

DataViz - Introducción a la visualización de datos con R

Escuela de Métodos de Análisis Sociopolítico

Pablo Cabrera Álvarez
Departamento de Sociología y Comunicación (USAL)

18 junio 2018

En este curso...

- ▶ pablocal@usal.es || @pablocalv
- ▶ Es un curso de visualización de datos
- ▶ Utiliza tus propios datos y juega
- ▶ Dudas sobre trabajos
- ▶ R is cool!

Materiales en...

<https://github.com/pablocal/EMASdataviz>

Día 1 - Introducción a la visualización de datos

- ▶ ¿Visualización de datos?
- ▶ Aspectos teórico-prácticos de la visualización de datos
- ▶ Introducción a la gramática de gráficos
- ▶ Introducción a R y R-studio (**SP**)

Día 2 - Gráficos con ggplot2 (I)

- ▶ La gramática de gráficos y ggplot2: data, aesthetics y geoms
(SP)
- ▶ Stats **(SP)**
- ▶ Presentación del proyecto

Día 3 - Gráficos con ggplot2 (II)

- ▶ Aspectos estéticos de los gráficos: facets y theme (**SP**)
- ▶ Extensiones de ggplot (**SP**)

Introducción a la visualización de datos

Agenda día 1

- ▶ **15.30 - 16.30** Introducción a la visualización de datos y gramática de gráficos
- ▶ **16.30 - 16.45** *Coffee break*
- ▶ **16.45 - 18.00** Introducción a R y R-studio
- ▶ **18.00 - 19.30** SP: Manejo de datos en R

Los gráficos son visuales

048932754628527485267542625548655464
147895236541258965478522106654899475
158544522445214525150151585478563637

Los gráficos son visuales

04**8**93275462**8**5274**8**526754262554**8**655464
147**8**9523654125**8**96547**8**522106654**8**99475
15**8**5445224452145251501515**8**547**8**563637

Los gráficos cuentan historias

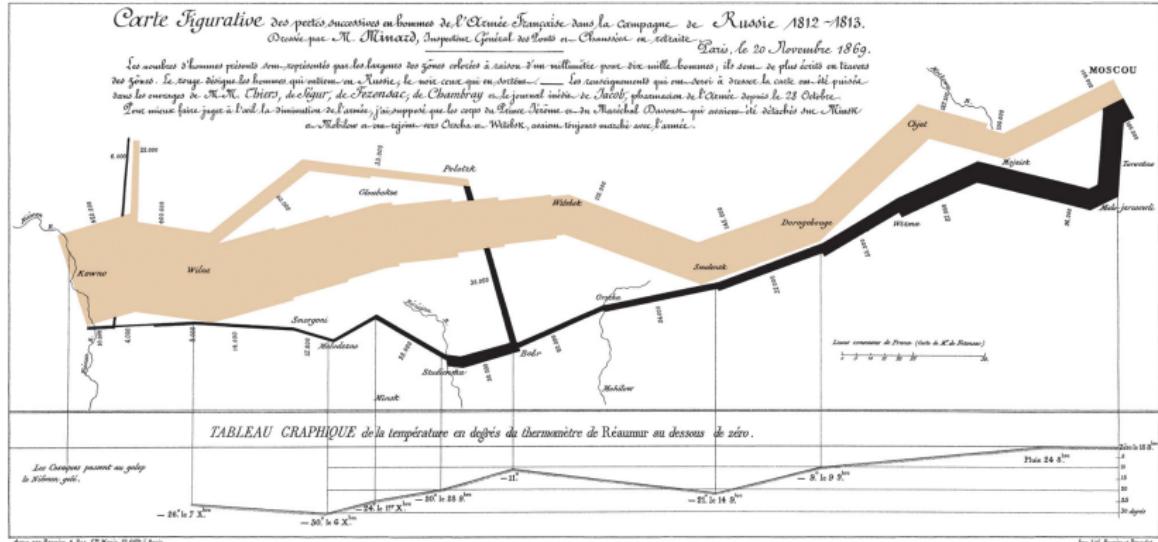


Figure 1: Gráfico de Minard (1869) sobre la campaña de Napoleón en Rusia

¿Qué variables están representadas en este gráfico?

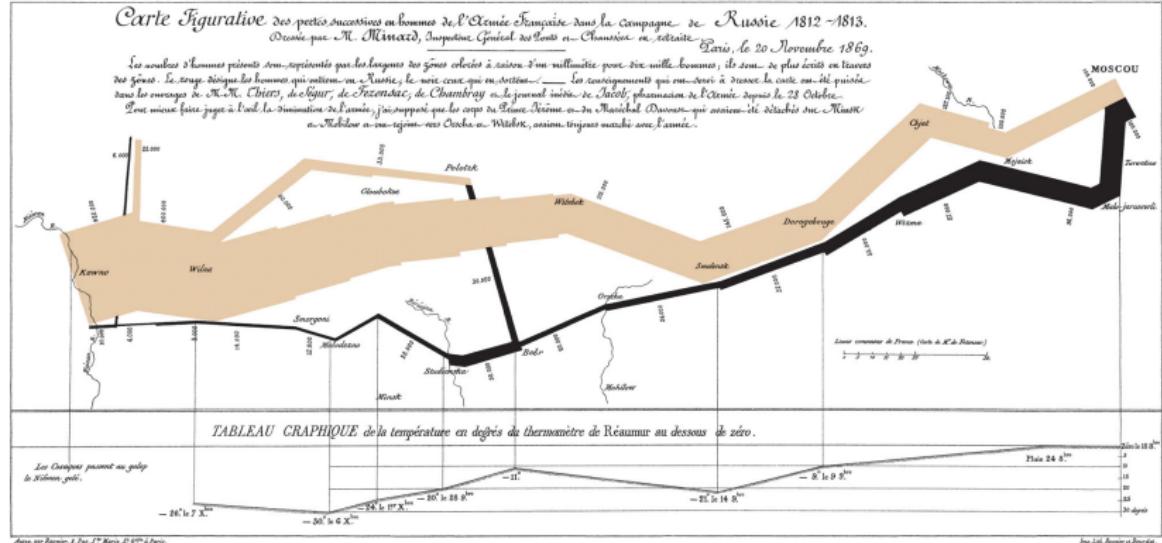


Figure 2: Gráfico de Minard (1869) sobre la campaña de Napoleón en Rusia

Variables en el gráfico de Minard

- ▶ Tamaño del ejército
- ▶ Localización del ejército (latitud)
- ▶ Localización del ejército (longitud)
- ▶ Dirección de los movimientos del ejército
- ▶ Retirada o avance del ejército
- ▶ Temperatura

¿Para qué data viz?

¿Para qué data viz?

- ▶ Explorar datos
- ▶ Analizar datos
- ▶ Presentar resultados

Explorar y analizar

```
cor(anscombe$x1, anscombe$y1) # r pearson x1 y1
```

```
## [1] 0.8164205
```

```
cor(anscombe$x2, anscombe$y2) # r pearson x2 y2
```

```
## [1] 0.8162365
```

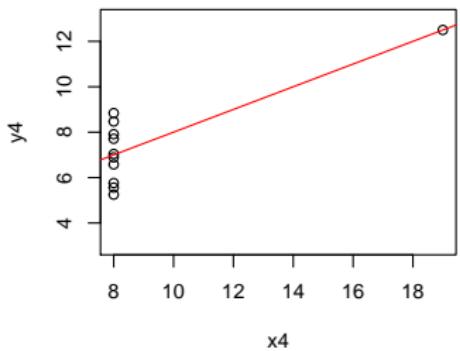
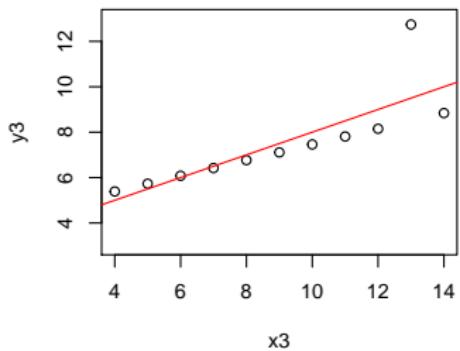
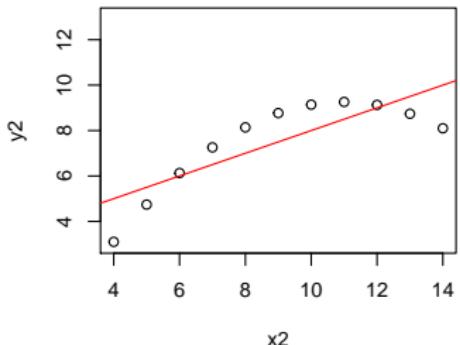
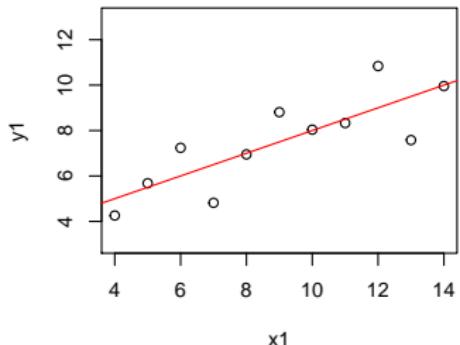
```
cor(anscombe$x3, anscombe$y3) # r pearson x3 y3
```

```
## [1] 0.8162867
```

```
cor(anscombe$x4, anscombe$y4) # r pearson x4 y4
```

```
## [1] 0.8165214
```

