

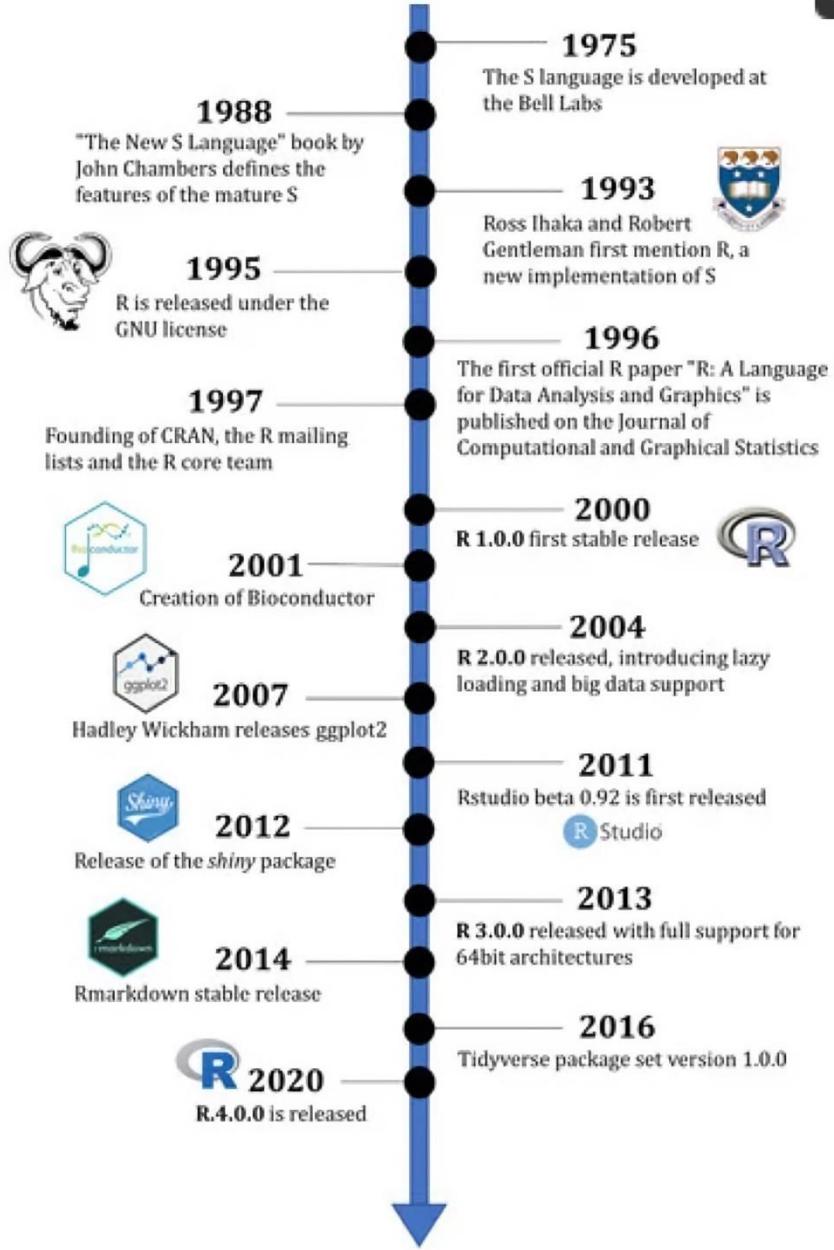


Introducción a R

José Luis Texcalac Sangrador

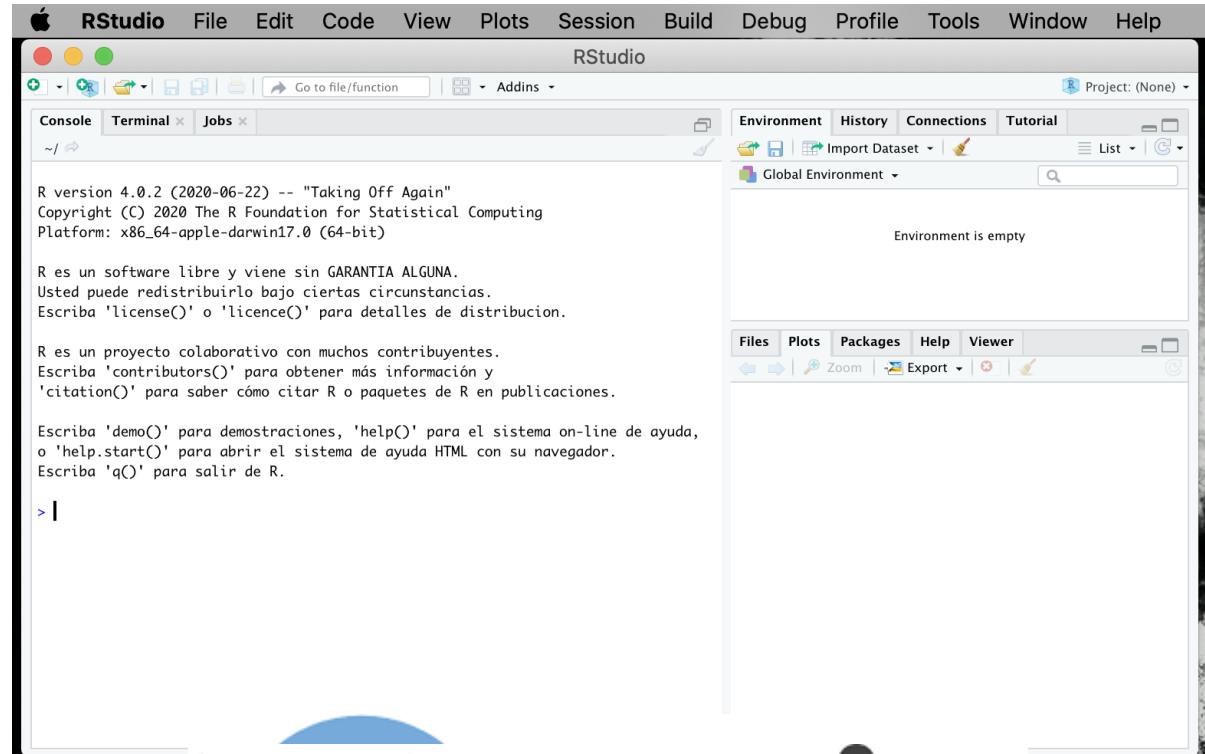
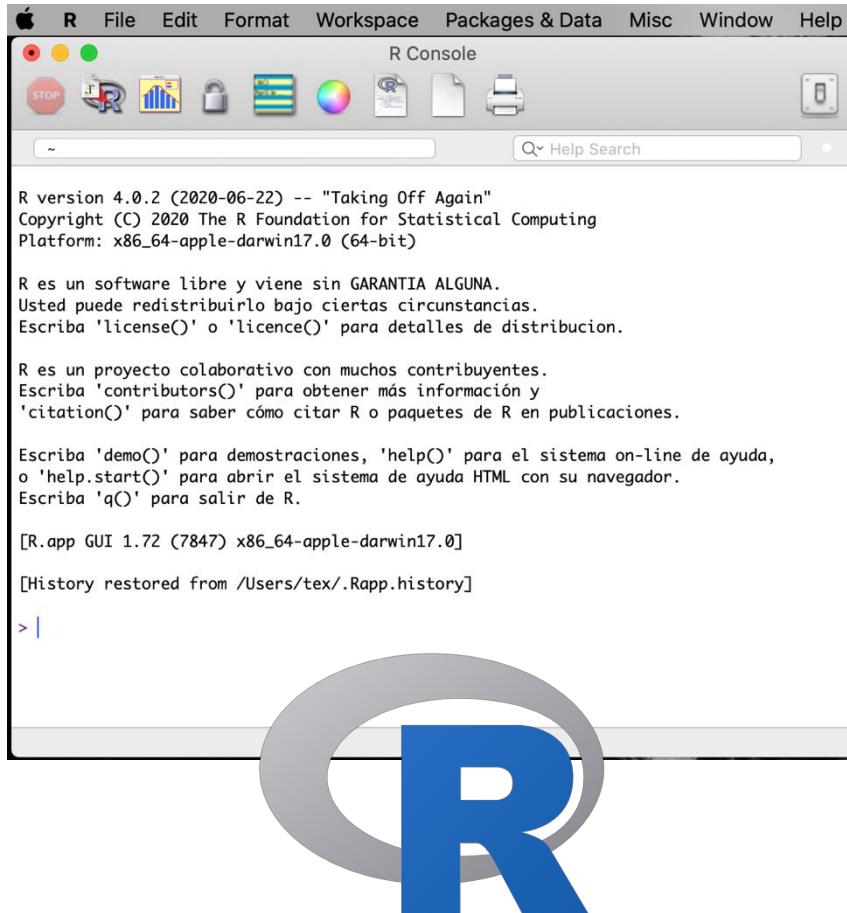
Procesamiento y visualización de datos espaciales en R







