

STATA, UN PAQUETE SENCILLO PERO POTENTE

RICARDO MESTRE

Servicio de Estudios, Banco de España

Stata¹ es un paquete de tratamiento de datos que permite analizarlos y describirlos mediante el uso de métodos estadísticos y gráficos. No es un recién llegado al mundo de los paquetes de análisis de datos, sino que tiene tras de sí una más que honrosa historia en instituciones académicas y fuera de ellas. La popularidad del paquete se explica por la potencia aliada a la sencillez que ofrece para el análisis de datos, muy en particular para datos de sección cruzada. Aquellas personas que deban enfrentarse a datos en circunstancias cambiantes, o bien a estudios de naturaleza muy distinta entre sí, y que necesitan por lo tanto un paquete ágil y versátil, deben apostar por Stata. El paquete demuestra, además, tener una gran capacidad de adaptación, muy en particular gracias a su capacidad para ejecutar procedimientos o programas definidos por el usuario. Esta característica permite que se esté produciendo un flujo continuo de añadidos al programa desde fuera de él, a través de procedimientos cedidos por terceros. El presente texto pretende servir de nota introductoria al paquete; como medio de resaltar sus principales características, se hará hincapié en el fácil manejo del mismo y su fantástica extensibilidad.

Stata permite unas libertades a la hora de definir, crear, manejar y describir datos que no es fácil encontrar en este tipo de paquetes. Al mismo tiempo, su sencillez de manejo le hace apto para ser usado por personas de todo tipo y nivel, sin perder por ello posibilidades en cuanto a potencia de cálculo estadístico, gracias a los procedimientos externos mencionados. Por supuesto, no todo esto se logra sin problemas, y Stata adolece de algunos defectos que, en el presente texto, no serán ocultados. Obsérvese, de todas formas, que el autor realiza buena parte de su trabajo en paquetes ajenos a Stata (por ser más específicos), pero que siente siempre el mismo placer cuando los avatares del trabajo le 'obligan' a retomar el uso del paquete.

La gran diversidad de tareas abordables con Stata hace que sea complejo ofrecer una visión panorámica del producto, por lo que se ha optado por describir las capacidades y la filosofía de uso del paquete. Para motivar la lectura, se procederá a describir su potencialidad relacionándola con los diversos pasos que

(1) Stata es una marca registrada de Stata Corporation, 702 University Drive East, College Station, TX 77840, USA.

debe seguir todo analista a la hora de formalizar su análisis, y que básicamente son: la reunión de los datos, su depuración, la formalización del modelo postulado y la presentación de resultados. Al contrario que la inmensa mayoría de paquetes de cálculo numérico, Stata tiene algo que aportar en cada uno de estos pasos. Las secciones que siguen, en consecuencia, servirán para presentar una rápida descripción de las sentencias que facilitan en Stata, respectivamente, la reunión de datos, su descripción y depuración (estadística o no), el uso de las técnicas de estimación más usuales y la presentación de resultados finales. Finalizamos con un pequeño resumen y las recomendaciones que se derivan del escrito para aquellos que se planteen la compra de un paquete estadístico.

1. INTRODUCCIÓN GENERAL AL PAQUETE

En general, Stata es un paquete pensado para gestionar datos de sección cruzada referidos a una serie de individuos pero sin dimensión temporal. El caso opuesto son los datos de series temporales, que describen la información de un sólo individuo bajo forma de variables que evolucionan en el tiempo. Ha existido tradicionalmente una gran dicotomía en el tratamiento dado a los dos conjuntos de datos, de tal forma que las herramientas informáticas utilizadas con unos no suelen servir, o son de uso dificultoso, para los otros. Stata debe ser encuadrado como una herramienta decididamente orientada al análisis de sección cruzada, pero su flexibilidad le confiere cierta utilidad como paquete de análisis de series temporales. Es el paquete adecuado si lo que se pretende es tratar datos que aúnan la dimensión temporal con la de sección cruzada: los llamados datos de panel.

