

THE STATA NEWS

April/May/June 2010

Vol 25 No 2

Stata 11.1 now available

Learn some of the new features available in the latest free update to Stata 11.

p. 1

Stata makes a difference

Get an inside look into how Stata is used by the World Bank.

p. 2

Data Management Using Stata: A Practical Handbook

Read about the new data-management book by Michael N. Mitchell.

p. 3

In the spotlight

Learn about factor variables and the generalized method of moments in Stata.

p. 4

Stata/MP Performance Report updated

Discover where all the details about Stata/MP are explored.

p. 7

Stata Conference Boston 2010

Make plans to attend the annual Stata Conference in Boston.

p. 12

Also in this issue

Public training courses	8
New from the Stata Bookstore	9
Users Group meetings	11
Upcoming NetCourses	11

The Stata News

Executive Editor: Brian Poi
Production Supervisor: Annette Fett

Stata 11.1 now available

Stata 11.1 is now available as a free update to Stata 11. It adds several new features and extensions to existing features. If you have Stata 11, just type **update all** in Stata, and then type **update swap**. Or select **Help** from the main Stata menu, then select **Official Updates** and follow the instructions in the resulting Viewer window.

Here are some of the new features in Stata 11.1:

Multiple imputation. The **mi** command now officially supports fitting panel-data and multilevel models.

Truncated count-data models. New commands **tpoisson** and **tnbreg** fit models of count-data outcomes with any form of left truncation, including truncation that varies by observations.

Mixed models. Linear mixed (multilevel) models have new covariance structures for these residuals: exponential, banded, and Toeplitz.

Probability predictions. **predict** after count-data models, such as **poisson** and **nbreg**, can now predict the probability of any count or any count range.

Survey bootstrap. Estimation commands can now estimate survey bootstrap standard errors (SEs) using user-supplied bootstrap replicate weights.

Survey SDR weights. Successive difference replicate (SDR) weights are now supported when estimating with survey data. These weights are supplied with many datasets from the United States Census Bureau.

Concordance. **estat concordance** adds a new measure of concordance, Gönen and Heller's *K*, that is robust in the presence of censoring.

Survey GOF. Goodness-of-fit (GOF) tests are now available after **probit** and **logistic** estimates on survey data.

Robust SEs. Cluster-robust SEs have been added to **xtpoisson**, **fe**.

Survey CV. The coefficient of variation (CV) is now available after estimation with survey data.

Estimation formatting. Numerical formats can now be customized on regression results. You can set the number of decimal places for coefficients, SEs, *p*-values, and confidence intervals using either command-line arguments or the **set** command.

Settings have also been added to control the display of factor variables in estimation tables. These display settings include extra space around factor variables, display of empty cells, display of base levels, and omitting variables that are excluded because of collinearity.

Stata/MP performance. Stata/MP, the parallel version of Stata, has several performance improvements, including even more parallelized panel-data estimators, improved parallelization of estimations with more than 200 covariates, and improved tuning of MP on large numbers of processors/cores.

Clipboard improvements. Clipboard support in the Data Editor has been enhanced. Copies to the Clipboard now retain variable formats and other characteristics of the data when pasted from within Stata.

Continued on p. 2

Windows XP. The amount of memory available to Stata has been increased on 32-bit Windows XP.

Do-file Editor. Syntax highlighting, bookmarks, and other Do-file Editor features have been added to Stata for Mac.

ODBC. ODBC support has been added for Sun Solaris (Oracle Solaris).

Dialog boxes. Dialog boxes on Unix now have varlist controls that allow you to select variables from a list of variables in your dataset.

To learn more, type **help whatsnew** after updating to 11.1 and follow the links to the individual commands.

If you have not already upgraded to Stata 11, you are also missing out on a host of new features, including:

- Multiple imputation (with its own manual)
- Generalized method of moments (GMM) estimation
- Competing-risks regression
- Factor variables
- State-space models
- Marginal means and predictive margins
- Average marginal effects
- Greek letters, italics, and other fonts in graphs
- Dynamic-factor models
- Multivariate GARCH models
- Error covariance structures in linear mixed models
- Panel-data unit-root tests
- Variables Manager
- New Data Editor
- New Do-file Editor
- Object-oriented programming in Mata
- More commands parallelized in Stata/MP
- PDF documentation

Find out more about Stata 11 at

www.stata.com/stata11/

.....

Stata makes a difference at the World Bank: Automated poverty analysis

The World Bank supports the United Nations Millennium Development Goals of eliminating poverty and providing for sustained development. To ameliorate poverty in an area, one must first know who is most affected by poverty and how poverty is distributed among society's members.

Poverty Assessments are key to the World Bank's poverty-reduction strategy. These reports are routinely produced for virtually every country the Bank studies. Each Poverty Assessment includes various statistics on poverty and income inequality and reports on how well each country is achieving its poverty-reduction targets. Historically, producing a Poverty Assessment for a country would involve hiring a consultant, often a newly minted PhD or a graduate student. The consultant would learn the principles of poverty analysis and write Stata programs to produce the requisite tables and graphs. This approach was prone to error because no standards were in place; instead, Stata programs and documentation were produced by people with varying degrees of skill. Methodologies and assumptions were often vague, and results were difficult to replicate. Maintaining the code and preparing data were costly procedures. To do an analysis similar to an existing one, a researcher would often have to start from scratch rather than reuse the existing code.

Poverty Analysis Toolkit

To rectify and streamline the process of producing Poverty Assessments, Michael Lokshin, a lead economist in the Development Research Group at the World Bank, and his team, including Sergiy Radyakin and Zurab Sajaia, wrote a set of ado-files to implement various poverty measurement and analysis algorithms. These ado-files eventually became known as the Poverty Analysis Toolkit, which was widely used throughout the World Bank. The popular user-written command **xml_tab**, available from the Statistical Software Components (SSC) archive, also grew out of this work; **xml_tab** allows users to save Stata results in a format that is easily incorporated into Microsoft Excel spreadsheets.

