UOC – UNIVERSITAT OBERTA CATALUNYA
Asignatura: Tipología y ciclo de vida de los datos
Actividad: Práctica 2 – Limpieza y análisis de datos

Alumno: José Cortés Novales

Enunciado

Competencias

En esta práctica se desarrollan las siguientes competencias del Máster de Data Science:

- 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.

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 Tipología y ciclo de vida de los datos Práctica 2 pág 2 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.

Descripción de la Práctica a realizar

El objetivo de esta actividad será el tratamiento de un dataset, que puede ser el creado en la práctica 1 o bien cualquier dataset libre disponible en Kaggle (https://www.kaggle.com). Algunos ejemplos de dataset con los que podéis trabajar son:

- Red Wine Quality (https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009)
- Titanic: Machine Learning from Disaster (https://www.kaggle.com/c/titanic)

El último ejemplo corresponde a una competición activa de Kaggle de manera que, opcionalmente, podéis aprovechar el trabajo realizado durante la práctica para entrar en esta competición.

Siguiendo las principales etapas de un proyecto analítico, las diferentes tareas a realizar (y justificar) son las siguientes:

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

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

3. Limpieza de los datos.
    3.1. ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?
    3.2. Identificación y tratamiento de valores extremos.

4. Análisis de los datos.
    4.1. Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).
    4.2. Comprobación de la normalidad y homogeneidad de la varianza.
    4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

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

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

Resolución

Previo: BBDD elegida

Descargas de varios datasets o conjuntos de datos desde Kaggle. Entre ellas, las siguientes:

- Multidimensional poverty measures
- Spotify song attributes
- World city population
- Laptop prices
- Star wars script sentences
- Datos sobre Airbnb en Madrid
- Ventas de Champagne
- Wine Reviews
- Red Wine Quality

La búsqueda fue hecha sobre varios temas que me atraían la atención (Wine, Ordenadores, Música o Pobreza) o sobre temas curiosos que iban apareciendo al hacer búsquedas sobre estos temas. Así apareció la de Airbnb o la de Star wars.

La mayoría de las BBDD las descarté porque estaba en algún formato que desconocía (por ejemplo, sqlite) o porque las variables contenidas en el csv eran frases que luego son difícilmente analizables (por ejemplo, los guiones de star wars o las características sobre los pisos incluidos en Airbnb).

Así que, me centré en las BBDD relacionadas con temas del vino, que además de ser un tema que me interesa, está relacionado con mi trabajo (aunque yo me dedico a temas económicos y no de elaboración).

De las tres encontradas, que son las últimas de la lista inicial, la de champagne la descarté por básica, ya que era una lista de ventas de ese tipo de vino por año. La de las wine reviews era muy interesante a priori, pero únicamente facilitaba el análisis en base a las puntuaciones de los vinos y las zonas o añadas. Así que me decidí por la BBDD de Red Wine Quality que parecía que ofrecía más oportunidades de análisis y limpieza de la BBDD, que además era una de las opciones propuestas en el enunciado.

El dataset del Titanic no lo escogí porque en una de las asignaturas niveladoras para la entrada en el Master creo recordar que ya habíamos trabajado sobre un dataset con esa temática.

