

# SESION 3. ANÁLISIS EXPLORATORIO DE DATOS (AED/EDA) CON R

Victor Miguel Terrón Macias

17/1/2021

## OBJETIVO

Lograr un mejor conocimiento o entendimiento del problema con el cual se relacionan los datos, de una manera relativamente fácil y rápida, sin utilizar modelos o teoría matemática avanzada

Durante el transcurso de esta sesión serás capaz de desarrollar las siguientes capacidades de R Conocer mejor el conjunto de datos con el que se esté trabajando: \* Llevar a cabo visualizaciones \* Plantear hipótesis \* Formular preguntas relevantes.

## INTRODUCCIÓN

Un diagrama de dispersión, gráfica de dispersión o gráfico de burbujas es un tipo de diagrama matemático que utiliza las coordenadas cartesianas para mostrar los valores de dos variables para un conjunto de datos1.

Los diagramas de dispersión son útiles para reconocer tendencias en datos, cuando estos son graficados en puntos, se deben tomar dos variables de tipo cuantitativo, a continuación, se presenta una imagen donde se pueden observar dos tipos de correlación que generalmente son de interés. Cuando se trata de una alta dispersión se puede suponer que la correlación es cercana a cero, y cuando tenemos poca dispersión se puede deber a correlaciones cercanas a 1 o a -1, sin embargo puede existir poca dispersión si los datos se aglomeran en clusters de información, es decir los datos están muy “compactados”, la correlación puede obtenerse mediante el comando `cor()`.

Cuando los diagramas de co-relación si parece haber una tendencia negativa se dice que la correlación es negativa y sucede cuando la variable crece en el eje  $x$  y la del eje  $y$  decrece. Cuando en un diagrama de co-relación si parece haber una tendencia positiva o alcista se da cuando la variable del eje  $x$  crece y la variable del eje  $y$  también.

La ilustración 1 es una correlación negativa, esto sucede cuando una variable crece (en este caso eje de las “x”) y la otra disminuye (eje de la “y”). Para el caso de la ilustración 2, se representa una correlación positiva, cuando una variable crece (eje “x”), la otra también lo hace (eje “y”). Puede suceder que no se pueda reconocer un patrón específico, esto también es útil ya que indica que las variables no tienen una correlación o que esta no es tan fácil de determinar visualmente debido a que su coeficiente de correlación es muy bajo.

## COMANDO PARA HACER GRÁFICO DE DISPERSIÓN

El comando en R para realizar un gráfico de dispersión es: `plot(var1, var2, ...)`

## BOXPLOTS Y OUTLIERS

También conocido como diagrama de caja y bigote, box plot, box-plot o boxplot. Es un método estandarizado para representar gráficamente una serie de datos numéricos a través de sus cuartiles (1ero; 25%, 2do; 50%, 3ro; 75%) . El máximo interés del box-plot es visualizar la distribución de una variable numérica de la manera más simplificada posible. Sólo utiliza los valores de los cuartiles, los extremos ( $q1 - 1.5IQR$  y  $q3 + 1.5IQR$ ) y valores raros o outliers. No depende de valores ponderados como la media. Simplemente se fija en las características de la posición. El diagrama siguiente será de mucha utilidad para comprenderlos. En R se utiliza el comando `boxplot` para graficarlos.

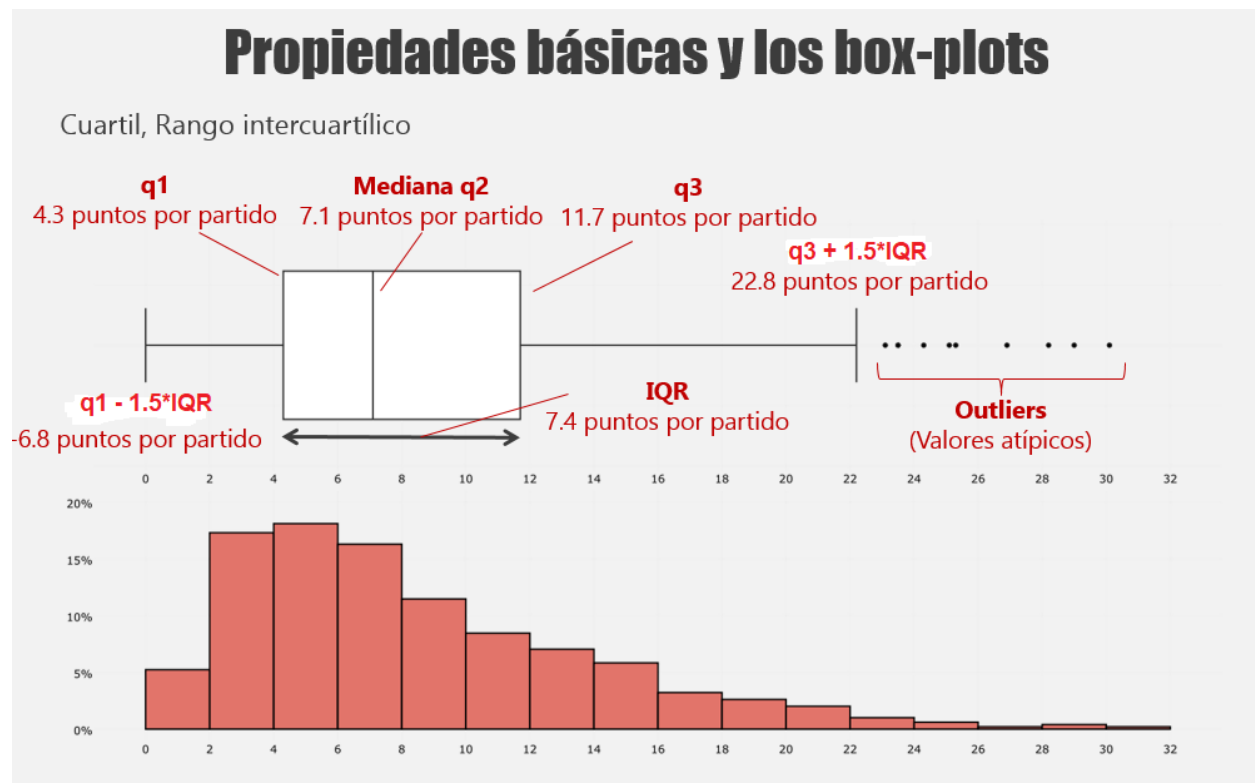
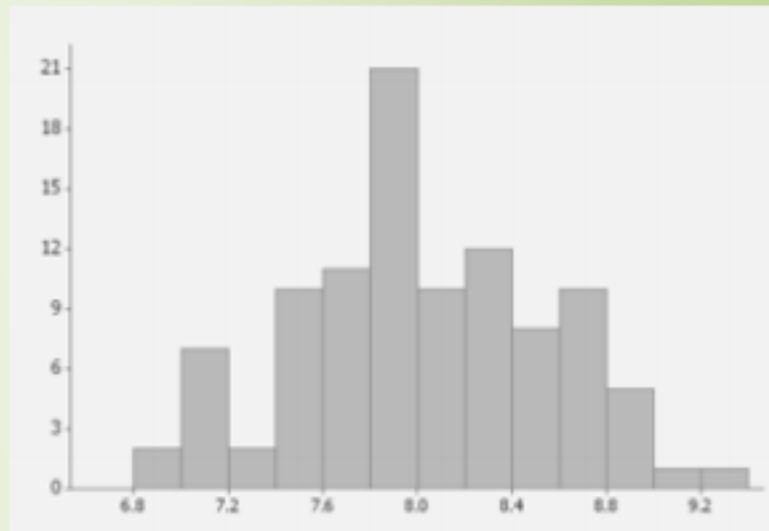


Figure 1: PROPIEDADES-BOX-PLOTS

## HISTOGRAMAS

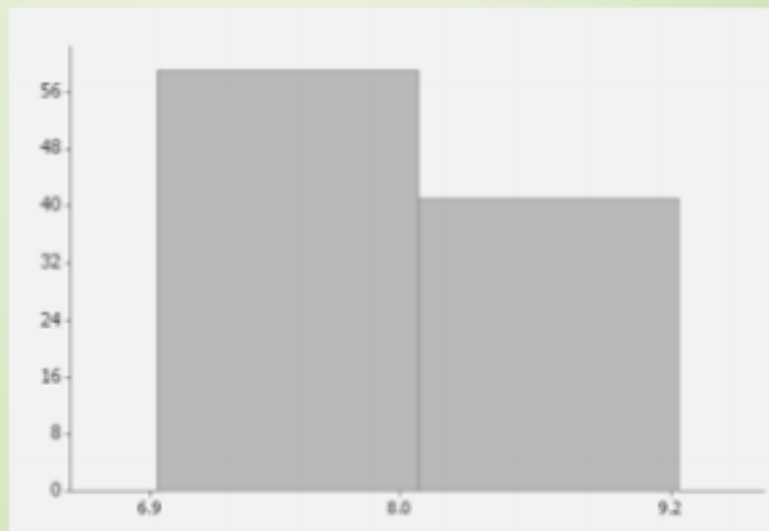
Es una gráfica de la distribución de un conjunto de datos. Es un tipo especial de gráfica de barras, en la cual una barra va pegada a la otra, es decir no hay espacio entre las barras. Cada barra representa un subconjunto de los datos. Un histograma muestra la acumulación ó tendencia, la variabilidad o dispersión y la forma de la distribución. Un histograma es una gráfica adecuada para representar variables continuas, aunque también se puede usar para variables discretas. Es decir, mediante un histograma se puede mostrar gráficamente la distribución de una variable cuantitativa o numérica. Los datos se deben agrupar en intervalos de igual tamaño, llamados clases.

Se grafican en el eje de las X las clases y en el eje Y las frecuencias de nuestros datos entonces de ese modo obtenemos el histograma, que es la representación visual de la distribución de frecuencias. Para realizar un boxplot se utiliza el comando en R: `Boxplot(var1, var2,...)`



***Poco resumen: muchas clases***

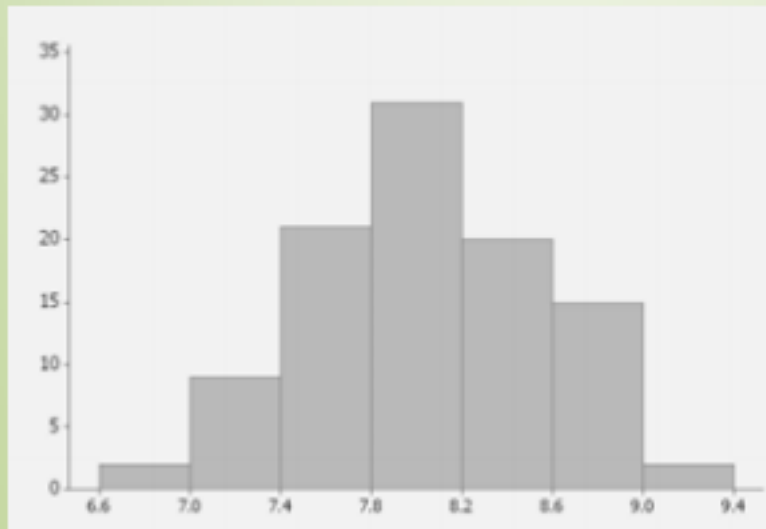
Figure 2: imagen1



***Demasiado resumen: pocas clases***

Figure 3: imagen2

***Demasiado resumen:  
pocas clases***



***Resumen Adecuado***

Figure 4: imagen3

## SERIES DE TIEMPO

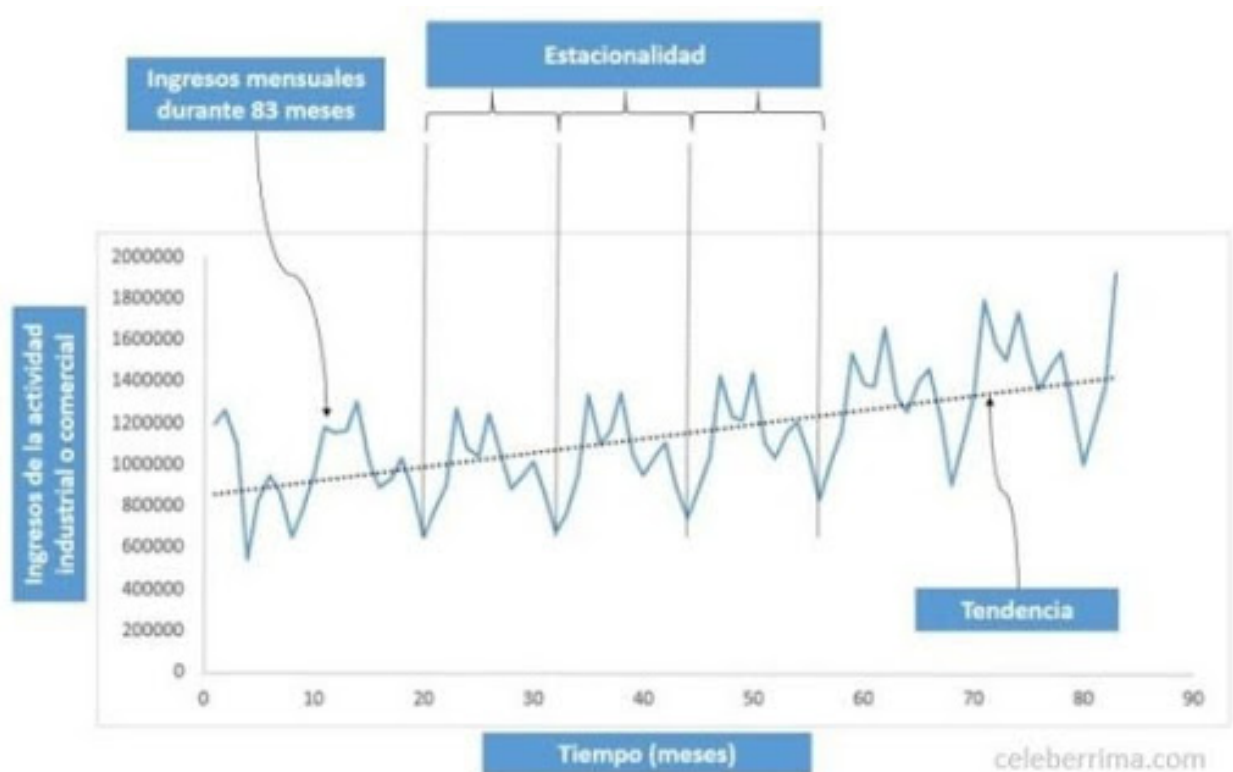
Es un conjunto de valores observados durante una serie de periodos temporales, secuencialmente ordenada. Son variables estadísticas bidimensionales en donde el tiempo es la variable independiente, y la otra es la variable dependiente. Se construyen modelos de series de tiempo para:

- Obtención del mecanismo
- Estudio de su evolución futura o predicción

Se realizan de la siguiente forma: \* Analizando los componentes o factores que determinan los resultados de la información

El método clásico para el análisis de series de tiempo identifica cuatro componentes:

- TENDENCIA (T).- El movimiento general a largo plazo de los valores de la serie de tiempo (Y) sobre un extenso periodo de años.
- FLUCTUACIONES CÍCLICAS (C).- Movimientos ascendentes y descendentes respecto de las tendencias recurrentes, con una duración de varios años.
- VARIACIONES ESTACIONALES (E).- Movimientos ascendentes y descendentes respecto de la tendencia que se consuman en el término de un año y se repiten anualmente, estas variaciones suelen identificarse con base en datos mensuales o trimestrales.
- VARIACIONES IRREGULARES (I).- Las variaciones erráticas respecto de la tendencia que no puedan atribuirse a las influencias cíclicas o estacionales. A continuación, se muestran las partes de una serie de tiempo.



Existe la descomposición aditiva y multiplicativa las cuales ayudan a entender el comportamiento de la serie de tiempo. Para realizar las series de tiempo se utilizará el comando `ts()`

# EJEMPLO 1. PAQUETE GGLOT2

## OBJETIVO

- Generar gráficos artativos
- Aprender la sintaxis del paquete ggplot2

## REQUISITOS

- Haber realizado la lectura y comprensión del prework
- Manejo de data frames
- Analizar el código que se va mostrando y tratar de comprender la sintaxis

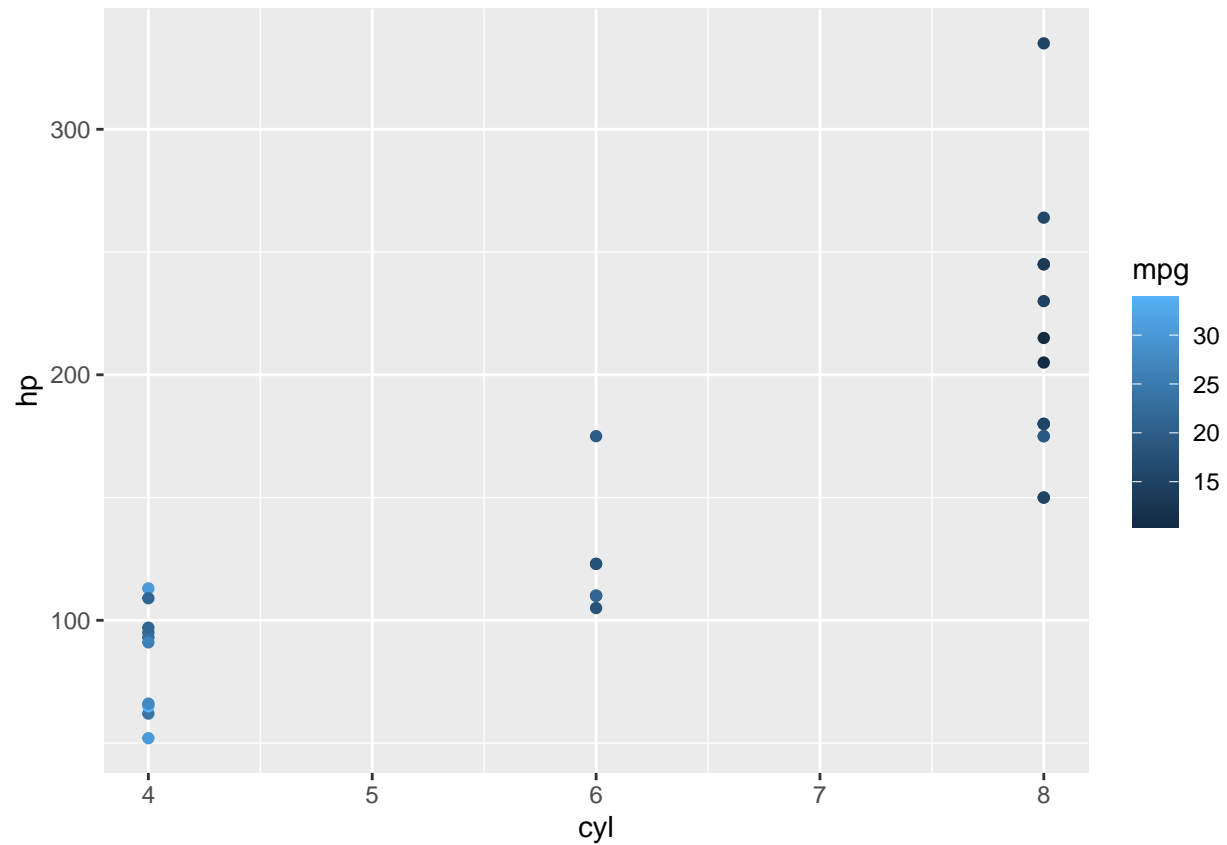
## DESARROLLO

Comenzando con gráficos simples; vamos a utilizar el dataset *mtcars*. Por lo que vamos a cargar la librería:

```
## EJEMPLO 1
library(ggplot2)
## RECORDAMOS LAS VARIABLES QUE CONTIENE EL DATASET MTCARS
names(mtcars)
```

```
[1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
[11] "carb"
```

```
##Graficamos las variables cyl en el eje x y hp en y, observa el comando geom_point()
ggplot(mtcars, aes(x=cyl, y = hp, colour = mpg )) + geom_point()
```



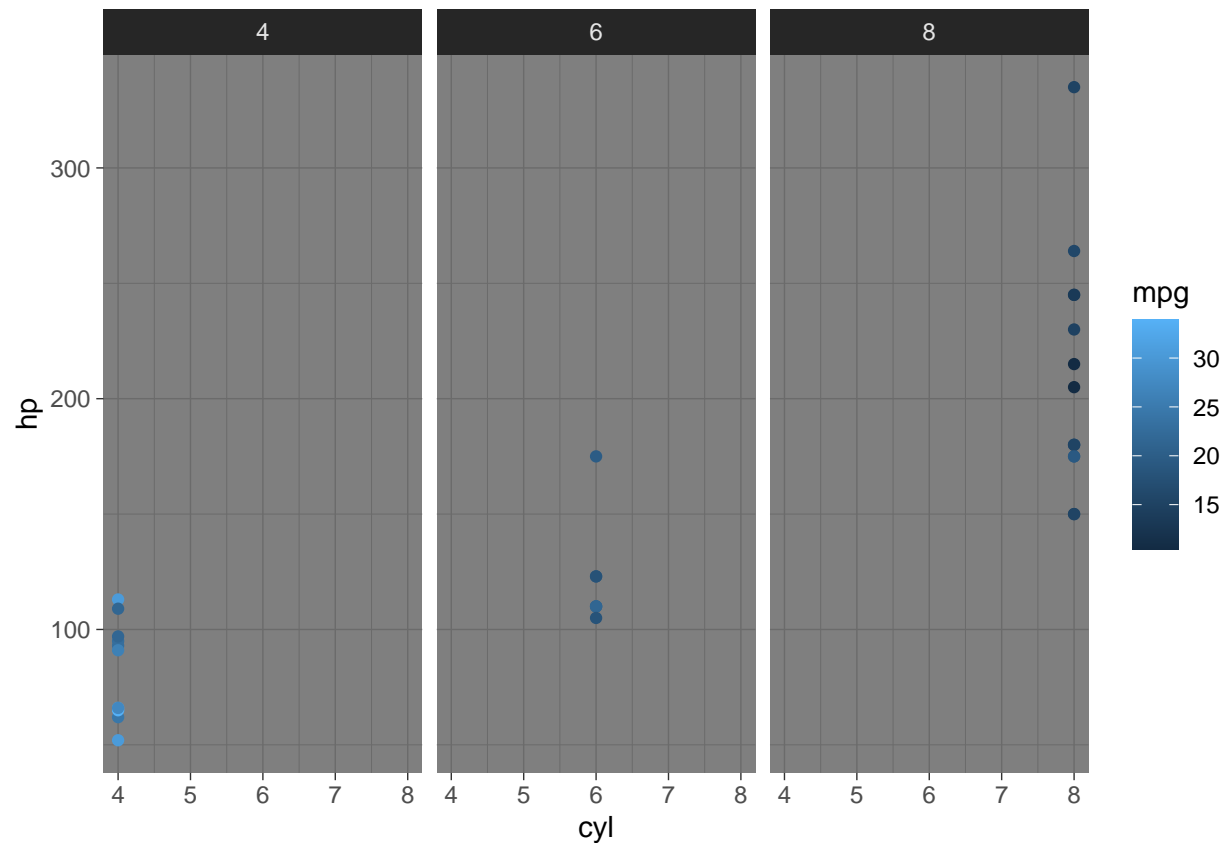
*# Tipo de geometría, intenta utilizar alguna otra, se encuentran en a cheat sheet*

*#AGREGANDO CARACTERÍSTICAS DE TEMA Y FACEWRAP*

```
names(mtcars)
```

```
[1] "mpg"  "cyl"  "disp" "hp"   "drat" "wt"   "qsec" "vs"   "am"   "gear"
[11] "carb"
```

```
ggplot(mtcars, aes(x=cyl, y = hp, colour = mpg )) +
  geom_point() +
  theme_dark() + # Temas (inteta cambiarlo)
  facet_wrap("cyl") # Lo divide por el núm de cilindros
```

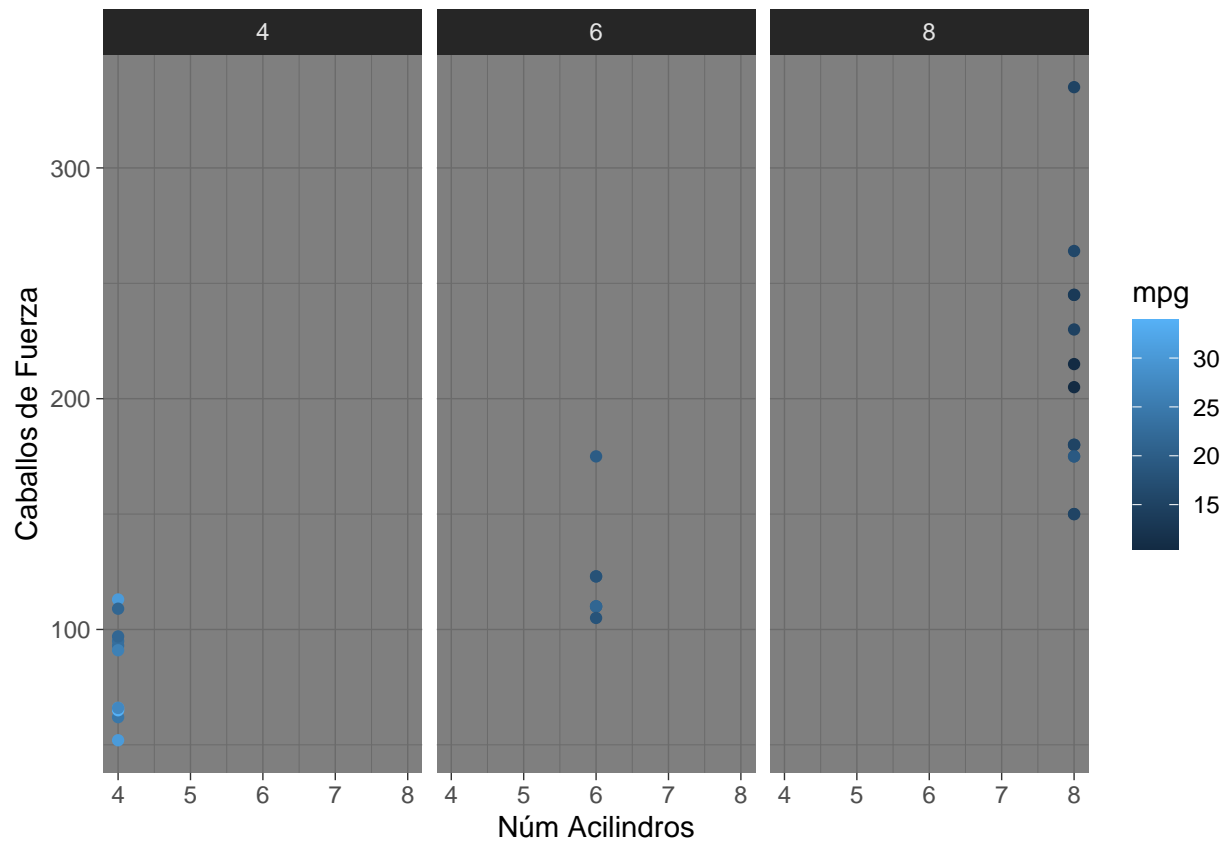


```
# CAMBIANDO LOS NOMBRES A LOS EJES X Y Y
names(mtcars)
```

```
[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear"
[11] "carb"
```

```
ggplot(mtcars, aes(x=cyl, y = hp, colour = mpg )) +
  geom_point() +
  theme_dark() + # Temas (inteta cambiarlo)
  facet_wrap("cyl") + # Lo divide por el núm de cilindros
  xlab('Núm Acilindros')+ # Nombre en los ejes
  ylab('Caballos de Fuerza')
```





## EJEMPLO 2. HISTOGRAMAS

### OBJETIVO

1. Generar histogramas de datasets
2. Cambiar propiedades de los histogramas
3. Comparar entre *hist()* y *ggplot*

### REQUISITOS

- Manipulación de datos con R
- Lectura de ficheros

### DESARROLLO

Procediendo a realizar el ejemplo tenemos lo siguiente:

```
# EJEMPLO 2
#IMPORTAMOS LA LIBRERÍA
library(dplyr)
```

Attaching package: 'dplyr'

The following objects are masked from 'package:stats':

filter, lag

The following objects are masked from 'package:base':

intersect, setdiff, setequal, union

```
#REALIZAMOS LECTURA DEL CSV
```

```
data2 <- read.csv("C:/Users/Victor Miguel Terron/Documents/PHASE2/DATA-SCIENCE-2PHASE/SESSION 3/boxp.csv")
```

```
#VEMOS LOS PRIMEROS 6 DATOS DEL DATASET
```

```
head(data2)
```

	Categoria	Grupo	Mediciones
1	C1	0	82.6
2	C1	1	112.6
3	C1	0	42.8
4	C1	0	44.6
5	C1	0	21.6
6	C1	0	11.3

```
#VEMOS LOS NOMBRES DE LAS COLUMNAS DEL DATASET
```

```
names(data2)
```

```
[1] "Categoria" "Grupo"      "Mediciones"
```

```
#GENERAMOS UN NUEVO DATASET DONDE MULTIPLICAMOS UNA CALUMNA
```

```
data <- mutate(data2, Mediciones = Mediciones*1.23)
```

```
#VEMOS EL CAMBIO
```

```
head(data)
```

	Categoria	Grupo	Mediciones
1	C1	0	101.598
2	C1	1	138.498
3	C1	0	52.644
4	C1	0	54.858
5	C1	0	26.568
6	C1	0	13.899

```
#GRAFICAMOS EN FORMA DE HISTOGRAMA UTILIZANDO HIST()
```

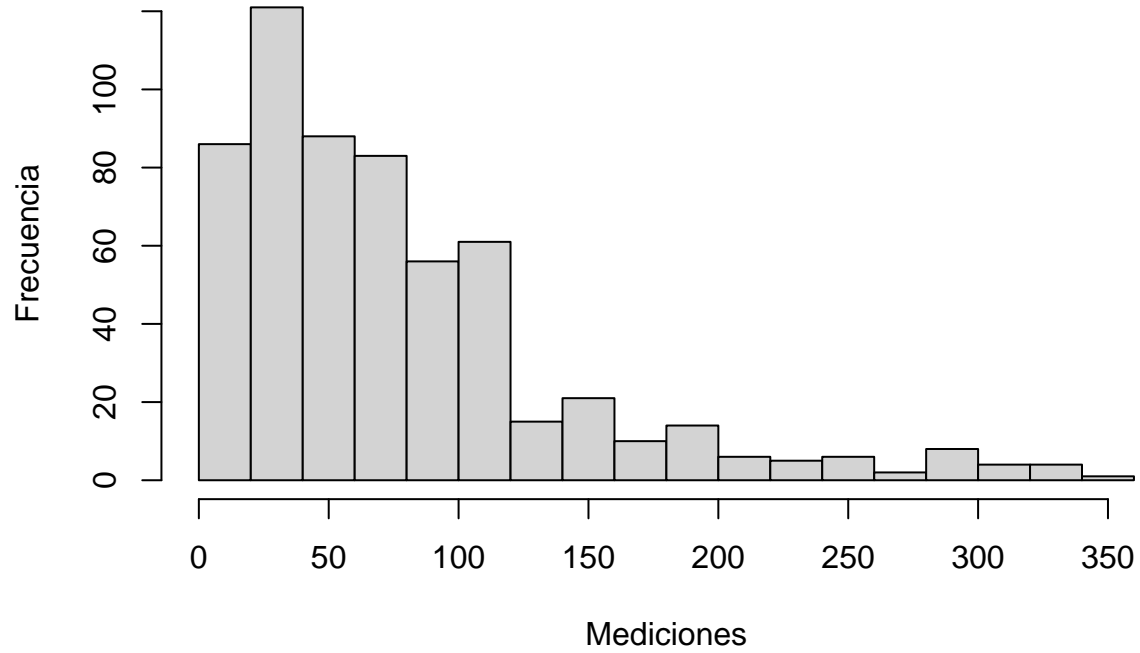
```
hist(data$Mediciones, breaks = (seq(0,360, 20)),
```

```
  main = "Histograma de Mediciones",
```

```
  xlab = "Mediciones",
```

