Interactive Scatterplots

Elizabeth Whalen

October 7, 2004

1 Overview

In the package *iSPlot*, the goal is to create interactive, linked scatterplots. The two required packages for *iSPlot* are *RGtk* and *gtkDevice*. *RGtk* is used to create a graphical user interface, which enables the user to load, view, plot and interact with the data. *gtkDevice* creates a device of type gtk that looks like a X11 device, but a gtk device can respond to events, such as a button press or a mouse over event. By generating a function call in response to a mouse or keyboard event, interactive scatterplots can be created.

2 Getting Started
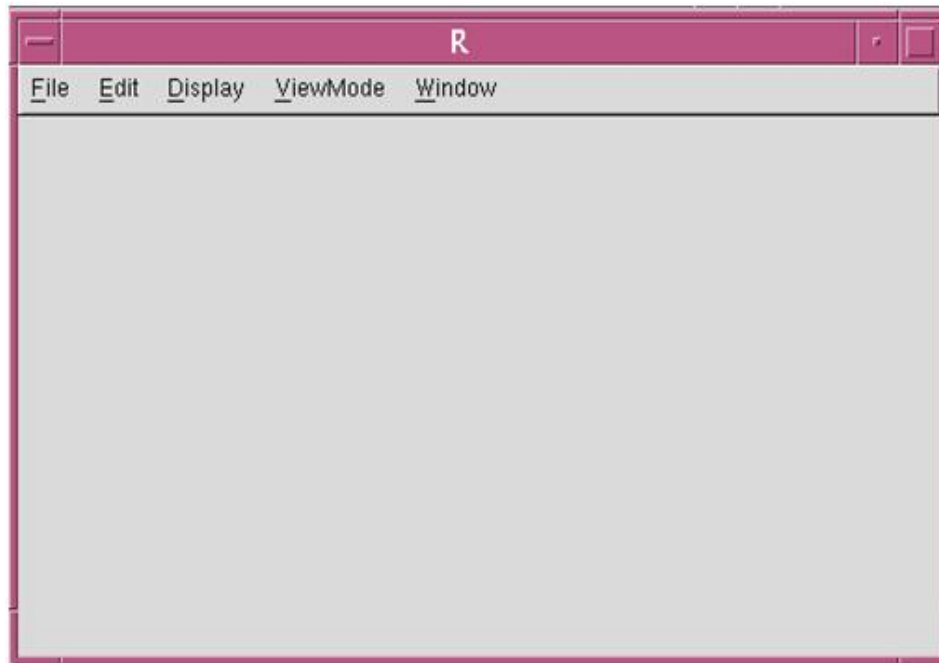
```
> library(iSPlot)

Loading required package: RGtk
Loading required package: gtkDevice
```

After loading the library, the first command will be to open the control window, using `createControlWindow`. Data can be loaded, viewed and plotted by selecting menu items on the control window.

```
> if (interactive()) {
+     createControlWindow()
+ }
```

The control window will look as follows:
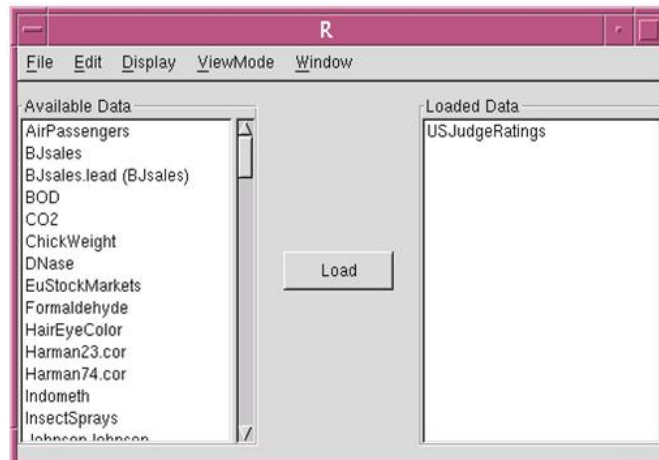
Figure 1: Control Window

3 Loading Data

Currently, this package can only load and plot either dataframes or matrices. No other class of data can be loaded at this time. Before plotting the data, it first must be loaded using either the Open Data menu item or the Open File menu item under the File menu. Either option will load the data into a data environment where the data and any updates to the data will be stored.

If the user wants to load data that is available from the function `data`, then highlight the File menu and select the Open Data menu item. Alternatively, the user can press Ctrl-D to activate the Open Data menu item.