The Poverty Analysis Toolkit includes several programs for dynamic policy analysis, including commands for plotting growth incidence curves; for plotting poverty incidence, deficit, and severity curves; and for analyzing the changes in poverty over time that are due to sectoral and population changes, and growth and redistribution.

The Toolkit greatly simplified the study of poverty at the World Bank by making available a standard set of Stata commands that researchers could use without having to reinvent the wheel. However, having a collection of programs instead of a single interface raised the learning curve for new researchers and limited researchers' ability to produce standard output that could easily be included in reports.

ADePT

To make the Poverty Analysis Toolkit appeal to a wider audience, Lokshin and his team decided to combine the separate routines and to provide a single easy-to-use graphical interface. The Toolkit was renamed ADePT (which stands for Automated DEC Poverty Tables) and was quickly adopted by researchers around the world. In contrast to the Toolkit, the ADePT software, available at www.worldbank.org/adept/, is no longer a set of isolated components, but rather an integrated platform. Having an integrated platform allows the components to work together and simplifies the development of additional modules.

ADePT was developed using a combination of Stata's ado-language, Mata, and dialog programming language, including over 150,000 lines of code. Certain routines were also developed in C++ and assembly language for maximum performance and used Stata's plug-in facilities. One example of such a routine is the **usespss** command, which is available from the SSC archive; this command allows Stata users to read datasets in SPSS format.

Different modules within ADePT perform an array of statistical analyses, from simple cross-tabulations to estimation of simultaneous equations via maximum likelihood. Routines

In the spotlight: Factor variables

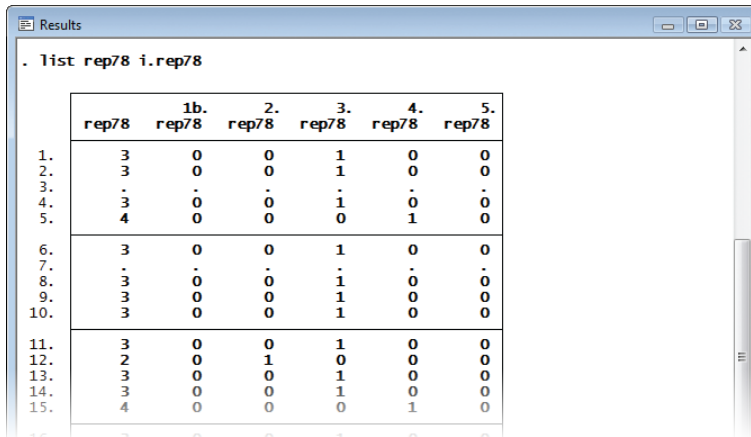
Factor variables were introduced in Stata 11. I like them for many reasons, including:

1. They save space.
2. They make full factorial specifications easy.
3. They work almost everywhere, which is convenient.
4. They make short work of tests of structural change.

First, in case you have not been using factor variables already, I will introduce them. Any categorical variable in Stata can be treated as a factor variable. (Well, not exactly *any* variable—the variable must take on only nonnegative integer values.) Often we do not want to treat a categorical variable as cardinal or ordinal, but rather as a set of indicator variables—one for each level (value) that the variable takes on. We do that using Stata's factor-variable operators.

The most commonly used operator is `i.`. There are other operators, and we will see a few of them in this article. Using the venerable `auto.dta`, when we type `i.rep78`, we are actually referring to five indicator variables—one for each level of `rep78`. (Yes, one of those is a base level, which is special, but I am not going to get into that here.) We can see the indicators by typing

```
. sysuse auto
. list rep78 i.rep78
```



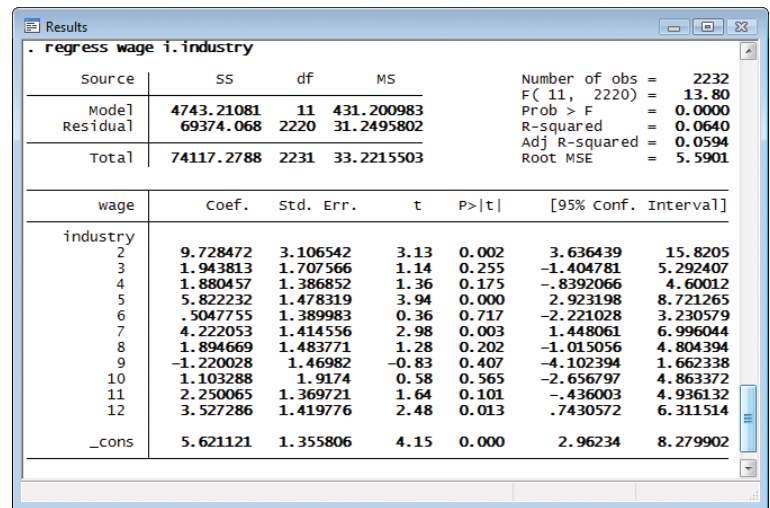
	rep78	1b. rep78	2. rep78	3. rep78	4. rep78	5. rep78
1.	3	0	0	1	0	0
2.	3	0	0	1	0	0
3.
4.	3	0	0	1	0	0
5.	4	0	0	0	1	0
6.	3	0	0	1	0	0
7.
8.	3	0	0	1	0	0
9.	3	0	0	1	0	0
10.	3	0	0	1	0	0
11.	3	0	0	1	0	0
12.	2	0	1	0	0	0
13.	3	0	0	1	0	0
14.	3	0	0	1	0	0
15.	4	0	0	0	1	0

Now, back to my partial list of reasons I like factor variables.

