This file consists of Appendix B to **MacAnova User's Guide** by Gary W. Oehlert and Christopher Bingham, issued as part of a revision of Technical Report Number 617, School of Statistics, University of Minnesota, August 1998, describing Version 4.07 of MacAnova.

This manual is Copyright © 1998 Gary W. Oehlert and Christopher Bingham, all rights reserved.

Fonts used in this manual are Palatino, Courier, and Symbol.

For information concerning MacAnova, write University of Minnesota, Department of Applied Statistics, 352 Classroom Office Building, 1994 Buford Avenue, St. Paul, MN 55108-6042.



Appendix B MacAnova on a Macintosh

B.1 **Introduction** This appendix summarizes features special to the Macintosh versions of MacAnova.

B.1.1 **MacAnova windows** All input and non-graphics output occurs in an editable, scrollable window, referred to below as a *command/output* window. When MacAnova starts up, the window is named **Untitled-1**.

The line or lines immediately after the last Cmd> prompt in the window are collectively known as the *command line*. When the insertion point (a flashing vertical line like this|) is at the end of the command line, pressing Return usually executes the line. If you press Return when the insertion point is *not* at the end of the command line, the line is broken but will not execute. Until the command line is executed, you can go back and edit it.

Enter and Shift Return behave differently from Return. No matter where the insertion point is, pressing Enter or Shift Return adds a return *at the end* of the command line, usually executing it. Because of this feature, you don't need to move back to the end of the line to execute it. Instead of Enter and Shift Return you can press ⌘\, ⌘Return or function key F6. The ⌘ key is sometimes called the "command" key and is pressed together with another character.

MacAnova saves up to about 32000 characters in a command/output window. If the window gets too full, a new command window named **Untitled-2** is opened after giving you a chance to save the one you are currently using. You can have up to nine such windows at any one time, although only the front most one is active. You can switch between them using the **Windows** menu (Sec. B.3.4) or by clicking on visible portions of them.

There can also be up to eight graph windows created by graphics commands such as `boxplot()`, `plot()`, `lineplot()` and `chplot()` (Sec. 2.15, 8.5). In addition there may be one or two more graph windows, the so called *panel windows*, each displaying the contents in miniature of up to four of the regular graph windows. When a graph window is in front, pressing Return brings the most recently displayed command/output window to the front. As with command/output windows you can swith between graph windows using the **Windows** menu or by clicking on a window.

B.1.2 **Versions for different types of Macintoshes** There are three MacAnova versions for Macintosh – the PPC version, the *no co-processor version* and the *co-processor version* and . All function identically, but have different hardware requirements. The PPC version runs *only* on a Power Macintosh or compatible with the Motorola Power PC™ processor. The no co-processor version should run on any Macintosh with sufficient memory, as little as 1 megabyte. It neither requires nor uses a 68881 math co-processor. The co-processor version requires a Motorola 68020 or better processor and a Motorola 68881 or better math co-processor. It will not run on a Power Macintosh. See Sec. B.8.1 for information on compilers used.

B.1.3 **File names** Although you seldom will need to use the information, you should know that the Macintosh OS uses ":" to separate folders (directories) in file and path names. A completely specified path starts with a volume (disk) name, followed by 0 or more folder names, followed by the file name, all separated by ":". An example might be `"Macintosh HD:MacAnova Folder:MacAnova.mac"`. The path to a file in a folder in the current folder has the form `":Folder Name:File Name"`, starting with ":". A file in the "parent" directory has path `"::File Name"`.

B.2 **Launching MacAnova** There are three types of MacAnova icons, the MacAnova application icon, an icon for files created by `save()` or `asciisave()`, and an icon for files created by **Save Output** or **Save Output & Quit** on the **File** menu.



Application Icon



Save File Icon



Output Window Icon

MacAnova may be started up in several ways.

1. Double clicking the Application icon.

2. Double clicking on a Save File icon. This restores the workspace previously saved.

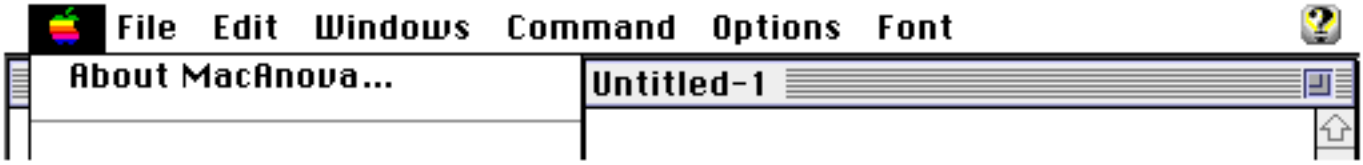
3. Double clicking on an Output Window icon. This initializes the Command Window with commands and output from a previous run.

4. Selecting both a Save File icon and a Output Window icon and then double clicking restores *both* the previous workspace and previous output. This means you can end a session and restore things exactly to how they were at the point you quit.

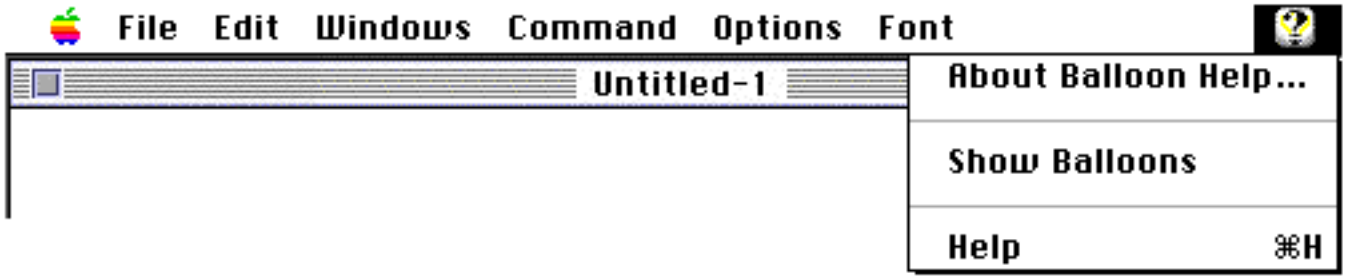
Under System 7 or later, you may want to put an alias of MacAnova in the Apple Menu Items folder in the System folder or install it in Launcher.

B.3 **Menus** In addition to the and **Help** menus, there are six menus, **File**, **Edit**, **Windows**, **Command**, **Options** and **Font**.