To make the control window look as it is shown below, complete the following steps. Choose Open Data under the File menu, highlight the list item, "USJudgeRatings", by scrolling down the Available Data list, and then click the Load button.
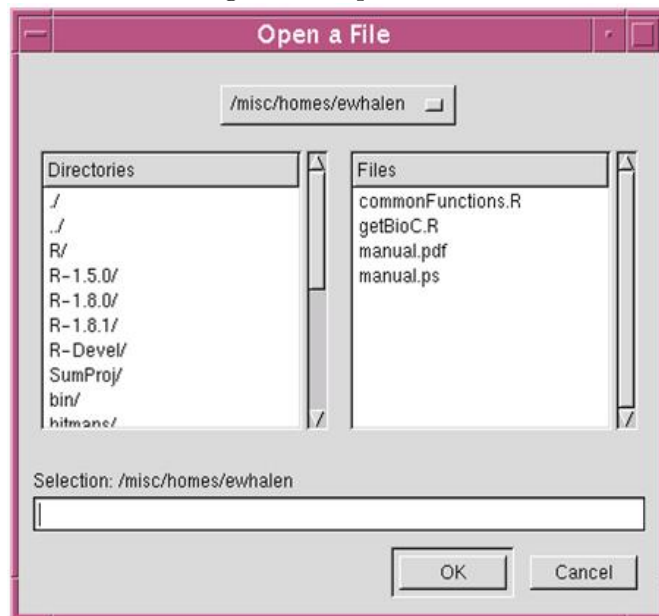
Figure 2: Loaded Data

Note that once "USJudgeRatings" data has been loaded, it will no longer appear in the Available Data list on the left hand side of the control window.

Instead, if the data the user wants to load is stored in a file, then highlight the File menu and select the Open File menu item. Similarly, the user can press Ctrl-O to activate the Open File menu item. A file explorer window will open allowing the user to choose which file they would like to load.

The file explorer window is shown below.

Figure 3: Open a File

Again, note that if the user tries to load data that is not of type dataframe or matrix, it
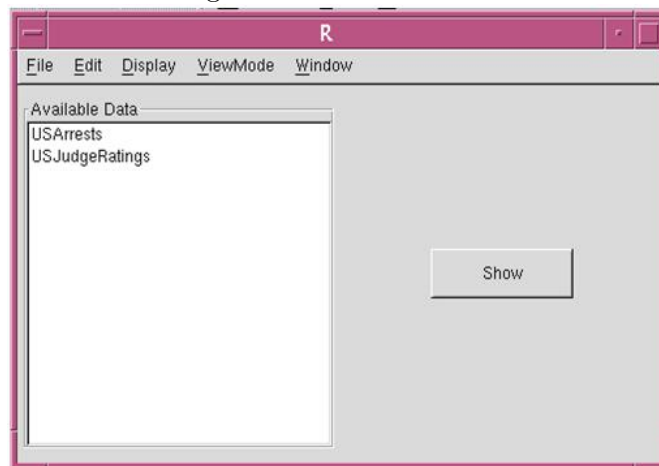
can not be loaded and a message box will appear to let the user know that the data could not be loaded.

After loading some data, the user can either view the data or go directly to plotting the data. Viewing the data will be discussed in section 4 and plotting the data will be discussed in section 5.

4   Viewing Data

After loading data into the data environment using the methods described in section 3, the data can be viewed in an Excel-type format. To view the data, highlight the Display menu and select the View Data menu item. Alternatively, typing Ctrl-V will activate the View Data menu item. The control window will now appear as follows after selecting the View Data menu item. Note that figure 4 shows that both the "USArrests" and "USJudgeRatings" dataframes have been loaded using the methods described in section 3.
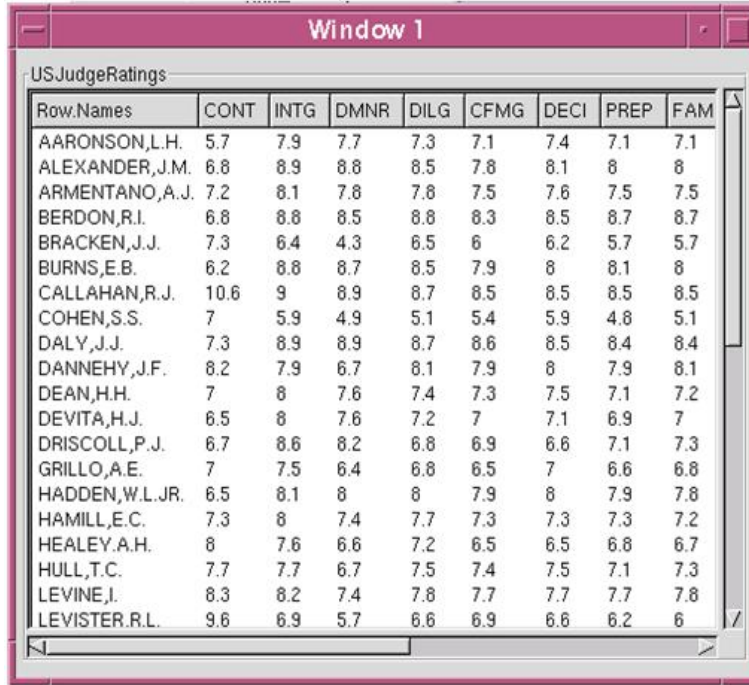
Figure 4: View the Data

Now highlight "USJudgeRatings" in the Available Data list and click the Show button. Another window will appear that shows the dataframe data. Note that the last four columns of the dataframe were added when the data was loaded. These last four columns are color, pch, highlit and hide. These columns are needed when the data are plotted. Initially, they should be set to the defaults of black, 1, FALSE, and FALSE, respectively. These columns will be discussed in more detail in sections 5 and 6.

