Stata users’ meeting, London, September 2014

Thursday, September 11

9:25-9:30  Opening remarks
9:30-10:00  Roger Newson on rank and spline parameters
10:00-10:30  Brendan Halpin on sequence analysis

10:30-11:00  Break

11:00-11:30  Ben Jann on coefplot
11:30-12:00  Robert Grant on interactive online graphics
12:00-12:45  Nick Cox on small multiples in graphics

12:45-1:45  Lunch

1:45-2:15  Michael Crowther on simulating survival data
2:15-3:15  Yulia Marchenko on transformation survival models

3:15-3:45  Break

3:45-4:45  William Gould on floating point numbers

6:30  Optional dinner

Friday, September 12

9:30-10:00  Vincenzo Verardi on generalized boxplots
10:00-10:30  Thomas Roca on dynamic visualization
10:30-11:00  Break

11:00-11:30  Daniel Bratton on multi-arm multi-stage clinical trial designs
11:30-12:00  Aurelio Tobías et al. on time-stratified case-crossover studies
12:00-12:30  Ian White on treatment changes in randomised trials

12:30-1:30  Lunch

1:30-2:30  John Thompson on Bayesian analysis with Stata
2:30-3:00  Giovanni Cerulli on treatment effects and neighbourhood interactions

3:00-3:30  Break

3:30-4:00  Kit Baum and Mark Schaffer on asymptotic covariance matrix estimation
4:00-4:30  Philippe Van Kerm on space-filling location selection

4:30  Wishes and grumbles
Easy-to-use packages for estimating rank and spline parameters

So-called ‘non-parametric’ methods are in fact based on estimating and testing parameters, usually either rank parameters or spline parameters. Two comprehensive packages for estimating these are somersd (for rank parameters) and bspline (for spline parameters). Both of these estimate a wide range of parameters, but both are frequently found to be difficult to use by casual users. This presentation introduces rcentile, an easy-to-use front-end for somersd, and polyspline, an easy-to-use front-end for bspline. rcentile estimates percentiles with confidence limits, optionally allowing for clustered sampling and/or sampling-probability weights. The confidence intervals are saved in a Stata matrix, with one row per percentile, which the user may save to a resultsset using the xsvmat package. polyspline inputs an X-variable and a user-defined list of reference points, and outputs a basis of variables for a polynomial, or for another unrestricted spline. This basis can be included in the covariate list for an estimation command, and the corresponding parameters will be values of the polynomial or spline at the reference points, or differences between these values. In default, the spline will simply be a polynomial, with a degree one less than the number of reference points. However, if the user specifies a lower degree, then the spline will have knots interpolated sensibly between the reference points.
Sequence analysis (SA) is a very different way of looking at categorical longitudinal data, such as life-course or labour-market histories (or any ordered categorical data, for that matter). Instead of focusing on transition rates (e.g., via hazard rate, Markov or panel models) it takes individual time-series and compares them as wholes. It has significant advantages at a descriptive and exploratory level, and may help detect patterns which conventional methods will overlook. As availability of longitudinal data increases, this becomes a significant advantage.

Sequence analysis hinges on defining measures of similarity between sequences, typically in order to generate data-driven classifications, for example by cluster analysis. Most SA uses the Optimal Matching distance, but other measures are in use. There is some controversy about the applicability of SA algorithms to social science data, and about their parameterisation. Comparison of different methods and parameterisations helps clarify the issues.

For a long time TDA was the only package social scientists had access to for sequence analysis, but in more recent years both Stata and R have had relevant functionality, in Stata’s case provided by the sq and sadi packages.

In this talk I will discuss the current state of the sadi package. sadi differs from sq in being plugin-based, and therefore is significantly faster: many of the distance measures are compute-intensive, and typically $N(N-1)/2$ comparisons will be made for $N$ observations. It also provides additional distance measures, including Dynamic Hamming, Time-Warp Edit Distance and a version of Elzinga’s Number of Matching Subsequences measure. It includes tools for inspecting and graphing sequence data, and for comparing distance measures and the resulting cluster analyses.

