# Tipología y ciclo de vida de los datos: Práctica 2 - Limpieza y análisis de datos

Autor: José Antonio González Constanza

Diciembre 2021

## 1. Detalles de la actividad

### 1.1. Descripción

En esta actividad se elabora un caso práctico, consistente en el tratamiento de un conjunto de datos, 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.

### 1.2. Objetivos

Los objetivos concretos de esta práctica son:

- 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.

### 1.3. Competencias

En cuanto a las competencias a desarrollar:

- 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.

# 2. Resolución

## 2.1. Descripción del dataset

El conjunto de datos objeto de análisis se ha obtenido a partir de este enlace "https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009" en Kaggle y está constituido por 12 características (columnas) que presentan 1.599 observaciones de vinos tintos (filas o registros).

Entre los campos de este conjunto de datos, encontramos los siguientes:

- fixded acidity: La acidez fija del vino. La acidez fija es el conjunto de ácidos naturales del vino (tartárico, málico, cítrico, succínico y láctico). La volatilidad de los ácidos en los vinos es muy baja, lo cual es un buen síntoma para que el vino tenga la calidad que el enólogo busca de cara a su comercialización.
- volatile acidity: La acidez volátil del vino. La acidez volátil es una parte de la acidez total de un vino, formada por los ácidos primarios que ya están presentes en el mosto de uva (málico y tartárico) y los secundarios que son los generados durante los procesos de fermentación.
- citric acid: El ácido cítrico del vino. El ácido cítrico (E-330) es un acidificante para corregir la acidez en mostos y vinos, además posee una acción estabilizante como antioxidante. El ácido cítrico forma complejos naturales con Fe(III), por tanto su adición puede reforzar esta acción secuestrando una cierta cantidad del hierro contenido en el vino.
- residual sugar: El azúcar residual en el contenido del vino. El azúcar residual es la cantidad total de azúcar que queda en el vino que no ha sido fermentada por las levaduras, y parte de ese azúcar no fermentado son las Pentosas, azúcares presentes en el vino en concentraciones cercanas a 1 gramo por litro de mosto.
- chlorides: Los cloruros que pueda presentar el vino.
- free sulfur dioxide: El dióxido de azufre libre. El dióxido de azufre (SO2) se utiliza en enología principalmente como conservante, pero también para otros fines (por ejemplo, para funciones antisépticas, antioxidantes, antioxidasicas, solubilizantes, combinadas y clarificantes).
- total sulfur dioxide: El dióxido de azufre total. El contenido total de anhídrido sulfuroso no puede superar los 150 mg/l para vinos tintos y los 200 mg/l para vinos blancos y rosados.
- density: La densidad del vino. La densidad relativa a 20ºC se obtiene multiplicando la masa volúmica por el factor 1,0018. Se expresa con cuatro

decimales y es adimensional. Los valores habituales de la masa volúmica a 20ºC para cada tipo de muestra son: 1) Vino blanco seco: 0,9880-0,9930 g/mL. 2) Vinos tinto seco: 0,9910-0,9950 g/mL.

- pH: El pH en los vinos varía entre 3 a 4, el de un vino blanco se encuentra aproximadamente entre 3,0- 3., mientras que el de un vino tinto entre 3,3 y 3,6.
- sulphates: Sulfatos. Los sulfatos de sodio y calcio aparecen en el agua y por lo tanto la uva y el vino pueden contenerlos.
- alcohol: El nivel de alcohol del vino. Los vinos, "habitualmente", se hallan entre valores de alcohol de 10 a 14° (diez a catorce grados). Los vinos tintos suelen estar comprendidos entre 12 y 13° y los blancos y rosados entre 10 y 12°. La cuestión no es simple para los blancos y rosados, no obstante, el dataset incluye solo vinos tintos.
- quality (score between 0 and 10): La calidad del vino fijada en un rango del 1 al 10. Que un vino sea o no de buena calidad depende mucho de la propia uva porque más del 70% de su calidad lo representan estas mismas, frente al propio proceso de elaboración.

## 2.2. Importancia y objetivos de los análisis

A partir de este conjunto de datos se plantea la problemática de determinar qué variables influyen más para determinar la calidad de un vino. Se procederá al desarrollo de modelos de regresión que permitan predecir la calidad del vino en función de sus características y contrastes de hipótesis que ayuden a identificar propiedades interesantes en las muestras que puedan ser inferidas con respecto a la población. Estos análisis serán de vital importancia a la hora de determinar la calidad de los vinos y por tanto, a la hora de fijar sus precios en los diferentes puntos de venta. Este tipo de análisis nos permiten diferenciar la calidad entre diferentes vinos tintos haciendo uso del aprendizaje automático. Este desarrollo abre el camino hacia la automatización en la capacidad de determinar la calidad de un vino, en base a unos criterios numéricos.

## 2.3. Limpieza de los datos

Antes de comenzar con la limpieza de los datos, procedemos a realizar la lectura del fichero en formato CSV en el que se encuentran. El resultado devuelto por la llamada a la función read.csv() será un objeto data.frame:

```
# Lectura de datos
red.wines.DT <- read.csv("D:/Master Ciencia de Datos/Tipología y ciclo de
vida de datos/Practica 2/winequality-red.csv")
head(red.wines.DT)

# Tipo de dato asignado a cada campo
sapply(red.wines.DT, function(x) class(x))
##        fixed.acidity     volatile.acidity          citric.acid
##            "numeric"           "numeric"            "numeric"
##       residual.sugar           chlorides  free.sulfur.dioxide
##            "numeric"           "numeric"            "numeric"
## total.sulfur.dioxide             density                   pH
```

```
##           "numeric"           "numeric"           "numeric"
##           sulphates             alcohol             quality
##           "numeric"           "numeric"           "integer"
str(red.wines.DT)
# Número de filas y columnas del dataset
rows <- dim(red.wines.DT)[1]
columns <- dim(red.wines.DT)[2]
rows
## [1] 1599
columns
## [1] 12
```

Comprobamos que todas las características son de tipo numérico o integer.

### 2.3.1. Selección de los datos de interés

La totalidad de los atributos presentes en el conjunto de datos son considerados interesantes a efectos de predecir la calidad de un vino. Por tanto, se tienen en cuenta el conjunto de todas las variables independientes que componen el dataset.

