# ggobi manual

Deborah F. Swayne, AT&T Labs – Research
Dianne Cook, Iowa State University
Andreas Buja, AT&T Labs – Research
Duncan Temple Lang, Lucent Bell Labs

January 2001

**Abstract**

The ggobi software is a data visualization system with state-of-the-art interactive and dynamic methods for the manipulation of views of data. It represents a significant improvement on its precedessor, XGobi, with multiple plotting windows, more flexible color management, XML file handling, and better portability to Windows.

The most significant change may be that ggobi can be embedded in other software and controlled using an API (application programming interface). This design has been developed and tested in partnership with R. When ggobi is used with R, the result is a full marriage between ggobi's direct manipulation graphical environment and R's familiar extensible environment for statistical data analysis.

It has the same graphical functionality whether it is running standalone or embedded in other software. That functionality includes 2-D displays of projections of points and lines in high-dimensional spaces, as well as scatterplot matrices, parallel coordinate and time series plots. Projection tools include average shifted histograms of single variables, plots of pairs of variables, and grand tours of multiple variables. Views of the data can be reshaped. Points can be labeled and brushed with glyphs and colors. Several displays can be open simultaneously and linked for labeling and brushing. Missing data are accommodated and their patterns can be examined.

## 1 Introduction

This paper gives an overview of the layout and functionality of ggobi, interactive graphical software for exploratory data analysis. Readers who are familiar with xgobi will find much that is familiar in ggobi's design, and might want to read section 12 first, where key differences between the two programs are described.

You will find that you can use ggobi for simple tasks with virtually no instruction. All that a user needs is some cursory knowledge of the developments in interactive statistical graphics of the last 15 years, as well as a willingness to experiment with the sample data provided guided by the tooltips. In parallel with the hands-on learning process, it is probably useful to acquire a basic understanding of the overall layout and functionality of the system. The greatest success is obtained by users who have gained experience with the system and combine it with creativity and data analytic sophistication.

We begin with a tutorial, and move on to describe ggobi in detail.

## 2 Tutorial

Several sample data files are included with the ggobi distribution, in a directory called **data,** and there you will find the file **pigs.xml**, a dataset on pig production in the United Kingdom, taken from *Data* by Andrews and Herzberg. Start ggobi for the pig production data by typing:

```
ggobi pigs
```

Two windows will appear, the ggobi control panel and a scatterplot, as shown in figure 1.

The control panel has a panel of controls on the left, labeled **XYPlot**, and a variable selection region on the right. You can see that the scatterplot contains a 2-dimensional projection of the data, a plot of YEAR vs TIME. Move the mouse to one of the variable labels in the variable selection region, and leave it in place until the tooltip appears, explaining how to select new variables for the plot. Begin to get a feeling for the data by looking at several of the 2d plots: PROFIT vs HERDSZ, HERDSZ vs TIME.

Some of the variables are not self-explanatory, and here's what they mean:

- GILTS: number of sows "in pig" (reproducing) for the first time
- PROFIT: ratio of price to an index of feed price
- S/HERDSZ: ratio of the number of breeding pigs slaughtered to the total breeding herd size
- PRODUCTION: number of pigs slaughtered that were reared for meat

Next, get acquainted with the main menubar for the control panel by exploring each of its menus. Pay particular attention to the Display, ViewMode and Tools menus.

- The Display menu is the interface for opening new plotting windows.
- The ViewMode menu is the interface for specifying both the projection (1d, 2d, 3d or higher) and mouse interactions (for scaling the plot, highlighting points, and so on) for the current plot.
- The Tools menu lets you open other windows to manipulate characteristics of the data and the view.

Using the Display menu, open another scatterplot display. Notice that the new window has a narrow white band drawn around the outside of the plotting area: that means that the new window is now the "current display" and its plot is the "current plot." Click in the plotting region of the other scatterplot to make it the current plot, and notice what happens in the variable selection region of the control panel when you alternate between the two: it should always show the variables that are plotted in the current plot.

Figure 1: Layout of a ggobi session. The plotting window contains a scatterplot of the PROFIT vs GILTS, from the pig production data.