I will also briefly discuss the advantages and disadvantages of using plugins rather than Mata, and make some remarks about cross-compiling plugins under Linux.
A new command for plotting regression coefficients and other estimates

Graphical display of regression results has become increasingly popular in presentations and the scientific literature, as graphs are much easier to read than tables in many cases. In Stata such plots can be produced by the \texttt{marginsplot} command. However, while \texttt{marginsplot} is very versatile and flexible, it has two major limitations: it can only process results left behind by \texttt{margins} and it can only handle one set of results at the time. In this presentation I will introduce a new command called \texttt{coefplot} that overcomes these limitations. It plots results from any estimation command and combines results from several models into a single graph. The default behavior of \texttt{coefplot} is to plot markers for coefficients and horizontal spikes for confidence intervals. However, \texttt{coefplot} can also produce various other types of graphs. The capabilities of \texttt{coefplot} are illustrated using a series of examples.

Making interactive online visualisations with \texttt{stata2d3} and \texttt{stata2leaflet}

The last three years have seen explosive growth in the variety and sophistication of interactive online graphics. These are mostly implemented in the web language JavaScript, with the D3 (Data Driven Documents) library being the most popular and flexible at present. Leaflet is a mapping library also being widely used. R users have some packages that translate their data and specifications into interactive graphics and maps, which write a text file containing the HTML and JavaScript instructions that make up a web page containing the desired visualisation. This translation into a web page is easily achieved in Stata, and I will present the \texttt{stata2leaflet} command which produces zoomable, clickable online maps.

Contemporary interactive graphs benefit from allowing the viewer to filter and select data of interest, which is a second layer of specification implemented in the \texttt{stata2d3} commands. \texttt{stata2d3} capitalises on the consistency of Stata graph syntax by parsing and translating a standard Stata graph command into a web page. Users can choose to include explanatory comments against each line in the source code, invisible to viewers but helping them to learn HTML and JavaScript and make further refinements.
Small multiples, or the science and art of combining graphs

Good graphics often exploit one simple graphical design that is repeated for different parts of the data, which Edward R. Tufte dubbed the use of small multiples. In Stata, small multiples are supported for different subsets of the data with by() or over() options of many graph commands; users can easily emulate this in their own programs by writing wrapper programs that call twoway or graph bar and its siblings. Otherwise, specific machinery offers repetition of a design for different variables, such as the (arguably much under-used) graph matrix command. Users can always put together their own composite graphs by saving individual graphs and then combining them.

I reviewed small multiples at the 2013 London meeting and this presentation offers further modest automation of the same design repeated for different data. Four recent general programs allow small multiples in different ways. crossplot offers a simple student-friendly graph matrix for each y and each x variable specified, which is more general than a scatter plot matrix. combineplot is a command for combining univariate or bivariate plots for different variables. designplot is a command for showing univariate summaries that generalizes and rewrites grmeanby in graph dot style. subsetplot automates display of different subsets of data with the complement as backdrop. Recent additions to some earlier programs are also reviewed.

Simulating simple and complex survival data

Simulation studies are conducted to assess novel and currently used methods in practice, to better assess and understand the frameworks under question. In survival analysis, we are interested in simulating both an event and censoring distribution to better reflect clinical data. In this talk, I will describe how to simulate survival times from simple parametric distributions, and then move to a more general framework, illustrating how to simulate from a general user-defined hazard function. This can incorporate any combination of a complex baseline hazard function with turning points, time-dependent effects, random effects, and nonlinear covariate effects. This is achieved through a two-stage algorithm incorporating numerical integration nested within root-finding techniques. The methods will be illustrated using the publicly available survsim package.
Transformation survival models

The Cox proportional hazards model is one of the most popular methods to analyze survival or failure-time data. The key assumption underlying the Cox model is that of proportional hazards. This assumption may often be violated in practice. Transformation survival models extend the Cox regression methodology to allow for non-proportional hazards. They represent the class of semiparametric linear transformation models, which relates an unknown transformation of the survival time linearly to covariates. In my presentation, I will describe these models and demonstrate how to fit them in Stata.

