

11-8-2020

Introducción a la microeconometría básica con aplicaciones en R

Ramiro Rodríguez Revilla

Universidad de La Salle, Bogotá, rrodriguezr@unisalle.edu.co

Follow this and additional works at: https://ciencia.lasalle.edu.co/edunisalle_administracion-economia

Recommended Citation

Rodríguez Revilla, Ramiro, "Introducción a la microeconometría básica con aplicaciones en R" (2020).
Administración y economía. 24.

https://ciencia.lasalle.edu.co/edunisalle_administracion-economia/24

This Libro is brought to you for free and open access by the Catálogo General at Ciencia Unisalle. It has been accepted for inclusion in Administración y economía by an authorized administrator of Ciencia Unisalle. For more information, please contact ciencia@lasalle.edu.co.

RAMIRO RODRÍGUEZ REVILLA

INTRODUCCIÓN A LA MICROECONOMETRÍA BÁSICA
CON APLICACIONES EN R

INTRODUCCIÓN A LA MICROECONOMETRÍA BÁSICA CON APLICACIONES EN R

RAMIRO RODRÍGUEZ REVILLA

Rodríguez Revilla, Ramiro
Introducción a la microeconomía básica con aplicaciones en R / Ramiro Rodríguez
Revilla. – Bogotá : Ediciones Unisalle, 2020.
80 páginas : gráficas ; 23 cm.

Incluye referencias bibliográficas
ISBN 978-958-5148-46-8 (impreso)
ISBN 978-958-5148-47-5 (ePub)
ISBN 978-958-5148-48-2 (PDF)

1. R (Lenguaje de programación de computadores) 2. Estadística – Programas para
computador 3. Estadística – Métodos gráficos 4. Econometría I. Título

CDD: 519.5 ed.22
CEP-Universidad de La Salle. Dirección de Bibliotecas

Ramiro Rodríguez Revilla
Introducción a la microeconomía básica con aplicaciones en R

Primera edición: Bogotá, D.C., noviembre del 2020
© 2020 Universidad de La Salle

ISBN impreso: 978-958-5148-46-8
ISBN ePub: 978-958-5148-47-5
ISBN PDF: 978-958-5148-48-2

EDICIÓN
Ediciones Unisalle
Cra. 5 n.º 59A-44, Edificio Administrativo, 3.º piso
PBX: (571) 348 8000, extensiones: 1224 y 1226
edicionesunisalle@lasalle.edu.co
<https://ediciones.lasalle.edu.co/>

DIRECCIÓN EDITORIAL
Alfredo Morales Roa

COORDINACIÓN EDITORIAL
Andrea del Pilar Sierra Gómez

CORRECCIÓN DE ESTILO
Sabina Ojeda

DISEÑO DE CARÁTULA
Milton Ruiz

DIAGRAMACIÓN
Andrés Conrado Montoya Acosta

IMPRESIÓN
DGP Editores

Queda prohibida la reproducción total o parcial de este libro por cualquier procedimiento, conforme a lo dispuesto por la ley. Impreso y hecho en Colombia.

Para Aura Sofía

CONTENIDO

Introducción	9
1 INTRODUCCIÓN A R	11
2 ESTADÍSTICA DESCRIPTIVA	19
3 REGRESIÓN SIMPLE	25
4 REGRESIÓN MÚLTIPLE	41
5 VARIABLES CUALITATIVAS INDEPENDIENTES	57
6 SUPUESTOS DE UN MODELO DE REGRESIÓN	63
Referencias	77

INTRODUCCIÓN

La microeconometría utiliza la teoría microeconómica y los métodos econométricos para estimar las relaciones que se establecen entre agentes y factores de producción en los mercados. Los modelos econométricos deben estar sintonizados con la realidad; de nada sirven los modelos más estilizados si no se pueden aplicar a la realidad económica que nos rodea.

La econometría representa para mí una pasión, porque nos da la posibilidad de crecer como economistas y realizar procesos de autoaprendizaje, al estimar nuevas relaciones que de alguna manera expliquen el comportamiento aleatorio de los eventos económicos.

