

Analisis gráfico con R

Gráficos con R base

Jaime Pinilla - C. González-Martel
27/10/2022

Análisis gráfico con R

1. Gráficos con la función plot de R base.
 - 1.1 Gráficos univariantes
 - 1.2 Gráficos bivariantes
 - 1.3 Gráficos múltiples
 - 1.4 Exportación
2. Preparación de datos para el análisis gráfico.

Gráficos con la función plot de R base.

`plot()` es la función básica para generar gráficos en R.

- El gráfico resultante dependerá del tipo de objeto que se le pase a la función `plot()`.

```
methods(plot)
```

```
## [1] plot.acf*          plot.data.frame*   plot.decomposed.ts*
## [4] plot.default       plot.dendrogram*   plot.density*
## [7] plot.ecdf          plot.factor*        plot.formula*
## [10] plot.function      plot.hclust*        plot.histogram*
## [13] plot.HoltWinters*   plot.isoreg*        plot.lm*
## [16] plot.medpolish*    plot.mlm*           plot.ppr*
## [19] plot.prcomp*       plot.princomp*      plot.profile.nls*
## [22] plot.R6*           plot.raster*        plot.spec*
## [25] plot.stepfun       plot.stl*           plot.table*
## [28] plot.ts            plot.tskernel*      plot.TukeyHSD*
## see '?methods' for accessing help and source code
```

- Existen otras funciones para generar ciertos tipos de gráficos como `boxplot()`, `hist()`, `barplot()`
- El gráfico resultante se genera en una ventana gráfica (`graphic.device()`), que puede ser la pantalla, la pestaña *Plots* de Rstudio o un archivo gráfico, si así se lo indicamos.

Gráficos univariantes.

Gráficos univariantes.

- `scatterplot` [Diagrama de dispersión]
- `Boxplot`. [Diagramas de caja]
- `Barplot`. [Gráfico de barras]
- Histogramas

```
numberofwords <- readr::read_csv("../Modulo 2/numberofwords.csv")
str(numberofwords)
```

```
## spec_tbl_df [268 x 8] (S3: spec_tbl_df/tbl_df/tbl/data.frame)
## $ id       : num [1:268] 1 2 3 4 5 6 7 8 9 10 ...
## $ age      : num [1:268] 20 20 20 20 20 20 20 20 18 18 ...
## $ gender   : chr [1:268] "male" "male" "female" "female" ...
## $ region   : chr [1:268] "urban" "rural" "urban" "urban" ...
## $ words    : num [1:268] 34 19 40 540 34 36 58 23 35 133 ...
## $ nouns    : num [1:268] 18 2 11 292 14 16 26 3 11 91 ...
## $ verbs    : num [1:268] 0 0 2 81 4 1 3 0 3 4 ...
## $ ageat1w  : num [1:268] 18 18 18 12 18 18 17 18 16 14 ...
## - attr(*, "spec")=
## .. cols(
## ..   id = col_double(),
## ..   age = col_double(),
## ..   gender = col_character(),
## ..   region = col_character(),
## ..   words = col_double(),
## ..   nouns = col_double(),
## ..   verbs = col_double(),
## ..   ageat1w = col_double()
## .. )
## - attr(*, "problems")=<externalptr>
```

Scatterplot. Diagrama de dispersión

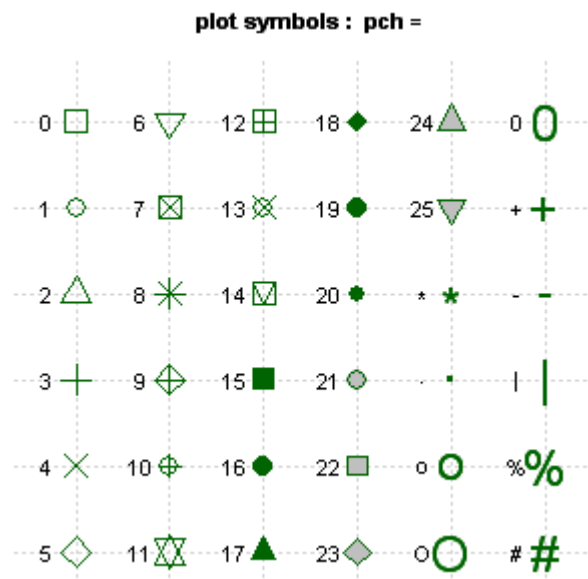
```
plot(numberofwords$words)
```


Personalización

- **type**: Formato del scatterplot.
 - "p" puntos,
 - "l" líneas,
 - "b" ambos,
 - "h" 'histograma', líneas verticales de alta densidad,
 - "s" escalones,
 - "S" otro tipo de escalones,
 - "n" sin gráfico.

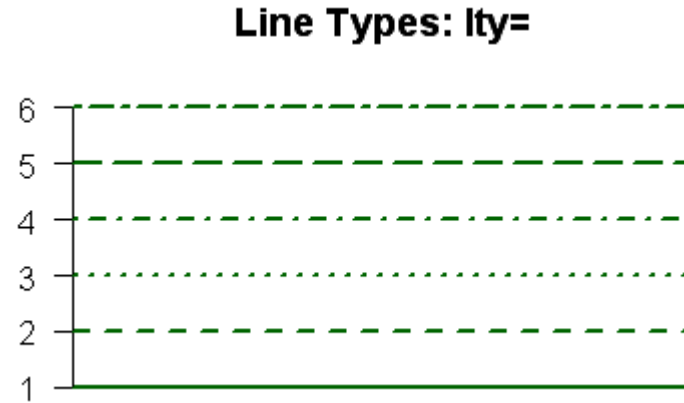
Personalización (puntos)

- `cex`, `cex.main`, `cex.sub`, `cex.xlab`, `cex.ylab`: Tamaño del punto intersección.
- `pch`: Forma del punto de intersección.