Floating point numbers: A visit to the looking glass world

In lieu of his usual Report to users, Bill Gould will talk on floating-point numbers. Researchers do not adequately appreciate that floating-point numbers are a simulation of real numbers and, as with all simulations, some features are preserved while others are not. When writing code, or even do-files, treating the computer’s floating numbers as if they were real numbers can lead to substantive problems and to numerical inaccuracy. In this, the relationship between computers and real numbers is not entirely unlike the relationship between tea and Douglas Adams’s Nutri-Matic drink dispenser. The Nutri-Matic produces a concoction that is “almost, but not quite, entirely unlike tea”. Gould shows what the universe would be like if it were implemented in floating-point rather than in real numbers. The floating-point universe turns out to be nothing like the real universe and probably could not be made to function. Without jargon and without resort to binary, Gould shows how floating-point numbers are implemented on an imaginary base-10 computer and quantifies the kinds of errors that can arise. In this, float-point subtraction stands out as really being almost, but not quite, entirely unlike subtraction. Gould shows how to work around such problems. The point of the talk is to build your intuition about the floating-point world so that you as a researcher can predict when calculations might go awry, know how to think about the problem, and determine how to fix it.
A generalized boxplot for skewed and heavy-tailed distributions implemented in Stata

The boxplot is probably the most commonly used tool to represent the distribution of the data and identify atypical observations in a univariate dataset. The problem with the standard boxplot is that, as soon as asymmetry or tail heaviness appears, the percentage of values identified as atypical becomes excessive. To cope with this issue, Hubert and Vandervieren (2008) proposed an adjusted boxplot for skewed data. Their idea is to move the whiskers of the boxplot according to the degree of asymmetry of the data. The rule to set the whiskers of the adjusted boxplot was found by running a large number of simulations using a wide range of (moderately) skewed distributions. The idea was to find a rule that guaranteed that 0.7% of the observations would lie outside the interval delimited by the whiskers. Even if their rule works satisfactorily for most commonly used distributions, it suffers from some limitations: (i) the adjusted boxplot is not appropriate for severely skewed distributions and/or for distributions with heavy tails; (ii) it is specifically related to a theoretical rejection rate of 0.7%; (iii) it is extremely sensitive to the estimated value of the asymmetry parameter; (iv) it requires a substantial computational complexity, $O(n \log n)$.

To tackle these drawbacks, we propose a much simpler method to find the whiskers of the boxplot in case of (eventually) skewed and heavy-tailed data. We apply a simple rank-preserving transformation on the original data so that the transformed data can be adjusted by a so-called Tukey $g$-and-$h$ distribution. Using the quantiles of this distribution, it is then relatively easy to recover whiskers of the boxplot related to the original data. The computational complexity of the proposed method is $O(n)$, the same as the standard boxplot.

Thomas Roca  
Agence Française de Développement, AFD  
rocat@afd.fr

**Data visualization with Stata:**

**How to adapt APIs and take benefit of HTML5 Canvas to create dynamic charts and maps with Stata**

Data visualization is a burgeoning field at the crossroads of design, computer science and statistics. Using HTML and applying data visualization technique allows producing elegant and insightful representations with Stata. Nevertheless, creating original ‘dataviz’ from an empty page is rough: it requires specific programming knowledge. Thankfully, many pieces of code have been developed under GNU/GPL or Apache licence, accessible for free (Google APIs are the most famous ones).