Este texto presenta los conocimientos de un primer curso de econometría en el ámbito de pregrado utilizando solo las estructuras de corte transversal. Se emplean el lenguaje R y el entorno RStudio para mostrar los ejemplos de cada capítulo, así, los estudiantes pueden replicar los comandos y funciones mientras avanzan en su lectura.

Existen muchos *softwares* especializados para obtener estadísticas, estimar, validar y evaluar modelos econométricos, pero pocos son libres como R. A medida que estos programas ganen popularidad será imprescindible aprender a manejar un *software* libre y el más utilizado por la comunidad académica en econometría es R.

Los primeros dos capítulos presentan generalidades sobre el manejo de R (por tanto, son de lectura obligatoria para los que se inician en su aprendizaje). A partir del tercer capítulo se muestran los modelos de regresión simple y múltiple con sus especificaciones e interpretaciones. El quinto y el sexto capítulo se pueden leer sin necesidad de los anteriores, si el lector ya posee conocimientos de econometría básica.

El libro se escribió con \LaTeX y RMarkdown. Este último es un entorno de RStudio para la escritura de reportes e informes técnicos que combina las salidas de R con un procesador de textos. Los animo a escribir sus reportes utilizando estas herramientas, las cuales, sin duda, representan el presente y el futuro de la escritura de textos formales científicos.

Agradezco de manera especial el apoyo y la asistencia de Luisa María Hoyos Franco, quien colaboró con algunas salidas de R para su inserción en el documento final. Asimismo, agradezco a las directivas de la Universidad de La Salle en Bogotá, D. C., Colombia, por la confianza depositada en este proyecto académico, el cual aportará al crecimiento disciplinar de sus estudiantes.

INTRODUCCIÓN A R

1.1. R

R es un lenguaje y entorno de programación de libre distribución orientado a objetos, que sirve para elaborar análisis estadístico y gráfico. Lo escribieron inicialmente Ross Ihaka y Robert Gentleman (el nombre R se originó de las iniciales de estos programadores), pertenecientes al Departamento de Estadística de la Universidad de Auckland (Nueva Zelanda). Estos autores tomaron como base el lenguaje científico S desarrollado por Becker *et al.* (1988) en los laboratorios AT&T Bell, y el lenguaje Scheme desarrollado por Steele y Sussman (1975) (Ihaka y Gentleman, 1996). Desde mediados de 1997 existe un grupo central que tiene acceso a la escritura del código fuente de R. Este código lo enriquecen programadores en todo el mundo, quienes aportan nuevas rutinas y actualizan el contenido original. Estos aportes se colocan en paquetes, los cuales son grupos de nuevas bases de datos, funciones y comandos alrededor de un tipo de análisis en particular.

Al ser un *software* libre, R se distribuye bajo una licencia general pública (GNU General Public License) y está disponible para los sistemas operativos Windows, Mac OS X y Linux.

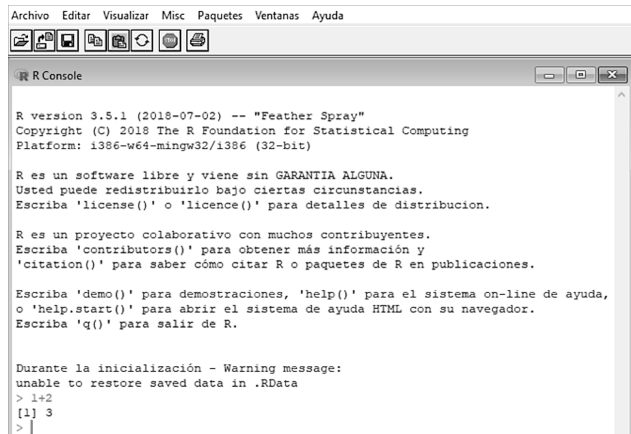
R tiene muchas ventajas sobre otros programas y paquetes para elaborar gráficos, realizar análisis estadístico y estimar modelos econométricos, entre las cuales se pueden enunciar:

- Muchos programas tienen un alto valor comercial, esto los convierte en inaccesibles para un gran sector de la comunidad de usuarios. En cambio, R es libre, lo cual representa un incentivo para que lo usen de forma masiva universidades, centros de investigación y empresas que buscan ahorrar costos en sus desarrollos de *software*.
- R es un lenguaje de programación flexible y orientado a objetos, esto significa que los resultados de funciones se almacenan como un objeto. No todos los resultados se muestran de inmediato, algunos se almacenan internamente para después procesarlos según el algoritmo del usuario. Esta característica lo nutre de flexibilidad: mientras otros programas muestran sus resultados al instante, en R se puede seleccionar lo que se va a mostrar o utilizar en otro proceso posterior.