Interfaz gráfica





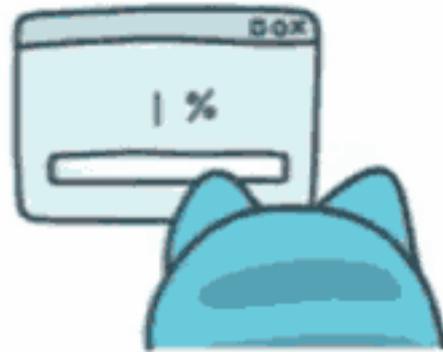
Su turno . . .

Descargue e instale...



Positron

<https://positron.posit.co/install.html>





Consola

- Es útil para ejecutar comandos sencillos que van a ser ejecutados de inmediato.
- Una vez que se abre **R** aparece el prompt **>**, lo que indica que **R** espera la introducción de algún comando.

A screenshot of the Positron IDE interface, specifically the R workspace. The main window title is "Welcome – workspace (Workspace)". On the left, there's a sidebar with icons for New, Open, and a search bar. The central area features the Positron logo and two buttons: "New Notebook" and "New File". To the right, there are sections for "Walkthroughs" (with links to "Migrating from VSC..." and "Migrating from RStudio...") and "Recent" (listing "prueba (Workspace) ~/Documents/r_puebas/prueba"). Below these is the R console output:

```
R version 4.5.1 (2025-06-13) -- "Great Square Root"
Copyright (C) 2025 The R Foundation for Statistical Computing
Platform: arch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

A red arrow points to the first line of the console output, which is the prompt ">". The word "Prompt" is written in red text next to the arrow. At the bottom of the console, it says "Quarto: 1.7.32".



Consola

El uso más básico de **R** sería utilizarlo como una calculadora científica

```
> 3 + 4
```

```
[1] 7
```

```
> sqrt(36)
```

```
[1] 6
```

```
> 3 + 3 * 3 + 3
```

```
[1] 15
```

```
> 10 - 7
```

```
[1] 3
```

```
> seq(1, 5)
```

```
[1] 1 2 3 4 5
```

```
> log(23.4) * 2
```

```
[1] 1.576368
```



Existe una amplia variedad de funciones matemáticas disponibles en **R**

- `log(x)`
- `exp(x)`
- `log(x, n)`
- `log10(x)`
- `sqrt(x)`

- `round(x, digits = 0)`
- `signif(x, digits = 6)`
- `cos(x)`
- `sin(x)`
- `tan(x)`

Todas ejecutables directamente en consola

```
> round(23.63496, 2)
```

[1] 23.63

```
> round(exp(1.744) * sqrt(7.1), 2)
```

[1] 15.24



Jerarquía de funciones

- Potencias
- División
- Multiplicación
- Restas
- Sumas

En caso de haber funciones de la misma jerarquía, entonces **R** las realizará en orden de aparición, de izquierda a derecha.

```
> 2 - 3 + 4 / 2 ^ 8 * 5
```

```
[1] -0.921875
```



En **R** tenemos 6 tipos básicos de datos

- **character**: texto
- **numeric**: números reales (double)
- **integer**: números enteros (3L, 45L)
- **complex**: números complejos (-1+0i)
- **logical**: TRUE/FALSE
- **date**: fechas



Valores faltantes e infinitos

- **NA**: valor faltante (missing)
- **NAN**: not a number (p.e. $0/0$)
- **Inf**: valor infinito positivo
- **-Inf**: valor infinito negativo



Buscando ayuda

```
> ?rep
```

```
> help(rep)
```

- Lo mejor será siempre buscar ayuda en internet
- por ejemplo ¿Cómo calculo la media de un vector?
- Preguntar en inglés puede dar mejores resultados
- [Stack Overflow](#) puede ser muy útil
- [Posit Community](#) puede ser muy útil



Welcome — workspace (Workspace)

+ New | Open | ...

Search

R 4.5.1 | prueba_01

SESSION CONNECTIONS HELP VIEWER

VARIABLES

R 4.5.1 | filter

No variables have been created.

Welcome x

Positron
an IDE for data science from

New Notebook | New File

Walkthroughs

Migrating from VSC... | Migrating from RStu... | More...

Recent

prueba (Workspace) ~/Documents/r_pruebas/prueba

CONSOLE TERMINAL PROBLEMS OUTPUT PORTS DEBUG CONSOLE

~ /Documents/r_pruebas/prueba_01

R 4.5.1 started.

```
R version 4.5.1 (2025-06-13) -- "Great Square Root"
Copyright (C) 2025 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

>

0 △ 0 Quarto: 1.7.32

SCRIPT



Positron File Edit Selection View Go Run Terminal Window Help

New Open

Welcome X

Positron
an IDE for data science from

New Notebook New File

Recent prueba (Workspace) ~/Documents/r_pruebas/prueba

CONSOLE TERMINAL PROBLEMS OUTPUT PORTS DEBUG CONSOLE

~/Documents/r_pruebas/prueba_01

R 4.5.1 started.

```
R version 4.5.1 (2025-06-13) -- "Great Square Root"
Copyright (C) 2025 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale
```

Select File Type or Enter File Name...

Text File Built-In
R File R
Python File Python
Jupyter Notebook .ipynb Support
Quarto Document Quarto
Quarto Project Quarto

LP VIEWER

No variables

Walkthroughs

Migrating from VSC... New

Migrating from RStu... New

More...

