# CONTENIDO

# 1. Análisis Exploratorio

```
setwd("C:\\Users\\81799\\OneDrive\\Documentos\\ESFM_CLASES\\Servicio Social ARTF\\Machine Learni
data <- read.csv("data/tema2/auto-mpg.csv",
                 header = TRUE,
                 stringsAsFactors = FALSE)
```

Reemplazo los valores (3,4,5,6,8) de la columna cylinders

```
data$cylinders <- factor(data$cylinders,
                         levels = c(3,4,5,6,8),
                         labels = c("3cil", "4cil", "5cil", "6cil", "8cil"))
```

Haciendo un resumen de nuestro data frame

```
summary(data)
```

```
##        No              mpg         cylinders   displacement     horsepower
##  Min.   :  1.0   Min.   : 9.00   3cil:  4   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:100.2   1st Qu.:17.50   4cil:204   1st Qu.:104.2   1st Qu.: 76.0
##  Median :199.5   Median :23.00   5cil:  3   Median :148.5   Median : 92.0
##  Mean   :199.5   Mean   :23.51   6cil: 84   Mean   :193.4   Mean   :104.1
##  3rd Qu.:298.8   3rd Qu.:29.00   8cil:103   3rd Qu.:262.0   3rd Qu.:125.0
##  Max.   :398.0   Max.   :46.60              Max.   :455.0   Max.   :230.0
##      weight      acceleration     model_year      car_name
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Length:398
##  1st Qu.:2224   1st Qu.:13.82   1st Qu.:73.00   Class :character
##  Median :2804   Median :15.50   Median :76.00   Mode  :character
##  Mean   :2970   Mean   :15.57   Mean   :76.01
##  3rd Qu.:3608   3rd Qu.:17.18   3rd Qu.:79.00
##  Max.   :5140   Max.   :24.80   Max.   :82.00
```

Observamos que para:

- Columna de clase factor nos hace un conteo de las veces que aparece el valor.
- Columna de clase numeric nos da los 6 estadísticos básicos.

La función *str()* nos da una idea inicial de como está organizado el Data Frame.

```
str(data)
```

```
## 'data.frame': 398 obs. of  9 variables:
##  $ No          : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ mpg         : num  28 19 36 28 21 23 15.5 32.9 16 13 ...
##  $ cylinders   : Factor w/ 5 levels "3cil","4cil",..: 2 1 2 2 4 2 5 2 4 5 ...
##  $ displacement: num  140 70 107 97 199 115 304 119 250 318 ...
##  $ horsepower  : int  90 97 75 92 90 95 120 100 105 150 ...
##  $ weight      : int  2264 2330 2205 2288 2648 2694 3962 2615 3897 3755 ...
##  $ acceleration: num  15.5 13.5 14.5 17 15 15 13.9 14.8 18.5 14 ...
##  $ model_year  : int  71 72 82 72 70 75 76 81 75 76 ...
##  $ car_name    : chr  "chevrolet vega 2300" "mazda rx2 coupe" "honda accord" "datsun 510 (sw)
```

## 1.1. Estadísticos y medidas básicas.

```
X=data$mpg
```

### 1.1.1. Medidas de tendencia central

Una medida de tendencia central es un número obtenido de un conjunto de datos, que tiende a posicionar el centro del conjunto de datos

## Media Aritmética

Para el caso de un conjunto de **n** datos pertenecientes a una **muestra**, la media aritmética se denota como $\bar{x}$, y se define de la siguiente manera:

$$\bar{x} = \frac{\sum_{i=1}^{n} x_i}{n}$$

```
mean(X) #sum(X)/length(X)

## [1] 23.51457
```

## Mediana

La mediana, denotada como Me, es la medida de tendencia central que se posiciona justamente al centro de los datos. Para calcular el valor de la mediana se debe seguir la siguiente secuencia, sin importar si el conjunto de datos es de una muestra o de una población.

1.

2. Ordenar los datos de menor a mayor.

   *a*) **Si n es impar:** Si el conjunto de datos en consideración es una cantidad impar, entonces la mediana será el valor del dato más central perteneciente al conjunto de datos ordenados, y se encuentra en la posición

   $$\frac{n+1}{2}$$

   *b*) **Si n es par:** Si el conjunto de datos en consideración es una cantidad par, entonces la mediana será el valor del promedio de los dos datos más centrales.

$$P(X \leq m) = 0.5$$

```
median(X)

## [1] 23
```

### Percentiles

El **percentil** es una medida de posición usada en estadística que indica, una vez ordenados los datos de menor a mayor, el valor de la variable por debajo del cual se encuentra un *porcentaje* dado de observaciones en un grupo.

$$P(X \leq x_p) = p \qquad\qquad p \in [0,1]$$

```
quantile(X)
```

```
##   0%  25%  50%  75% 100%
##  9.0 17.5 23.0 29.0 46.6
```

## 1.1.2. Medidas de Variabilidad

Las medidas de variabilidad nos permite conocer que tan dispersas se encuentran las observaciones a cada lado del centro de una serie de datos, o bien que tan alejadas se encuentran de la media de tendencia central.
Una medida de variabilidad es un número que indica el grado de dispersión (esparcimiento) en un conjunto de datos con respecto a un estadístico de tendencia central (por lo general, la media aritmética).

