Preface

Our goal in writing this book was to make it routine to carry out the complex calculations necessary to fully interpret regression models for categorical outcomes. Interpreting these models is complex because the models are nonlinear. Most software packages that fit these models do not provide options that make it simple to compute the quantities that are useful for interpretation. In this book, we briefly describe the statistical issues involved in interpretation, and then we show how you can use Stata to make these computations. As you read this book, we strongly encourage you to be at your computer so that you can experiment with the commands as you read. To facilitate this, we include two appendices. Appendix A summarizes each of the commands that we have written for interpreting regression models. Appendix B provides information on the datasets that we use as examples. We have also written a command called `spex`, standing for Stata postestimation examples, which makes it simple to load the datasets and run the sample programs from our book. We find this incredibly useful for teaching and for our own exploration of the methods.

Many of the commands that we discuss are not part of official Stata, but instead they are commands (in the form of ado-files) that we have written. To follow the examples in this book, you must install these commands. Details on how to do this are given in chapter 2. Although the book assumes that you are using Stata 9 or later, most commands will work in Stata 8 or Stata 7, but some of the output will appear differently. For details, see *http://www.indiana.edu/~jslsoc/spost.htm*. The screenshots that we present are from Stata 9 for Windows. If you are using a different operating system, your screen might not look the same. See the StataCorp publication *Getting Started with Stata* for your operating system for further details. All the examples, however, should work on all computing platforms that support Stata.

We use several conventions throughout the book. Stata commands, variable names, filenames, and output are presented in a typewriter-style font; for example, `logit`, `lfp`, `age`, `wc`, `hc`, and `k5`. Italics are used to indicate that something should be substituted for the word in italics. For example, `logit` *variablelist* indicates that the command `logit` is to be followed by a list of variables. When output from Stata is shown, the command is preceded by a period (which is the Stata prompt). For example,

```
. logit lfp age wc hc k5, nolog
Logistic regression                              Number of obs   =        753
  (output omitted)
```

If you want to reproduce the output, you do not type the period before the command. And, as just illustrated, when we have deleted part of the output, we indicate this with (*output omitted*). Keystrokes are set in this font. For example, Alt-f means that you are to hold down the Alt key and press f. The headings for sections that discuss advanced topics are tagged with an asterisk (*). These sections can be skipped with no loss of continuity with the rest of the book.

As we wrote this book and developed the accompanying software, many people provided suggestions and commented on early drafts. In particular, we thank Jun Xu, Ben Jann, Tait Medina, Simon Cheng, Claudia Geist, Lowell Hargens, and Patricia McManus. Pravin Trivedi was always willing to discuss problems and to provide sage advice. Many people at StataCorp provided their expertise in many ways. More than this, though, we are grateful for their engaging and encouraging our project in ways that are exemplary for academic publishing and that have made working with them enjoyable throughout. We particularly thank Richard Gates, Lisa Gilmore, Jeff Pitblado, and Gabe Waggoner. Last, we are indebted to David Drukker at StataCorp for his encouragement, valuable advice, and patience throughout the many stages of this project, not to mention his gentle way of imposing deadlines.

What's new in the second edition

The second edition has many changes, both large and small. Many of the changes reflect enhancements made in Stata 9. We discuss new regression models, including the zero-truncated Poisson and the zero-truncated negative binomial models, the hurdle model for counts, the stereotype logistic regression model, the rank-ordered logit model, and the multinomial probit model. We also discuss new Stata commands, such as `estat`, which provides a uniform way to access statistics useful for postestimation interpretation.

Also we have extended our suite of programs, known as SPost. The biggest change, and the most requested, is the inclusion of confidence intervals for predictions computed by `prvalue` and `prgen`. This work, completed with Jun Xu of Indiana University, has many applications that are illustrated throughout the book. We also added some new commands. `misschk` is a tool for examining the patterns of missing data. `leastlikely` provides a simple way to find observations that your model does not fit well. `asprvalue` computes predicted probabilities for the conditional logit model and other models for alternative-specific data that are discussed in the new chapter 7. And, `countfit` provides many measures of fit for comparing count models. Other commands have been expanded to work with more models, and we have added options that make the commands easier to use. (Users who complained about getting the tick marks right when using `prgen` will particularly appreciate the `gap` option!) Unfortunately, and despite encouragement from many users, SPost does not work with all the regression models fitted by Stata.

Getting help from the authors

We are gratified that many people have bought our book, but as a consequence, we have received many emails with questions and suggestions. We would like to respond to everyone who contacts us, but our time is limited. Here are things that you can do to make it easier for us to answer your questions, which will also increase the odds that you will get a prompt answer.

1. Make sure that you have the latest version of the Stata executable and ado-files. See [U] **28 Using the Internet to keep up to date** and [R] **update**.
2. Make sure that you have the last version of SPost. See page 9.
3. Check *http://www.indiana.edu/~jslsoc/spost.htm* for any new information.
4. Make sure that you do not have anything but letters, numbers, and underscores in your value labels. Some commands get hung up when value labels include special characters.
5. Look at the sample files in the `spost9_do` package. It is sometimes easiest to figure out how to use a command by seeing how others use it.

If none of these suggestions solves your problem, send us an email at *spostsup@indiana.edu*.

It is hard to figure out some problems by seeing just the log file. So, we suggest that you send a do-file, the resulting log file, and a small dataset (extract cases and variables from your full data).

1. Do not refer to specific directories. For example, do not include something like `use c:\data\project3\sample.dta` because it will not run on our machines unless we either edit your file or create your directory structure. The same is true if you use something like `log using c:\data\project3\problem, text`.
2. Send output in text rather than SMCL format. To do this, add the `text` option to your `log` command.
3. Include `prwhich` at the start of your do-file. This tells us which versions of the commands you are using. Then include `about` to tell us which version of Stata you are using.
4. Do not send a `.zip` file since the Indiana University email server will reject them. If you compress the data, change the suffix from `.zip` to something else.

Here is an example of what a do-file might look like:

```
capture log close
log using yourname.log, text replace
* prchange generates a variable not found error.
* scott long - jslong@indiana.edu - 4July2005
about
which prchange
use sample.dta
logit y x1 x2
* the following command causes the error
prchange, x(x1=1 z2=3)
log close
```

Wisconsin and Indiana *Jeremy Freese*
November 2005 *Scott Long*