

Curso de R y microarreglos

[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 (imagenes vectorizadas)
- Facil y eficiente manipulación de datos (Uso de recursos computacionales)
- Contiene herramientas enfocadas a las areas biológicas. (CRAN 12,548 Bioconductor 1477+909+326 Total 15,260)

** R es lo que requieres para ciencias biológicas*

¿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 <https://CRAN.MIRROR>/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/
```

Instalación

- Linux

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

 Ubuntu

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

 Fedora

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

 Arch Linux

```
$ 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?

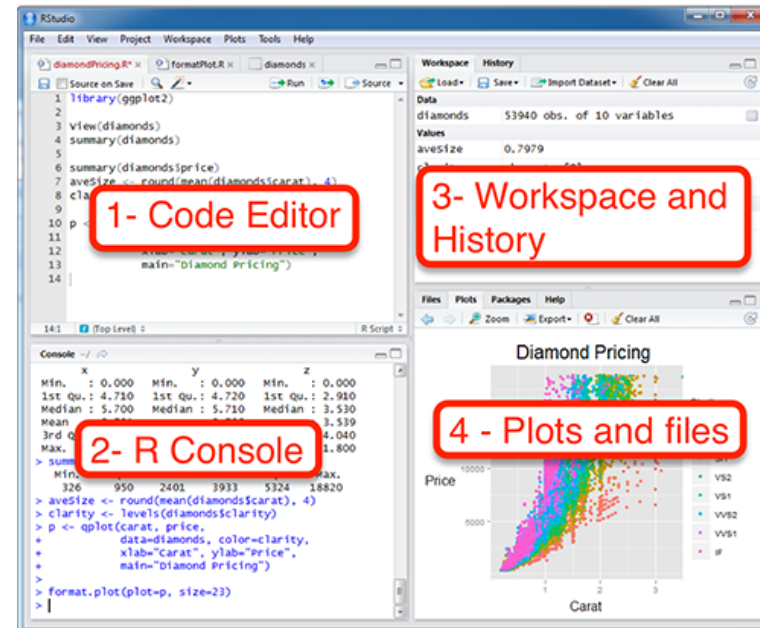
- Entornos graficos

Rstudio

Ambiente gráfico integrado, se basa en diversos compartimentos:

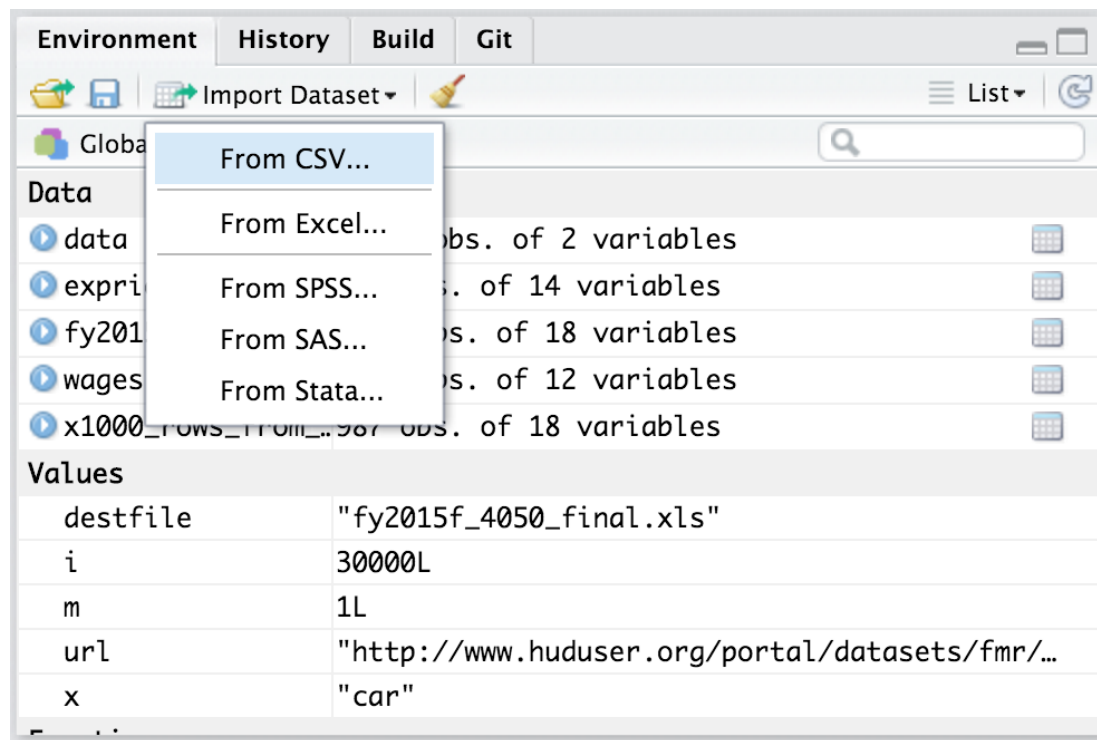
- Consola para editar código
- Ventana de datos e historial
- Ventana de la Consola
- Ventana de gráficas y archivos

