

# Ayudantía 1

Equipo Estadística 2

2025-03-11

## Ayudantías

- Las ayudantías serán semanales y con asistencia.
- Tendrán un enfoque en el refuerzo práctico de los contenidos de clase.
- Habrá controles en las ayudantías; estos no serán complejos y la idea es que puedan reforzar lo visto en clase.

## Fechas

- Control AyudaLab 1: 27/03/2025
- Control AyudaLab 2: 17/04/2025
- Control AyudaLab 3: 22/05/2025
- Control AyudaLab 4: 05/06/2025
- Control AyudaLab 5: 03/07/2025

\*\*Finalmente, la idea es que a lo largo de las ayudantías del curso, podamos desarrollar un Libro R que sirva de repositorio de los principales códigos utilizados.

## Contenidos Ayudantía 1

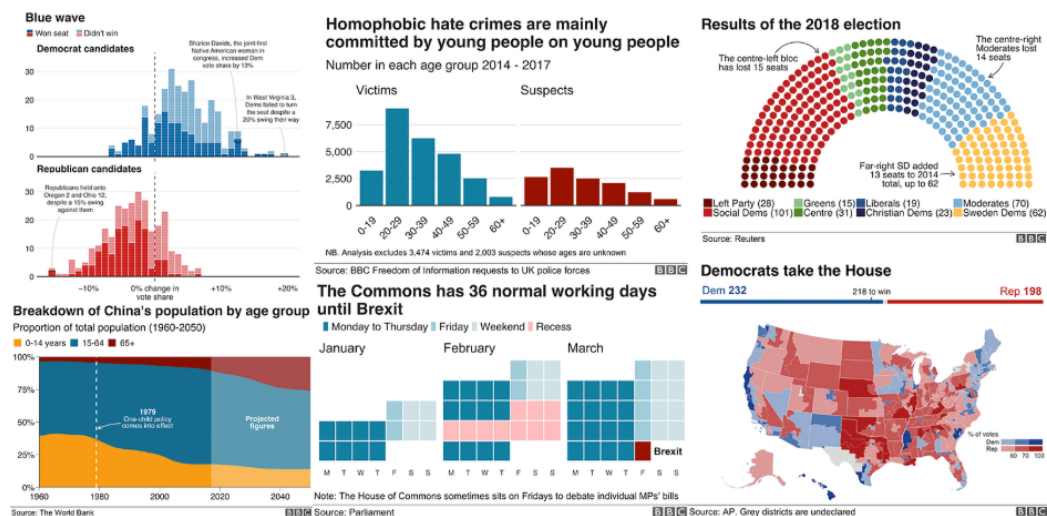
• Esta primera ayudantía tiene la finalidad de repasar los elementos centrales de R, para eso veremos lo siguiente:

1. Qué es R y RStudio
2. Interfaz RStudio
3. Funciones básicas
4. Vectores y bbdd
5. Trabajo en R
6. Paquetes
7. BBDD

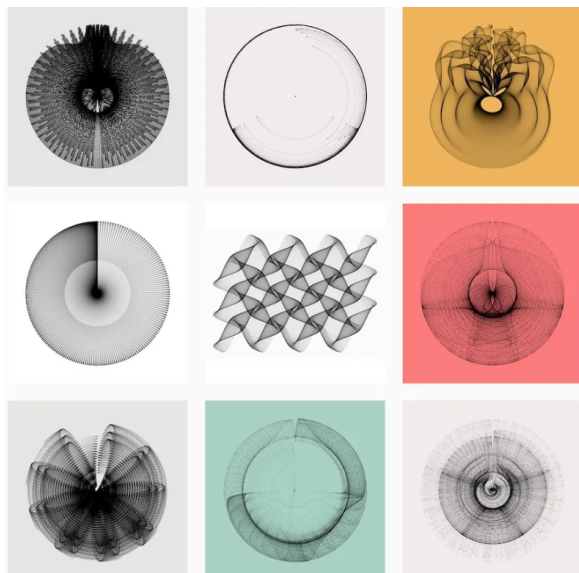
## R y Rstudio

- Existen múltiples programas para el análisis estadístico, siendo los más populares STATA, SPSS y R. R se ha vuelto el software más utilizado en los últimos años, debido a su accesibilidad gratuita, su potencia en análisis/presentación de información y su versatilidad por ser de código abierto. (Eso y muchas otras razones)
- R es el programa/lenguaje de programación y Rstudio es un entorno que actúa como interfaz para hacer más amigable la utilización y navegación por R.
- Si bien el programa se utiliza en mayor medida para análisis estadístico, también tiene gran potencialidad en análisis cualitativos, presentación de información, entre muchas otras funciones.

