

# PRAC2-WINE

*Ramón Serrano Valero*

*30/5/2020*

## Sección 1

### 1. Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?

El dataset elegido es “Wine Quality”, que podemos encontrar en el repositorio Kaggle, en la ruta: <https://www.kaggle.com/danielpanizzo/wine-quality?select=wineQualityWhites.csv>

Es un dataset que contiene propiedades fisicoquímicas de casi 5.000 de variantes de vino y su calidad. No se proporciona información sobre tipos de uva, marcas o precio de cada uno de los vinos. En este caso, existen dos datasets, uno para vino tinto y otro para vino blanco. Dado que el vino blanco tiene un mayor número de registros, será el que utilizaremos.

Pretende resolver la cuestión de predecir la calidad de un vino, en base a sus propiedades fisicoquímicas, una mejor composición química, hace que un vino pueda ser mejor que otro.

Exploración de los datos

Hay 4.898 registros, con 13 columnas, de las cuales, 12 de ellas son propiedades fisicoquímicas de entrada numéricas, mientras que la treceava es la variable de salida que se corresponde con la puntuación de calidad obtenida para cada vino en base a sus propiedades.

Entrada:

- X: Número de fila, aunque esta columna la eliminaremos, dado que no aporta.
- fixed acidity: La mayoría de los ácidos del vino, fijos o sin temprana evaporación. (g/dm3)
- volatile acidity: Cantidad de ácido acético, niveles altos pueden causar sabor a vinagrado. (g/dm3)
- citric acid: Pequeñas cantidades de ácido cítrico, causa frescura y sabor al vino. (g/dm3)
- residual sugar: Cantidad de azúcar sobrante tras la fermentación. (g/dm3)
- chlorides: Cantidad de sal en el vino. (g/dm3)
- free sulfur dioxide: Dióxido de azufre libre, previene la formación microbiana. (mg/dm3)
- total sulfur dioxide: Cantidad liberada de dióxido de azufre, detectable por nariz (mg/dm3)
- density: Densidad del agua, en base a cantidad de alcohol y azúcar. (g/cm3)
- pH: PH del ácido de 0 (muy ácido) a 14 (poco ácido)
- sulphates: Sulfato, aditivo que puede contribuir a niveles de dióxido de sulfato (g/dm3)
- alcohol: Porcentaje de alcohol del vino.

Salida:

- quality: Calidad del vino del 0 al 10.

Leemos los datos del csv, del dataset de Kaggle: <https://www.kaggle.com/danielpanizzo/wine-quality>

Donde podemos observar los valores que adquieren cada una de las columnas. Son de tipo numérico.

```
datos<-read.csv('datasets_3631_5794_wineQualityWhites.csv', header = TRUE, sep = ",", fill = TRUE)
summary(datos)
```

```

##      X      fixed.acidity volatile.acidity citric.acid
## Min. : 1      Min. : 3.800    Min. :0.0800  Min. :0.0000
## 1st Qu.:1225  1st Qu.: 6.300    1st Qu.:0.2100  1st Qu.:0.2700
## Median :2450  Median : 6.800    Median :0.2600  Median :0.3200
## Mean   :2450  Mean   : 6.855    Mean   :0.2782  Mean   :0.3342
## 3rd Qu.:3674  3rd Qu.: 7.300    3rd Qu.:0.3200  3rd Qu.:0.3900
## Max.  :4898   Max.  :14.200    Max.  :1.1000  Max.  :1.6600
## residual.sugar chlorides free.sulfur.dioxide
## Min. : 0.600  Min. :0.00900  Min. : 2.00
## 1st Qu.: 1.700 1st Qu.:0.03600 1st Qu.: 23.00
## Median : 5.200  Median :0.04300  Median : 34.00
## Mean   : 6.391  Mean   :0.04577  Mean   : 35.31
## 3rd Qu.: 9.900 3rd Qu.:0.05000 3rd Qu.: 46.00
## Max.  :65.800  Max.  :0.34600  Max.  :289.00
## total.sulfur.dioxide density pH sulphates
## Min. : 9.0      Min. :0.9871  Min. :2.720  Min. :0.2200
## 1st Qu.:108.0    1st Qu.:0.9917  1st Qu.:3.090  1st Qu.:0.4100
## Median :134.0    Median :0.9937  Median :3.180  Median :0.4700
## Mean   :138.4    Mean   :0.9940  Mean   :3.188  Mean   :0.4898
## 3rd Qu.:167.0    3rd Qu.:0.9961  3rd Qu.:3.280  3rd Qu.:0.5500
## Max.  :440.0     Max.  :1.0390  Max.  :3.820  Max.  :1.0800
## alcohol quality
## Min. : 8.00    Min. :3.000
## 1st Qu.: 9.50    1st Qu.:5.000
## Median :10.40    Median :6.000
## Mean   :10.51    Mean   :5.878
## 3rd Qu.:11.40    3rd Qu.:6.000
## Max.  :14.20    Max.  :9.000

```

## Sección 2

### 2. Integración y selección de los datos de interés a analizar

Disponíamos de dos datasets, no obstante nos quedamos con el dataset que contenía un mayor número de filas, el de vino blanco. Dado que la mayoría de columnas se corresponden con propiedades del vino, en forma de registros, nos deberemos quedar con ellas, para utilizarlos en la fase de análisis.

Por tanto disponemos de 13 columnas, aunque nos quedaremos con 12 de estas que son las propiedades de los vinos y su calidad, quitando la que no nos interesa, es decir, el número de fila:

```

white_wine_data<-datos[,2:13]
head(white_wine_data)

## fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1          7.0          0.27        0.36       20.7      0.045
## 2          6.3          0.30        0.34        1.6      0.049
## 3          8.1          0.28        0.40        6.9      0.050
## 4          7.2          0.23        0.32        8.5      0.058
## 5          7.2          0.23        0.32        8.5      0.058
## 6          8.1          0.28        0.40        6.9      0.050
## free.sulfur.dioxide total.sulfur.dioxide density pH sulphates alcohol
## 1             45           170  1.0010 3.00      0.45      8.8
## 2            14            132  0.9940 3.30      0.49      9.5

```

```

## 3          30          97  0.9951 3.26      0.44   10.1
## 4          47          186  0.9956 3.19      0.40    9.9
## 5          47          186  0.9956 3.19      0.40    9.9
## 6          30          97  0.9951 3.26      0.44   10.1
##   quality
## 1          6
## 2          6
## 3          6
## 4          6
## 5          6
## 6          6

```

Otra modificación que realizaremos es introducir una columna categórica que aglutine a los vinos como Buenos, Normales y Malos. Esta nueva columna será category.

Categorizamos los vinos en base a su puntuación, de tal forma que:

- Vino Malo: Puntuación Inferior a 5. [ $<5$ ]
- Vino Normal: Puntuación Igual o superior a 5 e inferior o igual a 7. [5,7]
- Vino Bueno: Puntuación Superior a 7. [ $>7$ ]

```

#Creamos la nueva columna
white_wine_data$category<-NA

#Categorizamos que un vino es malo si su puntuación es inferior a 5
white_wine_data[white_wine_data$quality<5,]$category='Malo'
#Categorizamos que un vino es normal si su puntuación se encuentra entre 5 y 7
white_wine_data[white_wine_data$quality>=5 & white_wine_data$quality<=7,]$category='Normal'
#Categorizamos que un vino es bueno si su puntuación es superior a 7
white_wine_data[white_wine_data$quality>7,]$category='Bueno'

head(white_wine_data)

##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1          7.0          0.27        0.36        20.7      0.045
## 2          6.3          0.30        0.34        1.6       0.049
## 3          8.1          0.28        0.40        6.9       0.050
## 4          7.2          0.23        0.32        8.5       0.058
## 5          7.2          0.23        0.32        8.5       0.058
## 6          8.1          0.28        0.40        6.9       0.050
##   free.sulfur.dioxide total.sulfur.dioxide density      pH sulphates alcohol
## 1                 45           170  1.0010 3.00      0.45     8.8
## 2                 14           132  0.9940 3.30      0.49     9.5
## 3                 30            97  0.9951 3.26      0.44   10.1
## 4                 47           186  0.9956 3.19      0.40    9.9
## 5                 47           186  0.9956 3.19      0.40    9.9
## 6                 30            97  0.9951 3.26      0.44   10.1
##   quality category
## 1          6   Normal
## 2          6   Normal
## 3          6   Normal
## 4          6   Normal
## 5          6   Normal
## 6          6   Normal

```

Finalmente transformamos la columna en categórica.

```

white_wine_data$category<-as.factor(white_wine_data$category)
str(white_wine_data)

## 'data.frame': 4898 obs. of 13 variables:
## $ fixed.acidity      : num 7 6.3 8.1 7.2 7.2 8.1 6.2 7 6.3 8.1 ...
## $ volatile.acidity   : num 0.27 0.3 0.28 0.23 0.23 0.28 0.32 0.27 0.3 0.22 ...
## $ citric.acid        : num 0.36 0.34 0.4 0.32 0.32 0.4 0.16 0.36 0.34 0.43 ...
## $ residual.sugar     : num 20.7 1.6 6.9 8.5 8.5 6.9 7 20.7 1.6 1.5 ...
## $ chlorides           : num 0.045 0.049 0.05 0.058 0.058 0.05 0.045 0.045 0.049 0.044 ...
## $ free.sulfur.dioxide: num 45 14 30 47 47 30 30 45 14 28 ...
## $ total.sulfur.dioxide: num 170 132 97 186 186 97 136 170 132 129 ...
## $ density              : num 1.001 0.994 0.995 0.996 0.996 ...
## $ pH                   : num 3 3.3 3.26 3.19 3.19 3.26 3.18 3 3.3 3.22 ...
## $ sulphates            : num 0.45 0.49 0.44 0.4 0.4 0.44 0.47 0.45 0.49 0.45 ...
## $ alcohol               : num 8.8 9.5 10.1 9.9 9.9 10.1 9.6 8.8 9.5 11 ...
## $ quality               : int 6 6 6 6 6 6 6 6 6 ...
## $ category              : Factor w/ 3 levels "Bueno","Malo",...: 3 3 3 3 3 3 3 3 3 3 ...

```

## Sección 3

### 3. Limpieza de los datos.

Tipo de dato asignado a cada campo, se corresponde con el dominio de estos.

```

# Tipo de dato asignado a cada campo
sapply(white_wine_data, 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"
##      category           "factor"

```

#### 3.1.a) ¿Los datos contienen ceros o elementos vacíos?

Realizar un análisis de los datos de calidad, debemos realizar la comprobación de valores perdidos y detectar los outliers en caso de que los haya. Para ello, primero buscaremos posibles valores perdidos y los outliers. Posteriormente realizaremos un análisis Multivariante, visualizando los datos de forma gráfica de relaciones entre variables.

Comprobamos los **nulos** de cada columna:

Número de nulos para fixed.acidity: 0

Número de nulos para volatile.acidity: 0

Número de nulos para citric.acid: 0

Número de nulos para residual.sugar: 0

Número de nulos para chlorides: 0

Número de nulos para free.sulfur.dioxide: 0

Número de nulos para total.sulfur.dioxide: 0

Número de nulos para density: 0

Número de nulos para pH: 0

Número de nulos para sulphates: 0

Número de nulos para quality: 0

Número de nulos para category: 0

Podemos inferir que no hay valores vacíos en los datos, en cuanto a nulos.

Procedemos a verificar si hay valores con valor 0, comprobamos los **ceros** de cada columna:

Número de ceros para fixed.acidity: 0

Número de ceros para volatile.acidity: 0

Número de ceros para citric.acid: 19

Número de ceros para residual.sugar: 0

Número de ceros para chlorides: 0

Número de ceros para free.sulfur.dioxide: 0

Número de ceros para total.sulfur.dioxide: 0

Número de ceros para density: 0

Número de ceros para pH: 0

Número de ceros para sulphates: 0

Número de ceros para quality: 0

Número de ceros para category: 0

Vemos que únicamente hay 19 valores 0 para la columna de cantidad de gramos de ácido cítrico, no obstante, este valor 0 forma parte del **dominio del atributo**. Por lo tanto, es posible que un vino no disponga de ácido cítrico, por tanto su valor numérico 0 tiene sentido. Siendo su valor comprendido entre [0,1.66]. No os valores muy altos, por lo que la cantidad de gramos de cítricos que posee un vino no suele ser superior a 1.66 y su valor medio suele ser 0.3341915

```
summary(white_wine_data)
```

```
##   fixed.acidity   volatile.acidity   citric.acid   residual.sugar
##   Min.    : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
##   1st Qu.: 6.300   1st Qu.:0.2100   1st Qu.:0.2700   1st Qu.: 1.700
##   Median  : 6.800   Median  :0.2600   Median  :0.3200   Median  : 5.200
##   Mean    : 6.855   Mean    :0.2782   Mean    :0.3342   Mean    : 6.391
##   3rd Qu.: 7.300   3rd Qu.:0.3200   3rd Qu.:0.3900   3rd Qu.: 9.900
##   Max.    :14.200   Max.    :1.1000   Max.    :1.6600   Max.    :65.800
##   chlorides      free.sulfur.dioxide total.sulfur.dioxide
##   Min.    :0.00900   Min.    : 2.00     Min.    : 9.0
##   1st Qu.: 0.03600   1st Qu.:23.00    1st Qu.:108.0
##   Median  :0.04300   Median  :34.00    Median  :134.0
##   Mean    :0.04577   Mean    :35.31    Mean    :138.4
##   3rd Qu.: 0.05000   3rd Qu.:46.00    3rd Qu.:167.0
##   Max.    :0.34600   Max.    :289.00   Max.    :440.0
##   density        pH           sulphates      alcohol
##   Min.    :0.9871   Min.    :2.720    Min.    :0.2200   Min.    : 8.00
##   1st Qu.: 0.9917   1st Qu.:3.090    1st Qu.:0.4100   1st Qu.: 9.50
```

```

## Median :0.9937   Median :3.180   Median :0.4700   Median :10.40
## Mean   :0.9940   Mean   :3.188   Mean   :0.4898   Mean   :10.51
## 3rd Qu.:0.9961   3rd Qu.:3.280   3rd Qu.:0.5500   3rd Qu.:11.40
## Max.   :1.0390   Max.   :3.820   Max.   :1.0800   Max.   :14.20
##      quality      category
## Min.   :3.000   Bueno : 180
## 1st Qu.:5.000   Malo  : 183
## Median :6.000   Normal:4535
## Mean   :5.878
## 3rd Qu.:6.000
## Max.   :9.000

```

### 3.1.b) ¿Cómo gestionarías cada uno de estos casos?

No disponíamos de ningún valor perdido, así que simulemos que teníamos varios valores perdidos en las 5 primeras filas de la columna citric.acid:

```

white_wine_data_lost<-white_wine_data
#Eliminamos los valores
white_wine_data_lost[0:5,]$citric.acid<-NA

```

Si hubieramos obtenido algún valor perdido, podríamos haber optado por varias opciones:

- Reemplazo manual: Si conocemos la información perdida y podemos corregirlo y supone una inversión de tiempo aceptable, podemos cambiar manualmente el valor.

```

white_wine_data_lost[1,]$citric.acid<-0.36
white_wine_data_lost[2,]$citric.acid<-0.34
white_wine_data_lost[3,]$citric.acid<-0.40
white_wine_data_lost[4,]$citric.acid<-0.32
white_wine_data_lost[5,]$citric.acid<-0.32

white_wine_data_lost[0:5,]

## fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1          7.0          0.27          0.36          20.7        0.045
## 2          6.3          0.30          0.34           1.6        0.049
## 3          8.1          0.28          0.40           6.9        0.050
## 4          7.2          0.23          0.32           8.5        0.058
## 5          7.2          0.23          0.32           8.5        0.058
## free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol
## 1            45           170  1.0010 3.00        0.45       8.8
## 2            14           132  0.9940 3.30        0.49       9.5
## 3            30            97  0.9951 3.26        0.44      10.1
## 4            47           186  0.9956 3.19        0.40       9.9
## 5            47           186  0.9956 3.19        0.40       9.9
##      quality      category
## 1      6  Normal
## 2      6  Normal
## 3      6  Normal
## 4      6  Normal
## 5      6  Normal

```

- Reemplazo por constante: Podemos reemplazar el valor perdido por una misma constante, para agrupar dichos valores en un mismo conjunto de valores desconocidos. Para el mismo ejemplo de valores perdidos de ácido cítrico anterior, reemplazamos por valor -1 todos los valores.

```

#Eliminamos los valores
white_wine_data_lost[0:5,]$citric.acid<-NA
#Asignamos mismo valor al valor perdido
white_wine_data_lost[is.na(white_wine_data_lost$citric.acid),]$citric.acid<--1
white_wine_data_lost[0:5,]

##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1      7.0          0.27       -1      20.7     0.045
## 2      6.3          0.30       -1       1.6     0.049
## 3      8.1          0.28       -1       6.9     0.050
## 4      7.2          0.23       -1       8.5     0.058
## 5      7.2          0.23       -1       8.5     0.058
##   free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol
## 1            45           170 1.0010 3.00     0.45     8.8
## 2            14           132 0.9940 3.30     0.49     9.5
## 3            30            97 0.9951 3.26     0.44    10.1
## 4            47           186 0.9956 3.19     0.40     9.9
## 5            47           186 0.9956 3.19     0.40     9.9
##   quality category
## 1      6   Normal
## 2      6   Normal
## 3      6   Normal
## 4      6   Normal
## 5      6   Normal

```

- Tendencia Central: Si se trata de un valor perdido categórico, podemos usar la moda, es decir el valor más repetido para dicho atributo. En caso de ser valor perdido numérico, empleamos la media o mediana para dicho atributo.

```

#Eliminamos los valores
white_wine_data_lost[0:5,]$citric.acid<-NA

valor_medio_acido_citrico<-mean(white_wine_data_lost[!is.na(white_wine_data_lost$citric.acid)],$citric.acid)

#Asignamos valor medio al valor perdido
white_wine_data_lost[is.na(white_wine_data_lost$citric.acid),]$citric.acid<-valor_medio_acido_citrico
white_wine_data_lost[0:5,]

##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1      7.0          0.27  0.3341774      20.7     0.045
## 2      6.3          0.30  0.3341774       1.6     0.049
## 3      8.1          0.28  0.3341774       6.9     0.050
## 4      7.2          0.23  0.3341774       8.5     0.058
## 5      7.2          0.23  0.3341774       8.5     0.058
##   free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol
## 1            45           170 1.0010 3.00     0.45     8.8
## 2            14           132 0.9940 3.30     0.49     9.5
## 3            30            97 0.9951 3.26     0.44    10.1
## 4            47           186 0.9956 3.19     0.40     9.9
## 5            47           186 0.9956 3.19     0.40     9.9
##   quality category
## 1      6   Normal
## 2      6   Normal
## 3      6   Normal
## 4      6   Normal

```

```
## 5      6  Normal
```

- KNN: K-Nearest Neighbours, que permite predecir valores en conjuntos de datos de valores discretos, continuos, ordinales y/o nominales. No obstante encontrar el valor K adecuado es el reto para la precisión. Utilizamos el paquete VIM de R, aplicamos kNN para obtener los valores perdidos, en base a una K=3 vecinos cercanos, en la columna citric.acid Donde vemos que resuelve los valores perdidos, obteniendo el mismo resultado que en otros ejemplos de este ejercicio.

```
#Eliminamos los valores
```

```
white_wine_data_lost[0:5,]$citric.acid<-NA  
#Cargamos libreria VIM  
library(VIM)
```

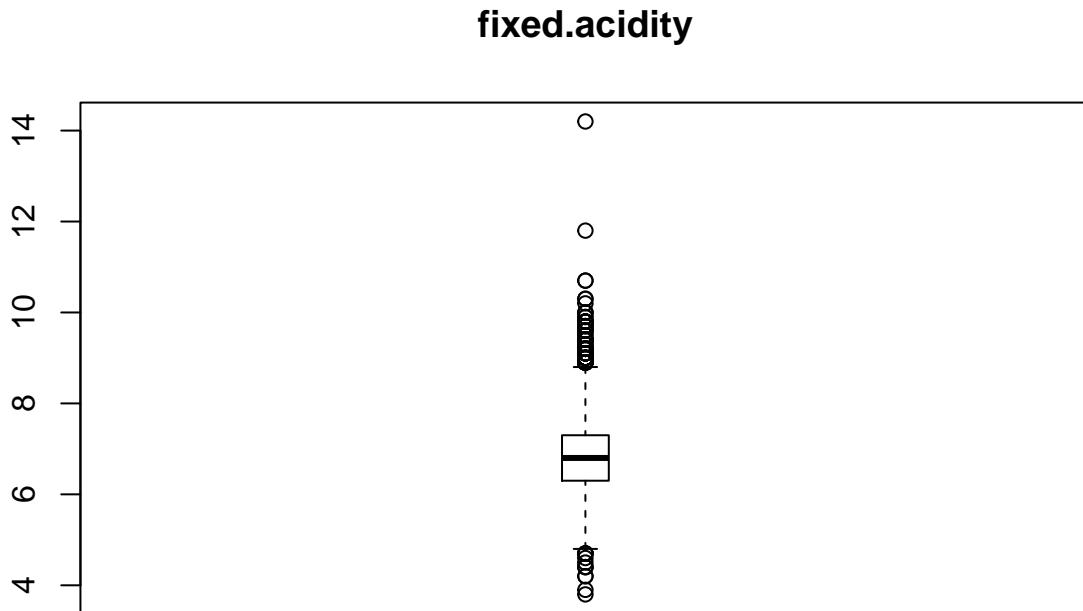
```
## Warning: package 'VIM' was built under R version 3.5.2  
## Loading required package: colorspace  
## Loading required package: grid  
## Loading required package: data.table  
## VIM is ready to use.  
## Since version 4.0.0 the GUI is in its own package VIMGUI.  
##  
##           Please use the package to use the new (and old) GUI.  
## Suggestions and bug-reports can be submitted at: https://github.com/alexkowa/VIM/issues  
##  
## Attaching package: 'VIM'  
## The following object is masked from 'package:datasets':  
##  
##     sleep  
#Aplicamos algoritmo KNN de vecinos cercanos con k=3  
white_wine_data_lost<-kNN(white_wine_data_lost, variable=c("citric.acid"), k=3)  
  
white_wine_data_lost[0:5,]  
  
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides  
## 1          7.0          0.27       0.41        20.7      0.045  
## 2          6.3          0.30       0.30        1.6      0.049  
## 3          8.1          0.28       0.40        6.9      0.050  
## 4          7.2          0.23       0.31        8.5      0.058  
## 5          7.2          0.23       0.31        8.5      0.058  
##   free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol  
## 1             45            170  1.0010 3.00      0.45      8.8  
## 2             14            132  0.9940 3.30      0.49      9.5  
## 3             30            97  0.9951 3.26      0.44     10.1  
## 4             47            186  0.9956 3.19      0.40      9.9  
## 5             47            186  0.9956 3.19      0.40      9.9  
##   quality category citric.acid_imp  
## 1      6  Normal        TRUE  
## 2      6  Normal        TRUE  
## 3      6  Normal        TRUE  
## 4      6  Normal        TRUE  
## 5      6  Normal        TRUE
```

### 3.2. Identificación y tratamiento de valores extremos.

Comprobemos si existen valores extremos en cada una de las variables:

Empezamos por la acidez **fixed.acidity**, vemos que tenemos dos valores que se escapan del conjunto de datos.

```
boxplot(white_wine_data$fixed.acidity, main="fixed.acidity", boxwex=0.1)
```



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

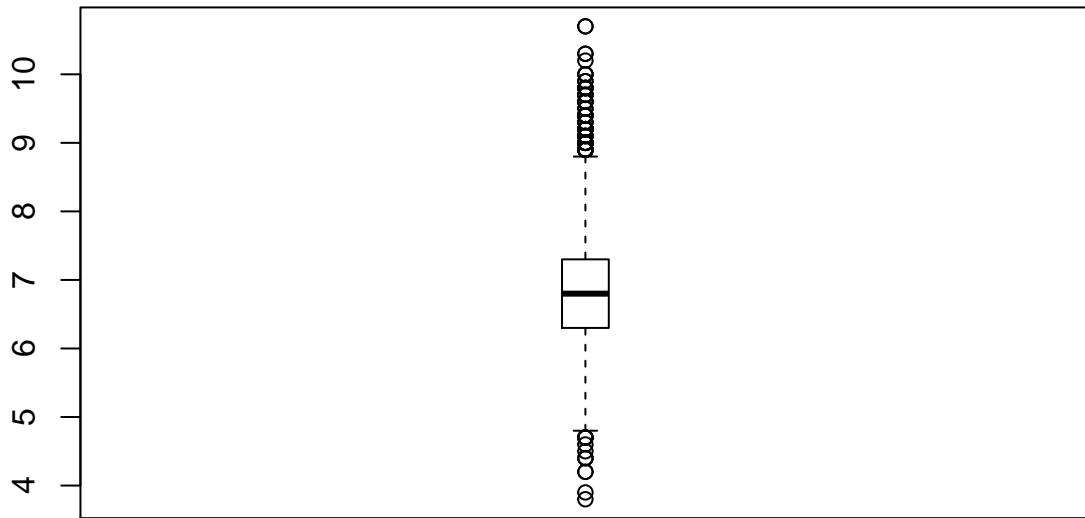
```
## [1] 9.8 9.8 10.2 9.1 10.0 9.2 9.2 9.0 9.1 9.2 10.3 9.4 9.2 9.8  
## [15] 9.6 9.2 9.0 9.3 9.2 9.1 8.9 9.8 8.9 9.2 9.7 9.4 10.3 9.6  
## [29] 9.0 9.7 9.2 9.4 9.6 9.2 9.0 9.2 10.7 10.7 9.0 9.2 9.8 9.2  
## [43] 14.2 8.9 8.9 9.1 9.1 9.8 9.0 9.3 8.9 9.0 9.0 8.9 9.0 9.3  
## [57] 9.2 9.6 9.4 9.4 10.0 8.9 8.9 10.0 9.2 9.2 9.2 9.9 9.5 9.0  
## [71] 9.0 8.9 9.5 11.8 9.4 9.1 9.8 9.9 9.2 8.9 9.2 9.4 9.4 9.4  
## [85] 4.6 8.9 9.4 9.2 9.2 9.8 9.0 9.0 9.0 8.9 8.9 4.5 9.2 9.6  
## [99] 4.2 9.7 9.7 9.0 4.2 9.4 8.9 8.9 8.9 4.7 4.7 3.8 4.4 4.7  
## [113] 9.0 9.0 4.7 4.4 3.9 4.7 4.4
```

```
extremos_acidez<-tail(sort(boxplot.stats(white_wine_data$fixed.acidity)$out),2)
```

Estos son 11.8, 14.2, procedemos a eliminarlos, dado que son solo 2 valores.

```
white_wine_data<-white_wine_data[white_wine_data$fixed.acidity<min(extremos_acidez),]  
boxplot(white_wine_data$fixed.acidity, main="fixed.acidity", boxwex=0.1)
```

## fixed.acidity

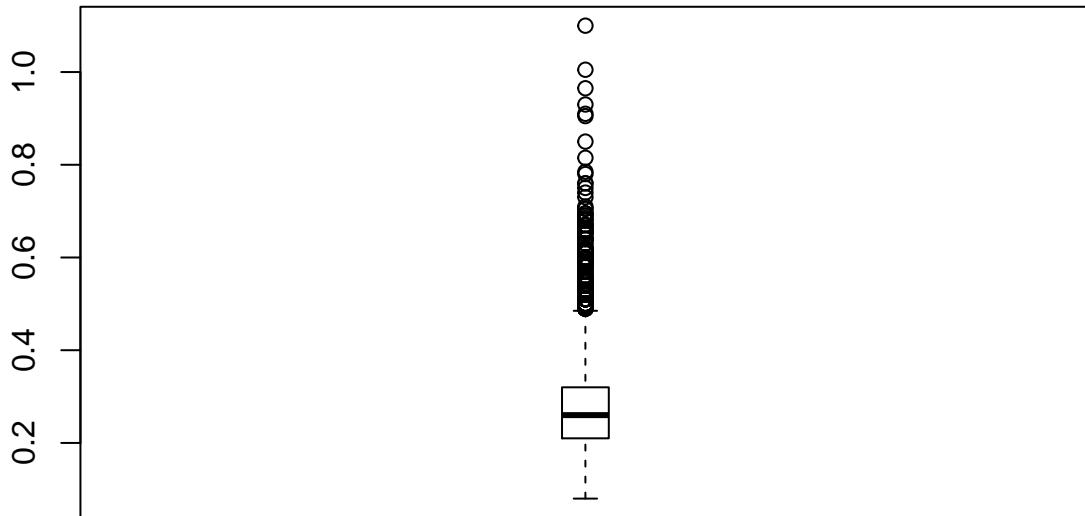


Continuamos por la acidez volátil **volatile.acidity**, vemos que tenemos dos valores que se escapan del conjunto de datos, pero en esta ocasión, parece ser un exceso de g/dm<sup>3</sup> de acidez volátil, dado que los valores están muy próximos a otros como 0.91, 0.93, 0.965, 1.005, 1.1, por lo tanto no debemos tratar valores extremos en este caso.

