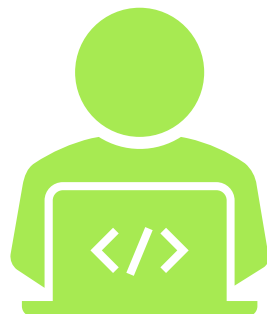




Curso Introductorio a R

Introducción: Presentación del curso



Reglas del taller

1. Seguir las reglas Netiqueta de la UANL.
2. Respeto a los compañeros presentes.
3. Todos estamos aprendiendo: seamos amables y comprensivos.
4. Se mantendrá micrófono y cámaras apagados durante las sesiones del taller para mejorar la conectividad.
5. Las preguntas se realizan los últimos minutos de la sesión del taller levantando la mano y esperando su turno.

Reglas del taller

6. Las preguntas fuera del horario del taller se tienen que escribir en el canal de Teams, anexando una imagen o el código del problema o cuestión.
7. Los comentarios, ideas, textos, trabajos, etc. publicadas en el canal deberán ser, exclusivamente, relacionadas con el tema del taller, dando crédito al autor cuando corresponda.
8. Cualquier publicación ajena al tema del taller será eliminada y contará como una llamada de atención.



Antes que cualquier cosa...

...¡tenemos que instalar R!

Entra a <https://cran.revolutionanalytics.com/>



[\[Home\]](#)

Download

[CRAN](#)

R Project

[About R](#)

[Logo](#)

[Contributors](#)

[What's New?](#)

[Reporting Bugs](#)

[Conferences](#)

[Search](#)

[Get Involved: Mailing Lists](#)

[Developer Pages](#)

[R Blog](#)

R Foundation

[Foundation](#)

[Board](#)

[Members](#)

[Donors](#)

[Donate](#)

Help With R

[Getting Help](#)

Documentation

The R Project for Statistical Computing

Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and MacOS. To [download R](#), please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our [answers to frequently asked questions](#) before you send an email.

News

- [R version 4.0.2 \(Taking Off Again\)](#) has been released on 2020-06-22.
- [useR! 2020 in Saint Louis has been cancelled](#). The European hub planned in Munich will not be an in-person conference. Both organizing committees are working on the best course of action.
- [R version 3.6.3 \(Holding the Windsock\)](#) has been released on 2020-02-29.
- You can support the R Foundation with a renewable subscription as a [supporting member](#)


News via Twitter

 The R Foundation Retweeted



useR2020muc

@useR2020muc





 Please let us know how you liked [#useR2020](#)

Even if you joined only 1 event or watched 1 talk, your feedback is helpful to us and will help plan future

[#rstats](#) events

https://twitter.com/R_Forwards/status/1290193576169803776

Index of /src/base/R-4

Name	Last modified	Size	Description
<hr/>			
 Parent Directory		-	
 R-4.0.0.tar.gz	2020-04-24 09:05	32M	
 R-4.0.1.tar.gz	2020-06-06 09:05	32M	
 R-4.0.2.tar.gz	2020-06-22 09:05	32M	

Antes que cualquier cosa...



Instalemos.

R Studio



Go to <https://rstudio.com/>

BLACK LIVES MATTER

JOIN US AND DONATE



DOWNLOAD

SUPPORT

DOCS

COMMUNITY



Products ▾

Solutions ▾

Customers

Resources ▾

About ▾

Pricing

R & Python

The best data science is done in code

LEARN MORE



OPEN SOURCE

Get started with R

RStudio

The premier IDE for R



RStudio Server

RStudio anywhere using a web browser



Shiny Server

Put Shiny applications online



R Packages

Shiny, R Markdown, Tidyverse and more

HOSTED SERVICES

Be our guest, be our guest

RStudio Cloud

Do, share, teach and learn data science



RStudio Public Package Manager

An easy way to access R packages



shinyapps.io

Let us host your Shiny applications

PROFESSIONAL

Enterprise-ready

RStudio Team

The premier software bundle for data science teams



RStudio Server Pro

RStudio for the Enterprise



RStudio Connect

Connect data scientists with decision makers



RStudio Package Manager

Control and distribute packages



RStudio

Take control of your R code

RStudio is an integrated development environment (IDE) for R. It includes a console, syntax-highlighting editor that supports direct code execution, as well as tools for plotting, history, debugging and workspace management. [Click here to see more RStudio features.](#)

RStudio is available in **open source** and **commercial** editions and runs on the desktop (Windows, Mac, and Linux) or in a browser connected to RStudio Server or RStudio Server Pro (Debian/Ubuntu, Red Hat/CentOS, and SUSE Linux).

