

# Introducción a



11-Set-2020 DataClubUY

Inés Rebollo

Sheila Scheffel

# Porqué usar



- Software libre
- Análisis estadísticos
- Trabajar con muchos datos
- Reproducible y repetible
- Gráficos



lenguaje de programación  
orientado a objetos

interfaz gráfica

Link descarga R: <http://cran.r-project.org/>

Ling descarga RStudio: <https://rstudio.com/products/rstudio/download/>

R para principiantes: [https://cran.r-project.org/doc/contrib/rdebuts\\_es.pdf](https://cran.r-project.org/doc/contrib/rdebuts_es.pdf)

1 #R como calculadora

2

3 # Suma

4 5 + 5

5

6 # Resta

7 5 - 5

8

9 # Multiplicacion

10 3 \* 5

11

12 # Division

11:1 (Top Level) R Script

Console

Terminal

Jobs

> df <- data.frame(Animal = 1:4,  
+ Edad = 5:7,  
+ Sexo = c("F", "M", "F", "M"))  
Error in data.frame(Animal = 1:4, Edad = 5:7, Sexo = c("F", "M",  
"F", :  
arguments imply differing number of rows: 4, 3  
> df <- data.frame(Animal = 1:4,  
+ Edad = 5:8,  
+ Sexo = c("F", "M", "F", "M"))  
>

Environment

History

Connections

Global Environment

Data

df 4 obs. of 3 variab...

Values

caracter "hold"

x 42

y 20

Functions

nombreF... function (argument...

Files

Plots

Packages

Help

Viewe

sum

R: Sum of Vector Elements

sum {base}

R Documentation

Sum of Vector Elements

Description

sum returns the sum of all the values present in its arguments.

Usage

## Scripts

Ejecutar línea: Run = Ctr/Cmd + Enter/R

## Consola

## Entorno

## Gráficos

## Archivos

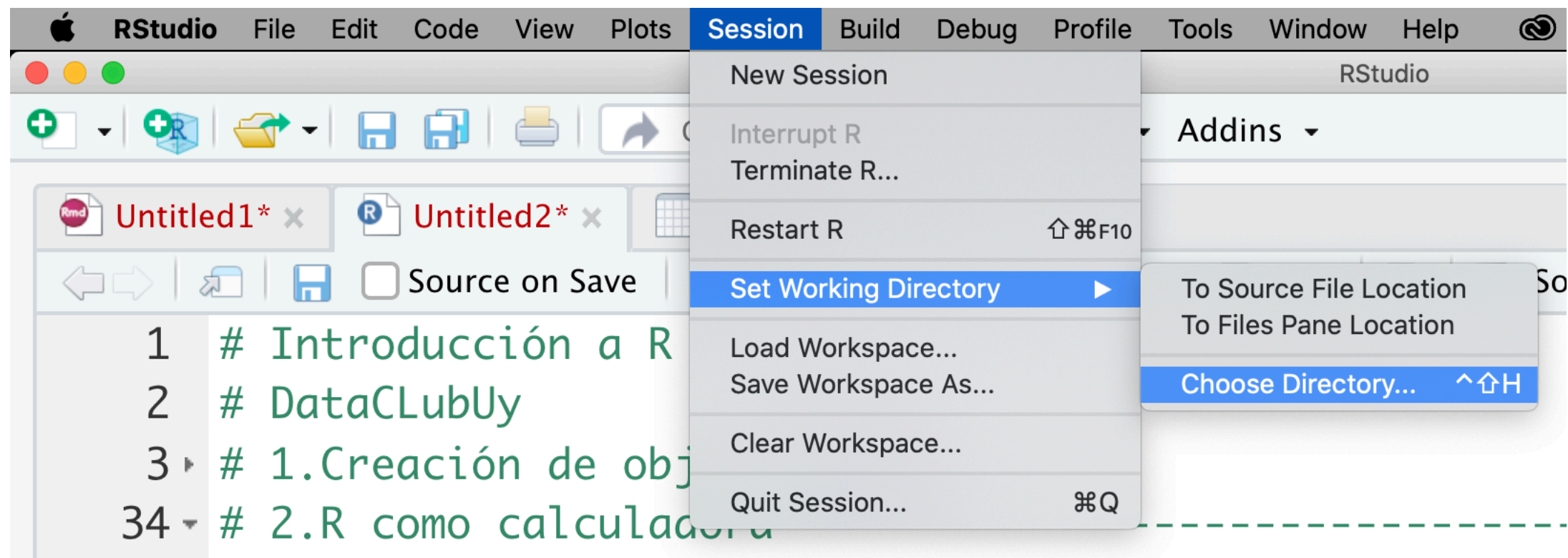
## Paquetes

## Ayuda

# Directorio de trabajo

Importante definirlo al empezar a trabajar

Es donde se van a buscar y guardar los archivos por defecto



alternativa:  
`setwd("path")`

# R como calculadora

+

suma

-

resta

\*

multiplicación

/

división

^ 0 \*\*

potencia

%%

resto

/%

cociente

# Nombre de objetos

- letras (mayúsculas, minúsculas),
- dígitos del 0 al 9 (no en posición inicial)
- `_` y `.` (no espacios)