Personalización (líneas)

- **lty**: Forma de la línea.



- **lwd**: Grosor de la línea o del borde del punto con respecto al grosor por defecto ($lwd=1$). P.ej. $lwd=2$ es el doble con respecto al grosor de la línea por defecto.

Personalización (colores)

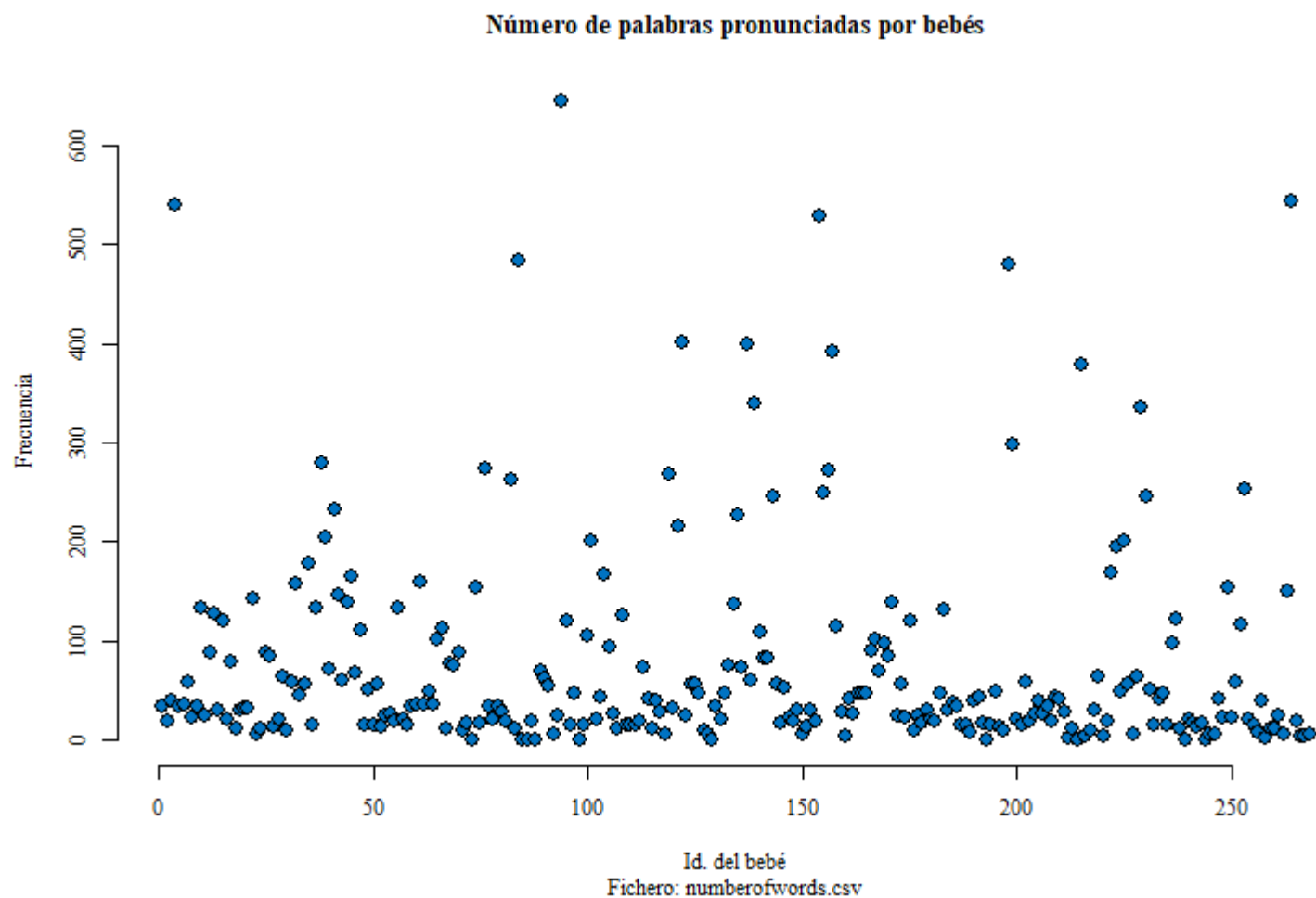
- **main, sub, xlab, ylab**: Título de gráfico, subtítulo y nombre de los ejes x e y, respectivamente.
- **col**: Color del borde del punto o color de la línea. Si es de puntos (`type="p"`) y se ha especificado valores de `pch` entre 21 y 25 se puede introducir el color de relleno con `bg=`. los colores se pueden establecer a través de **índices(números)**, **nombres** entre comillas ("red", "blue", etc.) o por su **código hexadecimal**. Por ejemplo, `col=1`, `col="white"` y `col="#FFFFFF"` son equivalentes a la hora de expresar el blanco.
- **col.main, col.sub, col.xlab, col.ylab, col.axis** : del título de gráfico, subtítulo y nombre de los ejes x e y, respectivamente.