New in RStudio 1.3

From spell check to a new tutorial pane, there is a lot to love. Now connect remote sessions between RStudio

	Open Source Edition	RStudio Desktop Pro
Overview	<ul style="list-style-type: none">• Access RStudio locally• Syntax highlighting, code completion, and smart indentation• Execute R code directly from the source editor• Quickly jump to function definitions• Easily manage multiple working directories using projects• Integrated R help and documentation• Interactive debugger to diagnose and fix errors quickly• Extensive package development tools	<p>All of the features of open source; plus:</p> <ul style="list-style-type: none">• A commercial license for organizations not able to use AGPL software• Access to priority support• RStudio Professional Drivers• Connect directly to your RStudio Server Pro instance remotely
Support	Community forums only	<ul style="list-style-type: none">• Priority Email Support• 8 hour response during business hours (ET)
License	AGPL v3	RStudio License Agreement
Pricing	Free	\$995/year

[DOWNLOAD RSTUDIO DESKTOP](#)[DOWNLOAD FREE RSTUDIO DESKTOP PRO TRIAL](#)

Choose Your Version

RStudio is a set of integrated tools designed to help you be more productive with R. It includes a console, syntax-highlighting editor that supports direct code execution, and a variety of robust tools for plotting, viewing history, debugging and managing your workspace.

[LEARN MORE ABOUT RSTUDIO FEATURES](#)



RStudio's new solution for every professional data science team. RStudio Team includes RStudio Server Pro, RStudio Connect and RStudio Package Manager.

[LEARN MORE](#)

RStudio Desktop

Open Source License

Free

[Learn more](#)

RStudio Desktop

Commercial License

\$995

/year

[Learn more](#)

RStudio Server

Open Source License

Free

DOWNLOAD

[Learn more](#)

RStudio Server Pro

Commercial License

\$4,975

/year

(5 Named Users)

BUY

[Evaluation](#) | [Learn more](#)

RStudio Desktop 1.3.1093 - [Release Notes](#)

1. Install R. RStudio requires [R 3.0.1+](#). 
2. Download RStudio Desktop. Recommended for your system:





DOWNLOAD RSTUDIO FOR WINDOWS
1.3.1093 | 171.62MB

Requires Windows 10/8/7 (64-bit)

All Installers

Linux users may need to [import RStudio's public code-signing key](#) prior to installation, depending on the operating system's security policy.

RStudio requires a 64-bit operating system. If you are on a 32 bit system, you can use an [older version of RStudio](#).

OS	Download	Size	SHA-256
Windows 10/8/7	 RStudio-1.3.1093.exe	171.62 MB	62b9e60a
OS 10/8/7	 RStudio-1.3.1093.exe	171.62 MB	62b9e60a

RStudio Desktop 1.3.1093 - [Release Notes](#)

1. Install R. RStudio requires [R 3.0.1+](#).
2. Download RStudio Desktop. Recommended for your system:





Requires Windows 10/8/7 (64-bit)



All Installers

Linux users may need to [import RStudio's public code-signing key](#) prior to installation, depending on the operating system's security policy.










RStudio requires a 64-bit operating system. If you are on a 32 bit system, you can use an [older version of RStudio](#).

OS	Download	Size	SHA-256
Windows 10/8/7	 RStudio-1.3.1093.exe	171.62 MB	62b9e60a
OS 10/8/7	 RStudio-1.3.1093.exe	171.62 MB	62b9e60a

All Installers

Linux users may need to [import RStudio's public code-signing key](#) prior to installation, depending on the operating system's security policy.

RStudio requires a 64-bit operating system. If you are on a 32 bit system, you can use an [older version of RStudio](#).