### 2.3.2. Ceros y elementos vacíos

Debemos determinar la posibilidad de que existan valores NA's o vacios que deban ser corregidos.

```
# Comprobamos si existen datos NA's o "".
colSums(is.na(red.wines.DT))
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                    0
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
##                    0                    0                    0
##            sulphates              alcohol              quality
##                    0                    0                    0
colSums(red.wines.DT == "")
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                    0
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
##                    0                    0                    0
##            sulphates              alcohol              quality
##                    0                    0                    0
# Otra forma de conocer los valores NA's
sapply(red.wines.DT, function(x) sum(is.na(x)))
##        fixed.acidity     volatile.acidity          citric.acid
##                    0                    0                    0
##       residual.sugar            chlorides  free.sulfur.dioxide
##                    0                    0                    0
## total.sulfur.dioxide              density                   pH
```

```
##                    0                    0                    0
##            sulphates              alcohol              quality
##                    0                    0                    0
```

No se han encontrado valores NA's ni "" vacios por lo que no es necesario realizar ningún tipo de corrección a tales efectos.

### 2.3.3. Valores extremos

Los valores extremos o outliers son aquellos que parecen no ser congruentes si los comparamos con el resto de los datos. Para identificarlos, podemos hacer uso de dos vías:

(1) representar un diagrama de caja por cada variable y ver qué valores distan mucho del rango intercuartílico (la caja). Comprobamos los outliers observando las medias y medianas, así como los valores máximos y minimos de las observaciones continuas. Cuando encontremos que la mediana está muy distante de la media, podemos intuir que estamos ante un fenomeno raro. La mayoría de las características presentan valores de media y mediana similares. Lo único que deberíamos proceder a examinar sería "total.sulfur.dioxide" con una diferencia entre media y mediana de 8,47.

Definiciones:

Rango intercuantílico (IQR: InterQuantile Range)

Procedo a determinar como valor atípico leve aquel que dista 1,5 veces el rango intercuantílico por debajo de Q1 o por encima de Q3. Un valor atípico extremo sería aquel que dista 3 veces el rango intercuantílico por debajo de Q1 o por encima del cuartil Q3.

```
# Análisis de outliers
summary(red.wines.DT)
# IQR para fixed.acidity
Q1.fa <- 22
Q3.fa <- 62

IQR.fa <- Q3.fa - Q1.fa
IQR.fa
## [1] 40
slight.outlier.value.fa <- Q1.fa - 1.5 * IQR.fa
extreme.outlier.value.fa <- Q1.fa - 3 * IQR.fa
slight.outlier.value.fa
## [1] -38
extreme.outlier.value.fa
## [1] -98
# Calculamos el umbral por encima para valores atípicos leves
Q1.fa + 1.5 * IQR.fa
## [1] 82
# Calculamos el umbral por debajo para valores atípicos leves
Q1.fa - 1.5 * IQR.fa
```

```
## [1] -38
# Calculamos el umbral por encima para valores atípicos extremos
Q1.fa + 3 * IQR.fa
## [1] 142
# Calculamos el umbral por debajo para valores atípicos extremos
Q1.fa - 3 * IQR.fa
## [1] -98
# Todos los valores del muestreo que superen el valor de 142 son outliers
# No hay valores atípicos inferiores al umbral pues los importes todos so
n positivos. El mínimo es 6.

# Comprobamos cuantas observaciones se encuentran por encima de 142.
nrow(red.wines.DT[red.wines.DT$total.sulfur.dioxide > 142,])
## [1] 23
# Procedemos a su eliminación
red.wines.DT <- red.wines.DT[red.wines.DT$total.sulfur.dioxide <= 142,]
nrow(red.wines.DT)
## [1] 1576
# Usar boxplot.stats
boxplot.stats(red.wines.DT$fixed.acidity)
## $stats
## [1]  4.6  7.1  7.9  9.2 12.3
##
## $n
## [1] 1576
##
## $conf
## [1] 7.816421 7.983579
##
## $out
##  [1] 12.8 12.8 15.0 15.0 12.5 13.3 13.4 12.4 12.5 13.8 13.5 12.6 12.5
12.8 12.8
## [16] 14.0 13.7 13.7 12.7 12.5 12.8 12.6 15.6 12.5 13.0 12.5 13.3 12.4
12.5 12.9
## [31] 14.3 12.4 15.5 15.5 15.6 13.0 12.7 13.0 12.7 12.4 12.7 13.2 13.2
13.2 15.9
## [46] 13.3 12.9 12.6 12.6
boxplot.stats(red.wines.DT$volatile.acidity)
## $stats
## [1] 0.12 0.39 0.52 0.64 1.01
##
## $n
## [1] 1576
##
## $conf
## [1] 0.5100501 0.5299499
##
## $out
##  [1] 1.130 1.020 1.070 1.330 1.330 1.040 1.090 1.040 1.185 1.020 1.035
```

```
1.025
## [13] 1.115 1.020 1.020 1.580 1.180 1.040
boxplot.stats(red.wines.DT$citric.acid)
## $stats
## [1] 0.00 0.09 0.25 0.42 0.79
##
## $n
## [1] 1576
##
## $conf
## [1] 0.2368661 0.2631339
##
## $out
## [1] 1
boxplot.stats(red.wines.DT$residual.sugar)
## $stats
## [1] 0.90 1.90 2.20 2.60 3.65
##
## $n
## [1] 1576
##
## $conf
## [1] 2.17214 2.22786
##
## $out
##   [1]  6.10  6.10  4.40 10.70  5.50  5.90  5.90  3.80  5.10  4.65  4.6
5  5.50
##  [13]  5.50  5.50  5.50  7.30  7.20  3.80  5.60  4.00  4.00  4.00  4.0
0  7.00
##  [25]  4.00  4.00  6.40  5.60  5.60 11.00 11.00  4.50  4.80  5.80  5.8
0  3.80
##  [37]  4.40  6.20  4.20  7.90  7.90  3.70  4.50  6.70  6.60  3.70  5.2
0 15.50
##  [49]  4.10  8.30  6.55  6.55  4.60  4.30  5.80  5.15  6.30  4.20  4.2
0  4.60
##  [61]  4.20  4.60  4.30  4.30  7.90  4.60  5.10  5.60  5.60  6.00  7.5
0  4.40
##  [73]  4.25  6.00  3.90  4.20  4.00  4.00  4.00  6.60  6.00  6.00  3.8
0  9.00
##  [85]  4.60  8.80  8.80  5.00  3.80  4.10  5.90  4.10  6.20  8.90  4.0
0  3.90
##  [97]  4.00  8.10  8.10  6.40  6.40  4.70  5.50  5.50  4.30  5.50  3.7
0  6.20
## [109]  5.60  7.80  4.60  5.80  4.10 12.90  4.30  4.80  6.30  4.50  4.5
0  4.30
## [121]  4.30  3.90  3.80  5.40  3.80  6.10  3.90  5.10  5.10  3.90 15.4
0 15.40
## [133]  4.80  5.20  5.20  3.75 13.80 13.80  5.70  4.30  4.10  4.10  4.4
0  3.70
## [145]  6.70 13.90  5.10  7.80
```

```
boxplot.stats(red.wines.DT$chlorides)
## $stats
## [1] 0.041 0.070 0.079 0.090 0.119
##
## $n
## [1] 1576
##
## $conf
## [1] 0.07820401 0.07979599
##
## $out
##   [1] 0.368 0.341 0.172 0.332 0.464 0.401 0.467 0.178 0.146 0.236 0.61
0 0.360
##  [13] 0.270 0.039 0.337 0.263 0.611 0.358 0.343 0.186 0.213 0.214 0.12
1 0.122
##  [25] 0.122 0.128 0.120 0.159 0.124 0.122 0.122 0.174 0.121 0.127 0.41
3 0.152
##  [37] 0.152 0.125 0.200 0.171 0.226 0.226 0.250 0.148 0.122 0.124 0.12
4 0.143
##  [49] 0.222 0.039 0.157 0.422 0.034 0.387 0.415 0.157 0.157 0.243 0.24
1 0.190
##  [61] 0.132 0.126 0.038 0.165 0.145 0.147 0.012 0.012 0.039 0.194 0.13
2 0.161
##  [73] 0.120 0.120 0.123 0.123 0.414 0.216 0.171 0.178 0.369 0.166 0.16
6 0.136
##  [85] 0.132 0.132 0.123 0.123 0.123 0.403 0.137 0.414 0.166 0.168 0.41
5 0.153
##  [97] 0.415 0.267 0.123 0.214 0.214 0.169 0.205 0.205 0.039 0.235 0.23
0 0.038
boxplot.stats(red.wines.DT$free.sulfur.dioxide)
## $stats
## [1]  1  7 13 21 42
##
## $n
## [1] 1576
##
## $conf
## [1] 12.44281 13.55719
##
## $out
##  [1] 50 68 68 43 47 54 46 45 53 52 51 45 57 50 45 48 43 48 43 51 51 52
55 55 48
## [26] 48 66
boxplot.stats(red.wines.DT$total.sulfur.dioxide)
## $stats
## [1]   6  22  37  60 116
##
## $n
## [1] 1576
##
```

```
## $conf
## [1] 35.48761 38.51239
##
## $out
##  [1] 119 119 136 125 140 136 133 134 141 129 128 129 128 121 121 127 1
26 120 120
## [20] 121 119 135 124 124 122 134 124 129 133 142 121 122 125 127 139 1
19 130 122
## [39] 119 135 119 119 141 141 133 131 131 131
boxplot.stats(red.wines.DT$density)
## $stats
## [1] 0.99235 0.99560 0.99673 0.99783 1.00100
##
## $n
## [1] 1576
##
## $conf
## [1] 0.9966412 0.9968188
##
## $out
##  [1] 0.99160 0.99160 1.00140 1.00150 1.00150 1.00180 1.00220 1.00220 1
.00140
## [10] 1.00140 1.00140 1.00140 1.00320 1.00260 1.00140 1.00315 1.00315 1
.00315
## [19] 1.00210 1.00210 0.99170 1.00260 0.99210 0.99154 0.99064 0.99064 1
.00289
## [28] 0.99162 0.99007 0.99007 0.99020 0.99220 0.99150 0.99157 0.99080 0
.99084
## [37] 0.99191 1.00369 1.00369 1.00242 0.99182 1.00242 0.99182
boxplot.stats(red.wines.DT$pH)
## $stats
## [1] 2.93 3.21 3.31 3.40 3.68
##
## $n
## [1] 1576
##
## $conf
## [1] 3.302438 3.317562
##
## $out
##  [1] 3.90 3.75 3.85 2.74 3.69 3.69 2.88 2.86 3.74 2.92 2.92 2.92 3.72
2.87 2.89
## [16] 2.89 2.92 3.90 3.71 3.69 3.69 3.71 3.71 2.89 2.89 3.78 3.70 3.78
4.01 2.90
## [31] 4.01 3.71 2.88 3.72 3.72
boxplot.stats(red.wines.DT$sulphates)
## $stats
## [1] 0.33 0.55 0.62 0.73 0.99
##
## $n
```