However, these pieces of code need to be adapted; many options are available to modify displays (texts, legends, labels, colours, size, type of representation, etc.) Besides, HTML5 now embeds a powerful graphic and drawing engine: Canvas, which can be mobilized to represent any data, starting from scratch. As underlying datasets need to be reshaped and organized, a fair amount of programming becomes necessary; thus, Stata users possess a great asset to fill in and format HTML and Java content.

Specific programs will be discussed:

**Googlegeomap** package creates heat maps using Googles API.

**Donutchart** package builds two layer pie charts supported by Highcharts API.

The project ‘How good are you converting your income into...’ is a simple data visualization created with Stata using HTML5 Canvas, which easily allows flagging outliers. This dataviz will be presented as an introduction to the use of Canvas with Stata.

**Dynamic Scatter Plot**, fuelled with HTML5 Canvas, can represent 3 variables, and displays additional information, scrolling over with the mouse. Furthermore, two Dynamic Scatter Plots can be superimposed to facilitate comparisons.

To help grasp the possibility offered by Stata in web programming, the **Country Dashboards** project will be presented. It shows how to create a comprehensive web portal, embedding hundreds of dataviz and web pages built with Stata.

Additional materials:
Using Mata to accelerate the familywise error rate calculation for multi-arm multi-stage clinical trial designs

More efficient trial designs are needed to combat the increasing cost of drug development. A class of trial designs which can help to accelerate this process, known as multi-arm multi-stage (MAMS) designs, have been proposed and used to much effect in oncology. To facilitate the design of these trials evaluating time to event outcomes, the nstage program was introduced for Stata in 2009. This program provides an estimate of the sample size requirement together with estimates of the pairwise operating characteristics and stage durations.

An important quantity in a multi-arm trial is its familywise error rate (FWER), that is, the probability of incorrectly recommending at least one ineffective treatment at the end of a trial. Strong control of the FWER is often a requirement in any confirmatory multi-arm study. A subroutine was therefore recently added to nstage to calculate the FWER of a MAMS design by simulating trial-level data, specifically the z-test statistics for the treatment effect in each arm at each stage of the study.

The calculation is relatively quick, taking less than ten seconds for designs with up to six arms and five stages. However, it can be made more efficient by performing the calculation in Mata. This involves the use of three-dimensional matrices which, despite being unable to be used directly in Mata, can be generated through the use of pointers. The speed of the Mata calculation over that using only Stata increases with the number of arms and stages, with almost a 50% reduction in computing time for a 6-arm 5-stage design. Although just a few seconds are saved for a single design, the cumulative savings in time are considerable when searching over multiple designs to find the one most suitable or most efficient for the MAMS trial in question.

In this talk, the calculations in Stata and Mata are described and their speeds for designs with various numbers of arms and stages are compared. The use of pointers in the FWER calculation in Mata is described and we discuss their advantages and potential use in other areas.
Analysis of time-stratified case-crossover studies in environmental epidemiology using Stata

The time-stratified case-crossover is a widely used design in environmental epidemiology to analyze the short-term effects of environmental risk factors, such as air pollution or temperature, on human health. It compares the exposure level in the day when the health event occurs (case day) with the levels in control days chosen with alternative selection methods. Standard analytical approaches to case-crossover analysis, based on conditional logistic regression (command `clogit` in Stata), require a considerable amount of data management. Here we introduce the `gencco` command to reshape datasets from time-series to time-stratified case-crossover. Then, we will discuss alternative statistical models to perform case-crossover analysis for aggregated data using Poisson and overdispersed Poisson regression (`poisson` and `glm`), and conditional Poisson regression (`xtpoisson`). Furthermore, we also introduce an updated command for conditional Poisson to allow for overdispersion (`xtpoisson_addOD`). Examples will be given using air pollution and mortality datasets, although these methods can be applicable generally in other contexts.
Handling treatment changes in randomised trials

Treatment changes in randomised trials are common: for example, in a trial evaluating psychotherapy, individuals allocated to psychotherapy may attend only partially or not at all; or in a trial evaluating a drug treatment, individuals allocated to no drug treatment may nevertheless receive the treatment. The issue is especially important in drug trials for late stage cancer where control group members typically receive the active treatment on disease progression. This talk focuses on time-to-event outcomes. In some cases, it is important to estimate the effect of the treatment if some or all of these treatment changes had not occurred: for example, for a health economic model exploring whether a drug should be available on the NHS, we would need to compare survival of a treated group with survival of a completely untreated group.