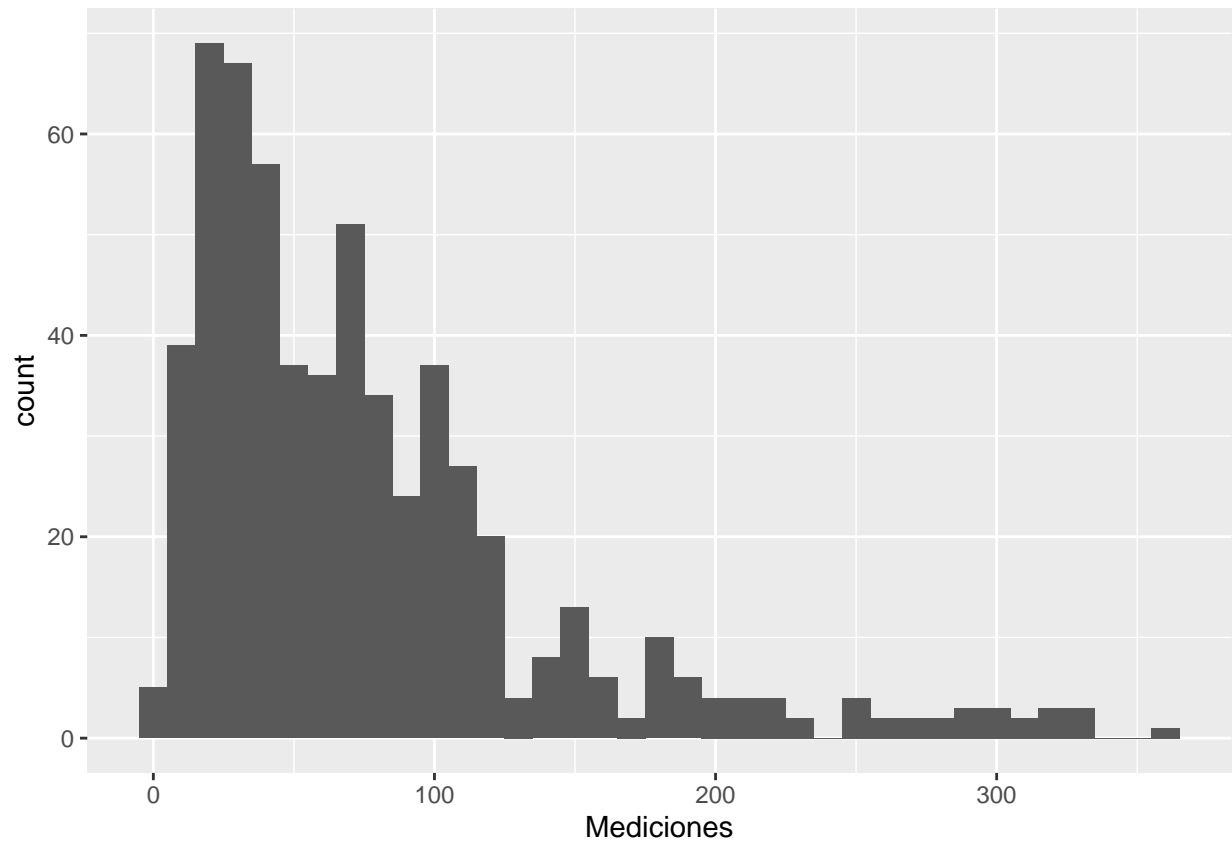
```
  ylab = "Frecuencia")
```

## Histograma de Mediciones



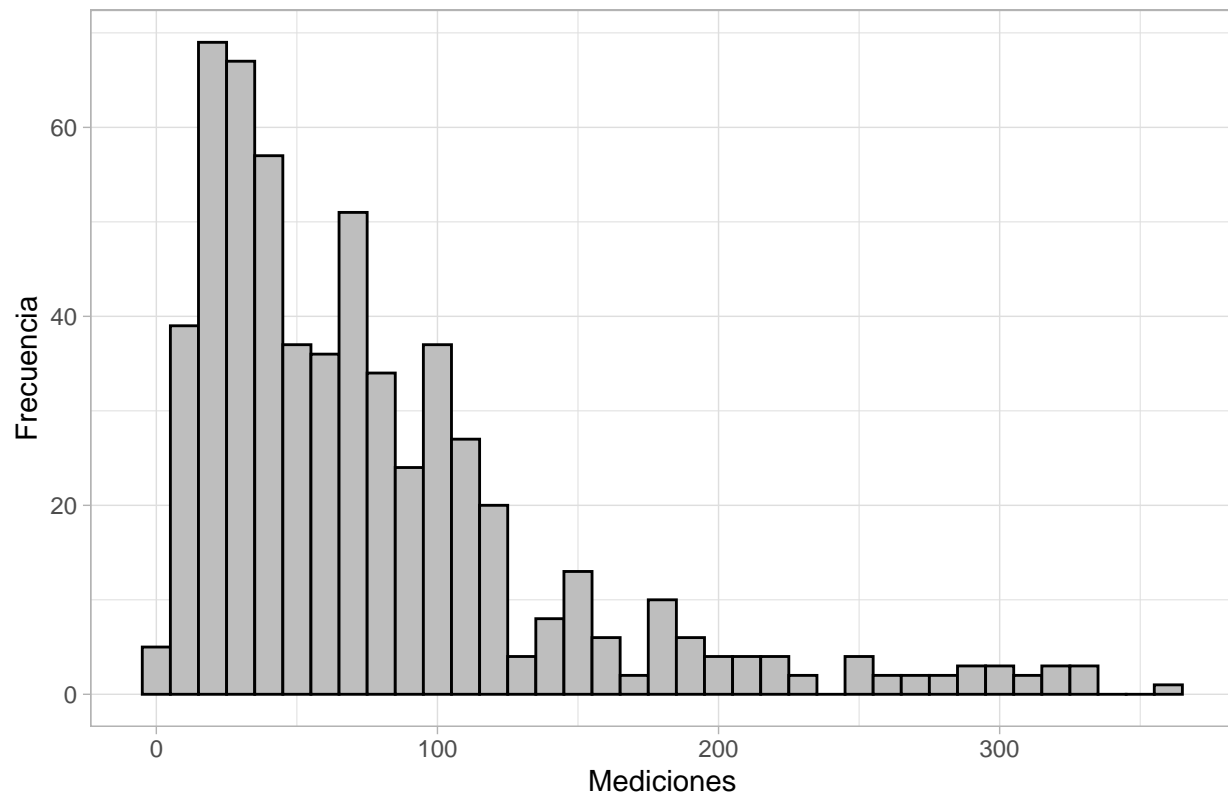
```
#GRAFICAMOS UTILIZANDO GGLOT
data <- na.omit(data)

data %>%
  ggplot() +
  aes(Mediciones) +
  geom_histogram(binwidth = 10)
```



```
#MODIFICANDO UN POCO EL ESTILO
data %>%
  ggplot() +
  aes(Mediciones) +
  geom_histogram(binwidth = 10, col="black", fill = "gray") +
  ggtitle("Histograma de Mediciones") +
  ylab("Frecuencia") +
  xlab("Mediciones") +
  theme_light()
```

## Histograma de Mediciones



Tanto `hist()` como `ggplot() + aes() + geom_histogram()` son útiles para generar histogramas, tu decide cual te funciona mejor.

## RETO 1. ALTURA DE LOS ALUMNOS

### OBJETIVO

- Crear histogramas con `hist()` y con `ggplot`
- Comparar las características de ambas funciones

### REQUISITOS

- Manipulación de datos
- Gráficas y datos

### DESARROLLO

Este ejemplo aunque es básico, servirá para realizar una comparación entre las funciones `hist()` y `ggplot`, los dos útiles, comparalos y decide cual es de tu agrado de acuerdo a los requerimientos.

1. Carga el data set `BD_Altura_Alunos.csv` (Hint: Si tienes algún inconveniente con los datos, lee sobre el argumento `sep` de la función `read.csv`)
2. Realiza el histograma con la función `hist()`, nativa de R

3. Ahora realiza el histograma con la función ggplot. (Recuerda que debes instalar el paquete ggplot2)

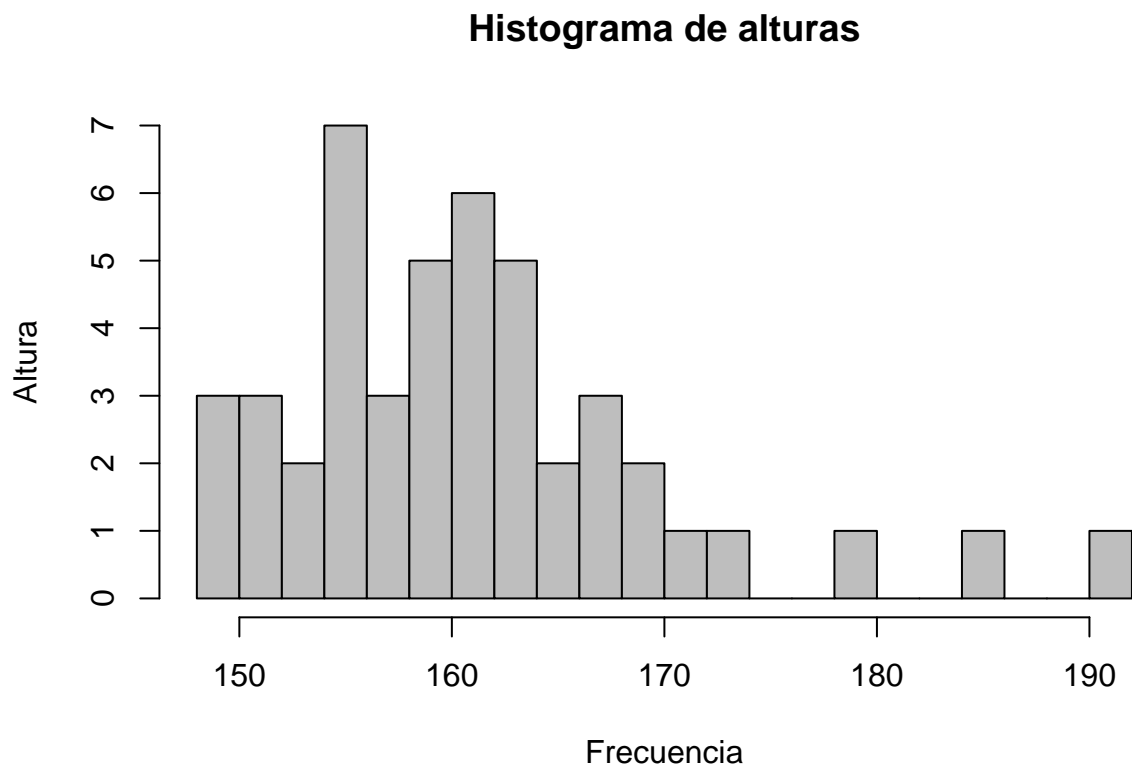
```
#RETO 1 SESSION 3
#Viendo el dataset tenemos la opción de que podemos limpiar los datos o especificando el caracter de sep
library(ggplot2)
var <- read.csv("C:/Users/Victor Miguel Terron/Documents/BD_Altura_Alunos.csv", sep = ";")
#VEMOS LAS COLUMNAS DEL DATAFRAME
names(var)
```

```
[1] "Aluno" "Altura"
```

```
dim(var)
```

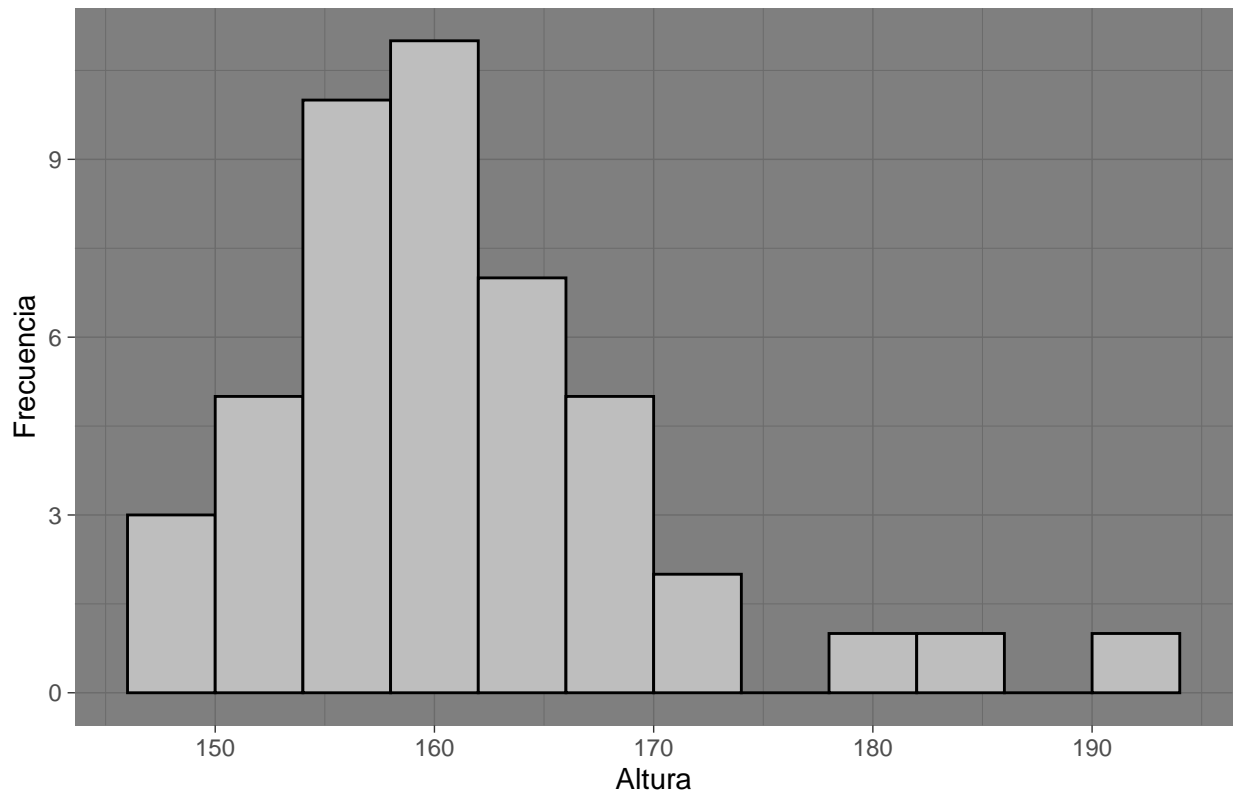
```
[1] 46 2
```

```
#UTILIZAMOS FUNCION HIST
hist(x = var$Altura,
     breaks = 20,
     main = "Histograma de alturas",
     xlab = "Frecuencia",
     ylab = "Altura",
     col="gray")
```



```
#UTILIZAMOS FUNCION GGLOT, DENTRO DE AES VA EL NOMBRE DE LA COLUMNA DEL DATAFRAME
ggplot(var,aes(Altura))+
  geom_histogram(binwidth=4,col="black",fill="gray")+
  ggtitle("Histograma de mediciones")+
  ylab("Frecuencia")+
  xlab("Altura")+
  theme_dark()
```

Histograma de mediciones



## EJEMPLO 3. GRÁFICOS DE DISPERSIÓN (SCATTER PLOTS)

### OBJETIVO

- Desarrollar habilidades para realizar e implementar scatter-plots
- Utilizar diferentes funciones de *ggplot*
- Variantes de scatter-plots

### REQUISITOS

- Tener conocimientos básicos de graficación
- Haber realizado el prework

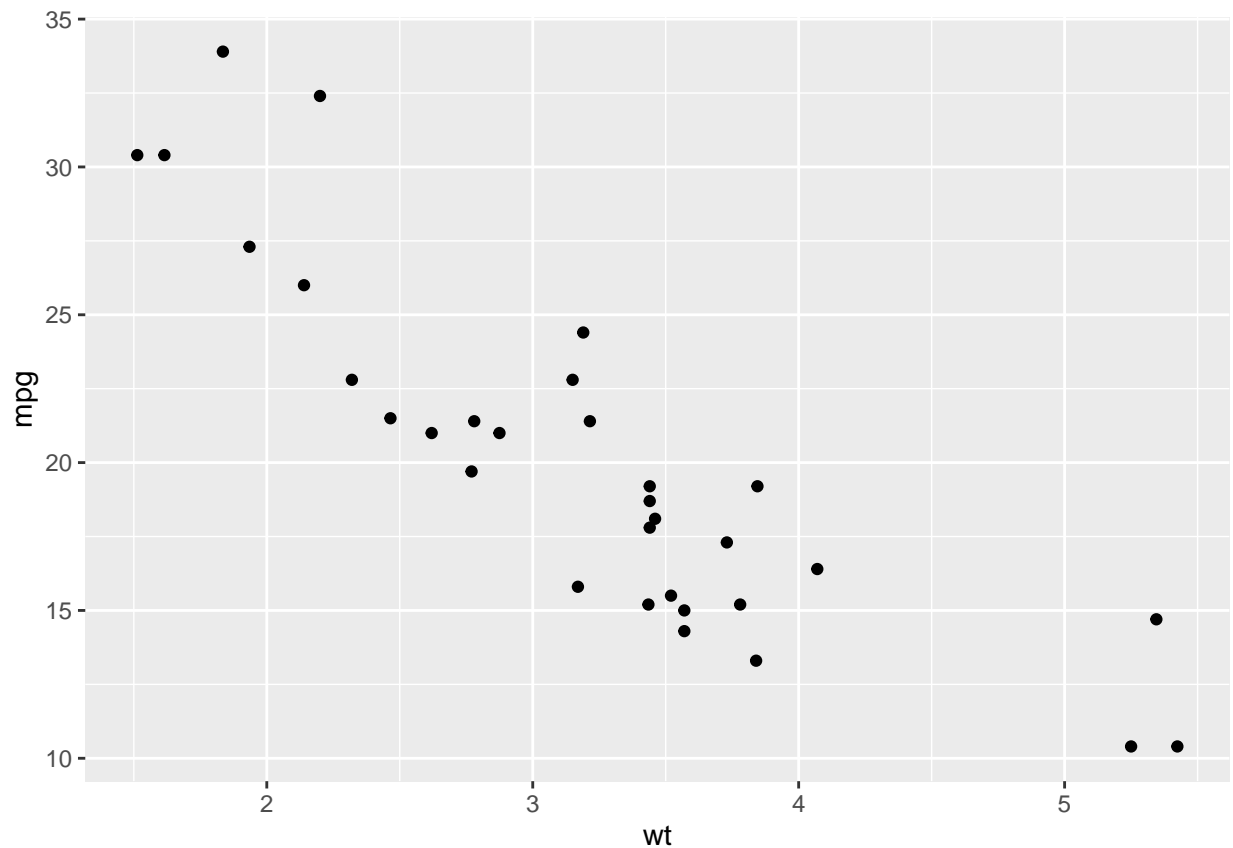
## DESARROLLO

Realizamos un scatter plot de las variables *wt* y *mpg*, debemos utilizar necesariamente *geom\_point()*. El dataset que se utilizará es el de *mtcars*

```
names(mtcars)
```

```
[1] "mpg" "cyl" "disp" "hp" "drat" "wt" "qsec" "vs" "am" "gear"  
[11] "carb"
```

```
(my_scatplot <- ggplot(mtcars, aes(x = wt, y = mpg)) + geom_point())
```

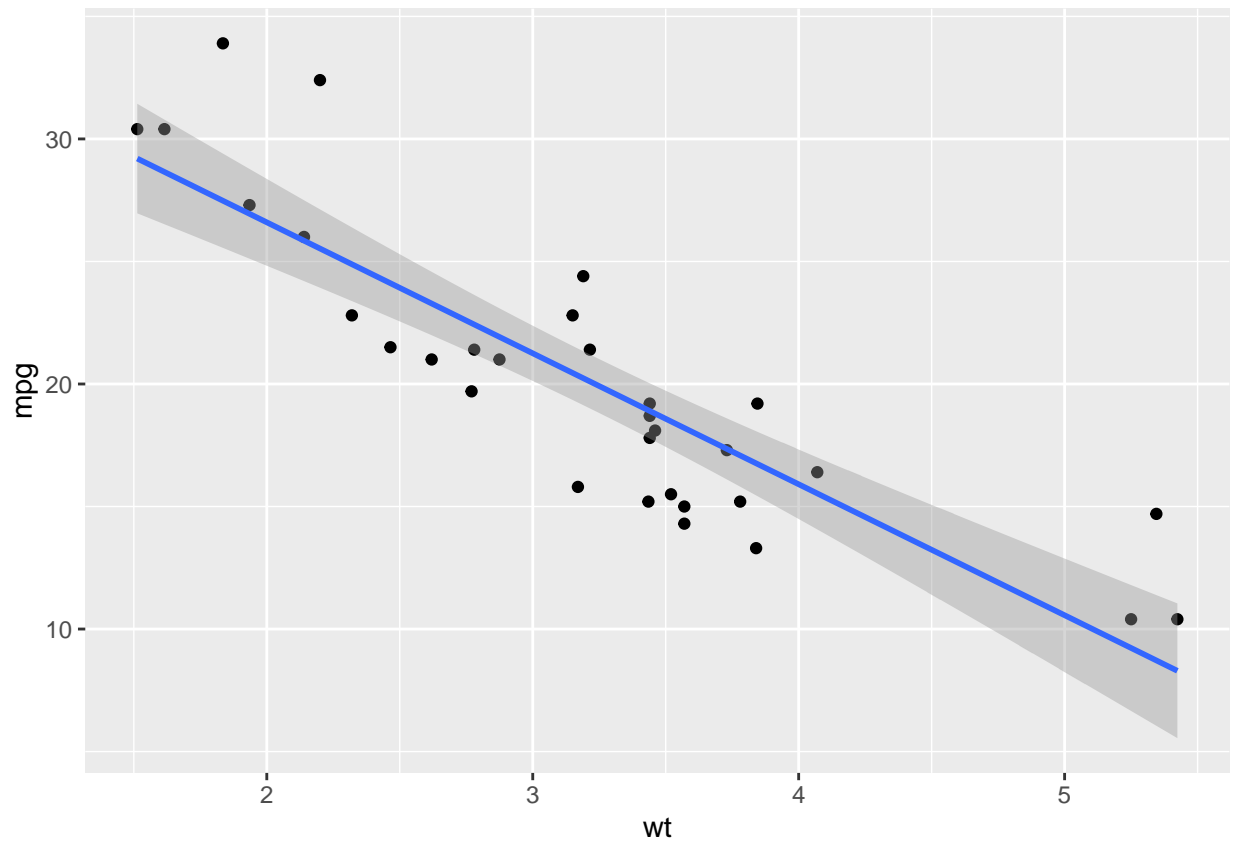


Adicionalmente se puede agregar una linea de tendencia

```
(my_scatplot <- ggplot(mtcars, aes(x = wt, y = mpg)) +  
  geom_point() +  
  geom_smooth(method = "lm", se = T)) # modelo lineal, cambia el parametro `se`, este hace referencia a
```

```
`geom_smooth()` using formula 'y ~ x'
```

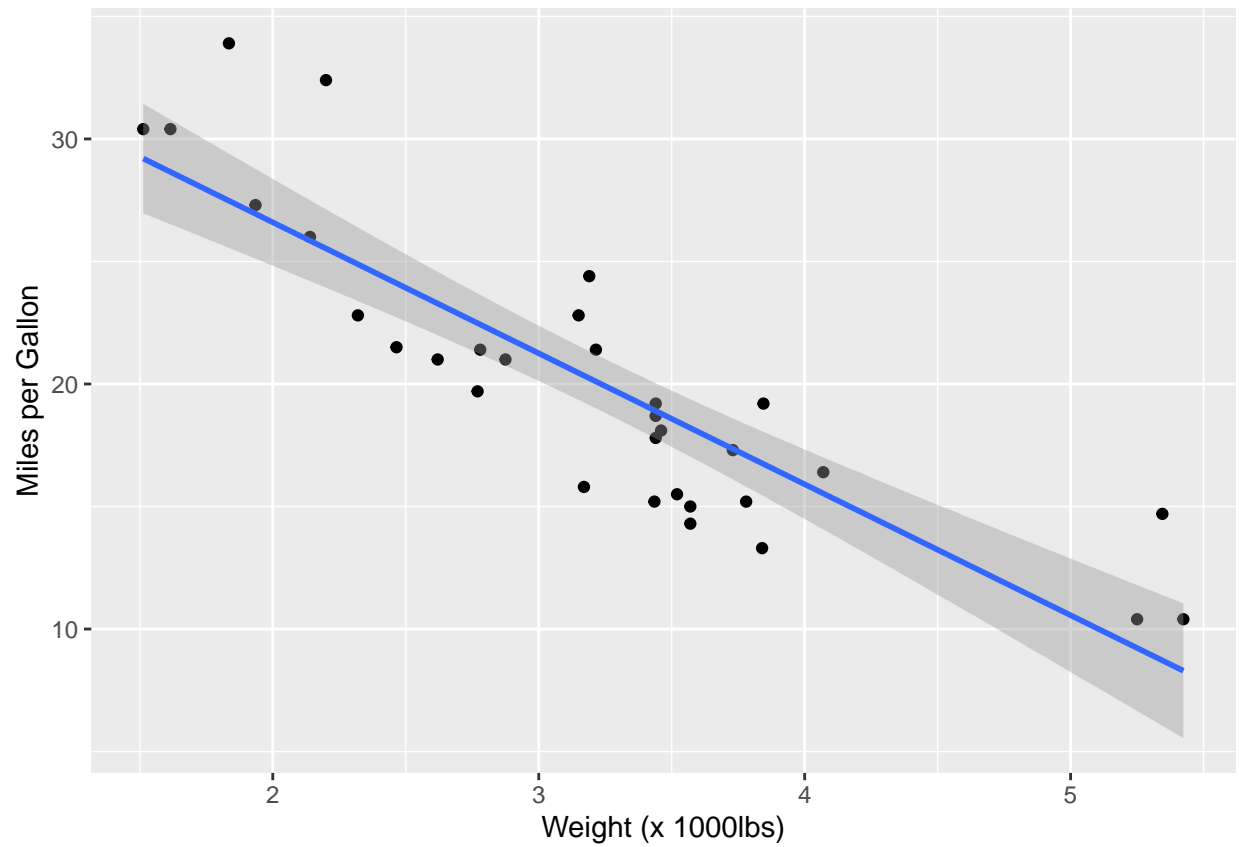




Agregando los nombres de los ejes, observa que se almacenó el gráfico en el objeto `my_scatterplot` (nota que pueden agregarse más características seguido del signo `+`).

```
my_scatterplot + xlab('Weight (x 1000lbs)') + ylab('Miles per Gallon')
```

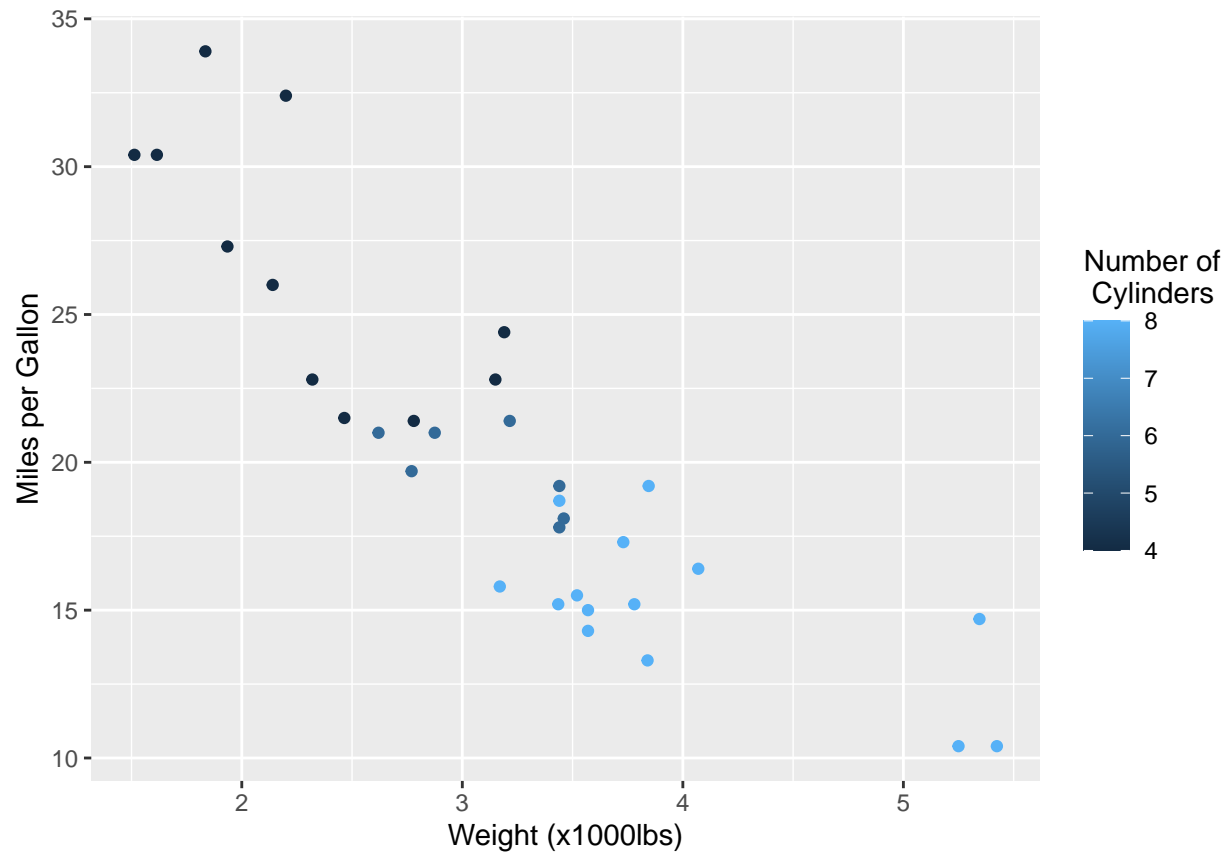
```
`geom_smooth()` using formula 'y ~ x'
```



Otras características interesantes

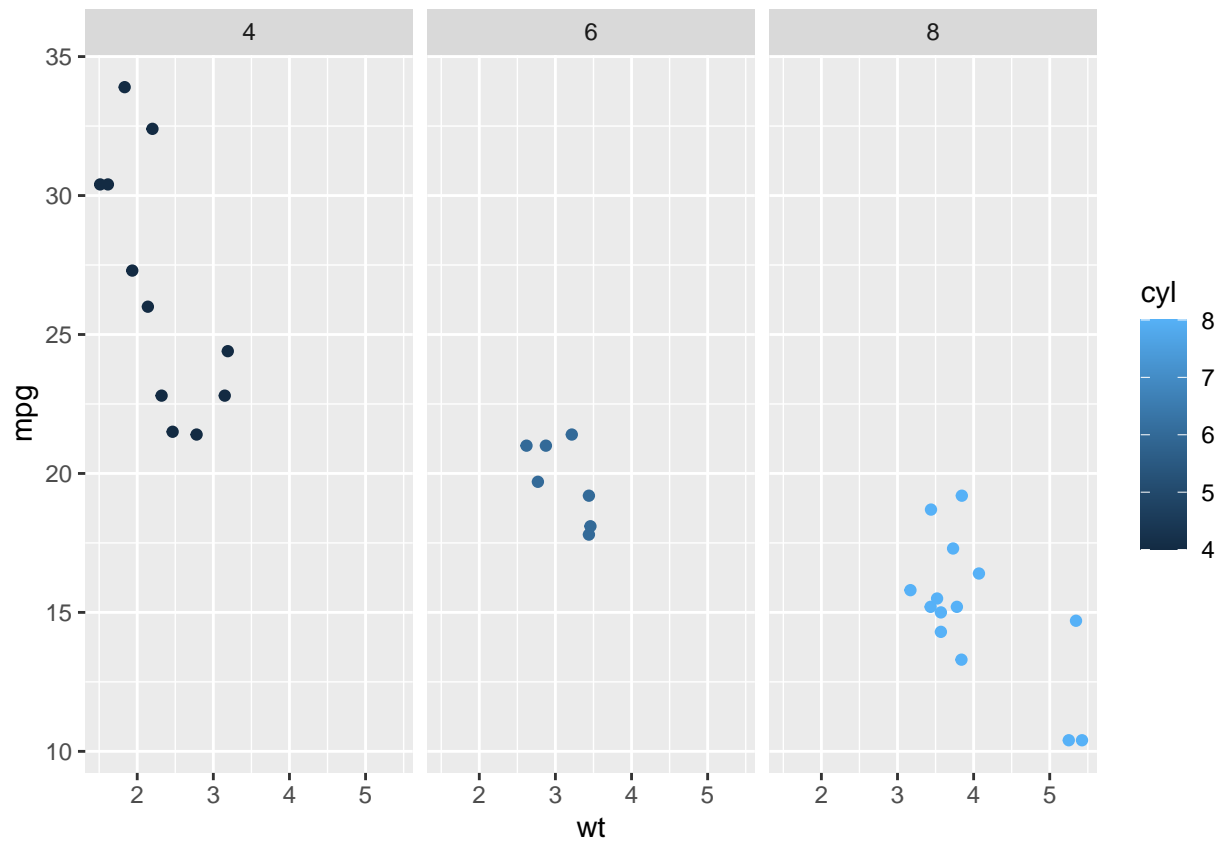
```
(my_scatterplot <- ggplot(mtcars, aes(x = wt, y = mpg, col = cyl)) + geom_point())
```





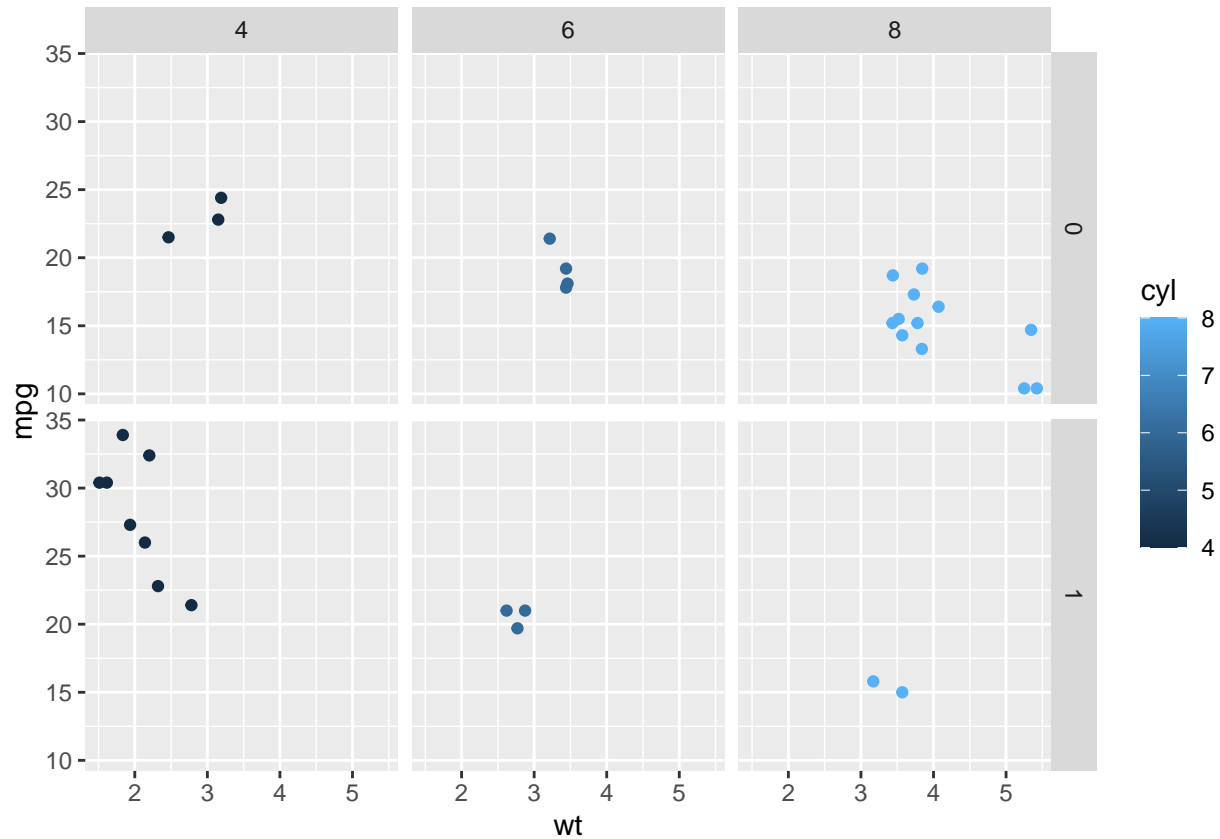
Haciendo un *facewrap* con la variable *cyl*

```
my_scatterplot + facet_wrap("cyl")
```



Separandolas por el tipo de transmisión: (am=Transmission,(0=automatic,1=manual))

```
my_scatterplot + facet_grid(am~cyl)
```



Como pudiste ver esas son algunas formas de visualizar el gráfico de dispersión, sin embargo, hay muchas más formas de representar un gráfico de dispersión.