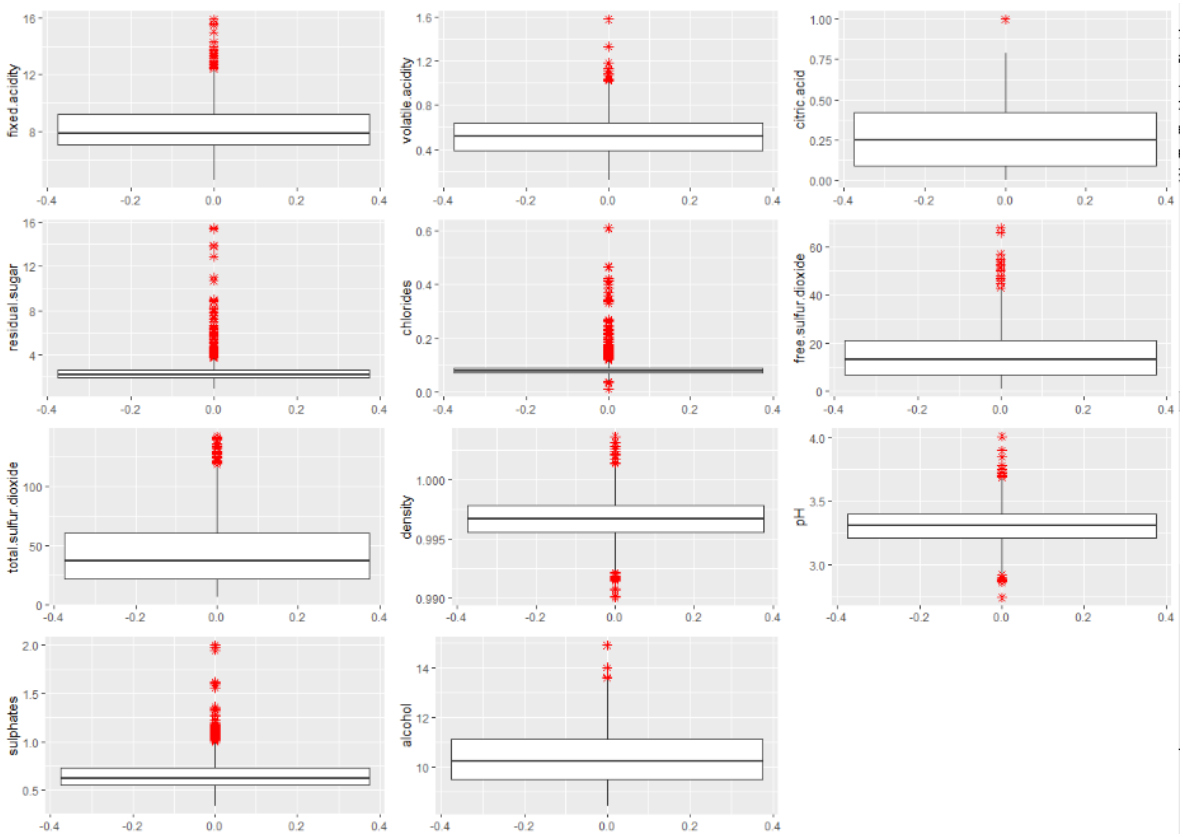
```
## [1] 1576
##
## $conf
## [1] 0.6128361 0.6271639
##
## $out
##  [1] 1.56 1.28 1.08 1.20 1.12 1.28 1.14 1.95 1.22 1.95 1.98 1.31 2.00
1.08 1.59
## [16] 1.02 1.61 1.09 1.26 1.08 1.00 1.36 1.18 1.13 1.04 1.11 1.13 1.07
1.06 1.06
## [31] 1.05 1.06 1.04 1.05 1.02 1.02 1.36 1.36 1.05 1.17 1.62 1.06 1.18
1.07 1.34
## [46] 1.16 1.10 1.15 1.17 1.17 1.33 1.18 1.17 1.03 1.17 1.10 1.01
boxplot.stats(red.wines.DT$alcohol)
## $stats
## [1]  8.4  9.5 10.2 11.1 13.5
##
## $n
## [1] 1576
##
## $conf
## [1] 10.13632 10.26368
##
## $out
##  [1] 14.00000 14.00000 14.00000 14.00000 14.90000 14.00000 13.60000 13
.60000
##  [9] 13.60000 14.00000 14.00000 13.56667 13.60000
```

