

EL PROGRAMA MATLAB Y SU USO EN EL ANÁLISIS ECONOMETRICO*

GABRIELE FIORENTINI

CEMFI

REGINA KAISER

Universidad Carlos III

El programa MATLAB¹ (Matrix Laboratory) es un sistema interactivo y un lenguaje de programación pensado fundamentalmente para resolver problemas de tipo matemático.

El origen de MATLAB puede encontrarse en dos librerías de subrutinas FORTRAN, conocidas como LINPACK Y EISPACK, desarrolladas por Cleve Moler para resolver problemas de ecuaciones lineales y factorización de matrices. A finales de la década de los 70, Cleve Moler escribió un programa interactivo, que permitiese un fácil acceso a esas subrutinas FORTRAN y le dio el nombre de MATLAB. Su bajo coste de aprendizaje inicial y la posibilidad de aplicación a diversos campos hizo que el número de usuarios creciese rápidamente. En 1984 se fundó la compañía Mathworks que lo comercializa y se encarga de incorporar los desarrollos posteriores (quizás el primer cambio fundamental fue la transcripción de las subrutinas originarias a lenguaje C).

El elemento fundamental de MATLAB es la matriz, real o compleja, siendo los vectores y escalares casos particulares de la misma. Las instrucciones en este lenguaje son interpretadas y no compiladas, lo que permite al usuario especificar comandos interactivamente desde su terminal. Las operaciones que pueden llevarse a cabo tienen expresiones sencillas e intuitivas parecidas a las usadas en matemáticas o ingeniería. Por ejemplo, la fórmula de la regresión por Mínimos Cuadrados Ordinarios,

$$\hat{\beta} = (X'X)^{-1} X'y$$

donde, X es una matriz ($n \times k$) e y es un vector ($n \times 1$), tiene la siguiente representación en MATLAB,

$$beta = inv(X' * X) * X' * y \quad \text{ó} \quad beta = X \backslash y$$

(*) En la realización de este trabajo nos hemos beneficiado de los comentarios de Miguel Angel Arranz, Raquel Carrasco, Humberto López e Ismael Sánchez.

(1) The Mathworks, Inc., 24 Prime Park Way, Natick, MA 01760-1520, USA. E-mail: info@mathworks.com.

La última versión disponible, por el momento, de MATLAB es la 4.2 que presenta varias novedades con respecto a las versiones de la generación 3. En particular, las posibilidades gráficas han sido bastante mejoradas en esta última versión.

El MATLAB básico se compone de cientos de funciones que permiten la solución de problemas de todo tipo, como pueden ser el cálculo de la media o la varianza de un grupo de datos, la obtención de las raíces de un polinomio y su factorización, o la descomposición de Cholesky de una matriz. Se comercializan por separado una serie de *toolboxes* o librerías compuestas por funciones válidas para propósitos más específicos como son control, optimización o análisis espectral, entre otros. Por último, Prentice-Hall distribuye una versión de MATLAB para estudiantes que contiene las funciones básicas y resulta más económica.

MATLAB está pensado para explotar al máximo las posibilidades de los ordenadores personales, tanto IBM como APPLE MACINTOSH O POWER MACINTOSH. MATLAB también está disponible para otras plataformas *hardware*, que van desde las estaciones de trabajo más populares hasta grandes máquinas como VAX, CONVEX O los superordenadores CRAY. El programa MATLAB puede utilizarse bajo los entornos DOS, WINDOWS, OS/2, UNIX, MACINTOSH, OS Y VMS. Para la versión 4.2 los requisitos en PC son un procesador Intel-based 486 o Pentium (para procesadores 486-SX también es necesario un coprocesador matemático), 8 Mb de RAM Y 15 Mb de espacio libre en el disco duro. Pueden utilizarse versiones precedentes de MATLAB con procesadores 386. Para máquinas con entornos UNIX se requieren 20 Mb de espacio en el disco duro, 16 Mb de memoria y se recomiendan 64 Mb de aérea virtual en disco.

