

# Econometría Aplicada con



```
R Console (32-bit)
Archivo Editar Misc. Ejecutar Ventanas Ayuda

> x <- c(1,2,3,4,5,6)
> y <- x^2
> print(y)
[1] 1 4 9 16 25 36
> mean(y)
[1] 15.16667
> var(y)
[1] 178.9444
> lm_1 <- lm(y ~ x)
> print(lm_1)

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept) -9.3333
x              7.0000

> summary(lm_1)

Call:
lm(formula = y ~ x)

Coefficients:
(Intercept) -9.3333
x              7.0000

Residuals:
1      2      3      4      5      6
3.3333 -0.6667 -2.6667 -2.6667 -0.6667  3.3333

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -9.3333    2.8441   -3.282 0.030453 *
x              7.0000    0.7303    9.585 0.000662 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.055 on 4 degrees of freedom
Multiple R-squared:  0.9583,    Adjusted R-squared:  0.9478
F-statistic: 91.87 on 1 and 4 DF,    p-value: 0.000662

> |
```



## SESIÓN 1: Introducción a R

## Contenido

|                                       |    |
|---------------------------------------|----|
| Introducción .....                    | 4  |
| ¿Qué es R? .....                      | 5  |
| Usos de R .....                       | 5  |
| Historia de R.....                    | 6  |
| Iniciando una sesión con R .....      | 8  |
| Primera sesión.....                   | 9  |
| Ayuda en R .....                      | 10 |
| R como calculadora .....              | 11 |
| Operadores aritméticos .....          | 11 |
| Operadores lógicos .....              | 12 |
| Operadores de comparación .....       | 12 |
| Funciones matemáticas.....            | 13 |
| Espacio y directorio de trabajo ..... | 15 |
| Paquetes en R.....                    | 16 |
| Instalar paquetes.....                | 16 |
| Cargar librerías.....                 | 17 |
| Bibliografía .....                    | 18 |
| Recursos informáticos .....           | 18 |

## Introducción

En la actualidad existen diversos paquetes estadísticos de paga, como por ejemplo SPSS, Stata, SAS, Minitab, etc., que tratan en la medida de ofrecer la mayor cantidad de técnicas estadísticas posibles. R surge frente a estas opciones como un software libre a fin de que la descarga del mismo está abierta en la red cualquier usuario con acceso a internet puede obtener dicho software.

En la siguiente sesión se explicará las características principales de R, su historia, sus usos y el uso básico del programa; a fin de que el estudiante comprenda la importancia del uso de este lenguaje de programación para la ciencia en general.

## ¿Qué es R?

R es un lenguaje de programación que se creó especialmente para el análisis estadístico. A diferencia de la mayoría de los programas que se usan frecuentemente, este se maneja a través de una consola en donde se indican ordenes por medio de códigos.

La importancia de R radica en que al ser éste un software libre, su descarga está abierta para cualquier usuario que cuente con internet.

### Usos de R

---

El lenguaje de programación R es muy utilizado en la actualidad en diferentes campos de la ciencia, entre los principales podemos distinguir los siguientes:

**Estadística:** Aplicación de técnicas de muestreo, técnicas básicas y avanzadas de estadística (pruebas de hipótesis, análisis de varianzas, etc.).

**Econometría:** Aplicación de técnicas de análisis para datos de corte transversal, de series de tiempo y de panel.

**Ciencias biológicas:** Análisis de modelos de elección discreta para la estimación de probabilidades, análisis de supervivencia, entre otros.

## Historia de R

R es uno de los programas más usados en la actualidad cuando se habla de análisis estadístico y de datos, empresas como Google, Bank of America y Shell lo usan.

R fue inicialmente diseñado por Robert Gentleman y Ross Ihaka en 1992, miembros del departamento de estadística de la Universidad de Auckland, en Nueva Zelanda.

Se basaron en el software propietario llamado S, que es propiedad de Laboratorios Bell, 1996 liberaron el código de R como software libre, es por esta razón que el lenguaje ha ido evolucionando gracias a la colaboración de programadores en todo el mundo, actualmente cientos de miles de personas usan dicho software.