## RETO 2. ESTADÍSTICOS DE LA NBA

Full players stats from the 2014-2015 season + personal details such as height. weight, etc.

### OBJETIVO

- Generar histogramas
- Generar gráficos de dispersión
- Identificar datos relevantes
- Hacer boxplots

### REQUISITOS

- Manejo de dataframes
- Gráficas básicas

### DESARROLLO

1. Generar un histograma de los minutos totales *MIN*, de los jugadores y agregar una línea donde se muestre la media (Para colocar la línea que muestre la media puedes consultar la documentación sobre

```
geom_line(xintercept)
```

2. Generar un histograma de edad *AGE* y agregar una línea con la media
3. Hacer un *scatterplot* de las variables *WEIGHT* y *HEIGHT* Las unidades de altura deben ser en metros.
4. Utiliza la función *WHICH.MAX* para saber quien es el jugador más alto, una vez hecho esto, presenta los resultados en una leyenda que diga “El jugador más alto es: *Name*, con una altura de: *Height*”. Las unidades de altura deben ser en metros.
5. Utiliza la función *which.min* para saber quién es el jugador más bajito, una vez hecho esto, presenta los resultados en una leyenda que diga “El jugador más bajito es: *Name*, con una altura de: *Height*”. Las unidades de altura deben ser en metros.
6. ¿Cuál es la altura promedio?, representa el resultado en una frase que diga: “La altura promedio es: *ALTURA*”
7. Generar un *scatterplot* donde se representen las Asistencias totales (*AST.TOV*) vs *Puntos* (PTS), además has un *facewrap* con la posición (*Pos*)

#### #RETO 2

```
library(ggplot2)
datasetnba <- read.csv("https://raw.githubusercontent.com/beduExpert/Programacion-con-R-Santander/master/datasetnba.csv")
names(datasetnba)
```

```
[1] "Name"           "Games.Played" "MIN"           "PTS"           "FGM"
[6] "FGA"            "FG."          "X3PM"          "X3PA"          "X3P."
[11] "FTM"            "FTA"          "FT."           "OREB"          "DREB"
[16] "REB"            "AST"          "STL"           "BLK"           "TOV"
[21] "PF"             "EFF"          "AST.TOV"       "STL.TOV"       "Age"
[26] "Birth_Place"    "Birthdate"    "Collage"       "Experience"     "Height"
[31] "Pos"            "Team"         "Weight"        "BMI"
```

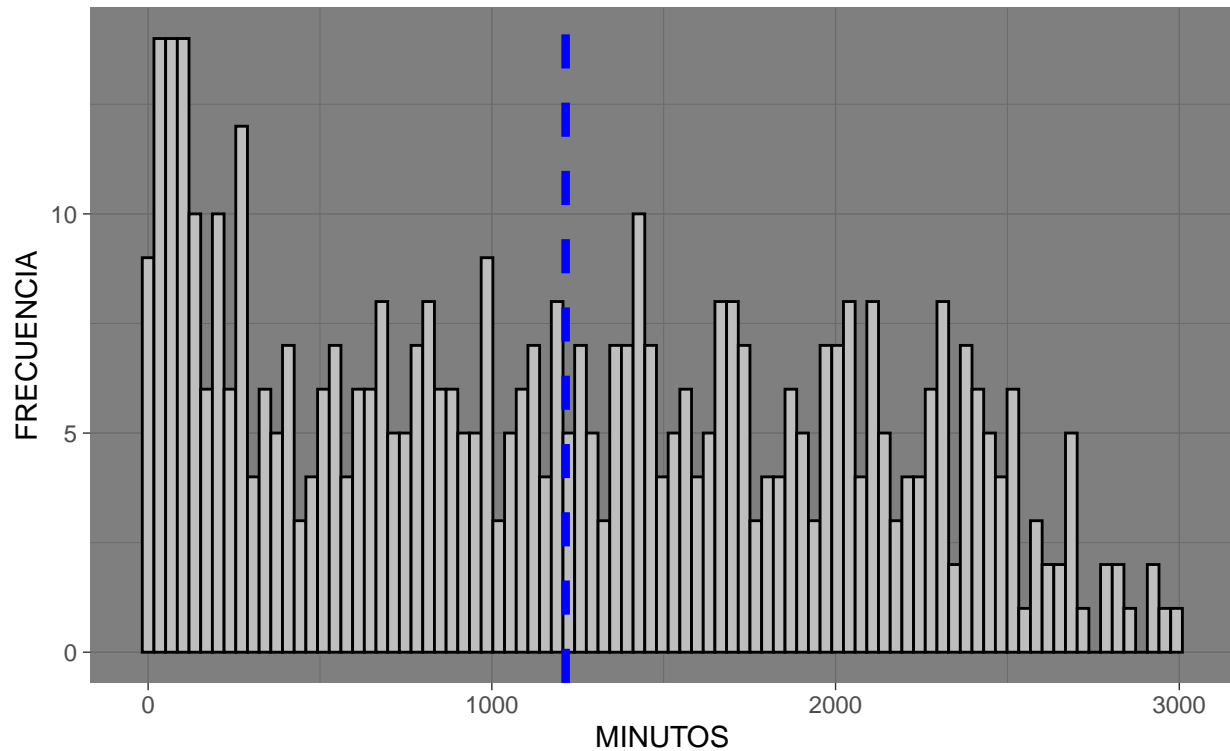
```
View(datasetnba)
```

#### #HISTOGRAMA DE LOS MINUTOS TOTALES UTILIZANDO GGLOT PUNTO 1

```
mediap1<-mean(datasetnba$MIN)
ggplot(datasetnba,aes(MIN))+
  geom_histogram(binwidth = 34,col="black",fill="gray")+
  ggtitle("HISTOGRAMA DE MINUTOS POR JUGADOR",paste("Media= ",mediap1))+
  xlab("MINUTOS")+
  ylab("FRECUENCIA")+
  geom_vline(xintercept = mediap1,col="blue",lwd=1.5,lty=2)+
  theme_dark()
```

## HISTOGRAMA DE MINUTOS POR JUGADOR

Media= 1214.71428571429



```
#HISTOGRAMA DE EDAD Y AGREGAR UNA LÍNEA CON LA MEDIA PUNTO 2  
mean(datasetnba$Age)
```

```
[1] NA
```

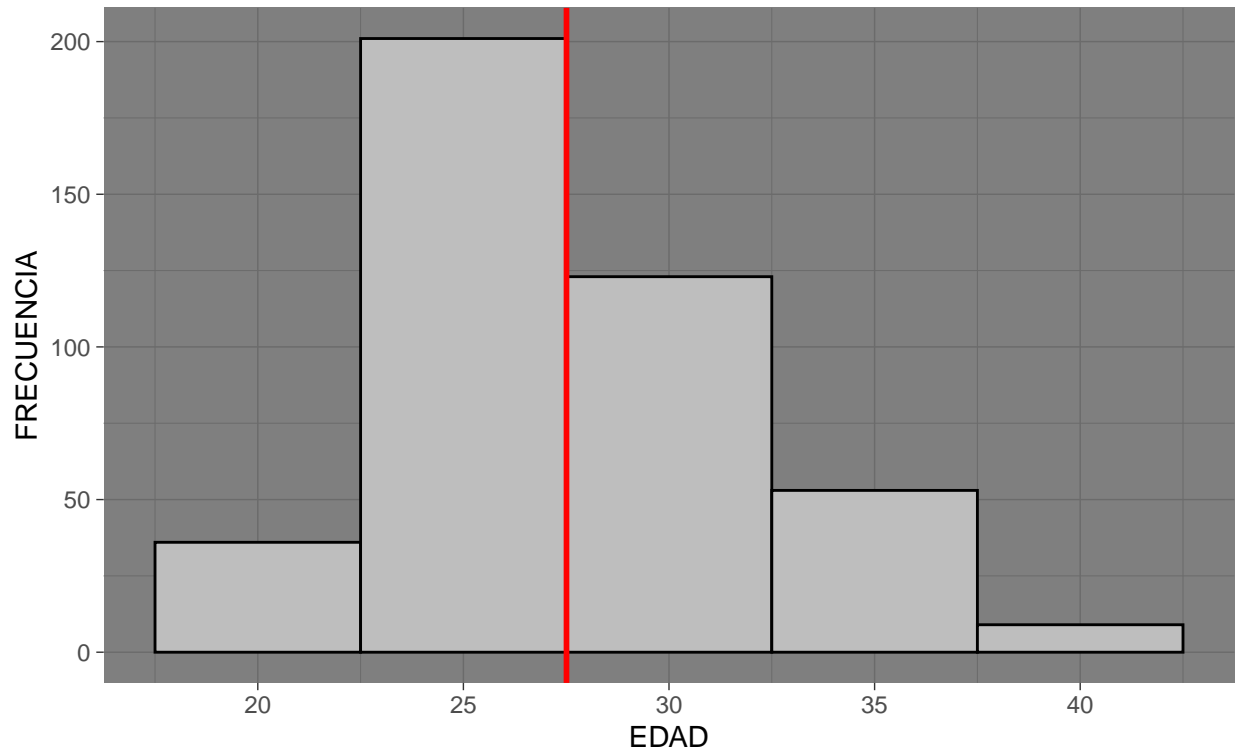
```
mediap2<-mean(na.omit(datasetnba$Age))  
ggplot(datasetnba,aes(Age))+  
  geom_histogram(binwidth = 5,col="black",fill="gray")+  
  ggtitle("EDADES DE JUGADORES",paste("Edad promedio: ",mediap2))+  
  xlab("EDAD")+  
  ylab("FRECUENCIA")+  
  geom_vline(xintercept = mediap2,col="red",lwd=1,lty=1)+  
  theme_dark()
```

Warning: Removed 68 rows containing non-finite values (stat\_bin).



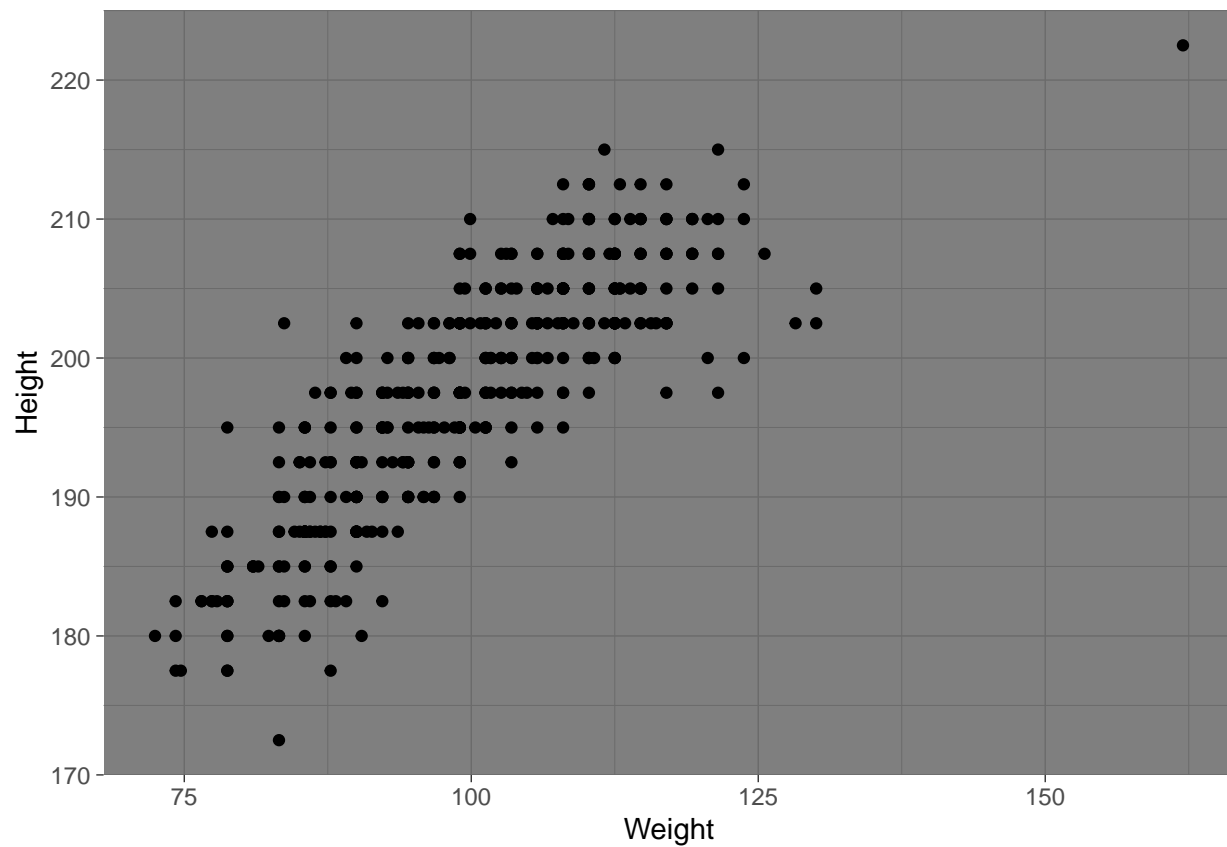
## EDADES DE JUGADORES

Edad promedio: 27.5071090047393



```
#SCATTERPLOT DE Weigth y height y observar correlación entre variables PUNTO 3  
col <- ggplot(datasetsnba, aes(x=Weight, y = Height)) +  
  geom_point() +  
  theme_dark()  
col
```

Warning: Removed 68 rows containing missing values (geom\_point).

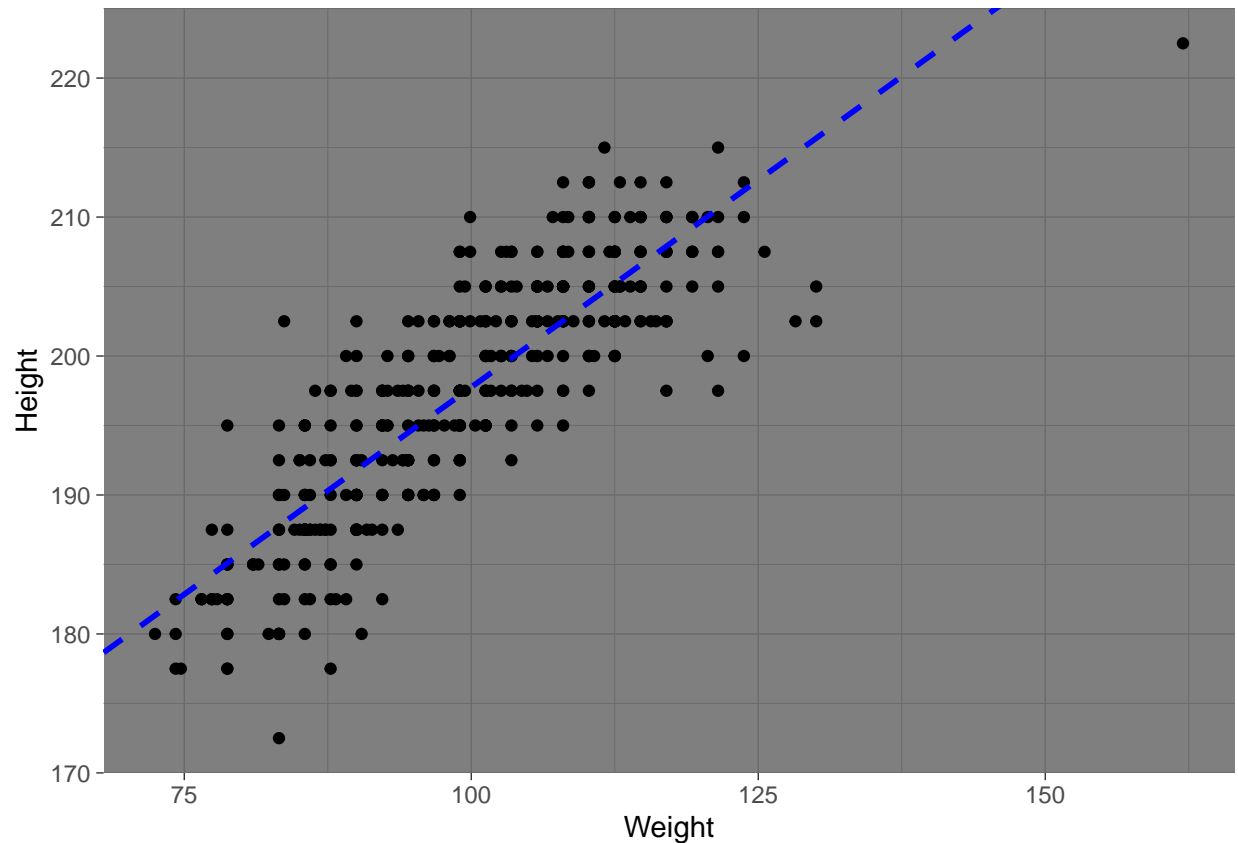


```
co2 <- coef(lm(Height~Weight,data=datasetnba ))
co2
```

```
(Intercept)      Weight
138.1294365      0.5962783
```

```
co1+geom_abline(intercept = co2[1],slope = co2[2],col="blue",lwd=1,lty=2)
```

Warning: Removed 68 rows containing missing values (geom\_point).



```
#ÉSTA ES OTRA FORMA DE REFERENCIAR EL ORIGEN DE LOS DATOS
# datasetnba%>% ggplot(aes(Weight,Height))+
#   geom_point()+
#   theme_dark()
```

```
#PUNTO 4 JUGADOR MÁS ALTO
```

```
taller<-which.max(datasetnba$Height)
```

```
#OBTENEMOS LA POSICIÓN POR LO QUE SE METE ENTRE CORCHETES
```

```
#PARA QUE AUTOMÁTICAMENTE REGRESE EL DATO
```

```
paste("El jugador más alto es: ",datasetnba$Name[taller],"con una altura de: ",round(datasetnba$Height[taller],2))
```

```
[1] "El jugador más alto es:  Sim Bhullar con una altura de:  2.22 metros"
```

```
#PUNTO 5 JUGADOR MÁS BAJITO
```

```
smaller<-which.min(datasetnba$Height)
```

```
paste("El jugador más bajito es: ",datasetnba$Name[smaller],"Con una altura de",round(datasetnba$Height[smaller],2))
```

```
[1] "El jugador más bajito es:  Isaiah Thomas Con una altura de 1.73  metros"
```

```
#PUNTO 6 ALTURA PROMEDIO, COMO HAY NA ENTONCES UTILIZAMOS UNA DE LAS FUNCIONES PARA EVITARLO
```

```
altprom<-round(mean(na.omit(datasetnba$Height)/100),2)
```

```
altprom
```

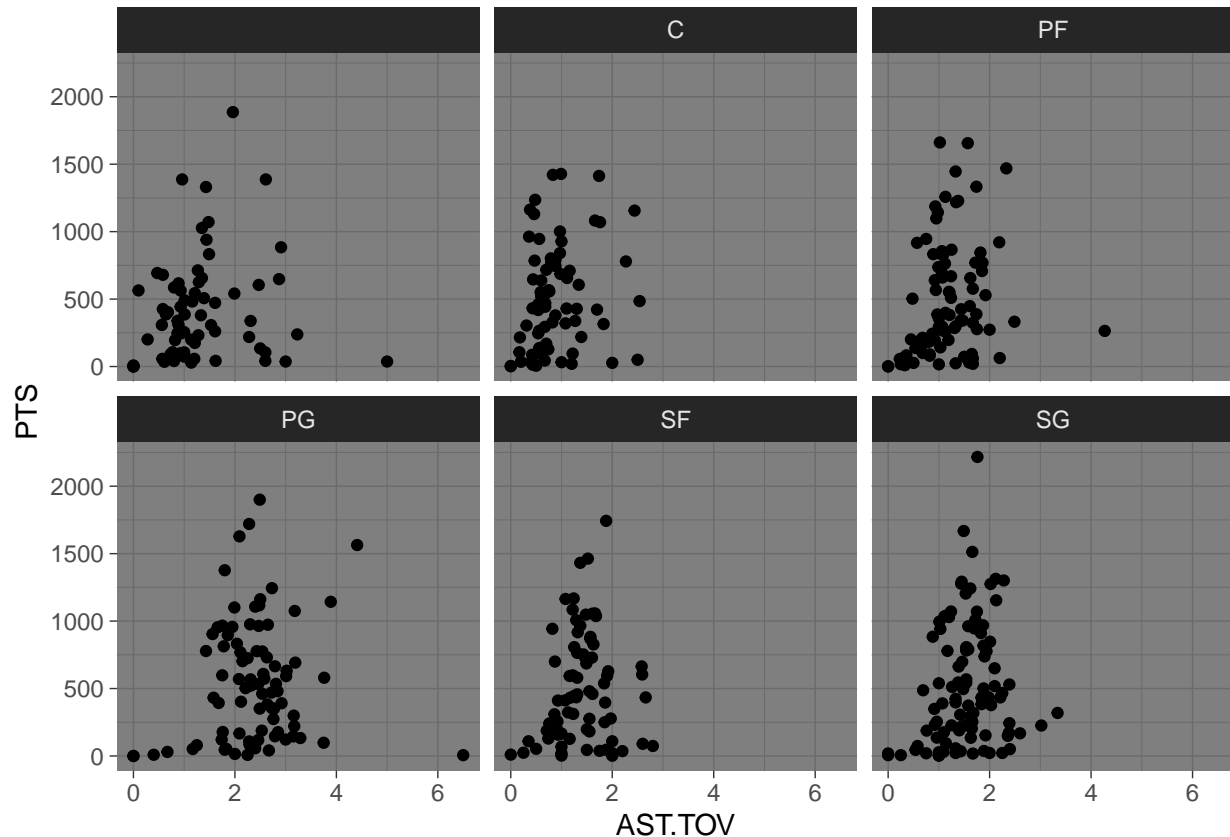
```
[1] 1.97
```

```
paste("La altura promedio de los jugadores es: ",altprom,"metros")
```

```
[1] "La altura promedio de los jugadores es:  1.97 metros"
```

*#PUNTO 7 SCATTERPLOT DE ASISTENCIAS TOTALES VS PUNTOS CON FACCEWRAP*

```
ggplot(datasetnba,aes(x=AST.TOV,y=PTS))+  
  geom_point()+  
  theme_dark()+  
  facet_wrap("Pos")
```



*#Un facewrap sirve para separar la gráfica y obtener información más detallada*

## EJEMPLO 4. BOX PLOTS Y OUTLIERS

### OBJETIVO

- Generar y comprender los tipos de gráfico boxplot
- Introducción a la distribución de los datos

### REQUISITOS

- Lectura de ficheros CSV
- Nociones básicas de *ggplot*

## DESARROLLO

Comenzamos leyendo un fichero, el cual contiene información sobre dos grupos de control G1 y G2, a los cuales se les realizó a cada uno una medición en 3 momentos diferentes C1, C2 y C3

```
data2 <- read.csv("https://raw.githubusercontent.com/beduExpert/Programacion-con-R-Santander/master/Ses.
```

Revisamos el encabezado del fichero y el nombre de sus variables o columnas

```
names(data2)
```

```
[1] "Categoria" "Grupo"      "Mediciones"
```

```
head(data2)
```

	Categoria	Grupo	Mediciones
1	C1	0	82.6
2	C1	1	112.6
3	C1	0	42.8
4	C1	0	44.6
5	C1	0	21.6
6	C1	0	11.3

Vamos a realizar un cambio en la variable *Mediciones* para practicar

```
data <- mutate(data2, Mediciones = Mediciones*1.23)  
head(data)
```

	Categoria	Grupo	Mediciones
1	C1	0	101.598
2	C1	1	138.498
3	C1	0	52.644
4	C1	0	54.858
5	C1	0	26.568
6	C1	0	13.899

Observamos algunos datos estadísticos sobre las variables

```
summary(data)
```

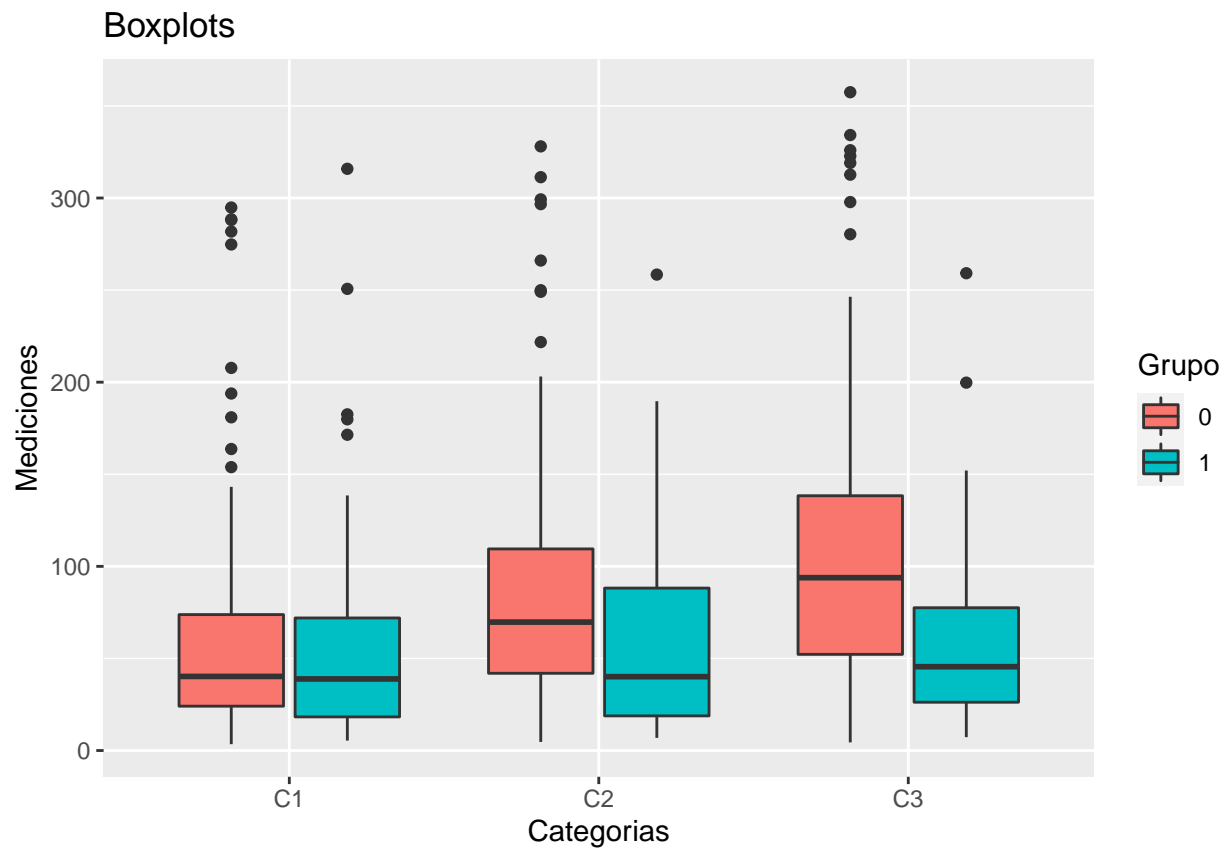
Categoria	Grupo	Mediciones
Length:615	Min. :0.0000	Min. : 3.444
Class :character	1st Qu.:0.0000	1st Qu.: 28.843
Mode :character	Median :0.0000	Median : 60.639
	Mean :0.2439	Mean : 77.349
	3rd Qu.:0.0000	3rd Qu.:101.906
	Max. :1.0000	Max. :357.438
		NA's :24

Como estamos ante la presencia de *NA's* los eliminamos con *complete.cases()* y solamente seleccionamos aquellos sin *NA's* y convertimos en factores la variable, Categoria y Grupo‘

```
bien <- complete.cases(data)
data <- data[bien,]
data <- mutate(data, Categoria = factor(Categoria), Grupo = factor(Grupo))
```

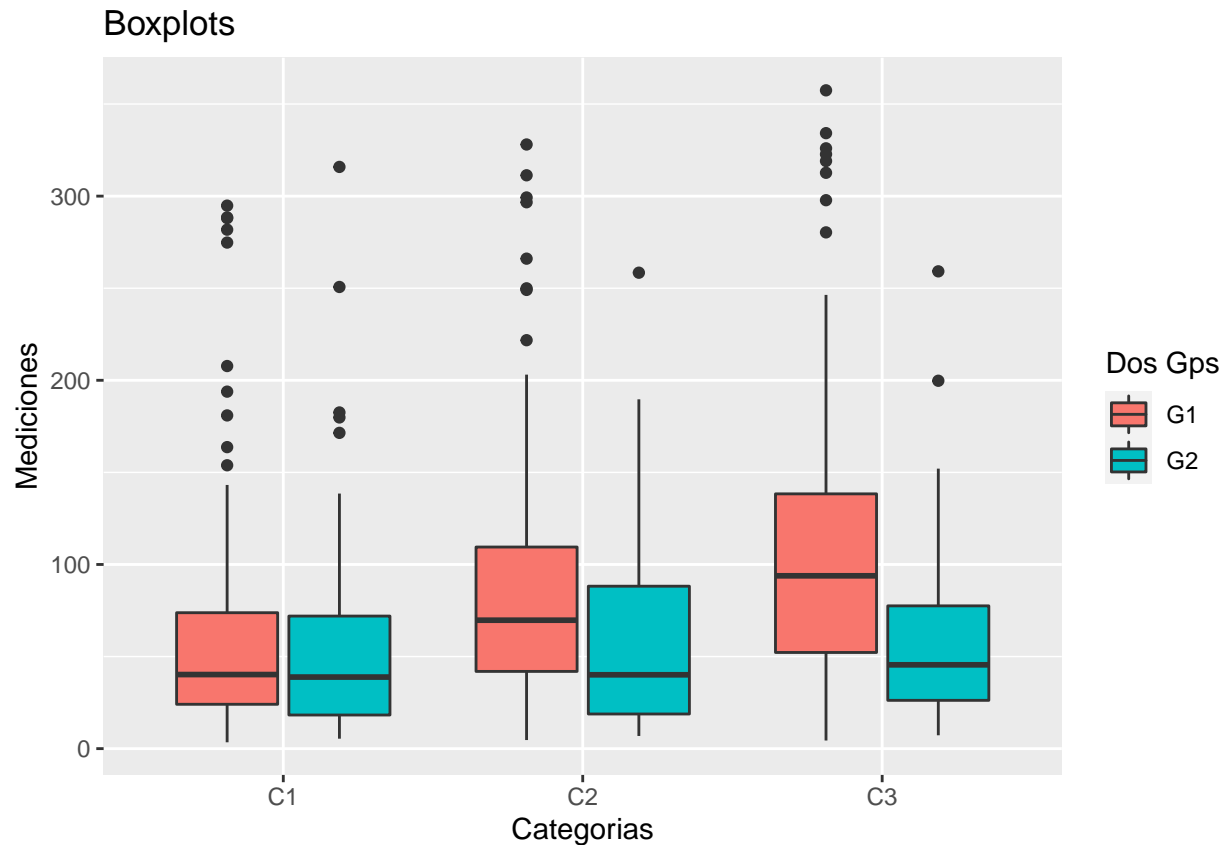
Finalmente realizamos el boxplot

```
ggplot(data, aes(x = Categoria, y = Mediciones, fill = Grupo)) + geom_boxplot() +
  ggtitle("Boxplots") +
  xlab("Categorias") +
  ylab("Mediciones")
```



Agregamos el nombre de las etiquetas para los grupos G1 y G2

```
ggplot(data, aes(x = Categoria, y = Mediciones, fill = Grupo)) + geom_boxplot() +
  scale_fill_discrete(name = "Dos Gps", labels = c("G1", "G2")) +
  ggtitle("Boxplots") +
  xlab("Categorias") +
  ylab("Mediciones")
```



## EJEMPLO 5 DIVERSOS GRÁFICOS DE TENDENCIAS APLICADOS AL COVID-19

### OBJETIVO

- Visualizar con diversos gráficos las tendencias de la enfermedad COVID-19 desde el inicio de la pandemia hasta la fecha actual en tiempo real.
- Creación de data frames especializados
- Creación de gráficos especializados

### REQUISITOS

- Lectura de ficheros locales y desde algún repositorio en internet
- Manejo de data frames con dplyr: mutate, select, rename, filter
- Uso de ggplot

### DESARROLLO

Al inicio es posible que no comprendas todo el código, trata de leerlo e ir asimilando que es lo que realiza cada línea.