Mathworks dispone de varias páginas en Internet donde puede encontrarse cualquier tipo de información sobre sus productos. De especial interés es el buzón donde los usuarios pueden dialogar entre ellos exponiendo problemas y soluciones. Más información puede obtenerse en la dirección de Internet: www.mathworks.com.

1. CARACTERÍSTICAS PRINCIPALES

La mayoría de las funciones en el MATLAB básico y en las librerías están escritas en lenguaje MATLAB, lo que permite el acceso del usuario a los ficheros fuente. Esta es una de las principales ventajas de lenguajes como MATLAB, ya que el usuario no sólo puede modificar las funciones existentes sino programar sus propias aplicaciones hasta llegar a organizar verdaderas librerías.

El coste fijo inicial necesario para manejar el programa MATLAB no es excesivamente grande. Conforme crecen las necesidades de los usuarios, la excelente ayuda en línea (comando *help*) se convierte en una herramienta de gran utilidad ya que es clara, detallada y aporta referencias cruzadas entre comandos. Las funciones escritas por el usuario también pueden incluirse en la ayuda en línea, facilitando el manejo por parte del propio usuario o el intercambio de funciones entre distintos usuarios.

El tratamiento de los datos resulta sencillo y flexible. Los datos pueden leerse y almacenarse con dos formatos diferentes, formato ASCII Y formato MATLAB. La ventaja del formato ASCII radica en que los datos y resultados pueden utilizarse en otros programas. Sin embargo, el formato MATLAB puede resultar más eficiente ya que las operaciones de lectura y escritura se realizan con mayor rapidez.

Dado que MATLAB es un lenguaje matricial conviene explotar sus capacidades, expresando el mayor número posible de operaciones en forma matricial y evitando el uso de bucles *for* que pueden ralentizar los procesos de ejecución. Por ejemplo, para construir un proceso x_t de media móvil de orden 1 y parámetro $\theta = .8$, a partir de una serie de innovaciones a_t , $t = 1, \dots, T$, resulta más rápido utilizar,

```
>> x(1)=a(1);
>> x(2:T)=a(2:T)+.8 * a(1:T-1);
```

en vez de,

```
>> x(1)=a(1);
>> for i=2:T,
>> x(i)=a(i)+.8 * a(i-1);
>> end
```

MATLAB permite trabajar interactivamente escribiendo sentencias directamente en la línea de comandos del programa (>>). También puede trabajarse en forma por lotes (*batch*), escribiendo ficheros con extensión *m* (ejemplo: prog.m), cuyas líneas son instrucciones de MATLAB, que pueden ejecutarse desde la línea de comandos (>> prog).

MATLAB no dispone de un editor propio pero el usuario puede acceder a su editor habitual con el comando *edit*. También es posible la ejecución de instrucciones del sistema operativo (comando !) sin necesidad de salir del programa. Estas dos características hacen que una forma habitual de trabajar en el entorno de MATLAB sea la combinación de los modos interactivo y por lotes. Esta combinación tiene la ventaja de permitir al usuario ejecutar sus programas y, a continuación, modificarlos si lo considera necesario.

En cada sesión de trabajo las variables que se crean se van guardando en la memoria del programa. Con el comando *whos* se obtiene un listado de las variables en memoria con sus dimensiones y el total de la memoria ocupada. En principio, no existe ningún límite para la dimensión de una matriz ya que MATLAB usa una asignación dinámica de la memoria, de forma que los límites para la memoria son los impuestos por el sistema operativo o por el hardware. En casos de problemas de agotamiento de memoria (mensaje OUT OF MEMORY) el comando *clear* elimina las variables que no se van a volver a utilizar y el comando *pack* comprime los datos en memoria. En general, conviene no guardar variables innecesarias ya que el control sintáctico del interprete del programa hace que la rapidez de ejecución de las sentencias sea inversamente proporcional al espacio ocupado.

