

Práctica 2: Limpieza y análisis de datos

Daria Gracheva, Zechao Jin

5 de January, 2021

Contents

Descripción del dataset.	1
Integración y selección de los datos de interés a analizar.	2
Limpieza y preprocesamiento de los datos	4
Valores nulos	6
Unidades de medida	7
Outliers	8
Discretización	12
Exportación de los datos preprocesados	15
Análisis	15
Selección de los grupos de datos que se quieren analizar/comparar	15
Comprobación de la normalidad y homogeneidad de la varianza	19
Estudio de correlación	21
Análisis inferencial	22
Modelización predictiva	23
Conclusión; Resolución del problema.	29

Descripción del dataset.

El dataset elegido es “Wine quality dataset”, el cual es originalmente publicado en el repositorio UCI Machine Learning y posteriormente en plataforma kaggle (<https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009>; <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>).

Se trata de 2 conjuntos de datos para tipos tinto y blanco del ‘vinho verde’ cada uno de cuales contiene 11 atributos numéricos (características fisicoquímicas del vino) y la marca de clase que corresponde a la calidad sensorial del vino en una escala de 0 a 10. El dataset de los vinos tintos contiene 1599 observaciones, y el de vinos blancos - 4898 observaciones, mientras la marca de calidad tiene un sesgo para los vinos de calidad “normal”, según los autores.

¿Por qué es importante y qué pregunta/problema pretende responder?

Con este estudio se pretende explicar y perfilar la calidad de los vinos de distintos colores según sus características fisicoquímicas. La calidad es un paradigma común y entendible, las características son fácilmente interpretables y el ámbito del tema es generalmente conocido pues tiene un alto potencial de dar un resultado divulgativo.

Teniendo en cuenta que se había usado la misma metodología para recopilar los datos (los dos datasets provienen de la misma fuente) y los atributos son los mismos, y aunque la dimensionalidad de datasets no es igual, eso nos permite comparar los dos conjuntos estadísticamente. Con los datos que disponemos, se puede llevar a cabo estudios tanto de cada una de las muestras (por ejemplo, para explicar la calidad de cada uno de los tipos de vinos), como de dos muestras (comparar estadísticamente parámetros o proporciones), y también tener intuición sobre similitudes y diferencias de los vinos.

Puesto que hay una marca de clase original, el dataset permite crear modelos supervisados de clasificación para poder predecir la calidad sensorial en función de los atributos fisicoquímicos.

Integración y selección de los datos de interés a analizar.

Cargamos los dos datasets:

```
# Carga del dataset
red <- read.csv('https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-red.csv')

white <- read.csv('https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv')

# Nombres de los atributos
names(red) <- c("fixed acidity", "volatile acidity", "citric acid", "residual sugar", "chlorides", "free sulfur dioxide", "total sulfur dioxide", "density", "pH", "sulphates", "alcohol", "quality")
names(white) <- c("fixed acidity", "volatile acidity", "citric acid", "residual sugar", "chlorides", "free sulfur dioxide", "total sulfur dioxide", "density", "pH", "sulphates", "alcohol", "quality")

# Verificamos la estructura de los conjuntos de datos
str(red)
```

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

str(white)
```

```
## 'data.frame': 4898 obs. of 12 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 6 ...
```

Se observa que tenemos todas las variables bien numéricas continuas o discretas.

Descripción de variables: son las características físicas o químicas del vino más su calidad y su color/tipo (tinto o blanco)

fixed acidity : acidez fija (g/l), v. continua

volatile acidity : acidez volátil (g/l), v. continua

citric acid : ácido cítrico (g/l), v. continua

residual sugar : azúcar residual (g/l), v. continua

chlorides : cloruros (g/l), v. continua
 free sulfur dioxide : dióxido de azufre libre (mg/l), v. continua
 total sulfur dioxide: dióxido de azufre total (mg/l), v. continua
 density : densidad (g/l), v. continua
 pH : pH, v. continua
 sulphates : sulfatos (g/l), v. continua
 alcohol : concentración de alcohol (%), v. continua
 quality : calidad, v. discreta

Veamos como son algunas de las observaciones:

```
rbind(head(red,3), tail(red,3))
```

```
##      fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.4           0.700           0.00           1.9       0.076
## 2           7.8           0.880           0.00           2.6       0.098
## 3           7.8           0.760           0.04           2.3       0.092
## 1597         6.3           0.510           0.13           2.3       0.076
## 1598         5.9           0.645           0.12           2.0       0.075
## 1599         6.0           0.310           0.47           3.6       0.067
##      free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates
## 1              11              34 0.99780 3.51       0.56
## 2              25              67 0.99680 3.20       0.68
## 3              15              54 0.99700 3.26       0.65
## 1597            29              40 0.99574 3.42       0.75
## 1598            32              44 0.99547 3.57       0.71
## 1599            18              42 0.99549 3.39       0.66
##      alcohol quality
## 1          9.4      5
## 2          9.8      5
## 3          9.8      5
## 1597       11.0      6
## 1598       10.2      5
## 1599       11.0      6
```

```
rbind(head(white,3), tail(white,3))
```

```
##      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
## 4896         6.5           0.24           0.19           1.2       0.041
## 4897         5.5           0.29           0.30           1.1       0.022
## 4898         6.0           0.21           0.38           0.8       0.020
##      free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates
## 1              45              170 1.00100 3.00       0.45
## 2              14              132 0.99400 3.30       0.49
## 3              30              97 0.99510 3.26       0.44
## 4896            30              111 0.99254 2.99       0.46
## 4897            20              110 0.98869 3.34       0.38
## 4898            22              98 0.98941 3.26       0.32
##      alcohol quality
## 1          8.8      6
## 2          9.5      6
## 3         10.1      6
## 4896         9.4      6
```

```
## 4897    12.8      7
## 4898    11.8      6
```

Limpieza y preprocesamiento de los datos

Veamos las estadísticas básicas:

```
summary(red)
```

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

```
summary(white)
```

```
## 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
## Min. :3.000
## 1st Qu.:5.000
## Median :6.000
## Mean :5.878
## 3rd Qu.:6.000
## Max. :9.000
```

Todas las columnas parecen ser bastante limpias, no obstante aquí se observa la heterogeneidad de las variables (por ejemplo, cloruros que no superan 0.611 g/l y dióxido de sulfuro total que “alcanza” 440 mg/l: dado que tienen las unidades distintas se produce esta brecha).

En cuanto a la calidad, el valor mínimo es 3 y el máximo es 9 (para vinos blancos). Por lo que la escala real sería 3-9. Comprobamos la distribución de calidad:

```
table(red$quality)
```

```
##
## 3 4 5 6 7 8
## 10 53 681 638 199 18
```

```
table(white$quality)
```

```
##
## 3 4 5 6 7 8 9
## 20 163 1457 2198 880 175 5
```

Tal y como se ha dicho anteriormente, la distribución de clases no es balanceada, con vinos normales más representados. Con el atributo “quality” podemos tener 7 marcas de clase diferentes, aunque puede ser conveniente agruparlo en una variable discreta como vemos más adelante.

Distribución de valores únicos de atributos:

```
apply(red,2, function(x) length(unique(x)))
```

```
## fixed.acidity volatile.acidity citric.acid
## 96 143 80
## residual.sugar chlorides free.sulfur.dioxide
## 91 153 60
## total.sulfur.dioxide density pH
## 144 436 89
## sulphates alcohol quality
## 96 65 6
```

```
apply(white,2, function(x) length(unique(x)))
```

```
## fixed.acidity volatile.acidity citric.acid
## 68 125 87
## residual.sugar chlorides free.sulfur.dioxide
## 310 160 132
## total.sulfur.dioxide density pH
## 251 890 103
## sulphates alcohol quality
```

```
##
```

```
79
```

```
103
```

```
7
```

Puesto que el número de observaciones de vinos tintos y blancos es muy distinto, también se observa un sesgo en la variabilidad natural de los atributos de los vinos.

Valores nulos

Comprobamos si hay valores nulos en el dataset:

```
any(is.na(red))
```

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

```
any(is.na(white))
```

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

```
any(red=="")
```

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

```
any(white=="")
```

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

A partir de las estadísticas se ve que todas las variables tienen un mínimo distinto del cero menos la variable “citric acid”, y podría tomar un 0 como un valor desconocido.

Veamos la distribución de “citric acid”:

```
table(red$citric.acid)
```

```
##
##      0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14
## 132  33  50  30  29  20  24  22  33  30  35  15  27  18  21
## 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24 0.25 0.26 0.27 0.28 0.29
##  19   9  16  22  21  25  33  27  25  51  27  38  20  19  21
## 0.3 0.31 0.32 0.33 0.34 0.35 0.36 0.37 0.38 0.39 0.4 0.41 0.42 0.43 0.44
##  30  30  32  25  24  13  20  19  14  28  29  16  29  15  23
## 0.45 0.46 0.47 0.48 0.49 0.5 0.51 0.52 0.53 0.54 0.55 0.56 0.57 0.58 0.59
##  22  19  18  23  68  20  13  17  14  13  12   8   9   9   8
## 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.72 0.73 0.74
##   9   2   1  10   9   7  14   2  11   4   2   1   1   3   4
## 0.75 0.76 0.78 0.79   1
##   1   3   1   1   1
```

```
table(white$citric.acid)
```

```
##
##      0 0.01 0.02 0.03 0.04 0.05 0.06 0.07 0.08 0.09 0.1 0.11 0.12 0.13 0.14
##  19   7   6   2  12   5   6  12   4  12  14   1  19  17  27
## 0.15 0.16 0.17 0.18 0.19 0.2 0.21 0.22 0.23 0.24 0.25 0.26 0.27 0.28 0.29
##  23  33  27  49  48  70  66 104  83 181 136 219 216 282 223
## 0.3 0.31 0.32 0.33 0.34 0.35 0.36 0.37 0.38 0.39 0.4 0.41 0.42 0.43 0.44
## 307 200 257 183 225 137 177 134 122 101 117  82  95  37  63
## 0.45 0.46 0.47 0.48 0.49 0.5 0.51 0.52 0.53 0.54 0.55 0.56 0.57 0.58 0.59
##  46  51  38  39 215  35  25  23  16  19  11  22  13  21   6
## 0.6 0.61 0.62 0.63 0.64 0.65 0.66 0.67 0.68 0.69 0.7 0.71 0.72 0.73 0.74
##   6   9  14   4   6   8   7   7   7   5   3   9   5   5  41
## 0.78 0.79 0.8 0.81 0.82 0.86 0.88 0.91 0.99   1 1.23 1.66
##   2   2   2   2   2   1   1   2   1   5   1   1
```

Observación “0” supone alrededor de 8% de la distribución de vinos tintos y 0,3% en vinos blancos, y es comparable con algunas otras frecuencias, por lo que puede ser valor real (=“no se añade el ácido cítrico”) y no perdido o nulo.

Unidades de medida

Como hemos visto, hay variables que tienen distintas unidades de medidas (hablando de variables de misma naturaleza), podemos reducirlas a las mismas medidas -por ejemplo, g/l.

Hay dos variables que usan otras unidades: “free/total sulfur dioxide”, que en g/l tendrían la distribución parecida a la de “chlorides”.