B.3.1 **The and Help menus**

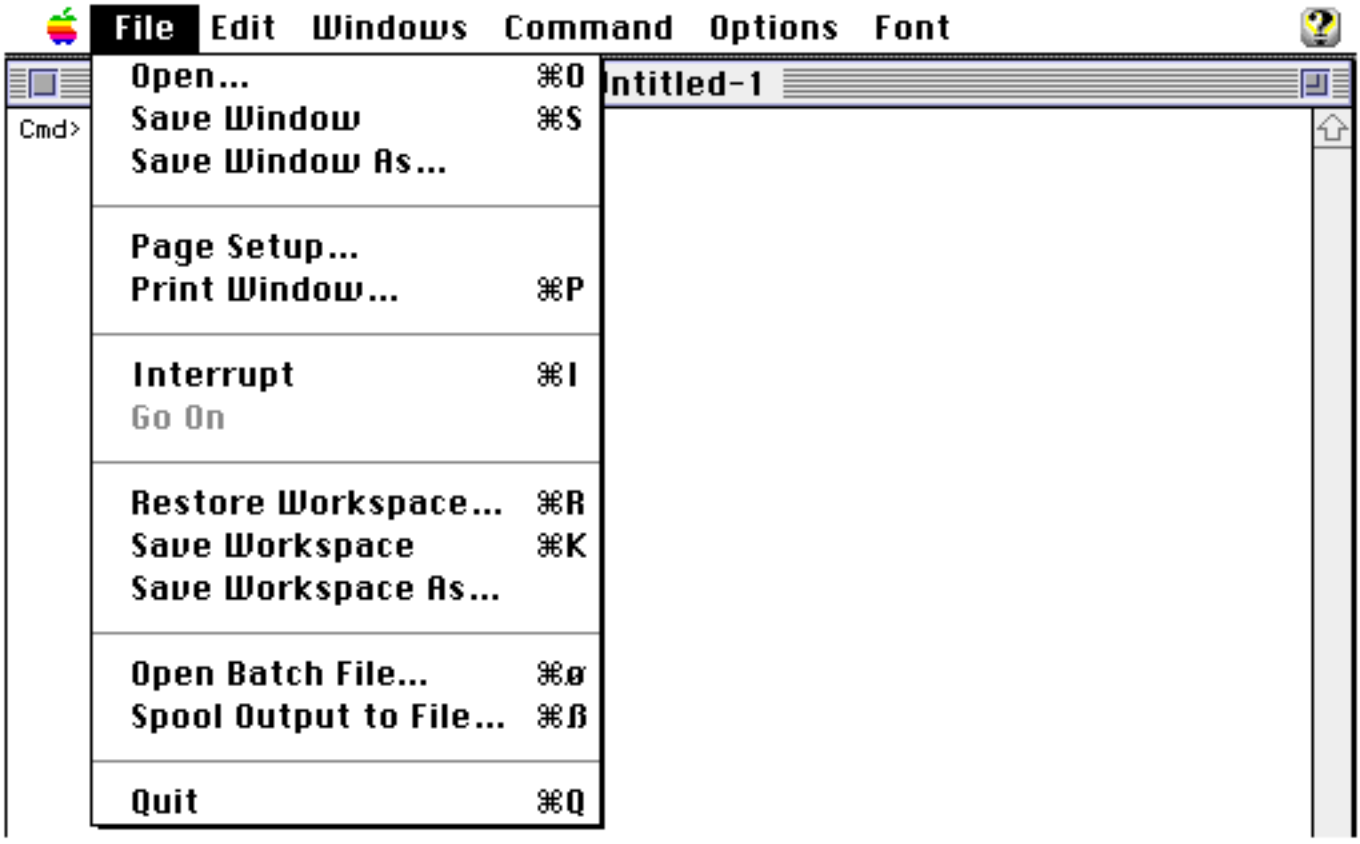


About MacAnova... on the menu brings up a dialog box that includes information about the version of MacAnova you are running, as well as some tips for help topics.



Help on the menu is equivalent to typing `help()`, producing a very brief summary of how to use `help()`. When you select a topic or command name in the window by double clicking on it, the item will read something like **help("anova")** and selecting it gives you help on that topic. You can also select this item by pressing **⌘H** (⌘ key together with H) or Help. See Sec. 2.9. When MacAnova is running under System 6 or earlier, this item is on the menu.

B.3.2 **The File menu**



Open... allows you to load the contents of a previously saved output window, or any plain text file, into the current window. It brings up a dialog box for you to select the file. You can select **Open...** from the keyboard this item by pressing ⌘**O**.

Save Window saves the front command/output window. The first time you save it, you are asked to provide a name in a dialog box. At that point, the name of the window changes to the file name. Subsequent use of **Save Window** saves it in this file without asking you anything. The file written may be printed or edited using any word processor such as Microsoft Word or an editor such as Simple Text. You can select **Save Window** from the keyboard by pressing ⌘**S**. Note that **Save Window** saves only the text in the window and not the actual variables. See menu item **Save Workspace...** below.

Save Window As... is like **Save Window**, except it always brings up a dialog box for you to enter a file name. When a graph window is in front, this item reads **Save Graph As...**. When selected it brings up a dialog box to name a file where the graph is written in PICT format. Such a file can be imported into many word processors and graphics editors.

Page Setup... works just like **Page Setup...** in any standard Macintosh program. It affects the printing of the window, command/output or graph, that is in front when it is selected.

Print Window... prints the command/output window. If you have selected text in the command/output window using a mouse, this item will read **Print Selection...** and just the selected text will be printed. When a graph window is in front, this item will read **Print Graph...** and the graph window will be printed. You can select **Print Window...** and **Print Graph ...** from the keyboard by pressing ⌘**P**.

Interrupt interrupts any executing command and prints a new prompt. A few commands ignore it, so it may not work instantly. You can select **Interrupt** from the keyboard by pressing ⌘**.** or ⌘**I**.

Go On This is active only immediately after you have drawn a graph with `pause:T` as an argument to the plotting command (Sec. 8.5.2). Selecting **Go On** ends the pause and allows MacAnova to continue. When **Go On** is active you can select it from the keyboard by pressing Return.

Restore Workspace... allows you to restore a workspace previously saved by `save()` or `asciisave()`. Selecting **Restore Workspace...** automatically enters `restore("")` in the command/output window and executes it. This brings up a scrolling dialog box for you to select the file. You can select **Restore Workspace...** from the keyboard by pressing ⌘**R**.

Save Workspace... saves the current workspace (Sec. 2.17) in a file. If the workspace has not been saved previously on the current run, this item automatically enters `save("")` in the command/output window and then executes it. This brings up a

dialog box for you to name the file. If the workspace has been saved previously on the run either using the menu or directly typing a `save()` command, **Save Workspace…** automatically enters `save()` in the command/output window and executes it. This saves the workspace on the same file, with the same options. You can select **Save Workspace…** from the keyboard by pressing ⌘**K**. This provides a quick and easy way to play it safe, saving your workspace by pressing ⌘**K** every few minutes to minimize your losses in case of a crash. See Sec. 2.17, 7.7.

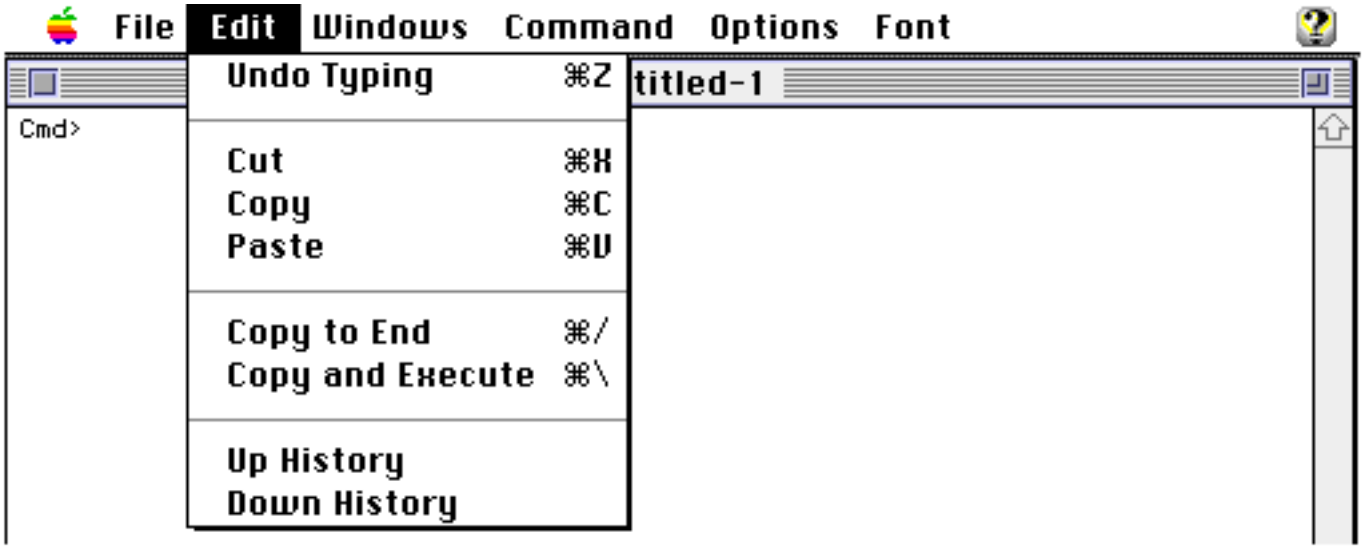
Save Workspace As… automatically enters `save("")` in the command/output window. This brings up a dialog box for you to name the file, even if you have previously saved the workspace.

Open Batch File… automatically enters `batch("")` in the command/output window. This brings up a dialog box for you to select a file of MacAnova commands to be executed. See Sec. 7.6. You can also select this item by pressing ⌘**ø** (⌘ Option o).

Spool Output to File… automatically enters `spool("")` in the command/output window. This brings up a dialog box for you to name a file where all commands and output are echoed. Once such a "spool" file has been started, the menu entry changes to **Stop Spooling** or **Resume Spooling**. Either automatically enters `spool()` in the command/output window, toggling the spooling of commands and output off and on. See Sec. 2.16. You can select this item from the keyboard by pressing ⌘**ß** (⌘ Option s).

Quit terminates MacAnova. Alternatively you can type `quit`, `bye`, `stop`, `exit` or `end`. A dialog box normally gives you a chance to save your workspace and any command/output windows. If you type `quit(F)` or press the option key while selecting **Quit**, MacAnova terminates immediately without giving any chance to save stuff. You can select **Quit** from the keyboard by pressing ⌘**Q** or option+⌘**Q**.

B.3.3 **The Edit menu**



The first menu item is generally referred to as **Undo** or **Redo**. You can select it by pressing ⌘**Z**. It actually is labelled one of **Undo Output**, **Undo Typing**, **Undo Paste**, **Undo Cut**, **Redo Output**, **Redo Typing** or **Can't Undo** and provides a limited facility

for undoing your previous action.

Immediately after executing a command line, **Undo Output** deletes any output back to the end of the just executed command line. This is useful if you made a typing error. Hit ⌘**Z**, make the correction, and press Enter to redo the command. If instead, you immediately select **Redo Output** again, the deleted output is restored. Similarly, after typing anything, **Undo Typing** deletes what you have typed or, if your typing deleted something, restores it. After pasting something (**Paste** on the **Edit** menu), **Undo Paste** removes what was pasted, returning it to the Clipboard, and restores anything that may have been deleted. After cutting a selection (**Cut** on the **Edit** menu), **Undo Cut** restores what was deleted.