(2) Otra forma de fijar los outliers es usando los boxplot utilizando la función boxplots.stats() de R, la cual se emplea a continuación. La caja nos representa el borde superior (tercer cuartil) y el inferior (Q1). Entre medias estan el 50% de las observaciones. La altura de la caja es el rango intercuartílico. La linea gruesa dentro de la caja, es la mediana. Por encima y debajo se ven dos límites que son los umbrales para los valores atípicos.

No obstante, si revisamos los demás datos para varios vinos escogidos aleatoriamente de esta web, comprobamos que son valores que perfectamente pueden darse. Es por ello que el manejo de estos valores extremos (los fijados a través de la función boxplot.stats(), salvo los de total.sulfur.dioxide) consistirá en simplemente dejarlos como actualmente están recogidos.

```
# Usando boxplots
grid.newpage()
ggg01 <- ggplot(red.wines.DT, aes(y=fixed.acidity)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
               outlier.size=2)
ggg02 <- ggplot(red.wines.DT, aes(y=volatile.acidity)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
```

```
                outlier.size=2)
ggg03 <- ggplot(red.wines.DT, aes(y=citric.acid)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg04 <- ggplot(red.wines.DT, aes(y=residual.sugar)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg05 <- ggplot(red.wines.DT, aes(y=chlorides)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg06 <- ggplot(red.wines.DT, aes(y=free.sulfur.dioxide)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg07 <- ggplot(red.wines.DT, aes(y=total.sulfur.dioxide)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg08 <- ggplot(red.wines.DT, aes(y=density)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg09 <- ggplot(red.wines.DT, aes(y=pH)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg10 <- ggplot(red.wines.DT, aes(y=sulphates)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
ggg11 <- ggplot(red.wines.DT, aes(y=alcohol)) +
  geom_boxplot(outlier.colour="red", outlier.shape=8,
                outlier.size=2)
grid.arrange(ggg01, ggg02, ggg03, ggg04,
             ggg05, ggg06, ggg07, ggg08,
             ggg09, ggg10, ggg11, ncol=3)
```




### 2.3.4. Exportación de los datos preprocesados

Una vez que hemos acometido sobre el conjunto de datos inicial los procedimientos de integración, validación y limpieza anteriores, procedemos a guardar estos en un nuevo fichero denominado RedWines_data_clean.csv:

```
# Exportación de los datos preprocesados
write.csv(red.wines.DT, "RedWines_data_clean.csv")
```

## 2.4. Análisis de los datos

### 2.4.1. Selección de los grupos de datos a analizar

A continuación, se seleccionan los grupos dentro de nuestro conjunto de datos que pueden resultar interesantes para analizar y/o comparar. No obstante, como se verá en el apartado consistente en la realización de pruebas estadísticas, no todos se utilizarán.

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

Para la comprobación de que los valores que toman nuestras variables cuantitativas provienen de una población distribuida normalmente, utilizaremos la prueba de normalidad de Anderson-Darling. Así, se comprueba que para que cada prueba se obtiene un p-valor superior al nivel de significación prefijado $\alpha = 0, 05$. Si esto se cumple, entonces se considera que variable en cuestión sigue una distribución normal.

```
# Comprobación de la normalidad y homogeneidad de la varianza
alpha = 0.05
col.names = colnames(red.wines.DT)

for (i in 1:ncol(red.wines.DT)) {
  if (i == 1) cat("Variables que no siguen una distribución normal:\n")
  if (is.integer(red.wines.DT[,i]) | is.numeric(red.wines.DT[,i])) {
    p_val = ad.test(red.wines.DT[,i])$p.value
    if (p_val < alpha) {
      cat(col.names[i])
      # Format output
      if (i < ncol(red.wines.DT)) cat(", ")
      if (i %% 3 == 0) cat("\n")
    }
  }
}
## Variables que no siguen una distribución normal:
## fixed.acidity, volatile.acidity, citric.acid,
## residual.sugar, chlorides, free.sulfur.dioxide,
## total.sulfur.dioxide, density, pH,
## sulphates, alcohol, quality
```

Comprobamos que todas las variables del dataset no siguen una distribución normal.