Dos utilidades de indudable interés práctico son, en primer lugar, el comando *diary* que permite grabar en un fichero todas las operaciones realizadas durante una sesión de trabajo y, en segundo lugar, la posibilidad de incorporar, mediante el uso de los ficheros MEX, rutinas escritas en lenguaje C o FORTRAN (también es posible realizar el proceso inverso).

Cabe destacar la gran potencia de las utilidades gráficas de MATLAB y su excelente calidad apta para publicaciones de tipo profesional. Se pueden realizar gráficos de una o varias series (*plot*), se puede dividir la pantalla en sectores con un máximo de cuatro (*subplot*), se pueden realizar gráficos en tres dimensiones (*mesh*), gráficos de contornos (*contour*) y, para aplicaciones de estadística descriptiva, se pueden representar histogramas (*hist*) o diagramas de barras (*bar*). Otros tipos de gráficos, como el correlograma o el espectro teórico de una serie temporal, pueden construirse

con relativa facilidad. El comando *axis* aporta bastante flexibilidad para cambiar la escala de los ejes en los gráficos y puede utilizarse como un *zoom* para evidenciar sectores de los gráficos que son de especial interés, en la versión 4.2 esto se realiza de forma más sencilla desplazando el ratón a la zona del gráfico que se quiera resaltar.

Los gráficos realizados en MATLAB se pueden grabar en ficheros (comando *meta*) que, posteriormente, con el programa GPP se pueden transformar, dependiendo del uso que se le quiera dar y del *hardware* de impresión de que se disponga. Por ejemplo, se podrían transformar en ficheros *postscript* e importarlos en procesadores de texto como LATEX, SCIENTIFIC WORD O WORD PERFECT.

A nuestro parecer el mayor problema de MATLAB está en el dispositivo de localización de los errores. En general, los mensajes de error no son excesivamente claros y, además, cuando los errores se producen en el interior de un bucle de programación el dispositivo sólo hace referencia a la línea donde comienza ese bucle. Una consecuencia de este hecho es que, en programaciones muy estructuradas, la información proporcionada por el dispositivo de localización no resulta ser de gran utilidad.

MATLAB no permite explícitamente la creación de *arrays* multidimensionales, por lo que, por ejemplo, no se pueden asignar valores a una sucesión de matrices. En teoría, los *arrays* multidimensionales podrían manejarse creando funciones en lenguaje MATLAB, sin embargo su utilización resulta bastante compleja.

Dos últimas características de MATLAB que, no constituyendo un verdadero problema, resultan molestas son, en primer lugar, el hecho de que cada sentencia debe terminar con punto y coma para evitar que el resultado de dicha sentencia sea reproducido en pantalla y, en segundo lugar, que MATLAB distingue entre letras mayúsculas y minúsculas, no sólo para nombrar variables sino también comandos, lo que en algunas ocasiones es fuente de confusiones para el usuario.

2. ANÁLISIS ECONÓMETRICO

Como se ha mencionado anteriormente, MATLAB es un lenguaje de programación y por lo tanto, no debe considerarse como un paquete específico para algún área de conocimiento, como lo son TSP O MICROFIT, entre otros, para el análisis econométrico.

Existen, sin embargo, varias librerías que contienen funciones y aplicaciones de cierto interés para el análisis econométrico.

System Identification Toolbox (IDENT) contiene funciones para la modelización paramétrica de series temporales univariantes. Están incluidos los modelos AR, ARMA, modelos tipo Box-Jenkins o modelos estructurales. La estimación de los parámetros se realiza con diferentes métodos como mínimos cuadrados ordinarios (para AR Y ARX), O el método de la descomposición del error de predicción. Esta librería está enfocada al campo de la ingeniería por lo que utiliza una terminología distinta a la habitual lo que hace complicado su uso para un analista econométrico.