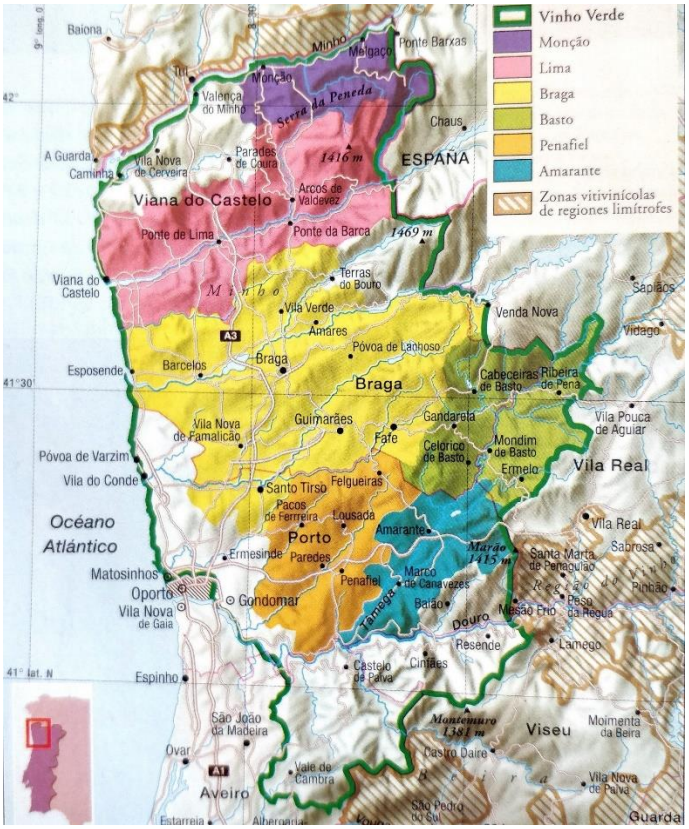
Descripción del dataset

Según la descripción que se ofrece en el contexto de Kaggle sobre el dataset, se recopilan dato de las variantes tinta y blanca del vino verde portugués, aunque yo creo que los datos son sólo del tinto, por el nombre del fichero y por otras variables (luego se confirma que así es, más abajo en el apartado “About this file”; en el original de la UCI si están los dos ficheros).

El conjunto de datos objeto de análisis se ha obtenido a partir del enlace mencionado en Kaggle y está constituido por 12 características (columnas) que presentan 1.599 vinos (filas o registros).

Se recogen diversas características de esos vinos, que listaremos a continuación, ya que son las variables con las que vamos a trabajar, a fin de conocer si un vino va a ser mejor que otro o no. Son características físicas de los vinos relacionadas con su composición química (acidez, volatilidad, …), que son tratadas como inputs en el fichero y otra característica que es la sensorial, que es el output de las características anteriores. La idea es que cuando mejor sea esa salida, habrá que considerar las características anteriores en la elaboración del vino para que este obtenga una mejor sensorial.

Es una pena, aunque comprensible, que debido a motivos de privacidad no se tengan datos de las marcas, el tipo de uva o los precios de los vinos, como se advierte en la descripción de los datos. De esa manera, podríamos estudiar, por ejemplo, si los vinos más caros son mejores que los más baratos o no. O si hay un tipo de uva mejor para la obtención de los mejores vinos. O si los productores grandes tienen más calidad que los pequeños, aunque su producción suele estar muy fragmentada y es el segundo vino que más se exporta en Portugal, después del Oporto.



El Vinho verde se produce entre las regiones del Miño y el Duero, como se puede ver en el mapa de la página anterior.

El fichero está en Kaggle, pero el autor menciona que lo ha traído del repositorio de UCI (UCI Machine Learning Repository: Wine Quality Data Set) y que los datos originales vienen de una investigación de Paulo Cortez de la Universidad de Guimaraes, cuya referencia recojo en recursos, puesto que también lo he consultado.

Las características recogidas en el fichero son las siguientes. Las once primeras corresponden las que se han obtenido mediante test de tipo fisicoquímico y la duodécima puntúa la calidad del vino, entre 0 y 10, basándose en una prueba sensorial,