Explorar y analizar



Comunicar resultados

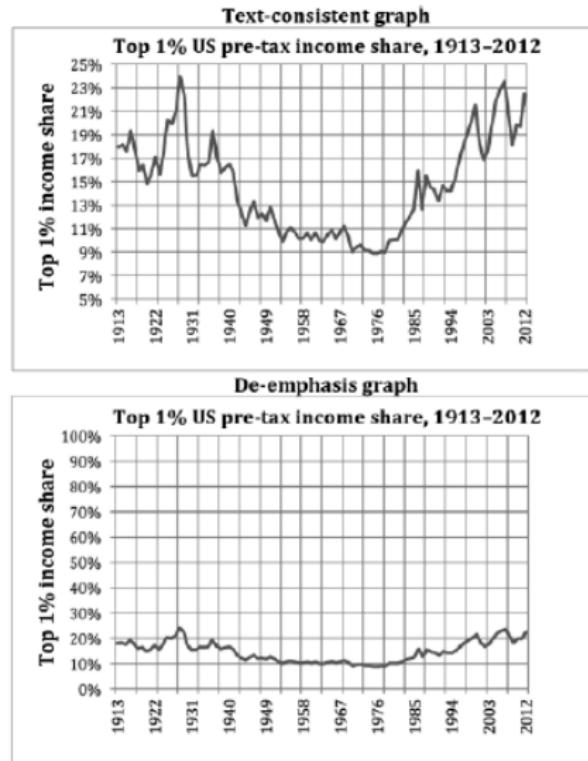


Figure 3: Hughes (2015)

Teoría de gráficos

- ▶ Los gráficos han sido tratados desde los '50 (e.g. incluir tres escalas, dobles dimensiones, colores...) (Haemer, 1947, 1948, 1949)
- ▶ Sistematización de la producción de gráficos (Tukey, 1972, 1977)
- ▶ Tufte (1983) *The Visual Display of Quantitative Information*
- ▶ Cleveland (1985) *Graphical Perception and Graphical Methods for Analyzing Scientific Data*
- ▶ *The Grammar of Graphics* (Wilkinson, 2005) y su adaptación a R a través de *ggplot2* por Wickham (2015)

¿Qué es un gráfico?

Los gráficos son elementos visuales que muestran medidas cuantitativas usando una combinación de puntos, líneas, un sistema de coordenadas, números, símbolos, palabras formas y colores.

(Tufte, 1983: 3)

¿Existen reglas para crear gráficos?

Existen principios generales,
líneas de actuación,
y un largo repertorio de ejemplos sobre lo que **no** hay que hacer.

Pensar en tres aspectos de los gráficos...

- ▶ **Datos**
- ▶ **Precisión** de la representación
- ▶ **Claridad** de la representación

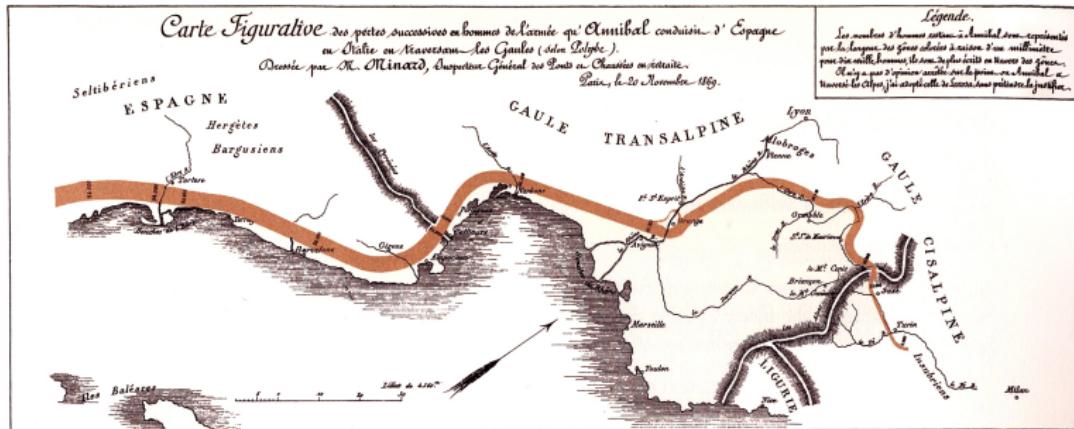


Figure 4: Minard (1869)

Datos: Principio 1

La calidad de un gráfico está determinada por la calidad de los datos representados (Tufte, 1983)

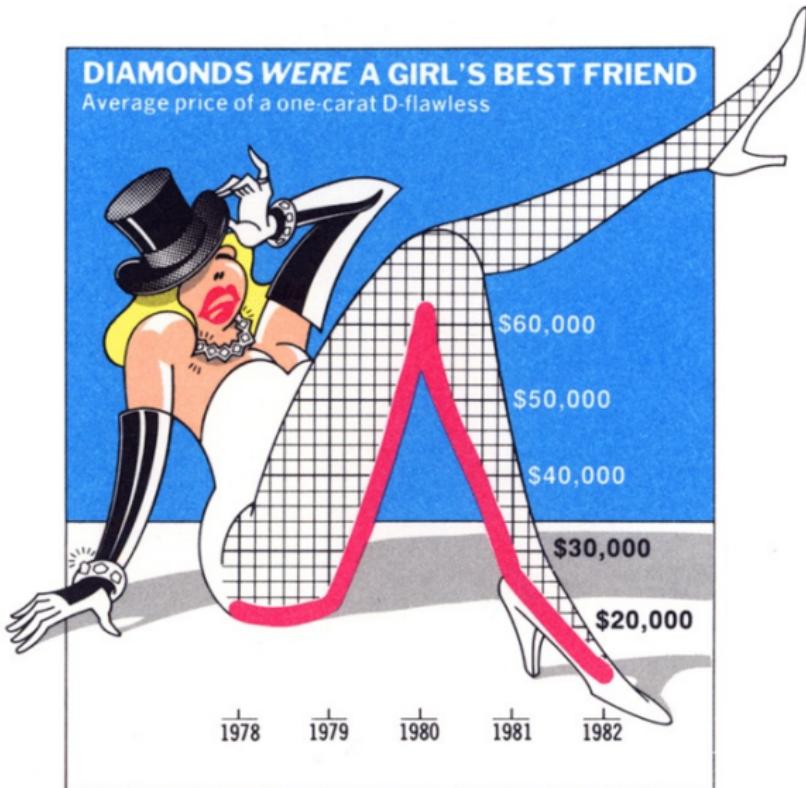


Figure 5: Nigel Holmes (1982) en revista TIME

Datos: Principio 2

Los datos siempre deben ser el centro del gráfico (Kelleher y Wagener, 2011)