### Varianza (muestral)

Esta es un estadístico que se define como el promedio de las desviaciones con respecto a la media.

$$s^2 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$$

```
var(X)
```

```
## [1] 61.08961
```

### Desviación típica (muestral)

La desviación estándar es una medida de variación absoluta que nos permite concluir que tan grande o pequeña es la dispersión de los datos.(Es la raíz cuadrada de la varianza)

$$s = \sqrt{s^2} = \sqrt{\frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}}$$

```
sd(X)
```

```
## [1] 7.815984
```

### Coeficiente de variación.

El Coeficiente de Variación (CV) es la medida relativa que permite tener una idea general de la magnitud de la desviación estándar en relación con la magnitud de la media aritmética.

$$CV = \frac{s}{\bar{x}} \cdot 100\,\%$$

```
sd(X)/mean(X)*100

## [1] 33.2389
```

## 1.2. Subconjunto de datos

Índices por posición

```
data[1:5, 8:9]

##   model_year            car_name
## 1         71 chevrolet vega 2300
## 2         72     mazda rx2 coupe
## 3         82        honda accord
## 4         72     datsun 510 (sw)
## 5         70         amc gremlin

data[1:5,c(8,9)]

##   model_year            car_name
## 1         71 chevrolet vega 2300
## 2         72     mazda rx2 coupe
## 3         82        honda accord
## 4         72     datsun 510 (sw)
## 5         70         amc gremlin
```

Índices por nombre

```
data[1:5,c("model_year", "car_name")]

##   model_year            car_name
## 1         71 chevrolet vega 2300
## 2         72     mazda rx2 coupe
## 3         82        honda accord
## 4         72     datsun 510 (sw)
## 5         70         amc gremlin
```

Por condiciones

```
#Operadores lógicos
# & : AND
# | : OR
# ! : NOT
# == : Igualdad
data[data$mpg == max(data$mpg ) | data$mpg == min(data$mpg), ]

##      No  mpg cylinders displacement horsepower weight acceleration model_year
## 190 190  9.0      8cil          304        193   4732         18.5         70
## 269 269 46.6      4cil           86         65   2110         17.9         80
##      car_name
## 190  hi 1200d
## 269 mazda glc
```

Filtro por condiciones

```
data[ data$mpg> 30 & data$cylinders == "6cil" , c("car_name", "mpg") ]
```

```
##                              car_name  mpg
## 12                       volvo diesel 30.7
## 300 oldsmobile cutlass ciera (diesel) 38.0
## 364                     datsun 280-zx 32.7
```

La versión reducida sirve que con solo las primeras 3 letras del nombre de la columna R infiere a que columna te estas refiriendo

```
data[ data$mpg> 30 & data$cyl == "6cil" , c("car_name", "mpg") ]
```

```
##                              car_name  mpg
## 12                       volvo diesel 30.7
## 300 oldsmobile cutlass ciera (diesel) 38.0
## 364                     datsun 280-zx 32.7
```

### Subset

```
subset(data, mpg>30 & cylinders =="6cil" , select = c("car_name", "mpg"))
```

```
##                              car_name  mpg
## 12                       volvo diesel 30.7
## 300 oldsmobile cutlass ciera (diesel) 38.0
## 364                     datsun 280-zx 32.7
```

Excluir columnas

```
data[1:5,-c(1,9)]
```

```
##   mpg cylinders displacement horsepower weight acceleration model_year
## 1  28      4cil          140         90   2264         15.5         71
## 2  19      3cil           70         97   2330         13.5         72
## 3  36      4cil          107         75   2205         14.5         82
## 4  28      4cil           97         92   2288         17.0         72
## 5  21      6cil          199         90   2648         15.0         70
```

```
data[1:5, !names(data) %in% c("No", "car_name")]
```

```
##   mpg cylinders displacement horsepower weight acceleration model_year
## 1  28      4cil          140         90   2264         15.5         71
## 2  19      3cil           70         97   2330         13.5         72
## 3  36      4cil          107         75   2205         14.5         82
## 4  28      4cil           97         92   2288         17.0         72
## 5  21      6cil          199         90   2648         15.0         70
```

## 1.3. Divisiones con split

La función *split()* lo que hace es dividir grupos basados en un factor o bien en en un vector, tiene la función inversa que se llama *unsplit()* que hace el efecto revertido.

```
setwd("C:\\Users\\81799\\OneDrive\\Documentos\\ESFM_CLASES\\Servicio Social ARTF\\Machine Learni
data <- read.csv("data/tema2/auto-mpg.csv",
                 header = TRUE,
                 stringsAsFactors = FALSE)
carslit <- split(data, data$cylinders)
carslit[1] #Accedemos a un valor clase lista

## $`3`
##      No  mpg cylinders displacement horsepower weight acceleration model_year
## 2     2 19.0         3           70         97   2330         13.5         72
## 199 199 18.0         3           70         90   2124         13.5         73
## 251 251 23.7         3           70        100   2420         12.5         80
## 365 365 21.5         3           80        110   2720         13.5         77
##            car_name
## 2   mazda rx2 coupe
## 199       maxda rx3
## 251   mazda rx-7 gs
## 365      mazda rx-4

class(carslit)

## [1] "list"

carslit[[1]] #Así nos devuelve el valor interno del Data Frame

##      No  mpg cylinders displacement horsepower weight acceleration model_year
## 2     2 19.0         3           70         97   2330         13.5         72
## 199 199 18.0         3           70         90   2124         13.5         73
## 251 251 23.7         3           70        100   2420         12.5         80
## 365 365 21.5         3           80        110   2720         13.5         77
##            car_name
## 2   mazda rx2 coupe
## 199       maxda rx3
## 251   mazda rx-7 gs
## 365      mazda rx-4

class(carslit[[1]])

## [1] "data.frame"

names(carslit[[1]])

## [1] "No"           "mpg"          "cylinders"    "displacement" "horsepower"
## [6] "weight"       "acceleration" "model_year"   "car_name"
```

## 1.4. Gráficos

```
setwd("C:\\Users\\81799\\OneDrive\\Documentos\\ESFM_CLASES\\Servicio Social ARTF\\Machine Learni
auto <- read.csv("data/tema2/auto-mpg.csv",
                 header = TRUE,
                 stringsAsFactors = FALSE)
auto$cylinders <- factor(auto$cylinders ,
                         levels = c(3,4,5,6,8),
                         labels = c("3cil", "4cil", "5cil","6cil", "8cil")
```