```
# Cambio de unidades de mg/l a g/l
red$free.sulfur.dioxide <- red$free.sulfur.dioxide * 0.001
red$total.sulfur.dioxide <- red$total.sulfur.dioxide * 0.001

white$free.sulfur.dioxide <- white$free.sulfur.dioxide * 0.001
white$total.sulfur.dioxide <- white$total.sulfur.dioxide * 0.001
```

Visualizamos de nuevo las estadísticas:

```
summary(red)
```

```
## fixed.acidity  volatile.acidity  citric.acid  residual.sugar
## Min.      : 4.60  Min.      :0.1200  Min.      :0.000  Min.      : 0.900
## 1st Qu.: 7.10  1st Qu.:0.3900  1st Qu.:0.090  1st Qu.: 1.900
## Median : 7.90  Median :0.5200  Median :0.260  Median : 2.200
## Mean   : 8.32  Mean   :0.5278  Mean   :0.271  Mean   : 2.539
## 3rd Qu.: 9.20  3rd Qu.:0.6400  3rd Qu.:0.420  3rd Qu.: 2.600
## Max.   :15.90  Max.   :1.5800  Max.   :1.000  Max.   :15.500
## chlorides      free.sulfur.dioxide total.sulfur.dioxide
## Min.      :0.01200  Min.      :0.00100  Min.      :0.00600
## 1st Qu.:0.07000  1st Qu.:0.00700  1st Qu.:0.02200
## Median :0.07900  Median :0.01400  Median :0.03800
## Mean   :0.08747  Mean   :0.01587  Mean   :0.04647
## 3rd Qu.:0.09000  3rd Qu.:0.02100  3rd Qu.:0.06200
## Max.   :0.61100  Max.   :0.07200  Max.   :0.28900
## density        pH          sulphates      alcohol
## Min.      :0.9901  Min.      :2.740  Min.      :0.3300  Min.      : 8.40
## 1st Qu.:0.9956  1st Qu.:3.210  1st Qu.:0.5500  1st Qu.: 9.50
## Median :0.9968  Median :3.310  Median :0.6200  Median :10.20
## Mean   :0.9967  Mean   :3.311  Mean   :0.6581  Mean   :10.42
## 3rd Qu.:0.9978  3rd Qu.:3.400  3rd Qu.:0.7300  3rd Qu.:11.10
## Max.   :1.0037  Max.   :4.010  Max.   :2.0000  Max.   :14.90
## quality
## Min.      :3.000
## 1st Qu.:5.000
## Median :6.000
## Mean   :5.636
## 3rd Qu.:6.000
## Max.   :8.000
```

```
summary(white)
```

```
## 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.   :0.00200    Min.   :0.0090
## 1st Qu.:0.03600    1st Qu.:0.02300    1st Qu.:0.1080
## Median :0.04300    Median :0.03400    Median :0.1340
## Mean   :0.04577    Mean   :0.03531    Mean   :0.1384
## 3rd Qu.:0.05000    3rd Qu.:0.04600    3rd Qu.:0.1670
## Max.   :0.34600    Max.   :0.28900    Max.   :0.4400
## 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
## Min.   :3.000
## 1st Qu.:5.000
## Median :6.000
## Mean   :5.878
## 3rd Qu.:6.000
## Max.   :9.000
```

Outliers

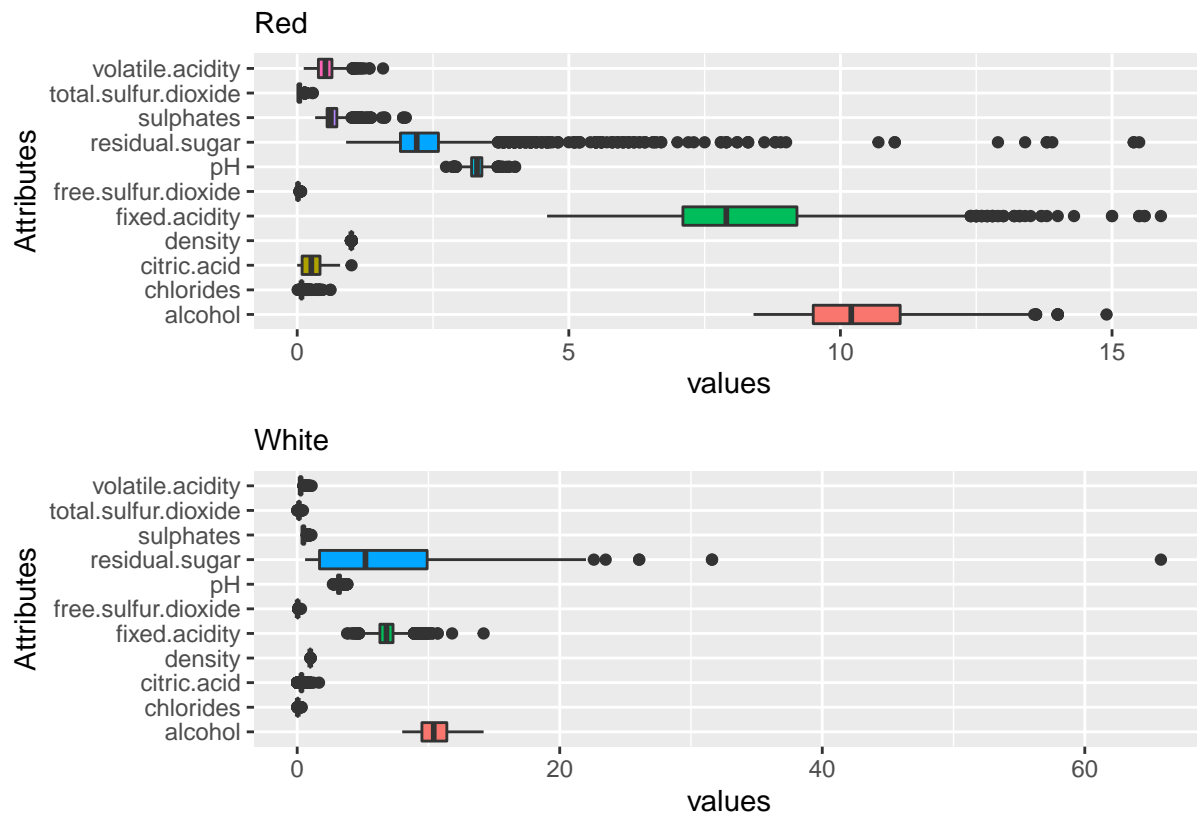
El dataset parece tener outliers ya que en muchas variables la diferencia entre el tercer cuantil y el máximo es considerable.

Visualizamos los boxplots de los datasets:

```
bp1 <- red %>%
  gather(Attributes, values, c(1:11)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

bp2 <- white %>%
  gather(Attributes, values, c(1:11)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

bp1 + labs(subtitle = "Red") + bp2 + labs(subtitle = "White") + plot_layout(nrow = 2)
```

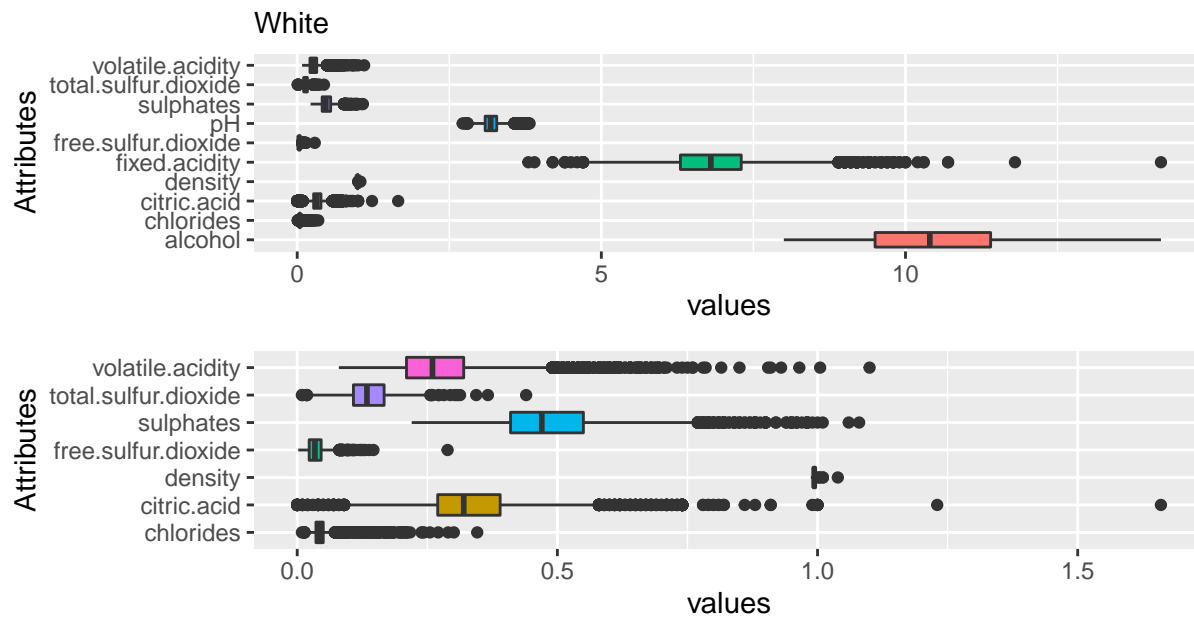



Se observa que las variable “residual sugar” de vinos blancos tiene outliers muy distantes, por lo que es difícil de visualizar otras características.