Optimization Toolbox (OPTIM) es una librería para la optimización lineal y no lineal. Realiza, entre otras cosas, minimización restringida y no restringida, mínimos cuadrados no lineales, programación lineal y cuadrática y solución de sistemas de ecuaciones no lineales. Esta librería podría, en principio, utilizarse para la estimación por máxima verosimilitud de cualquier tipo de modelo econométrico. Su uso no está muy extendido entre los analistas que, en general, prefieren la librería MAXLIK del

programa GAUSS ya que ésta última es más específica para la solución de problemas de tipo econométrico, al tener un mayor número de opciones para elegir el algoritmo de maximización y proporcionar directamente la matriz de varianzas y covarianzas de los parámetros estimados. [Rust (1993), compara los programas MATLAB Y GAUSS].

Statistics es una librería de funciones para el análisis descriptivo, inferencial o gráfico, o para la construcción de modelos lineales. Incluye catorce distribuciones de probabilidad, desde la normal hasta las Weibull, beta o Poisson, con sus respectivas funciones de densidad, distribución, función inversa y generador de números aleatorios. Considera casos particulares del modelo lineal general, entre ellos análisis de la varianza, intervalos de confianza para polinomios o regresión múltiple.

Financial es una librería desarrollada recientemente para responder al uso cada vez más extendido de herramientas matemáticas avanzadas en el análisis financiero moderno. Algunos ejemplos de las posibilidades de esta librería son el análisis de selección de cartera que permite hallar la frontera eficiente de acuerdo con el modelo de Markowitz, también se pueden calcular precios y rentabilidad de activos derivados y precios de las opciones europeas con la formula de Black-Sholes.

Symbolic Math añade al lenguaje MATLAB funciones simbólicas matemáticas basadas en Maple V. Algunas de las operaciones de computación simbólica más relevantes son diferenciación, integración, series de Taylor, inversión de matrices, simplificación de expresiones, solución de ecuaciones algebraicas o diferenciales.

Otras librerías que merece la pena destacar son, *Neural Network*, para el diseño y la simulación de redes neuronales (permite trabajar con varias versiones de la regla de retropropagación) y *Signal Processing*, para el procesamiento digital de señales.

La forma habitual en la que nosotros trabajamos es escribiendo nuestros propios programas en vez de utilizar exclusivamente las librerías de MATLAB. Describiremos brevemente, en primer lugar, algunas de las funciones útiles en el análisis econométrico de series temporales y, posteriormente, varias aplicaciones que se podrían llevar a cabo. Dado que nuestro interés se centra en el análisis econométrico de series temporales mencionaremos ejemplos en este ámbito; no obstante también son posibles las aplicaciones de tipo microeconométrico con análisis de datos de sección cruzada y paneles.

Las medidas descriptivas para un conjunto de datos pueden obtenerse rápidamente con las funciones *mean*, *std*, *cov*, *median*, *corrcoef*, *max*, *min*. Para generar un vector o una matriz de números aleatorios que siguen una distribución normal estándar se utiliza *rand*, también pueden generarse datos distribuidos uniformemente en el intervalo (0,1) con *rand*. El MATLAB puede realizar operaciones con polinomios, lo que resulta especialmente útil para el análisis de modelos Box-Jenkins ya que éstos se expresan como polinomios en el operador de retardo. Las operaciones de multiplicación y división pueden realizarse con *conv* y *deconv*, las raíces se calculan con *roots* lo que permite analizar la dinámica de un proceso autorregresivo, mientras que *poly* realiza el cálculo inverso hallando los coeficientes de un polinomio a partir de las raíces reales o complejas. La función *filter* consiste en un filtro racional que, por ejemplo, permite la generación de procesos ARIMA a partir de una serie de ruido blanco. La aplicación del filtro ARMA inverso a un vector con todos sus elementos igual a cero menos el primero igual a la unidad proporciona los pesos de la representación de Wold del proceso ARMA. La función *filter* también resulta útil para determinar el efecto de distintos tipos de valores atípicos sobre las observaciones o sobre la serie de

residuos estimados de un modelo ARIMA. Por último, para el análisis multivariante de series temporales conviene tener en cuenta las funciones *inv* que calcula la inversa de una matriz, *eig* que proporciona los autovalores y *chol* que realiza la descomposición de Cholesky.

