Click **More Information**. The Page Info dialog opens.

4. In the Page Info dialog, go to **Security** and click **View Certificate**.

5. Scroll down to **Miscellaneous** and download the PEM certification file.

6. From the command line, import the certification file to your IBM SPSS Modeler client directory as follows:

   On Windows:

   ```
   "%JAVA_HOME%"\bin\keytool.exe -importcert -trustcacerts -file <Certification file absolute
   path> -alias <alias> -keystore "<Modeler client installation path>"\jre\lib\security\cacerts
   ```

   On macOS:

   ```
   "%JAVA_HOME%"/bin/keytool -importcert -trustcacerts -file <Certification file absolute path>
   -alias <alias> -keystore "<Modeler client installation path>"/jre/lib/security/cacerts
   ```

   When prompted, enter `changeit` for the keystore password and enter `yes` to trust the certification.

7. On the main menu in IBM SPSS Modeler, click:

   Tools > **IBM Cloud Pak for Data Server** > **IBM Cloud Pak for Data Server Connections...**

8. Specify connection settings to the server. Use the same URL you used in step 1. If you're not sure which username or password to use, contact your local system administrator. If you select the API Key login method, use the following method to obtain the URL:

- Log on to the Cloud Pak for Data cluster and run the command `oc get routes -n ibm-common-services`, then copy the HOST/PORT value for `cp-console`. In the following example, the URL is `https://cp-console.apps.wml46x04.cp.myserver.com`:

```
[root@wml46x04-inf ocp-scripts]# oc get routes -n ibm-common-services
NAME          HOST/PORT                                    PATH   SERVICES
PORT    TERMINATION             WILDCARD
cp-console    cp-console.apps.wml46x04.cp.myserver.com            icp-management-ingress
https   reencrypt/Redirect      None
cp-proxy      cp-proxy.apps.wml46x04.cp.myserver.com             nginx-ingress-controller
https   passthrough/Redirect    None
```

9. Click **Connect** to connect to the server.

To save streams to Cloud Pak for Data

1. On the main menu, click:

   File > **Store as Flow...**. *Streams* are referred to as *flows* after they're migrated to Cloud Pak for Data.

   Alternatively, you can also right-click in the streams manager pane and select **Store as Flow...**

2. Choose the Cloud Pak for Data Server you configured previously. Then select the project to which you want to save the flow.

Tip: You can also select multiple streams to save at one time. On the main menu, click **Tools** > **IBM Cloud Pak for Data Server...** > **Bulk uploading streams to IBM Cloud Pak for Data Server**. You'll then be prompted to select streams from your local file system, select a server, and select a project.

Chapter 14. Exporting to external applications

About Exporting to External Applications

IBM SPSS Modeler provides a number of mechanisms to export the entire data mining process to external applications, so that the work you do to prepare data and build models can be used to your advantage outside of IBM SPSS Modeler as well.

The previous section showed how you can deploy streams to an IBM SPSS Collaboration and Deployment Services repository to take advantage of its multi-user access, job scheduling and other features. In a similar way, IBM SPSS Modeler streams can also be used in conjunction with:

• IBM SPSS Modeler Advantage

• Applications that can import and export files in PMML format

For more information about using streams with IBM SPSS Modeler Advantage, see "Opening a Stream in IBM SPSS Modeler Advantage " on page 213.

For information on exporting and importing models as PMML files, making it possible to share models with any other applications that support this format, see "Importing and exporting models as PMML" on page 213.

Opening a Stream in IBM SPSS Modeler Advantage

IBM SPSS Modeler streams can be used in conjunction with the thin-client application IBM SPSS Modeler Advantage. While it is possible to create customized applications entirely within IBM SPSS Modeler Advantage, you can also use a stream already created in IBM SPSS Modeler as the basis of an application workflow.

To open a stream in IBM SPSS Modeler Advantage:

1. Deploy the stream in the IBM SPSS Collaboration and Deployment Services repository, being sure to click the **Deploy as stream** option. See the topic "Deploying streams" on page 206 for more information.

2. Click the Open in IBM SPSS Modeler Advantage toolbar button, or from the main menu click:

File > **Open in IBM SPSS Modeler Advantage**

1. Specify connection settings to the repository if necessary. See the topic "Connecting to the Repository" on page 196 for more information. For specific port, password, and other connection details, contact your local system administrator.

Note: The repository server must also have the IBM SPSS Modeler Advantage software installed.

1. In the Repository: Store dialog, choose the folder where you want to store the object, specify any other information you want to record, and click the **Store** button. See the topic "Setting Object Properties" on page 197 for more information.

Doing so launches IBM SPSS Modeler Advantage with the stream already open. The stream is closed in IBM SPSS Modeler.

Importing and exporting models as PMML

PMML, or predictive model markup language, is an XML format for describing data mining and statistical models, including inputs to the models, transformations used to prepare data for data mining, and the parameters that define the models themselves. IBM SPSS Modeler can import and export PMML, making it possible to share models with other applications that support this format, such as IBM SPSS Statistics.

For more information about PMML, see the Data Mining Group website (*http://www.dmg.org*).

To export a model

PMML export is supported for most of the model types generated in IBM SPSS Modeler. See the topic "Model types supporting PMML" on page 214 for more information.

1. Right-click a model nugget on the models palette. (Alternatively, double-click a model nugget on the canvas and select the File menu.)
2. On the menu, click **Export PMML**.
3. In the Export (or Save) dialog box, specify a target directory and a unique name for the model.

Note:

You can change options for PMML export in the User Options dialog box. On the main menu, click:

Tools > **Options** > **User Options**

and click the PMML tab.

See the topic "Setting PMML Export Options" on page 229 for more information.

To import a model saved as PMML

Models exported as PMML from IBM SPSS Modeler or another application can be imported into the models palette. See the topic "Model types supporting PMML" on page 214 for more information.

1. In the models palette, right-click the palette and select **Import PMML** from the menu.
2. Select the file to import and specify options for variable labels as required.
3. Click **Open**.

Use variable labels if present in model. The PMML may specify both variable names and variable labels (such as Referrer ID for *RefID*) for variables in the data dictionary. Select this option to use variable labels if they are present in the originally exported PMML.

If you have selected the variable label option but there are no variable labels in the PMML, the variable names are used as normal.

Model types supporting PMML

PMML Export

IBM SPSS Modeler models. The following models created in IBM SPSS Modeler can be exported as PMML 4.3:

- C&R Tree
- QUEST
- CHAID
- Neural Net
- C5.0
- Logistic Regression
- Genlin
- SVM
- Apriori
- Carma
- K-Means
- Kohonen
- TwoStep
- TwoStep-AS

- GLMM (PMML is exported for all GLMM models, but the PMML has only fixed effects)
- Decision List
- Cox
- Sequence (scoring for Sequence PMML models is not supported)
- Random Trees
- Tree-AS
- Linear
- Linear-AS
- Regression
- Logistic
- GLE
- LSVM
- KNN
- Association Rules

Database native models. For models generated using database-native algorithms, PMML export is not available. Models created using Analysis Services from Microsoft or Oracle Data Miner cannot be exported.

PMML Import

IBM SPSS Modeler can import and score PMML models generated by current versions of all IBM SPSS Statistics products, including models exported from IBM SPSS Modeler as well as model or transformation PMML generated by IBM SPSS Statistics 17.0 or later. Essentially, this means any PMML that the scoring engine can score, with the following exceptions:

- Apriori, CARMA, Anomaly Detection, Sequence, and Association Rules models cannot be imported.
- PMML models may not be browsed after importing into IBM SPSS Modeler even though they can be used in scoring. (Note that this includes models that were exported from IBM SPSS Modeler to begin with. To avoid this limitation, export the model as a generated model file [*.gm] rather than PMML.)
- Limited validation occurs on import, but full validation is performed on attempting to score the model. Thus it is possible for import to succeed but scoring to fail or produce incorrect results.

Note: For third party PMML imported into IBM SPSS Modeler, IBM SPSS Modeler will attempt to score valid PMML that can be recognized and scored. But it is not guaranteed that all PMML will score or that it will score in the same way as the application that generated it.

Chapter 15. Projects and reports

Introduction to Projects

A **project** is a group of files related to a data mining task. Projects include data streams, graphs, generated models, reports, and anything else that you have created in IBM SPSS Modeler. At first glance, it may seem that IBM SPSS Modeler projects are simply a way to organize output, but they are actually capable of much more. Using projects, you can:

- Annotate each object in the project file.
- Use the CRISP-DM methodology to guide your data mining efforts. Projects also contain a CRISP-DM Help system that provides details and real-world examples on data mining with CRISP-DM.
- Add non-IBM SPSS Modeler objects to the project, such as a PowerPoint slide show used to present your data mining goals or white papers on the algorithms that you plan to use.
- Produce both comprehensive and simple update reports based on your annotations. These reports can be generated in HTML for easy publishing on your organization's intranet.

Note: If the project pane is not visible in the IBM SPSS Modeler window, click **Project** on the View menu.

Objects that you add to a project can be viewed in two ways: **Classes view** and **CRISP-DM view**. Anything that you add to a project is added to both views, and you can toggle between views to create the organization that works best.

CRISP-DM View

By supporting the Cross-Industry Standard Process for Data Mining (CRISP-DM), IBM SPSS Modeler projects provide an industry-proven and non-proprietary way of organizing the pieces of your data mining efforts. CRISP-DM uses six phases to describe the process from start (gathering business requirements) to finish (deploying your results). Even though some phases do not typically involve work in IBM SPSS Modeler, the project pane includes all six phases so that you have a central location for storing and tracking all materials associated with the project. For example, the Business Understanding phase typically involves gathering requirements and meeting with colleagues to determine goals rather than working with data in IBM SPSS Modeler. The project pane allows you to store your notes from such meetings in the *Business Understanding* folder for future reference and inclusion in reports.

The CRISP-DM view in the project pane is also equipped with its own Help system to guide you through the data mining life cycle. From IBM SPSS Modeler, this help can be accessed by clicking **CRISP-DM Help** on the Help menu.

Note: If the project pane is not visible in the window, click **Project** on the View menu.

Setting the Default Project Phase

Objects added to a project are added to a default phase of CRISP-DM. This means that you need to organize objects manually according to the data mining phase in which you used them. It is wise to set the default folder to the phase in which you are currently working.

To select which phase to use as your default:

1. In CRISP-DM view, right-click the folder for the phase to set as the default.
2. On the menu, click **Set as Default**.

The default folder is displayed in bold type.

Classes View

The Classes view in the project pane organizes your work in IBM SPSS Modeler categorically by the types of objects created. Saved objects can be added to any of the following categories:

- Streams
- Nodes
- Models
- Tables, graphs, reports
- Other (non-IBM SPSS Modeler files, such as slide shows or white papers relevant to your data mining work)

Adding objects to the Classes view also adds them to the default phase folder in the CRISP-DM view.

Note: If the project pane is not visible in the window, click **Project** on the View menu.