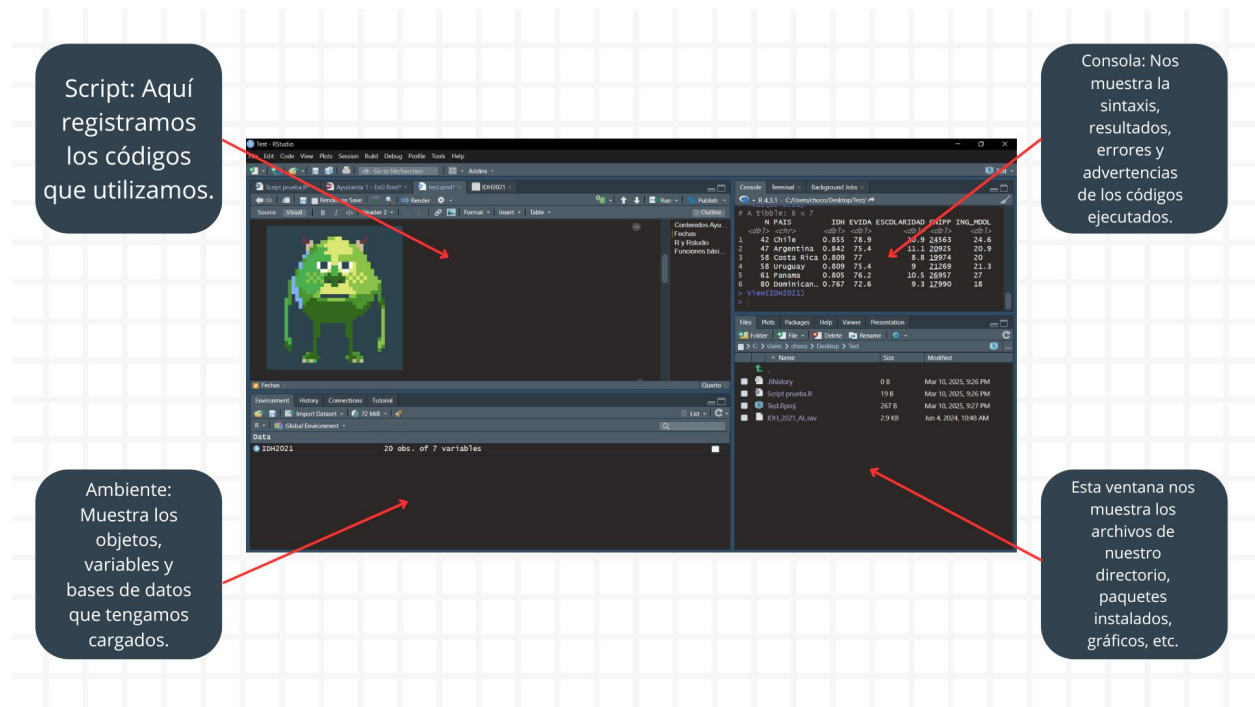
## R y Rstudio



## R y Rstudio



## Rstudio



## Funciones básicas en R

- Como bien recordarán, podemos utilizar R para realizar operaciones básicas:

```
300+200
```

```
## [1] 500
```

```
100-42
```

```
## [1] 58
```

```
4*7
```

```
## [1] 28
```

```
25/5
```

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

- También nos permite establecer comparaciones entre valores:

```
100 == 200 # "==" Esto significa igual a
```

```
## [1] FALSE
```

```
400 > 399 # Mayor a
```

```
## [1] TRUE
```

## Funciones básicas en R

- Además de la adición, sustracción, multiplicación y división, estas son otras funciones básicas en R

Operación	Significado
==	Igual a
!=	Distinto a
>	Mayor que
>=	Mayor o igual que
<	Menor que
<=	Menor o igual que

- Estas se vuelven importantes cuando queramos recodificar y filtrar variables.

## Vectores

- Los vectores son conjuntos de datos del mismo tipo, aplicado a las ciencias sociales estos serían variables. Podemos destacar 3 tipos de vectores más relevantes:

1. Numéricos: valores numéricos, incluye decimales.
2. Carácter: valores alfanuméricos, es decir, letras, números y signos mezclados.
3. Lógicos: valores lógicos. (TRUE o FALSE)

## Vectores

- Para la creación de vectores se debe utilizar la función concatenar.
- En el caso de vectores numéricos, los números decimales se deben escribir con un “.”, ya que, la coma (“,”) es la que indica separación de valores.

```
estaturas <- c(1.85, 1.63, 1.75, 1.57)
```

- En los vectores de carácter, las palabras deben ir entre comillas (“ ”)

```
nombres <- c("Ana", "Alejandro", "Andrés", "María")
```

- En los vectores lógicos se utiliza TRUE (T) y FALSE (F). Tanto las palabras, como la letra funcionan, en cualquier caso, deben ir en mayúscula.

```
mayorde18 <- c(T, F, T, T)
```

## Base de datos

- Ya habiendo creado vectores, los podemos combinar para crear una base de datos.

```
bddd <- data.frame (nombres, estaturas, mayorde18)
print(bddd)
```

```
##      nombres estaturas mayorde18
## 1      Ana      1.85      TRUE
## 2 Alejandro      1.63      FALSE
## 3    Andrés      1.75      TRUE
## 4    María      1.57      TRUE
```

