Análisis de datos con R

Cristóbal Honores

Descripción del curso

- Este curso corresponde a una introducción al uso de la herramienta R, con énfasis en el análisis de datos químicos en base a conceptos básicos tanto del programa, lenguaje, la estadística y gráficos que informen de manera adecuada los resultados obtenidos.

Objetivos

1. Conocer el programa R y sus diferentes ambientes.
2. Lograr importar diversos tipos de bases de datos, proveniente de diferentes equipos.
3. Comparar dos o más procesos.
4. Generar gráficos que informen de manera más óptima los resultados obtenidos.
5. Conocer métodos de calibración multivariada.
6. Realizar control de calidad utilizando el programa R.

Material del curso.

- **R Studio Cloud:**

Se compartirá todo el proyecto realizado en la clases.

- **Github:**

https://github.com/CristobalHonores/Analisis-de-datos-quimicos-con-R

Propósito de esta clase.

Aprender las bases de programación para pode realizar un buen análisis de diferentes base de datos utilizando el programa R.

Introducción a R, Estadística descriptiva y visualización de datos.

1. Introducción a R, R Studio y R Studio Cloud.
2. Objetos y estructuras básicas de R.
3. Importar bases de datos.
4. Selección de filas y columnas.
5. Crear filtros.
6. Recodificar bases de datos.
7. Paquete: dplyr

¿Qué es R?

R es un entorno de software libre para computación estadística y gráficos.

¿Por qué usar R?

1. *Es un lenguaje que permite manipular los datos rápidamente y de forma precisa.*

2. *Puede leer prácticamente cualquier tipo de datos.*

3. *Tiene capacidades avanzadas de gráficos.*

4. *Se puede automatizar fácilmente.*

5. *Es gratuito.*

¿Dónde se baja R?

- ***Microsoft Windows:***

  http://cran.r-project.org/bin/windows/base/

- ***OSX:***

  http://cran.r-project.org/bin/macosx/

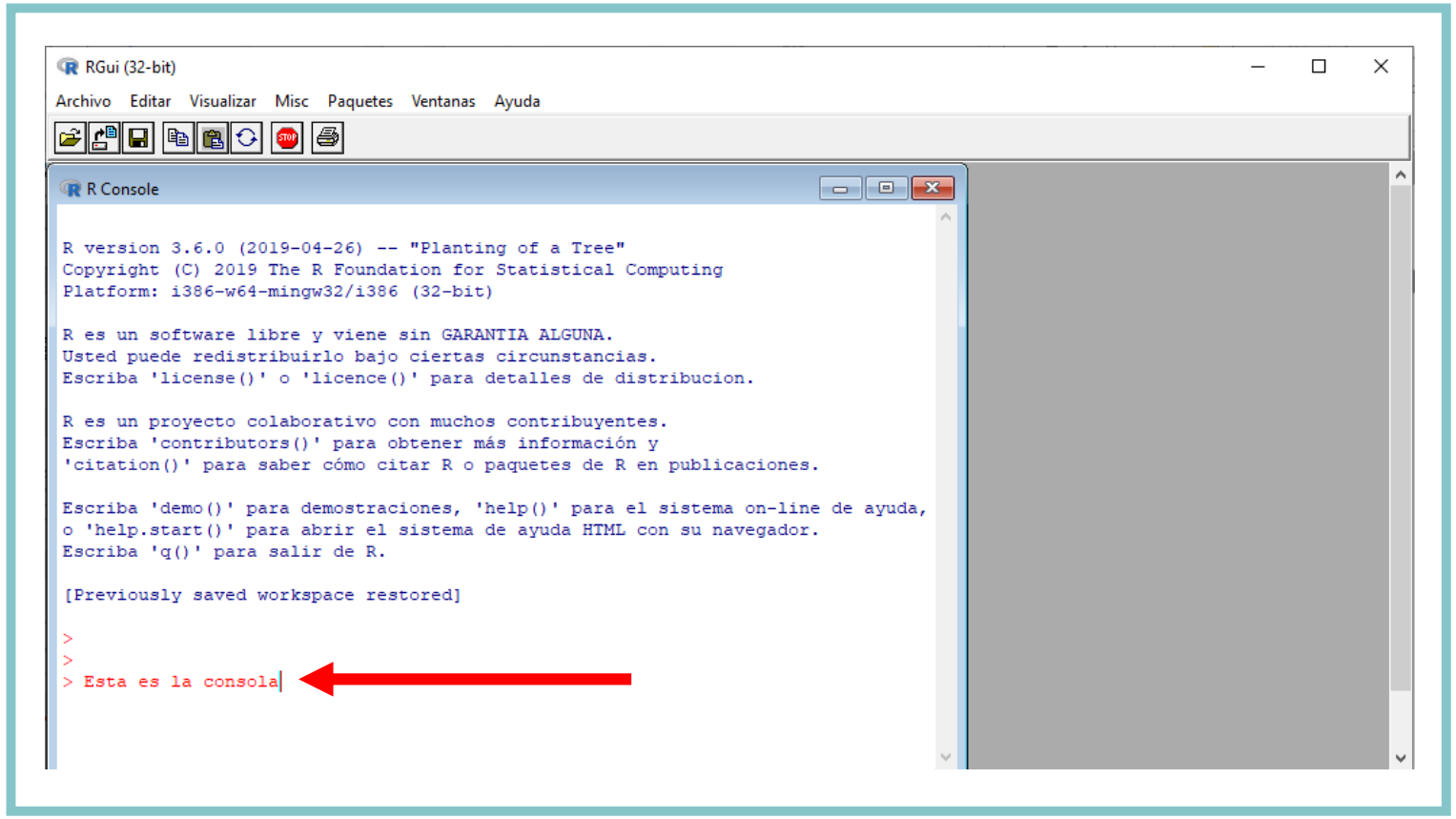
- ***Linux:***

  http://cran.r-project.org/bin/linux/

Conociendo R

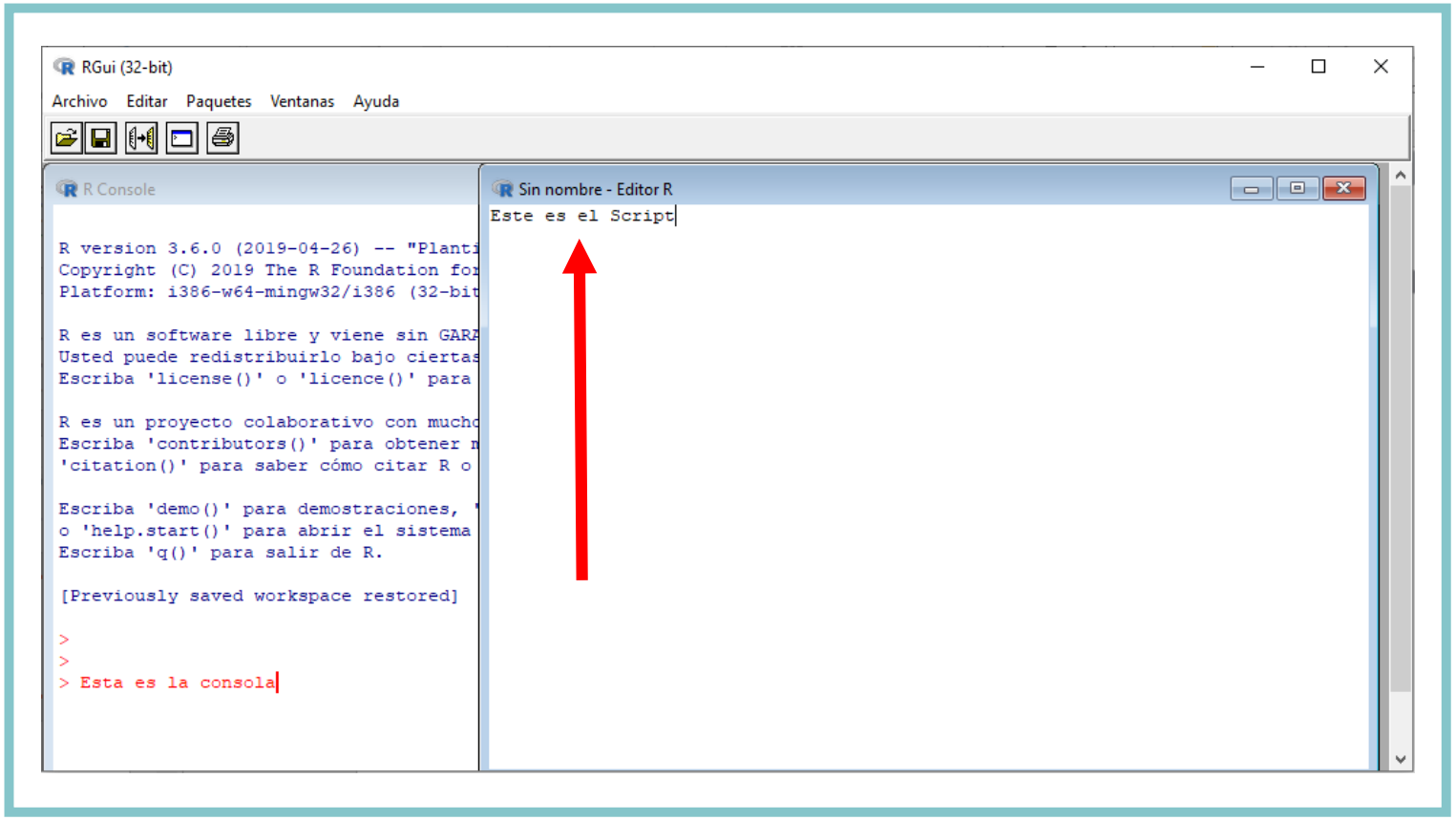
La consola:

Es donde se ejecutan todos los comandos y códigos.



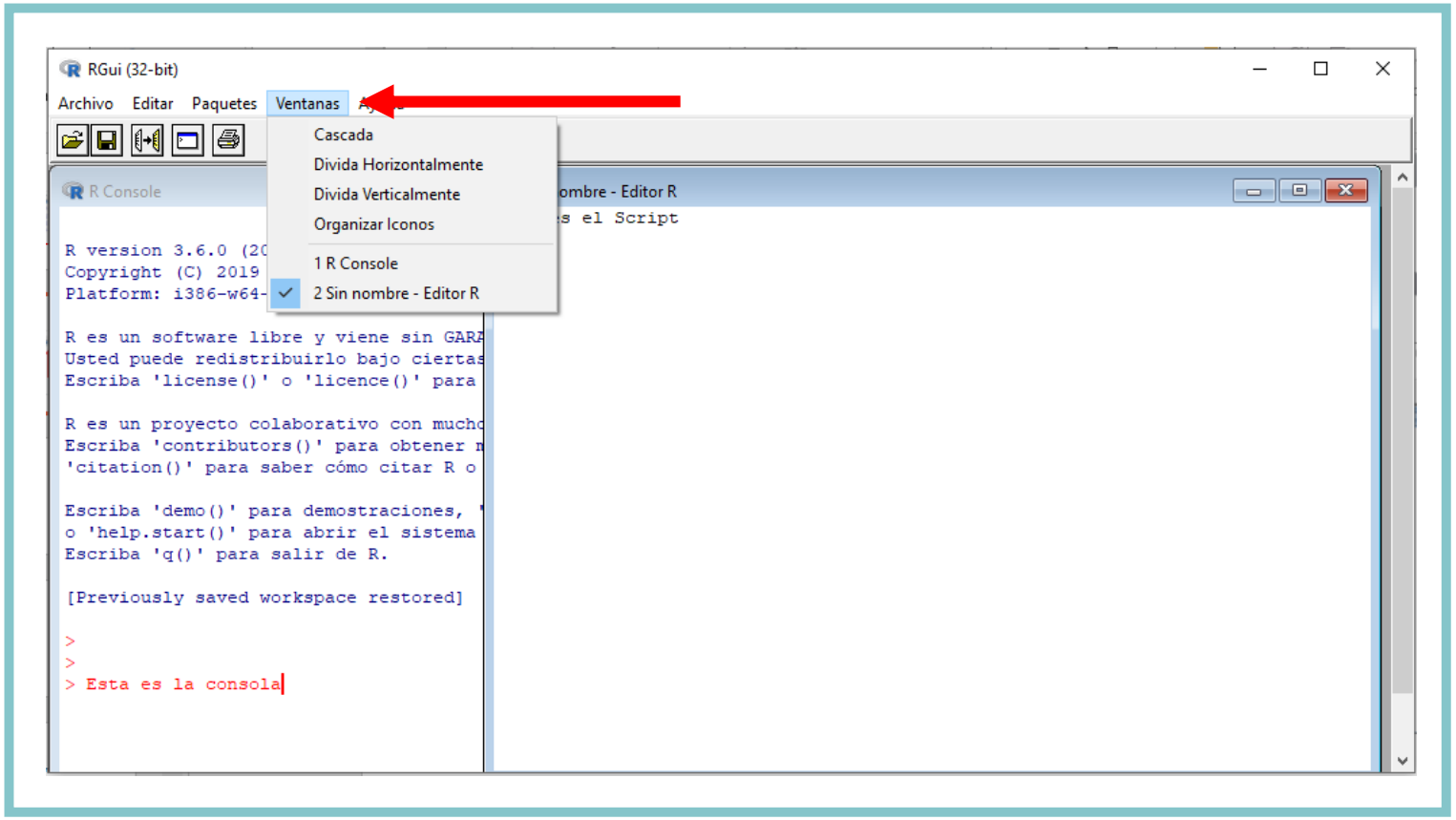
El Script:

Aquí se escriben los códigos d manera mas ordenado.

