Tipologia y ciclo de vida de los datos - PRA2

Autor: Eduardo Diaz Villanueva

Enero 2021

Contents

Lectura de datos

```
library(stringr)
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
```

```
library(ggplot2)

# Limpiamos la aplicacion de datos anteriores y cargo el fichero.
rm(list = ls())
datos <- read.csv("winequality-red.csv", sep=",")
datos_originales <- datos
head(datos,5)
```

```
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.4             0.70        0.00            1.9     0.076
## 2           7.8             0.88        0.00            2.6     0.098
## 3           7.8             0.76        0.04            2.3     0.092
## 4          11.2             0.28        0.56            1.9     0.075
## 5           7.4             0.70        0.00            1.9     0.076
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  11                   34  0.9978 3.51      0.56     9.4
## 2                  25                   67  0.9968 3.20      0.68     9.8
## 3                  15                   54  0.9970 3.26      0.65     9.8
## 4                  17                   60  0.9980 3.16      0.58     9.8
## 5                  11                   34  0.9978 3.51      0.56     9.4
##   quality
## 1       5
## 2       5
## 3       5
## 4       6
## 5       5
```

Descripción del dataset.

El dataset seleccionado contiene 11 variables que describen las propiedades quimicas de un vino, como puede ser la acided, ph nivel de azucar, etc… estas variables tendran influencia en la calidad final del vino.

Con este ejercicio queremos estudiar que variables son mas representativas y encontrar modelos que puedan predecir la calidad del vino.

Si pensamos por ejemplo en una industria, podriamos reducir el tiempo y coste reduciendo el numero de pruebas de calidad a las variables mas significativas. Incluso mejorar la calidad del producto final, focalizando esfuerzos y recursos a reducir la variabiliad de las varibles que mas contribuyan a la calidad final.

Integracion y seleccion de los datos de interes

Realizaremos un primer analisis estadistico para familiarizarnos con las variables.

```r
summary(datos)
```

```
##  fixed.acidity   volatile.acidity  citric.acid    residual.sugar 
##  Min.   : 4.60   Min.   :0.1200   Min.   :0.000   Min.   : 0.900  
##  1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090   1st Qu.: 1.900  
##  Median : 7.90   Median :0.5200   Median :0.260   Median : 2.200  
##  Mean   : 8.32   Mean   :0.5278   Mean   :0.271   Mean   : 2.539  
##  3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420   3rd Qu.: 2.600  
##  Max.   :15.90   Max.   :1.5800   Max.   :1.000   Max.   :15.500  
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density      
##  Min.   :0.01200   Min.   : 1.00       Min.   :  6.00       Min.   :0.9901  
##  1st Qu.:0.07000   1st Qu.: 7.00       1st Qu.: 22.00       1st Qu.:0.9956  
##  Median :0.07900   Median :14.00       Median : 38.00       Median :0.9968  
##  Mean   :0.08747   Mean   :15.87       Mean   : 46.47       Mean   :0.9967  
##  3rd Qu.:0.09000   3rd Qu.:21.00       3rd Qu.: 62.00       3rd Qu.:0.9978  
##  Max.   :0.61100   Max.   :72.00       Max.   :289.00       Max.   :1.0037  
##        pH          sulphates         alcohol         quality     
##  Min.   :2.740   Min.   :0.3300   Min.   : 8.40   Min.   :3.000  
##  1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000  
##  Median :3.310   Median :0.6200   Median :10.20   Median :6.000  
##  Mean   :3.311   Mean   :0.6581   Mean   :10.42   Mean   :5.636  
##  3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000  
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :8.000  
```

```
str(datos)
```

```
## 'data.frame':    1599 obs. of  12 variables:
##  $ fixed.acidity       : num  7.4 7.8 7.8 11.2 7.4 7.4 7.9 7.3 7.8 7.5 ...
##  $ volatile.acidity    : num  0.7 0.88 0.76 0.28 0.7 0.66 0.6 0.65 0.58 0.5 ...
##  $ citric.acid         : num  0 0 0.04 0.56 0 0 0.06 0 0.02 0.36 ...
##  $ residual.sugar      : num  1.9 2.6 2.3 1.9 1.9 1.8 1.6 1.2 2 6.1 ...
##  $ chlorides           : num  0.076 0.098 0.092 0.075 0.076 0.075 0.069 0.065 0.073 0.071 ...
##  $ free.sulfur.dioxide : num  11 25 15 17 11 13 15 15 9 17 ...
##  $ total.sulfur.dioxide: num  34 67 54 60 34 40 59 21 18 102 ...
##  $ density             : num  0.998 0.997 0.997 0.998 0.998 ...
##  $ pH                  : num  3.51 3.2 3.26 3.16 3.51 3.51 3.3 3.39 3.36 3.35 ...
##  $ sulphates           : num  0.56 0.68 0.65 0.58 0.56 0.56 0.46 0.47 0.57 0.8 ...
##  $ alcohol             : num  9.4 9.8 9.8 9.8 9.4 9.4 9.4 10 9.5 10.5 ...
##  $ quality             : int  5 5 5 6 5 5 5 7 7 5 ...
```

```
#Tipo de dato asignado a cada campo
sapply(datos, function(x) class(x))
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##            "numeric"            "numeric"            "numeric"
##       residual.sugar            chlorides  free.sulfur.dioxide
##            "numeric"            "numeric"            "numeric"
## total.sulfur.dioxide              density                   pH
##            "numeric"            "numeric"            "numeric"
##            sulphates              alcohol              quality
##            "numeric"            "numeric"            "integer"
```

Limpieza de los datos

Elementos vacios

Analizamos los valores para detectar falta o ausencia de daos

```
# Analizamos la existencia de datos NA
colSums(is.na(datos))
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                    0
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
##                    0                    0                    0
##            sulphates              alcohol              quality
##                    0                    0                    0
```

```
# Analizamos la existencia de datos vacios
colSums(datos=="")
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                    0
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
##                    0                    0                    0
##            sulphates              alcohol              quality
##                   0                   0                   0
```

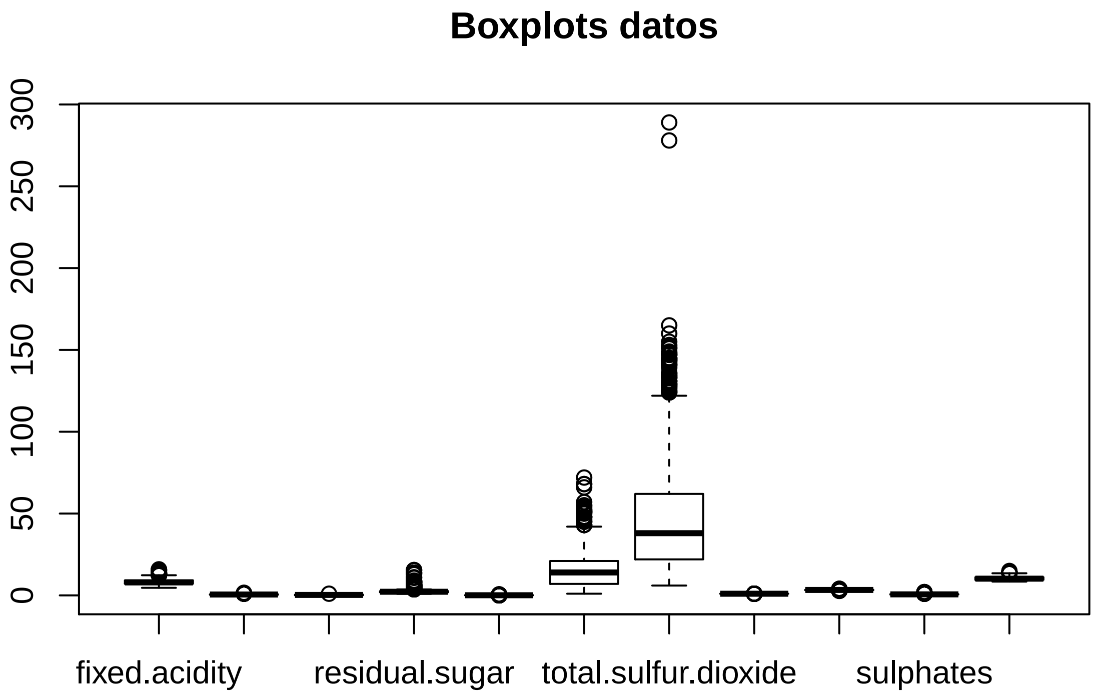
```
# Analizamos la existencia de datos con valor 0
colSums(datos==0)
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                  132
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
##                    0                    0                    0
##            sulphates              alcohol              quality
##                    0                    0                    0
```

Observamos la variable Citric.acid con una gran cantidad de valores 0.

Valores extremos

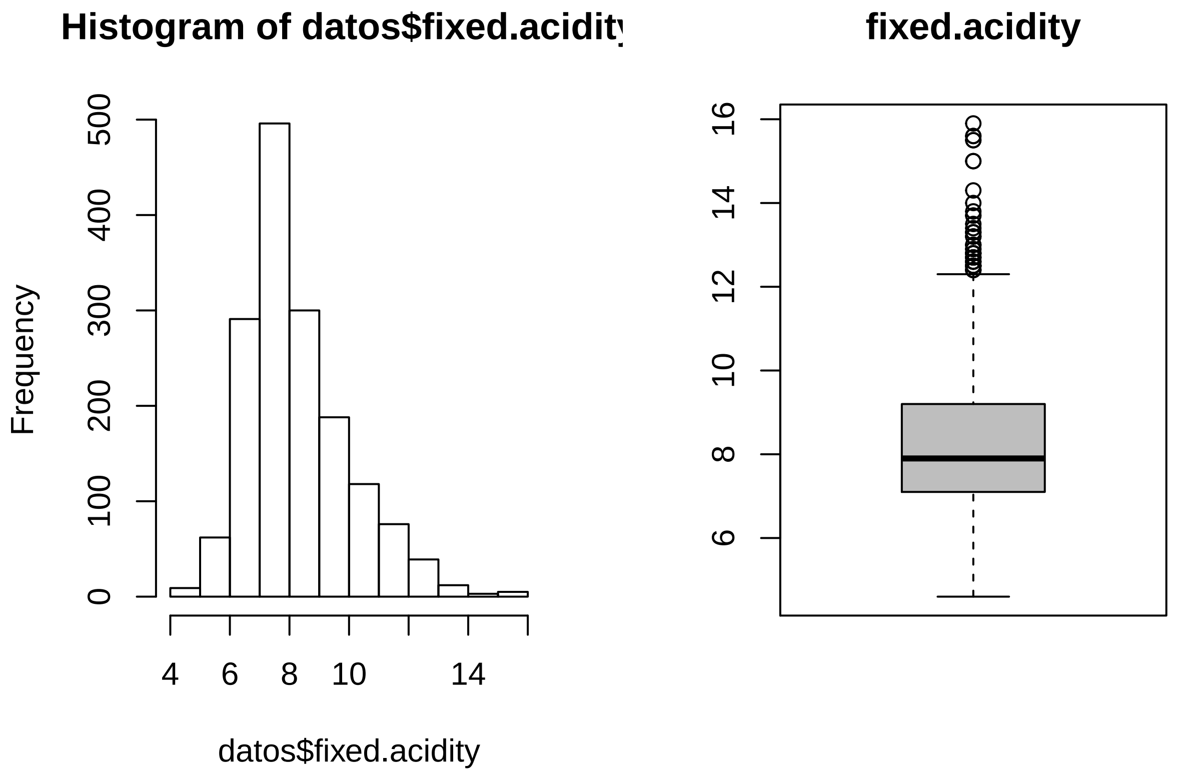
```
boxplot(x=datos[,1:11],main="Boxplots datos")
```



```
#boxplot(x=datos[,1:5],main="Boxplots datos")
#boxplot(x=datos[,6:7],main="Boxplots datos")
#boxplot(x=datos[,8:12],main="Boxplots datos")


par(mfrow=c(1,2))
hist(datos$fixed.acidity)
```