Personalización (fuentes)

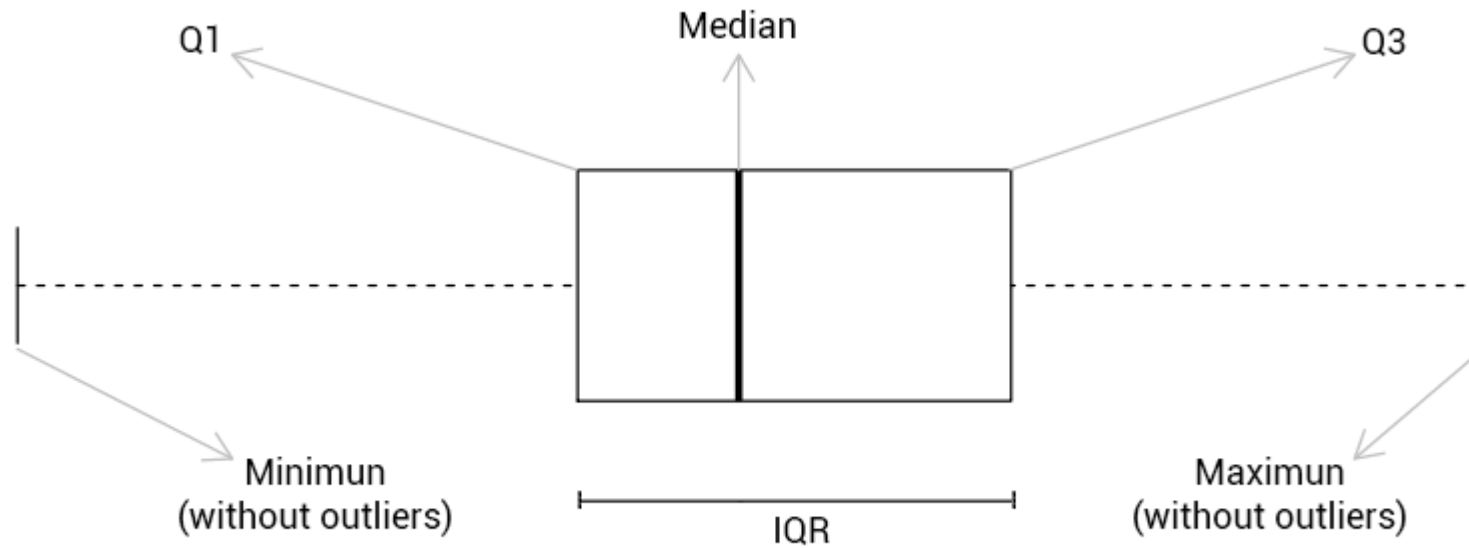
- `font`, `font.main`, `font.sub`, `font.xlab`, `font.ylab`: Fuente para los textos. 1=texto plano, 2=**negrita**, 3=*itálica*, 4= **negrita itálica**, 5=símbolos.
- `ps`: Tamaño del punto de la fuente. Tamaño de la fuente = $ps * cex$.
- `col.main`, `col.sub`, `col.xlab`, `col.ylab`, `col.axis` : del título de gráfico, subtítulo y nombre de los ejes x e y, respectivamente.
- **!!** `family`: Familia de la fuente. Serif, Sans, Mono, Symbol

```
windowsFonts(A = windowsFont("Times New Roman"))
```

```
plot(numberofwords$words, ylab = "Frecuencia", xlab = "Id. del bebé", main = "Número de palabras pronunciadas por
```



Boxplot



`boxplot()` considera *outliers* aquellos valores superiores a $Q_3 + 1.5 \cdot IQR$ e inferiores a $Q_1 - 1.5 \cdot IQR$.

```
summary(numberofwords$words)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.   
##      0.00   17.00   35.00   76.28   88.25  644.00
```

```
boxplot(numberofwords$words)
```


Personalización

- **col**: Color de la caja del gráfico.
- **outline**: Si es FALSE los valores *outliers* no se dibujan.
- **horizontal**.

Barplot

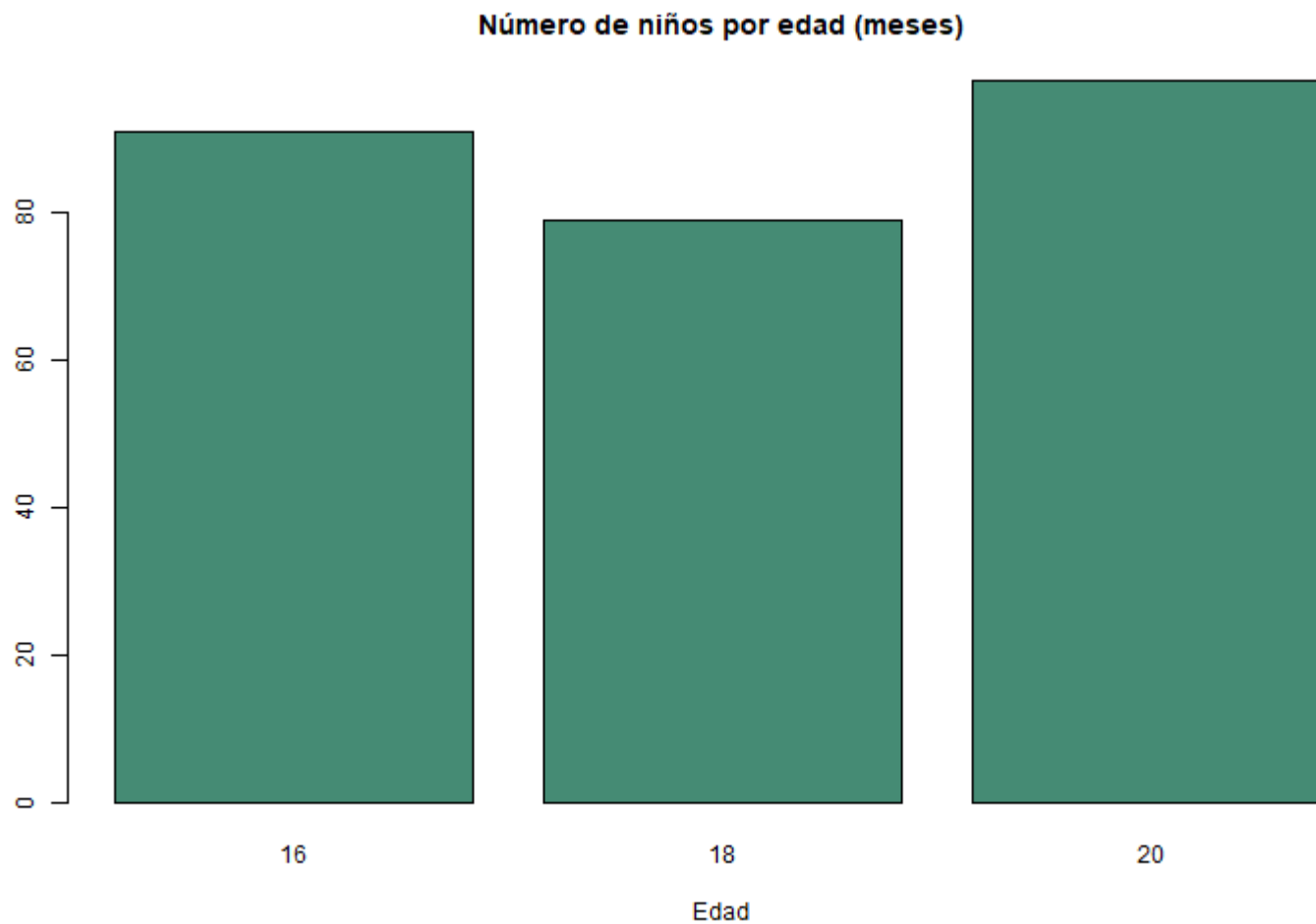
```
table(numberofwords$age)
```

```
##  
## 16 18 20  
## 91 79 98
```

```
barplot(table(numberofwords$age), col = "aquamarine3",  
        xlab = "Edad", main = "Número de niños por edad (meses)")
```



```
plot(factor(numberofwords$age), col = "aquamarine4",  
      xlab = "Edad", main = "Número de niños por edad (meses)")
```