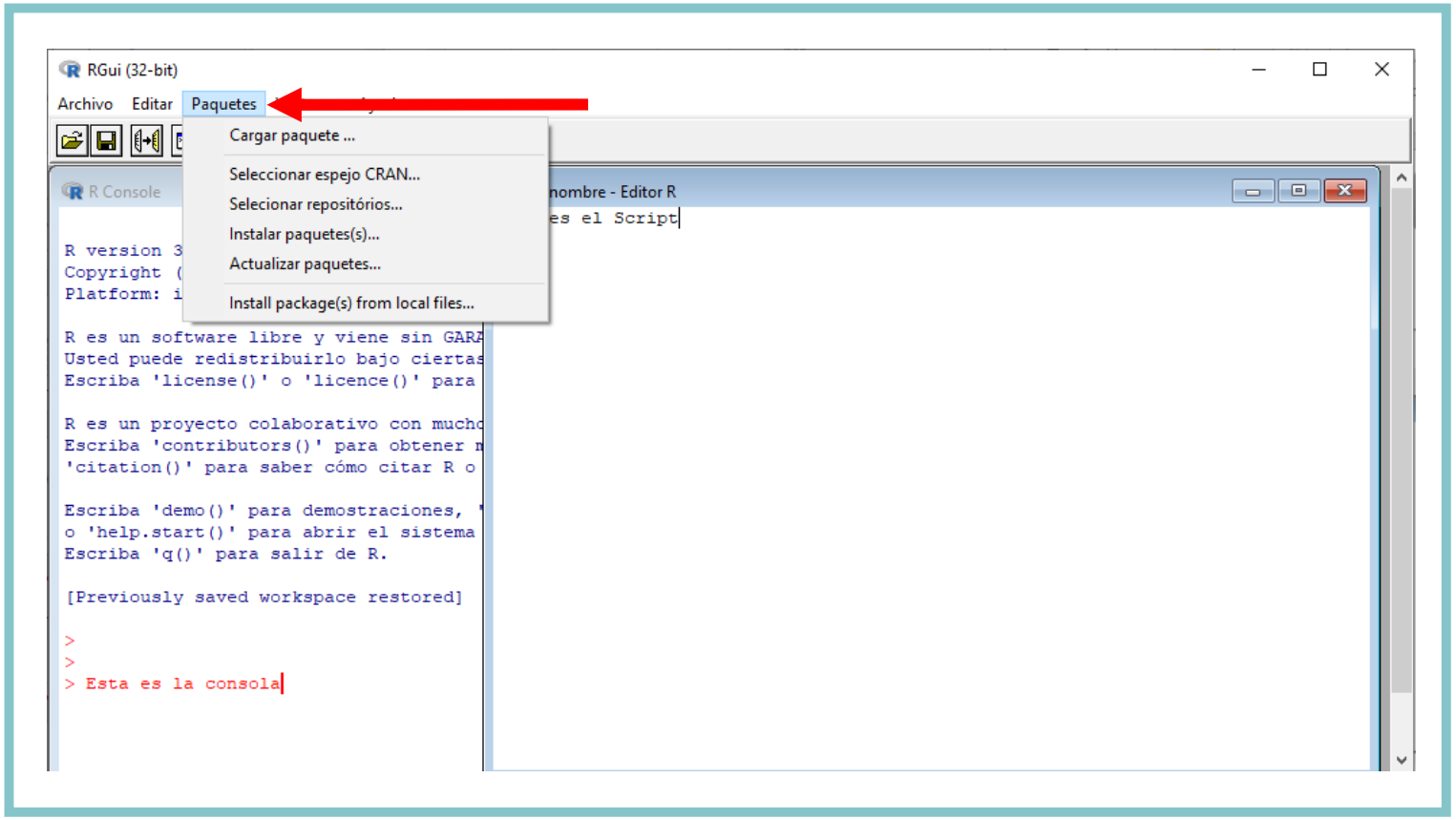


Organizar ventanas:

Muestra todas las ventanas que uno tenga abiertas. (Consola, Scripts, gráficos, etc.)



Cargar paquetes



R Studio

- Un programa que mejora la interfaz de R.

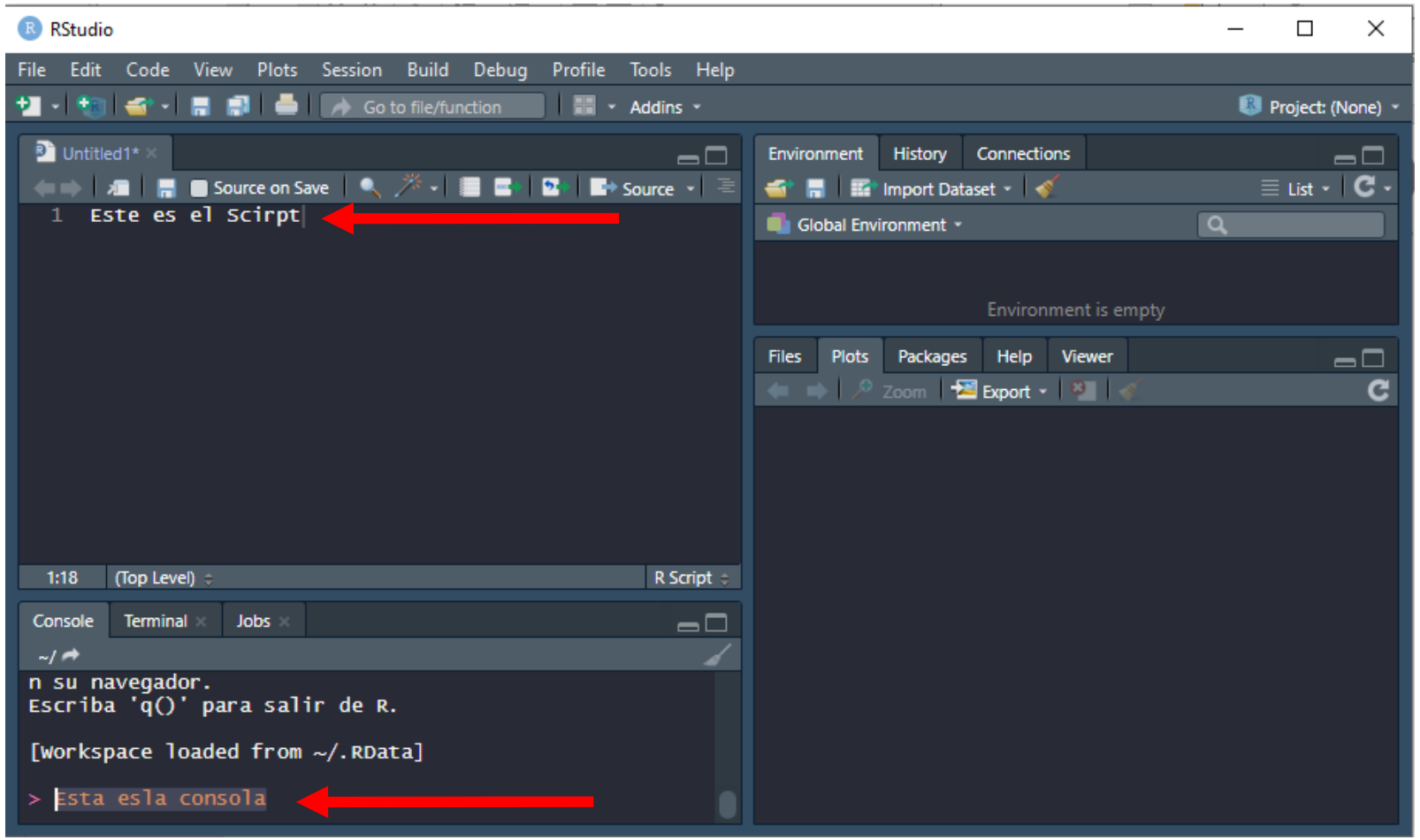
- Funciona con los mismos comandos de R.

Bajar R Studio

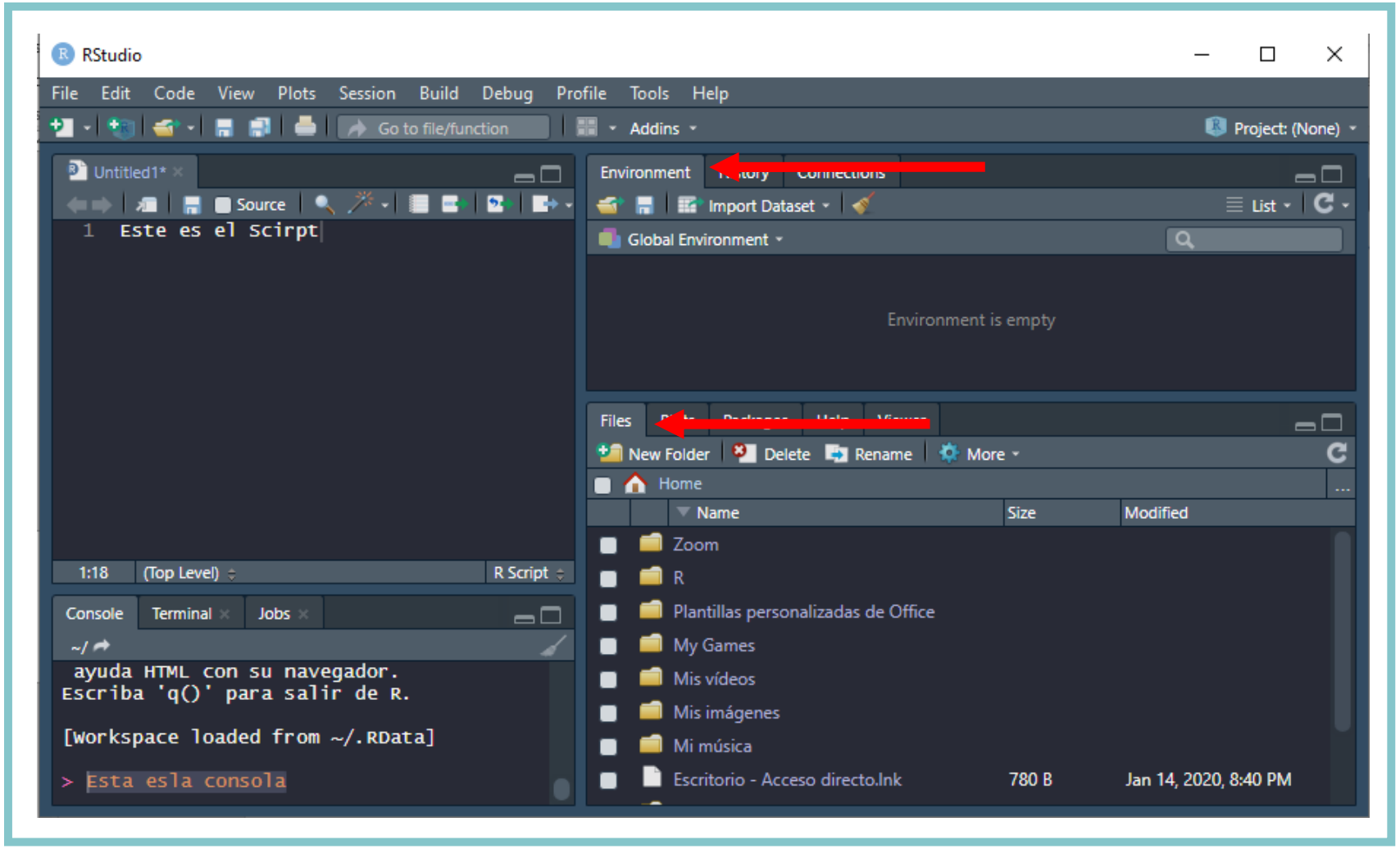
Una vez instalado R, puede instalar R Studio desde:

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

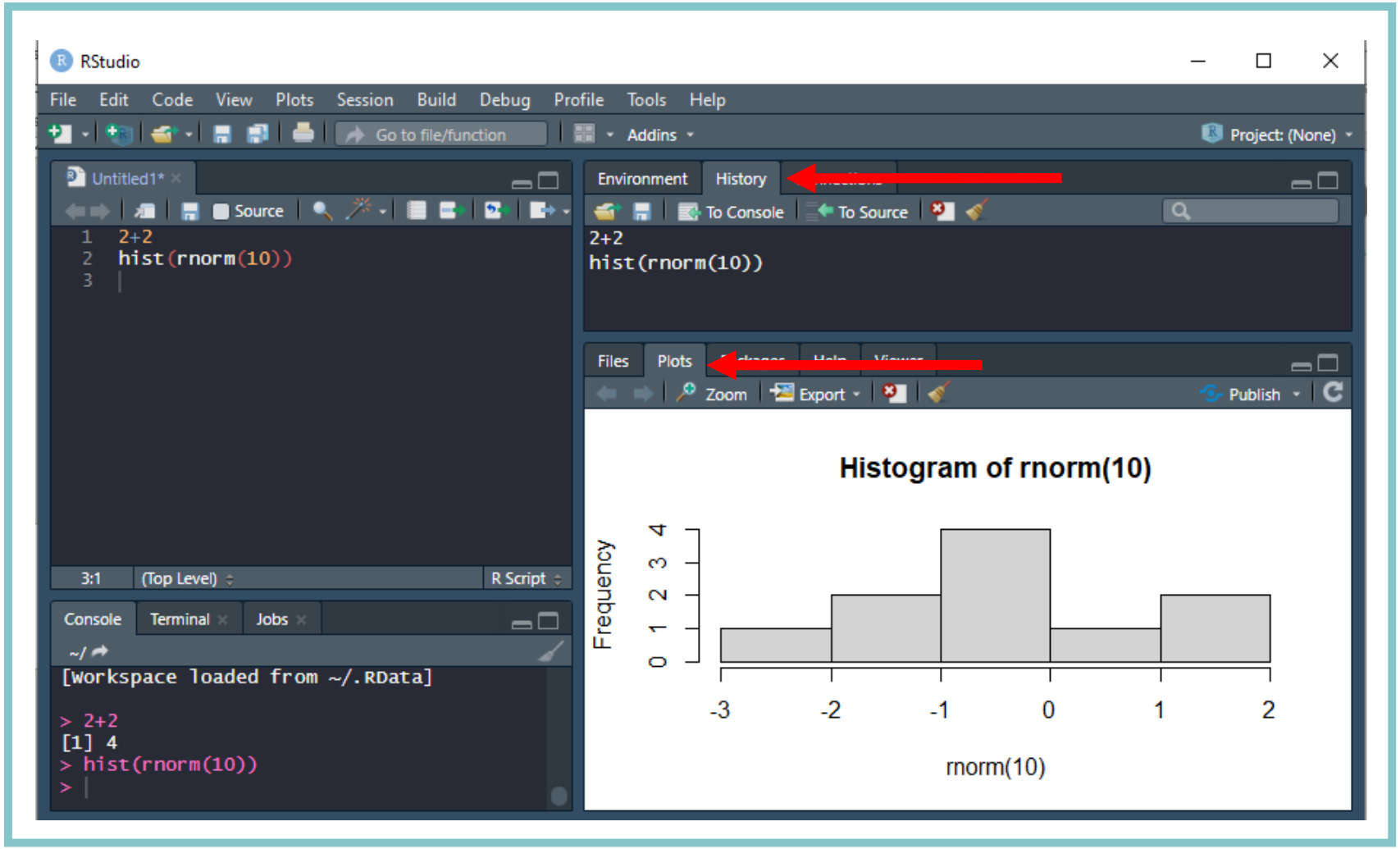
R vs R Studio


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1*
Source on Save
Source
1 Este es el Scirpt
1:18
(Top Level)
R Script
Console
Terminal
Jobs
~/
n su navegador.
Escriba 'q()' para salir de R.
[Workspace loaded from ~/.RData]
> Esta esla consola
Environment
History
Connections
Import Dataset
List
Global Environment
Environment is empty
Files
Plots
Packages
Help
Viewer
Zoom
Export

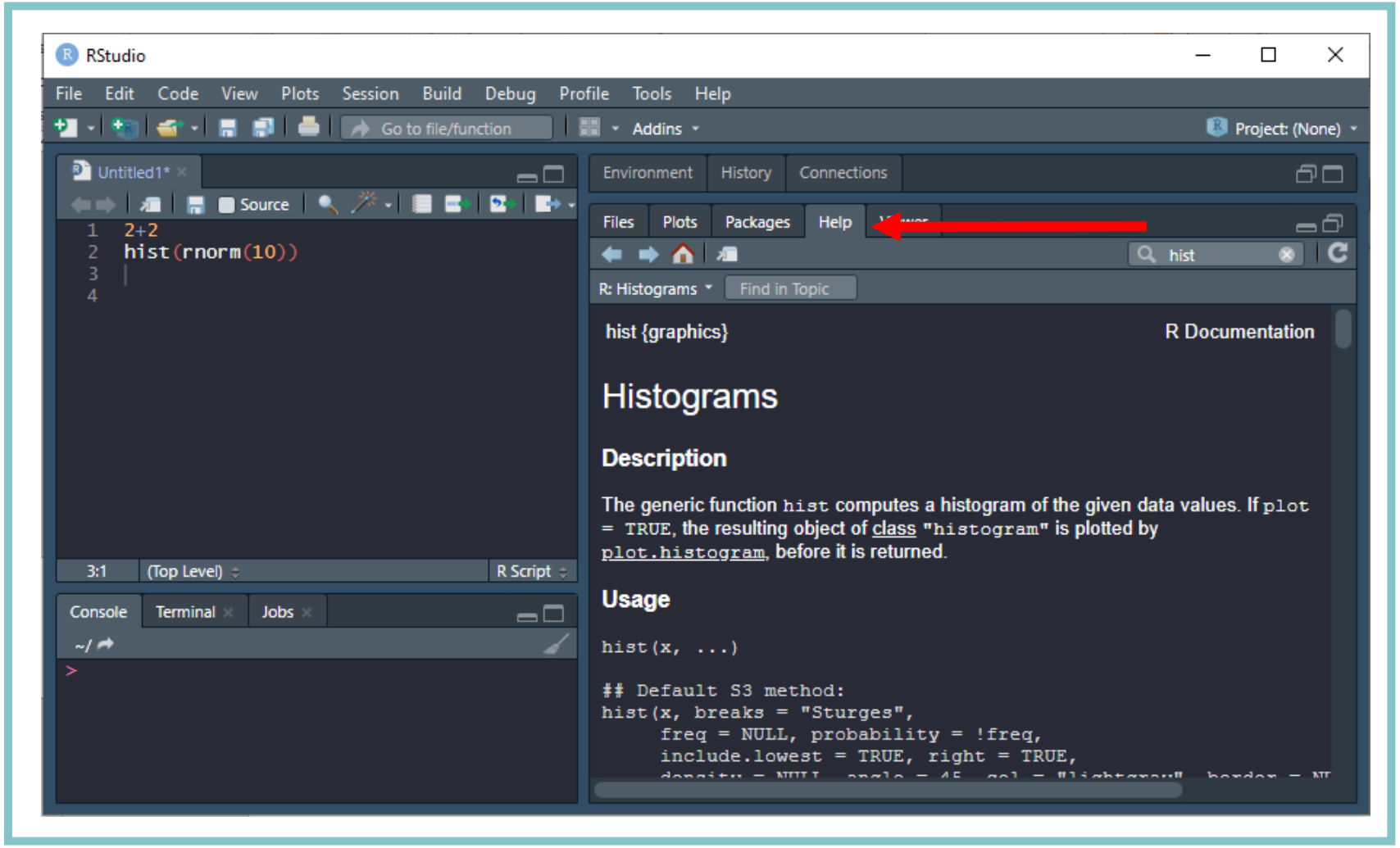
R VS R Studio


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1*
Source
1 Este es el Scirpt
1:18
(Top Level)
R Script
Console
Terminal
Jobs
~/
ayuda HTML con su navegador.
Escriba 'q()' para salir de R.
[Workspace loaded from ~/.RData]
> Esta esla consola
Environment
Import Dataset
List
Global Environment
Environment is empty
Files
New Folder
Delete
Rename
More
Home
Name
Size
Modified
Zoom
R
Plantillas personalizadas de Office
My Games
Mis vídeos
Mis imágenes
Mi música
Escritorio - Acceso directo.lnk
780 B
Jan 14, 2020, 8:40 PM