Stata es un paquete de tratamiento de datos de apariencia similar a la de otros, en el sentido de procesar secuencialmente sentencias o comandos tecleados por el usuario dentro de una sesión. Para acceder al mismo, debe teclearse *STATA*² desde el sistema operativo. Le aparecerá, entonces, una pantalla de presentación, cuyo formato varía con la versión del paquete, al final de la cuál verá un punto seguido del cursor. Stata le informa a través de esta línea, precisamente, que está dispuesto para iniciar una sesión interactiva. Ejemplos de sentencias que podría utilizar entonces podrían ser:

infile using datos

o

generate prod=empleo/vab

La primera sentencia informa a Stata que deberá leer los datos del fichero *DATOS.RAW*, cuyo contenido está descrito –en un formato preestablecido– en el fichero *DATOS.DCT*. La segunda genera la variable *prod* como el cociente entre las variables *empleo* y *vab*. Ambas sentencias se ejecutarán una vez que las teclee y pulse la tecla Intro.

Como habrá podido observarse, Stata es un programa de apariencia similar a multitud de otros paquetes de tratamiento de datos: la invocación se realiza desde la línea de comandos del sistema operativo, tras lo cual el paquete espera que se le vayan introduciendo sentencias en un formato preestablecido, que irá ejecutando de forma secuencial. Las posibles sentencias siguen sin excepción una

(2) En minúsculas si ejecuta desde UNIX.

sintaxis determinada, siendo la sistematicidad en el formato uno de los elementos que más facilitan el manejo del paquete.

Al igual que muchos otros paquetes de análisis de datos, Stata puede funcionar de forma no interactiva (en *batch*), procesando sin intervención humana un fichero con sentencias. Habitualmente es preferible una sesión interactiva, que permite que el usuario interactúe con el paquete y tome decisiones sobre la marcha. Pero a veces conviene mantener tareas repetitivas o versiones definitivas de estudios en ficheros que permitan reproducir fácilmente una sesión. Se habrá observado que el interfaz con el usuario es estándar, en el sentido de necesitar el tecleo de las sentencias a procesar: existen, sin embargo, planes para desarrollar un interfaz gráfico, sobre la base del manejo de iconos, sin perder por ello el formato actual orientado a caracteres. Sin embargo, esto es, hoy por hoy, sólo un proyecto.

2. GESTIÓN DE DATOS

El primer paso para completar un estudio es, por supuesto, la búsqueda y reunión de datos. Usualmente, éstos se hallan dispersos en varias fuentes, y es necesario realizar dos tareas preliminares: reunirlos y expresarlos en un formato que facilite su tratamiento, que normalmente es el que reconozca el paquete que se desea utilizar. Estos dos pasos no suelen ser los fuertes de los paquetes de tratamiento de datos estándar, cuyo uso debe esperar a que los datos hayan sido convertidos a un formato que entiendan, por lo general bastante restringido. No es el caso de Stata, que también utiliza un formato específico, pero cuya potencia para tratar ficheros externos genéricos ASCII es uno de los puntos fuertes del paquete. Otro punto fuerte esencial para abordar esta tarea es la riqueza de tipos de dato que Stata reconoce, que va desde los enteros de menor cabida o precisión hasta los reales más precisos, pasando por literales alfabéticos. Esta profusión de tipos desconcierta al principio, pero demuestra, con el tiempo, ser una de las características más apreciables de Stata.

El tratamiento de ficheros se ve, además, facilitado por un programa distribuido por *Stata Corporation*, *TRANSFER*, que permite la rápida conversión de ficheros en formatos populares a formato Stata, y viceversa. Funciona en interactivo, mediante menús en donde se le indica al programa el nombre del fichero a transformar, el formato de entrada y el de salida. Formatos reconocidos por *TRANSFER* son los de las principales hojas de cálculo, los de las principales bases de datos y los de otros paquetes de cálculo de reconocido prestigio. Es un programa de compra opcional junto con Stata, pero altamente recomendable. La potencia de Stata a la hora de leer ficheros externos, aliado al uso de *TRANSFER*, hacen que, prácticamente, no haya que utilizar lenguajes de programación tradicionales para la reunión y elaboración previa de datos.

Stata facilita enormemente la tarea de controlar internamente la marea de datos habitual en los estudios en los que suele verse implicado. Pero la gran diferencia con otros paquetes en cuanto a manejo básico de datos se encuentra en su acceso a ficheros externos. En primer lugar, se tiene la ayuda inestimable del programa *TRANSFER*, que acompaña a Stata sin formar parte de él, cuyas ventajas y uso ya se han comentado. Pero mientras este programa se limita a convertir ficheros de un formato a otro, Stata se atreve sin otra ayuda con ficheros ASCII, sin importar lo extraño de su estructura: pueden leerse datos numéricos

o literales, en formato libre o fijo, con uno o varios registros por individuo, saltando información intermedia, etc. Para hacerlo, basta lanzar una orden *infile* seguida de los nombres de las variables a leer y del nombre del fichero, en el formato más sencillo de la orden, o seguida de *using* y del nombre de un fichero que contenga un diccionario Stata, en su formato más completo. Lo que en terminología del paquete se llama diccionario no es más que un fichero ASCII con información sobre el contenido del fichero de datos (nombre de cada variable, tipo de dato, posición del dato, etc). La orden inversa de *infile*, e igual de potente que ella, es la orden de escritura en formato ASCII *outfile*.

