



PROBANDO EL LENGUAJE DE PROGRAMACIÓN MATA DE STATA

LUISA FERNANDA RIASCOS CAIPE

**DIRECTOR DEL PROYECTO:
CARLOS GIOVANNI GONZÁLEZ ESPITIA**

**UNIVERSIDAD ICESI
FACULTAD DE CIENCIAS ADMINISTRATIVAS Y ECONÓMICAS
ECONOMÍA Y NEGOCIOS INTERNACIONALES
SANTIAGO DE CALI
2015**

TABLA DE CONTENIDO

Resumen	4
1. Introducción	5
2. Revisión de la literatura.....	6
3. Metodología	7
4. Resultados	8
5. Conclusiones	8

PROBANDO EL LENGUAJE DE PROGRAMACIÓN MATA DE STATA

Luisa Fernanda Riascos Caipe

Resumen

El objetivo de este documento es contrastar la eficiencia el tiempo de ejecución de algoritmos en el lenguaje de programación MATA frente a algunos lenguajes tradicionales. Para realizar lo anterior, se usa la metodología planteada por Aruoba y Fernandez-Villaverde para calcular tiempos de ejecución a través de un modelo neoclásico de crecimiento estocástico con depreciación completa para la obtención de una solución cerrada.

Palabras clave: Lenguajes de programación en economía, Stata, Mata.

Abstract

The main objective of this document is to test the runtime efficiency of the algorithms in Mata programming language against some traditional languages. Therefore, to accomplish the objective mentioned above it is used the methodology proposed by Aruoba and Fernandez-Villaverde to calculate the execution time through a stochastic neoclassical growth model with full depreciation to obtain a closed form solution.

Key words: Programming languages in economics, Stata, Mata.

1. Introducción

Día a día las herramientas computacionales se van convirtiendo en herramientas principales en el mundo de los economistas. De esta manera, cada vez es más frecuente la pregunta ¿cuál es el mejor lenguaje de programación para los economistas?, sin embargo, la respuesta no es única puede variar pues radica en múltiples aspectos a evaluar y que en algunos casos resultan un tanto subjetivos como lo es la facilidad de aprendizaje del lenguaje, de interpretación o de codificación. Pero, también existen aspectos a evaluar que pueden ser objetivos y que pueden presentar un punto de comparación para ponderar ventajas y desventajas de un lenguaje de programación frente a otro, estos aspectos pueden ser la velocidad de ejecución del código y la capacidad de memoria del lenguaje de programación.

Teniendo en cuenta lo anterior, se toma el tiempo de ejecución para hacer una comparación de un aspecto de comparación entre lenguajes que es de fácil medición y que es muy útil a la hora de realizar modelos en economía ya que el tiempo es un recurso escaso y que se debe optimizar a la hora de plantear modelos. De esta manera, en este documento se plantea realizar una comparación entre los tiempos de ejecución del programa MATA de Stata y otros programas tradicionales como Fortran, Java, Matlab y R.

El documento se divide en 5 secciones. La primera la constituye esta introducción, la segunda sección presenta la revisión de la literatura en el tema, la tercera la metodología, la cuarta los resultados, en la quinta sección se presentan las conclusiones y por último la bibliografía.

2. Revisión de la literatura

Existe abundante literatura acerca de lenguajes de programación en general pues son útiles para diversas disciplinas principalmente para la ingeniería. Sin embargo, existe muy poca literatura acerca de lenguajes de programación en economía y el campo de estudio de la economía computacional aun es pequeño pero es creciente.

Una de las primeras aproximaciones a los lenguajes de programación en economía la hacen Kendrick y Amman (1995), ellos realizan un trabajo de análisis de las ventajas comparativas entre los lenguajes que en ese año eran los más usados por los economistas y sugieren que se debe comenzar con algún lenguaje de alto nivel o de modelación, luego ir hacia abajo en la cadena y aprender Fortran, Basic o C. De este modo, plantean una sugerencia de los programas que un economista debe usar dependiendo de su perfil pero no hace una comparación en tiempos de ejecución.

Por otra parte, Cribari-Neto y Jensen (1997) presentan una revisión del software MATLAB y concluyen que el lenguaje combinado con algunos complementos de módulos específicos ofrece un ambiente flexible para la programación y los análisis de datos y sugieren para la fecha que el lenguaje sería más atractivo para los economistas y econométristas si incluyese más funciones de estadística e implementaciones.