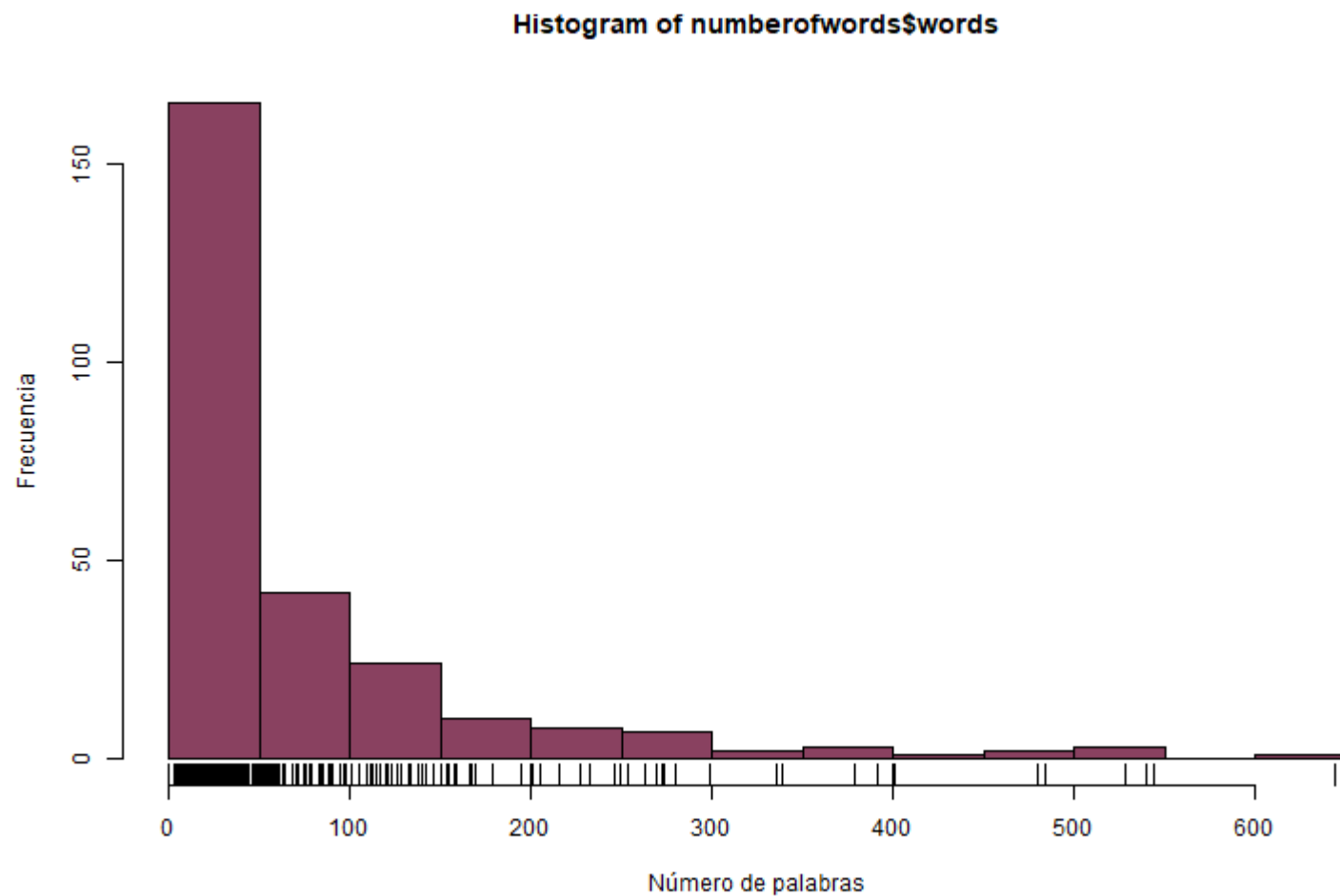


Histograma

```
hist(numberofwords$words, col = "#894160", xlab = "Número de palabras", ylab = "Frecuencia")
```

