

Curso de R y estadística básica

[Felipe de J. Muñoz González]

fmunoz@lcg.unam.mx

Introducción

[Descargar Presentación](#)

¿Qué es y por que debería usarlo?

¿Qué es R?



R es un lenguaje de programación para la estadística y el modelado de datos.

R tiene las siguientes características:

- Es elegante y versatil
- Sintaxis diseñada para trabajar con datos
- Capacidades gráficas altamente potentes
- Facil y eficiente manipulación de datos
- Contiene herramientas enfocadas a las areas biológicas.

** R es lo que requieres para investigar*

¿Dónde comienzo?

Instalación

- Windows

Lo primero es identificar donde descargamos R para instalarlo:

<https://cran.r-project.org/bin/windows/base/>

R-3.3.2 for Windows (32/64 bit)

[Download R 3.3.2 for Windows](#) (62 megabytes, 32/64 bit)

[Installation and other instructions](#)

[New features in this version](#)

If you want to double-check that the package you have downloaded exactly matches the package distributed by R, you can compare the [md5sum](#) of the .exe to the [true fingerprint](#). You will need a version of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

Frequently asked questions

- [Does R run under my version of Windows?](#)
- [How do I update packages in my previous version of R?](#)
- [Should I run 32-bit or 64-bit R?](#)

Please see the [R FAQ](#) for general information about R and the [R Windows FAQ](#) for Windows-specific information.

Other builds

- Patches to this release are incorporated in the [r-patched snapshot build](#).
- A build of the development version (which will eventually become the next major release of R) is available in the [r-devel snapshot build](#).
- [Previous releases](#)

Note to webmasters: A stable link which will redirect to the current Windows binary release is [~CRANMIRROR~bin/windows/base/release.htm](#).

Last change: 2016-10-31, by Duncan Murdoch

Instalación

- MAC

Lo primero es identificar donde descargamos R para instalarlo:

<https://cran.r-project.org/bin/macosx/>

R for Mac OS X

This directory contains binaries for a base distribution and packages to run on Mac OS X (release 10.6 and above). Mac OS 8.6 to 9.2 (and Mac OS X 10.1) are no longer supported but you can find the last supported release of R for these systems (which is R 1.7.1) [here](#). Releases for old Mac OS X systems (through Mac OS X 10.5) and PowerPC Macs can be found in the [old](#) directory.

Note: CRAN does not have Mac OS X systems and cannot check these binaries for viruses. Although we take precautions when assembling binaries, please use the normal precautions with downloaded executables.

As of 2016/03/01 package binaries for R versions older than 2.12.0 are only available from the [CRAN archive](#) so users of such versions should adjust the CRAN mirror setting accordingly.

R 3.3.2 "Sincere Pumpkin Patch" released on 2016/10/31

Please check the MD5 checksum of the downloaded image to ensure that it has not been tampered with or corrupted during the mirroring process. For example type

md5 R-3.3.2.pkg

in the *Terminal* application to print the MD5 checksum for the R-3.3.2.pkg image. On Mac OS X 10.7 and later you can also validate the signature using

pkgutil --check-signature R-3.3.2.pkg

Files:

R-3.3.2.pkg

MD5-hash: 77c2776b755d18c30d09b62843ba
SHA1-hash: 18446e185d31108ec236241ac51ed8b5e27d
(ca. 71MB)

R 3.3.2 binary for Mac OS X 10.9 (Mavericks) and higher, signed package. Contains R 3.3.2 framework, R.app GUI 1.68 in 64-bit for Intel Macs, Tcl/Tk 8.6.0 X11 libraries and Textinfo 5.2. The latter two components are optional and can be omitted when choosing "custom install", it is only needed if you want to use the tcltk R package or build package documentation from sources.

Note: the use of X11 (including tcltk) requires [XQuartz](#) to be installed since it is no longer part of OS X. Always re-install XQuartz when upgrading your OS X to a new major version.

R-3.2.1-snowleopard.pkg

MD5-hash: 58fe0b13148cb75f80b0c9b914d65
SHA1-hash: b6ef61db12b22a224f8cb91c7cf09033cc0b0
(ca. 68MB)

R 3.2.1 legacy binary for Mac OS X 10.6 (Snow Leopard) - 10.8 (Mountain Lion), signed package. Contains R 3.2.1 framework, R.app GUI 1.66 in 64-bit for Intel Macs. This package contains the R framework, 64-bit GUI (R.app), Tcl/Tk 8.6.0 X11 libraries and Textinfo 5.2. GNU Fortran is **NOT** included (needed if you want to compile packages from sources that contain FORTRAN code) please see [the tools directory](#). NOTE: the binary support for OS X before Mavericks is being phased out, we do not expect further releases!

