Tipología y ciclo de vida de los datos: PRA2

Autor: Iván López-Baltasar Benito / David Quiles Gómez

Junio 2019

Contents

Introducción

Presentación

En esta actividad se elabora un caso práctico, consistente en el tratamiento de un conjunto de datos (en inglés, dataset), orientado a aprender a identificar los datos relevantes para un proyecto analítico y usar las herramientas de integración, limpieza, validación y análisis de las mismas.

Objetivos

- Aprender a aplicar los conocimientos adquiridos y su capacidad de resolución de problemas en entornos nuevos o poco conocidos dentro de contextos más amplios o multidisciplinares.

- Saber identificar los datos relevantes y los tratamientos necesarios (integración, limpieza y validación) para llevar a cabo un proyecto analítico.

- Aprender a analizar los datos adecuadamente para abordar la información contenida en los datos.
- Identificar la mejor representación de los resultados para aportar conclusiones sobre el problema planteado en el proceso analítico.

- Actuar con los principios éticos y legales relacionados con la manipulación de datos en función del ámbito de aplicación.

- Desarrollar las habilidades de aprendizaje que les permitan continuar estudiando de un modo que tendrá que ser en gran medida autodirigido o autónomo.

- Desarrollar la capacidad de búsqueda, gestión y uso de información y recursos en el ámbito de la ciencia de datos.

Competencias

- Capacidad de analizar un problema en el nivel de abstracción adecuado a cada situación y aplicar las habilidades y conocimientos adquiridos para abordarlo y resolverlo.
- Capacidad para aplicar las técnicas específicas de tratamiento de datos (integración, transformación, limpieza y validación) para su posterior análisis

Descripcion del dataset

En ésta práctica vamos a trabajar con el juego de datos de https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/ el cual contiene dos datasets, uno de vinos blancos y otro de vinos tintos.

Ambos datasets contienen 11 atributos de entrada, correspondientes a pruebas fisioquímicas, y uno de salida: "quality".

El objetivo del análisis será por un lado construir un modelo que nos pueda predecir la calidad de un vino, y por otro, construir un modelo que nos permita clasificar un vino en un determinado tipo (blanco/tinto).

Carga y limpieza del dataset

Cargamos los paquetes R que vamos a usar

```
library(ggplot2)
library(dplyr)
```

```
blanco<-read.csv("vinos/winequality-white.csv", header=T, sep=";")
tinto<-read.csv("vinos/winequality-red.csv", header=T, sep=";")
```

Vamos a añadirle la clase a cada juego de datos para después unir ambos datasets.

```
blanco$tipo<-'B'
tinto$tipo<-'T'

nomCols <- c("acidez_fija", "acidez_volatil", "acido_citrico", "azucar_residual", "cloruros","diox_azuf

colnames(blanco) <- nomCols
colnames(tinto) <- nomCols

#str(blanco)
summary(blanco)
```

```
##   acidez_fija     acidez_volatil   acido_citrico    azucar_residual
##  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
##     cloruros       diox_azufre_libre diox_azufre_total    densidad
##  Min.   :0.00900   Min.   :  2.00    Min.   :  9.0     Min.   :0.9871
##  1st Qu.:0.03600   1st Qu.: 23.00    1st Qu.:108.0     1st Qu.:0.9917
##  Median :0.04300   Median : 34.00    Median :134.0     Median :0.9937
##  Mean   :0.04577   Mean   : 35.31    Mean   :138.4     Mean   :0.9940
##  3rd Qu.:0.05000   3rd Qu.: 46.00    3rd Qu.:167.0     3rd Qu.:0.9961
##  Max.   :0.34600   Max.   :289.00    Max.   :440.0     Max.   :1.0390
##        pH           sulfatos         alcohol         calidad
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   Min.   :3.000
##  1st Qu.:3.090   1st Qu.:0.4100   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.180   Median :0.4700   Median :10.40   Median :6.000
##  Mean   :3.188   Mean   :0.4898   Mean   :10.51   Mean   :5.878
##  3rd Qu.:3.280   3rd Qu.:0.5500   3rd Qu.:11.40   3rd Qu.:6.000
##  Max.   :3.820   Max.   :1.0800   Max.   :14.20   Max.   :9.000
##      tipo
##  Length:4898
##  Class :character
##  Mode  :character
##
##
##
```

```
#str(tinto)
summary(tinto)
```

```
##   acidez_fija    acidez_volatil   acido_citrico   azucar_residual
##  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
##     cloruros       diox_azufre_libre diox_azufre_total    densidad
##  Min.   :0.01200   Min.   : 1.00     Min.   :  6.00    Min.   :0.9901
##  1st Qu.:0.07000   1st Qu.: 7.00     1st Qu.: 22.00    1st Qu.:0.9956
##  Median :0.07900   Median :14.00     Median : 38.00    Median :0.9968
##  Mean   :0.08747   Mean   :15.87     Mean   : 46.47    Mean   :0.9967
##  3rd Qu.:0.09000   3rd Qu.:21.00     3rd Qu.: 62.00    3rd Qu.:0.9978
##  Max.   :0.61100   Max.   :72.00     Max.   :289.00    Max.   :1.0037
##        pH           sulfatos         alcohol         calidad
##  Min.   :2.740   Min.   :0.3300   Min.   : 8.40   Min.   :3.000
##  1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.310   Median :0.6200   Median :10.20   Median :6.000
##  Mean   :3.311   Mean   :0.6581   Mean   :10.42   Mean   :5.636
##  3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :8.000
##      tipo
##  Length:1599
##  Class :character
##  Mode  :character
##
```

##
##

Ahora unimos ambos datasets

```
# Unimos los dos juetos de datos en uno solo
totalData <- bind_rows(blanco,tinto)
filas=dim(totalData)[1]

# Factorizamos la variable tipo
totalData$tipo <- as.factor(totalData$tipo)

str(totalData)
```

```
## 'data.frame':    6497 obs. of  13 variables:
##  $ acidez_fija      : num  7 6.3 8.1 7.2 7.2 8.1 6.2 7 6.3 8.1 ...
##  $ acidez_volatil   : num  0.27 0.3 0.28 0.23 0.23 0.28 0.32 0.27 0.3 0.22 ...
##  $ acido_citrico    : num  0.36 0.34 0.4 0.32 0.32 0.4 0.16 0.36 0.34 0.43 ...
##  $ azucar_residual  : num  20.7 1.6 6.9 8.5 8.5 6.9 7 20.7 1.6 1.5 ...
##  $ cloruros         : num  0.045 0.049 0.05 0.058 0.058 0.05 0.045 0.045 0.049 0.044 ...
##  $ diox_azufre_libre: num  45 14 30 47 47 30 30 45 14 28 ...
##  $ diox_azufre_total: num  170 132 97 186 186 97 136 170 132 129 ...
##  $ densidad         : 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 ...
##  $ sulfatos         : 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 ...
##  $ calidad          : int  6 6 6 6 6 6 6 6 6 6 ...
##  $ tipo             : Factor w/ 2 levels "B","T": 1 1 1 1 1 1 1 1 1 1 ...
```

```
summary(totalData)
```

```
##   acidez_fija     acidez_volatil   acido_citrico    azucar_residual
##  Min.   : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
##  1st Qu.: 6.400   1st Qu.:0.2300   1st Qu.:0.2500   1st Qu.: 1.800
##  Median : 7.000   Median :0.2900   Median :0.3100   Median : 3.000
##  Mean   : 7.215   Mean   :0.3397   Mean   :0.3186   Mean   : 5.443
##  3rd Qu.: 7.700   3rd Qu.:0.4000   3rd Qu.:0.3900   3rd Qu.: 8.100
##  Max.   :15.900   Max.   :1.5800   Max.   :1.6600   Max.   :65.800
##     cloruros       diox_azufre_libre diox_azufre_total    densidad
##  Min.   :0.00900   Min.   :  1.00    Min.   :  6.0     Min.   :0.9871
##  1st Qu.:0.03800   1st Qu.: 17.00    1st Qu.: 77.0     1st Qu.:0.9923
##  Median :0.04700   Median : 29.00    Median :118.0     Median :0.9949
##  Mean   :0.05603   Mean   : 30.53    Mean   :115.7     Mean   :0.9947
##  3rd Qu.:0.06500   3rd Qu.: 41.00    3rd Qu.:156.0     3rd Qu.:0.9970
##  Max.   :0.61100   Max.   :289.00    Max.   :440.0     Max.   :1.0390
##        pH           sulfatos         alcohol         calidad      tipo
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   Min.   :3.000   B:4898
##  1st Qu.:3.110   1st Qu.:0.4300   1st Qu.: 9.50   1st Qu.:5.000   T:1599
##  Median :3.210   Median :0.5100   Median :10.30   Median :6.000
##  Mean   :3.219   Mean   :0.5313   Mean   :10.49   Mean   :5.818
##  3rd Qu.:3.320   3rd Qu.:0.6000   3rd Qu.:11.30   3rd Qu.:6.000
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :9.000
```