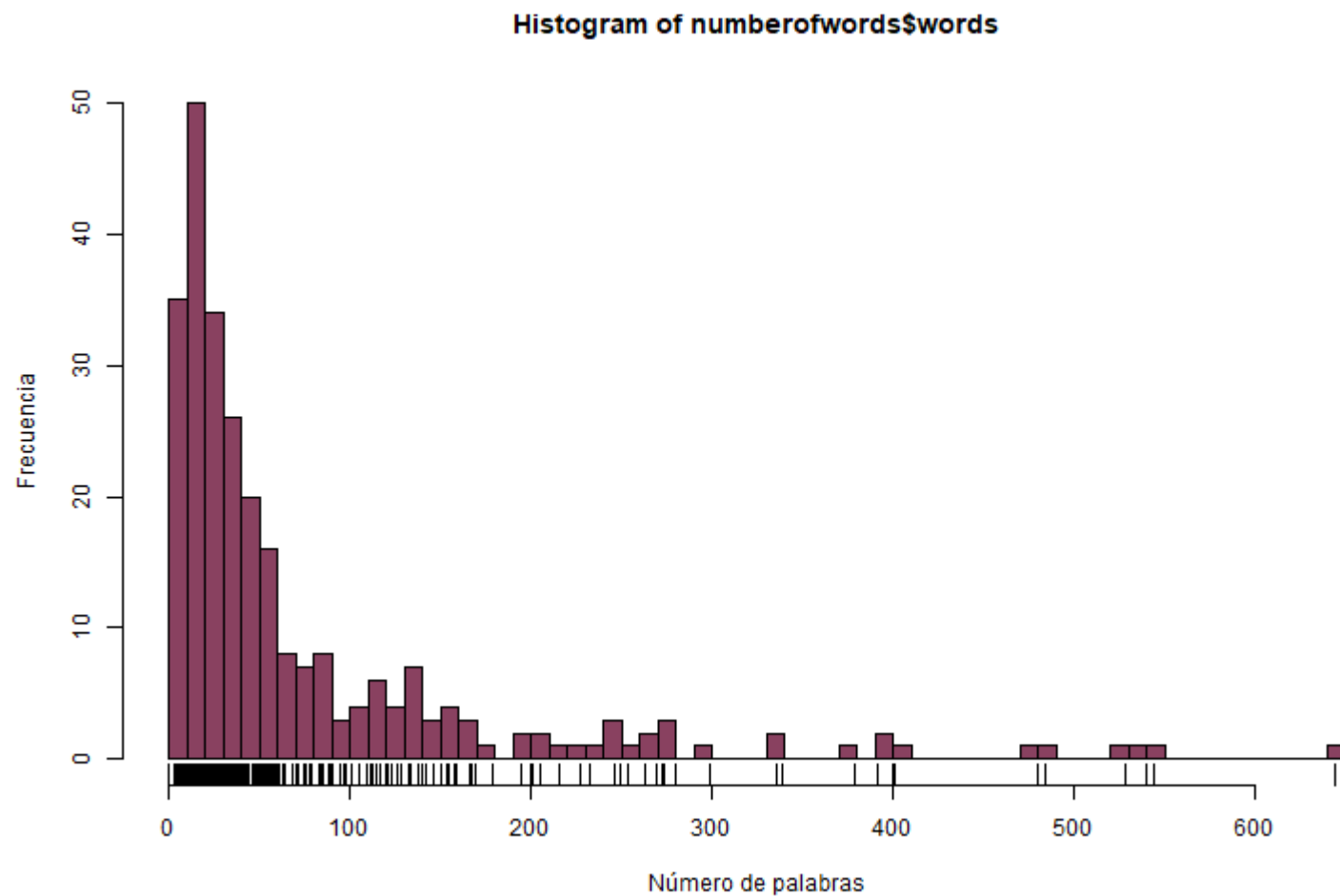
La función rug

```
hist(numberofwords$words, col = "#894160", xlab = "Número de palabras", ylab = "Frecuencia")  
rug(numberofwords$words)
```



Personalizar con el parámetro breaks

```
hist(numberofwords$words, col = "#894160", xlab = "Número de palabras", ylab = "Frecuencia", breaks = 50)  
rug(numberofwords$words)
```

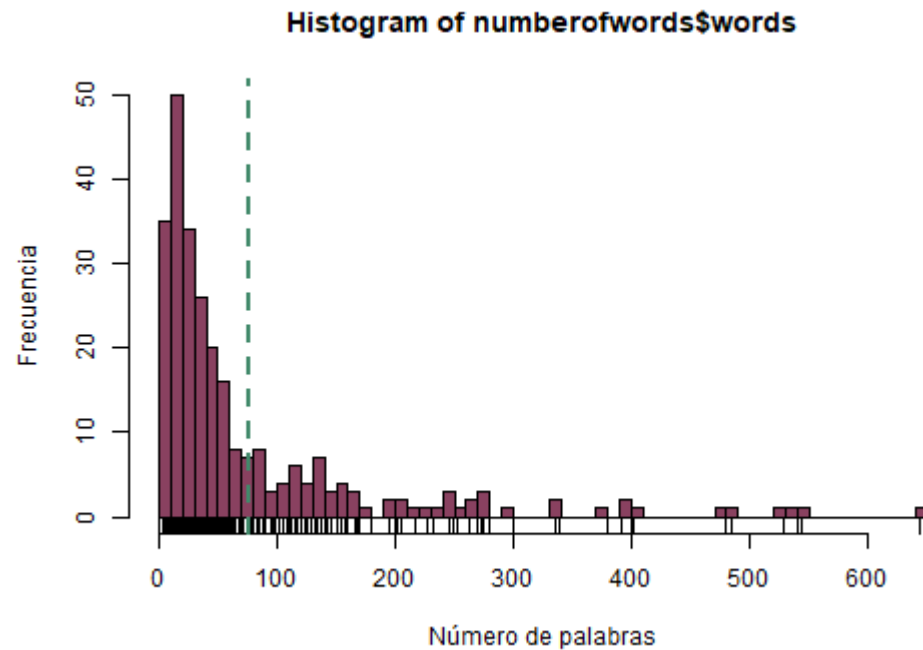


Capas.