[Mac-GUI-1.68.tar.gz](#)

Sources for the R.app GUI 1.68 for Mac OS X. This file is only needed if you want to

Instalación

- Linux

Abrimos una terminal de linux (Ctrl + Alt + T) y dentro de esta, dependiendo del sistema operativo:



```
$ sudo apt-get install r-base
```



```
$ su -c 'yum install R'
```



```
$ sudo pacman -S r
```

¿Cómo funciona?

- Ejecutar desde el cmd/terminal

- Entornos graficos

Desde Windows:

- Opcion A:
Inicio > Simbolo del sistema
- Opcion B:
Buscar > CMD

Desde MAC/Linux:

1. Se abre la terminal

Se ejecuta:

```
$ R
```


¿Cómo funciona?

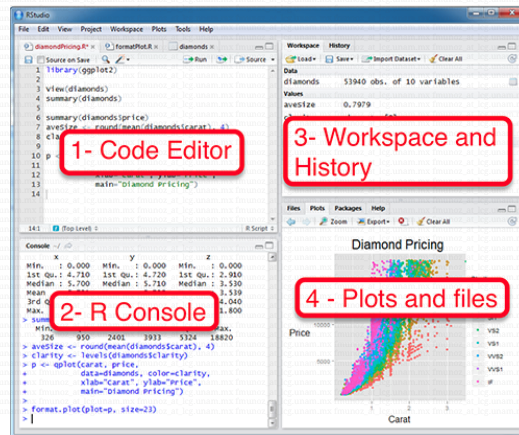
- Entornos graficos

Rstudio

Ambiente gráfico integrado, se basa en diversos compartimentos:

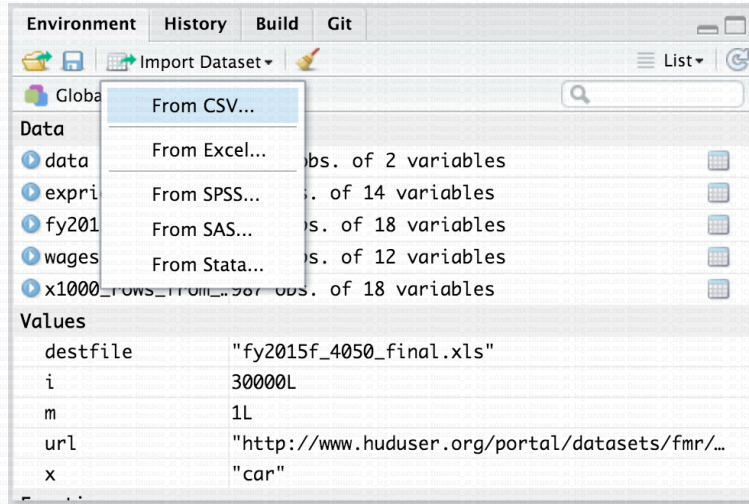
- Consola para editar código
- Ventana de datos e historial
- Ventana de la Consola
- Ventana de gráficas y archivos

Permite importar y ver los datos de una manera gráfica





Importando desde el ambiente grafico



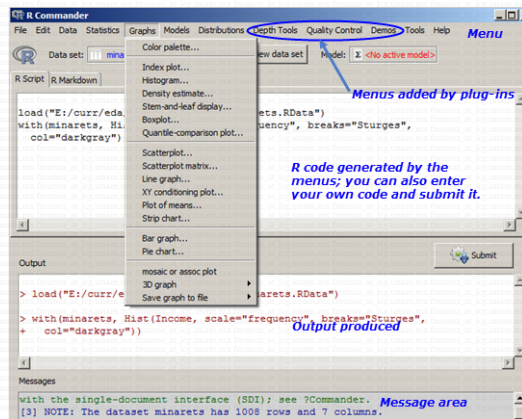
¿Cómo funciona?