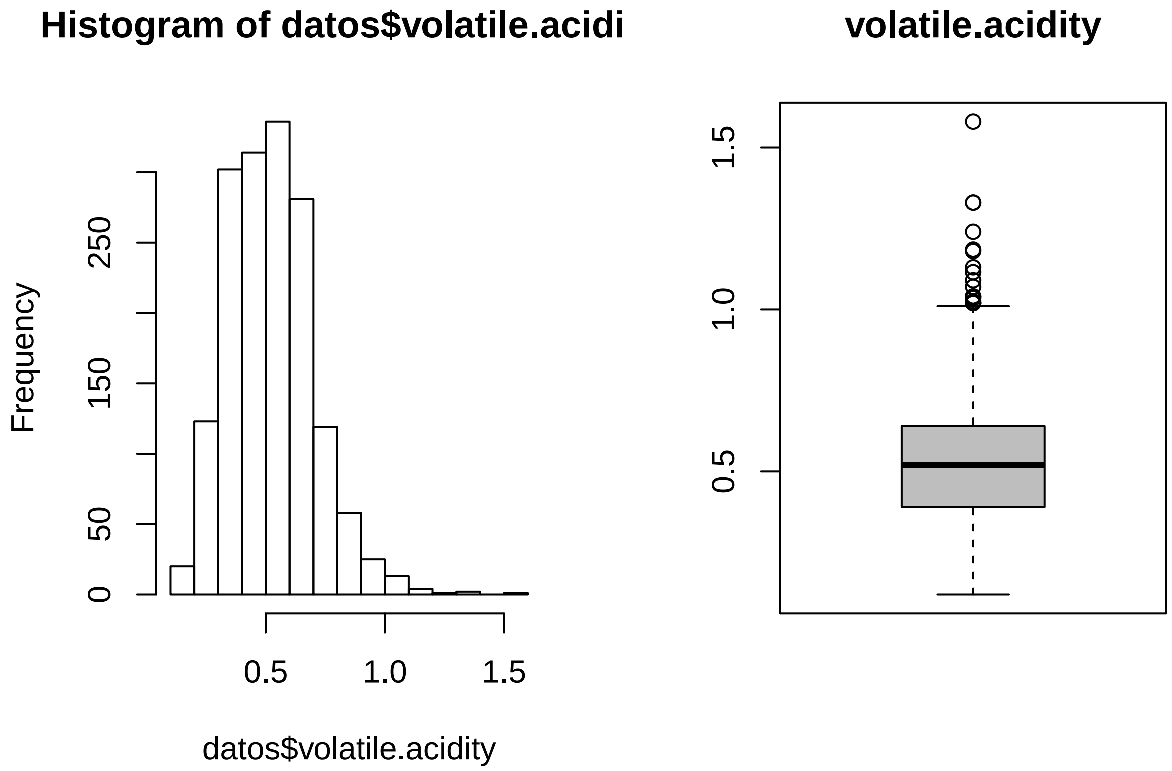
```
boxplot(datos$fixed.acidity,main="fixed.acidity", col="gray")
```



```
boxplot.stats(datos$fixed.acidity)$out
```

```
##  [1] 12.8 12.8 15.0 15.0 12.5 13.3 13.4 12.4 12.5 13.8 13.5 12.6 12.5 12.8 12.8
## [16] 14.0 13.7 13.7 12.7 12.5 12.8 12.6 15.6 12.5 13.0 12.5 13.3 12.4 12.5 12.9
## [31] 14.3 12.4 15.5 15.5 15.6 13.0 12.7 13.0 12.7 12.4 12.7 13.2 13.2 13.2 15.9
## [46] 13.3 12.9 12.6 12.6
```

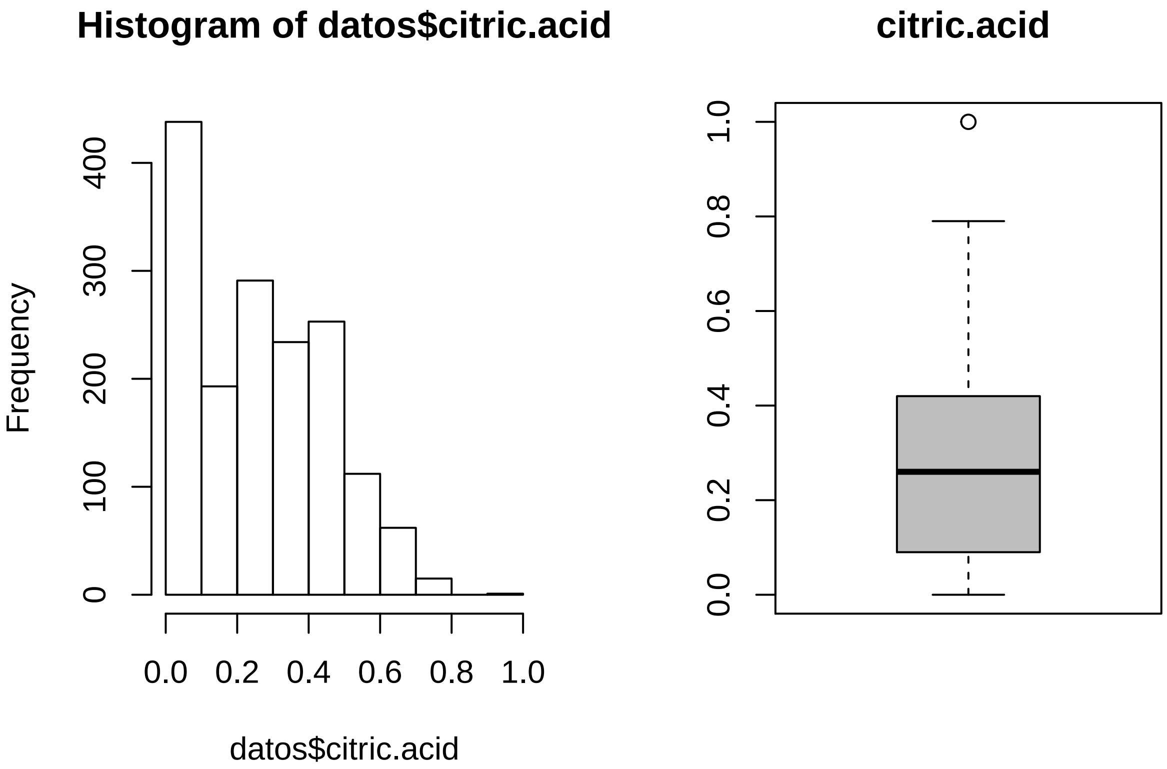
```
par(mfrow=c(1,2))
hist(datos$volatile.acidity)
boxplot(datos$volatile.acidity,main="volatile.acidity", col="gray")
```



```
boxplot.stats(datos$volatile.acidity)$out
```

```
##  [1] 1.130 1.020 1.070 1.330 1.330 1.040 1.090 1.040 1.240 1.185 1.020 1.035
## [13] 1.025 1.115 1.020 1.020 1.580 1.180 1.040
```

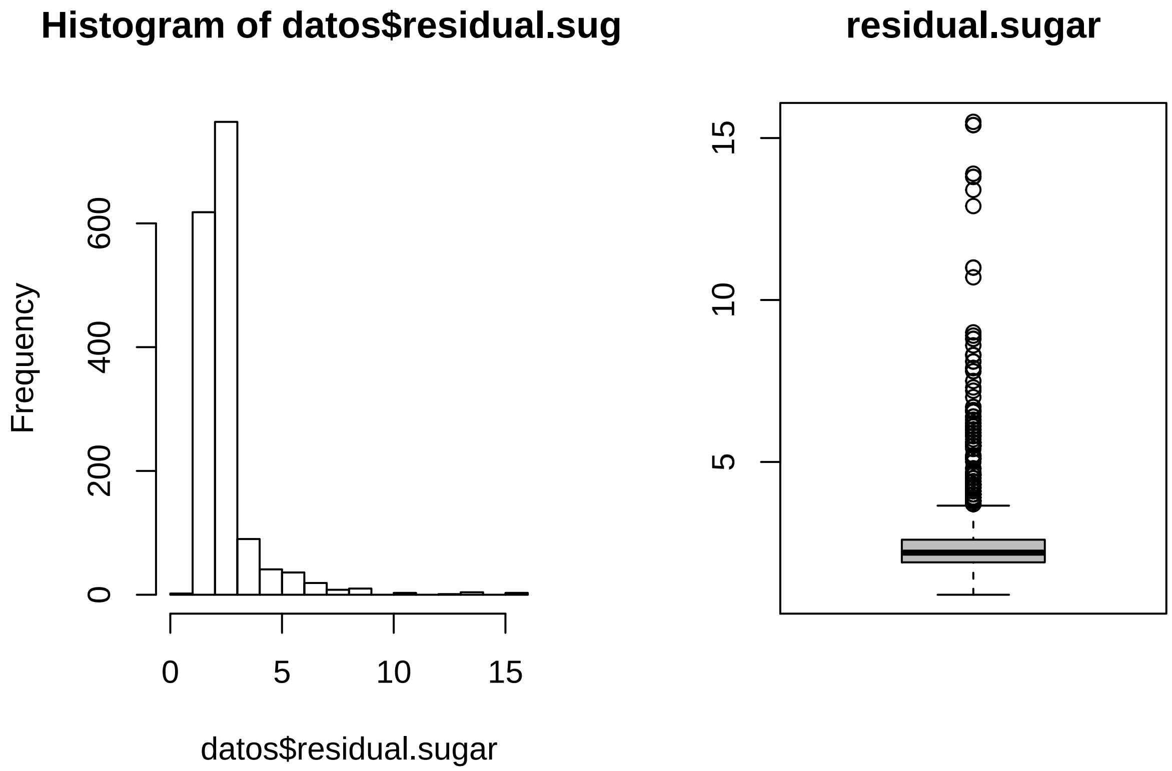
```
par(mfrow=c(1,2))
hist(datos$citric.acid )
boxplot(datos$citric.acid ,main="citric.acid ", col="gray")
```



```
boxplot.stats(datos$citric.acid )$out
```