1. ***Fixed acidity***. La acidez en los vinos se suele medir mediante dos variables. La acidez fija y la acidez volátil (que se mide en la siguiente característica). La acidez fija se mide en gramos de ácido tartárico por litro o por decímetros cúbicos (como ocurre en este caso). Un decímetro cúbico es equivalente a un litro. El ácido tartárico es un compuesto natural que se obtiene de las uvas, normalmente de los posos de la pulpa y el tartrato. Tiene un papel fundamental en el mantenimiento de la estabilidad química del vino, en el color y en el sabor del mismo.
2. ***Volatile acidity***. La acidez volátil se mide en gramos de ácido acético por litro o por decímetro cúbico (también se usa este segundo valor). El ácido acético se suele originar de la oxidación del alcohol, normalmente en la fermentación del vino. En ocasiones se afirma que, a menor acidez volátil, mejor es la calidad de un vino, ya que cuando este compuesto químico aumenta el sabor y olor del vino se tornan más agrios y parecidos al vinagre (aunque tampoco hay un consenso total sobre que el ácido acético sea el único responsable de esta transformación).
3. ***Citric acid***. Al igual que las dos anteriores, el ácido cítrico se mide en gramos por decímetro cúbico. Se encuentra en cantidades pequeñas en la uva. A veces se permite que lo añadan los enólogos con el fin de eliminar el exceso de hierro o cobre en el vino. Normalmente, cuando se añade se hace después de la primera fermentación de la uva, porque de lo contrario se convierte en ácido málico.
4. ***Residual sugar***. El azúcar residual se mide en gramos por decímetro cúbico. El azúcar está contenido en la uva y cuando las levaduras actúan sobre las uvas presadas, se comen el azúcar para convertirlo en alcohol. El azúcar que no son capaces de procesar esas levaduras es el azúcar residual. Le da al vino el dulzor que se puede apreciar al probarlo.
5. ***Chlorides***. Los cloruros se miden también en gramos por decímetro cúbico. Este componente llega al vino porque se suele encontrar en el suelo en el que crecen las vides. La presencia de cloruro está relacionada con el llamado cuerpo del vino, su consistencia y el sabor.
6. ***Free sulfur dioxide***. El dióxido de azufre se mide en miligramos por decímetro cúbico. Este compuesto se suele emplear para evitar la oxidación del vino y favorecer su conservación. Puede aparecer de manera natural en la fermentación o añadirse posteriormente. Suelen conocerse también como sulfitos.
7. ***Total sulfur dioxide***. También se mide en miligramos por decímetro cúbico. Suele incluir el dióxido de azufre libre y otros tipos de dióxido de azufre, como el combinado. Es importante medir el componente total porque, aunque es un elemento conservador, en cantidades altas puede llegar a ser peligroso en la ingesta (por ejemplo, por la producción de diarreas o dolores gástricos).

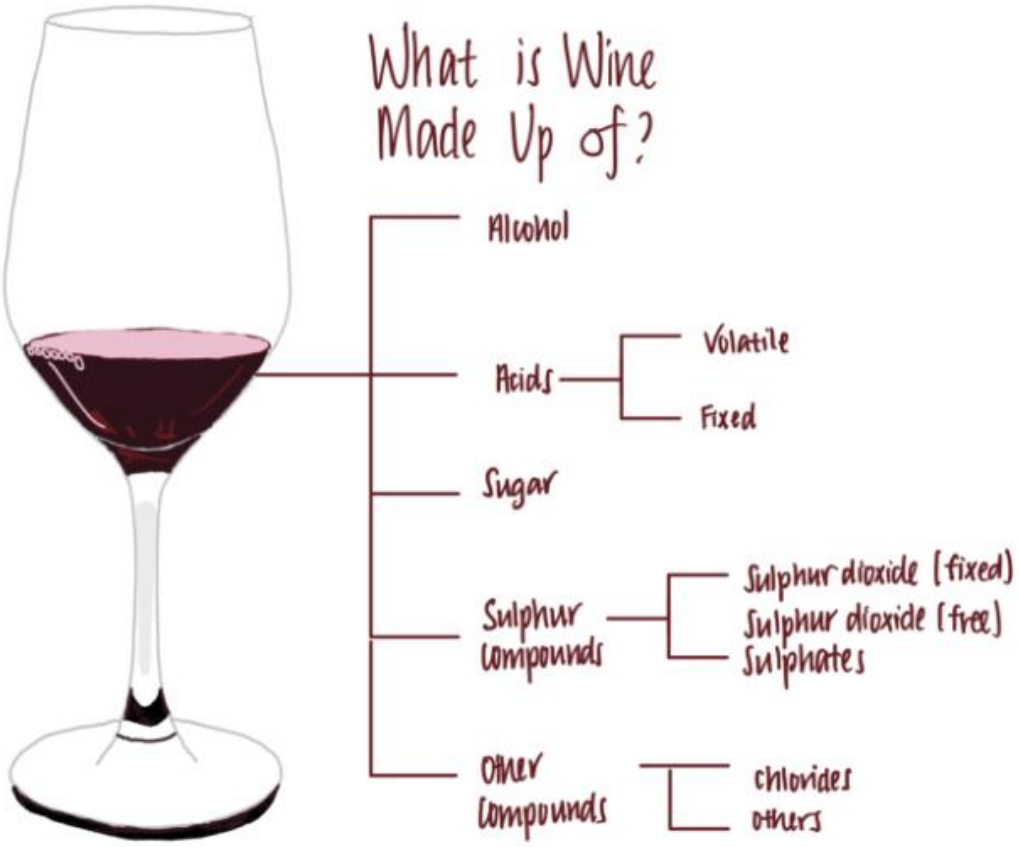
8. ***Density***. La densidad se mide en gramos por centímetro cúbico. Va comprobando el paso del azúcar a alcohol y, por lo tanto, el grado alcohólico, desde el mosto al vino. En el mosto es mayor la densidad y en el vino va bajando.
9. ***pH***. El pH tiene una escala propia de medición. Se suele decir que un vino tiene un nivel de pH de x. Sirve para medir su acidez (no sólo en el vino, se usa para medir ácidos/bases). Si el pH es elevado el vino será menos ácido y viceversa. Se mide en una escala de cero a catorce y en los vinos los valores suelen estar entre 2.9 y 4.2.
10. ***Sulphates***. Se mide en gramos de sulfito de potasio por decímetros cúbicos. Suele emplearse como conservante y se añade artificialmente. Aunque es verdad que la uva naturalmente tiene potasio.
11. ***Alcohol***. Se mide en grados alcohólicos. En los vinos suele estar entre los 12 y los 14 grados. Aunque hay vinos que presentan mayor graduación alcohólica y con el aumento de las temperaturas se está potenciando este fenómeno, buscando los agricultores viñedos en mayor altura o clones que aguanten mejor la temperatura. Como ya hemos explicado antes, en la fermentación las levaduras convierten en alcohol el azúcar.
12. ***Quality*** (score between 0 and 10). Se hacen catas a ciegas (catas en las que no se conoce ningún dato del vino) por tres catadores diferentes y se hace la mediana entre esas tres catas. La escala va del cero (para los vinos perores) al 10 (para los vinos mejores).