Nulos y/o elementos vacíos

Comprobamos que no haya valores vacíos o nulos.

```
# Estad??ssticas de valores vacios
colSums(is.na(totalData))
```

```
##       acidez_fija    acidez_volatil     acido_citrico   azucar_residual
##                 0                 0                 0                 0
##          cloruros diox_azufre_libre diox_azufre_total          densidad
##                 0                 0                 0                 0
##                pH          sulfatos           alcohol           calidad
##                 0                 0                 0                 0
##              tipo
##                 0
```

```
colSums(totalData=="")
```

```
##       acidez_fija    acidez_volatil     acido_citrico   azucar_residual
##                 0                 0                 0                 0
##          cloruros diox_azufre_libre diox_azufre_total          densidad
##                 0                 0                 0                 0
##                pH          sulfatos           alcohol           calidad
##                 0                 0                 0                 0
##              tipo
##                 0
```

Podemos ver como tenemos algunos valores a cero, en el atributo acido_citrico que además está presente tanto en los vinos tintos como en los blancos. Vamos a obtener cuantos valores a 0 tenemos y consultar con una fuente externa (https://www.aprenderdevino.es/acidos-acidez-vino/) si este valor representa un error o es un valor correcto.

```
a <- sum(blanco$acido_citrico==0)
b <- sum(tinto$acido_citrico==0)
sprintf("Número de muestras de vinos blancos con el ácido cítrico = 0 : %s",a)
```

```
## [1] "Número de muestras de vinos blancos con el ácido cítrico = 0 : 19"
```

```
sprintf("Número de muestras de vinos tintos con el ácido cítrico = 0 : %s",b)
```

```
## [1] "Número de muestras de vinos tintos con el ácido cítrico = 0 : 132"
```

Según la fuente externa consultada, los niveles habituales de acido cítrico en los vinos oscila entre 0 y 0,5. Este acido es otro más que está presente en gran cantidad de vinos pero no es extraño que el valor sea cero o que también esté por encima de 0.5, por tanto consideramos los valores como correctos.

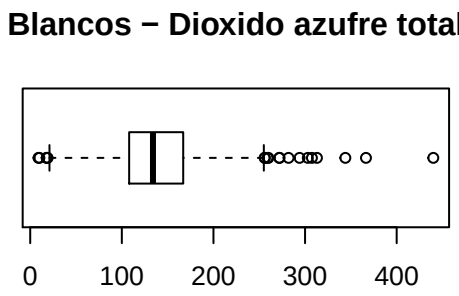
Valores extremos

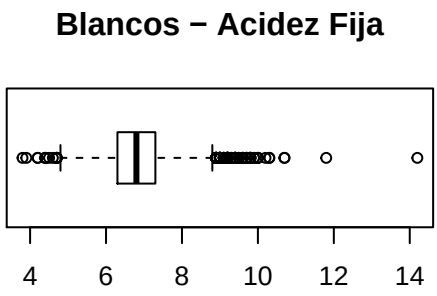
Vamos a realizar el análisis de cada una de las variables cualitativas del dataset. Para determinar si los outliers son valores correctos o no, nos apoyaremos en algunas fuentes externas que nos ayudarán a eliminar aquellos valores que concluyamos que no son correctos.

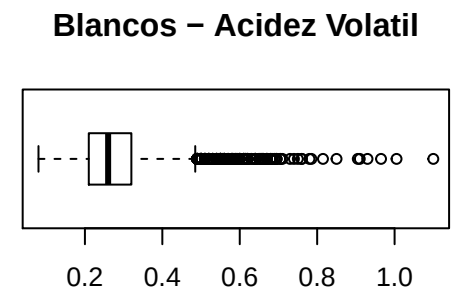
Comenzamos por realizar un gráfico **boxplot** para cada tipo de vino y variable.

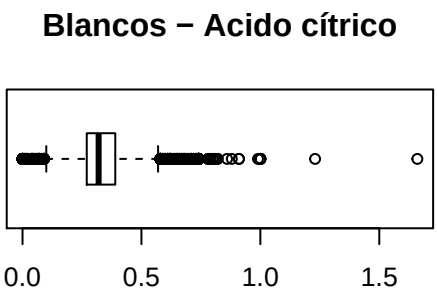
```
# Comprobamos outliers de las variables de los vinos blancos
#ggplot(totalData, aes(x=tipo, y=diox_azufre_total)) +  geom_point(size=2, shape=23)
par(mfrow = c(2,2))
datos.bp <-boxplot(blanco$diox_azufre_total, main="Blancos - Dioxido azufre total", horizontal = T)

datos.bp <-boxplot(blanco$acidez_fija, main="Blancos - Acidez Fija", horizontal = T)
```

```
datos.bp <-boxplot(blanco$acidez_volatil, main="Blancos - Acidez Volatil", horizontal = T)
datos.bp <-boxplot(blanco$acido_citrico, main="Blancos - Acido cítrico", horizontal = T)
```

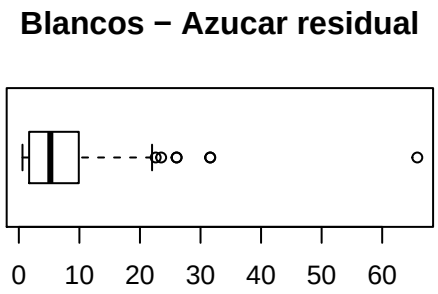


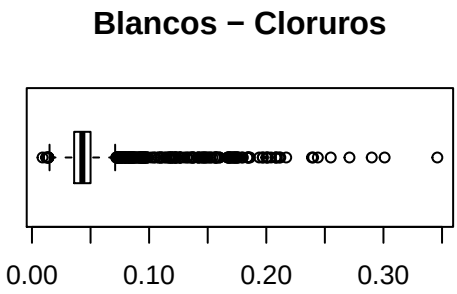


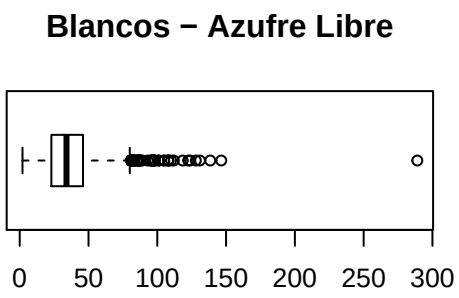


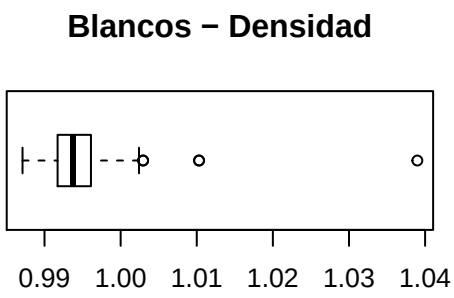



```
datos.bp <-boxplot(blanco$azucar_residual, main="Blancos - Azucar residual", horizontal = T)
datos.bp <-boxplot(blanco$cloruros, main="Blancos - Cloruros", horizontal = T)

datos.bp <-boxplot(blanco$diox_azufre_libre, main="Blancos - Azufre Libre", horizontal = T)
datos.bp <-boxplot(blanco$densidad, main="Blancos - Densidad", horizontal = T)
```

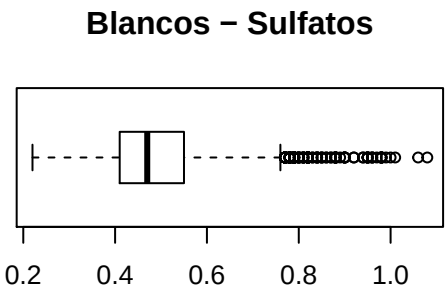


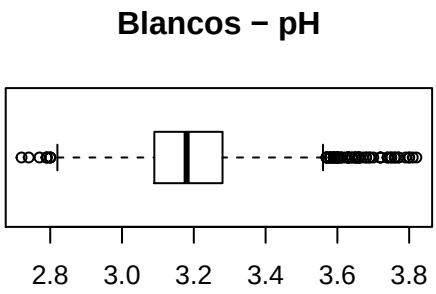


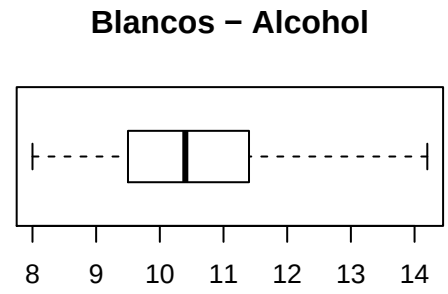





```
datos.bp <-boxplot(blanco$sulfatos, main="Blancos - Sulfatos", horizontal = T)
datos.bp <-boxplot(blanco$pH, main="Blancos - pH", horizontal = T)
datos.bp <-boxplot(blanco$alcohol, main="Blancos - Alcohol", horizontal = T)
```