The new window showing the "USJudgeRatings" dataframe will look as follows.

Figure 5: View USJudgeRatings Data

USJudgeRatings								
Row.Names	CONT	INTG	DMNR	DILG	CFMG	DECI	PREP	FAM
AARONSON,L.H.	5.7	7.9	7.7	7.3	7.1	7.4	7.1	7.1
ALEXANDER,J.M.	6.8	8.9	8.8	8.5	7.8	8.1	8	8
ARMENTANO,A.J.	7.2	8.1	7.8	7.8	7.5	7.6	7.5	7.5
BERDON,R.I.	6.8	8.8	8.5	8.8	8.3	8.5	8.7	8.7
BRACKEN,J.J.	7.3	6.4	4.3	6.5	6	6.2	5.7	5.7
BURNS,E.B.	6.2	8.8	8.7	8.5	7.9	8	8.1	8
CALLAHAN,R.J.	10.6	9	8.9	8.7	8.5	8.5	8.5	8.5
COHEN,S.S.	7	5.9	4.9	5.1	5.4	5.9	4.8	5.1
DALY,J.J.	7.3	8.9	8.9	8.7	8.6	8.5	8.4	8.4
DANNEHY,J.F.	8.2	7.9	6.7	8.1	7.9	8	7.9	8.1
DEAN,H.H.	7	8	7.6	7.4	7.3	7.5	7.1	7.2
DEVITA,H.J.	6.5	8	7.6	7.2	7	7.1	6.9	7
DRISCOLL,P.J.	6.7	8.6	8.2	6.8	6.9	6.6	7.1	7.3
GRILLO,A.E.	7	7.5	6.4	6.8	6.5	7	6.6	6.8
HADDEN,W.L.JR.	6.5	8.1	8	8	7.9	8	7.9	7.8
HAMILL,E.C.	7.3	8	7.4	7.7	7.3	7.3	7.3	7.2
HEALEY.A.H.	8	7.6	6.6	7.2	6.5	6.5	6.8	6.7
HULL,T.C.	7.7	7.7	6.7	7.5	7.4	7.5	7.1	7.3
LEVINE,I.	8.3	8.2	7.4	7.8	7.7	7.7	7.7	7.8
LEVISTER.R.L.	9.6	6.9	5.7	6.6	6.9	6.6	6.2	6

Currently, each dataframe's data can be viewed in only one window. For example, now that the "USJudgeRatings" dataframe is being shown in a window, it can not be shown in a different window. So if a user clicks the Show button while "USJudgeRatings" is highlighted in the Available Data list and a window showing "USJudgeRatings" is already open, then nothing will happen. To open a different window showing "USJudgeRatings", the first window with "USJudgeRatings" in it must be closed.

The reason why the data can be shown in only one window at a time is for simplicity. Also, if the data changes, then the window showing the data will also be updated. To keep the data and its views in synch, each dataframe can only be shown in one window.

Even though each dataframe can be shown in only one window, there is no limit on how many dataframes can be shown in separate windows. That means you can have one window showing "USJudgeRatings" and another window showing "USArrests", and so on. So the user can now highlight "USArrests" and click on the Show button and a new window will open showing the data from "USArrests".
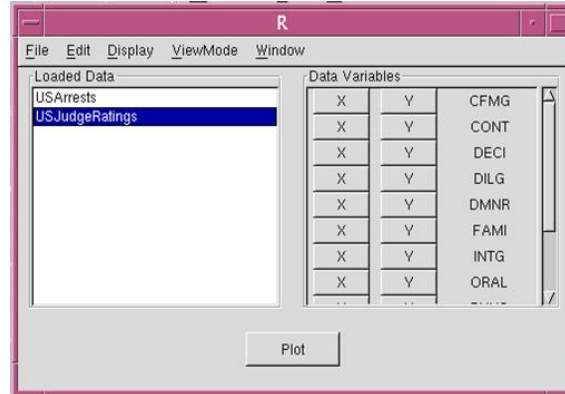
One last note about viewing the data: the data can not be changed through the view data windows.