Palabras reservadas (no usar)

`break` `else` `FALSE` `for` `function` `if` `in`  
`Inf` `NA` `NaN` `next` `NULL` `repeat` `TRUE` `while`

# Tipo de objetos

`integer`

números enteros

`numeric`

números con decimales

`character`

caracteres, ""

`logical`

TRUE (T) o FALSE (F)

`matrix`

tabla con mismo tipo de dato

`data.frame`

tabla, admite distinto tipo de dato

tibble  
data table

`list`

listas

`function`

funciones



# Creación de objetos

`x <- 5`  
asignación

x es un objeto de  
tipo numeric

 Alt + -

 Option + -

`y <- c(TRUE, FALSE)`  
concatenar

y es un **vector** de  
tipo logical

`z <- c("ID_1", "ID_2")`

Z es un **vector** de  
tipo character

# Creación de objetos

`matriz <- matrix()` Al apretar tabulador se muestran los argumentos de la función

- ◆ data =
- ◆ nrow =
- ◆ ncol =
- ◆ byrow =
- ◆ dimnames =
- ◆ matriz
- ◆ -

## data

an optional data vector (including a list or expression vector). Non-atomic classed **R** objects are coerced by `as.vector` and all attributes discarded.

Press F1 for additional help

del 1 al 100

```
matriz <- matrix(data = 1:100, nrow = 10,  
                 ncol = 10, byrow = TRUE)
```

1 2 3 vs. 1 3 5  
4 5 6 2 4 6

# Creación de objetos

```
x <- 5
```

```
y <- c(TRUE, FALSE)
```

```
z <- c("ID_1", "ID_2")
```

```
df <- data.frame(ID = z,  
                 esHembra = y,  
                 edad = x) Recycling rule
```

ID	esHembra	edad
ID_1	TRUE	5
ID_2	FALSE	5

# Indexado de vectores

```
vector <- c(1:20)
```

```
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

```
vector[2]
```

```
vector[18:20]
```

```
vector[c(4,10)]
```

```
vector[-2]
```

```
[1] 1 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
```

Se puede usar para reemplazar elementos

# Indexado de matrices

	[,1]	[,2]	[,3]	[,4]	[,5]	[,6]	[,7]	[,8]	[,9]	[,10]
[1,]	1	2	3	4	5	6	7	8	9	10
[2,]	11	12	13	14	15	16	17	18	19	20
[3,]	21	22	23	24	25	26	27	28	29	30
[4,]	31	32	33	34	35	36	37	38	39	40
[5,]	41	42	43	44	45	46	47	48	49	50
[6,]	51	52	53	54	55	56	57	58	59	60
[7,]	61	62	63	64	65	66	67	68	69	70
[8,]	71	72	73	74	75	76	77	78	79	80
[9,]	81	82	83	84	85	86	87	88	89	90
[10,]	91	92	93	94	95	96	97	98	99	100

`matriz[5,7]`

`matriz[1,c(2,4)]`

`matriz[9:10,]`

`matriz[,c(1:8)]`

# Indexado de data frames

ID	esHembra	edad
ID_1	TRUE	5
ID_2	FALSE	5

df\$      tabulador

◆ ID
◆ esHembra
◆ edad

```
df$ID = df[,1] = df[, "ID"]
```

```
df[, -1] = df[, 2:3] = df[, c(2, 3)] =  
df[, c("esHembra", "edad")]
```

# Indexado de listas