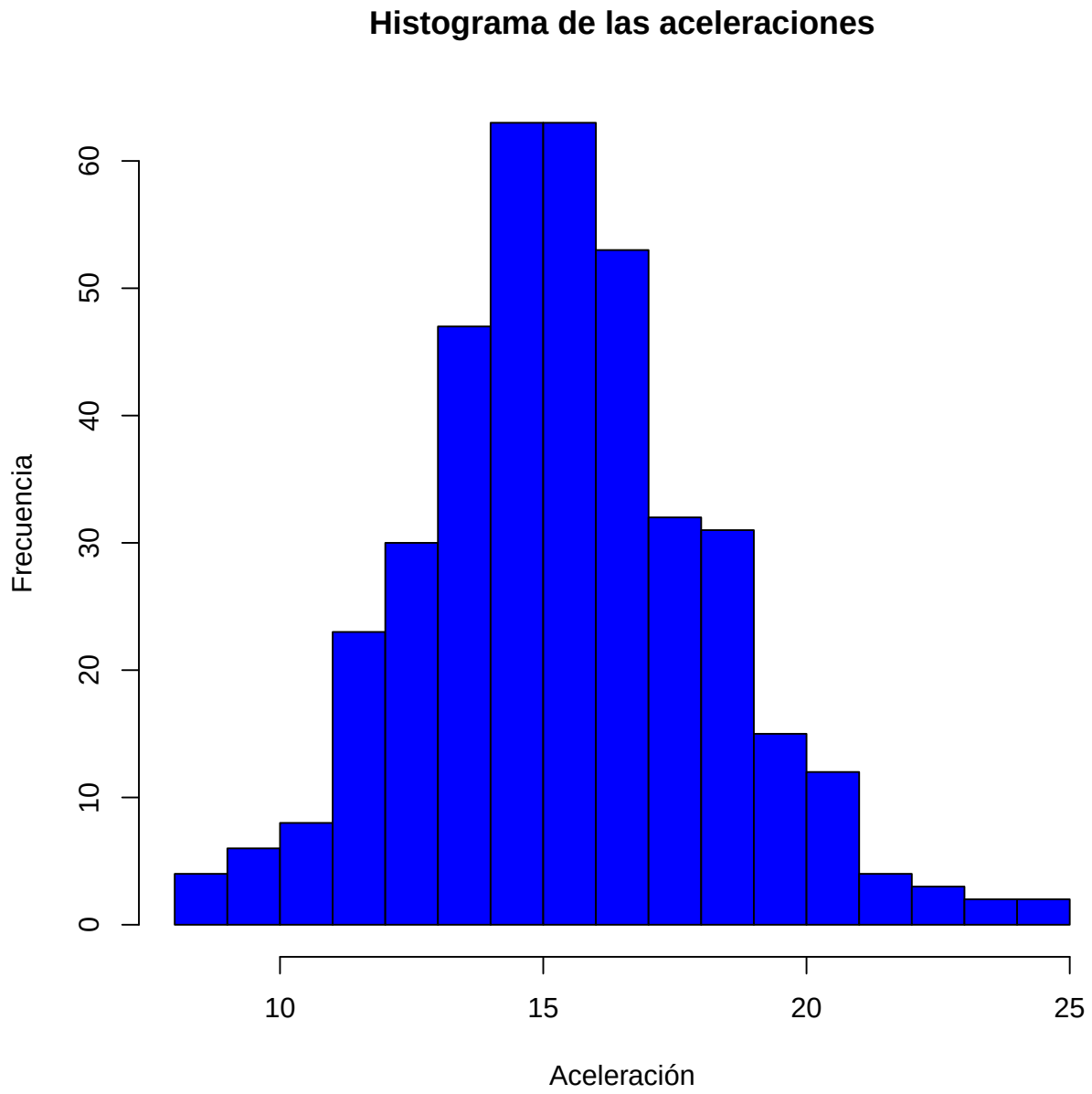
```
                        )
```

La función *attach()* hace que el Data Frame forme parte de la estructura principal de R, esto significa que cada vez que necesite acceder a la columna basta con colocar su nombre

```
attach(auto)
```

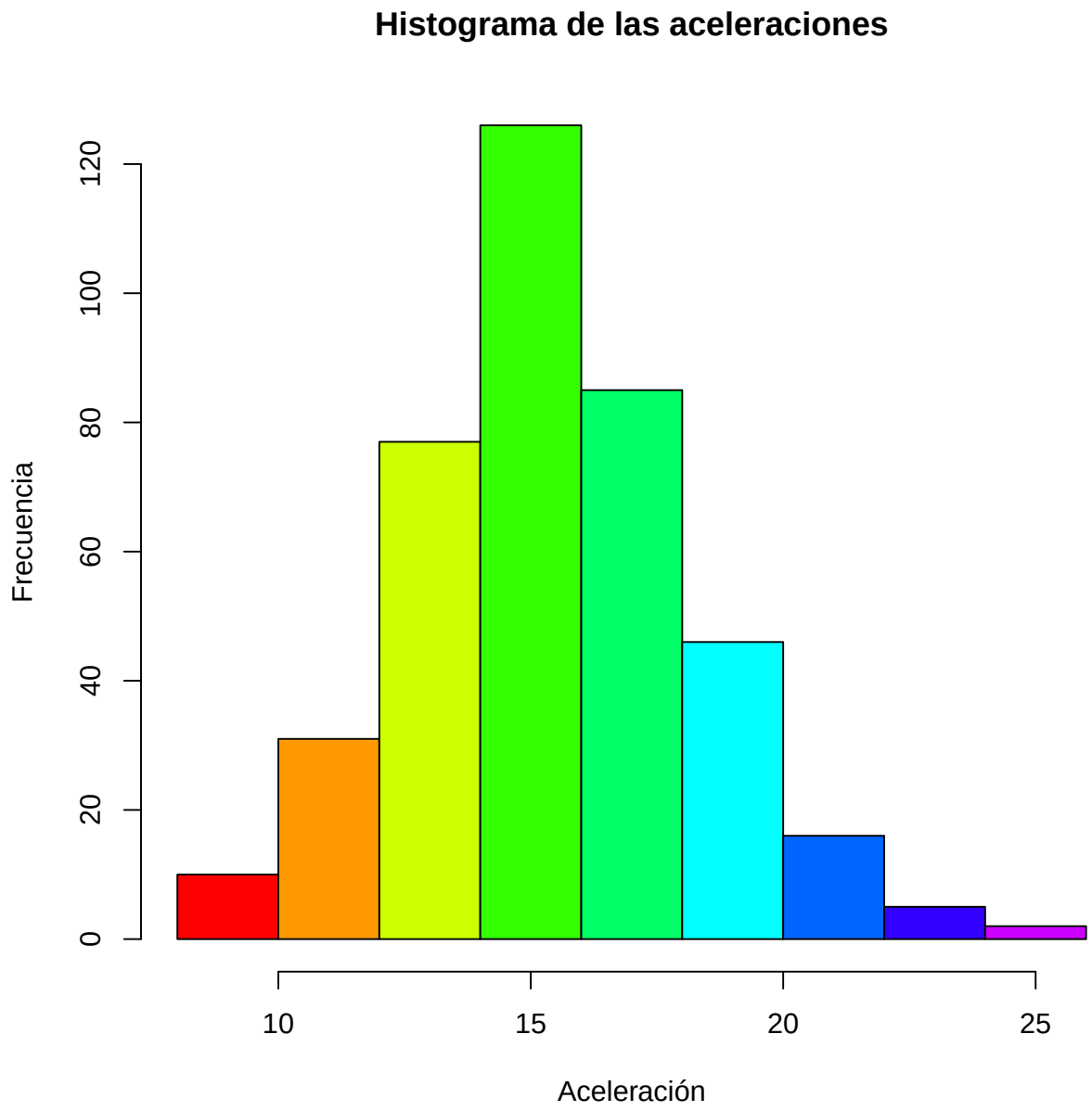
### 1.4.1. Histograma

```
hist(acceleration,
     col ="blue", xlab = "Aceleración", ylab = "Frecuencia",
     main = "Histograma de las aceleraciones", breaks = 16 )
```