Building a Project

A project is essentially a file containing references to all of the files that you associate with the project. This means that project items are saved both individually and as a reference in the project file (*.cpj*). Because of this referential structure, note the following:

- Project items must first be saved individually before being added to a project. If an item is unsaved, you will be prompted to save it before adding it to the current project.
- Objects that are updated individually, such as streams, are also updated in the project file.
- Manually moving or deleting objects (such as streams, nodes, and output objects) from the file system will render links in the project file invalid.

Creating a New Project

New projects are easy to create in the IBM SPSS Modeler window. You can either start building one, if none is open, or you can close an existing project and start from scratch.

On the main menu, click:

File > **Project** > **New Project...**

Adding to a Project

Once you have created or opened a project, you can add objects, such as data streams, nodes, and reports, using several methods.

Adding Objects from the Managers

Using the managers in the upper right corner of the IBM SPSS Modeler window, you can add streams or output.

1. Select an object, such as a table or a stream, from one of the manager tabs.
2. Right-click, and click **Add to Project**.

   If the object has been previously saved, it will automatically be added to the appropriate objects folder (in Classes view) or to the default phase folder (in CRISP-DM view).

3. Alternatively, you can drag and drop objects from the managers to the project pane.

Note: You may be asked to save the object first. When saving, be sure that **Add file to project** is selected in the Save dialog box. This will automatically add the object to the project after you save it.

Adding Nodes from the Canvas

You can add individual nodes from the stream canvas by using the Save dialog box.

1. Select a node on the canvas.

2. Right-click, and click **Save Node**. Alternatively, on the main menu click:

    Edit > **Node** > **Save Node...**

3. In the Save dialog box, select **Add file to project**.
4. Create a name for the node and click **Save**.

This saves the file and adds it to the project. Nodes are added to the *Nodes* folder in Classes view and to the default phase folder in CRISP-DM view.

Adding External Files

You can add a wide variety of non-IBM SPSS Modeler objects to a project. This is useful when you are managing the entire data mining process within IBM SPSS Modeler. For example, you can store links to data, notes, presentations, and graphics in a project. In CRISP-DM view, external files can be added to the folder of your choice. In Classes view, external files can be saved only to the *Other* folder.

To add external files to a project:

1. Drag files from the desktop to the project.

    or

2. Right-click the target folder in CRISP-DM or Classes view.
3. On the menu, click **Add to Folder**.
4. Select a file in the dialog box and click **Open**.

This will add a reference to the selected object inside IBM SPSS Modeler projects.

Transferring Projects to the IBM SPSS Collaboration and Deployment Services Repository

You can transfer an entire project, including all component files, to the IBM SPSS Collaboration and Deployment Services Repository in one step. Any objects that are already in the target location will not be moved. This feature also works in reverse: you can transfer entire projects from the IBM SPSS Collaboration and Deployment Services Repository to your local file system.

Transferring a Project

Make sure that the project you want to transfer is open in the project pane.

To transfer a project:

1. Right-click the root project folder and click **Transfer Project**.
2. If prompted, log in to IBM SPSS Collaboration and Deployment Services Repository.
3. Specify the new location for the project and click **OK**.

Setting Project Properties

You can customize a project's contents and documentation by using the project properties dialog box. To access project properties:

1. Right-click an object or folder in the project pane and click **Project Properties**.
2. Click the **Project** tab to specify basic project information.

Created. Shows the project's creation date (not editable).

Summary. You can enter a summary for your data mining project that will be displayed in the project report.

Contents. Lists the type and number of components referenced by the project file (not editable).

Save unsaved object as. Specifies whether unsaved objects should be saved to the local file system, or stored in the repository. See the topic "About the IBM SPSS Collaboration and Deployment Services Repository " on page 195 for more information.

Update object references when loading project. Select this option to update the project's references to its components. *Note*: The files added to a project are not saved in the project file itself. Rather, a reference to the files is stored in the project. This means that moving or deleting a file will remove that object from the project.

Annotating a Project

The project pane provides a number of ways to annotate your data mining efforts. Project-level annotations are often used to track "big-picture" goals and decisions, while folder or node annotations provide additional detail. The Annotations tab provides enough space for you to document project-level details, such as the exclusion of data with irretrievable missing data or promising hypotheses formed during data exploration.

To annotate a project:

1. Select the project folder in either CRISP-DM or Classes view.
2. Right-click the folder and click **Project Properties**.
3. Click the **Annotations** tab.
4. Enter keywords and text to describe the project.

Folder Properties and Annotations

Individual project folders (in both CRISP-DM and Classes view) can be annotated. In CRISP-DM view, this can be an extremely effective way to document your organization's goals for each phase of data mining. For example, using the annotation tool for the *Business Understanding* folder, you can include documentation such as "The business objective for this study is to reduce churn among high-value customers." This text could then be automatically included in the project report by selecting the **Include in report** option.

To annotate a folder:

1. Select a folder in the project pane.
2. Right-click the folder and click **Folder Properties**.

In CRISP-DM view, folders are annotated with a summary of the purpose of each phase as well as guidance on completing the relevant data mining tasks. You can remove or edit any of these annotations.

Name. This area displays the name of the selected field.

Tooltip text. Create custom ToolTips that will be displayed when you hover the mouse pointer over a project folder. This is useful in CRISP-DM view, for example, to provide a quick overview of each phase's goals or to mark the status of a phase, such as "In progress" or "Complete."

Annotation field. Use this field for more lengthy annotations that can be collated in the project report. The CRISP-DM view includes a description of each data mining phase in the annotation, but you should feel free to customize this for your own project.

Include in report. To include the annotation in reports, select **Include in report**.

Object Properties

You can view object properties and choose whether to include individual objects in the project report. To access object properties:

1. Right-click an object in the project pane.
2. On the menu, click **Object Properties**.

Name. This area lists the name of the saved object.

Path. This area lists the location of the saved object.

Include in report. Select this option to include the object details in a generated report.

Closing a Project

When you exit IBM SPSS Modeler or open a new project, the existing project file (*.cpj*) is closed.

Some files associated with the project (such as streams, nodes or graphs) may still be open. If you want to leave these files open, reply No to the message **... Do you want to save and close these files?**

If you modify and save any associated files after the close of a project, these updated versions will be included in the project the next time you open it. To prevent this behavior, remove the file from the project or save it under a different filename.

Generating a Report

One of the most useful features of projects is the ability to generate reports based on the project items and annotations. This is a critical component of effective data mining, as discussed throughout the CRISP-DM methodology. You can generate a report directly into one of several file types or to an output window on the screen for immediate viewing. From there, you can print, save, or view the report in a web browser. You can distribute saved reports to others in your organization.

Reports are often generated from project files several times during the data mining process for distribution to those involved in the project. The report culls information about the objects referenced from the project file as well as any annotations created. You can create reports based on either the Classes view or CRISP-DM view.

To generate a report:

1. Select the project folder in either CRISP-DM or Classes view.
2. Right-click the folder and click **Project Report**.
3. Specify the report options and click **Generate Report**.

The options in the report dialog box provide several ways to generate the type of report you need:

Output name. Specify the name of the output window if you choose to send the output of the report to the screen. You can specify a custom name or let IBM SPSS Modeler automatically name the window for you.

Output to screen. Select this option to generate and display the report in an output window. Note that you have the option to export the report to various file types from the output window.

Output to file. Select this option to generate and save the report as a file of the type specified in the File type list.

Filename. Specify a filename for the generated report. Files are saved by default to the IBM SPSS Modeler *bin* directory. Use the ellipsis button **(...)** to specify a different location.

File type. Available file types are:

- **HTML document.** The report is saved as a single HTML file. If your report contains graphs, they are saved as PNG files and are referenced by the HTML file. When publishing your report on the Internet, make sure to upload both the HTML file and any images it references.
- **Text document.** The report is saved as a single text file. If your report contains graphs, only the filename and path references are included in the report.
- **Microsoft Word document.** The report is saved as a single document, with any graphs embedded directly into the document.
- **Microsoft Excel document.** The report is saved as a single spreadsheet, with any graphs embedded directly into the spreadsheet.
- **Microsoft PowerPoint document.** Each phase is shown on a new slide. Any graphs are embedded directly into the PowerPoint slides.
- **Output object.** When opened in IBM SPSS Modeler, this file (*.cou*) is the same as the **Output to screen** option in the **Report Format** group.

Note: To export to a Microsoft Office file, you must have the corresponding application installed.

Title. Specify a title for the report.

Report structure. Select either **CRISP-DM** or **Classes**. CRISP-DM view provides a status report with "big-picture" synopses as well as details about each phase of data mining. Classes view is an object-based view that is more appropriate for internal tracking of data and streams.

Author. The default user name is displayed, but you can change it.

Report includes. Select a method for including objects in the report. Select **all folders and objects** to include all items added to the project file. You can also include items based on whether **Include in Report** is selected in the object properties. Alternatively, to check on unreported items, you can choose to include only items marked for exclusion (where **Include in Report** is not selected).

Select. This option allows you to provide project updates by selecting only **recent items** in the report. Alternatively, you can track older and perhaps unresolved issues by setting parameters for **old items**. Select **all items** to dismiss time as a parameter for the report.

Order by. You can select a combination of the following object characteristics to order them within a folder:

- **Type.** Group objects by type.
- **Name.** Organize objects alphabetically.
- **Added date.** Sort objects using the date they were added to the project.

Saving and Exporting Generated Reports

A report generated to the screen is displayed in a new output window. Any graphs included in the report are displayed as in-line images.

Report Terminology

The total number of nodes in each stream is listed within the report. The numbers are shown under the following headings, which use IBM SPSS Modeler terminology, not CRISP-DM terminology:

- **Data readers.** Source nodes.
- **Data writers.** Export nodes.
- **Model builders.** Build, or Modeling, nodes.
- **Model appliers.** Generated models, also known as nuggets.
- **Output builders.** Graph or Output nodes.
- **Other.** Any other nodes related to the project. For example, those available on the Field Ops tab or Record Ops tab on the Nodes Palette.

To save a report:

1. On the File menu, click **Save**.
2. Specify a filename.

   The report is saved as an output object.

   To export a report:

3. On the File menu, click **Export** and the file type to which you want to export.
4. Specify a filename.

The report is saved in the format you chose.

You can export to the following file types:

- HTML
- Text
- Microsoft Word
- Microsoft Excel

- Microsoft PowerPoint

Note: To export to a Microsoft Office file, you must have the corresponding application installed.

Use the buttons at the top of the window to:

- Print the report.
- View the report as HTML in an external web browser.

Chapter 16. Customizing IBM SPSS Modeler

Customizing IBM SPSS Modeler options

There are a number of operations you can perform to customize IBM SPSS Modeler to your needs. Primarily, this customization consists of setting specific user options such as memory allocation, default directories, and the use of sound and color. You can also customize the Nodes palette located at the bottom of the IBM SPSS Modeler window.