## 2.5. Pruebas estadísticas

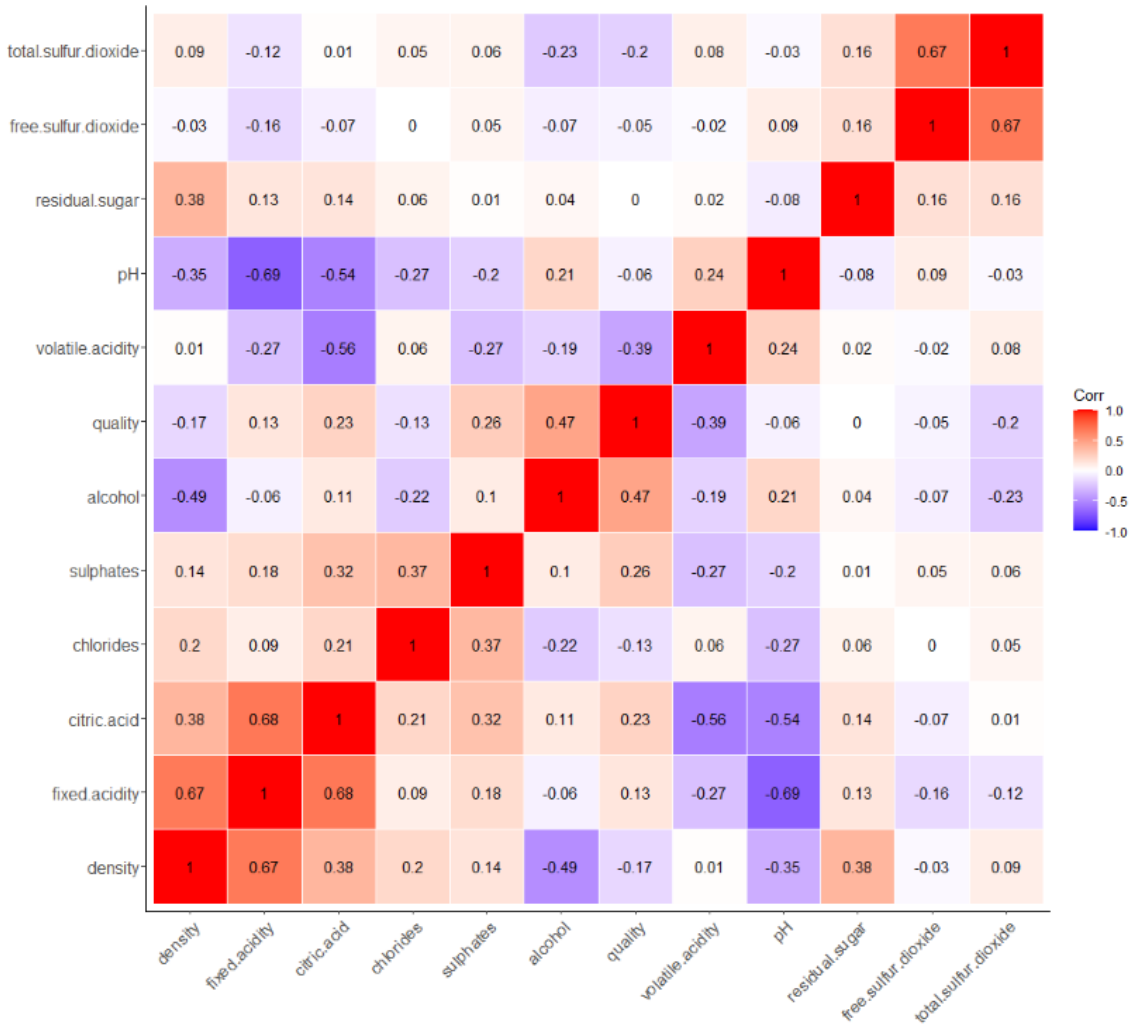
### 2.5.1. ¿Qué variables cuantitativas influyen más en el precio?

En primer lugar, procedemos a realizar un análisis de correlación entre las distintas variables para determinar cuáles de ellas ejercen una mayor influencia sobre la calidad final del vino. Para ello, se utilizará el coeficiente de correlación de Spearman, puesto que hemos comprobado que todos los datos no siguen una distribución normal.

```
# Análisis de correlación
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")
# Calcular el coeficiente de correlación para cada variable cuantitativa
# con respecto a la variables dependiente quality.
for (i in 1:(ncol(red.wines.DT) - 1)) {
  if (is.integer(red.wines.DT[,i]) | is.numeric(red.wines.DT[,i])) {
    spearman_test = cor.test(red.wines.DT[,i],
                             red.wines.DT[,length(red.wines.DT)],
                             method = "spearman", exact = FALSE)
    corr_coef = spearman_test$estimate
    p_val = spearman_test$p.value
    # Add row to matrix
    pair = matrix(ncol = 2, nrow = 1)
    pair[1][1] = corr_coef
    pair[2][1] = p_val
    corr_matrix <- rbind(corr_matrix, pair)
```

```
    rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(red.wines.DT)[i]
  }
}

print(corr_matrix)
##                          estimate      p-value
## fixed.acidity         0.12279250 1.011600e-06
## volatile.acidity     -0.37483098 9.551300e-54
## citric.acid           0.21414921 8.353412e-18
## residual.sugar        0.03088739 2.203817e-01
## chlorides            -0.18067697 4.954896e-13
## free.sulfur.dioxide  -0.05112446 4.242777e-02
## total.sulfur.dioxide -0.19161013 1.695207e-14
## density              -0.16902619 1.443491e-11
## pH                   -0.04619494 6.674158e-02
## sulphates             0.38573761 4.497827e-57
## alcohol               0.47289965 1.288115e-88
matriz.Correlacion <- cor(red.wines.DT)
round(matriz.Correlacion, 1)
##                        pH sulphates alcohol quality
## fixed.acidity        -0.7       0.2    -0.1     0.1
## volatile.acidity      0.2      -0.3    -0.2    -0.4
## citric.acid          -0.5       0.3     0.1     0.2
## residual.sugar       -0.1       0.0     0.0     0.0
## chlorides            -0.3       0.4    -0.2    -0.1
## free.sulfur.dioxide   0.1       0.1    -0.1     0.0
## total.sulfur.dioxide  0.0       0.1    -0.2    -0.2
## density              -0.4       0.1    -0.5    -0.2
## pH                    1.0      -0.2     0.2    -0.1
## sulphates            -0.2       1.0     0.1     0.3
## alcohol               0.2       0.1     1.0     0.5
## quality              -0.1       0.3     0.5     1.0
matriz.PValues <- rcorr(as.matrix(red.wines.DT))
matriz.PValues
# Chart correlaciones entre observaciones
ggcorrplot(matriz.Correlacion, hc.order = TRUE,
           outline.color = "white", lab = TRUE, ggtheme = ggplot2::theme_
classic())
```




