Style and Record Keeping: "If you cannot retrace your steps from the raw data to the final results, you have no results."

It is important that your programs be readable by humans as well as by machines. They are instructions to the computer on how to manipulate the data and as documentation for skeptics about what you have done. You may have to make changes to your programs months or years after you wrote them, you may want someone else to help get them working, or someone else may inherit your work. To improve readability:

- use long, descriptive variable names. The meaning of total_education = grade_school + university_years; is obvious; te = g + u is not.
- write comments at the top of each program file about what it does
- use comments to document everything that's not obvious from your code .
- if you are merging several data sets from several programs, consider making a flow chart showing where data comes from, which SAS / Stata programs process it, and how various steps rely on other steps. Header comments can describe where data comes from and where it goes.
- Write memos or keep clear written record about data problems and quirks you find and how you fixed them.
- some programs are more important and more likely to be reused or revisited than others; the data set construction code need to be well documented, well organized, and easy to read.
- Simpler, shorter code is easier to read and change. If you are doing the same thing to a large number of similar variables, consider using SAS Arrays and For loops.
- If you need to simplify a variable or break a variable into categories in your output, it is generally better to **format** the existing variable than to create a new one.
- There is a parallel art form to making SAS output compact and easy for others to read. Good titles tell the reader what the output is and what they should be looking for in it. Well designed formats make output compact and easy to interpret. The /norow nocol nopercent options for proc freq remove the row and column percentages from tables and make them simpler, smaller and more readable. Clear variable names or variable labels help.

Professor Severin Borenstein's e-mail discusses many of these issues.

Date: Fri, 17 Oct 2003 19:57:35 -0700 (PDT)
From: Severin Borenstein
Subject: ramblings on how to do empirical work

This has come up a few times in the last couple months, so I thought I would share with you a few thoughts on the nuts and bolts of doing empirical work. (For those of you who do RA work for me, please replace the word "thoughts" with the word "rules". For the non-RAs, feel free to delete.)

1. From the raw data to the final regression, keep a careful log of any changes made to the data: this includes variable tranformations, variable creations, *and* editing of specific observations (such as when an obvious data entry error is replaced). Just keep your text editor open with this data log file in it and flip over to it any time you need to record anything. (Save frequently.) **[One of the great advantages of SAS is that a well organized, well documented SAS program is this kind of documentation. You can fix bad entries using an if …then statement in a SAS data step and to comment clearly what you were fixing.]**

2. Resist the temptation to work interactively with Stata (or any other program). It makes it either difficult or impossible to follow rule #1. If you cannot resist the temptation, start a log file the moment you open up Stata and keep it running until you are finished.

3. You are not finished with any step until you have written a .do file [for Stata; which is the equivalent of a .SAS file for SAS] (or similar for other programs) that takes you from a dataset to either another dataset that ends the step or to the regression that ends the step. If you have failed to resist the evil of interactive data analysis, then you must do penance by reading through your log file (which you […] better have saved) and constructing a .do file that produces the same results.

4. In the end, you should have a raw dataset (or datasets) and a series of programs that transform the raw dataset(s) to the final dataset and then to the final results. Occasionally, you will need the data log you've been keeping to remind you of, for instance, the data substitution you made when observation 1483 had a value of -93 that obviously should have been 93, so you changed it by hand. That's what the data log is for.

5. If you cannot retrace your steps from the raw data to the final results, you have no results. This sounds straightforward, but is surprisingly difficult to adhere to even if you keep good records. If you don't keep good records, you are lost.....and your work will be no more reliable than the stuff produced by <insert here name of flim-flam "empirical IO economist">.