Setting IBM SPSS Modeler options

There are several ways to customize and set options for IBM SPSS Modeler:

- Set system options, such as memory usage and locale, by clicking **System Options** on the **Tools > Options** menu.
- Set user options, such as display fonts and colors, by clicking **User Options** on the **Tools > Options** menu.
- Specify the location of applications that work with IBM SPSS Modeler by clicking **Helper Applications** on the **Tools > Options** menu.
- Specify the default directories used in IBM SPSS Modeler by clicking **Set Directory** or **Set Server Directory** on the File menu.

You can also set options that apply to some or all of your streams. See the topic "Setting options for streams" on page 36 for more information.

System Options

You can specify the preferred language or locale for IBM SPSS Modeler by clicking **System Options** on the **Tools > Options** menu. Here you can also set the maximum memory usage for SPSS Modeler, and specify how often to automatically save streams. Note that changes made in this dialog box will not take effect until you restart SPSS Modeler.

Maximum memory. Select to impose a limit in megabytes on IBM SPSS Modeler's memory usage. On some platforms, SPSS Modeler limits its process size to reduce the toll on computers with limited resources or heavy loads. If you are dealing with large amounts of data, this may cause an "out of memory" error. You can ease memory load by specifying a new threshold.

For example, attempting to display a very large decision tree may cause a memory error. In this case, we recommend that you increase the memory to the high value such as 4096Mb. In cases such as these, where you are likely to be processing very large amounts of data, after you increase the memory allowance shut down SPSS Modeler and start it from a command line to ensure the maximum amount of memory is used when processing your data.

To start from a command line (assuming SPSS Modeler is installed in the default location), in a Command Prompt window, enter the following:

```
C:\Program Files\IBM\SPSS\Modeler\18.6.0\bin\modelerclient.exe" -J-Xss4096M
```

Note: The maximum memory threshold you can specify is 58368Mb due a Java constraint.

Use system locale. This option is selected by default and set to English (United States). Deselect to specify another language from the list of available languages and locales.

Stream auto save interval (minutes). Specify how often you want SPSS Modeler to save streams automatically. The maximum is 60 minutes, minimum is 1 minute, and default is 5 minutes.

Managing Memory

In addition to the **Maximum memory** setting specified in the System Options dialog box, there are several ways you can optimize memory usage:

- Adjust the **Maximum members for nominal fields** option in the stream properties dialog box. This option specifies a maximum number of members for nominal fields after which the measurement level of the field becomes *Typeless*. See the topic "Setting general options for streams" on page 37 for more information.
- Force IBM SPSS Modeler to free up memory by clicking in the lower right corner of the window where the memory that IBM SPSS Modeler is using and the amount allocated are displayed (*xx*MB / *xx*MB). Clicking this region turns it a darker shade, after which memory allocation figures will drop. Once the region returns to its regular color, IBM SPSS Modeler has freed up all the memory possible.

Setting Default Directories

You can specify the default directory used for file browsers and output by selecting **Set Directory** or **Set Server Directory** from the File menu.

- **Set Directory.** You can use this option to set the working directory. The default working directory is based on the installation path of your version of IBM SPSS Modeler or from the command line path used to launch IBM SPSS Modeler. In local mode, the working directory is the path used for all client-side operations and output files (if they are referenced with relative paths).
- **Set Server Directory.** The Set Server Directory option on the File menu is enabled whenever there is a remote server connection. Use this option to specify the default directory for all server files and data files specified for input or output. The default server directory is *$CLEO/data*, where *$CLEO* is the directory in which the Server version of IBM SPSS Modeler is installed. Using the command line, you can also override this default by using the -server_directory flag with the modelerclient command line argument.

Setting user options

You can set general options for IBM SPSS Modeler by selecting **User Options** from the **Tools > Options** menu. These options apply to all streams used in IBM SPSS Modeler.

The following types of options can be set by clicking the corresponding tab:

- Notification options, such as model overwriting and error messages.
- Display options, such as graph and background colors.
- Syntax color display options.
- PMML export options used when exporting models to Predictive Model Markup Language (PMML).
- User or author information, such as your name, initials, and e-mail address. This information may be displayed on the Annotations tab for nodes and for other objects that you create.
- Switching between traditional mode and Analytic Server mode.

To set stream-specific options, such as decimal separators, time and data formats, optimization, stream layout, and stream scripts, use the Stream Properties dialog box, available from the File and Tools menus.

Setting Notification Options

Using the Notifications tab of the User Options dialog box, you can set various options regarding the occurrence and type of warnings and confirmation windows in IBM SPSS Modeler. You can also specify the behavior of the Outputs and Models tabs in the managers pane when new output and models are generated.

Show stream execution feedback dialog Select to display a dialog box, that includes a progress indicator, when a stream has been running for three seconds. The dialog box also includes details of the output objects created by the stream.

- **Close dialog upon completion** By default, the dialog box closes when the stream finishes running. Clear this check box if you want the dialog box to remain visible when the stream finishes.

Warn when a node overwrites a file Select to warn with an error message when node operations overwrite an existing file.

Warn when a node overwrites a database table Select to warn with an error message when node operations overwrite an existing database table.

Sound Notifications

Use the list to specify whether sounds notify you when an event or error occurs. There are a number of sounds available. Use the Play (loudspeaker) button to play a selected sound. Use the ellipsis button **(...)** to browse for and select a sound.

Note: The `.wav` files used to create sounds in IBM SPSS Modeler are stored in the `/media/sounds` directory of your installation.

- **Mute all sounds** Select to turn off sound notification for all events.

Visual Notifications

The options in this group are used to specify the behavior of the Outputs and Models tabs in the managers pane at the top right of the display when new items are generated. Select **New Model** or **New Output** from the list to specify the behavior of the corresponding tab.

The following option is available for **New Model**:

Replace previous model If selected (default), overwrites an existing model from this stream in the Models tab and on the stream canvas. If this box is unchecked, the model is added to the existing models on the tab and the canvas. Note that this setting is overridden by the model replacement setting on a model link.

The following option is available for **New Output**:

Warn when outputs exceed [n] Select whether to display a warning when the number of items on the Outputs tab exceeds a prespecified quantity. The default quantity is 20; however, you can change this if needed.

The following options are available in all cases:

Select tab Choose whether to switch to the Outputs or Models tab when the corresponding object is generated while the stream runs.

- Select **Always** to switch to the corresponding tab in the managers pane.
- Select **If generated by current stream** to switch to the corresponding tab only for objects generated by the stream currently visible in the canvas.
- Select **Never** to restrict the software from switching to the corresponding tab to notify you of generated outputs or models.

Flash tab Select whether to flash the Outputs or Models tab in the managers pane when new outputs or models have been generated.

- Select **If not selected** to flash the corresponding tab (if not already selected) whenever new objects are generated in the managers pane.
- Select **Never** to restrict the software from flashing the corresponding tab to notify you of generated objects.

Scroll palette to make visible (**New Model** only). Select whether to automatically scroll the Models tab in the managers pane to make the most recent model visible.

- Select **Always** to enable scrolling.
- Select **If generated by current stream** to scroll only for objects generated by the stream currently visible in the canvas.

- Select **Never** to restrict the software from automatically scrolling the Models tab.

Open window (**New Output** only). Select whether to automatically open an output window upon generation.

- Select **Always** to always open a new output window.
- Select **If generated by current stream** to open a new window for output generated by the stream currently visible in the canvas.
- Select **Never** to restrict the software from automatically opening new windows for generated output.

To revert to the system default settings for this tab, click **Default Values**.

Setting display options

Using the Display tab of the User Options dialog box, you can set options for the display of fonts and colors in IBM SPSS Modeler.

Show welcome dialog on startup. Select to cause the welcome dialog box to be displayed on startup. The welcome dialog box has options to launch the application examples tutorial, open a demonstration stream or an existing stream or project, or to create a new stream.

Show stream and SuperNode markups. If selected, causes markup (if any) on streams and SuperNodes to be displayed by default. Markup includes stream comments, model links, and highlighting of scoring branches.

Standard Fonts & Colors (effective on restart). Options in this control box are used to specify the IBM SPSS Modeler screen design, color scheme, and the size of the fonts displayed. The options that you select here do not take effect until you close and restart IBM SPSS Modeler.

- **Look and feel.** Select a standard color scheme and screen design. You can choose from:
  - **SPSS Standard**, the default design.
  - **SPSS Classic**, a design familiar to users of earlier versions of SPSS Modeler.
  - **Windows**, a Windows design that can be useful for increased contrast in the stream canvas and palettes.
  - **Analytics Carbon**, a modern design with sleek icons and colors.
- **Default font size for nodes.** Specify a font size to be used in the node palettes and for nodes that are displayed in the stream canvas.
- **Specify fixed width font.** To select a fixed width font, and associated font **Size** for use in scripting and CLEM expression controls, select this check box. The default font is Monospace plain; click **Change...** to display a list of other fonts that you can select.

Note: You can set the size of the node icons for a stream on the Layout pane of the Options tab of the stream properties dialog box. From the main menu, choose **Tools** > **Stream Properties** > **Options** > **Layout**.

Custom Colors. This table lists the currently selected colors that are used for various display items. For each of the items that are listed in the table, you can change the current color by double-clicking the corresponding row in the **Color** column and selecting a color from the list. To specify a custom color, scroll to the bottom of the list and click the **Color...** entry.

Chart Category Color Order. This table lists the currently selected colors that are used for display in newly created graphs. The order of the colors reflects the order in which they are used in the chart. For example, if a nominal field used as a color overlay contains four unique values, then only the first four colors that are listed here are used. For each of the items that are listed in the table, you can change the current color by double-clicking the corresponding row in the **Color** column and selecting a color from the list. To specify a custom color, scroll to the bottom of the list and click the **Color...** entry. Changes that you make here do not affect previously created graphs.

To revert to the system default settings for this tab, click **Default Values**.

Setting Syntax Display Options

Using the Syntax tab of the User Options dialog box, you can set options for the font attributes and display colors in scripts that you create in IBM SPSS Modeler.

Syntax highlighting. This table lists the currently selected colors used for various syntax items, including both the font and the window in which it is displayed. For each of the items listed in the table you can change the color by clicking the corresponding drop-down list in the row and selecting a color from the list. In addition, for font items, you can choose to add bold and italic emphasis.

Preview. This table shows an example syntax display that uses the colors and font attributes that you select in the **Syntax highlighting** table. This preview updates as soon as you change any selection.

Click **Default Values** to revert to the system default settings for this tab.

Setting PMML Export Options

On the PMML tab, you can control how IBM SPSS Modeler exports models to Predictive Model Markup Language (PMML). See the topic "Importing and exporting models as PMML" on page 213 for more information.

Export PMML. Here you can configure variations of PMML that work best with your target application.

- Select **With extensions** to allow PMML extensions for special cases where there is no standard PMML equivalent. Note that in most cases this will produce the same result as standard PMML.
- Select **As standard PMML...** to export PMML that adheres as closely as possible to the PMML standard.