1. They save space

A dataset from the United States National Longitudinal Survey of Women can be loaded into Stata by typing `sysuse nlsw88`. Among the variables in that dataset is the 12-level variable `industry`, which represents the industry in which a woman was working, such as manufacturing or real estate. To run a regression on wages by industry, I can type

```
. regress wage i.industry
```



Source	SS	df	MS			
Model	4743.21081	11	431.200983			
Residual	69374.068	2220	31.2495802			
Total	74117.2788	2231	33.2215503			

	Coef.	Std. Err.	t	P> t	[95% Conf. Interval]	
industry						
2	9.728472	3.106542	3.13	0.002	3.636439	15.8205
3	1.943813	1.707566	1.14	0.255	-1.404781	5.292407
4	1.880457	1.386852	1.36	0.175	-.8392066	4.60012
5	5.822232	1.478319	3.94	0.000	2.923198	8.721265
6	.5047755	1.389983	0.36	0.717	-2.221028	3.230579
7	4.222053	1.414556	2.98	0.003	1.448061	6.996044
8	1.894669	1.483771	1.28	0.202	-1.015056	4.804394
9	-1.220028	1.46982	-0.83	0.407	-4.102394	1.662338
10	1.103288	1.9174	0.58	0.565	-2.656797	4.863372
11	2.250065	1.369721	1.64	0.101	-.436003	4.936132
12	3.527286	1.419776	2.48	0.013	.7430572	6.311514
_cons	5.621121	1.355806	4.15	0.000	2.96234	8.279902

I used the factor-variable notation `i.industry` to regress `wage` on indicator variables for each level of `industry`. What makes this regression different from other ways that I might have accomplished the same thing—such as creating the indicators with `tabulate`, `generate()` or by using `xi`—is that Stata did not have to create the 12 indicator variables required to run the regression. The values of the indicators were created “on the fly” based on the value of the `industry` variable.

2. They make full factorial specifications easy

Continuing with the wage regression from the National Longitudinal Survey, I might also be interested in the effect of the women's occupational categories—sales, professional, management, etc. I could add indicators for each level of occupation to the regression in the same way I added indicators for industry. What if, however, I also suspected that the effect of occupation differed across industries—would we see an interaction effect? We have a factor-variable shorthand for entering all 12 indicators for industry, all 13 indicators for occupation, and all indicators for each combination of industry and occupation. We just type

```
. regress wage industry##occupation
```

The first operator, `#`, says that we want the interaction, and the second operator, `##`, says that we want all lower-level interactions, right down to the levels of our constituent categorical variables.

For those versed in experimental design, this will all sound mundane. They refer to this as a full-factorial specification, and the `anova` command does it as a standard specification. What is not mundane is that with factor variables, this notation is available on almost every estimation command, and indeed, almost every command in Stata.

What is also nice is that Stata did not need to create the $12+13+12\times 13=181$ indicator variables. They were created “on the fly” as the `regress` command needed them for its computations.

Factor variables really save space.

3. They work almost everywhere, which is convenient

Yes, I know that `xi` did something similar for factorial specifications, but factor variables are much more deeply understood by Stata. Try predicting on a dataset other than the estimation dataset with `xi`.

```
. webuse margex, clear
. xi: logistic outcome i.treatment*i.group age
. webuse hstandard
. predict prbhat
```

The `predict` command will fail because Stata simply created the indicator variables when you used `xi`; and, despite the fact that `hstandard.dta` contains the same categorical variables, Stata does not know that it should re-create them for the `predict` command.

Try the same thing with factor variables:

```
. webuse margex
. logistic outcome treatment##group age
. webuse hstandard
. predict prbhat
```

It works! It works because Stata knows that, for example, `group==2` is an indicator for the second group, and it knows this across datasets. Of course, you should be sure that `2` means the same thing in both of your datasets.

4. They make short work of tests of structural change

Sometimes we believe that we have a good model specification, but we are not sure that the parameters for one group are applicable for another group. We can apply what economists often call a Chow test, after Chow (1960), but the test is performed in many disciplines. Imagine a model of hospital-stay lengths before and after a change in admission policy, a model of economic growth before and during a war, or a model of fish populations on a reef before and after a hurricane. We might want to test whether all the parameters in these models are jointly the same for each of the two groupings.

Again using `auto.dta`, if we assume that price is determined by mileage and weight (as a proxy for car size), we could fit that relationship by typing

```
. sysuse auto, clear
. regress price mpg weight
```

What if we further believe that the prices of high- and low-quality cars had fundamentally different relationships with `mpg` and `weight`? Perhaps the effect of quality was to reduce the importance of both fuel economy and size. We can group the automobiles based on whether their repair records are better than “fair”.

```
. gen highq = rep78 > 3
```

Now we can estimate both the overall effect of `mpg` and `weight` on price and also estimate the difference of those effects for high-quality cars. These are just the default results from a full-factorial specification that includes interactions with the continuous covariates.

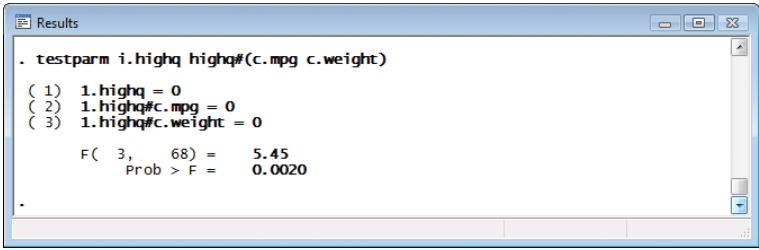
```
. regress price highq##(c.mpg c.weight)
```

We can test whether high quality changes the coefficients on `mpg` and `weight` by typing

```
. testparm highq#(c.mpg c.weight)
```

This is almost what we typed to estimate the regression; only the operator is different. We want to test only the interaction effects, not the overall effects, so we type `#` instead of `##`. The classic Chow test includes the test of whether the constant changes between the two groups, so we can add `i.highq` to our test:

```
. testparm i.highq highq#(c.mpg c.weight)
```



```
Results
. testparm i.highq highq#(c.mpg c.weight)
( 1) 1.highq = 0
( 2) 1.highq#c.mpg = 0
( 3) 1.highq#c.weight = 0

F( 3, 68) = 5.45
Prob > F = 0.0020
```

The test bears out our suspicion that something is structurally different about determining pricing in high-quality cars versus low-quality cars.