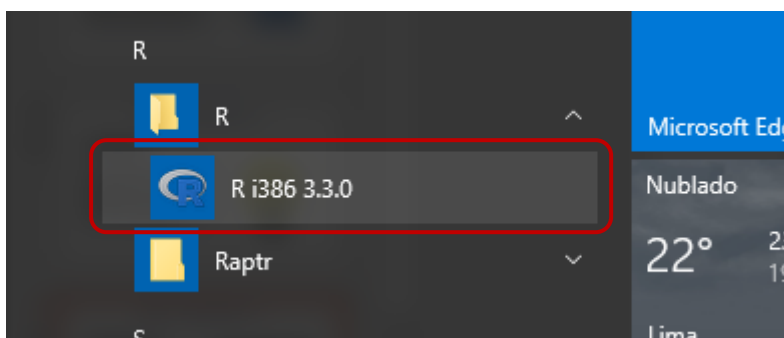
A continuación, se presentan algunos hechos importantes en la historia del lenguaje de programación R:

- Versión 0.16: Es la última versión alfa desarrollada esencialmente por Ihaka y Gentleman, que incluye gran parte de las características descritas en el "White Book".
- La lista de correo comenzó su andadura el 1 de abril de 1997.
- Versión 0.49 (23 de abril de 1997): Es la versión más antigua de la que se conserva el código (que todavía compila en algunas plataformas UNIX). En esta fecha arrancó también CRAN con tres espejos que albergaban 12 paquetes. Poco después aparecieron las versiones alfa para Windows y Mac OS.
- Versión 0.60 (5 de diciembre de 1997): R se integra oficialmente en el Proyecto GNU. El código se versiona a través de CVS.
- R efectuando múltiples tareas en Windows, versión 3.2.2.
- Versión 1.0.0 (29 de febrero de 2000): Los desarrolladores lo consideran suficientemente estable para su uso en producción.<sup>4</sup>
- Versión 1.4.0: Se introducen los métodos S4 y aparece la primera versión para Mac OS X.
- Versión 2.0.0 (4 de octubre de 2004): Introduce el lazy loading, que permite una carga rápida de datos con un coste de memoria mínimo.
- Versión 2.1.0: Aparece el soporte para UTF-8 y comienzan los esfuerzos de internacionalización para distintos idiomas.
- Versión 2.9.0: El paquete 'Matrix' se incluye en la distribución básica de R.

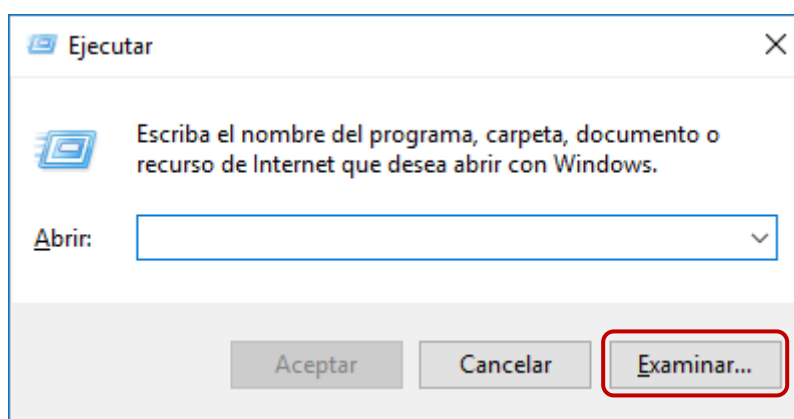
- Versión 2.11.0 (22 de abril de 2010): Soporte para sistemas Windows de 64 bits.
- Versión 2.13.0 (14 de abril de 2011): Añadida una nueva función al compilador que permite acelerar las funciones convirtiéndolas a byte-code.
- Versión 2.14.0 (31 de octubre de 2011): Añadidos espacios de nombres obligatorios para los paquetes. Añadido un nuevo paquete de paralelización.
- Versión 2.15.0 (30 de marzo de 2012): Nuevas funciones de balanceo de cargas. Mejorada la velocidad de serialización para grandes vectores.
- Versión 3.0.0 (3 de abril de 2013): Mejoras en GUI, funciones gráficas, gestión de memoria, rendimiento e internacionalización.