Permite importar y ver los datos de una manera gráfica





Importando desde el ambiente grafico



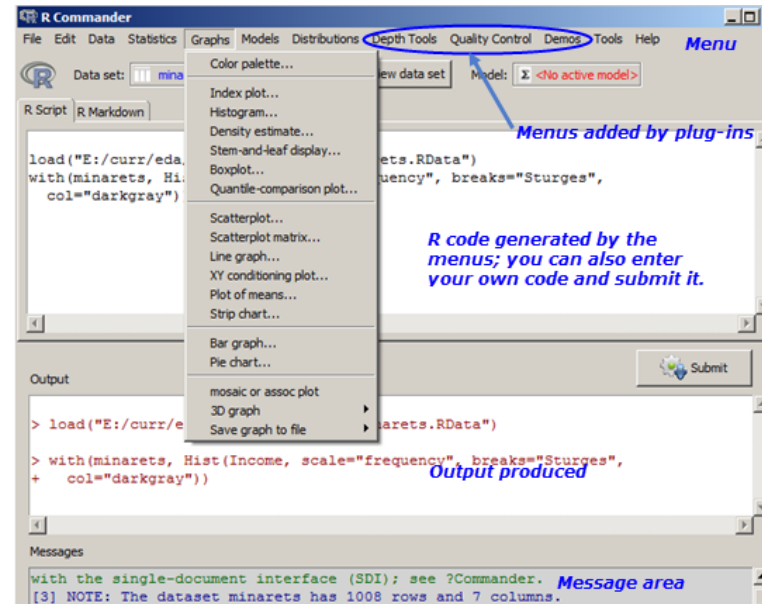
¿Cómo funciona?

- Entornos graficos

R commander (Rcmdr)

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

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





Utilizando las herramientas predefinidas

The screenshot shows the R Commander interface. The 'Statistics' menu is open, displaying options like Summaries, Contingency tables, Means, Proportions, Variances, Nonparametric tests, Dimensional analysis, and Fit models. The 'Fit models' submenu is also open, showing Linear regression..., Linear model..., Generalized linear model..., Multinomial logit model..., and Proportional-odds logit model... The 'Submit' button is visible. The Script Window contains the following code:

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

The Output Window shows the following messages:

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

The Messages window shows the following error message:

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

The 'One-Way Analysis of Variance' dialog box is open, showing the following options:

- Enter name for model: AnovaModel1
- Groups (pick one): CLASS_DESCR, DEPT_DESCR, GENDER, LOCATION_DESCR
- Response Variable (pick one): CostSum, DEPTID, FUND_CODE, TOTAL_COST
- Pairwise comparisons of means: ☒

Four callouts highlight specific features:

- Variable selection window, allows choosing from list
- Checkboxes for access to common options for a statistical test
- Sensible defaults with the option of customization for expert users

¿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	12d6d6ade0203a2fcef6fe3dea65c1ae
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (32-bit)	85.5 MB	2016-12-21	0a20fb89d8aaeb39b329a640ddadd2c5
RStudio 1.0.136 - Ubuntu 12.04+/Debian 8+ (64-bit)	92.1 MB	2016-12-21	2a73b88a12a9fbaf96251cecf8b41340
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (32-bit)	84.7 MB	2016-12-21	fa6179a7855bff0f939a34c169da45fd
RStudio 1.0.136 - Fedora 19+/RedHat 7+/openSUSE 13.1+ (64-bit)	85.7 MB	2016-12-21	2b3a148ded380b704e58496befb55545

Zip/Tarballs

Zip/tar archives	Size	Date	MD5
RStudio 1.0.136 - Windows Vista/7/8/10	117.5 MB	2016-12-21	f415939bf5012c0ab127c7cfbc9600be
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.