PLOTS

A red arrow points to the "Python File Python" option in the "Select File Type or Enter File Name..." dropdown menu.



Untitled-1

New Open

Search

SESSION CONNECTIONS HELP VIEWER

VARIABLES

R 4.5.1

No variables have been created.

filter

En esta ventana capturaremos todo nuestro código

CONSOLE TERMINAL PROBLEMS OUTPUT PORTS DEBUG CONSOLE

~/

```
R version 4.5.1 (2025-06-13) -- "Great Square Root"
Copyright (C) 2025 The R Foundation for Statistical Computing
Platform: aarch64-apple-darwin20

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.
```

>

LN 1, COL 1 Spaces: 2 UTF-8 LF { } R

A screenshot of the Positron IDE interface. The menu bar includes File, Edit, Selection, View, Go, Run, Terminal, Window, and Help. The title bar shows "S02_Texcalac.R". The left sidebar has icons for New, Open, and a search bar. The main pane displays an R script with numbered lines. Red arrows point from labels on the left to specific parts of the code:

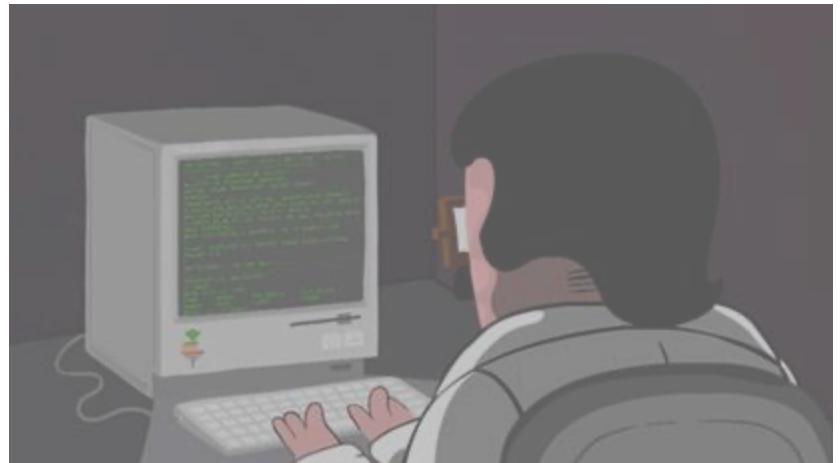
- An arrow points from the label "Encabezado" to the first few lines of the script, which are all comments starting with "#".
- An arrow points from the label "Sección" to line 12, which starts with "###".
- An arrow points from the label "Comentarios" to line 16, which contains a comment "# imprimo en pantalla el resultado almacenado en 'obj1'".
- An arrow points from the label "Sección" to line 20, which starts with "###".
- An arrow points from the label "Espaciado" to line 28, which shows a line of code with significant whitespace before the assignment operator "<-".

```
1 # =====#
2 # Script de la Sesión 02
3 # Autor: Tex
4 # Curso: "Procesamiento y visualización de datos espaciales en R"
5 # Fecha: 11 de septiembre de 2025
6 # =====#
7
8 # Activo los paquetes a utilizar en mi sesión
9 library(tidyverse)
10
11
12 ### Paso 1: Preparo datos ====
13 # Genero objeto de nombre "obj1"
14 obj1 <- 2 + 2
15
16 # imprimo en pantalla el resultado almacenado en "obj1"
17 obj1
18
19
20 ### Paso 2: Analizo datos ====
21 # Genero objeto de nombre "obj2"
22 5 * 7 -> obj2
23
24 # imprimo en pantalla el resultado almacenado en "obj2"
25 obj2
26
27 # Genero objeto de nombre "obj3"
28 obj3 <- obj1 + obj2
29
30 obj3<-obj1+obj2
31
32 # imprimo en pantalla el resultado almacenado en "obj3"
33 obj3
34
```



Su turno...

- Genere un nuevo script
- Agregue el encabezado a su script
- Guarde su script como
“S02_Nombre.R”





John M. Chambers

“Everything that exists in **R** is an object”.

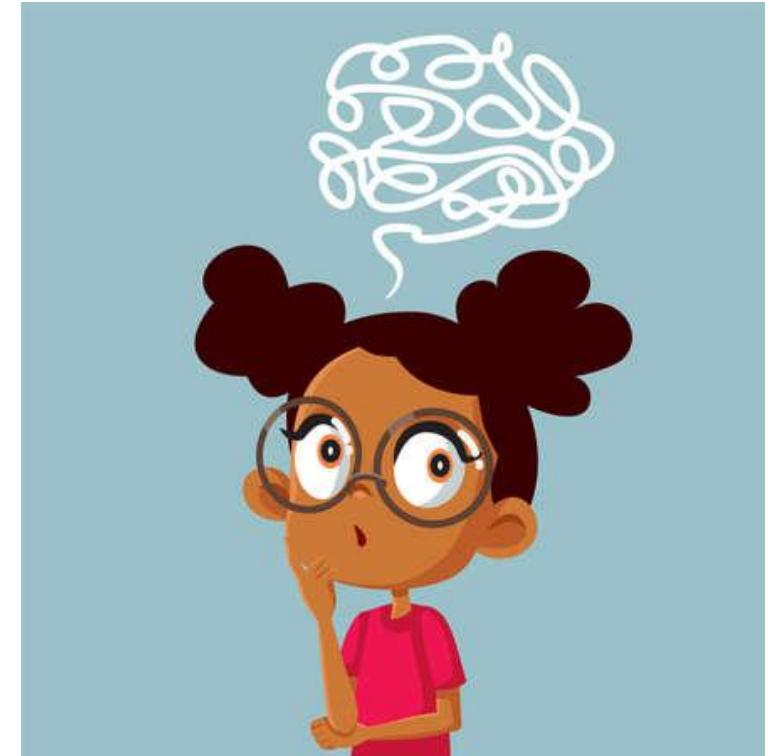
“Everything that happens in **R** is the result of a function call”.





```
función(argumento1, arg2..., na.rm = TRUE)
```

```
función(función(argumento1, arg2..., na.rm = TRUE), arg2..., na.rm = TRUE)
```





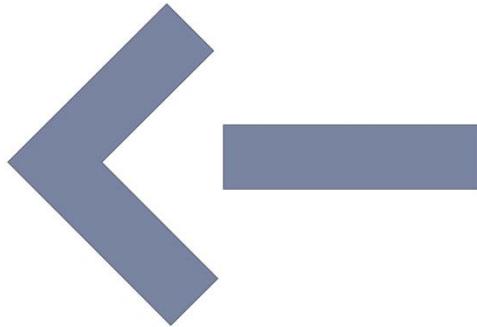
```
mean(x, trim = 0, na.rm = FALSE...)
```

```
round(mean(x, trim = 0, na.rm = FALSE...), digits = 0)
```





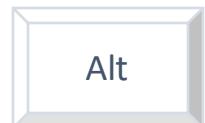
Operador de asignación



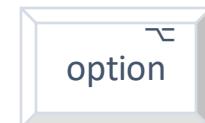
Menor que guión



Windows 10



+



+





Generar objetos

Podemos almacenar un resultado asignándole nombre

```
> obj1 <- 2 + 2 # Note que no se imprime en pantalla
```