## Iniciando una sesión con R

Para iniciar una sesión con R en el sistema operativo Windows, existen diversas formas, la más común es buscando el ícono en el menú de inicio:



Otra forma es haciendo uso de la herramienta de **Ejecutar** en el menú, haciendo la combinación de teclas **Windows + R**:

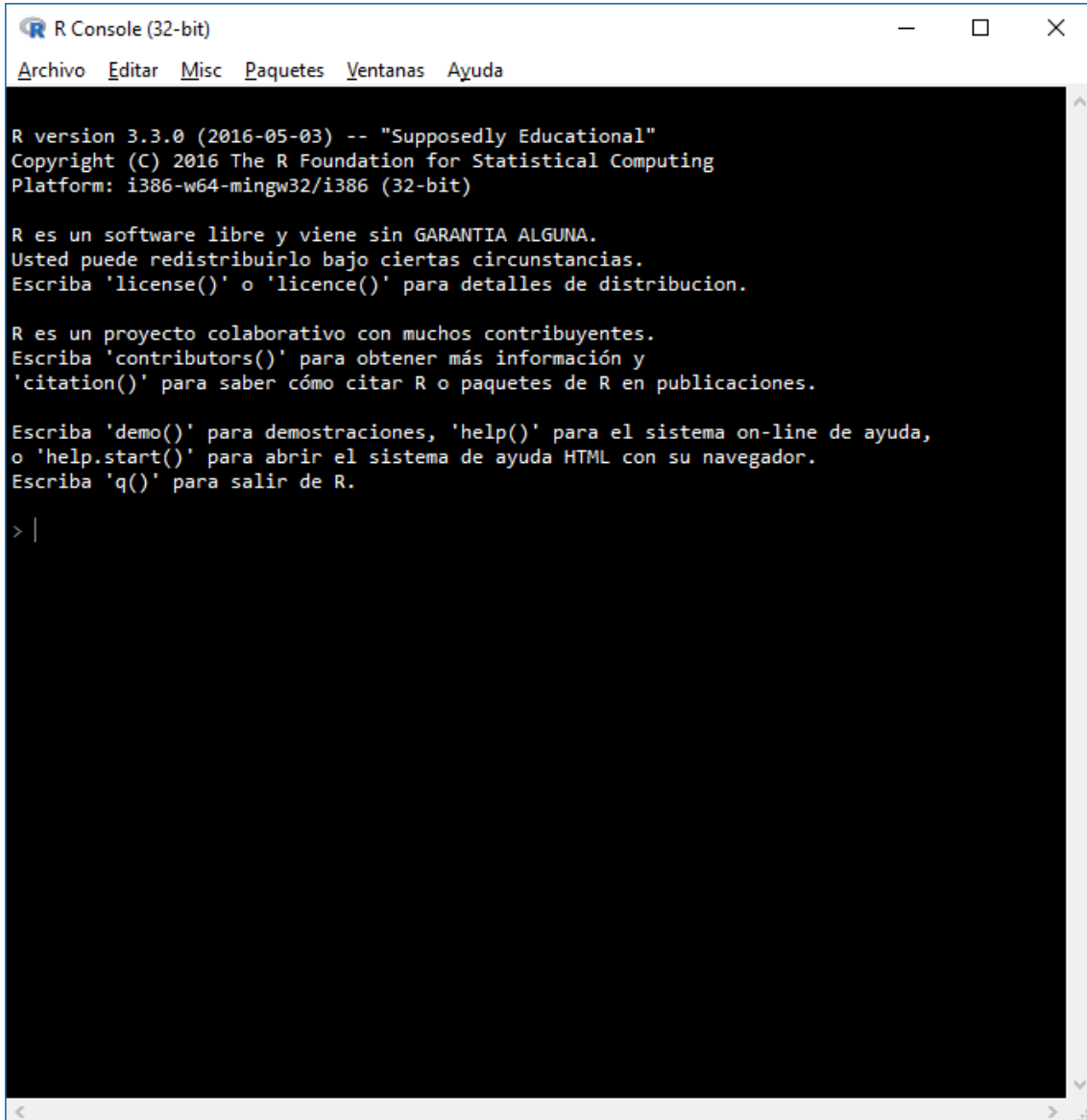


Se deberá dar clic en Examinar y se buscará el programa R, por ejemplo, en el equipo usado para este manual tendrá la siguiente ruta:

**"C:\Program Files (x86)\R-3.3.0\bin\R.exe"**

Una vez que se hayan completado cualquiera de las dos opciones aparecerá la ventana principal del software, que tiene la siguiente apariencia:





```
R Console (32-bit)
Archivo Editar Misc Paquetes Ventanas Ayuda

R version 3.3.0 (2016-05-03) -- "Supposedly Educational"
Copyright (C) 2016 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.

> |
```

## Primera sesión

R es un lenguaje que trabaja básicamente con objetos, por lo que la forma más práctica de trabajar ahí es creándolos bajo la siguiente sintaxis:

```
> x = rnorm(5) #Significa el principio de un comentario
> print(x) # Se mostrarán cada uno de los valores contenidos en el objeto
creado
[1] -0.1961129 -0.7532845 0.8839334 -0.8354918 0.4191791
```

Este es un ejemplo de cómo se trabaja en R, la importancia de los objetos es que se manipulan y se puede hacer interacción entre ellos en la medida posible.

## Ayuda en R

---

Para abrir la ayuda del programa existen diversos métodos, siendo los siguientes:

|                                 |  |
|---------------------------------|--|
| <b>&gt; help()</b>              | Muestra una ventana de ayuda general sobre R.  |
| <b>&gt; help.start()</b>        | Arranca un manual de ayuda completo en formato html, utilizando el navegador del sistema.        |
| <b>&gt; help(mean)</b>          | Muestra una ventana de ayuda sobre la función "media aritmética".                                |
| <b>&gt; ?mean</b>               | Lo mismo que el ejemplo anterior.  |
| <b>&gt; help("[")</b>           | Muestra una ayuda sobre el carácter [ , que es un carácter especial: forma parte del lenguaje R. |
| <b>&gt; apropos(mean)</b>       | Muestra las funciones relacionadas con la función mean.  |
| <b>&gt; help.search("mean")</b> | Busca ayuda sobre objetos o funciones que tengan nombre o título que contenga la cadena "mean"   |

## R como calculadora

R puede servirnos como calculadora, ya que ofrece diversos tipos de herramientas para el cálculo numérico.

Se usa de la siguiente manera:

```
> 2+2
[1] 4
> sqrt(10) # Calcula la raíz cuadrada de 10
[1] 3.162278
> pi # Número PI
[1] 3.141593
> log(1) # Logaritmo natural de 1
[1] 0
> log(10)
[1] 2.302585
> exp(1) # Número de Euler
[1] 2.718282
> log(exp(1))
[1] 1
```

Como se observa R es una potente herramienta no sólo para análisis estadístico, sino que también puede ser utilizado para el cálculo numérico.

A continuación, se presentan las principales funciones que R ofrece para hacer operaciones matemáticas:

### Operadores aritméticos

Los operadores aritméticos que podemos emplear en R, se detallan en la siguiente tabla.

| Operadores Aritméticos. |                 |                    |
|-------------------------|-----------------|--------------------|
| Operador                | Definición      | Ejemplo            |
| +                       | Suma            | > 9 + 2<br>[1] 11  |
| -                       | Resta           | > 9 - 2<br>[1] 7   |
| *                       | Multiplicación  | > 9 * 2<br>[1] 18  |
| /                       | División        | > 9 / 2<br>[1] 4.5 |
| ^                       | Potencia        | > 9^2<br>[1] 81    |
| %%                      | División Entera | > 9 %/% 2<br>[1] 4 |
| %%                      | Resto/Residuo   | > 9 %% 2<br>[1] 1  |

## Operadores lógicos

Los operadores lógicos se describen en la siguiente tabla.