Esto puede significar que aunque el grueso de los valores se encuentre entre 0.2 y 0.4, puede que haya otros valores que están agrupados en otros niveles de acidez.

```
boxplot(white_wine_data$volatile.acidity, main="volatile.acidity", boxwex=0.1)
```

## volatile.acidity



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

```
## [1] 0.660 0.660 0.670 0.540 0.595 0.670 0.530 0.540 0.570 0.685 0.495
## [12] 0.640 0.520 0.580 0.585 0.590 0.600 0.580 0.590 0.550 0.905 0.550
## [23] 0.490 0.550 0.520 0.600 0.550 0.510 0.620 0.510 0.560 0.570 0.670
## [34] 0.500 0.560 0.560 0.655 0.595 0.705 0.520 0.550 0.600 0.640 0.680
```

```

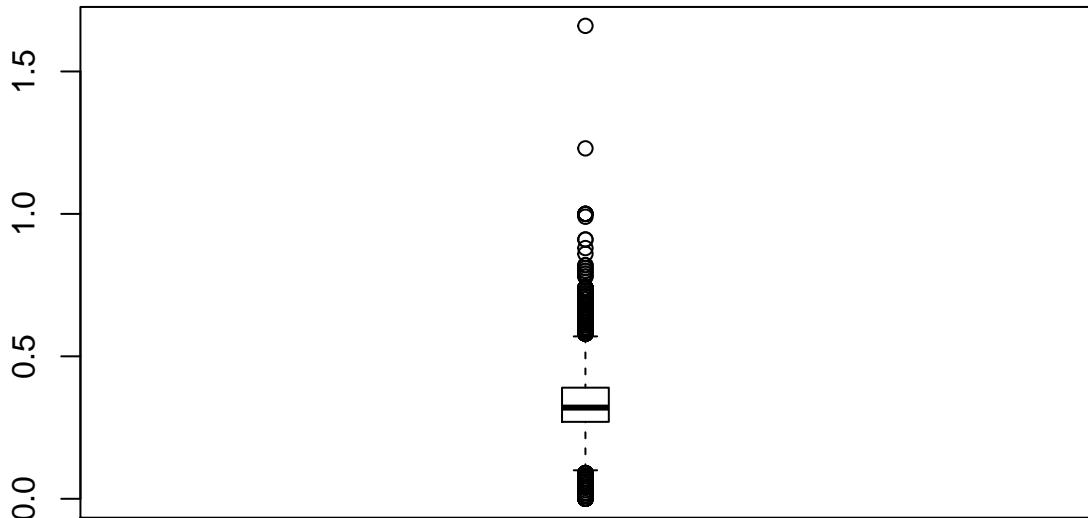
## [45] 0.490 0.510 0.550 0.520 0.500 0.550 0.600 0.610 0.610 0.610 0.660
## [56] 0.570 0.500 0.500 0.590 0.580 0.540 0.580 0.570 0.640 0.560 0.490
## [67] 0.490 0.670 0.550 0.560 0.520 0.520 0.850 0.510 0.620 0.510 0.530
## [78] 0.640 0.550 0.490 0.490 0.610 0.545 0.620 0.490 0.500 0.490 0.490
## [89] 0.550 0.490 0.910 0.530 0.490 0.710 1.005 0.490 0.550 0.550 0.760
## [100] 0.500 0.930 0.490 0.495 0.695 0.705 0.815 0.560 0.560 0.560 0.510
## [111] 0.540 0.540 0.500 0.615 0.500 0.520 0.600 0.680 0.655 0.510 0.510
## [122] 0.615 0.615 0.965 0.740 0.530 0.780 0.680 0.640 0.540 0.750 0.640
## [133] 0.640 0.655 0.580 0.520 0.530 0.600 0.530 0.580 0.670 0.610 0.730
## [144] 0.650 0.580 1.100 0.500 0.500 0.500 0.650 0.520 0.550 0.585 0.560
## [155] 0.555 0.555 0.540 0.610 0.550 0.530 0.660 0.615 0.500 0.620 0.500
## [166] 0.490 0.510 0.510 0.540 0.610 0.695 0.695 0.630 0.630 0.690 0.690
## [177] 0.590 0.620 0.785 0.760 0.500 0.540 0.520 0.600 0.540 0.530

```

Continuamos por el ácido cítrico **citric.acid**, vemos que hay algun valor que se sale del conjunto de valores, por ser un valor mas alto que el resto.

```
boxplot(white_wine_data$citric.acid, main="citric.acid", boxwex=0.1)
```

**citric.acid**



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

```

## [1] 0.62 0.04 0.59 0.07 0.03 0.61 0.62 0.63 0.61 0.62 0.63 0.66 0.66 0.00
## [15] 0.04 0.67 0.67 0.04 0.04 0.07 0.88 0.08 0.59 0.07 0.07 0.07 0.07 0.58
## [29] 0.70 0.00 0.00 0.60 0.07 0.09 0.04 0.62 0.58 0.62 0.70 0.62 0.62 0.58
## [43] 0.02 0.65 0.65 0.71 0.66 0.66 0.07 0.06 0.07 0.06 0.68 0.68 0.68 0.68
## [57] 0.06 0.72 0.69 0.58 0.70 1.66 0.04 0.63 0.60 0.00 0.08 0.58 0.58 0.05
## [71] 0.58 0.00 0.00 0.65 0.58 0.00 0.05 0.05 0.62 0.62 0.58 0.58 1.00 0.09
## [85] 0.01 0.71 0.71 0.60 0.06 0.74 0.81 0.69 0.58 0.69 0.00 0.07 0.64 0.72
## [99] 0.73 0.65 0.68 0.65 0.74 0.71 0.59 0.68 0.08 0.72 0.64 0.02 0.74 0.74
## [113] 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74
## [127] 0.74 0.74 0.74 0.74 0.74 0.99 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74
## [141] 0.74 0.74 0.74 0.74 0.74 0.74 0.01 0.74 0.01 0.74 0.74 1.00 0.04 0.58
## [155] 0.07 1.00 0.00 0.58 0.61 0.61 0.61 0.02 0.67 0.67 0.67 0.58 0.65 0.58
## [169] 0.09 0.08 0.71 0.04 0.03 0.05 0.64 0.64 0.58 0.58 0.81 0.58 0.61 0.62
## [183] 0.59 0.00 0.04 0.63 0.73 0.68 0.09 0.78 0.79 0.09 0.64 0.65 0.65 0.00
## [197] 0.73 0.73 0.64 0.60 0.71 0.72 0.82 0.07 0.58 0.58 1.00 0.66 0.80 0.80

```

```

## [211] 1.23 0.59 0.02 0.00 1.00 0.62 0.00 0.71 0.71 0.71 0.61 0.61 0.00 0.60
## [225] 0.58 0.09 0.09 0.72 0.62 0.62 0.79 0.82 0.67 0.01 0.01 0.86 0.61 0.02
## [239] 0.05 0.00 0.69 0.69 0.59 0.01 0.66 0.66 0.78 0.00 0.04 0.91 0.91 0.06
## [253] 0.06 0.04 0.04 0.74 0.09 0.09 0.60 0.62 0.73 0.00 0.09 0.00 0.09 0.67
## [267] 0.01 0.09 0.00 0.02

extremos_citrico<-max(boxplot.stats(white_wine_data$citric.acid)$out)

```

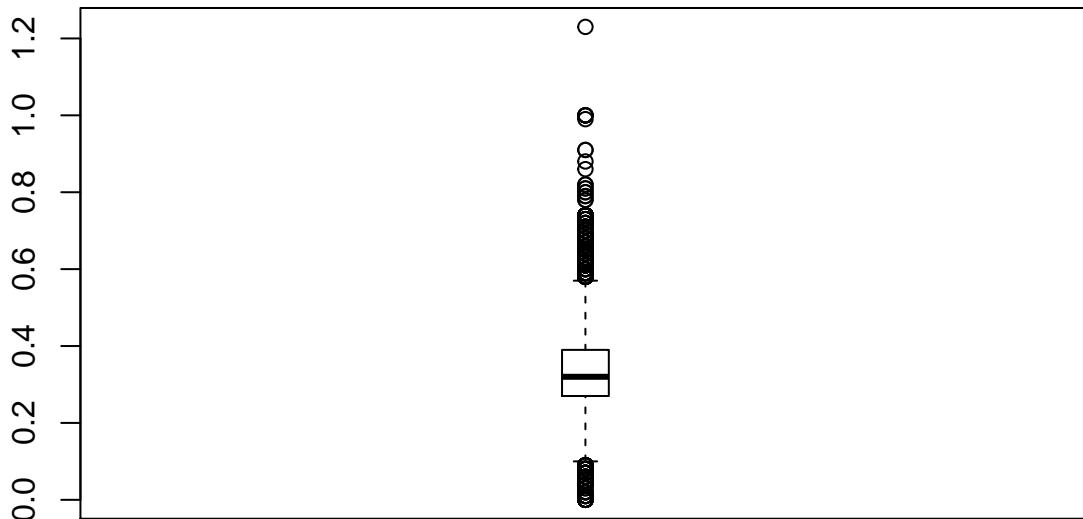
Estos son 1.66, procedemos a eliminarlo, dado que es únicamente 1 valor.

```

white_wine_data<-white_wine_data[white_wine_data$citric.acid<min(extremos_citrico),]
boxplot(white_wine_data$citric.acid, main="citric.acid", boxwex=0.1)

```

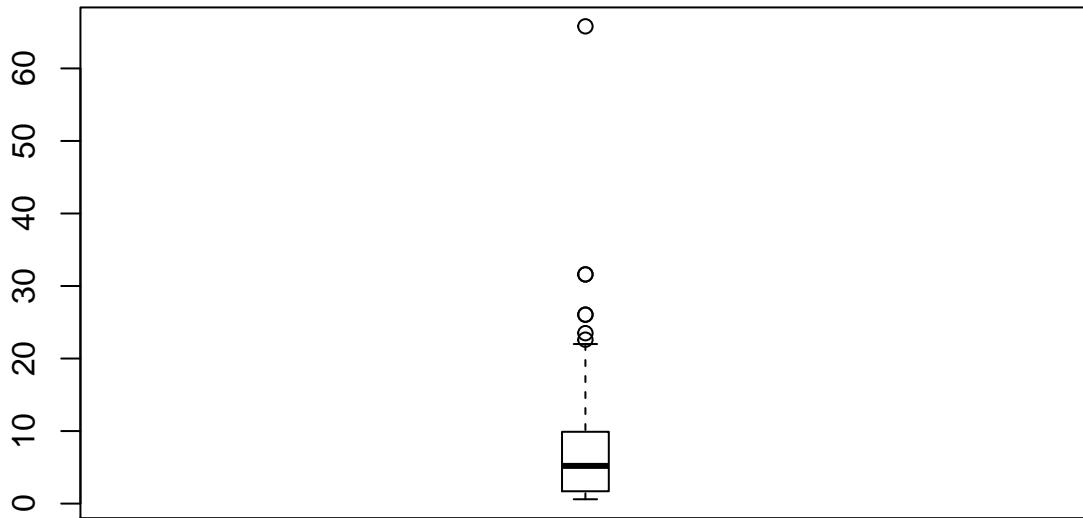
**citric.acid**



Continuamos por el azucar residual **residual.sugar**, vemos que hay algun valor que se sale del conjunto de valores, por ser un valor mas alto que el resto.

```
boxplot(white_wine_data$residual.sugar, main="residual.sugar", boxwex=0.1)
```

## residual.sugar

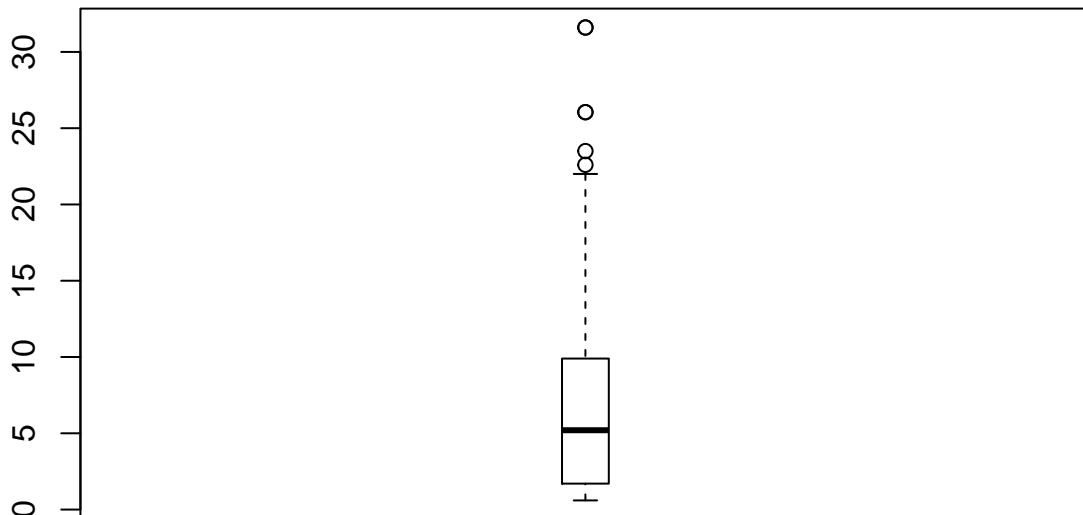


```
boxplot.stats(white_wine_data$residual.sugar)$out  
## [1] 23.50 31.60 31.60 65.80 26.05 26.05 22.60  
extremos_sugar<-max(boxplot.stats(white_wine_data$residual.sugar)$out)
```

Estos son 65.8, procedemos a eliminarlo, dado que es únicamente 1 valor.

```
white_wine_data<-white_wine_data[white_wine_data$residual.sugar<min(extremos_sugar),]  
boxplot(white_wine_data$residual.sugar, main="residual.sugar", boxwex=0.1)
```

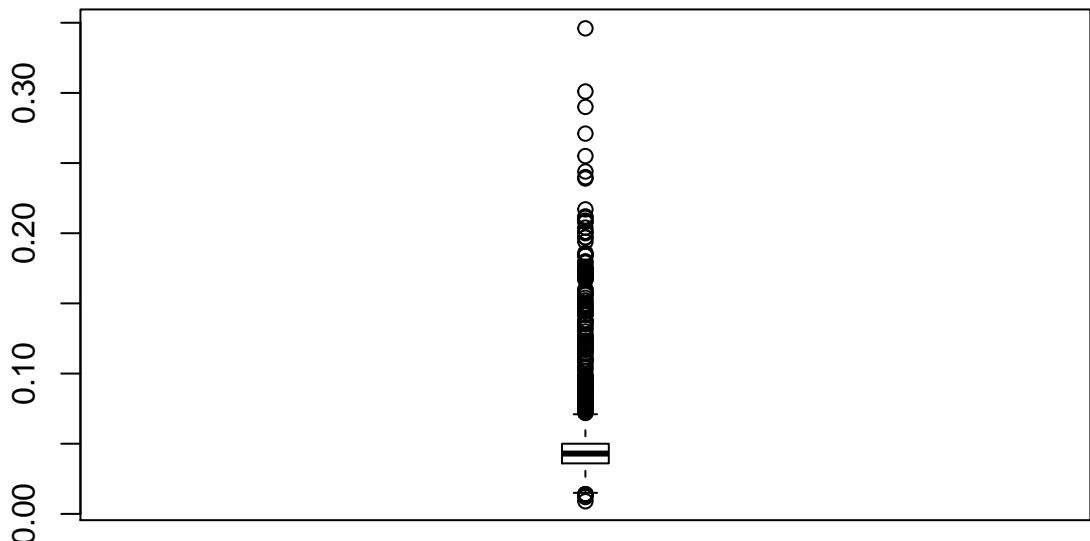
## residual.sugar



Continuamos por la cantidad de sal **chlorides**, vemos que hay algun valor que se sale del conjunto de valores, pero no excesivamente, por lo que puede no ser un valor extremo, sino un indice alto de sal.

```
boxplot(white_wine_data$chlorides, main="chlorides", boxwex=0.1)
```

## chlorides



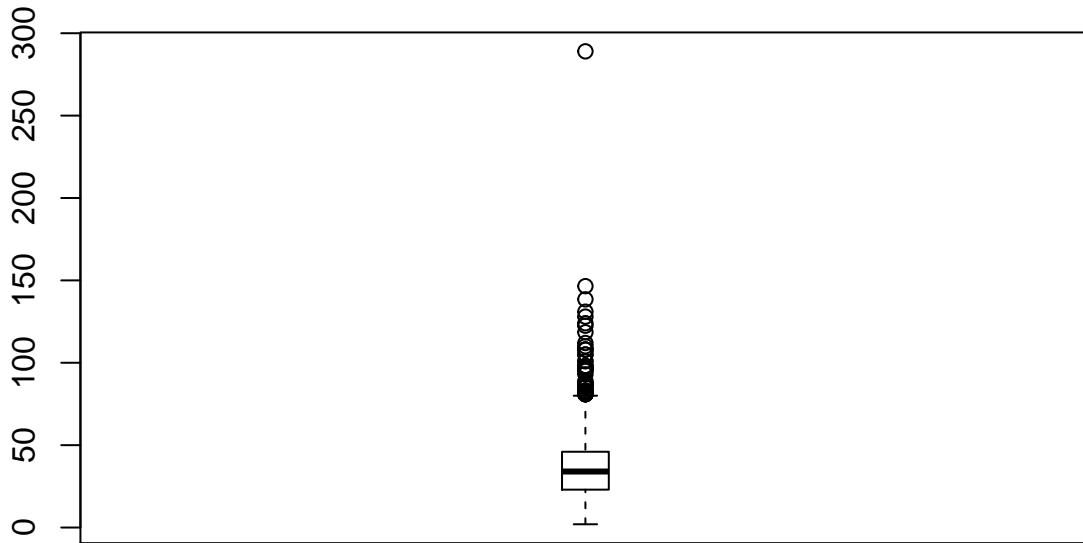
```
boxplot.stats(white_wine_data$chlorides)$out
```

```
## [1] 0.074 0.080 0.172 0.173 0.147 0.092 0.082 0.092 0.200 0.197 0.197
## [12] 0.074 0.132 0.089 0.108 0.081 0.073 0.346 0.090 0.114 0.186 0.180
## [23] 0.084 0.083 0.096 0.094 0.240 0.290 0.185 0.110 0.078 0.130 0.135
## [34] 0.115 0.072 0.170 0.080 0.119 0.126 0.150 0.152 0.088 0.244 0.137
## [45] 0.093 0.077 0.079 0.073 0.072 0.076 0.201 0.201 0.074 0.074 0.301
## [56] 0.138 0.169 0.083 0.093 0.168 0.122 0.172 0.167 0.239 0.076 0.138
## [67] 0.137 0.123 0.123 0.133 0.073 0.073 0.211 0.123 0.123 0.255 0.204
## [78] 0.208 0.083 0.080 0.076 0.086 0.084 0.084 0.168 0.160 0.179 0.076
## [89] 0.076 0.087 0.217 0.094 0.157 0.157 0.148 0.158 0.157 0.168 0.157
## [100] 0.092 0.099 0.084 0.085 0.091 0.093 0.080 0.095 0.096 0.096 0.147
## [111] 0.142 0.079 0.074 0.075 0.074 0.121 0.121 0.079 0.079 0.014 0.156
## [122] 0.012 0.119 0.119 0.081 0.170 0.171 0.082 0.083 0.083 0.152 0.169
## [133] 0.073 0.014 0.078 0.112 0.154 0.126 0.126 0.104 0.142 0.102 0.184
## [144] 0.184 0.096 0.076 0.146 0.117 0.117 0.118 0.014 0.085 0.087 0.085
## [155] 0.087 0.076 0.088 0.160 0.167 0.014 0.009 0.098 0.098 0.086 0.086
## [166] 0.194 0.094 0.013 0.144 0.149 0.185 0.084 0.175 0.090 0.098 0.110
## [177] 0.110 0.095 0.174 0.097 0.142 0.145 0.208 0.209 0.105 0.086 0.176
## [188] 0.176 0.108 0.096 0.271 0.120 0.212 0.094 0.094 0.117 0.173 0.074
## [199] 0.076 0.076 0.175 0.174 0.075 0.127 0.127 0.096 0.136
```

Proseguimos por la cantidad libre de dióxido de sulfuro **free.sulfur.dioxide**, vemos que hay algun valor que se sale del conjunto de valores.

```
boxplot(white_wine_data$free.sulfur.dioxide, main="free.sulfur.dioxide", boxwex=0.1)
```

## free.sulfur.dioxide



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

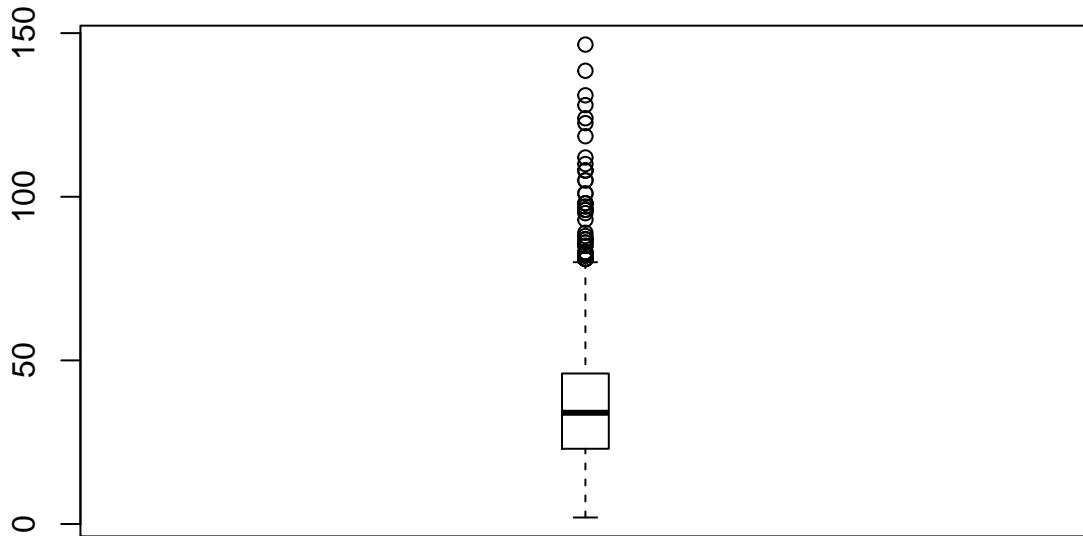
```
## [1] 81.0 82.0 131.0 82.5 87.0 87.0 83.0 122.5 83.0 81.0 88.0  
## [12] 82.0 118.5 81.0 96.0 83.0 83.0 146.5 128.0 110.0 85.0 89.0  
## [23] 86.0 86.0 96.0 96.0 93.0 85.0 81.0 138.5 95.0 124.0 87.0  
## [34] 87.0 105.0 105.0 101.0 101.0 108.0 108.0 98.0 98.0 112.0 108.0  
## [45] 98.0 81.0 81.0 81.0 289.0 97.0
```

```
extremos_sulfurdioxide<-max(boxplot.stats(white_wine_data$free.sulfur.dioxide)$out)
```

Estos son 289, procedemos a eliminarlo, dado que es únicamente 1 valor.

```
white_wine_data<-white_wine_data[white_wine_data$free.sulfur.dioxide<min(extremos_sulfurdioxide),]  
boxplot(white_wine_data$free.sulfur.dioxide, main="free.sulfur.dioxide", boxwex=0.1)
```