- Entornos graficos

R commander (Rcmdr)

Es una interfaz gráfica que cuenta con botones y menus extensos, las características son

- Contiene códigos precargados (SPSS, SAS o Stata)
- No provee acceso directo a la línea de comandos de R
- No es enriquecido gráficamente, contiene 3 paneles:
 - Ventana del script (código ejecutándose)
 - Ventana de Salida (Imprime los resultados)
 - Ventana de Mensajes (Errores/advertencias/notas)





Utilizando las herramientas predefinidas

The screenshot shows the R Commander application window. The 'Data set' dropdown is set to 'd'. The 'Script Window' contains the following R code:

```
y<-rnorm(100)
x<-rnorm(100)
data1<-data.frame(x,y)
attach(data1)
```

The 'Output Window' shows the execution of these commands:

```
> y<-rnorm(100)
> x<-rnorm(100)
> data1<-data.frame(x,y)
> attach(data1)
> Dataset <- edit(as.data.frame(NULL))
```

The 'Messages' window at the bottom displays an error:

```
ERROR: empty data set.
NOTE: The dataset data1 has 100 rows and 2 columns.
```

Annotations highlight key features:

- Variable selection window, allows choosing from list:** Points to the 'Statistics' menu, which is open, showing options like Summaries, Contingency tables, Means, Proportions, Variances, Nonparametric tests, Dimensional analysis, and Fit models.
- Sensible defaults with the option of customization for expert users:** Points to the 'Model:' dropdown, which is currently set to '<No active model>'. Below it, a list of models is visible, including Linear regression..., Linear model..., Generalized linear model..., Multinomial logit model..., and Proportional-odds logit model.
- Checkboxes for access to common options for a statistical test:** Points to the 'One-Way Analysis of Variance' dialog box. It shows 'Enter name for model: AnovaModel.1', 'Groups (pick one): CLASS_DESCR', and 'Response Variable (pick one): TOTAL_COST'. The 'Pairwise comparisons of means' checkbox is checked.

¿Cuál usar y como instalarlo?

Comparación

RStudio

- Provee acceso directo al código en R.
- Uso para proyectos que requieren interacción directa con el código o manipulación de datos compleja

Rcmdr

- Simple y amable para el usuario sobre todo en análisis estadísticos y diagnósticos.
- Uso para análisis tradicionales, datos convencionales y tests estadísticos.

NOTA: Es posible ejecutar Rcmdr desde R-Studio.

Instalación de R-Studio

RStudio tiene diferentes versiones:

- Version gratis para escritorio
- Version de paga para escritorio
- Version gratis para servidor
- Version pro para servidor

Para descargarlo entramos a

<https://www.rstudio.com/products/rstudio/download/>

RStudio Desktop 1.0.136 — Release Notes

RStudio requires R 2.11.1+. If you don't already have R, download it [here](#).

Installers for Supported Platforms

Installers

	Size	Date	MD5
RStudio 1.0.136 - Windows Vista/7/8/10	81.9 MB	2016-12-21	93b3f307f567c33f7a4db4c114099b3e
RStudio 1.0.136 - Mac OS X 10.6+ (64-bit)	71.2 MB	2016-12-21	12d6d6ade0203a2fce6fe3dea65c1ae
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (32-bit)	85.5 MB	2016-12-21	0a20fb89d8aeb39b329a640ddadd2c5
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (64-bit)	92.1 MB	2016-12-21	2a73b88a12a9fba7f96251cec78b41340
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	84.7 MB	2016-12-21	fa6179a7855bf6f9f939a34c169da45f1d
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	85.7 MB	2016-12-21	2b3a140ded380bf0704e58496be6b55545

Zip/Tarballs

Zip/tar archives

	Size	Date	NDS
RStudio 1.0.136 - Windows Vista/7/8/10	1117.5 MB	2016-12-21	f415939bf5012c0ab127c7c7fbc9608be
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (32-bit)	86.2 MB	2016-12-21	fca75f953dd425694b7fd4335bd29165
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (64-bit)	93.2 MB	2016-12-21	7cf0092653aa44fc76325a8f1325fb1f
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	85.4 MB	2016-12-21	30c89299d30ec03b38098e51e9bf49b8
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	86.6 MB	2016-12-21	ea2a262f650e92f568f48edc1c093902

Source Code

A tarball containing source code for RStudio v1.0.136 can be downloaded from [here](#)

Instalación de Rcmdr



Ejecutamos R

```
[usuario@equipo ~]$ R
```

Instalamos el paquete de Rcmdr

```
> install.packages("Rcmdr",dependencies=TRUE)
```

Seguimos las instrucciones de la salida

```
Aviso en install.packages("Rcmdr", dependencies = TRUE) :  
'lib = "/usr/lib/R/library"' is not writable  
Would you like to use a personal library instead? (y/n) y  
Would you like to create a personal library  
~/R/x86_64-unknown-linux-gnu-library/2.15  
to install packages into? (y/n) y
```

Se abrirá una ventana para seleccionar el repositorio de dónde descargar los paquetes necesarios. Seleccionamos el que queramos y después de aceptar empezará a descargar los paquetes.