Ahora se leerá el archivo C19Mexico.csv con los infectados y muertos acumulados para cada fecha creado previamente durante la sesión 2.

Un vector se diferencia de un factor por el hecho de que un vector es una lista de valores atómicos(char,logica,integer,double,complex,raw), mientras que un factor es una lista de vectores.

```
library(scales)
mex <- read.csv("C19Mexico.csv")
head(mex);tail(mex)
```

	Pais	Fecha	Infectados	Muertos	NI	NM	Letalidad	IDA	MDA	FCI	FCM	Dia
1	Mexico	2020-02-28	1	0	1	0	0	NA	NA	NA	NA	1
2	Mexico	2020-02-29	4	0	3	0	0	1	0	4.00	NA	2
3	Mexico	2020-03-01	5	0	1	0	0	4	0	1.25	NA	3
4	Mexico	2020-03-02	5	0	0	0	0	5	0	1.00	NA	4
5	Mexico	2020-03-03	5	0	0	0	0	5	0	1.00	NA	5
6	Mexico	2020-03-04	5	0	0	0	0	5	0	1.00	NA	6

	Pais	Fecha	Infectados	Muertos	NI	NM	Letalidad	IDA	MDA
321	Mexico	2021-01-13	1571901	136917	15873	1235	8.7	1556028	135682
322	Mexico	2021-01-14	1588369	137916	16468	999	8.7	1571901	136917
323	Mexico	2021-01-15	1609735	139022	21366	1106	8.6	1588369	137916
324	Mexico	2021-01-16	1630258	140241	20523	1219	8.6	1609735	139022
325	Mexico	2021-01-17	1641428	140704	11170	463	8.6	1630258	140241
326	Mexico	2021-01-18	1649502	141248	8074	544	8.6	1641428	140704

	FCI	FCM	Dia
321	1.010201	1.009102	321
322	1.010476	1.007296	322
323	1.013452	1.008019	323
324	1.012749	1.008768	324
325	1.006852	1.003301	325
326	1.004919	1.003866	326

```
mex <- mutate(mex, Fecha = as.Date(Fecha, "%Y-%m-%d"))
str(mex)
```

```
'data.frame': 326 obs. of 12 variables:
 $ Pais      : chr  "Mexico" "Mexico" "Mexico" "Mexico" ...
 $ Fecha     : Date, format: "2020-02-28" "2020-02-29" ...
 $ Infectados: int   1 4 5 5 5 5 5 6 6 7 ...
 $ Muertos   : int   0 0 0 0 0 0 0 0 0 0 ...
 $ NI        : int   1 3 1 0 0 0 0 1 0 1 ...
 $ NM        : int   0 0 0 0 0 0 0 0 0 0 ...
 $ Letalidad : num   0 0 0 0 0 0 0 0 0 0 ...
 $ IDA       : int  NA 1 4 5 5 5 5 5 6 6 ...
 $ MDA       : int  NA 0 0 0 0 0 0 0 0 0 ...
 $ FCI       : num  NA 4 1.25 1 1 ...
 $ FCM       : num  NA NA NA NA NA NA NA NA NA ...
 $ Dia       : int   1 2 3 4 5 6 7 8 9 10 ...
```

A continuación, te presentamos un panorama de la situación que se ha estado viviendo en México, debido al coronavirus. Es información simple, que puede resultar valiosa para algunas personas. Las gráficas, las hemos realizado utilizando datos que puedes encontrar en el siguiente sitio: <https://data.humdata.org/dataset/novel-coronavirus-2019-ncov-cases>



\* Acumulado de casos confirmados

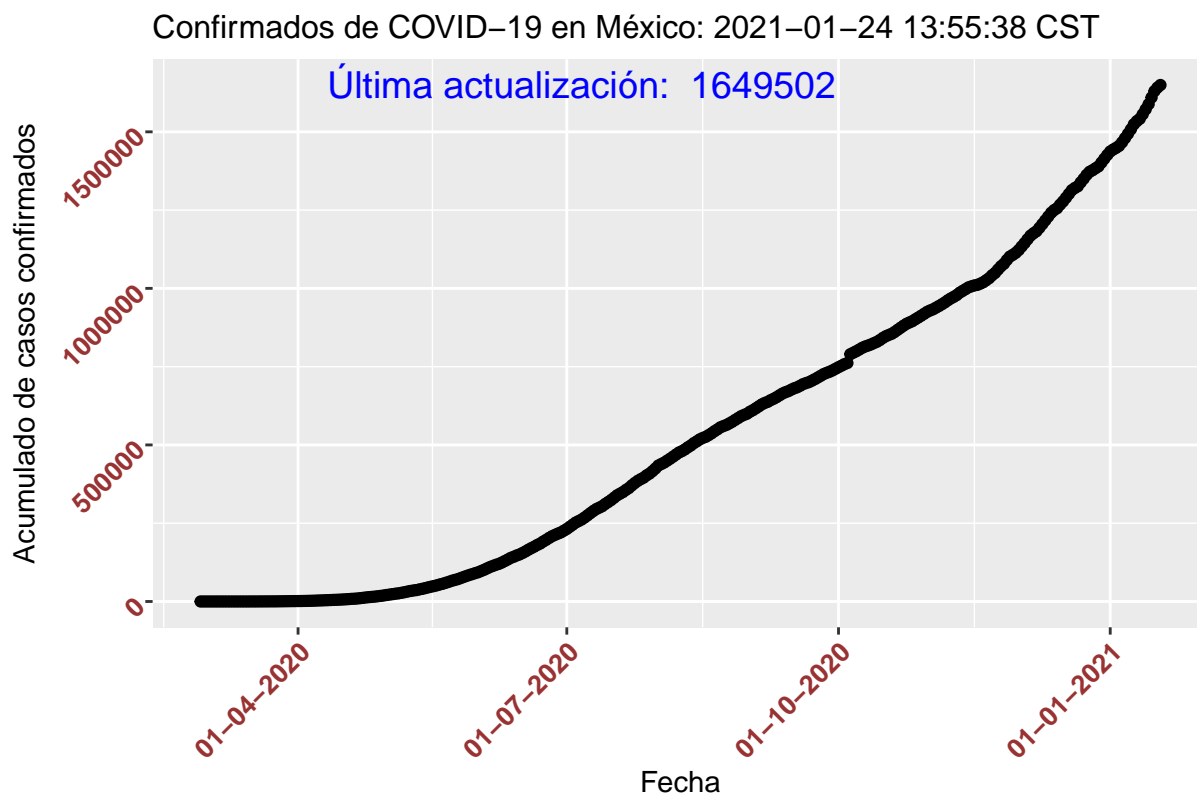
```
print("=====Acumulado de casos confirmados=====")
[1] "=====Acumulado de casos confirmados=====
p <- ggplot(mex, aes(x=Fecha, y=Infectados)) +
  geom_line( color="blue") +
  geom_point() +
  labs(x = "Fecha",
       y = "Acumulado de casos confirmados",
       title = paste("Confirmados de COVID-19 en México:",
                     format(Sys.time(),
                           tz="America/Mexico_City",
                           usetz=TRUE))) +
  theme(plot.title = element_text(size=12)) +
  theme(axis.text.x = element_text(face = "bold", color="#993333" ,
                                   size = 10, angle = 45,
                                   hjust = 1),
        axis.text.y = element_text(face = "bold", color="#993333" ,
                                   size = 10, angle = 45,
                                   hjust = 1)) # color, ángulo y estilo de las abcisas y ordenadas

p <- p + scale_x_date(labels = date_format("%d-%m-%Y")) # paquete scales

###

p <- p +
  theme(plot.margin=margin(10,10,20,10), plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mex$Fecha[round(dim(mex)[1]*0.4)], y = max(mex$Infectados), colour = "blue",

size = 5, label = paste("Última actualización: ", mex$Infectados[dim(mex)[1]]))
p
```



\* Casos confirmados por día

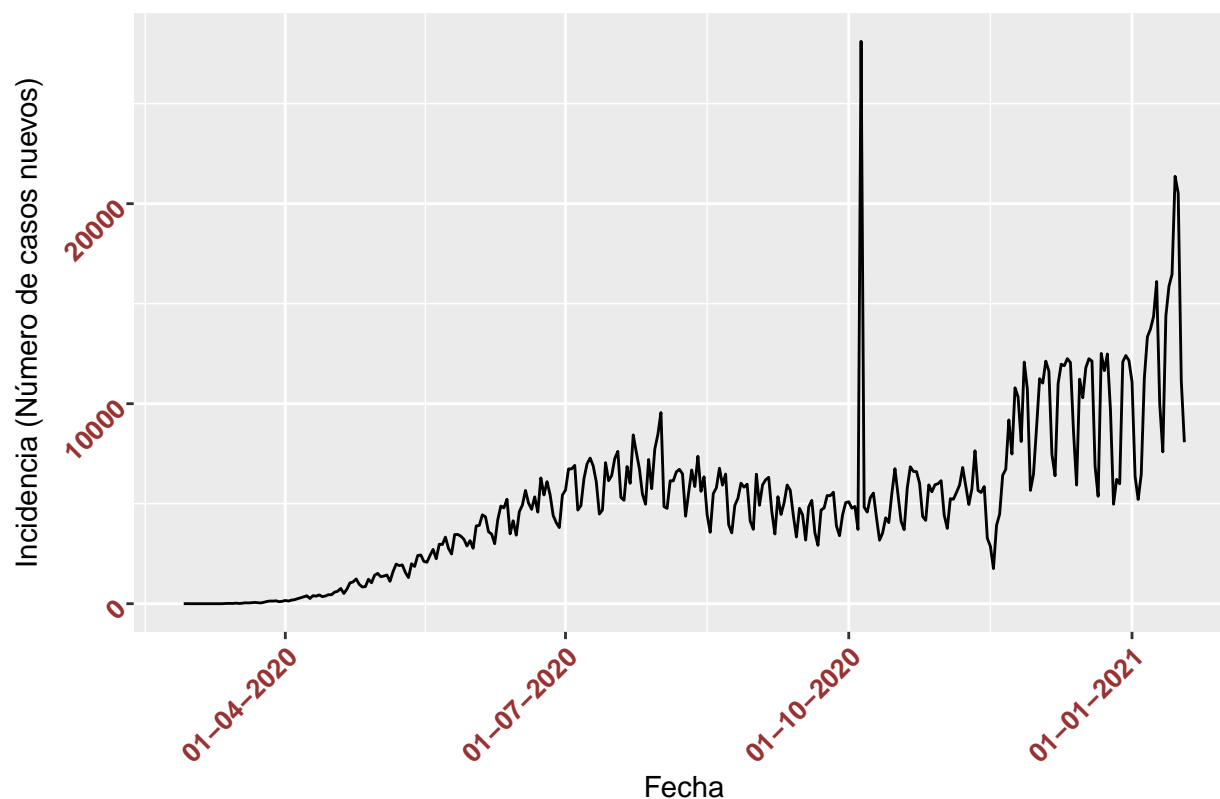
```
#CASOS CONFIRMADOS POR DÍA
print("=====Casos confirmados por día=====")
```

```
[1] "=====Casos confirmados por día====="
```

```
p <- ggplot(mex, aes(x=Fecha, y=NI)) +
  geom_line(stat = "identity") +
  labs(x = "Fecha", y = "Incidencia (Número de casos nuevos)",
       title = paste("Casos de Incidencia de COVID-19 en México:",
                     format(Sys.time(),
                           tz="America/Mexico_City", usetz=TRUE))) +
  theme(plot.title = element_text(size=12)) +
  theme(axis.text.x = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1),
        axis.text.y = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1))
# color, Ángulo y estilo de las abcisas y ordenadas

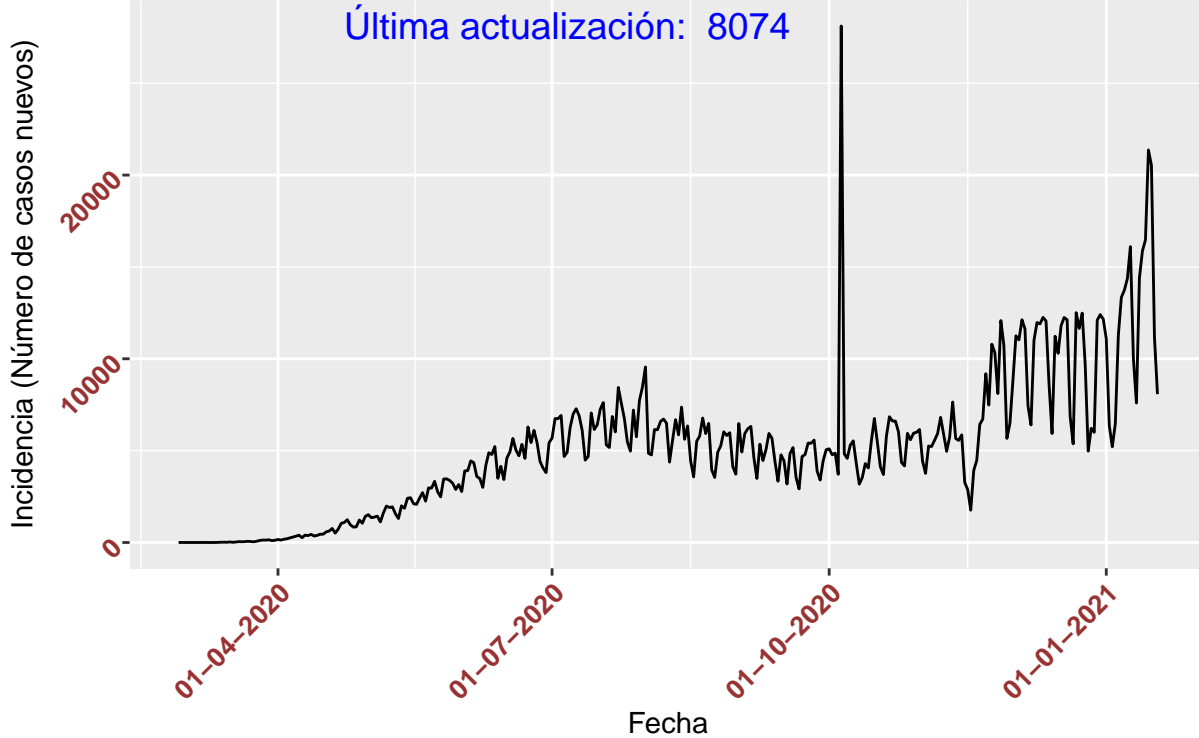
p <- p + scale_x_date(labels = date_format("%d-%m-%Y")) # paquete scales
p
```

Casos de Incidencia de COVID-19 en México: 2021-01-24 13:55:38 CST



```
###
p <- p +
  theme(plot.margin=margin(10,10,20,10),
        plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mex$Fecha[round(dim(mex)[1]*0.4)],
          y = max(mex$NI), colour = "blue", size = 5,
          label = paste("Última actualización: ", mex$NI[length(mex$NI)]))
p
```

Casos de Incidencia de COVID-19 en México: 2021-01-24 13:55:38 CST



\* Muertes Acumuladas

```
#Muertes acumuladas
print("=====Muertes acumuladas=====")
```

```
[1] "=====Muertes acumuladas====="
```

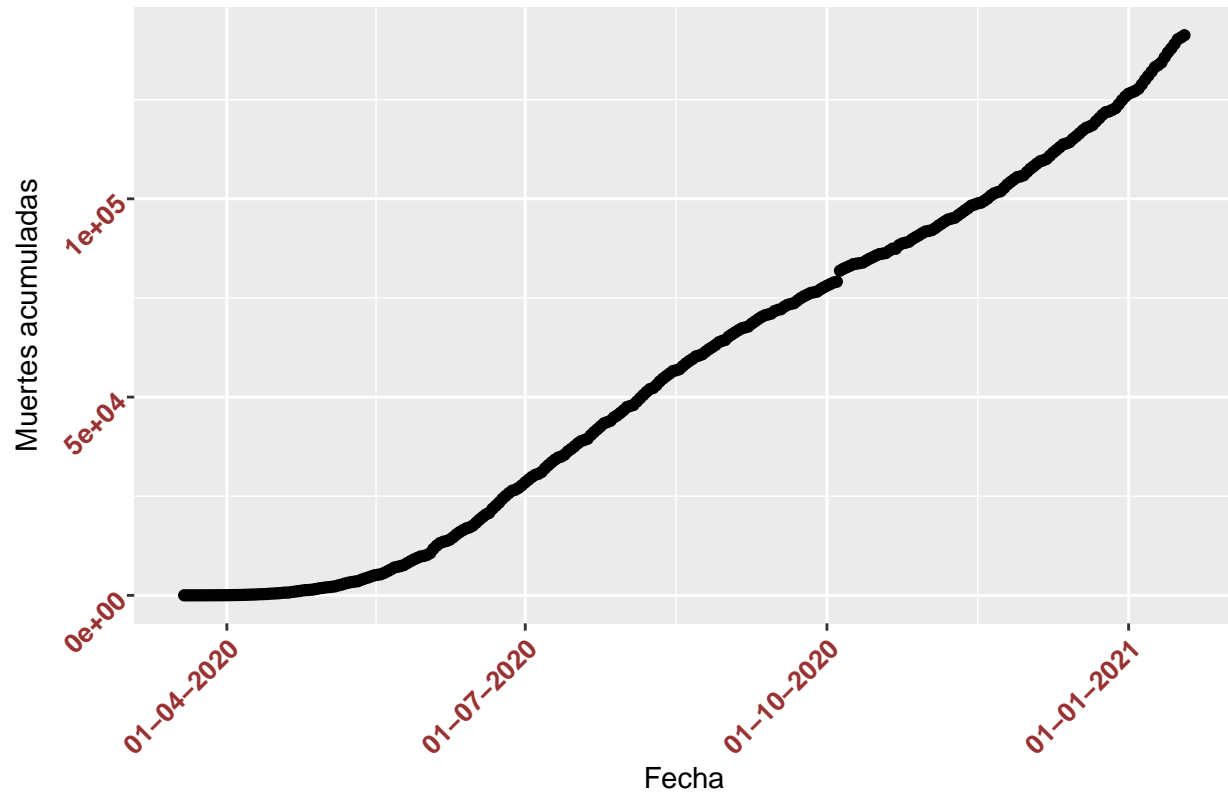
```
mexm <- subset(mex, Muertos > 0)
# Tomamos el subconjunto desde que comenzaron las muertes

p <- ggplot(mexm, aes(x=Fecha, y=Muertos)) + geom_line( color="red") +
  geom_point() +
  labs(x = "Fecha",
       y = "Muertes acumuladas",
       title = paste("Muertes por COVID-19 en México:",
                     format(Sys.time(), tz="America/Mexico_City",
                             usetz=TRUE))) +
  theme(axis.text.x = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1),
        axis.text.y = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1))
# color, Ángulo y estilo de las abscisas y ordenadas
```

```
p <- p + scale_x_date(labels = date_format("%d-%m-%Y"))
```

```
p
```

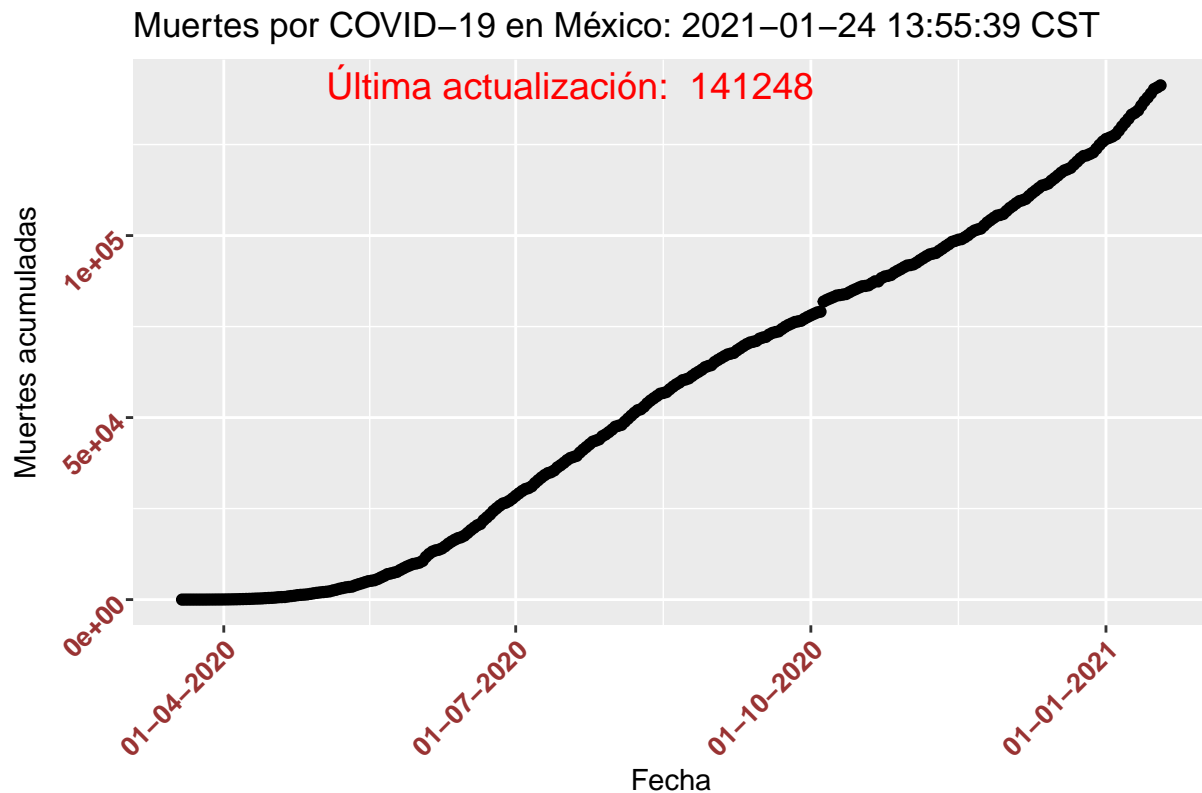
Muertes por COVID-19 en México: 2021-01-24 13:55:39 CST



```
###
```

```
p <- p +
  theme(plot.margin=margin(10,10,20,10),
        plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mexm$Fecha[round(dim(mexm)[1]*0.4)],
           y = max(mexm$Muertos), colour = "red",
           size = 5, label = paste("Última actualización: ",
                                   mexm$Muertos[dim(mexm)[1]]))
```

```
p
```



\* Muertes por día

```
print("=====Muertes por día=====")
```

```
[1] "=====Muertes por día====="
```

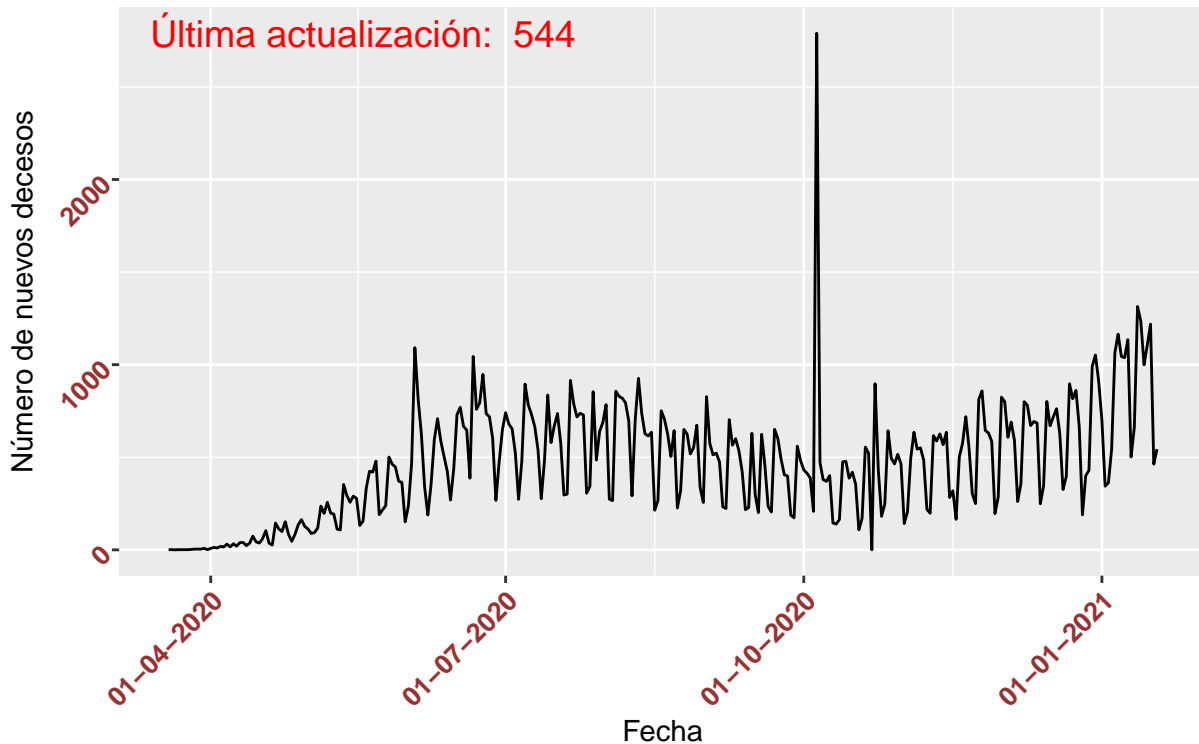
```
p <- ggplot(mexm, aes(x=Fecha, y=NM)) +
  geom_line(stat = "identity") +
  labs(x = "Fecha", y = "Número de nuevos decesos",
       title = paste("Nuevos decesos por COVID-19 en México:",
                     format(Sys.time(), tz="America/Mexico_City",
                           usetz=TRUE))) +
  theme(plot.title = element_text(size=12)) +
  theme(axis.text.x = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1),
        axis.text.y = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1))
# color, Ángulo y estilo de las abcisas y ordenadas

p <- p + scale_x_date(labels = date_format("%d-%m-%Y"))

###
```

```
p <- p +
  theme(plot.margin=margin(10,10,20,10),
        plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mexm$Fecha[round(dim(mexm)[1]*0.2)],
           y = max(mexm$NM), colour = "red", size = 5,
           label = paste("Última actualización: ", mexm$NM[dim(mexm)[1]]))
p
```

Nuevos decesos por COVID-19 en México: 2021-01-24 13:55:39 CST



\* Acumulado de casos confirmados y muertes

```
print("=====Acumulado de casos confirmados y muertes=====")
```

```
[1] "=====Acumulado de casos confirmados y muertes====="
```

```
p <- ggplot(mex, aes(x=Fecha, y=Infectados)) + geom_line(color="blue") +
  labs(x = "Fecha",
       y = "Acumulado de casos",
       title = paste("COVID-19 en México:", format(Sys.time(),
                                                    tz="America/Mexico_City",usetz=TRUE))) +
  geom_line(aes(y = Muertos), color = "red") +
  theme(axis.text.x = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1),
```

```

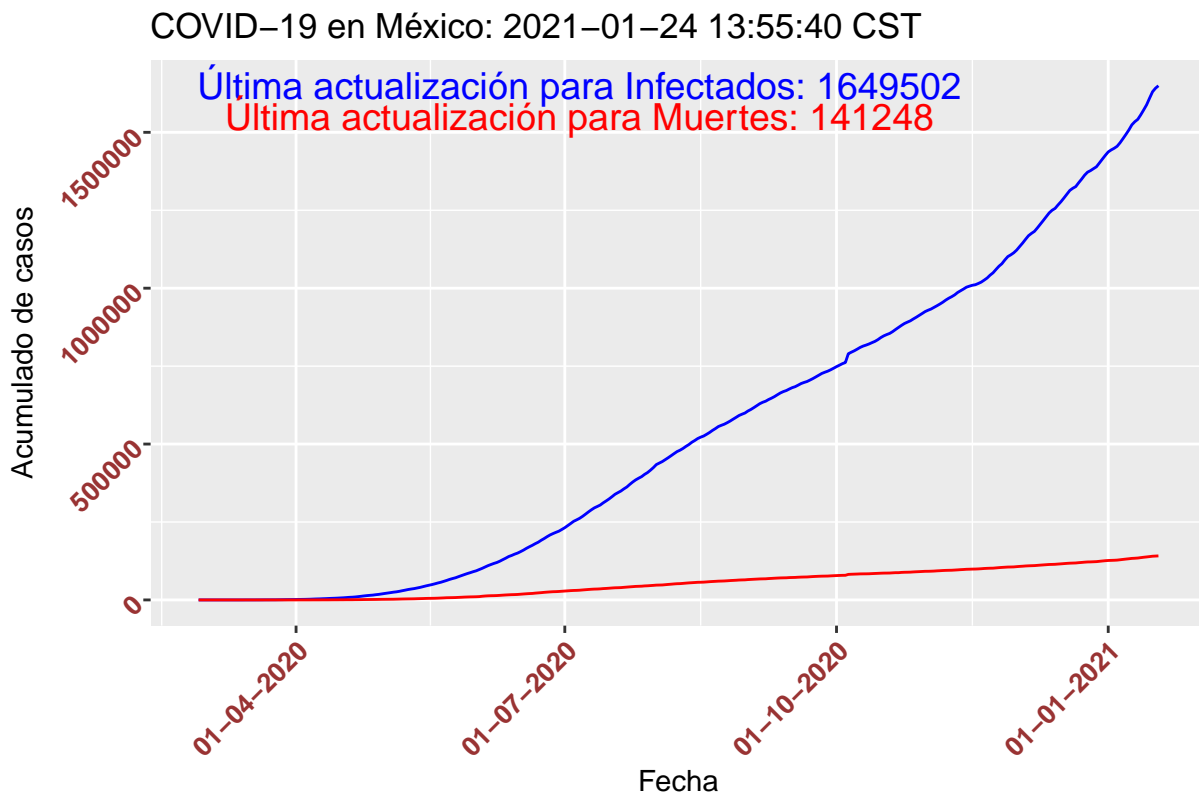
axis.text.y = element_text(face = "bold", color="#993333",
                             size = 10, angle = 45, hjust = 1))
# color, Ángulo y estilo de las abcisas y ordenadas

p <- p + scale_x_date(labels = date_format("%d-%m-%Y"))

###

p <- p +
  theme(plot.margin=margin(10,10,20,10),
        plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mex$Fecha[round(dim(mex)[1]*0.4)],
            y = max(mex$Infectados), colour = "blue",
            size = 5, label = paste("Última actualización para Infectados:",
                                     mex$Infectados[dim(mex)[1]])) +
  annotate("text", x = mex$Fecha[round(dim(mex)[1]*0.4)],
            y = max(mex$Infectados)-100000, colour = "red",
            size = 5, label = paste("Última actualización para Muertes:",
                                     mex$Muertos[dim(mex)[1]]))
p

```



### Tasa de letalidad

La tasa de letalidad observada para un día determinado, la calculamos dividiendo las muertes acumuladas reportadas hasta ese día, entre el acumulado de casos confirmados para el mismo día. Multiplicamos el



resultado por 100 para reportarlo en forma de porcentaje. Lo que obtenemos es el porcentaje de muertes del total de casos confirmados.

```
print("=====TASA DE LETALIDAD=====")
```

```
[1] "=====TASA DE LETALIDAD====="
```

```
p <- ggplot(mexm, aes(x=Fecha, y=Letalidad)) + geom_line(color="red") +
  labs(x = "Fecha",
       y = "Tasa de letalidad",
       title = paste("COVID-19 en México:",
                     format(Sys.time(), tz="America/Mexico_City",
                           usetz=TRUE))) +
  theme(axis.text.x = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1),
        axis.text.y = element_text(face = "bold", color="#993333",
                                    size = 10, angle = 45, hjust = 1)) +
  # color, Ángulo y estilo de las abcisas y ordenadas
  scale_y_discrete(name = "Tasa de letalidad",
                  limits=factor(seq(1, 13.5, 1)),
                  labels=paste(seq(1, 13.5, 1), "%", sep = ""))

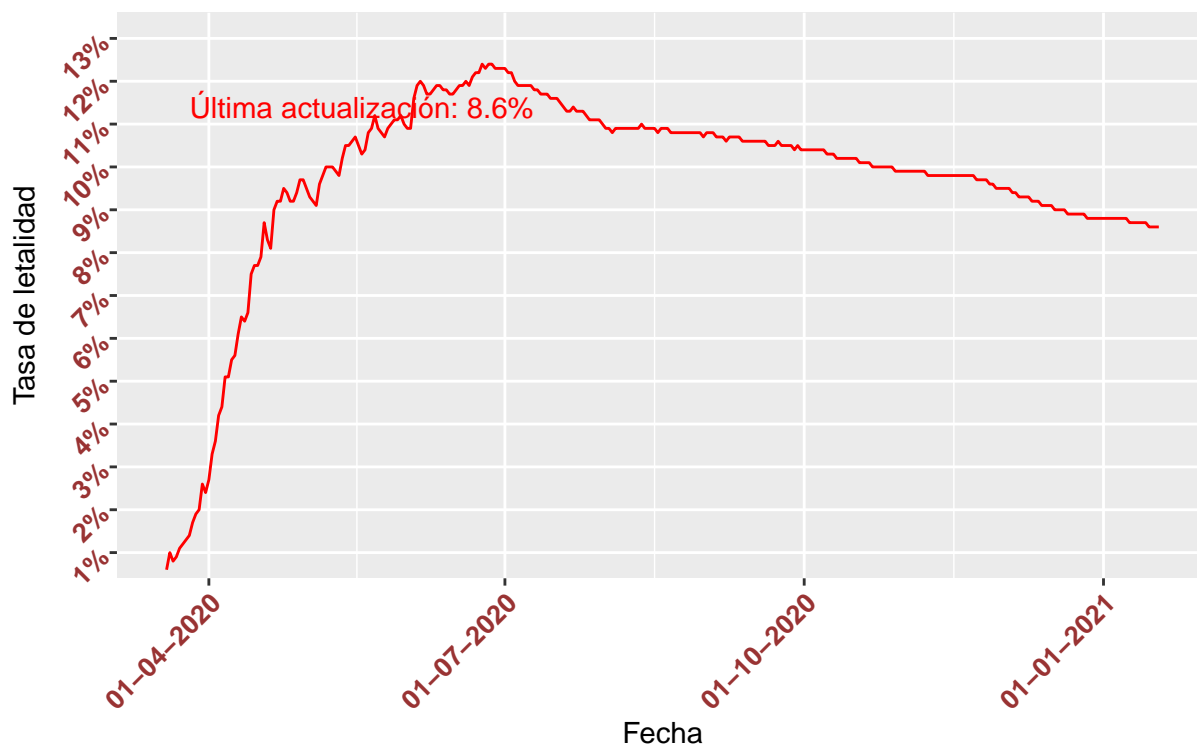
p <- p + scale_x_date(labels = date_format("%d-%m-%Y"))

###

p <- p +
  theme(plot.margin=margin(10,10,20,10),
        plot.caption=element_text(hjust=1.05, size=10)) +
  annotate("text", x = mexm$Fecha[round(length(mexm$Fecha)*0.2)],
           y = max(mexm$Letalidad)-1, colour = "red",
           size = 4, label = paste("Última actualización: ",
                                   mexm$Letalidad[dim(mexm)[1]],
                                   "%", sep = ""))

p
```