- Es posible tener en la memoria innumerables bases de datos abiertas para procesarlas, esto solo depende de la cantidad de memoria RAM del equipo.
- R tiene múltiples recursos gráficos para representar complejas estructuras de datos, incluso en tres dimensiones. Con la disponibilidad de mejores resoluciones en el *hardware*, se hace imprescindible utilizar toda la tecnología gráfica.
- Los usuarios con alguna experticia pueden publicar sus propias rutinas y funciones para extender las capacidades básicas del programa. Por esto, la cantidad de paquetes de funciones en R ha crecido de manera exponencial. A la fecha de la edición de este texto, la red Comprehensive R Archive Network (CRAN) tenía más de 16.000 paquetes de rutinas en R¹.
- Las rutinas exigentes en términos de recursos computacionales se pueden programar en C, C++ o Fortran, dado que R las interpreta de forma estándar.

1.2. RCONSOLE Y RSTUDIO

RConsole es el entorno a través del cual se ordenan los cálculos y se muestran los resultados (figura 1.1). Pero este entorno de R es poco amigable, por lo que se han desarrollado de manera gratuita editores de código R denominados *graphical user interface* (GUI), como RCommander y RStudio.



```

Archivo  Editar  Visualizar  Misc  Paquetes  Ventanas  Ayuda
R Console

R version 3.5.1 (2018-07-02) -- "Feather Spray"
Copyright (C) 2018 The R Foundation for Statistical Computing
Platform: i386-w64-mingw32/i386 (32-bit)

R es un software libre y viene sin GARANTIA ALGUNA.
Usted puede redistribuirlo bajo ciertas circunstancias.
Escriba 'license()' o 'licence()' para detalles de distribucion.

R es un proyecto colaborativo con muchos contribuyentes.
Escriba 'contributors()' para obtener más información y
'citation()' para saber cómo citar R o paquetes de R en publicaciones.

Escriba 'demo()' para demostraciones, 'help()' para el sistema on-line de ayuda,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.

Durante la inicialización - Warning message:
unable to restore saved data in .RData
> 1+2
[1] 3
>

```

Figura 1.1. Pantalla de RConsole

Fuente: el autor.

¹ Véase METACRAN (s. f.).

RStudio es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para R que incluye una consola, un editor de comandos y funciones que admite la ejecución directa de código, así como herramientas para trazar, mantener un historial de comandos, depurar y administrar el espacio de trabajo. De estos dos editores, RStudio es el más utilizado por la comunidad de usuarios (figura 1.2). Por esta razón, en este texto se mostrarán sus salidas. En la forma interactiva, los comandos se deben digitar en la ventana *Console*.

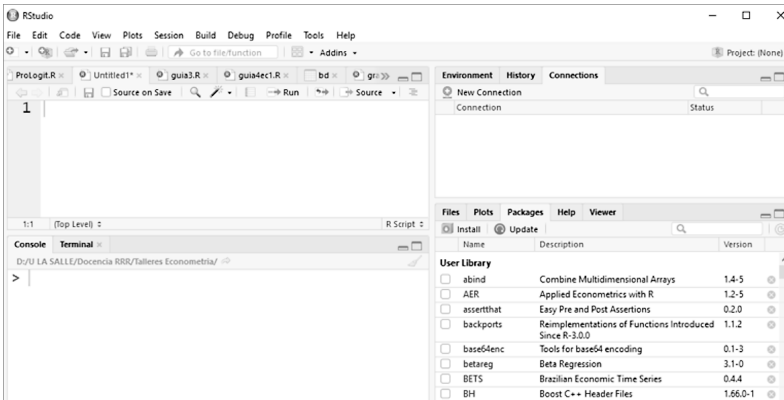


Figura 1.2. Pantalla de RStudio

Fuente: el autor.