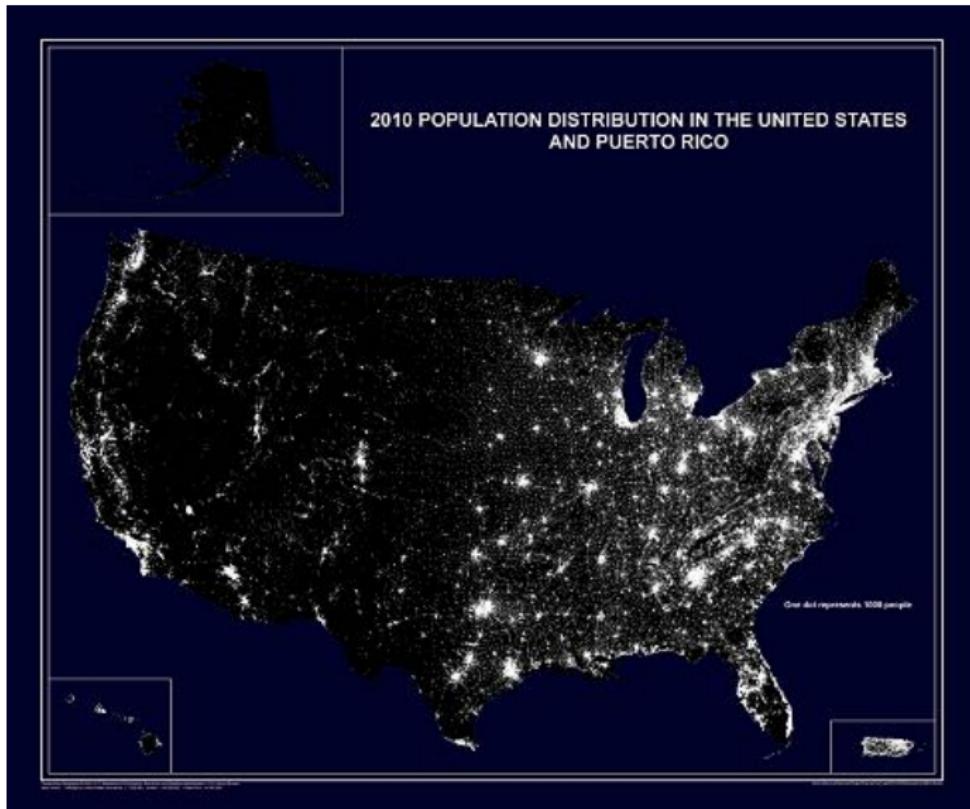
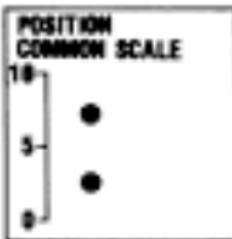


Figure 6: US Census Bureau (2011)

Datos: Principio 3

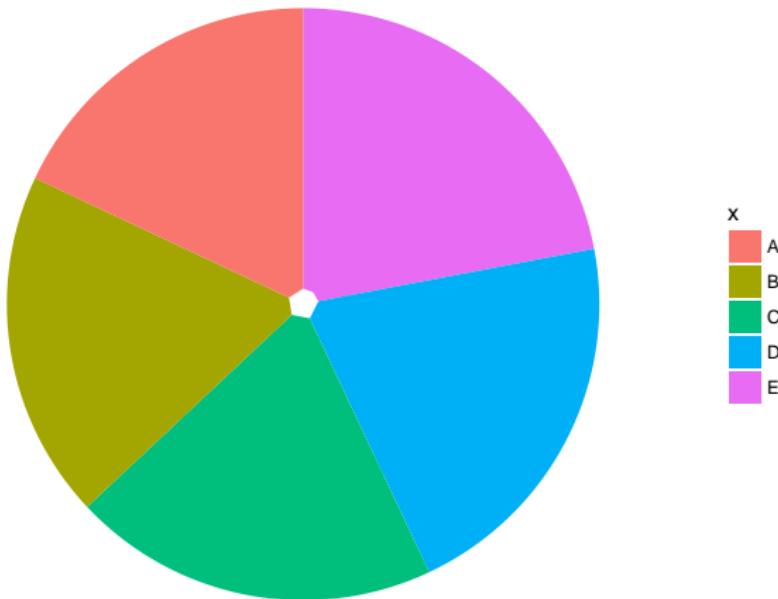
*Un gráfico debe representar el máximo de datos posible
(Wainer, 1984)*

La precisión y la percepción visual

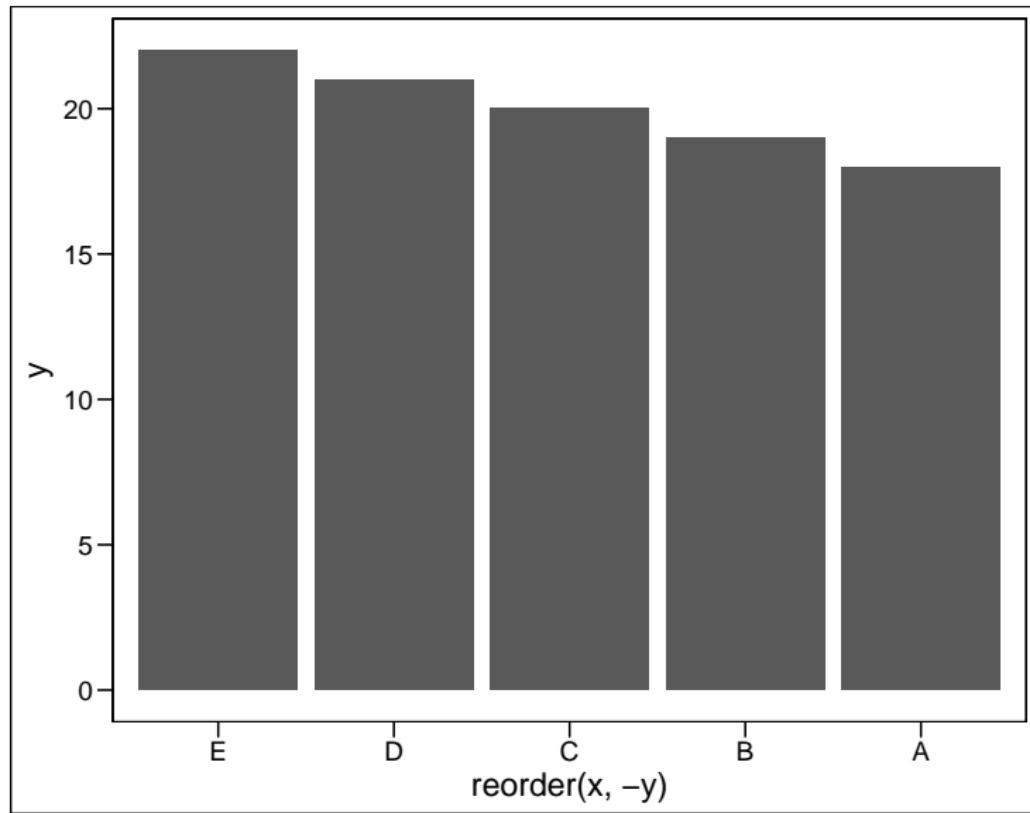


COLOR SATURATION

Precisión



Precisión



Tareas de decodificación y precisión:

1. Posición en una escala común
2. Posición en escalas no alineadas
3. Largo, dirección y ángulo
4. Área
5. Volumen y curvatura
6. Sombreado y color

EL BIPARTIDISMO SE RECUPERA, SEGÚN EL ÚLTIMO CIS -Edit. y P.10 a 12

El PP remonta y Podemos cae 16 puntos desde diciembre

Rajoy coge aire y aventaja al PSOE en 3,3 puntos a cinco meses de las generales

■ Pablo Iglesias se hunde y pasa de primera a tercera fuerza en intención directa de voto

ESTIMACIÓN DE VOTO EN % SOBRE VOTO VÁLIDO



Figure 8: La Razón (5 de agosto 2015)

Precisión: Principio 1

Un gráfico debe mostrar la variación de los datos, y no la variabilidad del diseño (Tufte, 1983)



Precisión: Principio 2

*El etiquetado de los datos debe ser completo y conciso
(Tufte, 1983)*

Fuente: INE



Figure 10: Telediario TVE (2016)

Precisión: Principio 3

Las cantidades numéricas representadas en el gráfico y las proporciones de las geometrías empleadas deben ser proporcionales (Tufte, 1983)