| Operadores Lógicos. |                |   |
|---------------------|----------------|---|
| Operador            | Definición     | Ejemplo   |
| !                   | Negación (Not) | <pre>&gt; vector1 &lt;- c(TRUE,FALSE,TRUE) &gt; !vector1 [1] FALSE TRUE FALSE</pre>                                     |
|                     | OR             | <pre>&gt; vector1 &lt;- c(TRUE,FALSE) &gt; vector2 &lt;- c(TRUE,TRUE) &gt; vector1   vector2 [1] TRUE TRUE</pre>        |
| &                   | AND            | <pre>&gt; vector1 &lt;- c(TRUE, FALSE) &gt; vector2 &lt;- c(TRUE, TRUE) &gt; vector1 &amp; vector2 [1] TRUE FALSE</pre> |
| xor(a, b)           | OR-Exclusivo   | <pre>&gt; vector1 &lt;- c(TRUE, FALSE) &gt; vector2 &lt;- c(TRUE, TRUE) &gt; xor(vector1, vector2) [1] FALSE TRUE</pre> |
|                     | Sentencia: OR  | <pre>if ((a &lt;&gt; 5))</pre>  |
| &&                  | Sentencia: AND | <pre>if ((a &lt;&gt; 5))</pre>  |

OR significa suma lógica, mientras AND es la multiplicación lógica. Los operandos || y &&, se usan para sentencias de control, normalmente en sentencias if, while, etc.

## Operadores de comparación

Los operadores de comparación se muestran en la siguiente tabla.

| Operadores de Comparación. |                   |                            |
|----------------------------|-------------------|----------------------------|
| Operador                   | Definición        | Ejemplo                    |
| >                          | Mayor que         | <pre>if (a &gt; 4)</pre>   |
| <                          | Menor que         | <pre>if (a &lt; 4)</pre>   |
| ==                         | Igual que         | <pre>if (b == 5)</pre>     |
| >=                         | Mayor o igual que | <pre>if (b &gt;= 11)</pre> |
| <=                         | Menor o igual que | <pre>if (a &lt;= b)</pre>  |
| !=                         | Distinto de       | <pre>if (a != 0)</pre>     |

Los operadores de comparación son útiles, principalmente, en las sentencias de control: if, while, etc, aunque si las usamos en la consola, R nos responderá como si se tratase de una operación lógica.

### Funciones matemáticas

Las funciones matemáticas predefinidas en R, se pueden englobar en cinco categorías: Funciones de Signo, Funciones Trigonómicas, Funciones Hiperbólicas, Funciones de Redondeo y Funciones Exponenciales/Logarítmicas.

| Funciones de Signo.  |   |  |
|----------------------|---|--|
| Operador             | Definición  | Ejemplo  |
| <code>abs(a)</code>  | Valor absoluto de a   | <code>&gt; abs(-4)</code><br>[1] 4   |
| <code>sign(a)</code> | Devuelve el signo de a:<br><br>1 $\equiv$ Positivo<br>0 $\equiv$ Cero<br>-1 $\equiv$ Negativo | <code>&gt; sign(-4)</code><br>[1] -1<br><code>&gt; sign(11-11)</code><br>[1] 0<br><code>&gt; sign(8)</code><br>[1] 1 |
| <code>sqrt(a)</code> | Raíz cuadrada de a  | <code>&gt; sqrt(4)</code><br>[1] 2   |

| Funciones de Trigonómicas. |                   |   |
|----------------------------|-------------------|---|
| Operador                   | Definición        | Ejemplo                                       |
| <code>cos(a)</code>        | Coseno de a       | <code>&gt; cos(0)</code><br>[1] 1             |
| <code>sin(a)</code>        | Seno de a         | <code>&gt; sin(pi/2)</code><br>[1] 1          |
| <code>tan(a)</code>        | Tangente de a     | <code>&gt; tan(0)</code><br>[1] 0             |
| <code>acos(a)</code>       | Arcocoseno de a   | <code>&gt; acos(0.60)</code><br>[1] 0.9272952 |
| <code>asin(a)</code>       | Arcoseno de a     | <code>&gt; asin(0.43)</code><br>[1] 0.4444928 |
| <code>atan(a)</code>       | Arcotangente de a | <code>&gt; atan(0.11)</code><br>[1] 0.1095595 |