Resumen de las unidades de mediación de las variables input,

Table 1
The physicochemical data statistics per wine type.

Attribute (units)
Fixed acidity (g(tartaric acid)/dm^3)
Volatile acidity (g(acetic acid)/dm^3)
Citric acid (g/dm^3)
Residual sugar (g/dm^3)
Chlorides (g(sodium chloride)/dm^3)
Free sulfur dioxide (mg/dm^3)
Total sulfur dioxide (mg/dm^3)
Density (g/cm^3)
pH
Sulphates (g(potassium sulphate)/dm^3)
Alcohol (vol.%)

Y resumen gráfico de las variables mencionadas



The composition of wine. Illustration by author

Integración y selección de los datos de interés a analizar

A partir de este conjunto de datos se plantea la problemática de determinar qué variables influyen más sobre la calidad de un vino. Además, a partir de esas variables, se podrán crear modelos de regresión que permitan predecir la calidad de un vino en base a sus características fisicoquímicas.

Estos análisis adquieren una gran relevancia en la elaboración del vino verde, puesto que se puede planificar la elaboración de vinos a medida que permitan un incremento mayor en el nivel de exportación o que incluso aumente el consumo interno de este tipo de vino. Además, se podrán optimizar recursos en la viña o mejorar procesos en la fermentación de cara a obtener vinos más apreciados por los consumidores.

Aunque podríamos quitar alguna variable que parece que aporta menor información, como la densidad, ya que, al ser todos vinos, deberán moverse en valores parecidos, sin que ello tenga efecto sobre la calidad del vino. O el total de dióxido de azufre, que es un valor que se mide para la salubridad, pero no para la calidad del vino, aunque sobrepasar un determinado nivel implicaría que ese vino no es apto para el mercado, así que sería interesante medirlo por otras circunstancias.

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. Para ello, abrimos un nuevo proyecto en R studio y cargamos el fichero mediante la función read.csv():

```
# Lectura de datos y prueba de las primera cinco líneas y columnas
vino <- read.csv("~/PRA2_Datos/winequality-red.csv ", header = TRUE)
head(vino[,1:5])
```

```
##  fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1         7.4             0.70        0.00            1.9     0.076
## 2         7.8             0.88        0.00            2.6     0.098
## 3         7.8             0.76        0.04            2.3     0.092
## 4        11.2             0.28        0.56            1.9     0.075
## 5         7.4             0.70        0.00            1.9     0.076
## 6         7.4             0.66        0.00            1.8     0.075
```

Mediante la función str() vemos las variables cargadas y su naturaleza

```
#Análisis de las variables cargadas
str(vino)
```