Aparte de una potente y cómoda lectura de ficheros ASCII, Stata permite el uso de ficheros en formato interno suyo mediante la orden *save*, que vuelca a disco duro toda la información disponible en memoria. El fichero puede volver a cargarse utilizando la orden *use* seguida del nombre de fichero generado, después de lo cual la memoria interna quedará tal y como estaba cuando se descargó el fichero. El programa acompañante *TRANSFER* reconoce este formato interno Stata.

Otras utilidades con ficheros externos incluyen la posibilidad de juntar varios en memoria, bien de forma secuencial o mediante pareado de claves coincidentes. Lo primero se logra cargando datos en memoria (mediante *use*, por ejemplo), y haciendo un *append* de otro fichero en formato Stata; lo segundo, mediante la orden *merge*. La diferencia entre ambas sentencias proviene de que la primera, *append*, añade nuevos individuos con las mismas variables, mientras que *merge* añade nuevas variables a individuos ya existentes. Este último comando exige que el fichero interno y el externo estén ordenados según la clave, pero esto es sencillo de realizar con Stata mediante el comando de ordenamiento *sort*, que permite el ordenamiento según una clave o varias anidadas, y que resulta una orden absolutamente fundamental. Como podrá atestiguar cualquiera que haya tenido experiencia con el paquete, los datos internos son reordenados de forma habitual y en multitud de ocasiones, en dependencia de lo que se pretenda hacer con ellos en cada momento dado. Este es uno de los elementos de Stata que mejor explican las infinitas posibilidades de manejo de datos que ofrece el paquete.

Los datos reconocidos por Stata incluyen tres tipos de enteros y dos tipos de reales de coma flotante, cada uno con mayor cabida que el anterior, pero más goloso de espacio en memoria. Stata genera por defecto las variables con el tipo de mayor precisión; para generarlas con otro tipo deberá señalarlo explícitamente y tener cuidado con las pérdidas de precisión o desbordamientos que puedan producirse si luego realiza cálculos sobre esa variable. En general, reserve esta posibilidad para variables contenidas en un fichero cuyo contenido conoce de forma previa y que no deban ser alteradas en la sesión. Se pueden definir y utilizar, también, variables de tipo carácter, en forma de literales alfanuméricos. Esta riqueza de datos permite, entre otras muchas cosas, que se pueda ajustar el espacio en memoria que ocupan los datos al mínimo, tema que, como se verá más adelante, tiene cierta importancia. El punto oscuro es la dificultad en el uso de los valores no disponibles ('missing values'). En otros paquetes, el valor correspondiente al 'missing value' suele ser el valor NAN definido en el estándar de representación de datos en coma flotante del IEEE, puesto que estos otros paquetes suelen utilizar exclusivamente reales de coma flotante. Stata no puede utilizar este valor con generalidad, ya que utiliza tipos de datos enteros, además de reales. Por ello, usa una representación de 'missing value' distinta para cada tipo de dato,

y que consiste en el valor numérico máximo que admita cada uno. Desgraciadamente, muchas operaciones no tienen en cuenta esta convención y fuerzan a menudo al usuario a controlar él mismo la presencia o ausencia de valores ‘missing value’.

Otro punto importante de Stata es la gestión excepcional de la memoria, aunque este es un punto en el que tiene mayor competencia por parte de otros paquetes. Stata reserva un espacio en memoria para los datos que viene definido por tres cantidades: el número máximo de observaciones o individuos, el número máximo de variables y el espacio máximo que ocuparán estas variables para cada uno de los individuos. El último dato es necesario dada la diversidad de tipos de dato que maneja Stata, que no permite una relación directa entre número de variables y espacio que ocuparán. El grado de finura que permiten tantos grados de libertad posibilita que Stata pueda abordar el tratamiento de cantidades enormes de datos, particularmente en sus versiones para UNIX³ y para DOS extendido (máquinas 386 bajo DOS). Esto lo hace particularmente útil para análisis de datos de panel.