Now set up a plot of GILTS vs HERDSZ in the first scatterplot, and PROFIT vs HERDSZ in the second, and make that the current plot. Using the ViewMode menu, choose **Brush.** Look at the buttons and menus inside the leftmost portion of the ggobi control panel. Notice that they're contained by a frame labelled **Brush,** which is the ViewMode of the current plot. This frame contains most of the brushing controls, which are described in the section 6.7. A few of the brushing controls, though, are in the main menubar: in the Reset menu and at the bottom of the Options menu.

The pink rectangle that appears in the current plot is the "paintbrush," and dragging it over points changes their color. Use the left button to drag it and the middle or right button to resize it. Make it almost as wide as the window and about a quarter as tall, and paint first the profitable herds, and then the unprofitable ones. While you do that, keep an eye on the of GILTS vs HERDSZ, and notice where the painted points fall in that scatterplot. Using two linked 2d plots is one way to explore the relationship among three variables.

Next use the Tools menu to open the **Variable manipulation tool**, a central part of the ggobi graphical user interface. It contains a table with a few basic statistics for each variable, and the buttons below the table allow you to set variable limits, add new variables, and a few other things. Try clicking the mouse in the table – you can select one row of the table at a time, or use the control and shift keys as modifiers to select more than one. Hold the control key down while clicking to select variables that are not contiguous in the table; use the shift key if you want to select a contiguous range of variables.

You select variables in this fashion when you want to use one of the tools to operate on them. For example, select just the GILTS variable, and then click on **Clone** at the bottom of the tool. You have just added a duplicate of the GILTS variable to the dataset, and you'll see it at the bottom of the table in this tool, and in the variable selection region of the ggobi control panel, too. Select that new GILTS variable in the **Variable manipulation tool**, then return to the Tools menu and open the **Variable transformation tool.**

[etc, do a transformation]

# 3 Layout and functionality

## 3.1 The major functions

Across the top of the control panel, as seen in in Figure 1, stretches a row of buttons for selecting major ggobi functions from pull-down menus. Buttons such as **File**, **Display, ViewMode**, **Tools**, **Options** and **DisplayTree** organize functions in menus.

As expected, the **File** button opens a menu for selecting input/output functions as well as exiting.

The **Display** button opens a manu for opening a new plotting window. The display types are

- scatterplot,
- scatterplot matrix,
- parallel coordinates plot, and
- time series plots.

If the data has missing values, a scatterplot or scatterplot matrix of the "missingness" information can be opened. Each display type is discussed in section 3.2.2.

The **ViewMode** menu contains links to ggobi's inventory of interactive graphics operations:

- 1DPlot: 1-D dotplots and average shifted histograms,
- XYPlot: 2-D scatterplots,
- 1DTour: random 1-Dimensional tour,
- 2DTour: random 2-Dimensional tour,
- 2x1DTour: correlation tour, random and guided,
- Scale: axis scaling,
- Brush: setting point glyphs and point and edge colors,
- Identify: labelling points
- Move Points: direct manipulation of point positions.

When you choose a new view mode, the controls at the left of the main window will change correspondingly: each mode has its own parameters, and its own rules for responding to mouse actions in the plotting windows. The view modes are discussed in section 6.

The **Tools** menu gives access to

- a variable manipulation table,
- a variable transformation pipeline,
- an variable sphering panel,
- jittering controls,
- a panel for hiding groups of cases,
- subsetting functions for systematic and random subsampling,
- a tool for managing missing values.

Each tool is discussed in section 7.

There are a couple of distinctions between view modes and tools The view mode functions determine the mouse interactions in the display windows, while none of the Tools does. Furthermore, view mode functions populate the control panel in the main ggobi window, while tools create their own permanent control panels or views of the data in separate windows, as shown in figure ?. For example, the **variable manipulation tool** is a window containing a table of data each variable and several buttons, and the **variable transformation tool** is also a separate window.

The **Options** menu allows users to set some options for the main control window: whether tooltips are displayed, for instance, and whether the control panel is shown. In some view modes, it contains additional options that are specific to that mode.

Other menus are present only during certain modes: you will sometimes find an **I/O** menu or a **Reset** menu.

The **DisplayTree** menu allows users to open a tree listing all the currently open display windows, each of which may contain several plots.