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

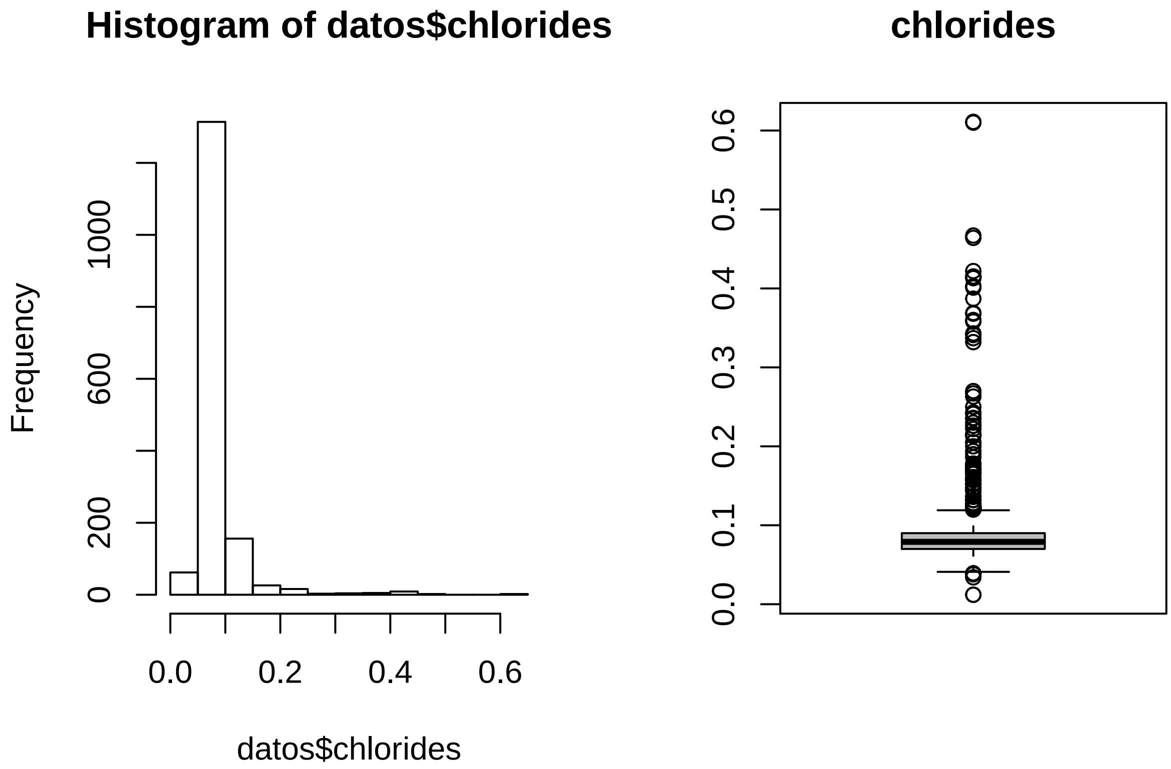
```
par(mfrow=c(1,2))
hist(datos$residual.sugar)
boxplot(datos$residual.sugar,main="residual.sugar", col="gray")
```



```
boxplot.stats(datos$residual.sugar)$out
```

```
##   [1]  6.10  6.10  3.80  3.90  4.40 10.70  5.50  5.90  5.90  3.80  5.10  4.65
##  [13]  4.65  5.50  5.50  5.50  5.50  7.30  7.20  3.80  5.60  4.00  4.00  4.00
##  [25]  4.00  7.00  4.00  4.00  6.40  5.60  5.60 11.00 11.00  4.50  4.80  5.80
##  [37]  5.80  3.80  4.40  6.20  4.20  7.90  7.90  3.70  4.50  6.70  6.60  3.70
##  [49]  5.20 15.50  4.10  8.30  6.55  6.55  4.60  6.10  4.30  5.80  5.15  6.30
##  [61]  4.20  4.20  4.60  4.20  4.60  4.30  4.30  7.90  4.60  5.10  5.60  5.60
##  [73]  6.00  8.60  7.50  4.40  4.25  6.00  3.90  4.20  4.00  4.00  4.00  6.60
##  [85]  6.00  6.00  3.80  9.00  4.60  8.80  8.80  5.00  3.80  4.10  5.90  4.10
##  [97]  6.20  8.90  4.00  3.90  4.00  8.10  8.10  6.40  6.40  8.30  8.30  4.70
## [109]  5.50  5.50  4.30  5.50  3.70  6.20  5.60  7.80  4.60  5.80  4.10 12.90
## [121]  4.30 13.40  4.80  6.30  4.50  4.50  4.30  4.30  3.90  3.80  5.40  3.80
## [133]  6.10  3.90  5.10  5.10  3.90 15.40 15.40  4.80  5.20  5.20  3.75 13.80
## [145] 13.80  5.70  4.30  4.10  4.10  4.40  3.70  6.70 13.90  5.10  7.80
```

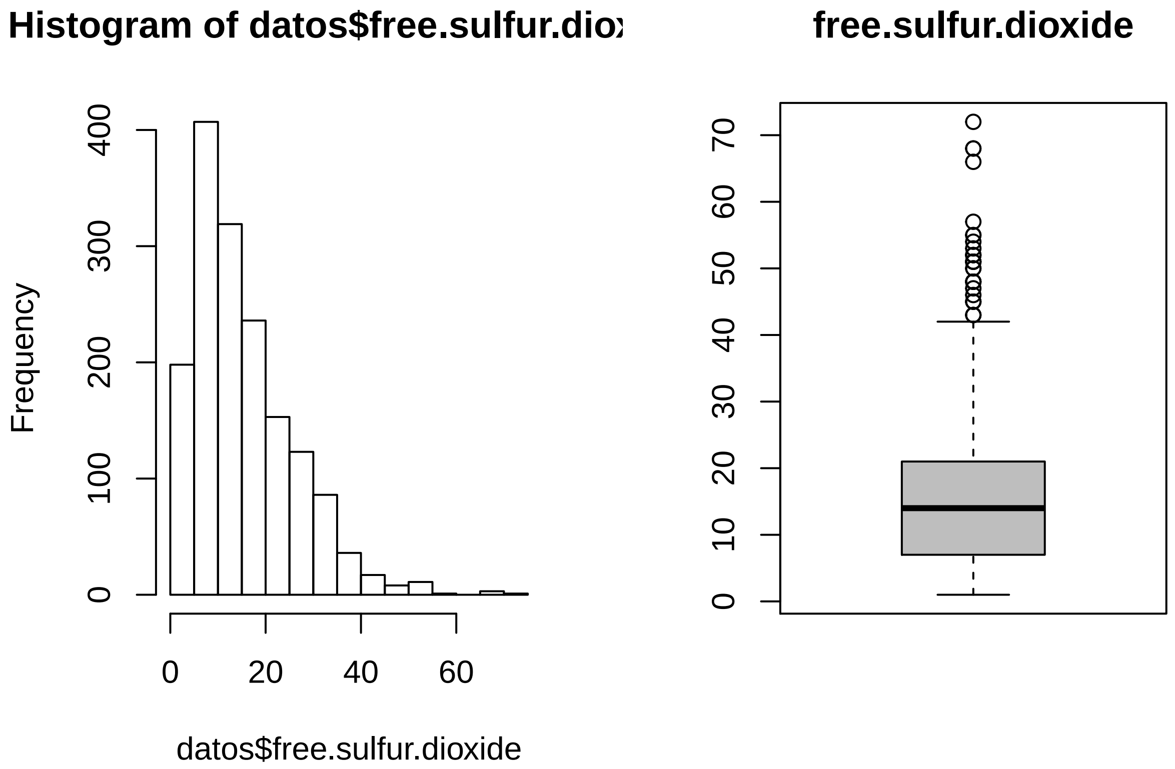
```
par(mfrow=c(1,2))
hist(datos$chlorides)
boxplot(datos$chlorides,main="chlorides", col="gray")
```



```r
boxplot.stats(datos$chlorides)$out
```

```
##   [1] 0.176 0.170 0.368 0.341 0.172 0.332 0.464 0.401 0.467 0.122 0.178 0.146
##  [13] 0.236 0.610 0.360 0.270 0.039 0.337 0.263 0.611 0.358 0.343 0.186 0.213
##  [25] 0.214 0.121 0.122 0.122 0.128 0.120 0.159 0.124 0.122 0.122 0.174 0.121
##  [37] 0.127 0.413 0.152 0.152 0.125 0.122 0.200 0.171 0.226 0.226 0.250 0.148
##  [49] 0.122 0.124 0.124 0.143 0.222 0.039 0.157 0.422 0.034 0.387 0.415 0.157
##  [61] 0.157 0.243 0.241 0.190 0.132 0.126 0.038 0.165 0.145 0.147 0.012 0.012
##  [73] 0.039 0.194 0.132 0.161 0.120 0.120 0.123 0.123 0.414 0.216 0.171 0.178
##  [85] 0.369 0.166 0.166 0.136 0.132 0.132 0.123 0.123 0.123 0.403 0.137 0.414
##  [97] 0.166 0.168 0.415 0.153 0.415 0.267 0.123 0.214 0.214 0.169 0.205 0.205
## [109] 0.039 0.235 0.230 0.038
```