No todo consiste en llenar la memoria de datos, es necesario además que sean fácilmente accesibles. Stata es similar a otros paquetes en el sentido de referenciar las variables mediante un nombre de un máximo de 8 caracteres, pero añade a esto la facilidad de asignar a cada variable una leyenda o *label*, que mostrará en gráficos y salidas similares. Esta leyenda permite, además, que una descripción del contenido de la variable acompañe a ésta, peculiaridad extremadamente útil cuando se tienen cientos de datos de cada individuo, en cuyo caso los nombres de las variables acaban no siendo muy descriptivos. A esto añade la posibilidad de asignar leyendas a valores específicos que tomen algunas variables: por ejemplo, en un fichero con una variable numérica indicativa de la provincia se le puede tanto asignar *label* a la variable (por ejemplo, “Provincia”), como a cada uno de los posibles códigos (por ejemplo, “Madrid” para el código 28). Ambas leyendas aparecerán posteriormente en gráficos, tabulaciones, etc, en vez del nombre de la variable o del código correspondiente. Otra posibilidad que admite Stata, muy útil para ahorrar espacio en memoria, es la inversa de la anterior; es decir, convertir un literal leído en un fichero a códigos numéricos definidos mediante una tabla de conversión. De esta forma se conserva en memoria el código numérico y se ignora el literal presente en el fichero.

Otro elemento de ayuda en el manejo de datos, más usual, es el uso de expresiones. Stata utiliza las expresiones, básicamente, en las instrucciones *generate* y *replace*. La primera genera variables nuevas, la segunda modifica el contenido de variables ya existentes. Las expresiones permiten el uso de los operadores habituales —operaciones relacionales, lógicas, aditivas y multiplicativas—, al tiempo que se pueden utilizar funciones a elegir dentro de un amplio menú, incluidas funciones estadísticas o de manejo de literales.

Se pueden, además, lograr descripciones rápidas del contenido de memoria mediante una serie de sentencias utilitarias, como *describe*, *list*, *count*, *tabulate*, o *summarize*. La primera da una descripción del contenido de memoria: nombres de las variables, leyendas correspondientes y variables según las cuales el fichero está ordenado. Las otras permiten listar el contenido de algunas variables (*list*), contar individuos (*count*) u obtener tabulaciones o descripciones estadísticas de

(3) UNIX es una marca registrada de *UNIX Systems Labs*.

las variables (*tabulate*, *summarize*). Aunque estas sentencias no alteran los valores en memoria, resultan fundamentales para hacerse una idea de su contenido, aunque es de lamentar alguna carencia en las posibilidades de tabulación de variables.

Un punto en el que Stata no está al nivel de muchos paquetes es en el de tratamiento de fechas. En principio, lo que se procesan son individuos, como ha quedado dicho, y Stata no es el paquete de tratamiento de datos adecuado si lo que se desea es realizar análisis de series temporales. En consecuencia, no existe ninguna variable interna que indique la 'fecha' de la observación corriente, ni puede indicarse de forma directa un rango de fechas al que limitar una acción. Todo ello puede lograrse gracias a la flexibilidad de Stata, siempre y cuando exista una variable que recoja el período temporal a que corresponde cada observación. Desgraciadamente, esto fuerza a veces a expresar acciones de una forma poco intuitiva. Aunque se pueden tratar series temporales, su uso es complicado.

3. ANÁLISIS Y DESCRIPCIÓN DE DATOS

El paso inmediatamente posterior a la carga de los datos en Stata es lograr una primera descripción de los mismos, ya sea para detectar errores en ellos, o ya sea para desechar los que sean menos fiables. Este es un punto en el que Stata resulta particularmente fuerte, dada su capacidad para representar los datos tanto gráfica como estadísticamente de una forma sencilla.