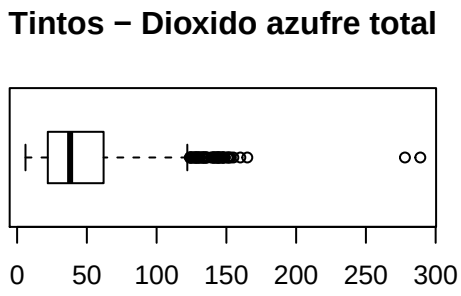


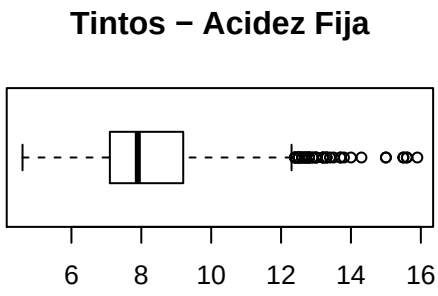


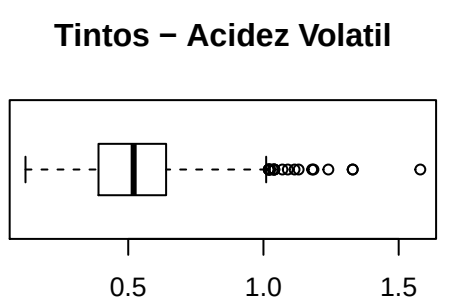


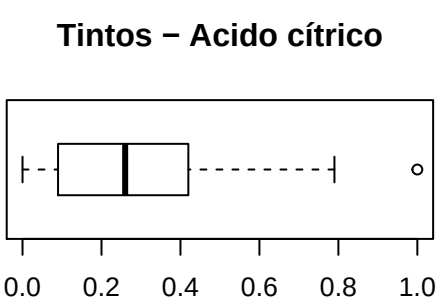
Seguidamente, hacemos el mismo ejercicio para observar los outliers de los vinos tintos:

```
par(mfrow = c(2,2))
datos.bp <-boxplot(tinto$diox_azufre_total, main="Tintos - Dioxido azufre total", horizontal = T)
datos.bp <-boxplot(tinto$acidez_fija, main="Tintos - Acidez Fija", horizontal = T)
datos.bp <-boxplot(tinto$acidez_volatil, main="Tintos - Acidez Volatil", horizontal = T)
datos.bp <-boxplot(tinto$acido_citrico, main="Tintos - Acido cítrico", horizontal = T)
```

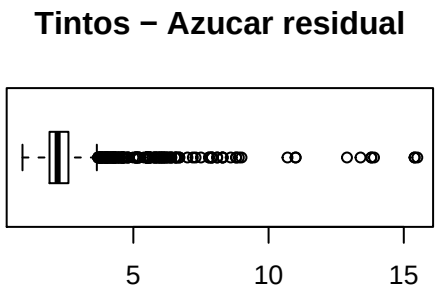


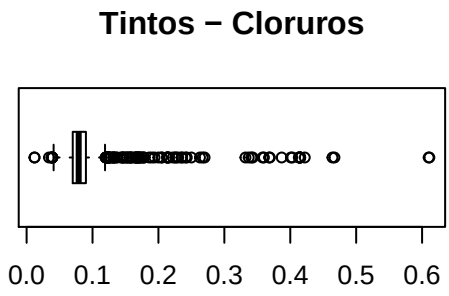


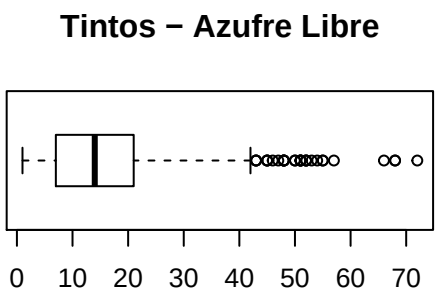


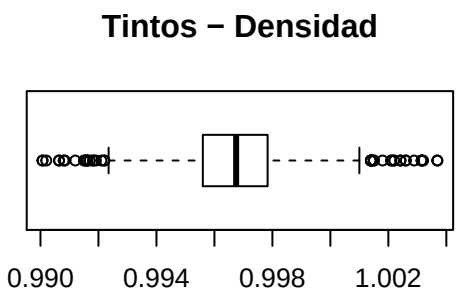



```
datos.bp <-boxplot(tinto$azucar_residual, main="Tintos - Azucar residual", horizontal = T)
datos.bp <-boxplot(tinto$cloruros, main="Tintos - Cloruros", horizontal = T)
datos.bp <-boxplot(tinto$diox_azufre_libre, main="Tintos - Azufre Libre", horizontal = T)
datos.bp <-boxplot(tinto$densidad, main="Tintos - Densidad", horizontal = T)
```

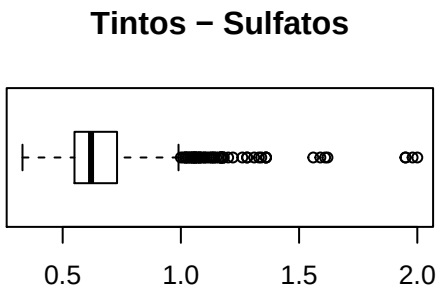


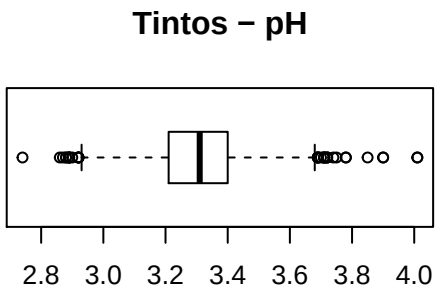


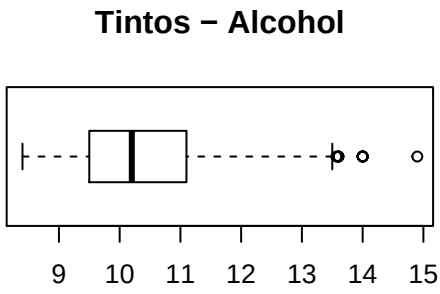





```
datos.bp <-boxplot(tinto$sulfatos, main="Tintos - Sulfatos", horizontal = T)
datos.bp <-boxplot(tinto$pH, main="Tintos - pH", horizontal = T)
datos.bp <-boxplot(tinto$alcohol, main="Tintos - Alcohol", horizontal = T)
```







A modo de ejemplo, se muestran los outliers de dos variables:

```
cat("OUTLIERS de la variable dióxido de azufre total en VINOS BLANCOS")
```

```
## OUTLIERS de la variable dióxido de azufre total en VINOS BLANCOS
```

```
boxplot.stats(blanco$diox_azufre_total)$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  10.0 259.0 440.0
```

```
cat("OUTLIERS de la variable dióxido de azufre total en VINOS TINTOS")
```

```
## OUTLIERS de la variable dióxido de azufre total en VINOS TINTOS
```

```
boxplot.stats(tinto$diox_azufre_total)$out
```

```
##  [1] 145 148 136 125 140 136 133 153 134 141 129 128 129 128 143 144 127
## [18] 126 145 144 135 165 124 124 134 124 129 151 133 142 149 147 145 148
## [35] 155 151 152 125 127 139 143 144 130 278 289 135 160 141 141 133 147
## [52] 147 131 131 131
```

```
write("  ")

cat("OUTLIERS de la variable ácido cítrico en VINOS BLANCOS")
```

```
## OUTLIERS de la variable ácido cítrico en VINOS BLANCOS
```

```
boxplot.stats(blanco$acido_citrico)$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
```

```
cat("OUTLIERS de la variable ácido cítrico en VINOS TINTOS")
```

```
## OUTLIERS de la variable ácido cítrico en VINOS TINTOS
```

```
boxplot.stats(tinto$acido_citrico)$out
```

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

Como se puede ver en los boxplots, en casi todas las variables, el sistema detecta valores atipicos. Al no tener un conocimiento suficiente como para valorar si se deben a algún error, al uso diferentes metodologias de medicion o o si por el contrario, son valores correctos, consultaremos fuentes externas que nos ayudarán en esta fase de limpieza de datos y que son referenciadas al final de este apartado.