## free.sulfur.dioxide

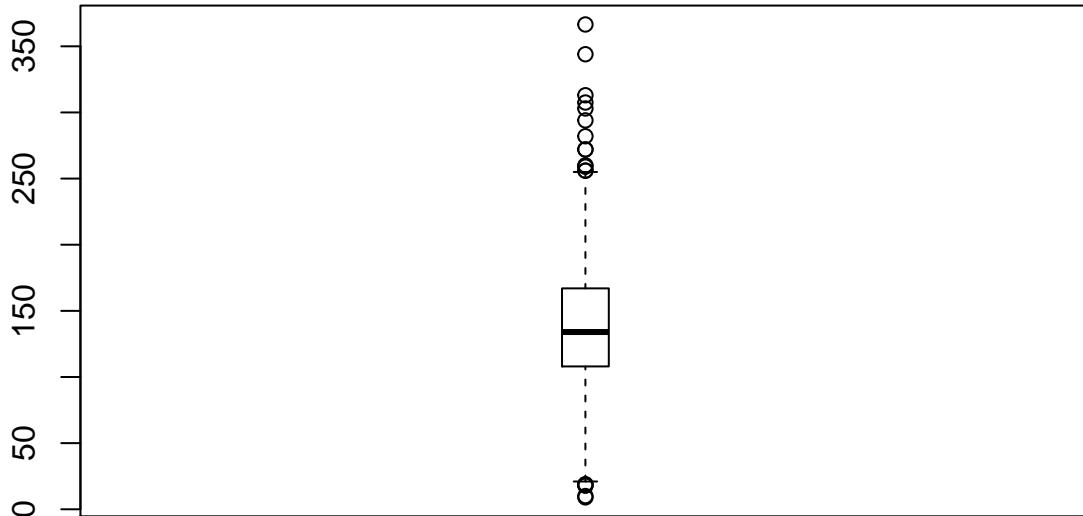


Proseguimos por la cantidad total de dióxido de sulfuro **total.sulfur.dioxide**, vemos que hay algun valor

que se sale del conjunto de valores, pero no los vamos a eliminar, dado a que hay algunos valores que tambien se aproximan a dichos valores.

```
boxplot(white_wine_data$total.sulfur.dioxide, main="total.sulfur.dioxide", boxwex=0.1)
```

**total.sulfur.dioxide**



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

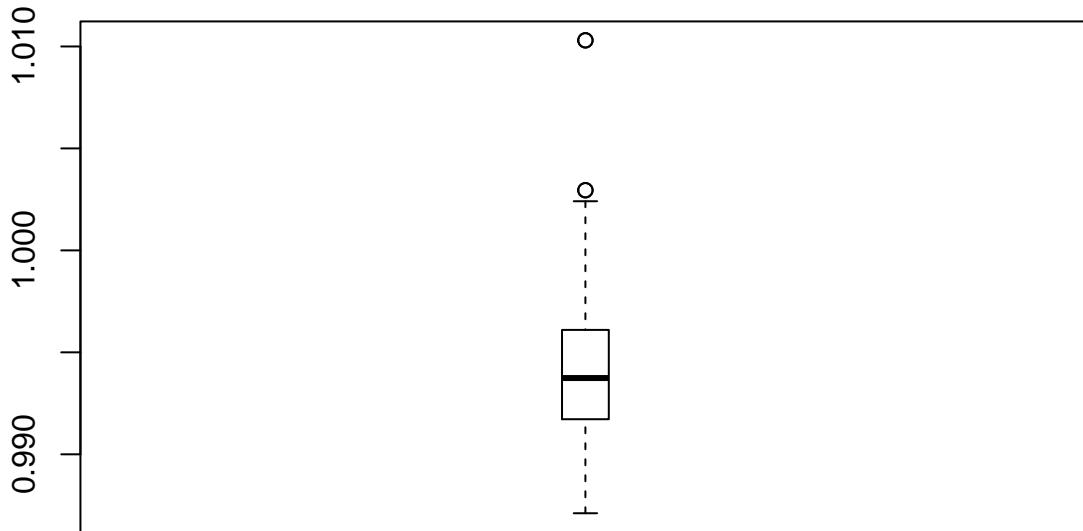
```
## [1] 272.0 313.0 260.0 19.0 366.5 307.5 256.0 256.0 344.0 282.0 303.0  
## [12] 272.0 18.0 18.0 294.0 9.0 10.0 259.0
```

```
extremos_total_sulfurdioxide<-tail(sort(boxplot.stats(white_wine_data$total.sulfur.dioxide)$out))
```

Proseguimos por la densidad **density**, vemos que hay algun valor que se sale del rango de valores.

```
boxplot(white_wine_data$density, main="density", boxwex=0.1)
```

**density**



```

boxplot.stats(white_wine_data$density)$out
## [1] 1.01030 1.01030 1.00295 1.00295
extremos_density<-tail(sort(boxplot.stats(white_wine_data$density)$out))

```

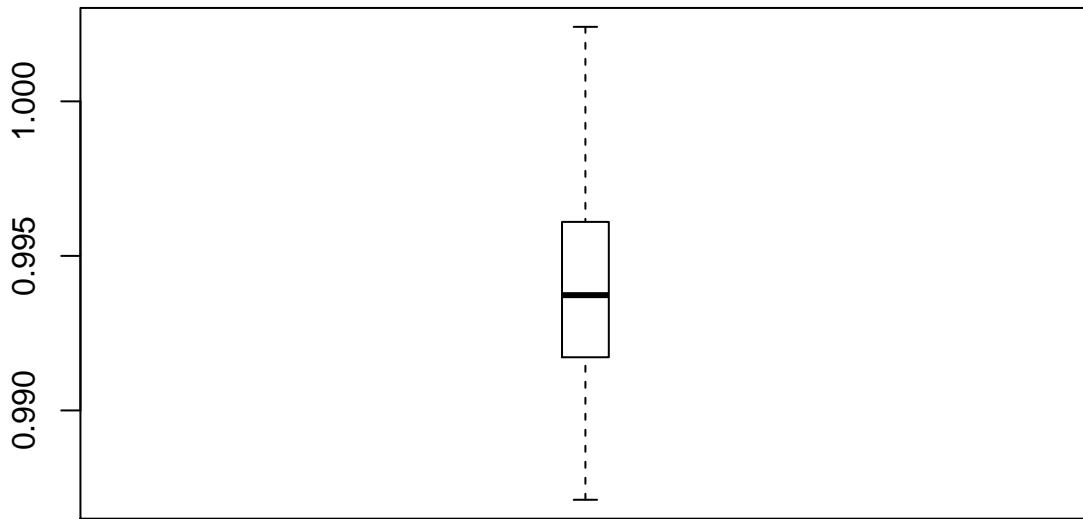
Estos son 1.00295, 1.00295, 1.0103, 1.0103, procedemos a eliminarlo, dado que son pocos valores.

```

white_wine_data<-white_wine_data[white_wine_data$density<min(extremos_density),]
boxplot(white_wine_data$density, main="density", boxwex=0.1)

```

**density**



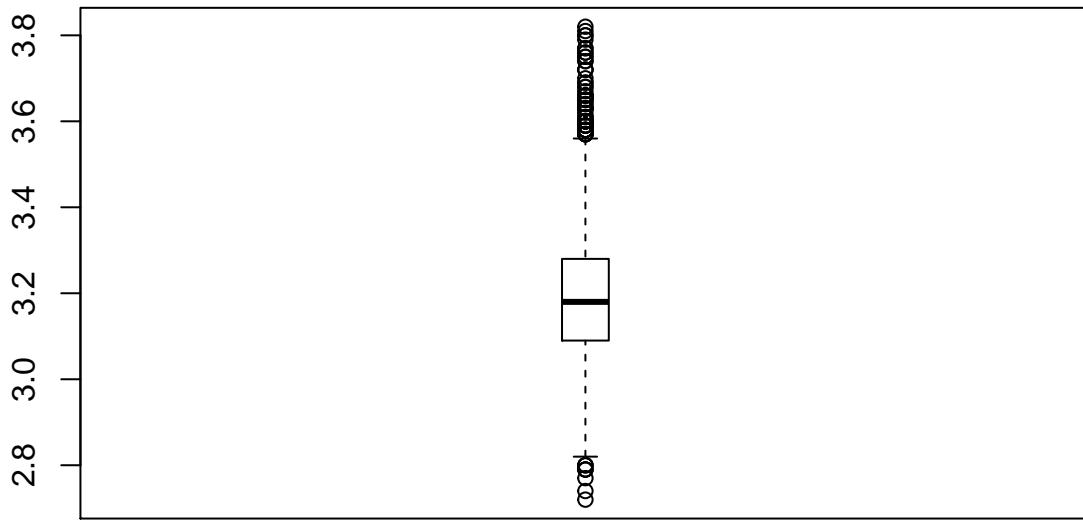
Proseguimos por el ph **pH**, vemos que hay algun valor que no está donde la nube de puntos media, pero si esta dentro de los valores posibles del ph entre 0 y 14. Por lo que no eliminaremos ningun valor.

```

boxplot(white_wine_data$pH, main="pH", boxwex=0.1)

```

**pH**



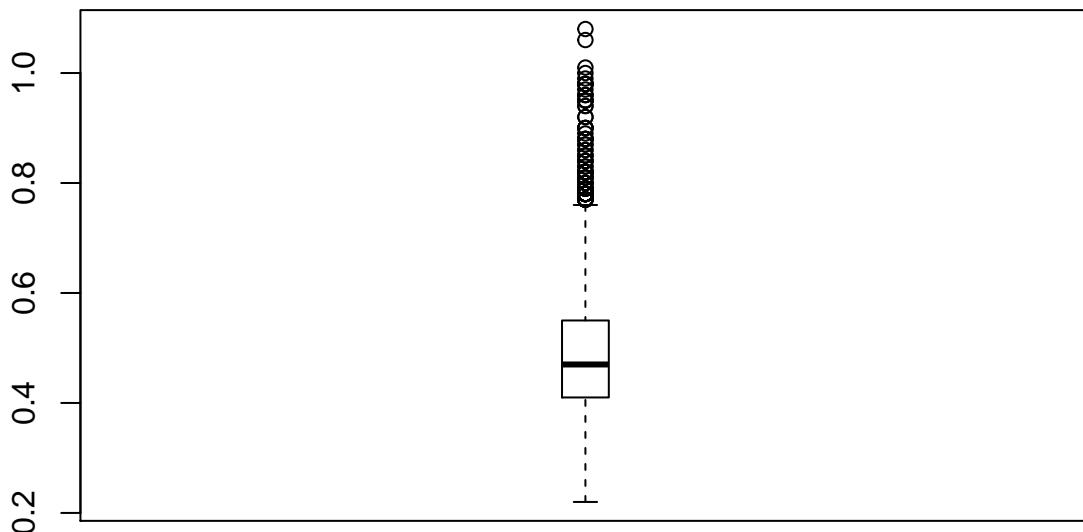
```
boxplot.stats(white_wine_data$pH)$out
```

```
## [1] 3.69 3.63 3.72 3.61 3.64 3.64 3.72 3.72 3.58 3.58 3.66 3.59 2.74 3.82  
## [15] 3.81 3.65 3.65 3.59 3.77 3.62 3.63 3.58 3.58 3.65 3.74 2.80 3.60 3.60  
## [29] 2.72 3.60 2.79 2.79 3.57 3.80 3.60 3.60 3.68 3.63 3.63 2.77 3.63 3.60  
## [43] 3.60 3.61 3.61 3.59 3.79 3.59 3.68 3.59 3.66 3.70 3.74 3.80 3.57 3.57  
## [57] 3.57 3.65 3.58 2.80 3.77 3.76 3.69 3.66 3.59 2.79 3.75 3.63 3.75 3.76  
## [71] 3.66 3.66 2.80 3.67 3.57
```

Proseguimos por la cantidad de sulfatos **sulphates**, vemos que hay algun valor que no está donde la nube de puntos media, no obstante no parece ser un outlier, ya que hay otros puntos cercanos a este.

```
boxplot(white_wine_data$sulphates, main="sulphates", boxwex=0.1)
```

**sulphates**



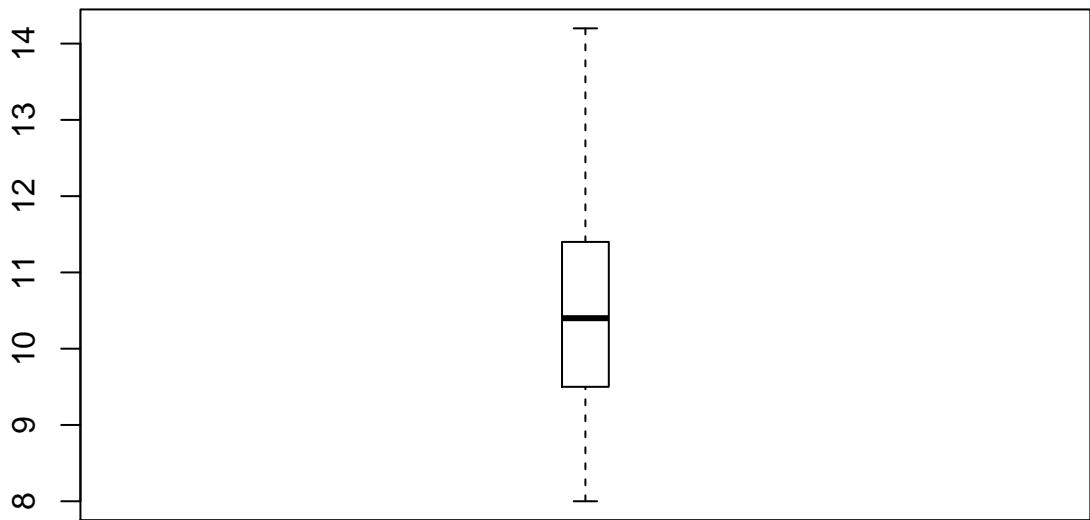
```
boxplot.stats(white_wine_data$sulphates)$out
```

```
## [1] 0.77 0.84 0.77 0.79 0.85 0.78 0.79 0.79 0.79 0.77 0.78 0.85 0.96 0.97  
## [15] 0.82 0.82 0.77 0.95 0.95 0.77 0.95 0.82 0.82 0.90 0.88 0.88 0.79 0.80  
## [29] 0.80 0.78 0.78 0.87 0.86 0.90 0.90 0.78 0.79 0.81 0.81 0.77 0.82 0.79  
## [43] 0.79 0.77 0.82 0.92 0.79 0.79 0.82 0.82 0.82 0.82 0.82 0.79 0.78 0.79  
## [57] 0.77 0.77 0.77 0.98 1.06 0.88 0.88 0.88 0.80 0.78 1.00 0.80 0.90 0.90  
## [71] 0.89 0.94 0.99 0.86 0.84 0.95 0.84 0.84 0.81 0.80 0.87 0.82 0.78 0.78  
## [85] 0.78 0.78 0.78 0.77 0.85 0.78 0.78 0.88 0.88 0.78 0.78 0.78 0.78 0.79  
## [99] 0.77 0.77 0.83 0.83 0.81 0.81 0.98 0.98 0.98 0.98 0.79 0.79 0.78 0.82  
## [113] 0.98 0.77 0.96 1.01 0.77 0.96 0.77 0.92 0.94 0.95 1.08 0.79
```

Comprobamos outliers en la cantidad de alcohol **alcohol** y vemos que no hay outliers

```
boxplot(white_wine_data$alcohol, main="alcohol", boxwex=0.1)
```

## alcohol



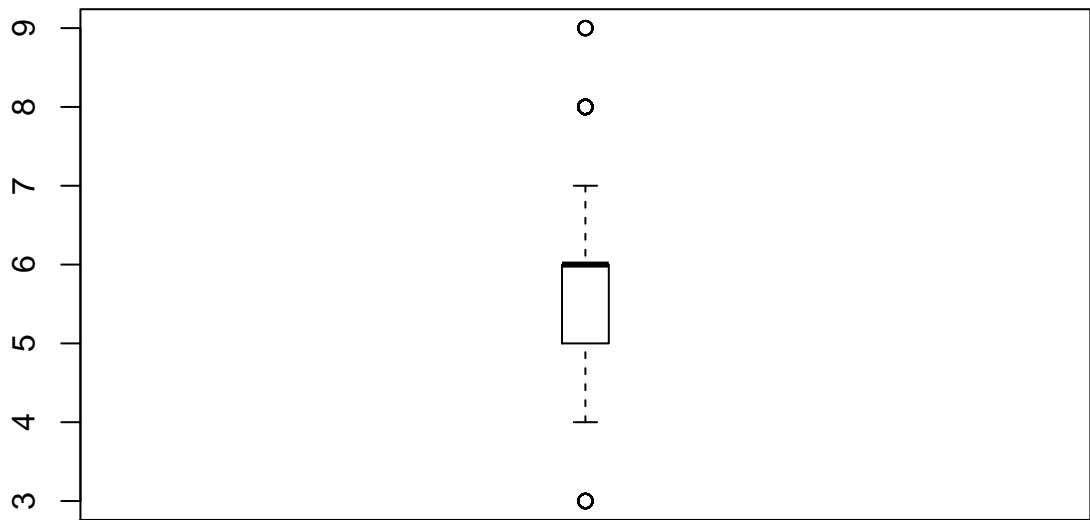
```
boxplot.stats(white_wine_data$alcohol)$out
```

```
## numeric(0)
```

Comprobamos outliers en la calidad **quality** y vemos que no hay outliers, dado que se encuentra dentro del rango de valores del 0 al 10.

```
boxplot(white_wine_data$quality, main="quality", boxwex=0.1)
```

## quality



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

```
## [1] 8 8 8 8 8 8 8 8 8 3 3 8 8 8 3 8 8 8 3 8 8 8 8 8 3 9 8 8 8 9 9 8 8 8  
## [36] 8 8 8 8 8 8 3 9 8 8 8 8 8 3 8 8 8 8 8 8 8 8 8 8 8 3 8 8 8 8 8 8 8 8 8  
## [71] 8 8 8 8 3 8 3 8 8 9 8 8 8 3 8 8 8 8 3 8 8 8 8 8 8 3 8 8 8 8 8 8 8 3 8 8 8  
## [106] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 3 8 8 8  
## [141] 8 8 8 8 3 8 8 8 3 8 8 8 3 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
```

```
## [176] 8 8 3 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8
```

## Sección 4

### 4. Análisis de los datos

#### 4.1. Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).

No hemos segmentado los datos en distintos tipos, dado que únicamente hemos tomado el dataset de vinos blancos, en lugar de combinar tintos y blancos, por tanto al trabajar únicamente con un tipo de vino, no segmentaremos los datos, ya que queremos emplear cada propiedad que atribuye a un vino su calidad.

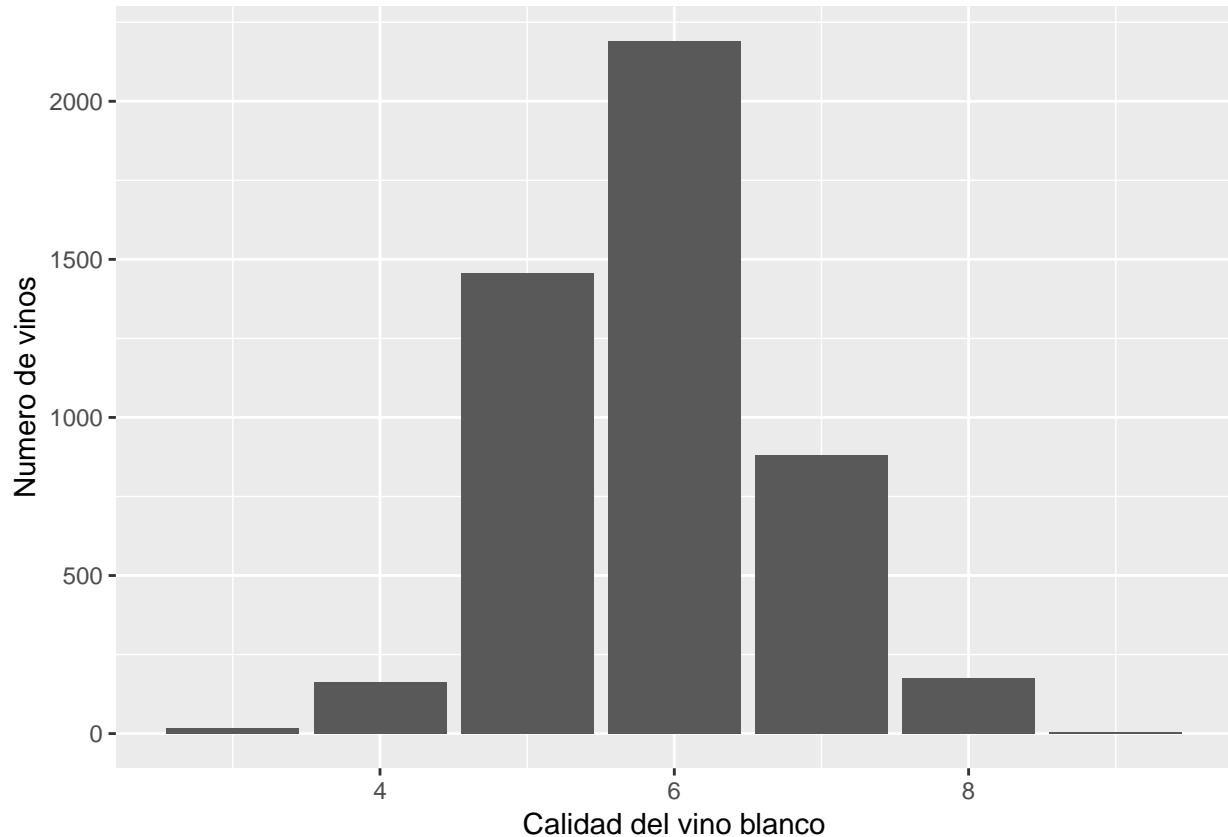
La planificación de análisis a realizar es ver si alguna de las propiedades del vino, es significativamente influyente para determinar la calidad del vino.

Analizaremos por tanto la normalidad de los datos así como la relación que hay entre cada una de las variables, hasta obtener un modelo que nos permita determinar la calidad de estos vinos blancos.

Empezamos por ver como se distribuyen los datos en la calidad:

```
library(ggplot2)
```

```
## Warning: As of rlang 0.4.0, dplyr must be at least version 0.8.0.  
## x dplyr 0.7.8 is too old for rlang 0.4.2.  
## i Please update dplyr with `install.packages("dplyr")`.  
print(" Numero de vinos por su puntuacion:")  
  
## [1] " Numero de vinos por su puntuacion:  
table(white_wine_data$quality)  
  
##  
##   3    4    5    6    7    8    9  
## 18 163 1457 2191  880  175    5  
ggplot(white_wine_data,aes(quality)) + geom_histogram(stat="count") +  
  xlab("Calidad del vino blanco") + ylab("Numero de vinos")  
  
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



La calidad del vino está distribuida en ratios de vino pobre inferior a 5, vino normal [5,7] y vino bueno superior a 7.

La calidad del vino sigue una distribución normal, en la que hay mas vinos que son de calidad normal, que superior o pésima.

Comprobemos cual es por tanto la categoría global de los vinos blancos:

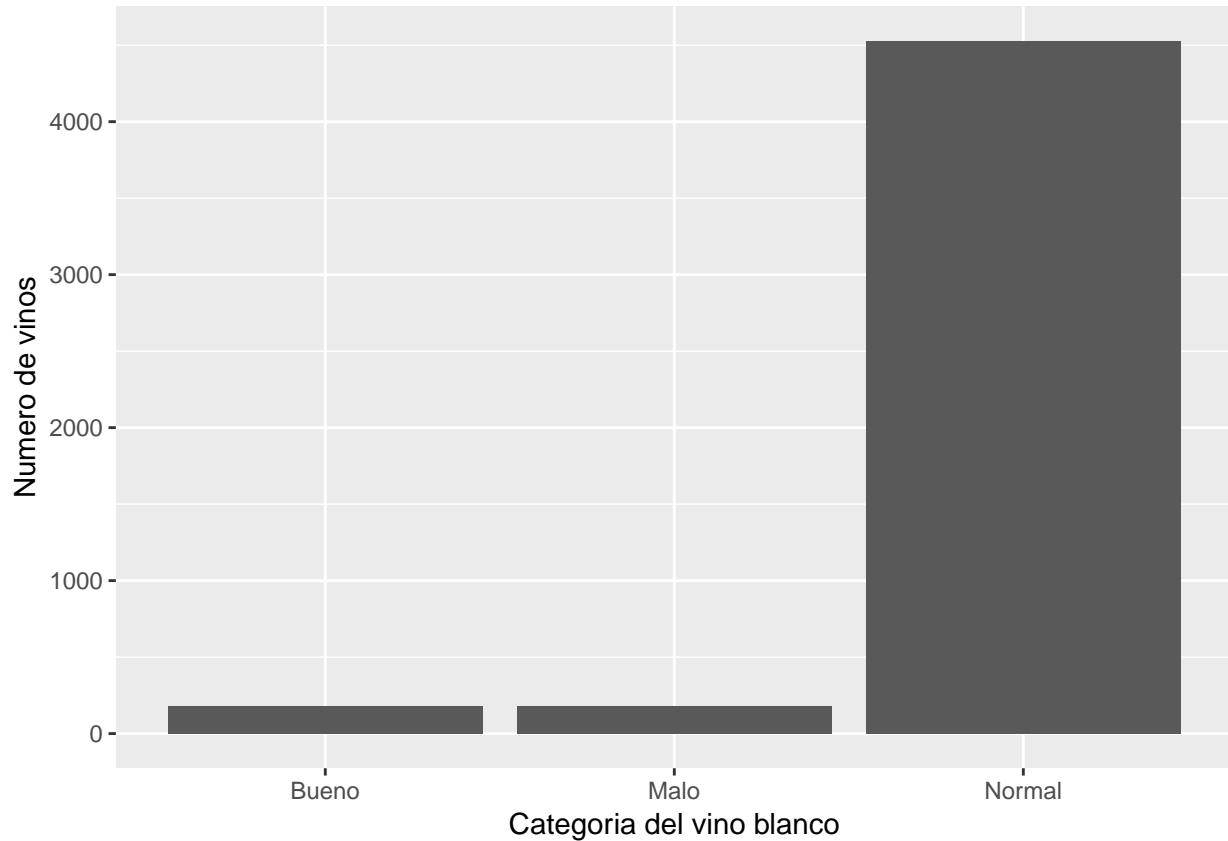
```
print(" Numero de vinos por su categoría:")
## [1] " Numero de vinos por su categoría:"
```

```
table(white_wine_data$category)
```

```
##
##   Bueno    Malo Normal
##     180     181    4528
```

```
ggplot(white_wine_data,aes(category)) + geom_histogram(stat="count") +
  xlab("Categoria del vino blanco") + ylab("Número de vinos")
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```



Observamos que tenemos tantos vinos de mala calidad, como de buena calidad y el grueso de los vinos son vinos normales, lo cual tiene sentido, ya que la mayor parte de los vinos deberian acercarse a tener las mismas propiedades o muy cercanas, sin embargo los vinos buenos tengan alguna propiedad que varia y lo hace un mejor vino, con lo que quizas incluso su precio sea mayor.