When the Command Window is in front, **Cut**, **Copy**, and **Paste** do just what you expect in a Macintosh program. When a graphics window is in front, only **Copy** is effective, copying the entire graph to the Clipboard.

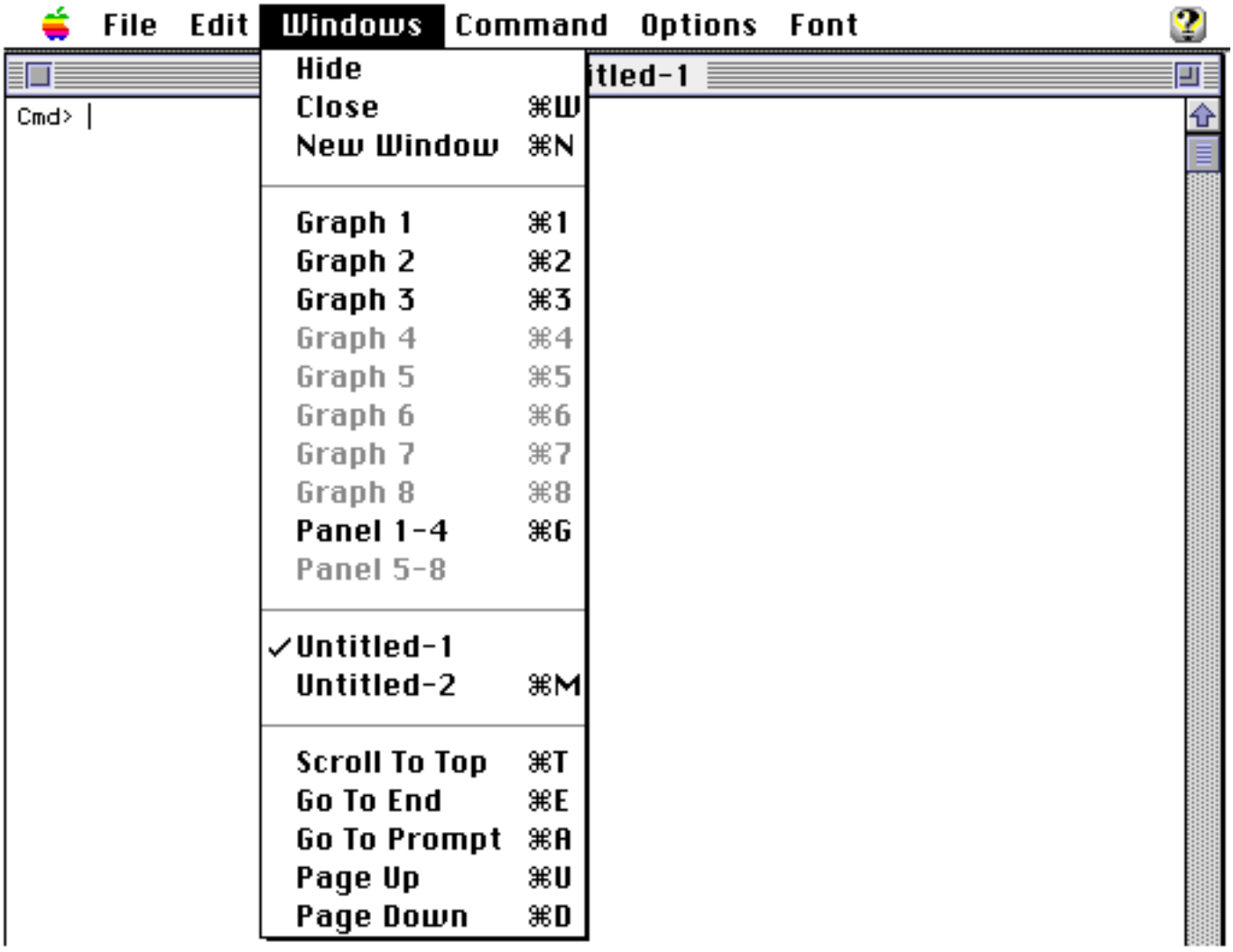
Copy to End When text is selected in the command/output window, **Copy to End** inserts a copy of the selected text at the end of the command line. You can then edit it, if necessary, before executing it. You can select **Copy to End** from the keyboard by pressing ⌘**/** or function key F5.

Copy and Execute is active only when text anywhere before the start of the command line has been selected using the mouse. The selected text is copied to the end of the command line followed by Return; in most cases this executes the entire command line including what was copied. This provides a quick way to re-execute previous commands.

If no text before the command line is selected, this menu entry reads simply **Execute**. When **Execute** is selected, Return is added to the end of the command line and the insertion point is moved there. This executes the line unless there is an incomplete quoted string (Sec. 2.5) or compound command (Sec. 9.2.1). You can select **Copy and Execute** and **Execute** from the keyboard by pressing ⌘****, function key F6, Enter, Shift Return or ⌘Return.

Up History and **Down History** recall previous commands to the command line where they can be executed, possibly after modification. Each selection of **Up History** moves backward one command line while each later selection of **Down History** moves forward. Pressing F7 or Option ↑ activates **Up History**; pressing F8 or Option ↓ activates **Down History**. The number of lines you can go back is controlled by option `history`. Shift Option ↑ moves to the earliest remembered command and Shift Option ↓ restores anything typed before you started recalling previous commands. See Sec. 8.1.3, 8.8 and B.6.4.

B.3.4 **The Windows menu**



Hide hides the front window; it can be made visible again by selecting its entry on its menu.

Close closes the front window, removing it entirely. It is equivalent to clicking in the Close box of the window. You are given a chance to save the contents of a command/output window being closed but you can bypass this by pressing the Option key while closing. You may select **Close** from the keyboard by pressing ⌘**W**.

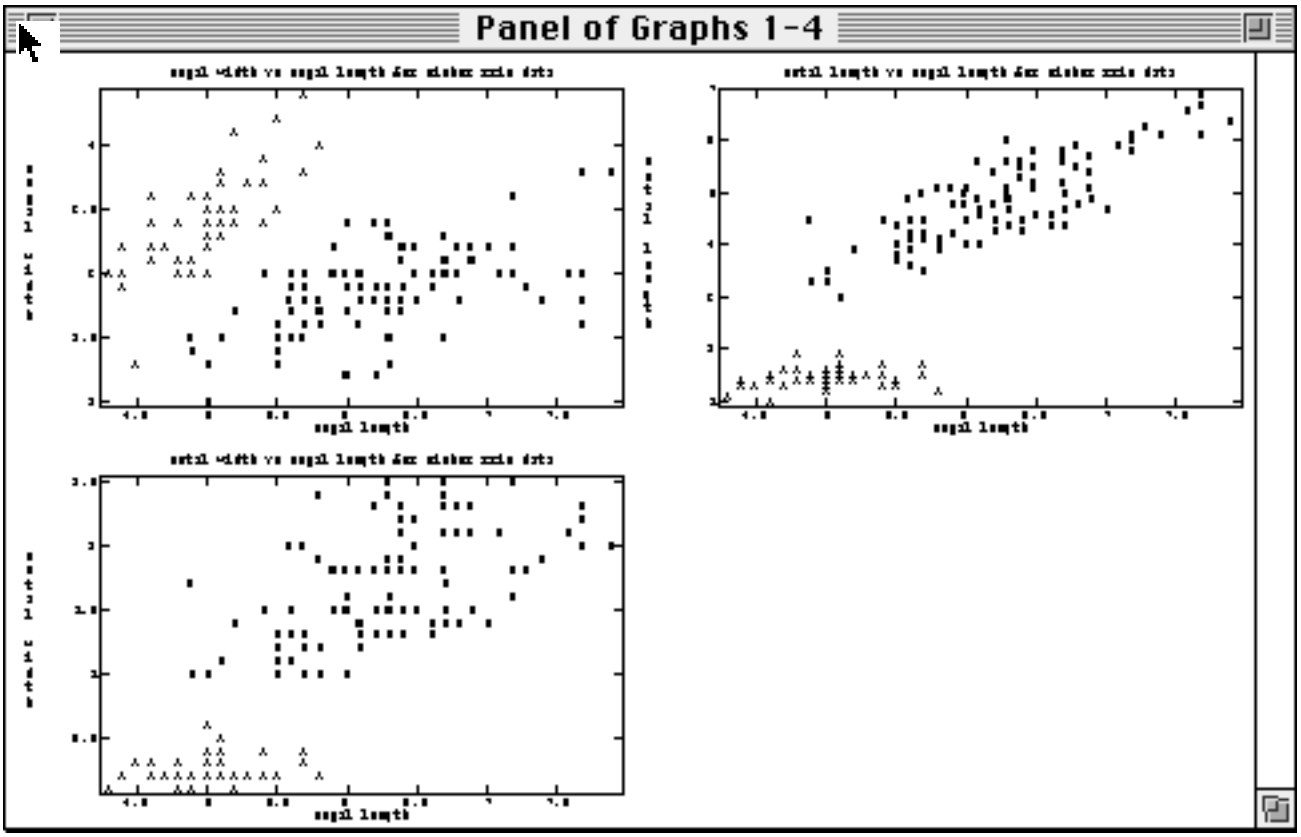
Neither **Hide** or **Close** are active when the front window is the only command/output window; the close box is always active.