1.3. INSTALACIÓN DE R Y PAQUETES

El *software* R se puede descargar libremente de la página oficial <https://www.r-project.org/>. A continuación, se debe seleccionar el servidor para descargar el programa ejecutable y el sistema operativo para utilizar entre Windows, Mac OS X y Linux. Luego, hay que instalar RStudio, el cual se descarga de manera libre de la página <https://www.rstudio.com/>.

Además del entorno básico, es posible que se necesite la instalación de uno o más paquetes² para elaborar distintos tipos de análisis estadísticos o econométricos. En la tabla 1.1 se observan algunos de los más utilizados y el tipo de análisis o proceso que permiten elaborar. Algunas acciones las puede realizar más de un paquete.

² Según METACRAN (s. f.), a la fecha de elaboración de este texto existían más de 16.000 paquetes activos en R.

Tabla 1.1. Paquetes más utilizados en R para aplicaciones econométricas

#	Paquete	Descripción
1	<i>AER</i> *	Variables instrumentales y ejemplos de bases de datos
2	<i>car</i>	Pruebas de regresiones y errores estándar robustos
3	<i>descr</i>	Estadísticas descriptivas
4	<i>ggplot2</i>	Gráficos avanzados
5	<i>gmodels</i>	Herramientas estadísticas
6	<i>haven</i>	Carga y graba datos de otros programas
7	<i>Hmisc</i>	Exporta a L ^A T _E X
8	<i>ivpack</i>	Variables instrumentales
9	<i>lmtest</i>	Pruebas Breusch-Pagan y Breusch-Godfrey
10	<i>knitr</i>	Elaboración de reportes dinámicos, por ejemplo: .pdf
11	<i>mfx</i>	Efectos marginales
12	<i>micEcon</i>	Herramientas para análisis microeconómico
13	<i>normtest</i>	Pruebas de normalidad
14	<i>quantreg</i>	Regresiones cuantílicas (<i>quantile regressions</i>)
15	<i>readxl</i>	Carga archivos de Excel
16	<i>rmarkdown</i>	Convierte documentos Markdown a .html, .pdf y .doc
17	<i>sandwich</i>	Errores estándar robustos
18	<i>stargazer</i>	Salidas de regresiones en formato estándar

* El paquete *AER* se creó con el libro del mismo nombre de Kleiber y Zeileis (2008). Las tablas de este texto se elaboraron con una herramienta gratuita en <https://www.rbloggers.com/creating-nice-tables-using-r-markdown/>

Fuente: el autor.

Para instalar un paquete es necesario escribir un comando; por ejemplo, con el paquete *knitr* es:

```
install.packages("knitr")
```

Para ejecutar las funciones y los comandos o para utilizar las bases de datos de un paquete particular es necesario cargarlo con el comando:

```
library(knitr)
```

1.4. OPERADORES

R considera los operadores aritméticos, de comparación y lógicos, mostrados en la tabla 1.2. Para asignar un valor a una variable se pueden utilizar indistintamente los símbolos (<- , =).

Tabla 1.2. Operadores en R

Aritméticos		De comparación		Lógicos	
+	Suma	==	Igual	!	Negación
-	Resta	!=	Distinto	&	Y lógico
*	Multiplicación	>	Mayor		O lógico
/	División	>=	Mayor o igual	xor	O exclusivo
^ o **	Potencia	<	Menor		
%%	Residuo o resto	<=	Menor o igual		
%* %	Producto de matrices				

Fuente: el autor.

1.5. OBJETOS

R procesa distintos tipos de objetos como números, textos, vectores, matrices, arreglos, bases de datos, listas, fechas, funciones, clase s3, clase s4. Aquí se describirá la forma de crear algunos objetos y su procesamiento³. En general, el contenido de cualquier objeto se puede visualizar escribiendo solo el nombre del objeto.

A. Números

Los números se almacenan en variables. Por ejemplo, a la variable a se le asigna el valor 10. Ahora se muestra el valor de a:

```
a = 10
a
[1] 10
```

³ Dado el carácter introductorio de este texto, el resto de los objetos se pueden consultar en Heiss (2016).