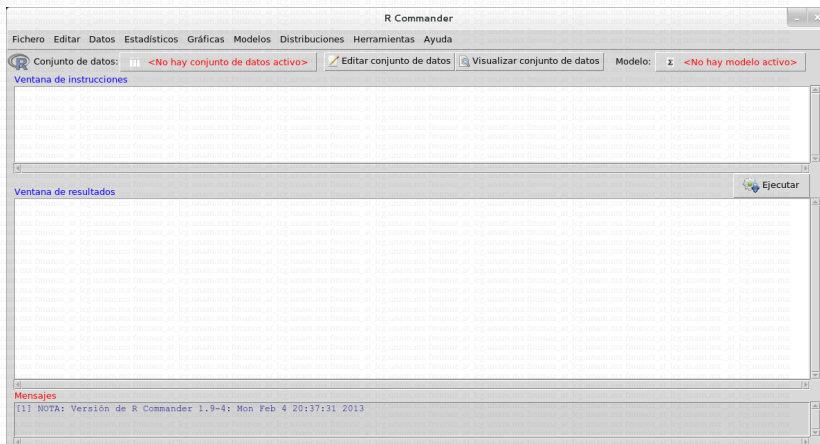
Ejecutar Rcmdr

Ejecutamos R

```
[usuario@equipo ~]$ R
```

Cargamos la libreria de Rcmdr

```
> library("Rcmdr")
```



Mis primeros pasos en R



Sintaxis

- Aritmetica

> 1+2

> 6*9/3 #6 multiplicado por nueve y dividido entre 3

> 2**3 #2 elevado al exponente 3

> 2^8 #2 elevado al exponente 8

Sintaxis

- Aritmetica

```
> options(digits =16) #ver mas digitos
> 10/3
```

```
> pi #numeros irracionales
```

```
> exp(1)
```

- Texto

21 / 43

Sintaxis en R

- Aritmetica

- Texto

- Operaciones Logicas

> 3 > 4

> 2+2 == 5

> T == TRUE

Sintaxis en R

- Aritmetica

```
> sqrt(-1)
```

- Texto

- Operaciones Logicas

```
> sqrt(-1+0i)
```

- No definidos

```
> sqrt(as.complex(-1))
```

```
> 0+1i
```

NaN Not a Number

Variables

Variables

- Definición

Definición de una variable

Estructura de datos que pueden cambiar de contenido a lo largo de la ejecución de un programa.

Una variable requiere

- Un espacio en el sistema de almacenaje
- Un identificador (nombre) asociado a ese espacio
- Cantidad o informacion ya sea conocida o desconocida (Valor)

Variables

- Definición

- Tipos

Tipos de Variables:

- **Tipo de dato lógico.**
 - TRUE/FALSE, T/F
- **Tipo de dato numerico entero**
 - 0, 1, -2, 3, 10, ...
- **Tipo de dato numerico doble**
 - 12.1, 200, 5.2E10, 0.1232, pi, e, sqrt(2)
- **Tipo de dato cadena**
 - "Esto es texto", "Se pueden usar simbolos,-.\$%&()/="

Variables

- Definición

- Tipos

- Creación

Crea una variable que tenga el valor 42 que se llame var_1

```
> var_1 <- 42 # o var_1 = 42 pero...
```

Divide esa variable entre 2

```
> var_1/2
```

Crea una cadena de texto y ponla en la variable txt

```
> txt<- "Arrg, Soy un pirata"
```

Asigna un valor de VERDADERO a la variable logic

```
> logic <- TRUE
```

Variables

- Definición

- Tipos

- Creación

Imprime los valores de txt, logic y de var_1

```
> txt
```

```
> logic
```

```
> var_1
```

Variables

- Definición

- Tipos

- Creación

Los nombres de las variables solo pueden tener:

- [a-zA-Z]+
- [0-9]+
- [_.]+

Que cosas **NO** pueden tener:

- Numeros para iniciar
- Simbolos seguidos de numeros (Ej. A+3)
- Solamente numeros
- Solamente simbolos

Variables

- Definición

- [a-zA-Z]+

- Tipos

- [0-9]+

- Creación

- [._]+

Que cosas **NO** pueden tener:

- Numeros para iniciar
- Simbolos seguidos de numeros (Ej. A+3)
- Solamente numeros
- Solamente simbolos

Vectores

- Introducción a vectores

Unidad básica de variable en R.