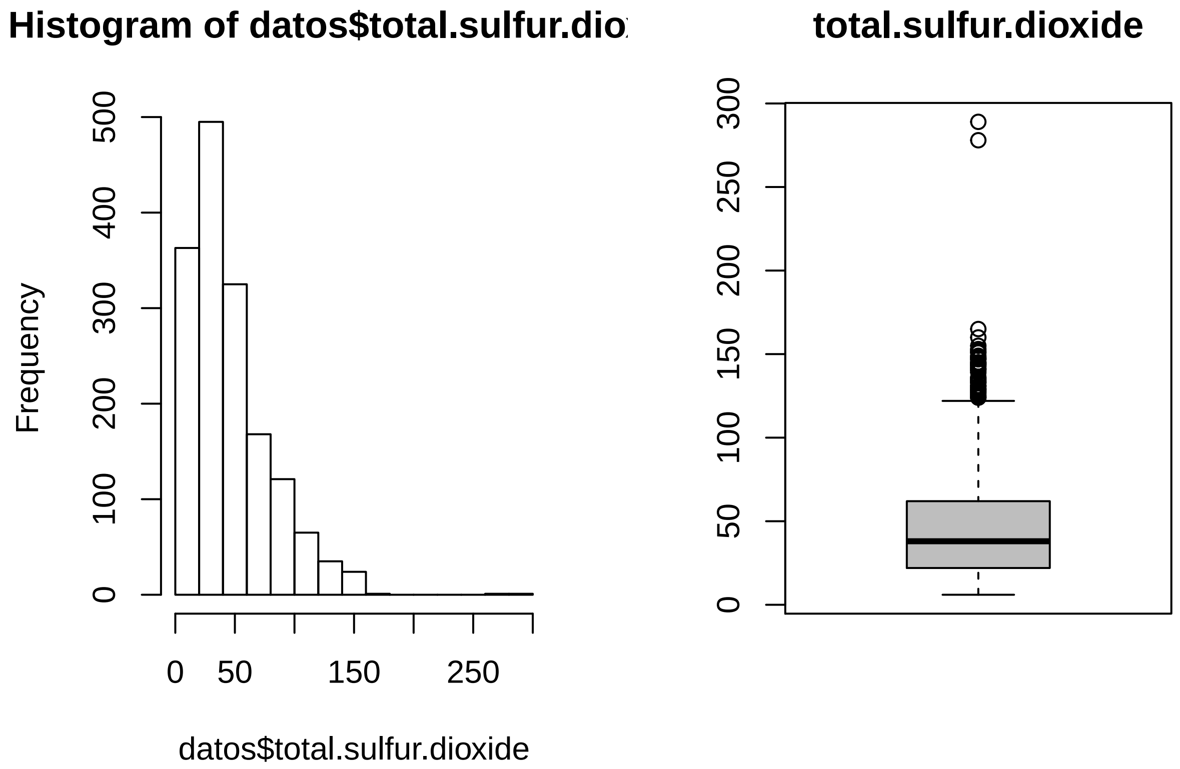
```r
par(mfrow=c(1,2))
hist(datos$free.sulfur.dioxide)
boxplot(datos$free.sulfur.dioxide,main="free.sulfur.dioxide", col="gray")
```



```
boxplot.stats(datos$free.sulfur.dioxide)$out
```

```
##  [1] 52 51 50 68 68 43 47 54 46 45 53 52 51 45 57 50 45 48 43 48 72 43 51 51 52
## [26] 55 55 48 48 66
```

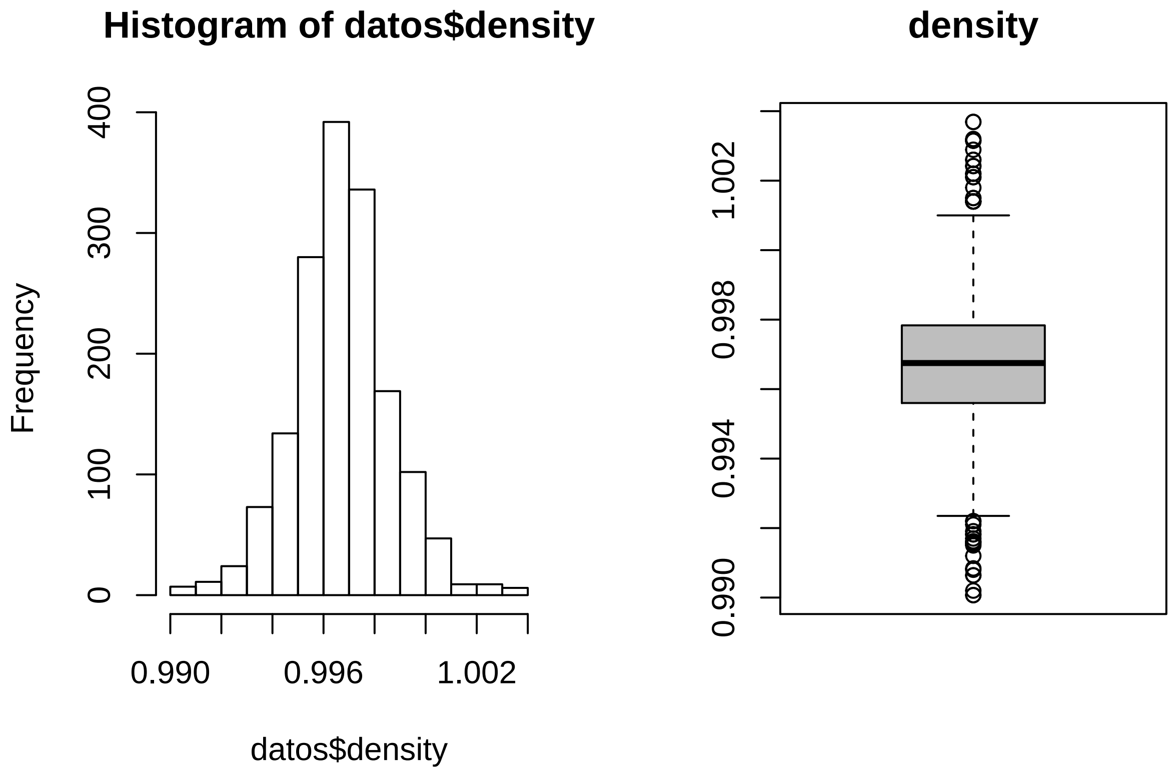
```
par(mfrow=c(1,2))
hist(datos$total.sulfur.dioxide)
boxplot(datos$total.sulfur.dioxide,main="total.sulfur.dioxide", col="gray")
```



```
boxplot.stats(datos$total.sulfur.dioxide)$out
```

```
##  [1] 145 148 136 125 140 136 133 153 134 141 129 128 129 128 143 144 127 126 145
## [20] 144 135 165 124 124 134 124 129 151 133 142 149 147 145 148 155 151 152 125
## [39] 127 139 143 144 130 278 289 135 160 141 141 133 147 147 131 131 131
```

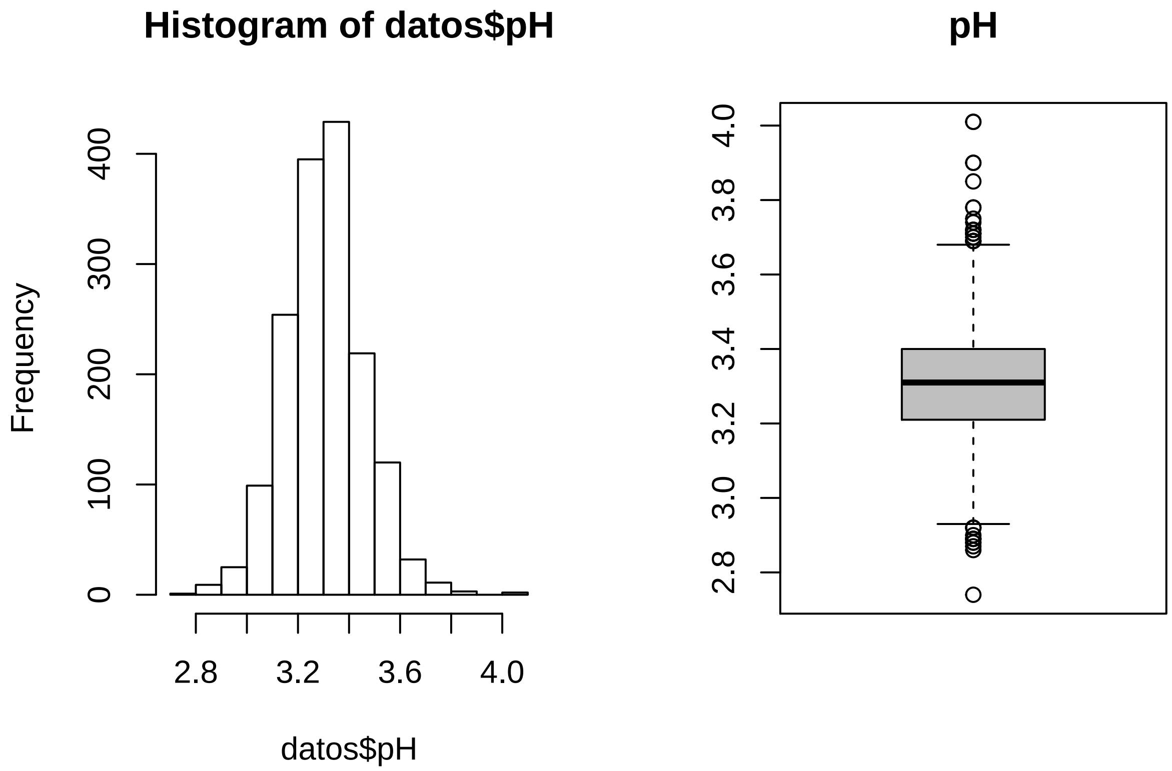
```
par(mfrow=c(1,2))
hist(datos$density)
boxplot(datos$density,main="density", col="gray")
```



```r
boxplot.stats(datos$density)$out
```

```
##  [1] 0.99160 0.99160 1.00140 1.00150 1.00150 1.00180 0.99120 1.00220 1.00220
## [10] 1.00140 1.00140 1.00140 1.00140 1.00320 1.00260 1.00140 1.00315 1.00315
## [19] 1.00315 1.00210 1.00210 0.99170 0.99220 1.00260 0.99210 0.99154 0.99064
## [28] 0.99064 1.00289 0.99162 0.99007 0.99007 0.99020 0.99220 0.99150 0.99157
## [37] 0.99080 0.99084 0.99191 1.00369 1.00369 1.00242 0.99182 1.00242 0.99182
```