Factor variables also let us compute marginal means, discrete marginal effects, etc., using the `margins` command, which is a subject for another article.

If you have not used Stata 11's factor variables, I highly recommend that you explore the possibilities. I suggest you start by reading [\[U\] 11.4.3 Factor variables](#), and then move on to [\[U\] 25 Working with categorical data and factor variables](#). The reference manuals also use factor variables in many examples.

There are many other reasons to like factor variables. Start using them and you will find some of your own.

Reference

Chow, G. C. 1960. Tests of equality between sets of coefficients in two linear regressions. *Econometrica* 28: 591–605.

— Vince Wiggins, Vice President,
Scientific Development

In the spotlight: The generalized method of moments

The generalized method of moments (GMM) is a very flexible estimation framework that has become a workhorse of modern econometric analysis. Unlike maximum likelihood estimation, GMM does not require the user to make strong distributional assumptions, thus providing for more robust estimates. Moreover, GMM is broad-based in that other commonly used estimators like least-squares and maximum likelihood can be viewed as special cases of GMM. GMM is popular in economics not only because of its favorable statistical properties, but also because many theoretical models, such as those involving rational expectations, naturally yield the moment conditions that underlie GMM.

OLS regression is a GMM estimator. In the model $y_i = \mathbf{x}_i'\boldsymbol{\beta} + e_i$, we assume the error term e_i —conditional on \mathbf{x}_i —has an expected value of zero: $E[e_i|\mathbf{x}_i] = 0$. By the law of iterated expectations, this implies that the unconditional moment condition $E[\mathbf{x}_i \times e_i] = E[\mathbf{x}_i(y_i - \mathbf{x}_i'\boldsymbol{\beta})] = 0$. GMM chooses $\boldsymbol{\beta}$ such that the sample analogue of this moment condition is as close to zero as possible. In the case of OLS regression, because the number of parameters equals the number of elements of \mathbf{x}_i , the moment condition will equal zero precisely at the chosen value of $\boldsymbol{\beta}$, and the GMM estimate of $\boldsymbol{\beta}$ will equal the OLS estimate.

In the more general case, we have moment conditions like $E[\mathbf{z}_i(y_i - \mathbf{x}_i'\boldsymbol{\beta})]$, where we have more variables in \mathbf{z}_i than parameters in $\boldsymbol{\beta}$, so the moment condition will not equal zero. In GMM, we choose $\boldsymbol{\beta}$ so that the matrix-weighted Euclidean distance of the moment conditions from zero is minimized. The matrix we use to compute that distance is aptly called the weight matrix. We typically choose the weight matrix based on the properties of the error term (such as heteroskedasticity or autocorrelation). A key feature of GMM is that if we select the appropriate weight matrix, the estimator will be efficient, meaning that it has smaller variance than any other estimator based on those moment conditions.

Stata's `gmm` command, introduced in Stata 11, makes fitting models via GMM a snap. You just enter your moment conditions using a simple syntax, specify your instruments, and select the type of weight matrix that you want. For more complicated models, you can instead write a program that computes the moment conditions, similar to the programs you write when using `ml`.

We have data on the number of doctor visits a person makes, and we want to model that based on the person's gender and income, as well as whether the person has a chronic disease or private insurance. Because the number of doctor visits is a count variable, we want to use Poisson regression. We suspect that unobserved factors that determine the number of times a person visits the doctor also affect the person's income. Thus, income is an endogenous regressor. Standard Poisson estimators like Stata's `poisson` command cannot deal with endogeneity, but that is no problem for `gmm`. The Poisson model leads to the moment conditions $E[\mathbf{z}_i(y_i - \exp(\mathbf{x}_i'\boldsymbol{\beta}))] = 0$. As instruments for income, we will use each person's age and race.

In Stata, we type

```
. webuse docvisits, clear
. gmm (docvis - exp({xb:private chronic female income}+{b0})),
    instruments(private chronic female age black hispanic)
```

Stata/SE 11.1 - http://www.stata-press.com/data/r11/docvisits.dta - [Results]

Review

Command	_rc
1 webuse docvisits, clear	
2 gmm (docvis - exp({xb:private ...	

Step 1
Iteration 0: GMM criterion q(b) = **16.910173**
Iteration 1: GMM criterion q(b) = **.82276104**
Iteration 2: GMM criterion q(b) = **.21832032**
Iteration 3: GMM criterion q(b) = **.12685935**
Iteration 4: GMM criterion q(b) = **.12672369**
Iteration 5: GMM criterion q(b) = **.12672365**

Step 2
Iteration 0: GMM criterion q(b) = **.00234641**
Iteration 1: GMM criterion q(b) = **.00215957**
Iteration 2: GMM criterion q(b) = **.00215911**
Iteration 3: GMM criterion q(b) = **.00215911**

GMM estimation
Number of parameters = 5
Number of moments = 7
Initial weight matrix: **unadjusted**
GMM weight matrix: **robust**
Number of obs = 4412

	Coef.	Robust Std. Err.	z	P> z	[95% Conf. Interval]	
/xb_private	.535335	.1599039	3.35	0.001	.2219291	.8487409
/xb_chronic	1.090126	.0617659	17.65	0.000	.9690668	1.211185
/xb_female	.6636579	.0959884	6.91	0.000	.4755241	.8517918
/xb_income	-.0142855	.0027162	5.26	0.000	-.0089618	-.0196092
/b0	-.5983477	.138433	-4.32	0.000	-.8696713	-.327024

Instruments for equation 1: **private chronic female age black hispanic _cons**

Command
gmm (docvis - exp({xb:private chronic female income}+{b0})), instruments(private chronic female age black hispanic)

C:\Program Files\Stata11

On the `gmm` command, we specified our residual equation $y_i - \exp(\mathbf{x}'_i\beta)$ using a substitutable expression like those used by Stata's `nl` and `nlstur` commands for nonlinear least-squares problems. We then specified the instruments (\mathbf{z}) in the `instruments()` option. By default, `gmm` used the two-step estimator with a weight matrix that allows for heteroskedastic residuals. `gmm` is capable of fitting a wide variety of models, so unlike commands such as `ivregress`, `ivprobit`, or the acclaimed user-written `ivreg2`, we must include all the exogenous variables in the `instruments()` option, not just the variables excluded from the equation we are estimating.