5 Plotting Data

Once a dataframe is loaded, the user can create scatterplots of the data. To plot a dataframe, highlight the Display menu and select the Plot Data menu item. Alternatively, the user can press Ctrl-P to activate the Plot Data menu item.
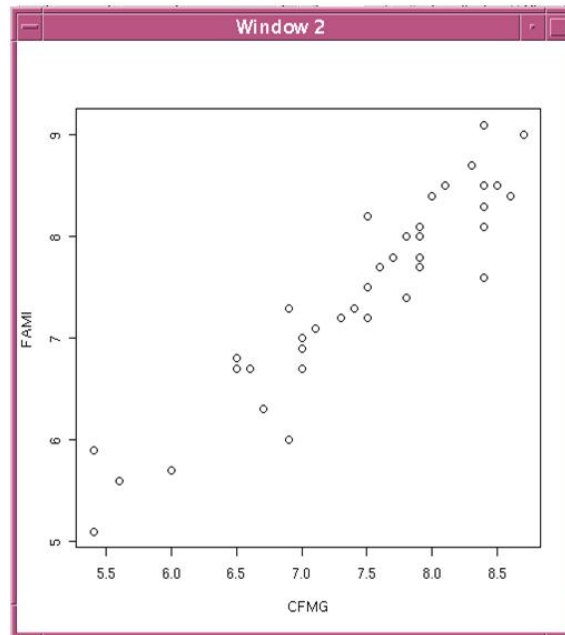
After activating the Plot Data menu item, highlight "USJudgeRatings" in the Loaded Data list on the left hand side of the control window. By highlighting "USJudgeRatings", the right hand side of the control window is filled in with the data variables. The control window should now appear as follows.

Figure 6: Plot a Dataframe

The variables available for plotting "USJudgeRatings" data are CFMG, CONT, DECI, DILG, DMNR, FAMI, INTG, ORAL, PHYS, PREP, RTEN, and WRIT. To learn more about these variables, the user can type help("USJudgeRatings") at the R prompt. As an example, click the X button for CFMG and click the Y button for FAMI, and then click the Plot button. A scatterplot of FAMI vs. CFMG will appear in a new window, as shown below.

Figure 7: Plot FAMI vs. CFMG for USJudgeRatings

Only one X button and Y button can be highlighted at a time because only two-way scatterplots can be made at this time. Also, unlike viewing the data where each data set can only appear in one window, the user can create as many plots of each data set as they like. Each new scatterplot will be presented in a new window.

To create another plot of "USJudgeRatings" data, click the X button for CONT and the Y button for ORAL, and then click the Plot button. Now there will be two plots of "USJudgeRatings" data. One of ORAL vs. CONT and one of FAMI vs. CFMG. Because these plots are based on the same data, they will automatically be linked. This idea will be discussed further in section 6.
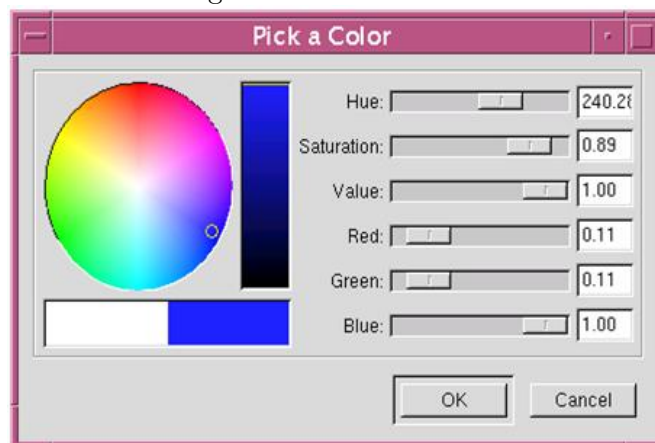
6 Setting ViewMode

Currently, there are four possibilities for the ViewMode: color, highlight, identify, and hide. The ViewMode controls what happens when the user interacts with a scatterplot.

Color

For example, set the ViewMode to color by highlighting the ViewMode menu and selecting the Color menu item. Alternatively, the Color menu item can be activated by pressing Ctrl-C. After setting the ViewMode to color, a color browser window will appear, which allows the user to set the color. Also, the title on all of the scatterplots will now say "color", showing the user what will happen if they interact with that scatterplot.

First, pick a color from the color wheel. As shown below, the color chosen on the color wheel is blue.
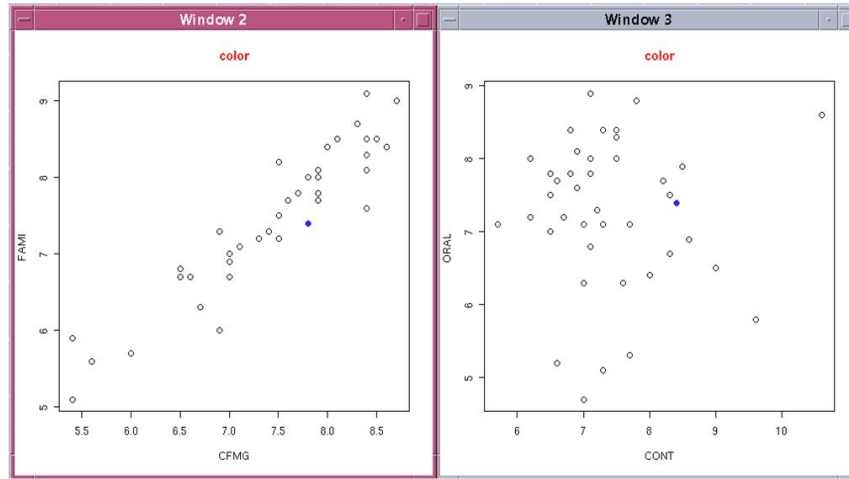
Figure 8:   Color Browser