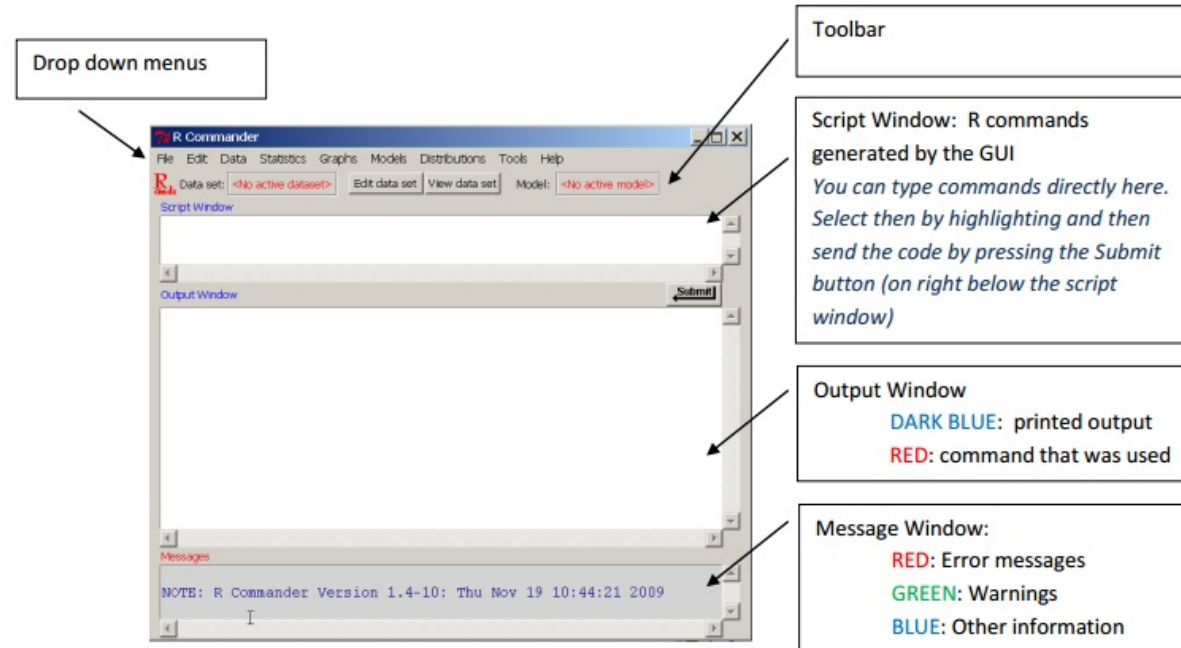
Ejecutar Rcmdr

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 por nueve añadir 3
```

```
> 2**3 #2 elevado al exponente 3  
> 2^8 #2 elevado al exponente 8
```

```
> log(exp(1)) #2 elevado al exponente 8  
> log10(1000)
```

Sintaxis

- Aritmetica

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

```
> pi #numeros irracionales
```

```
> exp(1)
```

Sintaxis

- Aritmetica

- Texto

```
> "Esto es una cadena de texto!"
```

```
> palabras<-c("Esto", "es", "un", "conjunto", "de", "palabras", "en", "R")
```

Sintaxis en R

- Aritmetica

- Texto

- Operaciones
Logicas

```
> 3 > 4 #¿Es 3 mayor que 4?
```

```
> 2+2 == 5 #¿2 más dos es igual a 5?
```

```
> T == TRUE #¿Es T igual a TRUE?
```

Sintaxis en R

- Aritmetica

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

- Texto

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

- Operaciones
Logicas

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

- No definidos

```
> 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 información ya sea conocida o desconocida (Valor)

Variables

- Definición

- Tipos

Tipos de Variables:

- **Tipo de dato cuantitativo.**
 - Numerico entero (Discreto)
 - 0, 1, -2, 3, 10, ...
 - **Numerico flotante/doble (Continuo)**
 - 12.1, 200, 5.2E10, 0.1232, pi, e, sqrt(2)
- **Tipo de dato cualitativo(categórico).**
 - Tipo de dato de texto

```
> str(state.abb)
> "Se pueden usar simbolos.,-.$%&/()="
```

- **Tipo de dato lógico.**

```
> TRUE/FALSE, T/F
```

- **Datos faltantes.**

```
> NA
> NaN
```

Variables

- Definición

Crea una variable que tenga el valor 42 que se llame var_1 y dividela entre 2

```
> var_1 <- 42  
> var_1 = 42  
> var_1/2
```