Líneas de referencia

```
abline(h = NULL, v = NULL, coef = NULL)
```

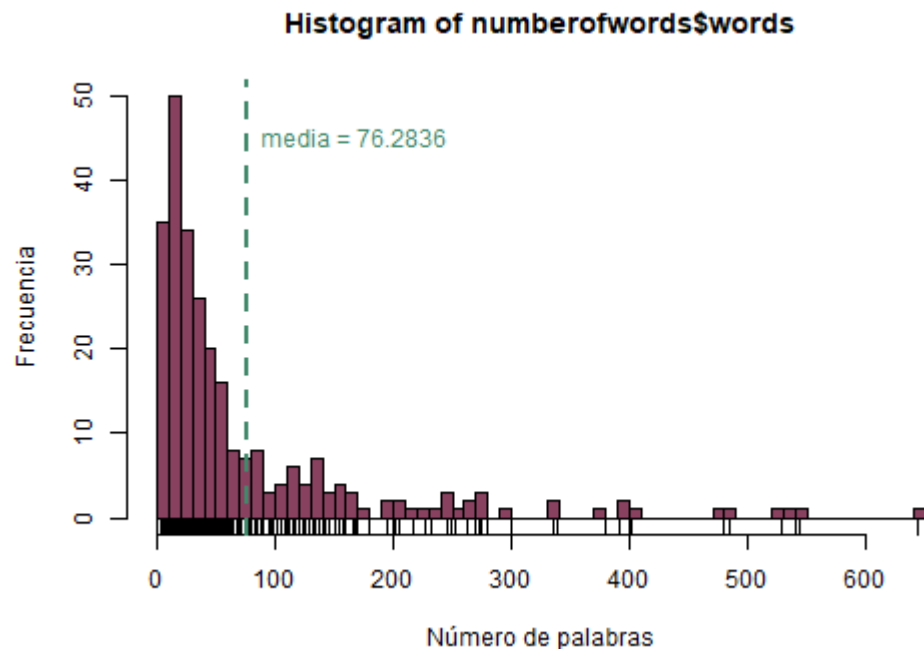
```
hist(numberofwords$words, col = "#894160", xlab = "Número de palabras", ylab = "Frecuencia", breaks = 50)  
rug(numberofwords$words)  
abline(v=mean(numberofwords$words, rm.na = T), lwd = 2, col="#41896a", lty = 2)
```



Anotaciones (texto)

text()

```
hist(numberofwords$words, col = "#894160", xlab = "Número de palabras", ylab = "Frecuencia", breaks = 50)
rug(numberofwords$words)
abline(v = mean(numberofwords$words, rm.na = T), lwd = 2, col="#41896a", lty = 2)
text(label = paste("media =", round(mean(numberofwords$words, rm.na = T), 4)), x = mean(numberofwords$words, rm.na = T), y = 45)
```



Gráficos bivariantes.

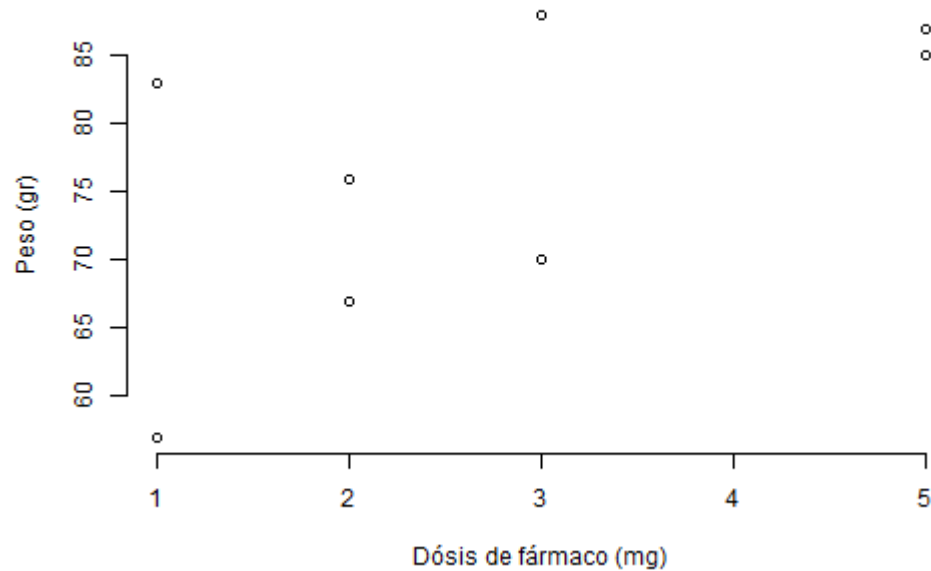
Gráficos bivariantes.