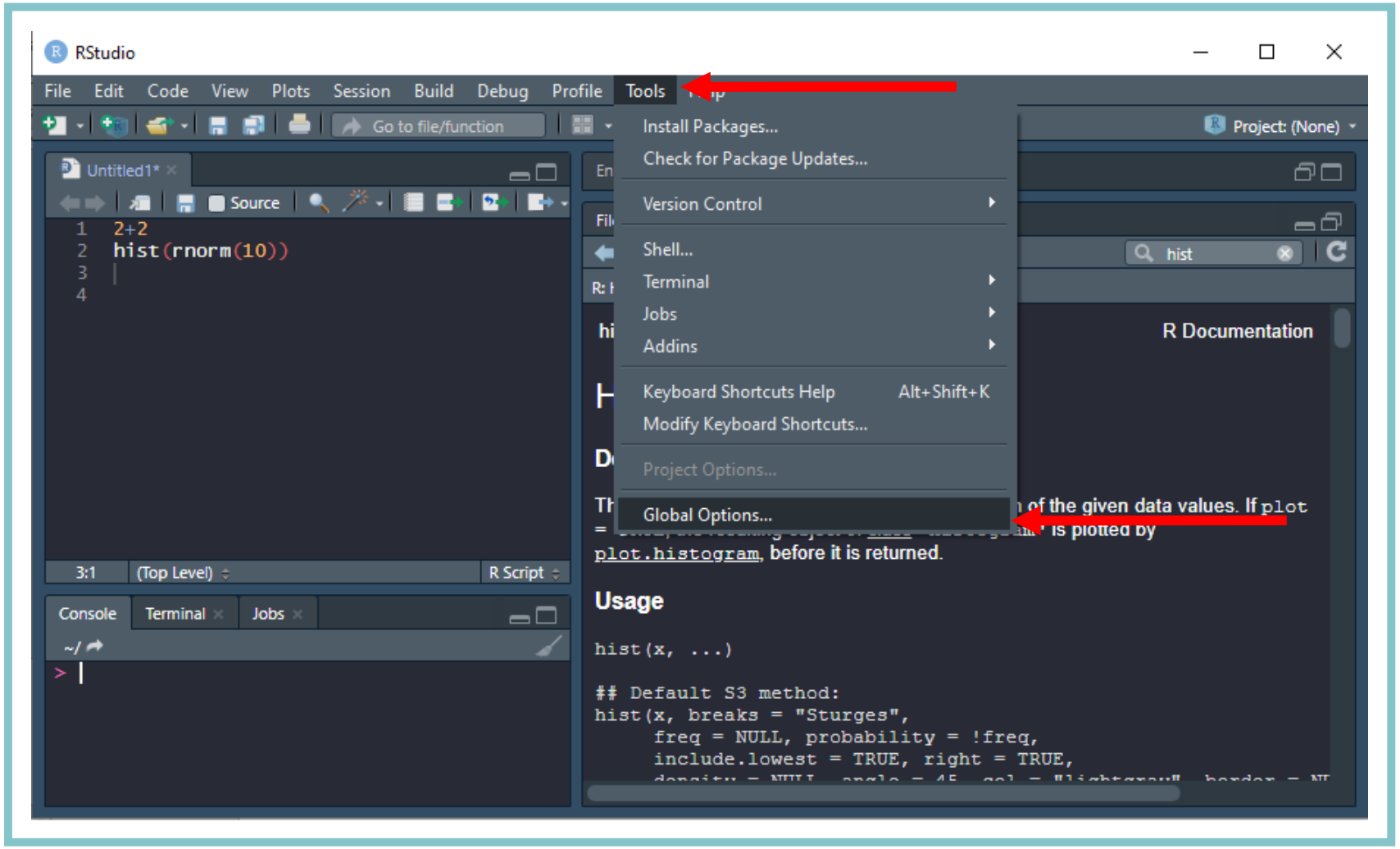
R VS R Studio

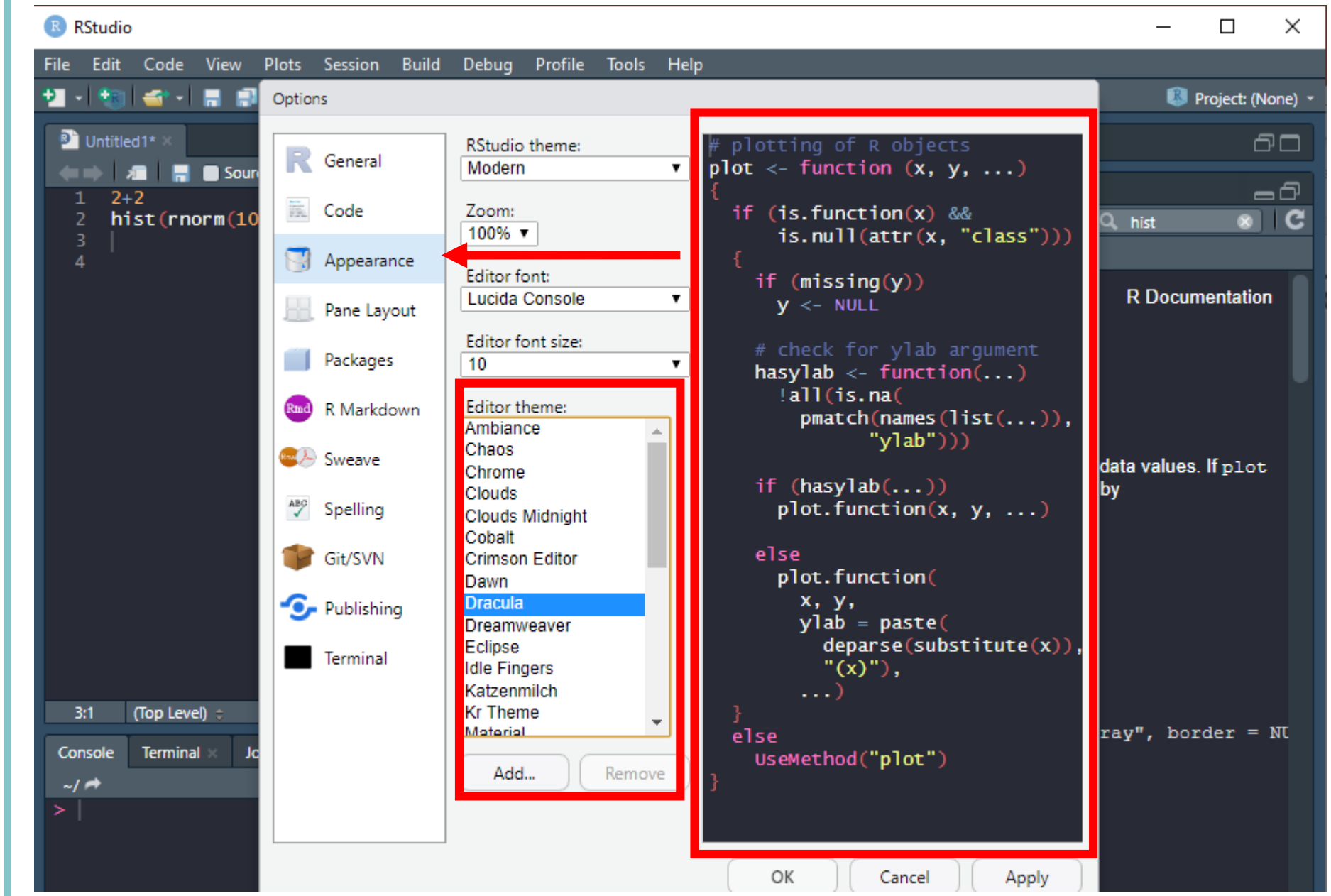

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1*
Source
1 2+2
2 hist(rnorm(10))
3
3:1
(Top Level)
R Script
Environment
History
To Console
To Source
2+2
hist(rnorm(10))
Files
Plots
Zoom
Export
Publish
Histogram of rnorm(10)
Frequency
0 1 2 3 4
-3 -2 -1 0 1 2
rnorm(10)
Console
Terminal
Jobs
~/
[Workspace loaded from ~/.RData]
> 2+2
[1] 4
> hist(rnorm(10))
>

R VS R Studio


RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
Untitled1*
Source
1 2+2
2 hist(rnorm(10))
3
4
3:1
(Top Level)
R Script
Console
Terminal
Jobs
~/
>
Environment
History
Connections
Files
Plots
Packages
Help
hist
R: Histograms
Find in Topic
hist {graphics}
R Documentation
Histograms
Description
The generic function hist computes a histogram of the given data values. If plot = TRUE, the resulting object of class "histogram" is plotted by plot.histogram, before it is returned.
Usage
hist(x, ...)
Default S3 method:
hist(x, breaks = "Sturges",
freq = NULL, probability = !freq,
include.lowest = TRUE, right = TRUE,

Cambiar el tema Studio




RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Options
General
Code
Appearance
Pane Layout
Packages
R Markdown
Sweave
Spelling
Git/SVN
Publishing
Terminal
RStudio theme:
Modern
Zoom:
100%
Editor font:
Lucida Console
Editor font size:
10
Editor theme:
Ambiance
Chaos
Chrome
Clouds
Clouds Midnight
Cobalt
Crimson Editor
Dawn
Dracula
Dreamweaver
Eclipse
Idle Fingers
Katzenmilch
Kr Theme
Material
Add...
Remove
OK
Cancel
Apply
Untitled1*
2+2
hist(rnorm(10
Console
Terminal
Project: (None)
hist
R Documentation
data values. If plot
by
ray", border = NU
# plotting of R objects
plot <- function (x, y, ...)
{
if (is.function(x) &&
is.null(attr(x, "class")))
{
if (missing(y))
y <- NULL
# check for ylab argument
hasylab <- function(...)
!all(is.na(
pmatch(names(list(...)),
"ylab")))
if (hasylab(...))
plot.function(x, y, ...)
else
plot.function(
x, y,
ylab = paste(
deparse(substitute(x)),
"(x)"),
...)
}
else
UseMethod("plot")
}

R Studio Cloud

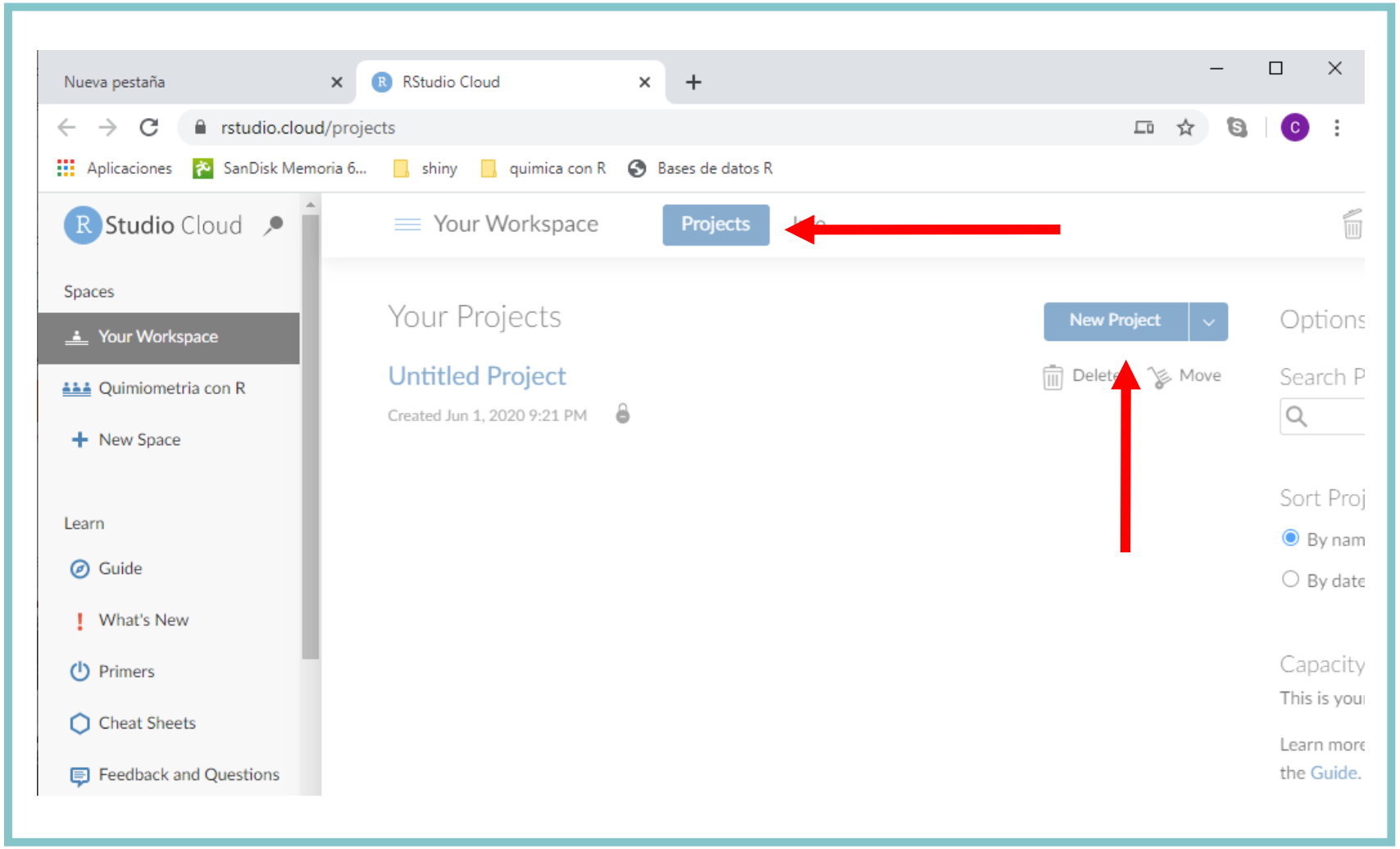
R Cloud es una plataforma digital donde se puede utilizar R Studio de manera digital.

https://rstudio.cloud/

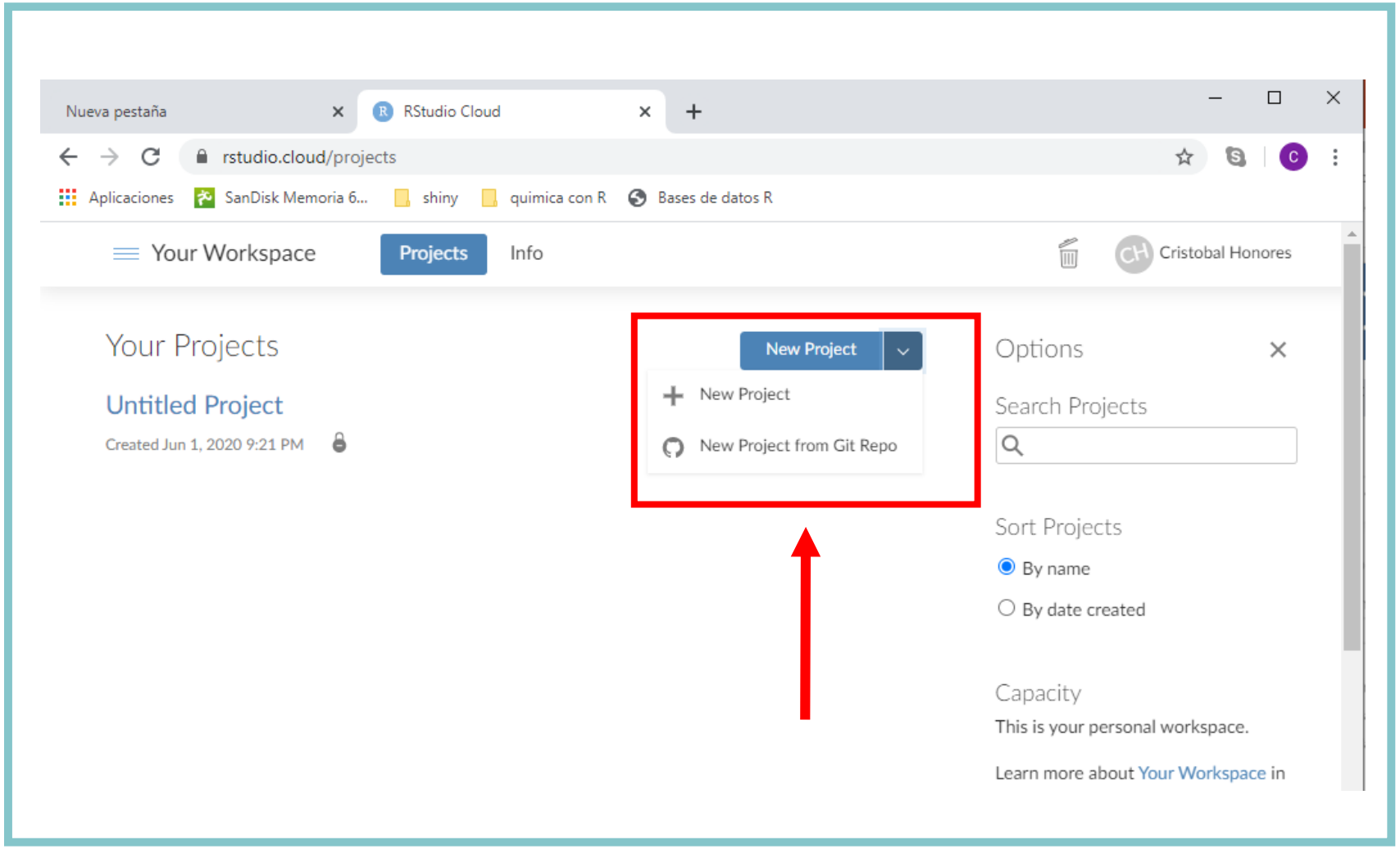
Ventajas de R Cloud

- No es necesario instalar ningún programa.
- Es gratis.
- Se puede compartir proyectos y script de manera fácil.
- Se guarda de manera automática en la cuenta.
- Puede utilizarse de cualquier computador.
- Fácil de usar.