Standard PMML Options. When the **As standard PMML...** option is selected, you can choose one of two valid ways to export linear and logistic regression models:

- As **PMML <GeneralRegression> models**
- As **PMML <Regression> models**

For more information on PMML, see the Data Mining Group website at http://www.dmg.org.

Setting User Information

User/Author Information. Information you enter here can be displayed on the Annotations tab of nodes and other objects that you create.

Setting the mode

Modeler Mode Settings. On the **Mode** tab, you can choose from the following modes:

- **Traditional SPSS Modeler mode** shows all available nodes and expressions in the user interface.
- **Analytic Server mode** shows only the nodes and expressions supported by Analytic Server. But note that some nodes and some CLEM expressions will still be displayed even though they're not *fully* supported by Analytic Server. The following table provides general information about which nodes are supported, partially supported, and unsupported by Analytic Server.

See the Analytic Server documentation for more information about Analytic Server.

Note that if you configure SPSS Modeler to show database nodes on the Database Modeling palette, they will not be impacted when switching modes. The database nodes will always be displayed. If you use IBM Db2 for z/OS or IBM Netezza integration, in some cases those nodes may disappear from the Database Modeling palette after switching to Analytic Server mode. If this happens, go to **Tools** > **Options** > **Helper Applications** and reset the check boxes.

Table 38. Node support			
Node type (palette name)	Supported by Analytic Server	Partially supported by Analytic Server	Unsupported by Analytic Server
Sources	• Analytic Server source node		• Database • Var. File • Fixed File • Statistics File • Data Collection • IBM Cognos • TWC Import • TM1 Import • SAS File • Excel • XML User Input • Sim Gen • Data View • Geospatial • Object Storage • Extension Import • R Import • SNA (Diffusion Analysis and Group Analysis)
Record Ops	• Select • Sort • Balance • Distinct • RFM Aggregate • Append • Streaming TS • Extension Transformation • Streaming TCM	• Sample (Only supports Random% for Simple method. Complex is not supported.) • Merge (only supports join by keys and conditions) • Aggregate (1st quantile, 3rd quantile, and median are not supported by Analytic Server)	• Space-Time-Boxes • CPLEX Optimization

Table 38. Node support (continued)			
Node type (palette name)	**Supported by Analytic Server**	**Partially supported by Analytic Server**	**Unsupported by Analytic Server**
Field Ops	• Type • Filter • Derive • Filler • Reclassify • Ensemble • SetToFlag • Restructure • Field Reorder • Reproject • Time Intervals	• Auto Data Prep (only supports transform) • Binning (binning method `equalFreq` is not supported when Ties setting **Keep in current** is selected) • RFM Analysis (binning method `Tiles` is not supported when Ties setting **Keep in current**" is selected) • Partition (not supported by Analytic Server unless a unique field is used to repeatably assign rows to partitions)	• Anonymize • History • Transpose
Graphs	• Plot • Multiplot • Time Plot • Distribution • Histogram • Collection • Web • Evaluation • Map Visualization • E-Plot (Beta) • t-SNE	• Graphboard (only supports the aggregation mode function for fields having a measurement level of discrete, nominal , ordinal, or flag)	

Table 38. Node support (continued)			
Node type (palette name)	**Supported by Analytic Server**	**Partially supported by Analytic Server**	**Unsupported by Analytic Server**

Modeling	• Time Series
	• TCM
	• Isotonic-AS
	• Random Trees
	• Tree-AS
	• Linear-AS
	• GLE
	• LSVM
	• STP
	• TwoStep-AS
	• Association Rules
	• XGBoost-AS
	• K-Means-AS
		• Auto Classifier
		• Auto Numeric (the two nodes only support split, and a field with a split role must be supplied when using the Auto Classifier option **Run on Analytic Server (splits enabled)**)
		• Extension (the R syntax building model is not supported by Analytic Server)
		• The following nodes only support split and PSM:
		  – C&R Tree
		  – Linear
		  – Neural Net
		  – CHAID
		  – Quest
			• Auto cluster
			• Decision List
			• C5.0
			• Regression
			• PCA/Factor
			• Feature Selection
			• Discriminant
			• Logistic
			• GenLin
			• GLMM
			• Bayes Net
			• Apriori
			• Carma
			• Sequence
			• K-Means
			• Kohonen
			• TwoStep
			• Anomaly
			• KNN
			• R
			• Random Forest
			• The following nodes have asl but just read-write asl:
			  – Cox
			  – SVM
			  – SLRM

Output	• Table • Matrix • Analysis • Data Audit • Transform • Statistics • Means • Report • Set Globals	• Extension Output • R Syntax Output	• Sim Eval • Sim Fit • R Output

Table 38. Node support (continued)			
Node type (palette name)	**Supported by Analytic Server**	**Partially supported by Analytic Server**	**Unsupported by Analytic Server**
Export	• Analytic Server export node	• Extension Export • R Syntax Export	• Database • Flat File • Statistics Export • Data Collection • Excel • IBM Cognos Export • TM1 Export • SAS • XML Export • Object Storage • R Export
IBM SPSS Statistics			• Statistics File • Statistics Transform • Statistics Model • Statistics Output • Statistics Export
IBM SPSS Text Analytics	• Text Link Analysis • Text Mining • Language node		• File List • Web Feed • Text Link Analysis • Translate • Text Mining • File Viewer
Python			• SMOTE • One-Class SVM • XGBoost Tree • XGBoost Linear • t-SNE • Random Forest • HDBSCAN
Spark	All supported		

Customizing the Nodes Palette

Streams are built using nodes. The Nodes Palette at the bottom of the IBM SPSS Modeler window contains all of the nodes it is possible to use in stream building. See the topic "Nodes palette" on page 13 for more information.

You can reorganize the Nodes Palette in two ways:

• Customize the Palette Manager. See the topic "Customizing the Palette Manager" on page 234 for more information.

- Change how palette tabs that contain subpalettes are displayed on the Nodes Palette. See the topic "Creating a Subpalette" on page 235 for more information.

Customizing the Palette Manager

The Palette Manager can be customized to accommodate your usage of IBM SPSS Modeler. For example, if you frequently analyze time-series data from a database, you might want to be sure that the Database source node, the Time intervals node, the Time Series node, and the Time Plot graph node are available together from a unique palette tab. The Palette Manager enables you to easily make these adjustments by creating your custom palette tabs in the Nodes Palette.

The Palette Manager enables you to carry out various tasks:

- Control which palette tabs are shown on the Nodes Palette below the stream canvas.
- Change the order in which palette tabs are shown on the Nodes Palette.
- Create and edit your own palette tabs and any associated subpalettes.
- Edit the default node selections on your tabs.

To access the Palette Manager, on the Tools menu, click **Manage Palettes**.

Palette Name. Each available palette tab, whether shown on the Nodes Palette or not, is listed. This includes any palette tabs that you have created. See the topic "Creating a Palette Tab" on page 234 for more information.

No. of nodes. The number of nodes displayed on each palette tab. A high number here means you may find it more convenient to create subpalettes to divide up the nodes on the tab. See the topic "Creating a Subpalette" on page 235 for more information.

Shown? Select this field to display the palette tab on the Nodes Palette. See the topic "Displaying Palette Tabs on the Nodes Palette" on page 235 for more information.

Sub Palettes. To select subpalettes for display on a palette tab, highlight the required **Palette Name** and click this button to display the Sub Palettes dialog box. See the topic "Creating a Subpalette" on page 235 for more information.

Restore Defaults. To completely remove all changes and additions you have made to the palettes and subpalettes and return to the default palette settings, click this button.

Creating a Palette Tab

To create a custom palette tab:

1. From the Tools menu, open the Palette Manager.
2. To the right of the *Shown?* column, click the Add Palette button; the Create/Edit Palette dialog box is displayed.
3. Type in a unique **Palette name**.
4. In the **Nodes available** area, select the node to be added to the palette tab.
5. Click the Add Node right-arrow button to move the highlighted node to the **Selected nodes** area. Repeat until you have added all the nodes you want.

   After you have added all of the required nodes, you can change the order in which they are displayed on the palette tab:
6. Use the simple arrow buttons to move a node up or down one row.
7. Use the line-arrow buttons to move a node to the bottom or top of the list.
8. To remove a node from a palette, highlight the node and click the Delete button to the right of the **Selected nodes** area.

Displaying Palette Tabs on the Nodes Palette

There may be options available within IBM SPSS Modeler that you never use; in this case, you can use the Palette Manager to hide the tabs containing these nodes.

To select which tabs are to be shown on the Nodes Palette:

1. From the Tools menu, open the Palette Manager.
2. Using the check boxes in the *Shown?* column, select whether to include or hide each palette tab.

To permanently remove a palette tab from the Nodes Palette, highlight the node and click the Delete button to the right of the *Shown?* column. Once deleted, a palette tab cannot be recovered.

Note: You cannot delete the default palette tabs supplied with IBM SPSS Modeler, except for the Favorites tab.

Changing the display order on the Nodes Palette

After you have selected which palette tabs you want to display, you can change the order in which they are displayed on the Nodes Palette:

1. Use the simple arrow buttons to move a palette tab up or down one row. Moving them up moves them to the left of the Nodes Palette, and vice versa.
2. Use the line-arrow buttons to move a palette tab to the bottom or top of the list. Those at the top of the list will be shown on the left of the Nodes Palette.

Displaying Subpalettes on a Palette Tab

In the same way that you can control which palette tabs are displayed on the Nodes Palette, you can control which subpalettes are available from their parent palette tab.

To select subpalettes for display on a palette tab:

1. From the Tools menu, open the Palette Manager.
2. Select the palette that you require.
3. Click the Sub Palettes button; the Sub Palettes dialog box is displayed.
4. Using the check boxes in the *Shown?* column, select whether to include each subpalette on the palette tab. The **All** subpalette is always shown and cannot be deleted.
5. To permanently remove a subpalette from the palette tab, highlight the subpalette and click the Delete button to the right of the *Shown?* column.

Note: You cannot delete the default subpalettes supplied with the Modeling palette tab.

Changing the display order on the Palette Tab

After you have selected which subpalettes you want to display, you can change the order in which they are displayed on the parent palette tab:

1. Use the simple arrow buttons to move a subpalette up or down one row.
2. Use the line-arrow buttons to move a subpalette to the bottom or top of the list.

The subpalettes you create are displayed on the Nodes Palette when you select their parent palette tab. See the topic "Changing a Palette Tab View" on page 236 for more information.

Creating a Subpalette

Because you can add any existing node to the custom palette tabs that you create, it is possible that you will select more nodes than can be easily displayed on screen without scrolling. To prevent having to scroll, you can create subpalettes into which you place the nodes you chose for the palette tab. For example, if you created a palette tab that contains the nodes you use most frequently for creating your streams, you could create four subpalettes that break the selections down by source node, field operations, modeling, and output.