Analizadas las graficas y consultadas las fuentes, de la muestra total podemos concluir que:

Dióxido de azufre total: Eliminamos del conjunto de datos aquellas muestras que tienen un valor > 400 y es blanco y de la de tintos los dos que tienen un valor superior a 250.
Acidez fija: Eliminaremos del conjunto tanto de vinos como de blancos los valores mayores de 12 que están muy alejados del rango intercuartilico y podria ser incorrecto.
Acidez volátil: La acidez volátil es crítica para la calidad de vino. Una acidez por encima de 1, nos dará un vino de pésima calidad, es probable también que el dato no sea correcto.
Ácido cítrico: Existen dos blancos que tienen esta variabale por encima de 1. Es muy raro que este presente en estas cantidades en los vinos. Los sacaremos del dataset.
Azúcar residual: Los valores habituales de los vinos oscilan entre 1 gr y 200 gr por litro de vino. Todos nuestros vinos están dentro de ese rango y por lo tanto los outilers son correctos. Los vinos dulces presentan un alto nivel de azúcar dentro de los valores indicados en las graficas. NO eliminaremos ninguna muestra.
Cloruros: Existen outliers tanto en tintos como en blancos, pero sus valores se consideran normales y no representan ningun error por lo que no eliminaremos ningun valor.
Sulfatos: No existen valores anormales en esta variable.
Dióxido de azufre libre: Vamos a optar por eliminar aquellos tengan un valor superior a 300. Es probable que sea incorrecto.
Densidad: La densidad del vino habitual suele estar entre 0.98-0.999 aprox. Hay algún valor indicado como outliers pero no son valores incorrectos porque la densidad habitual del vino dulce puede llegar hasta 1.115k, asi que podemos determinar que todos los valores son correctos. **pH**: El PH es una medida de acidez total que presenta un vino y que presenta un maximo de 4, siendo cuanto más alto menos acido. En nuestro dataset, eliminaremos las muestras por encima de 4 para eliminar muestras incorrectas.

FUENTES CONSULTADAS:

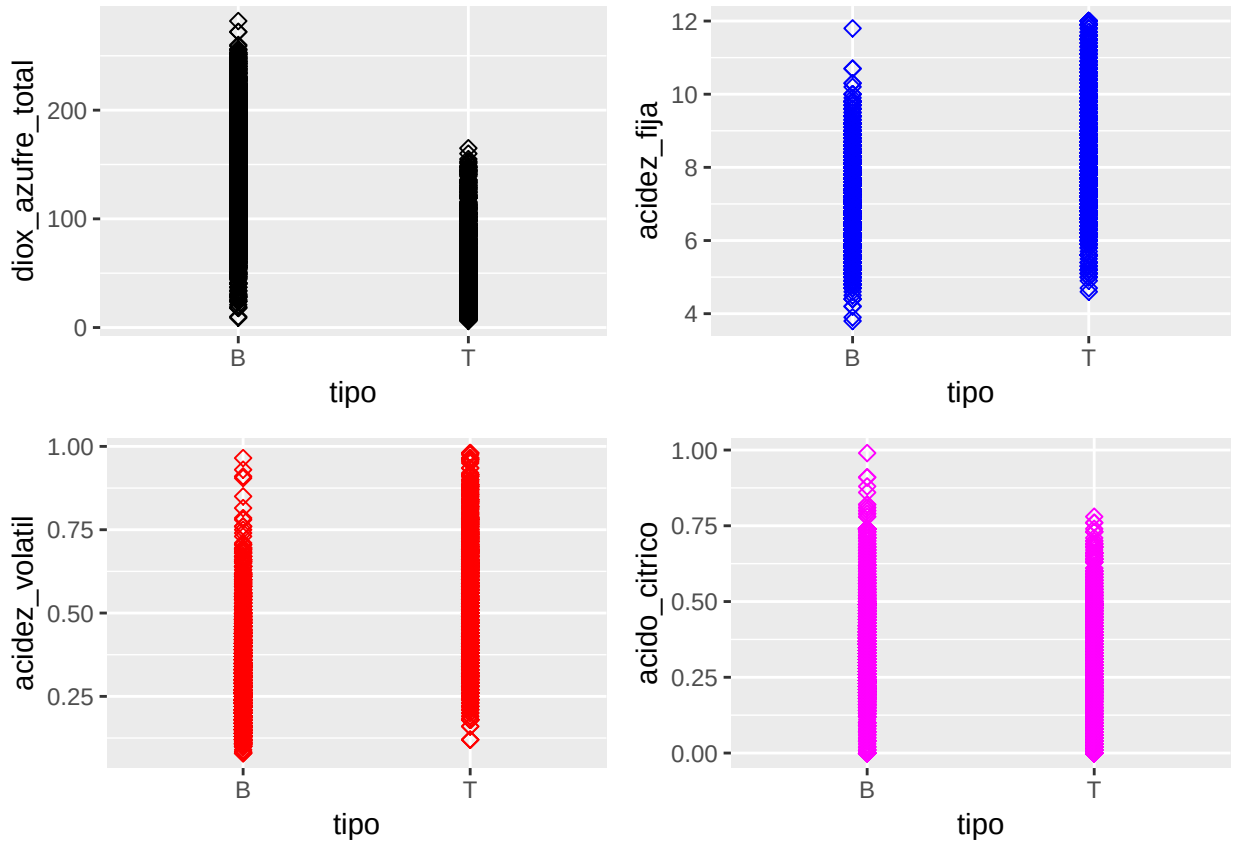
https://www.catadelvino.com/blog-cata-vino
https://foro.e-nologia.com/thread-37415-page-1.html
http://www.usc.es/caa/MetAnalisisStgo1/enologia.pdf

```
blanco <-subset(blanco, diox_azufre_total<400)
tinto <-subset(tinto, diox_azufre_total<250)

blanco <-subset(blanco, acidez_fija<=12)
tinto <-subset(tinto, acidez_fija<=12)

blanco <-subset(blanco, acidez_volatil<1)
tinto <-subset(tinto, acidez_volatil<1)

blanco<-subset(blanco, acido_citrico<1)
tinto<-subset(tinto, acido_citrico<1)

blanco<-subset(blanco, diox_azufre_total<300)
tinto<-subset(tinto, diox_azufre_total<300)

blanco<-subset(blanco, pH<4)
tinto<-subset(tinto, pH<4)

# , pero teniendo en cuenta que la suma de Cloruros, sulfatos y otras sales de un vino no tiene que sup
#blanco<-subset(blanco, sulfatos<1)
#tinto<-subset(tinto, sulfatos<1)
```

```
#unir ambos datasets
totalData <- bind_rows(blanco,tinto)

# Factorizamos la variable tipo
totalData$tipo <- as.factor(totalData$tipo)
```

Mostramos de los gráficos de dispersión una vez hemos eliminado los outliers.

```
par(mfrow = c(1,1))
p1<-ggplot(totalData, aes(x=tipo, y=diox_azufre_total)) + geom_point(size=2, shape=23)
p2<-ggplot(totalData, aes(x=tipo, y=acidez_fija)) + geom_point(size=2, shape=23, color="blue")
p3<-ggplot(totalData, aes(x=tipo, y=acidez_volatil)) + geom_point(size=2, shape=23, color= "red")
p4<-ggplot(totalData, aes(x=tipo, y=acido_citrico)) + geom_point(size=2, shape=23, color ="magenta")

gridExtra::grid.arrange(p1, p2,p3,p4, nrow = 2)
```



Análisis de los datos

Análisis descriptivo de la calidad

A continuación vamos a realizar un análisis descriptivo de la variable calidad.