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

```
#También podemos utilizar la función summary() para ver las principales medidas estadísticas de las variables
summary(vino)
```

```
## fixed.acidity   volatile.acidity  citric.acid    residual.sugar
## Min.   : 4.60   Min.   :0.1200   Min.   :0.000   Min.   : 0.900
## 1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090   1st Qu.: 1.900
## Median : 7.90   Median :0.5200   Median :0.260   Median : 2.200
## Mean   : 8.32   Mean   :0.5278   Mean   :0.271   Mean   : 2.539
## 3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420   3rd Qu.: 2.600
## Max.   :15.90   Max.   :1.5800   Max.   :1.000   Max.   :15.500
##   chlorides      free.sulfur.dioxide total.sulfur.dioxide    density
```

```
## Min.   :0.01200   Min.   : 1.00      Min.   :  6.00      Min.   :0.9901
## 1st Qu.:0.07000   1st Qu.: 7.00      1st Qu.: 22.00      1st Qu.:0.9956
## Median :0.07900   Median :14.00      Median : 38.00      Median :0.9968
## Mean   :0.08747   Mean   :15.87      Mean   : 46.47      Mean   :0.9967
## 3rd Qu.:0.09000   3rd Qu.:21.00      3rd Qu.: 62.00      3rd Qu.:0.9978
## Max.   :0.61100   Max.   :72.00      Max.   :289.00      Max.   :1.0037
##       pH          sulphates         alcohol         quality
## Min.   :2.740   Min.   :0.3300   Min.   : 8.40   Min.   :3.000
## 1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000
## Median :3.310   Median :0.6200   Median :10.20   Median :6.000
## Mean   :3.311   Mean   :0.6581   Mean   :10.42   Mean   :5.636
## 3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000
## Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :8.000
```

Identificación de los valores cero. Comúnmente, se utilizan los ceros como centinela para indicar la ausencia de ciertos valores. Así, se procede a conocer a continuación qué campos contienen elementos vacíos:

```
# Números de valores desconocidos por campo
sapply(vino, function(x) sum(is.na(x)))
```