Note: You can only select subpalette nodes from those added to the parent palette tab.

To create a subpalette:

1. From the Tools menu, open the Palette Manager.
2. Select the palette to which you want to add subpalettes.
3. Click the Sub Palettes button; the Sub Palettes dialog box is displayed.
4. To the right of the *Shown?* column, click the Add Sub Palette button; the Create/Edit Sub Palette dialog box is displayed.
5. Type in a unique **Sub palette name**.
6. In the **Nodes available** area, select the node to be added to the subpalette.
7. Click the Add Node right-arrow button to move a selected node to the **Selected nodes** area.
8. When you have added the required nodes, click **OK** to return to the Sub Palettes dialog box.

The subpalettes you create are displayed on the Nodes Palette when you select their parent palette tab. See the topic "Changing a Palette Tab View" on page 236 for more information.

Changing a Palette Tab View

Due to the large number of nodes available in IBM SPSS Modeler, they may not all be visible on smaller screens without scrolling to the left or right of the Nodes Palette; this is especially noticeable on the Modeling palette tab. To reduce the need to scroll, you can choose to display only the nodes contained in a subpalette (where available). See the topic "Creating a Subpalette" on page 235 for more information.

To change the nodes shown on a palette tab, select the palette tab and then, from the menu on the left, select to display either all nodes, or just those in a specific subpalette.

Chapter 17. Performance considerations for streams and nodes

You can design your streams to maximize performance by arranging the nodes in the most efficient configuration, by enabling node caches when appropriate, and by paying attention to other considerations as detailed in this section.

Aside from the considerations discussed here, additional and more substantial performance improvements can typically be gained by making effective use of your database, particularly through SQL optimization.

Order of Nodes

Even when you are not using SQL optimization, the order of nodes in a stream can affect performance. The general goal is to minimize downstream processing; therefore, when you have nodes that reduce the amount of data, place them near the beginning of the stream. IBM SPSS Modeler Server can apply some reordering rules automatically during compilation to bring forward certain nodes when it can be proven safe to do so. (This feature is enabled by default. Check with your system administrator to make sure it is enabled in your installation.)

When using SQL optimization, you want to maximize its availability and efficiency. Since optimization halts when the stream contains an operation that cannot be performed in the database, it is best to group SQL-optimized operations together at the beginning of the stream. This strategy keeps more of the processing in the database, so less data is carried into IBM SPSS Modeler.

The following operations can be done in most databases. Try to group them at the *beginning* of the stream:

- Merge by key (join)
- Select
- Aggregate
- Sort
- Sample
- Append
- Distinct operations in *include* mode, in which all fields are selected
- Filler operations
- Basic derive operations using standard arithmetic or string manipulation (depending on which operations are supported by the database)
- Set-to-flag

The following operations cannot be performed in most databases. They should be placed in the stream *after* the operations in the preceding list:

- Operations on any nondatabase data, such as flat files
- Merge by order
- Balance
- Distinct operations in *discard* mode or where only a subset of fields are selected as distinct
- Any operation that requires accessing data from records other than the one being processed
- State and count field derivations
- History node operations
- Operations involving "@" (time-series) functions

- Type-checking modes *Warn* and *Abort*
- Model construction, application, and analysis

    Note: Decision trees, rulesets, linear regression, and factor-generated models can generate SQL and can therefore be pushed back to the database.
- Data output to anywhere other than the same database that is processing the data

Node Caches

To optimize stream running, you can set up a *cache* on any nonterminal node. When you set up a cache on a node, the cache is filled with the data that passes through the node the next time you run the data stream. From then on, the data is read from the cache (which is stored on disk in a temporary directory) rather than from the data source.

Caching is most useful following a time-consuming operation such as a sort, merge, or aggregation. For example, suppose that you have a source node set to read sales data from a database and an Aggregate node that summarizes sales by location. You can set up a cache on the Aggregate node rather than on the source node because you want the cache to store the aggregated data rather than the entire data set.

Note: Caching at source nodes, which simply stores a copy of the original data as it is read into IBM SPSS Modeler, will not improve performance in most circumstances.

Nodes with caching enabled are displayed with a small document icon at the top right corner. When the data is cached at the node, the document icon is green.

To Enable a Cache

1. On the stream canvas, right-click the node and click **Cache** on the menu.
2. On the caching submenu, click **Enable**.
3. You can turn the cache off by right-clicking the node and clicking **Disable** on the caching submenu.

Caching Nodes in a Database

For streams run in a database, data can be cached midstream to a temporary table in the database rather than the file system. When combined with SQL optimization, this may result in significant gains in performance. For example, the output from a stream that merges multiple tables to create a data mining view may be cached and reused as needed. By automatically generating SQL for all downstream nodes, performance can be further improved.

To take advantage of database caching, both SQL optimization and database caching must be enabled. Note that Server optimization settings override those on the Client. See the topic "Setting optimization options for streams" on page 39 for more information.

With database caching enabled, simply right-click any nonterminal node to cache data at that point, and the cache will be created automatically directly in the database the next time the stream is run. If database caching or SQL optimization is not enabled, the cache will be written to the file system instead.

Note: The following databases support temporary tables for the purpose of caching: Db2, Oracle, SQL Server, and Teradata. Other databases, such as Netezza, will use a normal table for database caching. The SQL code can be customized for specific databases - contact Services for assistance.

Performance: Process Nodes

Sort. The Sort node must read the entire input data set before it can be sorted. The data is stored in memory up to some limit, and the excess is spilled to disk. The sorting algorithm is a combination algorithm: data is read into memory up to the limit and sorted using a fast hybrid quick-sort algorithm. If all the data fits in memory, then the sort is complete. Otherwise, a merge-sort algorithm is applied. The sorted data is written to file and the next chunk of data is read into memory, sorted, and written to disk. This is repeated until all the data has been read; then the sorted chunks are merged. Merging may require repeated passes over the data stored on disk. At peak usage, the Sort node will have two complete copies of the data set on disk: sorted and unsorted.

The overall running time of the algorithm is on the order of $N*\log(N)$, where N is the number of records. Sorting in memory is faster than merging from disk, so the actual running time can be reduced by allocating more memory to the sort. The algorithm allocates to itself a fraction of physical RAM controlled by the IBM SPSS Modeler Server configuration option *Memory usage multiplier*. To increase the memory used for sorting, provide more physical RAM or increase this value. Note that when the proportion of memory used exceeds the working set of the process so that part of the memory is paged to disk, performance degrades because the memory-access pattern of the in-memory sort algorithm is random and can cause excessive paging. The sort algorithm is used by several nodes other than the Sort node, but the same performance rules apply.

Binning. The Binning node reads the entire input data set to compute the bin boundaries, before it allocates records to bins. The data set is cached while the boundaries are computed; then it is rescanned for allocation. When the binning method is *fixed-width* or *mean+standard deviation*, the data set is cached directly to disk. These methods have a linear running time and require enough disk space to store the entire data set. When the binning method is *ranks* or *tiles*, the data set is sorted using the sort algorithm described earlier, and the sorted data set is used as the cache. Sorting gives these methods a running time of $M*N*\log(N)$, where M is the number of binned fields and N is the number of records; it requires disk space equal to twice the data set size.

Generating a Derive node based on generated bins will improve performance in subsequent passes. Derive operations are much faster than binning.

Merge by Key (Join). The Merge node, when the merge method is *keys* (equivalent to a database join), sorts each of its input data sets by the key fields. This part of the procedure has a running time of $M*N*\log(N)$, where M is the number of inputs and N is the number of records in the largest input; it requires sufficient disk space to store all of its input data sets plus a second copy of the largest data set. The running time of the merge itself is proportional to the size of the output data set, which depends on the frequency of matching keys. In the worst case, where the output is the Cartesian product of the inputs, the running time may approach NM. This is rare—most joins have many fewer matching keys. If one data set is relatively larger than the other(s), or if the incoming data is already sorted by a key field, then you can improve the performance of this node using the Optimization tab.

Aggregate. When the *Keys are contiguous* option is not set, this node reads (but does not store) its entire input data set before it produces any aggregated output. In the more extreme situations, where the size of the aggregated data reaches a limit (determined by the IBM SPSS Modeler Server configuration option *Memory usage multiplier*), the remainder of the data set is sorted and processed as if the *Keys are contiguous* option were set. When this option is set, no data is stored because the aggregated output records are produced as the input data is read.

Distinct. The Distinct node stores all of the unique key fields in the input data set; in cases where all fields are key fields and all records are unique it stores the entire data set. By default the Distinct node sorts the data on the key fields and then selects (or discards) the first distinct record from each group. For smaller data sets with a low number of distinct keys, or those that have been pre-sorted, you can choose options to improve the speed and efficiency of processing.

Type. In some instances, the Type node caches the input data when reading values; the cache is used for downstream processing. The cache requires sufficient disk space to store the entire data set but speeds up processing.

Evaluation. The Evaluation node must sort the input data to compute tiles. The sort is repeated for each model evaluated because the scores and consequent record order are different in each case. The running time is $M*N*\log(N)$, where M is the number of models and N is the number of records.

Performance: Modeling Nodes

Neural Net and Kohonen. Neural network training algorithms (including the Kohonen algorithm) make many passes over the training data. The data is stored in memory up to a limit, and the excess is spilled to disk. Accessing the training data from disk is expensive because the access method is random, which can lead to excessive disk activity. You can disable the use of disk storage for these algorithms, forcing all data to be stored in memory, by selecting the **Optimize for speed** option on the Model tab of the node's dialog

box. Note that if the amount of memory required to store the data is greater than the working set of the server process, part of it will be paged to disk and performance will suffer accordingly.

When **Optimize for memory** is enabled, a percentage of physical RAM is allocated to the algorithm according to the value of the IBM SPSS Modeler Server configuration option *Modeling memory limit percentage*. To use more memory for training neural networks, either provide more RAM or increase the value of this option, but note that setting the value too high will cause paging.

The running time of the neural network algorithms depends on the required level of accuracy. You can control the running time by setting a stopping condition in the node's dialog box.

K-Means. The K-Means clustering algorithm has the same options for controlling memory usage as the neural network algorithms. Performance on data stored on disk is better, however, because access to the data is sequential.

Performance: CLEM Expressions

CLEM sequence functions ("@ functions") that look back into the data stream must store enough of the data to satisfy the longest look-back. For operations whose degree of look-back is unbounded, all values of the field must be stored. An unbounded operation is one where the offset value is not a literal integer; for example, `@OFFSET(Sales, Month)`. The offset value is the field name *Month*, whose value is unknown until executed. The server must save all values of the *Sales* field to ensure accurate results. Where an upper bound is known, you should provide it as an additional argument; for example, `@OFFSET(Sales, Month, 12)`. This operation instructs the server to store no more than the 12 most recent values of *Sales*. Sequence functions, bounded or otherwise, almost always inhibit SQL generation.