Del mismo modo, Cribari-Neto y Zarkos (1999) realizan la evaluación de dos programas: R y S-PLUS y a través de una simulación concluyen que R y S-PLUS son más o menos iguales en eficiencia cuando el código es vectorizado, pero R es más eficiente que S-PLUS cuando se trata de hacer *loops*. Por lo tanto, para ellos R es una alternativa más llamativa que el otro programa presentado porque en primer lugar es un software libre y en segundo lugar tiende a ser más eficiente en términos de velocidad y de capacidad de memoria. Cribari-Neto (1999) afirma que C es un lenguaje de programación flexible que ha probado ser útil y que puede ser de gran ayuda para cualquier econométrista interesado en la programación numérica y que esté insatisfecho con las limitaciones de velocidad y memoria de lenguajes de alto nivel como GAUSS, MATLAB o S-PLUS.

Cabe resaltar que hasta este punto las comparaciones entre los lenguajes de programación se habían realizado entre dos o tres programas. Sin embargo, Prechelt (2000) realiza una comparación entre 7 lenguajes de programación: C, C++, Java, Perl, Python, Rexx y Tcl. Su comparación investiga aspectos de cada idioma como la duración del programa, la eficiencia en el tiempo de ejecución, el consumo de memoria y la confiabilidad. De esta manera concluyó que para el método de evaluación utilizado, en la fase de inicialización, C y C++ completaron entre 3 a 4 veces más rápido que Java y cerca de 5 a 10 veces más rápido que los lenguajes script (Perl, Python, Rexx y Tcl) y en la fase principal C y C++ corrieron cerca de dos veces más rápido que Java.

Así mismo, Cruz Vieira y Santana de Lelis (2005) comparan tres lenguajes de programación teniendo en cuenta cuatro aspectos: Eficiencia en el tiempo, legibilidad, facilidad de aprendizaje y fiabilidad. Su conclusión acerca del tiempo de ejecución fue que aunque Fortran ha incluido nuevas funciones haciéndolo menos eficiente, aún es más eficiente que C++ y Java si se considera la misma tarea ejecutada por el mismo *hardware*. Sin embargo, en oposición a lo planteado por estos dos autores, Aruoba y Fernandez-Villaverde (2015) encuentran que C++ es más eficiente en tiempo que cualquier otro lenguaje de programación entre ellos Fortran, Java, Julia, Matlab, Python, R y Mathematica.

Dado que no se ha encontrado literatura que relacione el lenguaje de programación Mata con los lenguajes mencionados y que son más frecuentes. Se ve la necesidad de comparar este lenguaje relativamente nuevo y ver si es un lenguaje eficiente en el tiempo de ejecución en comparación con los demás.

3. Metodología

Para poder alcanzar el objetivo propuesto se implementó un algoritmo para el modelo neoclásico de crecimiento estocástico que plantea Fernández-Villaverde y Aruoba (2015). En este modelo un planificador central elige una secuencia de consumo y capital para resolver el problema de maximización al que se enfrenta en los periodos. Calibrando para obtener una solución cerrada se llega a un nivel óptimo de consumo y capital en el estado estacionario.

4. Resultados

Una primera aproximación a la eficiencia del programa muestra que es competitivo dentro de los lenguajes seleccionados. La prueba se realizó en un computador con Windows 10, procesador Intel® Core™ i7 @2.60 GHz, con 4 cores y 16 GB de RAM y se obtuvo un resultado dentro del rango esperado.

5. Conclusiones

El lenguaje de programación MATA es relativamente nuevo comparado con los lenguajes tradicionales. Sin embargo, es de fácil uso, acceso y aprendizaje lo que permite que cada vez más usuarios estén interesados en su aprendizaje. El programa ha ido creciendo en los últimos años pues cada vez más economistas hacen uso de él y se ha convertido en una herramienta competitiva.

6. Bibliografía

- Cribari-Neto, F. (1999). C for econometricians, *Computational Economics*, 14 p. 35-149.
- Cribari-Neto, F. y Jensen, M. J. (1997). MATLAB as an econometric programming environment, *Journal of Applied Econometrics*, 12, p. 735-744.
- Cribari-Neto, F. y Zarkos, S. G. (1999). R: yet another econometric programming Environment, *Journal of Applied Econometrics*, 14, p. 319-329.
- Cruz Vieira, W. y Santana de Lelis, L. H. (2005). Programming languages in economics: A comparison among FORTRAN77, C++and JAVA. *Revista de economia e agronegócio*. 3, p. 1679-1614.
- Kendrick, D. A. y Amman, H. M. (1999). Programming languages in economics, *Computational Economics*, 14, p. 151-181.
- Prechelt, L. (2000). An empirical comparison of seven programming languages. *IEEE Computer*, 33(10), p. 23-29.