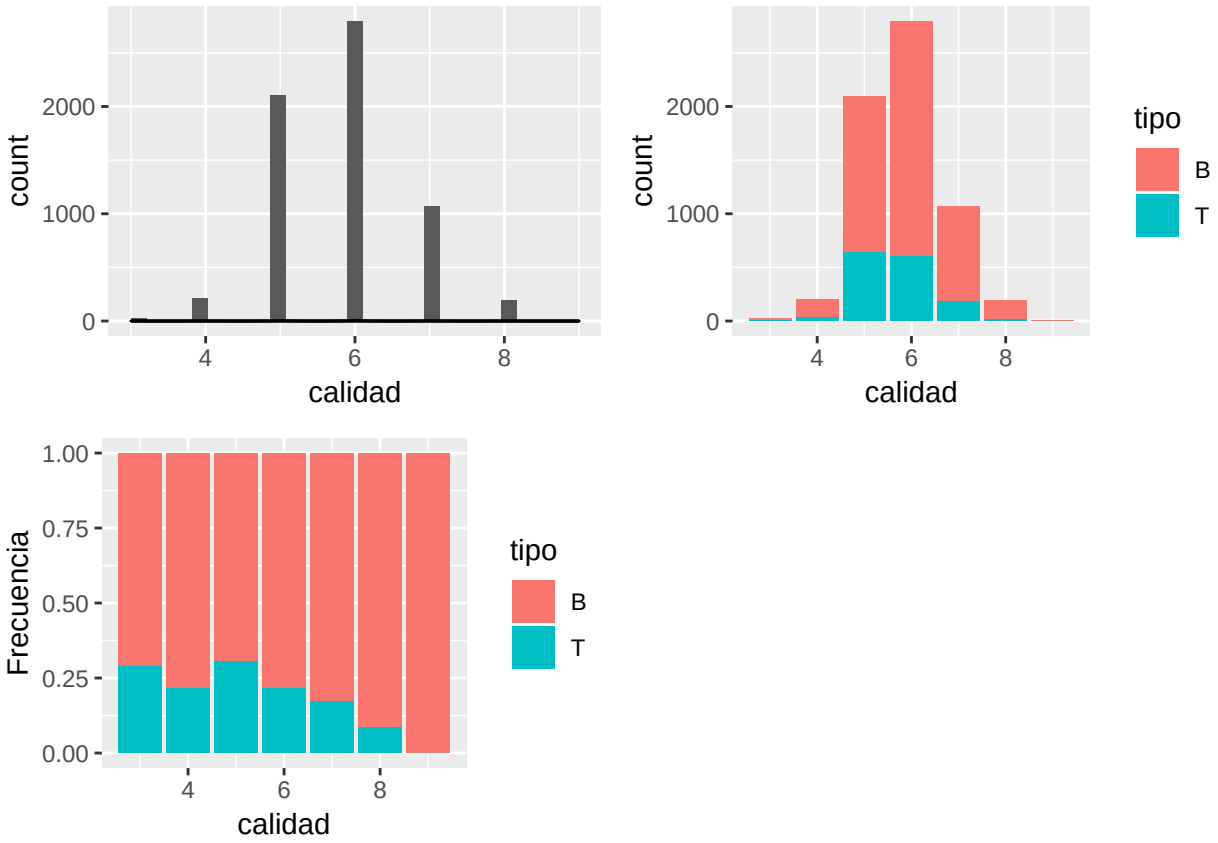
```
summary(totalData$calidad)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.000   5.000   6.000   5.825   6.000   9.000
```

```
#desviacion estandara
sd(totalData$calidad)
```

```
## [1] 0.8688916
```

```
# mostramos un histograma de la calidad
filas=dim(totalData)
p1<-ggplot(data = totalData[1:filas,],aes(x=calidad))+geom_histogram()+ geom_density(alpha=.2, fill="#F

# Relacion entre calidad y tipo de vino
p2<-ggplot(data=totalData[1:filas,],aes(x=calidad,fill=tipo))+geom_bar()

# Grafico de frecuencias
p3<-ggplot(data = totalData[1:filas,],aes(x=calidad,fill=tipo))+geom_bar(position="fill")+ylab("Frecuen

gridExtra::grid.arrange(p1, p2,p3,nrow = 2)
```



Se puede deducir de los gráficos que los vinos blancos de la muestra tienen más calidad que los tintos.

Análisis de la normalidad y homogeneidad de la varianza

Vamos a comprobar la normalidad en ambos grupos de vinos para cada una de la variables numéricas de nuestro dataset. Utilizaremos el tests de **Kolmogorov-Smirnov** y **Shapiro-Wilk**.

```
##
col.names = colnames(tinto)
alpha <- 0.05

for (i in 1:ncol(tinto)){
  if (i == 1) {cat("Variables que siguen una distribución normal en el grupo de vinos tintos:")}

  if (is.integer(tinto[,i]) | is.numeric(tinto[,i])){
    p_val = ks.test(tinto[,i], pnorm, mean(tinto[,i]), sd(tinto[,i]))$p.value
    if (p_val >= alpha){
      cat(col.names[i])
      #formatear la salida
      if (i< ncol(tinto) -1){cat(" ,")}
      if (i %% 3 == 0){cat("\n")}
    }
    p_val = shapiro.test(tinto[,i])$p.value
    if (p_val >= alpha){
      cat(col.names[i])
      #formatear la salida
      if (i< ncol(tinto) -1){cat(" ,")}
```

```
      if (i %% 3 == 0){cat("\n")}
    }
  }
}
```

```
## Variables que siguen una distribución normal en el grupo de vinos tintos:
```

```
for (i in 1:ncol(tinto)){
  if (i == 1) {cat("Variables que siguen una distribución normal en el grupo de vinos blancos:")}
  if (is.integer(blanco[,i]) | is.numeric(blanco[,i])){
    p_val = ks.test(blanco[,i], pnorm, mean(blanco[,i]), sd(blanco[,i]))$p.value
    if (p_val >= alpha){
      cat(col.names[i])
      #formatear la salida
      if (i< ncol(blanco) -1){cat(" ,")}
      if (i %% 3 == 0){cat("\n")}
    }
    p_val = shapiro.test(blanco[,i])$p.value
    if (p_val >= alpha){
      cat(col.names[i])
      #formatear la salida
      if (i< ncol(tinto) -1){cat(" ,")}
      if (i %% 3 == 0){cat("\n")}
    }
  }
}
```

```
## Variables que siguen una distribución normal en el grupo de vinos blancos:
```

Comprobamos que debemos rechazar la hipótesis nula en todas las variables de ambos grupos de vinos. No obstante, por el **teorema central del límite** se podria considerar que los datos siguen una distribución normal.

Analizaremos la homocedasticidad de la varianza de la variable calidad mediante el **test de Flinger-Killen** en para el conjunto total de vinos.

```
##
#b <- blanco$calidad
#t <- tinto$calidad
fligner.test(calidad ~ tipo, data= totalData)
```

```
##
##  Fligner-Killeen test of homogeneity of variances
##
## data:  calidad by tipo
## Fligner-Killeen:med chi-squared = 0.090506, df = 1, p-value =
## 0.7635
```

Dado que el p-valor es > 0.05 podemos aceptar la hipóstesis nula de que las varianzas de ambas muestras son homogéneas.

Vamos ahora a considerar la Calidad como una variable categórica y a comprobar la homogeneidad de la varianza del resto de variables cuando agrupamos las muestras por Calidad.

Dado que hemos comprobado que ninguna variable se distribuye con una Normal, aplicaremos el test de **Kruskal-Wallis** para comprobar si alguna variable presenta diferencias significativas en función de la calidad.

```
##
totalData2<-totalData

totalData2$calidadFactor<-totalData2$calidad
totalData2$calidadFactor <- as.factor(totalData2$calidadFactor)

#aplicamos test Kruskal-Wallis
matriz <-matrix(nc=2, nr=0)
colnames(matriz) <- c( "variable","p-value")
for (i in 1:(ncol(totalData2)-3)){
  if (is.integer(totalData2[,i]) | is.numeric(totalData2[,i])){
    kruskal.test = kruskal.test(totalData2[,i] ~ totalData2$calidadFactor, data=totalData2)
    p_val = kruskal.test$p.value

    tupla = matrix(ncol=2,nrow=1)
    tupla[1][1]=colnames(totalData2)[i]
    tupla[2][1]=p_val
    matriz <- rbind(matriz, tupla)

  }
}
print(matriz)
```

```
##       variable            p-value
##  [1,] "acidez_fija"       "5.4005185025087e-13"
##  [2,] "acidez_volatil"    "2.90738610120818e-94"
##  [3,] "acido_citrico"     "4.88643999651156e-14"
##  [4,] "azucar_residual"   "6.3338334080799e-07"
##  [5,] "cloruros"          "3.27259130072707e-127"
##  [6,] "diox_azufre_libre" "1.32801716621801e-24"
##  [7,] "diox_azufre_total" "2.48283286920673e-10"
##  [8,] "densidad"          "1.36788365438699e-162"
##  [9,] "pH"                "0.0302755337790672"
## [10,] "sulfatos"          "0.000408267131801604"
## [11,] "alcohol"           "4.59604922370829e-301"
```

Comprobamos como ninguna variable presenta un *p-value > 0.05*, con lo que debemos rechazar la hipótesis nula de homocedasticidad, por tanto todas las variables presentan varianzas estadisticamente diferentes para los diferentes grupos de calidad.

