Computing and Software:
Editor’s Notes

JOSEPH HILBE
Arizona State University

This review of the Stata statistical and data management system represents the first such review for this column. Others are planned for future issues. I invite readers to consider authoring reviews, as well as serving as referees. The aim is to provide an overview of software packages which are used in health care and health policy analysis. For those interested in contributing to this column, contact me at either jhilbe@aol.com or hilbe@asu.edu

This initial review should not be thought of as a paradigm on which to base other such reviews. In fact, this review is quite unusual. Typically a reviewer is someone who has never been connected with the software in question. This case is an exception. I did have a working relationship with Stata from 1990 to 1993. I know the software quite well. I also know its history, its strengths, and the areas where there are shortcomings. I work with it on a near daily basis. But, having no vested interest in the vendor, nor of the commercial success of the package, I believe that my remarks may provide the reader with an insight that is not typically obtained in the normal review process.

STATA: AN HISTORICAL OVERVIEW

In this review I shall provide a brief history of the Stata statistical package, and inform the reader—especially those who know little of the program—why it has grown from a small two-person corporation some ten years ago to one of the most currently well used packages worldwide. In the process I shall provide an overview of Stata’s capabilities, as well as its present limitations.

Stata was originally written by Bill Gould, the current CEO, and Bill Rogers beginning in about 1985 under the vendor name, Computing Resource Center (CRC). First located in Southern California, the authors attempted to develop a package which highlighted speed and graphics. They also intended to develop a comprehensive data management system. Today, Stata still emphasizes speed and data management; but in addition, they have made a point of developing a truly exhaustive host of regression routines. Efforts in graphical capability have not kept pace. This is not to say that Stata has poor graphics, but rather that this early emphasis has been surperceded by other concerns. It is my understanding that graphics will again be put on the front-burner for the next release.

Stata is written as a single executable which fits on one 1.4 Meg diskette. A number of ASCII text programs accompany the executable; i.e. are run by the executable. Originally these text files allowed the programmers and users to write batch-type algorithms to manage
data. They were aptly termed “do” files. One could write a do file to execute a string of commands, typically dealing with the interaction of data management and statistical analysis. After a couple of version upgrades, the authors developed the notion of an automatic “do” file, or “ado” file for short. This type of program could be called directly from the command line and would sequentially execute the commands listed in the text file. A feature added to the “ado” program ended up changing the future direction of the package. The ado file allowed while-loops and if-then branching. From that point on, the authors could write statistical routines which, on the surface, were independent of the executable.

By 1990, Stata had a well developed data management system, good graphics—especially for 1990—and a limited number of statistical routines. Included were the standard host of univariate statistics, linear regression, ANOVA, logistic and Cox regression, together with associated Kaplan-Meier statistics, and some quality control features. The software also included several excellent run-time tutorials. Its clear emphasis was in the area of biostatistics.

In mid-1990, Henry Krakauer of the Health Care Financing Administration (HCFA) recommended that Stata be used as the management and statistical system for the Peer Review Organization (PRO) analysis of Medicare data. Sponsored under the so-called Medicare Infrastructure Project, PROs were mandated to use Stata for all analysis reports during their 4th Scope of Work beginning in 1993. This provided Stata with a boost into national prominence.

I was the Infrastructure team’s senior biostatistician, and Rogers served as a consultant. Roger’s task was to develop a 3-parameter Bailey-Makeham survival model using both C and Stata code. One of my roles was to extend Stata statistical offerings for use by PRO analysts. This was accomplished by using the ado file format to develop a variety of statistical routines.

Several statistical programs had already been written by users outside of Stata. Krakauer had co-authored an eloquent life table ado program, and Larry Hamilton, a sociologist at the University of New Hampshire, had written a robust regression routine. I had already contributed by writing logistic regression extensions based on the work of Hosmer and Lemeshow. It became acutely obvious that Stata’s programming language could provide users with far more analytic capability than was found in the base package.

Many statistical packages have associated programming languages. Notable macros have been developed in SAS and in GLIM and in a variety of other packages. In fact, the first GEE algorithm was developed as a not-so-simple SAS macro. But whereas most of these packages considered macros as an ancillary feature to the base package, Stata seemed to be designed, even if not intentionally at first, as giving preeminent importance to the macro or, in Stata language, to ado files per se. And to be sure, at least 75% of Stata’s current scope of statistical routines are written as ado files rather than as hard-coded into the executable. But this was not the case at the beginning.

In March 1991 I made a proposal to Gould that CRC sponsor a technical journal or bulletin, to be called the Stata Technical Bulletin (STB), which could serve as a forum for those who desired to write and share statistical and data management programs which they had written using Stata’s programming language. He enthusiastically supported the notion,