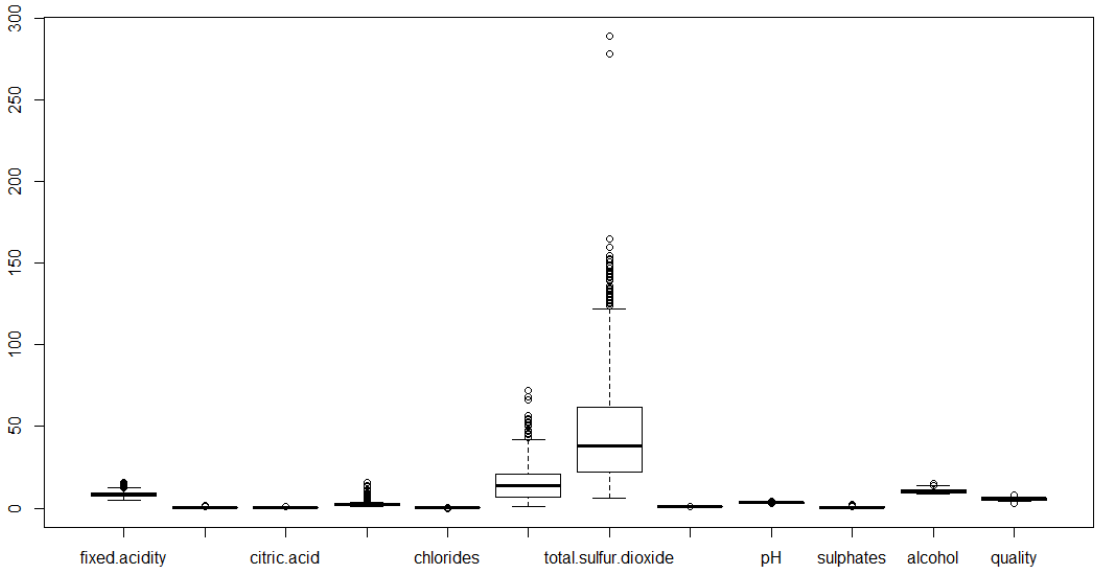
```
## fixed.acidity    volatile.acidity         citric.acid      residual.sugar
##             0                   0                   0                   0
## chlorides  free.sulfur.dioxide total.sulfur.dioxide             density
##             0                   0                   0           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

Podemos observar que no hay ningún valor NA, hecho que ya habíamos observado al hacer la función summary(), puesto que tampoco detectaba ningún NA. Sin embargo, si sabemos que hay valores cero para algunas variables, como se ha podido observar cuando hemos hecho una visualización de las primeras filas y columnas del grupo de datos, como pasaba en el ácido cítrico, pero esos valores cero no parece que sean porque falta el valor para esa variable en ese vino observado, sino porque para ese vino esa valor es cero, es decir, siguiendo con el ejemplo que habíamos mencionado, el valor del ácido cítrico en ese vino es inexistente.

Identificación de los valores extremos. Para los llamados outliers o valores extremos lo que haremos es analizar los gráficos de caja o emplear la función boxplot.stats().

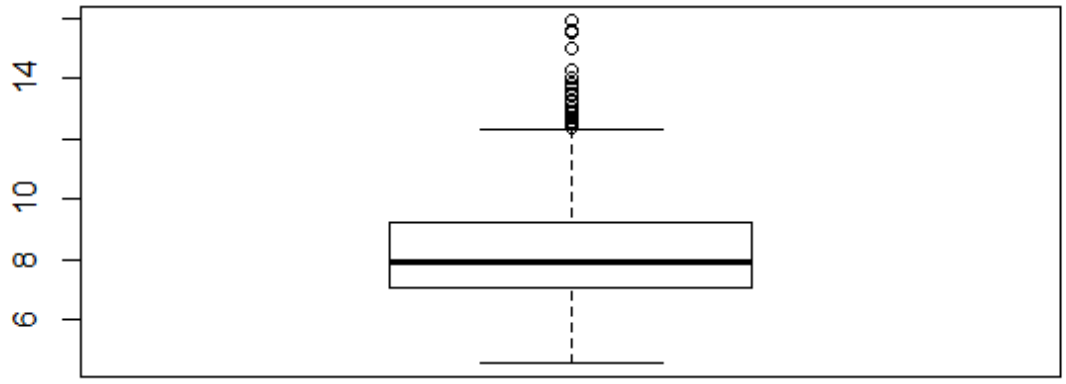
Si lo hacemos de forma gráfica, podemos dibujar todos los gráficos de caja de una vez,

```
# dibujamos los boxplot de todas las variables
boxplot(vino)
```



O lo podemos hacer una a una, como, por ejemplo, con el caso de la variable fixed acidity

```
#Boxplot de la variable fixed.acidity
boxplot(vino$fixed.acidity)
```



En el gráfico se observa mejor que hay valores extremos, por lo menos por la parte superior, hecho que se percibía, pero no se observaba con nitidez, en el gráfico conjunto, por eso es mejor que analicemos cada variable separadas. Complementamos este análisis con la función antes mencionada.

```
#Aplicación de la función boxplot.stats()
boxplot.stats(vino$fixed.acidity)
```

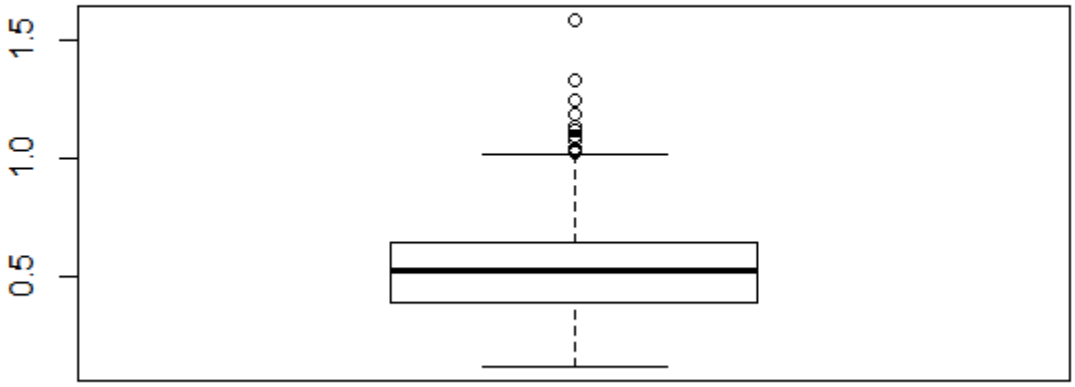
```
## $stats
## [1]  4.6  7.1  7.9  9.2 12.3
##
##$n
```