- Tipos

Crea una cadena de texto y ponla en la variable txt

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

- Creación

Asigna un valor de VERDADERO a una variable logic

```
> z <- TRUE
```

Datos Faltantes

```
> x <- c(3, 7, NA, 4, 7)  
> y <- c(5, NA, 1, 2, 2)  
> x + y  
> sum(x)  
> sum(x, na.rm = TRUE)  
> is.na(x)
```

Variables

- Definición

- Tipos

- Creación

Imprime los valores de txt, logic y de var_1

```
> print(txt)
```

```
> logic
```

```
> var_1
```

Variables

- Definición

- Tipos

- Creación

Los nombres de las variables solo pueden tener (Nombres significativos):

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

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)
```

```
> x <- seq(from = 2, by = -0.1, length.out = 20)  
> y <- seq(from = 2, to= 100, by = 5)
```

```
> 1:5
```

Vectores

- Introducción a vectores

```
> x[1]
```

- Indexación

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

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

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

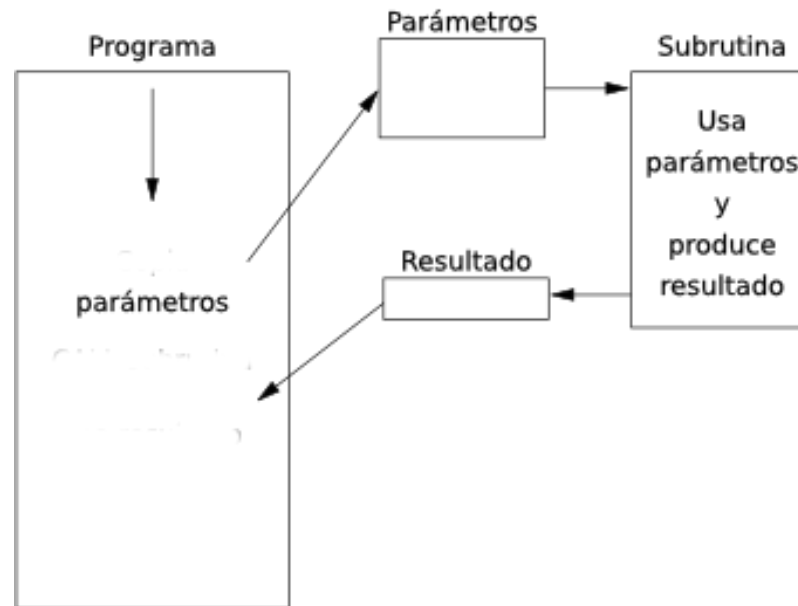
```
letters[-(6:24)]
```


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

- Ejemplos

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

Algunas funciones requieren argumentos los cuales tienen nombres determinados

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

Ejercicio: Obten la raiz de 16 utilizando la función sqrt

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
- Código

Escribe el nombre de una función sin parentesis

```
> intersect
```

UseMethod

```
> rev
```

```
> methods(rev)
```

```
> rev.default
```

```
> wilcox.test  
> methods(wilcox.test)
```

Obtener ayuda

- Interna

- **Console:** Shortcuts (Ej. Ctrl+L)
 - **FAQ on R:** <https://cran.r-project.org/doc/FAQ/R-FAQ.html>
 - **FAQ on R for Windows:** <https://cran.r-project.org/bin/windows/base/rw-FAQ.html>
 - **R Manuals:** <https://cran.r-project.org/manuals.html>
 - **R functions (text):** `help("mean")`, `?plot`
 - **HTML Help:** `help.start()`
 - **Search help:** `help.search("plo")` // ?? plo
 - **search.r-project.org**
 - **Apropos** `?apropos` // `apropos("lm")` // `find("cor")`
 - **R Help Mailing Lists** <http://www.r-project.org/mail.html>

Obtener ayuda

- Interna

- Externa

- **R-Forge:** <http://r-forge.r-project.org/>
 - **R Wiki:** <http://wiki.r-project.org/rwiki/doku.php>
 - **R Graph Gallery:** <http://addictedtor.free.fr/graphiques/>
 - **R Graphical Manual** <http://bm2.genes.nig.ac.jp/RGM2/index.php>
 - **RSeek** (<http://www.rseek.org>)
 - **Google** www.google.com

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 código determinado desde la terminal se utiliza la función **source**:

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

get code: [bottle_code.R](#)

That's all folks (for now)!