## 3.2 Graphical displays

### 3.2.1 Current display, current plot

Since there are multiple displays, some of which contain multiple plots, the question arises: Which plot in which display window corresponds to the control panel? If you select the **Brush** mode, how can you tell which plot is going to respond to brushing?

There is in ggobi a notion of the "current display" and the "current plot." (We need both because some displays, like the scatterplot matrix, contain multiple plots.) The current plot is the one which is outlined with a thick white border; the current display is the one which contains the current plot.

To reset the current plot and display, just click once (left, right or middle) in the plot you wish to address. To understand the effects of this selection, open a few displays and set them in different ViewModes, then click on different plots and see what happens. The white border should follow your actions, and the control panel should update so that its panels correspond to the current display type and ViewMode.

### 3.2.2 Display types

Each display type is briefly described here. As mentioned earlier, there are presently four main display types:

The **scatterplot** display is a window containing a single scatterplot. By default, it includes axes; they can be turned off using the Options menu in the main control window. It has the largest number of view modes of any display, and each of the projection modes has its own rules for variable selection. The variable selection interface for the 1DPlot and XYPlot modes is a column of checkboxes: clicking left on one of these selects a variable to be plotted horizontally, clicking middle or right selects a vertical variable. The tour modes all use a set of labelled circles for variable selection, and they provide some feedback about current projection. The variable selection behavior for the tour modes will be described in the section for each mode.

The **scatterplot matrix** is a window containing a symmetric matrix of scatterplots for the chosen variables. The plots along the diagonal are ASHes (Average Shifted Histograms). The matrix is required to be symmetric, and that constraint affects its variable selection behavior.

- Replace: First select one of the plots along the diagonal to tell ggobi uniquely which variable to replace, then click on one of the checkboxes in the variable selection region.
- Insert (Append): First select one of the plots along the diagonal to tell ggobi uniquely where to insert (append) the new variable, then click on one of the variable checkboxes. (For now, ggobi will not add a variable that's already plotted.)
- Delete: No plot selection is required; just click on the variable you want to delete.

The **parallel coordinates** display contains a single parallel coordinates plot, which can be arranged horizontally or vertically. (To understand this plot if you are encountering it for the first time, imagine deconstructing a high-dimensional scatterplot and arranging its axes in parallel instead of orthogonally. To represent case $i$, think of drawing a dot on axis $j$ at that point's value for variable

$j$, and then connecting the dots into one set of connected line segments. For a more detailed explanation, see [].) The line segments are drawn by default, but you can turn them off using the **Options** menu on the display menubar.

By default, the plots are simple dotplots, but they can also be drawn using one of the other two methods for 1D plots: as a textured dot strip or an ASH.

The variable selection behavior works as follows:

- Replace: First select one of the plots, then click on one of the variable checkboxes to replace its plotted variable.
- Insert (Append): First select one of the plots, then click on one of the variable checkboxes to insert (append) a new plot. (For now, ggobi will not add a variable that's already plotted.)
- Delete: No plot selection is required; just click on the variable whose plot you want to delete.

The **time series display** contains a row or column of 2-variable plots with a common axis, usually a time variable. By default, the points are connected with line segments.

The time series display uses the checkbox variable selection interface, and the behavior of the checkboxes depends on the state of the **Selection mode** option menu in the control panel.

- Replace: If you want to replace the horizontal (time) variable, no plot selection is required; simply click left on the variable you want to choose. To replace a vertical variable, first select the plot you want to change, and then click right or middle on the variable you want. (For now, ggobi won't let you use a variable that's already plotted.)
- Insert (Append): Select a plot in the display, and then click middle or right on a variable checkbox. (This applies to verticallly plotted variable only.)
- Delete: No plot selection is required; just click middle or right on the variable checkbox for the variable you want to delete. (This applies to verticallly plotted variable only.)

#### 3.2.3 Missing values displays

When your data includes missing values, two additional display types are available. The missing values scatterplot and scatterplot matrix display not the data itself but the "missingness" for each point, namely a jittered scatterplot of indicator variables encoding the presence or absence of values for each variable. Examining a number of views in one of these displays allows us to explore the joint distribution of missing values across variables. Since these displays are linked to all others, brushing the missings in a missing values display allows us to explore the association between missing values and the variables.

## 4 Data format

### 4.1 ASCII

The basics of the ascii data format used in xgobi are still supported, with some changes.