B. Textos

Los textos se deben escribir entre comillas:

```
b = "Texto de prueba"
b
[1] "Texto de prueba"
```

C. Vectores

Los vectores en R se tratan por defecto como filas. Para crear un vector se debe usar la función `c()`. Para hacer referencia a un elemento particular de un vector hay que escribir el nombre del vector seguido de corchetes (`[]`), incluyendo el número del elemento dentro de estos:

```
vectorc = c(2, 3, 4, 5, 6)
vectorc
[1] 2 3 4 5 6
```

```
vectora = c(5)
vectora
[1] 5
```

```
vectord = c(2, 3, "texto")
vectord[3]
[1] "texto"
```

D. Bases de datos

Las bases de datos son conjuntos de vectores columna que contienen datos. Cada columna debe tener la misma cantidad de elementos. Aquí se presenta como ejemplo la base de datos "CPS1985" del paquete *AER*. El comando que permite cargarla es `data()`. Para hacer referencia al vector columna denominado `wage` de la base de datos CPS1985 se utiliza el operador `$`:

```
library(AER)
data("CPS1985")
CPS1985$wage
```

E. Listas

Las listas son colecciones que combinan varios tipos de objetos; esto las diferencia de los vectores, que contienen solo un tipo. En el ejemplo se construirá una lista con el comando `list()` y se incluirá el número representado por la variable `a`, el vector `vectorc` y la base de datos CPS1985. Solo se muestran los cuatro primeros elementos de esta base de datos:

```
lista = list(numero = a, vector = vectorc, base = CPS1985)
lista$numero
[1] 10
lista$vector
[1] 2 3 4 5 6
```

	base	wage	education	experience	age	ethnicity	region	gender
	1	5,10	8	21	35	hispanic	other	female
1100	4,95	9	42	57	cauc	other	female	
2	6,67	12	1	19	cauc	other	male	
3	4,00	12	4	22	cauc	other	male	

Los nombres de la lista creada se pueden visualizar con el comando `names()`:

```
names(lista)
[1] "numero" "vector" "base"
```

Al igual que las columnas de la base de datos, los elementos de la lista se referencian con el operador `$`:

```
lista$vector
[1] 2 3 4 5 6
```

1.6. MANEJO INTERACTIVO Y “SCRIPTS”

El manejo interactivo en RStudio se produce cuando se escriben comandos en la ventana *Console*, después del símbolo `>`; luego se tiene que presionar `Enter` para visualizar su resultado⁴ en la ventana. Si se desea limpiar el contenido de la ventana *Console*, se debe presionar al mismo tiempo `CTRL + L`.

Un *script* es un conjunto de comandos y funciones, los cuales se pueden ejecutar uno por uno o en bloque. Para crear un *script* se debe ir al menú *File*→*New File*→*R Script* y digitar todos los comandos. Si se quiere ejecutar una línea que contenga un comando, se debe presionar el botón `Run` en la parte superior o `Ctrl + Enter` (para ejecutar varias líneas de comandos primero se deben seleccionar). En los *scripts* se pueden utilizar líneas de comentarios, los cuales deben ir precedidos del símbolo `#`. Los comentarios se presentarán en color verde.

1.7. RUTAS DE CARPETAS

En R es importante definir la ruta de la carpeta de trabajo para que los archivos de salida se guarden en esa ubicación; esto permite tener un control efectivo de los archivos que se cargarán en el entorno.

⁴ No todos los mensajes en R de color rojo son errores, algunos son advertencias.

Para definir la ruta hay que utilizar el comando `setwd()`. Por ejemplo:

```
setwd("D:/Carpeta de trabajo/Econometria/")
```

Se puede digitar el comando `getwd()` para recuperar la carpeta de trabajo previamente establecida por el comando `setwd()`:

```
getwd()
```

1.8. RMARKDOWN

Este es un entorno dentro de RStudio para combinar código de comandos, gráficos y funciones de R (denominados *chunk*), y un procesador de textos. No solo acepta código de R, sino también de Python, SQL, Bash, Rcpp, Stan, JavaScript y CSS. Tiene una amplia variedad de formatos de salida como documentos de Word o pdf y html. También es posible elaborar presentaciones del tipo Power Point o Beamer (presentación .pdf con \LaTeX). Este texto se escribió totalmente en RMarkdown y \LaTeX . Su página web es <https://rmarkdown.rstudio.com/>