The *Stata Journal* publishes reviewed papers together with shorter notes or comments, regular columns, book reviews, and other material of interest to Stata users. Examples of the types of papers include 1) expository papers that link the use of Stata commands or programs to associated principles, such as those that will serve as tutorials for users first encountering a new field of statistics or a major new technique; 2) papers that go “beyond the Stata manual” in explaining key features or uses of Stata that are of interest to intermediate or advanced users of Stata; 3) papers that discuss new commands or Stata programs of interest either to a wide spectrum of users (e.g., in data management or graphics) or to some large segment of Stata users (e.g., in survey statistics, survival analysis, panel analysis, or limited dependent variable modeling); 4) papers analyzing the statistical properties of new or existing estimators and tests in Stata; 5) papers that could be of interest or usefulness to researchers, especially in fields that are of practical importance but are not often included in texts or other journals, such as the use of Stata in managing datasets, especially large datasets, with advice from hard-won experience; and 6) papers of interest to those who teach, including Stata with topics such as extended examples of techniques and interpretation of results, simulations of statistical concepts, and overviews of subject areas.

For more information on the *Stata Journal*, including information for authors, see the web page

[http://www.stata-journal.com](http://www.stata-journal.com)

The *Stata Journal* is indexed and abstracted in the following:

- Science Citation Index Expanded (also known as SciSearch®)
- CompuMath Citation Index®

**Copyright Statement:** The *Stata Journal* and the contents of the supporting files (programs, datasets, and help files) are copyright © by StataCorp LP. The contents of the supporting files (programs, datasets, and help files) may be copied or reproduced by any means whatsoever, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the *Stata Journal*.

The articles appearing in the *Stata Journal* may be copied or reproduced as printed copies, in whole or in part, as long as any copy or reproduction includes attribution to both (1) the author and (2) the *Stata Journal*.

Written permission must be obtained from StataCorp if you wish to make electronic copies of the insertions. This precludes placing electronic copies of the *Stata Journal*, in whole or in part, on publicly accessible web sites, file servers, or other locations where the copy may be accessed by anyone other than the subscriber.

Users of any of the software, ideas, data, or other materials published in the *Stata Journal* or the supporting files understand that such use is made without warranty of any kind, by either the *Stata Journal*, the author, or StataCorp. In particular, there is no warranty of fitness of purpose or merchantability, nor for special, incidental, or consequential damages such as loss of profits. The purpose of the *Stata Journal* is to promote free communication among Stata users.

The *Stata Journal*, electronic version (ISSN 1536-8734) is a publication of Stata Press. Stata and Mata are registered trademarks of StataCorp LP.
Stata tip 72: Using the Graph Recorder to create a pseudograph scheme

Kevin Crow
StataCorp
College Station, TX
kcrow@stata.com

Following the update of 25 February 2008, the Graph Editor can now record a series of edits, name the recording, and apply the edits from the recording to other graphs. You can apply the recorded edits from the Graph Editor or from the command line. The edits can be applied from the command line when a graph is created, when it is used from disk, or whenever it is the active graph. See Graph Recorder in help graph editor for creating and playing recordings in the Graph Editor. For applying edits from the command line, see help graph play and the option play(recordingname) in help std.options and help graph use.

In this tip, I focus on the use of the Graph Recorder to create a graph scheme. A graph scheme specifies the overall look of the graph. If you want to create your own look for your graphs, you will want to create a scheme file. There is a problem with scheme files, however, because unless you know exactly how you want your scheme to be set up, creating a scheme can be very time consuming.

A shortcut to creating a scheme file is saving a graph recording to disk and replaying that recording on your graphs by using the play() option of the graph command. Using the Graph Recorder to create your graph recording also allows you to tinker with your graph's look on the fly without having to edit a scheme file. Let's walk through an example.

Suppose you want to create several graphs for a report and you want those graphs to have a specific look. To create your recording, you first need to draw the first graph of the report. Try

```
.su yuse auto
(1978 automobile data)
.scatter mpg weight
```

Now that your graph is in the Graph window, you can right-click on the window and select Start Graph Editor to start the Graph Editor. Next start the Graph Recorder by clicking on the Start Recording button, so that you save your changes to memory. Once the graph looks the way you want, you then click on the same button, (which now has the tool tip End Recording), to save your changes to a .grec file. By default, Stata saves .grec files to your PERSONAL/grec directory.

If you want to tinker with the graph during a recording, but you do not want the changes to be saved to the .grec file, click on the Pause button, to temporarily stop saving the changes. To unpause the Recorder, click on the Pause button, again.

© 2008 StataCorp LP
Now that you have a recorder file saved to disk, you can type your next `graph` command in your do-file or from the Command window and apply your scheme to the graph with the `play()` option. For example,

```
.scatter mpg turn, play("test.grec") saving(test1, replace)
```

Also, if you have graphs already created and saved to disk, you can apply your recording to those graphs by using the `play()` option of `graph use`. For example,

```
.graph use "oldfile.gph", play("test.grec") saving("new_file", replace)
```