```
## [1] 1599
##
##$conf
## [1] 7.817024 7.982976
##
##$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 14.0
## [17] 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 14.3 12.4
## [33] 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 13.3 12.9 12.6
## [49] 12.6
```

La parte de la salida que nos interesa es la de $out, que nos da los valores extremos. Vemos que en el caso de esta variable hay 50 valores que superan el límite superior.

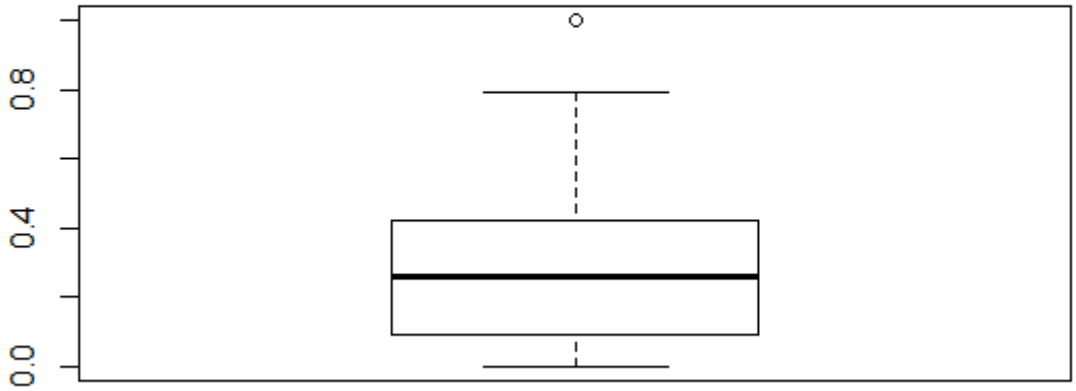
Hacemos el mismo análisis con el resto de las variables.

```
#variable volatile.acidity
boxplot(vino$volatile.acidity)
boxplot.stats(vino$volatile.acidity)$out
```



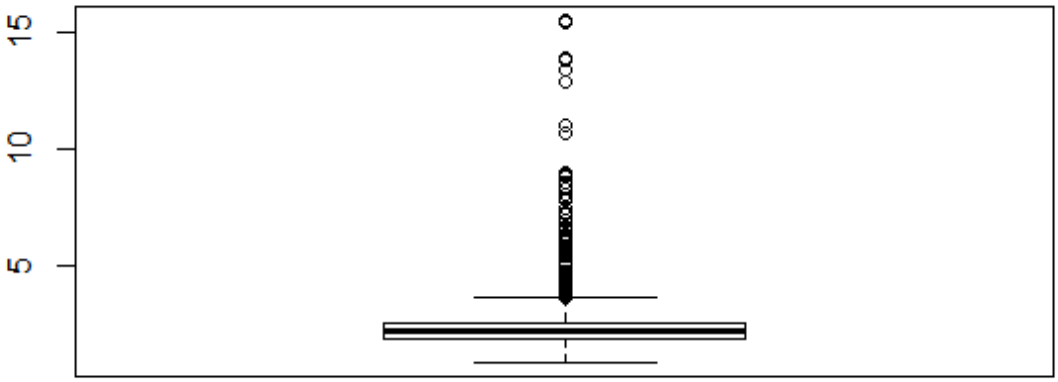

```
## [1] 1.130 1.020 1.070 1.330 1.330 1.040 1.090 1.040 1.240 1.185 1.020 1.035 1.025
## [14] 1.115 1.020 1.020 1.580 1.180 1.040
```