Chapter 18. Accessibility in IBM SPSS Modeler

Overview of Accessibility in IBM SPSS Modeler

IBM SPSS Modeler provides accessibility support for all users, as well as specific support for users with visual and other functional impairments. This section describes the features and methods of working using accessibility enhancements, such as screen readers and keyboard shortcuts.

Types of Accessibility Support

Whether you have a visual impairment or are dependent on the keyboard for manipulation, there is a wide variety of alternative methods for using this data mining toolkit. For example, you can build streams, specify options, and read output, all without using the mouse. Available keyboard shortcuts are listed in the topics that follow. Additionally, IBM SPSS Modeler provides extensive support for screen readers, such as JAWS for Windows. You can also optimize the color scheme to provide additional contrast. These types of support are discussed in the following topics.

Accessibility for the Visually Impaired

There are a number of properties you can specify in IBM SPSS Modeler that will enhance your ability to use the software.

Display Options

You can select colors for the display of graphs. You can also choose to use your specific Windows settings for the software itself. This may help to increase visual contrast.

1. To set display options, on the Tools menu, click **User Options**.
2. Click the **Display** tab. The options on this tab include the software color scheme, chart colors, and font sizes for nodes.

Note: The screen reader is not able to read graphs, so these are not accessible to visually-impaired users.

Use of Sounds for Notification

By turning sounds on or off, you can control the way you are alerted to particular operations in the software. For example, you can activate sounds for events such as node creation and deletion or the generation of new output or models.

1. To set notification options, on the Tools menu, click **User Options**.
2. Click the **Notifications** tab.

Controlling the Automatic Launching of New Windows

The Notifications tab on the User Options dialog box is also used to control whether newly generated output, such as tables and charts, are launched in a separate window. It may be easier for you to disable this option and open an output window only when required.

1. To set these options, on the Tools menu, click **User Options**.
2. Click the **Notifications** tab.
3. In the dialog box, select **New Output** from the list in the **Visual Notifications** group.
4. Under **Open Window**, select **Never**.

Node Size

Nodes can be displayed using either a standard or small size. You may want to adjust these sizes to fit your needs.

1. To set node size options, on the File menu, click **Stream Properties**.
2. Click the **Layout** tab.
3. From the **Icon Size** list, select **Standard**.

Accessibility for Blind Users

Support for blind users is predominately dependent on the use of a screen reader, such as JAWS for Windows. To optimize the use of a screen reader with IBM SPSS Modeler, you can specify a number of settings.

Display Options

Screen readers tend to perform better when the visual contrast is greater on the screen. If you already have a high-contrast Windows setting, you can choose to use these Windows settings for the software itself.

1. To set display options, on the Tools menu, click **User Options**.
2. Click the **Display** tab.

Note: The screen reader is not able to read graphs, so these are not accessible to blind users.

Use of Sounds for Notification

By turning on or off sounds, you can control the way you are alerted to particular operations in the software. For example, you can activate sounds for events such as node creation and deletion or the generation of new output or models.

1. To set notification options, on the Tools menu, click **User Options**.
2. Click the **Notifications** tab.

Controlling the Automatic Launching of New Windows

The Notifications tab on the User Options dialog box is also used to control whether newly generated output is launched in a separate window. It may be easier for you to disable this option and open an output window as needed.

1. To set these options, on the Tools menu, click **User Options**.
2. Click the **Notifications** tab.
3. In the dialog box, select **New Output** from the list in the **Visual Notifications** group.
4. Under **Open Window**, select **Never**.

Keyboard Accessibility

The product's functionality is accessible from the keyboard. At the most basic level, you can press Alt plus the appropriate key to activate window menus (such as Alt+F to access the File menu) or press the Tab key to scroll through dialog box controls. However, there are special issues related to each of the product's main windows and helpful hints for navigating dialog boxes.

This section will cover the highlights of keyboard accessibility, from opening a stream to using node dialog boxes to working with output. Additionally, lists of keyboard shortcuts are provided for even more efficient navigation.

Shortcuts for navigating the main window

You do most of your data mining work in the main window of IBM SPSS Modeler. The main area is called the **stream canvas** and is used to build and run data streams. The bottom part of the window contains the **node palettes**, which contain all available nodes. The palettes are organized on tabs corresponding to the type of data mining operation for each group of nodes. For example, nodes used to bring data into IBM SPSS Modeler are grouped on the Sources tab, and nodes used to derive, filter, or type fields are grouped on the Field Ops tab (short for Field Operations).

The right side of the window contains several tools for managing streams, output, and projects. The top half on the right contains the **managers** and has three tabs that are used to manage streams, output, and generated models. You can access these objects by selecting the tab and an object from the list. The bottom half on the right contains the **project pane**, which allows you to organize your work into projects. There are two tabs in this area reflecting two different views of a project. The **Classes view** sorts project objects by type, while the **CRISP-DM view** sorts objects by the relevant data mining phase, such as Data Preparation or Modeling. These various aspects of the IBM SPSS Modeler window are discussed throughout the Help system and User's Guide.

Following is a table of shortcuts used to move within the main IBM SPSS Modeler window and build streams. Shortcuts for dialog boxes and output are listed in the topics that follow. Note that these shortcut keys are available only from the main window.

Table 39. Main Window Shortcuts

Shortcut Key	Function
Ctrl+F5	Moves focus to the node palettes.
Ctrl+F6	Moves focus to the stream canvas.
Ctrl+F7	Moves focus to the managers pane.
Ctrl+F8	Moves focus to the project pane.

Table 40. Node and Stream Shortcuts

Shortcut Key	Function
Ctrl+N	Creates a new blank stream canvas.
Ctrl+O	Displays the Open dialog box, from where you can select and open an existing stream.
Ctrl+number keys	Moves focus to the corresponding tab on a window or pane. For example, within a tabbed pane or window, Ctrl+1 moves to the first tab starting from the left, Ctrl+2 to the second, etc.
Ctrl+Down Arrow	Used in the node palette to move focus from a palette tab to the first node under that tab.
Ctrl+Up Arrow	Used in the node palette to move focus from a node to its palette tab.
Enter	When a node is selected in the node palette (including refined models in the generated models palette), this keystroke adds the node to the stream canvas. Pressing **Enter** when a node is already selected on the canvas opens the dialog box for that node.
Ctrl+Enter	When a node is selected in the palette, adds that node to the stream canvas without selecting it, and moves focus to the first node in the palette.

Table 40. Node and Stream Shortcuts (continued)	
Shortcut Key	**Function**
Alt+Enter	When a node is selected in the palette, adds that node to the stream canvas and selects it, while moving focus to the first node in the palette.
Shift+Spacebar	When a node or comment has focus in the palette, toggles between selecting and deselecting that node or comment. If any other nodes or comments are also selected, this causes them to be deselected.
Ctrl+Shift+Spacebar	When a node or comment has focus in the stream, or a node or comment has focus on the palette, toggles between selecting and deselecting the node or comment. This does not affect any other selected nodes or comments.
Left/Right Arrow	If the stream canvas has focus, moves the entire stream horizontally on the screen. If a palette tab has focus, cycles between tabs. If a palette node has focus, moves between nodes in the palette.
Up/Down Arrow	If the stream canvas has focus, moves the entire stream vertically on the screen. If a palette node has focus, moves between nodes in the palette. If a subpalette has focus, moves between other subpalettes for this palette tab.
Alt+Left/Right Arrow	Moves selected nodes and comments on the stream canvas horizontally in the direction of the arrow key.
Alt+Up/Down Arrow	Moves selected nodes and comments on the stream canvas vertically in the direction of the arrow key.
Ctrl+A	Selects all nodes in a stream.
Ctrl+Q	When a node has focus, selects it and all nodes downstream, and deselects all nodes upstream.
Ctrl+W	When a selected node has focus, deselects it and all selected nodes downstream.
Ctrl+Alt+D	Duplicates a selected node.
Ctrl+Alt+L	When a model nugget is selected in the stream, opens an Insert dialog box to enable you to load a saved model from a .nod file into the stream.
Ctrl+Alt+R	Displays the Annotations tab for a selected node, enabling you to rename the node.
Ctrl+Alt+U	Creates a User Input source node.
Ctrl+Alt+C	Toggles the cache for a node on or off.
Ctrl+Alt+F	Flushes the cache for a node.
Tab	On the stream canvas, cycles through all the source nodes and comments in the current stream. On a node palette, moves between nodes in the palette. On a selected subpalette, moves to the first node in the subpalette.

Table 40. Node and Stream Shortcuts (continued)	
Shortcut Key	**Function**
Shift+Tab	Performs the same operation as Tab but in reverse order.
Ctrl+Tab	With focus on the managers pane or project pane, moves focus to the stream canvas. With focus on a node palette, moves focus between a node and its palette tab.
Any alphabetic key	With focus on a node in the current stream, gives focus and cycles to the next node whose name starts with the key pressed.
F1	Opens the Help system at a topic relevant to the focus.
F2	Starts the connection process for a node selected in the canvas. Use the Tab key to move to the required node on the canvas, and press Shift+Spacebar to finish the connection.
F3	Deletes all connections for the selected node on the canvas.
F6	Moves focus between the managers pane, project pane and node palettes.
F10	Opens the File menu.
Shift+F10	Opens the pop-up menu for the node or stream.
Delete	Deletes a selected node from the canvas.
Esc	Closes a pop-up menu or dialog box.
Ctrl+Alt+X	Expands a SuperNode.
Ctrl+Alt+Z	Zooms in on a SuperNode.
Ctrl+Alt+Shift+Z	Zooms out of a SuperNode.
Ctrl+E	With focus in the stream canvas, this runs the current stream.

A number of standard shortcut keys are also used in IBM SPSS Modeler, such as Ctrl+C to copy. See the topic "Using shortcut keys" on page 18 for more information.

Shortcuts for Dialog Boxes and Tables

Several shortcut and screen reader keys are helpful when you are working with dialog boxes, tables, and tables in dialog boxes. A complete list of special keyboard and screen reader shortcuts follows.

Table 41. Dialog Box and Expression Builder Shortcuts	
Shortcut Key	**Function**
Alt+4	Used to dismiss all open dialog boxes or output windows. Output can be retrieved from the Outputs tab in the managers pane.
Ctrl+End	With focus on any control in the Expression Builder, this will move the insertion point to the end of the expression.
Ctrl+1	In the Expression Builder, moves focus to the expression edit control.
Ctrl+2	In the Expression Builder, moves focus to the function list.
Ctrl+3	In the Expression Builder, moves focus to the field list.

Table Shortcuts

Table shortcuts are used for output tables as well as table controls in dialog boxes for nodes such as Type, Filter, and Merge. Typically, you will use the Tab key to move between table cells and Ctrl+Tab to leave the table control. *Note*: Occasionally, a screen reader may not immediately begin reading the contents of a cell. Pressing the arrow keys once or twice will reset the software and start the speech.

