1 Introduction

This book focuses on the tools needed to carry out applied econometric research in economics and finance. These include both the theoretical foundations of econometrics and a solid understanding of how to use those econometric tools in the research process. That understanding is motivated in this book through an integration of theory with practice, using Stata on research datasets to illustrate how those data may be organized, transformed, and used in empirical estimation. My experience in working with students of econometrics and doctoral candidates using econometric tools in their research has been that you learn to *use* econometrics only by *doing* econometrics with realistic datasets. Thankfully, a growing number of introductory econometrics text-books[1] follow this approach and focus on the theoretical aspects that are likely to be encountered in empirical work. This book is meant to complement those textbooks and provide hands-on experience with a broad set of econometric tools using Stata.

The rest of this chapter presents my "top 11" list of Stata's distinctive features: aspects of Stata's design and capabilities that make the program an excellent tool for applied econometric research. Sections 1.2 and 1.3 provide essential information for those who want to execute the examples used in the text. Many of those examples use user-written Stata commands that must be installed in your copy of Stata. A convenience program described in that section, `itmeus`, will make doing so a painless task.

1.1 An overview of Stata's distinctive features

Stata is a powerful tool for researchers in applied economics. Stata can help you analyze research easily and efficiently—no matter what kind of data you are working with—whether time-series, panel, or cross-sectional data. Stata gives you the tools you need to organize and manage your data and then to obtain and analyze statistical results.

For many users, Stata is a statistical package with menus that allow users to read data, generate new variables, compute statistical analyses, and draw graphs. To others, Stata is a command line–driven package, commonly executed from a do-file of stored commands that will perform all the steps above without intervention. Some consider Stata to be a programming language for developing ado-files that define programs or new Stata commands that extend Stata by adding data-management, statistics, or graphics capabilities.

---

1. E.g., Wooldridge (2006) and Stock and Watson (2006).

Understanding some of Stata's distinctive features will help you use Stata more effectively and efficiently. You will be able to avoid typing (or copying and pasting) repetitive commands and constantly reinventing the wheel. Learning to write computationally efficient do-files (say, one that runs in 10 seconds rather than in 2 minutes) is helpful, but more importantly you need to be able to write do-files that can be easily understood and modified. This book will save you time by teaching you to generate comprehensible and extensible do-files that you can rerun with one command.

Consider several of Stata's distinctive features, which I discuss in more detail later:

You can easily learn Stata commands, even if you do not know the syntax. Stata has a dialog for almost every official command, and when you execute a command with a dialog, the Review window displays the command syntax, just as if you had typed it. Although you can submit a command without closing the dialog, you will often want to execute several commands in succession (for instance, generating a new variable and then summarizing its values). Even if you are using Stata dialogs, you can reissue, modify, and resubmit commands by using the Review and Command windows. You can save the contents of the Review window to a file or copy them into the Do-file Editor window so that you can modify and resubmit them. To use these options, control-click or right-click on the Review window.

You can use Stata's Do-file Editor to save time developing your analysis. Once you are familiar with common commands, you will find it easier to place them in a do-file and execute that file rather than entering them interactively (using dialogs or the Command window). Using your mouse, you can select any subset of the commands appearing in the Do-file Editor and execute only those commands. That ability makes it easy to test whether these commands will perform the desired analysis. If your do-file does the entire analysis, it provides a straightforward, reproducible, and documented record of your research strategy (especially if you add comments to describe what is being done, by whom, on what date, etc.).

A simple command performs all computations for all the desired observations. Stata differs from several other statistical packages in its approach to variables. When you read in a Stata dataset, Stata puts in memory a matrix with rows corresponding to the observations and columns representing the variables. You can see this matrix by clicking the Data Viewer or Data Editor icon on Stata's toolbar. Most Stata commands do not require you to explicitly specify observations. Unlike with other statistical package languages, few circumstances in Stata require you to refer to the specific observation, and Stata will run much faster if you avoid doing so. When you must explicitly refer to the prior observation's value—for instance, when you are generating a lagged value in time-series data—*always* use Stata's time-series operators, such as `L.x` for the lagged value of `x` or `D.x` for the first difference.

Looping over variables saves time and effort. One of Stata's most valuable features is the ability to repeat steps (data transformations, estimation, or creating graphics) over several variables. The relevant commands are documented in [P] **forvalues**, [P] **foreach**, and [P] **macro**; see the online help (e.g., `help forvalues`) and

appendix B for more details. Using these commands can help you produce a do-file that will loop over variables rather than issuing a separate command for each one; you can easily modify your file later if you need a different list of variables; see chapter 2.

Stata's by-groups reduce the need for programming. Stata lets you define *by-groups* from one or more categorical (integer valued) variables, so you can do sophisticated data transformations with short, simple commands; see chapter 2.