In our example, we used cross-sectional data. However, `gmm` also provides weight matrices that are suitable for use with time-series, panel, and clustered data, as well. `gmm` is like a Swiss army knife: it can be used right away to estimate simpler models, and as you use it you soon discover that it can do so many other tasks, as well.

— Brian Poi, Executive Editor and Senior Economist

Stay informed

For up-to-the-minute news about Stata, be sure to check our web site:

www.stata.com

There you will find announcements regarding updates to Stata, upcoming public training courses, Stata Conferences and Users Group meetings, Stata Press books, and more. You can also subscribe to an RSS feed to have our news headlines delivered straight to your browser.

Prefer to receive an email alert for news that interests you? You can subscribe to our email alert service at

www.stata.com/alerts/

Stata/MP Performance Report updated

Stata/MP is the flavor of Stata that has been programmed to take full advantage of multiprocessor and multicore computers. It is exactly like Stata/SE in all ways, except that it distributes many of Stata's most computationally demanding tasks across all the cores in your computer and thereby runs faster—much faster. If you have a dual-core computer, you can expect Stata/MP to run linear regressions, logistic regressions, and many other estimation commands in about half the time required by Stata/SE. If you have a quad-core computer, those commands will run in about one-fourth the time; and if you have an 8-core computer, they will run in about one-eighth the time. You do not have to change anything to obtain these speed improvements—Stata/MP is just faster on multicore computers.

Across all commands, Stata/MP runs 1.6 times faster on dual-core computers, 2.1 times faster on quad-core computers, and 2.7 times faster on 8-core computers. Those are median speed improvements; half the commands run even faster. Commands that take longer to run are even more parallelized. Across all estimation commands, Stata/MP runs 1.8 times faster on dual-core computers, 2.8 times faster on quad-core computers, and 4.1 times faster on 8-core computers.

Figure 1 summarizes the performance improvements of Stata/MP.

Stata/MP is faster because we have hand-parallelized 250 crucial sections (over 10,000 lines) of Stata's C code. These sections now split their computations across the cores in your computer while carefully optimizing how that splitting is done, based on the size of your dataset, the number of variables, and the structure of each computation. Because of this care, Stata/MP is highly scalable (some commands are almost 100% parallelized and will run over 40 times faster on 64 cores) yet efficient in using computer

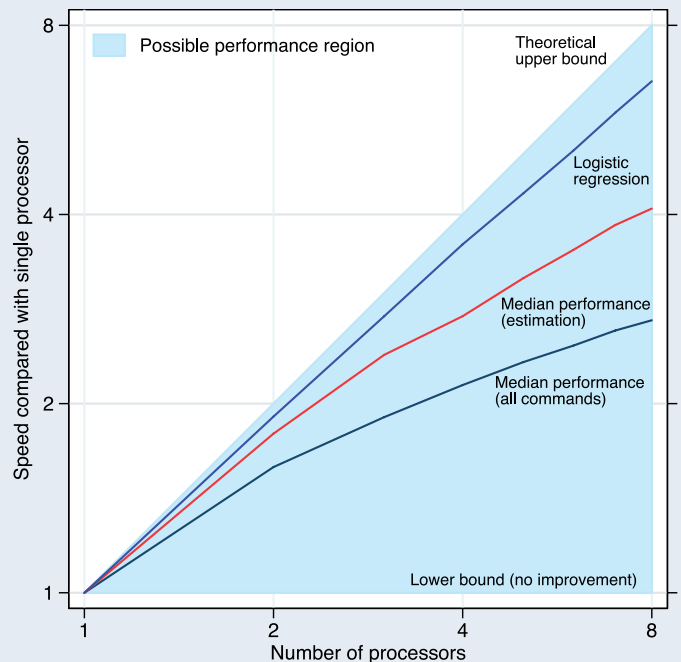


Figure 1

resources for less parallelized commands. You cannot obtain this type of scalability with automated techniques for parallelizing code.

The *Stata/MP Performance Report* provides a complete assessment of Stata/MP's performance, including command-by-command performance statistics. The report has been updated for Stata 11 and now includes coverage for all of Stata's commands, along with a section discussing Mata. Performance graphs are shown for 488 Stata commands.

Read more about Stata/MP at www.stata.com/statamp/, or go straight to the updated performance report at www.stata.com/statamp/report.pdf.

Public training courses

Course	Dates	Location	Cost
Using Stata Effectively: Data Management, Analysis, and Graphics Fundamentals	July 27–28	San Francisco, CA	\$950
	August 25–26	Washington, D.C.	\$950
	October 5–6	New York, NY	\$950
Multilevel/Mixed Models Using Stata	September 16–17	San Francisco, CA	\$1295
	October 7–8	New York, NY	\$1295
Multiple Imputation Using Stata	September 22–23	Washington, D.C.	\$1295

Multiple Imputation Using Stata

Instructor: Yulia Marchenko, Senior Statistician at StataCorp and primary developer of Stata's official multiple-imputation features

This two-day course covers the use of Stata to perform multiple-imputation analysis. Multiple imputation (MI) is a simulation-based technique for handling missing data. The course will provide a brief introduction to multiple imputation and will focus on how to perform MI in Stata using the **mi** command. The three stages of MI (imputation, complete-data analysis, and pooling) will be discussed in detail with accompanying Stata examples. Various imputation techniques will be discussed, with the main focus on multivariate normal imputation. Also, a number of examples demonstrating how to efficiently manage multiply imputed data within Stata will be provided. Linear and logistic regression analysis of multiply imputed data as well as several postestimation features will be presented.