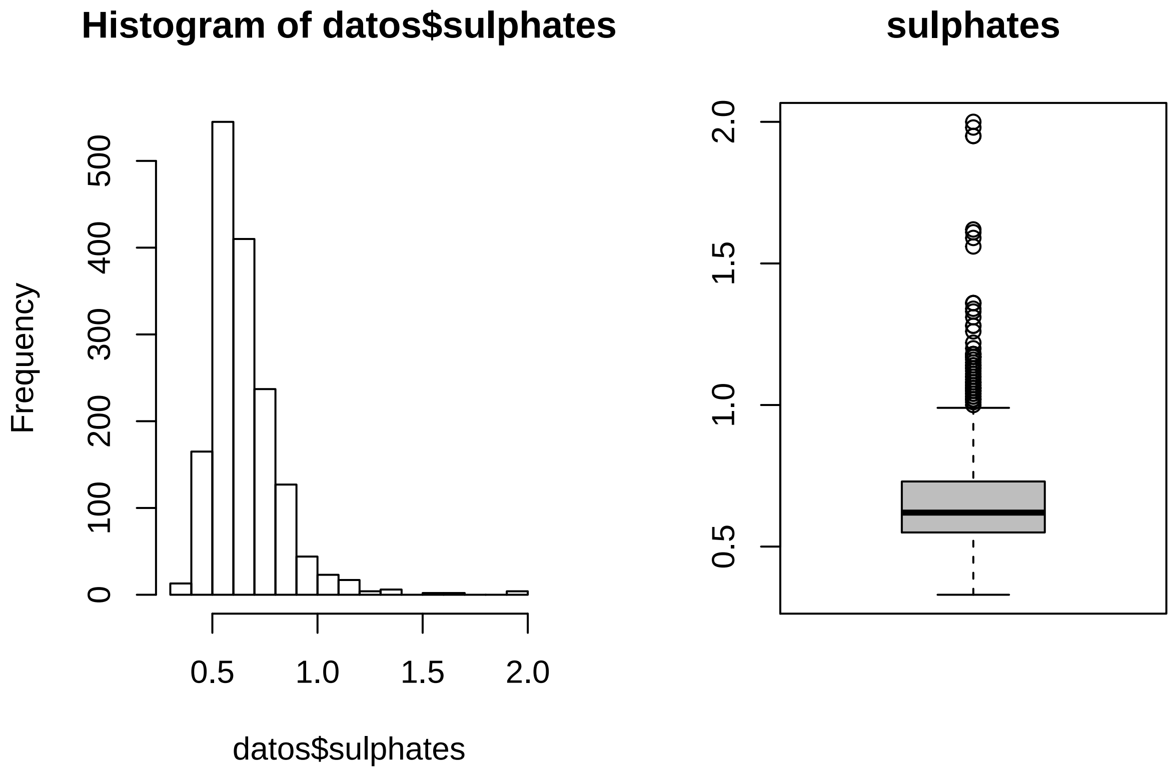
```r
par(mfrow=c(1,2))
hist(datos$pH)
boxplot(datos$pH,main="pH", col="gray")
```



```
boxplot.stats(datos$pH)$out
```

```
##  [1] 3.90 3.75 3.85 2.74 3.69 3.69 2.88 2.86 3.74 2.92 2.92 2.92 3.72 2.87 2.89
## [16] 2.89 2.92 3.90 3.71 3.69 3.69 3.71 3.71 2.89 2.89 3.78 3.70 3.78 4.01 2.90
## [31] 4.01 3.71 2.88 3.72 3.72
```

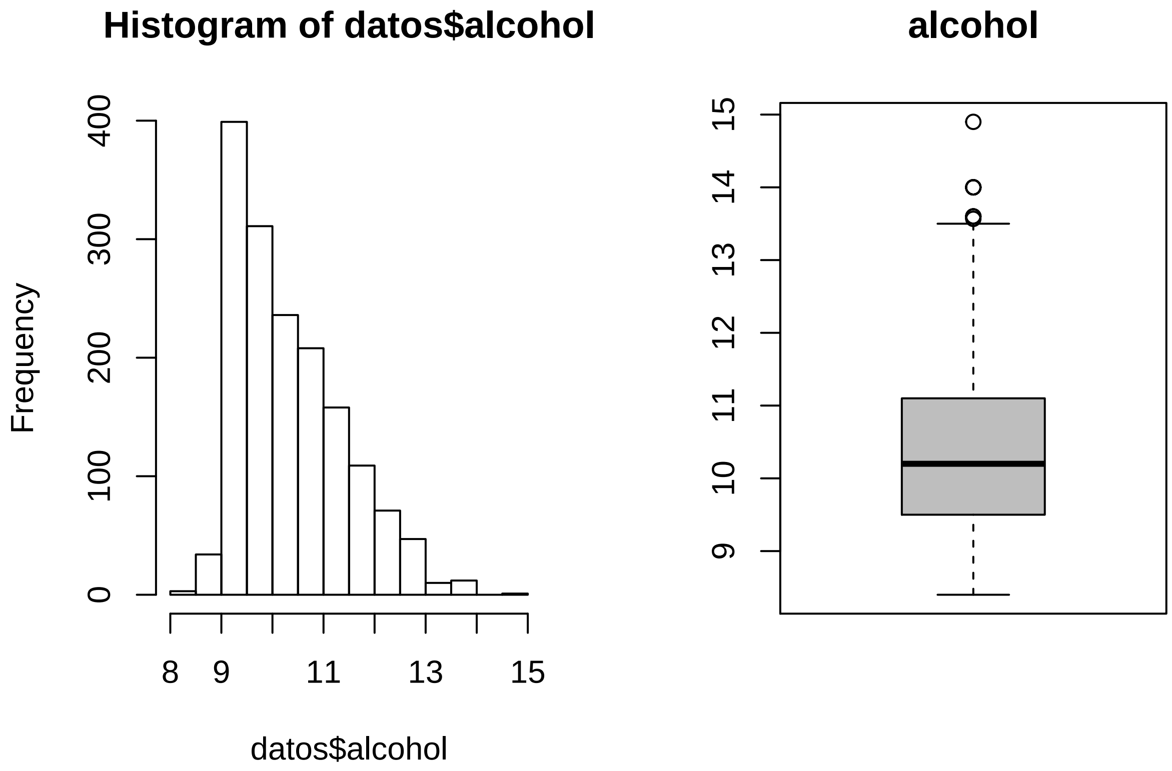
```
par(mfrow=c(1,2))
hist(datos$sulphates)
boxplot(datos$sulphates,main="sulphates", col="gray")
```



```
boxplot.stats(datos$sulphates)$out
```

```
##  [1] 1.56 1.28 1.08 1.20 1.12 1.28 1.14 1.95 1.22 1.95 1.98 1.31 2.00 1.08 1.59
## [16] 1.02 1.03 1.61 1.09 1.26 1.08 1.00 1.36 1.18 1.13 1.04 1.11 1.13 1.07 1.06
## [31] 1.06 1.05 1.06 1.04 1.05 1.02 1.14 1.02 1.36 1.36 1.05 1.17 1.62 1.06 1.18
## [46] 1.07 1.34 1.16 1.10 1.15 1.17 1.17 1.33 1.18 1.17 1.03 1.17 1.10 1.01
```

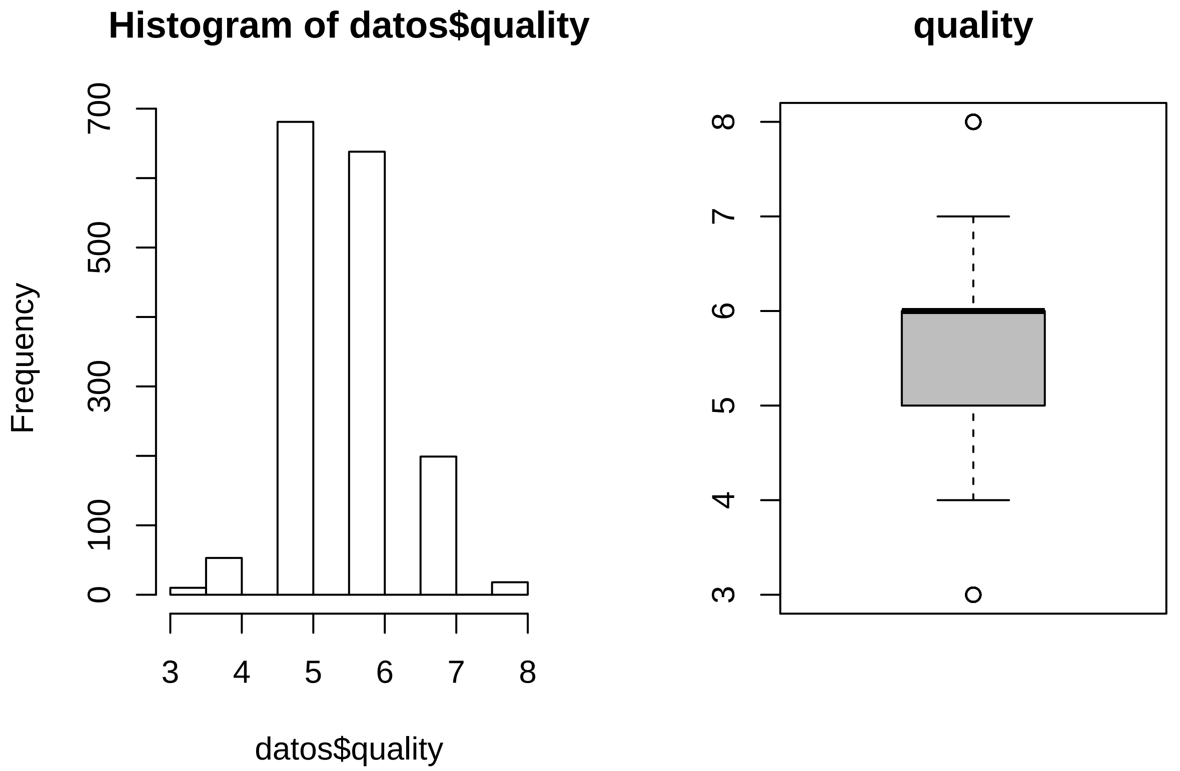
```
par(mfrow=c(1,2))
hist(datos$alcohol)
boxplot(datos$alcohol,main="alcohol", col="gray")
```



```
boxplot.stats(datos$alcohol)$out
```

```
##  [1] 14.00000 14.00000 14.00000 14.00000 14.90000 14.00000 13.60000 13.60000
##  [9] 13.60000 14.00000 14.00000 13.56667 13.60000
```

```
par(mfrow=c(1,2))
hist(datos$quality)
boxplot(datos$quality,main="quality", col="gray")
```



```
boxplot.stats(datos$quality)$out
```

```
##  [1] 8 8 8 8 8 3 8 8 8 3 8 3 8 3 3 8 8 8 8 8 3 3 8 8 3 3 3 8
```

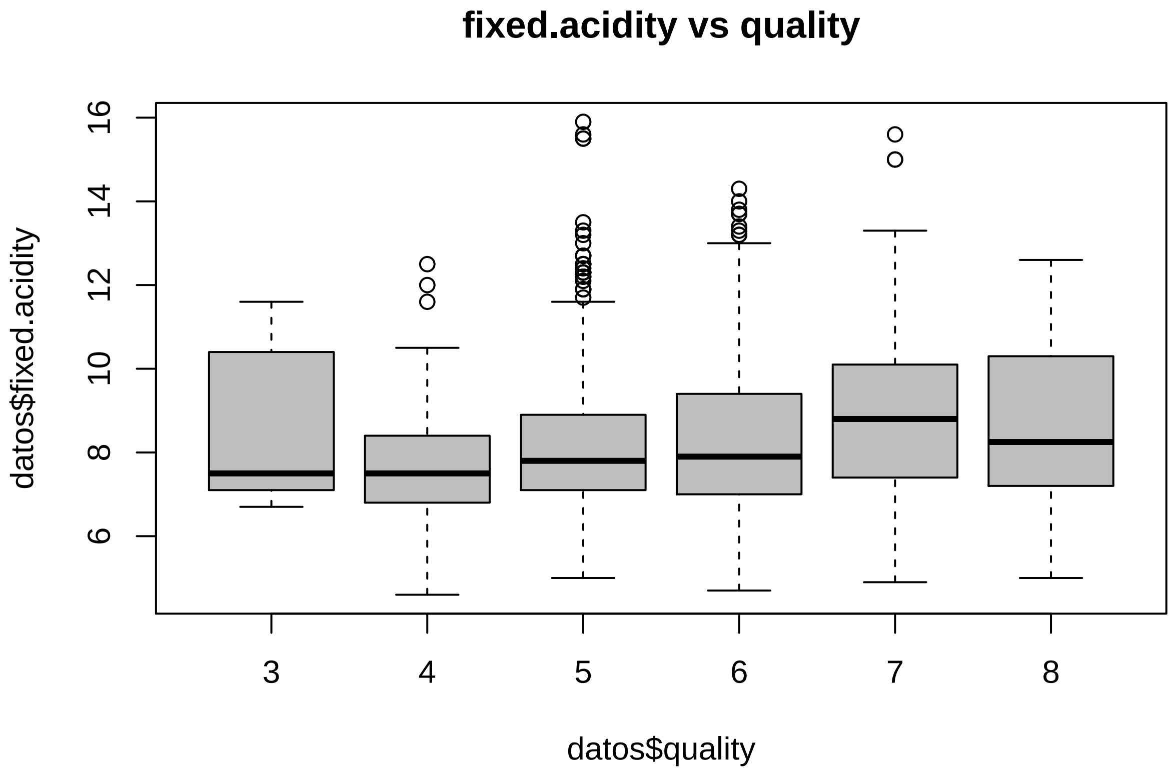
Analisis

Antes de comenzar con el analisis guardaremos una copia de los datos despues del proceso de limpieza