#### 4.1.a) Análisis Multivariante

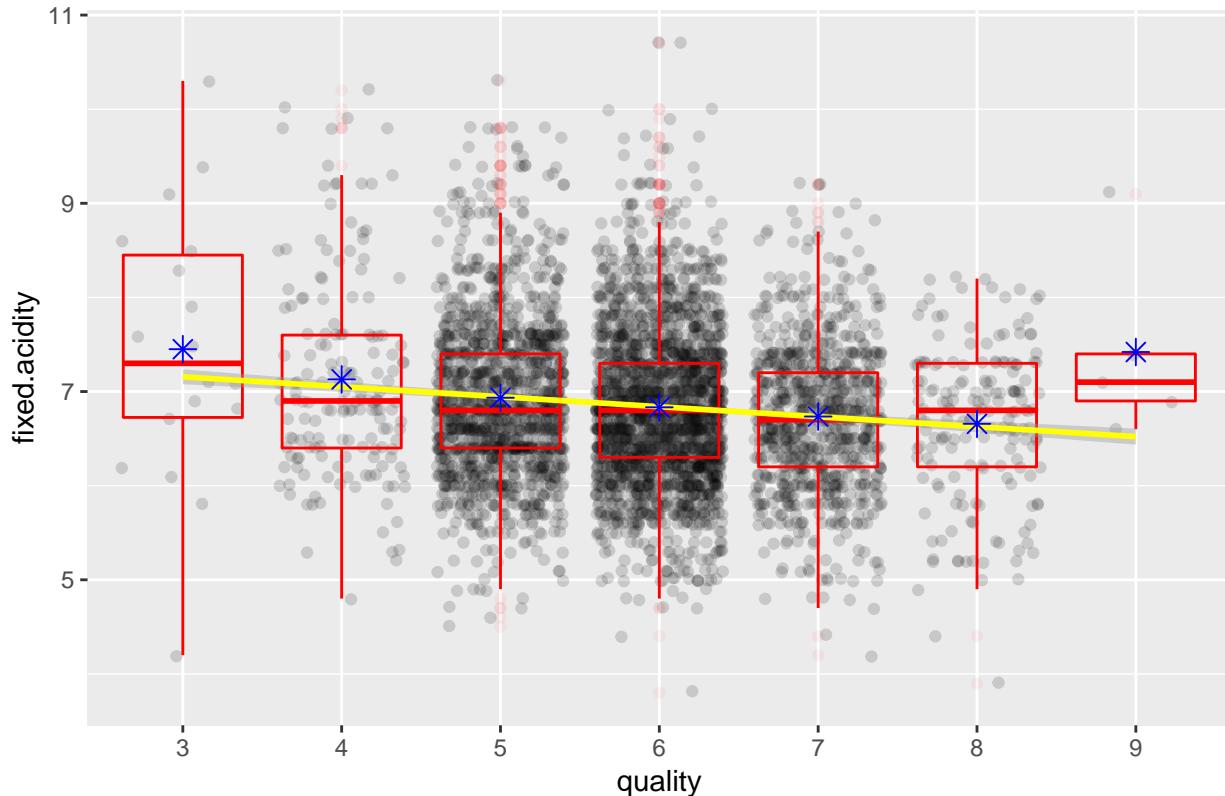
Utilizaremos un visual de cada variable contra la calidad, así como la correlación de Pearson, que son: - Correlación Débil  $0.1 < r < 0.3$  - Correlación Moderada  $0.3 < r < 0.5$  - Correlación Fuerte  $r > 0.5$

Veamos como influye la variable **fixed.acidity** en la calidad del vino:

```
library(ggplot2)

ggplot(data = white_wine_data, aes(x = factor(quality), y = fixed.acidity)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,fixed.acidity),method='lm',color='yellow')+
  xlab('quality')+ ylab('fixed.acidity') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs fixed.acidity')
```

## Calidad del vino vs fixed.acidity



```
corr_quality_fixed_acidity<-cor.test(white_wine_data$quality,white_wine_data$fixed.acidity,method='pearson')
corr_quality_fixed_acidity
```

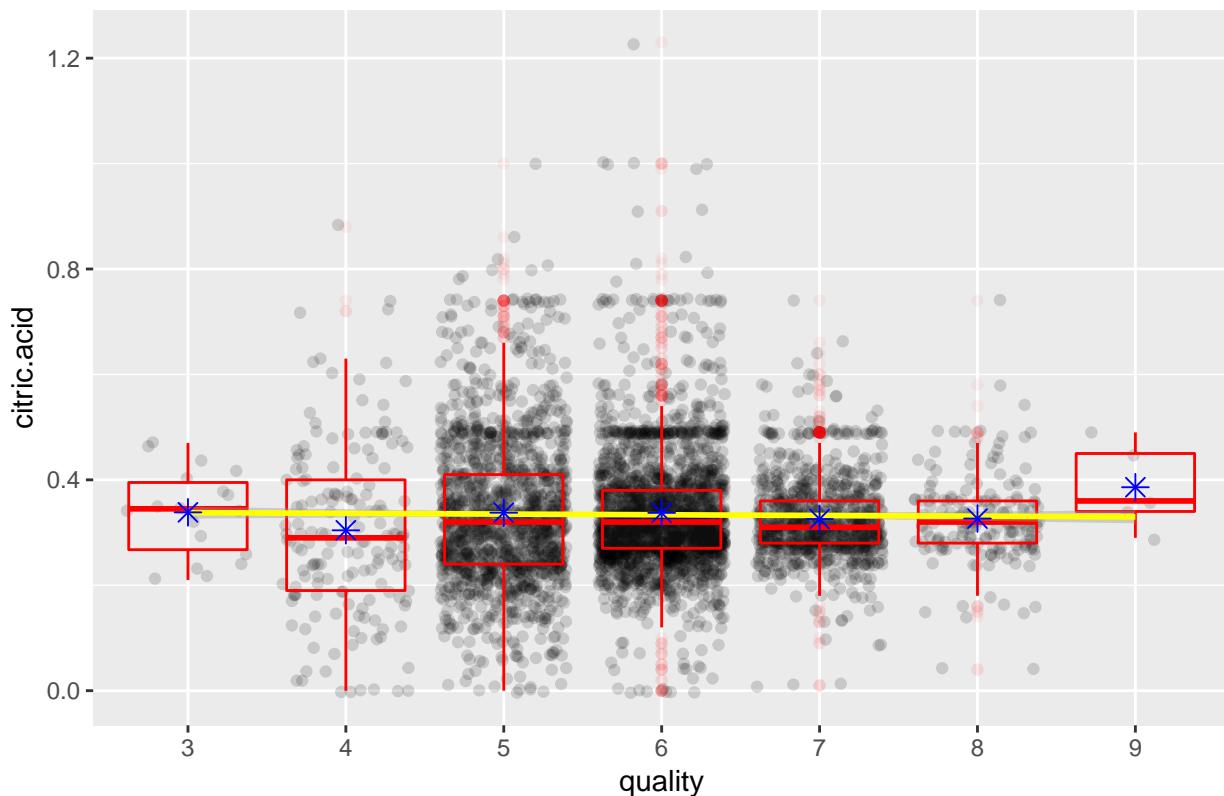
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$fixed.acidity
## t = -7.9006, df = 4887, p-value = 3.405e-15
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.13989216 -0.08453421
## sample estimates:
##       cor
## -0.1123003
```

Como podemos observar, la cantidad de **fixed.acidity** o acidez fija, no es un factor que tenga repercusión en la calidad del vino, dado que la calidad del vino no parece verse afectada por este factor, al permanecer los valores de acidez fija, cercanos a la media para cada tipo de calidad. Además su correlación es débil, teniendo un valor de -0.1123003, por lo que no parece que tenga mucha repercusión.

Veamos como influye la variable **citric.acid** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = citric.acid)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,citric.acid),method='lm',color='yellow')+
  xlab('quality')+ ylab('citric.acid') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs citric.acid')
```

## Calidad del vino vs citric.acid



```
corr_quality_citric<-cor.test(white_wine_data$quality,white_wine_data$citric.acid,method='pearson')
corr_quality_citric
```

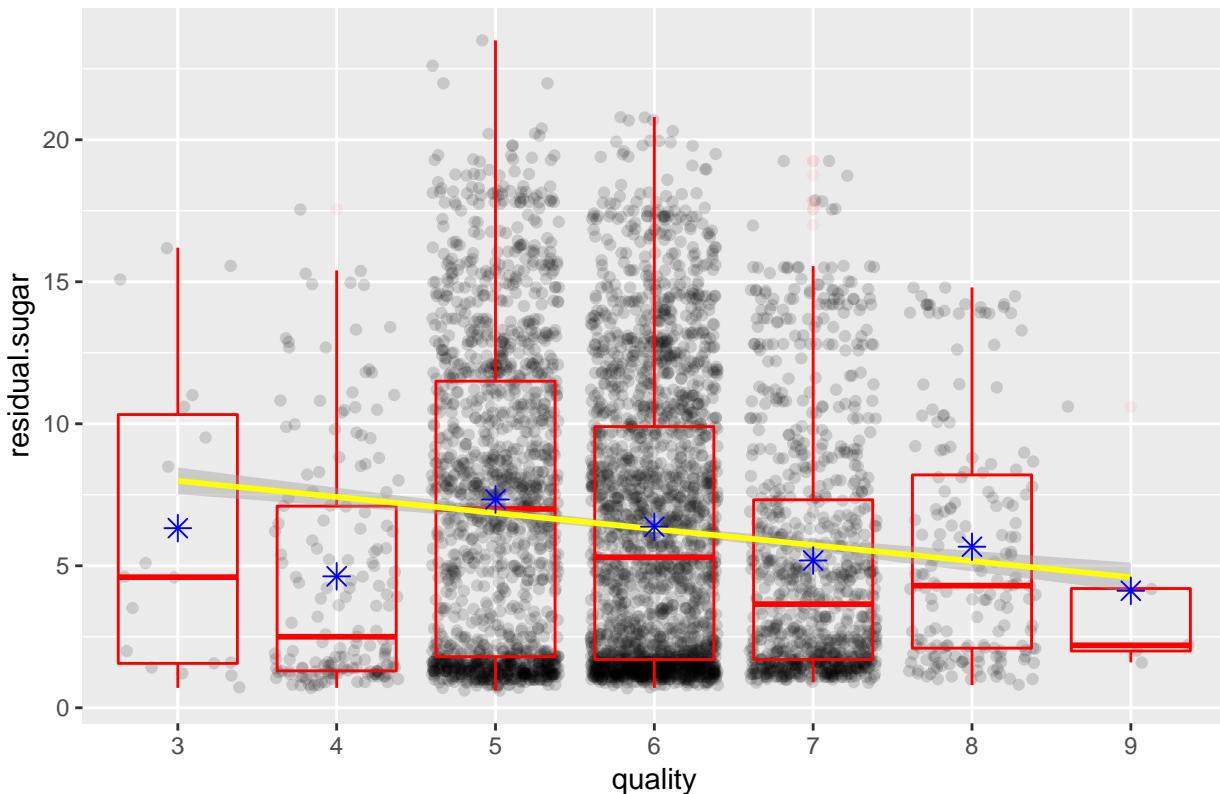
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$citric.acid
## t = -0.69383, df = 4887, p-value = 0.4878
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.03794623 0.01811270
## sample estimates:
## cor
## -0.009924564
```

Como podemos observar, la cantidad de **citric.acid** o citrico, tiene una correlación muy débil entre citrico y calidad, su correlación es -0.0099246, además el rango de sus valores se encuentran entre 0.1 y 0.7.

Veamos como influye la variable **residual.sugar** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = residual.sugar)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,residual.sugar),method='lm',color='yellow')+
  xlab('quality')+ ylab('residual.sugar') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs residual.sugar')
```

## Calidad del vino vs residual.sugar



```
corr_quality_sugar<-cor.test(white_wine_data$quality,white_wine_data$residual.sugar,method='pearson')
corr_quality_sugar
```

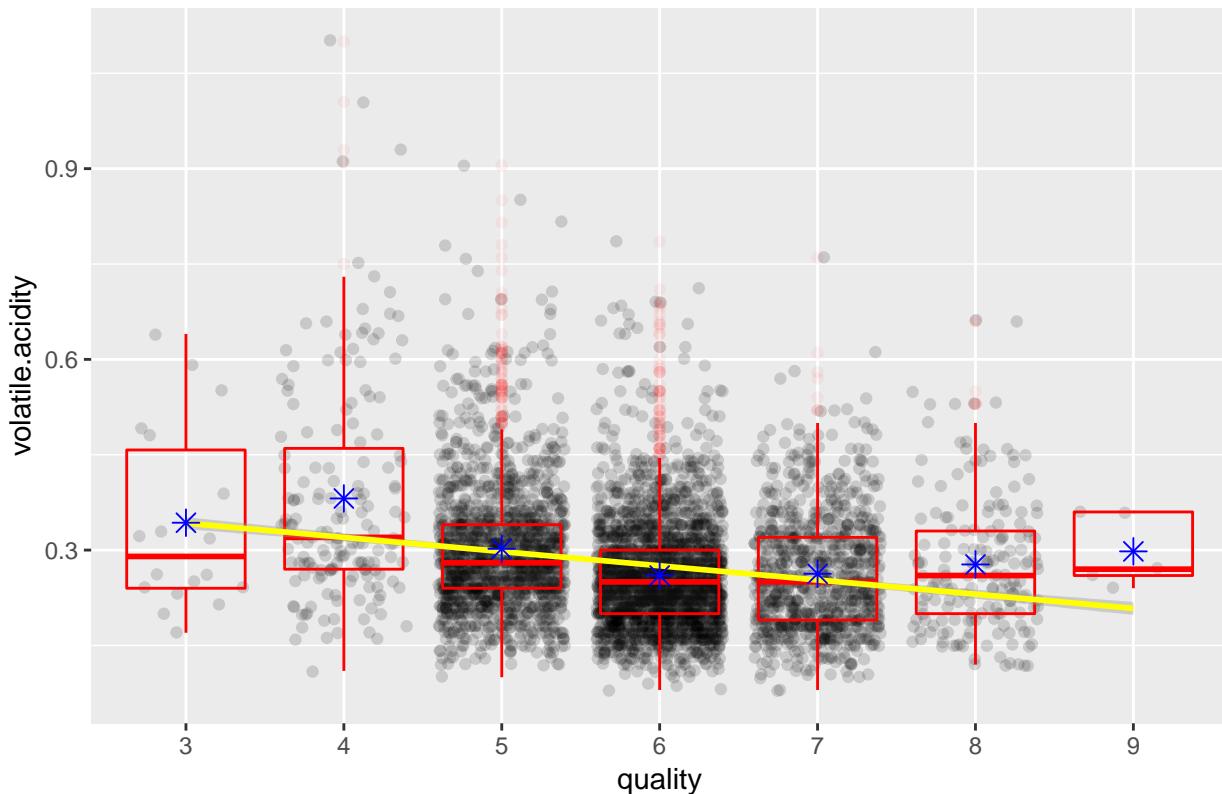
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$residual.sugar
## t = -7.0747, df = 4887, p-value = 1.71e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.12835655 -0.07286003
## sample estimates:
##       cor
## -0.1006866
```

Como podemos observar, la cantidad de **residual.sugar** tiene una correlación débil, es negativa, afectando minimamente a la calidad del vino, su correlacion es -0.1006866, por lo que no es determinante para la calidad del vino.

Veamos como influye la variable **volatile.acidity** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = volatile.acidity)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,volatile.acidity),method='lm',color='yellow') +
  xlab('quality') + ylab('volatile.acidity') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs volatile.acidity')
```

## Calidad del vino vs volatile.acidity



```
corr_quality_volatile_acidity<-cor.test(white_wine_data$quality,white_wine_data$volatile.acidity,method="pearson")
corr_quality_volatile_acidity
```

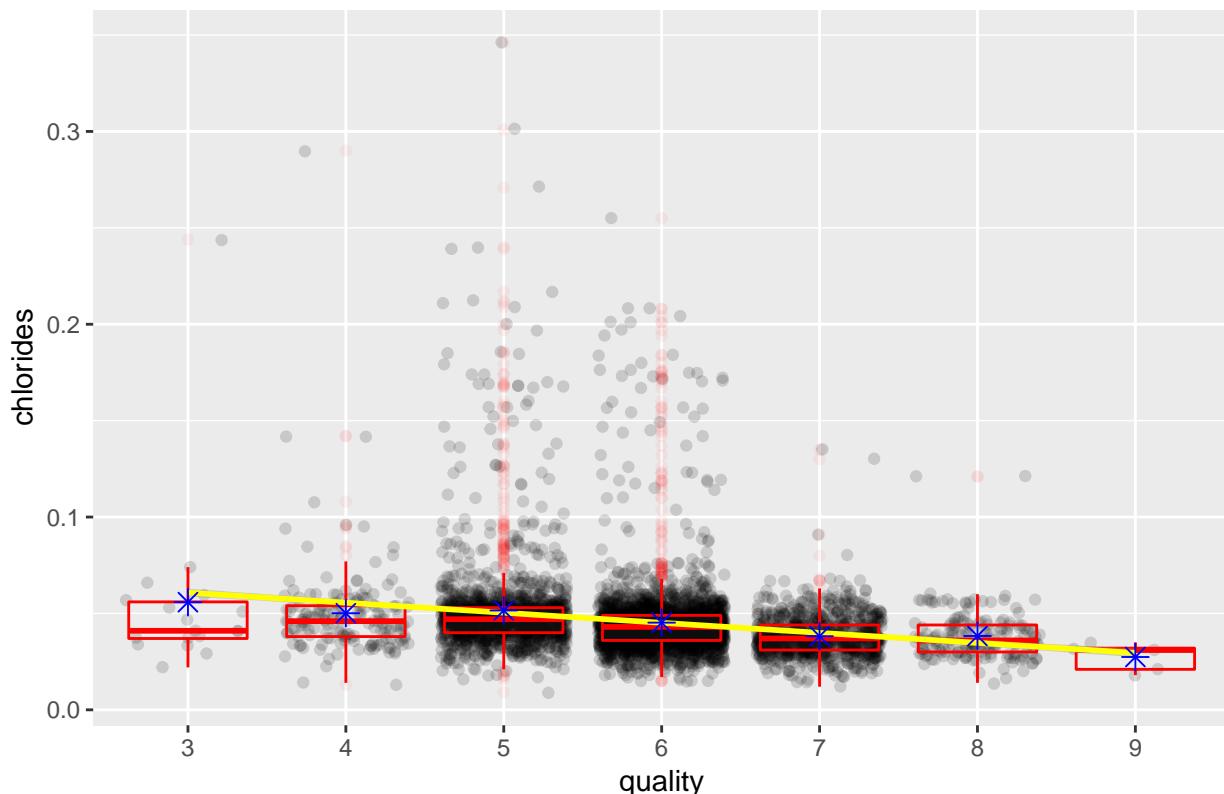
```
## 
## Pearson's product-moment correlation
## 
## data: white_wine_data$quality and white_wine_data$volatile.acidity
## t = -14.044, df = 4887, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.2237552 -0.1698640
## sample estimates:
##       cor
## -0.1969584
```

Como podemos observar, la cantidad de **volatile.acidity** o acidez volatil, no es un factor que tenga repercusión en la calidad del vino, dado que la calidad del vino no parece verse afectada por este factor, al permanecer los valores de acidez fija, cercanos a la media para cada tipo de calidad. Su correlación es débil -0.1969584.

Veamos como influye la variable **chlorides** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = chlorides)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,chlorides),method='lm',color='yellow')+
  xlab('quality')+ ylab('chlorides') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs chlorides')
```

## Calidad del vino vs chlorides



```
corr_quality_chlorides<-cor.test(white_wine_data$quality,white_wine_data$chlorides,method='pearson')
corr_quality_chlorides
```

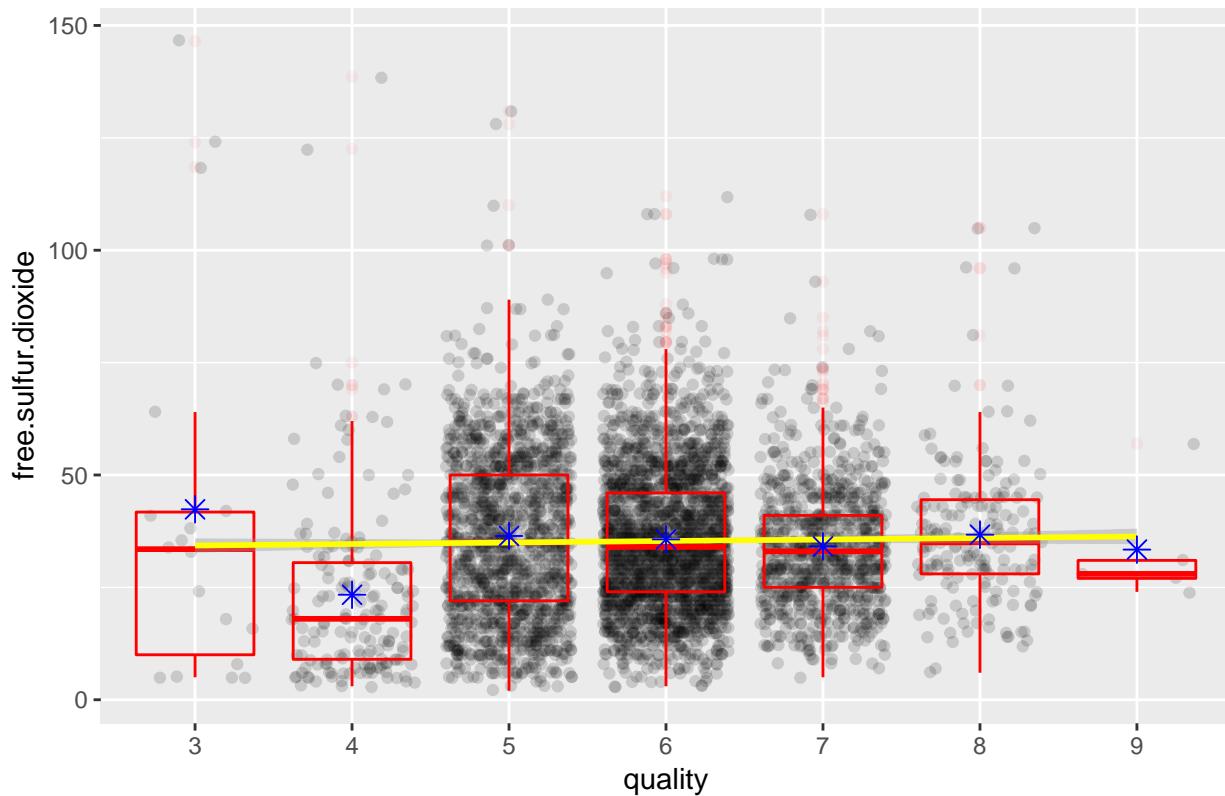
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$chlorides
## t = -15.074, df = 4887, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.2374128 -0.1838374
## sample estimates:
## cor
## -0.2107834
```

La cantidad de sal en vino **chlorides**, tambien tiene una correlación débil, siendo su valor -0.2107834 por lo que no se ve afectada por esta variable.

Veamos como influye la variable **free.sulfur.dioxide** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = free.sulfur.dioxide)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,free.sulfur.dioxide),method='lm',color='yellow')+
  xlab('quality')+ ylab('free.sulfur.dioxide') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs free.sulfur.dioxide')
```

## Calidad del vino vs free.sulfur.dioxide



```
corr_quality_free_sulfur<-cor.test(white_wine_data$quality,white_wine_data$free.sulfur.dioxide,method='pearson')
corr_quality_free_sulfur
```

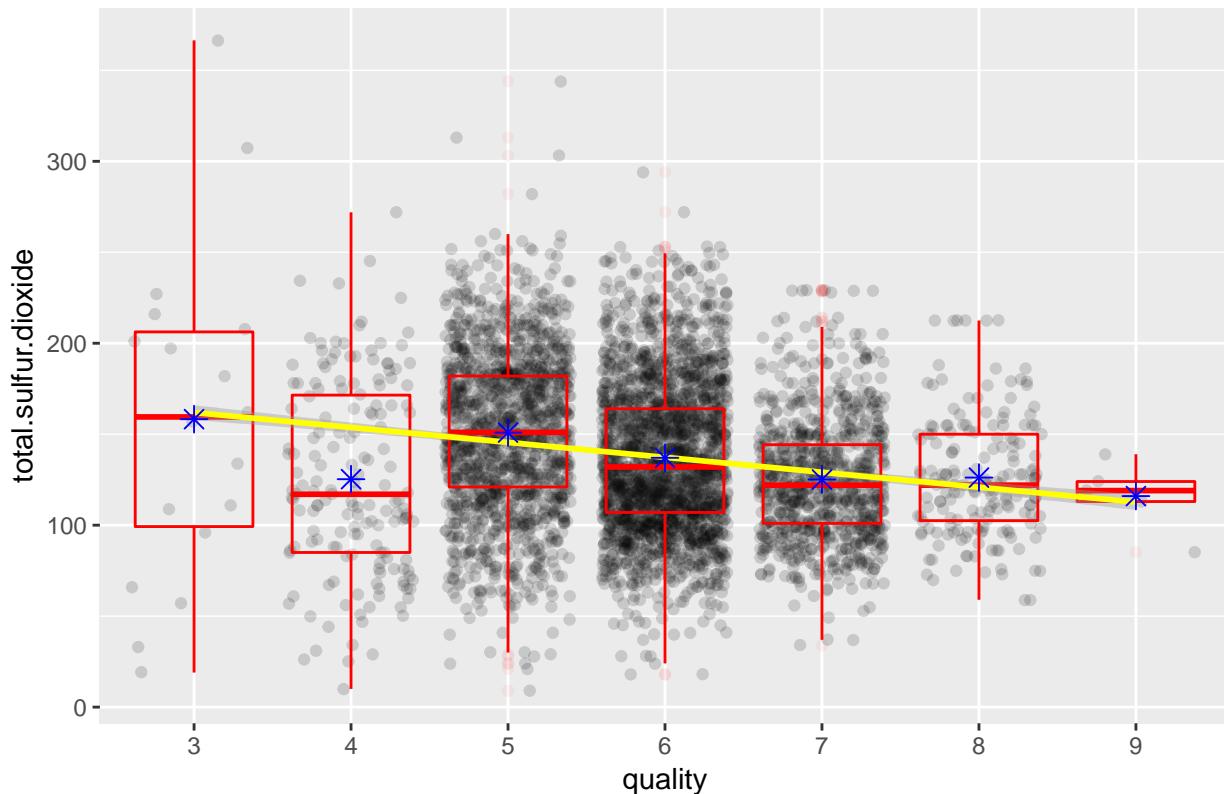
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$free.sulfur.dioxide
## t = 1.2447, df = 4887, p-value = 0.2133
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.01023564 0.04581105
## sample estimates:
##       cor
## 0.01780169
```

La cantidad libre de dioxido de azufre en vino **free.sulfur.dioxide**, tambien tiene una correlación débil positiva, siendo su valor 0.0178017 por lo que no se ve afectada por esta variable.