Dos son los aspectos que más ayudan a facilitar el manejo de las posibilidades descriptivas de Stata: en primer lugar, la sintaxis sistemática de las sentencias o comandos Stata, tema ya comentado; en segundo lugar, la eficaz ayuda interactiva que posee el paquete, que permite localizar rápidamente cuál es la orden precisa que más nos interesa para una acción en particular, y las opciones de esa orden que mejor nos servirán para lograr nuestro propósito. A la ayuda se puede acceder mediante el comando *help*, que puede darse o bien aisladamente, o bien seguida de la orden cuya descripción nos interesa. Para obtener la descripción de un comando deberemos teclear *help* seguido del nombre del comando, tras lo cual veremos desfilar una serie de pantallas explicativas del mismo, a razón de una pantalla cada vez que pulsemos la tecla Intro. Tecleando *help* solamente, se nos listarán las sentencias cuya descripción podremos pedir. Podemos, además, añadir libremente textos de ayuda adicionales, que deberemos editar en un fichero ASCII, y que luego el comando *help* reconocerá.

La ayuda interactiva resulta indispensable si tenemos en cuenta que el número de sentencias Stata que permiten describir datos es bastante grande, y la inherente extensibilidad del paquete hace que sea potencialmente infinito. Ejemplos de sentencias de este tipo son: *anova*, *boxcox*, *canon*, *logit*, *correlate*, *factor*, *hist*, *logistic*, *means*, *regress*, *summarize*, *tabsum*, y una multitud de otras, de las cuales sólo una minoría son internas a Stata —es decir, forman parte intrínseca del programa. Algunas de estas sentencias son, propiamente hablando, métodos de estimación, y difícilmente pueden calificarse como meros modos de describir datos; pero, como se verá más adelante, la estimación en sí puede ser abordada con herramientas mucho más poderosas que las mencionadas.

Se ha comentado la posibilidad que existe de extender Stata con programas escritos en un lenguaje propio. De hecho, el paquete admite dos modalidades de ejecución diferida. En primer lugar, es posible indicar a Stata que procese desde una sesión interactiva un fichero que contenga instrucciones a ejecutar, mediante

las sentencias *do* o *run* seguidas del nombre de fichero. Ambas difieren en que *do* presenta por pantalla las mismas salidas que se hubiesen obtenido tecleando las sentencias directamente, mientras *run* realiza el trabajo silenciosamente. Esto permite editar un fichero con estas sentencias y, posteriormente, ejecutarlas cuantas veces se quiera. La segunda modalidad diferida es bastante más interesante: programas Stata propiamente dichos, usualmente también contenidos en un fichero externo, pero cuya lectura permite cargar el juego de instrucciones en memoria sin ejecutarlas. El fichero debe comenzar con una sentencia *program define* seguida del nombre del procedimiento que queremos definir, seguido a su vez de las sentencias Stata que permiten completar la tarea que tiene asignada. Es posible pasar argumentos al procedimiento —parámetros sobre los que actuará—, aunque su uso es un tanto peculiar en Stata; existe también la posibilidad de definir variables locales al procedimiento, sólo activas y presentes en memoria durante las ejecuciones del mismo. La ejecución se produce, dentro de la sesión, sin más que invocar el programa por su nombre tantas veces como se quiera. En ese caso, no hay relectura del fichero a cada ejecución.

Por si fuera poco lo dicho, existe una tercera posibilidad de ejecutar procedimientos no internos: si Stata encuentra una orden que desconoce, antes de abortar la acción busca un fichero cuyo nombre sea el del verbo mencionado y su extensión sea *.ADO*, y lo carga en memoria en caso de encontrarlo. De esta forma, por ejemplo, la orden *canon* ya mencionada —que realiza un análisis de correlaciones canónicas— no forma parte de Stata, sino que es un programa contenido en el fichero *CANON.ADO*. Cuando invocamos *canon* por primera vez en una sesión, Stata carga en memoria este fichero y ejecuta el procedimiento *canon* de forma absolutamente transparente para nosotros. En las sucesivas invocaciones, Stata conserva en memoria las sentencias del procedimiento, por lo que no repite la lectura del fichero. Este método es tan sencillo y elegante que acaba resultando una lástima que el número de programas ADO existentes no sea mayor, aunque tanto desde la casa *Stata Corporation* como desde otras fuentes hay un flujo continuo de procedimientos de este tipo que están ayudando a extender tremendamente la utilidad del paquete.