## COVID-19 en México: 2021-01-24 13:55:40 CST



### Factores de crecimiento

El factor de crecimiento de infectados para un día determinado, lo calculamos al dividir el acumulado de infectados para ese día, entre el acumulado de infectados del día anterior. El factor de crecimiento de muertes lo calculamos de forma similar.

```
print("=====FACTOR DE CRECIMIENTO=====")
```

```
[1] "=====FACTOR DE CRECIMIENTO====="
```

```
mex <- filter(mex, FCM < Inf) # Tomamos solo valores reales de factores de crecimiento

p <- ggplot(mex, aes(x=Fecha, y=FCI)) + geom_line(color="blue") +
  labs(x = "Fecha",
       y = "Factor de crecimiento",
       title = paste("COVID-19 en México:",
                     format(Sys.time(), tz="America/Mexico_City",
                           usetz=TRUE))) +
  geom_line(aes(y = FCM), color = "red") +
  theme(plot.title = element_text(size=12)) +
  theme(axis.text.x = element_text(face = "bold",
                                    color="#993333" ,
                                    size = 10,
                                    angle = 45,
```

```

                                hjust = 1),
axis.text.y = element_text(face = "bold",
                             color="#993333" ,
                             size = 10,
                             angle = 45,
                             hjust = 1))
# color, Ángulo y estilo de las abscisas y ordenadas

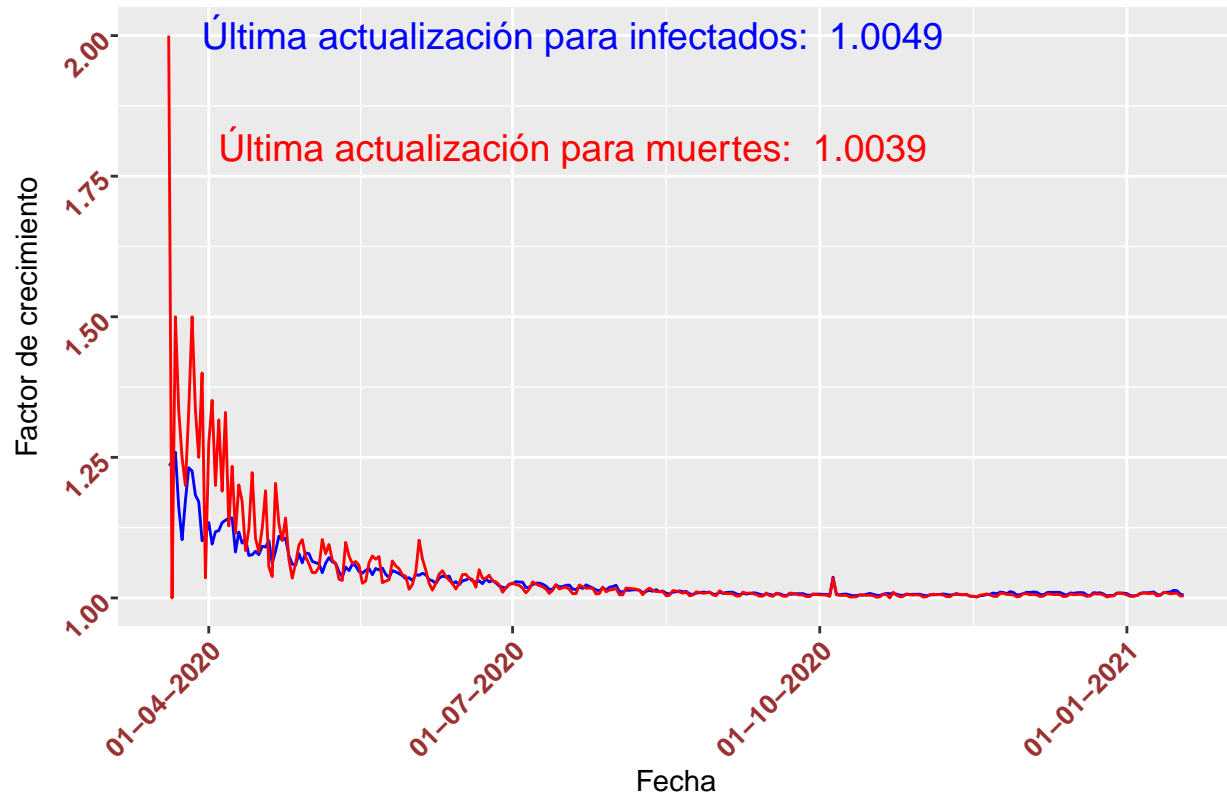
p <- p + scale_x_date(labels = date_format("%d-%m-%Y"))

###

p <- p +
  annotate("text", x = mex$Fecha[round(length(mex$Fecha)*0.4)],
           y = max(mex$FCM), colour = "blue", size = 5,
           label = paste("Última actualización para infectados: ",
                         round(mex$FCI[dim(mex)[1]], 4))) +
  annotate("text", x = mex$Fecha[round(length(mex$Fecha)*0.4)],
           y = max(mex$FCM)-0.2, colour = "red", size = 5,
           label = paste("Última actualización para muertes: ",
                         round(mex$FCM[dim(mex)[1]], 4)))
p

```

COVID-19 en México: 2021-01-24 13:55:40 CST



## EJEMPLO 6. SERIES DE TIEMPO Y DESCOMPOSICIÓN

### OBJETIVO

- Aprender a graficar y descomponer series de tiempo

### REQUISITOS

- Prework
- Gráficos: función plot

### DESARROLLO

Utilizaremos un dataset del paquete *datasets*

```
#EJEMPLO6.SERIES DE TIEMPO Y DESCOMPOSICIÓN
```

```
##UTILIZAREMOS UN DATASET DEL PAQUETE DATASETS
```

```
library(datasets)
```

```
(AP <- AirPassengers) # Número de pasajeros (en miles) por mes de una aerolínea
```

	Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
1949	112	118	132	129	121	135	148	148	136	119	104	118
1950	115	126	141	135	125	149	170	170	158	133	114	140
1951	145	150	178	163	172	178	199	199	184	162	146	166
1952	171	180	193	181	183	218	230	242	209	191	172	194
1953	196	196	236	235	229	243	264	272	237	211	180	201
1954	204	188	235	227	234	264	302	293	259	229	203	229
1955	242	233	267	269	270	315	364	347	312	274	237	278
1956	284	277	317	313	318	374	413	405	355	306	271	306
1957	315	301	356	348	355	422	465	467	404	347	305	336
1958	340	318	362	348	363	435	491	505	404	359	310	337
1959	360	342	406	396	420	472	548	559	463	407	362	405
1960	417	391	419	461	472	535	622	606	508	461	390	432

```
class(AP)
```

```
[1] "ts"
```

```
##Inicio, fin y frecuencia de la serie
```

```
start(AP); end(AP); frequency(AP)
```

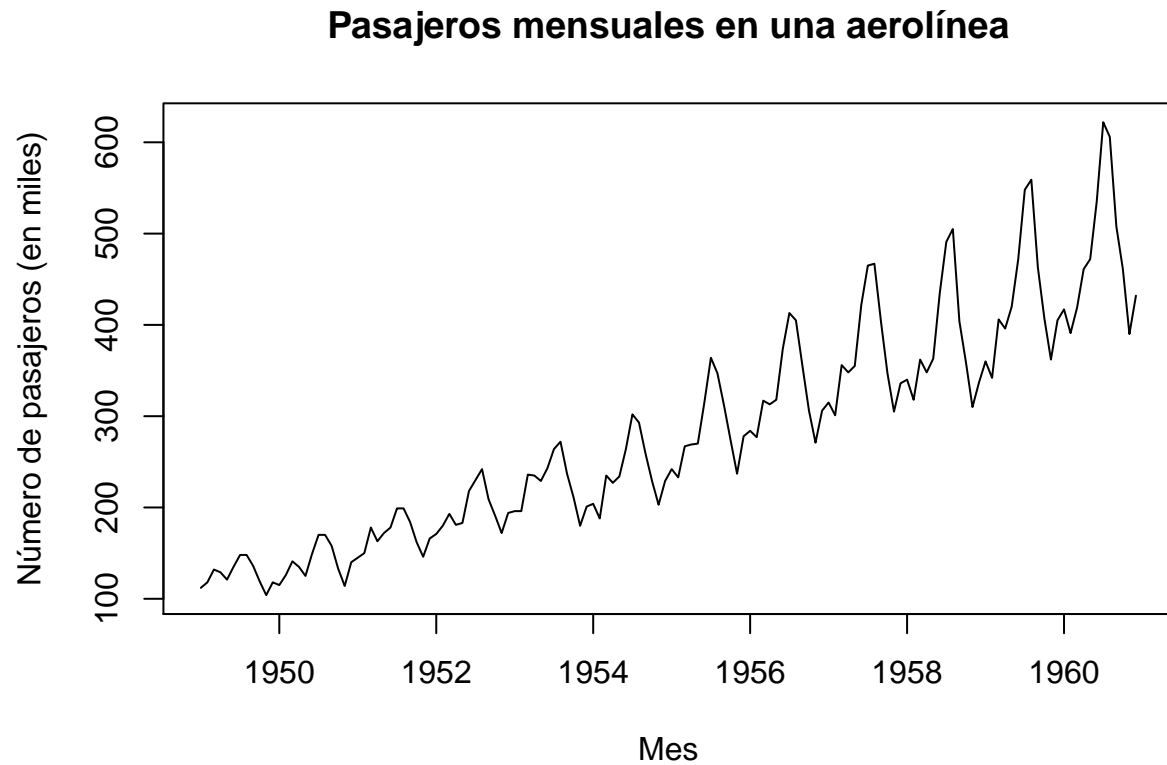
```
[1] 1949    1
```

```
[1] 1960   12
```

```
[1] 12
```

```
##Graficamos la serie de tiempo
```

```
plot(AP, main = "Pasajeros mensuales en una aerolínea", ylab = "Número de pasajeros (en miles)", xlab =
```



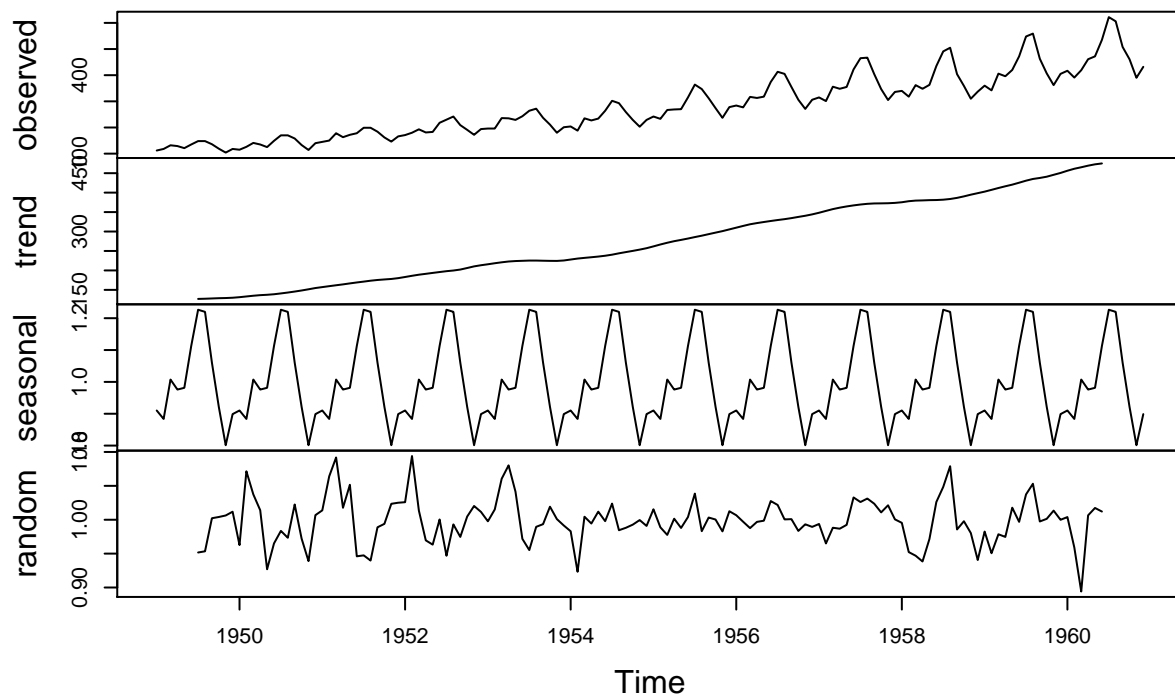
```
##DESCOMPOSICIÓN MULTIPLICATIVA
```

```
comp <- decompose(AP, type = "multiplicative")
```

```
##Graficamos la serie original, tendencia, estacionalidad, y componente aleatoria
```

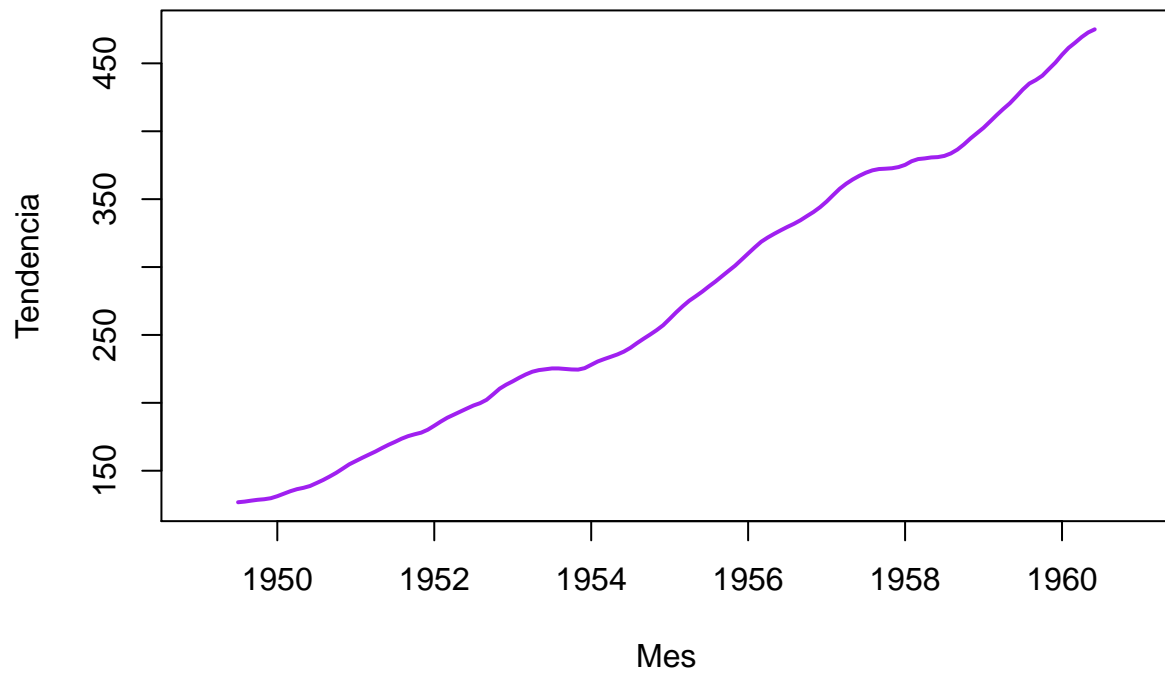
```
plot(comp)
```

## Decomposition of multiplicative time series



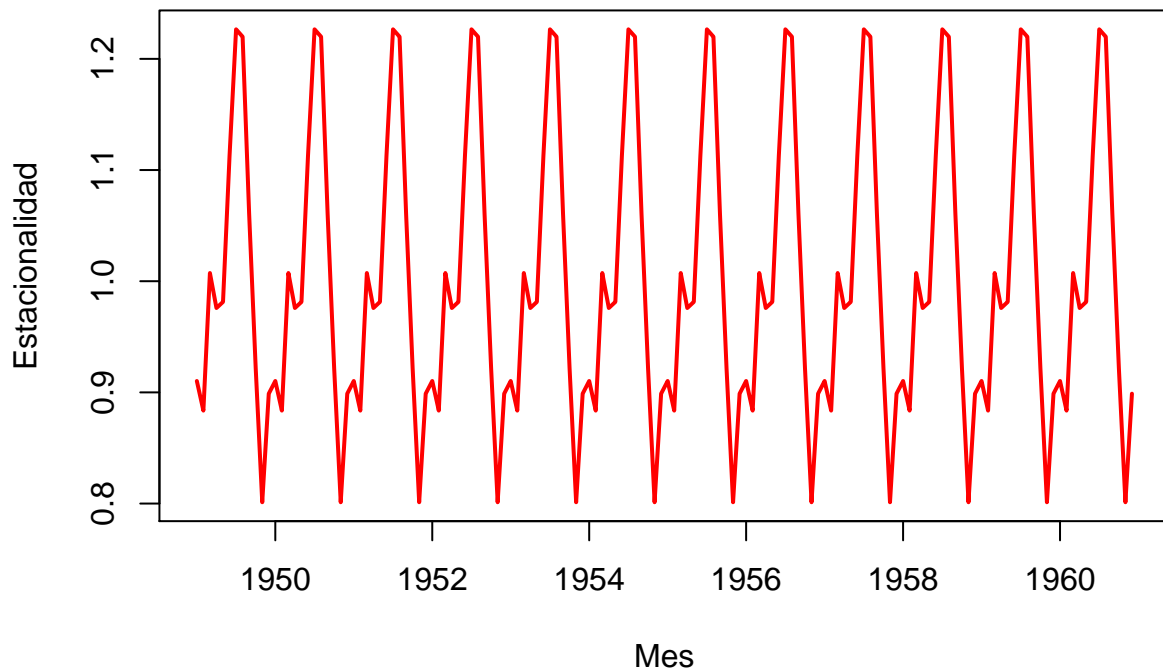
```
##Graficamos únicamente la serie de tendencia
plot(comp$trend, col = "purple", lwd = 2,
      main = "Componente de tendencia",
      ylab = "Tendencia", xlab = "Mes") # Gráfica de la tendencia
```

## Componente de tendencia



```
##Graficamos la componente estacional  
plot(comp$seasonal, col = "red", lwd = 2, main = "Componente de estacionalidad",  
      ylab = "Estacionalidad", xlab = "Mes") # Gráfica de la estacionalidad
```

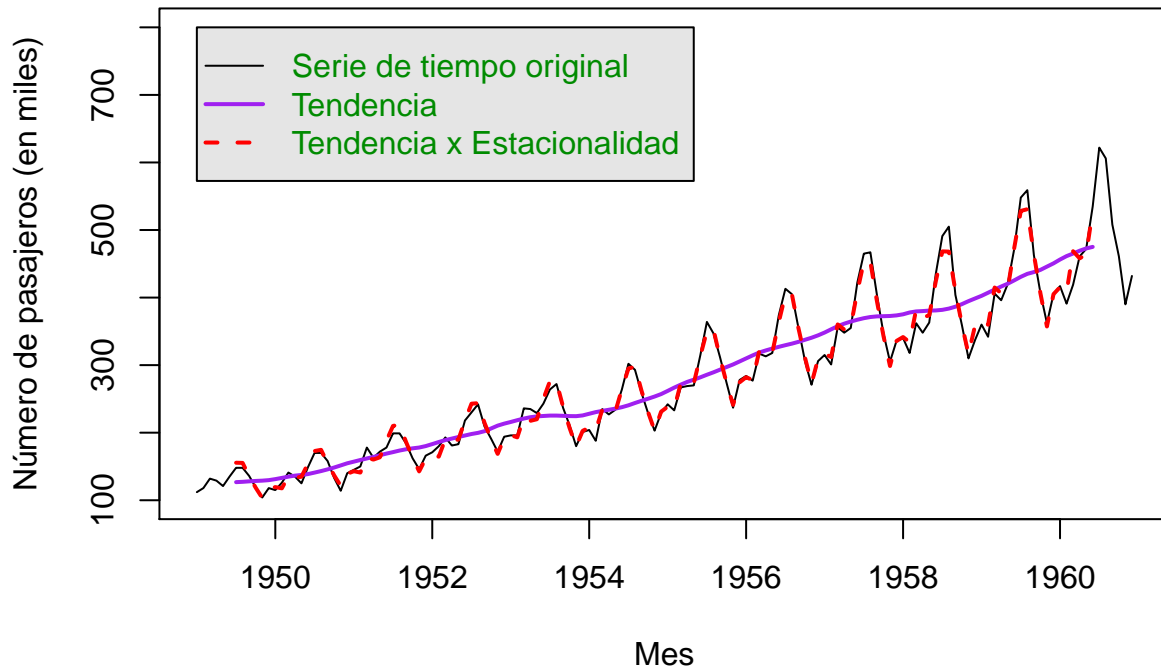
## Componente de estacionalidad



```
##Finalmente graficamos la serie original, junto con la tendencia y tendencia*estacionalidad
plot(AP, main = "Pasajeros mensuales en una aerolínea",
      ylab = "Número de pasajeros (en miles)",
      xlab = "Mes", ylim = c(100, 800))
lines(comp$trend , col = "purple", lwd = 2)
lines(comp$seasonal * comp$trend, col = "red", lty = 2, lwd = 2 )
legend(1949, 800,
       c('Serie de tiempo original', 'Tendencia',
         'Tendencia x Estacionalidad'),
       col = c('black', 'purple', 'red'),
       text.col = "green4", lty = c(1, 1, 2),
       lwd = c(1, 2, 2),
       merge = TRUE,
       bg = 'gray90')
```



## Pasajeros mensuales en una aerolínea



## RETO 3. SERIES DE TIEMPO

### OBJETIVO

- Poner en práctica los conocimientos de series de tiempo para poder manipularlas y así obtener información relevante.
- Implementar la descomposición multiplicativa

### REQUISITOS

- Haber llevado seguimiento de los ejercicios de clase
- Prework

### DESARROLLO

1. Importa los datos de producción de electricidad que se encuentran en el archivo *cbe.csv* a R .
2. Crea la serie de tiempo mensual de producción de electricidad en R a partir del año 1958.
3. Realiza la descomposición multiplicativa de la serie de tiempo y grafica la serie original junto con sus componentes (tendencia, estacionalidad y componente aleatoria).
4. Realiza la gráfica de tendencia y coloca la gráfica de tendencia por estacionalidad superpuesta a ésta.

```
#RETO 3
data<-read.csv("cbe.csv")
View(data)
class(data)
```

```
[1] "data.frame"
```

```
start(AP); end(AP); frequency(AP)
```

```
[1] 1949    1
```

```
[1] 1960   12
```

```
[1] 12
```

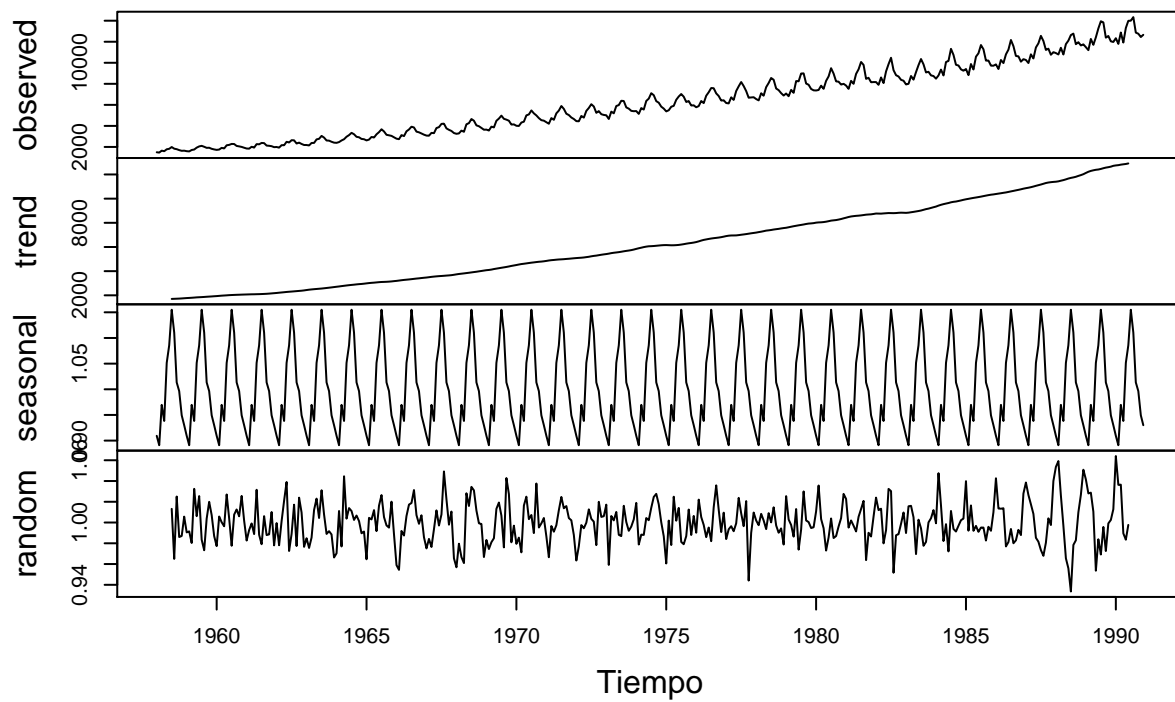
```
View(data)
datats <-ts(data[,3],start = 1958,frequency = 12)
#SE COLOCA EL ORIGEN DE DATOS Y EN LA TERCERA COLUMNA ES EL AÑO
#POR LO QUE SE SELECCIONA LA TERCER COLUMNA INICIANDO POR EL AÑO
#1958 Y COMO CADA AO TIENE 12 MESES SE COLOCA 12 EN FRECUENCIA
class(datats)
```

```
[1] "ts"
```

```
View(datats)

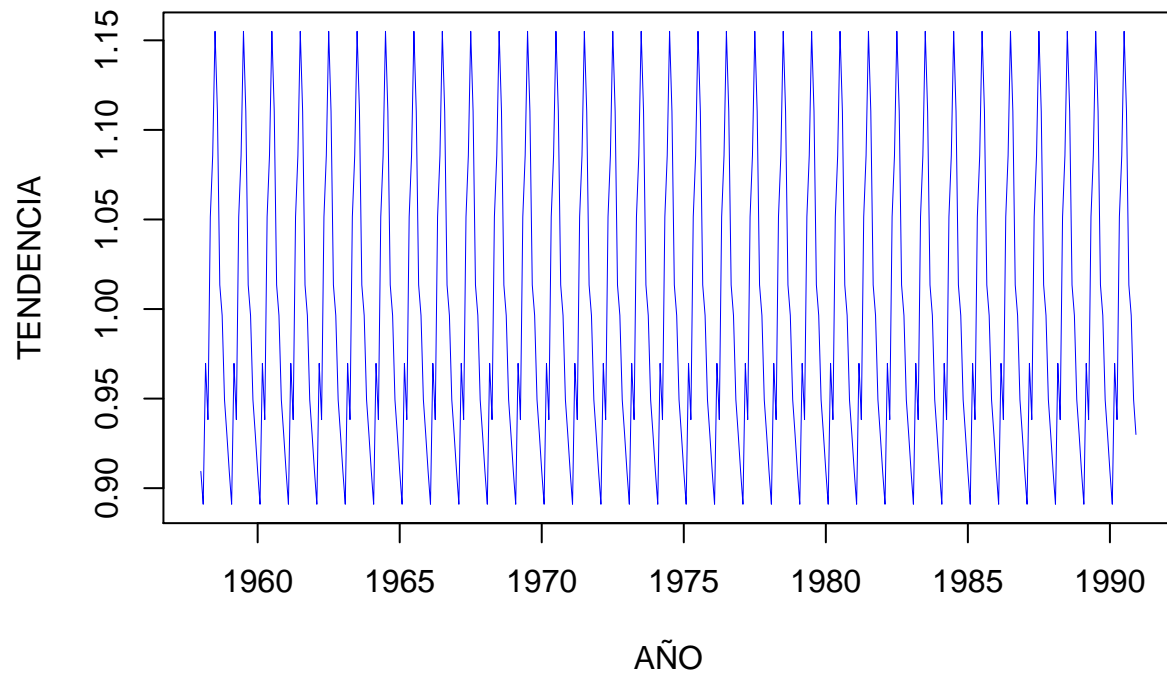
#REALIZANDO LA DESCOMPOSICIÓN MULTIPLICATIVA DE LA SERIE DE TIEMPO
datadecommulti <- decompose(datats,type = "mult" )
#GRAFICANDO LA DESCOMPOSICIÓN MULTIPLICATIVA
plot(datadecommulti,
     xlab="Tiempo",
     sub="Descomposición de los datos de producción de electricidad")
```

## Decomposition of multiplicative time series



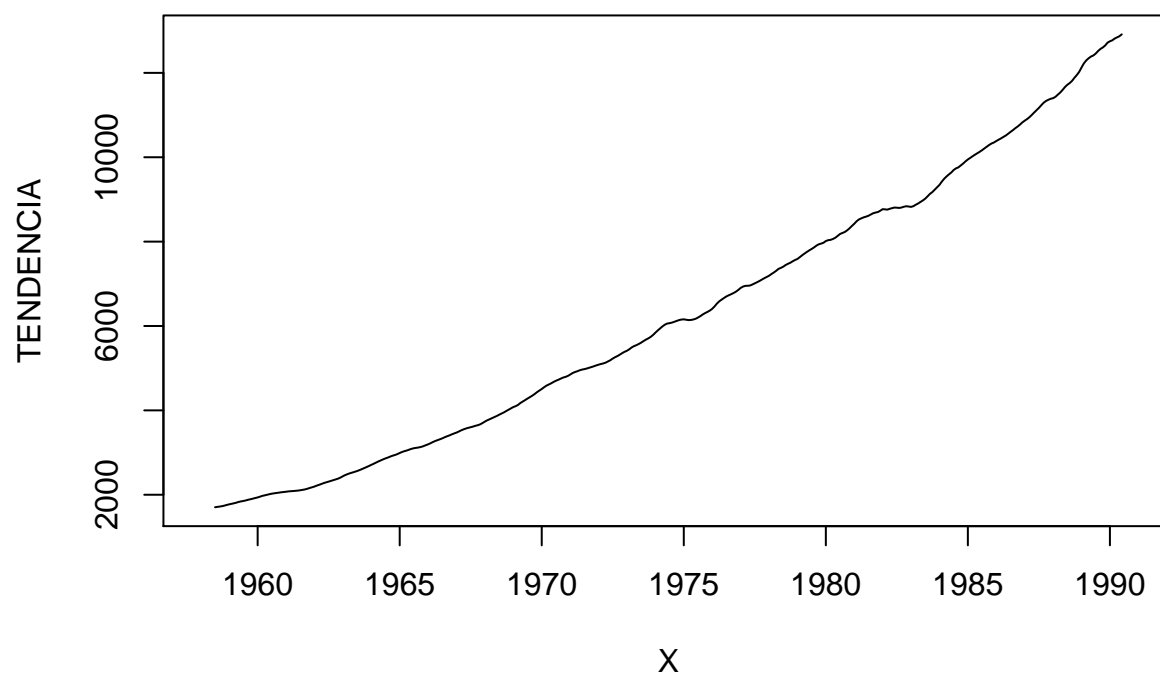
```
#GRAFICANDO COMPONENTES SEPARADOS
#GRAFICANDO COMPONENTE DE ESTACIONALIDAD
plot(datadecommulti$seasonal,col="blue",lwd=0.5,
     main="Componente de temporada",
     ylab="TENDENCIA",
     xlab="AÑO")
```