Así, identificamos cuáles son las variables más correlacionadas con calidad en función de su proximidad con los valores -1 y +1. Teniendo esto en cuenta, queda patente cómo la variable más relevante en la fijación de la calidad es el alcohol. Es decir, las mayores correlaciones tienen valores inferiores al 0,5. Destacamos volatile.acidity, sulphates y alcohol con mayores correlaciones.

### 2.5.2. Modelo de regresión lineal

Tal y como se planteó en los objetivos de la actividad, resultará de mucho interés poder realizar predicciones sobre la calidad de un vino dadas sus características. Así, se calculará un modelo de regresión lineal utilizando regresores cuantitativos con el que poder realizar las predicciones de la calidad de los vinos. Para obtener un modelo de regresión lineal considerablemente eficiente, lo que haremos será obtener varios modelos de regresión utilizando las variables que estén más correladas con respecto al precio, según la tabla obtenido en el apartado 2.5.1. Así, de entre todos los modelos que tengamos, escogeremos el mejor utilizando como criterio aquel que presente un mayor coeficiente de determinación (R2).

```
lm.fit.mul.RW <- lm(quality~., data = red.wines.DT)
lm.fit.mul.RW
##
## Call:
## lm(formula = quality ~ ., data = red.wines.DT)
##
## Coefficients:
##          (Intercept)         fixed.acidity      volatile.acidity
##            10.980095              0.016100             -1.102656
##          citric.acid        residual.sugar             chlorides
##            -0.200897              0.008502             -1.866405
##  free.sulfur.dioxide  total.sulfur.dioxide               density
##             0.005517             -0.004084             -6.704518
##                   pH             sulphates               alcohol
##            -0.437026              0.941083              0.279100
summary(lm.fit.mul.RW)
##
## Call:
## lm(formula = quality ~ ., data = red.wines.DT)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71368 -0.36977 -0.05038  0.44981  2.07961
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          10.9800949 21.6596200   0.507   0.6123
## fixed.acidity         0.0161004  0.0262962   0.612   0.5404
## volatile.acidity     -1.1026556  0.1236439  -8.918  < 2e-16 ***
## citric.acid          -0.2008968  0.1488434  -1.350   0.1773
## residual.sugar        0.0085019  0.0156465   0.543   0.5869
## chlorides            -1.8664051  0.4219950  -4.423 1.04e-05 ***
## free.sulfur.dioxide   0.0055167  0.0022491   2.453   0.0143 *
## total.sulfur.dioxide -0.0040839  0.0008119  -5.030 5.47e-07 ***
## density              -6.7045182 22.1000905  -0.303   0.7616
## pH                   -0.4370262  0.1930410  -2.264   0.0237 *
## sulphates             0.9410834  0.1155112   8.147 7.55e-16 ***
## alcohol               0.2791000  0.0267757  10.424  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6502 on 1564 degrees of freedom
## Multiple R-squared:  0.3587, Adjusted R-squared:  0.3542
## F-statistic: 79.54 on 11 and 1564 DF,  p-value: < 2.2e-16
```

El modelo con todas las variables introducidas como predictores tiene un R-Squared de (0.3542), es capaz de explicar el 35,42% de la variabilidad observada en la calidad de un vino. La función summary() nos proporciona información de los coeficientes estimados por el modelo, es decir, los parámetros ocultos, betas de la ecuación. Estas son la ordenada al origen (Intercept) y las pendientes estimadas para cada variable.

### 2.5.3. Selección de los mejores predictores

Procedemos a considerar aquellas variables dependientes con p-value cercanos a cero, es decir, las que permiten rechazar la hipótesis nula (coeficiente igual a cero) y con ellas diseñamos diferentes modelos de regresión. Buscamos el mejor modelo de regresión, aquel con mayor R-Squared.

```
# Regresores cuantitativos con mayor coeficiente
# de correlación con respecto a la variable quality
# volatile.acidity, chlrides, free.sulfur.dioxide, total.sulfur.dioxide,
pH, sulphates y alcohol




# Generación de varios modelos
lm.fit.mul.RW.01 <- lm(quality ~ volatile.acidity + chlorides + free.sulf
ur.dioxide +
                         total.sulfur.dioxide + pH + sulphates + alcohol,
data = red.wines.DT)
lm.fit.mul.RW.02 <- lm(quality ~ volatile.acidity + chlorides + free.sulf
ur.dioxide +
                         total.sulfur.dioxide + pH + sulphates, data = re
d.wines.DT)
lm.fit.mul.RW.03 <- lm(quality ~ volatile.acidity + chlorides + free.sulf
ur.dioxide +
                         total.sulfur.dioxide + pH, data = red.wines.DT)
lm.fit.mul.RW.04 <- lm(quality ~ volatile.acidity + chlorides + free.sulf
ur.dioxide +
                         total.sulfur.dioxide, data = red.wines.DT)
lm.fit.mul.RW.05 <- lm(quality ~ volatile.acidity + alcohol, data = red.w
ines.DT)

# Tabla con los coeficientes de determinación de cada modelo
tabla.coeficientes <- matrix(c(1, summary(lm.fit.mul.RW.01)$r.squared,
                               2, summary(lm.fit.mul.RW.02)$r.squared,
                               3, summary(lm.fit.mul.RW.03)$r.squared,
                               4, summary(lm.fit.mul.RW.04)$r.squared,
                               5, summary(lm.fit.mul.RW.05)$r.squared),
                             ncol = 2, byrow = TRUE)
colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
##      Modelo       R^2
## [1,]      1 0.3579059
## [2,]      2 0.2477355
## [3,]      3 0.1941108
## [4,]      4 0.1939258
## [5,]      5 0.3121268
```