New Window creates a new command/output window with name **Untitled-n** where *n* is an integer, and adds its name to the **Windows** menu. You can have up to 9 such windows open at any one time. The only other way a new command/output window is created is when you overflow the capacity of a window. You may select **New Window** from the keyboard by pressing ⌘**N**.

Entries **Graph 1** through **Graph 8** bring one of the graph windows to the front. If an

entry is "grayed out" it means the window doesn't exist. There are three graph windows in use in the example menu. You can select these items from the keyboard by pressing ⌘**1,** ⌘**2**, ..., or ⌘**8** or ⌘F1**,** ⌘F2, ..., or ⌘F8**.**

Panel 1-4 and **Panel 5-8** bring the panel windows to the front. Each displays in miniature up to four of the regular graph windows. Here is an example of a partially filled panel window. When a panel window is not in front, pressing ⌘**G** brings forward the panel window associated with the most recently displayed regular graph window; when a panel window is in front, ⌘**G** switches to the other panel window, if any.



Below the graph and panel window entries is a list of all the open command/output windows. In the example, there are two – **Untitled-1** and **Untitled-2**. The one where that has most recently been in front is checked. Selecting an entry in this list brings the corresponding window to the front. Pressing ⌘**M** always switches to one of the windows in this list. The active command/output window has a ✓ next to its name.

The last five entries on the **Windows** menu are active only when a command/output window is in front.

Scroll To Top scrolls to the start of the text, but does not change the selection point. You can select **Scroll To Top** from the keyboard by pressing ⌘**T** or Home.

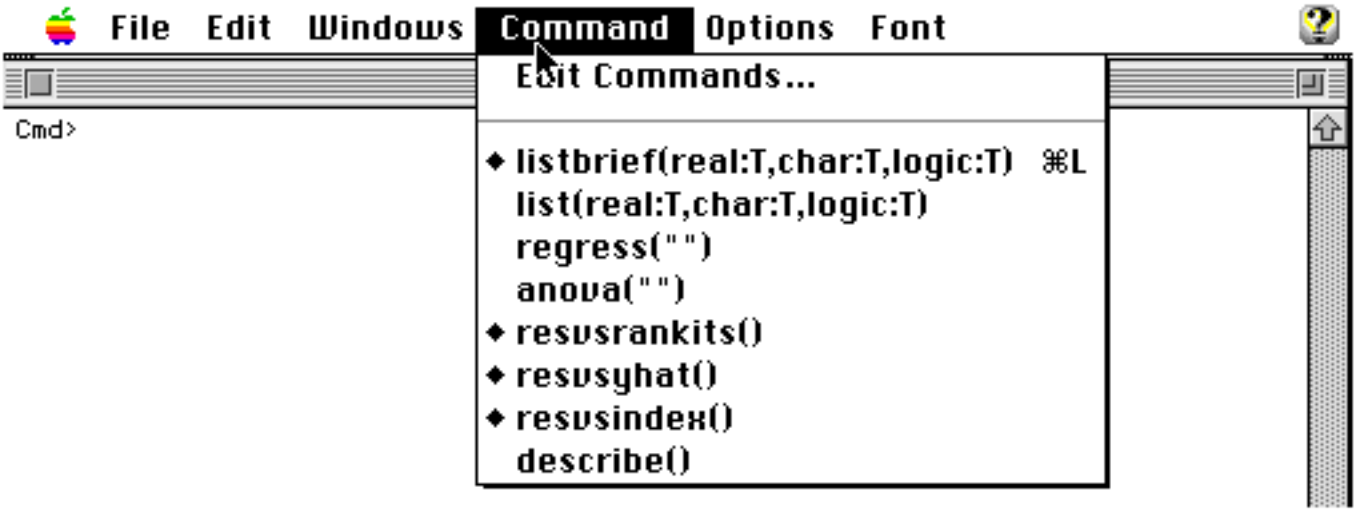
Go To End scrolls to the end of the text in the command/output window and moves the selection point there. You can select **Go To End** from the keyboard by pressing ⌘**E**

or End.

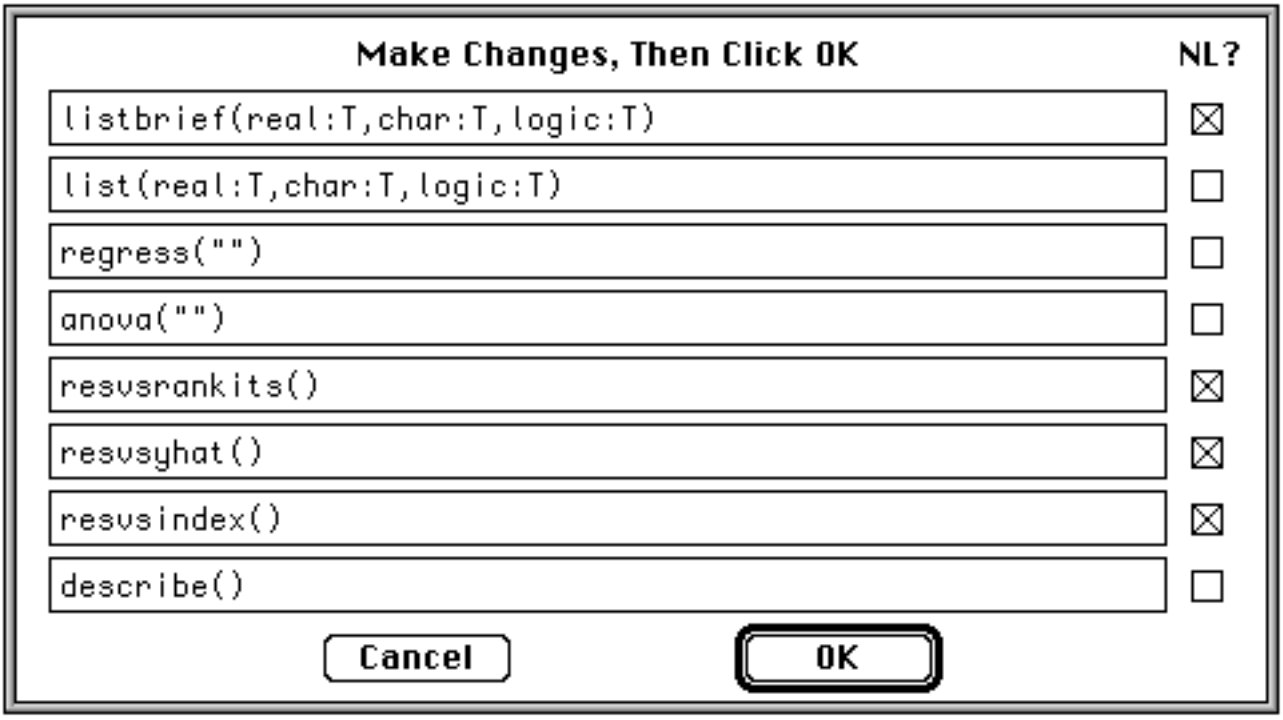
Go To Prompt moves the insertion point to the beginning of the command line, immediately after the prompt, scrolling the command/output window if necessary to bring the prompt into view. You can select **Go To Prompt** from the keyboard by pressing ⌘**A**.

Page Up and **Page Down** scroll the command/output window back and forward a screenful at a time. You can also select **Page Up** from the keyboard by pressing ⌘**U** or Page up and **Page Down** by pressing ⌘**D** or Page down.