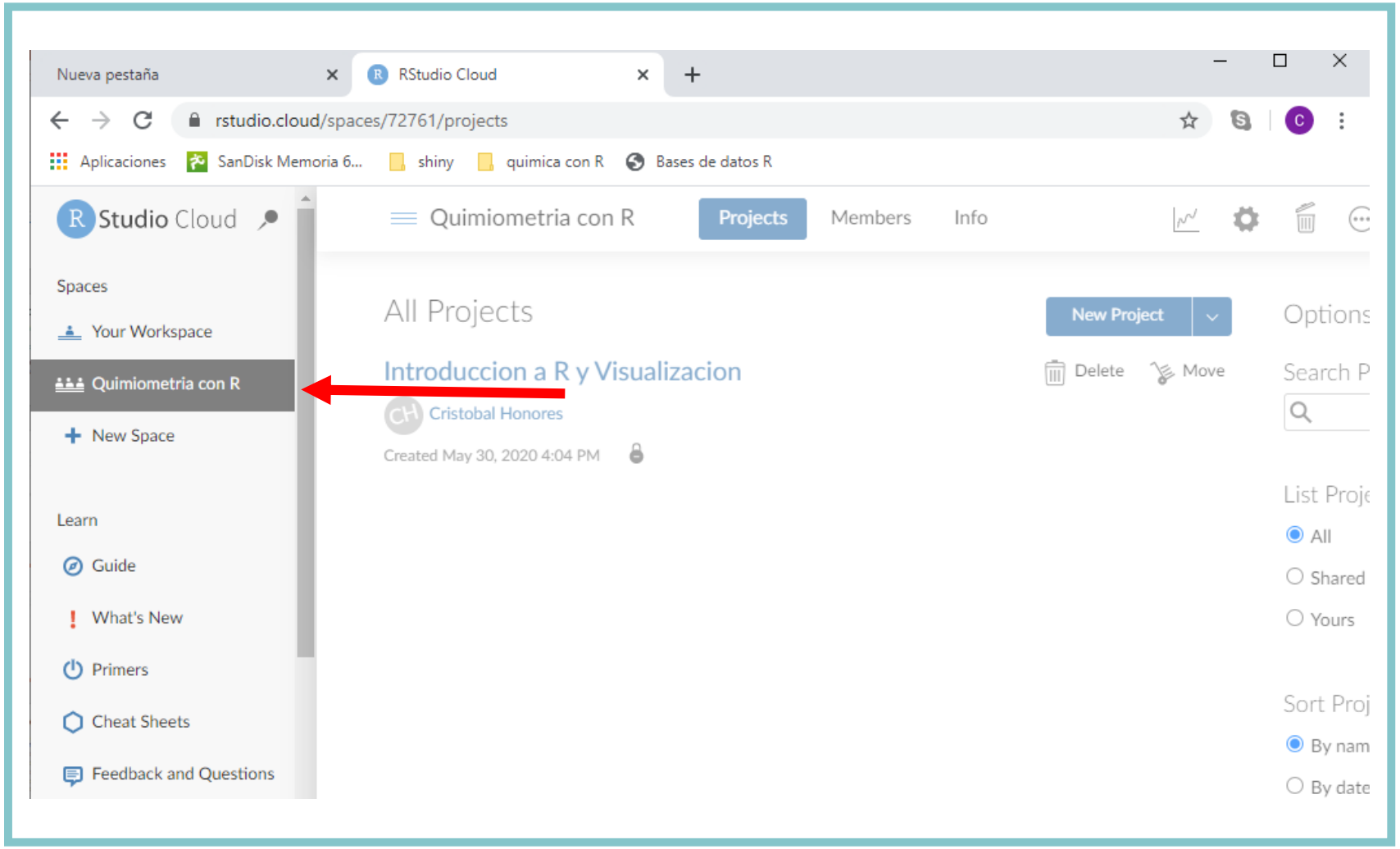
R Studio Cloud



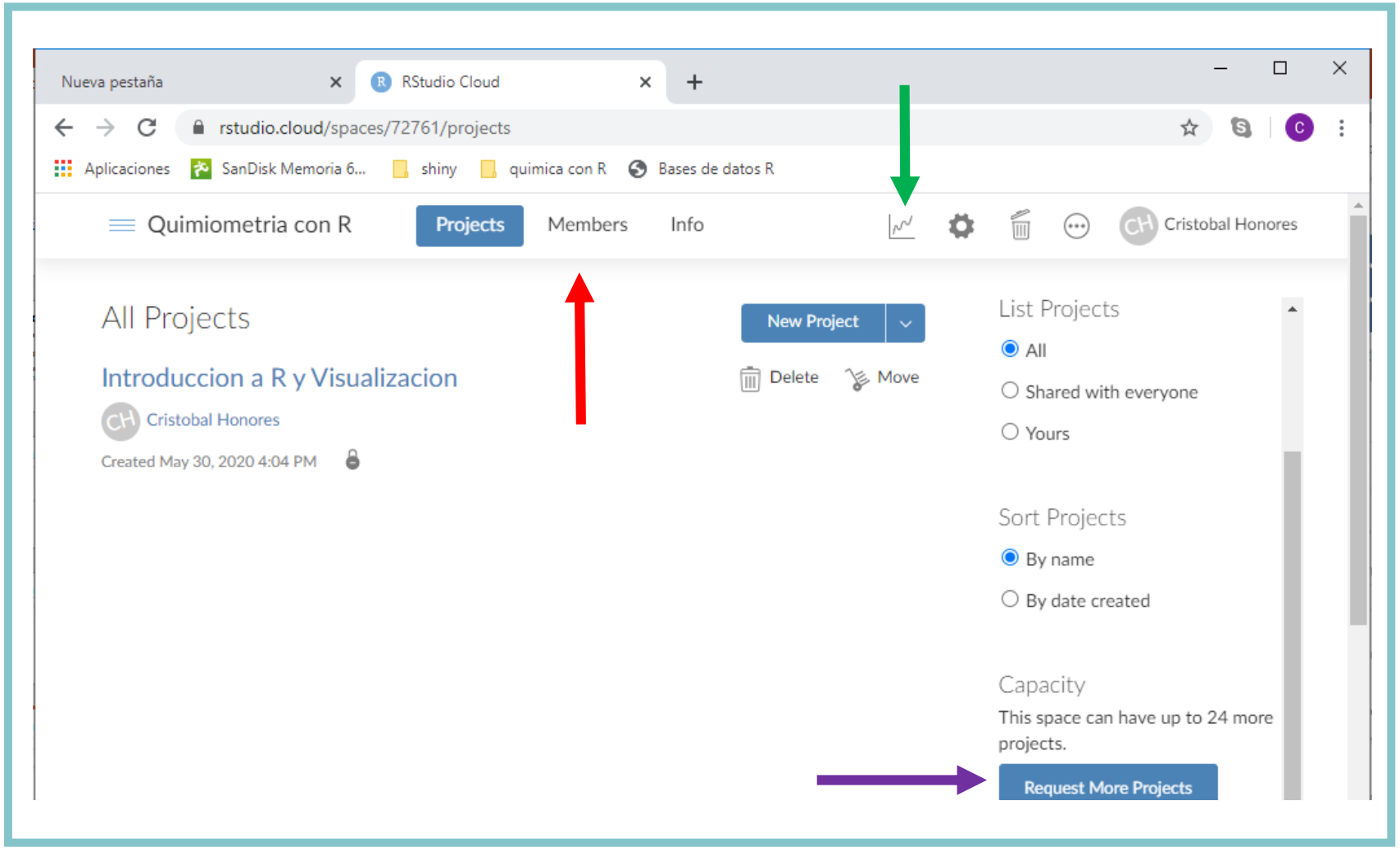
R Studio Cloud



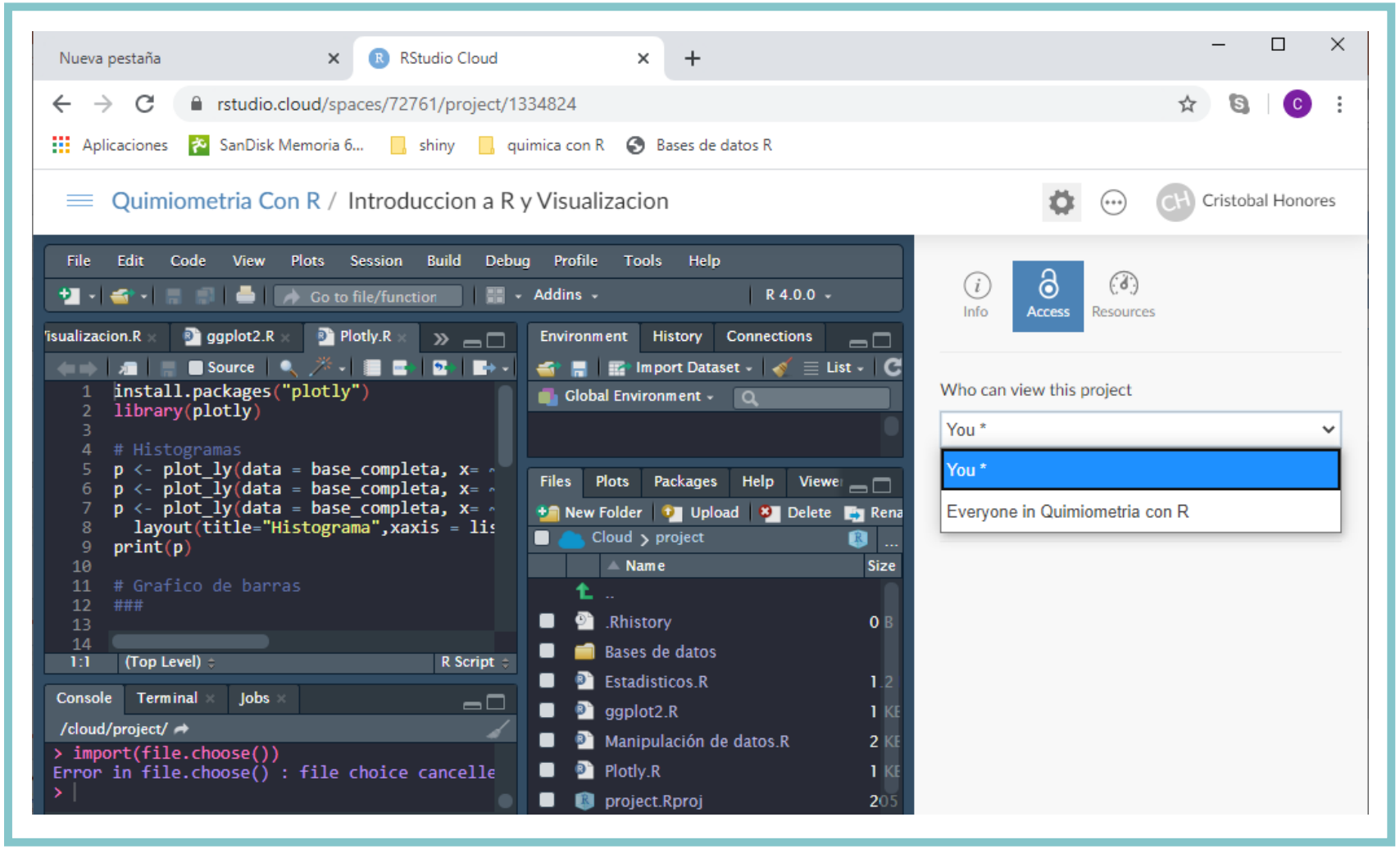
R Studio Cloud



R Studio Cloud



R Studio Cloud



Funciones básicas

En R se pueden hacer diversas operaciones usando operadores binarios (entre dos objetos).

+ operador binario para sumar.

- operador binario para restar.

* operador binario para multiplicar.

/ operador binario para dividir.

^ operador binario para potencia.

Objetos

Un objeto es la manera simplificada que tiene R para trabajar con bases de datos, marcos de datos, vectores, funciones, matrices, etc.