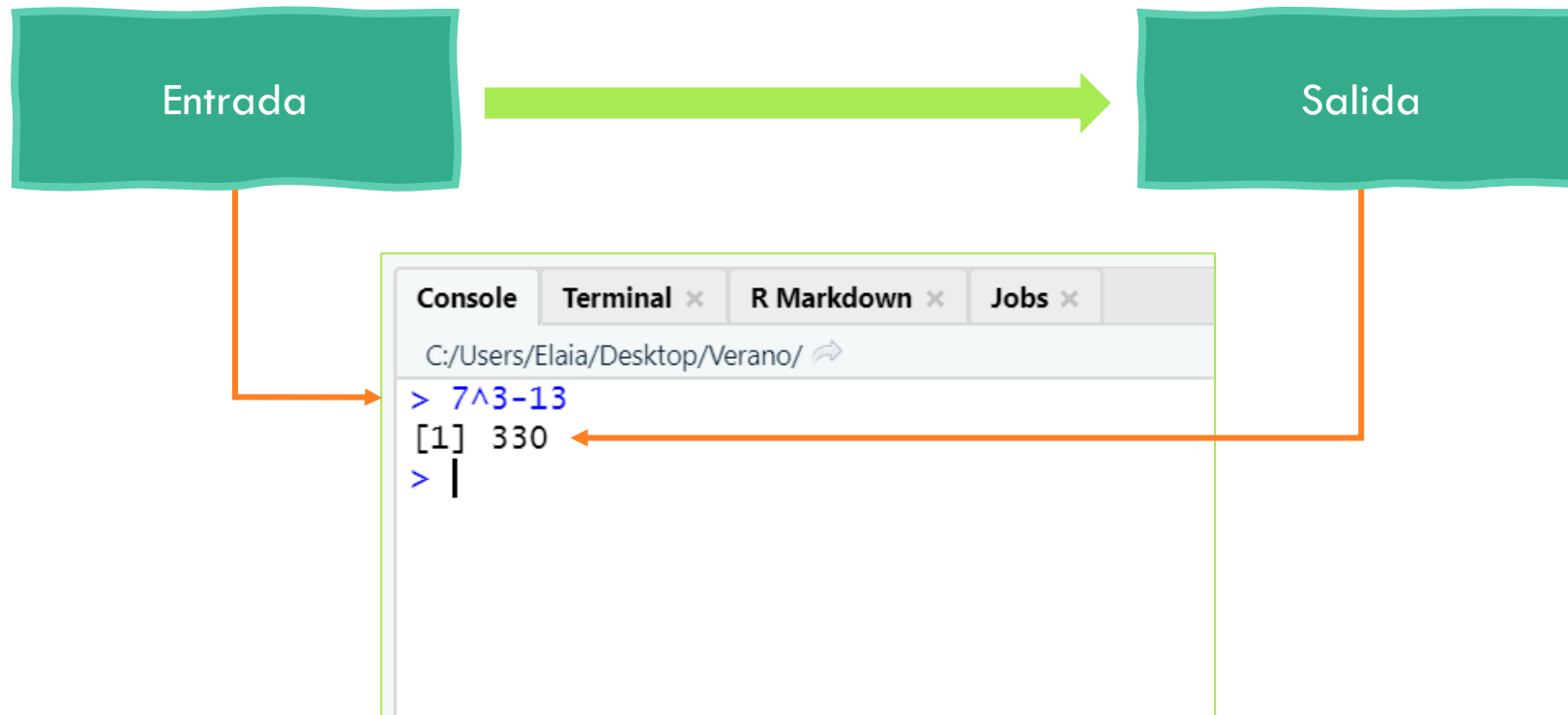
OS	Download	Size	SHA-256
Windows 10/8/7	 RStudio-1.3.1093.exe	171.62 MB	62b9e60a
macOS 10.13+	 RStudio-1.3.1093.dmg	148.66 MB	bdc4d3a4
Ubuntu 16	 rstudio-1.3.1093-amd64.deb	124.33 MB	72f05048
Ubuntu 18/Debian 10	 rstudio-1.3.1093-amd64.deb	126.80 MB	ff222177
Fedora 19/Red Hat 7	 rstudio-1.3.1093-x86_64.rpm	146.96 MB	ed1f6ef8
Fedora 28/Red Hat 8	 rstudio-1.3.1093-x86_64.rpm	151.05 MB	01a978f3
Debian 9	 rstudio-1.3.1093-amd64.deb	127.00 MB	a747f9f9
SLES/OpenSUSE 12	 rstudio-1.3.1093-x86_64.rpm	119.43 MB	5016cbcf
OpenSUSE 15	 rstudio-1.3.1093-x86_64.rpm	128.40 MB	cf47e32d