B.3.5 **The Command menu**



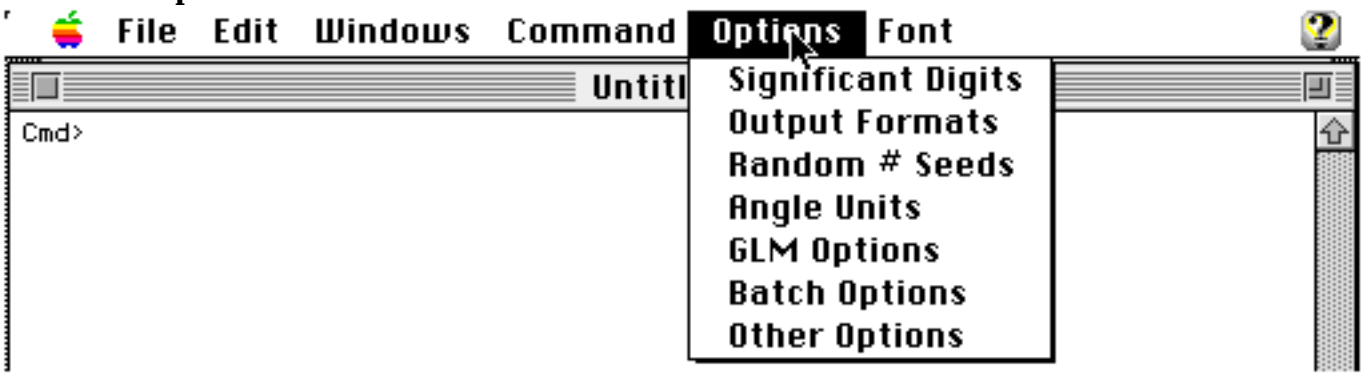
Edit Commands brings up the following dialog box so that you can change any or all of the commands in the last 8 lines of the **Command** menu.



You can change any of the eight entries. Check the box at the right if a Return should automatically be added to the command. When your change or changes are complete, click on **OK** or press Return

The last 8 items in the **Command** menu are predefined commands that are copied to the command line by selecting them with the mouse. A ◆ in the left margin indicates that the command will be followed by Return, resulting in immediate execution. If ◆ is absent, you can edit the command after it is put in the window. The selection point will be inside a set of empty parenthesis (**describe(|)**) or inside empty quotation marks (**anova("|")**). Thus selecting **resvsrankits()** results in an immediate residuals versus normal scores plot (Sec. 3.17), while selecting **anova("")** allows you to specify a model. You can replace or modify any of these commands using item **Edit Commands**. If you want to do anything complicated you should probably write a macro to do what you want (see Sec. 9.3) and have only the macro invocation in the menu. You can also select these items from the keyboard by pressing ⌘Option **1**, ⌘Option **2**, ... or ⌘Option **8**.

B.3.6 **The Options menu**



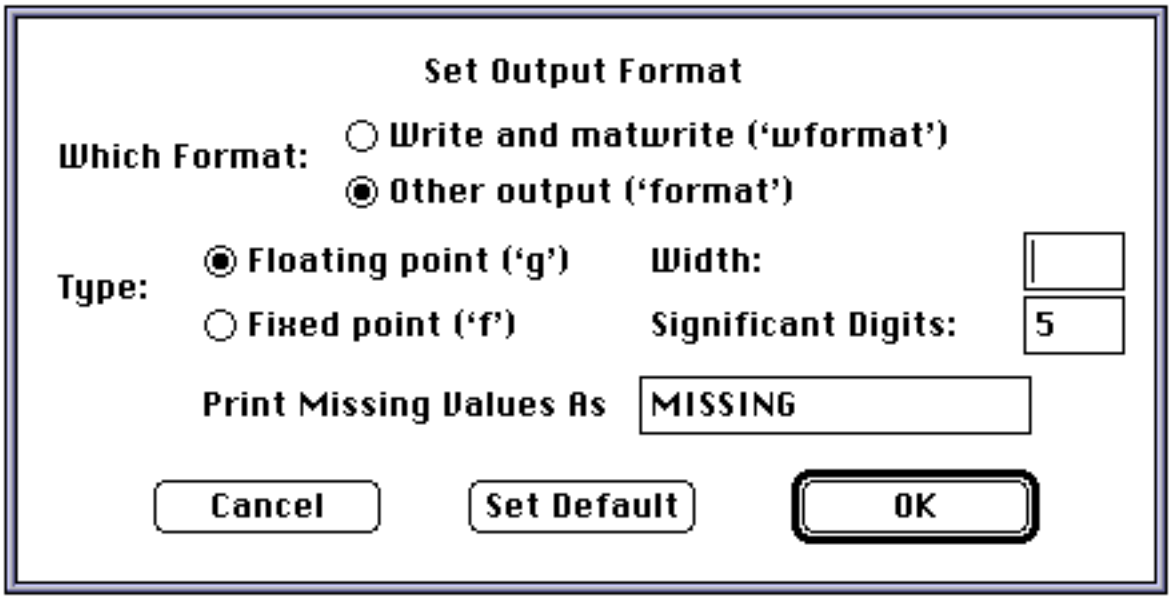
Selecting an item from the **Options** menu brings up a dialog box that allows you to set one or more of the options that can be set by `setoptions()` (Sec. 8.1.3). In fact, once you have clicked the **OK** button in the dialog box, the appropriate `setoptions()` is automatically entered in the command/output window and executed.

Significant Digits



This generates command `setoptions(nsig:`*n*`)`, where *n* is the value in the box. This sets the default number of significant digits in printed output to *n*.

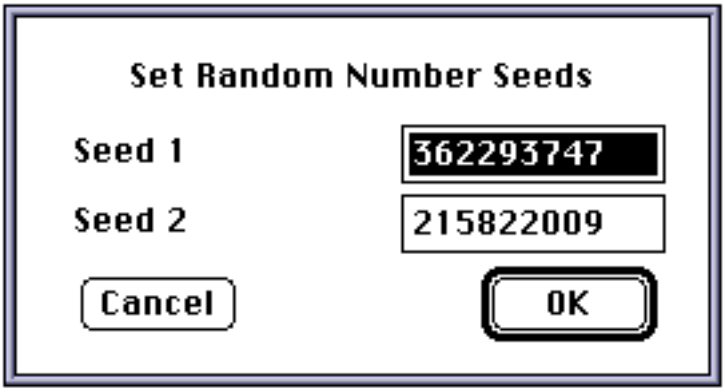
Output Formats



This sets the default formats for output. The format you specify while **Write and matwrite ('wformat')** is selected applies to `write()` and `matwrite()`, while the

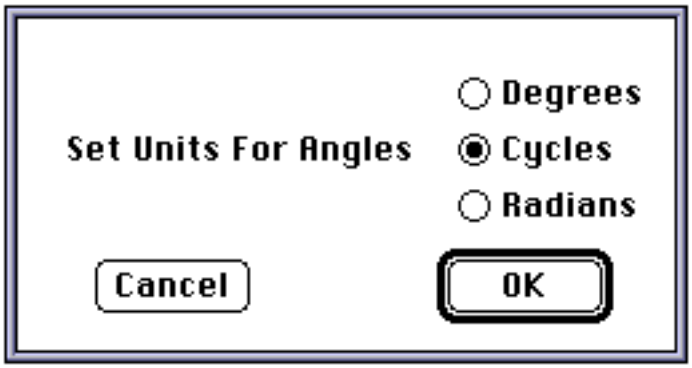
format specified while **Other output ('format')** is checked applies to `print()`, `matprint()`, and most other output. See Sec. 7.4 and 8.1.3. Any value you supply for **Print Missing Values As** changes option `missing`. Clicking the **Set Default** button restores the default formats.

Random # seeds



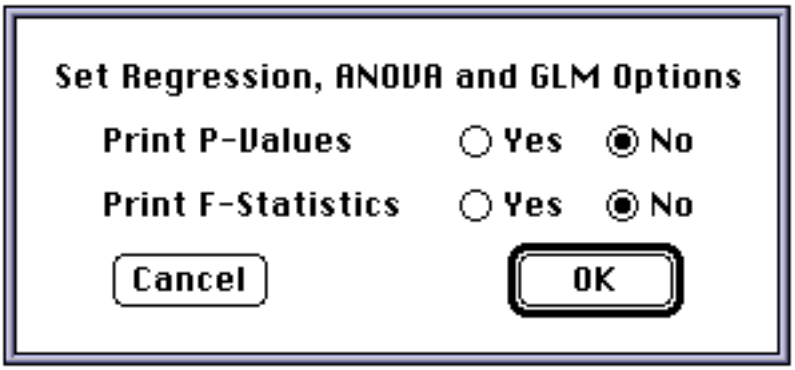
This generates the command `setoptions(seeds:vector(`n_1`,`n_2`))`, where n_1 and n_2 are the numbers in the two boxes. It initializes the seeds used by the random number generating functions `runi()` and `rnorm()` (Sec. 2.13.1).