Veamos como influye la variable **total.sulfur.dioxide** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = total.sulfur.dioxide)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,total.sulfur.dioxide),method='lm',color='yellow')+
  xlab('quality')+ ylab('total.sulfur.dioxide') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs total.sulfur.dioxide')
```

## Calidad del vino vs total.sulfur.dioxide



```
corr_quality_total_sulfur<-cor.test(white_wine_data$quality,white_wine_data$total.sulfur.dioxide,method="pearson")
corr_quality_total_sulfur
```

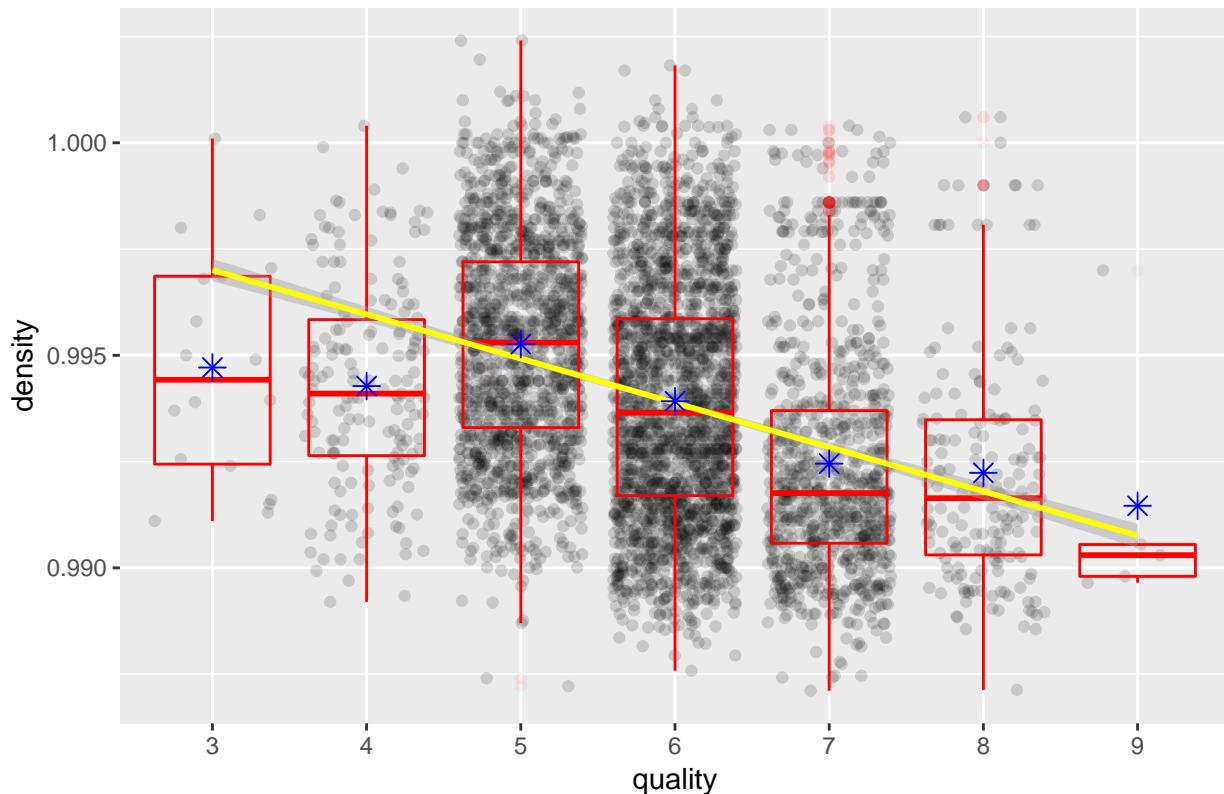
```
## 
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$total.sulfur.dioxide
## t = -12.177, df = 4887, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.1986843 -0.1442696
## sample estimates:
##       cor
## -0.1716078
```

La cantidad total de dioxido de azufre en vino **total.sulfur.dioxide**, tambien tiene una correlación débil negativa, siendo su valor -0.1716078 por lo que no se ve afectada por esta variable.

Veamos como influye la variable **density** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = density)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,density),method='lm',color='yellow')+
  xlab('quality')+ ylab('density') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs density')
```

## Calidad del vino vs density



```
corr_quality_density<-cor.test(white_wine_data$quality,white_wine_data$density,method='pearson')
corr_quality_density
```

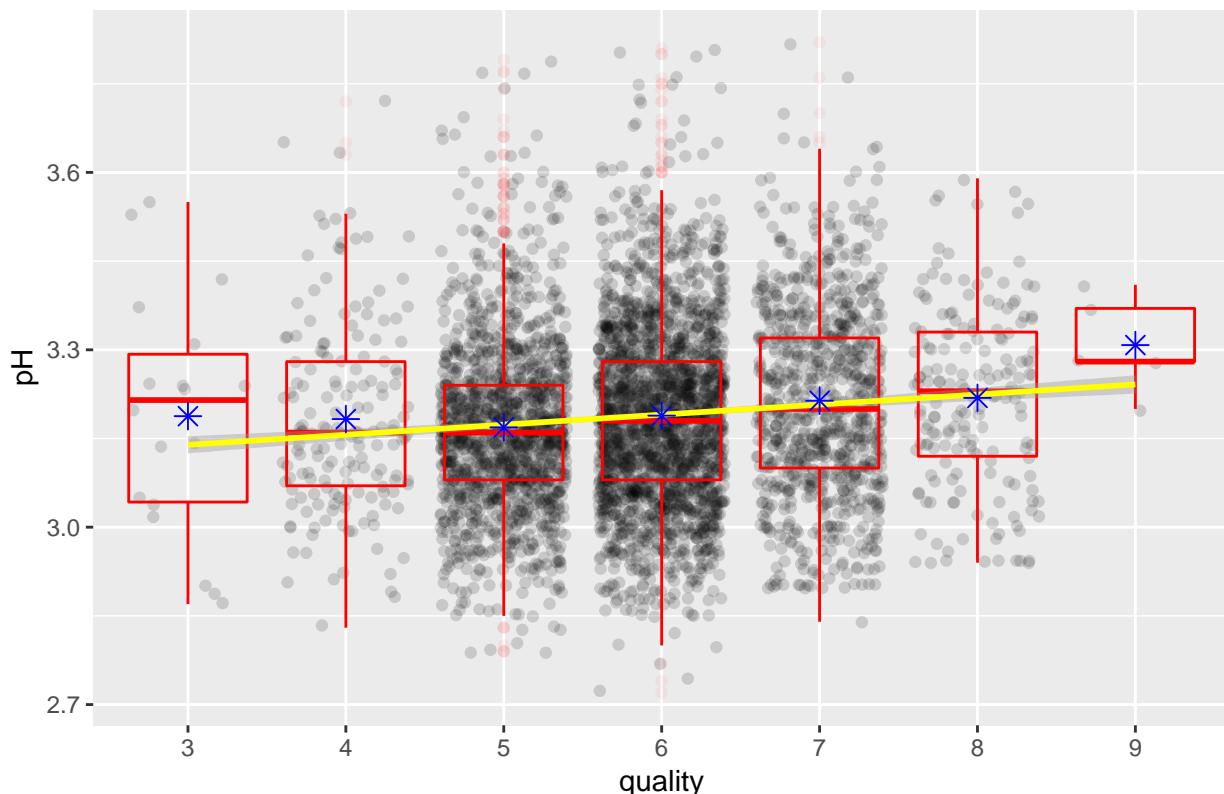
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$density
## t = -23.425, df = 4887, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## -0.3427085 -0.2922999
## sample estimates:
##       cor
## -0.3177287
```

La densidad en vino **density**, tiene una correlación moderada negativa, siendo su valor -0.3177287 por lo que SI se ve afectada por esta variable, aunque de manera moderada, cuanto menor densidad, mejor vino.

Veamos como influye la variable **pH** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = pH)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,pH),method='lm',color='yellow')+
  xlab('quality')+ ylab('pH') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs pH')
```

## Calidad del vino vs pH



```
corr_quality_pH<-cor.test(white_wine_data$quality,white_wine_data$pH,method='pearson')
corr_quality_pH
```

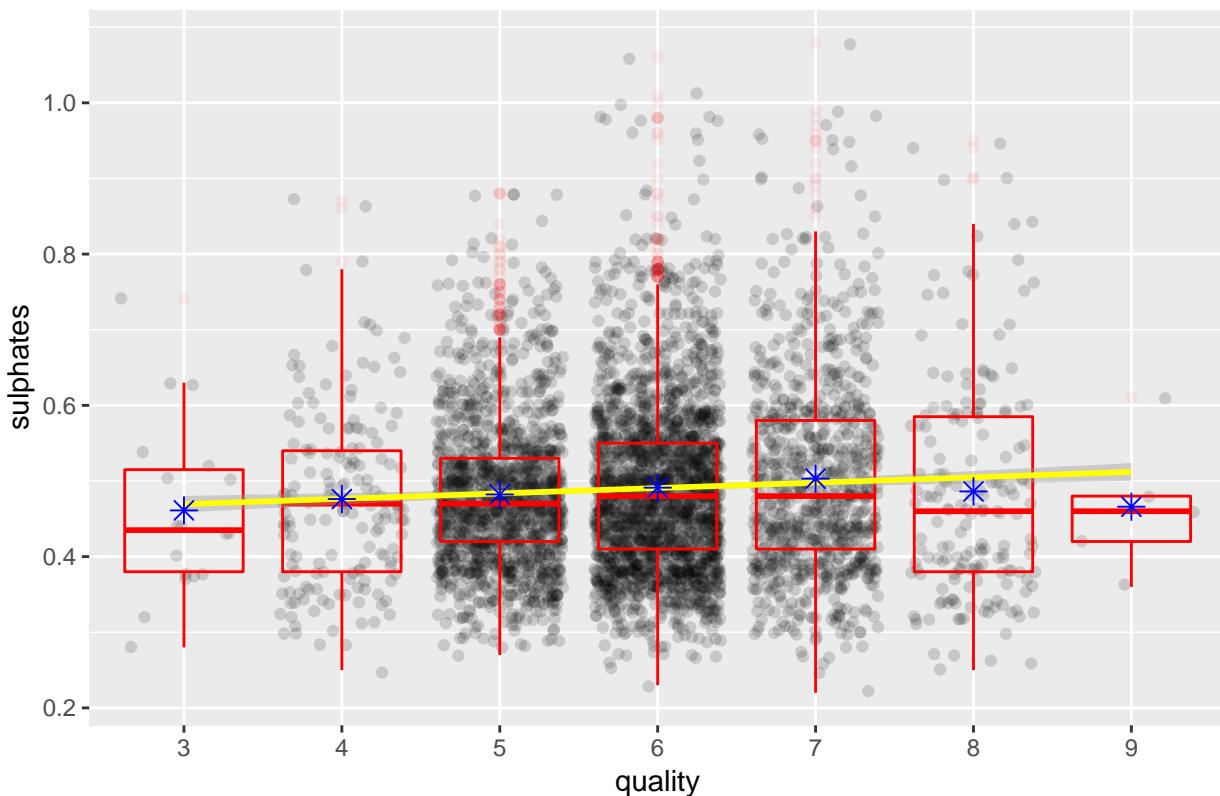
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$pH
## t = 7.0066, df = 4887, p-value = 2.773e-12
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.07189715 0.12740443
## sample estimates:
## cor
## 0.09972838
```

La cantidad de ph en vino **pH**, tiene una correlación débil negativa, siendo su valor 0.0997284 por lo que no se ve afectada por esta variable, muy débilmente cuanto mayor ph, mejor vino, aunque no es relevante. Sus valores oscilan entre 3 y 3.4 por lo que hay muy poco margen de diferencia entre el ph de los vinos.

Veamos como influye la variable **sulphates** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = sulphates)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,sulphates),method='lm',color='yellow')+
  xlab('quality')+ ylab('sulphates') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs sulphates')
```

## Calidad del vino vs sulphates



```
corr_quality_sulphates<-cor.test(white_wine_data$quality,white_wine_data$sulphates,method='pearson')
corr_quality_sulphates
```

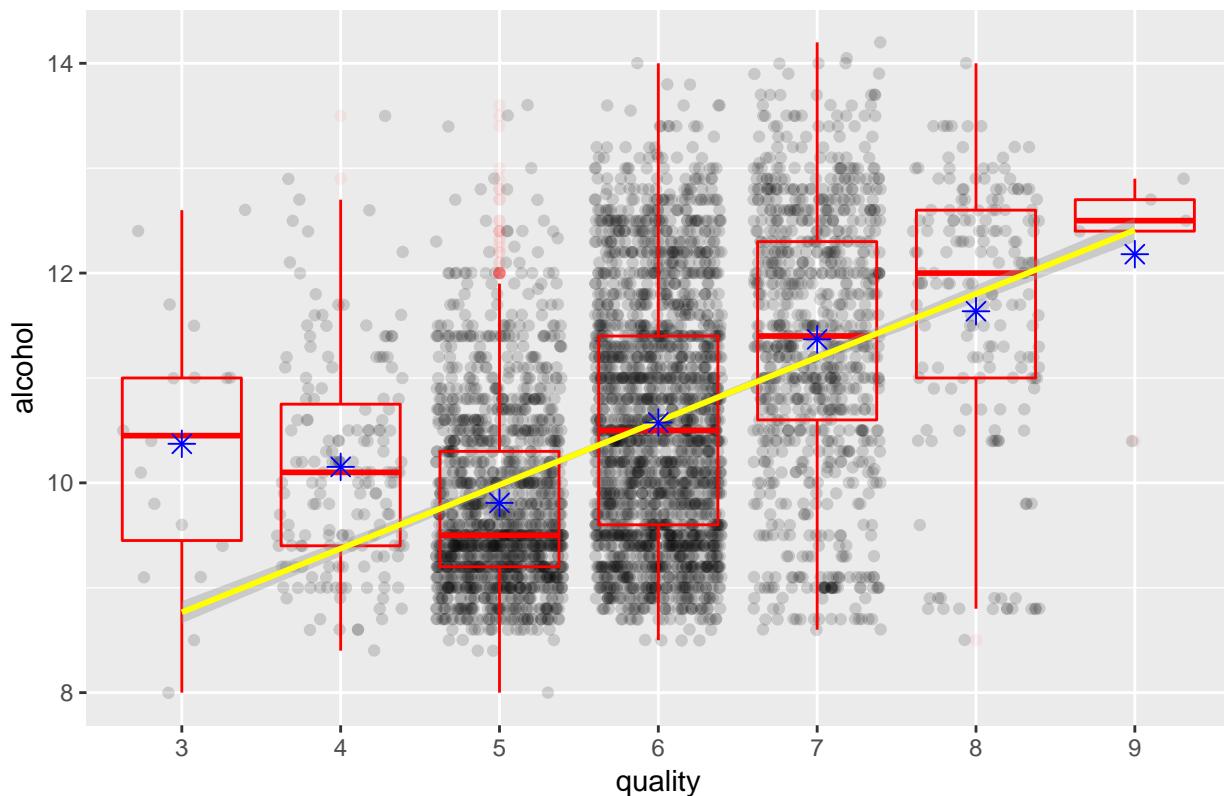
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$sulphates
## t = 3.8562, df = 4887, p-value = 0.0001166
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
## 0.02708816 0.08298266
## sample estimates:
## cor
## 0.05507856
```

La cantidad de sulfatos en vino **sulphates**, tiene una correlación muy débil negativa, siendo su valor 0.0550786 por lo que no se ve afectada por esta variable.

Veamos como influye la variable **sulphates** en la calidad del vino:

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = alcohol)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,alcohol),method='lm',color='yellow')+
  xlab('quality')+ ylab('alcohol') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs alcohol')
```

## Calidad del vino vs alcohol



```
corr_quality_alcohol<-cor.test(white_wine_data$quality,white_wine_data$alcohol,method='pearson')
corr_quality_alcohol
```

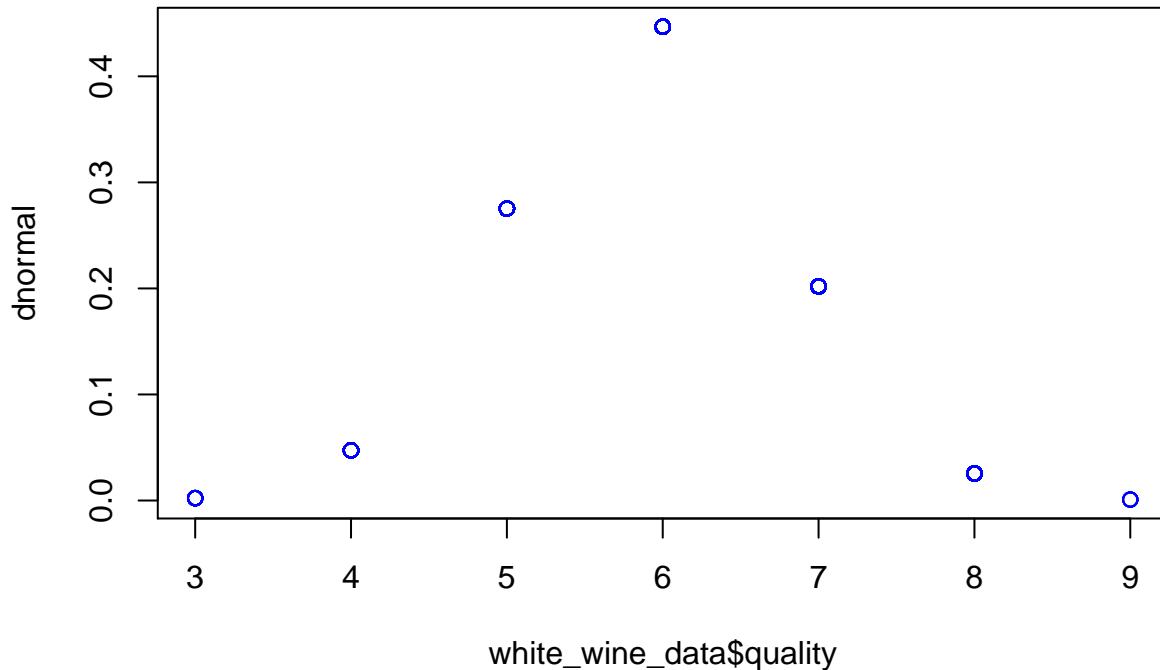
```
##
## Pearson's product-moment correlation
##
## data: white_wine_data$quality and white_wine_data$alcohol
## t = 33.906, df = 4887, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.4134241 0.4588183
## sample estimates:
##        cor
## 0.4363989
```

La cantidad de sulfatos en vino **alcohol**, tiene una correlación moderada positiva, siendo su valor 0.4363989 por lo que SI se ve afectada por esta variable. A mayor cantidad de alcohol, mejora la calidad del vino, encontrando valores entre 9 y 13. Siendo para vinos muy bueno, valores superiores a 11 de cantidad de alcohol.

### 4.2. Comprobación de la normalidad y homogeneidad de la varianza.

Verificamos la normalidad sobre la calidad de los vinos disponibles.

```
dnormal<-dnorm(white_wine_data$quality,mean(white_wine_data$quality),sd(white_wine_data$quality))
plot(white_wine_data$quality,dnormal,type='p', col='blue',lwd=1)
```

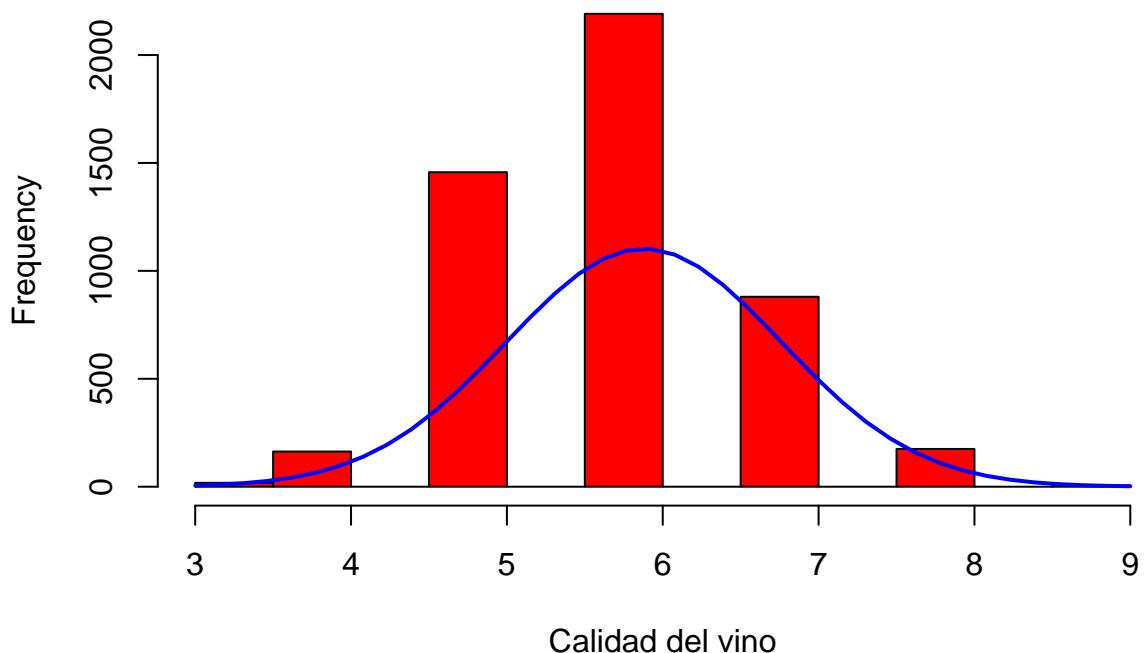


`white_wine_data$quality`

Obtenemos la curva normal sobre el gráfico, para ver la distribución normal.

```
x <- white_wine_data$quality
h<-hist(x, col="red", xlab="Calidad del vino",main="Histograma con Curva Normal")
xfit<-seq(min(x),max(x),length=40)
yfit<-dnorm(xfit,mean=mean(x),sd=sd(x))
yfit <- yfit*diff(h$mid[1:2])*length(x)
lines(xfit, yfit, col="blue", lwd=2)
```