Course topics

- Multiple-imputation overview
 - › MI as a statistical procedure
 - › Stages of MI: Imputation, analysis, and pooling
 - › MI in Stata—the **mi** suite of commands
- Imputation
 - › Imputation techniques
 - › Univariate imputation
 - › Multivariate imputation
 - › Checking the sensibility of imputations
- Data management
 - › Storing multiply imputed data
 - › Importing existing multiply imputed data
 - › Verifying multiply imputed data
 - › Variable management (passive variables)
 - › Merging, appending, and reshaping multiply imputed data
 - › Exporting multiply imputed data to a non-Stata application
- Estimation
 - › Using **mi estimate** to perform the analysis and pooling stages of MI in one easy step
 - › Estimating linear and nonlinear functions of coefficients
 - › Testing linear and nonlinear hypotheses

For more information or to enroll, visit www.stata.com/training/mi.html.

Multilevel/Mixed Models Using Stata

Instructor: Roberto G. Gutierrez, StataCorp's Director of Statistics and primary developer of Stata's official multilevel/mixed models features

This two-day course is an introduction to using Stata to fit multilevel/mixed models. Mixed models contain both fixed effects analogous to the coefficients in standard regression models and random effects not directly estimated, but instead summarized through the unique elements of their variance–covariance matrix. Mixed models may contain more than one level of nested random effects. Hence, these models are also referred to as multilevel or hierarchical models, particularly in the social sciences. Stata's approach to linear mixed models is to assign random effects to independent panels, where a hierarchy of nested panels can be defined for handling nested random effects.

Course topics

- Introduction to linear mixed models
- Random coefficients and hierarchical models
- Postestimation analysis
- Nonlinear models
- Advanced topics

For more information or to enroll, visit

www.stata.com/training/mixed.html.

Using Stata Effectively: Data Management, Analysis, and Graphics Fundamentals

Instructor: Bill Rising, StataCorp's Director of Educational Services

Become intimately familiar with all three components of Stata: data management, analysis, and graphics. This two-day course is aimed at new Stata users and at those who want to optimize their workflow and learn tips for efficient day-to-day usage of Stata. Upon completion of the course, you will be able to use Stata efficiently for basic analyses and graphics. You will be able to do this in a reproducible manner, making collaborative changes and follow-up analyses much simpler. You also will be able to make your datasets self-explanatory to your co-workers and to your future self.

Whether you currently own Stata 11 or you are considering an upgrade or

a new purchase, this course will unquestionably make you more proficient with Stata's wide-ranging capabilities.

Course topics

- Stata basics
- Data management
- Workflow
- Analysis
- Graphics

For more information or to enroll, visit

www.stata.com/training/eff_stata.html.

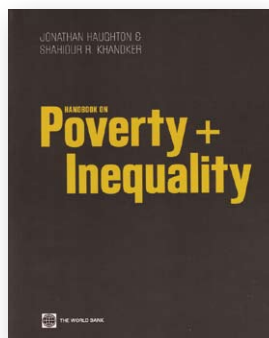
Enrollment in public training courses is limited. Computers with Stata 11 installed are provided at all public training sessions. A continental breakfast, lunch, and an afternoon snack will also be provided. All training courses run from 8:30 AM to 4:30 PM each day. Participants are encouraged to bring a USB flash drive to all public training sessions; this is the safest and simplest way to save your work from the session.

For a complete schedule of upcoming training courses, visit

www.stata.com/training/public.html.

New from the Stata Bookstore

Handbook on Poverty + Inequality



Authors: Jonathan Haughton and Shahidur R. Khandker

Publisher: World Bank

Copyright: 2009

Pages: 444; paperback

ISBN-10: 0-8213-7613-6

ISBN-13: 978-0-8213-7613-3

Price: \$36.00

In *Handbook on Poverty + Inequality*, Jonathan Haughton and Shahidur Khandker provide introductions to problems of defining, measuring, and analyzing poverty and inequality. While this book is more focused on the subjects of poverty and inequality than on data-analytic methods, several chapters describe important aspects of how to interpret results from different types of data analysis. The authors also provide quick introductions to basic data analysis, regression modeling, and the use of complex survey data, in addition to providing an appendix that succinctly introduces Stata in regard to the methods discussed.

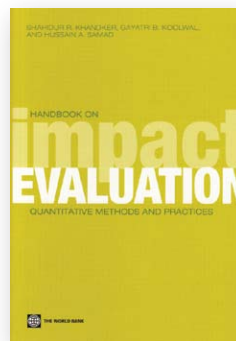
This book will be useful for students and researchers who need an introduction to the literature on poverty and inequality and who need a quick primer on data analysis. Researchers and students who are analyzing data will want an additional book on statistics or econometrics, such as *Data Analysis Using Stata, Second Edition* by Ulrich Kohler and Frauke Kreuter or

Introductory Econometrics: A Modern Approach, Fourth Edition by Jeffrey M. Wooldridge.

You can find the table of contents and online ordering information at

www.stata.com/bookstore/hopi.html.

Handbook on Impact Evaluation: Quantitative Methods and Practices



Authors: Shahidur R. Khandker, Gayatri B. Koolwal, and Hussain A. Samad

Publisher: World Bank

Copyright: 2010

Pages: 280; paperback

ISBN-10: 0-8213-8028-1

ISBN-13: 978-0-8213-8028-4

Price: \$37.75

In *Handbook on Impact Evaluation: Quantitative Methods and Practices*, Shahidur Khandker, Gayatri Koolwal, and Hussain Samad provide an excellent, relatively nontechnical introduction to the estimation and interpretation of treatment effects. While this book is aimed at practitioners in development economics, it will also be useful to graduate students and researchers who need background on and intuition for the modern, more technical literature. The authors also provide a succinct introduction to Stata for the purpose of estimating treatment effects.