Es necesario llamar al objeto para imprimir el resultado en pantalla

```
> obj1
```

```
[1] 4
```

La asignación también funciona en sentido inverso

```
> 5 * 7 -> obj2
```

```
> obj2
```

```
[1] 35
```



Operaciones entre objetos

Se pueden realizar operaciones entre objetos y crear nuevos

```
> obj1
```

```
[1] 4
```

```
> obj2
```

```
[1] 35
```

Cálculos con la información almacenada en un objeto

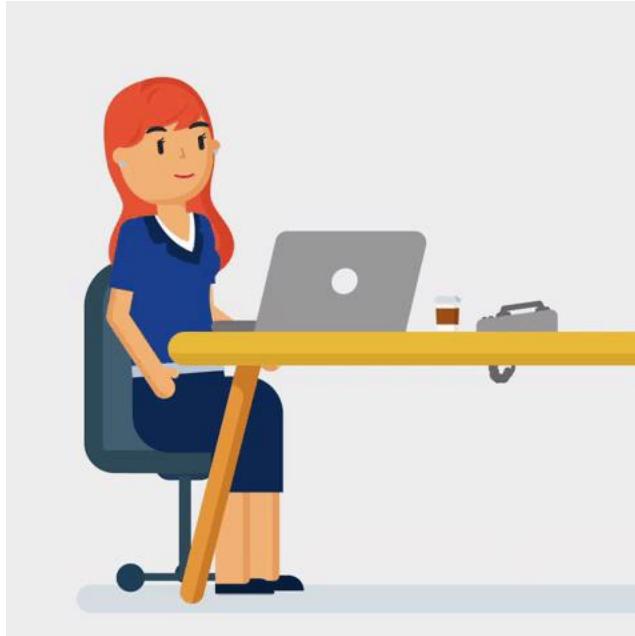
```
> obj3 <- obj1 + obj2
```

```
[1] 39
```



Su turno

- Continue en su script
- Genere tres objetos diferentes y nombrelos como obj4, obj5 y obj6
- Realice operaciones entre los objetos





Buenas prácticas para la redacción de código y nombrado de objetos

Utilice espacios para separar elementos

```
> objeto <- 2 + 2
```



```
> objeto<-2+2
```



No utilice acentos para nombrar objetos

```
> poblacion <- 2 + 2
```



```
> población <- 2 + 2
```



No utilice espacios en el nombre de sus objetos

```
> pob_2021 <- 2 + 2
```



```
> pob 2021 <- 2 + 2
```





Buenas prácticas para la redacción de código y nombrado de objetos

Revisar que cada bloque de código se ejecute correctamente

```
> objeto_1 <- c("Juan", "Pablo", "Pedro")
```

```
objeto_2 <- 2 + 2
```

Evite nombrar objetos con una sola letra

```
> w <- 2 + 2
```

Evite utilizar nombres de comandos existentes o funciones

```
> mean <- 2 + 2
```

```
> log <- 2 + 2
```

```
> sqrt <- 2 + 2
```



S02_Texcalac.R

+ New | Open | ... | Search

S02_Texcalac.R ●

Users > tex > Desktop > S02_Texcalac.R > Paso 2: Analizo datos

```
5 # Fecha: 11 de septiembre de 2025
6 # =====#
7
8 # Activo los paquetes a utilizar en mi sesión
9 library(tidyverse)
10
11
12 ### Paso 1: Preparo datos ====
13 # Genero objeto de nombre "obj1"
14 obj1 <- 2 + 2
15
16 # imprimo en pantalla el resultado almacenado en "obj1"
17 obj1
18
19
20 ### Paso 2: Analizo datos ====
21 # Genero objeto de nombre "obj2"
22 5 * 7 -> obj2
23
24 # imprimo en pantalla el resultado almacenado en "obj2"
25 obj2
26
27 # Genero objeto de nombre "obj3"
28 obj3 <- obj1 + obj2
29
30 obj3<-obj1+obj2
31
32 # imprimo en pantalla el resultado almacenado en "obj3"
33 obj3
34
35
36
37
```

CONSOLE TERMINAL PROBLEMS OUTPUT PORTS DEBUG CONSOLE

~/

```
[1] 35
> obj3 <- obj1 + obj2
>
```

SESSION CONNECTIONS HELP VIEWER

VARIABLES

R 4.5.1

VALUES

	obj1	4	dbl
obj2	35		dbl
obj3	39		dbl

filter

PLOTS

Variables

Ln 35, Col 1 Spaces: 2 UTF-8 LF { } R



Remover objetos

```
> rm(obj1)
```

```
> rm(obj1, obj2)
```

Eliminar todos los objetos del espacio de trabajo ¡Cuidado!

```
> rm(list = ls())
```

- Si borré todo, tendré que ejecutar de nuevo el script
- Si no tengo script entonces perdí mi trabajo (revisar historial)



1 ERRORES

En un error R dice "NO, no haré lo que me pides porque no puedo llevar a cabo la orden". Y el mensaje de error te dice por qué no se puede ejecutar y en qué fragmento del código debes buscar el fallo.

```
> y[1]  
Error: objeto 'y' no encontrado
```

El problema que lo causa
El código que lo produjo

2 ADVERTENCIAS

Las advertencias significan "OK, haré lo que me pides, pero tal vez no te guste lo que vas a obtener". Es la forma en que R te dice que el código se comporta de una manera diferente a la que se podrías esperar.

```
> data(airquality)  
> ggplot(airquality, aes(x = Solar.R, y = Ozone)) +  
+   geom_point()  
Warning message:  
Removed 42 rows containing missing values (geom_point).
```

El problema que lo causa
El comportamiento que se advierte
El código que lo produjo