U.S. trade with China and Taiwan

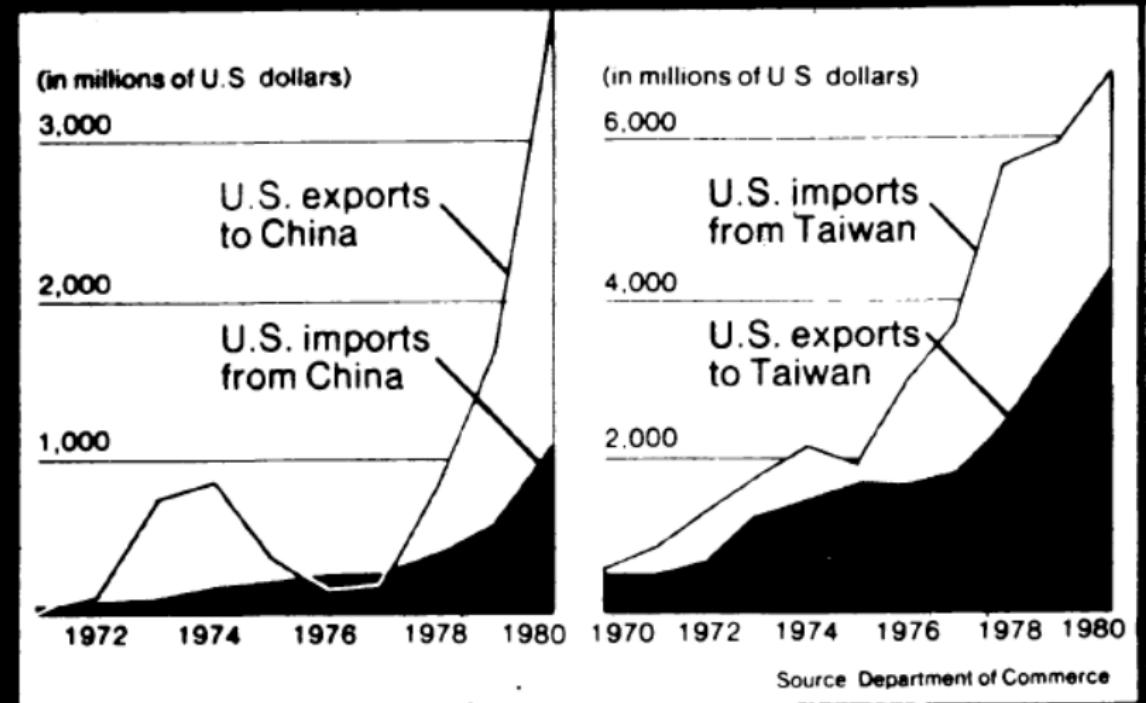


Figure 11: NYT (1981)

Precisión: Principio 4

Los ejes del gráfico deben estar alineados en la misma escala para facilitar las comparaciones (Gordon y Finch, 2015)



Figure 12: Telediario TVE (2013)

Precisión: Principio 5

Los ejes utilizados para el gráfico deben ayudar a presentar la información en contexto (Kelleher y Wagener, 2011)

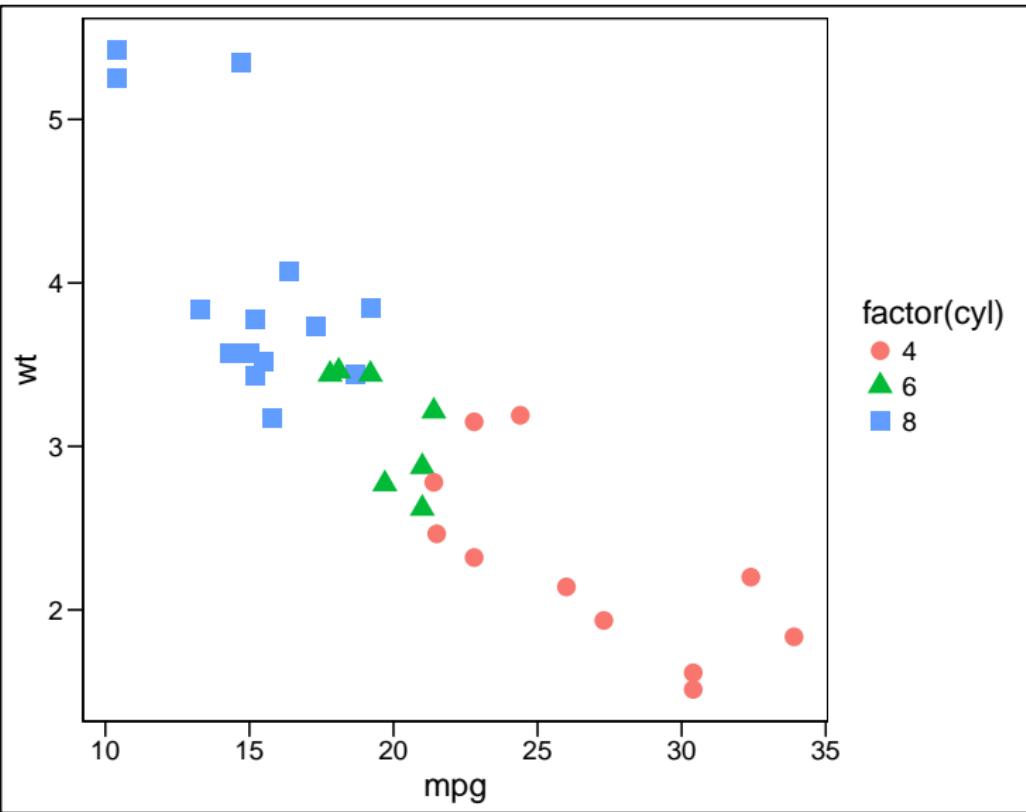
The soaraway Post — the daily paper New Yorkers trust



Figure 13: New York Post (1983)

Precisión: Principio 6

*No se debe cambiar el intervalo de progresión del eje
(Weiner, 1984)*

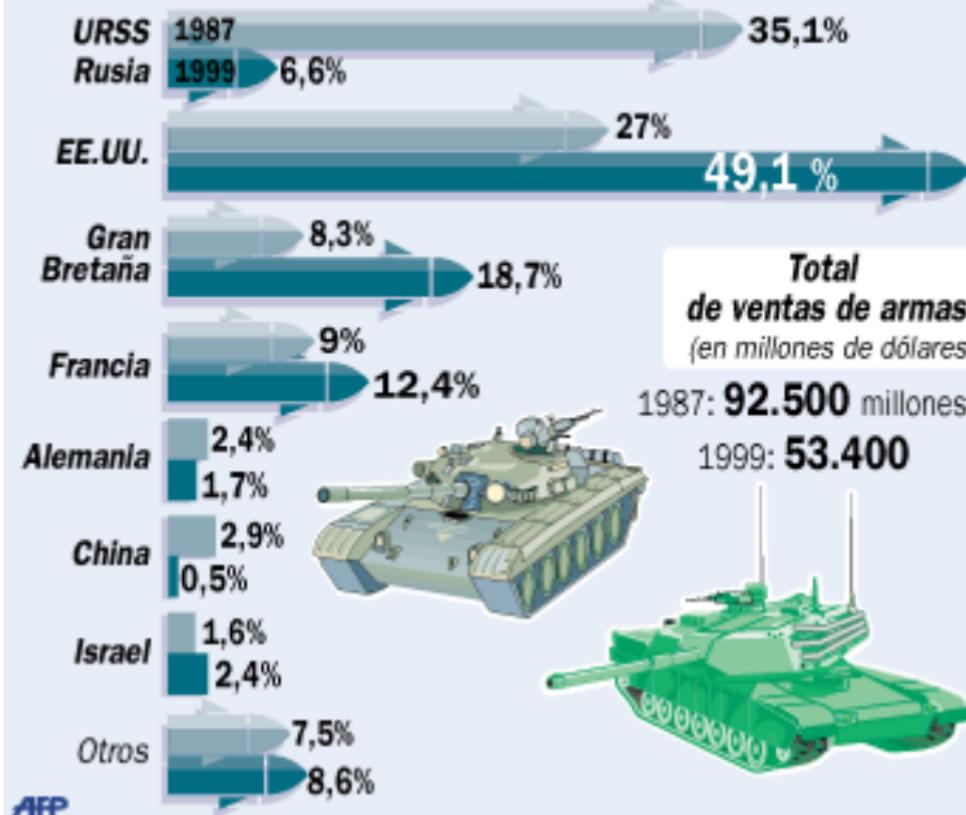


Claridad: Principio 1

El número de atributos y variables debe coincidir (Tufte, 1983)