Otra forma de determinar el mejor modelo es a través de la función step(). En este caso se van a emplear la estrategia de stepwise mixto. El valor matemático empleado para determinar la calidad del modelo va a ser Akaike(AIC).

```
step(object = lm.fit.mul.RW, direction = "both", trace = 1)
## Start:  AIC=-1345.05
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.su
gar +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + sulphates + alcohol
##
##                        Df Sum of Sq    RSS     AIC
## - density               1     0.039 661.17 -1347.0
## - residual.sugar        1     0.125 661.26 -1346.8
## - fixed.acidity         1     0.158 661.29 -1346.7
## - citric.acid           1     0.770 661.91 -1345.2
## <none>                              661.14 -1345.0
## - pH                    1     2.167 663.30 -1341.9
## - free.sulfur.dioxide   1     2.543 663.68 -1341.0
## - chlorides             1     8.269 669.40 -1327.5
## - total.sulfur.dioxide  1    10.695 671.83 -1321.8
## - sulphates             1    28.058 689.19 -1281.5
## - volatile.acidity      1    33.619 694.76 -1268.9
## - alcohol               1    45.930 707.07 -1241.2
##
## Step:  AIC=-1346.96
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.su
gar +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     pH + sulphates + alcohol
##
##                        Df Sum of Sq    RSS     AIC
## - residual.sugar        1     0.086 661.26 -1348.8
## - fixed.acidity         1     0.154 661.33 -1348.6
## - citric.acid           1     0.774 661.95 -1347.1
## <none>                              661.17 -1347.0
## + density               1     0.039 661.14 -1345.0
## - free.sulfur.dioxide   1     2.638 663.81 -1342.7
## - pH                    1     3.692 664.87 -1340.2
## - chlorides             1     8.444 669.62 -1329.0
## - total.sulfur.dioxide  1    11.004 672.18 -1322.9
## - sulphates             1    29.180 690.35 -1280.9
## - volatile.acidity      1    34.387 695.56 -1269.0
## - alcohol               1   110.504 771.68 -1105.4
##
## Step:  AIC=-1348.75
## quality ~ fixed.acidity + volatile.acidity + citric.acid + chlorides +
##     free.sulfur.dioxide + total.sulfur.dioxide + pH + sulphates +
##     alcohol
##
```

```
##                         Df Sum of Sq    RSS     AIC
## - fixed.acidity          1     0.176 661.44 -1350.3
## - citric.acid            1     0.739 662.00 -1349.0
## <none>                               661.26 -1348.8
## + residual.sugar         1     0.086 661.17 -1347.0
## + density                1     0.000 661.26 -1346.8
## - free.sulfur.dioxide    1     2.742 664.00 -1344.2
## - pH                     1     3.661 664.92 -1342.0
## - chlorides              1     8.373 669.63 -1330.9
## - total.sulfur.dioxide   1    10.918 672.18 -1324.9
## - sulphates              1    29.096 690.36 -1282.9
## - volatile.acidity       1    34.341 695.60 -1271.0
## - alcohol                1   111.938 773.20 -1104.3
##
## Step:  AIC=-1350.33
## quality ~ volatile.acidity + citric.acid + chlorides + free.sulfur.dio
xide +
##     total.sulfur.dioxide + pH + sulphates + alcohol
##
##                         Df Sum of Sq    RSS     AIC
## - citric.acid            1     0.568 662.01 -1351.0
## <none>                               661.44 -1350.3
## + fixed.acidity          1     0.176 661.26 -1348.8
## + residual.sugar         1     0.108 661.33 -1348.6
## + density                1     0.102 661.34 -1348.6
## - free.sulfur.dioxide    1     2.857 664.29 -1345.5
## - pH                     1     6.405 667.84 -1337.1
## - chlorides              1     9.660 671.10 -1329.5
## - total.sulfur.dioxide   1    12.306 673.74 -1323.3
## - sulphates              1    29.489 690.93 -1283.6
## - volatile.acidity       1    35.169 696.61 -1270.7
## - alcohol                1   112.315 773.75 -1105.2
##
## Step:  AIC=-1350.98
## quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol
##
##                         Df Sum of Sq    RSS     AIC
## <none>                               662.01 -1351.0
## + citric.acid            1     0.568 661.44 -1350.3
## + residual.sugar         1     0.045 661.96 -1349.1
## + density                1     0.005 662.00 -1349.0
## + fixed.acidity          1     0.005 662.00 -1349.0
## - free.sulfur.dioxide    1     3.276 665.28 -1345.2
## - pH                     1     6.076 668.08 -1338.6
## - chlorides              1    10.737 672.74 -1327.6
## - total.sulfur.dioxide   1    13.335 675.34 -1321.5
## - sulphates              1    29.064 691.07 -1285.3
## - volatile.acidity       1    41.206 703.21 -1257.8
## - alcohol                1   113.587 775.59 -1103.4
```

```
##
## Call:
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.diox
ide +
##     total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wines.
DT)
##
## Coefficients:
##          (Intercept)      volatile.acidity             chlorides
##              4.403485             -1.014597             -2.014398
##  free.sulfur.dioxide  total.sulfur.dioxide                    pH
##              0.006130             -0.004335             -0.449961
##            sulphates               alcohol
##              0.925101              0.280698
```

El mejor modelo resultante del proceso de selección ha sido:

```
# Mejor modelo de regresión del proceso de selección
lm.fit.mul.RW.best <- lm(quality ~ volatile.acidity + chlorides + free.su
lfur.dioxide +
                           total.sulfur.dioxide + pH + sulphates +
                           alcohol, data = red.wines.DT)
summary(lm.fit.mul.RW.best)
##
## Call:
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.diox
ide +
##     total.sulfur.dioxide + pH + sulphates + alcohol, data = red.wines.
DT)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71896 -0.37195 -0.04845  0.46062  2.07973
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.4034846  0.4050377  10.872  < 2e-16 ***
## volatile.acidity     -1.0145967  0.1026995  -9.879  < 2e-16 ***
## chlorides            -2.0143976  0.3994574  -5.043 5.12e-07 ***
## free.sulfur.dioxide   0.0061301  0.0022005   2.786 0.005405 **
## total.sulfur.dioxide -0.0043346  0.0007713  -5.620 2.26e-08 ***
## pH                   -0.4499610  0.1186104  -3.794 0.000154 ***
## sulphates             0.9251010  0.1114986   8.297 2.27e-16 ***
## alcohol               0.2806981  0.0171133  16.402  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6498 on 1568 degrees of freedom
## Multiple R-squared:  0.3579, Adjusted R-squared:  0.355
## F-statistic: 124.9 on 7 and 1568 DF,  p-value: < 2.2e-16
```