3 MENSAJES

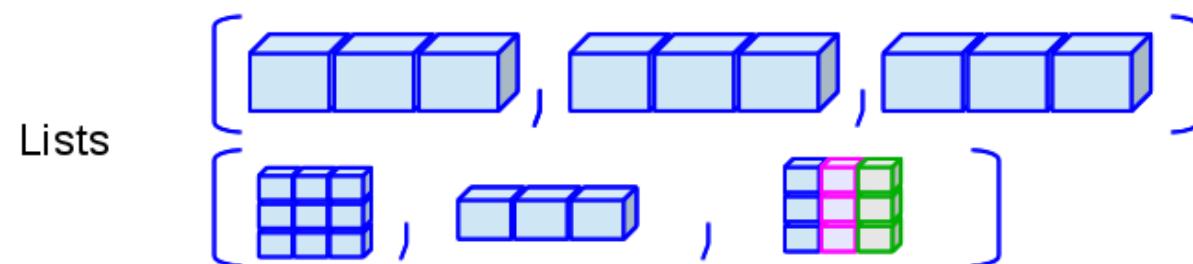
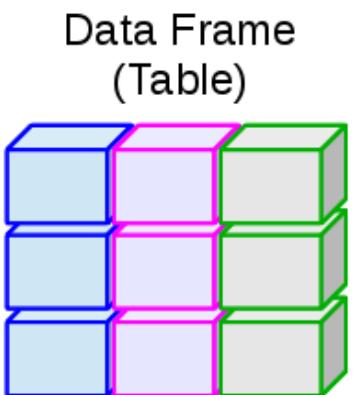
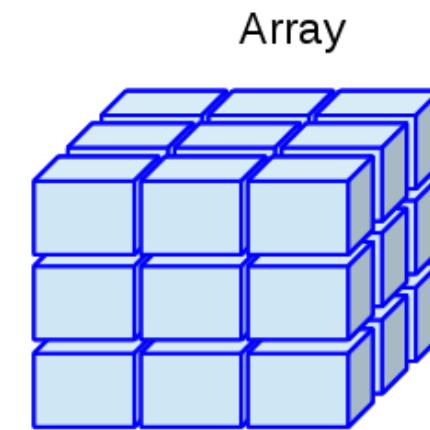
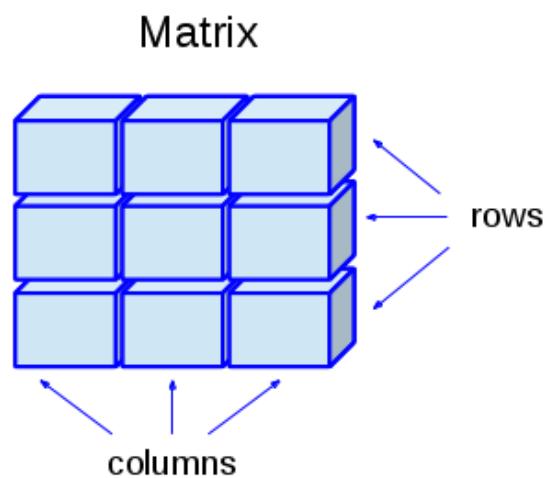
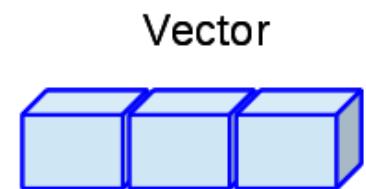
Cuando R te muestra un mensaje que no comienza con "Error" ni con "Warning", entonces solo indica que se han realizado algunas acciones en tu nombre. No impiden que el código funcione ni se trata de un comportamiento inesperado.

```
> library(vegan)  
Loading required package: permute  
Loading required package: lattice  
This is vegan 2.5-6
```

Solo información



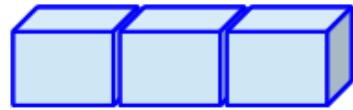
Tipos de datos





Vectores

Vector



Nombre
Brayan
Kimberly
Kevin
Britany
Brandon
Jenifer
Iker
Nicole

Edad
16
14
15
17
15
16
15
18

Altura
167
158
171
154
165
161
166
159

Peso
69.4
59.6
74.3
55.1
69.8
57.4
64.1
61.1

```
nombre <- c("Brayan", "Kimberly", "Kevin", "Britany", "Brandon", "Jenifer", "Iker", "Nicole")
```

```
edad <- c(16L, 14L, 15L, 17L, 15L, 16L, 15L, 18L)
```

```
altura <- c(167, 158, 171, 154, 165, 161, 166, 159)
```

```
peso <- c(69.4, 59.6, 74.3, 55.1, 69.8, 57.4, 64.1, 61.1)
```



Generación de vectores

```
> seq(7, 16)
```

```
[1] 7 8 9 10 11 12 13 14 15 16
```

```
> rep(1:3, times = 3, each = 2)
```

```
[1] 1 1 2 2 3 3 1 1 2 2 3 3 1 1 2 2 3 3
```

```
> letters[7:13]
```

```
[1] "g" "h" "i" "j" "k" "l" "m"
```

```
> rnorm(15, mean = 0, sd = 1)
```

```
[1] 0.2719539 -0.6148791 -0.2768793 -1.1126266 -1.9346567 0.5766766  
[7] 0.4586722 -0.9866584 0.4151321 0.2463699 2.9209443 -0.2661786  
[13] 0.1429584 -1.0427412 1.2250898
```



Generación de vectores

```
> c("a1", "a2", "a3", "a4", "a5")
```

```
[1] "a1" "a2" "a3" "a4" "a5"
```

```
> c(3L, 7L, 9L, 15L, 12L)
```

```
[1] 3 7 9 15 12
```

```
> rep(c("a1", "a2", "a3"), times = 3, each = 2)
```

```
[1] "a1" "a1" "a2" "a2" "a3" "a3" "a1" "a1" "a2" "a2" "a3" "a3" "a1" "a1"  
[15] "a2" "a2" "a3" "a3"
```

```
> sample(1:50, 5)
```

```
[1] 5 48 7 31 38
```



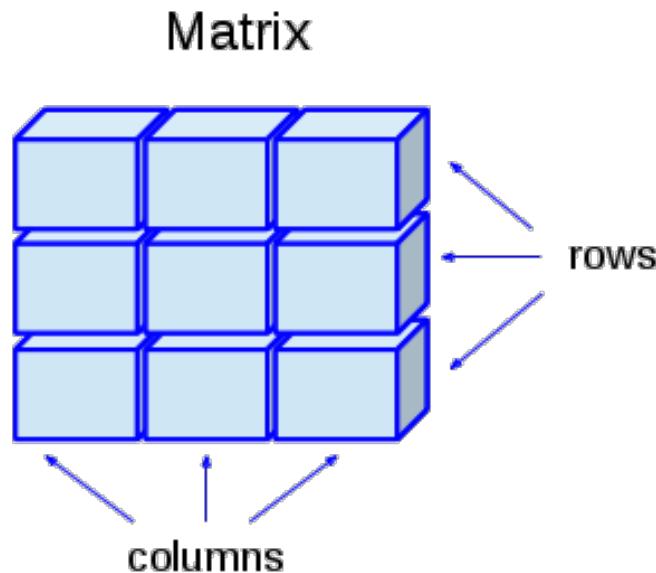
Su turno...

- Continue en su script
- Genere los siguientes vectores con 10 elementos cada uno y agregue algún valor NA, NaN, Inf o -Inf
 - nombre
 - peso
 - años cumplidos (edad)
 - estado civil (edo_civ)
 - titulado (TRUE, FALSE)





Matrices



```
> matrix(data, nrow, ncol , byrow = F)
```

```
> matrix(1:15, nrow = 3, byrow = T )
```

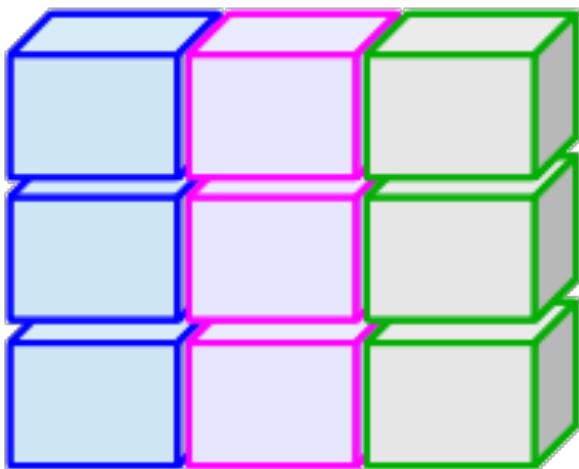
```
[,1] [,2] [,3] [,4] [,5]
[1,] 1 2 3 4 5
[2,] 6 7 8 9 10
[3,] 11 12 13 14 15
```

```
[,1] [,2] [,3] [,4]
[1,] "Brayan" "16" "167" "69.4"
[2,] "Kimberly" "14" "158" "59.6"
[3,] "Kevin" "15" "171" "74.3"
[4,] "Britany" "17" "154" "55.1"
[5,] "Brandon" "15" "165" "69.8"
[6,] "Jenifer" "16" "161" "57.4"
[7,] "Yovani" "15" "166" "64.1"
[8,] "Nicole" "18" "159" "61.1"
```