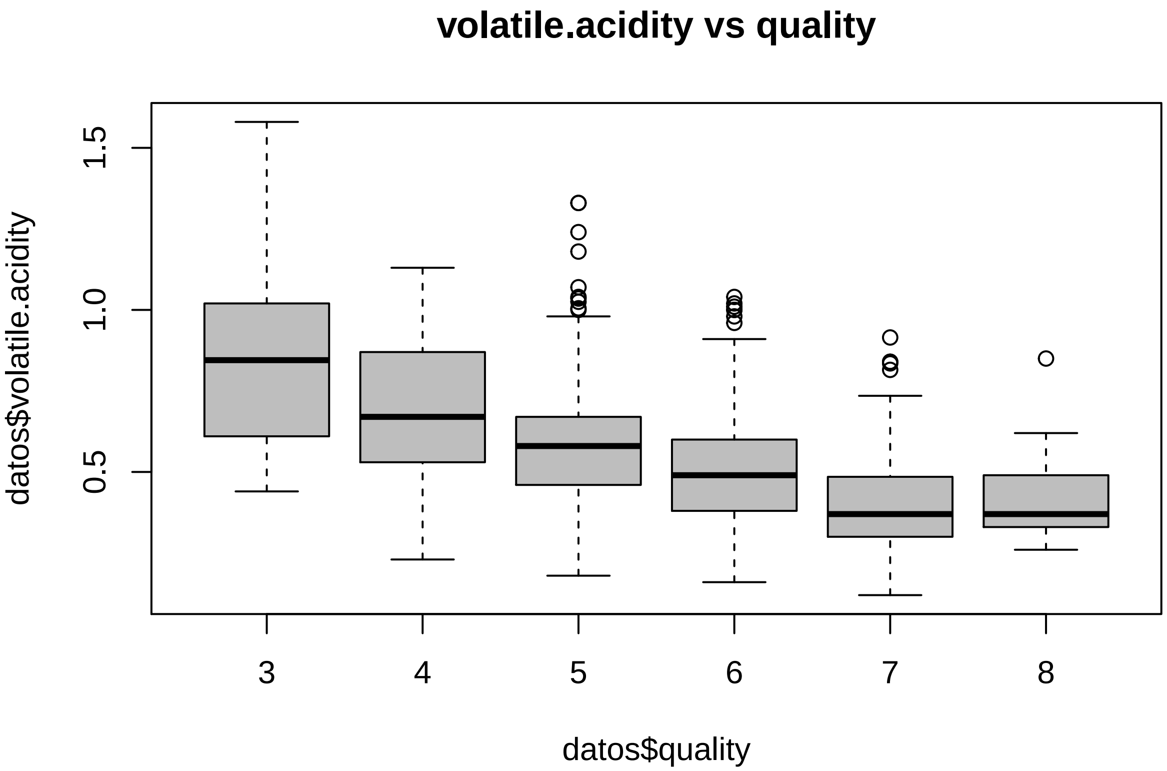
```
# Exportación de los datos limpios en .csv
write.csv(datos, "datos_data_clean.csv")
```

Analizaremos las variables frente a la calidad para decidir cuaes utilizar en el resto del analisis

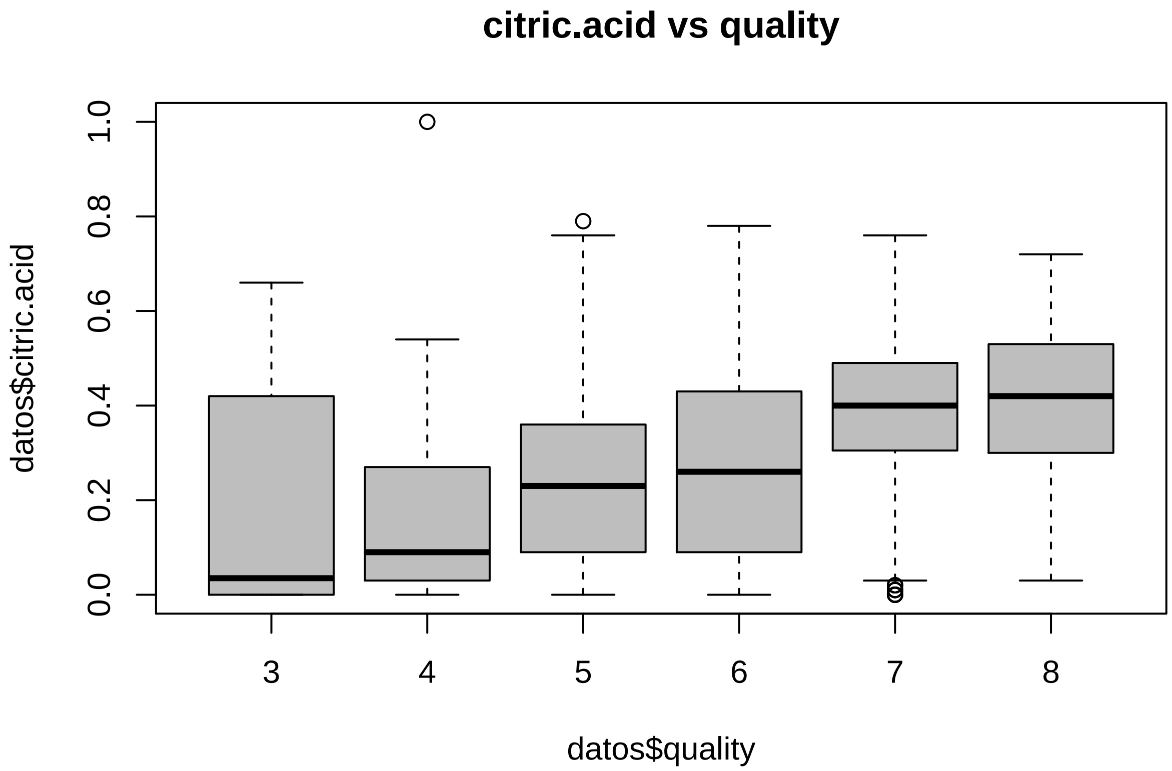
```
#boxplot(datos$pH,main="quality", col="gray")
boxplot(formula = datos$fixed.acidity ~ datos$quality, main="fixed.acidity vs quality", col="gray")
```



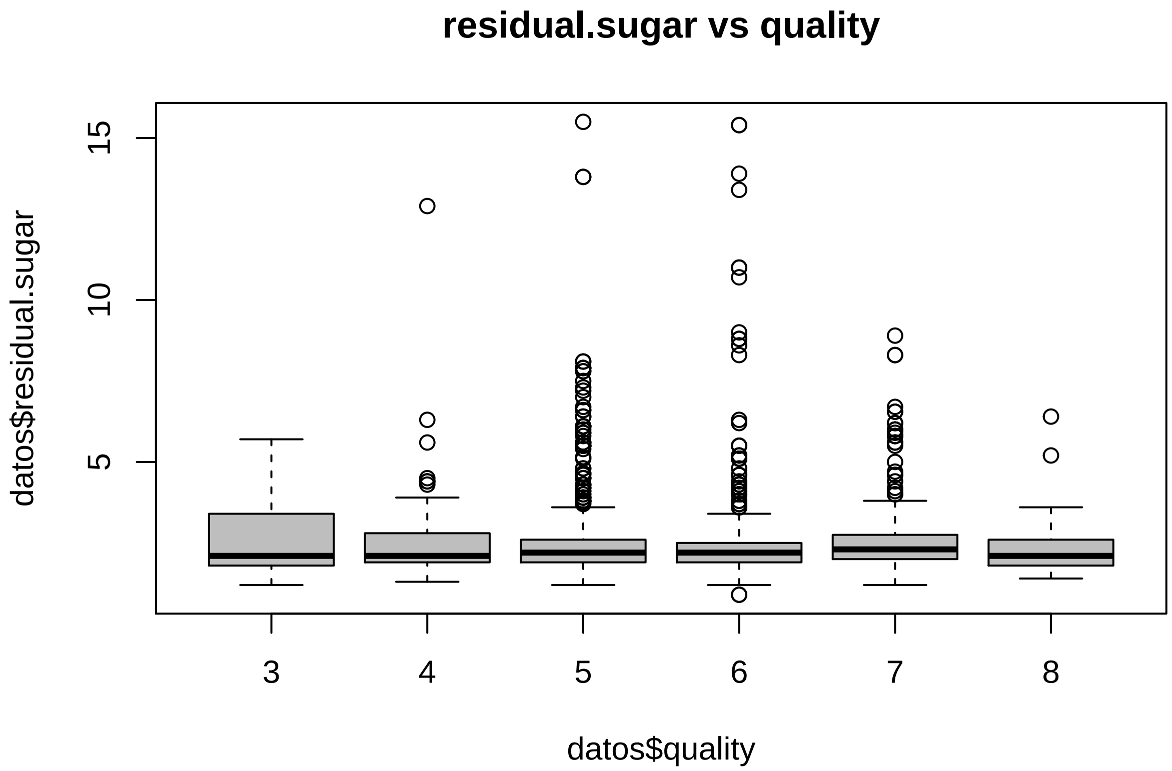
```
boxplot(formula = datos$volatile.acidity ~ datos$quality, main="volatile.acidity vs quality", col="gray
```