```
p1 <- white %>%
  gather(Attributes, values, c(1:3,5:11)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

p2 <- white %>%
  gather(Attributes, values, c(2:3,5:8,10)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

p1 + labs(subtitle = "White") + p2 + plot_layout(nrow = 3)
```



Tenemos algunos valores bastante anómalos con las colas por la derecha, sobre todo en variables “residual sugar”, citric acid“,”free sulfur dioxide“,”sulphates“,”volatile acidity“,”sulphates“,”chlorides”.

No obstante, son relativamente pocas observaciones por lo que se puede realizar una imputación por la mediana.

Por ello, reemplazamos los valores extremos según el estadístico de boxplot por NA:

```
for (x in c("residual.sugar", "citric.acid", "free.sulfur.dioxide", "sulphates", "volatile.acidity", "chlorides")) {
  red[,x][red[,x] %in% (boxplot.stats(red[,x])$out) ] <- NA
  white[,x][white[,x] %in% (boxplot.stats(white[,x])$out) ] <- NA
}
```

Cantidad de outliers detectados:

```
sapply(red, function(red) sum(is.na(red)))
```

```
##      fixed.acidity    volatile.acidity    citric.acid
##             49             19             1
##      residual.sugar    chlorides    free.sulfur.dioxide
##             155             112             30
## total.sulfur.dioxide    density    pH
##             55             45             35
##      sulphates    alcohol    quality
##             59             13             0
```

```
sapply(white, function(white) sum(is.na(white)))
```

```
##      fixed.acidity    volatile.acidity    citric.acid
##             119             186             270
##      residual.sugar    chlorides    free.sulfur.dioxide
##              7             208             50
## total.sulfur.dioxide    density    pH
##             19              5             75
##      sulphates    alcohol    quality
##            124              0             0
```

Un ejemplo de observaciones con outliers:

```
head(red[is.na(red$alcohol),],5)
```

```
##      fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 143          5.2           0.34         0.00           1.8      0.050
## 145          5.2           0.34         0.00           1.8      0.050
## 468          8.8           0.46         0.45           2.6      0.065
## 589          5.0           0.42         0.24           2.0      0.060
## 653          NA           0.36         0.65           NA      0.096
##      free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates
## 143          0.027           0.063         NA 3.68      0.79
## 145          0.027           0.063         NA 3.68      0.79
## 468          0.007           0.018      0.9947 3.32      0.79
## 589          0.019           0.050         NA  NA      0.74
## 653          0.022           0.071      0.9976 2.98      0.84
##      alcohol quality
## 143      NA        6
## 145      NA        6
## 468      NA        6
## 589      NA        8
## 653      NA        5
```

Podemos ver que algunos valores atípicos se encuentran en las mismas observaciones.

Imputación por la mediana:

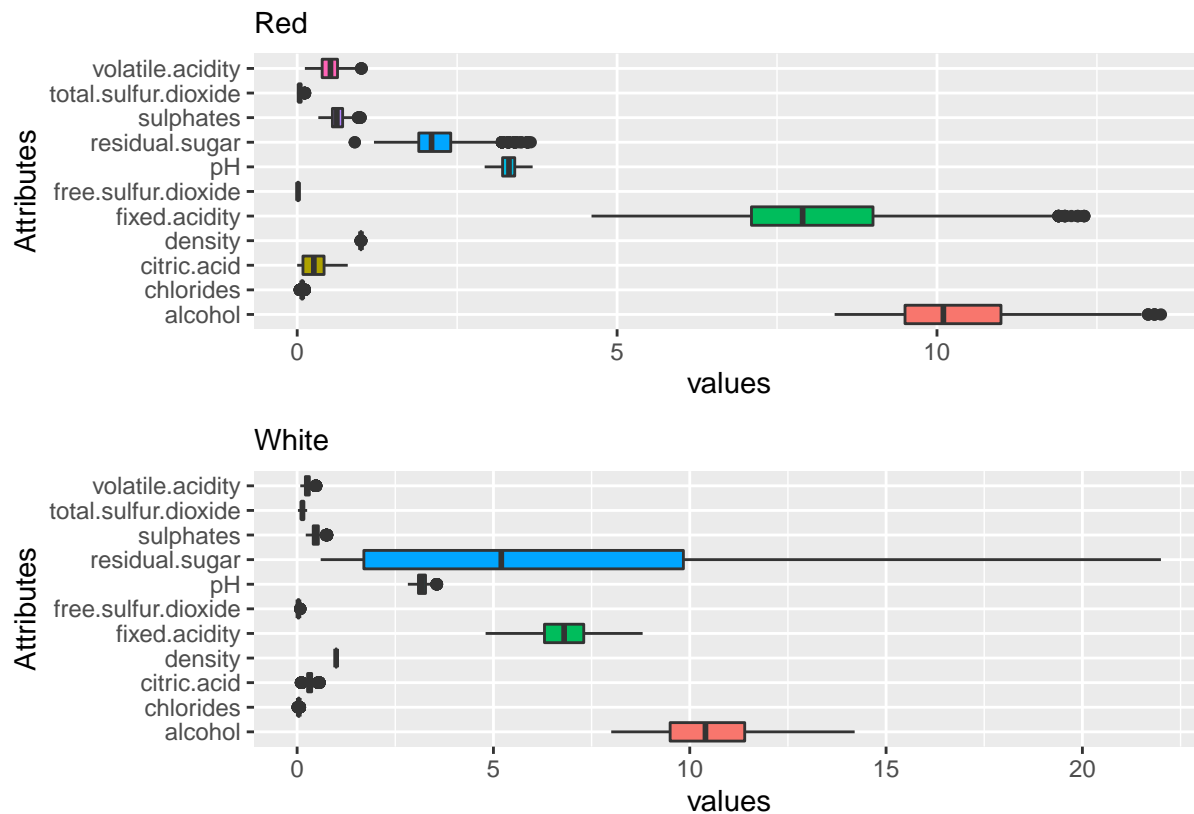
```
red[,c(1:11)] <- apply(red[,c(1:11)], 2, impute)
white[,c(1:11)] <- apply(white[,c(1:11)], 2, impute)
```

Visualizamos los boxplots de los atributos de nuevo:

```
bp1 <- red %>%
  gather(Attributes, values, c(1:11)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

bp2 <- white %>%
  gather(Attributes, values, c(1:11)) %>%
  ggplot(aes(x=Attributes, y=values, fill=Attributes)) + geom_boxplot(show.legend=FALSE) + coord_

bp1 + labs(subtitle = "Red") + bp2 + labs(subtitle = "White") + plot_layout(nrow = 2)
```



Aunque seguimos teniendo outliers en su definición más teórica, están más agrupados habiendo eliminado los valores muy anómalos y demasiado dispersos que podían ser errores o inconsistencias. Por ello, hemos obtenido la distribución mucho menos sesgada y mas representativa de variebilidad natural fisicoquímica.

Discretización

Como se observa, la variable quality no está balanceada, y las clases que tienen pocas observaciones pueden presentar problemas en análisis así que es conveniente crear particiones con más observaciones. Por ello con el fin de equilibrar la marca de calidad y agruparlo de manera natural, se puede realizar la discretización de la variable. Para poder trabajar posteriormente con una variable dicotómica de calidad, fijaremos el número de bins de 2, que representaria calidad alta/no alta.

Veamos de nuevo sus estadísticas:

```
summary(red$quality)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  3.000  5.000   6.000   5.636  6.000   8.000
```

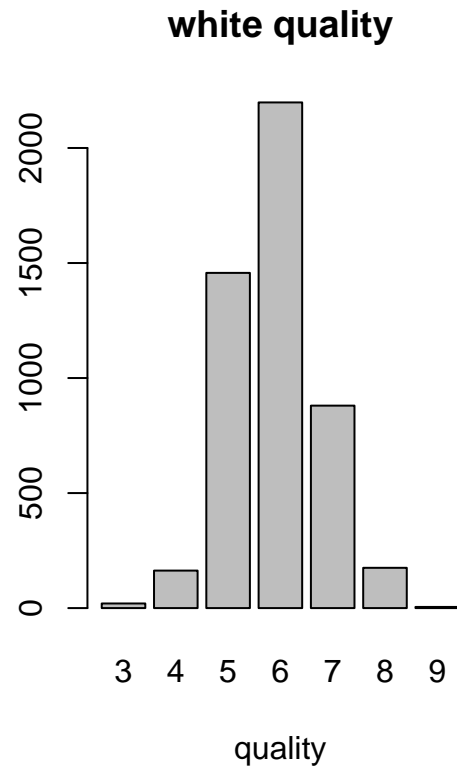
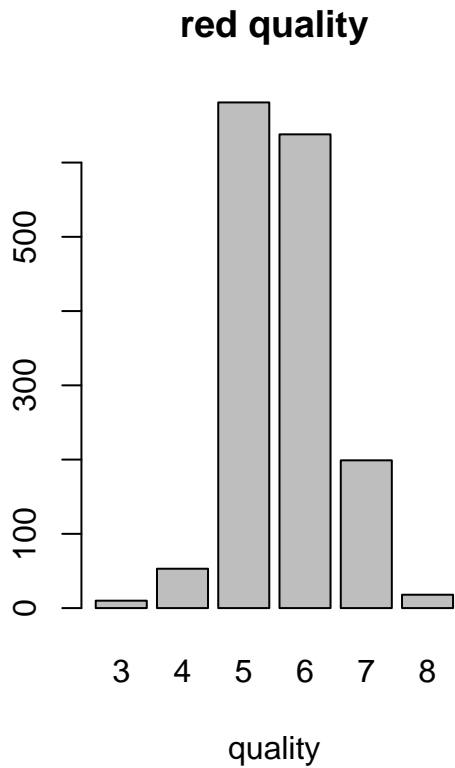
```
summary(white$quality)
```

```
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  3.000  5.000   6.000   5.878  6.000   9.000
```

```
par(mfrow=c(1,2))
```

```
barplot(table(red$quality), xlab="quality", main = "red quality")
```

```
barplot(table(white$quality), xlab="quality", main = "white quality")
```



```
# Distribución de valores únicos
table(red$quality)
```

```
##
##  3  4  5  6  7  8
## 10 53 681 638 199 18
```

```
table(white$quality)
```

```
##
##  3  4  5  6  7  8  9
## 20 163 1457 2198 880 175 5
```

Las marcas de clases en distintos vinos no estan igualmente distribuidas, sin embargo lo más lógico sería tener bins homogéneos para ambos vinos, teniendo así las clases de calidad consistentes.

Para poder determinar qué observaciones agrupamos en qué bins de calidad, visualizamos las clases discretas segun si la partición se hace por igual frecuencia / igual amplitud o clustering:

```
# red vines
par(mfrow=c(2,3))
set.seed(13)

hist(red$quality, breaks = 30, main = "red equal frequency")
abline(v = discretize(red$quality, breaks = 2, onlycuts = TRUE), col = "red")