Instalemos R Studio

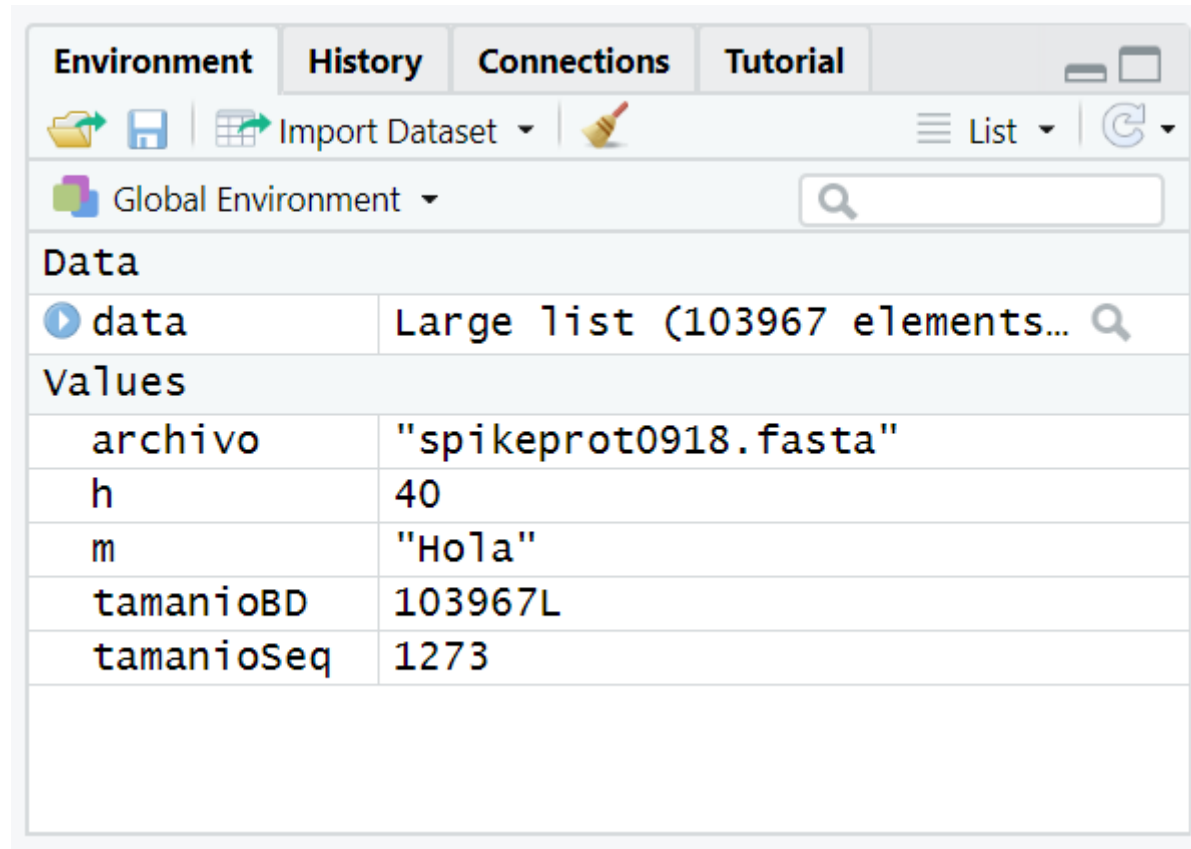


Instalemos.