Now click on a point in the scatterplot of FAMI vs. CFMG. The point that was just clicked will be filled in blue. Also, a point on the scatterplot of ORAL vs. CONT will also be filled in blue because these two points are from the same row of the dataframe. This can be seen by viewing "USJudgeRatings" data, using the View Data menu item under the Display menu. See section 4 for more information. If the view window for "USJudgeRatings" is open (as discussed

7

in section 4), you can see that the column for color will now have one row with a label that is not black (the label will be a set of numbers and/or letters, which correspond to the red, green, and blue values). This row corresponds with the point that was just clicked. Also this row in the spreadsheet will be selected to indicate that this row was just colored. As an alternative method of coloring points, you can also color a point by selecting a row in the spreadsheet.

Figure 9: Coloring Points in the Scatterplot

This behavior shows how the scatterplots are linked. Even though a point on the FAMI vs. CFMG plot was clicked, all plots that are based on the "USJudgeRatings" dataframe will be updated. Note that this also explains why the columns for color, pch, highlit and hide are added to data sets when they are loaded (see section 4). This information is needed to keep all plots of the data in synch.

For reference, color refers to the color of the point, pch refers to the plotting character of the point, highlit refers to whether the point is highlighted or not, and hide refers to whether a point is hidden or not.

Now, if the user clicks on a point in the scatterplot of ORAL vs. CONT, both plots will show two blue points each. The color can be changed at any time by clicking on the color wheel.

Clicking the Ok or Cancel buttons on the color browser will cause the color browser window to be destroyed. To make the color browser reappear just press Ctrl-C or select the Color menu item under the ViewMode menu.
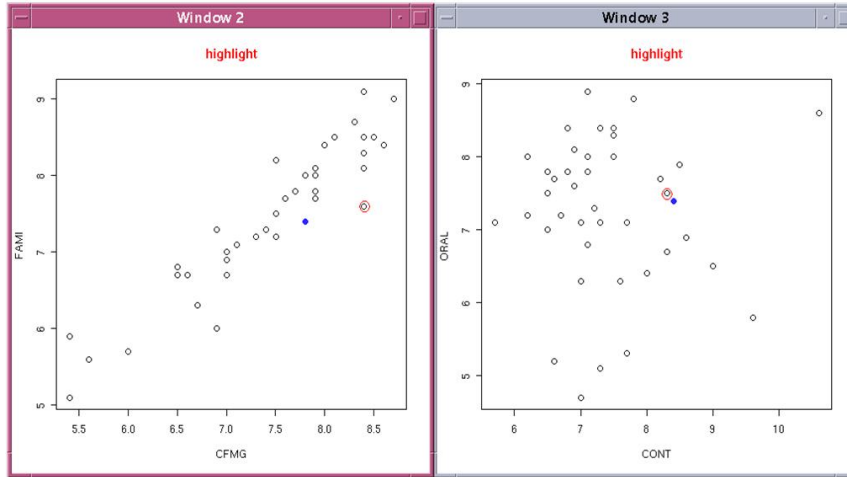
Highlight

To set the ViewMode to highlight, either press Ctrl-H or select the Highlight menu item under the ViewMode menu. The title of all the scatterplots will now be set to "highlight".

Highlighting a point means that an outer red circle will appear around that point. For example, click on a point in the scatterplot of FAMI vs. CFMG. A red circle will appear around that point. Again, because these plots are linked, the plot of ORAL vs. CONT will

8

also show a point that has a red circle around it. These points correspond to the same row in the "USJudgeRatings" dataframe.

Figure 10: Highlighting Points in the Scatterplot

If the view window for "USJudgeRatings" is open (as discussed in section 4), you can see that the column for highlit will now be set to TRUE for one row. This row corresponds with the point that was just clicked. Also this row in the spreadsheet will be selected to indicate that this row was just highlighted. As an alternative to clicking on a point in a plot, you can set highlit to TRUE for a row by selecting a row in the spreadsheet.

To remove the highlighting, just click on the same point again and the red circle will disappear. Also, you can remove the highlighting by unselecting a row in the spreadsheet if you have the view window open. Thus, highlighting a point is an action that can be toggled. This is not true when the color of a point is set - this action can not be toggled.
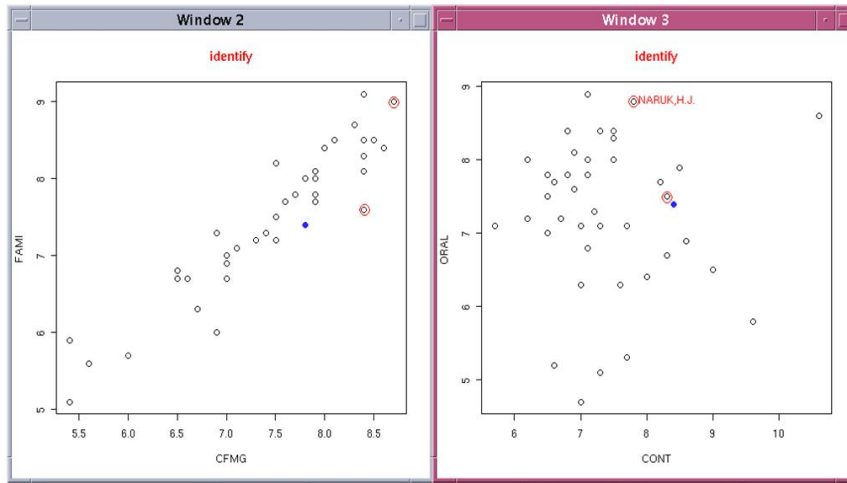
Identify