The only essential file is the one containing the data itself. Each line in the file contains one row of the input data matrix, and lines must be separated by carriage returns. Columns, or variables, can be separated by any number of tabs or spaces. The file can have the suffix *.dat*, but no suffix is required, so it is named either **filename.dat** or just *filename*.

You can supply variable and case labels in associated files. Variable (column) labels can be in a file named **datafile.col** (or **datafile.column**, **datafile.collab**, or **datafile.var**). Case (row) labels can be supplied in a file named **datafile.row** (or **datafile.rowlab** or **datafile.case**). There should be one label per line, and the label can include blanks. If variable or case labels are not supplied, default labels using column or row numbers are used.

The files **datafile.glyphs** and **datafile.colors** can be used to set the plotting characters and colors to be used in drawing each point. Glyphs can be specified in two ways:

- with a string for glyph type and a number for glyph size, where the string is one of "plus", "x", "or" (open rectangle), "fr" (filled rectangle), "oc" (open circle), "fc" (filled circle), "." (a one-pixel point), and the number is between 1 and 8, or
- with one number per line, where the number is between one and 49. Here's how to generate that number: the type is between 1 and 7, using the ordering just presented, and the size is between 1 and 8 (though it must be 1 for the single-pixel glyph). The number is then $8 \times (type - 1) + size$.

### 4.2 XML

The XML format is described in *Using XML Input Formats*, which you should find where you found this manual. The XML format allows more detailed specification, such as

- multiple datasets within a single process,
- rules for linking between datasets,
- edges: line segments connecting pairs of points, and
- the colormap to be used in brushing.

### 4.3 Database access

## 5 Integration of ggobi with other software systems

## 6 View modes: projection and interaction

Selecting any mode on the **ViewMode** menu changes the interactions available for the display and plot that are current, and pops the corresponding control panel into the left portion of the main control window. The modes in the top half of the menu do something more as well: they set the projection method for the display, and the meaning of actions in the variable selection panel always conforms to the current projection method.

As an example, start ggobi with some data, and watch what happens in the main ggobi window and a a single scatterplot display. When ggobi starts, it's in **XYPlot** mode by default, so the scatterplot window shows a 2-dimensional projection of the data. If you select **Scale** or **Brush** (or any choice in the bottom half of the menu), the control panel at the left of the main window changes, reflecting the different interactions available to you in each mode. However, the projection in the window doesn't change. If you click on the checkboxes in the variable selection panel at the right of the main ggobi window, you replace one of the plotted variables.

Now select *2D Tour* in the ViewMode menu. *Everything* changes at once, because you've effectively selected both a new mode and a new projection type at the same time.

- The control panel changes, because a new set of interactions just became available.
- The variable selection panel changes, because the variable selection behavior for high-dimensional projection types is quite different than that for low-dimensional projection types.
- The plot in the scatterplot display changes, because it's now showing a projection of 3 variables instead of 2. Furthermore, it's moving, because a grand tour process is running.

If you now select one of the modes in the bottom half of the ViewMode menu, you'll see again that the variable selection panel doesn't change, and the plot in the scatterplot doesn't change – except that it stops moving. The only thing that changes with every selection is the control panel at the left of the main window.

Now we'll describe each view mode in more detail, starting at the top of the ViewMode menu.

## 6.1 1D plots

## 6.2 XY plots

## 6.3 1D Tour

## 6.4 2D Tour

## 6.5 2x1D Tour

## 6.6 Scaling of axes

The two styles of interaction, **Drag** and **Click**, are quite different. Drag-style scaling is a perfect example of a direct manipulation interface, in which the points follow cursor motion in a very simple way. However, if you're looking at a lot of data, the points may sometimes lay behind the cursor motion, making the degree of panning or zooming hard to control.

Click-style scaling may take you a few minutes to get used to, but you'll find that it gives you very precise control and is especially useful when you have a lot of data.

### 6.6.1 Drag

For the default setting, Drag, the actions of the mouse can be described in terms of a camera: you're operating a camera and looking at a projection of the data in the viewfinder. When you use

the left button, the camera is panning freely around, following the mouse exactly. When you use the middle or right button, you're zooming the camera in and out.

#### 6.6.2 Click

When you select the Click interaction style, the manipulation is not so direct, but your control of the panning and zooming becomes more precise.