Para asignar un objeto se hace de la siguiente manera:

nombre **<-** *función()*

a <- 2+2

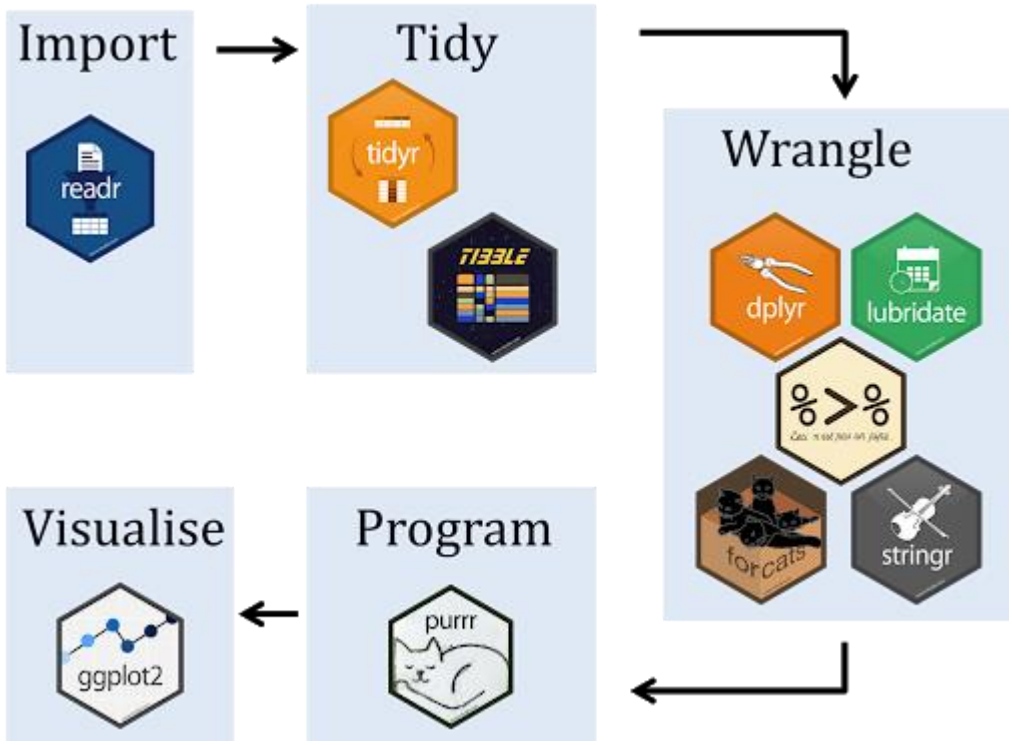
Para remover un objeto se ocupa la siguiente función:

rm(*nombre****)***

rm(a)

Paquetes

Los paquetes son colecciones de funciones R, datos y código compilado en un formato bien definido.



Repositorios

- **CRAN:** El repositorio oficial es una red de servidores mantenidos por la comunidad R.

- **Bioconductor:** Este es un repositorio específico del tema, destinado al software de código abierto para bioinformática.

- **Github:** Aunque esto no es específico de R, Github es probablemente el repositorio más popular para proyectos de código abierto.

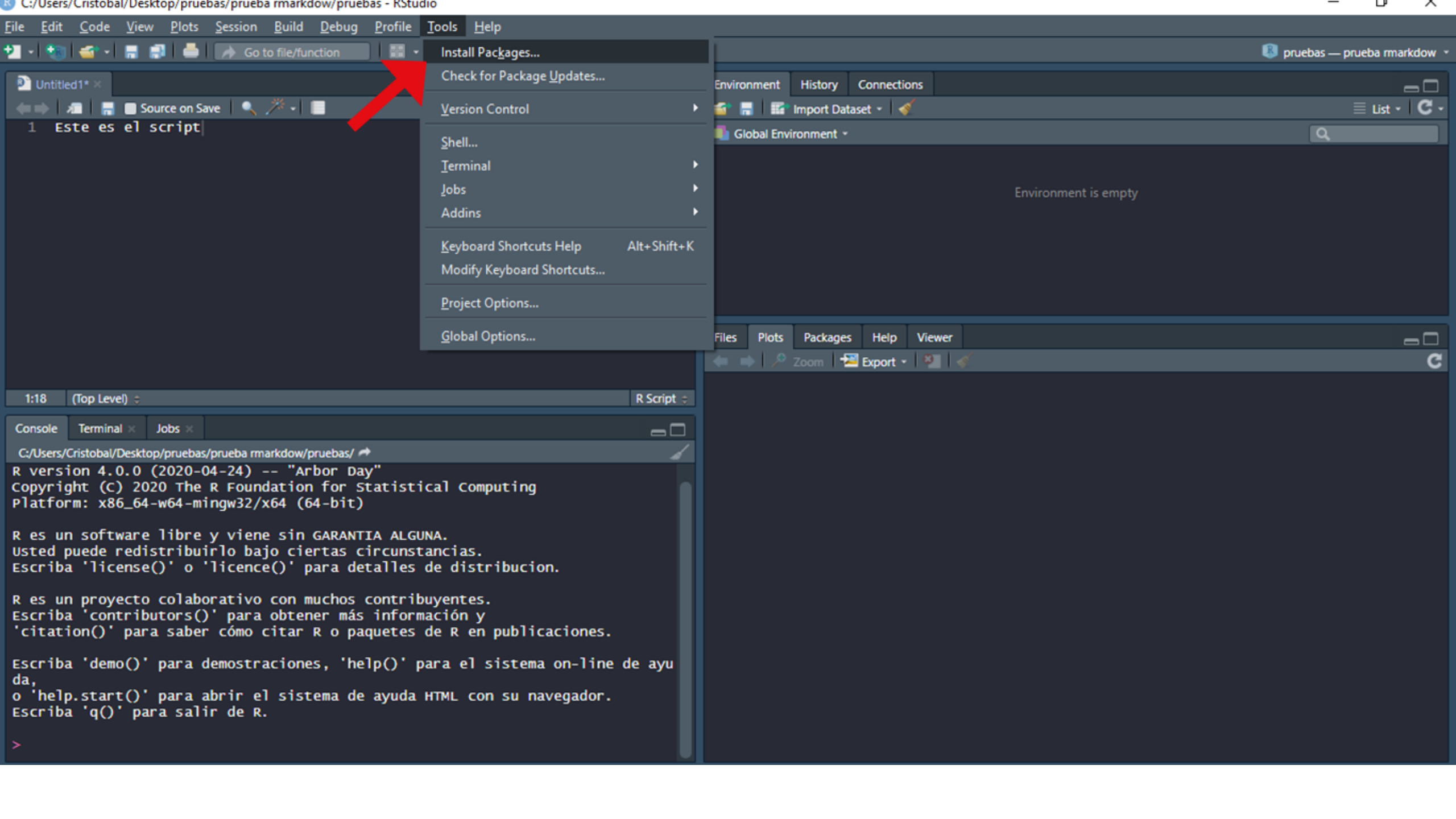
¿Cómo instalar paquetes?

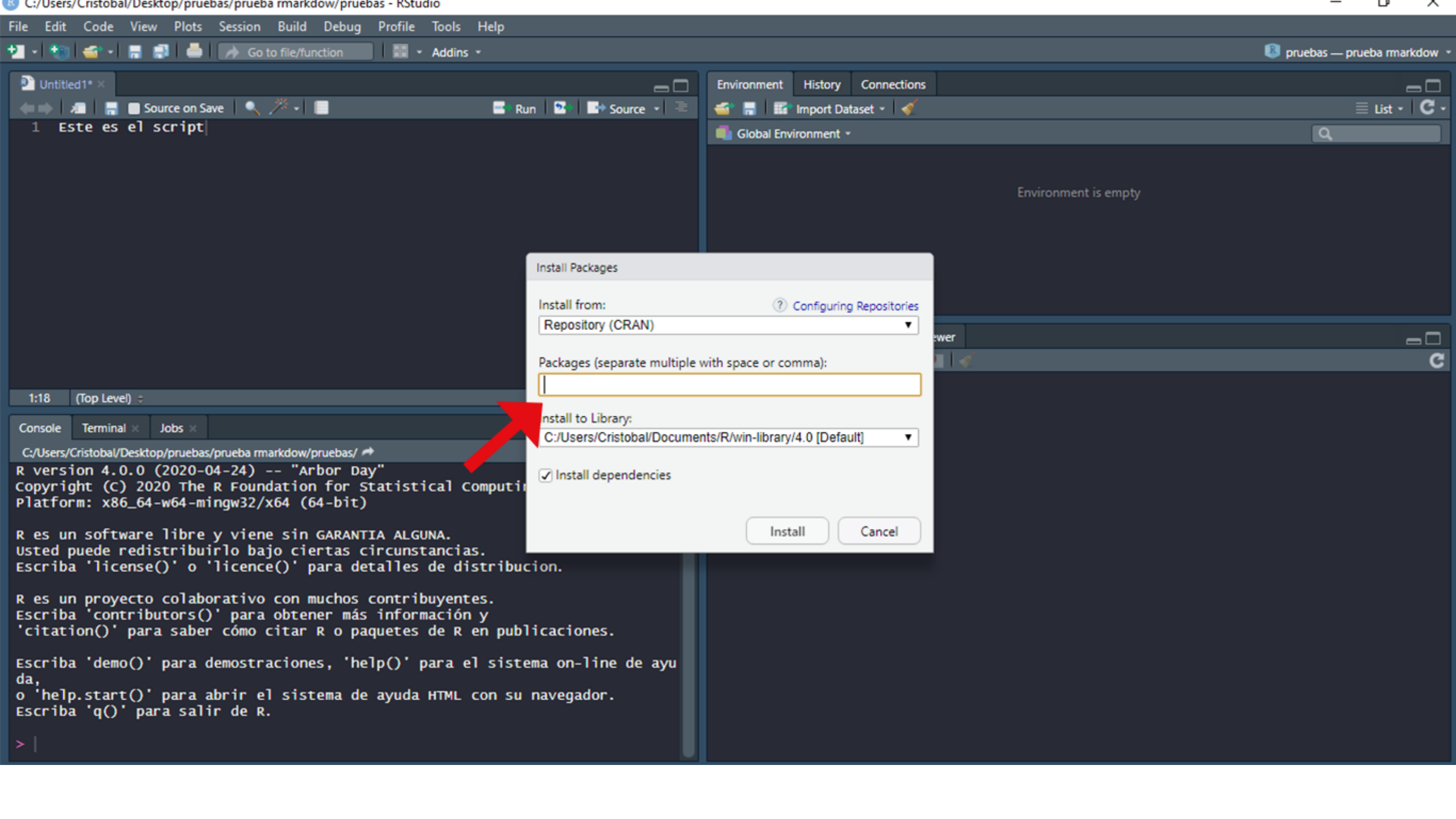
Para instalar paquetes desde CRAN es necesario utilizar la función:

install.packages("*paquete*")

También es posible instalar mas de in paquete a la vez:

install.packages(c("*paquete1*","*paquete2*",...))


C:/Users/Cristobal/Desktop/pruebas/prueba rmarkdow/pruebas - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
pruebas — prueba rmarkdow
Install Packages...
Check for Package Updates...
Version Control
Shell...
Terminal
Jobs
Addins
Keyboard Shortcuts Help Alt+Shift+K
Modify Keyboard Shortcuts...
Project Options...
Global Options...
Untitled1*
Source on Save
1 Este es el script
1:18 (Top Level) R Script
Environment History Connections
Import Dataset
List
Global Environment
Environment is empty
Files Plots Packages Help Viewer
Zoom Export
Console Terminal Jobs
C:/Users/Cristobal/Desktop/pruebas/prueba rmarkdow/pruebas/
R version 4.0.0 (2020-04-24) -- "Arbor Day"
Copyright (C) 2020 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-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 ayu
da,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.
>


C:/Users/Cristobal/Desktop/pruebas/prueba rmarkdow/pruebas - RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
pruebas — prueba rmarkdow
Untitled1*
Source on Save
Run
Source
1 Este es el script
1:18
(Top Level)
Environment
History
Connections
Import Dataset
List
Global Environment
Environment is empty
Install Packages
Install from:
Configuring Repositories
Repository (CRAN)
Packages (separate multiple with space or comma):
Install to Library:
C:/Users/Cristobal/Documents/R/win-library/4.0 [Default]
Install dependencies
Install
Cancel
Console
Terminal
Jobs
C:/Users/Cristobal/Desktop/pruebas/prueba rmarkdow/pruebas/
R version 4.0.0 (2020-04-24) -- "Arbor Day"
Copyright (C) 2020 The R Foundation for Statistical Computi
Platform: x86_64-w64-mingw32/x64 (64-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 ayu
da,
o 'help.start()' para abrir el sistema de ayuda HTML con su navegador.
Escriba 'q()' para salir de R.
>