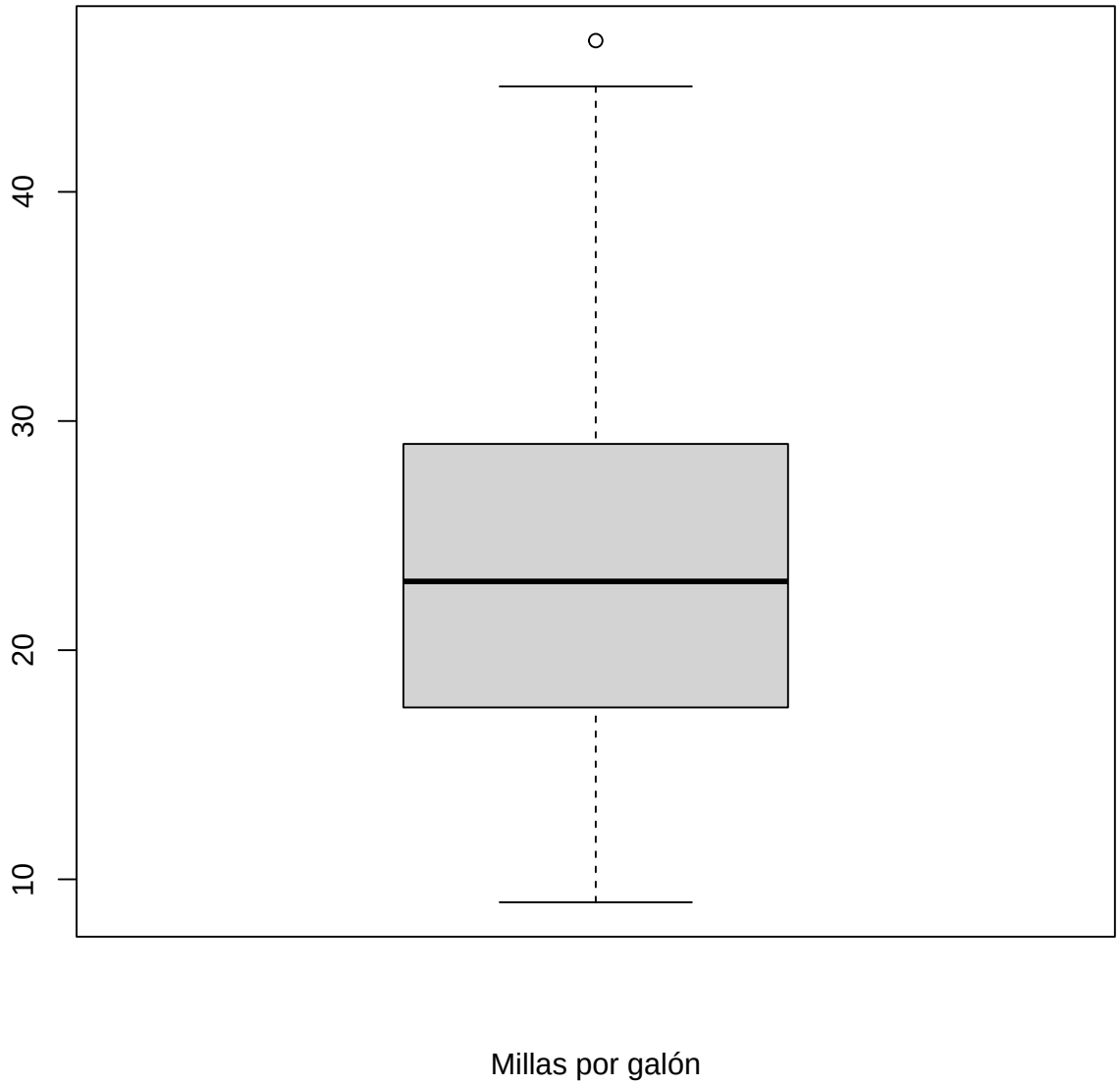

```
#Nota que ya no hubo necesidad de hacer auto$acceleration
```

```
hist(acceleration,
     col =rainbow(10) , xlab = "Aceleración", ylab = "Frecuencia",
     main = "Histograma de las aceleraciones", breaks = 10 )
```