To set the ViewMode to identify, either press Ctrl-I or select the Identify menu item under the ViewMode menu. The title of all the scatterplots will now be set to "identify".

Identifying means that the row name of the point will appear next to it and the point will also be highlighted. Identify mode is different from color and highlight in that the user does not need to click on the point to identify it. The mouse only needs to hover over a point to identify it.

Set the scatterplot of ORAL vs. CONT to be the active window by clicking on the plot window. Place the mouse so it hovers over a point on the plot, the point will have a red circle appear around it and the row name will appear next to it. If the mouse is moved off of the point, the red circle and the row name will disappear. Also, note that when a point is identified on the scatterplot of ORAL vs. CONT, the corresponding point on the scatterplot of FAMI vs. CFMG will also appear to be highlighted. However, the row name will only appear on the active plot.

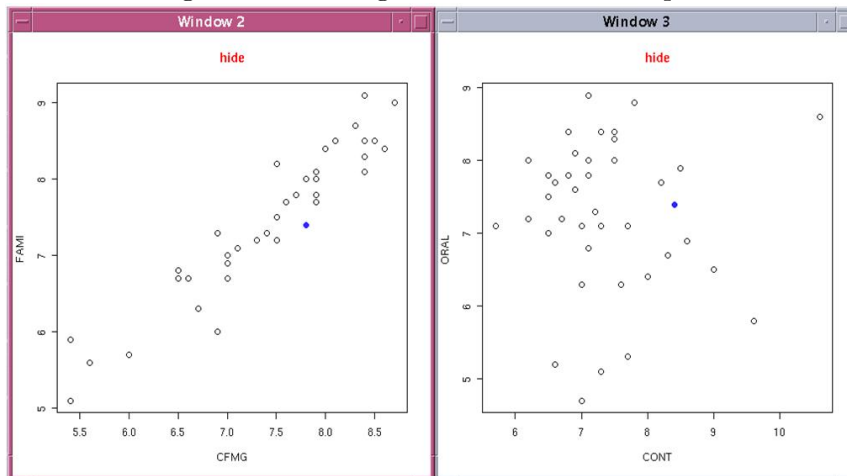Figure 11: Identifying Points in the Scatterplot

Hide

To set the ViewMode to hide, either press Ctrl-E or select the Hide menu item under the ViewMode menu. The title of all scatterplots will now be set to "hide".

Hiding a point means that the point will be drawn in the same color as the background so that the point is "hidden". For example, click on the point in the scatterplot of FAMI vs. CFMG that is currently highlighted. This point along with its highlighting will disappear. This point will also be hidden on the plot of ORAL vs. CONT because the two plots are based on the same data.

Figure 12: Hiding Points in the Scatterplot

If the view window for "USJudgeRatings" is open (as discussed in section 4), you can see that the column for hide will now be set to TRUE for one row (its highlit column will also

be set to TRUE). This row corresponds with the point that was just clicked. Also this row in the spreadsheet will be selected to indicate that this row was just hidden. As an alternative to clicking on a point in a plot, you can set hide to TRUE for a row by selecting a row in the spreadsheet.
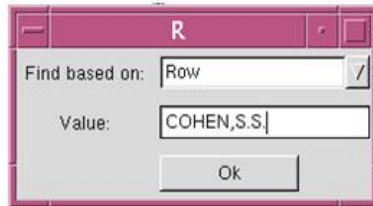
To redraw the hidden point, just click on the same point again and the point will reappear (along with the highlighting - note that hiding a point does not affect the highlit column). Also, you can set the hide column to FALSE by unselecting a row in the spreadsheet if you have the view window open. Thus, hiding a point is an action that can be toggled, just as highlighting can be toggled.