LOS GRANDES MERCADERES DE ARMAS

Partes del mercado de las exportaciones mundiales



Claridad: Principio 2

Los gráficos deben mostrar los datos (Tufte, 1983)

Ratio datos-tinta

- ▶ Tufte propone una medida:

$$\text{Ratio datos} - \text{tinta} = \frac{\text{Tinta representando datos}}{\text{Tinta representando el resto de elementos}}$$

TECIDOS DE ALGODÃO (COTTON TEXTILES)

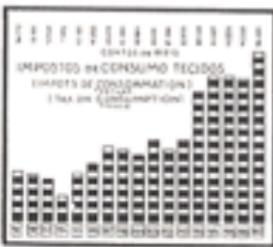
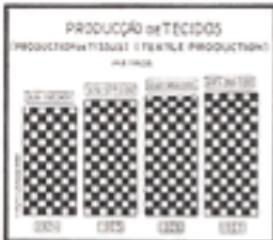
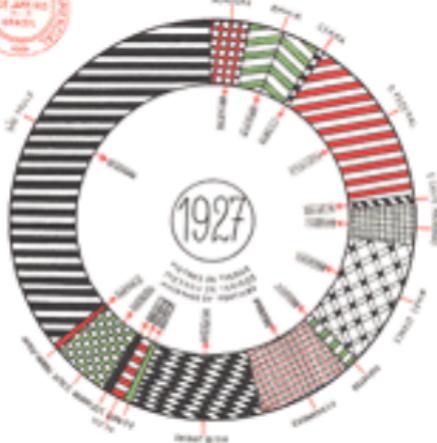
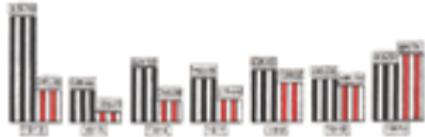
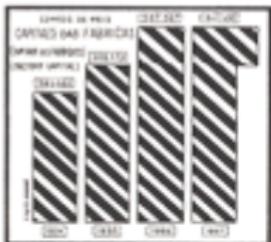


Figure 15: Instituto de Expansão Commercial (1929)

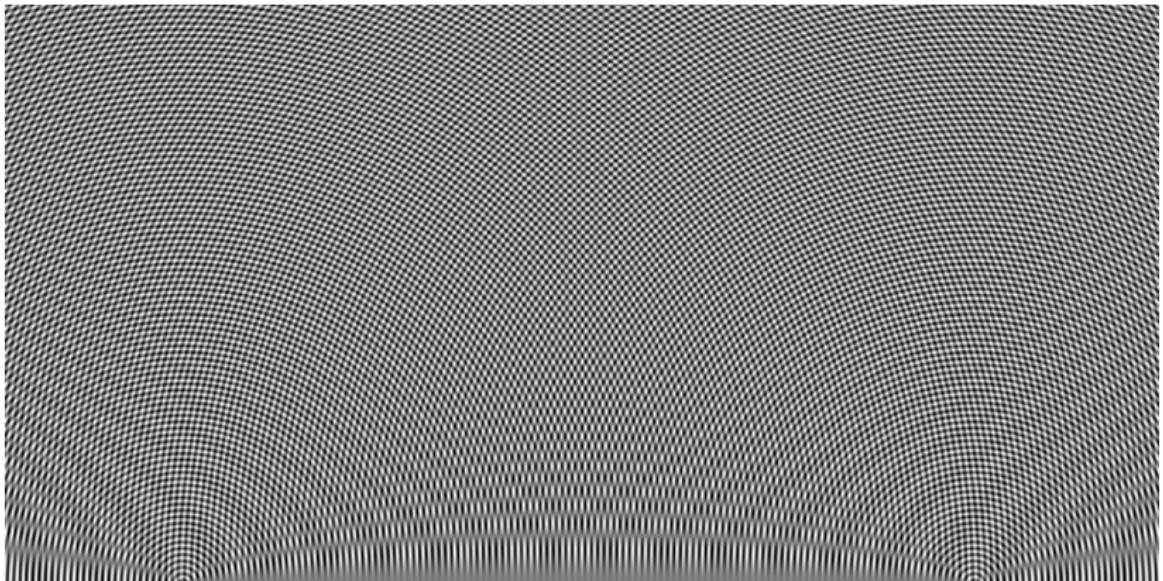


Figure 16: Efecto Moiré

Claridad: Principio 3

Evitar la presencia de chartjunk (Tufte, 1983)

¿Es el grid *chartjunk*?

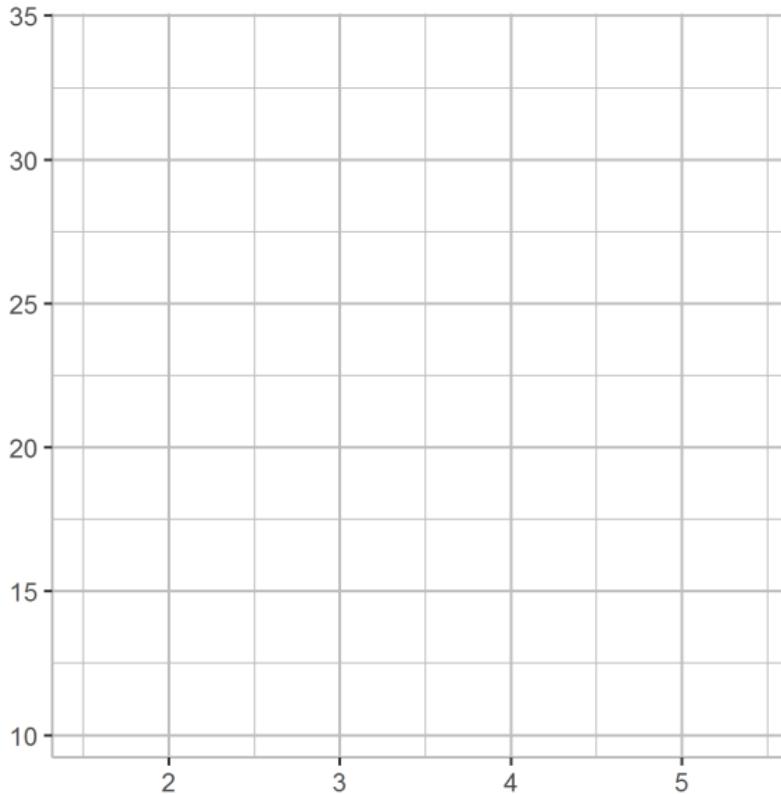


Figure 17: Grid

¿Es el grid *chartjunk*?

Algunos autores señalan que el grid puede ocultar información relevante (Tufte, 1983; Wainer, 1984; Bigwood y Spore, 2003), pero...

...según Gordon y Finch (2015) un grid de líneas gris claro puede ayudar a estimar la cantidad.