Entrada y Salida



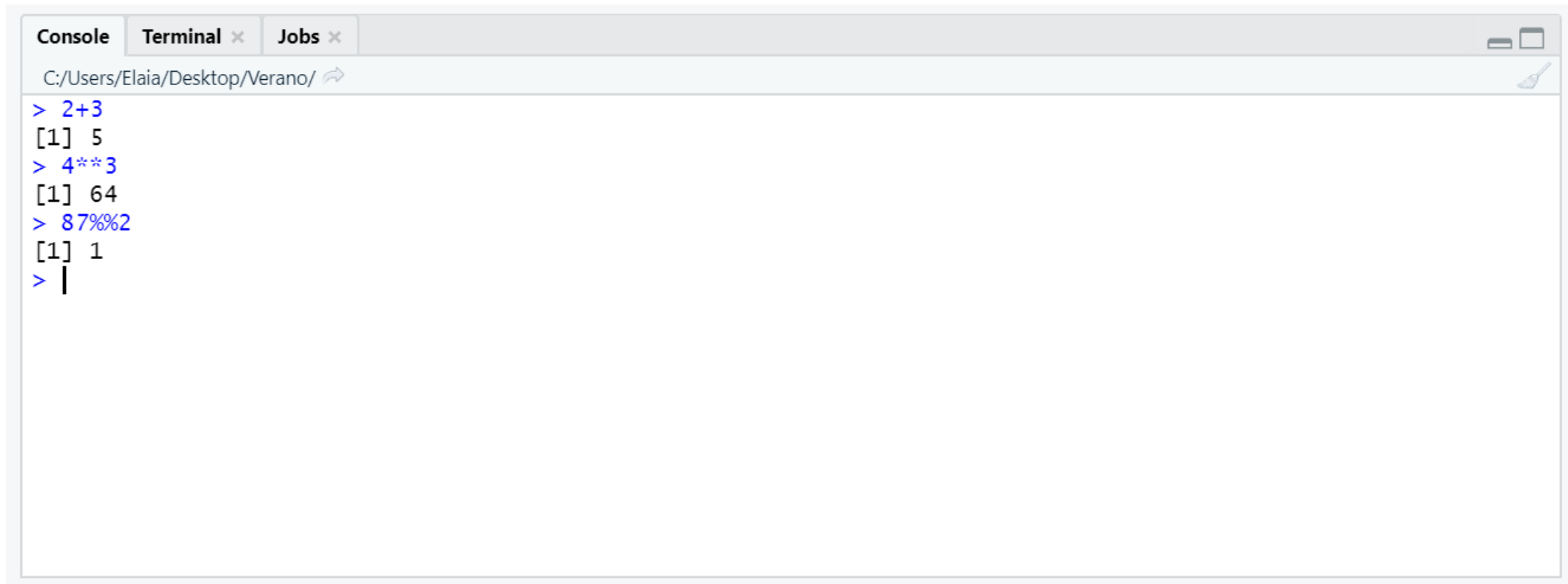
Objetos



The screenshot shows a software interface with a tabbed menu at the top containing 'Environment', 'History', 'Connections', and 'Tutorial'. The 'Environment' tab is active, displaying a toolbar with icons for file operations and a search bar. Below the toolbar, a section labeled 'Global Environment' contains a table of variables. The table has two columns: the variable name and its value. The variables listed are 'data' (a large list), 'archivo' (a file path), 'h' (a number), 'm' (a string), 'tamañoBD' (a number), and 'tamañoSeq' (a number).

Data	
data	Large list (103967 elements...)
Values	
archivo	"spikeprot0918.fasta"
h	40
m	"Hola"
tamañoBD	103967L
tamañoSeq	1273

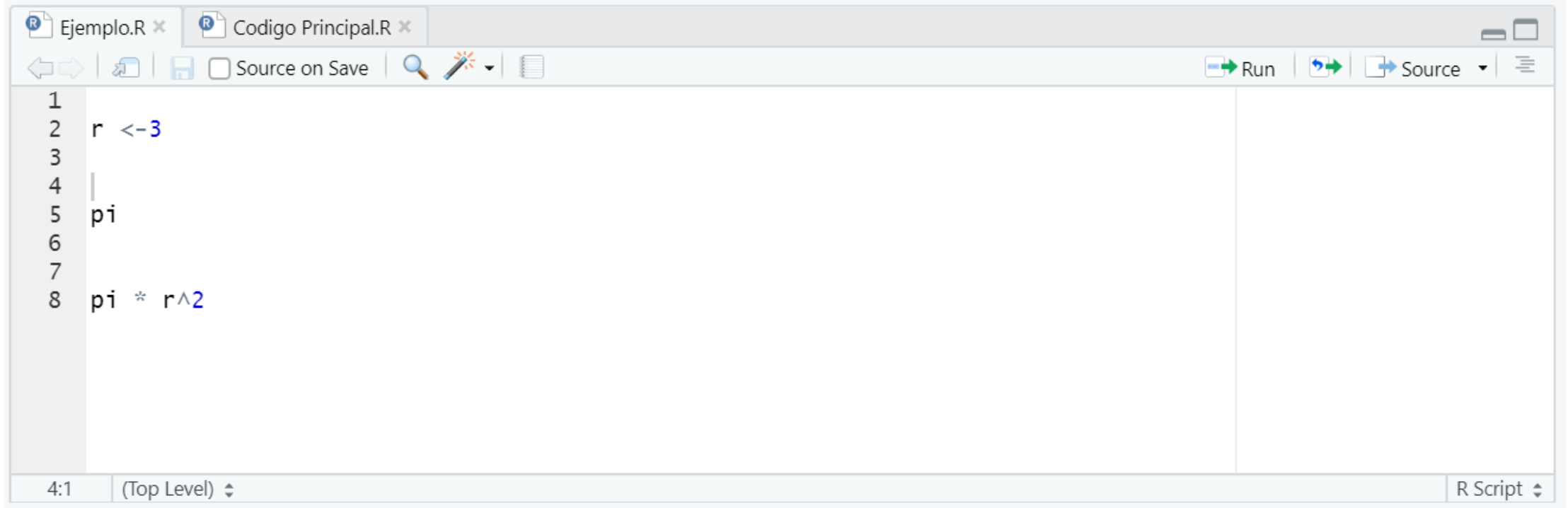
Interfaz de Línea de Comandos



The image shows a JupyterLab console window with three tabs: 'Console', 'Terminal', and 'Jobs'. The 'Console' tab is active, displaying the current directory path 'C:/Users/Elaia/Desktop/Verano/' and a series of arithmetic commands and their outputs. The commands are entered in blue text, and the outputs are in black text. The console shows the results of three calculations: 2+3, 4**3, and 87%%2. The prompt character is '>'.

```
> 2+3  
[1] 5  
> 4**3  
[1] 64  
> 87%%2  
[1] 1  
> |
```