| Funciones Hiperbólicas. |                         |                                    |
|-------------------------|-------------------------|------------------------------------|
| Operador                | Definición              | Ejemplo                            |
| <code>cosh(a)</code>    | Coseno hiperbólico de a | <code>&gt; cosh(0)</code><br>[1] 1 |

|                       |                               |   |
|-----------------------|-------------------------------|---|
| <code>sinh(a)</code>  | Seno hiperbólico de a         | <code>&gt; sinh(pi/2)</code><br><code>[1] 2.301299</code>   |
| <code>tanh(a)</code>  | Tangente hiperbólica de a     | <code>&gt; tanh(0)</code><br><code>[1] 0</code>             |
| <code>acosh(a)</code> | Arcocoseno hiperbólico de a   | <code>&gt; acosh(pi)</code><br><code>[1] 1.811526</code>    |
| <code>asinh(a)</code> | Arcoseno hiperbólico de a     | <code>&gt; asinh(pi/2)</code><br><code>[1] 1.233403</code>  |
| <code>atanh(a)</code> | Arcotangente hiperbólico de a | <code>&gt; atanh(0.11)</code><br><code>[1] 0.1104469</code> |

| Funciones de Redondeo.  |  |  |
|-------------------------|--|--|
| Operador                | Definición   | Ejemplo  |
| <code>ceiling(a)</code> | Devuelve el menor entero $\geq a$  | <code>&gt; ceiling(4.5)</code><br><code>[1] 5</code>   |
| <code>floor(a)</code>   | Devuelve el mayor entero $\leq a$  | <code>&gt; floor(4.5)</code><br><code>[1] 4</code>   |
| <code>trunc(x)</code>   | Devuelve entero más cercano a x, entre x y 0, inclusive.<br><br>Esta función es como <i>floor</i> para valores positivos y como <i>ceiling</i> para valores negativos. | <code>&gt; trunc(4.5)</code><br><code>[1] 4</code><br><br><code>&gt; trunc(-4.5)</code><br><code>[1] -4</code> |

| Funciones Exponenciales y Logarítmicas. |   |  |
|---|---|--|
| Operador                                | Definición  | Ejemplo  |
| <code>log(a, base=exp(1))</code>        | Logaritmo de a de base exponente 1                    | <code>&gt; log(4, base=exp(1))</code><br><code>[1] 1.386294</code> |
| <code>log10(a)</code>                   | Logaritmo de a de base 10                             | <code>&gt; log10(10)</code><br><code>[1] 1</code>                  |
| <code>log2(a)</code>                    | Logaritmo de a de base 2                              | <code>&gt; log2(2)</code><br><code>[1] 1</code>                    |
| <code>log1p(a)</code>                   | Realiza el logaritmo $\log(1+a)$ para $ a  \ll 1$     | <code>&gt; log1p(0.0001)</code><br><code>[1] 9.9995e-05</code>     |
| <code>exp(a)</code>                     | Función exponencial de a                              | <code>&gt; exp(1)</code><br><code>[1] 2.718282</code>              |
| <code>expm1(a)</code>                   | Realiza el exponencial $\exp(a) - 1$ para $ a  \ll 1$ | <code>&gt; expm1(0.0001)</code><br><code>[1] 0.000100005</code>    |

## Espacio y directorio de trabajo

El **"workspace"** es el espacio de trabajo en que se incluyen todos los objetos definidos por el usuario (ya veremos qué son estos objetos, que incluyen variables, vectores, dataframes...), se almacena en memoria intermedia mientras trabajas con R.

Cuando termina una sesión de R el propio R te pregunta si quieres guardar el **"workspace"** para usos futuros. Este espacio, **"workspace"**, se recarga al volver a iniciar la sesión. Directorio de trabajo o **"working directory"** es el directorio donde por defecto "lee" R. También es donde guardará el **workspace** al finalizar la sesión y donde buscará un **workspace** guardado al inicio. Si se quiere que R lea un fichero que no esté en **"working directory"** hay que especificar la ruta completa.