Data frame (“bases de datos”)

**Data Frame
(Table)**



Nombre	Edad	Altura	Peso
Brayan	16	167	69.4
Kimberly	14	158	59.6
Kevin<	15	171	74.3
Britany	17	154	55.1
Brandon	15	165	69.8
Jenifer	16	161	57.4
Iker	15	166	64.1
Nicole	18	159	61.1



Data Editor (Browse) - [auto.dta]

File Edit View Data Tools

make[1] AMC Concord

	make	price	mpg	rep78	headroom
1	AMC Concord	4,099	22	3	.
2	AMC Pacer	4,749	17	3	.
3	AMC Spirit	3,799	22	.	.
4	Buick Century	4,816	20	3	.
5	Buick Electra	7,827	15	4	.
6	Buick LeSabre	5,788	18	3	.
7	Buick Opel	4,453	26	.	.
8	Buick Regal	5,189	20	3	.
9	Buick Riviera	10,372	16	3	.
10	Buick Skylark	4,082	19	3	.
11	Cad. Deville	11,385	14	3	.
12	Cad. Eldorado	14,500	14	2	.
13	Cad. Seville	15,906	21	3	.
14	Chev. Chevette	3,299	29	3	.

Variables

Filter variables here

<input checked="" type="checkbox"/> Variable	<input type="checkbox"/> Label
<input checked="" type="checkbox"/> make	Make and Model
<input checked="" type="checkbox"/> price	Price
<input checked="" type="checkbox"/> mpg	Mileage (mpg)
<input checked="" type="checkbox"/> rep78	Repair Record 1978
<input checked="" type="checkbox"/> headroom	Headroom (in.)

Ready Vars: 12 Order: Dataset Obs: 74 Filter Off Mode: Browse

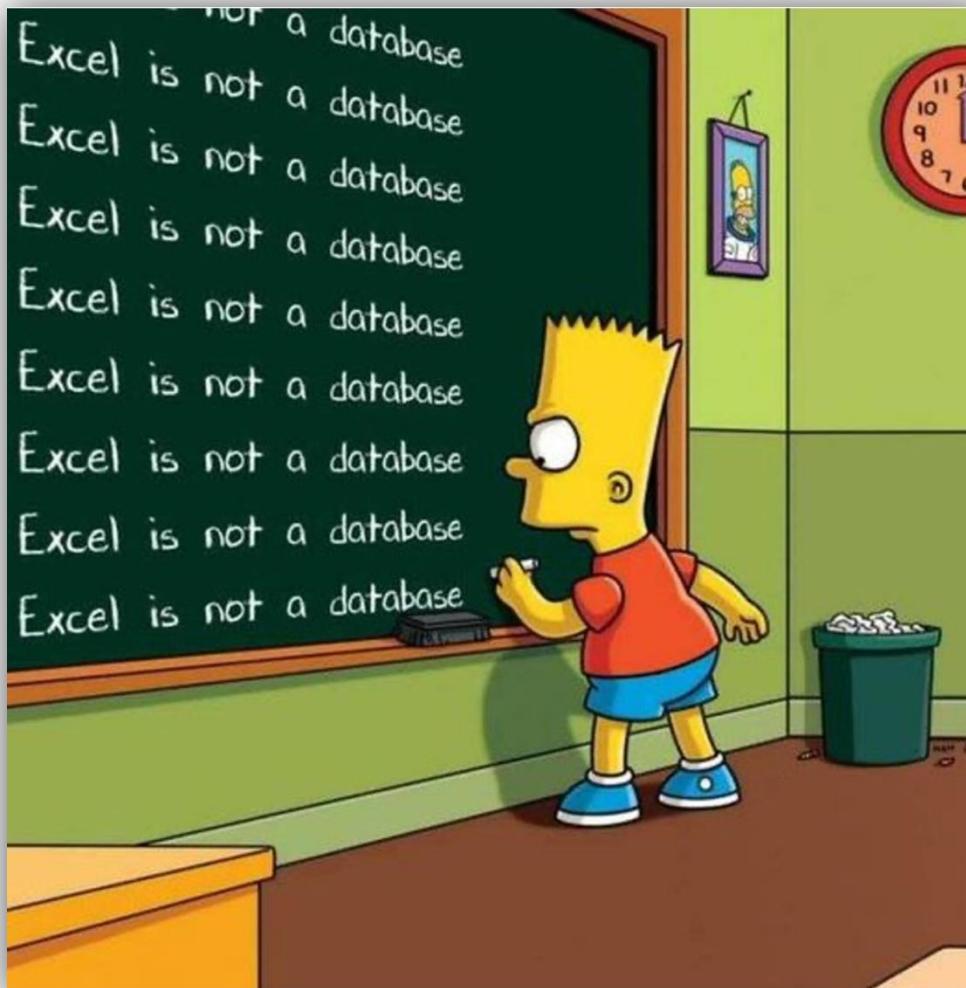
A small, semi-transparent image of former US President Barack Obama is positioned in the bottom right corner of the software window. He is shown from the chest up, wearing a dark suit and tie, with his hands raised as if gesturing during a speech.



Base de datos

- Una base de datos es una colección de información organizada de tal forma que un programa de computadora pueda seleccionar rápidamente los fragmentos de datos que necesite
- Una base de datos es un sistema de archivos electrónico.
- Las bases de datos tradicionales se organizan por campos, registros y archivos.







PubMed.gov

Search PubMed

Search

Advanced Create alert Create RSS User Guide

Found 1 result for *Gene name errors are widespread in the scie...*

Save Email Send to Display options

> [Genome Biol.](#) 2016 Aug 23;17(1):177. doi: 10.1186/s13059-016-1044-7.

Gene name errors are widespread in the scientific literature

Mark Ziemann ¹, Yotam Eren ^{1 2}, Assam El-Osta ^{3 4}

Affiliations + expand

PMID: 27552985 PMCID: [PMC4994289](#) DOI: [10.1186/s13059-016-1044-7](#)

[Free PMC article](#)

Abstract

The spreadsheet software Microsoft Excel, when used with default settings, is known to convert gene names to dates and floating-point numbers. A programmatic scan of leading genomics journals reveals that approximately one-fifth of papers with supplementary Excel gene lists contain erroneous gene name conversions.

Keywords: Gene symbol; Microsoft Excel; Supplementary data.