```
lista <- list(x, y, z)
```

```
[[1]]  
[1] 5
```

```
[[2]]  
[1] TRUE FALSE
```

```
[[3]]  
[1] "ID_1" "ID_2"
```

`lista[[2]]` Devuelve el elemento

`lista[2]` Devuelve una lista

# Operadores lógicos

<

menor

>

mayor

<=

menor o igual

>=

mayor o igual

==

igual

!=

distinto

%in%

está en

&

y

|

o



# Valores especiales

NA

"Not Available"

NULL

Objeto vacío

NaN

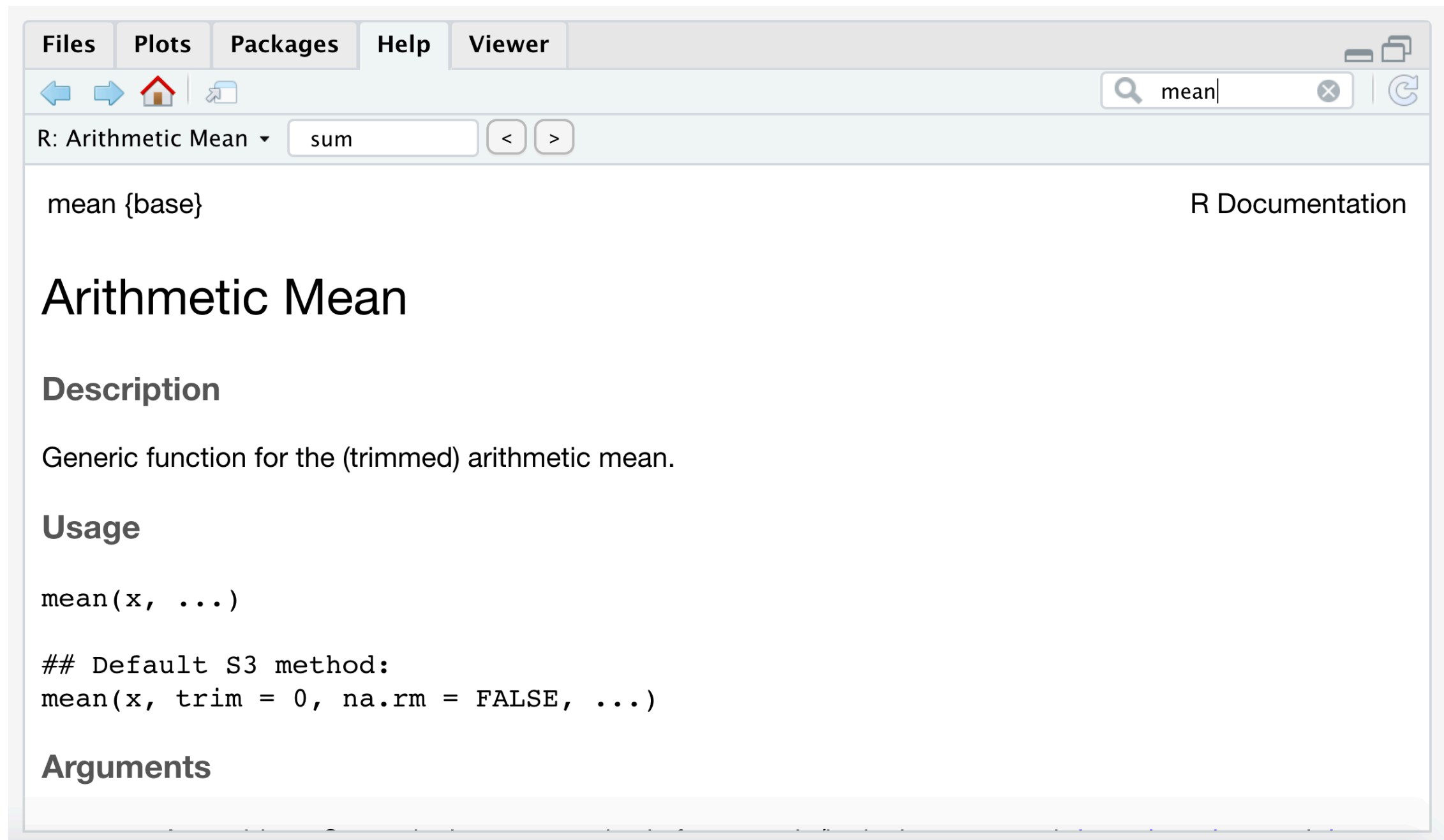
"Not a Number" ej.:  $\sqrt{-1}$

# Operaciones con vectores

<code>max(vector)</code>	máximo
<code>min(vector)</code>	mínimo
<code>range(vector)</code>	rango
<code>mean(vector)</code>	media
<code>sum(vector)</code>	suma de los elementos
<code>cumsum(vector)</code>	suma acumulada
<code>sd(vector)</code>	desvío estándar
<code>quantile(vector)</code>	cuantiles

# Help

?mean



The screenshot shows the R Help viewer window. The title bar includes tabs for Files, Plots, Packages, Help, and Viewer. The Help tab is active, and the search bar contains the text 'mean'. The main content area displays the documentation for the 'mean' function, which is titled 'Arithmetic Mean'. The documentation includes a description, usage, and arguments section.

mean {base} R Documentation

## Arithmetic Mean

### Description

Generic function for the (trimmed) arithmetic mean.

### Usage