Stata has many statistical features that make it uniquely powerful. Stata can calculate robust and cluster–robust estimates of the variance–covariance matrix of the estimator for nearly all the estimation commands.[2] The `mfx` command estimates marginal effects after estimation. `test`, `testnl`, `lincom`, and `nlcom` provide Wald tests of linear and nonlinear restrictions and confidence intervals for linear and nonlinear functions of the estimated parameters.

You can avoid problems by keeping Stata up to date. If you have an Internet connection, Stata's [R] **update** facility periodically updates Stata's executable and ado-files, free of charge. Most updates contain bug fixes and enhancements to existing commands (and sometimes brand-new commands). To find available updates, use the command `update query` and follow its recommendations. Many problems identified by Stata users have already been addressed by updates, so you should always update your Stata executable and ado-files before reporting any apparent error in the program. Be sure to update your copy of Stata when you reinstall the program on a new computer or hard disk since the installation CD contains the original code (i.e., version 9.0 without updates versus version 9.2 with updates, which is available at this writing).

Stata is infinitely extensible. You can create your own commands that are indistinguishable from official Stata commands. You can add a new command to Stata, whether you or someone else developed it, by writing an ado-file and help file. Any properly constructed ado-files on the `adopath` will define new commands with those names, so Stata's capabilities are open ended (see [P] **sysdir**). Since most Stata commands are written in the do-file language, they are available for viewing and modification, and they demonstrate good programming practice.

Stata's user community provides a wealth of useful additions to Stata. StataCorp's development strategy gives users the same development tools used by the company's own professional programmers. This practice has encouraged a vibrant user community of Stata developers who freely share their contributions. Although any Stata developers may set up their own `net from` sites, most user-written programs are available from the Statistical Software Components (SSC) archive that I maintain at Boston College, which you can access by using Stata's `ssc` command; see [R] **ssc**. You can use a web browser to search the SSC archive, but you should use the `ssc` command to download any of its contents to ensure that the files are handled properly and installed in the appropriate directory. Typing `ssc whatsnew` lists recent additions and updates in the SSC archive. Typing `adoupdate` updates the packages you have installed from the SSC archive, the *Stata Journal*, or individual users' sites as needed.

---

2. Don't worry if you do not know what these are; I discuss them in detail in the text.

Stata is cross-platform compatible. Unlike many statistical packages, Stata's feature set does not differ across the platforms (Windows, Macintosh, Linux, and Unix) on which it runs. The Stata documentation is not platform specific (with the exception of the *Getting Started with Stata* manuals). A do-file that runs on one platform will run on another (as long as each system has enough memory). This compatibility allows you to move binary data files easily among platforms: that is, all Stata `.dta` files have the same binary data format, so any machine running the same version of Stata can read and write to those files. Stata can also read a data file stored on a web server with the command `use http://...` regardless of platform.

Stata can be fun. Although empirical research is serious business, you need only follow a few threads in Statalist[3] discussions to learn that many users greatly enjoy using Stata and participating in the Stata user community. Although learning to use Stata effectively—like learning to speak a foreign language—is hard work, learning to solve data-management and statistical analysis problems is rewarding. Who knows? Someday your colleagues may turn to you, asking for help with Stata.

1.2   Installing the necessary software

This book uses Stata to illustrate many aspects of applied econometric research. As mentioned, Stata's capabilities are not limited to the commands of official Stata documented in the manuals and in online help but include a wealth of commands documented in the *Stata Journal*, *Stata Technical Bulletin*, and the SSC archive.[4] Those commands will not be available in your copy of Stata unless you have installed them. Because the book uses several of those user-written commands to illustrate the full set of tools available to the Stata user, I have provided a utility command, `itmeus`, that will install all the unofficial commands used in the book's examples. To install that command, you must be connected to the Internet and type

```
ssc install itmeus
```

which will retrieve the command from the SSC archive. When the `ssc` command succeeds, you may type

```
help itmeus
```

as you would with any Stata command, or just

```
itmeus
```

to start the download procedure. All necessary commands will be installed in your copy of Stata. Any example in the book (see the next section to obtain the do-files and datasets used to produce the examples) may then be executed.

---

3. See http://www.stata.com/statalist/.
4. Type `help ssc` for information on the SSC ("Boston College") archive.

Newer versions of the user-written commands that you install today may become available. The official Stata command `adoupdate`, which you may give at any time, will check to see whether newer versions of these user-written commands are available. Just as the command `update query` will determine whether your Stata executable and official ado-files are up to date, `adoupdate` will perform the same check for user-written commands installed in your copy of Stata.

1.3   Installing the support materials

Except for some small expository datasets, all the data I use in this book are freely available for you to download from the Stata Press web site, http://www.stata-press.com. In fact, when I introduce new datasets, I merely load them into Stata the same way that you would. For example,

```
. use http://www.stata-press.com/data/imeus/tablef7-1.dta, clear
```

Try it.

To download the datasets and do-files for this book, type

```
. net from http://www.stata-press.com/data/imeus/
. net describe imeus
. net get imeus-dta
. net get imeus-do
```

The materials will be downloaded to your current working directory. I suggest that you create a new directory and copy the materials there.