7   Editing Data

Find

Now suppose you are interested in finding the points that corresponds to a particularly row in a data set. Select the Find menu item under the Edit menu or press Ctrl-F. A new window that looks like figure 13 will appear (except the text box will be empty).
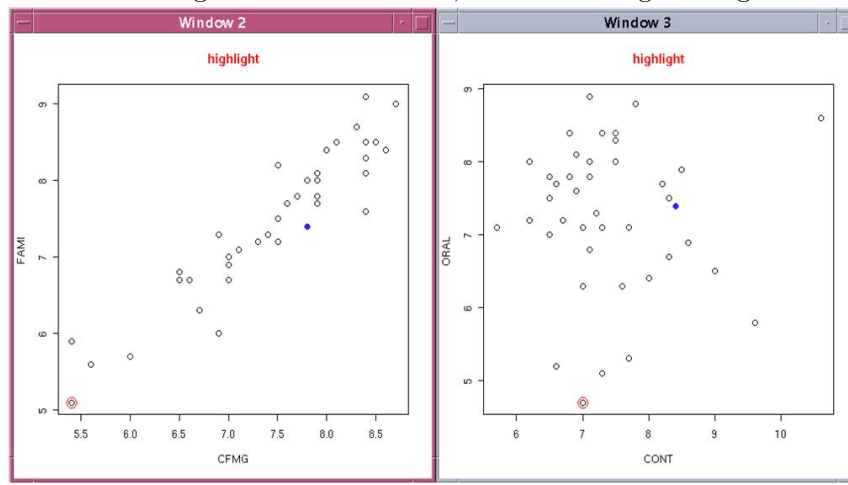
Figure 13: Finding Points in the Scatterplot

First set the ViewMode to highlight by either pressing Ctrl-H or selecting the Highlight menu item under the ViewMode menu. Then set the Find based on drop down box to Row, enter the row name you want to find and press the Ok button. For example, in the text box enter "COHEN,S.S.", except without the quotes. Make sure you do not enter any spaces and that you capitalize all of the letters, and then press the Ok button. The active scatterplot will have the point corresponding to COHEN,S.S. highlighted. Any other plots that depend on the data from "USJudgeRatings" will show that point as being highlighted.

Note that if the row is found, then the action taken will depend on the view mode. For example, if the view mode was hide, then if the row was found, that row would have the hide column set to TRUE. So make sure you have the view mode set to the action you want to happen.

Figure 14: Finding the Point COHEN,S.S. in USJudgeRatings Scatterplot

You can also find rows based on a data element. This time set the Find based on drop down box to Data Element, enter a value from the data set (for example, enter the number, 8.8) and press the Ok button. Now the first row that contains this data element will have its highlit column set to TRUE because the view mode is set to highlight. For the USJudgeRatings data set, the first row containing the value, 8.8, is the second row, ALEXANDER,J.M. This point will now be highlighted on the plots.

Replot

Replotting the data is useful if you have made many changes to the data set and thus, your scatterplots will be cleaned up by replotting. Replotting will plot the data set as it is stored in the data environment. In other words, it will not make changes to the data set like Reset does, which is described below. Replotting will only refresh the plots and remove any selected rows on the spreadsheet view.

To replot your active scatterplot and any other plots that depend on the same data as your active scatterplot, choose the Replot menu item under the Edit menu or press Ctrl-R. A message box will appear asking if the user wants to replot the active scatterplot. Click the Yes button and all views based on the same data as the active scatterplot will be replotted.
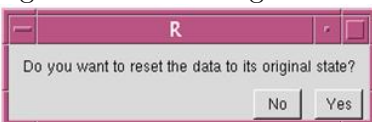
Figure 15: Replotting the Data

If you have multiple plots drawn that depend on different dataframes, only plots that depend on the same dataframe as the active plot will be replotted.

Reset

If you want to reset the data so it appears as it did when it was first loaded, then you want to use reset. This will remove any changes you have made to the data, such as setting the color or highlighting a point. Select the Reset menu item under the Edit menu or press Ctrl-S, which results in a message box appearing to ask if the user really wants to reset the data. Resetting the data can not be undone. Click the Yes button and all plots based on "USJudgeRatings" will be replotted.

Figure 16: Resetting the Data

If you have multiple data sets loaded, reset will only change the data set that corresponds with the active plot.

8   Adding Menu Items

If the user wants to add menu items to the main menu on the control window from the R command line, then the following two R functions will allow the user to do that: `addMenuItem` and `addSubMenuItem`.

Adding Menu Items

The function `addMenuItem` adds a menu item to the menu bar on the control window. So the new menu item will appear after Window on the menu bar (other menu items on the menu bar are File, Edit, Display, ViewMode, and Window).
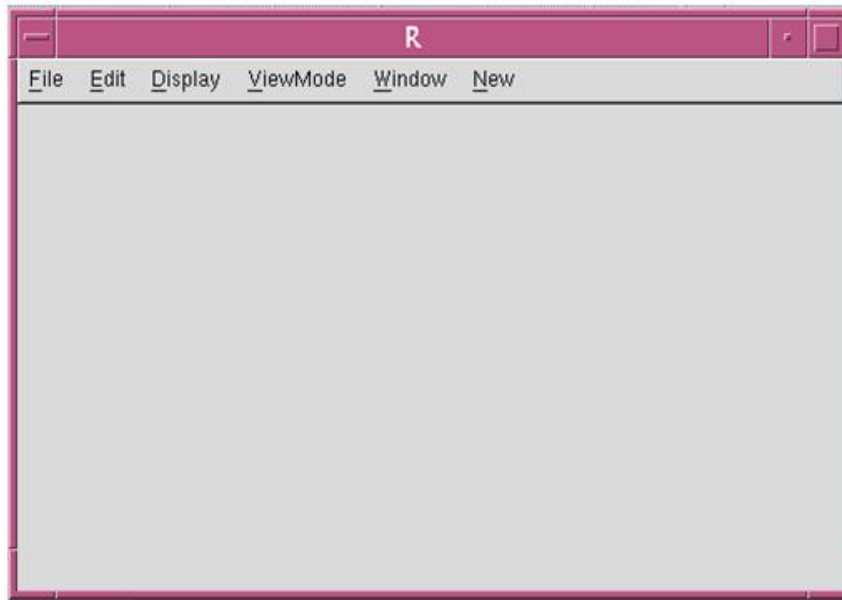
To add an accelerator to this menu item, place an underscore before the letter that you want to be the accelerator. In the following code example, the accelerator will be 'N' because there is an underscore placed before the 'N' in New.