To illustrate the technical points, Khandker, Koolwal, and Samad use many case studies and intuitive examples drawn both from their own work and from development economics. The authors' ability to make technical information easy to understand is a definite strength of this book.

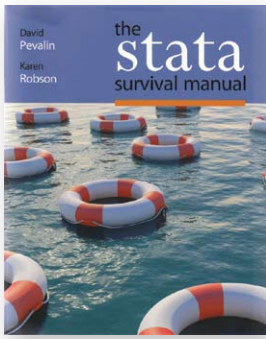
The breadth of coverage is impressive: after distinguishing quantitative impact evaluation from other forms of program evaluation, the authors provide introductions to the counterfactual model, the random-assignment model, propensity-score matching for selection on observables, and several methods for selection on unobservables.

The authors do not address many important technical details, but each chapter contains a good bibliography for readers who need more information. The authors provide a high-level introduction that will serve as an excellent outline or jump-off point per the reader's needs.

You can find the table of contents and online ordering information at

www.stata.com/bookstore/hoie.html.

The Stata Survival Manual



Authors: David Pevalin and Karen Robson
 Publisher: McGraw-Hill
 Copyright: 2009
 Pages: 373; paperback
 ISBN-10: 0-335-22388-5
 ISBN-13: 978-0-335-22388-6
 Price: \$54.75

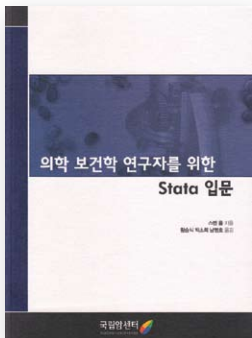
The Stata Survival Manual, by David Pevalin and Karen Robson, is a nicely written introduction to the practical use of Stata 10. The style is friendly and flows well, and the authors do not assume prior knowledge of Stata or statistical sophistication from the reader. Both Stata and statistical usage are explained throughout the book.

The authors step through the basics of using Stata, starting with basic usage of Stata and working through common data-management techniques for analysis, table and graph creation, and presentation of results. Special focus is given to working with categorical variables and building scales from instruments. The analysis sections detail how to fit interactions and explain them to nonstatistical audiences using graphs. Each chapter begins with a presentation of new tools in Stata and simple examples of their use. The tools are then applied via a “Demonstration Exercise” to an example that runs throughout the book. Thus the reader can learn new tools in a simple setting and see their use in an analysis on a real-life dataset from start to finish.

At several points in the book, especially in the chapters focusing on data management, the authors point out differences between Stata and IBM SPSS Statistics for those making the transition from IBM SPSS Statistics to Stata. While the authors focus on using do-files for reproducibility, they also show how to use the menus and dialog boxes for those accustomed to working in this fashion.

You can find the table of contents and online ordering information at www.stata.com/bookstore/ssm.html.

An Introduction to Stata for Health Researchers, 2nd Edition (Korean)



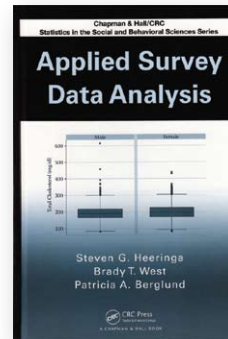
Author: Svend Juul
 Publisher: JasonTG Co., Seoul
 Copyright: 2009
 Pages: 450; paperback
 ISBN-10: 89-92864-03-5
 ISBN-13: 978-89-92864-03-9
 Price: \$52.00

Translated by JasonTG, Stata's distributor in South Korea, *An Introduction to Stata for Health Researchers, Second Edition* provides Korean speakers

an overview of Stata's data management, graphics, and statistics capabilities. Distinguished in its careful attention to detail, the book not only teaches how to use Stata, but also teaches the skills needed to create the reproducible analyses that are so necessary in the field. The book is based on the assumption that the reader has some basic knowledge of statistics but no knowledge of Stata. Juul builds the reader's abilities as a builder would build a house, laying a firm foundation in Stata, framing a general structure in which good work can be accomplished, and finally filling in details that are particular to various types of statistical analysis.

You can find the table of contents and online ordering information at www.stata.com/bookstore/ishr2_korean.html.

Applied Survey Data Analysis



Authors: Steven G. Heeringa, Brady T. West, and Patricia A. Berglund
 Publisher: CRC Press/Taylor & Francis
 Copyright: 2010
 Pages: 462; hardback
 ISBN-10: 1-4200-8066-0
 ISBN-13: 978-1-4200-8066-7
 Price: \$64.00

Applied Survey Data Analysis is an intermediate-level, example-driven treatment of current methods for complex survey data. It will appeal to researchers of all disciplines who work with survey data and have basic knowledge of applied statistical methodology for standard (nonsurvey) data.

The authors begin with some history and by discussing some widely used survey datasets, such as the National Health and Nutrition Examination Survey (NHANES). They then follow with the basic concepts of survey data: sampling plans, weights, clustering, prestratification and poststratification, design effects, and multistage samples. Discussion then turns to the types of variance estimators: Taylor linearization, jackknife, bootstrap, and balanced and repeated replication.

The middle sections of the text provide in-depth coverage of the types of analyses that can be performed with survey data, including means and proportions, correlations, tables, linear regression, regression with limited dependent variables (including logit and Poisson), and survival analysis (including Cox regression). Two final chapters are devoted to advanced topics, such as multiple imputation, Bayesian analysis, and multilevel models. The appendix provides overviews of popular statistical software, including Stata.

You can find the table of contents and online ordering information at www.stata.com/bookstore/asda.html.

Erratum: While editing the previous issue of the Stata News, on page 2 we inadvertently referred to Donald Rubin, a leading researcher in the area of multiple imputation, as Daniel Rubin. We regret the error.