La descripción estadística de datos no es la única posible: la descripción gráfica es muchas veces igual de reveladora. Suele ser un paso indispensable en el análisis preliminar de los datos. Stata tiene razonablemente cubierto este extremo mediante una orden para realizar gráficos que permite obtener con alguna facilidad salidas de cierta calidad. Esta orden es *graph*, cuyo generoso juego de opciones posibles permite realizar gráficos muy distintos entre sí: histogramas, gráficos de entrada única, de doble entrada, de tarta, de barras, etc. Permite, además, realizar gráficos ‘matriciales’, presentando en una misma pantalla todos los cruces bidimensionales entre un conjunto de variables. El control que permite la orden *graph* de elementos como títulos, subtítulos, leyendas, etc, es relativamente bueno; se permite, también, acumular gráficos en una sola pantalla, aunque con cierta dificultad. Es posible imprimir los gráficos mediante el uso de los programas *gphdot* o *gphpen*, que vuelcan respectivamente a una impresora o a un *plotter* el gráfico conseguido, y que pueden ser invocados desde Stata.

El único problema serio de Stata a este respecto es la ausencia de gráficos tridimensionales, cuya utilidad es cada vez menos cuestionada y que ya ofrecen muchos paquetes. De hecho, las posibilidades del paquete permiten la rápida y cómoda ejecución de gráficos en borrador, muy útiles durante el trabajo de análisis

preliminar de datos, pero no está orientado a la confección de gráficos de calidad. Stata no es, ni pretende ser, un paquete gráfico, ya que no cubre ni mucho menos todas las posibilidades que ofrecen éstos, ni puede sustituirlos. Si es usted consciente de esto, no tendrá problemas serios con las capacidades gráficas del paquete.

4. FORMALIZACIÓN DEL MODELO

Un paquete de tratamiento de datos que cubra correctamente la formalización de un modelo, si ésta es deseada, debe, necesariamente, ofrecer un buen abanico de posibilidades en cuanto a técnicas estadísticas. Muchas veces resulta indispensable que se puedan programar aquellas que no traiga ya incluidas. Stata da a este respecto una de cal y otra de arena: curiosamente, a pesar de su manifiesta potencia, incluye pocos métodos estadísticos como parte intrínseca del paquete. La mayor parte de ellos son procedimientos externos, en lenguaje Stata, ya existentes; el resto, aquellos que no están incluidos, deberán ser programados por el propio usuario u obtenerse a través de aportaciones de terceros. Es aquí donde la relativa falta de procedimientos Stata pre-programados se hace sentir más angustiosamente.

Por supuesto, Stata incluye las técnicas de estimación más comunes: entre otras, regresión lineal, modelos logit y probit, o modelos con distribuciones truncadas. Existen, también, técnicas multivariantes, como análisis factorial o canónico, estimación de modelos lineales o no lineales, probit y logit multivariantes, o máxima verosimilitud. La mayor parte de ellos son procedimientos pre-programados, usualmente ficheros ADO que se cargan automáticamente, y que se entregan con el paquete o pueden conseguirse a través de *Stata Corporation*. Las aportaciones de terceros no abundan, a pesar de que el paquete está relativamente extendido, siendo esto, sin duda alguna, uno de los mayores problemas de Stata, al menos para los usuarios más avanzados.

Otro punto de crítica ya esbozado anteriormente es el tratamiento pobre de las series temporales. Si desea hacerse una regresión que implique a variables retardadas, caso muy común con datos de panel, es necesario crear una variable nueva por cada retardo a incluir. Dado que Stata suele manejar bases de datos ya sobrecargadas, esta característica del paquete es particularmente ineficiente. Es una lástima que las sentencias de Stata que actúan sobre variables citadas por su nombre no lo hagan sobre listas de expresiones, evitándose así la creación de inútiles variables intermedias.

Éstas son las malas noticias. Las buenas son la alta programabilidad del paquete, que permite que prácticamente cualquier cosa pueda ser incluida bajo la forma de procedimiento. Muy pocos paquetes son capaces de incluir, como lo hace Stata, técnicas de estimación no lineal o de máxima verosimilitud exclusivamente en base a estos procedimientos, como extensión al propio paquete. Un elemento que ha potenciado mucho este aspecto de Stata ha sido la inclusión, a partir de la versión 3.1, de un lenguaje matricial propio. Es decir, ahora se pueden definir y manejar matrices directamente desde Stata, lo que permite incluir en el paquete la mayor parte de los métodos estadísticos más recientes de una forma bastante intuitiva. Ahora es posible crear matrices, sumarlas, multiplicarlas, aplicarles el producto de Kronecker, obtener su transformada por Cholesky o por SVD (descomposición de valor singular), etc. Este añadido al paquete puede suponer

una revolución en el mundo de Stata. En todo caso, el lenguaje de procedimientos del paquete era ya muy potente antes de esto, aunque si usted posee una versión de Stata anterior a la 3.1 le recomendamos que pida a la casa una actualización a la nueva versión.