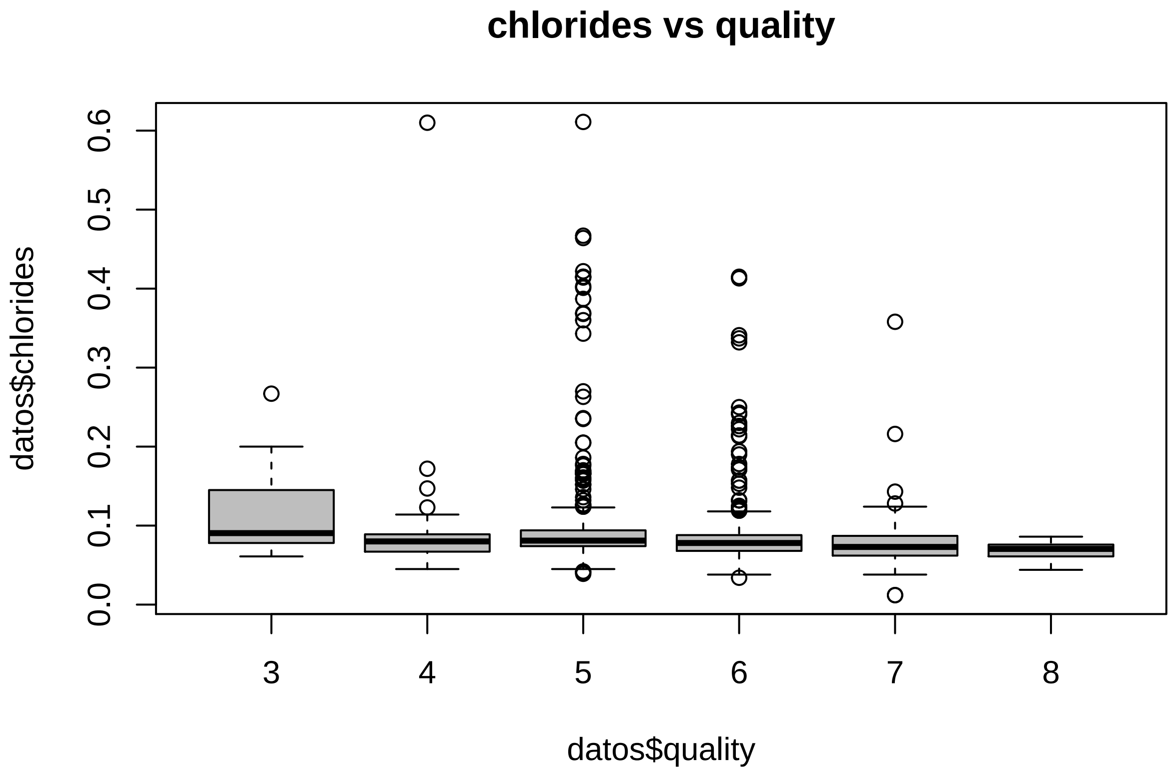
```r
boxplot(formula = datos$citric.acid ~ datos$quality, main="citric.acid vs quality", col="gray")
```



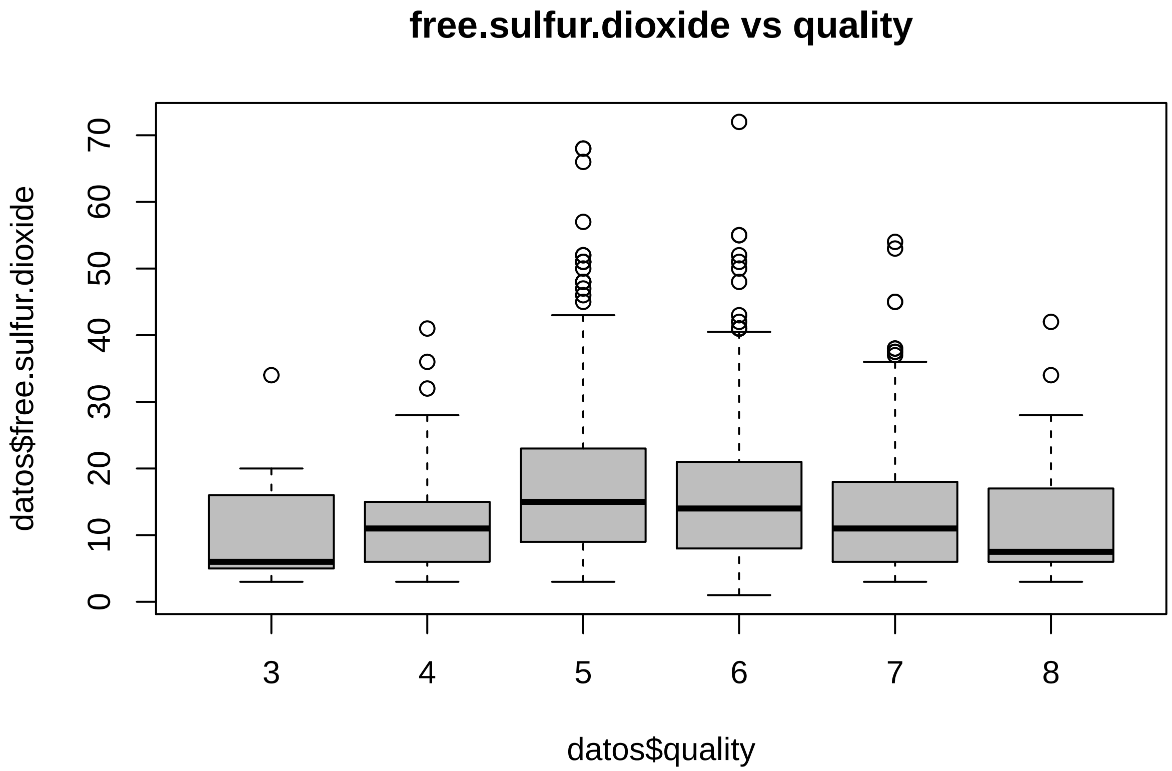
```
boxplot(formula = datos$residual.sugar ~ datos$quality, main="residual.sugar vs quality", col="gray")
```



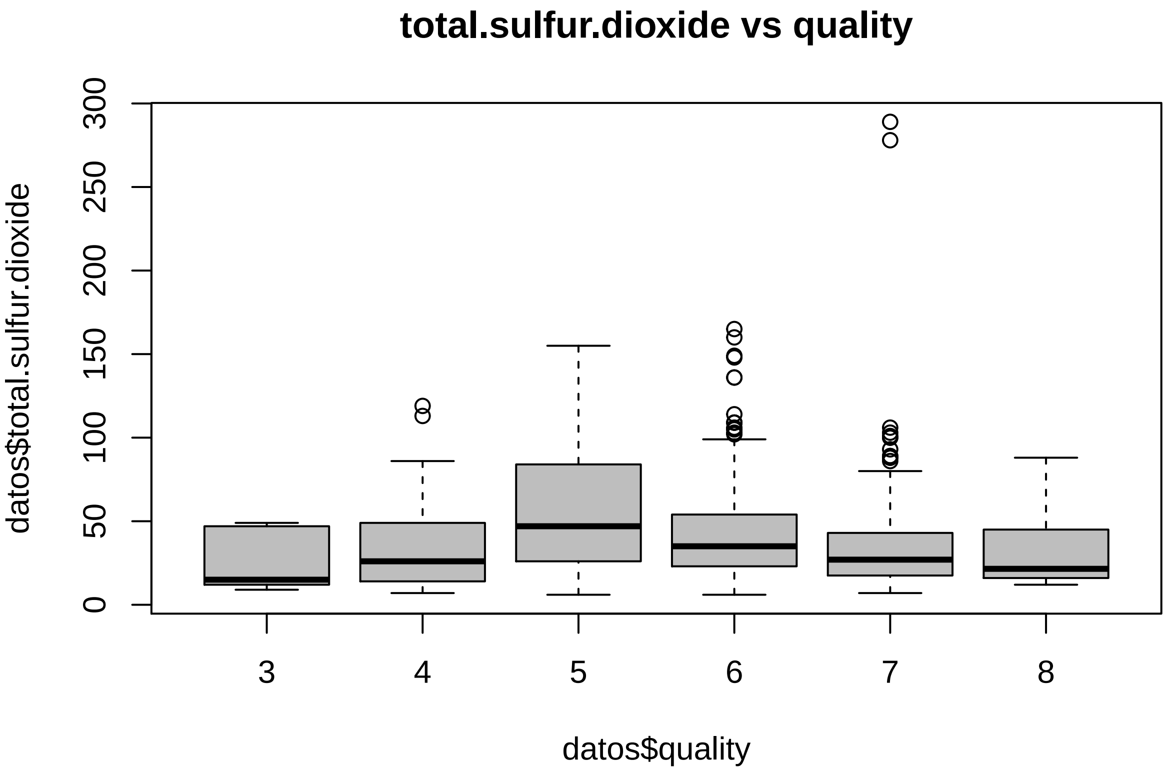
```
boxplot(formula = datos$chlorides ~ datos$quality, main="chlorides vs quality", col="gray")
```



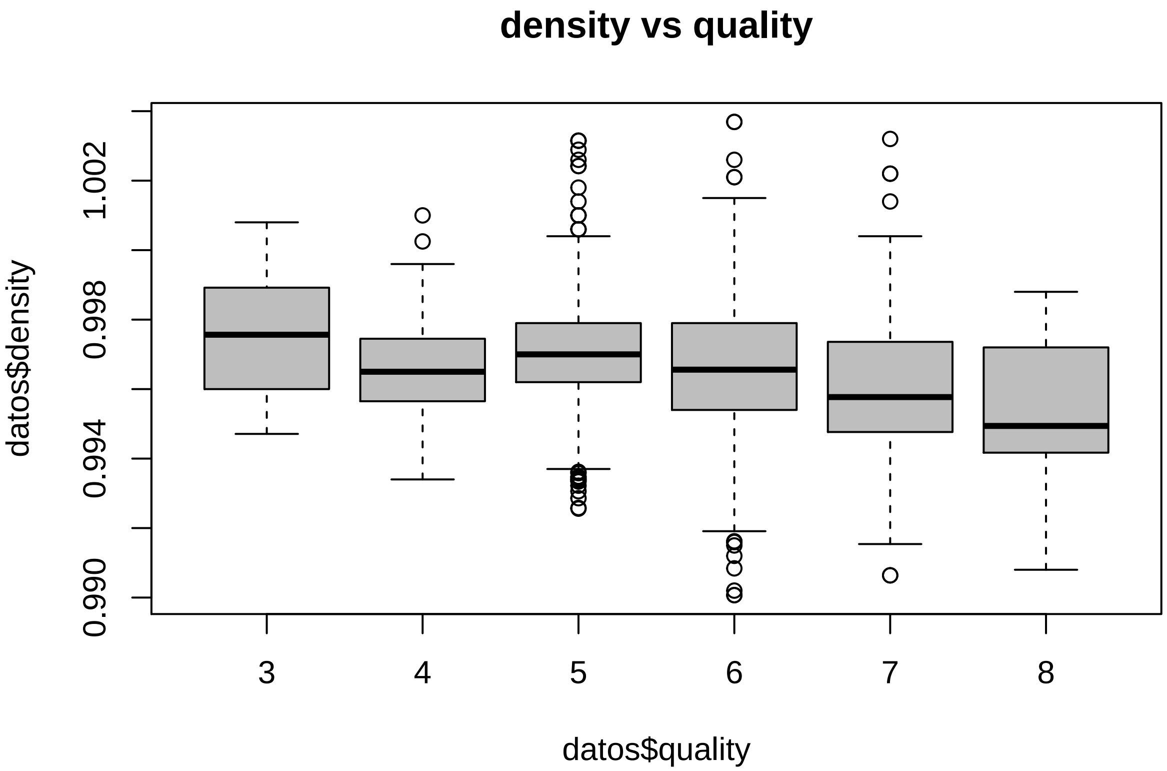
```
boxplot(formula = datos$free.sulfur.dioxide ~ datos$quality, main="free.sulfur.dioxide vs quality", col
```



```
boxplot(formula = datos$total.sulfur.dioxide ~ datos$quality, main="total.sulfur.dioxide vs quality", co
```