Introducir 74,31,95,61,76,34,23,54,96 en R

1. Funcion c

```
x<-c(74,31,95,61,76,34,23,54,96) #Numerico
```

```
x<-c("74",31,95,61,76,34,23,54,96) #String
```

1. Funcion scan

```
x<-scan() # empty to end
```

Vectores

- Introducción a vectores

1. Repeated data; Regular patterns

```
> seq(from = 1, to = 5)
```

```
> seq(from = 2, by = -0.1, length.out = 4)
```

```
> 1:5
```


Vectores

- Introducción a vectores

- Indexación

```
> x[1]
```

```
> x[c(1, 3, 4, 8)]
```

```
> x[-c(1, 3, 4, 8)]
```

```
>LETTERS[1:5]
```

```
letters[-(6:24)]
```

Funciones

- Definición

Definición:

Subrutina o subprograma (también llamada procedimiento, función o rutina), que se presenta como un subalgoritmo que forma parte del algoritmo principal, el cual permite resolver una tarea específica.

Funciones

- Definición

Definición:

Subrutina o subprograma (también llamada procedimiento, función o rutina), que se presenta como un subalgoritmo que forma parte del algoritmo principal, el cual permite resolver una tarea específica.

Funciones

- Definición

Definición:

Subrutina o subprograma (también llamada procedimiento, función o rutina), que se presenta como un subalgoritmo que forma parte del algoritmo principal, el cual permite resolver una tarea específica.

Funciones

- Definición
- Ejemplos

```
> sum(1,2,6,7,9)
```

Algunas funciones requieren argumentos los cuales tienen nombres determinados

```
> rep("Yo ho! un pirata soy", times=3)
```

Funciones

- Definición

- Ejemplos

- Help, example

De todas las funciones, la que debemos de considerar como básica es **help**

```
> help (sum)
```

```
> ?sum
```

Otra es la función **example**

```
> example(min)
```

Ejercicio: Obten el help de la función **rep**

Funciones

- Definición

- Ejemplos

- Help, example

-Codigo

1. Escribe el nombre de una función sin parentesis

```
> intersect
```

1. UseMethod

```
> rev
```

```
> methods(rev)
```

```
> rev.default
```

Obtener ayuda - Interna

- **Consola: Shortcuts (Ej. Ctrl+L)**

- **FAQ on R:**

- **FAQ on R for Windows:**

- **R Manuals:**

- **R functions (text):**

- **HTML Help:**

- **Search help: `help.seaerch("plot") // ??`**

- **`search.r-project.org`**

- **`Apropos ?apropos`**

Obtener ayuda

- Interna

- Externa

- **The R Project for Statistical Computing:** <http://www.r-project.org/>
 - **The Comprehensive R Archive Network:** <http://cran.r-project.org/>
 - **R-Forge:** (<http://r-forge.r-project.org/>)
 - **R Wiki:** <http://wiki.r-project.org/rwiki/doku.php>
 - **Other:** R Graph Gallery (<http://addictedtor.free.fr/graphiques/>) y R Graphical Manual (<http://bm2.genes.nig.ac.jp/RGM2/index.php>). RSeek (<http://www.rseek.org>)

Obtener ayuda

- Interna

- Externa

- Otros tips

• Up Arrow Para comandos utilizados

◦ objects() o ls() Variables actuales

◦ remove(var1,var2,var3) o rm(var1,var2) Borrar variables

◦ rm(list=ls()) Borrar todo

Archivos

- Ver archivos

Con el fin de leer una carpeta dentro del sistema podemos utilizar la función **list.files()**

```
> list.files()
```

Archivos

- Ver archivos

- Scripts R

Cuando se programa en R, se generan documentos llamados **Scripts**, los cuales en general por convención finalizan en .R y pueden ser ejecutados en el momento que se desee.

Si se quiere ejecutar un codigo determinado desde la terminal se utiliza la función **source**:

```
> source("bottle1.R")
```

That's all folks (for now)!

Slideshow created using **remark**.