Figures

FULL TEXT LINKS

Read free full text at [BMC](#)

PMC **FREE** Full text

ACTIONS

Cite

Favorites

SHARE

[Twitter](#) [Facebook](#) [Email](#)

PAGE NAVIGATION

>Title & authors

Abstract



STREAMING NEWS & INSIGHT

From Vulture's Joe Adalian

GET THE
NEWSLETTER ➞

VULTURE
BUFFERING

MICROSOFT REPORT SCIENCE

Scientists rename human genes to stop Microsoft Excel from misreading them as dates

Sometimes it's easier to rewrite genetics than update Excel

By James Vincent | Aug 6, 2020, 8:44am EDT

- **Symbols that affect data handling and retrieval.** For example, all symbols that autoconverted to dates in Microsoft Excel have been changed (for example, SEPT1 is now SEPTIN1; MARCH1 is now MARCHF1); tRNA synthetase symbols that were also common words have been changed (for example, WARS is now WARS1; CARS is now CARS1).



Support The Guardian

Available for everyone, funded by readers

[Contribute →](#)

[Subscribe →](#)

Search jobs

[Sign in](#)



News

Opinion

Sport

Culture

Lifestyle

More ▾

UK ► UK politics

Education Media Society Law Scotland Wales Northern Ireland

Health policy

Covid: how Excel may have caused loss of 16,000 test results in England

Public Health England data error blamed on limitations of Microsoft spreadsheet

● [Coronavirus - latest updates](#)

● [See all our coronavirus coverage](#)



RESEARCH ARTICLE

Gene name errors: Lessons not learned

Mandhri Abeysooriya , Megan Soria , Mary Savvy Kasu , Mark Ziemann *^{*}

Deakin University, School of Life and Environmental Sciences, Geelong, Australia

* m.ziemann@deakin.edu.au

Abstract



OPEN ACCESS

Citation: Abeysooriya M, Soria M, Kasu MS, Ziemann M (2021) Gene name errors: Lessons not learned. PLoS Comput Biol 17(7): e1008984. <https://doi.org/10.1371/journal.pcbi.1008984>

Editor: Christos A. Ouzounis, CPERI, GREECE

Received: May 4, 2021

Accepted: July 1, 2021

Published: July 30, 2021

Author summary

Autocorrection is a feature of modern softwares including messaging apps, word processors and spreadsheets. These are designed to avoid data entry errors but “autocorrect”



Book1 - Microsoft Excel

The screenshot shows a Microsoft Excel spreadsheet titled "Book1 - Microsoft Excel". The data is organized into columns A through P, with rows numbered from 61 to 83. Column A contains car models, and column L contains foreign exchange rates. The data includes the following entries:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
61	Fiat Strada	4296	21	3	2.5	16	2130	161	36	105	3.37	Foreign				
62	Honda Accord	5799	25	5	3	10	2240	172	36	107	3.05	Foreign				
63	Honda Civic	4499	28	4	2.5	5	1760	149	34	91	3.3	Foreign				
64	Mazda GLC	3995	30	4	3.5	11	1980	154	33	86	3.73	Foreign				
65	Peugeot 604	12990	14		3.5	14	3420	192	38	163	3.58	Foreign				
66	Renault Le Car	3895	26	3	3	10	1830	142	34	79	3.72	Foreign				
67	Subaru	3798	35	5	2.5	11	2050	164	36	97	3.81	Foreign				
68	Toyota Celica	5899	18	5	2.5	14	2410	174	36	134	3.06	Foreign				
69	Toyota Corolla	3748	31	5	3	9	2200	165	35	97	3.21	Foreign				
70	Toyota Corona	5719	18	5	2	11	2670	175	36	134	3.05	Foreign				
71	VW Dasher	7140	23	4	2.5	12	2160	172	36	97	3.74	Foreign				
72	VW Diesel	5397	41	5	3	15	2040	155	35	90	3.78	Foreign				
73	VW Rabbit	4697	25	4	3	15	1930	155	35	89	3.78	Foreign				
74	VW Scirocco	6850	25	4	2	16	1990	156	36	97	3.78	Foreign				
75	Volvo 260	11995	17	5	2.5	14	3170	193	37	163	2.98	Fore	n			
76																
77																
78																
79																
80																
81																
82																
83																

Sheet1 Sheet2 Sheet3

Ready Count: 12 100%

Start Recording... Microsoft Excel - B... Stata/SE 12.1 - [Res... STATA 12



Generando un data frame

```
> mi_df <- data.frame(nombre, peso, edad, edo_civ, titulado)  
> mi_df
```

```
> as.matrix(my_df)
```

Nombre	Edad	Altura	Peso
Brayan	16	167	69.4
Kimberly	14	158	59.6
Kevin	15	171	74.3
Britany	17	154	55.1
Brandon	15	165	69.8
Jenifer	16	161	57.4
Iker	15	166	64.1
Nicole	18	159	61.1

```
nombre edad altura peso  
1 Brayan 16 167 69.4  
2 Kimberly 14 158 59.6  
3 Kevin 15 171 74.3  
4 Britany 17 154 55.1  
5 Brandon 15 165 69.8  
6 Jenifer 16 161 57.4  
7 Iker 15 166 64.1  
8 Nicole 18 159 61.1
```



Explore su data frame

```
> class(mi_df)
```

```
[1] "data.frame"
```

```
> is(mi_df)
```

```
[1] "data.frame" "list"      "oldClass"   "vector"
```

```
> str(mi_df)
```

```
'data.frame': 8 obs. of 4 variables:  
 $ nombres: chr  "Brayan" "Kimberly" "Kevin" "Britany" ...  
 $ edad    : num  16 14 15 17 15 16 15 18  
 $ altura  : num  167 158 171 154 165 161 166 159  
 $ peso    : num  69.4 59.6 74.3 55.1 69.8 57.4 64.1 61.1
```



Explore su data frame

```
> mean(mi_df$edad, na.rm = TRUE)
```

```
[1] 15.75
```

```
> sd(mi_df$edad, na.rm = TRUE)
```

```
[1] 1.28
```