The default modifier type for adding a menu item (parameter modType) is the Alt button. So in the new menu item that the following code will create, the accelerator will be Alt-N. See the man page for `addMenuItem` for descriptions of the function's parameters.

```
> if (interactive()) {
+     addMenuItem("new", "_New")
+ }
```

After performing the above code, the control window will look as shown in figure 17.

Figure 17: Adding a Menu Item

Adding Sub Menu Items

If instead the user wants to add a sub menu item to an already existing menu, then the user should call the function `addSubMenuItem`. Examples of sub menu items that are already in the main menu are Open Data, Open File, and Quit under the File menu.

Again, to add an accelerator to the sub menu item, place an underscore before the letter that you want to be the accelerator. In the following code example, the accelerator will be 'N' because the underscore appears before the letter 'N' in Different.

The default modifier type for adding a sub menu item (parameter modType) is the Ctrl button. So in the new sub menu item that the following code will create, the accelerator will be Ctrl-N. See the man page for `addSubMenuItem` for descriptions of the function's parameters.

Also, the user must give a character string for curAction (the second parameter) that corresponds to the name of a function that will be called when this sub menu item is activated. This function may have parameters. In the code below, curAction is set to "testfun", the name of the function that was previously defined.
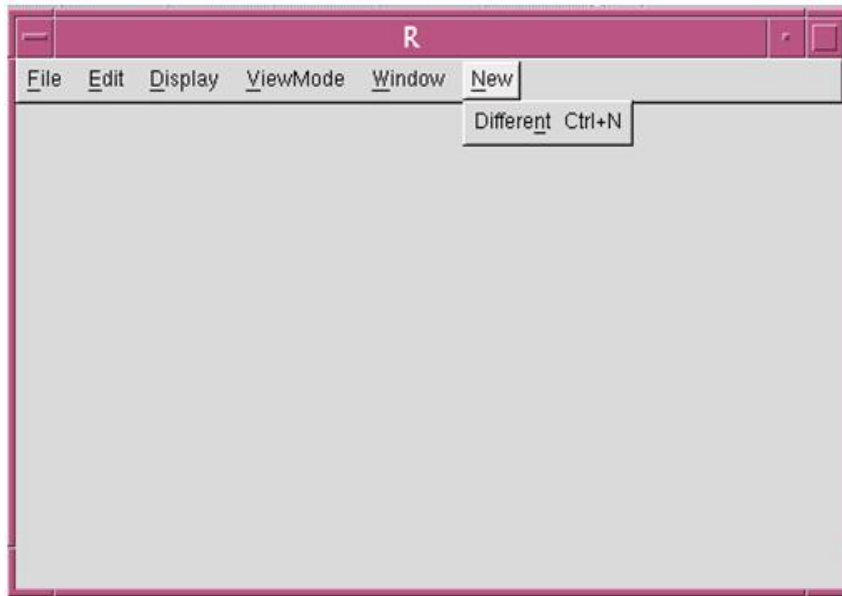
In the following example, the sub menu item, Different, is added to the menu item, New.

```
> if (interactive()) {
+     testfun <- function() {
+         w <- gtkWindow(show = FALSE)
+         lab <- gtkLabel("It works!")
+         w$Add(lab)
+         w$Show()
+     }
+     addSubMenuItem(menuName = "new", labelText = "Differe_nt  Ctrl+N",
```

14

```
+          action = "testfun")
+ }
```

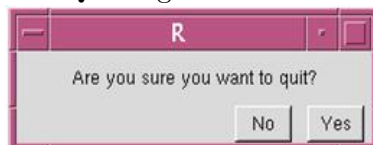After performing the above code, the control window will look as shown in figure 18.

Figure 18: Adding a Sub Menu Item

9 Quitting

When the user is ready to quit, select the Quit menu item under the File menu or press Ctrl-Q. A message box will appear that asks if the user really wants to quit. By clicking the Yes button, all windows such as plot and view data windows as well as the control window will be closed.

Figure 19: Quitting Interactive Scatterplots

10 Future Additions