12 years ago I published strbee, which implements in Stata the ‘rank-preserving structural failure time model’ (RPSFTM) of Robins and Tsiatis (1991). This is a model which infers a comparison of the randomised groups if they had had different treatment experiences; estimation is based only on comparisons of randomised groups.

Over the intervening years, the RPSFTM has been increasingly (though not widely) used, and various problems have been identified. First, it assumes treatment benefit is the same whenever treatment is received, and a sensitivity analysis to address possible departures from this assumption is needed. Second, the method’s power is low and declines as follow-up extends, because later times which contribute little information are given the same weight as earlier times. Third, a wider range of estimation procedures is required.

I will review the RPSFTM method and its alternatives in a Stata context, and I will describe an update of strbee which addresses the above issues.

Bayesian analysis with Stata: application to neonatal mortality in the UK

The Bayesian approach to statistical analysis has many theoretical advantages but in the past its use has been limited by a lack of suitable statistical software. In a book published this year by Stata Press called *Bayesian analysis with Stata*, I have tried to show that Stata can be used for Bayesian as well as frequentist statistical analysis.

In this talk I will present a Bayesian analysis of neonatal mortality rates in England and Wales and show how it can be implemented in Stata or Mata or by calling WinBUGS from within Stata.

Over the last few decades, neonatal mortality has been falling steadily throughout the world but the UK lags some way behind many other developed countries. Each year, data on neonatal mortality categorized by birth weight and maternal age are published. We will use a Bayesian analysis of these data to see if the declining rate over time has been similar in all of the categories.

ntreatreg: a Stata module for estimation of treatment effects in the presence of neighbourhood interactions

This paper presents a parametric counter-factual model identifying Average Treatment Effects (ATEs) by Conditional Mean Independence when externality (or neighbourhood) effects are incorporated within the traditional Rubin potential outcome model. As such, it tries to generalize the usual control-function regression, widely used in program evaluation and epidemiology, when SUTVA (i.e. Stable Unit Treatment Value Assumption) is relaxed. As a by-product, the paper presents also *ntreatreg*, a user-written Stata routine for estimating ATEs when social interaction may be present. Finally, an instructional application of the model and of its Stata implementation (using *ntreatreg*) through two examples (the first on the effect of housing location on crime; the second on the effect of education on fertility) is shown and results compared with a no-interaction setting.
Extending Stata’s capabilities for asymptotic covariance matrix estimation

The avar routine (Baum and Schaffer, SSC) constructs the ‘filling’ for a number of flavors of ‘sandwich’ covariance matrix estimators, including HAC, one- and two-way clustering, common cross-panel autocorrelated errors, etc. We show how avar may be used as a building block to construct VCEs that go beyond the Eicker-Huber-White and one-way cluster-robust VCEs provided by official Stata’s robust command. We also show how avar may be used to provide multiple-equation VCE estimates in a wider variety of circumstances than official Stata’s suest command.

Space-filling location selection

This presentation describes a Stata implementation of a space-filling location selection algorithm. The objective is to select a subset from a discrete list of locations so that the spatial coverage of the locations by the selected subset is optimized according to a geometric criterion. Such an algorithm designed for geographical site selection is useful more generally to determine a grid of points that ‘covers’ a data matrix as needed in various non-parametric estimation procedures. Various examples illustrate usage of the user-written command spacefill.

Wishes and grumbles

William Gould, President of StataCorp and more importantly in this context chief developer of Stata, and colleagues will be happy to receive wishes for developments in Stata and almost as happy to receive grumbles about the software.