With **Pan** selected, the mouse controls the endpoint of a line segment which is anchored at the center of the plot (just where the center of the crosshair is in Drag style). When you press the space bar, you'll cause the plot to pan so that the endpoint becomes the new center – ie, short segments yield small movements. Repeated presses repeat the mostion in the same direction – convenient for browsing time-dependent data, for instance.

With **Zoom** selected, the visual guide changes again: this time, the mouse controls a rectangle, and two keys are used: "¡" to zoom in and "¿" to zoom out an amount inversely proportional to the size of the rectangle – ie, large rectangles yield small movements.

#### 6.6.3 Reset

To reset the plot, use the menu marked Reset in the main menubar: it has two entries, allowing pan and zoom to be reset separately.

#### 6.6.4 Pan and zoom options

By the default, the panning and zooming of the plot is unconstrained, moving or rescaling vertically and horizontally with each action. The pan and zoom options allow it to be constrained so that only one axis is affected, convenient for browsing one variable at a time.

### 6.7 Brush: brushing of points and lines

Brushing is often performed when only a single display is visible, but it is most interesting and useful to perform brushing with more than one linked display showing different views of the same data.

To interactively paint points, drag left to move the "brush" within the plotting window, or drag middle to change the size or shape of the brush while you paint. (If you lose the brush by pulling it outside the plotting window, you can grab it again if you press the left or middle button while the cursor is inside the display window.)

#### 6.7.1 Brush On

When the **Brush on** toggle is checked, moving the brush over a plotted point causes that point to change its color or plotting character (called a glyph). If the brush is turned off, the brush can be freely moved across the plotting window and it does not change the points.

This is useful if you are plotting a very large number of points, and you want to position the brush before painting, because you can move it much more quickly across the plot. [This isn't true in the default mode, where the brush jumps to the cursor. Shall I re-add the other mode?]

### 6.7.2 Points and edges

If **Points** is selected, the brush is drawn as a rectangle. As the brush is moved across the points, any points contained by the brush are redrawn using the currently selected glyph and color. If **Edges** is selected, the brush is drawn as a crosshair, and as the brush is moved in the window, any edges (line segments) intersecting either the vertical or horizontal "hair" are redrawn in the chosen color. (There is no line brushing implementation for line types yet.) If both **Points** and **Edges** are selected, the brush is drawn as a crosshair inside a rectangle, and both point and edge brushing are performed.

### 6.7.3 Color and glyph

Use the **Color and glyph** menu to choose whether to brush with color, glyph, or both, or whether to hide the points you paint.

### 6.7.4 Brushing modes: persistent, transient

These are two brushing modes.

**Persistent**: When you brush a point, it retains its new color or glyph when the brush has moved away.
**Transient**: As the brush moves off a point, it returns to the color and glyph it had before the brush covered it.

### 6.7.5 Undo

Clicking on the **Undo** button restores the glyph and color and visibility of all points painted between the last mouse-down and mouse-up.

### 6.7.6 Options menu

When the brushing mode is active, the Options menu in the main menu bar contains this item:

**Update brushing continuously**: Update linked brushing with every mouse motion. The alternative is to update linked views only when the mouse is released.

### 6.7.7 Reset menu

When the brushing mode is active, the Reset Menu in the main menu bar contains these items:

**Show all points** Make all points visible.

| | |
|---|---|
| **Show all edges** | Make all edges visible. |
| **Reset brush size** | Reset the brush to its default size and position. |

## 6.8 Identification

This mode is used to display labels near points in the plotting window. The labels are taken from the file of row or case labels supplied by the user; if the file is not present, the row number of each case is used. To see these labels, simply move the cursor inside the plotting window. The label of the point nearest the cursor is displayed.

Identification in one window is instantly reflected in all linked windows.

To cause a label to become "sticky," click left when the target label is printed. The printing style changes and the label now remains printed as the cursor moves off, and even remains printed as you leave the **Identify** mode. It is possible to rescale or rotate data, and the sticky labels will continue to be printed next to their associated points.

To cause a label to become "unsticky," return to the **Identify** mode and click left again when the target point is nearest the cursor. It is also possible to restore all labels to unsticky status by clicking on the **Remove labels** button. You can also see all the labels at once by clicking on **Make all sticky**.