## Componente de temporada



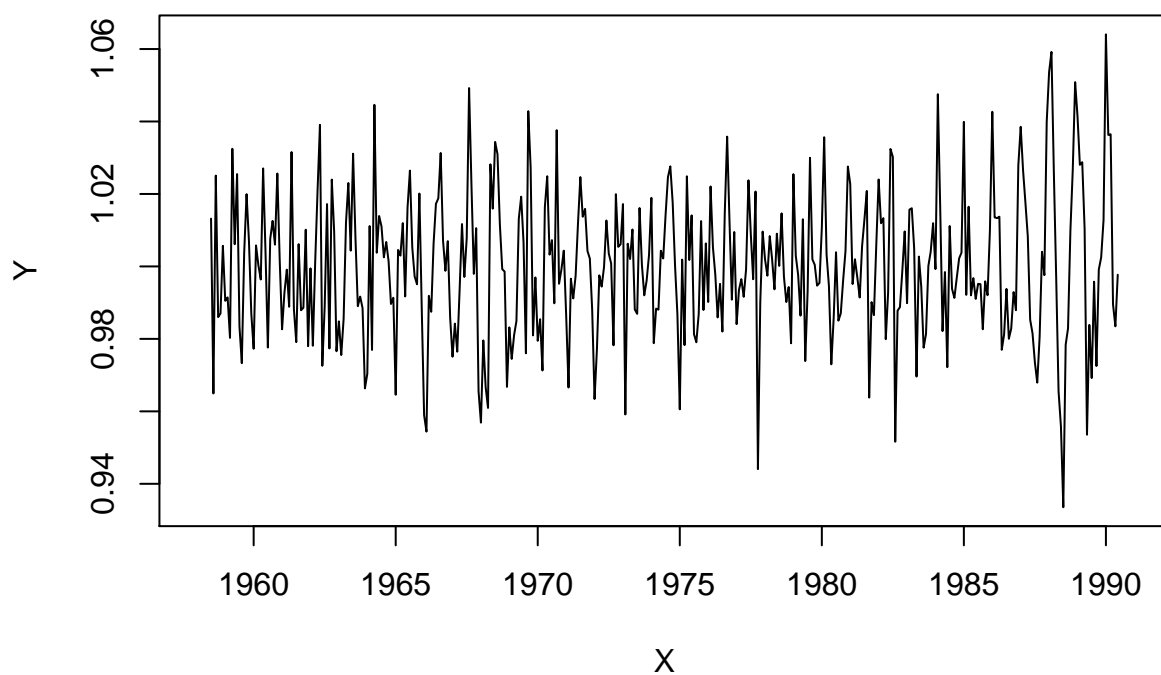
```
#GRAFICANDO COMPONENTE DE TENDENCIA
plot(datadecommulti$trend,col="black",lwd=1,
     main="Componente de tendencia",
     ylab="TENDENCIA",
     xlab="X")
```

## Componente de tendencia



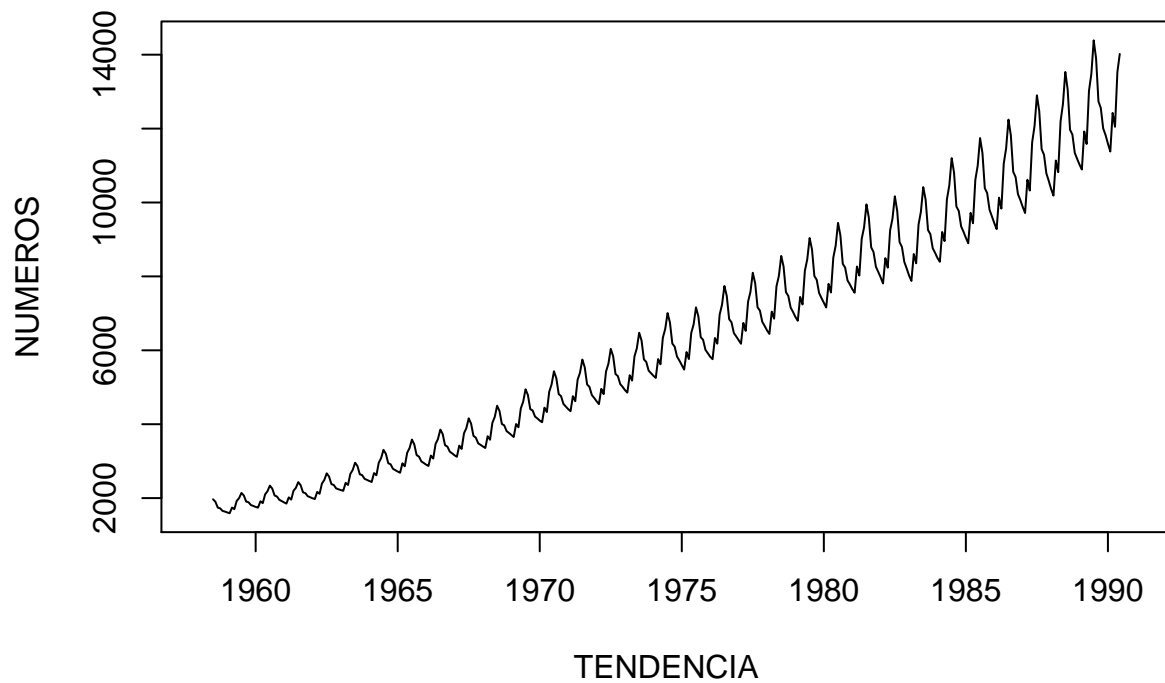
```
#GRAFICANDO LA COMPONENTE ALEATORIA  
plot(datadecommulti$random,col="black",lwd=1,  
      main="Componente aleatoria",  
      xlab="X",  
      ylab="Y")
```

## Componente aleatoria



```
#GRAFICANDO COMPONENTE DE TENDENCIA POR ESTACIONALIDAD  
plot(datadecommulti$trend*datadecommulti$seasonal,lwd=1,  
     main="TENDENCIA*ESTACIONALIDAD",xlab="TENDENCIA",  
     ylab="NUMEROS")
```

## TENDENCIA\*ESTACIONALIDAD



```
#SUPERPUESTOS COMPONENTES SUPERPUESTAS
```

```
Tendencia=datadecommulti$trend
```

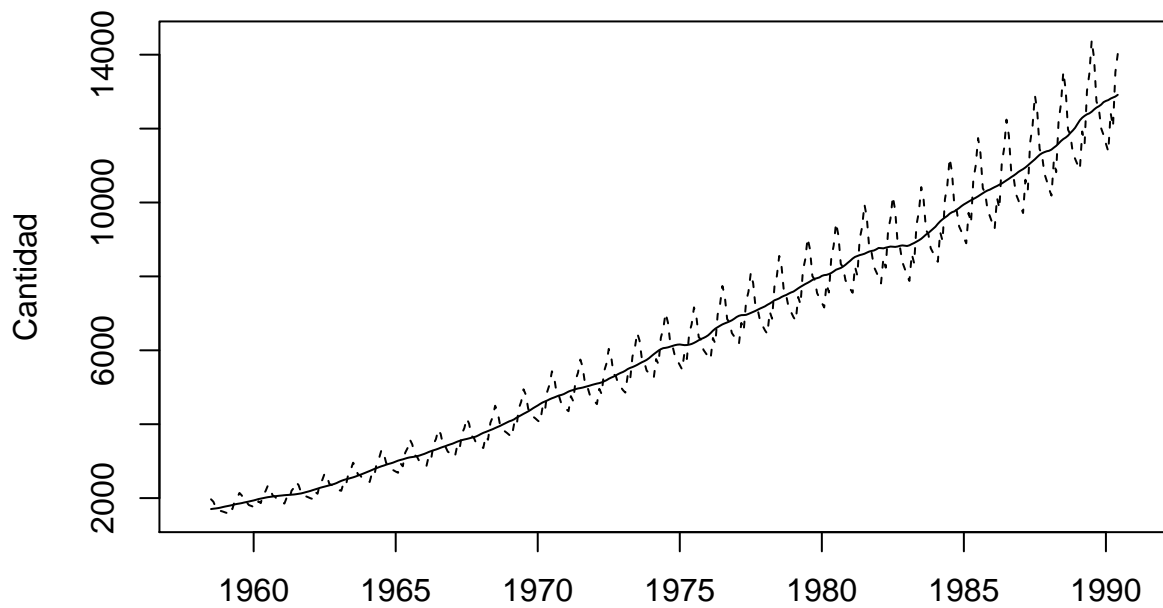
```
Seasonal=datadecommulti$seasonal
```

```
Random=datadecommulti$random
```

```
#GRAFICANDO
```

```
ts.plot(cbind(Tendencia,Tendencia*Seasonal),  
        main="Datos de producción de electricidad",  
        xlab="Tiempo",  
        ylab="Cantidad",  
        lty=1:2,  
        sub="Tendencia con efectos estacionales superpuestos")
```

## Datos de producción de electricidad



Tendencia con efectos estacionales superpuestos

```
#RECUERDA QUE lty INDICA EL TIPO DE LINEA QUE SE COLOCA  
#COMPROBANDO TENEMOS LO SIGUIENTE  
Tendencia[7]*Seasonal[7]*Random[7]
```

```
[1] 1994
```

```
datats[7]
```

```
[1] 1994
```

```
Tendencia[100]*Seasonal[100]*Random[100]
```

```
[1] 3033
```

```
datats[100]
```

```
[1] 3033
```

## POSTWORK SESION 3

### OBJETIVOS

- Realizar la descarga desde archivos de internet
- Generar nuevos dataframes
- Visualizar probabilidades estimadas con la ayuda de gráficas



## REQUISITOS

- R y RStudio
- Haber realizado el prework y seguir el curso de los ejemplos de la sesión
- Curiosidad por investigar nuevos tópicos y funciones de R

## DESARROLLO

Ahora graficaremos probabilidades (estimadas) marginales y conjuntas para el número de goles que anotan en un partido el equipo de casa o el equipo visitante.

1. Con el último data frame obtenido en el postwork de la sesión 2, elabora tablas de frecuencias relativas para estimar las siguientes probabilidades:

- La probabilidad (marginal) de que el equipo que juega en casa anote  $x$  goles ( $x=0,1,2,$ )
- La probabilidad (marginal) de que el equipo que juega como visitante anote  $y$  goles ( $y=0,1,2,$ )
- La probabilidad (conjunta) de que el equipo que juega en casa anote  $x$  goles y el equipo que juega como visitante anote  $y$  goles ( $x=0,1,2,$ ,  $y=0,1,2,$ )

2. Realiza lo siguiente:

- Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo de casa
- Un gráfico de barras para las probabilidades marginales estimadas del número de goles que anota el equipo visitante.
- Un HeatMap para las probabilidades conjuntas estimadas de los números de goles que anotan el equipo de casa y el equipo visitante en un partido.

```
#POSTWORK SESION 3 VICTOR MIGUEL TERRON MACIAS
# Ahora graficaremos probabilidades (estimadas) marginales y
# conjuntas para el número
# de goles que anotan en un partido el equipo de casa o el equipo visitante.

# 1. Con el último data frame obtenido en el postwork de la sesión 2,
#elabora tablas de frecuencias relativas para estimar las
# siguientes probabilidades:

# La probabilidad (marginal) de que el equipo que juega en casa anote x goles (x=0,1,2,)
# La probabilidad (marginal) de que el equipo que juega como visitante anote y goles (y=0,1,2,)
# La probabilidad (conjunta) de que el equipo que juega en casa anote x goles y el equipo que juega com

# 2. Realiza lo siguiente:
# * Un gráfico de barras para las probabilidades marginales estimadas del
# número de goles que anota el equipo de casa
# * Un gráfico de barras para las probabilidades marginales estimadas del
# número de goles que anota el equipo visitante.
# * Un HeatMap para las probabilidades conjuntas estimadas de los números
#de goles que anotan el equipo de casa y el equipo visitante en un partido.
```

```

#Lo primero que haremos es cargar los paquetes que usaremos más adelante.
# Usamos las funciones suppressWarnings
# y suppressMessages para que no se impriman mensajes ni
# advertencias al cargar el paquete.
suppressWarnings(suppressMessages(library(dplyr)))
suppressWarnings(suppressMessages(library(reshape2)))
suppressWarnings(suppressMessages(library(ggplot2)))

#Comenzamos importando los datos que se encuentran en archivos csv a R

url1718 <- "https://www.football-data.co.uk/mmz4281/1718/SP1.csv"
url1819 <- "https://www.football-data.co.uk/mmz4281/1819/SP1.csv"
url1920 <- "https://www.football-data.co.uk/mmz4281/1920/SP1.csv"
d1718 <- read.csv(file = url1718) # Importación de los datos a R
d1819 <- read.csv(file = url1819)
d1920 <- read.csv(file = url1920)

#Obtenemos una mejor idea de los datos que se encuentran
# en los data frames con las funciones str, head, View y summary

str(d1718); str(d1819); str(d1920)

```

```

'data.frame':  380 obs. of  64 variables:
 $ Div      : chr  "SP1" "SP1" "SP1" "SP1" ...
 $ Date     : chr  "18/08/17" "18/08/17" "19/08/17" "19/08/17" ...
 $ HomeTeam : chr  "Leganes" "Valencia" "Celta" "Girona" ...
 $ AwayTeam : chr  "Alaves" "Las Palmas" "Sociedad" "Ath Madrid" ...
 $ FTHG     : int   1 1 2 2 1 0 2 0 1 0 ...
 $ FTAG     : int   0 0 3 2 1 0 0 3 0 1 ...
 $ FTR      : chr   "H" "H" "A" "D" ...
 $ HTHG     : int   1 1 1 2 1 0 2 0 0 0 ...
 $ HTAG     : int   0 0 1 0 1 0 0 2 0 0 ...
 $ HTR      : chr   "H" "H" "D" "H" ...
 $ HS       : int  16 22 16 13 9 12 15 12 14 10 ...
 $ AS       : int   6 5 13 9 9 8 3 16 9 13 ...
 $ HST      : int   9 6 5 6 4 2 2 6 3 4 ...
 $ AST      : int   3 4 6 3 6 2 0 8 1 6 ...
 $ HF       : int  14 25 12 15 14 16 16 16 18 16 ...
 $ AF       : int  18 13 11 15 12 15 15 12 14 15 ...
 $ HC       : int   4 5 5 6 7 7 8 4 11 3 ...
 $ AC       : int   2 2 4 0 3 6 0 4 6 7 ...
 $ HY       : int   0 3 3 2 2 1 2 5 1 2 ...
 $ AY       : int   1 3 1 4 4 3 1 1 3 3 ...
 $ HR       : int   0 0 0 0 1 0 0 0 0 0 ...
 $ AR       : int   0 1 0 1 0 1 0 1 0 0 ...
 $ B365H    : num  2.05 1.75 2.38 8 1.62 1.5 1.17 9.5 3.25 2.1 ...
 $ B365D    : num  3.2 3.8 3.25 4.33 4 4 8 5.75 3.25 3.3 ...
 $ B365A    : num  4.1 4.5 3.2 1.45 5.5 7.5 15 1.3 2.3 3.7 ...
 $ BWH      : num  2.05 1.75 2.4 7.5 1.62 1.48 1.18 9.25 3.25 2.15 ...
 $ BWD      : num  3.1 3.9 3.3 4.33 3.9 4.25 7.5 5.75 3.2 3.3 ...
 $ BWA      : num  4.1 4.6 3 1.45 5.75 7 14.5 1.3 2.3 3.5 ...
 $ IWH      : num  2.1 1.75 2.5 7.2 1.55 1.5 1.17 7.5 3.3 2.1 ...
 $ IWD      : num  3.4 3.6 3.3 4.4 4 4.2 7.5 5.5 3.35 3.4 ...

```

```

$ IWA      : num  3.5 4.8 2.85 1.45 6.2 6.5 15 1.35 2.2 3.5 ...
$ LBH      : num  2.05 1.75 2.35 7.5 1.6 1.5 1.2 9.5 3.25 2.1 ...
$ LBD      : num  3 3.8 3.25 4 3.9 4 6.5 5.25 3.1 3.1 ...
$ LBA      : num  4.2 4.33 3 1.5 5.5 7 15 1.3 2.3 3.4 ...
$ PSH      : num  2.03 1.78 2.44 8.36 1.62 ...
$ PSD      : num  3.25 4.01 3.4 4.38 4.17 4.37 7.35 5.79 3.24 3.36 ...
$ PSA      : num  4.52 4.83 3.16 1.49 6.18 7.31 15.5 1.33 2.36 3.49 ...
$ WHH      : num  2.05 1.8 2.4 8 1.67 1.5 1.22 11 3.1 2.2 ...
$ WHD      : num  3.1 3.75 3.4 4.2 3.6 4 6 4.5 3.1 3.3 ...
$ WHA      : num  4 4.2 2.9 1.44 5.5 7 13 1.33 2.4 3.3 ...
$ VCH      : num  2.05 1.8 2.4 7.5 1.65 1.5 1.2 9.5 3.25 2.15 ...
$ VCD      : num  3.2 4 3.4 4.3 4 4.2 7 5.75 3.25 3.3 ...
$ VCA      : num  4.4 4.6 3.13 1.5 5.75 7 13 1.3 2.3 3.5 ...
$ Bb1X2    : int  35 35 35 35 35 34 35 35 34 34 ...
$ BbMxH    : num  2.12 1.83 2.5 8.36 1.69 ...
$ BbAvH    : num  2.03 1.77 2.39 7.53 1.63 1.5 1.19 9.68 3.26 2.18 ...
$ BbMxD    : num  3.4 4.04 3.5 4.4 4.17 4.4 8 5.86 3.35 3.4 ...
$ BbAvD    : num  3.15 3.86 3.32 4.17 3.93 4.17 7.11 5.44 3.17 3.26 ...
$ BbMxA    : num  4.52 4.83 3.2 1.51 6.2 7.5 17 1.35 2.4 3.7 ...
$ BbAvA    : num  4.17 4.46 3.01 1.48 5.58 ...
$ BbOU     : int  31 33 34 34 33 32 27 27 32 32 ...
$ BbMx.2.5 : num  2.84 1.69 2.03 2.2 1.81 2.01 1.44 1.5 2.42 2.25 ...
$ BbAv.2.5 : num  2.68 1.64 1.98 2.11 1.75 1.94 1.4 1.46 2.36 2.14 ...
$ BbMx.2.5.1 : num  1.53 2.4 1.9 1.8 2.14 1.96 3.1 2.95 1.63 1.76 ...
$ BbAv.2.5.1 : num  1.46 2.27 1.84 1.74 2.09 1.87 2.88 2.64 1.58 1.7 ...
$ BbAH     : int  18 16 18 16 16 17 17 16 15 17 ...
$ BbAHh    : num  -0.5 -0.75 -0.25 1.25 -1 -1 -2 1.5 0.25 -0.25 ...
$ BbMxAHH  : num  2.07 2.05 2.08 1.77 2.12 1.9 2.05 2.03 1.93 1.92 ...
$ BbAvAHH  : num  2.03 1.97 2.05 1.75 2.06 1.86 2 1.98 1.89 1.88 ...
$ BbMxAHA  : num  1.9 1.96 1.87 2.25 1.86 2.05 1.91 1.95 2.03 2.04 ...
$ BbAvAHA  : num  1.86 1.91 1.83 2.16 1.82 2.01 1.86 1.89 1.98 1.99 ...
$ PSCH     : num  1.98 1.78 2.12 6.93 1.64 1.53 1.2 12.4 3.31 2.2 ...
$ PSCD     : num  3.35 4.24 3.53 3.83 4.18 4.48 8.25 7 3.32 3.27 ...
$ PSCA     : num  4.63 4.43 3.74 1.63 5.82 6.91 15.2 1.26 2.4 3.85 ...

```

```

'data.frame': 380 obs. of 61 variables:

```

```

$ Div      : chr  "SP1" "SP1" "SP1" "SP1" ...
$ Date     : chr  "17/08/2018" "17/08/2018" "18/08/2018" "18/08/2018" ...
$ HomeTeam : chr  "Betis" "Girona" "Barcelona" "Celta" ...
$ AwayTeam : chr  "Levante" "Valladolid" "Alaves" "Espanol" ...
$ FTHG     : int  0 0 3 1 1 1 2 1 2 1 ...
$ FTAG     : int  3 0 0 1 2 2 0 4 1 1 ...
$ FTR      : chr  "A" "D" "H" "D" ...
$ HTHG     : int  0 0 0 0 1 0 1 0 1 0 ...
$ HTAG     : int  1 0 0 1 1 2 0 3 1 1 ...
$ HTR      : chr  "A" "D" "D" "A" ...
$ HS       : int  22 13 25 12 16 18 10 13 17 13 ...
$ AS       : int  6 2 3 14 8 8 4 17 12 9 ...
$ HST      : int  8 1 9 2 7 6 3 2 5 4 ...
$ AST      : int  4 1 0 5 4 6 1 8 2 3 ...
$ HF       : int  10 21 6 13 16 12 11 6 12 10 ...
$ AF       : int  10 20 13 14 10 13 27 15 13 15 ...
$ HC       : int  5 3 7 8 4 7 3 2 6 4 ...
$ AC       : int  3 2 1 7 6 0 0 6 2 10 ...

```

```

$ HY      : int  0 1 0 3 2 1 1 1 4 2 ...
$ AY      : int  2 1 2 2 3 1 7 0 5 3 ...
$ HR      : int  0 0 0 0 0 0 0 0 0 0 ...
$ AR      : int  0 0 0 0 0 0 0 0 0 0 ...
$ B365H   : num  1.66 1.75 1.11 1.85 2.04 1.66 1.2 3.25 1.75 3 ...
$ B365D   : num  4 3.6 10 3.5 3.4 3.75 7 3.6 3.3 3.2 ...
$ B365A   : num  5 5 21 4.5 3.8 5.5 13 2.14 5.5 2.5 ...
$ BWH     : num  1.7 1.75 1.11 1.91 2.05 1.7 1.18 3.5 1.78 2.85 ...
$ BWD     : num  3.7 3.5 10 3.4 3.3 3.7 7.25 3.5 3.5 3.25 ...
$ BWA     : num  5.25 5.25 20 4.25 3.9 5.25 16 2.1 5 2.55 ...
$ IWH     : num  1.75 1.8 1.12 1.9 2 1.7 1.2 3.5 1.85 2.85 ...
$ IWD     : num  3.6 3.6 9 3.5 3.4 3.75 6.5 3.4 3.5 3.2 ...
$ IWA     : num  4.9 4.5 20 4.1 3.8 5 15 2.1 4.4 2.55 ...
$ PSH     : num  1.69 1.8 1.11 1.93 2.06 1.72 1.2 3.46 1.79 3.12 ...
$ PSD     : num  4.19 3.7 11.27 3.64 3.51 ...
$ PSA     : num  5.11 4.99 25.4 4.27 3.91 ...
$ WHH     : num  1.67 1.75 1.08 1.91 2.05 1.73 1.22 3.3 1.8 3 ...
$ WHD     : num  3.9 3.6 9 3.5 3.3 3.6 6 3.7 3.4 3.2 ...
$ WHA     : num  4.75 4.6 29 4 3.6 4.75 13 2.05 4.75 2.4 ...
$ VCH     : num  1.67 1.8 1.1 1.93 2.05 1.7 1.2 3.4 1.8 3 ...
$ VCD     : num  4.2 3.7 10.5 3.5 3.5 3.8 7 3.6 3.4 3.2 ...
$ VCA     : num  5.2 4.8 34 4.4 3.9 5 13 2.1 5 2.45 ...
$ Bb1X2   : int  40 40 40 38 40 40 39 40 40 39 ...
$ BbMxH   : num  1.75 1.85 1.13 1.97 2.11 1.76 1.24 3.53 1.85 3.12 ...
$ BbAvH   : num  1.68 1.78 1.1 1.9 2.03 1.7 1.21 3.38 1.78 2.99 ...
$ BbMxD   : num  4.25 3.83 11.5 3.73 3.62 3.93 7.36 3.75 3.64 3.29 ...
$ BbAvD   : num  4 3.6 9.82 3.53 3.43 3.77 6.66 3.56 3.43 3.14 ...
$ BbMxA   : num  5.25 5.27 41 4.5 3.93 ...
$ BbAvA   : num  4.95 4.79 25.67 4.2 3.76 ...
$ BbOU    : int  38 38 32 36 37 37 33 37 36 36 ...
$ BbMx.2.5 : num  1.82 2.21 1.39 2.13 2.05 1.95 1.5 1.83 2.49 2.45 ...
$ BbAv.2.5 : num  1.76 2.13 1.34 2.06 1.99 1.88 1.45 1.76 2.35 2.33 ...
$ BbMx.2.5.1 : num  2.15 1.78 3.4 1.84 1.88 1.98 2.75 2.13 1.64 1.65 ...
$ BbAv.2.5.1 : num  2.06 1.71 3.18 1.76 1.81 1.91 2.66 2.04 1.58 1.59 ...
$ BbAH    : int  20 20 19 18 18 19 19 19 18 17 ...
$ BbAHh   : num  -0.75 -0.75 -2.5 -0.75 -0.25 -0.75 -1.75 0.25 -0.75 0.25 ...
$ BbMxAHH : num  1.89 2.06 1.95 2.26 1.76 1.96 1.85 2.08 2.11 1.82 ...
$ BbAvAHH : num  1.85 2.01 1.91 2.18 1.74 1.91 1.8 2.03 2.04 1.75 ...
$ BbMxAHA : num  2.07 1.9 2 1.74 2.23 2.01 2.15 1.86 1.86 2.23 ...
$ BbAvAHA : num  2 1.85 1.95 1.71 2.14 1.94 2.07 1.83 1.82 2.12 ...
$ PSCH    : num  1.59 1.76 1.1 2.18 2.32 1.77 1.19 4.57 1.69 3.55 ...
$ PSCD    : num  4.42 3.57 11.85 3.26 3.21 ...
$ PSCA    : num  5.89 5.62 32.17 3.85 3.53 ...

'data.frame':  380 obs. of  105 variables:
$ Div      : chr  "SP1" "SP1" "SP1" "SP1" ...
$ Date     : chr  "16/08/2019" "17/08/2019" "17/08/2019" "17/08/2019" ...
$ Time     : chr  "20:00" "16:00" "18:00" "19:00" ...
$ HomeTeam : chr  "Ath Bilbao" "Celta" "Valencia" "Mallorca" ...
$ AwayTeam : chr  "Barcelona" "Real Madrid" "Sociedad" "Eibar" ...
$ FTHG     : int  1 1 1 2 0 4 1 0 1 1 ...
$ FTAG     : int  0 3 1 1 1 4 0 2 2 0 ...
$ FTR      : chr  "H" "A" "D" "H" ...
$ HTHG     : int  0 0 0 1 0 1 0 0 0 1 ...

```

```

$ HTAG      : int  0 1 0 0 0 1 0 1 0 0 ...
$ HTR       : chr  "D" "A" "D" "H" ...
$ HS        : int  11 7 14 16 13 12 9 7 13 5 ...
$ AS        : int  11 17 12 11 4 14 16 12 14 6 ...
$ HST       : int  5 4 6 4 2 7 2 2 4 5 ...
$ AST       : int  2 11 3 5 2 7 4 4 3 0 ...
$ HF        : int  14 17 13 13 17 10 18 11 11 19 ...
$ AF        : int  9 12 14 14 11 16 15 17 19 22 ...
$ HC        : int  3 6 3 9 8 2 2 8 6 3 ...
$ AC        : int  8 4 3 3 0 7 9 4 1 4 ...
$ HY        : int  1 5 4 2 1 3 2 2 2 3 ...
$ AY        : int  1 2 4 3 4 1 1 2 6 4 ...
$ HR        : int  0 0 1 0 1 0 0 0 1 1 ...
$ AR        : int  0 1 0 0 0 0 0 0 0 1 ...
$ B365H     : num  5.25 4.75 1.66 2.8 2 1.6 2.15 3.2 1.66 1.44 ...
$ B365D     : num  3.8 4.2 3.75 3.2 3.2 3.8 3.2 3.3 3.75 4.33 ...
$ B365A     : num  1.65 1.65 5.5 2.6 4.2 6.5 3.6 2.3 5.5 8 ...
$ BWH       : num  5.5 4.4 1.67 2.95 2.05 1.6 2.15 3.1 1.65 1.45 ...
$ BWD       : num  3.8 4.2 3.75 3.1 3.25 3.8 3.3 3.4 3.75 4.33 ...
$ BWA       : num  1.65 1.72 5.5 2.6 3.9 6.25 3.6 2.3 5.75 7.5 ...
$ IWH       : num  5 5.3 1.67 2.9 2.05 1.63 2.2 3.1 1.63 1.45 ...
$ IWD       : num  3.8 4.2 3.75 3.1 3.1 4 3.25 3.4 3.75 4.4 ...
$ IWA       : num  1.7 1.6 5.3 2.6 4.05 5.5 3.4 2.3 5.7 7.2 ...
$ PSH       : num  5.15 4.73 1.68 2.98 2.1 1.62 2.29 3.13 1.63 1.49 ...
$ PSD       : num  3.84 4.18 3.94 3.14 3.21 3.99 3.31 3.56 3.81 4.34 ...
$ PSA       : num  1.74 1.72 5.47 2.66 4.13 6.13 3.45 2.33 6.38 7.58 ...
$ WHH       : num  5 5.25 1.67 2.9 2.05 1.6 2.25 3 1.62 1.47 ...
$ WHD       : num  3.8 4.2 3.8 3.1 3.2 3.9 3.3 3.5 3.75 4.2 ...
$ WHA       : num  1.7 1.6 5.25 2.62 4 5.8 3.3 2.3 6 8 ...
$ VCH       : num  5 4.75 1.67 2.9 2.1 1.65 2.25 3 1.62 1.45 ...
$ VCD       : num  3.8 4.2 3.9 3.13 3.2 4 3.3 3.5 3.8 4.2 ...
$ VCA       : num  1.75 1.73 5.75 2.7 4.1 5.75 3.3 2.3 5.75 8 ...
$ MaxH      : num  5.5 5.3 1.72 3.05 2.1 1.65 2.31 3.2 1.67 1.52 ...
$ MaxD      : num  3.95 4.4 3.98 3.2 3.3 4.15 3.4 3.56 3.9 4.5 ...
$ MaxA      : num  1.76 1.73 5.75 2.7 4.25 6.5 3.6 2.4 6.5 8.5 ...
$ AvgH      : num  5.07 4.67 1.68 2.91 2.06 1.61 2.23 3.08 1.64 1.47 ...
$ AvgD      : num  3.81 4.12 3.8 3.09 3.18 3.95 3.25 3.41 3.76 4.23 ...
$ AvgA      : num  1.71 1.69 5.29 2.62 4.02 5.8 3.43 2.33 5.78 7.63 ...
$ B365.2.5  : num  1.8 1.53 2 2.3 2.5 1.8 2.1 1.9 2.1 2.2 ...
$ B365.2.5.1 : num  2 2.5 1.8 1.61 1.53 2 1.72 1.9 1.72 1.66 ...
$ P.2.5     : num  1.81 1.52 2.08 2.45 2.72 1.88 2.16 1.95 2.16 2.3 ...
$ P.2.5.1   : num  2.09 2.66 1.82 1.6 1.5 2.02 1.76 1.95 1.76 1.68 ...
$ Max.2.5   : num  1.85 1.53 2.14 2.47 2.75 1.9 2.2 1.98 2.21 2.3 ...
$ Max.2.5.1 : num  2.11 2.72 1.83 1.65 1.54 2.05 1.77 1.95 1.78 1.71 ...
$ Avg.2.5   : num  1.79 1.49 2.07 2.34 2.59 1.84 2.13 1.92 2.13 2.23 ...
$ Avg.2.5.1 : num  2.05 2.58 1.77 1.6 1.49 1.98 1.72 1.89 1.72 1.66 ...
$ AHh       : num  0.75 0.75 -0.75 0 -0.5 -1 -0.25 0.25 -0.75 -1 ...
$ B365AHH   : num  1.99 2.04 1.91 2.05 2.08 2.05 1.95 1.88 1.86 1.88 ...
$ B365AHA   : num  1.94 1.89 2.02 1.88 1.85 1.75 1.98 2.05 2.07 2.05 ...
$ PAHH      : num  1.98 2.01 1.91 2.07 2.1 2.11 1.96 1.9 1.84 1.88 ...
$ PAHA      : num  1.94 1.91 2.01 1.85 1.82 1.81 1.96 2.02 2.08 2.04 ...
$ MaxAHH    : num  2 2.05 1.93 2.07 2.1 2.14 1.97 1.9 1.87 1.89 ...
$ MaxAHA    : num  1.95 1.91 2.03 1.88 1.85 1.85 1.99 2.06 2.08 2.08 ...
$ AvgAHH    : num  1.96 2 1.89 2.04 2.06 2.07 1.93 1.87 1.83 1.85 ...