```
mean(x, ...)
```

## Default S3 method:  
mean(x, trim = 0, na.rm = FALSE, ...)

### Arguments

# Help


??mean


FilesPlotsPackagesHelpViewer

mean

R: Search Results ▾sum<>

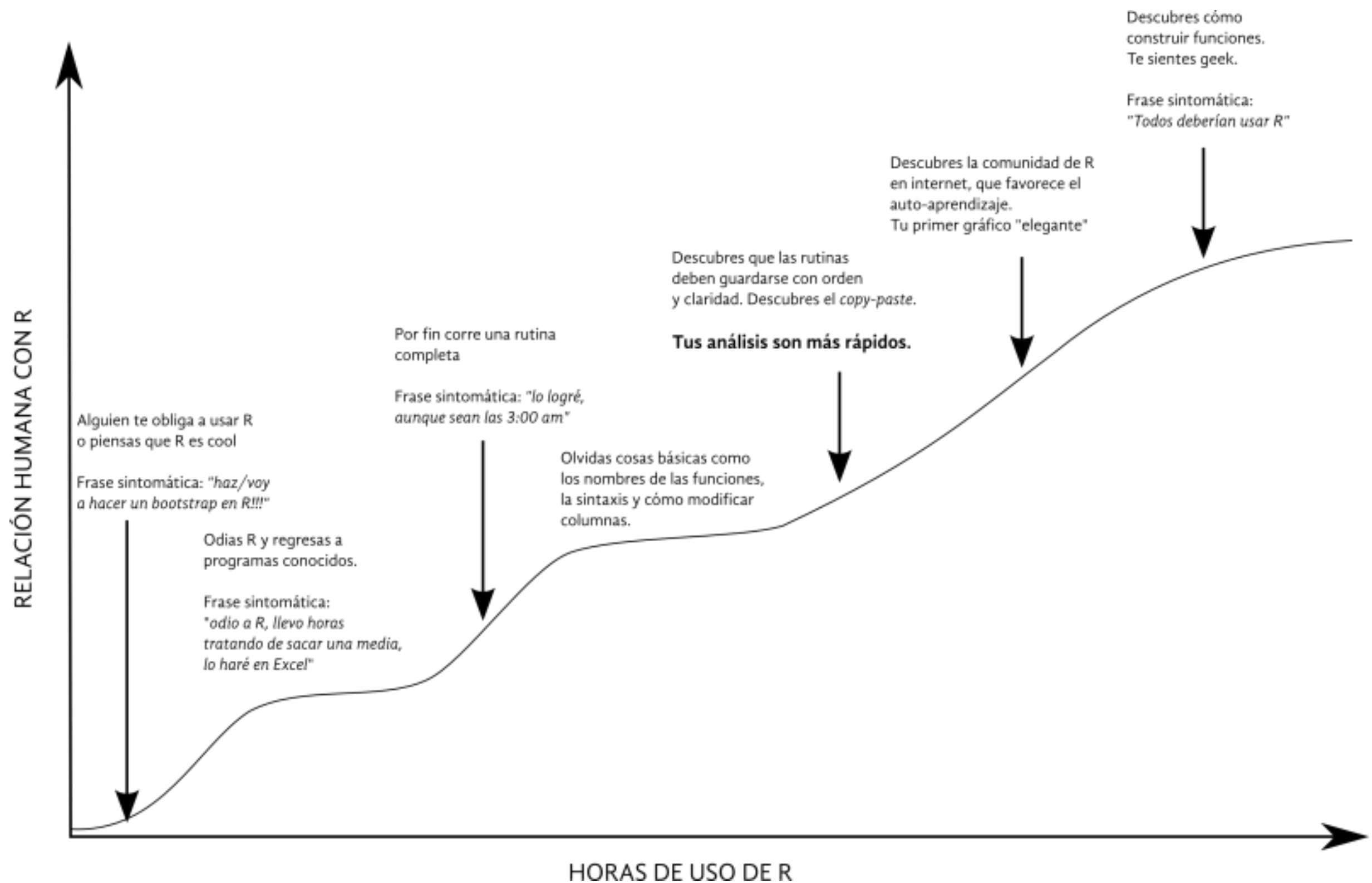
Search Results





Vignettes:

<a href="#">broom::kmeans</a>	kmeans with dplyr and broom	<a href="#">HTML</a>	<a href="#">source</a>	<a href="#">R code</a>
<a href="#">emmeans::FAQs</a>	FAQs for emmeans	<a href="#">HTML</a>	<a href="#">source</a>	<a href="#">R code</a>
<a href="#">emmeans::interactions</a>	Interaction analysis in emmeans	<a href="#">HTML</a>	<a href="#">source</a>	<a href="#">R code</a>
<a href="#">emmeans::models</a>	Models supported by emmeans	<a href="#">HTML</a>	<a href="#">source</a>	
<a href="#">emmeans::predictions</a>	Prediction in emmeans	<a href="#">HTML</a>	<a href="#">source</a>	<a href="#">R code</a>
<a href="#">emmeans::sophisticated</a>	Sophisticated models in emmeans	<a href="#">HTML</a>	<a href="#">source</a>	<a href="#">R code</a>



# Levantado de datos

`read.table()`      `archivos.txt`