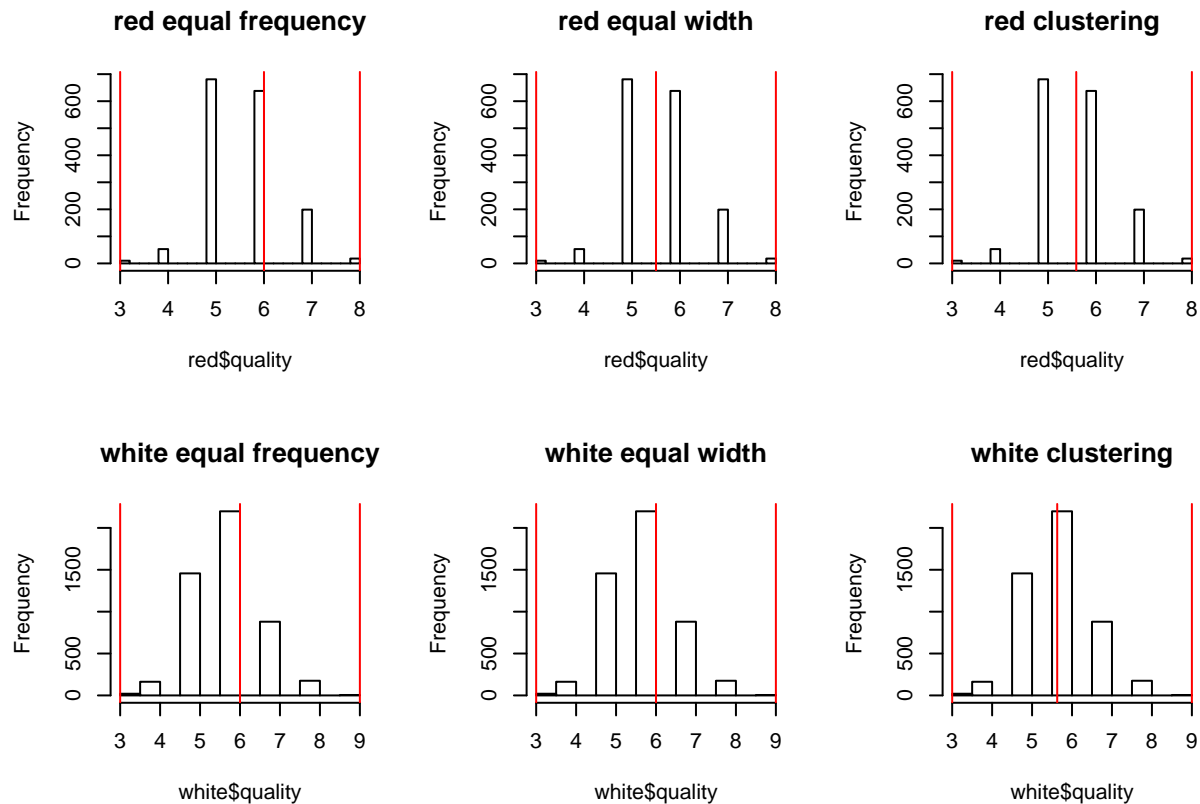
hist(red$quality, breaks = 30, main = "red equal width")
abline(v = discretize(red$quality, method = "interval", breaks = 2, onlycuts = TRUE), col = "red")

hist(red$quality, breaks = 30, main = "red clustering")
abline(v = discretize(red$quality, method = "cluster", breaks = 2, onlycuts = TRUE), col = "red")
```

```
# white vines
hist(white$quality, breaks = 20, main = "white equal frequency")
abline(v = discretize(white$quality, breaks = 2, onlycuts = TRUE), col = "red")

hist(white$quality, breaks = 20, main = "white equal width")
abline(v = discretize(white$quality, method = "interval", breaks = 2, onlycuts = TRUE), col = "red")

hist(white$quality, breaks = 20, main = "white clustering")
abline(v = discretize(white$quality, method = "cluster", breaks = 2, onlycuts = TRUE), col = "red")
```



Por la mayoría de intervalos, la calidad alta sería representada por vinos con calidad de 6 o mas, por ello creamos el atributo dicotómico correspondiente:

```
red$quality.class[red$quality<=5]="low"
red$quality.class[red$quality>5]="high"

white$quality.class[white$quality<=5]="low"
white$quality.class[white$quality>5]="high"

red$quality.class <- factor(red$quality.class, levels = c("low","high"))
white$quality.class <- factor(white$quality.class, levels = c("low","high"))

# Frecuencia de la calidad
table(red$quality.class)
```

```
##
## low high
## 744 855
```

```
table(white$quality.class)
```

```
##  
## low high  
## 1640 3258
```

Exportación de los datos preprocesados

```
write.csv(red, "red_clean.csv")  
write.csv(white, "white_clean.csv")
```

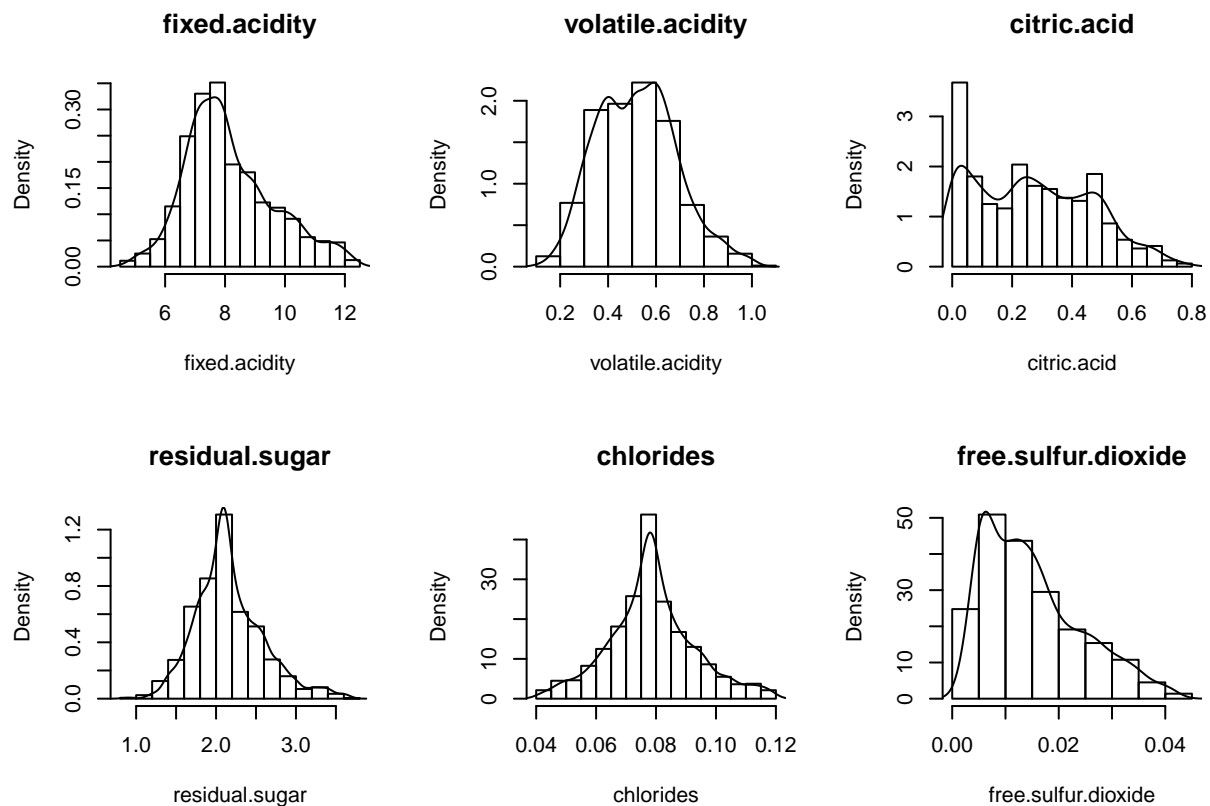
Análisis

Selección de los grupos de datos que se quieren analizar/comparar

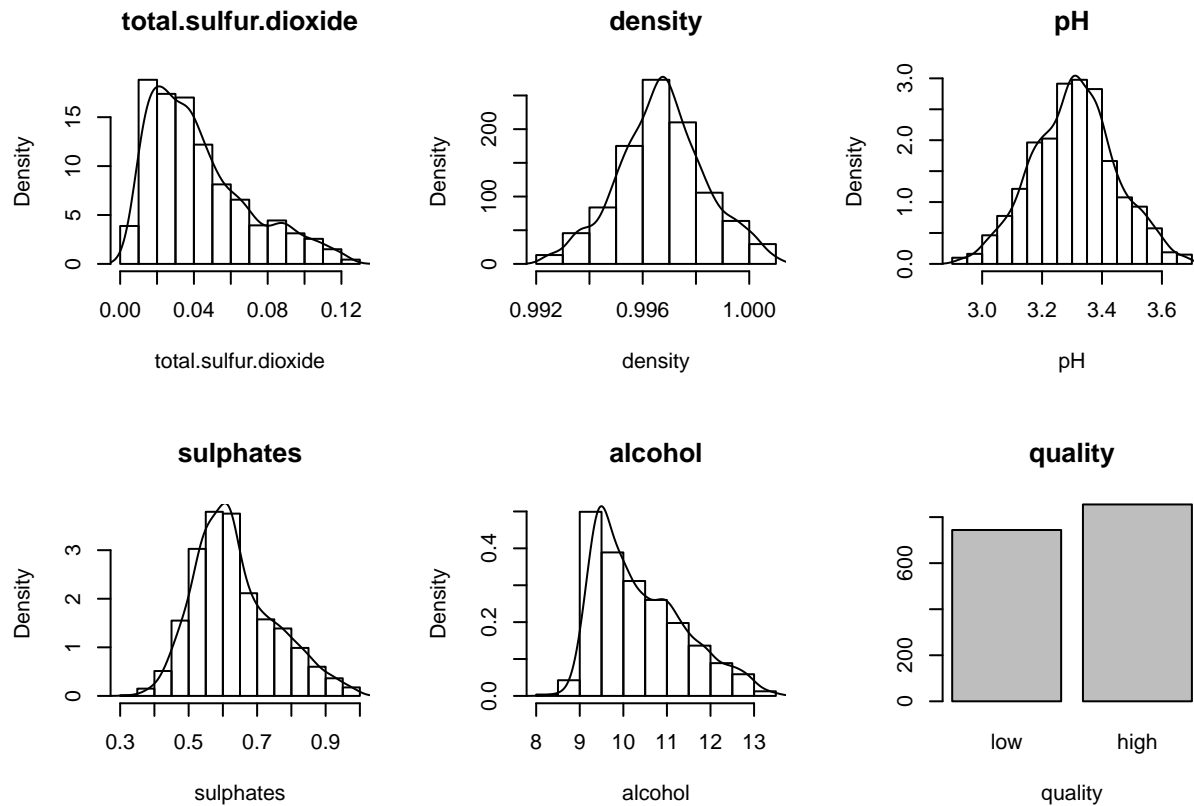
Seguimos trabajando con dos conjuntos de vinos - tintos y blancos, mientras cada uno tiene dos grupos de calidad, baja y alta, que sería el principal criterio de comparación.

Primero, realizamos un breve análisis exploratorio visual:

```
col <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar", "chlorides", "free.sulfur.dioxide")  
par(mfrow=c(2,3))  
  
for (name in col) {  
  hist(red[,name], prob=TRUE, xlab=name, main = name)  
  lines(density(na.omit(red[,name])))  
}
```

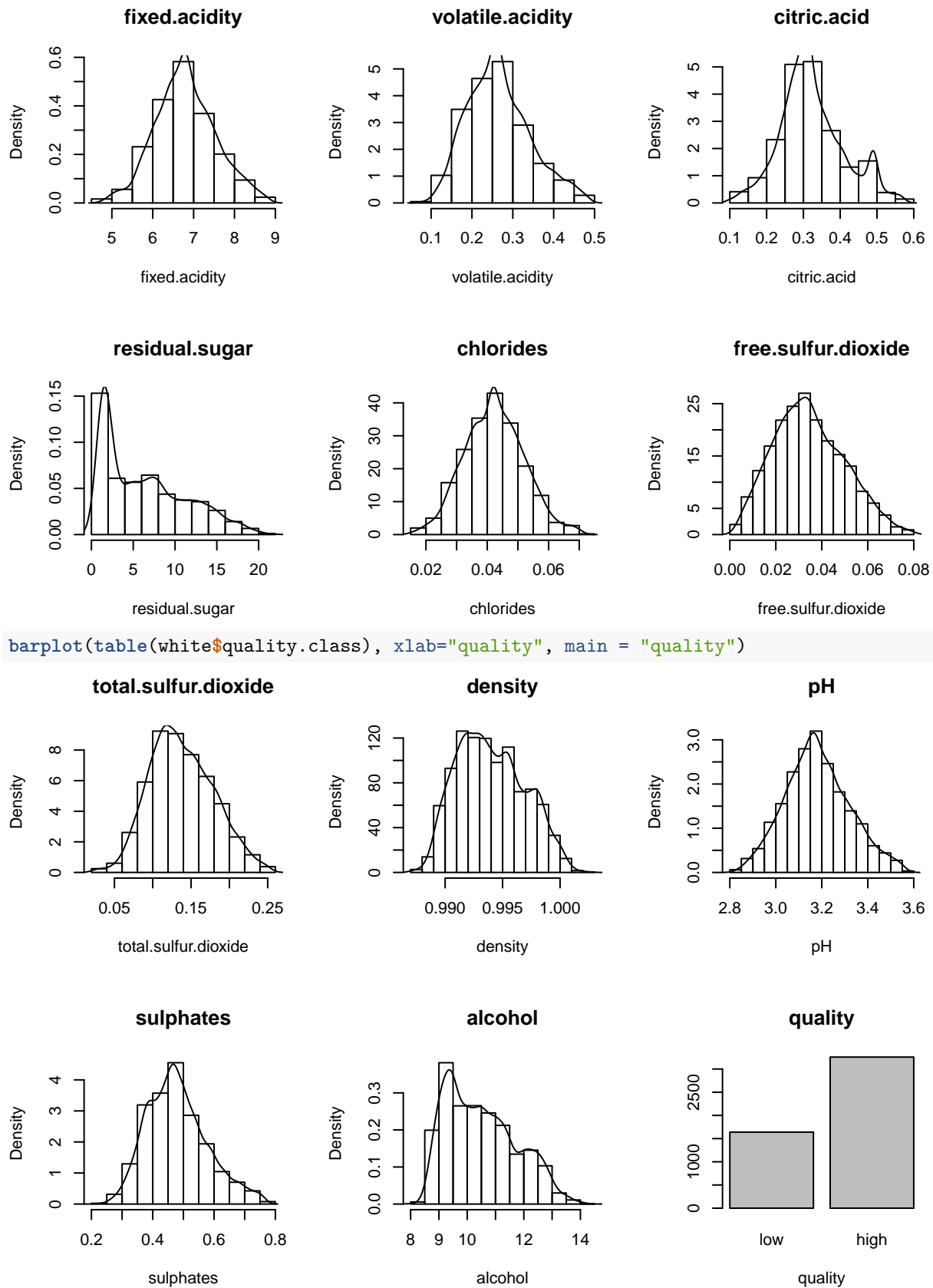


```
barplot(table(red$quality.class), xlab="quality", main = "quality")
```



```
col <- c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar", "chlorides", "free.sulfur.dioxide")
par(mfrow=c(2,3))

for (name in col) {
  hist(white[,name], prob=TRUE, xlab=name, main = name)
  lines(density(na.omit(white[,name])))
}
```

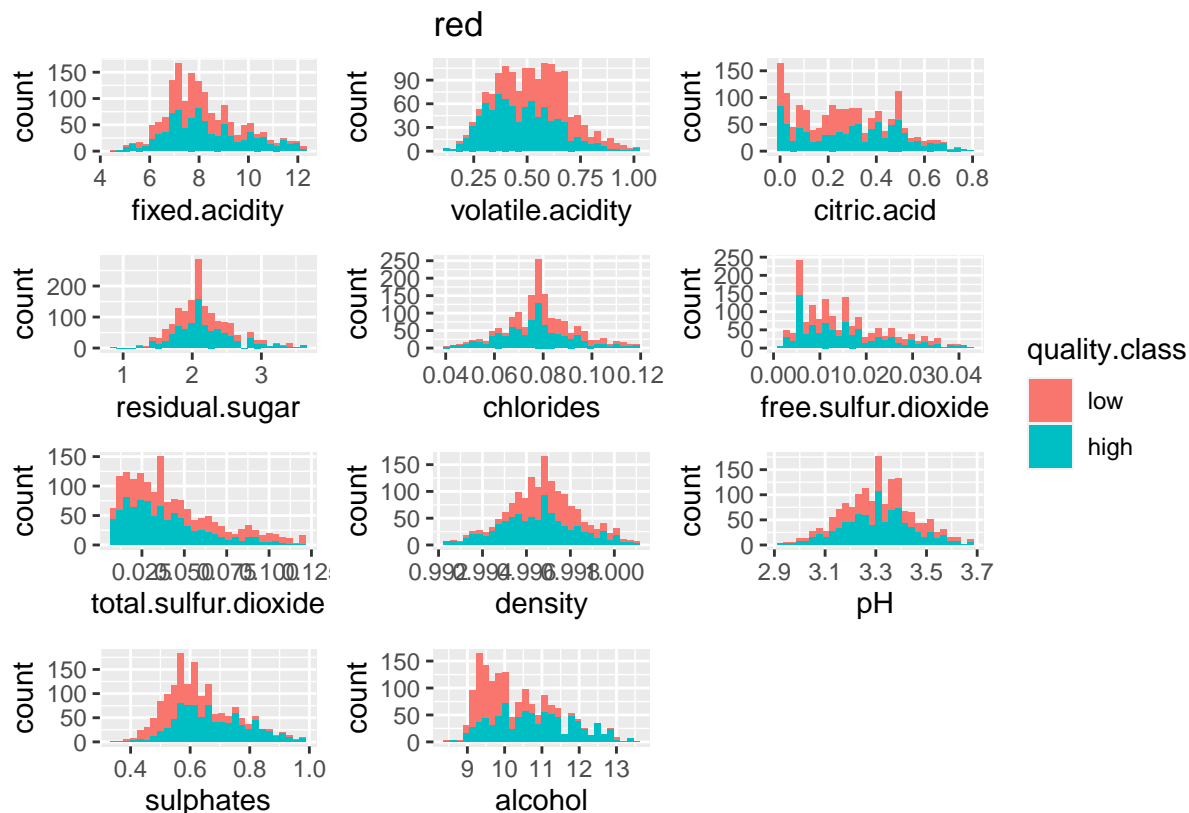
Se observa que la distribución de la mayoría de las variables para los dos datasets es relativamente normal,

aunque también hay variables cuya distribución es bastante sesgada y parece visualmente a la F-distribution, por ejemplo residual sugar de vinos blancos con una cola por la derecha.

También, se pueden visualizar los atributos separados por la marca de calidad:

```
p1 <- ggplot(data=red,aes(x=fixed.acidity,fill=quality.class))+geom_histogram()
p2 <- ggplot(data=red,aes(x=volatile.acidity,fill=quality.class))+geom_histogram()
p3 <- ggplot(data=red,aes(x=citric.acid,fill=quality.class))+geom_histogram()
p4 <- ggplot(data=red,aes(x=residual.sugar,fill=quality.class))+geom_histogram()
p5 <- ggplot(data=red,aes(x=chlorides,fill=quality.class))+geom_histogram()
p6 <- ggplot(data=red,aes(x=free.sulfur.dioxide,fill=quality.class))+geom_histogram()
p7 <- ggplot(data=red,aes(x=total.sulfur.dioxide,fill=quality.class))+geom_histogram()
p8 <- ggplot(data=red,aes(x=density,fill=quality.class))+geom_histogram()
p9 <- ggplot(data=red,aes(x=pH,fill=quality.class))+geom_histogram()
p10 <- ggplot(data=red,aes(x=sulphates,fill=quality.class))+geom_histogram()
p11 <- ggplot(data=red,aes(x=alcohol,fill=quality.class))+geom_histogram()
```

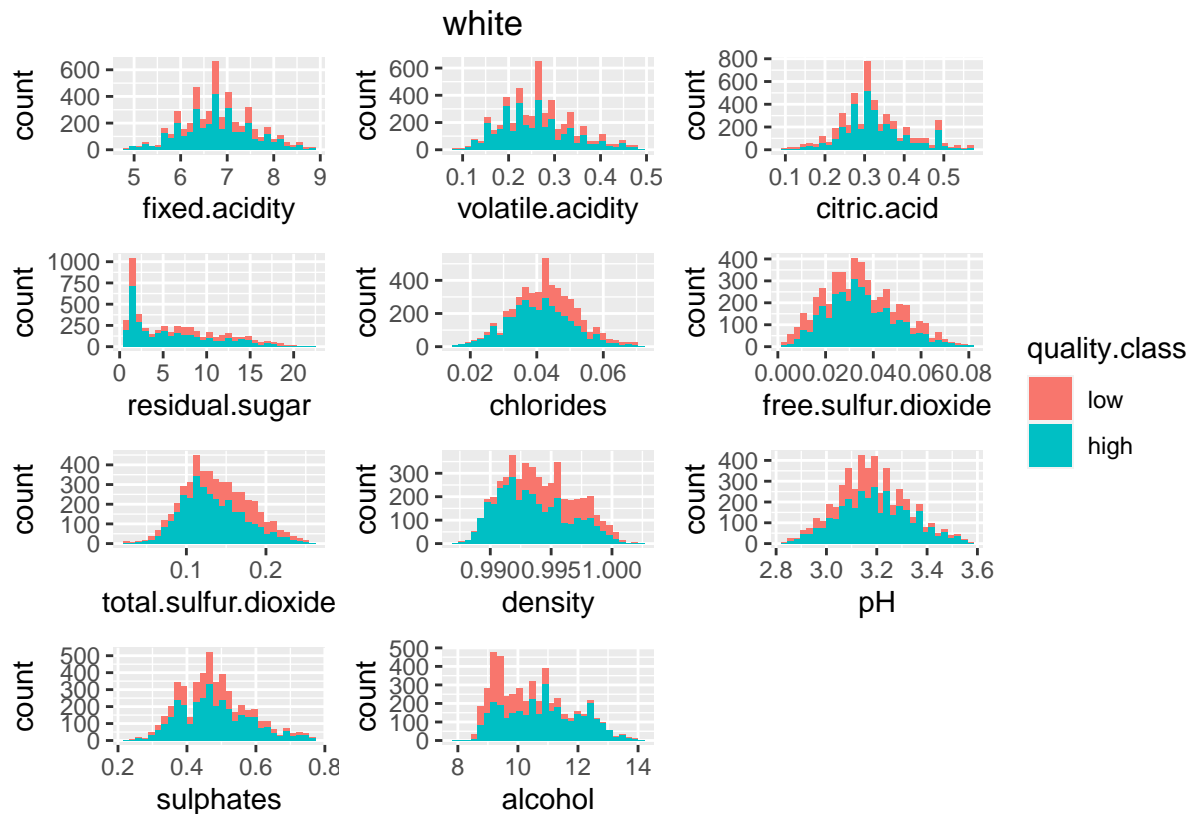
```
p1 + p2 + ggtitle("red") + p3 + p4 + p5 + p6 + p7 + p8 + p9 + p10 + p11 + plot_layout(ncol = 3) + plot_
```



```
p1 <- ggplot(data=white,aes(x=fixed.acidity,fill=quality.class))+geom_histogram()
p2 <- ggplot(data=white,aes(x=volatile.acidity,fill=quality.class))+geom_histogram()
p3 <- ggplot(data=white,aes(x=citric.acid,fill=quality.class))+geom_histogram()
p4 <- ggplot(data=white,aes(x=residual.sugar,fill=quality.class))+geom_histogram()
p5 <- ggplot(data=white,aes(x=chlorides,fill=quality.class))+geom_histogram()
p6 <- ggplot(data=white,aes(x=free.sulfur.dioxide,fill=quality.class))+geom_histogram()
p7 <- ggplot(data=white,aes(x=total.sulfur.dioxide,fill=quality.class))+geom_histogram()
p8 <- ggplot(data=white,aes(x=density,fill=quality.class))+geom_histogram()
p9 <- ggplot(data=white,aes(x=pH,fill=quality.class))+geom_histogram()
p10 <- ggplot(data=white,aes(x=sulphates,fill=quality.class))+geom_histogram()
```

```
p11 <- ggplot(data=white,aes(x=alcohol,fill=quality.class))+geom_histogram()
```

```
p1 + p2 + ggtitle("white") + p3 + p4 + p5 + p6 + p7 + p8 + p9 + p10 + p11 + plot_layout(ncol = 3) + plo
```



Se observan casi las mismas frecuencias para ambas calidades de los vinos, sin embargo algunos atributos se distribuyen de manera distinta - sobre todo el atributo alcohol, presente en los dos subconjuntos, que podría indicar una correlación del atributo con la calidad.

Comprobación de la normalidad y homogeneidad de la varianza

Para poder llevar a cabo un análisis inferencial y modelización predictiva, comprobamos la asunción de la normalidad y homoscedsticidad de los datos.

Para los tests de normalidad, usamos la prueba de normalidad de Shapiro-Wilk. Segun el nivel de significancia fijado a 0.05, aceptamos la hipótesis nula de normalidad si el p-value resultante es mayor al nivel de significancia, y rechazamos la hipótesis nula a favor de la alternativa si el p-value es menor de 0.05.

```
for (x in c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar", "chlorides", "free.su
  if (shapiro.test(red[,x])$p.value < 0.05) {
    cat("No sigue una distribución normal (tintos):",x,"\n")
  }
}
```

```
## No sigue una distribución normal (tintos): fixed.acidity
## No sigue una distribución normal (tintos): volatile.acidity
## No sigue una distribución normal (tintos): citric.acid
## No sigue una distribución normal (tintos): residual.sugar
## No sigue una distribución normal (tintos): chlorides
## No sigue una distribución normal (tintos): free.sulfur.dioxide
```

```
## No sigue una distribución normal (tintos): total.sulfur.dioxide
## No sigue una distribución normal (tintos): density
## No sigue una distribución normal (tintos): pH
## No sigue una distribución normal (tintos): sulphates
## No sigue una distribución normal (tintos): alcohol

for (x in c("fixed.acidity", "volatile.acidity", "citric.acid", "residual.sugar", "chlorides", "free.su
  if (shapiro.test(white[,x])$p.value < 0.05) {
    cat("No sigue una distribución normal (blancos):",x,"\n")
  }
}
```

```
## No sigue una distribución normal (blancos): fixed.acidity
## No sigue una distribución normal (blancos): volatile.acidity
## No sigue una distribución normal (blancos): citric.acid
## No sigue una distribución normal (blancos): residual.sugar
## No sigue una distribución normal (blancos): chlorides
## No sigue una distribución normal (blancos): free.sulfur.dioxide
## No sigue una distribución normal (blancos): total.sulfur.dioxide
## No sigue una distribución normal (blancos): density
## No sigue una distribución normal (blancos): pH
## No sigue una distribución normal (blancos): sulphates
## No sigue una distribución normal (blancos): alcohol
```

Hay evidencia que ninguna de las variables sigue una distribución normal según el test Shapiro-Wilk ya que no podemos aceptar la hipótesis nula de normalidad de distribución. Sin embargo, para lidiar con ello, por el teorema de límite central, teniendo un número suficiente de observaciones (1599 y 4898), podemos asumir la normalidad para ambos datasets.

Para comprobar la homocedasticidad de las variables, se usa el F-test, que aplicamos a la variable numérica alcohol de ambos datasets, que, tal y como hemos visto, puede ser bastante significativa para los modelos posteriores:

```
var.test(alcohol~quality.class, red, alternative = "two.sided")
```

```
##
## F test to compare two variances
##
## data: alcohol by quality.class
## F = 0.49039, num df = 743, denom df = 854, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.4268325 0.5638033
## sample estimates:
## ratio of variances
## 0.4903932
```

```
var.test(alcohol~quality.class, white, alternative = "two.sided")
```

```
##
## F test to compare two variances
##
## data: alcohol by quality.class
## F = 0.49345, num df = 1639, denom df = 3257, p-value < 2.2e-16
## alternative hypothesis: true ratio of variances is not equal to 1
## 95 percent confidence interval:
## 0.4539728 0.5369855
```

```
## sample estimates:
## ratio of variances
##          0.4934461
```

De la misma manera, por el p-value menor que el nivel de significancia, no podemos aceptar la hipótesis de homocedasticidad de alcohol en distintas calidades en vinos blancos y tintos. La proporción real de las varianzas en los grupos por calidad para ambos vinos con confianza de 95% es de aproximadamente 0.5.

Estudio de correlación

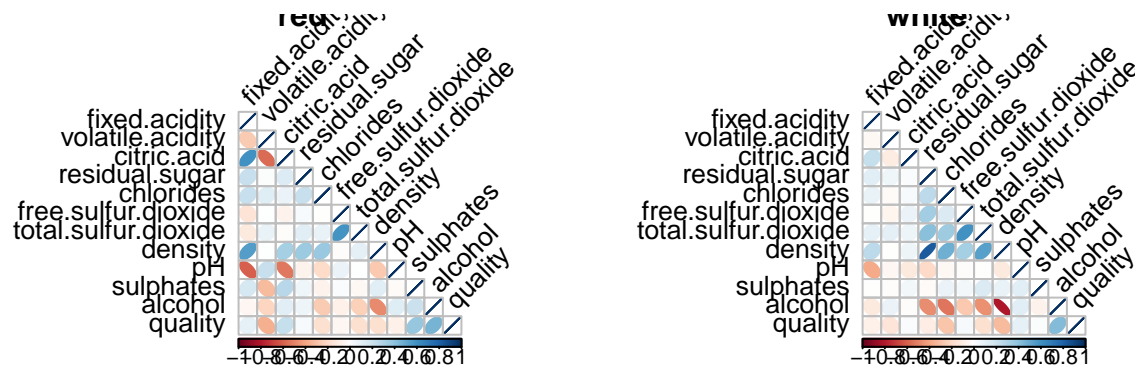
Para poder explicar las relaciones entre las variables para cada tipo de vino y sus clases de calidad y ver el grado de influencia, podemos visualizar los correlogramas:

```
r <- red[1:11]
r$quality = as.numeric(red$quality)

w <- white[1:11]
w$quality = as.numeric(white$quality)

M1<-cor(r)
M2<-cor(w)