THE
NORMAL
LAW OF ERROR
STANDS OUT IN THE
EXPERIENCE OF MANKIND
AS ONE OF THE BROADEST
GENERALIZATIONS OF NATURAL
PHILOSOPHY ♦ IT SERVES AS THE
GUIDING INSTRUMENT IN RESEARCHES
IN THE PHYSICAL AND SOCIAL SCIENCES AND
IN MEDICINE, AGRICULTURE AND ENGINEERING ♦
IT IS AN INDISPENSABLE TOOL FOR THE ANALYSIS AND THE
INTERPRETATION OF THE BASIC DATA OBTAINED BY OBSERVATION AND EXPERIMENT

Figure 18: La distribución normal

Claridad: Principio 4

Usar palabras, números e imágenes en conjunto para potenciar el mensaje de los datos (Tufte, 1983)

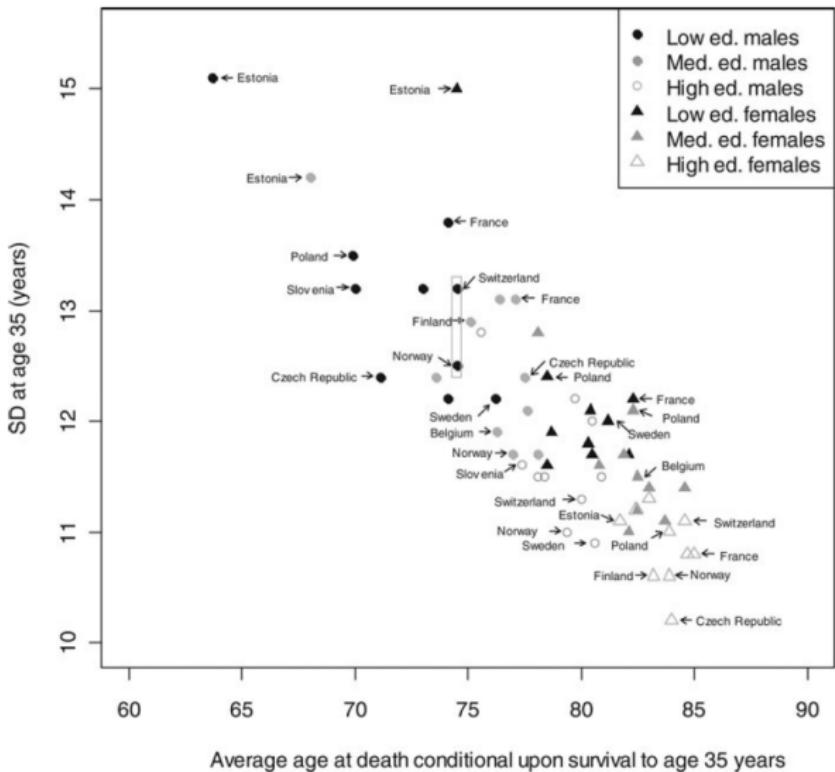


Figure 19: International Epidemiological Association (2014)

Claridad: Principio 5

El proceso de decodificación visual debe ser lo más simple posible (Gordon y Finch, 2015)

Education and Exports of Office Machines

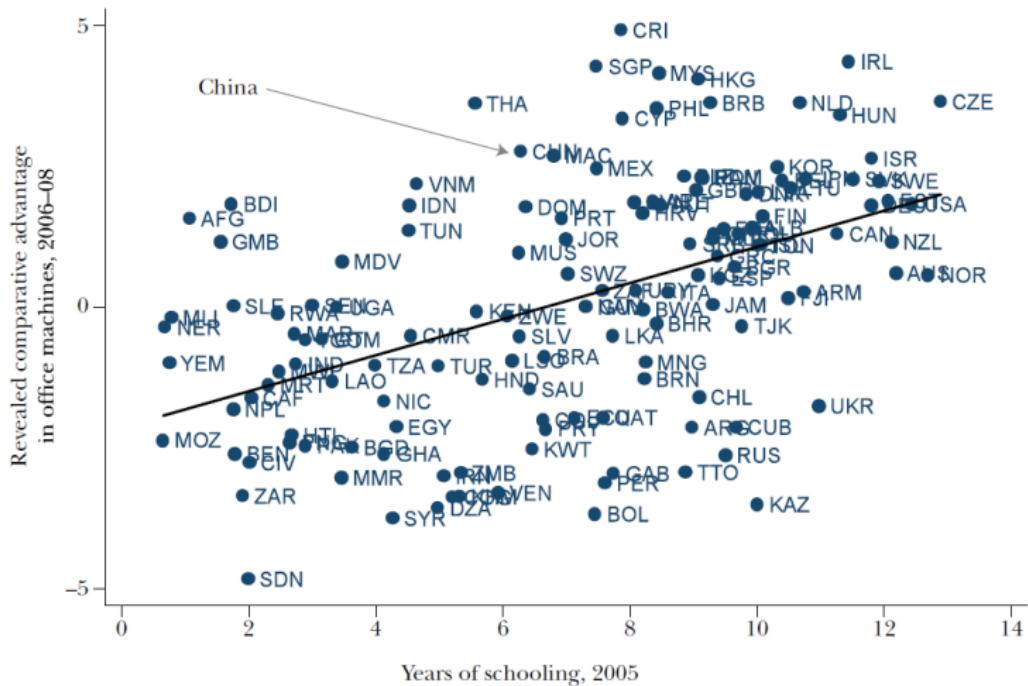


Figure 20: Hanson (2012)

Claridad: Principio 6

*Evitar la superposición de elementos opacos en el gráfico
(Kelleher y Wagener, 2011)*

Education and Exports of Office Machines

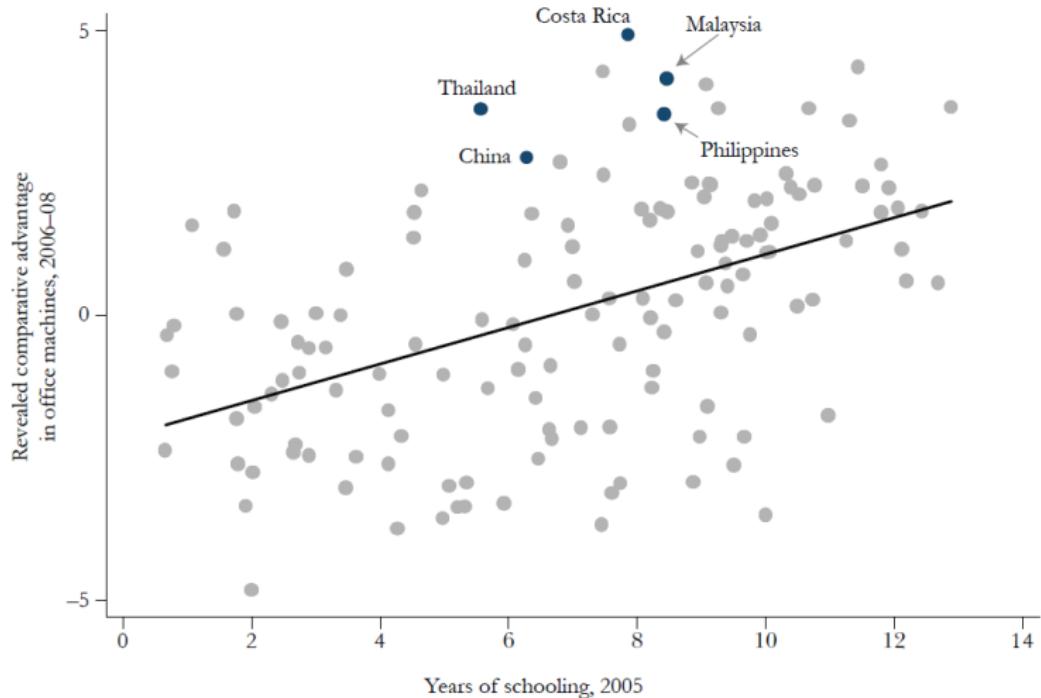


Figure 21: Schwabish (2014)

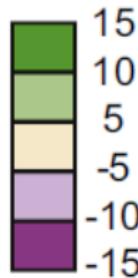
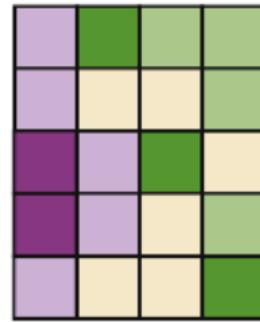
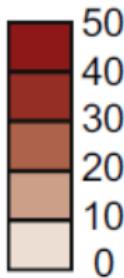
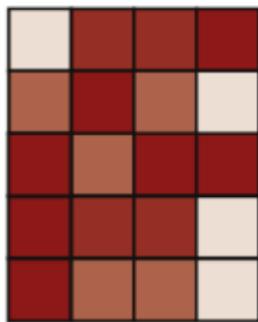


Figure 22: Kelleher y Wagener, (2011)

Claridad: Principio 7

Utilizar un esquema de colores consistente con los datos representados (Kelleher y Wagener, 2011)

Claridad: Principio 8

Evitar el uso excesivo de decimales (Wainer, 1984)