- `scatterplot` [Diagrama de dispersión]
- `Boxplot`. [Diagramas de caja]
- `Barplot`. [Gráfico de barras]

Scatterplot o Gráficos de dispersión

Muestra la relación de dos variables numéricas

```
ratones <- readr::read_csv("../Modulo 2/calcium.csv")  
plot(x = ratones$dose, y = ratones$weight, xlab = "Dosis de fármaco (mg)", ylab = "Peso (gr)", frame = FALSE)
```

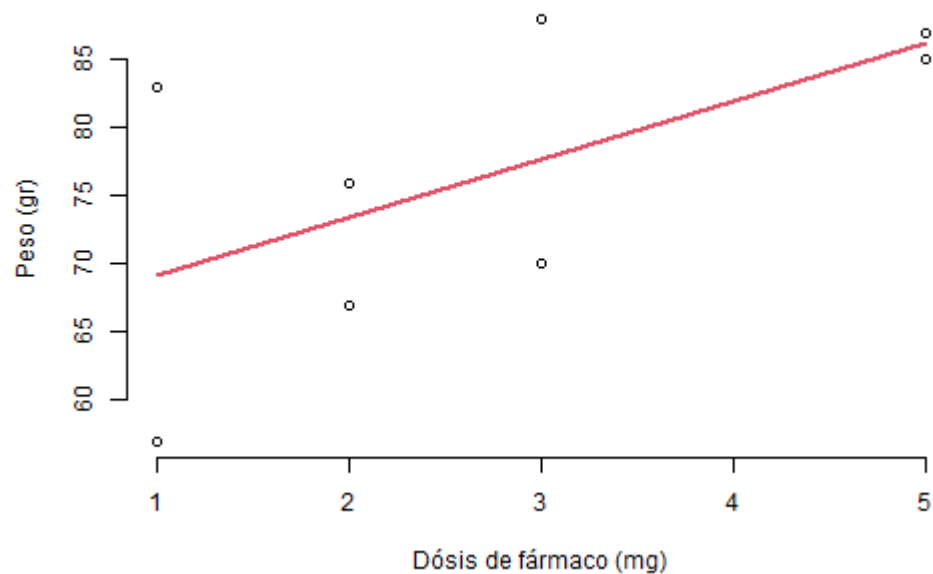


```
# plot(weight ~ dose, data = ratones, xlab = "Dosis de fármaco (mg)", ylab = "Peso (gr)", frame = FALSE)
```

Añadir recta de regresión

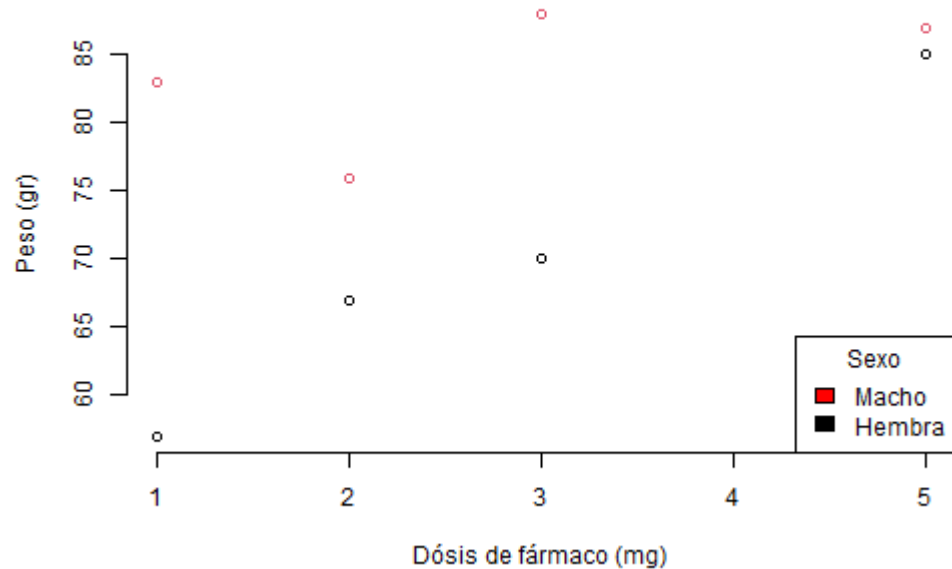
Muestra la relación de dos variables numéricas

```
mod_lm <- lm(weight ~ dose, data = ratones)
plot(weight ~ dose, data = ratones, xlab = "Dosis de fármaco (mg)", ylab = "Peso (gr)", frame = FALSE)
lines(ratones$dose, predict(mod_lm), col = 2, lwd = 2)
```



Añadir una tercera dimensión

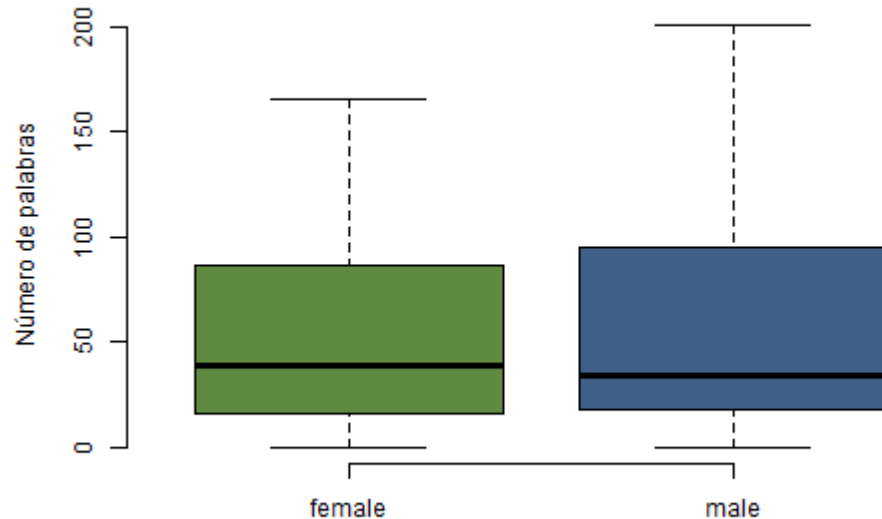
```
plot(weight ~ dose, data = ratones, xlab = "Dosis de fármaco (mg)", ylab = "Peso (gr)", frame = FALSE, col = factor(dose),  
legend(x = "bottomright", legend = c("Macho", "Hembra"), fill = c("red", "black"), title = "Sexo")
```



Boxplot o Gráficos de caja

Muestra la relación entre una variables categórica y otra numérica

```
plot(words ~ factor(gender), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = c("green", "blue"))
```