Constantes y Variables

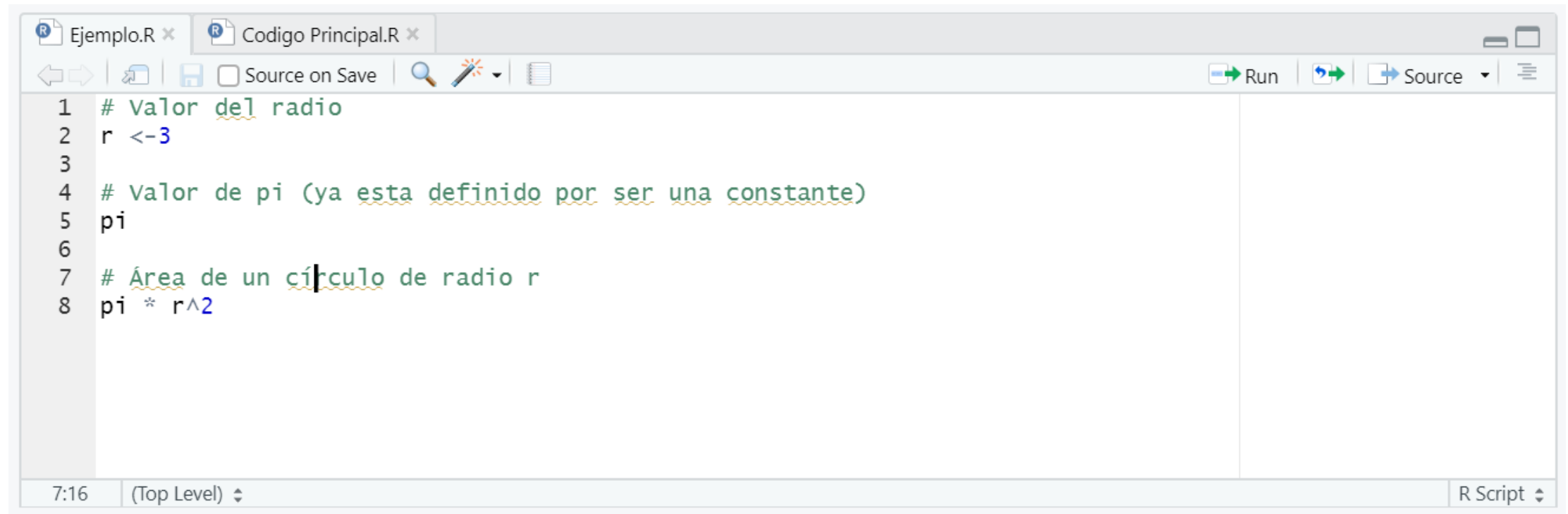


The image shows a screenshot of the R Studio IDE. The top pane displays the source editor with two tabs: 'Ejemplo.R' and 'Codigo Principal.R'. The 'Codigo Principal.R' tab is active, showing the following R code:

```
1  
2 r <- 3  
3  
4 |  
5 pi  
6  
7  
8 pi * r^2
```

The bottom status bar indicates the current position is '4:1' at the '(Top Level)' and the file type is 'R Script'.

Comentarios



The screenshot shows an R script editor window with two tabs: 'Ejemplo.R' and 'Codigo Principal.R'. The 'Codigo Principal.R' tab is active. The editor contains the following R code with comments:

```
1 # Valor del radio
2 r <- 3
3
4 # Valor de pi (ya esta definido por ser una constante)
5 pi
6
7 # Área de un círculo de radio r
8 pi * r^2
```

The interface includes a toolbar with icons for navigation, saving, and running. The status bar at the bottom shows the cursor position at line 7, column 16, and the file type as 'R Script'.