Users Group meetings: Save the date

German Stata Users Group meeting

Sophia Rabe-Hesketh will be the keynote speaker.

Date: June 25, 2010
 Venue: Berlin Graduate School of Social Sciences
 Luisenstraße 56, House 1
 10117 Berlin-Mitte, Germany
 Cost: €35 professionals; €15 students
 Details: www.stata.com/meeting/germany10/

UK Stata Users Group meeting

Dates: September 9–10, 2010
 Venue: London School of Hygiene and Tropical Medicine
 Keppel Street, London WC1E 7HT, UK
 Cost: Both days: £94.00 professionals; £64.63 students
 Single day: £64.63 professionals; £47.00 students
 Details: www.stata.com/meeting/uk10/

Spanish Stata Users Group meeting

Date: September 14, 2010
 Venue: Universidad Carlos III de Madrid
 C/ Madrid 126, 28903 Getafe, Madrid, Spain
 Cost: €30 professionals; €30 students
 Submissions: June 14, 2010
 Details: www.stata.com/meeting/spain10/

Portuguese Stata Users Group meeting

Date: September 17, 2010
 Venue: University of Minho
 Gualtar University Campus, 4710-057 Braga, Portugal
 Cost: €45 professionals; €20 students
 Submissions: June 14, 2010
 Details: www.stata.com/meeting/portugal10/

Italian Stata Users Group meeting

Dates: November 11–12, 2010
 Venue: Grand Hotel Baglioni
 Via Indipendenza, 8
 40121 Bologna, Italy
 Cost: €90, day 1 only; €375, day 1 + a training course
 Submissions: August 30, 2010
 Details: www.stata.com/meeting/italy10/

Upcoming NetCourses®

Enroll by visiting www.stata.com/netcourse/.

NC101: Introduction to Stata

An introduction to using Stata interactively

Dates: July 9–August 20, 2010
 Enrollment deadline: July 8, 2010
 Price: \$95
 Details: www.stata.com/netcourse/nc101.html

NC151: Introduction to Stata Programming

An introduction to Stata programming dealing with what most statistical software users mean by programming, namely, the careful performance of reproducible analyses

Dates: July 9–August 20, 2010
 Enrollment deadline: July 8, 2010
 Price: \$125
 Details: www.stata.com/netcourse/nc151.html

NC152: Advanced Stata Programming

This course teaches you how to create and debug new commands that are indistinguishable from those of official Stata. It is assumed that you know why and when to program and, to some extent, how. You will learn how to parse both standard and nonstandard Stata syntax by using the intuitive **syntax** command, how to manage and process saved results, how to process by-groups, and more.

Dates: October 8–November 26, 2010
 Enrollment deadline: October 7, 2010
 Price: \$150
 Details: www.stata.com/netcourse/nc152.html

NC461: Introduction to Univariate Time Series with Stata

This course introduces univariate time-series analysis, emphasizing the practical aspects most needed by practitioners and applied researchers. The course is written to appeal to a broad array of users, including economists, forecasters, financial analysts, managers, and anyone who encounters time-series data.

Dates: October 8–November 26, 2010
 Enrollment deadline: October 7, 2010
 Price: \$295
 Details: www.stata.com/netcourse/nc461.html

Stata Conference Boston 2010

Dates: July 15–16, 2010

Venue: Omni Parker House, Boston
60 School Street
Boston, MA 02108



Cost: Single day \$125, students \$50
Both days \$195, students \$75

Register: www.stata.com/meeting/boston10/

The Stata Conference Boston 2010 will be held on July 15 and 16 at the Omni Parker House hotel, located in downtown Boston near the Boston Common and the Park Street T.

Program

Thursday, July 15

Regression for nonnegative skewed dependent variables

Austin Nichols, Urban Institute

Margins and the Tao of interaction

Phil Ender, UCLA Statistical Consulting Group

To the vector belong the spoils: Circular statistics in Stata

Nicholas J. Cox, Durham University

System for formatting tables

John Gallup, Portland State University

Hunting for genes with longitudinal phenotype data using Stata

Chuck Huber, Texas A&M Health Science Center School of Rural Public Health

Bayesian bivariate diagnostic meta-analysis via R-INLA

Ben Adarkwa Dwamena, University of Michigan and VA Ann Arbor Health Systems

Storing, analyzing, and presenting Stata output

Julian Reif, University of Chicago

An efficient data envelopment analysis with a large dataset in Stata

Choonjoo Lee, Korea National Defense University

Competing-risks regression in Stata 11

Roberto G. Gutierrez, StataCorp

Structural equation models with latent variables

Stas Kolenikov

Friday, July 16

Multiple imputation using Stata's mi command

Yulia Marchenko, StataCorp

CEM: Coarsened exact matching in Stata

Matthew Blackwell, Harvard University

Evaluating one-way and two-way cluster-robust covariance matrix estimates

Christopher F. Baum, Boston College

Bootstrap LM test for the Box-Cox tobit model

David Vincent, Hewlett Packard

Teaching a statistical program in emergency medicine research rotations: Command-driven or click-driven?

Muhammad Waseem, Lincoln Medical and Mental Health Center

Report to users

William Gould, StataCorp

Wishes and grumbles: User feedback and Q&A

Scientific organizers

Elizabeth Allred, *Harvard School of Public Health*

Christopher F. Baum (chair), *Boston College*

Amresh Hanchate, *Boston University*

Marcello Pagano, *Harvard School of Public Health*

Logistics organizers

Chris Farrar, *StataCorp*

Gretchen Farrar, *StataCorp*

Sarah Marrs, *StataCorp*

Contact us

StataCorp
4905 Lakeway Dr.
College Station, TX 77845
USA

Phone 979-696-4600
Fax 979-696-4601
Email service@stata.com
Web www.stata.com

Please include your Stata serial number with all correspondence.



Copyright 2010 by StataCorp LP.

To locate a Stata international distributor near you, visit www.stata.com/worldwide/.