Angle Units



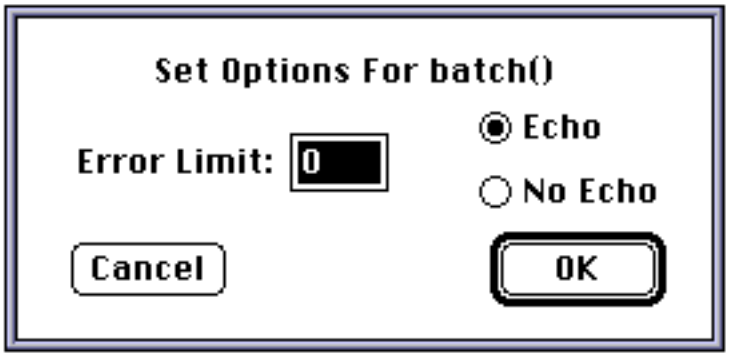
This generates the command `setoptions(angles:`*units*`)`, where *units* is one of `"degrees"`, `"cycles"` or `"radians"`. This determines the units assumed for *arguments* to `cos()`, `sin()`, `tan()`, (Sec. 2.8.6), `hrect()` and `crect()` (Sec. 5.2.4), and for values computed by `acos()`, `asin()`, `atan()`, `hpolar()` and `cpolar()`. The correspondence is 1 cycle = 360° = 2π radians.

GLM Options



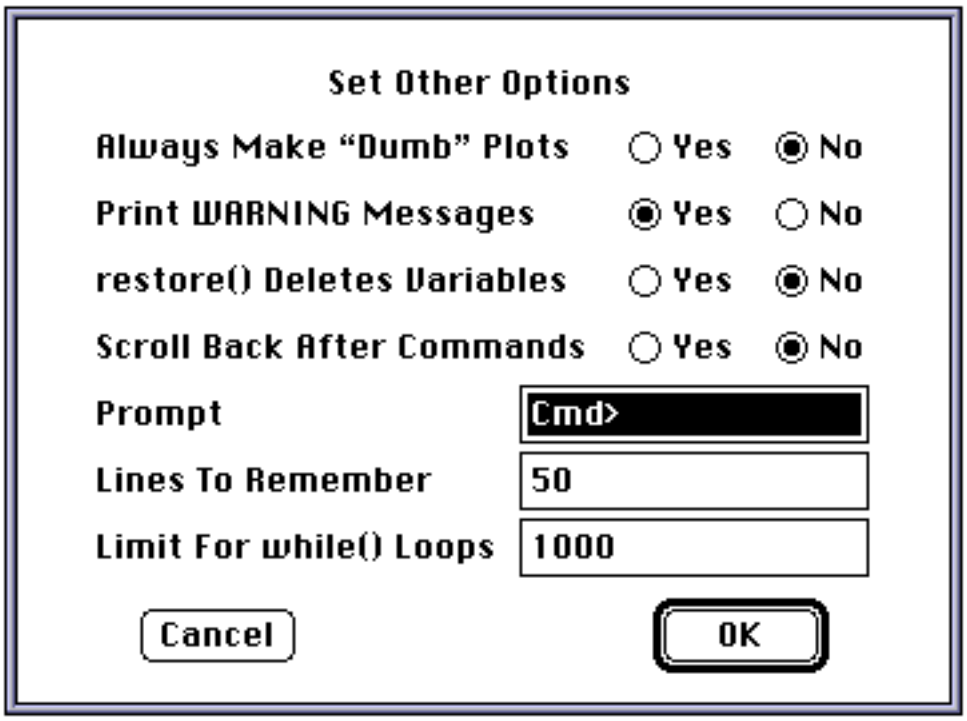
This generates the command `setoptions(pvalues:T or F,fstats:T or F)`, with `T` corresponding to **Yes** and `F` corresponding to **No**.

Batch Options



This generates the command `setoptions(batchecho:T or F,errors:`*n*`)`, with `T` corresponding to **Echo**, `F` corresponding to **No Echo** and *n* the number in the **Error Limit** box. See Sec. 7.6 and 8.2.

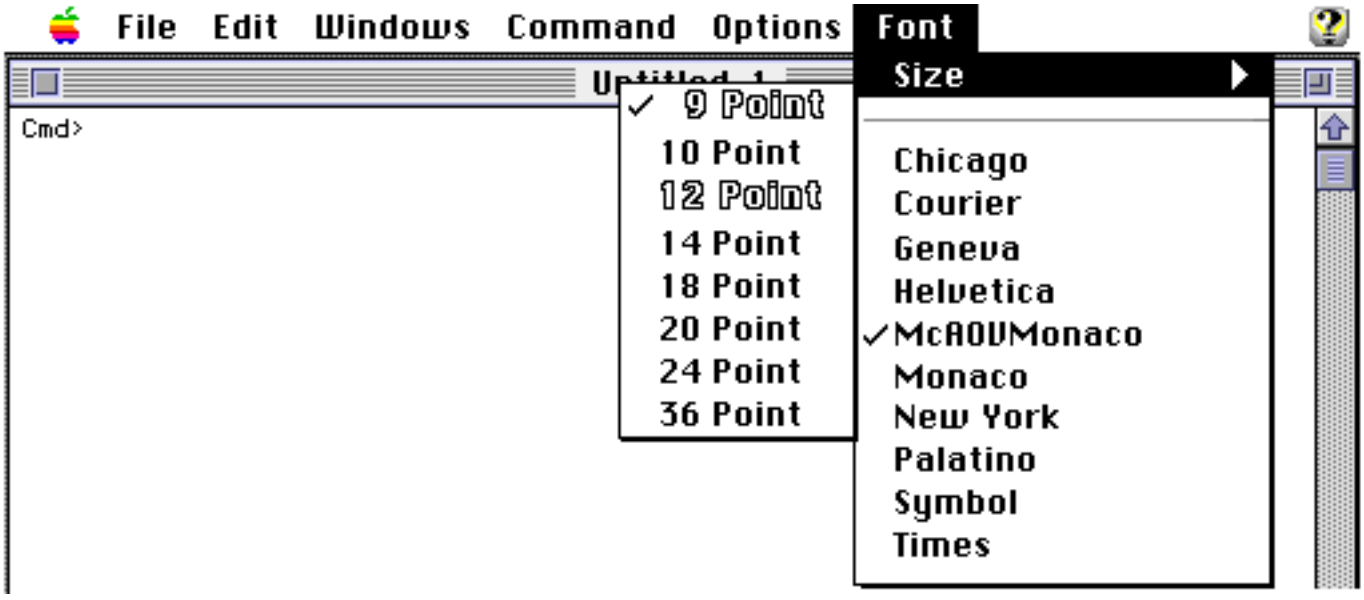
Other Options



This generates a `setoptions()` command to set some or all of options `dumbplot`, `warning`, `restoredel`, `scrollback`, `prompt`, `history` and `maxwhile`. **Yes** and **No** translate to `T` and `F`, respectively. See Sec. 8.1.3.

When you resize a command/output window, options `width` and `height` are automatically changed (see Sec. 8.1.3). These options affect the number of items printed in a line, the size of "dumb" plots and the default size of a stem and leaf display.

B.3.7 **The Font Menu**



The **Font** menu allows you to change the font and font size of everything in the front

command/output window. It also affects the font used in a window subsequently created by **New Window** on the **Windows** menu. The submenu shown is visible only when you select the **Size** item. The entries in ⦶outline⦶ form are the sizes available for the checked font. Except in unusual circumstances (we can't think of any), the font chosen should be non-proportional, that is, all characters having the same width. Of the standard Macintosh fonts, only Courier and Monaco fill the bill. The default font is McAOVMonaco which is built in to MacAnova. It is a special version of Monaco modified so that all characters have distinct shapes so you will be less likely to confuse O (oh) and 0 (zero) or 1 (one), l (el) and I (eye).

You can also change fonts using `setoptions()`. For example,

```
Cmd> setoptions(font:"Courier 18")
```

sets the font to Courier with size 18. An equivalent command would be

```
Cmd> setoptions(font:"Courier", fontsize:18)
```

B.4 **Graph windows** When a plotting command is executed, a new graph window is created, up to a maximum of eight, not counting the panel windows. These are named **Graph 1**, **Graph 2**, ..., **Graph 8**. The appropriate panel window is also updated, **Panel 1-4** if the window is one of **Graph 1**, ..., **Graph 4**, and **Panel 5-8** for **Graph 5**, ..., **Graph 8**. If you bring a panel window to the front and click on one of the "panes", it brings to the front the full regular graph window corresponding to the pane. You can also bring a graph window to the front by pressing, say, ⌘**2** (⌘ key together with 2), which would bring **Graph 2** to the front. Alternatively, you can press ⌘F2. You can bring a panel window to the front or toggle between panel windows by pressing ⌘**G**.

When you try to create a ninth graph window when there are already eight in use, MacAnova prints

```
ERROR: graph windows full - delete a graph and type showplot()
```

Before any more plots can be made, you will have to close at least 1 graph window. When you have freed up a window, type `showplot()` and the graph you wanted will appear there. Alternatively, you can direct the plot to a specific window by, for example,

```
Cmd> showplot(window:3) # See Sec. 8.5.2
```

If the window specified is in use, its contents will be replaced. `window:0` specifies that the plot will be drawn in the window most recently drawn in.

When you close a graph window, the appropriate panel window is updated. If you close all the regular graph windows associated with a panel window, the panel window also goes away.