Estas son las funciones asociadas al workspace:

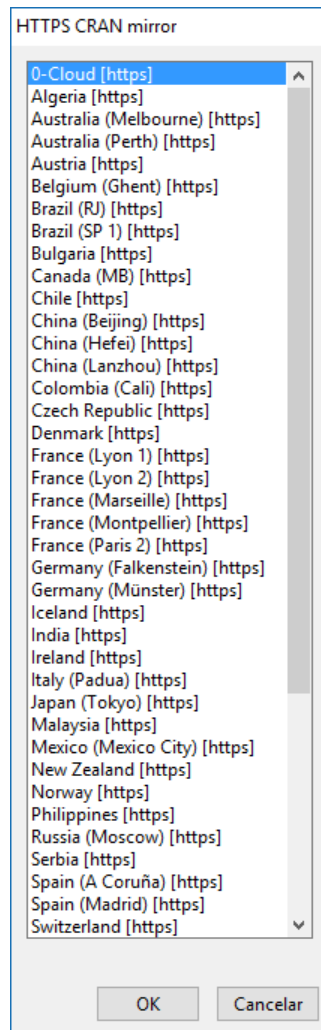
| Función                                      | Acción   |
|--|--|
| <code>&gt;getwd()</code>                     | Muestra el wd: <i>working directory</i>                          |
| <code>&gt;setwd("midirectorio")</code>       | Ajusta el wd al especificado                                     |
| <code>&gt;ls()</code> o <code>dir()</code>   | Lista lo que hay en el <b>wd</b>                                 |
| <code>&gt;history()</code>                   | Muestra los últimos comandos ejecutados                          |
| <code>&gt;savehistory()</code>               | Guarda el historial de comandos, por defecto en <b>.RHistory</b> |
| <code>&gt;loadhistory()</code>               | Carga el historial de comandos                                   |
| <code>&gt;save.image("myworkspace.R")</code> | Guarda los objetos del workspace, por defecto en <b>.RData</b>   |
| <code>&gt;load("myworkspace.R")</code>       | Carga el workspace <b>myworkspace.R</b>                          |

## Paquetes en R

Los paquetes de R permiten incrementar la capacidad de R: funciones, importación, exportación y manipulación de datos, gráficos, técnicas estadísticas, creación de informes etc. Son bibliotecas de código gratuitas creadas por la comunidad de usuarios y disponibles en varios repositorios como **Comprehensive R Archive Network (CRAN)** o **Bioconductor**.

### Instalar paquetes

Existen básicamente dos formas para instalar paquetes en R una puede ser yendo a **Paquetes > Instalar paquete(s)**, donde se deberá seleccionar el "CRAN mirror".



La otra forma es haciendo uso del comando **install.packages()**. Al igual que la forma anterior también se deberá iniciar el CRAN Mirror para instalar el paquete.



## Cargar librerías

---

Una vez instalado el paquete se debe cargar el mismo a fin de que se puedan usar las funciones relacionadas a dicha librería. Para eso se debe uso del comando **library()**.

```
> update.packages("ggplot2")  
--- Please select a CRAN mirror for use in this session ---  
> library("ggplot2")
```

## Bibliografía

Castillo, A. J. S. (2010). Métodos Estadísticos con R y R Commander. *Recurso electrónico gratuito*.

### Recursos informáticos

---

Cran Project - R History, Genesis:

[https://cran.r-project.org/doc/html/interface98-paper/paper\\_1.html](https://cran.r-project.org/doc/html/interface98-paper/paper_1.html)

Nube de datos - Instalar y gestionar paquetes en R:

<http://nubededatos.blogspot.pe/2014/01/instalar-y-gestionar-paquetes-en-r.html>

R: Operaciones matemáticas:

<http://unbarquero.blogspot.pe/2009/03/r-operaciones-matematicas.html>

Statistics Wordpress - Historia de R:

<https://sajumavi.wordpress.com/2013/05/11/historia-de-r/>

Universidad de Murcia - Entorno de trabajo R:

<http://www.um.es/ae/FEIR/10/#por-que-emplear-r>