Funciones*

Formato: nombreDeLaFunción (argumentos)

Ejemplos:

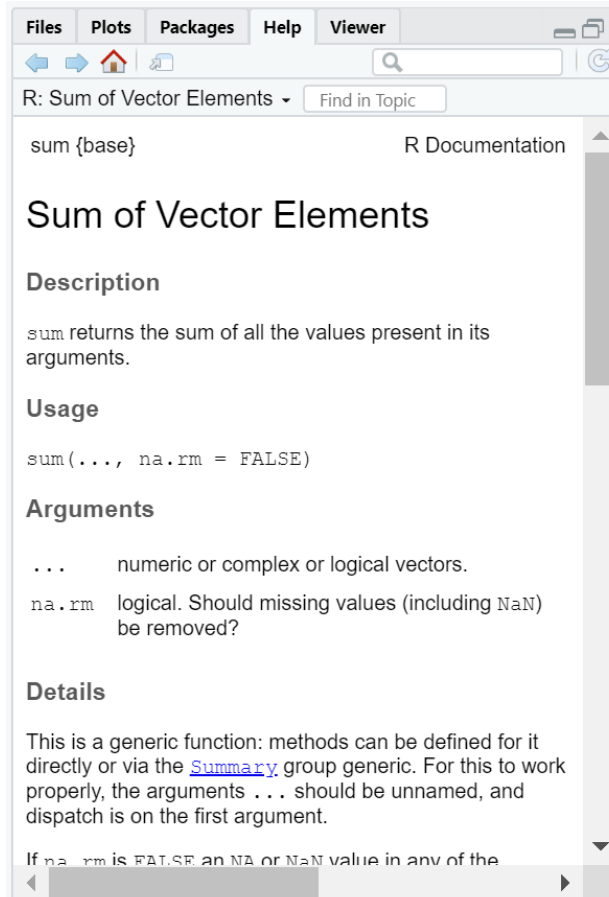
mean (x)

names (x)

length (x)

sum (x)

Documentación



```
help("mean")
```

```
?length
```

```
help(package = "stats")
```


Directorio de Trabajo

Conocer el directorio de trabajo: `getwd()`

Cambiar el directorio de trabajo: `setwd("[Aquí va el directorio"])`

Ver archivos del directorio: `list.files()`

Ver directorios del directorio: `list.dirs()`

Sesión de Trabajo

Conoce los objetos de tu sesión actual: `ls()`

Elimina objetos de tu sesión actual: `rm(...)`

Paquetes

Conoce los objetos de tu sesión actual: `install.packages(ggplot2)`

Cargar el paquete a tu sesión actual: `library(ggplot2)`

Tipos de datos

Estructuras de datos: Vectores

Estructuras de datos

Objetos que contienen datos.

Dimensiones	Homogéneas	Heterogéneas
1	vector	list
2	matrix	dataframe
n	array	

Vectores

Colección de 1 o más datos del mismo tipo.

Propiedades:

1. Tipo. Solo pueden contener datos de un solo tipo.
2. Longitud. Cantidad de elementos del vector.
3. Atributos. Metadatos de los vectores (no veremos esto en este curso).

Tipos de Vectores

Numérico (Enteros y Dobles)

Lógico

Caracter

Valores Faltantes (NA)

NULL

Valores Numéricos

Diferencias entre enteros y dobles:

1. Los dobles son aproximaciones.
2. Los números enteros tienen un valor especial (`NA`). Los dobles tienen cuatro: `NA`, `NaN`, `Inf` e `-Inf`.

NA, NaN, Inf, -Inf

Función	0	Inf	NA	NaN
<code>is.finite()</code>	x			
<code>is.infinite()</code>		x		
<code>is.na()</code>			x	x
<code>is.nan()</code>				x

Valores Faltantes (NA)

Tipo	NA
Lógico	NA
Entero	NA_integer_
Doble	NA_real_
Caracter	NA_character_

Normalmente no es necesario conocerlos (¡bendita coerción!), salvo en funciones específicas que son estrictas con sus datos de entrada.

Funciones relacionadas

`a:b`  `a, b enteros`

`c(x)`

`typeof(x)`

`length(x)`

Coerción

`a:b` ← `a, b` enteros

`c(x)`

`typeof(x)`

`length(x)`

Estructuras de datos: Matrices y Arreglos

Estructuras de datos

Objetos que contienen datos.

Dimensiones	Homogéneas	Heterogéneas
1	vector	list
2	matrix	dataframe
n	array	

Matrices y Arrays

Pueden ser descritas como **vectores multidimensionales**. Es decir, solo contienen datos de un solo tipo pero ordenados en 2 ó más dimensiones.

Propiedades (heredadas de los vectores):

1. Tipo. Solo pueden contener datos de un solo tipo.
2. Dimensiones. Dos para matrices, n para arreglos.
3. Atributos. Metadatos de los vectores (no veremos esto en este curso).

Notas sobre Arrays

Su uso es poco común. Es preferible usar listas en lugar de arrays.

Valores Numéricos

Diferencias entre enteros y dobles:

1. Los dobles son aproximaciones.
2. Los números enteros tienen un valor especial (`NA`). Los dobles tienen cuatro: `NA`, `NaN`, `Inf` e `-Inf`.