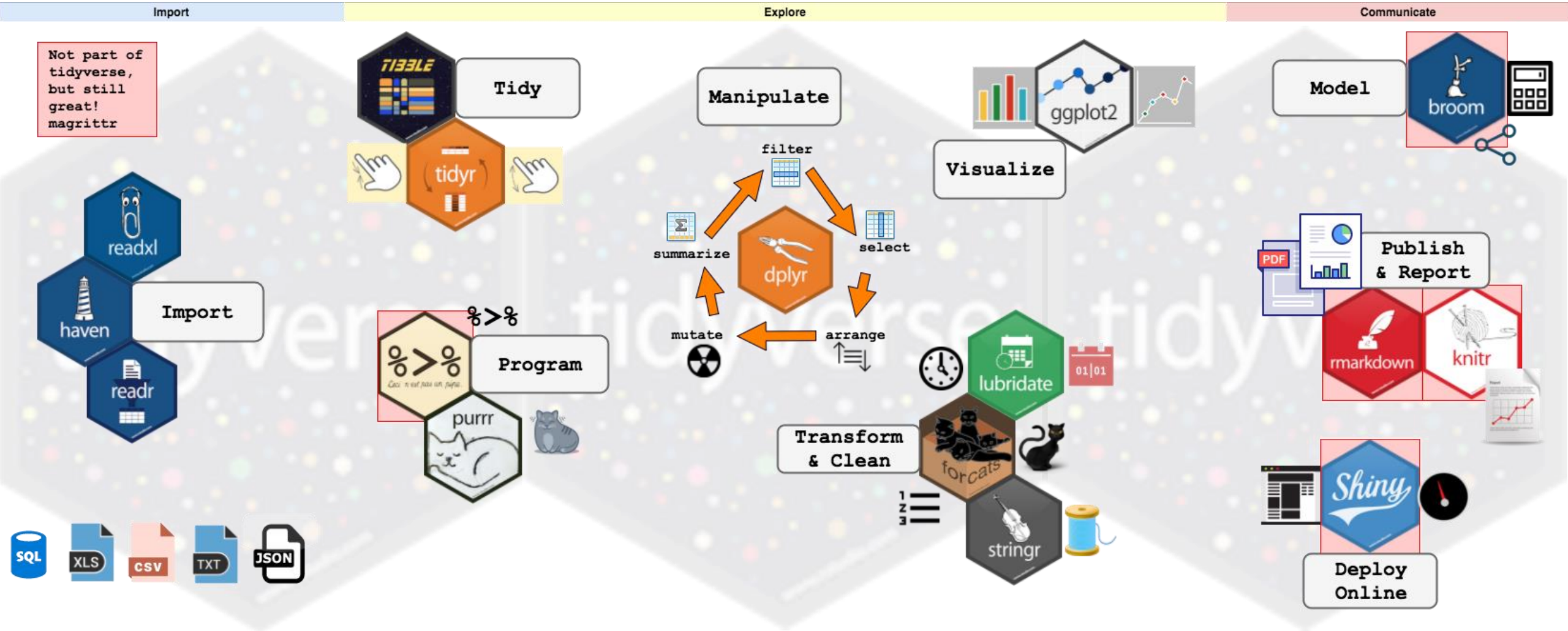
Cargar paquetes

Una vez instalados los paquetes es necesario cargarlos, para eso se utiliza la siguiente función.

library(paquete)

Los paquetes tienen que cargarse cada vez que se abre la sesión de R, sin embargo, se instalan solo una vez.

Tipos de paquetes



Bases de datos

- R trae muchas bases de datos de practica.
- Algunos paquetes también traen bases de datos de practica.

https://vincentarelbundock.github.io/Rdatasets/datasets.html

Importar bases de datos

Existen diferentes funciones para importar bases de datos en diferentes formatos en R:

- **read.table():** Importa Base de Datos en formato TXT, DAT y CSV.
- **read.csv2():** Importa Base de Datos en formato CSV.
- **readXL():** Importa Base de Datos en formato XLS y XLSX. (paquete "readxl").
- **import():** Importa cualquier tipo de archivo. (paquete “rio").

OJO!

- Para guardar bases de datos deben darle un nombre.
- Cuidado con el nombre que le den.
- Uno de los errores recurrentes de porqué no se ejecutan los comandos es debido a que se olvidan de como llamaron a la base de datos.

Paquete rio

Este paquete permite importar casi cualquier tipo de archivo a R.

Es necesario instalar el paquete “rio”.

install.packages(“rio”) Este comando instala el paquete.

Library(rio) Este comando carga el paquete.

Import()

El comando impor() es el que permite importar las bases.

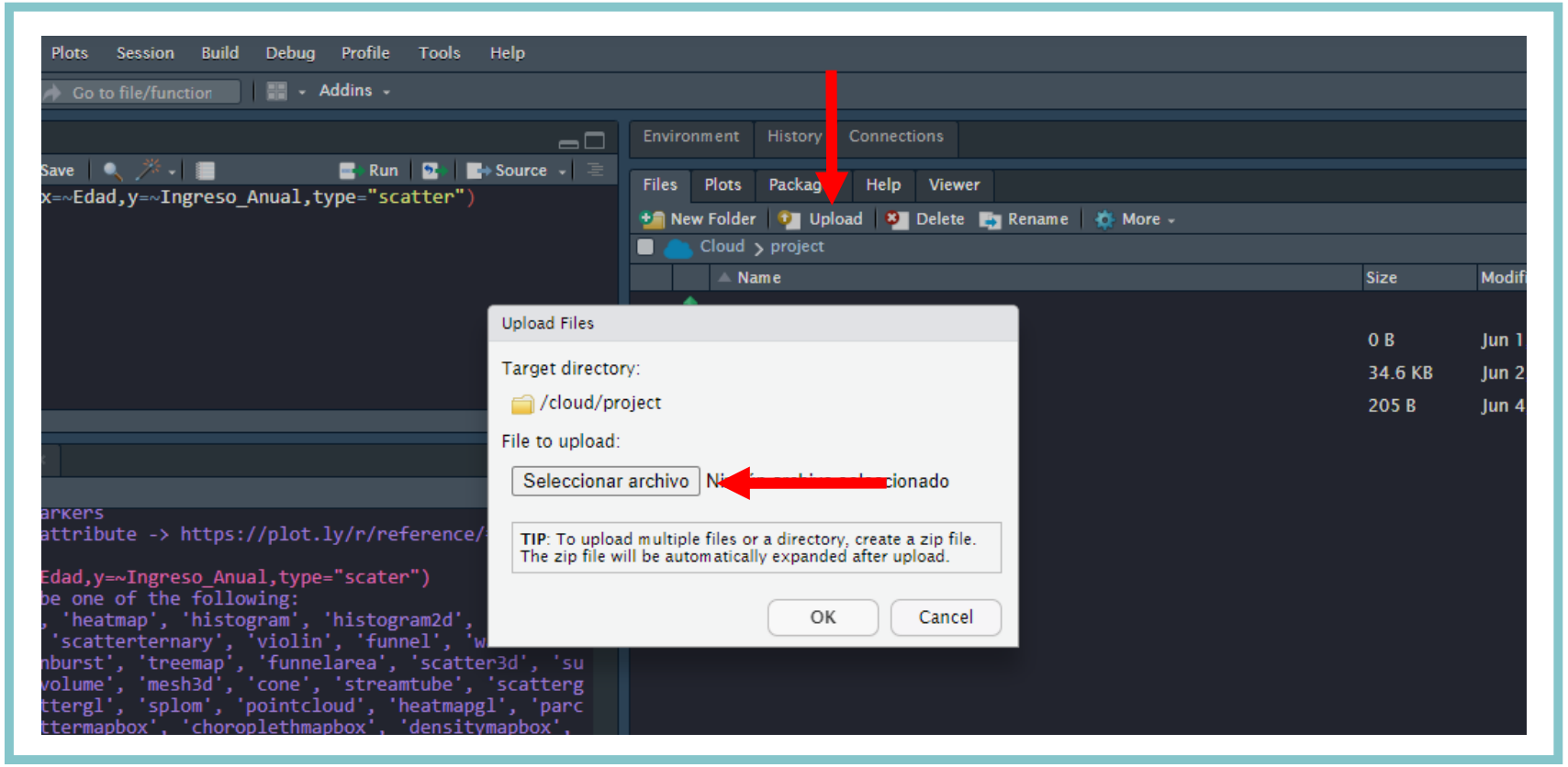
El comando file.choose() permite seleccionar un archivo simplemente a través de clicks.

import(file.choose()) carga la base de datos seleccionada.

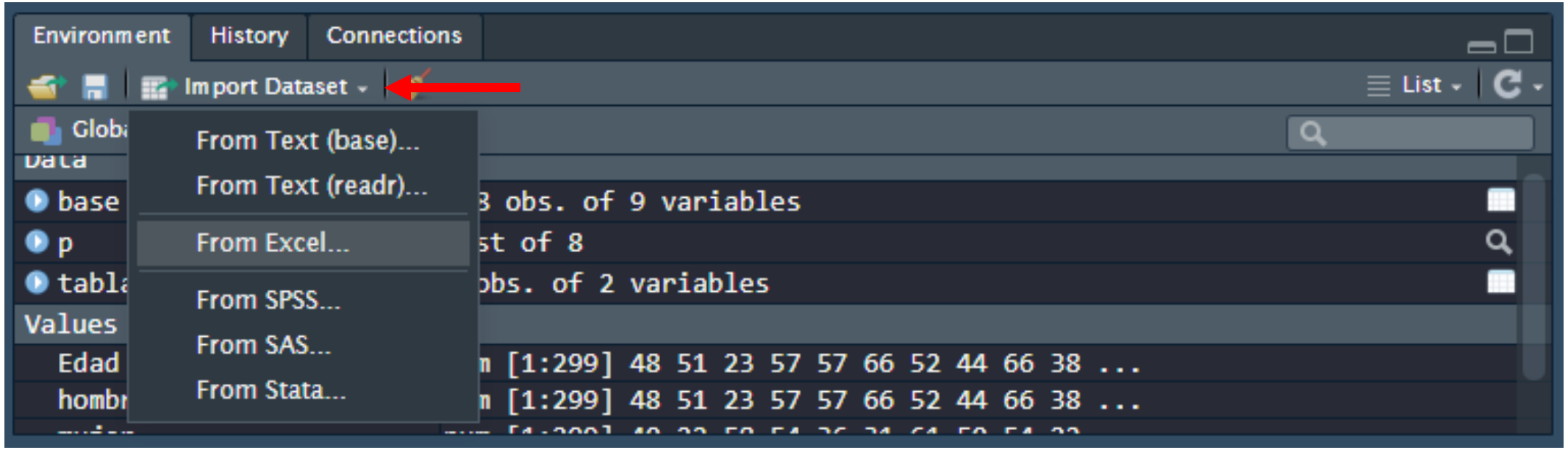
Base <- import(file.choose())

Es necesario guardar la base como objeto para poder trabajar la base de datos.