Table 42. Table Shortcuts	
Shortcut Key	**Function**
Ctrl+W	For tables, reads the short description of the selected ro**W**. For example, "Selected row 2 values are sex, flag, m/f, etc."
Ctrl+Alt+W	For tables, reads the long description of the selected ro**W**. For example, "Selected row 2 values are field = sex, type = flag, sex = m/f, etc."
Ctrl+D	For tables, reads the short **D**escription of the selected area. For example, "Selection is one row by six columns."
Ctrl+Alt+D	For tables, provides the long **D**escription of the selected area. For example, "Selection is one row by six columns. Selected columns are Field, Type, Missing. Selected row is 1."
Ctrl+T	For tables, provides a short description of the selected columns. For example, "Fields, Type, Missing."
Ctrl+Alt+T	For tables, provides a long description of the selected columns. For example, "Selected columns are Fields, Type, Missing."
Ctrl+R	For tables, provides the number of **R**ecords in the table.
Ctrl+Alt+R	For tables, provides the number of **R**ecords in the table as well as column names.
Ctrl+I	For tables, reads the cell **I**nformation, or contents, for the cell that has focus.
Ctrl+Alt+I	For tables, reads the long description of cell **I**nformation (column name and contents of the cell) for the cell that has focus.
Ctrl+G	For tables, provides short **G**eneral selection information.
Ctrl+Alt+G	For tables, provides long **G**eneral selection information.
Ctrl+Q	For tables, provides a **Q**uick toggle of the table cells. Ctrl+Q reads long descriptions, such as "Sex=Female," as you move through the table using the arrow keys. Selecting Ctrl+Q again will toggle to short descriptions (cell contents).
F8	For tables, when the focus is the table, sets the focus to the column header.
Spacebar	For tables, when the focus is the column header, enables column sorting.

Shortcuts for Comments

When working with on-screen comments, you can use the following shortcuts.

Table 43. Comment Shortcuts	
Shortcut Key	**Function**
Alt+C	Toggles the show/hide comment feature.

Table 43. Comment Shortcuts (continued)	
Shortcut Key	**Function**
Alt+M	Inserts a new comment if comments are currently displayed; shows comments if they are currently hidden.
Tab	On the stream canvas, cycles through all the source nodes and comments in the current stream.
Enter	When a comment has focus, indicates the start of editing.
Alt+Enter or Ctrl+Tab	Ends editing and saves editing changes.
Esc	Cancels editing. Changes made during editing are lost.
Alt+Shift+Up Arrow	Reduces the height of the text area by one grid cell (or one pixel) if snap-to-grid is on (or off).
Alt+Shift+Down Arrow	Increases the height of the text area by one grid cell (or one pixel) if snap-to-grid is on (or off).
Alt+Shift+Left Arrow	Reduces the width of the text area by one grid cell (or one pixel) if snap-to-grid is on (or off).
Alt+Shift+Right Arrow	Increases the width of the text area by one grid cell (or one pixel) if snap-to-grid is on (or off).

Shortcuts for Cluster Viewer and Model Viewer

Shortcut keys are available for navigating around the Cluster Viewer and Model Viewer windows.

Table 44. General Shortcuts - Cluster Viewer and Model Viewer	
Shortcut Key	**Function**
Tab	Moves focus to the next screen control.
Shift+Tab	Moves focus to the previous screen control.
Down Arrow	If a drop-down list has focus, opens the list or moves to the next item on the list. If a menu has focus, moves to the next item on the menu. If a thumbnail graph has focus, moves to the next one in the set (or to the first one if the last thumbnail has focus).
Up Arrow	If a drop-down list is open, moves to the previous item on the list. If a menu has focus, moves to the previous item on the menu. If a thumbnail graph has focus, moves to the previous one in the set (or to the last one if the first thumbnail has focus).
Enter	Closes an open drop-down list, or makes a selection on an open menu.
F6	Toggles focus between the left- and right-hand panes of the window.

Table 44. General Shortcuts - Cluster Viewer and Model Viewer (continued)

Shortcut Key	Function
Left and Right Arrows	If a tab has focus, moves to the previous or next tab. If a menu has focus, moves to the previous or next menu.
Alt+*letter*	Selects the button or menu having this letter underlined in its name.
Esc	Closes an open menu or drop-down list.

Cluster Viewer only

The Cluster Viewer has a Clusters view that contains a cluster-by-features grid.

To choose the Clusters view instead of the Model Summary view:

1. Press Tab repeatedly until the **View** button is selected.
2. Press Down Arrow twice to select **Clusters**.

   From here you can select an individual cell within the grid:
3. Press Tab repeatedly until you arrive at the last icon in the visualization toolbar.

Figure 19. Show Visualization Tree icon

4. Press Tab once more, then Spacebar, then an arrow key.

The following keyboard shortcuts are now available:

Table 45. Cluster Viewer Shortcuts

Shortcut Key	Function
Arrow key	Moves focus between individual cells in the grid. The cell distribution display in the right-hand pane changes as the focus moves.
Ctrl+, (comma)	Selects or deselects the entire column in the grid in which a cell has focus. To add a column to the selection, use the arrow keys to navigate to a cell in that column and press Ctrl+, again.
Tab	Moves focus out of the grid and onto the next screen control.
Shift+Tab	Moves focus out of the grid and back to the previous screen control.
F2	Enters edit mode (label and description cells only).
Enter	Saves editing changes and exits edit mode (label and description cells only).
Esc	Exits edit mode without saving changes (label and description cells only).

Shortcut Keys Example: Building Streams

To make the stream-building process more clear for users dependent on the keyboard or on a screen reader, following is an example of building a stream without the use of the mouse. In this example,

you will build a stream containing a Variable File node, a Derive node, and a Histogram node using the following steps:

1. **Start IBM SPSS Modeler.** When IBM SPSS Modeler first starts, focus is on the Favorites tab of the node palette.
2. **Ctrl+Down Arrow.** Moves focus from the tab itself to the body of the tab.
3. **Right Arrow.** Moves focus to the Variable File node.
4. **Spacebar.** Selects the Variable File node.
5. **Ctrl+Enter.** Adds the Variable File node to the stream canvas. This key combination also keeps selection on the Variable File node so that the next node added will be connected to it.
6. **Tab.** Moves focus back to the node palette.
7. **Right Arrow 4 times.** Moves to the Derive node.
8. **Spacebar.** Selects the Derive node.
9. **Alt+Enter.** Adds the Derive node to the canvas and moves selection to the Derive node. This node is now ready to be connected to the next added node.
10. **Tab.** Moves focus back to the node palette.
11. **Right Arrow 5 times.** Moves focus to the Histogram node in the palette.
12. **Spacebar.** Selects the Histogram node.
13. **Enter.** Adds the node to the stream and moves focus to the stream canvas.

Continue with the next example, or save the stream if you want to try the next example at a later time.

Shortcut Keys Example: Editing Nodes

In this example, you will use the stream built in the earlier example. The stream consists of a Variable File node, a Derive node, and a Histogram node. The instructions begin with focus on the third node in the stream, the Histogram node.

1. **Ctrl+Left Arrow 2 times.** Moves focus back to the Variable File node.
2. **Enter.** Opens the Variable File dialog box. Tab through to the File field and type a text file path and name to select that file. Press Ctrl+Tab to navigate to the lower part of the dialog box, tab through to the OK button and press Enter to close the dialog box.
3. **Ctrl+Right Arrow.** Gives focus to the second node, a Derive node.
4. **Enter.** Opens the Derive node dialog box. Tab through to select fields and specify derive conditions. Press Ctrl+Tab to navigate to the OK button and press Enter to close the dialog box.
5. **Ctrl+Right Arrow.** Gives focus to the third node, a Histogram node.
6. **Enter.** Opens the Histogram node dialog box. Tab through to select fields and specify graph options. For drop-down lists, press Down Arrow to open the list and to highlight a list item, then press Enter to select the list item. Tab through to the OK button and press Enter to close the dialog box.

At this point, you can add additional nodes or run the current stream. Keep in mind the following tips when you are building streams:

- When manually connecting nodes, use F2 to create the start point of a connection, tab to move to the end point, then use Shift+Spacebar to finalize the connection.
- Use F3 to destroy all connections for a selected node in the canvas.
- Once you have created a stream, use Ctrl+E to run the current stream.

A complete list of shortcut keys is available. See the topic "Shortcuts for navigating the main window" on page 243 for more information.

Using a Screen Reader

A number of screen readers are available on the market. IBM SPSS Modeler is configured to support JAWS for Windows using the Java Access Bridge, which is installed along with IBM SPSS Modeler. If you have JAWS installed, simply launch JAWS before launching IBM SPSS Modeler to use this product.

Note: We recommend that you have at least 6GB space to run JAWS with SPSS Modeler.

Due to the nature of IBM SPSS Modeler's unique graphical representation of the data mining process, charts and graphs are optimally used visually. It is possible, however, for you to understand and make decisions based on output and models viewed textually using a screen reader.

Note: With 64-bit client machines, some assistive technology features do not work. This is because the Java Access Bridge is not designed for 64-bit operation.

Using the IBM SPSS Modeler Dictionary File

An IBM SPSS Modeler dictionary file (*Awt.JDF*) is available for inclusion with JAWS. To use this file:

1. Navigate to the */accessibility* subdirectory of your IBM SPSS Modeler installation and copy the dictionary file (*Awt.JDF*).
2. Copy it to the directory with your JAWS scripts.

You may already have a file named *Awt.JDF* on your machine if you have other JAVA applications running. In this case, you may not be able to use this dictionary file without manually editing the dictionary file.

Using a Screen Reader with HTML Output

When viewing output displayed as HTML within IBM SPSS Modeler using a screen reader, you may encounter some difficulties. A number of types of output are affected, including:

- Output viewed on the Advanced tab for Regression, Logistic Regression, and Factor/PCA nodes
- Report node output

In each of these windows or dialog boxes, there is a tool on the toolbar that can be used to launch the output into your default browser, which provides standard screen reader support. You can then use the screen reader to convey the output information.

Accessibility in the Interactive Tree Window

The standard display of a decision tree model in the Interactive Tree window may cause problems for screen readers. To access an accessible version, on the Interactive Tree menus click:

View > **Accessible Window**

This displays a view similar to the standard tree map, but one which JAWS can read correctly. You can move up, down, right, or left using the standard arrow keys. As you navigate the accessible window, the focus in the Interactive Tree window moves accordingly. Use the Spacebar to change the selection, or use Ctrl+Spacebar to extend the current selection.

Tips for use

There are several tips for making the IBM SPSS Modeler environment more accessible to you. The following are general hints when working in IBM SPSS Modeler.

- **Exiting extended text boxes.** Use Ctrl+Tab to exit extended text boxes. Note that Ctrl+Tab is also used to exit table controls.
- **Using the Tab key rather than arrow keys.** When selecting options for a dialog box, use the Tab key to move between option buttons. The arrow keys will not work in this context.
- **Drop-down lists.** In a drop-down list for dialog boxes, you can use either the Escape key or the space bar to select an item and then close the list. You can also use the Escape key to close drop-down lists that do not close when you have tabbed to another control.