NA, NaN, Inf, -Inf

Función	0	Inf	NA	NaN
<code>is.finite()</code>	x			
<code>is.infinite()</code>		x		
<code>is.na()</code>			x	x
<code>is.nan()</code>				x

Valores Faltantes (NA)

Tipo	NA
Lógico	NA
Entero	NA_integer_
Doble	NA_real_
Caracter	NA_character_

Normalmente no es necesario conocerlos (¡bendita coerción!), salvo en funciones específicas estrictas con sus datos de entrada.

Creación de matrices

```
matrix(datos, nrow=3, ncol=4)
```

Ojo: Si la cantidad de datos no es igual a la cantidad de celdas de la matriz, ocurrirán dos cosas distintas.

Creación de matrices con vectores

`cbind()`

`rbind()`

Ojo: Si los vectores no son del mismo tamaño, se reciclarán los valores de los vectores.

Dimensiones

`dim()`

Ojo: Si usamos esta función con un vector, no nos dará el valor que esperamos.

Operaciones con matrices

La operación se aplica a cada elemento de la matriz (element-wise).

Funciones especiales

%*%

ncol()

nrow()

det()

t()

which()

solve(A,B)

solve(A)

$$AX = B$$

Funciones especiales

`is.matrix()`

`is.vector()`

`as.matrix()`

`as.vector()`

Estructuras de datos: Listas

Estructuras de datos

Objetos que contienen datos.

Dimensiones	Homogéneas	Heterogéneas
1	vector	list
2	matrix	dataframe
n	array	

Listas

También conocidas como “vectores recursivos” por la posibilidad de contener objetos de su misma clase.

Propiedades:

1. Tipo. Pueden contener datos de cualquier tipo.
2. Largo. Es unidimensional. El largo es la cantidad de elementos que tiene la lista.

Creación de listas

```
list()
```

Ojo: Es posible poner vectores, matrices, arrays, dataframes e incluso listas dentro de las listas.

Funciones especiales

`length()`

Ojo: Es posible poner vectores, matrices, arrays, dataframes e incluso listas dentro de las listas.



Estructuras de datos: Dataframes

Estructuras de datos

Objetos que contienen datos.

Dimensiones	Homogéneas	Heterogéneas
1	vector	list
2	matrix	dataframe
n	array	

Dataframes

Son el tipo de dato más común en análisis de datos.

Propiedades:

1. Tipo. Pueden contener datos de cualquier tipo (por columna).
2. Dimensiones. Bidimensional. Como una matriz, pero “más flexible”.

Creación de dataframes

```
data.frame()
```

Ojo: Todos los vectores proporcionados deben ser del mismo largo. Se tienen que dar la cantidad de vectores tantas como columnas queramos que contenga nuestro dataframe.

Ojo: Podemos asignar un nombre a cada vector que se convertirá en nuestro nombre de columna.

Funciones especiales

`dim()`

`length()`

`names()`

Ojo: Es posible poner vectores, matrices, arrays, dataframes e incluso listas dentro de las listas.

Estructuras de control

Estructuras de control

Establecen condicionales en nuestros códigos. ¿Qué condiciones deben cumplirse para realizar una operación o qué debe ocurrir para ejecutar una función?

Estructura de control	Descripción
if, else	Si... de otro modo...
for	Para cada uno en...
while	Mientras...
repeat	Repetir...
break	Interrupción
next	Siguiente

IF / ELSE

Estructura:

```
if (valorBooleano) {  
    operacionesSiElValorBooleanoEsTRUE  
}else{  
    operacionesSiElValorBooleanoEsFALSE  
}
```

Normalmente el `valorBooleano` **es uno o varios condicionales.**

IF / ELSE

Es possible “vectorizar” un `if` utilizando la función `ifelse()`

FOR

Estructura:

```
for(elemento in objeto) {  
    operacionConElemento  
}
```

Hay funciones populares en R que hacen operaciones con elementos que no son vectores de forma más eficiente (para algunos casos) que un `for`, como son las funciones de la familia `apply`.

WHILE

Estructura:

```
while (valorBooleano) {  
    operaciones  
}
```

Normalmente el `valorBooleano` es uno o varios condicionales.

NOTA: ¡Proceder con precaución!

BREAK / NEXT

La palabra reservada `break` nos permite interrumpir un bucle.

La palabra reservada `next` nos permite avanzar a la siguiente iteración.

REPEAT

Estructura:

```
repeat{  
    operaciones  
    ...  
    unBreakParaDetener  
}
```

NOTA: ¡Proceder con precaución!