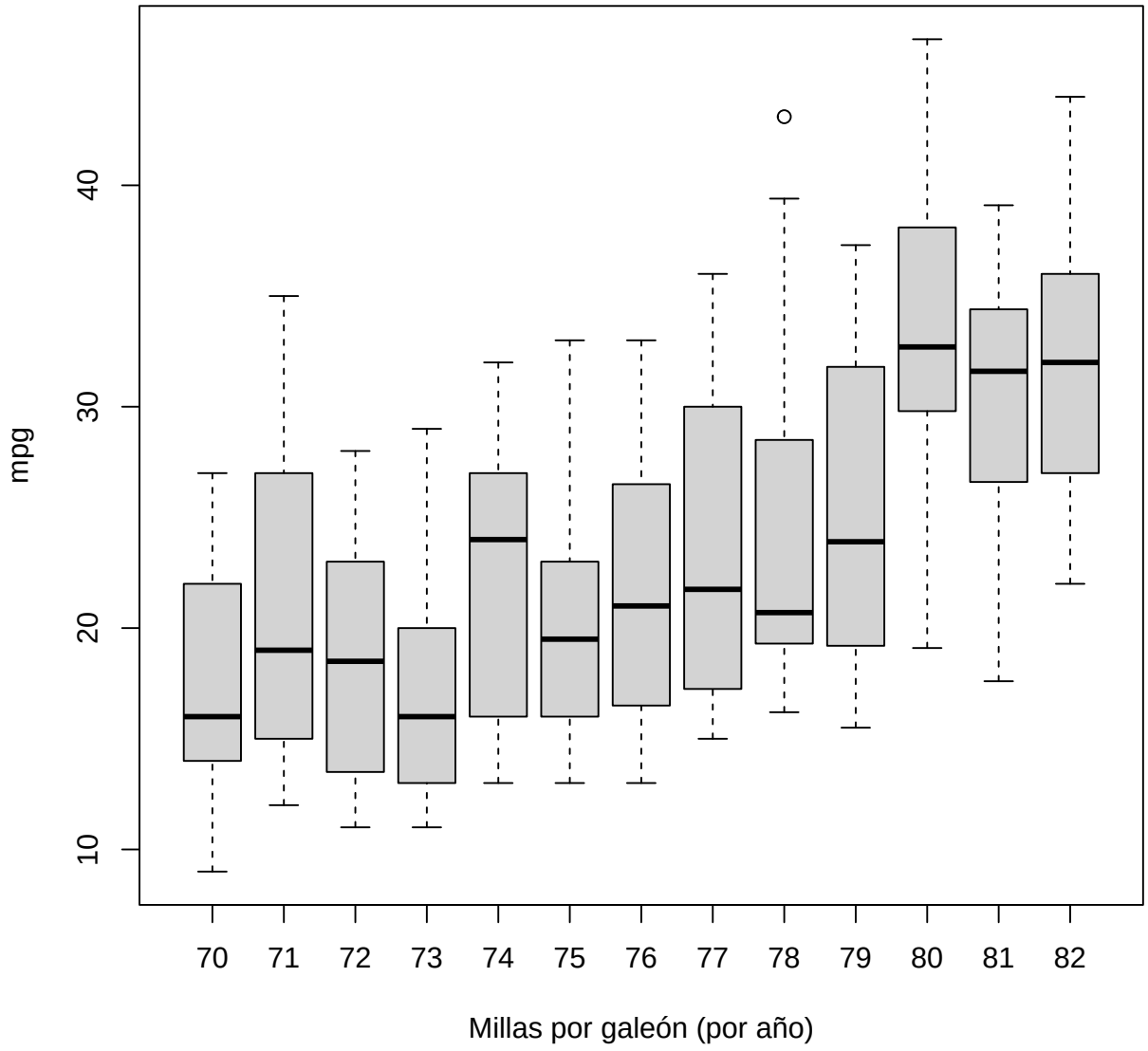

### 1.4.2. Diagrama de cajas

```
boxplot(mpg, xlab ="Millas por galón")
```



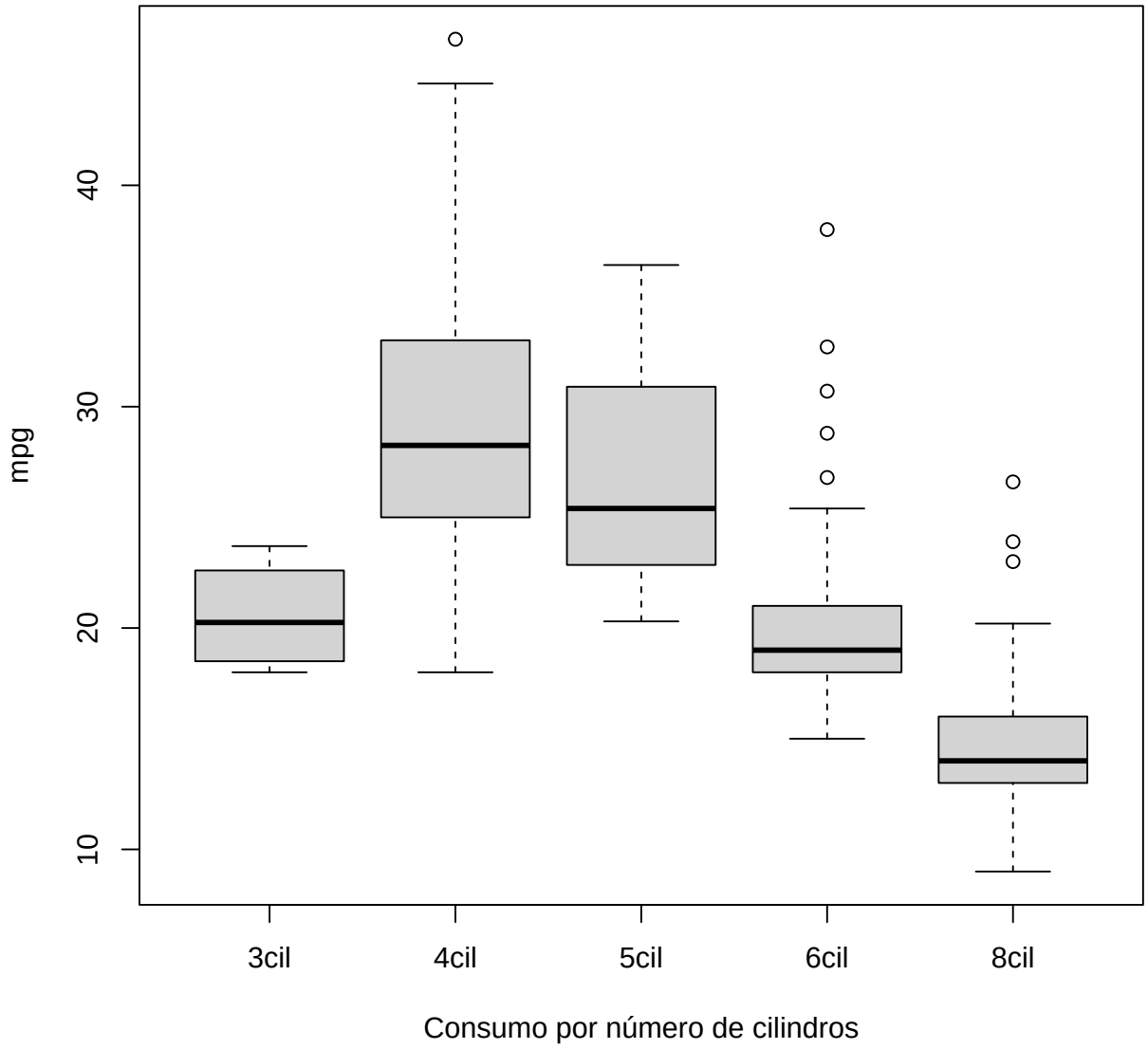

Para representar las millas por galeón pero por año, hacemos los siguiente:

```
boxplot(mpg~model_year, xlab = "Millas por galeón (por año)" )
```




Consumo por número de cilindros

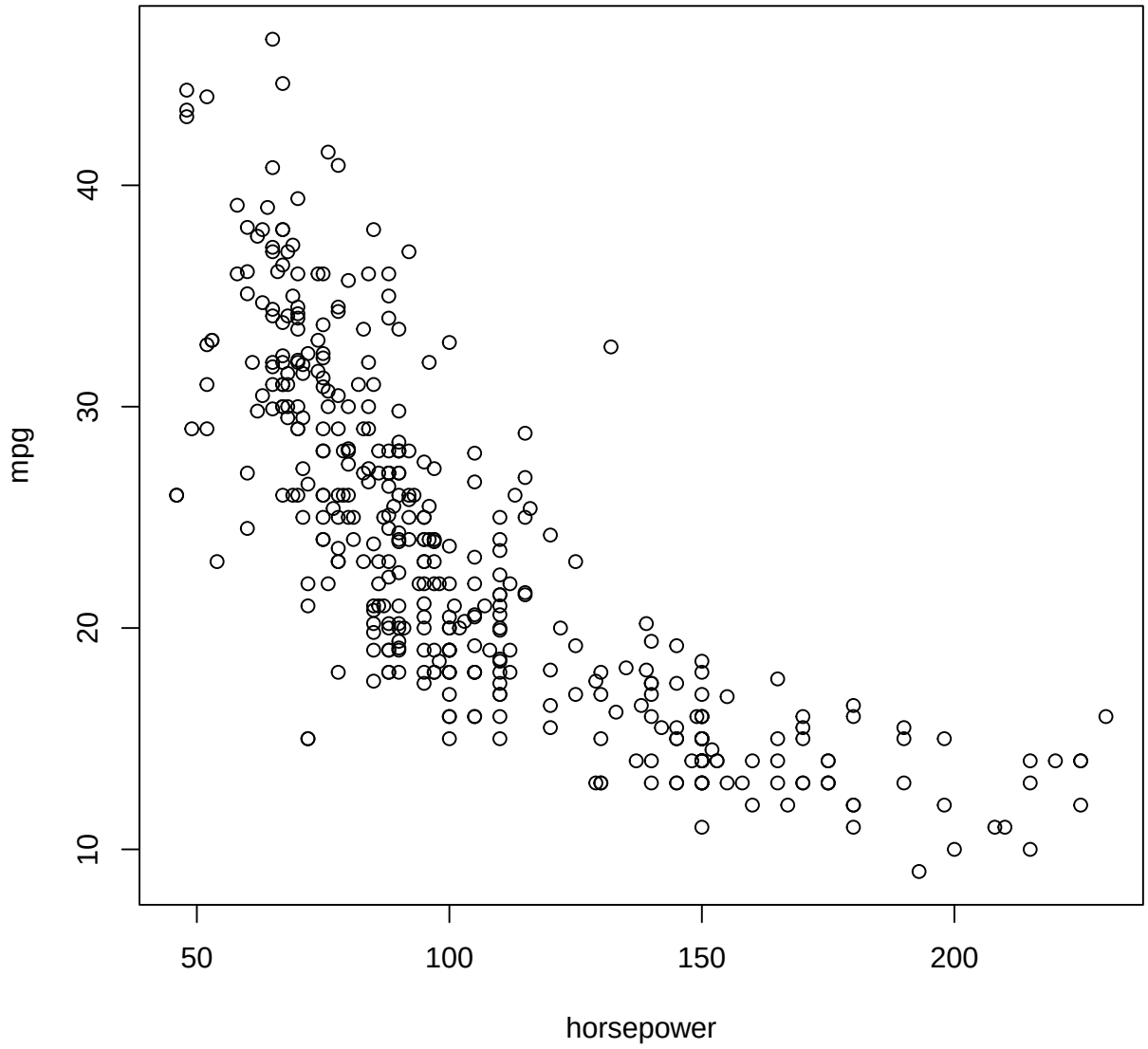
```
boxplot(mpg~cylinders , xlab = "Consumo por número de cilindros" )
```