## 6.9 Line editing

# 7 Tools

## 7.1 Variable manipulation tool

This powerful tool is opened by selecting the first entry on the **Tools** menu. It has several important functions:

- the display of variable statistics,
- variable selection for other tools,
- setting variable ranges,
- cloning variables, and
- adding other new variables.

Its first purpose is to display a few statistics for each variable: the current variable transformation (if any); the minimum, maximum, mean, and median of the raw data; the number of missing values per variable.

You can also use it to select variables – not for plotting, but to be operated on by other tools. Variables are selected by highlighting rows, and the control and shift keys are modifiers that allow multiple rows to be highlighted. The variable transformation tool, among others, will operate on the variables selected in this way.

These selected variables will also respond to operations contained within this panel: you can reset the variable ranges that are used for projecting the data into the plotting window. This is especially useful for a data set where several of the variables have the same units.

The selected variables can also be cloned, and the new variables you create will be added to the table as well as the main control panel.

There's another way to add new variables, and that relies on the *New ...* button, which brings up a small panel. Use that panel to specify the variable's name and to set its values: either the row numbers or a set of integers reflecting the groups currently defined by brushing.

## 7.2 Variable transformation tool

The first step in variable transformation is to specify the variables you want to transform. By default, the variables plotted in the current display will be affected; if you wish to select others, use the **Variable manipulation tool.**

There are three stages in the transformation pipeline, with a transformation function in each stage operating on the output of the previous stage. It's equally acceptable to use any or all of them.

You can think of stage 0 as a domain adjustment stage: if a variable has negative values, for instance, many transformation functions can't be applied to it, so you may need to add an increment to each value. The transformations in the next two stages don't have such neat definitions. Stage 1 transformations include the Box-Cox transformation, and you can either type the Box-Cox parameter into the text box and hit return, or use the spin button to gradually increase or decrease the parameter. Many of the stage 2 transformations are not linear; they include sorting and ranking.

## 7.3 Jittering

Select *Variable jittering ...* to open a panel that allows random noise to be added to selected variables.

First specify the variables you wish to jitter. By default, the variables plotted in the current display will be affected; if you wish to select others, use the **Variable manipulation tool.**

Choose between uniform and random jitter, and then set the degree of jitter using the slider. To rejitter without changing the degree of jitter, simply click on the **Jitter** button.

## 7.4 Controls for missing data

By default, missing values are assigned the value 0, but you may sometimes find that to be an inconvenient choice. Use the *Missing values* panel to assign alternative numbers.

By default, all missings in all variables will be affected together, but you can choose to assign values only for selected variables using the menu at the top of the panel.

Below the menu, a notebook widget allows you to specify the type of imputation you'd like to use.

**Random imputation**: Sample from the present values for each variable to populate the missing values. If you have done some brushing to partition the cases, you can specify that you want the sampling to be done using only cases brushed with the same color and glyph.
**Fixed value**: Specify any value to use instead of the default.
**Percentage below minimum**: Specify a value that is $x$ percent above the minimum value. For example, if the variable ranges from 40 to 80, specifying 10 will assign the missings the value $40 - (10\% * 40) = 36$.
**Percentage above minimum**: Specify a value that is $x$ percent above the maximum.

If you want all plots to rescale immediately when you assign new values, turn on *Rescale* toggle. Finally, click on the *Impute* button to perform the imputation or assignment of values.

## 7.5 Sphering

The *Sphering* panel starts by displaying a scree plot for the currently selected variables (see section 7.1). If the scree plot doesn't reflect the variables you want to sphere, then open the Variable statistics panel, select the variables of interest, and click on *Update scree plot.*

It's advisable to standardize the variables first, so a label at the top of the panel reminds you if they aren't. In that case, open the *Variable transformation* tool and standardize the selected variables, then click *Update scree plot.*

Now you're ready to sphere the selected variables. Working your way down the panel, use your visual interpretation of the scree plot together with the information in the labelled section "Prepare to sphere" to decide how many principal components you want to create. By default, all the selected variables will be sphered, but you decide that the first few principal components account for a sufficiently high proportion of the variance. In that case, you can use the spin button to the right of the label "Set number of PCs" to decrease the number of principal components you're going to generate. The variance and condition number are displayed to help you make that choice.