Pruebas estadísticas

¿Que tipo de vino tiene más calidad?

En los histogramas y gráficos de frecuencias pudimos observar que la calidad de los vinos blancos de la muestra era más alta que la de los tintos, vamos a realizar un contraste de hipótesis para comprobar si tenemos diferencias estadísticamente significativas en la media de la calidad de ambos grupos de vinos.

Considerando el análisis de la normalidad y homogeneidad de la varianza del punto anterior, aplicaremos la prueba **t de Student** formulando las siguientes hipótesis:

```
H0: μB - μT = 0
H1: μB - μT > 0
```

donde μB es la media muestral de la calidad de los vinos blancos y μT es la media muestral de la calidad de

los vinos tintos.

```
## Realizamos el test por tipo de vino
t.test(calidad ~ tipo, data = totalData, alternative="greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  calidad by tipo
## t = 9.886, df = 2758.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.1981451       Inf
## sample estimates:
## mean in group B mean in group T
##        5.880991        5.643283
```

Dado que el p-valor es inferior al nivel de significancia (0.05), debemos rechazar la hipótesis nula, por tanto podemos concluir que efectivamente, la calidad de los vinos blancos es superior que la de los vinos tintos de la muestra.

¿Qué prueba fisioquímica es más determinante para la calidad de un vino?

Vamos a calcular la matriz de correlaciones de las variables cuantitativas de cada grupo de vinos para determinar cuáles de ellas ejercen una mayor influencia sobre la calidad. Mediremos el coeficiente de correlación de **Spearman**.

```
library(rminer)
set.seed(123)
#TINTOS
tintoQ<-tinto[,-13]
matriz_corr <-matrix(nc=2, nr=0)
colnames(matriz_corr) <- c("estimate", "p-value")
for (i in 1:(ncol(tintoQ)-1)){
  if (is.integer(tintoQ[,i]) | is.numeric(tintoQ[,i])){
    spearman.test = cor.test(tintoQ[,i], tintoQ$calidad,method = "spearman")
    coeficiente_corr = spearman.test$estimate
    p_val = spearman.test$p.value

    tupla = matrix(ncol=2,nrow=1)
    tupla[1][1]=coeficiente_corr
    tupla[2][1]=p_val
    matriz_corr <- rbind(matriz_corr, tupla)
    rownames(matriz_corr)[nrow(matriz_corr)]<-colnames(tintoQ)[i]
  }
}
cat("Matriz de correlaciones en el grupo de vinos TINTO")
```

```
## Matriz de correlaciones en el grupo de vinos TINTO
```

```
print(matriz_corr)
```

```
##                     estimate       p-value
## acidez_fija       0.10591261  3.704271e-05
## acidez_volatil   -0.37034585  2.529387e-50
## acido_citrico     0.20493098  8.629631e-16
## azucar_residual   0.02173702  3.984723e-01
```

```
## cloruros          -0.19310650  3.688931e-14
## diox_azufre_libre -0.06537837  1.102273e-02
## diox_azufre_total -0.22475907  9.327689e-19
## densidad          -0.21175528  8.873686e-17
## pH                -0.02130805  4.078483e-01
## sulfatos           0.36945658  4.506027e-50
## alcohol            0.51199483 9.758564e-102
```

```
#BLANCOS
blancoQ<-blanco[,-13]
matriz_corr <-matrix(nc=2, nr=0)
colnames(matriz_corr) <- c("estimate", "p-value")
for (i in 1:(ncol(blancoQ)-1)){
  if (is.integer(blancoQ[,i]) | is.numeric(blancoQ[,i])){
    spearman.test = cor.test(blancoQ[,i], blancoQ$calidad,method = "spearman")
    coeficiente_corr = spearman.test$estimate
    p_val = spearman.test$p.value

    tupla = matrix(ncol=2,nrow=1)
    tupla[1][1]=coeficiente_corr
    tupla[2][1]=p_val
    matriz_corr <- rbind(matriz_corr, tupla)
    rownames(matriz_corr)[nrow(matriz_corr)]<-colnames(blancoQ)[i]
  }
}
cat("Matriz de correlaciones en el grupo de vinos BLANCO")
```

```
## Matriz de correlaciones en el grupo de vinos BLANCO
```

```
print(matriz_corr)
```

```
##                      estimate       p-value
## acidez_fija       -0.08350479  5.117605e-09
## acidez_volatil    -0.19422087  1.049079e-42
## acido_citrico      0.01730335  2.267439e-01
## azucar_residual   -0.08265992  7.295469e-09
## cloruros          -0.31389827 4.352436e-112
## diox_azufre_libre  0.02602957  6.897819e-02
## diox_azufre_total -0.19406575  1.223150e-42
## densidad          -0.34876749 1.194121e-139
## pH                 0.10993440  1.332736e-14
## sulfatos           0.03479720  1.503894e-02
## alcohol            0.44216548 7.616177e-233
```

```
#TOTAL
matriz_corr <-matrix(nc=2, nr=0)
colnames(matriz_corr) <- c("estimate", "p-value")
for (i in 1:(ncol(totalData)-2)){
  if (is.integer(totalData[,i]) | is.numeric(totalData[,i])){
    spearman.test = cor.test(totalData[,i], totalData$calidad,method = "spearman")
    coeficiente_corr = spearman.test$estimate
    p_val = spearman.test$p.value

    tupla = matrix(ncol=2,nrow=1)
    tupla[1][1]=coeficiente_corr
    tupla[2][1]=p_val
```

```
    matriz_corr <- rbind(matriz_corr, tupla)
    rownames(matriz_corr)[nrow(matriz_corr)]<-colnames(totalData)[i]
  }
}
cat("Matriz de correlaciones en el conjunto de vinos")
```

```
## Matriz de correlaciones en el conjunto de vinos
```

```
print(matriz_corr)
```

```
##                      estimate       p-value
## acidez_fija       -0.09895104  2.197245e-15
## acidez_volatil    -0.25163912  6.514234e-93
## acido_citrico      0.10304834  1.459869e-16
## azucar_residual   -0.02033888  1.039346e-01
## cloruros          -0.29568142 3.647608e-129
## diox_azufre_libre  0.08535199  8.168213e-12
## diox_azufre_total -0.06114246  9.964279e-07
## densidad          -0.32836516 1.404146e-160
## pH                 0.03757584  2.656713e-03
## sulfatos           0.02835340  2.338755e-02
## alcohol            0.45481010  0.000000e+00
```

Vemos que las correlaciones son bajas, que presentan bastantes diferencias entre los distintos grupos de vinos y generalmente están más fuertemente correladas en el grupo de vinos tinto por lo que seguramente funcione mejor un modelo de regresión lineal en dicho tipo de vinos.

Regresión lineal

Vamos a intentar encontrar un modelo de regresión lineal que nos permita inferir la calidad de un vino a partir de ciertas características fisioquímicas.

Probaremos varios modelos utilizando la información obtenida en el punto 4.2

Regresión lineal para el conjunto total de vinos.