- **Execution status**. When you are running a stream on a large database, JAWS can lag behind in reading the stream status to you. Press the Ctrl key periodically to update the status reporting.
- **Using the node palettes.** When you first enter a tab of the node palettes, JAWS will sometimes read "groupbox" instead of the name of the node. In this case, you can use Ctrl+Right Arrow and then Ctrl+Left Arrow to reset the screen reader and hear the node name.
- **Reading menus.** Occasionally, when you are first opening a menu, JAWS may not read the first menu item. If you suspect that this may have happened, use the Down Arrow and then the Up Arrow to hear the first item in the menu.
- **Cascaded menus.** JAWS does not read the first level of a cascaded menu. If you hear a break in speaking while moving through a menu, press the Right Arrow key to hear the child menu items.

Additionally, if you have IBM SPSS Modeler Text Analytics installed, the following tips can make the interactive workbench interface more accessible to you.

- **Entering dialog boxes.** You may need to press the Tab key to put the focus on the first control upon entering a dialog box.
- **Exiting extended text boxes.** Use Ctrl+Tab to exit extended text boxes and move to the next control. Note that Ctrl+Tab is also used to exit table controls.
- **Typing the first letter to find element in tree list.** When looking for an element in the categories pane, extracted results pane, or library tree, you can type the first letter of the element when the pane has the focus. This will select the next occurrence of an element beginning with the letter you entered.
- **Drop-down lists.** In a drop-down list for dialog boxes, you can use the space bar to select an item and then close the list.

Interference with Other Software

When testing IBM SPSS Modeler with screen readers, such as JAWS, our development team discovered that the use of a Systems Management Server (SMS) within your organization may interfere with JAWS' ability to read Java-based applications, such as IBM SPSS Modeler. Disabling SMS will correct this situation. Visit the Microsoft website for more information on SMS.

JAWS and Java

Different versions of JAWS provide varying levels of support for Java-based software applications. Although IBM SPSS Modeler will work with all recent versions of JAWS, some versions may have minor problems when used with Java-based systems. Visit the JAWS for Windows website at http://www.FreedomScientific.com.

Using Graphs in IBM SPSS Modeler

Visual displays of information, such as histograms, evaluation charts, multiplots, and scatterplots, are difficult to interpret with a screen reader. Please note, however, that web graphs and distributions can be viewed using the textual summary available from the output window.

Chapter 19. Unicode support

Unicode Support in IBM SPSS Modeler

IBM SPSS Modeler is fully Unicode-enabled for both IBM SPSS Modeler and IBM SPSS Modeler Server. This makes it possible to exchange data with other applications that support Unicode, including multi-language databases, without any loss of information that might be caused by conversion to or from a locale-specific encoding scheme.

- IBM SPSS Modeler stores Unicode data internally and can read and write multi-language data stored as Unicode in databases without loss.

- IBM SPSS Modeler can read and write UTF-8 encoded text files. Text file import and export will default to the locale-encoding but support UTF-8 as an alternative. This setting can be specified in the file import and export nodes, or the default encoding can be changed in the stream properties dialog box. See the topic "Setting general options for streams" on page 37 for more information.

- Statistics, SAS, and Text data files stored in the locale-encoding will be converted to UTF-8 on import and back again on export. When writing to any file, if there are Unicode characters that do not exist in the locale character set, they will be substituted and a warning will be displayed. This should occur only where the data has been imported from a data source that supports Unicode (a database or UTF-8 text file) and that contains characters from a different locale or from multiple locales or character sets.

- IBM SPSS Modeler Solution Publisher images are UTF-8 encoded and are truly portable between platforms and locales.

About Unicode

The goal of the Unicode standard is to provide a consistent way to encode multilingual text so that it can be easily shared across borders, locales, and applications. The Unicode Standard, now at version 4.0.1, defines a character set that is a superset of all of the character sets in common use in the world today and assigns to each character a unique name and code point. The characters and their code points are identical to those of the Universal Character Set (UCS) defined by ISO-10646. For more information, see the Unicode Home Page .

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you provide in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
US

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBMproducts. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to actual people or business enterprises is entirely coincidental.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Terms and conditions for product documentation

Permissions for the use of these publications are granted subject to the following terms and conditions.

Applicability

These terms and conditions are in addition to any terms of use for the IBM website.

Personal use

You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative work of these publications, or any portion thereof, without the express consent of IBM.

Commercial use

You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Rights

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Index

Special Characters

@BLANK function 139, 164, 191
@DIFF function 184
@FIELD function 139, 192
@FIELDS_BETWEEN function 139, 146, 192
@FIELDS_MATCHING function 139, 146, 192
@INDEX function 184
@LAST_NON_BLANK function 184, 191
@MAX function 184
@MEAN function 184
@MIN function 184
@MULTI_RESPONSE_SET function 147, 192
@NULL function 139, 164, 191
@OFFSET function
    performance considerations 240
@PARTITION_FIELD function 192
@PREDICTED function 192
@SDEV function 184
@SINCE function 184
@SUM function 184
@TARGET function 192
@TESTING_PARTITION function 192
@THIS function 184
@TODAY function 179
@TRAINING_PARTITION function 192
@VALIDATION_PARTITION function 192

Numerics

3-D bar charts 63
3-D scatter plots 91
3D charts 62
508 compliance 241

A

abs function 168
accessibility
    example 248, 249
    features in IBM SPSS Modeler 241
    tips in IBM SPSS Modeler 250
adding
    to a project 218
adding group labels 121
adding IBM SPSS Modeler Server connections 9
Aggregate node
    performance 238
alignment
    output 106, 132
allbutfirst function 173
allbutlast function 173
alphabefore function 173
alternating row colors
    pivot tables 126
and operator 168
annotating

annotating *(continued)*
    nodes 49, 53
    streams 49, 53
annotations
    converting to comments 53
    folder 220
    project 220
application examples 3
applications 21
applications of data mining 22
arccos function 170
arccosh function 170
arcsin function 170
arcsinh function 170
arctan function 170
arctan2 function 170
arctanh function 170
area
    spatial functions 171
area function 171
attribute 21
automation 141

B

background color 127
backslash character in CLEM expressions 158
backup stream files
    restoring 54
bar charts 63
Binning node
    performance 238
bitwise functions 171
blank handling
    CLEM functions 191
blanks 135, 145
BMP files
    exporting charts 111, 116
borders
    displaying hidden borders 130
box plots 65
branches, modeling and scoring 49, 208, 209
bubble charts 66
build rule node
    loading 56

C

cache
    enabling 226
    flushing 34, 37
    saving 34
    setting up a cache 34
cache file node
    loading 56
candlestick charts 67
canvas 12

degrees
    measurements units 39
deleting output 106
dendrogram charts 69
deployment 195
deployment options 207
deployment type 207
dialog boxes
    displaying variable labels 132
    displaying variable names 132
    variable display order 132
dictionary file 250
DIFF function 184
directory
    default 226
disable nodes 33, 34
display formats
    currency 39
    decimal places 39
    geospatial coordinates 42
    grouping symbol 39
    numbers 39
    scientific 39
display order 120, 121
distance
    spatial functions 171
distance function 171
Distinct node
    performance 238
distribution functions 170
div function 168
documentation 3
domain name (Windows)
    IBM SPSS Modeler Server 8
drop-line charts 91, 92
DTD 213
dual y-axes charts 70
dual-axes charts 70

E

enable nodes 33
encoding 37, 253
endstring function 173
EPS files
    exporting charts 111, 117
equals operator 166
error bar charts 63, 71
error in rendering view
    insufficient memory 225
error messages 43
essential fields 56, 57
Evaluation charts 73
Evaluation node
    performance 238
examples
    Applications Guide 3
    overview 4
Excel format
    exporting output 111, 113
    exporting workbook 111
execution times, viewing 43
exponential function 168
exporting

exporting *(continued)*
    PMML 213, 214
    stream descriptions 48
exporting charts 111, 116, 117
exporting output
    Excel format 111, 113
    HTML 111
    HTML format 111
    PDF format 111, 114
    PowerPoint format 111
    web report 112
    Word format 111, 113
exporting workbook
    Excel format 111
    HTML format 111
    PDF format 111
    PowerPoint format 111
    Word format 111
Expression Builder
    accessing 148
    overview 148
    using 149
expressions 157

F

f distribution
    probability functions 170
factor 250
fast pivot tables 133
Feature Selection node
    missing values 136
fields
    in CLEM expressions 152
    viewing values 152
files
    adding a text file to the Viewer 108
filler node
    missing values 139
find and replace
    Viewer documents 108
first_index function 147, 166
first_non_null function 147, 166
first_non_null_index function 147, 166
folders, IBM SPSS Collaboration and Deployment Services
Repository 203, 204
fonts
    in the outline pane 107
footers 117
footnotes
    markers 126
    renumbering 129
fracof function 168
frequency polygons 76
functions
    @BLANK 139
    @FIELD 148, 192
    @GLOBAL_MAX 190
    @GLOBAL_MEAN 190
    @GLOBAL_MIN 190
    @GLOBAL_SDEV 190
    @GLOBAL_SUM 190
    @PARTITION 192
    @PREDICTED 148, 192

U

undef function 191
undo 15
Unicode support 253
unicode_char function 173
unicode_value function 173
unlocking IBM SPSS Collaboration and Deployment Services
Repository objects 203
unmapping fields 56
uppertolower function 173
URL
    IBM SPSS Analytic Server 10
user ID
    IBM SPSS Modeler Server 8
user options 226
user-defined functions (UDFs) 149, 150
UTF-8 encoding 37, 253

V

value labels
    in outline pane 133
    in pivot tables 133
value_at function 147, 166
values
    adding to CLEM expressions 152
    viewing from a data audit 152
variable labels
    in dialog boxes 132
    in outline pane 133
    in pivot tables 133
variable names
    in dialog boxes 132
variables
    display order in dialog boxes 132
version labels, IBM SPSS Collaboration and Deployment
Services Repository object 205
vertical label text 121
Viewer
    changing outline font 107
    changing outline levels 107
    changing outline sizes 107
    collapsing outline 107
    deleting output 106
    display options 132
    displaying data values 133
    displaying value labels 133
    displaying variable labels 133
    displaying variable names 133
    expanding outline 107
    find and replace information 108
    hiding results 105
    outline 106
    outline pane 105
    results pane 105
    saving document 118
    search and replace information 108
    space between output items 118
visual programming 11
visualization 100

W

warnings
    setting options 226
web report
    exporting output 112
welcome dialog box 228
white space
    removing from strings 145, 173
wide tables
    pasting into Microsoft Word 109
within
    spatial functions 171
within function 171
word cloud charts 100
Word format
    exporting output 111, 113
    exporting workbook 111
    wide tables 111
workbook
    exporting 111
Workbook tab
    moving output 106
wrapping
    controlling column width for wrapped text 125

Z

zooming 15