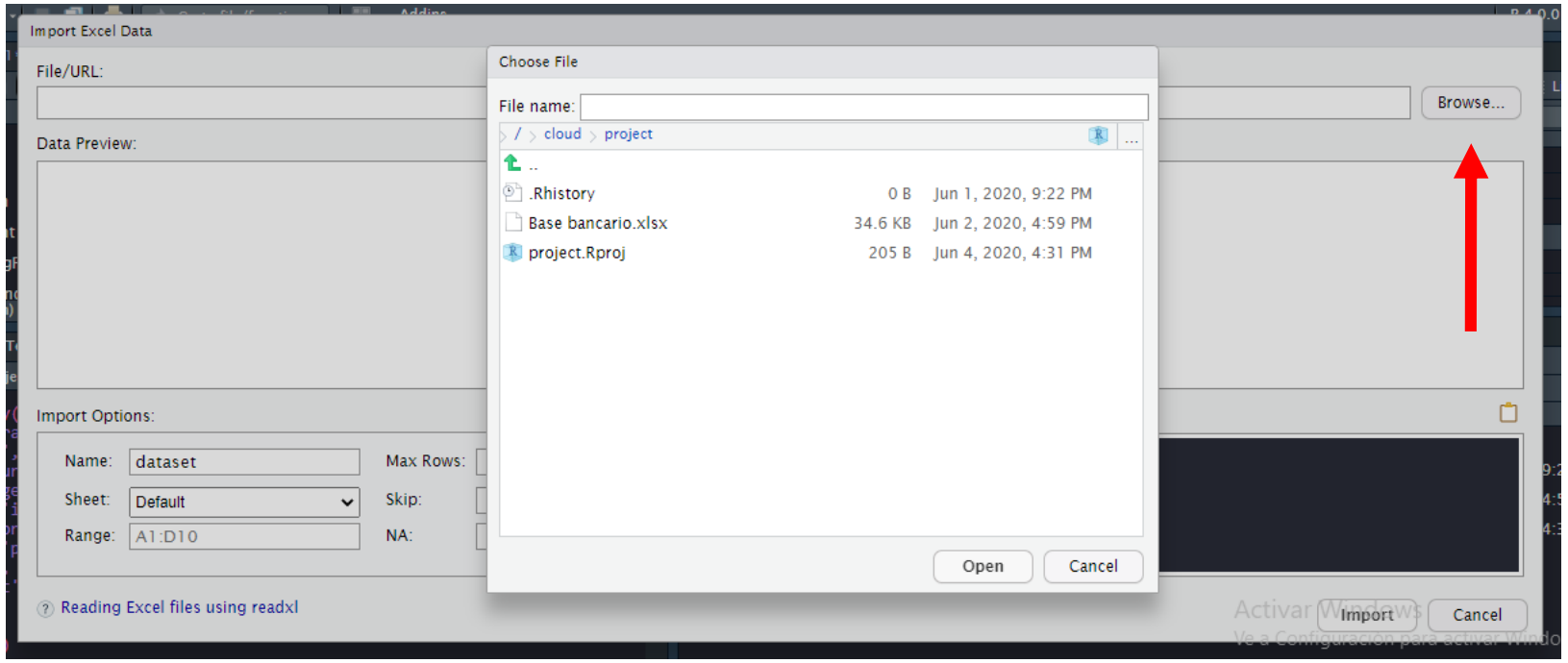
Importar bases de datos



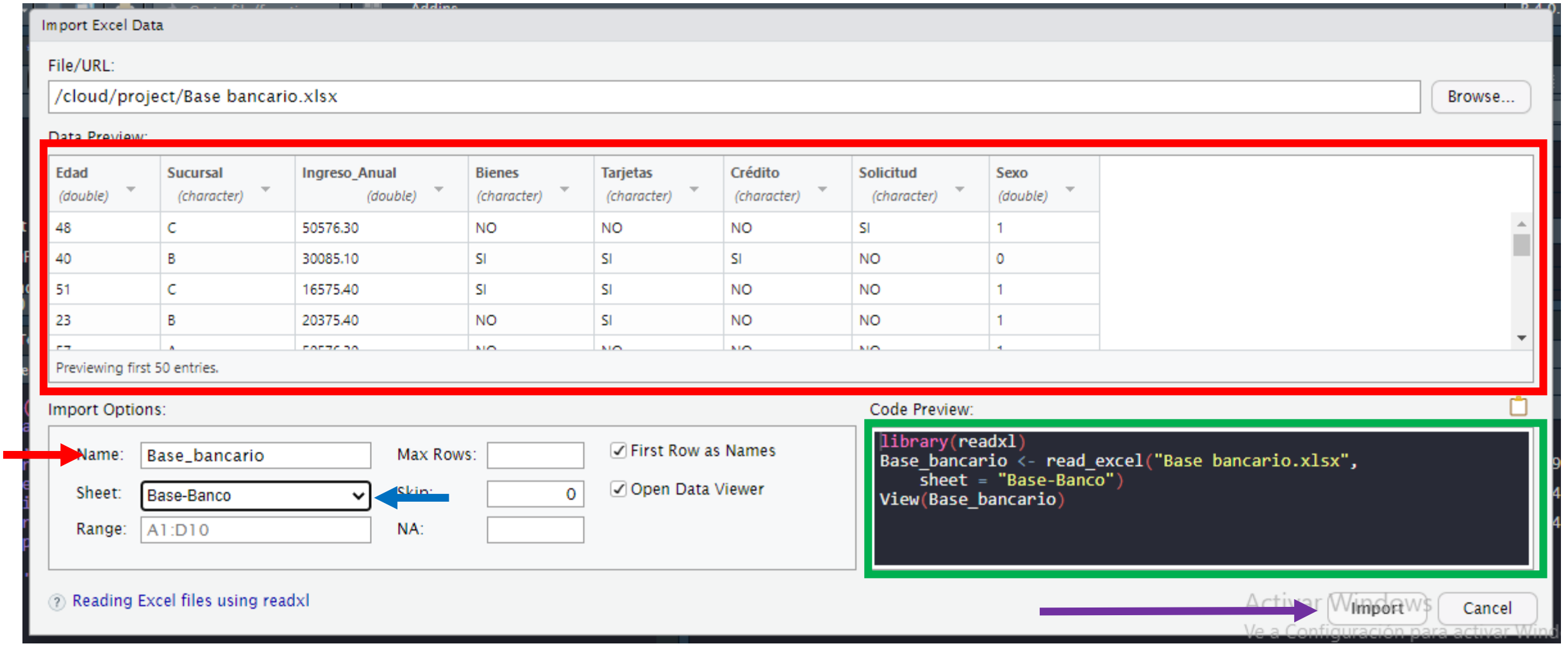
Importar bases de datos



Importar bases de datos



Importar bases de datos



Leer bases de datos

- **head():** Esta función entrega las primeras filas de mi base de datos, vector o tabla.
- **summary():** Produce un resumen simplificado de una base de datos, una función, un modelo de ajuste, etc.
- **names():** Entrega los nombres de las columnas que tiene una base de datos.
- **Tail():** Esta función entrega las ultimas filas de mi base de datos, vector o tabla.

Seleccionar columnas y filas

Para seleccionar una columna en especifico de la base de dato se utiliza $ y luego el nombre de la columna.

Base$Columna

base_completa$EDAD: selecciona la columna llamada EDAD.

Seleccionar columnas y filas

También se pueden seleccionar columnas y filas utilizando [ ] después del objeto.

Bases[,]:

Fila: base_completa[numero de fila,]

Columna: base_completa[, numero de columna]

Dato especifico: base_completa[numero de fila,numero de columna]

Filtros

Para filtrar bases de datos utilizamos [ ] colocando dentro el criterio por el cual queremos filtrar.

Base[Criterio,]

Criterio: Puede ir cualquier operación lógica que tenga sentido sobre una variable.

Base$variable + >, <=, >=, ==, != + valor

Filtros

Seleccionaremos los datos de las personas que fuman.

```
base_completa[base_completa$FUMA==1,]
```

Si queremos seleccionar solo a las personas menores o iguales a 35

```
base_completa[base_completa$EDAD <= 35,]
```

Filtros

Utilizando el filtro se puede seleccionar una único atributo observación especifica (columna) de esta base filtrada.

Para esto hay dos opciones:

1. Seleccionamos la variable de interés y luego reducimos la base con el filtro.

Base$variable[Criterio]

2. Reducimos la base con el filtro y luego seleccionamos la variable de interés de esta base filtrada.

Base[Criterio,]$variable

OJO! No olvidar la coma luego del criterio en este caso.

Generar nuevas columnas

Para generar nuevas columnas colocamos el nombre de la base luego $ y el nombre de la columna nueva.

```
Base$nombre_nueva_variable = .....
```

Para remover una columna se coloca el nombre de esta columna luego <- NULL

```
Base$nombre_variable <- NULL
```

Generar nuevas columnas

```
base_completa$Peso_libras = base_completa$PESO * 2.2
```

Calcula el peso de las personas en libras.

```
base_completa$Peso_libras <- NULL
```

Elimina la nueva columna de peso en libras.

Recodificar variables

- Para recodificar se utiliza la función ***ifelse()***

```
ifelse(base_completa$FUMA==1,"Si","No")
```

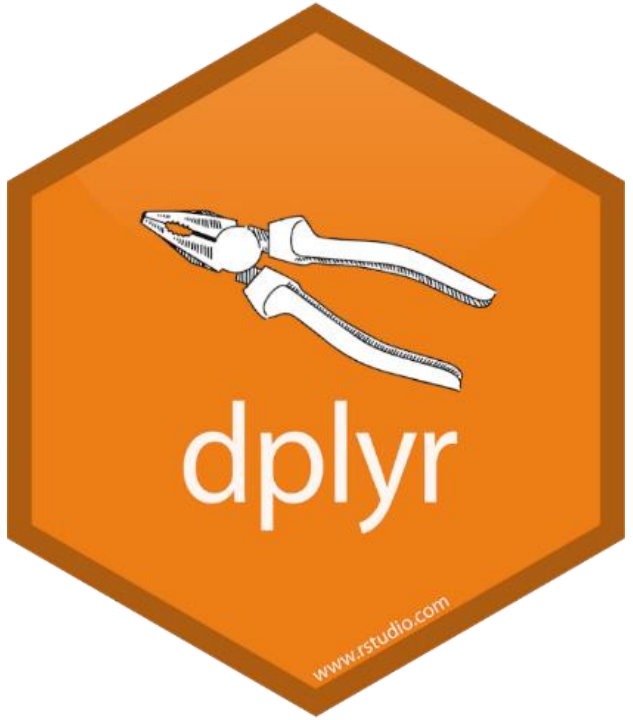
- Podemos generar una nueva columna recodificada.

```
base_completa$FUMA_RECOD = ifelse(base_completa$FUMA==1,"Si","No")
```

Paquete: Dplyr

Es un paquete de R diseñado para manipular, limpiar y resumir datos no estructurados.

Hace que la exploración y manipulación de datos sea fácil y rápida en R.



Funciones importantes

Función	Descripción
select()	Selecciona columnas. (variables)
filter()	Filtra por filas.
summarise()	Resumir datos.
join()	Une marcos de datos.

Combine Data Sets

a

x1	x2
A	1
B	2
C	3

+

b

x1	x3
A	T
B	F
D	T

=

Mutating Joins

x1	x2	x3
A	1	T
B	2	F
C	3	NA

dplyr::**left_join(a, b, by = "x1")**

Join matching rows from b to a.

x1	x3	x2
A	T	1
B	F	2
D	T	NA

dplyr::**right_join(a, b, by = "x1")**

Join matching rows from a to b.

x1	x2	x3
A	1	T
B	2	F

dplyr::**inner_join(a, b, by = "x1")**

Join data. Retain only rows in both sets.

x1	x2	x3
A	1	T
B	2	F
C	3	NA
D	NA	T

dplyr::**full_join(a, b, by = "x1")**

Join data. Retain all values, all rows.

Filtering Joins

x1	x2
A	1
B	2

dplyr::**semi_join(a, b, by = "x1")**

All rows in a that have a match in b.

x1	x2
C	3

dplyr::**anti_join(a, b, by = "x1")**

All rows in a that do not have a match in b.

y

x1	x2
A	1
B	2
C	3

+

z

x1	x2
B	2
C	3
D	4

=

Set Operations

x1	x2
B	2
C	3

dplyr::**intersect(y, z)**

Rows that appear in both y and z.

x1	x2
A	1
B	2
C	3
D	4

dplyr::**union(y, z)**

Rows that appear in either or both y and z.

x1	x2
A	1

dplyr::**setdiff(y, z)**

Rows that appear in y but not z.

Binding

x1	x2
A	1
B	2
C	3
B	2
C	3
D	4

dplyr::**bind_rows(y, z)**

Append z to y as new rows.

x1	x2	x1	x2
A	1	B	2
B	2	C	3
C	3	D	4

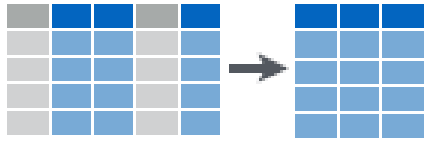
dplyr::**bind_cols(y, z)**

Append z to y as new columns.

Caution: matches rows by position.

join()

Subset Variables (Columns)



dplyr::**select(iris, Sepal.Width, Petal.Length, Species)**

Select columns by name or helper function.

Helper functions for select - ?select

select(iris, contains(":"))

Select columns whose name contains a character string.

select(iris, ends_with("Length"))

Select columns whose name ends with a character string.

select(iris, everything())

Select every column.

select(iris, matches(".t."))

Select columns whose name matches a regular expression.

select(iris, num_range("x", 1:5))

Select columns named x1, x2, x3, x4, x5.

select(iris, one_of(c("Species", "Genus")))

Select columns whose names are in a group of names.

select(iris, starts_with("Sepal"))

Select columns whose name starts with a character string.

select(iris, Sepal.Length:Petal.Width)

Select all columns between Sepal.Length and Petal.Width (inclusive).

select(iris, -Species)

Select all columns except Species.

select()

Subset Observations (Rows)

dplyr::filter(iris, Sepal.Length > 7)

Extract rows that meet logical criteria.

dplyr::distinct(iris)

Remove duplicate rows.

dplyr::sample_frac(iris, 0.5, replace = TRUE)

Randomly select fraction of rows.

dplyr::sample_n(iris, 10, replace = TRUE)

Randomly select n rows.

dplyr::slice(iris, 10:15)

Select rows by position.

dplyr::top_n(storms, 2, date)

Select and order top n entries (by group if grouped data).

Logic in R - ?Comparison, ?base::Logic			
<	Less than	!=	Not equal to
>	Greater than	%in%	Group membership
==	Equal to	is.na	Is NA
<=	Less than or equal to	!is.na	Is not NA
>=	Greater than or equal to	&, |, !, xor, any, all	Boolean operators

filter()

Summarise Data

dplyr::**summarise(iris, avg = mean(Sepal.Length))**

Summarise data into single row of values.

dplyr::**summarise_each(iris, funs(mean))**

Apply summary function to each column.

dplyr::**count(iris, Species, wt = Sepal.Length)**

Count number of rows with each unique value of variable (with or without weights).

Summarise uses **summary functions**, functions that take a vector of values and return a single value, such as:

dplyr::**first**
First value of a vector.

dplyr::**last**
Last value of a vector.

dplyr::**nth**
Nth value of a vector.

dplyr::**n**
# of values in a vector.

dplyr::**n_distinct**
# of distinct values in a vector.

IQR
IQR of a vector.

min
Minimum value in a vector.

max
Maximum value in a vector.

mean
Mean value of a vector.

median
Median value of a vector.

var
Variance of a vector.

sd
Standard deviation of a vector.

summarise()

Análisis de datos con R

Cristóbal Honores