Finalmente, efectuamos una predicción de la calidad de un vino en base a determinados valores de los atributos.

```
# Predicción de la calidad de los vinos tintos
newdata <- data.frame(
  volatile.acidity = 0.5,
  chlorides = 0.06,
  free.sulfur.dioxide = 10,
  total.sulfur.dioxide = 45,
  pH = 3,
  sulphates = 0.5,
  alcohol = 10)


predict(lm.fit.mul.RW.01, newdata)
##        1
## 5.561213
```

### 2.5.4. Contraste de hipótesis

Otra prueba estadística que se aplicará consistirá en un contraste de hipótesis sobre dos muestras para determinar si la calidad del vino es superior dependiendo de la cantidad de pH de que se trate (mayor o menor de la media, situada en 3.31). Para ello, tendremos dos muestras: la primera de ellas se corresponderá con los vinos con un pH por debajo de la media y, la segunda, con aquellos que presentan una cantidad de pH por encima del promedio.

Para las muestras debemos discretizar las observaciones del atributo pH. Se ha creado un nuevo atributo discreto, “segment_pH”.

Se debe destacar que un test paramétrico como el que a continuación se utiliza necesita que los datos sean normales, si la muestra es de tamaño inferior a 30. Como en nuestro caso, n > 30, el contraste de hipótesis siguiente es válido (aunque podría utilizarse un test no paramétrico como el de Mann-Whitney, que podría resultar ser más eficiente para este caso).

Así, se plantea el siguiente contraste de hipótesis de dos muestras sobre la diferencia de medias, el cual es unilateral atendiendo a la formulación de la hipótesis alternativa:

$H0 : \mu1 - \mu2 = 0$
$H1 : \mu1 - \mu2 < 0$

donde $\mu1$ es la media de la población de la que se extrae la primera muestra y $\mu2$ es la media de la población de la que extrae la segunda. Así, tomaremos $\alpha = 0,05$.

```
```{r message= FALSE, warning=FALSE}
# Contraste de hipótesis
min(red.wines.DT$pH)
max(red.wines.DT$pH)
```
```

```
mean.pH <- mean(red.wines.DT$pH)
sum(red.wines.DT$pH < mean.pH)

red.wines.DT["segment_pH"] <- cut(red.wines.DT$pH,
            breaks = c(0, mean.pH, 4),
            labels = c("< mean.pH", ">= mean.pH"))

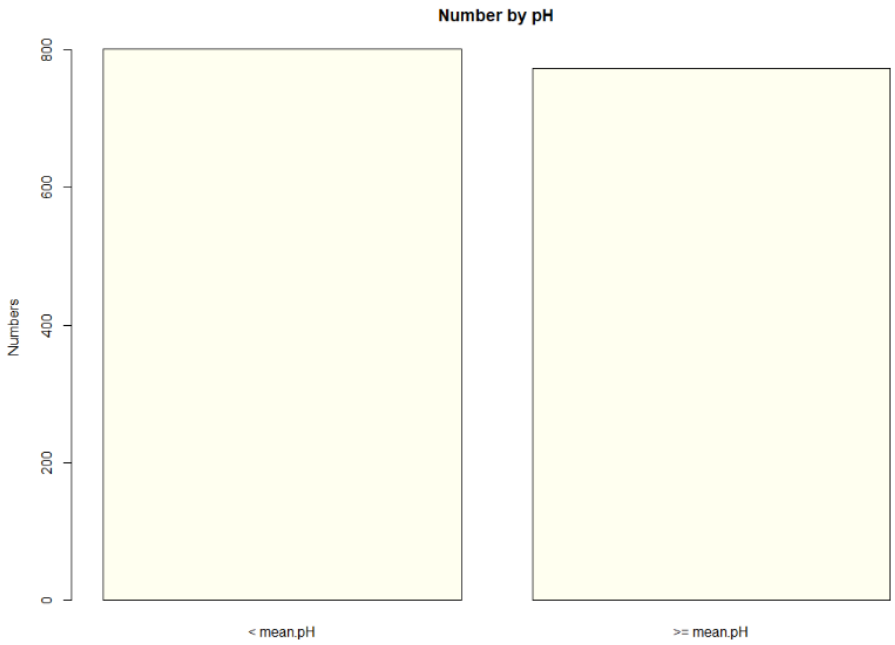
head(red.wines.DT)

plot(red.wines.DT$segment_pH, main="Number by pH",
     xlab="pH", ylab="Numbers",col = "ivory")
red.wines.LM.quality <- red.wines.DT[red.wines.DT$segment_pH == "< mean.p
H",]$quality
red.wines.BEM.quality <- red.wines.DT[red.wines.DT$segment_pH == ">= mean
.pH",]$quality
t.test(red.wines.LM.quality, red.wines.BEM.quality, alternative = "less")
```

```
## [1] 2.74
## [1] 4.01
## [1] 801

##    Welch Two Sample t-test

## data:  red.wines.LM.quality and red.wines.BEM.quality
## t = 2.0917, df = 1570.6, p-value = 0.9817
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##       -Inf 0.1523645
## sample estimates:
## mean of x mean of y
##  5.681648  5.596378
```




Puesto que obtenemos un p-value (0,9817) mayor que el valor de significación fijado (0,05), aceptamos la hipótesis nula. Por tanto, podemos concluir que, efectivamente, la calidad del vino es superior si la cantidad de pH es inferior a la media.

## 2.6. Conclusiones

Destacar las diferentes fases de análisis realizadas en la práctica, desde la necesidad de limpiar y adecuar los datos, hasta las pruebas estadísticas para determinar la normalidad o no de las distribuciones de las diferentes variables como la ideoneidad de los atributos en el modelo de regresión. Lamentablemente, los resultados obtenidos no nos aportan un modelo de regresión razonable que al menos explique la variabilidad de la variable dependiente en un porcentaje superior al azar. Tan solo se ha logrado obtener un modelo que explique como máximo el 35,79% de la variabilidad observada. En resumen, práctica muy interesante y desafiante que nos permite profundizar en los metodos de predicción de una variable dependiente mediante modelos lineales y en base a una serie de predictores o variables independientes que explican la variable objetivo.