### 1.4.3. Scatterplot

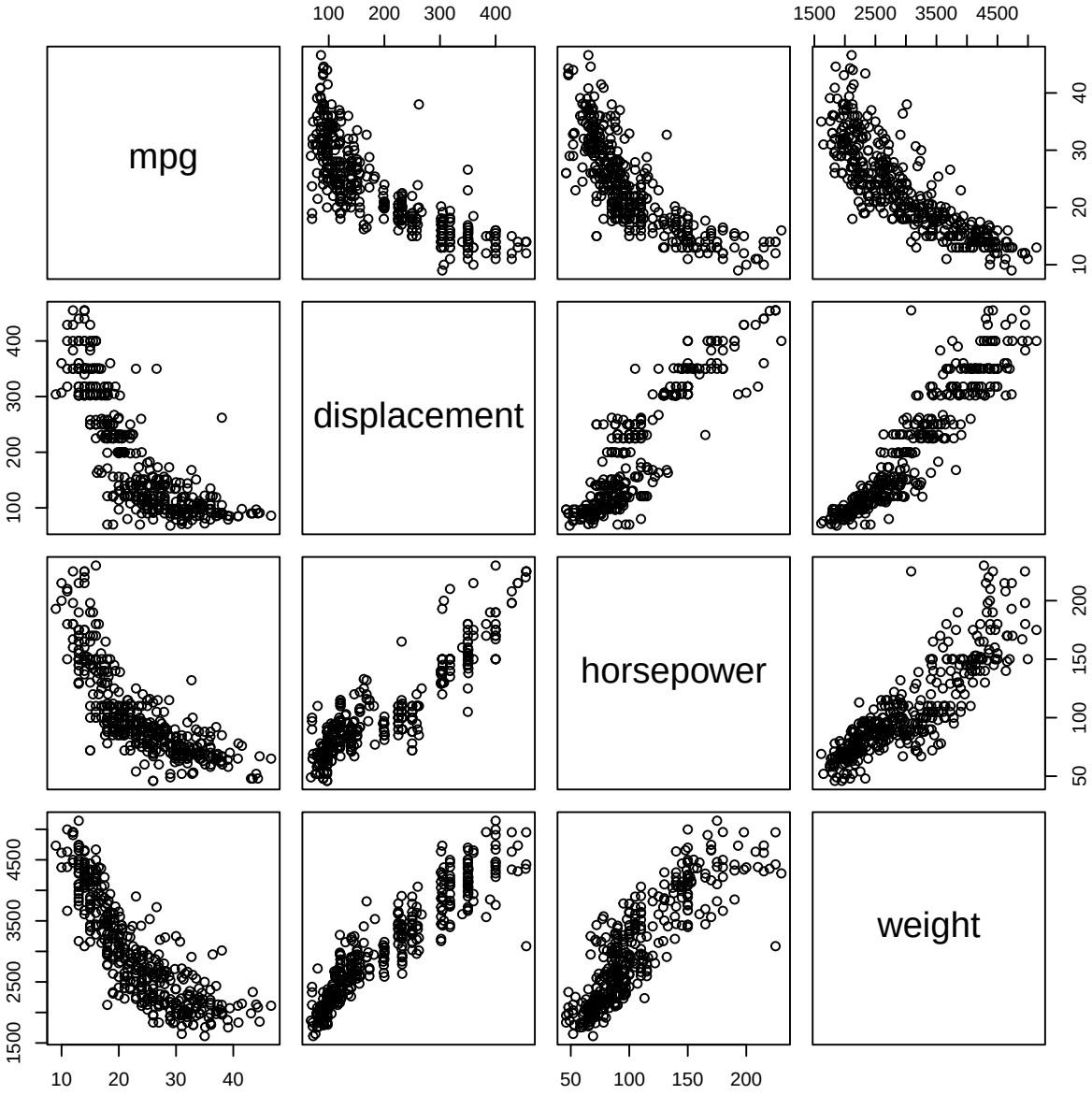
Millas por galeón (Variable dependiente ) en función de los caballos de fuerza (Variable independiente )

```
plot(mpg ~ horsepower)
```