Insistamos, de todas formas, en que hay una escasez de procedimientos distribuidos. Faltaría, por ejemplo, incluir las técnicas estadísticas de “clusterización”, de análisis factorial de correspondencias, de análisis discriminante y otras igual de relevantes; en el campo econométrico, es una lástima que no exista mayor riqueza en el tratamiento de variables instrumentales y, en particular, que no se incluyan procedimientos de estimación por el método generalizado de momentos (GMM) y, también, que no se pueda simular fácilmente con los modelos estimados. Obsérvese que cualquiera de estas técnicas puede programarse con Stata, y probablemente lo será en el futuro.

5. PRESENTACIÓN DE RESULTADOS

Muchos estudios del tipo de los que se realizan con Stata agradecen el que los resultados se presenten acompañados de gráficos. Para hacerlos puede utilizarse el propio paquete de tratamiento de datos, si permite esta posibilidad, o bien acudir a un paquete específico de confección de gráficos. El problema de la primera opción es la escasa calidad del gráfico resultante, dadas las limitaciones en este área de muchos paquetes de tratamiento de datos. Los gráficos de Stata son lo suficientemente buenos como para ser aprovechados de esta forma: les faltaría, si acaso, la posibilidad de elegir en los títulos los tipos de letra y su grosor. Stata permite, en todo caso, mejorar sus gráficos mediante la inclusión de títulos para el gráfico y cada uno de sus ejes, cambiando la forma de conectar los puntos, cambiando las escalas, etc.

Una ayuda adicional que presta Stata para que puedan aprovecharse sus gráficos es un programa que puede adquirirse junto con él: *STAGE*. *STAGE*, al igual que *TRANSFER*, es un añadido opcional que puede adquirirse o no junto con Stata, y que permite editar gráficos obtenidos por éste. Las posibilidades de edición no son excesivas y se limitan, a efectos prácticos, a reescalar los títulos, desplazar elementos dentro del gráfico, rotarlo u operaciones básicas de ese tipo. Permite realizar operaciones más complejas, pero el aprendizaje necesario para ello es parecido al de un paquete específicamente gráfico. En general, por lo tanto, la utilidad de este programa es menor que la de *TRANSFER*.

6. CONCLUSIONES Y RECOMENDACIONES

En resumidas cuentas, Stata es un paquete de tratamiento de datos que le puede dar respuesta a sus problemas, sean estos pequeños y simples o grandes y complicados, sin resultar un monstruo en cuanto a tamaño y conocimientos requeridos. La fácil sintaxis de Stata le permitirá cargar los datos y obtener una descripción de los mismos prácticamente desde que tome contacto con el paquete. Los procedimientos entregados con el mismo le permitirán abordar tareas de mayor complejidad, y el flujo continuo de nuevos procedimientos que se están distribuyendo continuamente le permitirán incrementar este grado de complejidad. Finalmente, si se toma el trabajo de aprender a programar procedimientos Stata,

y muy en particular su lenguaje matricial, obtendrá un control total del paquete y la posibilidad de abordar cualquier tipo de tarea sin restricciones.

Debe quedar claro a estas alturas que Stata *no* es el mejor paquete de tratamiento de datos, si es que existe alguno que lo sea. Pero sin duda alguna está entre los primeros, y es una opción muy interesante para ciertos perfiles de usuario. En particular, si sus preocupaciones mayores no se centran en el análisis de series temporales, sino más bien en el de datos de sección cruzada; tiene, de entrada, una gran masa de datos que procesar; no le gusta entretenerse más de lo necesario en aprender el uso del paquete; pero pide, por otro lado, una extensibilidad futura total al mismo, entonces Stata es probablemente su paquete. Si la mayor parte de su trabajo se realiza con series temporales, prefiere manejar paquetes en entornos gráficos o su trabajo exige trabajar con las técnicas de estimación más recientes, existen para usted opciones mejores que Stata. Si, finalmente, trabaja sobre todo con datos de panel, Stata es probablemente uno de los paquetes que le resultarán más cómodos, al menos mientras espera que aparezcan otros específicamente diseñados para la tarea. En todo caso, si planea comprar algún paquete de tratamiento de datos, tómese la molestia de evaluar Stata: no es un paquete que pueda ignorarse.

