Clase 3

1. Graficación en R
2. graphics
3. ggplot2
4. Live coding

Graficación en R

Podemos crear graficas en R de dos distintas maneras:

1. Usando paquete **graphics (precargado en R)**
    - Conveniente para gráficas sencillas
    - Difícil de usar para gráficas más complicadas/personalizadas
2. Usando el paquete **ggplot2** de tidyverse
    - Requiere un poco más de trabajo
    - Podemos hacer gráficas más complicadas
    - Fácil (cuando ya aprendimos bien) de personalizar

graphics

Paquete graphics

Funciones más comunes y usadas

Function	Description
`plot(x)`	Plot of the values of x (on the y-axis) ordered on the x-axis
`plot(x, y)`	Bivariate plot of x (on the x-axis) and y (on the y-axis)
`pie(x)`	Circular pie chart
`boxplot(x)`	"Box-and-whiskers" plot
`stripchart(x)`	Plot of the values of x on a line (an alternative to `boxplot()` for small sample sizes)
`coplot(x~y|z)`	Bivariate plot of x and y for each value (or interval of values) of z
`dotchart(x)`	If x is a data frame, plots a Cleveland dot plot (stacked plots line-by-line and column-by-column)
`fourfoldplot(x)`	Visualizes, with quarters of circles, the association between two dichotomous variables for different populations (x must be an array with `dim=c(2, 2, k)`, or a matrix with `dim=c(2, 2)` if $k = 1$)
`assocplot(x)`	Cohen-Friendly graph showing the deviations from independence of rows and columns in a two dimensional contingency table
`mosaicplot(x)`	"Mosaic" graph of the residuals from a log-linear regression of a contingency table
`pairs(x)`	If x is a matrix or a data frame, draws all possible bivariate plots between the columns of x
`hist(x)`	Histogram of the frequencies of x
`barplot(x)`	Histogram of the values of x
`qqnorm(x)`	Quantiles of x with respect to the values expected under a normal law
`qqplot(x, y)`	Quantiles of y with respect to the quantiles of x
`contour(x, y, z)`	Contour plot (data are interpolated to draw the curves), x and y must be vectors and z must be a matrix so that `dim(z)=c(length(x), length(y))` (x and y may be omitted)
`symbols(x,y,...)`	Draws, at the coordinates given by x and y, symbols (circles, squares, rectangles, stars, thermometers or "boxplots") which sizes, colors, etc, are specified by supplementary arguments
`termplot(mod.obj)`	Plot of the (partial) effects of a regression model (`mod.obj`)

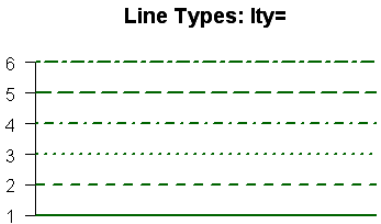
Parámetros para personalizar funciones

Agregar título y nombre a los ejes

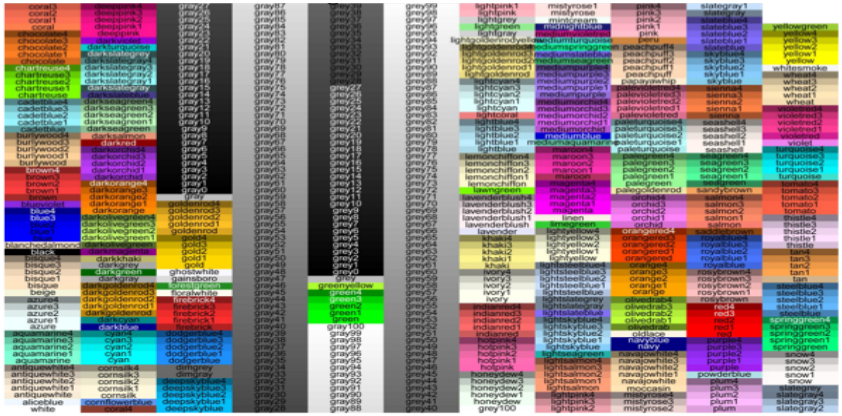
- main = “Título”
- xlab = “Nombre eje x”
- ylab = “Nombre eje y” ‘

Personalizar

- pch = tipo de punto (número del 1 al 25)
- lty = tipo de linea (número del 1 al 6)
- col = “color”
  ‘



R colors



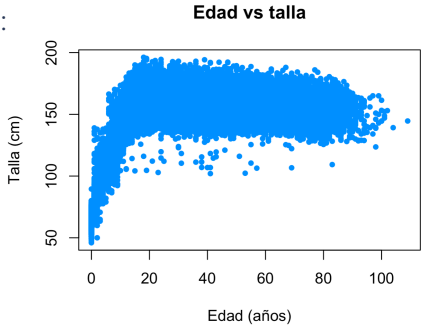
Nota: Podemos usar números, nombres o códigos hexadecimales

Ejemplo

El código:

```
> plot(edad, talla,
+      main = "Edad vs talla", xlab = "Edad (años)",
+      ylab = "Talla (cm)", col = "dodgerblue",
+      pch= 20)
```

Nos devuelve la gráfica:



ggplot2

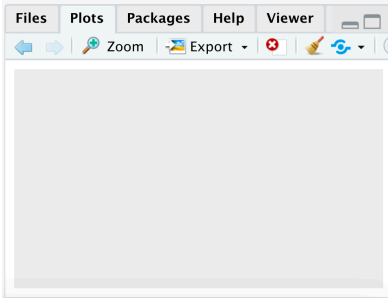
La idea principal es ir trabajando en capas:

1. Lienzo:

El codigo

```
> ggplot()
```

Nos devuelve



2. Agrego siguiente capa con un símbolo de +

El codigo

```
> ggplot() +
+   geom_point(data = Antropometria,
+              aes(x = edad, y = talla),
+              color = "tomato3")
```

Nos devuelve

Template básico de una gráfica

En general, el template para ggplot siempre se ve así:

```
ggplot(data = <DATA>) +
  <GEOM_FUNCTION>(mapping = aes(<MAPPINGS>))
```

En el ejemplo anterior, usamos la función *geom_point*, pero podríamos usar por ejemplo líneas (*geom_line*) o barras (*geom_bar*)

geom functions (funciones de mapeo)

Dentro de todas las (*geom_functions*) debemos agregar la función (*aes()*) para indicar las variables que queremos graficar, y también las *aesthetics* que son propiedades visuales de los objetos en la gráfica. Estas propiedades incluyen cosas como el tamaño, la forma o el color de los puntos/lineas/barras etc.
En el ejemplo anterior:

```
> ggplot(data = Antropometria) +
+   geom_point(aes(x = edad, y = talla),
+              color = "tomato3")
```

Live coding

Referencias y recursos

- P. Kuhnert & B. Venables, An Introduction to R: Software for Statistical Modeling & Computing
- Grolemund, G., & Wickham, H. (2017). R for Data Science. O'Reilly Media. *https://r4ds.had.co.nz/*
- *https://www.tidyverse.org/*
- *https://ggplot2.tidyverse.org/*