Once you're satisfied with the selected variables and the number of principal components, proceed to the last step. Click on *Apply sphering* to create new variables and add them ggobi's variable selection panels. The names of the selected variables will be added to the "sphered variables" to help you remember which variables you sphered.

[And I forget what 'Restore scree plot' is for – dfs]

### 7.6 Subsetting

Select *Case subsetting and sampling ...* to open a panel that allows subsets to be specified in one of five ways.

**Random sample without replacement**: Specify the number of cases to be in the sample.
**Consecutive block**: Specify the first and last row of the block.
**Every nth case**: Specify the interval and the first row.
**Sticky labels**: All cases with a "sticky" label will be in the subset. (See *Identification* for a description of sticky labels.)
**Row labels**: Type in a row label, and all cases with that label will be in the subset.

Select one of those five, then click on **Subset** in the bottom row of the panel. If you want to re-include all rows, select the **Include all** button in the bottom row.

If the **Rescale** button is checked, then the plots will be rescaled to exclude all points not in the subset. If it is not checked, the points will be hidden but not excluded from plot scaling operations.

One purpose of subsetting is to allow the use of ggobi on data matrices that are so large that dynamic and interactive operations begin to become painfully slow. By selecting a smaller subset, a user can work on that subset at a comfortable speed of rotation and interaction. Another purpose is to do graphical cross-validation: if the feature you see is still there in repeated subsamples, there's a good chance it's not just an artifact of visualization.

## 8 Multiple datasets

## 9 Linking

If two display windows represent the same dataset, then points which represent the same case are linked: that is, they are drawn with the same symbol (color and glyph), and if a case is labelled in one display, it will be identically labelled in all displays.

If the ggobi process has multiple datasets, then the linking rules have been defined in the XML file (or through the API).

1. If you have specified a categorical variable to use for linking, then all datasets with a variable of the same name are affected, and all cases which have the same level of that variable are linked. (The labels for the levels aren't used in that determination, just the numerical representation of the level.)

2. If you are not using a categorical variable for linking, then cases are linked by **id**: if two cases share the same **id**, then they are linked. Ids are unique within a dataset. If you have not specifically

assigned an id to a case, then it doesn't have one.

This is described in more detail in the XML documentation.

# 10 Edges

# 11 Implementation

# 12 Differences from xgobi

In this section, we summarize the key differences between ggobi and xgobi for those readers who are already familiar with xgobi.

## 12.1 Multiple displays

The first thing you'll notice when you look at a ggobi display is that the plotting window has become separated from the control panels. The main reason for that change is so that a ggobi process can have multiple display windows, of the same type or of different types. In addition to the basic scatterplot, ggobi currently has scatterplot matrices, parallel coordinates plots, and time series plots. (See 3.2 for more detail.)

This design change has had far-reaching effects.

First, user interactions available for the simple scatterplot display can now be made available for other display types. In xgobi, for instance, there is a parallel coordinates display, but it's not possible to brush it – in ggobi, it is.

Unfortunately, having multiple displays introduces a new source of ambiguity: you now have to tell ggobi which display, and which plot within a display, you want to address. Do that by simply clicking inside the target plot. ggobi will draw a thick white outline around that plot so that you can check which plot your actions will be addressing. The ViewMode panel in the main ggobi window should always correspond to the state of the current display, too. We need more user experience before we can tell whether this approach is satisfactory.

The basis for linking has changed, too: since all displays are linked by default, it's no longer necessary to run multiple processes in order to achieve linked displays.

One of the most interesting implications of using multiple displays is that a ggobi process is no longer restricted to a single data set. The XML file format makes it easy to specify two or more data matrices in a single file, as described in section 4.2. In addition, it's possible to add data matrices using the Read button on the File menu or using R. (The R - ggobi interface will be introduced below, and it's more fully described in section 5.)

The rules and the interface for linking across data sets have not yet been defined.

With multiple displays, too, we no longer have to launch a new process to open missing value plots – the plots of 1's and 0's which represent the presence and absence of data in each cell.

A convenient side effect of multiple displays is that, since each display now sits in a window of its own, it's now very simple to adjust the aspect ratio of a plot; this simple operation is very awkward in xgobi.