Usando, entre otras, las funciones mencionadas anteriormente hemos escrito, con relativa facilidad, programas para calcular el espectro teórico de un proceso estocástico, para realizar el contraste de cointegración de Johansen, para identificar el tipo y la posición de varios tipos de valores atípicos, para calcular las predicciones y bandas de confianza en modelos ARIMA y para realizar simulaciones de Monte Carlo que permiten obtener el valor crítico de un contraste. López (1995) ha desarrollado ECOTX, una librería de programas escritos en lenguaje MATLAB que contiene varias funciones útiles para el análisis econométrico de series temporales. En particular, lleva a cabo contrastes de raíces unitarias (ej. Dickey-Fuller Aumentado y Phillips y Perron), estimación de los parámetros de modelos ARIMA, ajuste de la estacionalidad de una serie por medio de una versión simplificada del filtro X-11, estimación de la tendencia con el filtro de Hodrick y Prescott, estimación del espectro de una serie mediante el uso de ventanas de Barlett, análisis de persistencia y de funciones de impulso-respuesta y aplicaciones del filtro de Kalman.

Aunque ninguna de las librerías de las que MATLAB dispone actualmente contiene aplicaciones de tipo econométrico, la creciente difusión del programa entre estadísticos y analistas econométricos han motivado el compromiso por parte de MATHWORKS de crear una librería, actualmente en desarrollo, específica para este colectivo.

Un programa similar a MATLAB es GAUSS. Ambos proporcionan en sus versiones básicas parecida cobertura de las principales herramientas matemáticas y estadísticas. La orientación de GAUSS es estadística/econométrica y contiene una extensa gama de funciones para estimar de forma lineal o no lineal modelos de sección cruzada o datos de panel. Por el contrario, MATLAB es más específico para el tratamiento de modelos macroeconómicos y de series temporales.

3. CONCLUSIONES

MATLAB es un lenguaje interactivo de programación con el que se pueden resolver problemas de tipo matemático. Consideramos que MATLAB constituye globalmente un paquete de gran potencia y flexibilidad; sin embargo es importante resaltar algunos aspectos negativos del mismo:

- Aunque la versión básica de MATLAB no tiene un precio demasiado elevado, resulta caro completar el paquete con las numerosas librerías auxiliares.

- El localizador de errores es bastante primitivo y proporciona mensajes poco claros que generan confusión.

- El hecho de que MATLAB no sea un lenguaje compilable hace que la ejecución de los programas se ralentice considerablemente. Este aspecto parece haberse solucionado en la última versión que incluye un compilador.

Sin duda nuestra experiencia de trabajo con MATLAB ha sido muy positiva y entre sus ventajas destacaremos las siguientes:

- Sigue una lógica intuitiva, lo que facilita la tarea de traducir ideas en programas. El usuario, por tanto, no necesita ser un programador experto.

- Para empezar a utilizarlo basta con un versión básica a la que se pueden ir añadiendo librerías según las necesidades de cada usuario.
- Es un programa multiplataforma.
- Es abierto y extensible. Los programas y funciones dentro de las librerías son de libre acceso, lo que permite al usuario inspeccionar y modificar los ya existentes o crear otros nuevos.
- Pueden incluirse programas en C o FORTRAN.
- Amplias posibilidades gráficas.
- Los modelos y algoritmos son revisados constantemente por investigadores, académicos y teóricos.



REFERENCIAS BIBLIOGRÁFICAS

- Ljung, L. (1986): *System Identification Toolbox: Manual*, The Mathworks, Inc., Sherborn, Mass.
- Ljung, L. (1987): *System Identification: Theory for the User*, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.
- López, H. (1995): "An Econometric Library for MATLAB: ECOTX", European University Institute, *mimeo*.
- Rust, J. (1993): "GAUSS and MATLAB: a Comparison", *Journal of Applied Econometrics*, 8, 307-24.