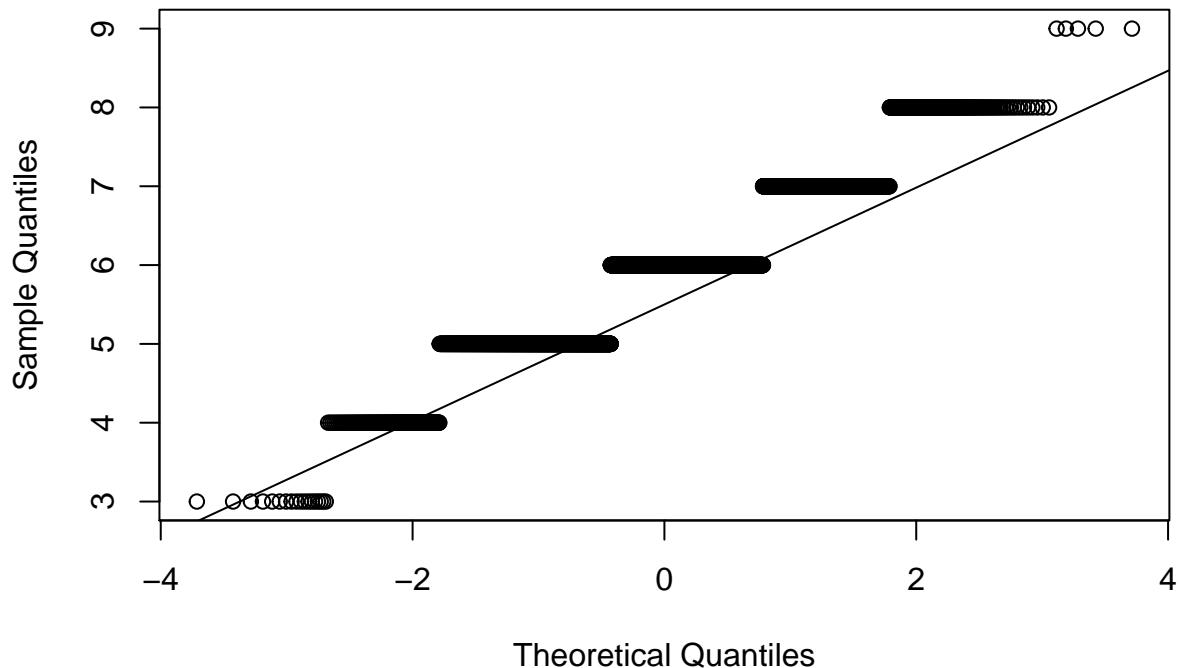
### Histograma con Curva Normal



Obtenemos en gráfico de nube de puntos y recta, para ver como se distribuyen

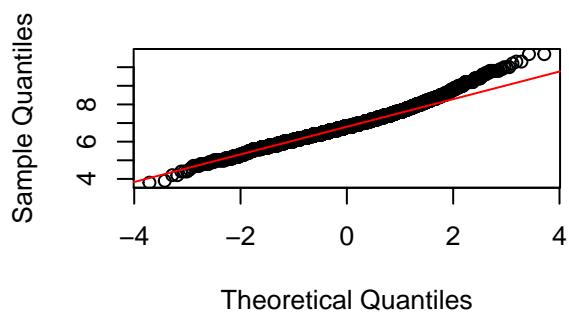
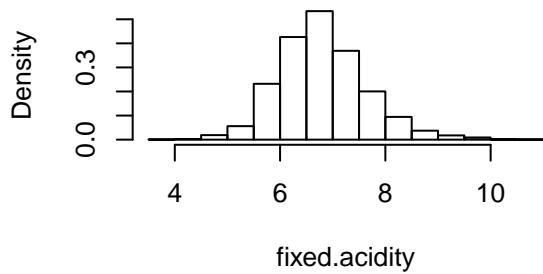
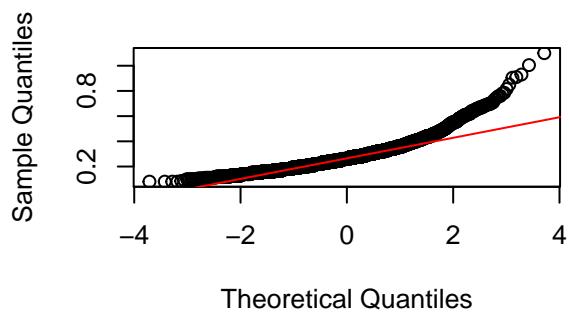
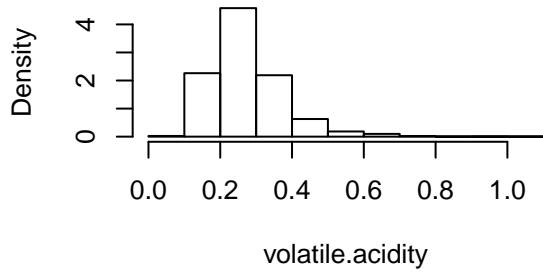
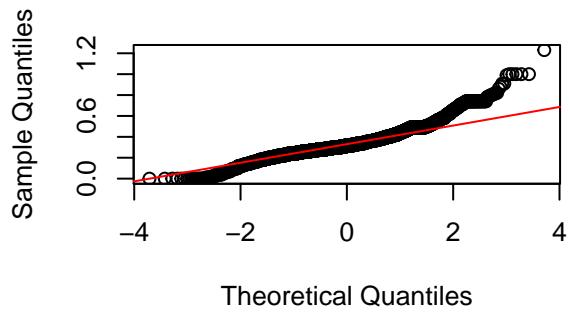
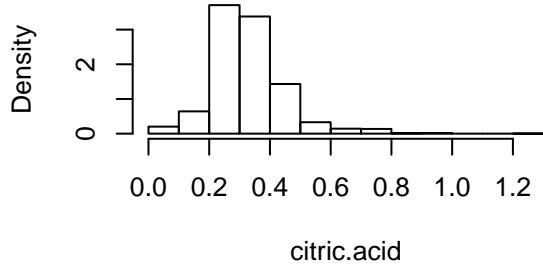
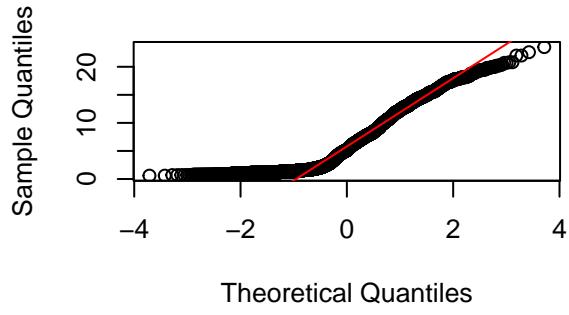
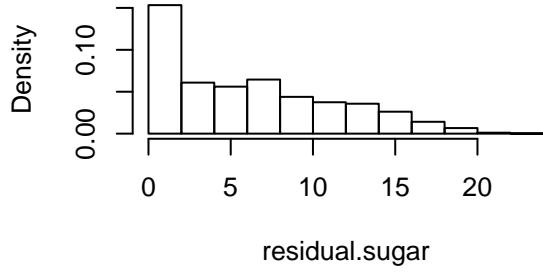
```
qqnorm( white_wine_data$quality ) # la nube de puntos  
qqline( white_wine_data$quality ) # la recta
```

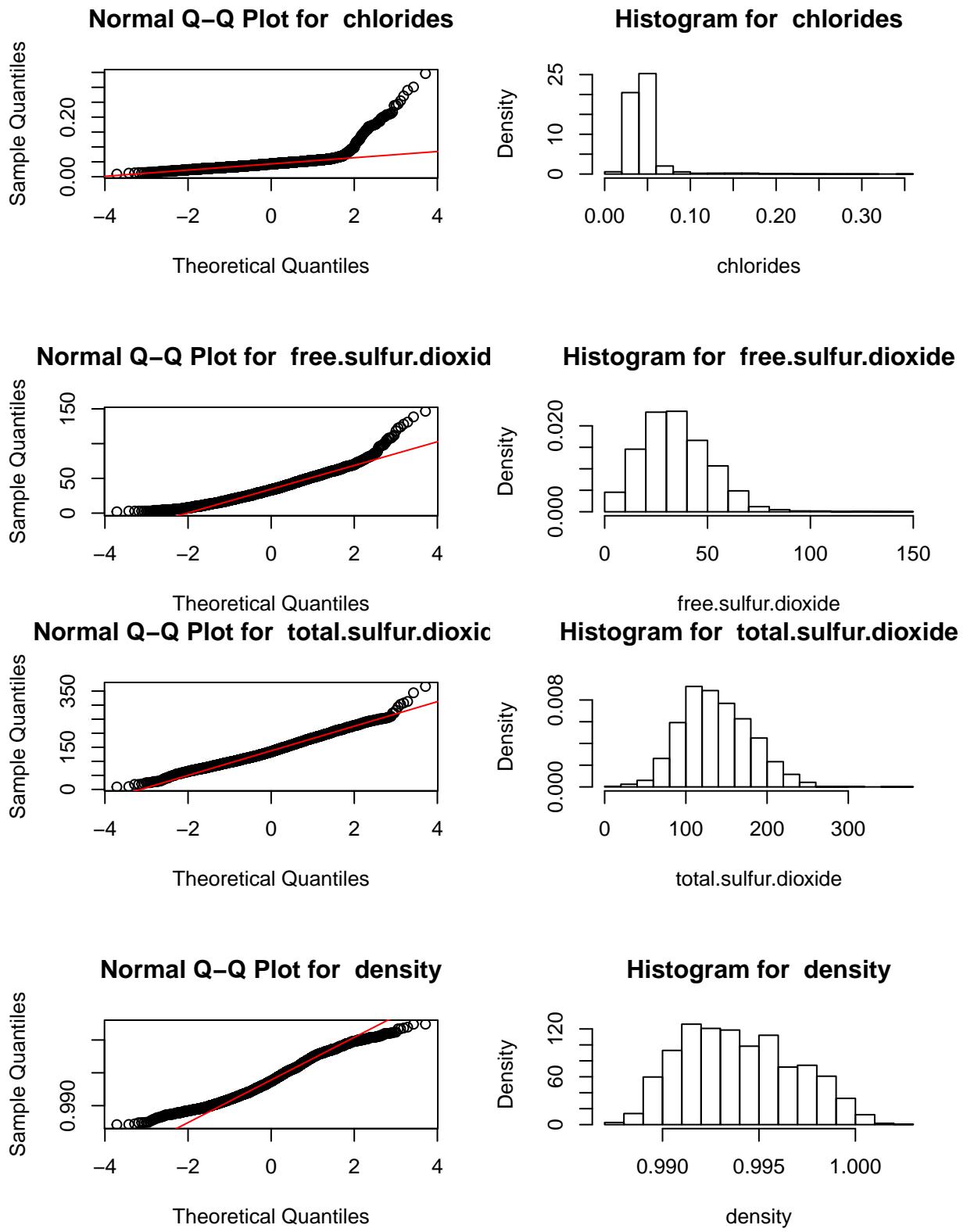
### Normal Q-Q Plot

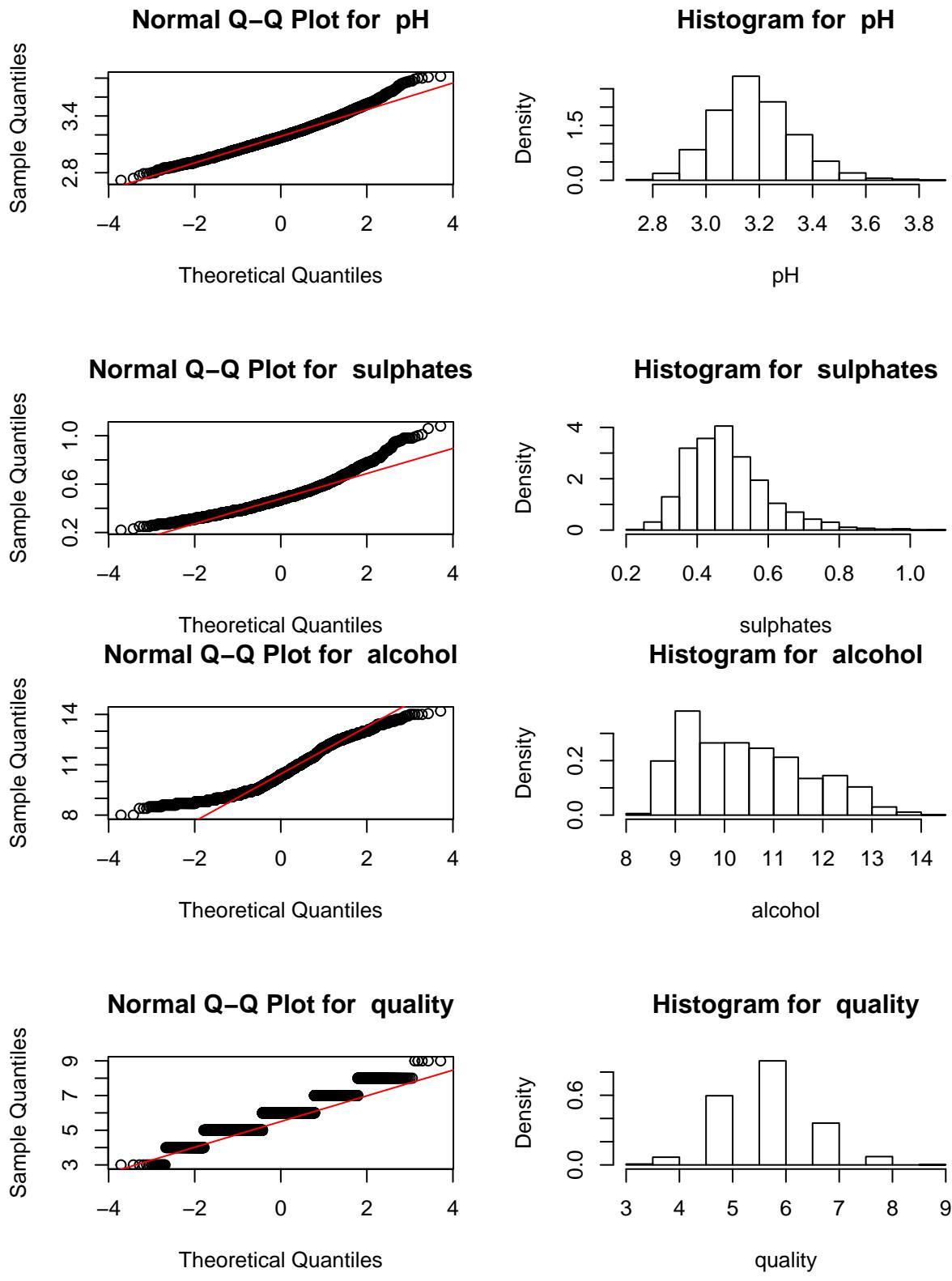


La nube de puntos se aproxima a la recta, por tanto se trata de una distribución normal sobre la calidad.

```
par(mfrow=c(2,2))  
for(i in 1:ncol(white_wine_data)) {  
  if (is.numeric(white_wine_data[,i])){  
    qqnorm(white_wine_data[,i],main = paste("Normal Q-Q Plot for ", colnames(white_wine_data)[i]))  
    qqline(white_wine_data[,i],col="red")  
    hist(white_wine_data[,i],  
         main=paste("Histogram for ", colnames(white_wine_data)[i]),  
         xlab=colnames(white_wine_data)[i], freq = FALSE)  
  }  
}
```

**Normal Q-Q Plot for fixed.acidity****Histogram for fixed.acidity****Normal Q-Q Plot for volatile.acidity****Histogram for volatile.acidity****Normal Q-Q Plot for citric.acid****Histogram for citric.acid****Normal Q-Q Plot for residual.sugar****Histogram for residual.sugar**





A simple vista, parece que las variables, siguen una distribución normal, no obstante para ver si las variables están normalizadas, utilizaremos el Test Shapiro y validaremos si su valor p-valor es mayor que 0.05, de serlo, se aceptaría la hipótesis nula y las variables serían normalizadas.

Asumiendo como hipótesis nula que la población está distribuida normalmente, si el p-valor es menor al nivel

de significancia, generalmente alfa=0.05, entonces la hipótesis nula es rechazada y se concluye que los datos no cuentan con una distribución normal:

Validamos fixed.acidity:

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

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$fixed.acidity  
## W = 0.98331, p-value < 2.2e-16
```

Validamos volatile.acidity:

```
shapiro.test(white_wine_data$volatile.acidity)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$volatile.acidity  
## W = 0.90724, p-value < 2.2e-16
```

Validamos citric.acid:

```
shapiro.test(white_wine_data$citric.acid)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$citric.acid  
## W = 0.9329, p-value < 2.2e-16
```

Validamos residual.sugar:

```
shapiro.test(white_wine_data$residual.sugar)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$residual.sugar  
## W = 0.89614, p-value < 2.2e-16
```

Validamos chlorides:

```
shapiro.test(white_wine_data$chlorides)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$chlorides  
## W = 0.59037, p-value < 2.2e-16
```

Validamos free.sulfur.dioxide:

```
shapiro.test(white_wine_data$free.sulfur.dioxide)
```

```
##  
## Shapiro-Wilk normality test  
##  
## data: white_wine_data$free.sulfur.dioxide
```

```

## W = 0.96739, p-value < 2.2e-16
Validamos total.sulfur.dioxide:
shapiro.test(white_wine_data$total.sulfur.dioxide)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$total.sulfur.dioxide
## W = 0.99162, p-value < 2.2e-16

Validamos density:
shapiro.test(white_wine_data$density)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$density
## W = 0.98102, p-value < 2.2e-16

Validamos pH:
shapiro.test(white_wine_data$pH)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$pH
## W = 0.98808, p-value < 2.2e-16

Validamos sulphates:
shapiro.test(white_wine_data$sulphates)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$sulphates
## W = 0.95151, p-value < 2.2e-16

Validamos alcohol:
shapiro.test(white_wine_data$alcohol)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$alcohol
## W = 0.9552, p-value < 2.2e-16

Validamos quality:
shapiro.test(white_wine_data$quality)

##
## Shapiro-Wilk normality test
##
## data: white_wine_data$quality
## W = 0.88883, p-value < 2.2e-16

```

Al aplicar el test de Shapiro, ninguna de las variables nos retorna un valor superior al p-valor 0.05, por lo que se puede inferir que no son variables normalizadas. Aunque al tener mas de 30 registros, mediante el teorema de límite central se pueden aproximar a la normal.

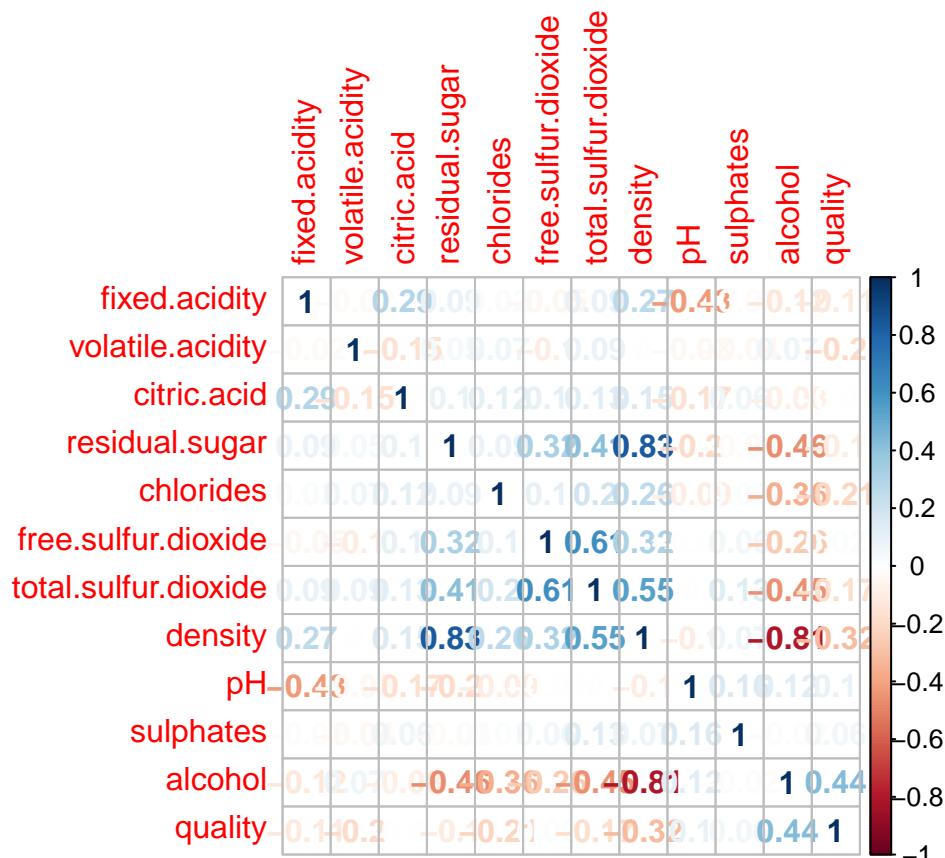
Siendo el **teorema central del límite**: se aplica a la distribución de la media de la muestra de un conjunto de datos. La media de una muestra de cualquier conjunto de datos es cada vez más normal a medida que aumenta la cantidad de observaciones. Así, a medida que aumenta el tamaño de la muestra N, la distribución de la media de la muestra se parece cada vez más a una distribución normal con una (verdadera) media de la población mu y varianza alpha2/N

**4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos.** En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

#### 4.3.a) Correlación de las variables

Visualizamos la correlación de las variables, para ello tomaremos las variables de tipo numérico, que son las que puede utilizar la función **cor** de R.

```
library(corrplot)
## corrplot 0.84 loaded
MatrizCorrelacion <- cor(white_wine_data[, 0:12])
corrplot(MatrizCorrelacion, method = "number")
```



De esta tabla, podemos inferir las siguientes relaciones entre variables:

- Podemos observar como la calidad está algo relacionada con el alcohol con un 44% de correlación. Lo cual es interesante.

- Además de estar relacionado el alcohol con la calidad, se ve también relacionado con la densidad. 78%
- Siendo la densidad fuertemente relacionada con la cantidad de azúcar residual. 84%
- La cantidad libre de dióxido de azufre está fuertemente relacionada con la cantidad total de dióxido de azufre. 62%

```
print(MatrizCorrelacion)
```

```
## fixed.acidity volatile.acidity citric.acid
## fixed.acidity 1.00000000 -0.024107494 0.291831084
## volatile.acidity -0.02410749 1.000000000 -0.153203218
## citric.acid 0.29183108 -0.153203218 1.000000000
## residual.sugar 0.08764786 0.045427684 0.095972675
## chlorides 0.02441937 0.069297815 0.117915472
## free.sulfur.dioxide -0.04625756 -0.096808505 0.100776502
## total.sulfur.dioxide 0.09316754 0.089749156 0.125578225
## density 0.26991041 0.003125434 0.152883703
## pH -0.42882030 -0.033379113 -0.167537193
## sulphates -0.01855138 -0.037587657 0.060958973
## alcohol -0.12217456 0.067200556 -0.080663482
## quality -0.11230032 -0.196958375 -0.009924564
## residual.sugar chlorides free.sulfur.dioxide
## fixed.acidity 0.08764786 0.02441937 -0.046257558
## volatile.acidity 0.04542768 0.06929782 -0.096808505
## citric.acid 0.09597267 0.11791547 0.100776502
## residual.sugar 1.00000000 0.08786592 0.320651334
## chlorides 0.08786592 1.00000000 0.103835022
## free.sulfur.dioxide 0.32065133 0.10383502 1.000000000
## total.sulfur.dioxide 0.41161478 0.19950681 0.611481854
## density 0.83137099 0.26179464 0.318501601
## pH -0.19960259 -0.09124336 -0.005991704
## sulphates -0.02833527 0.01647853 0.057191781
## alcohol -0.46055564 -0.36039907 -0.256070075
## quality -0.10068661 -0.21078340 0.017801690
## total.sulfur.dioxide density pH
## fixed.acidity 0.0931675392 0.269910409 -0.4288203006
## volatile.acidity 0.0897491562 0.003125434 -0.0333791130
## citric.acid 0.1255782254 0.152883703 -0.1675371934
## residual.sugar 0.4116147820 0.831370986 -0.1996025916
## chlorides 0.1995068139 0.261794645 -0.0912433614
## free.sulfur.dioxide 0.6114818545 0.318501601 -0.0059917036
## total.sulfur.dioxide 1.00000000000 0.547723613 -0.0003248467
## density 0.5477236129 1.0000000000 -0.0985260011
## pH -0.0003248467 -0.098526001 1.00000000000
## sulphates 0.1336286360 0.074453487 0.1551462925
## alcohol -0.4511763161 -0.805870749 0.1209201252
## quality -0.1716078413 -0.317728677 0.0997283753
## sulphates alcohol quality
## fixed.acidity -0.01855138 -0.12217456 -0.112300317
## volatile.acidity -0.03758766 0.06720056 -0.196958375
## citric.acid 0.06095897 -0.08066348 -0.009924564
## residual.sugar -0.02833527 -0.46055564 -0.100686608
## chlorides 0.01647853 -0.36039907 -0.210783403
## free.sulfur.dioxide 0.05719178 -0.25607008 0.017801690
## total.sulfur.dioxide 0.13362864 -0.45117632 -0.171607841
## density 0.07445349 -0.80587075 -0.317728677
```

```

## pH          0.15514629  0.12092013  0.099728375
## sulphates 1.00000000 -0.01845305  0.055078559
## alcohol    -0.01845305  1.00000000  0.436398860
## quality    0.05507856  0.43639886  1.000000000

```

#### 4.3.b) Modelo de Regresión Lineal

Para poder encontrar variables significativas, mejoraremos el modelo eliminando las variables que no tengan un fuerte significado para el análisis, verificandolo mediante valor de R-Cuadrado.