## 12.2 Data format

Before we describe other key changes in the visible design, we'll introduce the changes in data format. The xgobi format, in which a set of files with a common base name is used, is still supported, though some file formats have changed (*.colors*), and some are no longer supported (*.bin*, *.vgroups*, *.lines*). (See section 4 for more detail.)

The new format, in which all the data lives in one file, is written in XML. XML (Extensible Markup Language) is a widely used language for specifying structured documents and data to be viewed and exchanged. It was initially intended to be read by browsers, but it is also used to define documents that are read by other software. XML files can be validated automatically, and XML specifications can be easily extended, too, by adding to the set of tags in use.

As ggobi grows and develops, we expect to favor the XML format: it will become increasingly difficult to keep the old format up to date. For instance, it's no longer possible to specify line segments in the ascii data format, but they can be specified in XML – and they can have associated data values.

A few XML data files are included in the sample ggobi data, and the details of their format is described in ??.

Another innovation in data access is the ability to read data directly from a MySQL database, as described in section 4.3.

## 12.3 Integration

Many xgobi users are also users of the SPlus or R statistics software, and have used the S function which launches an xgobi process viewing S data. Once that launch occurred, the resulting process was utterly independent of its parent. The xgobi authors did some experiments in the early 90s to achieve a more intimate connection, but made little headway. (reference)

With ggobi, that problem has been solved, and it's now possible to have real-time integration between ggobi and a variety of other software environments. An example of this integration is the embedding of ggobi into R (or S), with the addition of a set of R (or S) functions that manipulate ggobi data and displays, resetting data values, projection, and the appearance of the plot.

For more details, see section 5, or read (reference).

## 12.4 Variable selection

There have been a few changes in variable selection. The familiar variable circles are still used for high-dimensional view modes, but we've switched to a simple checkbox interface for plots where only one or two variables can be selected simultaneously. In these plots, the rich feedback provided by the variable circles is not needed, and may just be confusing to novices.

The basics of the user interface haven't changed, though: click with the left button to select a variable to be plotted horizontally and the middle (or right) button to plot vertically.

## 12.5 The variable manipulation tool

The table in the **Variable manipulation tool** can also be used to select variables – not for plotting, but to be operated on by other tools. Several of the tools, such as variable transformation and jittering, will operate on the plotted variables in the current display by default. However, if any rows in this table are highlighted, then all selected variables will be affected.

This table can also be used to specify limits for variables or groups of variables, and so we have

dispensed with the *.vgroups* functionality in xgobi.

Variable cloning is another new feature: it appears as a button on the variable selection and statistics table.

For more detail on this table, see section 7.1.

## 12.6 Changes in ViewModes and Tools

**Brush** may be the **ViewMode** that has changed the most, because ggobi has a much richer notion of color selection than xgobi. Open the **Choose Symbol** panel, and then double-click on any element of the color palette to bring up a color selection widget with access to the full color map. Notice that you can change the background color as well as any of the foreground colors.

**Scale** has also changed. The direct manipulation shifting and scaling methods work as they did in xgobi, but we've added what we call "Click-style interaction" for more precise control. See section 6.6.

The tour methods in ggobi are still evolving: many things are not yet implemented, such as projection pursuit, but new things are beginning to appear. In the **1DTour**, a projection of several variables is viewed an an average shifted histogram, as described in section 6.3.

The redesign of the tour methods is reflected in the **Sphering** tool, which also appears in the most recent versions of xgobi. Instead of automatically sphering variables in projection pursuit, ggobi allows you the choice: you can use the **Sphering** tool to decide which variables to sphere. Since this method makes use of variable cloning, it creates new variables, allowing you to look at plots of principal components against the original data.

See section 7.5 for details and pictures.

## 12.7 Linking

The rules for linking in xgobi had evolved into a rather complicated hodge-podge with special handling of "row groups," the "nlinkable" notion to exclude points from linking, and linking points to line segments. This has been replaced with a single set of rules that can be specified in the xml file. See section 9 for details.

## 12.8 On-line help

The on-line help system used in xgobi has been replaced with "tooltips," so leaving the mouse over a widget for a couple of seconds brings up a phrase describing the function of that widget. If the tooltips annoy you, you can turn then off using a checkbox on the **Options** menu.