When a regular graph window or panel window is in front, selecting **Copy** on the **Edit** menu (⌘**C**) copies the entire graph to the Clipboard. From there it can be pasted into the Scrapbook, or into a window belonging to another program such as a word processor or graphics editing program.

When `file:fileName` is an argument to a plotting command, PostScript™ is written to the named file (see Sec. 8.5.4). As always `fileName` can be `""`, bringing up a dialog

box for you to name the file. With the right program (Drop•PS is distributed with MacAnova), the file can be printed directly on a LaserWriter.

When both `file:fileName` and `ps:F` are arguments to a plotting command, a PICT format representation of the graph is written to the file instead of PostScript. Such a file can be imported into many word processors and graphics editing programs.

When `epsf:fileName` is an argument, an encapsulated PostScript file representing the graph is written to the file. This includes a PICT preview of the graph.

When `screendump:fileName` is an argument, the graph is plotted and simultaneously a PICT representation of the graph is written to the file.

B.5 **Location of files** All files associated with MacAnova should be in the same folder as the MacAnova application itself. The most important of these are the help files `MacAnova.hlp`, `Design.hlp` and `Tser.hlp`, and the start up file `MacAnova.ini` (see Sec. 7.8.1). In addition, `MacAnova.mac`, `Tser.mac` and `Design.mac`, the macro files distributed with MacAnova, should also be in this same folder. If they are somewhere else, say in folder `MacAnova Macros` on disk `Macintosh Hard Disk`, you should add the following line to the start up file `MacAnova.ini` (see Sec. 2.11.6 and B.1.3).

```
adddatapath("Macintosh Hard Disk:MacAnova Macros:")
```

Your own data or macro files can be in any folder although you may want to add a similar line to your start up file so that MacAnova can find them. Files created by `save()`, `asciisave()` or `spool()`, or by **Save** on the **File** menu can also go anywhere.

B.6 **Other features**

B.6.1 **Interrupting MacAnova** Some operations such as listing the values in a long vector or a large analysis of variance may take a long time to complete. If you wish to interrupt MacAnova during a long operation or to stop printing to the screen, select **Interrupt** from the **File** menu or press ⌘**.** or ⌘**I**. MacAnova may take a few moments to respond but then will stop what it is doing, print

```
*****  Interrupt  *****
```

and then print a `Cmd>` prompt.

B.6.2 **Running other programs while in MacAnova** You can run other programs without quitting MacAnova. If you are running under System 7 or later or under Multifinder on earlier systems, just click on the desk top to return to the finder and start another application up in the usual manner. If you are not using Multifinder or System 7 or later, you are limited to executing Desk Accessories (DA's) installed in your menu.

Command `shell()` is not implemented on the MacIntosh.

B.6.3 **Editing Macros** You can edit a macro in at least two ways.

(a) Write the macro to a file using, say,

```
Cmd> macrowrite(fileName,mymacro)
```

(Sec. 7.5.2) and then use an editor or word processor to edit the file. Make the changes required and then save it as a text file when you are done. Return to MacAnova (or restart it if necessary) and read the macro back in using

```
Cmd> macroread(fileName,"mymacro")
```

It is a good idea to print it (just type its name) to make sure all of it was read.

(b) Print it to the command / output window by, for example,

```
Cmd> macrowrite(CONSOLE, mymacro)
```

This prints the macro exactly as it would appear on a file. Make any changes needed. Then use the mouse to select the entire macro, including the name line and the trailing line of the form %name% (click at the start of the name line and then, while holding down the Shift key, click at the end of the trailing %name% line). Select **Copy** on the **File** menu (or press ⌘**C**) and execute the following:

```
Cmd> mymacro <- macroread(string:CLIPBOARD)
```

This "reads" the revised macro from the Clipboard. See Sec. 7.5.1 for the form of a macro in a file.

You might find it convenient to first create a new window for editing macros using **New Window** on the **Windows** menu and then do all your editing there.

B.6.4 **Using keys to move around the command/output window** You can use the arrow keys (←, ↑, → or ↓) to move the insertion point in the obvious way. Pressing ⌘ while using an arrow key moves to the left (←) or right (→) end of the line or to the top (↑) or bottom (↓) of the screen. Option ← moves one "word" to the left and Option → moves one "word" to the right. If you hold the shift key down while moving with the arrow keys, everything the cursor moves over is selected.

Option ↑ and Option ↓ do not move the insertion point; they recall previously executed commands to the command line. See Sec. 8.8 and B.6.5.

If your keyboard lacks arrow keys, you can still move the insertion point without using the mouse. Press ⌘**B** (**B**ack) and ⌘**F** (**F**orward) instead of ← and →, respectively

Items **Scroll To Top** and **Go To End** on the **Windows** menu scroll to the start or end of the Command window, respectively. They are selected by Home or ⌘**T** (**T**op) and End or ⌘**E** (**E**nd), respectively. **Go To End** puts the insertion point at the end; **Scroll To Top** does not move the insertion point.

Items **Page Up** and **Page Down** on the **Windows** menu scroll back and forward a screenful without moving the insertion point. **Page Up** is activated by Page up or ⌘**U** (**U**p) and **Page Down** by Page down or ⌘**D** (**D**own).

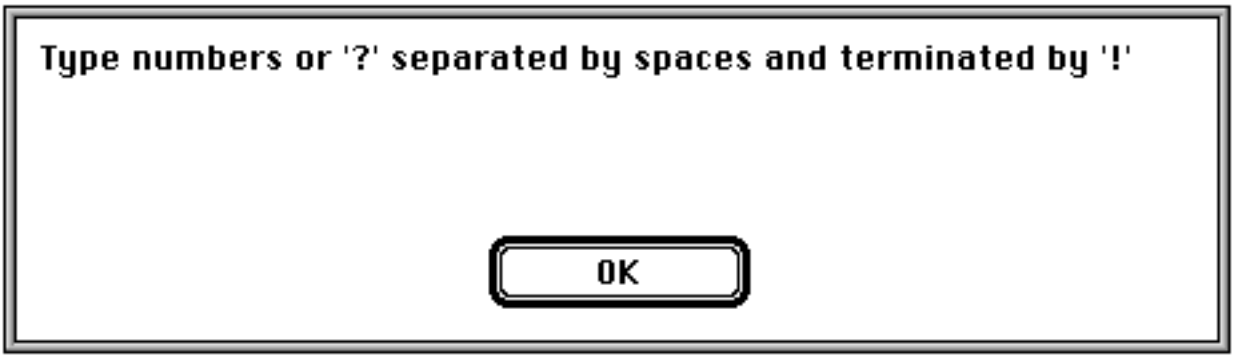
Finally, **Go To Prompt** (⌘**A**) on the **Windows** menu moves the insertion point to immediately after the prompt, before the command line if anything has already been typed.

B.6.5 **Recalling previous commands** You can recall previous commands to the command line using the keyboard (see Sec. 8.8.2). Repeated use of Option ↑ recalls earlier and earlier commands. If you go too far, you can move forward with Option ↓. The number of commands that are remembered is determined by the value of option `history` (see Sec. 8.1.3); the default value is 100. It can be changed by item **Other Options** on the **Options** menu (Sec. B.3.6). For compatibility with the Windows and Motif versions, you can use function keys F7 and F8 instead of Option ↑ and Option ↓.

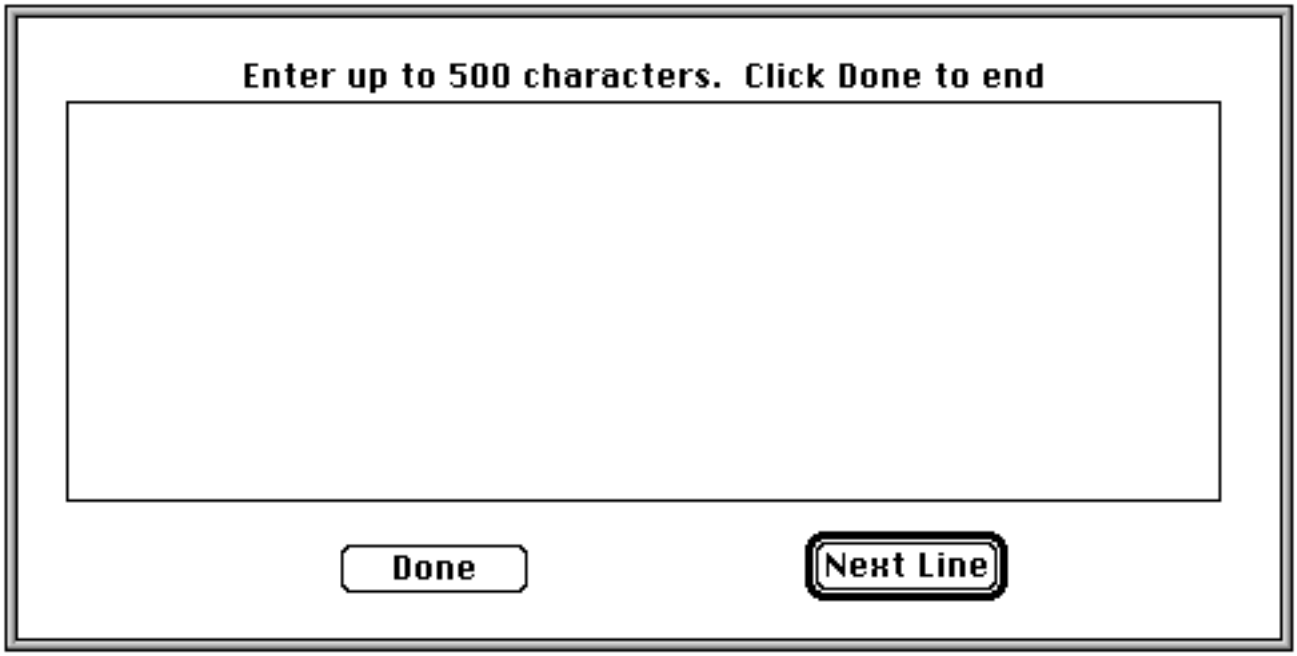
B.6.6 **"Console" input on the Macintosh** You can use `CONSOLE` in place of a file name on functions `vecread()`, `matread()`, `read()` and `macroread()`. For example,

```
Cmd> x <- vecread(CONSOLE)
```