## Ejercicio 1

- Hacer una base de datos que contenga los siguientes vectores:
- casas: Casa1, Casa2, Casa3, Casa4
- ingresos: 30, 70, 50, 300
- npersonas: 3, 4, 2, 3

**Al crear la base de datos, es fundamental que todas las variables contengan la misma cantidad de datos.**

## R project

- En nuestro quehacer de sociólogos es crucial mantener el orden, claridad, transparencia y replicabilidad de nuestros análisis. Esto es fundamental para que otras personas puedan comprender y reproducir nuestros resultados.
- Una de las mejores formas de garantizar el orden y transparencia del trabajo en R es utilizando “R project”. Estos son entornos en que se almacenan datos, códigos y procesos del trabajo realizado.
- Para crear uno apretaremos File -> New Project. Esto creará una carpeta que contenga todo el trabajo que vayamos haciendo. También funcionará como directorio, permitiéndonos cargar bases de datos externas desde esa carpeta.

## Paquetes y bases de datos

- Los paquetes (Packages) son funciones descargables que permiten realizar una diversidad de tareas en R, como, por ejemplo, análisis estadísticos complejos, creación de gráficos, manipulación de bases de datos y muchas otras funcionalidades.
- Las bases de datos normalmente vienen en formato sav (SPSS), dta (STATA), xlsx (Excel) y csv (Excel con valores separados por coma). Para cada una utilizaremos los siguientes paquetes:

Formato	Programa	Paquete	Función
.sav	SPSS	haven	read_sav(“base.sav”)
.dta	STATA	haven	read_dta(“base.dta”)
.xlsx	Excel	readxl	read_xlsx(“base.xlsx”)

Formato	Programa	Paquete	Función
.csv	Excel con valores separados por comas	readr	read_csv("base.csv")

## Cargar paquetes y BBDD

• La mayoría de los paquetes no vienen por defecto en R, entonces, debemos instalarlos. Para esto seguiremos el siguiente procedimiento:

### 1. Instalar la librería deseada

```
install.packages("haven")
```

```
## Installing package into 'C:/Users/choco/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

*#Aquí siempre debe ir el paquete entre comillas.*

### 2. Llamar a la librería instalada

```
library(haven)
```

```
## Warning: package 'haven' was built under R version 4.3.3
```

*#Aquí sin comillas*

### 3. Utilizar las funciones de la librería

```
lapop <- read_sav("LAPOP 2023.sav")
```

*#Se asigna un nombre a la base de datos*

## Manipulación y observación de bases de datos

• Antes de hacer cualquier análisis, es muy importante observar la base de datos y las variables que nos interesan. En este caso quiero analizar las variables edad (q2) y escala política izquierda-derecha del 1 al 10 (l1), ambas de la bases de datos Lapop.

• Una buena práctica es separar en una nueva base las variables que queremos utilizar.

## Manipulación y observación de bases de datos

• Como vieron en Estadística 1, existe una librería que facilita la manipulación de datos, esta es tidyverse. Este paquete nos permite filtrar, seleccionar, graficar, etc. En este caso queremos seleccionar las variables de interés q2 y l1, para esto necesitamos el comando select()

```
install.packages("tidyverse") #Instalamos la librería
```

```
## Installing package into 'C:/Users/choco/AppData/Local/R/win-library/4.3'
## (as 'lib' is unspecified)
```

```
## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
## C:\Users\choco\AppData\Local\Temp\Rtmpg3vxd\downloaded_packages
```

```
library(tidyverse) #Llamamos a la libreria
```

```
## Warning: package 'tidyverse' was built under R version 4.3.3
```

```
## Warning: package 'dplyr' was built under R version 4.3.3
```

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.4      v readr      2.1.4
## v forcats    1.0.0      v stringr   1.5.0
## v ggplot2    3.4.2      v tibble    3.2.1
## v lubridate  1.9.2      v tidyr     1.3.0
## v purrr      1.0.1
```

```
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
lapopselect<-lapop %>% select(q2, l1) #El %>% es un pipeline, es decir, una tubería que une en orden di
```

```
#Ahora exploraré la nueva base con el comando head()
```

```
head(lapopselect)
```

```
## # A tibble: 6 x 2
##   q2      l1
##   <dbl> <lbl>
## 1  50    NA
## 2  50     1 [Izquierda]
## 3  29    NA
## 4  61    10 [Derecha]
## 5  44    NA
## 6  19     5
```

## Ejercicio 2

- Crear un proyecto de R nuevo.
- Abrir un script nuevo.
- Descargar y abrir la base de datos IDH\_2021\_AL.

- Seleccionar las variables “índice de desarrollo humano” (IDH) y “esperanza de vida” (EVIDA).
- Luego observar la nueva base de datos.

**Al descargar la base, es importante siempre verificar el formato en el que viene, esto para saber qué paquete es necesario utilizar.**