```
read.table(file, header = FALSE, sep = "", quote = "\"", dec = ".", numerals = c("allow.loss",  
"warn.loss", "no.loss"), row.names, col.names, as.is = !stringsAsFactors, na.strings = "NA",  
colClasses = NA, <...truncated...> )
```

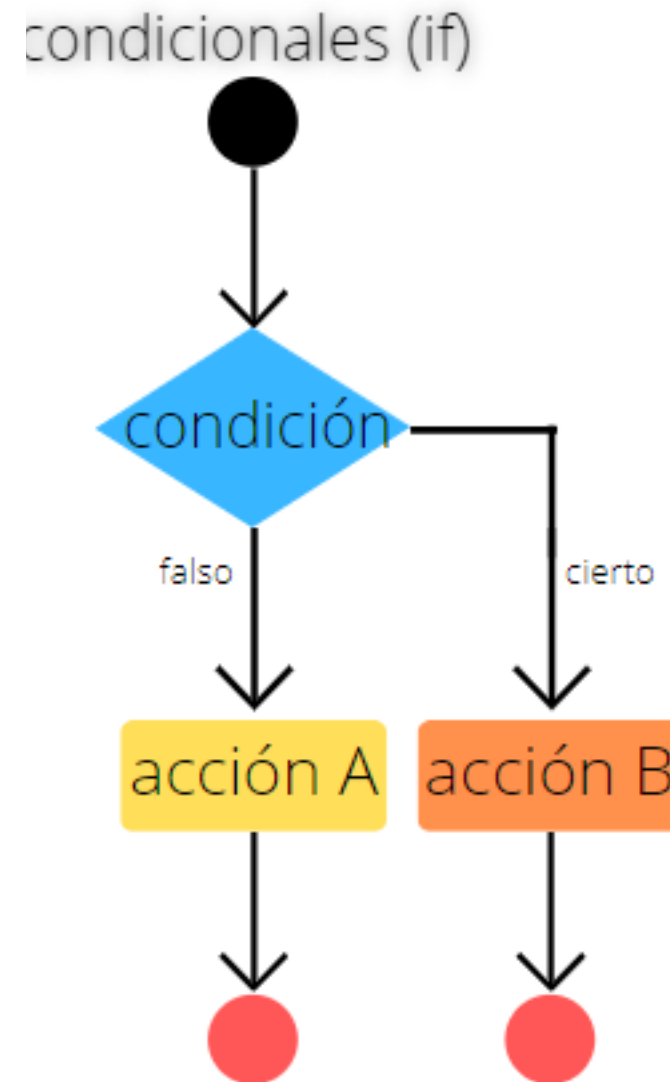
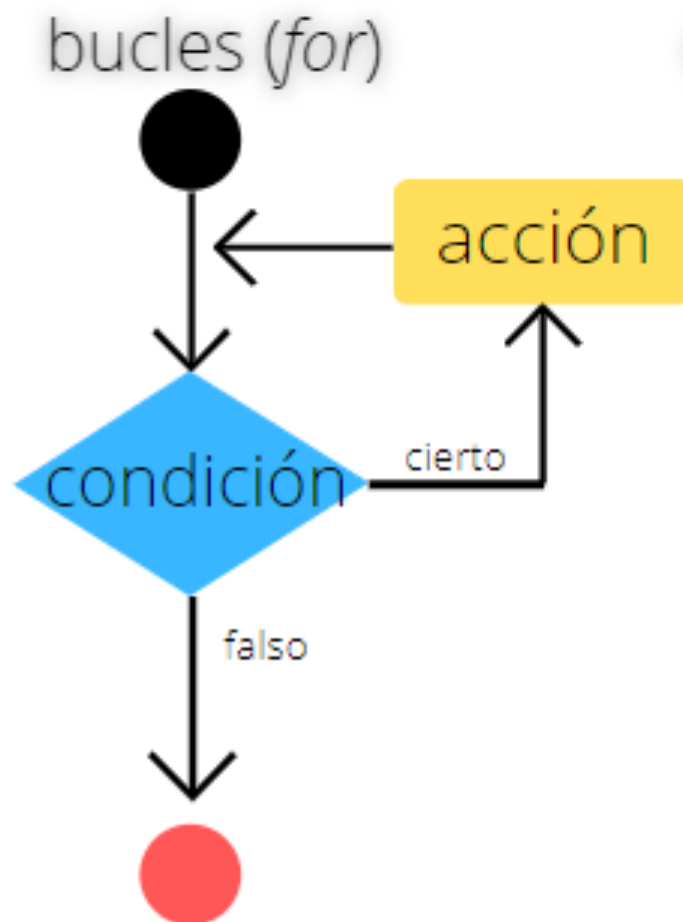
`read.table()`

```
read.table(file = "datos.txt", header = TRUE,  
           sep = "\t", stringsAsFactors = FALSE)
```

`read.csv()`      csv delimitado por "," y decimal con "."

`read.csv2()`      csv delimitado por ";" y decimal con ","

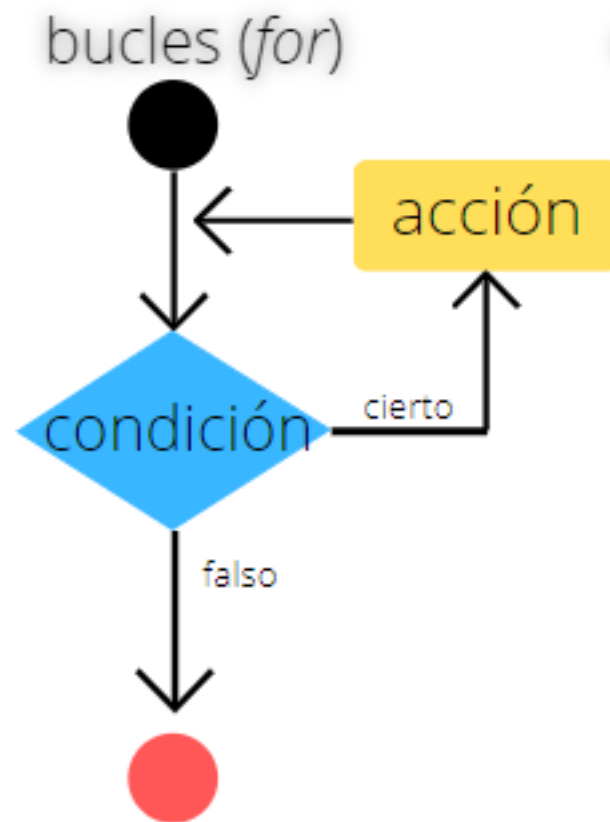
# Loops



Otros: while, repeat

# Loop for

Supuestamente poco eficiente

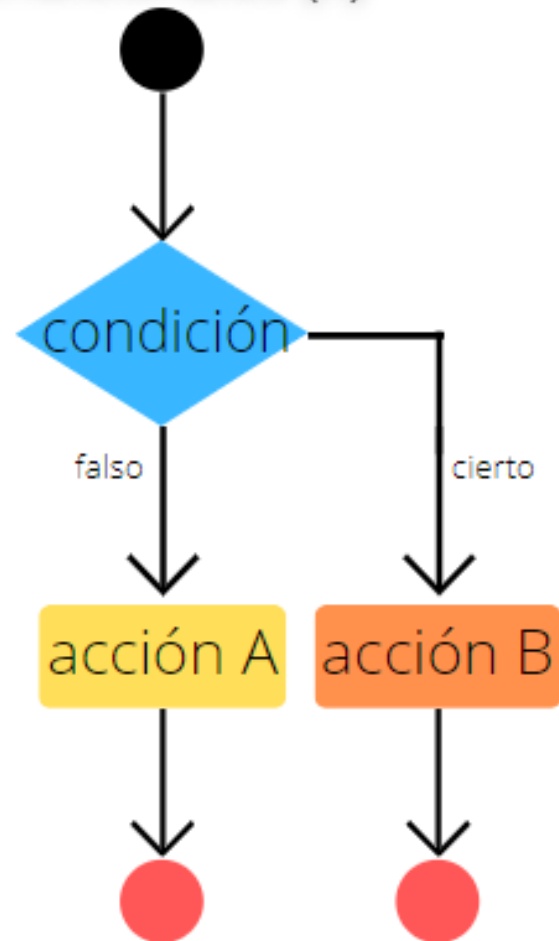