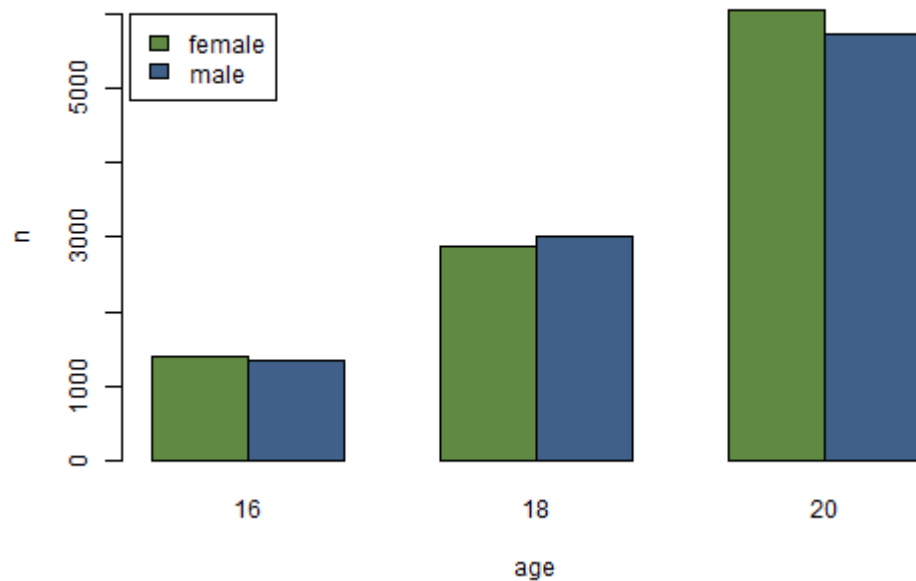


```
# boxplot(words ~ factor(gender), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = c("green", "blue"))
```

Barplot o Gráficos de barra

Muestra la frecuencia o valor de varias variables categóricas

```
library(dplyr)
freq <- numberofwords %>% group_by(age, gender) %>% summarise(n = sum(words))
barplot(n ~ gender + age, data = freq, beside = T, col = c("#608941", "#416089"), legend = levels(factor(freq$gender)))
```

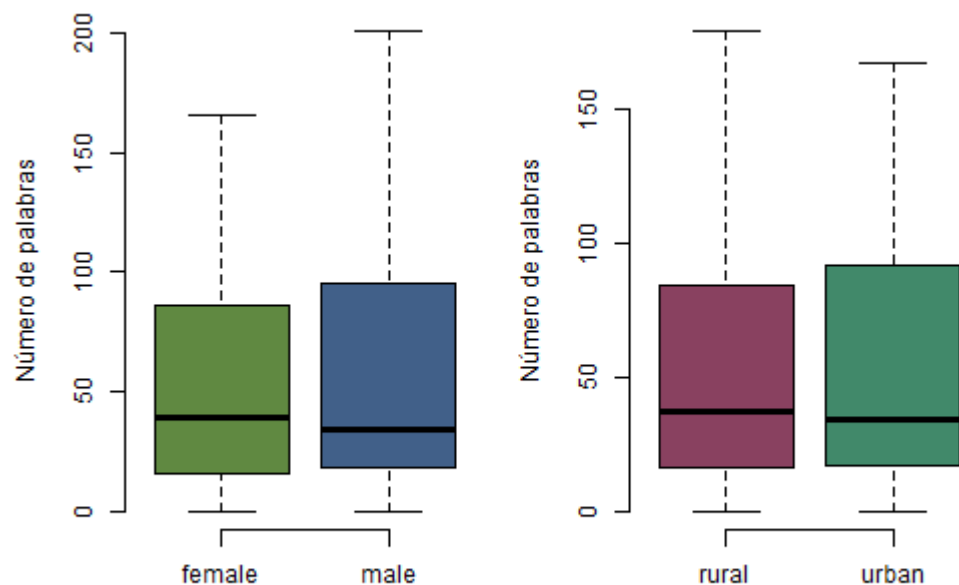


Gráficos múltiples.

Gráficos múltiples.

```
par(mfrow, mfc, mar)
```

```
par(mfrow = c(1, 2), mar = c(5, 4, 2, 1))  
boxplot(words ~ factor(gender), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col =  
boxplot(words ~ factor(region), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col =
```



Exportación.

Tipos de fichero

- Vectoriales
 - **pdf**. Redimensiona bien, fácilmente portable pero no es eficiente si contiene muchos objetos.
 - **svg**. Basado en lenguaje XML. Redimendiona bien y soporta animaciones y puede contener elementos interactivos. Para gráficos web.
 - **win.metafile**. Para palicaciones windows.
 - **postscript**. Mismas características que el pdf con el inconveniente añadido que windows no suele tener un visor para este tipo de fichero.
- Bitmap
 - **png**. Redimensiona mal, no emplea compresión. Para imágenes con colores sólidos.
 - **jpg**. Redimensiona mal, emplea compresión. Para fotografías y escenas naturales.
 - **tiff**. Alta calidad, sin compresión.
 - **bmp**. bitmap nativo de windows.

```
pdf(file = "fig/now_boxplots.pdf", height = 7, width = 8.5)
par(mfrow = c(1, 2), mar = c(5, 4, 2, 1))
boxplot(words ~ factor(gender), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = "red", las = 1)
boxplot(words ~ factor(region), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = "red", las = 1)
dev.off()
```

```
png(file = "fig/now_boxplots.png", height = 480, width = 600)
par(mfrow = c(1, 2), mar = c(5, 4, 2, 1))
boxplot(words ~ factor(gender), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = "red", las = 1)
boxplot(words ~ factor(region), data = numberofwords, xlab = "", ylab = "Número de palabras", frame = FALSE, col = "red", las = 1)
dev.off()
```

**I'm done
with everything
today.**