### Matriz de Scatterplots

```
pairs(~mpg + displacement + horsepower +weight )
```



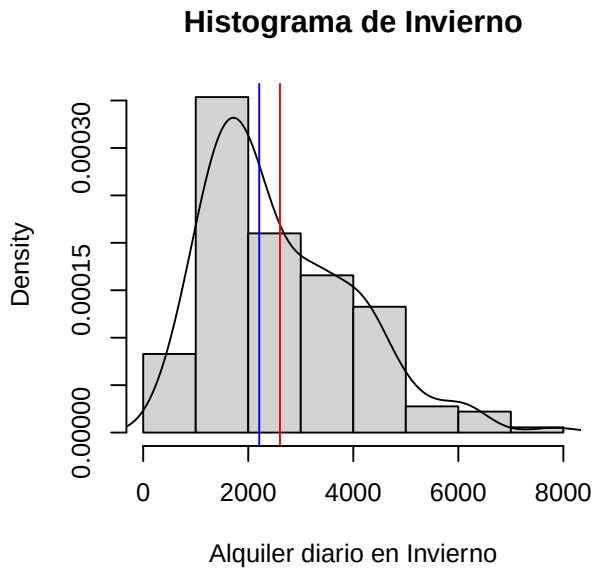

### 1.4.4. Comparación a través de representaciones

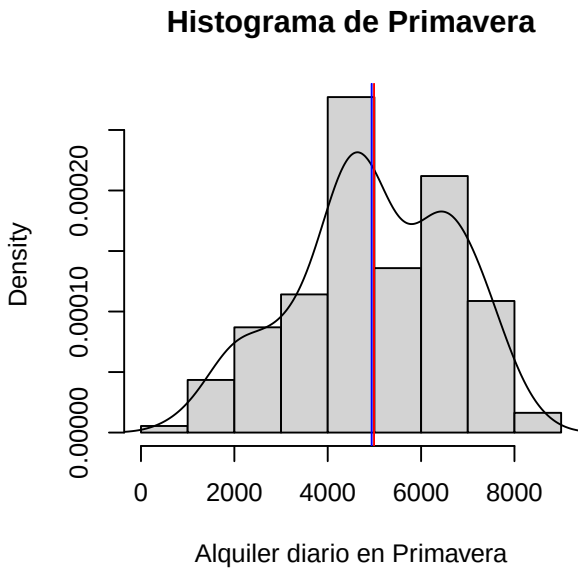
```
setwd("C:\\Users\\81799\\OneDrive\\Documentos\\ESFM_CLASES\\Servicio Social ARTF\\Machine Learni
Bicicletas <- read.csv("data/tema2/daily-bike-rentals.csv")
Bicicletas$season <- factor(Bicicletas$season, levels = c(1,2,3,4),
                            labels = c("Invierno","Primavera","Verano","Otoño"))
Bicicletas$workingday <- factor(Bicicletas$workingday, levels = c(0,1), labels = c("Festivo", "I
Bicicletas$weathersit <-  factor(Bicicletas$weathersit , levels = c(1,2,3),
                                 labels = c("Despejado", "Nublado", "Lluvia"))
attach(Bicicletas) #Para guardarlo en R y evitar poner Bicicletas$columna
```

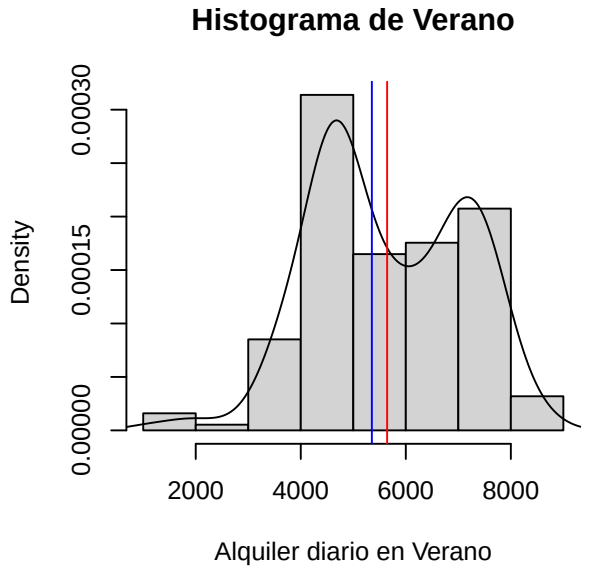
```
winter <- subset(Bicicletas , season == "Invierno")$cnt
spring <- subset(Bicicletas , season == "Primavera")$cnt
summer <- subset(Bicicletas , season == "Verano")$cnt
fall <- subset(Bicicletas , season == "Otoño")$cnt
```

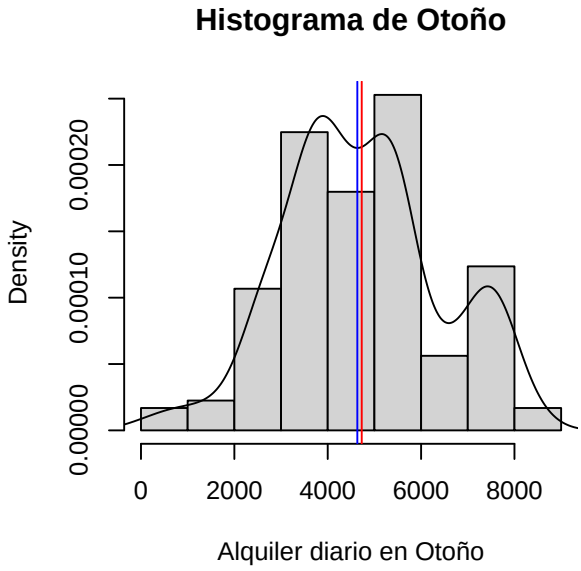
Haciendo los histogramas, tendremos que:

```
par(mfrow = c(2,2)) #4 gráficos en forma matricial 2x2
hist(winter, prob = T, xlab = "Alquiler diario en Invierno",
     main = "Histograma de Invierno")
lines(density(winter)) #Graficamos la distribucion
abline(v = mean(winter), col = "red") #Nos indica el valor de la media
abline(v = median(winter), col = "blue") #Nos indica el valor de la mediana

hist(spring, prob = T, xlab = "Alquiler diario en Primavera",
     main = "Histograma de Primavera")
lines(density(spring)) #Graficamos la distribucion
abline(v = mean(spring), col = "red") #Nos indica el valor de la media
abline(v = median(spring), col = "blue") #Nos indica el valor de la mediana

hist(summer, prob = T, xlab = "Alquiler diario en Verano",
     main = "Histograma de Verano")
lines(density(summer)) #Graficamos la distribucion
abline(v = mean(summer), col = "red") #Nos indica el valor de la media
abline(v = median(summer), col = "blue") #Nos indica el valor de la mediana

hist(fall, prob = T, xlab = "Alquiler diario en Otoño",
     main = "Histograma de Otoño")
lines(density(fall)) #Graficamos la distribucion
abline(v = mean(fall), col = "red") #Nos indica el valor de la media
abline(v = median(fall), col = "blue") #Nos indica el valor de la mediana
```




```r
par(mfrow = c(1,1)) #Para que nos haga solo una gráfica (Normal)
```

### 1.4.5. Diagrama Cuantil-Cuantil

Los diagramas cuantll-cuantil son una herramienta de exploración utilizada para evaluar las similitudes entre la distribución de una variable numérica y una distribución normal, o entre las distribuciones de dos variables numéricas.
Existen dos tipos de diagramas cuantil-cuantil:

1. **Diagrama cuantil-cuantil normales**: Estos se construyen trazando los cuantiles de una variable numérica respecto de los cuantiles de una distribución normal.

2. **Diagrama cuantil-cuantil generales**: Estos se construyen trazando los cuantiles de una variable numérica respecto de los cuantiles de una segunda variable numérica.

**Nota:** Si las distribuciones de los cuantiles comparados son idénticas, los puntos del diagrama formaran una linea recta de 45 grados. Cuanto más lejos se desvíen los puntos del diagrama de una linea recta, menos similares serán las distribuciones comparadas.