GRANDES PRESAS EN ESPAÑA POR CUENCA HIDROGRÁFICA

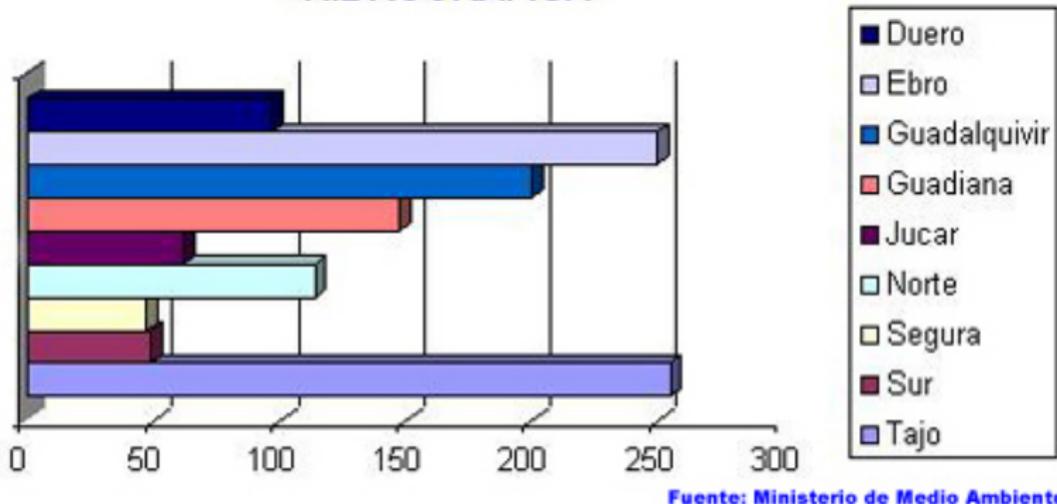


Figure 23: Ministerio de Medio Ambiente (2011)

Claridad: Principio 9

Evitar el uso de más de dos dimensiones (Wainer, 1984)

¡Práctica!

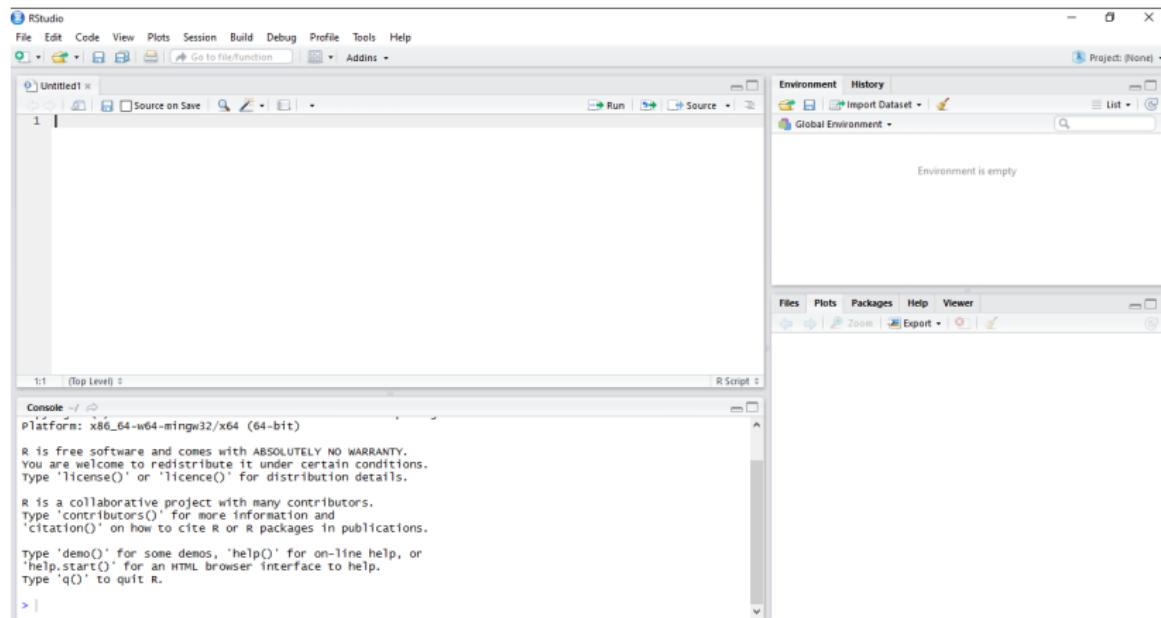
Introducción a R

¿Qué es R?

- ▶ R es un **lenguaje** que permite realizar análisis estadístico y gráfico
- ▶ R es el **software** que permite ejecutar el lenguaje
- ▶ Funciona a partir de paquetes
- ▶ ¿Por qué R?
 - ▶ Flexible
 - ▶ Abierto
 - ▶ Gratuito
 - ▶ Potencial para la visualización de datos

R-studio

- ▶ R-studio es un entorno para ejecutar R
- ▶ R-studio facilita el uso de R (e.g. auto-completar, gestor gráficos, visor de datos...)



Mi primer R-script

- ▶ El *R script* es el archivo de sintaxis o código en el que se escribe un programa
- ▶ Para abrir un nuevo script en R-studio File -> New File -> R Script
- ▶ Para ejecutar una orden desde el script **RUN** o **Ctrl + Enter**
- ▶ Para añadir comentarios a la sintaxis utilizar **#**

Objetos de R

- ▶ Vectores
- ▶ Matrices
- ▶ Data frames
- ▶ Listas
- ▶ Factores
- ▶ Funciones

Estructura de R: vectores

Los **vectores** son el objeto más sencillo, conjuntos ordenados de números

```
a <- c(10, 20, 10) #crear vector a  
b <- c(2, 2, 4) #crear vector b  
a
```

```
## [1] 10 20 10
```

```
b
```

```
## [1] 2 2 4
```

```
c <- a + b #sumar vectores a y b y asignar a c  
c
```

```
## [1] 12 22 14
```

Estructura de R: matrices

Los **matrices** son un conjunto de datos del mismo tipo en más de una dimensión

```
mat <- array(1:20,dim=c(4,5)) # genera una matriz (4 x 5)
mat
```

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

Estructura de R: data frames

Los **data frames** son matrices de datos en las que las columnas pueden ser de diferente tipo, el formato de almacenaje de datos más utilizado en Ciencias Sociales

```
df <- data.frame(varl= letters[1:5], #generar data frame  
                  varnum =rnorm(5),  
                  varlogic= runif(5)>0.5)
```

```
df
```

```
##   varl      varnum varlogic  
## 1   a -0.08187148 FALSE  
## 2   b  1.38126146 FALSE  
## 3   c -1.34869202 FALSE  
## 4   d -1.25997337  TRUE  
## 5   e -0.37971486 FALSE
```

Estructura de R: listas

Los **listas** son conjuntos de diferentes tipos de elementos
(e.g. vectores, matrices, df, funciones)

```
lista <- list(c, mat, df) #generar lista
lista
```

```
## [[1]]
## [1] 12 22 14
##
## [[2]]
##      [,1] [,2] [,3] [,4] [,5]
## [1,]     1     5     9    13    17
## [2,]     2     6    10    14    18
## [3,]     3     7    11    15    19
## [4,]     4     8    12    16    20
##
## [[3]]
##   varl      varnum varlogic
## 1  -0.08187148    FALSE
```

Estructura de R: factores

Los **factores** son vectores de datos categóricos

```
df$var1
```

```
## [1] a b c d e  
## Levels: a b c d e
```

```
class(df$var1) #función: evaluar la clase del objeto
```

```
## [1] "factor"
```

```
levels(df$var1) #función: evaluar los niveles del factor
```

```
## [1] "a" "b" "c" "d" "e"
```

Estructura de R: funciones I

Las **funciones** son las herramientas que sirven para trabajar con los datos, también son objetos almacenables y modificables

```
mean(a)
```

```
## [1] 13.33333
```

```
d <- mean(a)  
d
```

```
## [1] 13.33333
```

Estructura de R: funciones II

Las **funciones** siempre tienen la misma estructura:

Función(arg1 = x1, arg2 = x2, argX = xX)

```
mean(mi_objeto)
head(midf$mivar, n=20)
```

Ayuda con las funciones:

```
help(mi_funcion)
?mi_fucnión
```

R: paquetes de funciones

Algunas funciones vienen por defecto en R (8 paquetes), pero otras se encuentran en paquetes creados por autores. Los paquetes se encuentran disponibles en un repositorio llamado **CRAN**.