brings up the following dialog box.



After you click **OK**, another dialog box appears:



You can type anything that can be read by `vecread()`. When you press Return or click on **Next Line**, what you have typed is passed on to `vecread()`, and the dialog box reappears. This keeps happening until you type "`!`" or click on **Done**. If you press ⌘ together with Return, an end of line is entered but is not yet passed on to `vecread()`.

Instead of `vecread(CONSOLE)` you can use pre-defined macro `console`:

```
Cmd> x <- console(stop:"$",skip:"#")
```

As illustrated, you can use `vecread()` keyword phrases with `console`.

After `matread(CONSOLE)`, the message in the first dialog box is

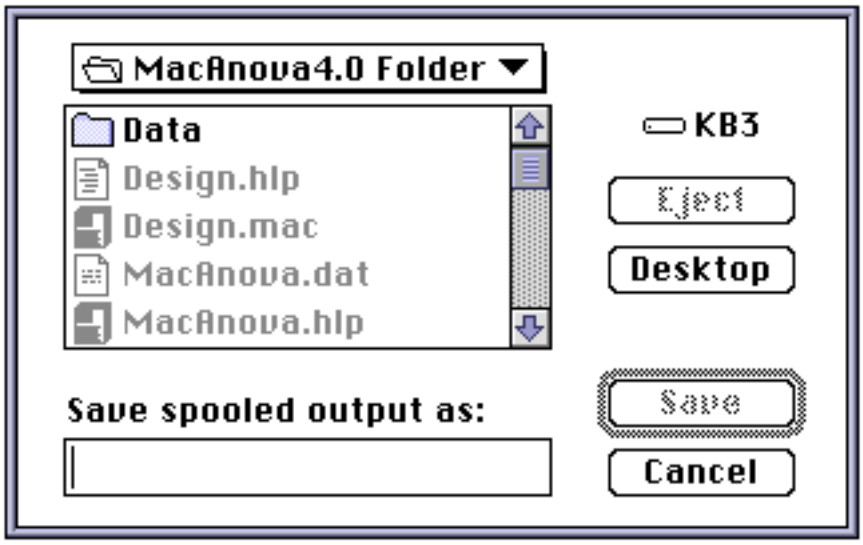
Type header, followed by data

After `macroread(CONSOLE)`, the message in the first dialog box is

Type header, followed by macro

You then type a data set or a macro in the form prescribed in Sec. 7.1 and 7.5.1, complete with a header lines, any necessary comment lines and followed by a line of the form %name%.

B.7 **Non-interactive mode** If you prefer non-interactive statistical computing, you can use MacAnova's non-interactive mode. You enable this by holding down the command key (⌘) while launching MacAnova. When you do, the following dialog box appears.

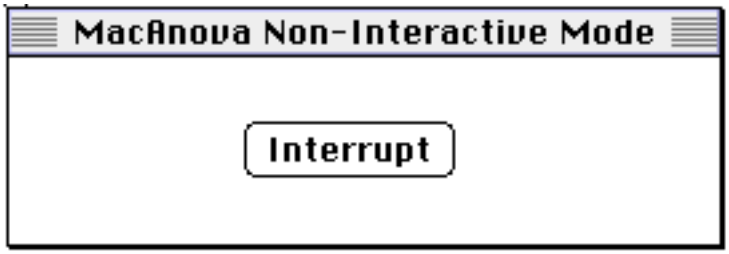


This is the same dialog box as is produced by selecting **Spool Output To File...** on the **File** menu. It allows you to name a file to which all echoed commands and output are to be written. Since such a file is required in non-interactive mode, clicking **Cancel** means execution will be in ordinary interactive mode.

Whether you click on **Save** or **Cancel** you will be presented with a dialog box to select a file of MacAnova commands to execute, exactly as if you selected **Open Batch File...**

on the **File** menu. In non-interactive mode, any attempts to use `spool()` are ignored and any plotting commands will produce "dumb" plots.

Provided you didn't select **Cancel** on the first dialog box, the following dialog box will then appear and remain until the run is done.



You can click on **Interrupt** at any time, although another dialog box gives you a chance to change your mind. When the run is over you are given an opportunity to save the workspace.

If you did click on **Cancel** in the first dialog box, the file of commands you specified will be executed as if you had used `batch()`. Echoed commands and output go in the command/output window. After the last command in the file, a prompt is printed and you can continue interactively.

B.8 **Miscellaneous information**

B.8.1 **Compilers** MacAnova for the Power Macintosh (Motorola PPC 60x processor) is compiled using Metrowerks Code Warrior C compiler. It will not run on an original type Macintosh, such as a Macintosh SE 30 or IIsi, that has a Motorola 680xx (68000, 68020 , 68030 or 68040) processor.

MacAnova for the original type Macintosh is compiled using MPW C. It comes in two versions, one using Motorola 68020 and 68881 math co-processor instructions and one, the no co-processor version, that will run on any 680xx computer. They work identically, but the co-processor version is much faster. They require System 4.2 or later and at least 1 megabyte of ram. The non co-processor version will work on a Power Macintosh but will be much slower than the version specifically compiled for the PPC processor.

B.8.2 **Distribution of MacAnova for Macintosh** As distributed, the PPC application has name of the form `MacAnova4.0x.YYMMDDp`, where `4.0x` is the version number and `YYMMDD` is the date (for example `970820`) of compilation. Similarly the co-processor and no co-processor applications have names of the form `MacAnova4.0x.YYMMDDc` and `MacAnova4.0x.YYMMDDn`, respectively.

MacAnova for the Macintosh is distributed in the form of self extracting archive (SEA) files. When such a file is executed, copies of the original files are created in a folder selected using a dialog box. The diskette distribution contains different SEA files from those available on the internet.

The five basic *diskette* SEA files are MV*mnn*Aux.*r*.sea , MV*mnn*ppc.*r*.sea, MV*mnn*coproc.*r*.sea, MV*mnn*nocoproc.*r*.sea, and MV*mnn*Userfun.*r*.sea, where *mnn* is the version number (*mnn* = 405 means version 4.05) and *r* is the release

number.

Executing MV*m n n* Aux.*r*.sea creates various auxiliary files including help and macro files, plus sample versions of a data and start up file that are needed by all versions.

Executing MV*m n n* ppc.*r*.sea, MV*m n n* coproc.*r*.sea or MV*m n n* nocoproc.*r*.sea create the PPC application, the co-processor application or the no co-processor application, respectively. You probably need only one of these.

Executing MV*m n n* Userfun.*r*.sea creates folder Userfun containing files related to dynamically loaded user functions. You do not need it unless you are planning to program user functions.

The three basic *internet* SEA files are MV*m n n* ppcAll.3.sea, MV*m n n* nocoprocAll.3.sea and MV*m n n* coprocAll.3.sea. When executed each of these creates all the auxiliary files, folder Userfun, and one of the application files and folder. You need only one of them.

File Install.txt distributed with MacAnova gives installation instructions and details the contents of each SEA file.

You can obtain the most recent release of any of these versions through the MacAnova home page with URL

http://www.stat.umn.edu/~gary/macanova/macanova.home.html.

If you prefer to use ftp, the Macintosh files are in

ftp://ftp.stat.umn.edu/pub/macanova/mac.

They are encoded in Binhex form (file name ends with `.Hqx`) which is automatically expanded by most browsers and some ftp programs.

Diskettes and a printed manual are available for nominal cost in the University of Minnesota Bookstore on the St. Paul Campus.