En primer lugar estudiamos un modelo de regresión para el conjunto total de vinos, utilizaremos el método de exclusión o *holdout* con partición de datos estratificada para clasificar los datos originales en entrenamiento y test.

```
h <- holdout(totalData$calidad,ratio=2/3,mode="stratified")
training <- totalData[h$tr,]
test <- totalData[h$ts,]

modelo1 <- lm(calidad ~ ., data = training)
modelo2 <- lm(calidad ~ alcohol+densidad+cloruros+acidez_volatil+azucar_residual, data = training)
modelo3 <- lm(calidad ~ tipo+alcohol+densidad+cloruros+acidez_volatil+azucar_residual, data = training)
modelo4 <- lm(calidad ~ alcohol, data = training)
tabla.coeficientes <-matrix(c(1,summary(modelo1)$r.squared,
                              2, summary(modelo2)$r.squared,
                              3, summary(modelo3)$r.squared,
                              4, summary(modelo4)$r.squared),
                            ncol=2,byrow=TRUE)
colnames(tabla.coeficientes)<-c("Modelo", "R^2")
tabla.coeficientes
```

```
##      Modelo       R^2
```

```
## [1,]      1 0.2953709
## [2,]      2 0.2671876
## [3,]      3 0.2787676
## [4,]      4 0.1981255
```

El modelo que tiene el R2 más alto es el primero, el que contiene todas las variables, vamos a aplicar una seleccion de predictores empleando *stepwise selection*

```
step(modelo1, direction = "both",trace=0)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
##     diox_azufre_libre + diox_azufre_total + densidad + pH + sulfatos +
##     alcohol + tipo, data = training)
##
## Coefficients:
##       (Intercept)        acidez_fija     acidez_volatil
##         1.139e+02          9.591e-02         -1.501e+00
##   azucar_residual  diox_azufre_libre  diox_azufre_total
##         6.915e-02          5.001e-03         -1.317e-03
##          densidad                 pH           sulfatos
##        -1.137e+02          5.009e-01          6.760e-01
##            alcohol              tipoT
##         2.300e-01          3.864e-01
```

La selección de predictores ha identificado como mejor modelo el formado por las variables acidez_fija, acidez_volatil, azucar_residual, cloruros, diox_azufre_libre, diox_azufre_total, densidad, pH, sulfatos, alcohol y tipo. Ha eliminado del modelo el ácido cítrico. Vamos a generar el nuevo modelo

```
modelo1 <- lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
     cloruros + diox_azufre_libre + diox_azufre_total + densidad +
     pH + sulfatos + alcohol + tipo, data = training)
summary(modelo1)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
##     cloruros + diox_azufre_libre + diox_azufre_total + densidad +
##     pH + sulfatos + alcohol + tipo, data = training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5962 -0.4819 -0.0404  0.4645  2.9829
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.116e+02  1.715e+01   6.507 8.55e-11 ***
## acidez_fija         9.364e-02  1.961e-02   4.776 1.85e-06 ***
## acidez_volatil     -1.486e+00  9.950e-02 -14.935  < 2e-16 ***
## azucar_residual     6.784e-02  7.383e-03   9.188  < 2e-16 ***
## cloruros           -5.839e-01  4.197e-01  -1.391  0.16420
## diox_azufre_libre   5.035e-03  9.774e-04   5.151 2.70e-07 ***
## diox_azufre_total  -1.323e-03  4.089e-04  -3.235  0.00123 **
## densidad           -1.112e+02  1.743e+01  -6.381 1.94e-10 ***
## pH                  4.758e-01  1.119e-01   4.251 2.18e-05 ***
```

```
## sulfatos           6.970e-01  9.562e-02   7.289 3.70e-13 ***
## alcohol            2.283e-01  2.200e-02  10.374  < 2e-16 ***
## tipoT              3.980e-01  7.155e-02   5.562 2.83e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7338 on 4250 degrees of freedom
## Multiple R-squared:  0.2953, Adjusted R-squared:  0.2934
## F-statistic: 161.9 on 11 and 4250 DF,  p-value: < 2.2e-16
```

Comprobamos como el coeficiente R2 es prácticamente idéntico y hemos reducido la dimensionalidad. Vamos a verificar su precisión calculando la media de los cuadrados de las desviaciones.

```
# funcion que calcula la media de los cuadrados de las desviaciones
dm <- function(actual, predicted){
  mean((actual - predicted)^2)
}

# MSE empleando las observaciones de entrenamiento
training_mse <- dm(modelo1$fitted.values, training$calidad)

# MSE empleando nuevas observaciones
predicciones <- predict(modelo1, newdata = test)
test_mse <- dm(predicciones, test$calidad)

sprintf("MSE de la muestra de entrenamiento (total): %s", training_mse)
```

```
## [1] "MSE de la muestra de entrenamiento (total): 0.536981693299343"
```

```
sprintf("MSE de la muestra de test (total): %s", test_mse)
```

```
## [1] "MSE de la muestra de test (total): 0.514231151414263"
```

Regresión lineal para el conjunto de vinos tintos.

Ahora vamos a repetir el proceso para el grupo de vinos tinto, probando el modelo con todas las variables y con las variables más correladas con respecto a la calidad obtenidas en el punto 4.2.

```
h <- holdout(tintoQ$calidad,ratio=2/3,mode="stratified")
training <- tintoQ[h$tr,]
test <- tintoQ[h$ts,]

modelo1 <- lm(calidad ~ ., data = training)
modelo2 <- lm(calidad ~ acidez_fija+acidez_volatil+acido_citrico+ cloruros+diox_azufre_total+densidad+su
tabla.coeficientes <-matrix(c(1,summary(modelo1)$r.squared,
                              2, summary(modelo2)$r.squared),
                            ncol=2,byrow=TRUE)
colnames(tabla.coeficientes)<-c("Modelo", "R^2")
tabla.coeficientes
```

```
##      Modelo      R^2
## [1,]      1 0.374502
## [2,]      2 0.372272
```

El modelo que tiene el R2 más alto es el primero, el que contiene todas las variables, aplicamos también una selección de predictores.

```
step(modelo1, direction = "both",trace=0)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + acido_citrico +
##     cloruros + diox_azufre_libre + diox_azufre_total + sulfatos +
##     alcohol, data = training)
##
## Coefficients:
##       (Intercept)        acidez_fija     acidez_volatil
##          2.167188           0.046766          -0.998238
##     acido_citrico           cloruros  diox_azufre_libre
##         -0.345666          -0.965749           0.004432
## diox_azufre_total           sulfatos             alcohol
##         -0.003591           0.950347           0.314505
```

```
modelo1<-lm(formula = calidad ~ acidez_fija + acidez_volatil + cloruros +  diox_azufre_libre + diox_azu
   alcohol, data = training)
summary(modelo1)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + cloruros +
##     diox_azufre_libre + diox_azufre_total + densidad + sulfatos +
##     alcohol, data = training)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.35555 -0.37535 -0.05975  0.43174  1.89995
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -1.3914715 19.4002038  -0.072   0.9428
## acidez_fija        0.0223592  0.0195531   1.144   0.2531
## acidez_volatil    -0.8233512  0.1321682  -6.230 6.88e-10 ***
## cloruros          -1.2155411  0.5495770  -2.212   0.0272 *
## diox_azufre_libre  0.0052175  0.0025352   2.058   0.0399 *
## diox_azufre_total -0.0040485  0.0008506  -4.760 2.22e-06 ***
## densidad           3.6745259 19.4434072   0.189   0.8501
## sulfatos           0.9479250  0.1382359   6.857 1.23e-11 ***
## alcohol            0.3093502  0.0260456  11.877  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6243 on 998 degrees of freedom
## Multiple R-squared:  0.372,  Adjusted R-squared:  0.3669
## F-statistic: 73.89 on 8 and 998 DF,  p-value: < 2.2e-16
```

En este caso comprobamos que el R2 es un poco peor, vamos a verificar su precisión calculando la media de los cuadrados de las desviaciones.

```
# MSE empleando las observaciones de entrenamiento
training_mse <- dm(modelo1$fitted.values, training$calidad)

# MSE empleando nuevas observaciones
```

```
predicciones <- predict(modelo1, newdata = test)
test_mse <- dm(predicciones, test$calidad)

sprintf("MSE de la muestra de entrenamiento (Tintos): %s", training_mse)
```

```
## [1] "MSE de la muestra de entrenamiento (Tintos): 0.386314566928667"
```

```
sprintf("MSE de la muestra de test (Tintos): %s", test_mse)
```

```
## [1] "MSE de la muestra de test (Tintos): 0.417886443054408"
```

Regresión lineal para el conjunto de vinos blancos.