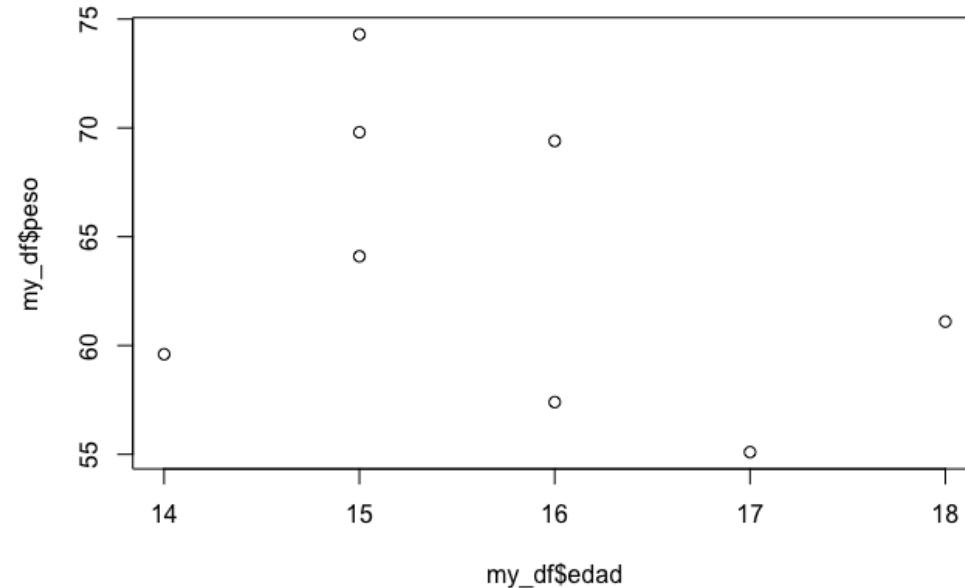
```
> summary(mi_df)
```

	nombres	edad	altura	peso
Length:	8	Min. :14.00	Min. :154.0	Min. :55.10
Class :	character	1st Qu.:15.00	1st Qu.:158.8	1st Qu.:59.05
Mode :	character	Median :15.50	Median :163.0	Median :62.60
		Mean :15.75	Mean :162.6	Mean :63.85
		3rd Qu.:16.25	3rd Qu.:166.2	3rd Qu.:69.50
		Max. :18.00	Max. :171.0	Max. :74.30



Gráfico básico

```
> plot(mi_df$edad, mi_df$peso)
```





Data frames instalados por defecto

- **R** y algunos paquetes tienen e instalan por defecto algunas mallas de datos para el análisis de datos.

```
> paquetes <- library(help = "datasets")
```

```
> head(paquetes$info[[2]])
```

[1] "AirPassengers	Monthly Airline Passenger Numbers 1949-1960"
[2] "Bjsales	Sales Data with Leading Indicator"
[3] "BOD	Biochemical Oxygen Demand"
[4] "CO2	Carbon Dioxide Uptake in Grass Plants"
[5] "ChickWeight	Weight versus age of chicks on different diets"
[6] "Dnase	Elisa assay of DNase"



Ejemplos de dataframes instalados por defecto

> AirPassengers

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1949	112	118	132	129	121	135	148	148	136	119	104	118
1950	115	126	141	135	125	149	170	170	158	133	114	140
1951	145	150	178	163	172	178	199	199	184	162	146	166
1952	171	180	193	181	183	218	230	242	209	191	172	194
1953	196	196	236	235	229	243	264	272	237	211	180	201
1954	204	188	235	227	234	264	302	293	259	229	203	229
1955	242	233	267	269	270	315	364	347	312	274	237	278
1956	284	277	317	313	318	374	413	405	355	306	271	306
1957	315	301	356	348	355	422	465	467	404	347	305	336
1958	340	318	362	348	363	435	491	505	404	359	310	337
1959	360	342	406	396	420	472	548	559	463	407	362	405
1960	417	391	419	461	472	535	622	606	508	461	390	432

> head(CO2, 12)

	Plant	Type	Treatment	conc	uptake
1	Qn1	Quebec	nonchilled	95	16.0
2	Qn1	Quebec	nonchilled	175	30.4
3	Qn1	Quebec	nonchilled	250	34.8
4	Qn1	Quebec	nonchilled	350	37.2
5	Qn1	Quebec	nonchilled	500	35.3
6	Qn1	Quebec	nonchilled	675	39.2
7	Qn1	Quebec	nonchilled	1000	39.7
8	Qn2	Quebec	nonchilled	95	13.6
9	Qn2	Quebec	nonchilled	175	27.3
10	Qn2	Quebec	nonchilled	250	37.1
11	Qn2	Quebec	nonchilled	350	41.8
12	Qn2	Quebec	nonchilled	500	40.6





Instalar paquetes

The Comprehensive R Archive Network (CRAN)

Available CRAN Packages By Name

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

[A3](#)

[aaSEA](#)

[AATtools](#)

[ABACUS](#)

[abbyyR](#)

[abc](#)

[abc.data](#)

[ABC.RAP](#)

[abcADM](#)

[ABCanalysis](#)

[abcdeFBA](#)

[ABCoptim](#)

[ABCp2](#)

[abcf](#)

[abcrda](#)

[abctools](#)

[abd](#)

[abdiv](#)

[abe](#)

[abf2](#)

[ABHgenotypeR](#)

Accurate, Adaptable, and Accessible Error Metrics for Predictive Models

Amino Acid Substitution Effect Analyser

Reliability and Scoring Routines for the Approach-Avoidance Task

Apps Based Activities for Communicating and Understanding Statistics

Access to Abbyy Optical Character Recognition (OCR) API

Tools for Approximate Bayesian Computation (ABC)

Data Only: Tools for Approximate Bayesian Computation (ABC)

Array Based CpG Region Analysis Pipeline

Fit Accumulated Damage Models and Estimate Reliability using ABC

Computed ABC Analysis

ABCDE_FBA: A-Biologist-Can-Do-Everything of Flux Balance Analysis with this package

Implementation of Artificial Bee Colony (ABC) Optimization

Approximate Bayesian Computational Model for Estimating P2

Approximate Bayesian Computation via Random Forests

Asymptotically Bias-Corrected Regularized Linear Discriminant Analysis

Tools for ABC Analyses

The Analysis of Biological Data

Alpha and Beta Diversity Measures

Augmented Backward Elimination

Load Gap-Free Axon ABF2 Files

Easy Visualization of ABH Genotypes

https://cran.r-project.org/web/packages/available_packages_by_name.html



<https://rsanchezs.gitbooks.io/ciencia-de-datos-con-r/content/paquetes/paquetes.html>



Instalar paquetes desde el CRAN

- Ampliar las capacidades del software requiere la instalación de paquetes adicionales
- El paquete sólo se instala una vez
- No es necesario instalar en cada sesión el paquete

Estructura básica para la instalación de paquetes

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

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

```
> install.packages("remotes")
```

Generalmente con esto es suficiente



Instalar paquetes desde Github

- No todos los paquetes se hospedan en CRAN.
- Muchos de ellos son rutinas específicas para problemas concretos.
- Puede ser y es muy útil para ampliar las capacidades del programa.
- Los paquetes `remotes` y `devtools` nos permite instalar paquetes de GitHub.

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

```
> library(remotes)
```

```
> install_github("sctyner/memer")
```



¡Importante!

- Al momento de instalar un paquete puede aparecerle el mensaje que se muestra en la imagen
- Siempre seleccionar la opción “CRAN packages only” (identificar el número que corresponda)

These packages have more recent versions available.
It is recommended to update all of them.
Which would you like to update?

- 1: All
- 2: CRAN packages only
- 3: None
- 4: rlang (1.0.2 -> 1.0.3) [CRAN]
- 5: rprojroot (2.0.2 -> 2.0.3) [CRAN]
- 6: waldo (0.3.1 -> 0.4.0) [CRAN]
- 7: ps (1.7.0 -> 1.7.1) [CRAN]
- 8: processx (3.5.3 -> 3.6.1) [CRAN]
- 9: pkgload (1.2.4 -> 1.3.0) [CRAN]



Su turno

```
> library(memer)
```

```
> meme_get("BatmanRobin") %>%
  meme_text_batman(batman = "R",
                    robin = "Excel/Stata",
                    size = 40)
```

```
> install_github("texcalac/addinsTex")
```