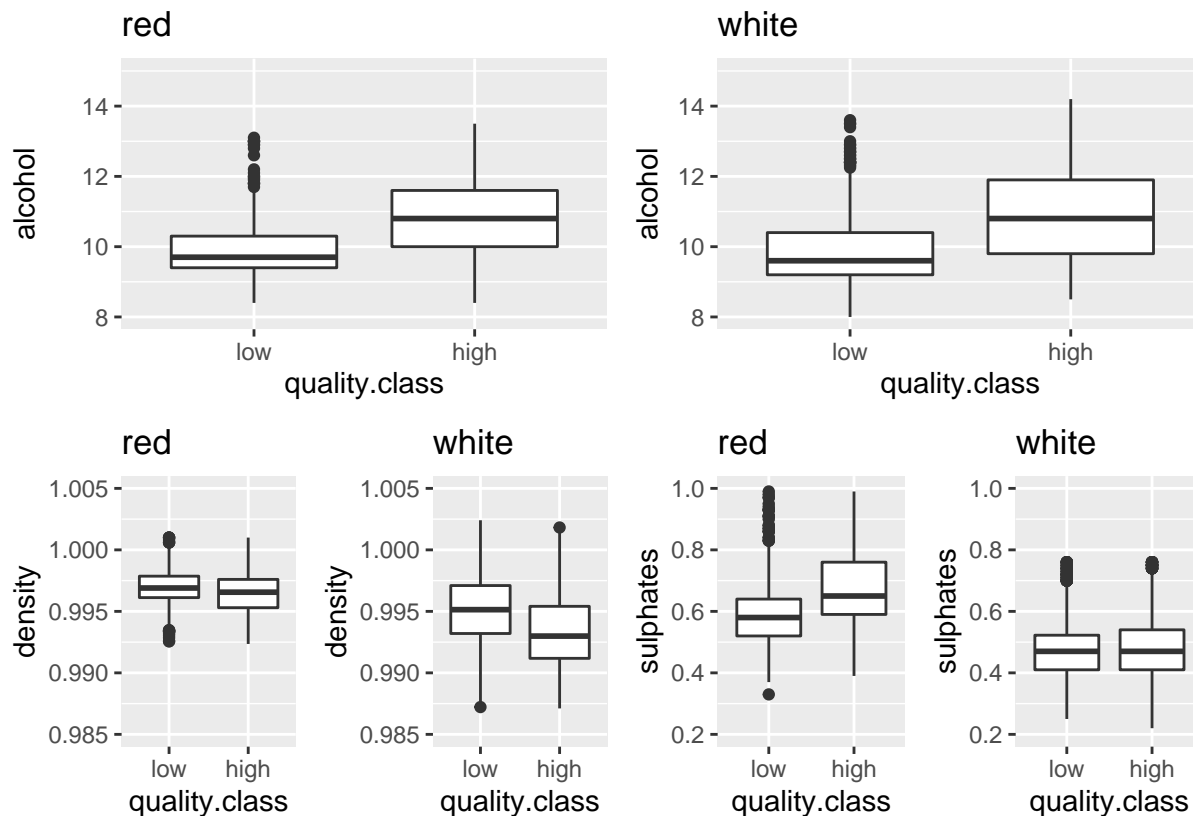
par(mfrow=c(2,2))
corrplot(M1, method="ellipse", type='lower', tl.col="black", tl.srt=45, title='red')
corrplot(M2, method="ellipse", type='lower', tl.col="black", tl.srt=45, title='white')
```



Se observa que los vinos tintos y blancos tienen correlaciones distintas tanto entre sus atributos, como con la calidad. Por ejemplo, alcohol influye en la calidad tanto en vinos tintos como en blancos. Sin embargo la calidad de los tintos también está correlacionada con sulphates y con volatile acidity, mientras la calidad de los blancos - con density y chlorides. En cuanto a los atributos de los vinos, muchas de las correlaciones son parecidas (parejas alcohol-density, pH-fixed acidity), pero se puede concluir que en general las relaciones entre las características de los vinos son distintas para los dos tipos.

Como la finalidad del estudio es intentar perfilar y predecir la calidad de los vinos, visualizamos las correlaciones más significativas con calidad, primero la correlación presente en ambos vinos, y posteriormente las correlaciones propias de cada uno de los tipos:

```
p1 <- ggplot(data = red, aes(x=quality.class, y=alcohol)) + geom_boxplot() + ggtitle("red") + ylim(8, 15)
p2 <- ggplot(data = red, aes(x=quality.class, y=density)) + geom_boxplot() + ggtitle("red") + ylim(0.985, 1)
p3 <- ggplot(data = red, aes(x=quality.class, y=sulphates)) + geom_boxplot() + ggtitle("red") + ylim(0.2, 0.4)
p1+p2+p3
```



Observamos la correlación positiva con alcohol en ambos tipos de vinos, y comprobamos la diferencia en correlación de la calidad en vinos: la densidad como la característica con una influencia más fuerte en vinos blancos y sulphates - en vinos tintos.

Análisis inferencial

Teniendo las muestras lo suficientemente grandes, podemos realizar los test paramétricos sobre los datos. La marca clase es discreta, por ello uno de los tests más significativos sería un test sobre la proporción de vinos de alta calidad en vinos blancos y tintos, respondiendo así la pregunta si el color y la calidad son independientes (con los datos disponibles, que, como sabemos, presentan un sesgo de marca de clase).

Contraste de hipótesis de dos muestras sobre la proporción de vinos de alta calidad según el color

Para el test, las hipótesis son:

hipótesis nula: las proporciones de alta calidad en dos muestras son iguales ($p_R = p_W$)

hipótesis alternativa: las proporciones de alta calidad en dos muestras no son iguales ($p_R \neq p_W$)

Por ello, es un contraste bilateral, y fijamos el nivel de significancia en 95%.

```
pR <- dim(red[red$quality.class=='high',])[1]/dim(red)[1]
pW <- dim(white[white$quality.class=='high',])[1]/dim(white)[1]
nR <- dim(red)[1]
nW <- dim(white)[1]

success<-c(pR*nR, pW*nW) # vector de casos de "éxito"
nn<-c(nR,nW) # vector de tamaño de muestras
prop.test(success, nn, alternative="two.sided", conf.level=0.95, correct=FALSE)
```

```
##
## 2-sample test for equality of proportions without continuity
## correction
##
## data: success out of nn
## X-squared = 88.323, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
## -0.1582521 -0.1026684
## sample estimates:
## prop 1 prop 2
## 0.5347092 0.6651695
```

Por el p-value no podemos aceptar la hipótesis nula de igualdad de proporciones, por ello concluimos que las proporciones de vinos de alta calidad son distintos para vinos de distinto color, siendo el color blanco el que tiende a tener calidad más alta en nuestro conjunto de datos.

Modelización predictiva

Uno de los objetivos del estudio ha sido poder realizar predicciones sobre la calidad, en este apartado se intentará crear y comparar clasificadores (modelos supervisados) lineales y no lineales. No sabemos qué tipo de frontera de decisión funcionará mejor sobre los datasets, por ello se prueban dos algoritmos: el primer modelo que se creará es un modelo lineal de regresión con regresores múltiples; luego, el modelo no lineal será un random forest, que como un algoritmo de bagging permite obtener modelos robustos.

Para poder evaluar la precisión de predicción y obtener las matrices de confusión, separamos los datasets en conjuntos de train y test:

```
set.seed(13)

split = sample.split(red$quality, SplitRatio = 0.8)
train_red = subset(red[c(1:11,13)],split == TRUE)
test_red = subset(red[c(1:11,13)],split == FALSE)

split = sample.split(white$quality, SplitRatio = 0.8)
train_white = subset(white[c(1:11,13)],split == TRUE)
test_white = subset(white[c(1:11,13)],split == FALSE)
```

Regresión logística

Como la variable respuesta es binaria, el modelo de regresión es la regresión logística con atributos cuantitativos. Intentamos explicar la calidad con todos los atributos del dataset, que posteriormente podemos eliminar del modelo si no son significativos (según el p-value obtenido del algoritmo).

Modelo para vinos tintos:

```
red_glm <- glm(quality.class ~ ., train_red, family=binomial(link=logit))
summary(red_glm)
```

```
##
## Call:
## glm(formula = quality.class ~ ., family = binomial(link = logit),
## data = train_red)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.8231 -0.8228 0.2796 0.7977 2.3556
```

```
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -12.06119   72.75307  -0.166  0.86833
## fixed.acidity     0.08418    0.07726   1.090  0.27588
## volatile.acidity  -3.39870    0.56778  -5.986 2.15e-09 ***
## citric.acid      -1.69315    0.56718  -2.985  0.00283 **
## residual.sugar     0.18912    0.18622   1.016  0.30982
## chlorides        -11.18006    5.33976  -2.094  0.03628 *
## free.sulfur.dioxide 10.05950    9.63008   1.045  0.29621
## total.sulfur.dioxide -10.09750    3.50720  -2.879  0.00399 **
## density           6.77843   73.63242   0.092  0.92665
## pH               -1.58082    0.68291  -2.315  0.02062 *
## sulphates         5.75564    0.69801   8.246 < 2e-16 ***
## alcohol           0.90596    0.10170   8.908 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 1765.6  on 1277  degrees of freedom
## Residual deviance: 1303.0  on 1266  degrees of freedom
## AIC: 1327
##
## Number of Fisher Scoring iterations: 4
```

Según el modelo, solo la mitad de las variables son estadísticamente significativas, asimismo lo podemos precisar:

```
red_glm <- glm(quality.class~volatile.acidity+citric.acid+total.sulfur.dioxide+pH+sulphates+alcohol, tra
summary(red_glm)
```

```
##
## Call:
## glm(formula = quality.class ~ volatile.acidity + citric.acid +
##      total.sulfur.dioxide + pH + sulphates + alcohol, family = binomial(link = logit),
##      data = train_red)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.6820  -0.8282   0.2878   0.7988   2.4395
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.79386    2.03807  -2.352  0.01866 *
## volatile.acidity  -3.44089    0.54156  -6.354 2.1e-10 ***
## citric.acid      -1.45928    0.50128  -2.911  0.00360 **
## total.sulfur.dioxide -8.08318    2.70612  -2.987  0.00282 **
## pH               -1.78363    0.59427  -3.001  0.00269 **
## sulphates         5.85740    0.67717   8.650 < 2e-16 ***
## alcohol           0.93799    0.08285  11.322 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
```



```
##
## Null deviance: 1765.6 on 1277 degrees of freedom
## Residual deviance: 1310.7 on 1271 degrees of freedom
## AIC: 1324.7
##
## Number of Fisher Scoring iterations: 4
```

Para ver los odds ratio para cada unidad de las características:

```
exp(coefficients(red_glm))
```

```
## (Intercept) volatile.acidity citric.acid
## 8.280448e-03 3.203627e-02 2.324042e-01
## total.sulfur.dioxide pH sulphates
## 3.086880e-04 1.680267e-01 3.498131e+02
## alcohol
## 2.554845e+00
```

Puesto que las características tienen distintas escalas de valores, podemos considerar alcohol y sulphates los más influyentes a la probabilidad en el modelo obtenido, lo que coincide con las correlaciones.

La bondad de ajuste se obtiene a través del índice de Akaike AIC, que en este caso asciende a 1325.

Modelo vinos blancos:

```
white_glm <- glm(quality.class ~ ., train_white, family=binomial(link=logit))
summary(white_glm)
```