```
for (variable in vector) {  
    acción1  
    acción2  
    ...  
}
```



# Loop if

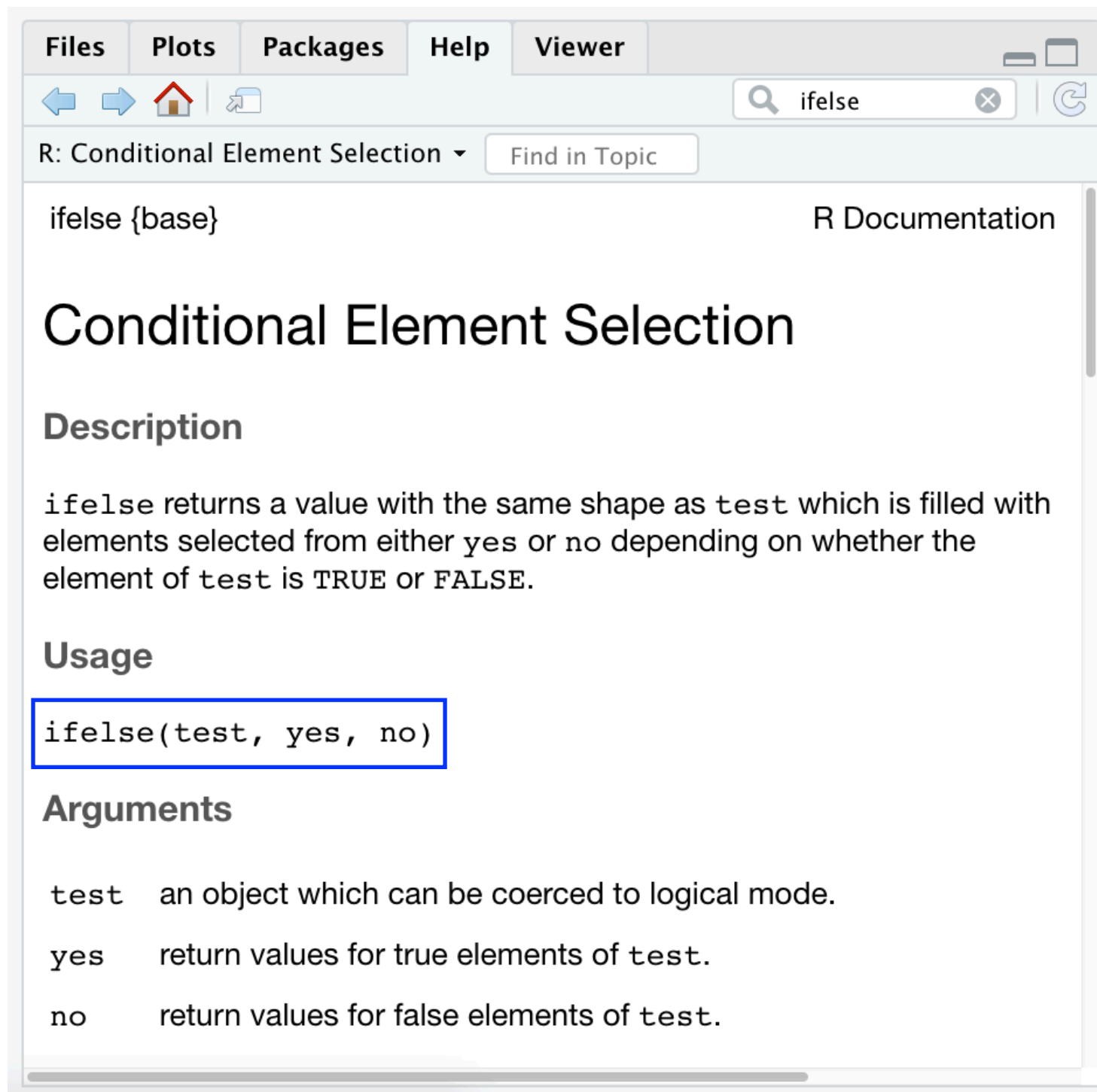
condicionales (if)