Para instalar un paquete:

```
install.packages("mi_paquete")
```

Una vez instalado, para cargar un paquete:

```
library(mi_paquete)
```

R: ¿qué paquetes vamos a utilizar?

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

tidyverse

- ▶ Es un compendio de paquetes que permiten: **leer** (readr y haven), **manipular** (tibble, tidyr, dplyr, purrr) y **visualizar** datos (ggplot2)

```
library(tidyverse) #carga el paquete tidyverse
```



sjmisc

- ▶ Es un paquete que permite realizar tablas de frecuencias y manipular datos de encuesta

```
library(sjmisc)
```

Abrir archivos en R

Cuando se trabaja con R, los datos generalmente se importan de ficheros csv (separados por comas), para ello se utiliza el comando:

```
df <- read.csv("midir\misDatos.csv",
                sep = ";", header = T)
```

En ocasiones los datos están en formato *sav* (SPSS) o *dta* (Stata). Para abrir estos archivo usar el paquete **foreign**:

```
df <- foreign::read.spss("midir\misDatos.sav")
df <- foreign::read.dta("midir\misDatos.dta")
```

Siempre que abrimos un archivo lo asignamos a un objeto, para después poder trabajar con él.

Explorar datos en R

- ▶ Funciones para el *df*:
 - ▶ `str()`
 - ▶ `head()`
 - ▶ `tail()`
 - ▶ `summary()`
- ▶ Funciones para las *variables*:
 - ▶ `frq()`
 - ▶ `descr()`

str() - estructura del objeto

```
str(mtcars)
```

```
## 'data.frame': 32 obs. of 11 variables:  
## $ mpg : num 21 21 22.8 21.4 18.7 18.1 14.3 24.4 22.8 ...  
## $ cyl : num 6 6 4 6 8 6 8 4 4 6 ...  
## $ disp: num 160 160 108 258 360 ...  
## $ hp : num 110 110 93 110 175 105 245 62 95 123 ...  
## $ drat: num 3.9 3.9 3.85 3.08 3.15 2.76 3.21 3.69 3.92 ...  
## $ wt : num 2.62 2.88 2.32 3.21 3.44 ...  
## $ qsec: num 16.5 17 18.6 19.4 17 ...  
## $ vs : num 0 0 1 1 0 1 0 1 1 1 ...  
## $ am : num 1 1 1 0 0 0 0 0 0 0 ...  
## $ gear: num 4 4 4 3 3 3 3 4 4 4 ...  
## $ carb: num 4 4 1 1 2 1 4 2 2 4 ...
```

head() - presenta primeras filas del df

```
head(mtcars, n=4)
```

```
##          mpg cyl disp  hp drat    wt  qsec vs am
## Mazda RX4     21.0   6 160 110 3.90 2.620 16.46  0  1
## Mazda RX4 Wag 21.0   6 160 110 3.90 2.875 17.02  0  1
## Datsun 710    22.8   4 108  93 3.85 2.320 18.61  1  1
## Hornet 4 Drive 21.4   6 258 110 3.08 3.215 19.44  1  0
```

tail() - presenta últimas filas del df

```
tail(mtcars, n=4)
```

```
##          mpg cyl disp  hp drat    wt  qsec vs am gear
## Ford Pantera L 15.8   8 351 264 4.22 3.17 14.5  0  1
## Ferrari Dino  19.7   6 145 175 3.62 2.77 15.5  0  1
## Maserati Bora 15.0   8 301 335 3.54 3.57 14.6  0  1
## Volvo 142E     21.4   4 121 109 4.11 2.78 18.6  1  1
```

summary() - presenta resumen de todas las variables

```
summary(mtcars)
```

```
##          mpg              cyl             disp  
##  Min.   :10.40   Min.   :4.000   Min.   : 71.1   Min.  
##  1st Qu.:15.43   1st Qu.:4.000   1st Qu.:120.8   1st Qu.  
##  Median :19.20   Median :6.000   Median :196.3   Median  
##  Mean    :20.09   Mean    :6.188   Mean    :230.7   Mean  
##  3rd Qu.:22.80   3rd Qu.:8.000   3rd Qu.:326.0   3rd Qu.  
##  Max.    :33.90   Max.    :8.000   Max.    :472.0   Max.  
  
##          drat              wt              qsec  
##  Min.   :2.760   Min.   :1.513   Min.   :14.50   Min.  
##  1st Qu.:3.080   1st Qu.:2.581   1st Qu.:16.89   1st Qu.  
##  Median :3.695   Median :3.325   Median :17.71   Median  
##  Mean    :3.597   Mean    :3.217   Mean    :17.85   Mean  
##  3rd Qu.:3.920   3rd Qu.:3.610   3rd Qu.:18.90   3rd Qu.  
##  Max.    :4.930   Max.    :5.424   Max.    :22.90   Max.  
  
##          am              gear             carb  
##  Min.   :0.0000   Min.   :3.000   Min.   :1.000
```

descr() - descriptivos

```
descr(mtcars$mpg)
```

```
##  
## ## Basic descriptive statistics  
##  
##   var     type label   n NA.prc   mean     sd     se     md trimme  
##   dd numeric   dd 32       0 20.09 6.03 1.07 19.2    19.  
##   skew  
##   0.67
```

table() - presenta tabla

```
table(mtcars$am)
```

```
##  
## 0 1  
## 19 13
```

frq() - tabla de frecuencias

```
frq(mtcars$am)
```

```
##  
## # x <numeric>  
## # total N=32  valid N=32  mean=0.41  sd=0.50  
##  
##   val frq raw.prc valid.prc cum.prc  
##     0 19    59.38      59.38    59.38  
##     1 13    40.62      40.62   100.00  
##   <NA> 0     0.00        NA      NA
```

Transformar datos en R

- ▶ Crear variable: `mutate()`
- ▶ Seleccionar variables: `select()`
- ▶ Seleccionar casos: `filter()`
- ▶ Ordenar casos: `arrange()`
- ▶ Recodificar: `recode()`

mutate() - crear variables

```
mtcars <- mutate(mtcars, acepeso = qsec/wt)
mtcars <- mutate(mtcars, acepeso2 = ifelse(am==1,
                                             qsec/wt, NA))
head(subset(mtcars, select =
            c(am, qsec, wt, acepeso, acepeso2)), n=4)

##   am  qsec      wt  acepeso acepeso2
## 1  1 16.46 2.620 6.282443 6.282443
## 2  1 17.02 2.875 5.920000 5.920000
## 3  1 18.61 2.320 8.021552 8.021552
## 4  0 19.44 3.215 6.046656          NA
```

filter() - seleccionar casos

```
table(mtcars$am)
```

```
##  
## 0 1  
## 19 13
```

```
mtcars2 <- filter(mtcars, am==0)  
table(mtcars2$am)
```

```
##  
## 0  
## 19
```

select() - seleccionar variables

```
mtcars2 <- select(mtcars, am, qsec)  
str(mtcars2)
```

```
## 'data.frame':      32 obs. of  2 variables:  
##   $ am  : num  1 1 1 0 0 0 0 0 0 0 ...  
##   $ qsec: num  16.5 17 18.6 19.4 17 ...
```

arrange() - ordenar casos

```
head(select(mtcars, wt, qsec, am, mpg), n=4)
```

```
##      wt  qsec am  mpg
## 1 2.620 16.46  1 21.0
## 2 2.875 17.02  1 21.0
## 3 2.320 18.61  1 22.8
## 4 3.215 19.44  0 21.4
```

```
mtcars <- arrange(mtcars, wt)
head(select(mtcars, wt, qsec, am, mpg), n=4)
```

```
##      wt  qsec am  mpg
## 1 1.513 16.90  1 30.4
## 2 1.615 18.52  1 30.4
## 3 1.835 19.90  1 33.9
## 4 1.935 18.90  1 27.3
```

recode() - recodificar una variable

```
table(mtcars$cyl)
```

```
##  
## 4 6 8  
## 11 7 14
```

```
mtcars$cyl2 <- recode(mtcars$cyl,  
                      `4` = "Bajo", `6` = "Medio",  
                      `8` = "Alto")  
table(mtcars$cyl, mtcars$cyl2)
```

```
##  
##      Alto Bajo Medio  
## 4     0    11    0  
## 6     0     0    7  
## 8    14     0    0
```

¡Práctica!