```

# Splitting the data into Training and Testing sets
library(caTools)
set.seed(144)
spl = sample.split(white_wine_data$quality, 0.7)
train = subset(white_wine_data, spl == TRUE)
test=subset(white_wine_data,spl==FALSE)

model1 <- lm(quality ~ .-category, data = train)
summary(model1)

##
## Call:
## lm(formula = quality ~ . - category, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -3.3774 -0.5105 -0.0325  0.4653  2.7740 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)              2.361e+02  2.870e+01   8.227 2.70e-16 ***
## fixed.acidity            1.361e-01  2.900e-02   4.694 2.78e-06 ***
## volatile.acidity         -1.755e+00  1.346e-01 -13.038 < 2e-16 ***
## citric.acid              1.190e-01  1.186e-01   1.004  0.3156    
## residual.sugar           1.089e-01  1.070e-02   10.181 < 2e-16 ***
## chlorides                 -7.280e-02 6.676e-01  -0.109  0.9132    
## free.sulfur.dioxide      5.264e-03  1.054e-03   4.994 6.22e-07 ***
## total.sulfur.dioxide     6.861e-05  4.616e-04   0.149  0.8818    
## density                  -2.373e+02  2.909e+01  -8.158 4.73e-16 ***
## pH                        9.674e-01  1.391e-01   6.957 4.15e-12 ***
## sulphates                6.495e-01  1.231e-01   5.277 1.39e-07 ***
## alcohol                   8.446e-02  3.616e-02   2.335  0.0196 *  
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7517 on 3411 degrees of freedom
## Multiple R-squared:  0.2815, Adjusted R-squared:  0.2791 
## F-statistic: 121.5 on 11 and 3411 DF,  p-value: < 2.2e-16

Obtenemos un modelo completo con todas las variables:

m1<-lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acid +
       white_wine_data$citric.acid + white_wine_data$residual.sugar + white_wine_data$chlorides +
       white_wine_data$free.sulfur.dioxide + white_wine_data$total.sulfur.dioxide +
       white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol)
summary(m1)

```

```

## 
## Call:
## lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity +
##     white_wine_data$volatile.acidity + white_wine_data$citric.acid +
##     white_wine_data$residual.sugar + white_wine_data$chlorides +
##     white_wine_data$free.sulfur.dioxide + white_wine_data$total.sulfur.dioxide +
##     white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##     white_wine_data$alcohol)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -3.6054 -0.4996 -0.0432  0.4655  3.1020 
##
## Coefficients:
##                               Estimate Std. Error t value
## (Intercept)                2.296e+02  2.359e+01  9.734
## white_wine_data$fixed.acidity   1.312e-01  2.367e-02  5.542
## white_wine_data$volatile.acidity -1.858e+00  1.132e-01 -16.411
## white_wine_data$citric.acid    2.657e-02  9.659e-02  0.275
## white_wine_data$residual.sugar  1.063e-01  8.811e-03 12.060
## white_wine_data$chlorides     -5.301e-02  5.450e-01 -0.097
## white_wine_data$free.sulfur.dioxide 4.292e-03  8.575e-04  5.005
## white_wine_data$total.sulfur.dioxide 7.171e-05  3.828e-04  0.187
## white_wine_data$density        -2.308e+02  2.391e+01 -9.651
## white_wine_data$pH             9.380e-01  1.139e-01  8.237
## white_wine_data$sulphates     7.364e-01  1.013e-01  7.266
## white_wine_data$alcohol       9.727e-02  2.978e-02  3.266
##                               Pr(>|t|)    
## (Intercept) < 2e-16 ***
## white_wine_data$fixed.acidity 3.14e-08 ***
## white_wine_data$volatile.acidity < 2e-16 ***
## white_wine_data$citric.acid    0.7833
## white_wine_data$residual.sugar < 2e-16 ***
## white_wine_data$chlorides      0.9225
## white_wine_data$free.sulfur.dioxide 5.79e-07 ***
## white_wine_data$total.sulfur.dioxide 0.8514
## white_wine_data$density        < 2e-16 ***
## white_wine_data$pH             2.25e-16 ***
## white_wine_data$sulphates     4.27e-13 ***
## white_wine_data$alcohol        0.0011 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7466 on 4877 degrees of freedom
## Multiple R-squared:  0.2892, Adjusted R-squared:  0.2876
## F-statistic: 180.4 on 11 and 4877 DF,  p-value: < 2.2e-16

```

Esto nos da un R-cuadrado ajustado de 0.7466, de modo que ahora vamos a ir eliminando variables para ver con que modelo nos quedamos.

Quitemos la cantidad de sal **chlorides**, que es el que mayor p-value **0.9225** tenia en el modelo anterior, a ver como afecta:

```
m2<-lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acid +
+ white_wine_data$residual.sugar + white_wine_data$citric.acid +
```

```

white_wine_data$free.sulfur.dioxide + white_wine_data$total.sulfur.dioxide +
white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol
summary(m2)

##
## Call:
## lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity +
##     white_wine_data$volatile.acidity + white_wine_data$residual.sugar +
##     white_wine_data$citric.acid + white_wine_data$free.sulfur.dioxide +
##     white_wine_data$total.sulfur.dioxide + white_wine_data$density +
##     white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -3.6056 -0.4998 -0.0431  0.4651  3.1022 
##
## Coefficients:
##                               Estimate Std. Error t value
## (Intercept)                2.300e+02  2.323e+01  9.902
## white_wine_data$fixed.acidity    1.316e-01  2.325e-02  5.661
## white_wine_data$volatile.acidity -1.859e+00  1.126e-01 -16.505
## white_wine_data$residual.sugar   1.064e-01  8.617e-03 12.352
## white_wine_data$citric.acid      2.555e-02  9.602e-02  0.266
## white_wine_data$free.sulfur.dioxide 4.289e-03  8.569e-04  5.005
## white_wine_data$total.sulfur.dioxide 7.193e-05  3.828e-04  0.188
## white_wine_data$density          -2.312e+02  2.353e+01 -9.825
## white_wine_data$pH               9.400e-01  1.120e-01  8.394
## white_wine_data$sulphates        7.369e-01  1.012e-01  7.281
## white_wine_data$alcohol          9.716e-02  2.976e-02  3.265
##
## Pr(>|t|) 
## (Intercept) < 2e-16 ***
## white_wine_data$fixed.acidity 1.59e-08 ***
## white_wine_data$volatile.acidity < 2e-16 ***
## white_wine_data$residual.sugar < 2e-16 ***
## white_wine_data$citric.acid    0.7901
## white_wine_data$free.sulfur.dioxide 5.78e-07 ***
## white_wine_data$total.sulfur.dioxide 0.8510
## white_wine_data$density         < 2e-16 ***
## white_wine_data$pH              < 2e-16 ***
## white_wine_data$sulphates       3.83e-13 ***
## white_wine_data$alcohol         0.0011 **
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7465 on 4878 degrees of freedom
## Multiple R-squared:  0.2892, Adjusted R-squared:  0.2877
## F-statistic: 198.4 on 10 and 4878 DF,  p-value: < 2.2e-16

```

Observamos que no se ve afectado el R-cuadrado ajustado, su valor 0.7465, por lo que podemos eliminar la variable.

Quitemos la cantidad total de dioxido de azufre **total.sulfur.dioxide**, que tenia el mayor p-value **0.8510** del modelo anterior a ver como afecta:

```

m3<-lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
       + white_wine_data$residual.sugar + white_wine_data$citric.acid +
       white_wine_data$free.sulfur.dioxide +
       white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol)
summary(m3)

##
## Call:
## lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity +
##      white_wine_data$volatile.acidity + +white_wine_data$residual.sugar +
##      white_wine_data$citric.acid + white_wine_data$free.sulfur.dioxide +
##      white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##      white_wine_data$alcohol)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -3.6075 -0.4987 -0.0424  0.4652  3.1010 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                2.289e+02  2.236e+01 10.236 < 2e-16  
## white_wine_data$fixed.acidity 1.311e-01  2.308e-02  5.681 1.41e-08  
## white_wine_data$volatile.acidity -1.855e+00  1.104e-01 -16.802 < 2e-16  
## white_wine_data$residual.sugar  1.061e-01  8.415e-03 12.608 < 2e-16  
## white_wine_data$citric.acid   2.622e-02  9.594e-02  0.273 0.784637  
## white_wine_data$free.sulfur.dioxide 4.383e-03  6.945e-04  6.311 3.02e-10  
## white_wine_data$density        -2.300e+02  2.264e+01 -10.156 < 2e-16  
## white_wine_data$pH            9.381e-01  1.115e-01  8.413 < 2e-16  
## white_wine_data$sulphates    7.375e-01  1.011e-01  7.293 3.52e-13  
## white_wine_data$alcohol       9.797e-02  2.944e-02  3.328 0.000882  
##
## (Intercept)                 ***
## white_wine_data$fixed.acidity ***
## white_wine_data$volatile.acidity ***
## white_wine_data$residual.sugar ***
## white_wine_data$citric.acid   ***
## white_wine_data$free.sulfur.dioxide ***
## white_wine_data$density        ***
## white_wine_data$pH            ***
## white_wine_data$sulphates    ***
## white_wine_data$alcohol       ***
## ---                        
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7464 on 4879 degrees of freedom
## Multiple R-squared:  0.2892, Adjusted R-squared:  0.2878 
## F-statistic: 220.5 on 9 and 4879 DF,  p-value: < 2.2e-16

```

Observamos que no se ve afectado el R-cuadrado ajustado, su valor 0.7464, por lo que podemos eliminar la variable.

Quitemos la cantidad libre de acido cítrico **citric.acid**, que tenia en el modelo anterior el mayor p-value **0.784637**, a ver como afecta:

```

m4<-lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
       + white_wine_data$residual.sugar +
       white_wine_data$free.sulfur.dioxide +
       white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol)
summary(m4)

##
## Call:
## lm(formula = white_wine_data$quality ~ white_wine_data$fixed.acidity +
##      white_wine_data$volatile.acidity + +white_wine_data$residual.sugar +
##      white_wine_data$free.sulfur.dioxide + white_wine_data$density +
##      white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol)
##
## Residuals:
##    Min      1Q  Median      3Q     Max 
## -3.6123 -0.4980 -0.0443  0.4650  3.1021 
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)    
## (Intercept)                2.283e+02  2.225e+01 10.258 < 2e-16  
## white_wine_data$fixed.acidity   1.316e-01  2.300e-02  5.723 1.11e-08  
## white_wine_data$volatile.acidity -1.860e+00  1.089e-01 -17.068 < 2e-16  
## white_wine_data$residual.sugar    1.059e-01  8.388e-03 12.628 < 2e-16  
## white_wine_data$free.sulfur.dioxide  4.399e-03  6.921e-04  6.355 2.27e-10  
## white_wine_data$density        -2.294e+02  2.253e+01 -10.179 < 2e-16  
## white_wine_data$pH            9.348e-01  1.109e-01  8.433 < 2e-16  
## white_wine_data$sulphates      7.385e-01  1.011e-01  7.307 3.18e-13  
## white_wine_data$alcohol        9.874e-02  2.930e-02  3.370 0.000758 
##
## (Intercept)      ***
## white_wine_data$fixed.acidity ***
## white_wine_data$volatile.acidity ***
## white_wine_data$residual.sugar ***
## white_wine_data$free.sulfur.dioxide ***
## white_wine_data$density ***
## white_wine_data$pH ***
## white_wine_data$sulphates ***
## white_wine_data$alcohol ***
##
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7464 on 4880 degrees of freedom
## Multiple R-squared:  0.2891, Adjusted R-squared:  0.288
## F-statistic: 248.1 on 8 and 4880 DF,  p-value: < 2.2e-16

```

Observamos que no se ve afectado el R-cuadrado ajustado, su valor 0.7464, por lo que podemos eliminar la variable.

Ahora vemos que el modelo que mejor describe la calidad del vino, son: -fixed.acidity -volatile.acidity -residual.sugar -free.sulfur.dioxide -density -pH -sulphates -alcohol

Dado a que el p-value que queda en cada una de las variables es muy baja, mucho mas bajo al p-value de **0.05**, por lo que podemos dejar de eliminar, basandonos en los p-values.

Veamos si mediante ANOVA podemos eliminar mas variables.

ANOVA nos permite comparar multiples muestras, para analizar su varianza. La hipotesis nula implica que no existen diferencias entre las distintas muestras, mientras que la hipotesis alternativa implica lo contrario, que si hay diferencia entre las muestras. Gracias a ANOVA, podremos ver la variabilidad entre modelos de datos, conjuntos de datos y la variabilidad residual.

Probemos a quitarle al modelo la acidez volatil, para ello comparamos el modelo sin acidez volatil, con el modelo completo reducido, obtenido de la fase anterior de los p-value.

Supongamos que tenemos la hipótesis nula de que ambas ecuaciones son iguales, eso quiere decir que si eliminamos una variable, no deberia afectar su resultado.

```

modeloCompletoReducido<-lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                           white_wine_data$volatile.acidity +
                           white_wine_data$residual.sugar +
                           white_wine_data$free.sulfur.dioxide +
                           white_wine_data$density +
                           white_wine_data$pH +
                           white_wine_data$sulphates +
                           white_wine_data$alcohol, data=white_wine_data)
m1anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     #white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     white_wine_data$pH +
                     white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                  ,modeloCompletoReducido)
m1anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$residual.sugar +
##           white_wine_data$free.sulfur.dioxide + white_wine_data$density +
##           white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##           white_wine_data$alcohol
##   Res.Df     RSS Df Sum of Sq      F    Pr(>F)
## 1    4881 2880.8
## 2    4880 2718.6  1    162.29 291.32 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m1anova, 'CriticalValue' = qf(1-.05, m1anova[1,1], m1anova[2,1]))

##   Res.Df     RSS Df Sum of Sq      F    Pr(>F) CriticalValue
## 1    4881 2880.845 NA      NA      NA      1.048221
## 2    4880 2718.555  1    162.2897 291.3216 1.727579e-63      1.048221
alpha = .05
qf(1-alpha, m1anova[1,1], m1anova[2,1])

## [1] 1.048221

```

Al ser el valor de **F 291.3216**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable de acidez volatil.

Probemos ahora a eliminar la variable de azucar residual:

```
m2anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     #white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     white_wine_data$pH +
                     white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                     ,modeloCompletoReducido)

m2anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$free.sulfur.dioxide + white_wine_data$density +
##           white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##           white_wine_data$alcohol
##   Res.Df   RSS Df Sum of Sq    F    Pr(>F)
## 1  4881 2807.4
## 2  4880 2718.6  1     88.83 159.46 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m2anova, 'CriticalValue' = qf(1-.05, m2anova[1,1], m2anova[2,1]))

##   Res.Df   RSS Df Sum of Sq    F    Pr(>F) CriticalValue
## 1  4881 2807.385 NA      NA      NA      NA      1.048221
## 2  4880 2718.555  1  88.82999 159.4562 5.408866e-36      1.048221

alpha = .05
qf(1-alpha, m2anova[1,1], m2anova[2,1])

## [1] 1.048221
```

Al ser el valor de **F 159.46**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable de azucar residual.

Probemos ahora a eliminar la variable de cantidad libre de dioxido de azufre:

```
m3anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     #white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     white_wine_data$pH +
                     white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                     ,modeloCompletoReducido)

m3anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
```

```

##      white_wine_data$residual.sugar + white_wine_data$density +
##      white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##      white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##      white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##      white_wine_data$alcohol
##   Res.Df      RSS Df Sum of Sq      F    Pr(>F)
## 1  4881  2741.1
## 2  4880  2718.6  1      22.5 40.389 2.271e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m3anova, 'CriticalValue' = qf(1-.05, m3anova[1,1], m3anova[2,1]))

##   Res.Df      RSS Df Sum of Sq      F    Pr(>F) CriticalValue
## 1  4881  2741.055 NA      NA      NA      1.048221
## 2  4880  2718.555  1  22.49986 40.38885 2.27086e-10      1.048221
alpha = .05
qf(1-alpha, m3anova[1,1], m3anova[2,1])

## [1] 1.048221

```

Al ser el valor de **F 40.389**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable free.sulfur.dioxide.

Probemos ahora a eliminar la variable de densidad:

```

m4anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     #white_wine_data$density +
                     white_wine_data$pH +
                     white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                  ,modeloCompletoReducido)
m4anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##      white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##      white_wine_data$pH + white_wine_data$sulphates + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##      white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##      white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##      white_wine_data$alcohol
##   Res.Df      RSS Df Sum of Sq      F    Pr(>F)
## 1  4881  2776.3
## 2  4880  2718.6  1      57.72 103.61 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m4anova, 'CriticalValue' = qf(1-.05, m4anova[1,1], m4anova[2,1]))

##   Res.Df      RSS Df Sum of Sq      F    Pr(>F) CriticalValue
## 1  4881  2776.276 NA      NA      NA      1.048221

```

```

## 2 4880 2718.555 1 57.7203 103.6121 4.276933e-24      1.048221
alpha = .05
qf(1-alpha, m4anova[1,1], m4anova[2,1])

## [1] 1.048221

```

Al ser el valor de **F 103.6121**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable density.

Probemos ahora a eliminar la variable de pH:

```

m5anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     #white_wine_data$pH +
                     white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                  ,modeloCompletoReducido)
m5anova

```

```

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$sulphates + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##           white_wine_data$alcohol
##   Res.Df   RSS Df Sum of Sq    F    Pr(>F)
## 1 4881 2758.2
## 2 4880 2718.6  1    39.616 71.114 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m5anova, 'CriticalValue' = qf(1-.05, m5anova[1,1], m5anova[2,1]))

```

```

##   Res.Df   RSS Df Sum of Sq    F    Pr(>F) CriticalValue
## 1 4881 2758.172 NA     NA     NA     1.048221
## 2 4880 2718.555  1  39.61646 71.11436 4.388597e-17    1.048221
alpha = .05
qf(1-alpha, m5anova[1,1], m5anova[2,1])

```

```
## [1] 1.048221
```

Al ser el valor de **F 71.11436**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable pH.

Probemos ahora a eliminar la variable de sulphates:

```

m6anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     white_wine_data$pH +
                     #white_wine_data$sulphates +
                     white_wine_data$alcohol, data=white_wine_data)
                  )

```

```

        ,modeloCompletoReducido)
m6anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$alcohol
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##           white_wine_data$alcohol
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1   4881 2748.3
## 2   4880 2718.6  1    29.742 53.389 3.18e-13 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m6anova, 'CriticalValue' = qf(1-.05, m6anova[1,1], m6anova[2,1]))

##   Res.Df   RSS Df Sum of Sq      F    Pr(>F) CriticalValue
## 1   4881 2748.297 NA     NA     NA     NA       1.048221
## 2   4880 2718.555  1  29.74199 53.38899 3.180237e-13       1.048221
alpha = .05
qf(1-alpha, m6anova[1,1], m6anova[2,1])

## [1] 1.048221

```

Al ser el valor de **F 53.38899**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable sulphates.

Probemos ahora a eliminar la variable de alcohol:

```

m7anova<-anova(lm(white_wine_data$quality ~ white_wine_data$fixed.acidity +
                     white_wine_data$volatile.acidity +
                     white_wine_data$residual.sugar +
                     white_wine_data$free.sulfur.dioxide +
                     white_wine_data$density +
                     white_wine_data$pH +
                     white_wine_data$sulphates
                     #white_wine_data$alcohol
                     , data=white_wine_data)
                  ,modeloCompletoReducido)
m7anova

## Analysis of Variance Table
##
## Model 1: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates
## Model 2: white_wine_data$quality ~ white_wine_data$fixed.acidity + white_wine_data$volatile.acidity +
##           white_wine_data$residual.sugar + white_wine_data$free.sulfur.dioxide +
##           white_wine_data$density + white_wine_data$pH + white_wine_data$sulphates +
##           white_wine_data$alcohol
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1   4881 2724.9
## 2   4880 2718.6  1    6.3259 11.355 0.0007581 ***

```

```

## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
cbind(m7anova, 'CriticalValue' = qf(1-.05, m7anova[1,1], m7anova[2,1]))

##   Res.Df      RSS Df Sum of Sq      F      Pr(>F) CriticalValue
## 1    4881 2724.881 NA       NA       NA       NA     1.048221
## 2    4880 2718.555  1  6.325857 11.35536 0.0007581409     1.048221
alpha = .05
qf(1-alpha, m7anova[1,1], m7anova[2,1])

## [1] 1.048221

```

Al ser el valor de **F 11.35536**, mayor que el valor crítico 1.0482208, no podemos eliminar la variable alcohol.

Con esto podemos observar que valores de F: -Sin volatile.acidity = **F 291.3216** -Sin residual.sugar = **F 159.46** -Sin free.sulfur.dioxide = **F 40.389** -Sin density = **F 103.6121** -Sin pH = **F 71.11436** -Sin sulphates = **F 53.38899** -Sin alcohol = **F 11.35536**

Si comparamos los valores de F de estas variables, ninguna ha bajado el valor crítico, por tanto no deberíamos eliminarlos, pero si eliminásemos el valor de alcohol de la ecuación, al ser el menor de todas las F, pero quizás bajaría su valor R-cuadrado ajustado. De modo que lo mantenemos.

#### 4.3.c)Contraste Hipótesis

Para hacer el contraste de hipótesis nula, utilizamos el Test de Kolmogorov-Smirnov, una prueba no paramétrica que determinará la bondad del ajuste: Asumiendo como hipótesis nula, que los datos están distribuidos de forma normal, si el p-value < 0.05 que es el valor de alpha, se rechaza la hipótesis nula, concluyendo por tanto que no sigue una distribución normal.

```

ks.test(x = as.numeric(white_wine_data$quality)
        , "pnorm"
        , mean(as.numeric(white_wine_data$quality))
        , sd(as.numeric(white_wine_data$quality)))

## Warning in ks.test(x = as.numeric(white_wine_data$quality), "pnorm",
## mean(as.numeric(white_wine_data$quality)), : ties should not be present for
## the Kolmogorov-Smirnov test

##
## One-sample Kolmogorov-Smirnov test
##
## data: as.numeric(white_wine_data$quality)
## D = 0.22874, p-value < 2.2e-16
## alternative hypothesis: two-sided

```

## Sección 5

### 5.Representación de los resultados a partir de tablas y gráficas.

Además de los gráficos vistos ya en puntos anteriores, quisiera destacar un breve resumen de algún gráfico que nos ayude a entender mejor el resultado.

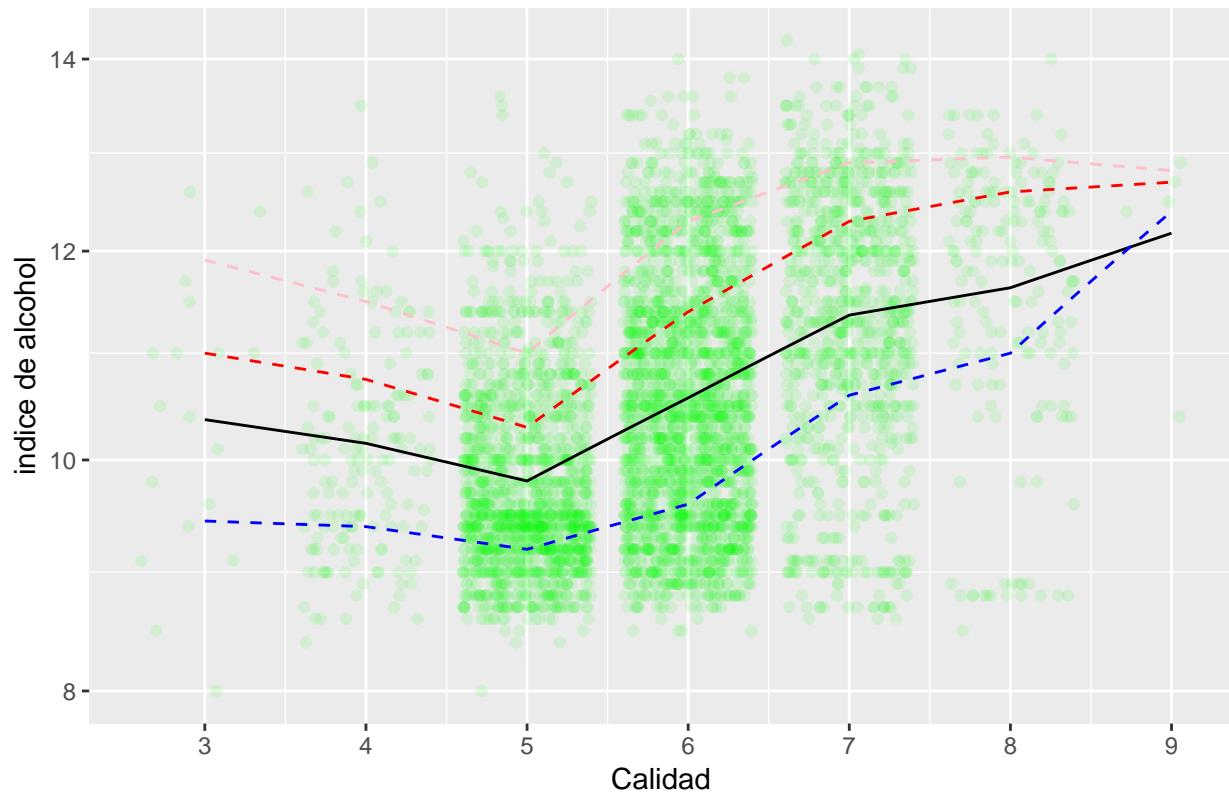
Podemos observar que la cantidad de alcohol para el 25% de las muestras (línea azul), 50% de las muestras (línea negra), 75% de las muestras (línea roja) y el 90% de las muestras (línea rosa) de cada tipo de calidad, muestran que un incremento de alcohol hace que suba la calidad desde el punto 5. Es decir mayor cantidad de alcohol hace que la calidad suba desde el nivel 5. Hasta llegar a un índice de alcohol entre 12 y 13, que es donde se encuentra la mayor calidad

```

ggplot(aes(x=quality, y = alcohol), data = white_wine_data) +
  geom_point(alpha = 0.1, position = position_jitter(h=0), color='green') +
  coord_cartesian(xlim = c(2,8)) + coord_cartesian(ylim = c(0.8,1.2)) + coord_trans(y = 'sqrt') +
  geom_line(stat = 'summary', fun.y = 'mean') +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.25), linetype =2, color = 'blue') +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.75), linetype =2, color = 'red') +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.9), linetype =2, color = 'pink') +
  labs(x = "Calidad", y = "indice de alcohol", title = "Influencia del Indice de alcohol en la calidad del vino")
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
## Coordinate system already present. Adding new coordinate system, which will replace the existing one

```

### Influencia del Indice de alcohol en la calidad del vino



Si lo vemos por categorías, podemos observar la misma conclusión, donde podemos ver que la calidad del vino para niveles de alcohol 12, donde está la media de valores de dicha calidad, es alta. La calidad del vino para el valor medio de alcohol de 9.5 a 10 es un vino de baja calidad. La calidad del vino para el valor medio de 10 a 10.5 es un valor de calidad media.

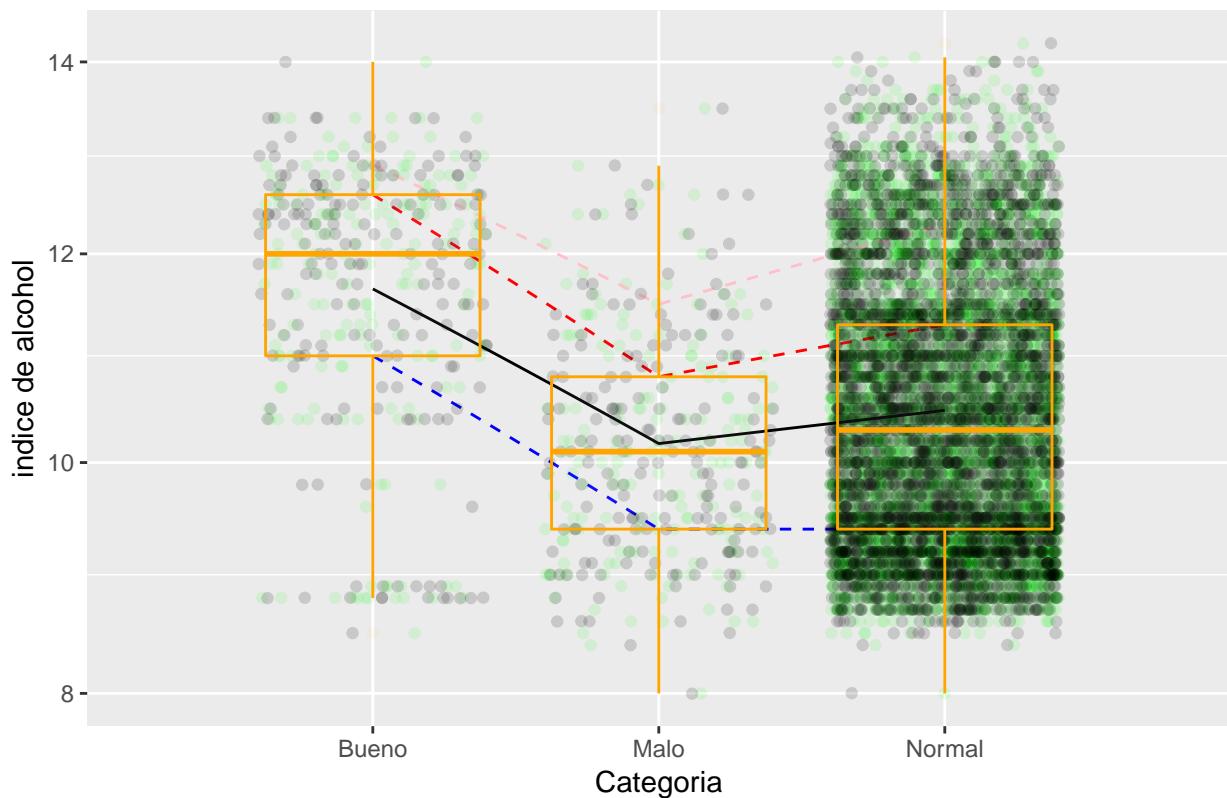
```

ggplot(aes(x=category, y = alcohol), data = white_wine_data) +
  geom_point(alpha = 0.1, position = position_jitter(h=0), color='green') +
  coord_cartesian(xlim = c(2,8)) + coord_cartesian(ylim = c(0.8,1.2)) + coord_trans(y = 'sqrt') +
  geom_line(stat = 'summary', fun.y = 'mean',group = 1) +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.25), linetype =2, color = 'blue') +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.75), linetype =2, color = 'red') +
  geom_line(stat = 'summary', fun.y = 'quantile', fun.args=list(probs=0.9), linetype =2, color = 'pink') +
  geom_jitter( alpha = .15) + geom_boxplot(alpha = .05,color = 'orange') +
  labs(x = "Categoria", y = "indice de alcohol",
       title = "Influencia del Indice de alcohol en la calidad del vino")

```

```
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
## Coordinate system already present. Adding new coordinate system, which will replace the existing one
```

### Influencia del Indice de alcohol en la calidad del vino



Aqui observamos que el numero de vinos calificados como buenos o malos son la gran minoría, es normal, ya que los vinos normales son los mas accesibles al consumidor medio y se comprará mas que uno bueno, quizas por mayor precio.