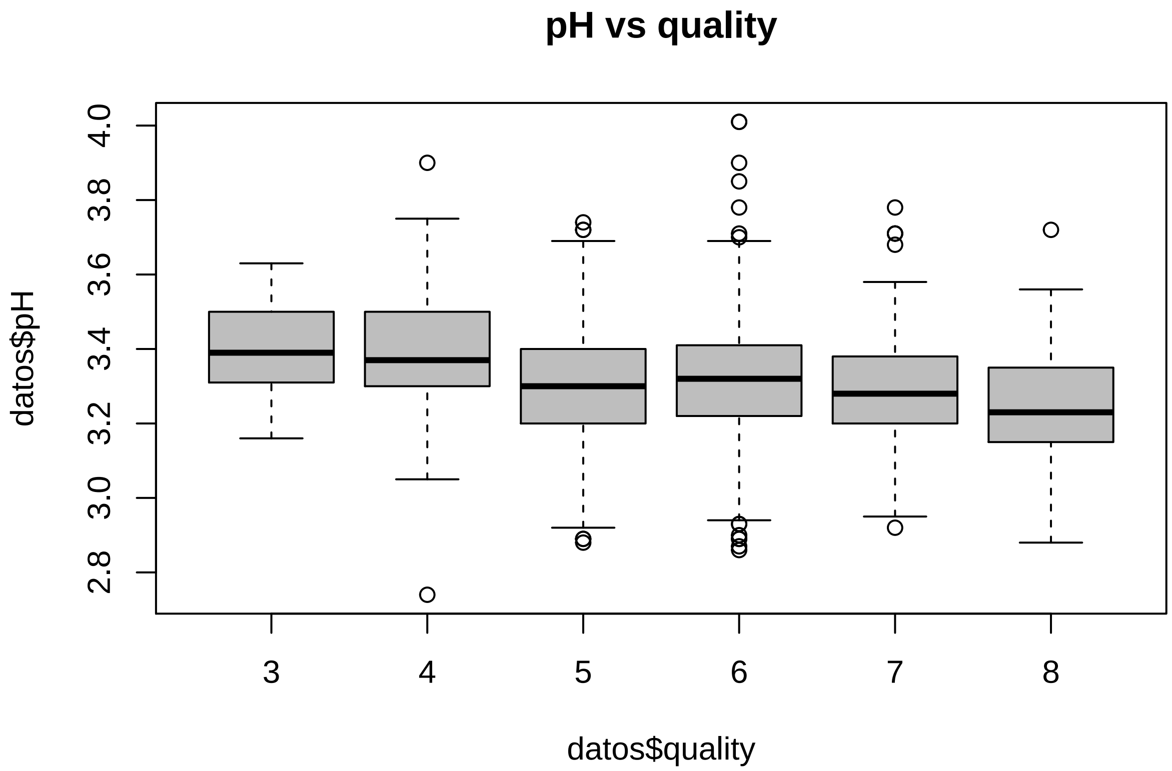


```
boxplot(formula = datos$density ~ datos$quality, main="density vs quality", col="gray")
```

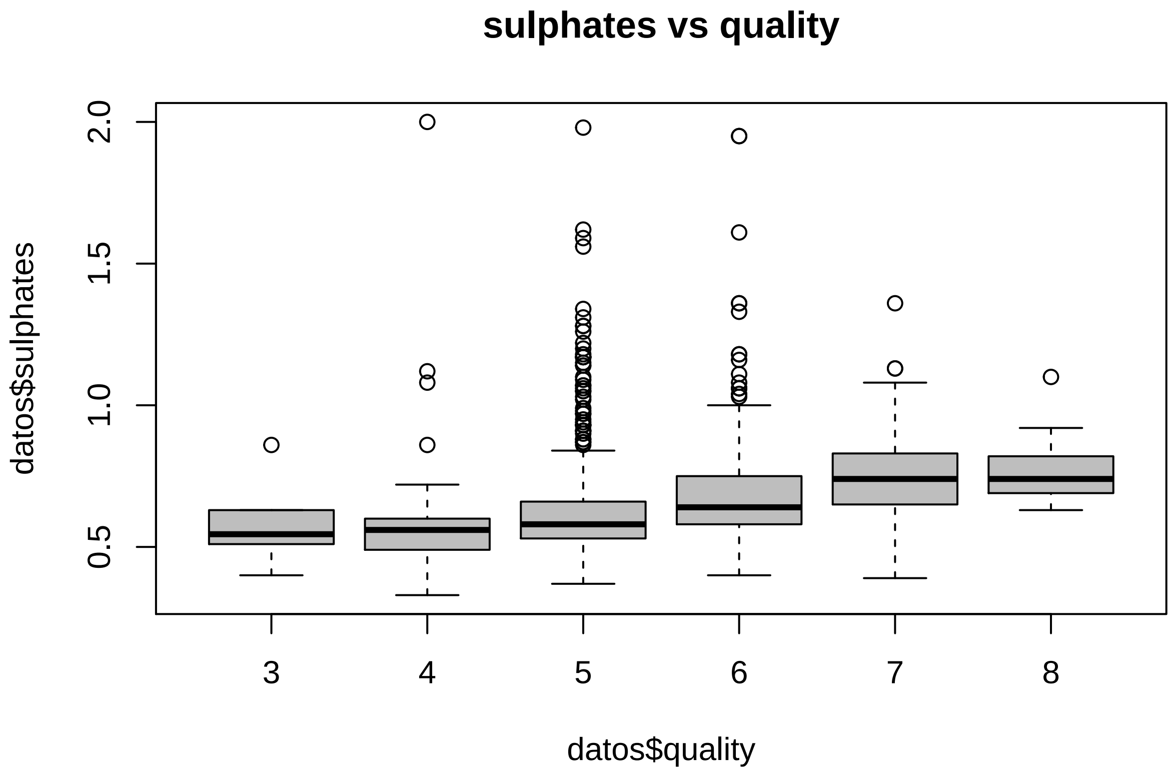


density vs quality

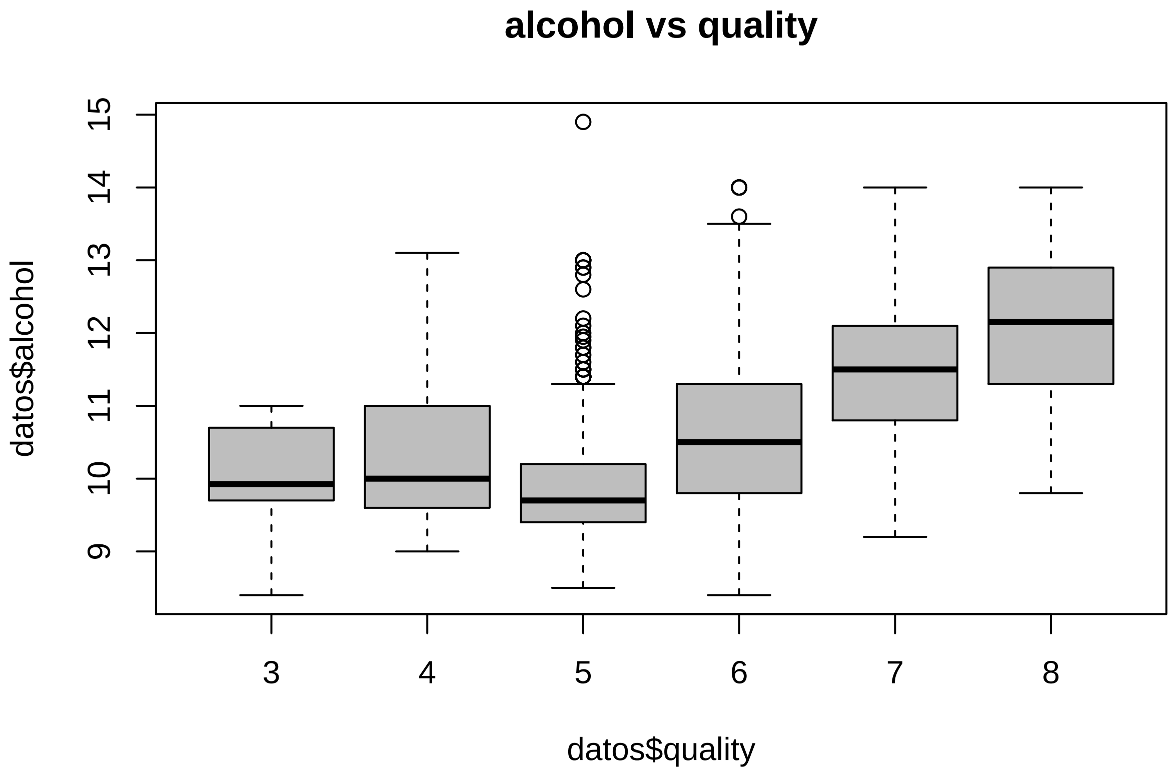
```r
boxplot(formula = datos$pH ~ datos$quality, main="pH vs quality", col="gray")
```



```
boxplot(formula = datos$sulphates ~ datos$quality, main="sulphates vs quality", col="gray")
```



```
boxplot(formula = datos$alcohol ~ datos$quality, main="alcohol vs quality", col="gray")
```



Seleccion grupo de datos

Comprobación de la normalidad y homogeneidad de la varianza.

```
shapiro.test(datos$fixed.acidity)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datos$fixed.acidity
## W = 0.94203, p-value < 2.2e-16
```

El test nos indica que ninguna variable esta normalizada, ya que el p-valor es inferior al coeficiente 0.05, por lo que se puede rechazar la hipotesis nula y entender que no es normal.

Que no sea normal no quiere decir que no pueda ser normalizable, ya que segun el teorema del limite central al tener mas de 30 elementos en las observaciones podemos aproximarla como una distribución normal de media 0 y desviación estandard 1.

Aplicación de pruebas estadísticas para comparar los grupos de datos

Representación de los resultados

Resolución del problema