```

```

$ AvgAHA      : num  1.92 1.88 1.99 1.85 1.83 1.8 1.95 2.01 2.06 2.03 ...
$ B365CH      : num  5.25 5.25 1.66 2.87 1.9 1.53 2.3 3 1.8 1.5 ...
$ B365CD      : num  3.8 4.2 3.75 3.2 3.1 4 3.4 3.4 3.6 4 ...
$ B365CA      : num  1.65 1.57 5.5 2.55 5 6.5 3.2 2.4 4.75 8 ...
$ BWCH        : num  4.75 4.5 1.65 2.95 1.95 1.57 2.35 3 1.8 1.5 ...
$ BWCD        : num  3.75 4.1 3.8 3.1 3.2 3.8 3.2 3.4 3.4 3.9 ...
$ BWCA        : num  1.75 1.7 5.5 2.6 4.5 6.5 3.2 2.35 5 7.75 ...
$ IWCH        : num  5 4.6 1.67 2.9 1.9 1.55 2.35 3 1.85 1.5 ...
$ IWCD        : num  3.8 3.8 3.8 3.1 3.15 4.05 3.25 3.35 3.55 3.9 ...
$ IWCA        : num  1.7 1.75 5.3 2.6 4.85 6.3 3.15 2.35 4.4 7.6 ...
$ PSCH        : num  5.34 5.1 1.69 2.96 1.9 1.54 2.43 3.13 1.82 1.57 ...
$ PSCD        : num  3.62 4.46 3.88 3.26 3.18 4.19 3.27 3.38 3.53 3.78 ...
$ PSCA        : num  1.78 1.65 5.47 2.6 5.3 6.87 3.2 2.41 5.07 7.66 ...
$ WHCH        : num  5 5 1.65 2.9 2.05 1.62 2.25 3 1.78 1.5 ...
$ WHCD        : num  3.8 4.2 3.9 3.1 3.2 3.9 3.3 3.4 3.5 3.8 ...
$ WHCA        : num  1.7 1.63 5.25 2.6 4 5.8 3.3 2.35 5 8 ...
$ VCCH        : num  4.8 5.2 1.7 3 1.9 1.57 2.45 3.13 1.87 1.55 ...
$ VCCD        : num  3.8 4.4 3.9 3.13 3.2 4 3.3 3.4 3.5 3.9 ...
$ VCCA        : num  1.8 1.65 5.5 2.63 5.2 7 3.13 2.4 4.6 8 ...
$ MaxCH       : num  5.8 6 1.72 3.05 1.95 1.58 2.46 3.38 1.87 1.58 ...
$ MaxCD       : num  3.9 4.52 3.95 3.29 3.26 4.2 3.42 3.47 3.65 4.05 ...
$ MaxCA       : num  1.81 1.75 6.2 2.72 5.3 7.3 3.58 2.48 5.35 8.9 ...
$ AvgCH       : num  5.03 4.93 1.68 2.93 1.9 1.54 2.37 3.05 1.83 1.53 ...
$ AvgCD       : num  3.66 4.26 3.82 3.14 3.16 4.05 3.25 3.34 3.5 3.84 ...
$ AvgCA       : num  1.76 1.65 5.37 2.59 4.91 6.66 3.18 2.39 4.74 7.68 ...
$ B365C.2.5   : num  1.9 1.44 2 2.2 2.75 1.9 2.1 2 2 2.37 ...
$ B365C.2.5.1 : num  1.9 2.75 1.8 1.66 1.44 1.9 1.72 1.8 1.8 1.57 ...
$ PC.2.5      : num  1.98 1.49 2.06 2.2 2.84 1.95 2.18 2.04 2.03 2.43 ...
$ PC.2.5.1    : num  1.93 2.76 1.85 1.74 1.47 1.95 1.75 1.85 1.87 1.61 ...
$ MaxC.2.5    : num  1.99 1.51 2.08 2.38 2.85 1.98 2.18 2.09 2.07 2.46 ...
$ MaxC.2.5.1  : num  2.11 2.88 1.98 1.74 1.5 2.1 1.83 2.05 1.92 1.65 ...
$ AvgC.2.5    : num  1.86 1.47 2 2.24 2.69 1.9 2.1 1.97 1.99 2.36 ...
$ AvgC.2.5.1  : num  1.97 2.63 1.82 1.66 1.46 1.92 1.74 1.85 1.83 1.59 ...
$ AHCh        : num  0.75 1 -0.75 0 -0.5 -1 -0.25 0.25 -0.75 -1 ...
$ B365CAHH    : num  1.93 1.82 1.94 2.11 1.89 1.96 2.08 1.86 2.02 2.06 ...
$ B365CAHA    : num  2 1.97 1.99 1.82 2.04 1.97 1.85 2.07 1.77 1.87 ...
[list output truncated]

```

```
head(d1718); head(d1819); head(d1920)
```

	Div	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST	AST				
1	SP1	18/08/17	Leganes	Alaves	1	0	H	1	0	H	16	6	9	3				
2	SP1	18/08/17	Valencia	Las Palmas	1	0	H	1	0	H	22	5	6	4				
3	SP1	19/08/17	Celta	Sociedad	2	3	A	1	1	D	16	13	5	6				
4	SP1	19/08/17	Girona	Ath Madrid	2	2	D	2	0	H	13	9	6	3				
5	SP1	19/08/17	Sevilla	Espanol	1	1	D	1	1	D	9	9	4	6				
6	SP1	20/08/17	Ath Bilbao	Getafe	0	0	D	0	0	D	12	8	2	2				
	HF	AF	HC	AC	HY	AY	HR	AR	B365H	B365D	B365A	BWH	BWD	BWA	IWH	IWD	IWA	LBH
1	14	18	4	2	0	1	0	0	2.05	3.20	4.10	2.05	3.10	4.10	2.10	3.4	3.50	2.05
2	25	13	5	2	3	3	0	1	1.75	3.80	4.50	1.75	3.90	4.60	1.75	3.6	4.80	1.75
3	12	11	5	4	3	1	0	0	2.38	3.25	3.20	2.40	3.30	3.00	2.50	3.3	2.85	2.35
4	15	15	6	0	2	4	0	1	8.00	4.33	1.45	7.50	4.33	1.45	7.20	4.4	1.45	7.50
5	14	12	7	3	2	4	1	0	1.62	4.00	5.50	1.62	3.90	5.75	1.55	4.0	6.20	1.60
6	16	15	7	6	1	3	0	1	1.50	4.00	7.50	1.48	4.25	7.00	1.50	4.2	6.50	1.50

	LBD	LBA	PSH	PSD	PSA	WHH	WHD	WHA	VCH	VCD	VCA	Bb1X2	BbMxH	BbAvH	BbMxD
1	3.00	4.20	2.03	3.25	4.52	2.05	3.10	4.00	2.05	3.2	4.40	35	2.12	2.03	3.40
2	3.80	4.33	1.78	4.01	4.83	1.80	3.75	4.20	1.80	4.0	4.60	35	1.83	1.77	4.04
3	3.25	3.00	2.44	3.40	3.16	2.40	3.40	2.90	2.40	3.4	3.13	35	2.50	2.39	3.50
4	4.00	1.50	8.36	4.38	1.49	8.00	4.20	1.44	7.50	4.3	1.50	35	8.36	7.53	4.40
5	3.90	5.50	1.62	4.17	6.18	1.67	3.60	5.50	1.65	4.0	5.75	35	1.69	1.63	4.17
6	4.00	7.00	1.53	4.37	7.31	1.50	4.00	7.00	1.50	4.2	7.00	34	1.53	1.50	4.40

	BbAvD	BbMxA	BbAvA	BbOU	BbMx.2.5	BbAv.2.5	BbMx.2.5.1	BbAv.2.5.1	BbAH	BbAHh
1	3.15	4.52	4.17	31	2.84	2.68	1.53	1.46	18	-0.50
2	3.86	4.83	4.46	33	1.69	1.64	2.40	2.27	16	-0.75
3	3.32	3.20	3.01	34	2.03	1.98	1.90	1.84	18	-0.25
4	4.17	1.51	1.48	34	2.20	2.11	1.80	1.74	16	1.25
5	3.93	6.20	5.58	33	1.81	1.75	2.14	2.09	16	-1.00
6	4.17	7.50	6.94	32	2.01	1.94	1.96	1.87	17	-1.00

	BbMxAHH	BbAvAHH	BbMxAHA	BbAvAHA	PSCH	PSCD	PSCA
1	2.07	2.03	1.90	1.86	1.98	3.35	4.63
2	2.05	1.97	1.96	1.91	1.78	4.24	4.43
3	2.08	2.05	1.87	1.83	2.12	3.53	3.74
4	1.77	1.75	2.25	2.16	6.93	3.83	1.63
5	2.12	2.06	1.86	1.82	1.64	4.18	5.82
6	1.90	1.86	2.05	2.01	1.53	4.48	6.91

Div	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS	HST
1	SP1 17/08/2018	Betis	Levante	0	3	A	0	1	A	22	6	8
2	SP1 17/08/2018	Girona	Valladolid	0	0	D	0	0	D	13	2	1
3	SP1 18/08/2018	Barcelona	Alaves	3	0	H	0	0	D	25	3	9
4	SP1 18/08/2018	Celta	Espanol	1	1	D	0	1	A	12	14	2
5	SP1 18/08/2018	Villarreal	Sociedad	1	2	A	1	1	D	16	8	7
6	SP1 19/08/2018	Eibar	Huesca	1	2	A	0	2	A	18	8	6

AST	HF	AF	HC	AC	HY	AY	HR	AR	B365H	B365D	B365A	BWH	BWD	BWA	IWH	IWD	IWA
1	4	10	10	5	3	0	2	0	1.66	4.00	5.0	1.70	3.7	5.25	1.75	3.60	4.9
2	1	21	20	3	2	1	1	0	1.75	3.60	5.0	1.75	3.5	5.25	1.80	3.60	4.5
3	0	6	13	7	1	0	2	0	1.11	10.00	21.0	1.11	10.0	20.00	1.12	9.00	20.0
4	5	13	14	8	7	3	2	0	1.85	3.50	4.5	1.91	3.4	4.25	1.90	3.50	4.1
5	4	16	10	4	6	2	3	0	2.04	3.40	3.8	2.05	3.3	3.90	2.00	3.40	3.8
6	6	12	13	7	0	1	1	0	1.66	3.75	5.5	1.70	3.7	5.25	1.70	3.75	5.0

	PSH	PSD	PSA	WHH	WHD	WHA	VCH	VCD	VCA	Bb1X2	BbMxH	BbAvH	BbMxD	BbAvD
1	1.69	4.19	5.11	1.67	3.9	4.75	1.67	4.2	5.2	40	1.75	1.68	4.25	4.00
2	1.80	3.70	4.99	1.75	3.6	4.60	1.80	3.7	4.8	40	1.85	1.78	3.83	3.60
3	1.11	11.27	25.40	1.08	9.0	29.00	1.10	10.5	34.0	40	1.13	1.10	11.50	9.82
4	1.93	3.64	4.27	1.91	3.5	4.00	1.93	3.5	4.4	38	1.97	1.90	3.73	3.53
5	2.06	3.51	3.91	2.05	3.3	3.60	2.05	3.5	3.9	40	2.11	2.03	3.62	3.43
6	1.72	3.90	5.26	1.73	3.6	4.75	1.70	3.8	5.0	40	1.76	1.70	3.93	3.77

	BbMxA	BbAvA	BbOU	BbMx.2.5	BbAv.2.5	BbMx.2.5.1	BbAv.2.5.1	BbAH	BbAHh	BbMxAHH
1	5.25	4.95	38	1.82	1.76	2.15	2.06	20	-0.75	1.89
2	5.27	4.79	38	2.21	2.13	1.78	1.71	20	-0.75	2.06
3	41.00	25.67	32	1.39	1.34	3.40	3.18	19	-2.50	1.95
4	4.50	4.20	36	2.13	2.06	1.84	1.76	18	-0.75	2.26
5	3.93	3.76	37	2.05	1.99	1.88	1.81	18	-0.25	1.76
6	5.50	5.08	37	1.95	1.88	1.98	1.91	19	-0.75	1.96

	BbAvAHH	BbMxAHA	BbAvAHA	PSCH	PSCD	PSCA
1	1.85	2.07	2.00	1.59	4.42	5.89
2	2.01	1.90	1.85	1.76	3.57	5.62
3	1.91	2.00	1.95	1.10	11.85	32.17

4	2.18	1.74	1.71	2.18	3.26	3.85
5	1.74	2.23	2.14	2.32	3.21	3.53
6	1.91	2.01	1.94	1.77	3.68	5.32

Div	Date	Time	HomeTeam	AwayTeam	FTHG	FTAG	FTR	HTHG	HTAG	HTR	HS	AS								
1	SP1	16/08/2019	20:00	Ath Bilbao	Barcelona	1	0	H	0	0	D	11 11								
2	SP1	17/08/2019	16:00	Celta	Real Madrid	1	3	A	0	1	A	7 17								
3	SP1	17/08/2019	18:00	Valencia	Sociedad	1	1	D	0	0	D	14 12								
4	SP1	17/08/2019	19:00	Mallorca	Eibar	2	1	H	1	0	H	16 11								
5	SP1	17/08/2019	20:00	Leganes	Osasuna	0	1	A	0	0	D	13 4								
6	SP1	17/08/2019	20:00	Villarreal	Granada	4	4	D	1	1	D	12 14								
HST	AST	HF	AF	HC	AC	HY	AY	HR	AR	B365H	B365D	B365A	BWH	BWD	BWA	IWH	IWD			
1	5	2	14	9	3	8	1	1	0	0	5.25	3.80	1.65	5.50	3.80	1.65	5.00	3.80		
2	4	11	17	12	6	4	5	2	0	1	4.75	4.20	1.65	4.40	4.20	1.72	5.30	4.20		
3	6	3	13	14	3	3	4	1	0	0	1.66	3.75	5.50	1.67	3.75	5.50	1.67	3.75		
4	4	5	13	14	9	3	2	3	0	0	2.80	3.20	2.60	2.95	3.10	2.60	2.90	3.10		
5	2	2	17	11	8	0	1	4	1	0	2.00	3.20	4.20	2.05	3.25	3.90	2.05	3.10		
6	7	7	10	16	2	7	3	1	0	0	1.60	3.80	6.50	1.60	3.80	6.25	1.63	4.00		
IWA	PSH	PSD	PSA	WHH	WHD	WHA	VCH	VCD	VCA	MaxH	MaxD	MaxA	AvgH	AvgD						
1	1.70	5.15	3.84	1.74	5.00	3.8	1.70	5.00	3.80	1.75	5.50	3.95	1.76	5.07	3.81					
2	1.60	4.73	4.18	1.72	5.25	4.2	1.60	4.75	4.20	1.73	5.30	4.40	1.73	4.67	4.12					
3	5.30	1.68	3.94	5.47	1.67	3.8	5.25	1.67	3.90	5.75	1.72	3.98	5.75	1.68	3.80					
4	2.60	2.98	3.14	2.66	2.90	3.1	2.62	2.90	3.13	2.70	3.05	3.20	2.70	2.91	3.09					
5	4.05	2.10	3.21	4.13	2.05	3.2	4.00	2.10	3.20	4.10	2.10	3.30	4.25	2.06	3.18					
6	5.50	1.62	3.99	6.13	1.60	3.9	5.80	1.65	4.00	5.75	1.65	4.15	6.50	1.61	3.95					
AvgA	B365	2.5	B365	2.5	1	P	2.5	P	2.5	1	Max	2.5	Max	2.5	1	Avg	2.5	Avg	2.5	1
1	1.71	1.80	2.00	1.81	2.09	1.85	2.11	1.79	2.05											
2	1.69	1.53	2.50	1.52	2.66	1.53	2.72	1.49	2.58											
3	5.29	2.00	1.80	2.08	1.82	2.14	1.83	2.07	1.77											
4	2.62	2.30	1.61	2.45	1.60	2.47	1.65	2.34	1.60											
5	4.02	2.50	1.53	2.72	1.50	2.75	1.54	2.59	1.49											
6	5.80	1.80	2.00	1.88	2.02	1.90	2.05	1.84	1.98											
AHh	B365AHh	B365AHA	PAHh	PAHA	MaxAHh	MaxAHA	AvgAHh	AvgAHA	B365CH	B365CD										
1	0.75	1.99	1.94	1.98	1.94	2.00	1.95	1.96	1.92	5.25	3.80									
2	0.75	2.04	1.89	2.01	1.91	2.05	1.91	2.00	1.88	5.25	4.20									
3	-0.75	1.91	2.02	1.91	2.01	1.93	2.03	1.89	1.99	1.66	3.75									
4	0.00	2.05	1.88	2.07	1.85	2.07	1.88	2.04	1.85	2.87	3.20									
5	-0.50	2.08	1.85	2.10	1.82	2.10	1.85	2.06	1.83	1.90	3.10									
6	-1.00	2.05	1.75	2.11	1.81	2.14	1.85	2.07	1.80	1.53	4.00									
B365CA	BWCH	BWCD	BWCA	IWCH	IWCD	IWCA	PSCH	PSCD	PSCA	WHCH	WHCD	WHCA	VCCH	VCCD						
1	1.65	4.75	3.75	1.75	5.00	3.80	1.70	5.34	3.62	1.78	5.00	3.8	1.70	4.80	3.80					
2	1.57	4.50	4.10	1.70	4.60	3.80	1.75	5.10	4.46	1.65	5.00	4.2	1.63	5.20	4.40					
3	5.50	1.65	3.80	5.50	1.67	3.80	5.30	1.69	3.88	5.47	1.65	3.9	5.25	1.70	3.90					
4	2.55	2.95	3.10	2.60	2.90	3.10	2.60	2.96	3.26	2.60	2.90	3.1	2.60	3.00	3.13					
5	5.00	1.95	3.20	4.50	1.90	3.15	4.85	1.90	3.18	5.30	2.05	3.2	4.00	1.90	3.20					
6	6.50	1.57	3.80	6.50	1.55	4.05	6.30	1.54	4.19	6.87	1.62	3.9	5.80	1.57	4.00					
VCCA	MaxCH	MaxCD	MaxCA	AvgCH	AvgCD	AvgCA	B365C	2.5	B365C	2.5	1	PC	2.5							
1	1.80	5.80	3.90	1.81	5.03	3.66	1.76	1.90	1.90	1.98										
2	1.65	6.00	4.52	1.75	4.93	4.26	1.65	1.44	2.75	1.49										
3	5.50	1.72	3.95	6.20	1.68	3.82	5.37	2.00	1.80	2.06										
4	2.63	3.05	3.29	2.72	2.93	3.14	2.59	2.20	1.66	2.20										
5	5.20	1.95	3.26	5.30	1.90	3.16	4.91	2.75	1.44	2.84										
6	7.00	1.58	4.20	7.30	1.54	4.05	6.66	1.90	1.90	1.95										
PC	2.5	1	MaxC	2.5	1	AvgC	2.5	AvgC	2.5	1	AHCh	B365CAHh	B365CAHA							



1	1.93	1.99	2.11	1.86	1.97	0.75	1.93	2.00
2	2.76	1.51	2.88	1.47	2.63	1.00	1.82	1.97
3	1.85	2.08	1.98	2.00	1.82	-0.75	1.94	1.99
4	1.74	2.38	1.74	2.24	1.66	0.00	2.11	1.82
5	1.47	2.85	1.50	2.69	1.46	-0.50	1.89	2.04
6	1.95	1.98	2.10	1.90	1.92	-1.00	1.96	1.97

	PCAHH	PCAHA	MaxCAHH	MaxCAHA	AvgCAHH	AvgCAHA
1	1.91	2.01	2.02	2.03	1.91	1.98
2	1.85	2.07	2.00	2.20	1.82	2.06
3	1.92	2.00	1.96	2.12	1.89	2.00
4	2.09	1.83	2.12	1.88	2.07	1.83
5	1.90	2.01	1.95	2.06	1.90	1.99
6	1.96	1.96	1.98	2.12	1.93	1.95

View(d1718); View(d1819); View(d1920)  
summary(d1718); summary(d1819); summary(d1920)

Div	Date	HomeTeam	AwayTeam
Length:380	Length:380	Length:380	Length:380
Class :character	Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character	Mode :character

FTHG	FTAG	FTR	HTHG
Min. :0.000	Min. :0.000	Length:380	Min. :0.0000
1st Qu.:0.750	1st Qu.:0.000	Class :character	1st Qu.:0.0000
Median :1.000	Median :1.000	Mode :character	Median :0.0000
Mean :1.547	Mean :1.147		Mean :0.6605
3rd Qu.:2.000	3rd Qu.:2.000		3rd Qu.:1.0000
Max. :7.000	Max. :6.000		Max. :5.0000

HTAG	HTR	HS	AS
Min. :0.0000	Length:380	Min. : 2.00	Min. : 1.00
1st Qu.:0.0000	Class :character	1st Qu.:10.00	1st Qu.: 8.00
Median :0.0000	Mode :character	Median :13.00	Median :10.00
Mean :0.4868		Mean :13.53	Mean :10.47
3rd Qu.:1.0000		3rd Qu.:16.00	3rd Qu.:13.00
Max. :3.0000		Max. :30.00	Max. :24.00

HST	AST	HF	AF
Min. : 0.000	Min. : 0.000	Min. : 4.00	Min. : 0.00
1st Qu.: 3.000	1st Qu.: 2.000	1st Qu.:11.00	1st Qu.:11.00
Median : 4.500	Median : 3.000	Median :13.00	Median :14.00
Mean : 4.758	Mean : 3.805	Mean :13.73	Mean :13.95
3rd Qu.: 6.000	3rd Qu.: 5.000	3rd Qu.:17.00	3rd Qu.:17.00
Max. :14.000	Max. :13.000	Max. :29.00	Max. :29.00

HC	AC	HY	AY
Min. : 0.000	Min. : 0.000	Min. :0.000	Min. :0.000
1st Qu.: 4.000	1st Qu.: 2.000	1st Qu.:1.000	1st Qu.:2.000
Median : 5.000	Median : 4.000	Median :2.000	Median :3.000
Mean : 5.613	Mean : 4.192	Mean :2.339	Mean :2.676

3rd Qu.: 7.000	3rd Qu.: 6.000	3rd Qu.:3.000	3rd Qu.:4.000
Max. :16.000	Max. :14.000	Max. :8.000	Max. :9.000
HR	AR	B365H	B365D
Min. :0.0000	Min. :0.00000	Min. : 1.050	Min. : 2.790
1st Qu.:0.0000	1st Qu.:0.00000	1st Qu.: 1.617	1st Qu.: 3.290
Median :0.0000	Median :0.00000	Median : 2.075	Median : 3.500
Mean :0.1105	Mean :0.07895	Mean : 2.777	Mean : 4.259
3rd Qu.:0.0000	3rd Qu.:0.00000	3rd Qu.: 2.790	3rd Qu.: 4.330
Max. :2.0000	Max. :2.00000	Max. :17.000	Max. :15.000
B365A	BWH	BWD	BWA
Min. : 1.170	Min. : 1.050	Min. : 2.950	Min. : 1.180
1st Qu.: 2.600	1st Qu.: 1.650	1st Qu.: 3.300	1st Qu.: 2.600
Median : 3.700	Median : 2.100	Median : 3.600	Median : 3.700
Mean : 5.192	Mean : 2.744	Mean : 4.278	Mean : 5.204
3rd Qu.: 5.500	3rd Qu.: 2.750	3rd Qu.: 4.330	3rd Qu.: 5.500
Max. :34.000	Max. :14.500	Max. :15.500	Max. :34.000
IWH	IWD	IWA	LBH
Min. : 1.070	Min. : 3.050	Min. : 1.200	Min. : 1.050
1st Qu.: 1.650	1st Qu.: 3.300	1st Qu.: 2.600	1st Qu.: 1.610
Median : 2.100	Median : 3.500	Median : 3.500	Median : 2.050
Mean : 2.721	Mean : 4.161	Mean : 5.041	Mean : 2.742
3rd Qu.: 2.700	3rd Qu.: 4.200	3rd Qu.: 5.300	3rd Qu.: 2.750
Max. :15.000	Max. :12.000	Max. :27.000	Max. :19.000
			NA's :1
LBD	LBA	PSH	PSD
Min. : 2.900	Min. : 1.170	Min. : 1.050	Min. : 3.020
1st Qu.: 3.250	1st Qu.: 2.575	1st Qu.: 1.660	1st Qu.: 3.410
Median : 3.500	Median : 3.600	Median : 2.120	Median : 3.705
Mean : 4.152	Mean : 5.375	Mean : 2.857	Mean : 4.539
3rd Qu.: 4.200	3rd Qu.: 5.500	3rd Qu.: 2.850	3rd Qu.: 4.455
Max. :17.000	Max. :41.000	Max. :19.650	Max. :20.380
NA's :1	NA's :1		
PSA	WHH	WHD	WHA
Min. : 1.180	Min. : 1.060	Min. : 2.900	Min. : 1.170
1st Qu.: 2.670	1st Qu.: 1.665	1st Qu.: 3.250	1st Qu.: 2.600
Median : 3.845	Median : 2.100	Median : 3.500	Median : 3.550
Mean : 5.522	Mean : 2.738	Mean : 4.092	Mean : 5.041
3rd Qu.: 5.942	3rd Qu.: 2.750	3rd Qu.: 4.200	3rd Qu.: 5.500
Max. :36.500	Max. :17.000	Max. :15.000	Max. :26.000
VCH	VCD	VCA	Bb1X2
Min. : 1.040	Min. : 3.000	Min. : 1.180	Min. : 3.00
1st Qu.: 1.650	1st Qu.: 3.400	1st Qu.: 2.630	1st Qu.:35.00
Median : 2.100	Median : 3.700	Median : 3.700	Median :37.00
Mean : 2.762	Mean : 4.416	Mean : 5.472	Mean :37.71
3rd Qu.: 2.800	3rd Qu.: 4.400	3rd Qu.: 5.750	3rd Qu.:40.00
Max. :15.000	Max. :17.000	Max. :36.000	Max. :43.00
BbMxH	BbAvH	BbMxD	BbAvD
Min. : 1.080	Min. : 1.050	Min. : 3.110	Min. : 2.940
1st Qu.: 1.700	1st Qu.: 1.640	1st Qu.: 3.478	1st Qu.: 3.328

Median : 2.200	Median : 2.090	Median : 3.750	Median : 3.570
Mean : 2.966	Mean : 2.743	Mean : 4.636	Mean : 4.261
3rd Qu.: 2.882	3rd Qu.: 2.765	3rd Qu.: 4.553	3rd Qu.: 4.272
Max. :19.650	Max. :16.300	Max. :20.380	Max. :15.320

BbMxA	BbAvA	BbOU	BbMx.2.5
Min. : 1.210	Min. : 1.170	Min. : 3.00	Min. :1.130
1st Qu.: 2.728	1st Qu.: 2.607	1st Qu.:31.75	1st Qu.:1.667
Median : 3.920	Median : 3.665	Median :34.00	Median :1.960
Mean : 6.107	Mean : 5.190	Mean :34.06	Mean :1.950
3rd Qu.: 6.105	3rd Qu.: 5.543	3rd Qu.:37.00	3rd Qu.:2.203
Max. :67.000	Max. :33.420	Max. :42.00	Max. :3.080

BbAv.2.5	BbMx.2.5.1	BbAv.2.5.1	BbAH
Min. :1.120	Min. :1.470	Min. :1.410	Min. : 1.00
1st Qu.:1.617	1st Qu.:1.780	1st Qu.:1.718	1st Qu.:17.00
Median :1.880	Median :2.000	Median :1.920	Median :18.00
Mean :1.872	Mean :2.284	Mean :2.162	Mean :18.16
3rd Qu.:2.120	3rd Qu.:2.402	3rd Qu.:2.283	3rd Qu.:19.00
Max. :2.850	Max. :7.000	Max. :5.970	Max. :24.00

BbAHh	BbMxAHH	BbAvAHH	BbMxAHA
Min. :-3.2500	Min. :1.610	Min. :1.580	Min. :1.680
1st Qu.: -0.7500	1st Qu.:1.890	1st Qu.:1.840	1st Qu.:1.897
Median :-0.2500	Median :1.985	Median :1.930	Median :1.970
Mean :-0.4059	Mean :1.988	Mean :1.938	Mean :1.988
3rd Qu.: 0.0625	3rd Qu.:2.070	3rd Qu.:2.020	3rd Qu.:2.080
Max. : 2.0000	Max. :2.420	Max. :2.340	Max. :2.520

BbAvAHA	PSCH	PSCD	PSCA
Min. :1.630	Min. : 1.060	Min. : 2.930	Min. : 1.160
1st Qu.:1.850	1st Qu.: 1.640	1st Qu.: 3.410	1st Qu.: 2.590
Median :1.930	Median : 2.120	Median : 3.700	Median : 3.850
Mean :1.937	Mean : 2.839	Mean : 4.508	Mean : 5.695
3rd Qu.:2.030	3rd Qu.: 2.980	3rd Qu.: 4.560	3rd Qu.: 6.095
Max. :2.440	Max. :18.700	Max. :18.500	Max. :46.000
	NA's :1	NA's :1	NA's :1

Div	Date	HomeTeam	AwayTeam
Length:380	Length:380	Length:380	Length:380
Class :character	Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character	Mode :character

FTHG	FTAG	FTR	HTHG
Min. :0.000	Min. :0.000	Length:380	Min. :0.0000
1st Qu.:1.000	1st Qu.:0.000	Class :character	1st Qu.:0.0000
Median :1.000	Median :1.000	Mode :character	Median :0.0000
Mean :1.453	Mean :1.134		Mean :0.5447
3rd Qu.:2.000	3rd Qu.:2.000		3rd Qu.:1.0000
Max. :8.000	Max. :6.000		Max. :3.0000
HTAG	HTR	HS	AS
Min. :0.0000	Length:380	Min. : 3.00	Min. : 2.00

1st Qu.:0.0000	Class :character	1st Qu.:10.00	1st Qu.: 8.00
Median :0.0000	Mode :character	Median :13.00	Median :10.00
Mean :0.5132		Mean :13.87	Mean :10.43
3rd Qu.:1.0000		3rd Qu.:17.00	3rd Qu.:13.00
Max. :5.0000		Max. :34.00	Max. :21.00
HST	AST	HF	AF
Min. : 0.000	Min. : 0.000	Min. : 1.00	Min. : 3.00
1st Qu.: 3.000	1st Qu.: 2.000	1st Qu.:11.00	1st Qu.:11.00
Median : 5.000	Median : 3.000	Median :13.00	Median :13.00
Mean : 4.834	Mean : 3.589	Mean :13.63	Mean :13.45
3rd Qu.: 6.000	3rd Qu.: 5.000	3rd Qu.:16.00	3rd Qu.:16.00
Max. :15.000	Max. :11.000	Max. :26.00	Max. :27.00
HC	AC	HY	AY
Min. : 0.000	Min. : 0.000	Min. :0.000	Min. :0.000
1st Qu.: 4.000	1st Qu.: 2.000	1st Qu.:1.000	1st Qu.:2.000
Median : 5.000	Median : 4.000	Median :2.000	Median :3.000
Mean : 5.574	Mean : 4.021	Mean :2.529	Mean :2.642
3rd Qu.: 7.000	3rd Qu.: 6.000	3rd Qu.:4.000	3rd Qu.:3.000
Max. :15.000	Max. :12.000	Max. :8.000	Max. :7.000
HR	AR	B365H	B365D
Min. :0.00000	Min. :0.0000	Min. : 1.080	Min. : 2.870
1st Qu.:0.00000	1st Qu.:0.0000	1st Qu.: 1.660	1st Qu.: 3.300
Median :0.00000	Median :0.0000	Median : 2.120	Median : 3.500
Mean :0.08684	Mean :0.1211	Mean : 2.596	Mean : 3.996
3rd Qu.:0.00000	3rd Qu.:0.0000	3rd Qu.: 2.800	3rd Qu.: 4.000
Max. :1.00000	Max. :2.0000	Max. :17.000	Max. :11.000
B365A	BWH	BWD	BWA
Min. : 1.160	Min. : 1.060	Min. : 2.900	Min. : 1.190
1st Qu.: 2.547	1st Qu.: 1.670	1st Qu.: 3.300	1st Qu.: 2.600
Median : 3.500	Median : 2.150	Median : 3.500	Median : 3.500
Mean : 4.790	Mean : 2.579	Mean : 3.991	Mean : 4.745
3rd Qu.: 5.062	3rd Qu.: 2.800	3rd Qu.: 4.000	3rd Qu.: 5.250
Max. :29.000	Max. :15.000	Max. :12.000	Max. :36.000
IWH	IWD	IWA	PSH
Min. : 1.070	Min. : 2.850	Min. : 1.200	Min. : 1.080
1st Qu.: 1.700	1st Qu.: 3.300	1st Qu.: 2.600	1st Qu.: 1.700
Median : 2.150	Median : 3.500	Median : 3.450	Median : 2.180
Mean : 2.553	Mean : 3.943	Mean : 4.587	Mean : 2.639
3rd Qu.: 2.763	3rd Qu.: 4.000	3rd Qu.: 4.950	3rd Qu.: 2.840
Max. :13.000	Max. :13.000	Max. :28.000	Max. :19.070
PSD	PSA	WHH	WHD
Min. : 2.990	Min. : 1.180	Min. : 1.050	Min. : 2.500
1st Qu.: 3.357	1st Qu.: 2.652	1st Qu.: 1.700	1st Qu.: 3.300
Median : 3.640	Median : 3.575	Median : 2.150	Median : 3.500
Mean : 4.133	Mean : 4.994	Mean : 2.564	Mean : 3.997
3rd Qu.: 4.170	3rd Qu.: 5.230	3rd Qu.: 2.800	3rd Qu.: 4.000
Max. :13.220	Max. :36.830	Max. :17.000	Max. :13.000
WHA	VCH	VCD	VCA
Min. : 1.150	Min. : 1.060	Min. : 3.000	Min. : 1.18
1st Qu.: 2.587	1st Qu.: 1.700	1st Qu.: 3.300	1st Qu.: 2.60
Median : 3.500	Median : 2.150	Median : 3.600	Median : 3.60
Mean : 4.779	Mean : 2.627	Mean : 4.097	Mean : 5.00
3rd Qu.: 5.000	3rd Qu.: 2.800	3rd Qu.: 4.200	3rd Qu.: 5.20
Max. :34.000	Max. :21.000	Max. :13.000	Max. :41.00

Bb1X2	BbMxH	BbAvH	BbMxD
Min. :31.00	Min. : 1.100	Min. : 1.070	Min. : 3.040
1st Qu.:34.00	1st Qu.: 1.750	1st Qu.: 1.690	1st Qu.: 3.420
Median :36.00	Median : 2.250	Median : 2.160	Median : 3.735
Mean :36.13	Mean : 2.739	Mean : 2.582	Mean : 4.251
3rd Qu.:38.00	3rd Qu.: 2.913	3rd Qu.: 2.792	3rd Qu.: 4.250
Max. :41.00	Max. :21.000	Max. :16.550	Max. :15.000

BbAvD	BbMxA	BbAvA	BbOU
Min. : 2.920	Min. : 1.200	Min. : 1.170	Min. :28.00
1st Qu.: 3.288	1st Qu.: 2.750	1st Qu.: 2.620	1st Qu.:32.00
Median : 3.550	Median : 3.695	Median : 3.495	Median :34.00
Mean : 4.005	Mean : 5.361	Mean : 4.763	Mean :33.84
3rd Qu.: 4.032	3rd Qu.: 5.370	3rd Qu.: 5.043	3rd Qu.:35.00
Max. :12.430	Max. :52.000	Max. :33.380	Max. :39.00