```
if (condición) {  
    acción1  
    acción2  
    ...  
} else if (condición) {  
    acción3  
    acción4  
    ...  
} else {  
    acción5  
    acción6  
    ...  
}
```

`next` pasa al siguiente y `break` termina al loop

# Loop ifelse



The screenshot shows the R Documentation window for the `ifelse` function. The window has a menu bar with 'Files', 'Plots', 'Packages', 'Help', and 'Viewer'. Below the menu bar is a search bar containing 'ifelse' and a 'Find in Topic' button. The main content area displays the title 'Conditional Element Selection' and a description of the `ifelse` function. The function signature `ifelse(test, yes, no)` is highlighted with a blue box. The arguments section lists `test`, `yes`, and `no` with their respective descriptions.

Files Plots Packages Help Viewer

ifelse {base} R Documentation

## Conditional Element Selection

### Description

`ifelse` returns a value with the same shape as `test` which is filled with elements selected from either `yes` or `no` depending on whether the element of `test` is `TRUE` or `FALSE`.

### Usage

```
ifelse(test, yes, no)
```

### Arguments

`test` an object which can be coerced to logical mode.

`yes` return values for true elements of `test`.

`no` return values for false elements of `test`.

# Funciones apply

Alternativas al for, supuestamente más eficientes

Permiten aplicar una función a una matriz, lista o vector

`apply(x, filas/columnas 1/2, FUN)`

`lapply(x, FUN)`

`sapply`

`tapply`

`mapply`

`vapply`

recibe, devuelve

vector matriz, =

lista, lista

lista, vector matriz

vector matriz lista, lista

matriz, lista o vector

listaS, vector con largo

# Funciones

```
nombreFuncion <- function (Arg1 = default1, Arg2 =  
                           default2) {  
  cuerpoFuncion  
}
```

# Secciones en un Script

Cualquier línea de comentario que termine en 4 = - o # se convierte en una sección

Crear una sección



Ctrl + Shift + R



Cmd + Shift + R

Colapsar la sección actual



Alt + L



Cmd + Option + L

Colapsar todas las secciones



Alt + O



Cmd + Option + O

Expandir la sección actual



Shift + Alt + L



Shift + Cmd + Option + L

Expandir todas las secciones



Shift + Alt + O



Shift + Cmd + Option + O

# Mirar los errores

```
> x + 5
```

```
Error: object 'x' not found
```

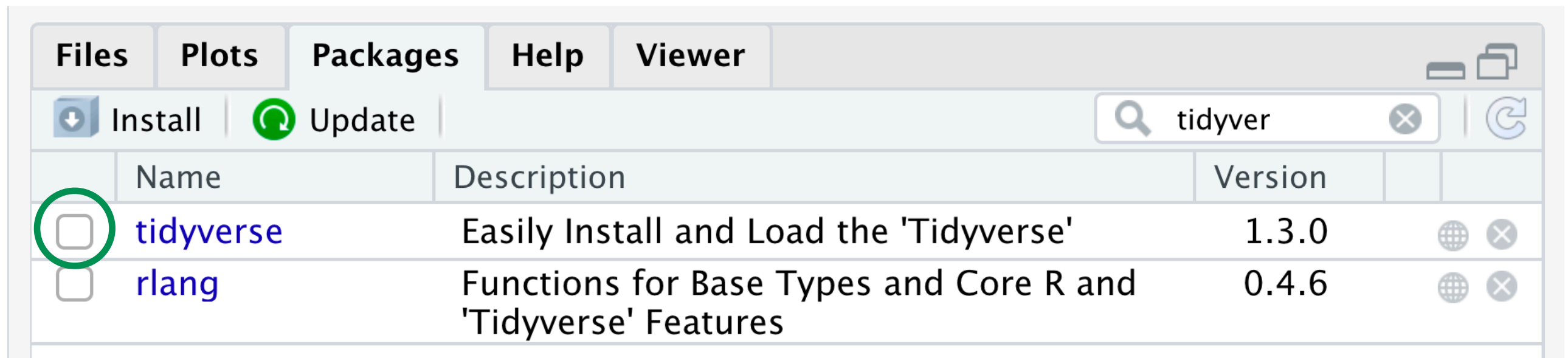
```
> select()
```