Por último repetimos el proceso para el grupo de vinos blanco, probando el modelo con todas las variables y con las variables más correladas con respecto a la calidad obtenidas en el punto 4.2.

```
h <- holdout(blancoQ$calidad,ratio=2/3,mode="stratified")
training <- blancoQ[h$tr,]
test <- blancoQ[h$ts,]

modelo1 <- lm(calidad ~ ., data = training)
modelo2 <- lm(calidad ~ acidez_volatil+ cloruros+diox_azufre_total+densidad+pH+alcohol, data = training)
tabla.coeficientes <-matrix(c(1,summary(modelo1)$r.squared,
                              2, summary(modelo2)$r.squared),
                            ncol=2,byrow=TRUE)
colnames(tabla.coeficientes)<-c("Modelo", "R^2")
tabla.coeficientes
```

```
##      Modelo       R^2
## [1,]      1 0.2766202
## [2,]      2 0.2447963
```

El modelo que tiene el R2 más alto es el primero, el que contiene todas las variables, aplicamos la selección de predictores.

```
step(modelo1, direction = "both",trace=0)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
##     diox_azufre_libre + densidad + pH + sulfatos + alcohol, data = training)
##
## Coefficients:
##       (Intercept)        acidez_fija     acidez_volatil
##         1.298e+02          4.765e-02         -1.840e+00
##   azucar_residual  diox_azufre_libre           densidad
##         7.280e-02          4.722e-03         -1.296e+02
##                pH           sulfatos            alcohol
##         5.342e-01          5.954e-01          2.278e-01
```

```
modelo1<-lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
     cloruros + diox_azufre_libre + densidad + pH + sulfatos +
     alcohol, data = training)
summary(modelo1)
```

```
##
## Call:
## lm(formula = calidad ~ acidez_fija + acidez_volatil + azucar_residual +
```

```
##     cloruros + diox_azufre_libre + densidad + pH + sulfatos +
##     alcohol, data = training)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5358 -0.5068 -0.0353  0.4661  2.7619
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         1.248e+02  2.174e+01   5.742 1.02e-08 ***
## acidez_fija         4.217e-02  2.527e-02   1.669   0.0952 .
## acidez_volatil     -1.821e+00  1.391e-01 -13.096  < 2e-16 ***
## azucar_residual     7.037e-02  8.974e-03   7.842 5.98e-15 ***
## cloruros           -8.489e-01  6.650e-01  -1.277   0.2019
## diox_azufre_libre   4.782e-03  8.606e-04   5.556 2.98e-08 ***
## densidad           -1.244e+02  2.207e+01  -5.635 1.91e-08 ***
## pH                  5.043e-01  1.287e-01   3.919 9.07e-05 ***
## sulfatos            5.904e-01  1.238e-01   4.768 1.95e-06 ***
## alcohol             2.275e-01  2.850e-02   7.982 1.98e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.755 on 3244 degrees of freedom
## Multiple R-squared:  0.2762, Adjusted R-squared:  0.2742
## F-statistic: 137.6 on 9 and 3244 DF,  p-value: < 2.2e-16
```

En este caso tampoco mejora el modelo con la selección de predictores, aunque al igual que en las pruebas anteriores, se ha reducido la dimensionalidad sin perder calidad en el modelo.

Verificamos su precisión calculando la media de los cuadrados de las desviaciones.

```
# MSE empleando las observaciones de entrenamiento
training_mse <- dm(modelo1$fitted.values, training$calidad)

# MSE empleando nuevas observaciones
predicciones <- predict(modelo1, newdata = test)
test_mse <- dm(predicciones, test$calidad)

sprintf("MSE de la muestra de entrenamiento (Blancos): %s", training_mse)
```

```
## [1] "MSE de la muestra de entrenamiento (Blancos): 0.568313585999137"
```

```
sprintf("MSE de la muestra de test (Blancos): %s", test_mse)
```

```
## [1] "MSE de la muestra de test (Blancos): 0.538487741712506"
```

Como habíamos podido intuir viendo las matrices de correlaciones de las variables con respecto a la calidad del punto 4.2, el mejor modelo lo hemos obtenido con el grupo de vinos tinto pero al no estar fuertemente correladas, la precisión del modelo no es buena.

Modelos de clasificación.

Random forest

A continuación vamos a aplicar un método de clasificación random forest mediante una validación cruzada con 4 folds para clasificar los vinos en tintos o blancos.

```
library(caret)


h <- holdout(totalData$tipo,ratio=2/3,mode="stratified")
vino_entrenamiento <- totalData[h$tr,]
vino_prueba <- totalData[h$ts,]

train_control <-trainControl(method = "cv", number = 4)
mod<-train(tipo~., data=vino_entrenamiento, method="rf",trControl=train_control)
pred <- predict(mod, newdata=vino_prueba)
```

Obtenemos la matriz de confusión para comprobar la bondad del modelo.

```
confusionMatrix(pred,vino_prueba$tipo)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    B    T
##          B 1625    9
##          T    2  495
##
##                Accuracy : 0.9948
##                  95% CI : (0.9908, 0.9974)
##     No Information Rate : 0.7635
##     P-Value [Acc > NIR] : < 2e-16
##
##                   Kappa : 0.9856
##
##  Mcnemar's Test P-Value : 0.07044
##
##             Sensitivity : 0.9988
##             Specificity : 0.9821
##          Pos Pred Value : 0.9945
##          Neg Pred Value : 0.9960
##              Prevalence : 0.7635
##          Detection Rate : 0.7626
##    Detection Prevalence : 0.7668
##       Balanced Accuracy : 0.9905
##
##        'Positive' Class : B
##
```

Vemos que el resultado es excelente, el modelo nos clasifica los vinos con una precisión del 99.45% con un índice **kappa=0.985** que nos indica que nuestra clasificación es un 98.5% mejor que una clasificación aleatoria.

Arbol de clasificacion para calidad

Vamos ahora a considerar la calidad como una variable categórica y a utilizar un modelo de clasificación.

```
totalData$calidad <- as.factor(totalData$calidad)
h <- holdout(totalData$calidad,ratio=2/3,mode="stratified")
vino_entrenamiento <- totalData[h$tr,]
vino_prueba <- totalData[h$ts,]
```

```
train_control <-trainControl(method = "repeatedcv", number = 4)
mod<-train(calidad ~., data=vino_entrenamiento, method="rf",trControl=train_control)
pred <- predict(mod, newdata=vino_prueba)
```

Obtenemos la matriz de confusión para comprobar la bondad del modelo.

```
confusionMatrix(pred,vino_prueba$calidad)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   3   4   5   6   7   8   9
##          3   0   0   0   0   0   0   0
##          4   0   7   6   1   0   0   0
##          5   4  33 504 155   5   0   0
##          6   4  27 189 729 177  25   2
##          7   0   2   2  47 174  23   0
##          8   0   0   0   0   0  16   0
##          9   0   0   0   0   0   0   0
##
## Overall Statistics
##
##                Accuracy : 0.6707
##                  95% CI : (0.6503, 0.6907)
##     No Information Rate : 0.4371
##     P-Value [Acc > NIR] : < 2.2e-16
##
##                   Kappa : 0.482
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 3 Class: 4 Class: 5 Class: 6 Class: 7 Class: 8
## Sensitivity          0.000000 0.101449   0.7190   0.7822  0.48876 0.250000
## Specificity          1.000000 0.996607   0.8623   0.6467  0.95833 1.000000
## Pos Pred Value            NaN 0.500000   0.7190   0.6323  0.70161 1.000000
## Neg Pred Value       0.996248 0.970727   0.8623   0.7926  0.90340 0.977316
## Prevalence           0.003752 0.032364   0.3288   0.4371  0.16698 0.030019
## Detection Rate       0.000000 0.003283   0.2364   0.3419  0.08161 0.007505
## Detection Prevalence 0.000000 0.006567   0.3288   0.5408  0.11632 0.007505
## Balanced Accuracy    0.500000 0.549028   0.7907   0.7144  0.72355 0.625000
##                      Class: 9
## Sensitivity          0.0000000
## Specificity          1.0000000
## Pos Pred Value             NaN
## Neg Pred Value       0.9990619
## Prevalence           0.0009381
## Detection Rate       0.0000000
## Detection Prevalence 0.0000000
## Balanced Accuracy    0.5000000
```

Tal y como sucedía con el modelo de regresión lineal, el modelo random forest no nos da una precisión muy alta (67.07%) clasificando por calidad.

Conclusiones

Se han realizado varias pruebas estadísticas con el objetivo de inferir la calidad de los vinos y de clasificarlos por tipo Tinto y Blanco tal y como se había planteado al principio. Para inferir la calidad de los vinos se han realizado modelos de regresión cuyos resultados no han sido del todo satisfactorios. Posteriormente se ha factorizado la calidad para construir un modelo random forest con el que clasificar los vinos por calidad aunque tampoco se ha obtenido mucha precisión en el modelo. Por lo tanto, podemos concluir que las características fisioquímicas de los vinos no son un buen indicador para medir la calidad de un vino.

En cuanto a los métodos de clasificación, sí que podemos concluir que las características fisioquímicas nos permiten clasificar con gran precisión entre los tipos de vino.

También se ha realizado un contraste de hipótesis mediante el cual se ha podido concluir que los vinos que pertenecen al tipo de vinos blanco tienen más calidad que las de los vinos tintos.