BbMx.2.5	BbAv.2.5	BbMx.2.5.1	BbAv.2.5.1
Min. :1.200	Min. :1.170	Min. :1.420	Min. :1.380
1st Qu.:1.710	1st Qu.:1.650	1st Qu.:1.710	1st Qu.:1.650
Median :2.000	Median :1.940	Median :1.950	Median :1.870
Mean :2.029	Mean :1.947	Mean :2.116	Mean :2.018
3rd Qu.:2.315	3rd Qu.:2.230	3rd Qu.:2.303	3rd Qu.:2.210
Max. :3.200	Max. :2.890	Max. :5.250	Max. :4.740

BbAH	BbAHh	BbMxAHH	BbAvAHH
Min. :15.00	Min. : -3.0000	Min. :1.560	Min. :1.520
1st Qu.:19.00	1st Qu.: -1.0000	1st Qu.:1.850	1st Qu.:1.800
Median :20.00	Median : -0.2500	Median :2.030	Median :1.970
Mean :19.88	Mean : -0.4033	Mean :2.055	Mean :1.992
3rd Qu.:21.00	3rd Qu.: 0.2500	3rd Qu.:2.200	3rd Qu.:2.130
Max. :24.00	Max. : 2.0000	Max. :3.270	Max. :3.020

BbMxAHA	BbAvAHA	PSCH	PSCD
Min. :1.450	Min. :1.410	Min. : 1.070	Min. : 2.860
1st Qu.:1.800	1st Qu.:1.750	1st Qu.: 1.698	1st Qu.: 3.310
Median :1.950	Median :1.890	Median : 2.190	Median : 3.610
Mean :1.972	Mean :1.915	Mean : 2.726	Mean : 4.102
3rd Qu.:2.130	3rd Qu.:2.070	3rd Qu.: 2.970	3rd Qu.: 4.202
Max. :2.850	Max. :2.670	Max. :18.040	Max. :14.910

PSCA

Min. : 1.190

1st Qu.: 2.612

Median : 3.645

Mean : 5.100

3rd Qu.: 5.575

Max. :36.030

Div	Date	Time	HomeTeam
Length:380	Length:380	Length:380	Length:380
Class :character	Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character	Mode :character

AwayTeam	FTHG	FTAG	FTR
Length:380	Min. :0.000	Min. :0.000	Length:380
Class :character	1st Qu.:1.000	1st Qu.:0.000	Class :character

Mode :character	Median :1.000	Median :1.000	Mode :character
	Mean :1.437	Mean :1.042	
	3rd Qu.:2.000	3rd Qu.:2.000	
	Max. :6.000	Max. :5.000	

HTHG	HTAG	HTR	HS
Min. :0.0000	Min. :0.00	Length:380	Min. : 3.00
1st Qu.:0.0000	1st Qu.:0.00	Class :character	1st Qu.: 9.00
Median :0.0000	Median :0.00	Mode :character	Median :12.00
Mean :0.6026	Mean :0.45		Mean :12.46
3rd Qu.:1.0000	3rd Qu.:1.00		3rd Qu.:15.00
Max. :4.0000	Max. :3.00		Max. :25.00

AS	HST	AST	HF
Min. : 1.00	Min. : 0.000	Min. : 0.000	Min. : 4.00
1st Qu.: 7.00	1st Qu.: 3.000	1st Qu.: 2.000	1st Qu.:11.00
Median :10.00	Median : 4.000	Median : 3.000	Median :13.00
Mean :10.14	Mean : 4.337	Mean : 3.511	Mean :13.66
3rd Qu.:12.25	3rd Qu.: 6.000	3rd Qu.: 5.000	3rd Qu.:16.00
Max. :24.00	Max. :17.000	Max. :12.000	Max. :28.00

AF	HC	AC	HY
Min. : 5.00	Min. : 0.000	Min. : 0.000	Min. :0.000
1st Qu.:11.00	1st Qu.: 3.000	1st Qu.: 2.750	1st Qu.:1.000
Median :13.00	Median : 5.000	Median : 4.000	Median :2.000
Mean :13.79	Mean : 5.042	Mean : 4.195	Mean :2.547
3rd Qu.:16.00	3rd Qu.: 7.000	3rd Qu.: 6.000	3rd Qu.:4.000
Max. :30.00	Max. :14.000	Max. :12.000	Max. :7.000

AY	HR	AR	B365H
Min. :0.000	Min. :0.0	Min. :0.0000	Min. : 1.120
1st Qu.:2.000	1st Qu.:0.0	1st Qu.:0.0000	1st Qu.: 1.700
Median :2.000	Median :0.0	Median :0.0000	Median : 2.200
Mean :2.584	Mean :0.1	Mean :0.1263	Mean : 2.595
3rd Qu.:4.000	3rd Qu.:0.0	3rd Qu.:0.0000	3rd Qu.: 2.900
Max. :8.000	Max. :2.0	Max. :2.0000	Max. :10.000

B365D	B365A	BWH	BWD
Min. :2.800	Min. : 1.280	Min. : 1.120	Min. :2.700
1st Qu.:3.200	1st Qu.: 2.500	1st Qu.: 1.700	1st Qu.:3.200
Median :3.400	Median : 3.500	Median : 2.200	Median :3.400
Mean :3.805	Mean : 4.465	Mean : 2.599	Mean :3.813
3rd Qu.:4.000	3rd Qu.: 5.062	3rd Qu.: 2.900	3rd Qu.:4.000
Max. :9.500	Max. :21.000	Max. :10.000	Max. :9.500

BWA	IWH	IWD	IWA
Min. : 1.280	Min. :1.130	Min. :2.750	Min. : 1.280
1st Qu.: 2.550	1st Qu.:1.730	1st Qu.:3.188	1st Qu.: 2.550
Median : 3.500	Median :2.200	Median :3.400	Median : 3.450
Mean : 4.438	Mean :2.603	Mean :3.770	Mean : 4.378
3rd Qu.: 5.250	3rd Qu.:2.913	3rd Qu.:4.000	3rd Qu.: 5.000
Max. :20.000	Max. :9.900	Max. :8.800	Max. :19.500

PSH	PSD	PSA	WHH
-----	-----	-----	-----

Min. : 1.120	Min. :2.780	Min. : 1.290	Min. : 1.110
1st Qu.: 1.720	1st Qu.:3.230	1st Qu.: 2.600	1st Qu.: 1.700
Median : 2.260	Median :3.515	Median : 3.520	Median : 2.225
Mean : 2.650	Mean :3.894	Mean : 4.673	Mean : 2.616
3rd Qu.: 2.958	3rd Qu.:4.088	3rd Qu.: 5.230	3rd Qu.: 2.900
Max. :10.020	Max. :9.950	Max. :25.500	Max. :10.000
NA's :2	NA's :2	NA's :2	

WHD	WHA	VCH	VCD
Min. :2.800	Min. : 1.270	Min. : 1.100	Min. :2.800
1st Qu.:3.200	1st Qu.: 2.550	1st Qu.: 1.700	1st Qu.:3.200
Median :3.400	Median : 3.450	Median : 2.200	Median :3.500
Mean :3.802	Mean : 4.623	Mean : 2.582	Mean :3.831
3rd Qu.:4.000	3rd Qu.: 5.250	3rd Qu.: 2.885	3rd Qu.:4.025
Max. :9.000	Max. :26.000	Max. :10.500	Max. :9.500

VCA	MaxH	MaxD	MaxA
Min. : 1.250	Min. : 1.160	Min. : 2.900	Min. : 1.320
1st Qu.: 2.500	1st Qu.: 1.778	1st Qu.: 3.340	1st Qu.: 2.678
Median : 3.400	Median : 2.310	Median : 3.600	Median : 3.625
Mean : 4.453	Mean : 2.757	Mean : 4.019	Mean : 4.951
3rd Qu.: 5.000	3rd Qu.: 3.055	3rd Qu.: 4.202	3rd Qu.: 5.500
Max. :26.000	Max. :11.000	Max. :10.500	Max. :31.000

AvgH	AvgD	AvgA	B365.2.5
Min. :1.120	Min. :2.780	Min. : 1.280	Min. :1.280
1st Qu.:1.718	1st Qu.:3.210	1st Qu.: 2.550	1st Qu.:1.800
Median :2.220	Median :3.435	Median : 3.455	Median :2.100
Mean :2.608	Mean :3.813	Mean : 4.483	Mean :2.108
3rd Qu.:2.917	3rd Qu.:4.022	3rd Qu.: 5.093	3rd Qu.:2.500
Max. :9.910	Max. :9.160	Max. :21.610	Max. :3.400

B365.2.5.1	P.2.5	P.2.5.1	Max.2.5
Min. :1.330	Min. :1.310	Min. :1.370	Min. :1.310
1st Qu.:1.530	1st Qu.:1.810	1st Qu.:1.570	1st Qu.:1.850
Median :1.720	Median :2.145	Median :1.770	Median :2.175
Mean :1.867	Mean :2.162	Mean :1.912	Mean :2.196
3rd Qu.:2.000	3rd Qu.:2.520	3rd Qu.:2.098	3rd Qu.:2.553
Max. :3.750	Max. :3.340	Max. :3.640	Max. :3.400

Max.2.5.1	Avg.2.5	Avg.2.5.1	AHh
Min. :1.380	Min. :1.260	Min. :1.350	Min. : -2.5000
1st Qu.:1.610	1st Qu.:1.780	1st Qu.:1.560	1st Qu.: -0.7500
Median :1.820	Median :2.080	Median :1.750	Median : -0.2500
Mean :1.950	Mean :2.099	Mean :1.869	Mean : -0.3289
3rd Qu.:2.132	3rd Qu.:2.440	3rd Qu.:2.040	3rd Qu.: 0.0000
Max. :3.950	Max. :3.160	Max. :3.680	Max. : 1.7500

B365AHH	B365AHA	PAHH	PAHA
Min. :1.670	Min. :1.650	Min. :1.750	Min. :1.680
1st Qu.:1.900	1st Qu.:1.890	1st Qu.:1.890	1st Qu.:1.880
Median :1.970	Median :1.960	Median :1.970	Median :1.950
Mean :1.963	Mean :1.954	Mean :1.966	Mean :1.954
3rd Qu.:2.040	3rd Qu.:2.030	3rd Qu.:2.040	3rd Qu.:2.030
Max. :2.200	Max. :2.160	Max. :2.310	Max. :2.200

NA's :10	NA's :10	NA's :2	NA's :2
MaxAHH	MaxAHA	AvgAHH	AvgAHA
Min. :1.800	Min. :1.710	Min. :1.740	Min. :1.670
1st Qu.:1.920	1st Qu.:1.910	1st Qu.:1.870	1st Qu.:1.860
Median :1.990	Median :1.980	Median :1.940	Median :1.930
Mean :1.995	Mean :1.988	Mean :1.939	Mean :1.932
3rd Qu.:2.070	3rd Qu.:2.060	3rd Qu.:2.010	3rd Qu.:2.000
Max. :2.330	Max. :2.250	Max. :2.270	Max. :2.170
B365CH	B365CD	B365CA	BWCH
Min. : 1.100	Min. : 2.700	Min. : 1.250	Min. :1.100
1st Qu.: 1.700	1st Qu.: 3.200	1st Qu.: 2.600	1st Qu.:1.715
Median : 2.150	Median : 3.400	Median : 3.550	Median :2.200
Mean : 2.598	Mean : 3.857	Mean : 4.628	Mean :2.603
3rd Qu.: 2.870	3rd Qu.: 4.000	3rd Qu.: 5.250	3rd Qu.:2.900
Max. :11.000	Max. :10.000	Max. :26.000	Max. :9.750
BWCD	BWCA	IWCH	IWCD
Min. : 2.750	Min. : 1.28	Min. : 1.120	Min. :2.700
1st Qu.: 3.200	1st Qu.: 2.60	1st Qu.: 1.730	1st Qu.:3.150
Median : 3.400	Median : 3.50	Median : 2.225	Median :3.400
Mean : 3.817	Mean : 4.55	Mean : 2.609	Mean :3.747
3rd Qu.: 4.000	3rd Qu.: 5.25	3rd Qu.: 2.900	3rd Qu.:4.000
Max. :10.000	Max. :23.00	Max. :11.000	Max. :9.000
IWCA	PSCH	PSCD	PSCA
Min. : 1.250	Min. : 1.100	Min. : 2.710	Min. : 1.270
1st Qu.: 2.600	1st Qu.: 1.720	1st Qu.: 3.188	1st Qu.: 2.688
Median : 3.500	Median : 2.275	Median : 3.480	Median : 3.640
Mean : 4.368	Mean : 2.672	Mean : 3.900	Mean : 4.873
3rd Qu.: 5.100	3rd Qu.: 2.975	3rd Qu.: 4.062	3rd Qu.: 5.440
Max. :20.000	Max. :10.930	Max. :11.520	Max. :28.530
WHCH	WHCD	WHCA	VCCH
Min. : 1.080	Min. : 2.620	Min. : 1.250	Min. : 1.080
1st Qu.: 1.700	1st Qu.: 3.200	1st Qu.: 2.600	1st Qu.: 1.722
Median : 2.225	Median : 3.400	Median : 3.600	Median : 2.225
Mean : 2.647	Mean : 3.827	Mean : 4.793	Mean : 2.600
3rd Qu.: 2.900	3rd Qu.: 4.000	3rd Qu.: 5.250	3rd Qu.: 2.880
Max. :11.000	Max. :11.000	Max. :26.000	Max. :10.500
VCCD	VCCA	MaxCH	MaxCD
Min. : 2.750	Min. : 1.250	Min. : 1.130	Min. : 2.860
1st Qu.: 3.200	1st Qu.: 2.600	1st Qu.: 1.788	1st Qu.: 3.320
Median : 3.450	Median : 3.400	Median : 2.355	Median : 3.610
Mean : 3.845	Mean : 4.549	Mean : 2.836	Mean : 4.062
3rd Qu.: 4.000	3rd Qu.: 5.050	3rd Qu.: 3.105	3rd Qu.: 4.242
Max. :10.500	Max. :26.000	Max. :13.000	Max. :12.400
MaxCA	AvgCH	AvgCD	AvgCA
Min. : 1.320	Min. : 1.100	Min. : 2.710	Min. : 1.280
1st Qu.: 2.743	1st Qu.: 1.710	1st Qu.: 3.167	1st Qu.: 2.618
Median : 3.870	Median : 2.235	Median : 3.430	Median : 3.545
Mean : 5.215	Mean : 2.625	Mean : 3.824	Mean : 4.630



3rd Qu.: 5.750	3rd Qu.: 2.908	3rd Qu.: 4.032	3rd Qu.: 5.282
Max. :31.370	Max. :10.390	Max. :10.410	Max. :24.600

B365C.2.5	B365C.2.5.1	PC.2.5	PC.2.5.1	MaxC.2.5
Min. :1.220	Min. :1.30	Min. :1.220	Min. :1.320	Min. :1.260
1st Qu.:1.720	1st Qu.:1.53	1st Qu.:1.795	1st Qu.:1.570	1st Qu.:1.857
Median :2.100	Median :1.72	Median :2.125	Median :1.790	Median :2.190
Mean :2.142	Mean :1.87	Mean :2.188	Mean :1.922	Mean :2.241
3rd Qu.:2.500	3rd Qu.:2.10	3rd Qu.:2.530	3rd Qu.:2.110	3rd Qu.:2.560
Max. :3.500	Max. :4.33	Max. :3.720	Max. :4.520	Max. :3.720

MaxC.2.5.1	AvgC.2.5	AvgC.2.5.1	AHCh
Min. :1.330	Min. :1.220	Min. :1.290	Min. :-2.7500
1st Qu.:1.627	1st Qu.:1.760	1st Qu.:1.550	1st Qu.: -0.7500
Median :1.840	Median :2.080	Median :1.750	Median :-0.2500
Mean :1.997	Mean :2.118	Mean :1.878	Mean :-0.3329
3rd Qu.:2.223	3rd Qu.:2.440	3rd Qu.:2.062	3rd Qu.: 0.0000
Max. :4.610	Max. :3.520	Max. :4.130	Max. : 1.7500

B365CAHH	B365CAHA	PCAHH	PCAHA
Min. :1.700	Min. :1.670	Min. :1.750	Min. :1.750
1st Qu.:1.880	1st Qu.:1.880	1st Qu.:1.880	1st Qu.:1.880
Median :1.960	Median :1.960	Median :1.960	Median :1.960
Mean :1.959	Mean :1.956	Mean :1.962	Mean :1.959
3rd Qu.:2.040	3rd Qu.:2.040	3rd Qu.:2.040	3rd Qu.:2.040
Max. :2.160	Max. :2.160	Max. :2.200	Max. :2.210

MaxCAHH	MaxCAHA	AvgCAHH	AvgCAHA
Min. :1.780	Min. :1.800	Min. :1.72	Min. :1.750
1st Qu.:1.940	1st Qu.:1.930	1st Qu.:1.86	1st Qu.:1.860
Median :2.020	Median :2.020	Median :1.93	Median :1.940
Mean :2.021	Mean :2.016	Mean :1.94	Mean :1.938
3rd Qu.:2.100	3rd Qu.:2.100	3rd Qu.:2.02	3rd Qu.:2.010
Max. :2.260	Max. :2.270	Max. :2.15	Max. :2.180

```
#Ahora seleccionaremos únicamente las columnas Date, HomeTeam,
# AwayTeam, FTHG, FTAG y FTR en cada uno de los data frames.
# Primero guardaremos los data frames en una lista con nombre lista
# y después con ayuda de las funciones lapply y select
# (del paquete dplyr), seleccionaremos las columnas deseadas.
# Los nuevos data frames quedarán guardados en nlista.
```

```
lista <- list(d1718, d1819, d1920)
nlista <- lapply(lista, select, Date, HomeTeam, AwayTeam, FTHG, FTAG, FTR)
```

```
#Con las funciones lapply y str observaremos la estructura
# de nuestros nuevos data frames
lapply(nlista, str)
```

```
'data.frame': 380 obs. of 6 variables:
 $ Date : chr "18/08/17" "18/08/17" "19/08/17" "19/08/17" ...
 $ HomeTeam: chr "Leganes" "Valencia" "Celta" "Girona" ...
 $ AwayTeam: chr "Alaves" "Las Palmas" "Sociedad" "Ath Madrid" ...
```

```

$ FTHG : int 1 1 2 2 1 0 2 0 1 0 ...
$ FTAG : int 0 0 3 2 1 0 0 3 0 1 ...
$ FTR : chr "H" "H" "A" "D" ...
'data.frame': 380 obs. of 6 variables:
$ Date : chr "17/08/2018" "17/08/2018" "18/08/2018" "18/08/2018" ...
$ HomeTeam: chr "Betis" "Girona" "Barcelona" "Celta" ...
$ AwayTeam: chr "Levante" "Valladolid" "Alaves" "Espanol" ...
$ FTHG : int 0 0 3 1 1 1 2 1 2 1 ...
$ FTAG : int 3 0 0 1 2 2 0 4 1 1 ...
$ FTR : chr "A" "D" "H" "D" ...
'data.frame': 380 obs. of 6 variables:
$ Date : chr "16/08/2019" "17/08/2019" "17/08/2019" "17/08/2019" ...
$ HomeTeam: chr "Ath Bilbao" "Celta" "Valencia" "Mallorca" ...
$ AwayTeam: chr "Barcelona" "Real Madrid" "Sociedad" "Eibar" ...
$ FTHG : int 1 1 1 2 0 4 1 0 1 1 ...
$ FTAG : int 0 3 1 1 1 4 0 2 2 0 ...
$ FTR : chr "H" "A" "D" "H" ...

```

```
[[1]]
```

```
NULL
```

```
[[2]]
```

```
NULL
```

```
[[3]]
```

```
NULL
```

```

#Arreglamos las columnas Date para que R reconozca
# los elementos como fechas, esto lo hacemos con las funciones
# mutate (paquete dplyr) y as.Date.
nlista[[1]] <- mutate(nlista[[1]], Date = as.Date(Date, "%d/%m/%y"))
nlista[[2]] <- mutate(nlista[[2]], Date = as.Date(Date, "%d/%m/%Y"))
nlista[[3]] <- mutate(nlista[[3]], Date = as.Date(Date, "%d/%m/%Y"))

#Verificamos que nuestros cambios se hayan llevado a cabo
lapply(nlista, str)

```

```

'data.frame': 380 obs. of 6 variables:
$ Date : Date, format: "2017-08-18" "2017-08-18" ...
$ HomeTeam: chr "Leganes" "Valencia" "Celta" "Girona" ...
$ AwayTeam: chr "Alaves" "Las Palmas" "Sociedad" "Ath Madrid" ...
$ FTHG : int 1 1 2 2 1 0 2 0 1 0 ...
$ FTAG : int 0 0 3 2 1 0 0 3 0 1 ...
$ FTR : chr "H" "H" "A" "D" ...
'data.frame': 380 obs. of 6 variables:
$ Date : Date, format: "2018-08-17" "2018-08-17" ...
$ HomeTeam: chr "Betis" "Girona" "Barcelona" "Celta" ...
$ AwayTeam: chr "Levante" "Valladolid" "Alaves" "Espanol" ...
$ FTHG : int 0 0 3 1 1 1 2 1 2 1 ...
$ FTAG : int 3 0 0 1 2 2 0 4 1 1 ...
$ FTR : chr "A" "D" "H" "D" ...
'data.frame': 380 obs. of 6 variables:
$ Date : Date, format: "2019-08-16" "2019-08-17" ...

```

```
$ HomeTeam: chr "Ath Bilbao" "Celta" "Valencia" "Mallorca" ...
$ AwayTeam: chr "Barcelona" "Real Madrid" "Sociedad" "Eibar" ...
$ FTHG : int 1 1 1 2 0 4 1 0 1 1 ...
$ FTAG : int 0 3 1 1 1 4 0 2 2 0 ...
$ FTR : chr "H" "A" "D" "H" ...
```

```
[[1]]
NULL
```

```
[[2]]
NULL
```

```
[[3]]
NULL
```

```
#Finalmente, con ayuda de las funciones rbind y do.call
# combinamos los data frames contenidos en nlista como un único data frame
data <- do.call(rbind, nlista)
dim(data)
```

```
[1] 1140 6
```

```
str(data)
```

```
'data.frame': 1140 obs. of 6 variables:
 $ Date : Date, format: "2017-08-18" "2017-08-18" ...
 $ HomeTeam: chr "Leganes" "Valencia" "Celta" "Girona" ...
 $ AwayTeam: chr "Alaves" "Las Palmas" "Sociedad" "Ath Madrid" ...
 $ FTHG : int 1 1 2 2 1 0 2 0 1 0 ...
 $ FTAG : int 0 0 3 2 1 0 0 3 0 1 ...
 $ FTR : chr "H" "H" "A" "D" ...
```

```
tail(data)
```

	Date	HomeTeam	AwayTeam	FTHG	FTAG	FTR
1135	2020-07-19	Espanol	Celta	0	0	D
1136	2020-07-19	Granada	Ath Bilbao	4	0	H
1137	2020-07-19	Leganes	Real Madrid	2	2	D
1138	2020-07-19	Levante	Getafe	1	0	H
1139	2020-07-19	Osasuna	Mallorca	2	2	D
1140	2020-07-19	Sevilla	Valencia	1	0	H

```
View(data)
```

```
summary(data)
```

	Date	HomeTeam	AwayTeam	FTHG
Min.	:2017-08-18	Length:1140	Length:1140	Min. :0.000
1st Qu.	:2018-03-17	Class :character	Class :character	1st Qu.:1.000
Median	:2019-01-16	Mode :character	Mode :character	Median :1.000
Mean	:2019-01-15			Mean :1.479
3rd Qu.	:2019-10-27			3rd Qu.:2.000

```

Max.      :2020-07-19
          FTR
Min.      :0.000   Length:1140
1st Qu.:0.000   Class :character
Median :1.000   Mode  :character
Mean    :1.108
3rd Qu.:2.000
Max.     :6.000

```

```

#Con ayuda de la función table obtenemos las estimaciones de
# probabilidades solicitadas
(pcasa <- round(table(data$FTHG)/dim(data)[1], 3)) # Probabilidades

```

```

      0      1      2      3      4      5      6      7      8
0.232 0.327 0.267 0.112 0.035 0.019 0.005 0.001 0.001

```

```

# marginales estimadas para los equipos que juegan en casa

(pvisita <- round(table(data$FTAG)/dim(data)[1], 3)) # Probabilidades

```

```

      0      1      2      3      4      5      6
0.352 0.340 0.212 0.054 0.029 0.010 0.003

```

```

# marginales estimadas para los equipos que juegan como visitante

(pcta <- round(table(data$FTHG, data$FTAG)/dim(data)[1], 3)) # Probabilidades

```

```

      0      1      2      3      4      5      6
0 0.078 0.081 0.046 0.018 0.005 0.004 0.000
1 0.116 0.115 0.068 0.018 0.009 0.002 0.000
2 0.088 0.094 0.061 0.011 0.009 0.002 0.002
3 0.045 0.032 0.025 0.006 0.002 0.002 0.001
4 0.014 0.011 0.007 0.000 0.004 0.000 0.000
5 0.009 0.005 0.004 0.000 0.001 0.000 0.000
6 0.003 0.002 0.000 0.001 0.000 0.000 0.000
7 0.000 0.001 0.000 0.000 0.000 0.000 0.000
8 0.000 0.000 0.001 0.000 0.000 0.000 0.000

```

```

# conjuntas estimadas para los partidos

```

```

pcasa <- as.data.frame(pcasa)
str(pcasa)

```

```

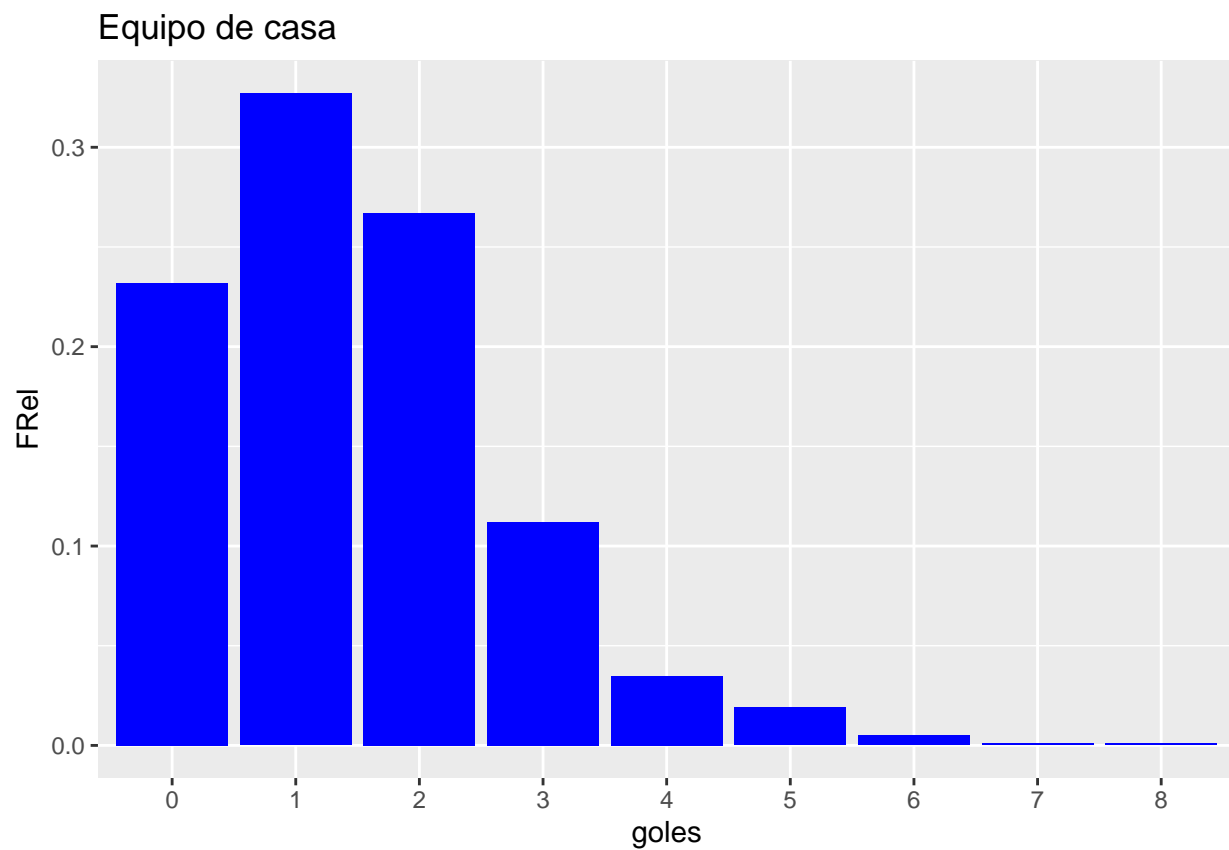
'data.frame':   9 obs. of  2 variables:
 $ Var1: Factor w/ 9 levels "0","1","2","3",...: 1 2 3 4 5 6 7 8 9
 $ Freq: num  0.232 0.327 0.267 0.112 0.035 0.019 0.005 0.001 0.001

```

```
pcasa <- pcasa %>% rename(goles = Var1, FRel = Freq)
tail(pcasa)
```

```
  goles  FRel
4      3 0.112
5      4 0.035
6      5 0.019
7      6 0.005
8      7 0.001
9      8 0.001
```

```
p <- ggplot(pcasa, aes(x = goles, y = FRel)) +
  geom_bar (stat="identity", fill = 'blue') +
  ggtitle('Equipo de casa')
p
```

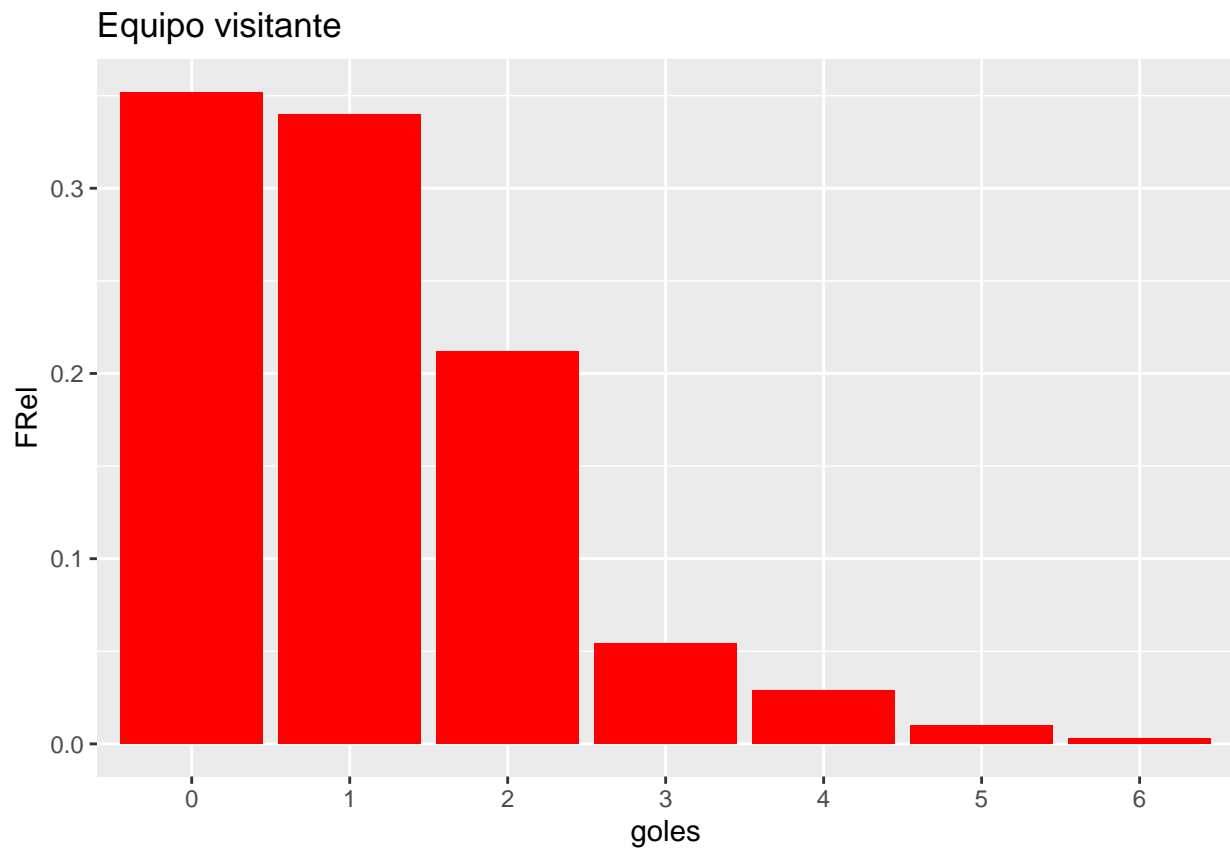


```
pvisita <- as.data.frame(pvisita)
pvisita <- rename(pvisita, goles = Var1, FRel = Freq)
tail(pvisita)
```

```
  goles  FRel
2      1 0.340
3      2 0.212
4      3 0.054
```

```
5    4 0.029
6    5 0.010
7    6 0.003
```

```
p <- ggplot(pvisita, aes(x = goles, y = FRel)) +
  geom_bar (stat="identity", fill = 'red') +
  ggtitle('Equipo visitante')
p
```



```
pcta <- melt(pcta) # Función del paquete reshape2
pcta <- rename(pcta, gcasa = Var1, gvisita = Var2, ProbEst = value)
pcta %>% ggplot(aes(gcasa, gvisita)) +
  geom_tile(aes(fill = ProbEst)) +
  ggtitle('Probabilidades conjuntas estimadas') +
  scale_fill_gradient(low = 'white', high = 'purple') +
  theme(axis.text.x = element_text(angle = 90, hjust = 0))
```