```
print(" Numero de vinos por su categoria:")
```

```
## [1] " Numero de vinos por su categoria:"
```

```
table(white_wine_data$category)
```

```
##
```

```
##   Bueno    Malo Normal
```

```
##     180     181    4528
```

No hay vinos de calidad superior a 9, es decir vinos de puntuación 10, ni tampoco vinos de calidad inferior a 3. Los vinos buenos suelen encontrarse entre 8 y 9, mientras que la mayor parte de los vinos se encuentran entre 5 y 7.9

```
print(" Numero de vinos por su calidad:")
```

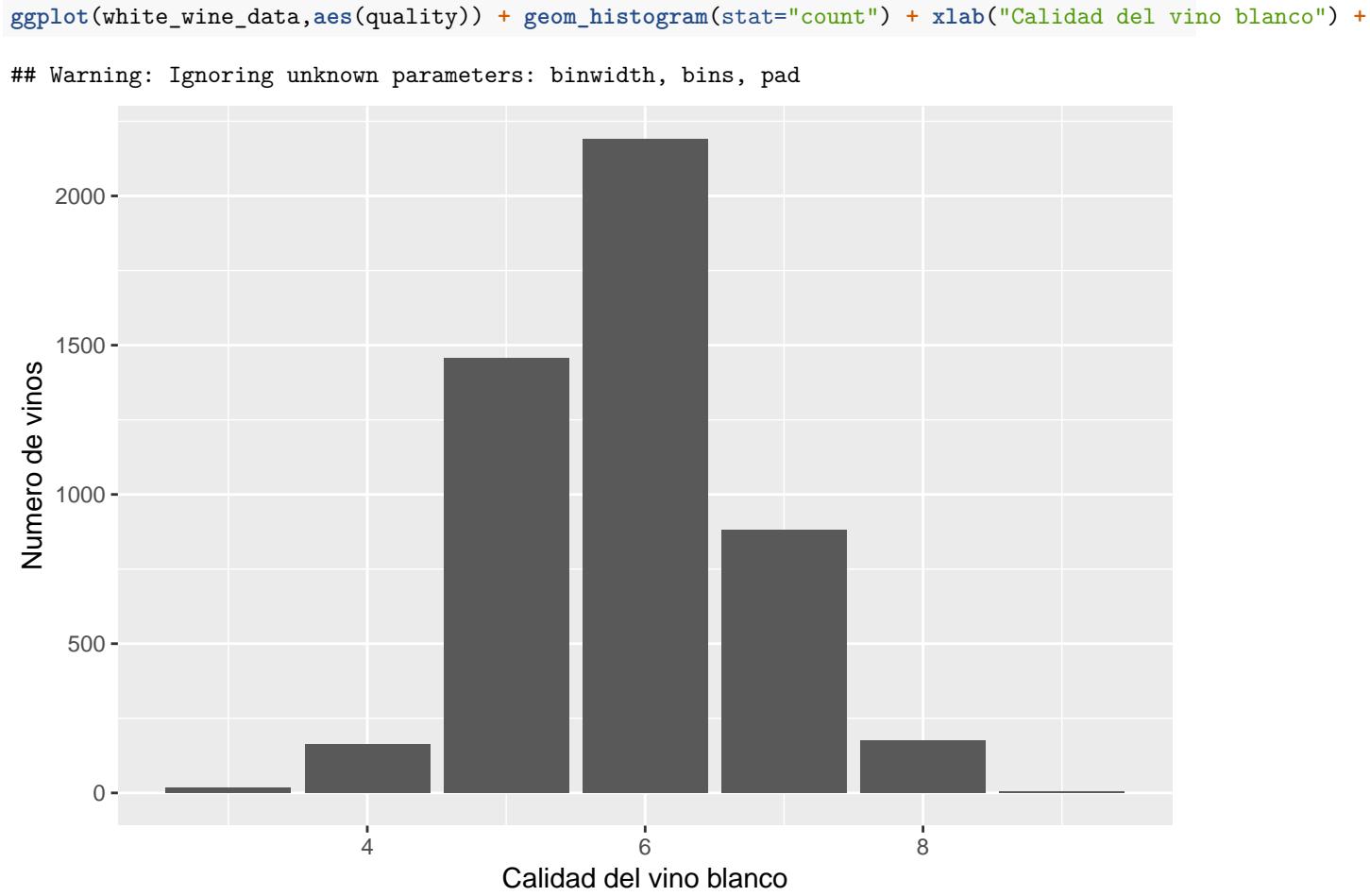
```
## [1] " Numero de vinos por su calidad:"
```

```
table(white_wine_data$quality)
```

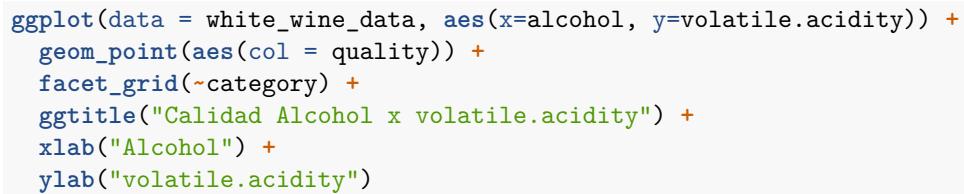
```
##
```

```
##   3    4    5    6    7    8    9
```

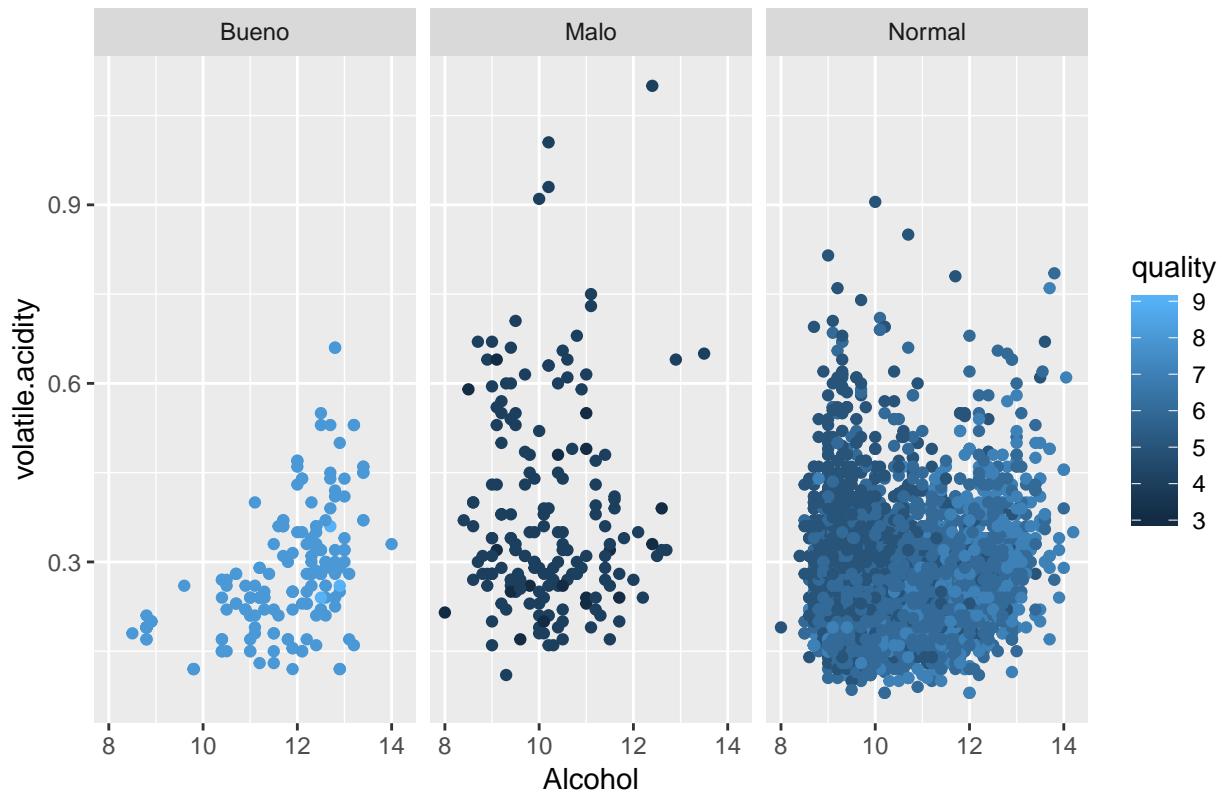
```
##   18   163  1457  2191   880   175    5
```



Vemos que cuanto mayor alcohol y menor acidez colatil, mejor vino.



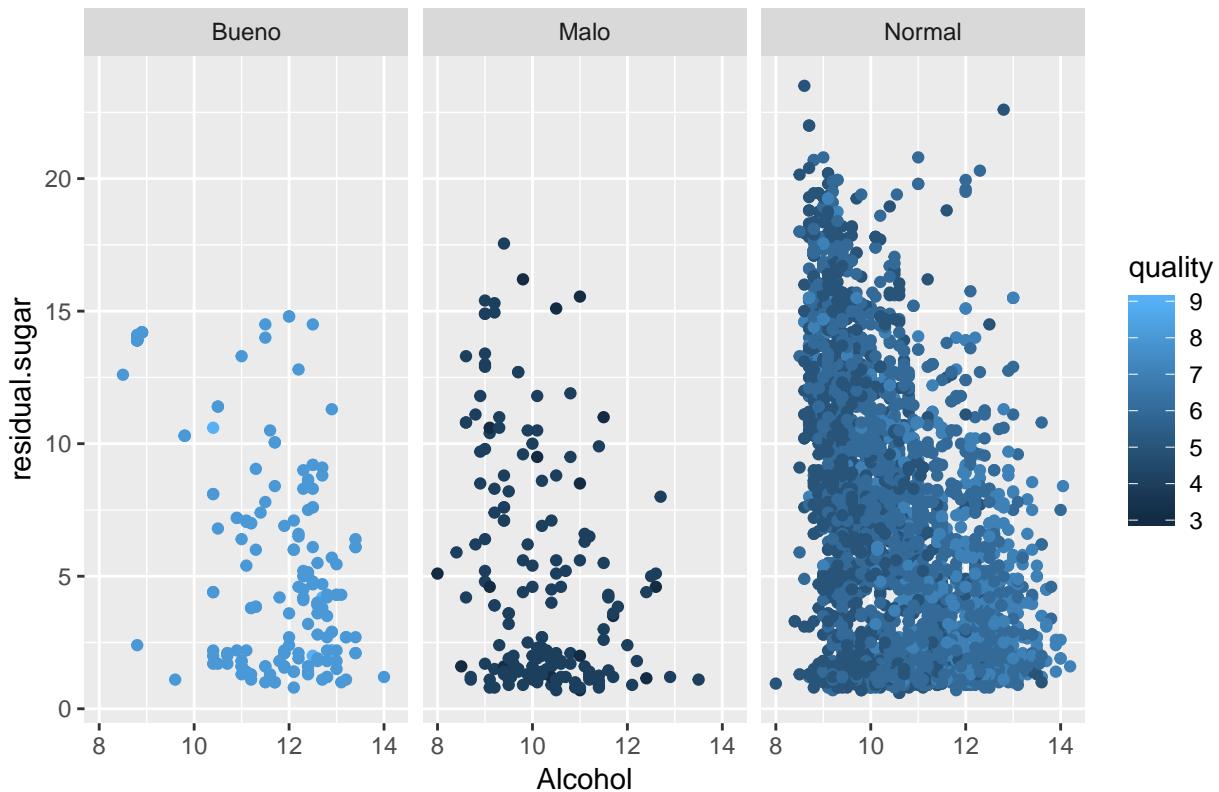
## Calidad Alcohol x volatile.acidity



Vemos que cuanto menor alcohol y mayor azucar residual, peor vino.

```
ggplot(data = white_wine_data, aes(x=alcohol, y=residual.sugar)) +  
  geom_point(aes(col = quality)) +  
  facet_grid(~category) +  
  ggtitle("Calidad Alcohol x residual.sugar") +  
  xlab("Alcohol") +  
  ylab("residual.sugar")
```

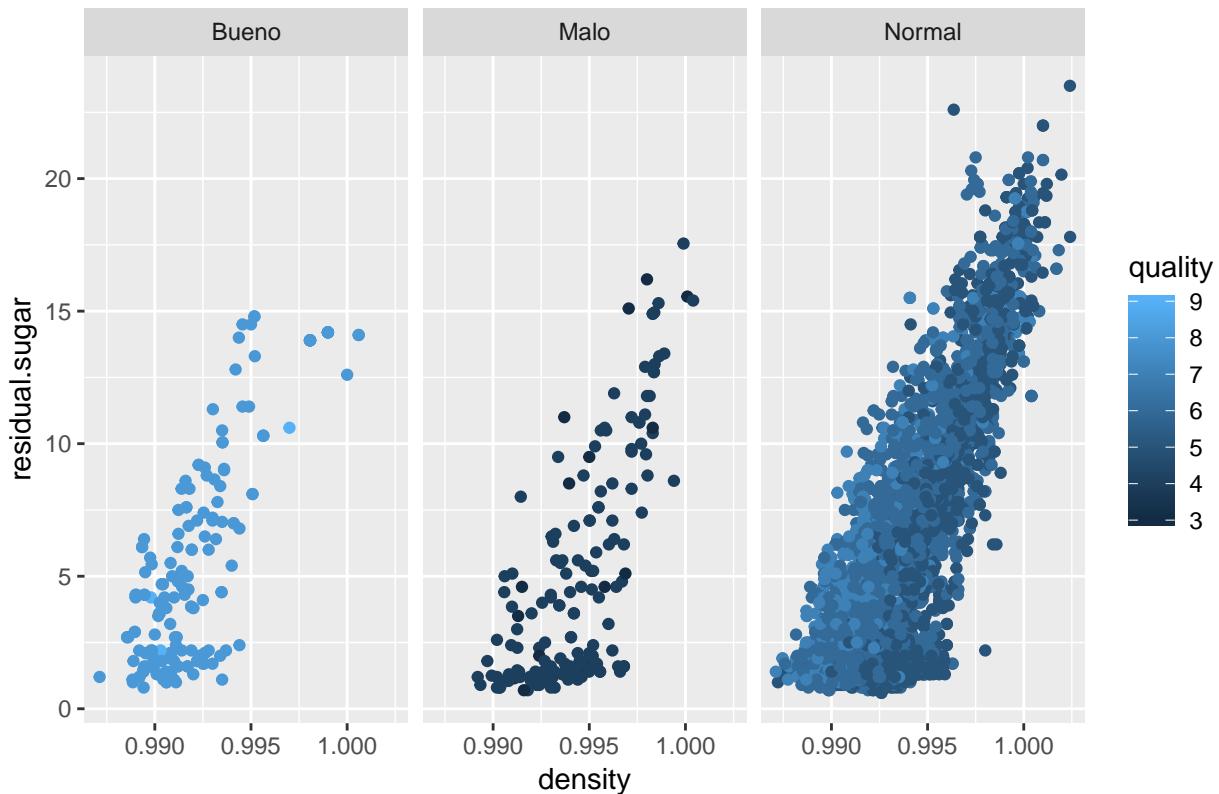
## Calidad Alcohol x residual.sugar



La densidad y el azucar estan relacionados, cuanto mayor densidad, mas azucar contiene.

```
ggplot(data = white_wine_data, aes(x=density, y=residual.sugar)) +  
  geom_point(aes(col = quality)) +  
  facet_grid(~category) +  
  ggtitle("Calidad density x residual.sugar") +  
  xlab("density") +  
  ylab("residual.sugar")
```

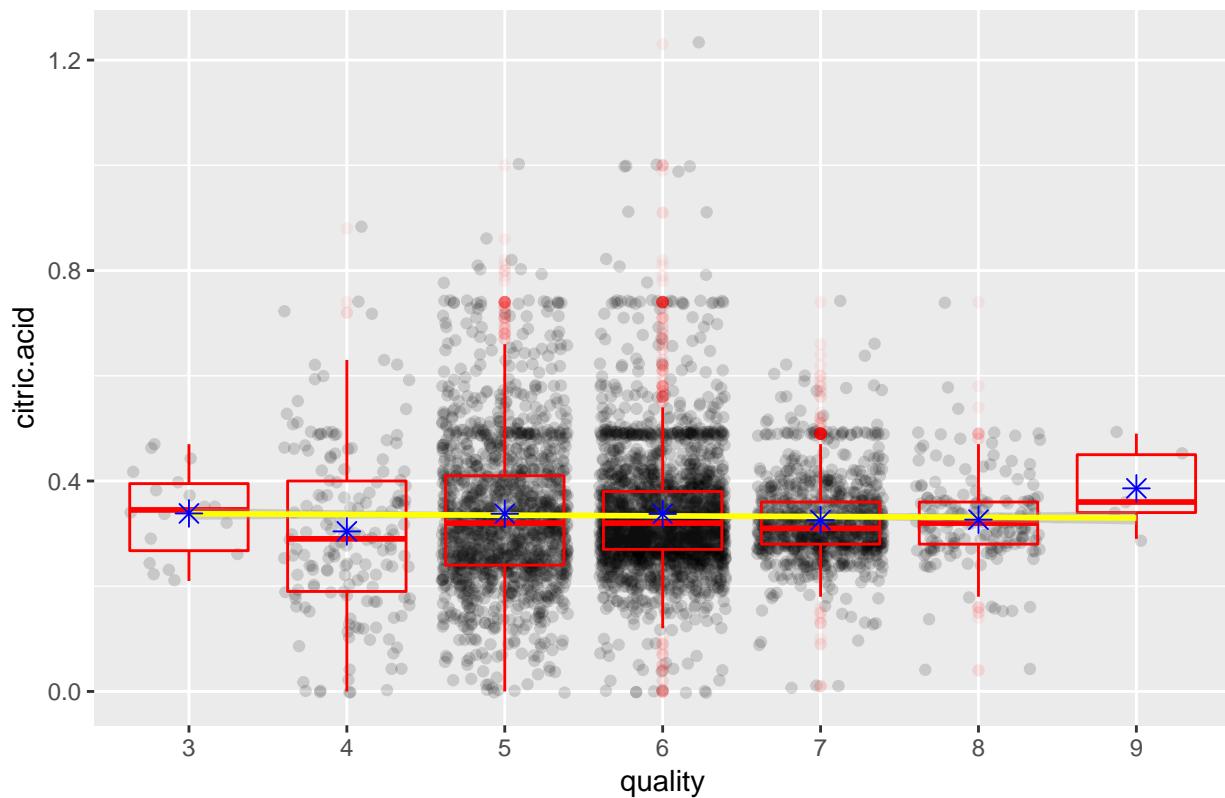
## Calidad density x residual.sugar



Acidez citrica, apenas tiene repercusion el la calidad.

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = citric.acid)) +
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +
  geom_smooth(aes(quality-2,citric.acid),method='lm',color='yellow')+
  xlab('quality')+ ylab('citric.acid') +
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +
  ggtitle('Calidad del vino vs citric.acid')
```

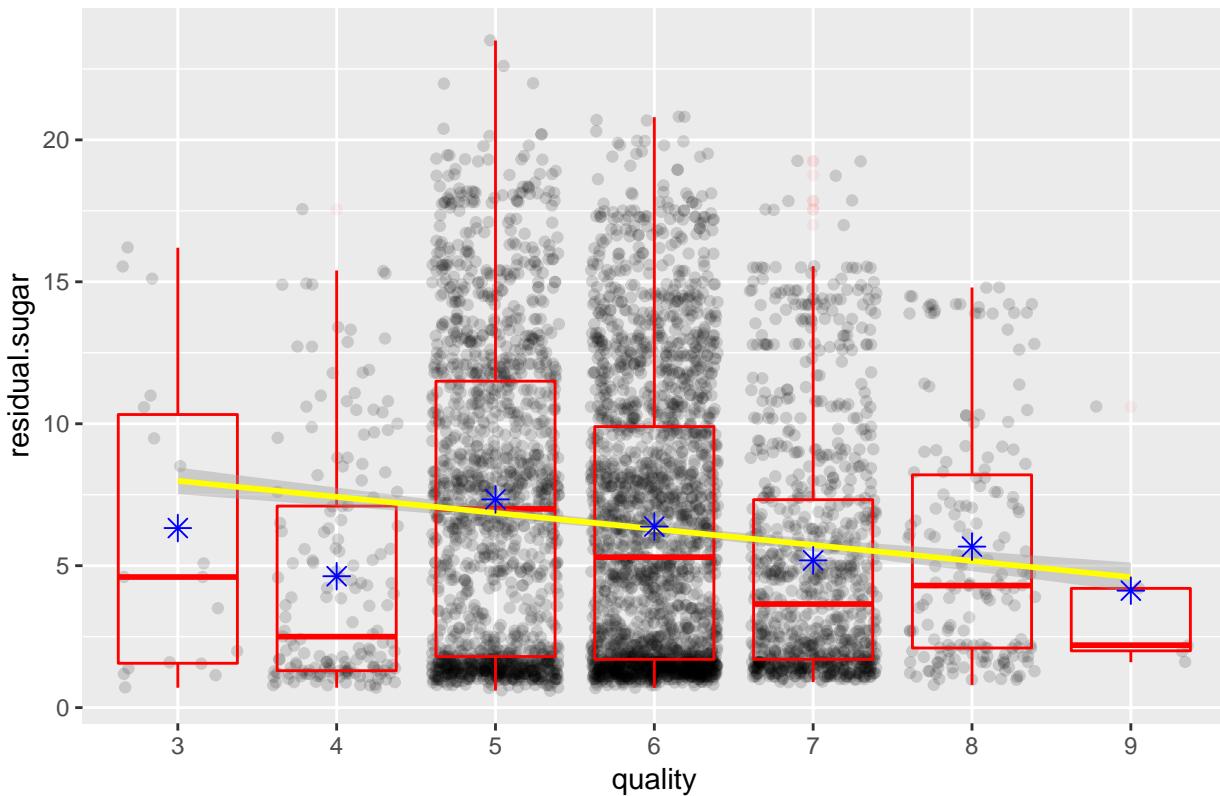
## Calidad del vino vs citric.acid



El azucar, puede influir en la calidad del vino, encontrando menor cantidad de azucar en mejores vinos.

```
ggplot(data = white_wine_data, aes(x = factor(quality), y = residual.sugar)) +  
  geom_jitter(alpha = .15) + geom_boxplot(alpha = .05,color = 'red') +  
  geom_smooth(aes(quality-2,residual.sugar),method='lm',color='yellow')+  
  xlab('quality')+ ylab('residual.sugar') +  
  stat_summary(fun.y = "mean", geom = "point",color = "blue",shape = 8,size = 3) +  
  ggtitle('Calidad del vino vs residual.sugar')
```

## Calidad del vino vs residual.sugar



## Sección 6

**6. Resolución del problema.** A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Tras determinar las variables que se veian relacionadas con el factor de calidad:

- volatile.acidity
- residual.sugar
- free.sulfur.dioxide
- density
- pH
- sulphates
- alcohol

Inicialmente pensaba que el citrico podia influir en gran medida en la calidad del vino, incluso que la cantidad de azucar lo hariadotaria de mejor calidad no obstante, he podido observar, que la cantidad de azucar encontrado, se ve relacionado de forma muy débil con la calidad del vino, si que veo que tiene mínimamente una influencia sobre ella, pero se puede despreciar.

Por otra parte, en cuanto a la cantidad de acido cítrico, he podido observar que no influye para nada en la calidad del vino, por lo que podemos despreciarla.

En cuanto al pH, la mayoria de vinos se encuentran en la misma franja de 3 a 3.4, por lo que no es determinante para determinar su calidad.

Finalmente podemos inferir que la calidad del vino se ve influenciado por el indice de alcohol, a mayor indice, mayor calidad, tampoco es pasarse de poner el máximo valor de alcohol, pero si una cantidad que sea entre 10.5 y 13, permitirá encontrar la franja de mejores vinos.

#Sección Guardar Dataframe:

```
write.csv(white_wine_data, "white_wine_data_final.csv", row.names = FALSE)
```

## **Sección Contribuciones:**

**Investigación Previa:** Ramón Serrano Valero

**Redacción Respuestas:** Ramón Serrano Valero

**Desarrollo Código:** Ramón Serrano Valero