```
Error in select() : could not find function "select"
```

# Cargar paquetes

## Importante:

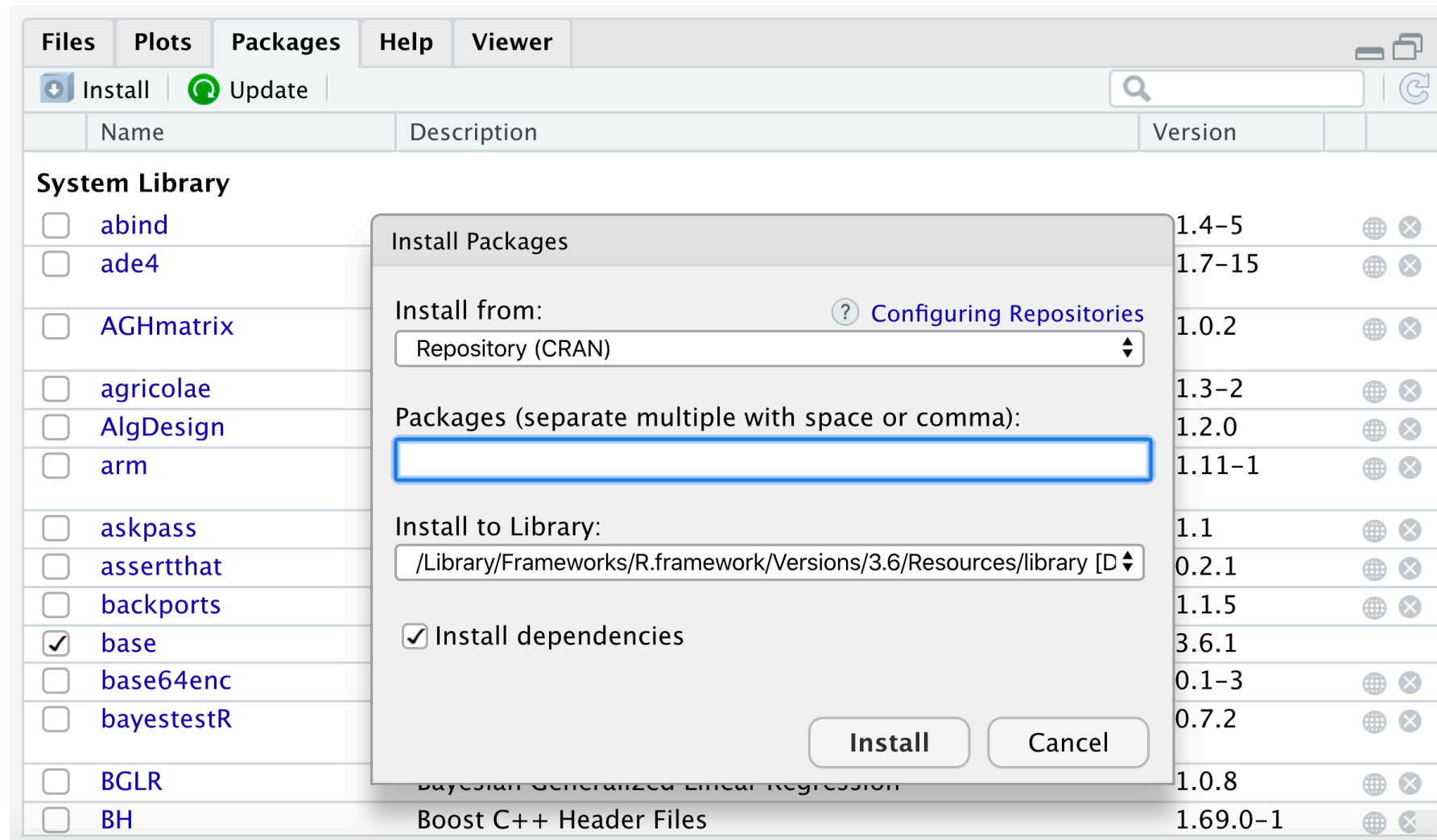
Si no se carga el paquete salta un error porque no se encuentra la función



## Alternativas:

```
library(tidyverse)
tidyverse::select()
```

# Instalar paquetes



Alternativa:

```
install.packages("tidyverse")
```



# Otros

(Des)comentar líneas



Ctrl + Shift + C



Cmd + Shift + C

%>%

pipe, = | en bash



Ctrl + Shift + M



Cmd + Shift + M

# Otros

	<b>data.frame</b>	<b>data.table</b>	<b>tibble</b>
	base R	Paquete data.table	Paquete tidyverse
<b>Visualización</b>	imprime todo el df	cuando el n° de filas > 100, automáticamente imprime las 5 primeras y 5 últimas	mejor visualización que df imprime 10 filas y las columnas que se ajusten a la pantalla -imprime colores y fuentes para resaltar
<b>Subseteo</b>		sintaxis: DT[i, j, by] dado DT, subset/reorder filas usando i, luego calcula j, agropado por by	más estricto que df, [ ] siempre devuelve un tibble
<b>Convierte clase de columnas, pe, de caracter a vector</b>	si	no	no
<b>Nombre de variables</b>	no permite algunos caracteres (espacio pe)		permite caracteres que df no permite
<b>Nombre de filas</b>	numéricos, generados automáticamente	impresos con : para separar el numero de fila de la primer columna	nunca usa rownames()

**THANK YOU FOR  
YOUR ATTENTION**



**ANY QUESTIONS,  
ASK GOOGLE**

*memegenerator.es*