```
##
## Call:
## glm(formula = quality.class ~ ., family = binomial(link = logit),
## data = train_white)
##
## Deviance Residuals:
## Min 1Q Median 3Q Max
## -2.9479 -0.8891 0.4396 0.7949 2.4265
##
## Coefficients:
## Estimate Std. Error z value Pr(>|z|)
## (Intercept) 389.16236 64.88839 5.997 2.00e-09 ***
## fixed.acidity 0.12311 0.07185 1.713 0.0866 .
## volatile.acidity -7.24654 0.55698 -13.010 < 2e-16 ***
## citric.acid 0.80351 0.45466 1.767 0.0772 .
## residual.sugar 0.21370 0.02448 8.729 < 2e-16 ***
## chlorides -6.89491 4.75835 -1.449 0.1473
## free.sulfur.dioxide 16.60295 3.33197 4.983 6.26e-07 ***
## total.sulfur.dioxide -1.34440 1.35952 -0.989 0.3227
## density -403.65209 65.59975 -6.153 7.59e-10 ***
## pH 1.74891 0.35807 4.884 1.04e-06 ***
## sulphates 1.95131 0.44516 4.383 1.17e-05 ***
## alcohol 0.55936 0.08836 6.330 2.45e-10 ***
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 4996.0 on 3917 degrees of freedom
```

```
## Residual deviance: 3956.1 on 3906 degrees of freedom
## AIC: 3980.1
##
## Number of Fisher Scoring iterations: 5
```

De la misma manera, se puede precisar el modelo:

```
white_glm <- glm(quality.class~volatile.acidity+residual.sugar+free.sulfur.dioxide+density+pH+sulphates
summary(white_glm)
```

```
##
## Call:
## glm(formula = quality.class ~ volatile.acidity + residual.sugar +
##     free.sulfur.dioxide + density + pH + sulphates + alcohol,
##     family = binomial(link = logit), data = train_white)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -3.0612  -0.9030   0.4362   0.8040   2.3370
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    329.89055    49.43848   6.673 2.51e-11 ***
## volatile.acidity    -7.54165     0.54068  -13.949 < 2e-16 ***
## residual.sugar      0.19183     0.01987   9.654 < 2e-16 ***
## free.sulfur.dioxide  15.02143     2.73269   5.497 3.86e-08 ***
## density          -342.96174    49.50439  -6.928 4.27e-12 ***
## pH                1.32512     0.30151   4.395 1.11e-05 ***
## sulphates         1.78029     0.43662   4.077 4.55e-05 ***
## alcohol           0.68019     0.07319   9.294 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 4996.0 on 3917 degrees of freedom
## Residual deviance: 3966.4 on 3910 degrees of freedom
## AIC: 3982.4
##
## Number of Fisher Scoring iterations: 5
```

Para los vinos blancos, obtenemos un modelo con el índice AIC de 3996 que es mucho mas alto que en el modelo anterior, por ello el ajuste debe de ser peor.

Veamos los odds ratio para cada unidad de las características:

```
exp(coefficients(white_glm))
```

```
##      (Intercept)    volatile.acidity    residual.sugar
##      1.860576e+143    5.305220e-04    1.211461e+00
## free.sulfur.dioxide    density    pH
##      3.339829e+06    1.131377e-149    3.762651e+00
##      sulphates    alcohol
##      5.931599e+00    1.974247e+00
```

Para los vinos blancos, density parece tener mayor importancia para la probabilidad de clase, tal y como se ha observado en correlaciones.

Sin embargo, la bondad de ajuste (por el AIC) no es muy buena, también podemos mirar las matrices de confusión y obtener precisión de los modelos:

Vinos tintos:

```
confusion_matrix(red_glm,test_red)
```

```
##           Predicted low Predicted high Total
## Actual low           43           106    149
## Actual high          120           52    172
## Total                163          158    321
```

Por ello, tanto la exactitud del modelo ($(43+51)/321$), como la sensibilidad ($51/172$) y la especificidad ($43/149$) son de 29%, un valor por debajo de 50% como un umbral de predicciones aleatorias.

Vinos blancos:

```
confusion_matrix(white_glm,test_white)
```

```
##           Predicted low Predicted high Total
## Actual low          163          165    328
## Actual high         574           78    652
## Total               737          243    980
```

Para los vinos blancos, la exactitud del modelo es de 25% con sensibilidad de 12% y especificidad de 50%, que significa que puede clasificar mejor los vinos de baja calidad, pero la bondad de ajuste sigue siendo baja.

Podemos concluir que no es trivial encontrar un modelo lineal multiple que explique la varianza de los atributos de los vinos para la probabilidad de la calidad, entonces que el dataset no es linealmente separable.

Modelo supervisado Random Forest

Para probar modelo no lineal, se ha elegido el algoritmo random forest que combina varios árboles de decisión con la técnica de muestreo subaleatorio (algoritmo bagging), lo que suele crear modelos que puedan generalizar bien sobre datos nuevos y tener un buen bias-variance tradeoff.

Modelo vinos tintos:

```
red_randomforest <- randomForest(quality.class ~ ., data=train_red)
red_randomforest
```

```
##
## Call:
## randomForest(formula = quality.class ~ ., data = train_red)
##           Type of random forest: classification
##           Number of trees: 500
## No. of variables tried at each split: 3
##
## OOB estimate of error rate: 18.94%
## Confusion matrix:
##           low high class.error
## low  478  117  0.1966387
## high 125  558  0.1830161
```

Modelo vinos blancos:

```
white_randomforest <- randomForest(quality.class ~ ., data=train_white)
white_randomforest
```

```
##
## Call:
```

```
## randomForest(formula = quality.class ~ ., data = train_white)
##           Type of random forest: classification
##           Number of trees: 500
## No. of variables tried at each split: 3
##
##           OOB estimate of  error rate: 16.51%
## Confusion matrix:
##           low high class.error
## low  917  395  0.30106707
## high 252 2354  0.09669992
```

En ambos modelos el out-of-bag error es bastante bajo, de 19% y 17%, que indica la alta capacidad de predicción puesto que es el error promedio de cada árbol en datos nuevos (no la muestra usada para entrenamiento de cada uno). Sin embargo, ya que tenemos el el conjunto de prueba, podemos hacer otra validación y obtener matrices de confusión:

```
pred_r <- predict(red_randomforest, test_red)

confusionMatrix(factor(pred_r),factor(test_red[,12]), positive = "high")
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction low high
##           low 123  32
##           high  26 140
##
##           Accuracy : 0.8193
##           95% CI : (0.7728, 0.8598)
##           No Information Rate : 0.5358
##           P-Value [Acc > NIR] : <2e-16
##
##           Kappa : 0.6377
##
## Mcnemar's Test P-Value : 0.5115
##
##           Sensitivity : 0.8140
##           Specificity : 0.8255
##           Pos Pred Value : 0.8434
##           Neg Pred Value : 0.7935
##           Prevalence : 0.5358
##           Detection Rate : 0.4361
##           Detection Prevalence : 0.5171
##           Balanced Accuracy : 0.8197
##
##           'Positive' Class : high
##
```

Tanto exactitud, como sensibilidad y especificidad del modelo de vinos tintos es de 81-82% (coincidiendo con out-of-bag error de 19%) que indica la alta capacidad de predicción, además de robustez y buena generalización del modelo puesto que ambas calidades se predicen igual de bien.

```
pred_w <- predict(white_randomforest, test_white)

confusionMatrix( factor(pred_w) , factor(test_white[,12]) , positive = "high")
```

```

## Confusion Matrix and Statistics
##
##           Reference
## Prediction low high
##      low  231   56
##      high  97  596
##
##           Accuracy : 0.8439
##           95% CI : (0.8196, 0.8661)
##      No Information Rate : 0.6653
##      P-Value [Acc > NIR] : < 2.2e-16
##
##           Kappa : 0.6382
##
##  Mcnemar's Test P-Value : 0.001222
##
##           Sensitivity : 0.9141
##           Specificity : 0.7043
##      Pos Pred Value : 0.8600
##      Neg Pred Value : 0.8049
##           Prevalence : 0.6653
##      Detection Rate : 0.6082
##      Detection Prevalence : 0.7071
##      Balanced Accuracy : 0.8092
##
##      'Positive' Class : high
##

```

Para los vinos blancos, la exactitud es aún mas alta con 84%, sensibilidad es de 91%, lo que puede predecir la alta calidad exelentemente, mientras la especificidad es un poco más baja con 70%. Como sabemos, en vinos blancos hay un sesgo en la marca de clase, lo que produce la diferencia en la predicción de las clases -si tuvieramos más datos para vinos blancos con calidad mas balanceada, el modelo posiblemente podría demostrar mejor rendimiento.

Para concluir, el modelo random forest ha obtenido una alta precisión en ambos tipos de vinos siendo el modelo preferible para predecir la calidad sensorial en función de las características físicoquímicas.

Conclusión; Resolución del problema.

A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?

Hemos trabajado con los dos subconjuntos de vinos realizando varios análisis explicativos o inferenciales (correlaciones, contraste de hipotesis) y creando modelos de clasificación (regresión logística y random forest) con el fin de poder perfilar y predecir la calidad de vinos tanto tintos como blancos, tal y como hemos predetminado con el objetivo inicial del estudio. Puesto que los datos has sido previamente limpiados, los procesos de limpieza y preprocesamiento han consistido en imputación de outliers, cambios de unidades de medida para conseguir una consistencia entre las variables y discretización de marca de clase de calidad convirtiendola en una variable dicotómica, que ha permitido agilizar los análisis y modelización posteriores.

Los resultados obtenidos nos demuestran que la calidad en vinos tintos y blancos no es estadísticamente igual. Los atributos mas explicativos y/o correlacionados con la calidad son distintos (i.e density para vinos blancos y sulphates para los tintos), aunque hay similitudes, como la influencia de alcohol en la calidad de ambos tipos de vinos. Para la clasificacion, el mejor algoritmo predictivo ha sido random forest con una alta precisión por encima de 80%, puesto que los datasets parecen no ser linealmente separables y el modelo

de regresión no ha podido explicar la calidad sensorial en función de los atributos físicoquímicos. Por lo tanto, el estudio nos ha permitido tanto explicar la calidad de distintos tipos de vinos (como intuición, mayor calidad tendrán los vinos blancos con más alcohol y menos densidad), y permitir predecir la calidad de manera satisfactoria, que puede ser usado tanto por los productores, como los consumidores.

```
data.frame("Contribuciones"=c("Investigación previa","Redacción de las respuestas","Desarrollo código"))
```

##	Contribuciones	Firma
## 1	Investigación previa	D.G.,Z.J.
## 2	Redacción de las respuestas	D.G.,Z.J.
## 3	Desarrollo código	D.G.,Z.J.