```
#variable citric.acid
boxplot(vino$citric.acid)
boxplot.stats(vino$citric.acid)$out
```



[1] 1

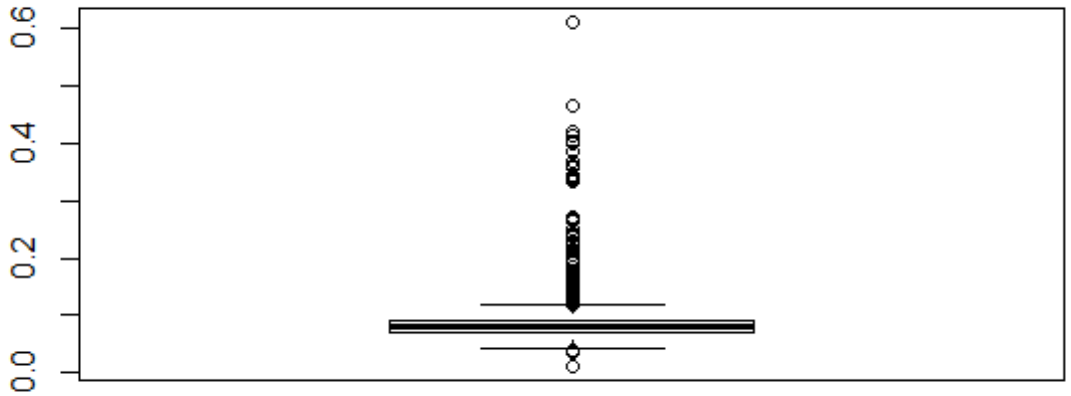
```
#variable residual.sugar
boxplot(vino$residual.sugar)
boxplot.stats(vino$residual.sugar)$out
```




```
##  [1]  6.10  6.10  3.80  3.90  4.40 10.70  5.50  5.90  5.90  3.80  5.10  4.65  4.65
## [14]  5.50  5.50  5.50  5.50  7.30  7.20  3.80  5.60  4.00  4.00  4.00  4.00  7.00
## [27]  4.00  4.00  6.40  5.60  5.60 11.00 11.00  4.50  4.80  5.80  5.80  3.80  4.40
## [40]  6.20  4.20  7.90  7.90  3.70  4.50  6.70  6.60  3.70  5.20 15.50  4.10  8.30
## [53]  6.55  6.55  4.60  6.10  4.30  5.80  5.15  6.30  4.20  4.20  4.60  4.20  4.60
## [66]  4.30  4.30  7.90  4.60  5.10  5.60  5.60  6.00  8.60  7.50  4.40  4.25  6.00
## [79]  3.90  4.20  4.00  4.00  4.00  6.60  6.00  6.00  3.80  9.00  4.60  8.80  8.80
## [92]  5.00  3.80  4.10  5.90  4.10  6.20  8.90  4.00  3.90  4.00  8.10  8.10  6.40
## [105]  6.40  8.30  8.30  4.70  5.50  5.50  4.30  5.50  3.70  6.20  5.60  7.80  4.60
```

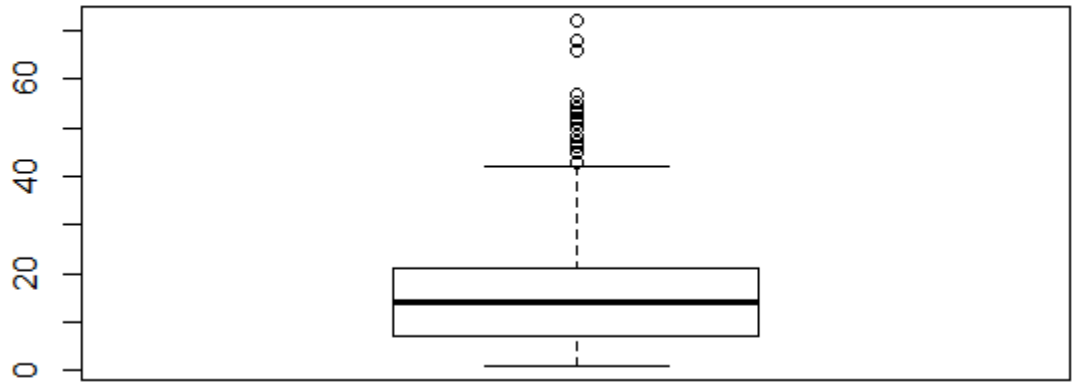
```
## [118]  5.80  4.10 12.90  4.30 13.40  4.80  6.30  4.50  4.50  4.30  4.30  3.90  3.80
## [131]  5.40  3.80  6.10  3.90  5.10  5.10  3.90 15.40 15.40  4.80  5.20  5.20  3.75
## [144] 13.80 13.80  5.70  4.30  4.10  4.10  4.40  3.70  6.70 13.90  5.10  7.80
```

```
#variable chlorides
boxplot(vino$chlorides)
boxplot.stats(vino$chlorides)$out
```



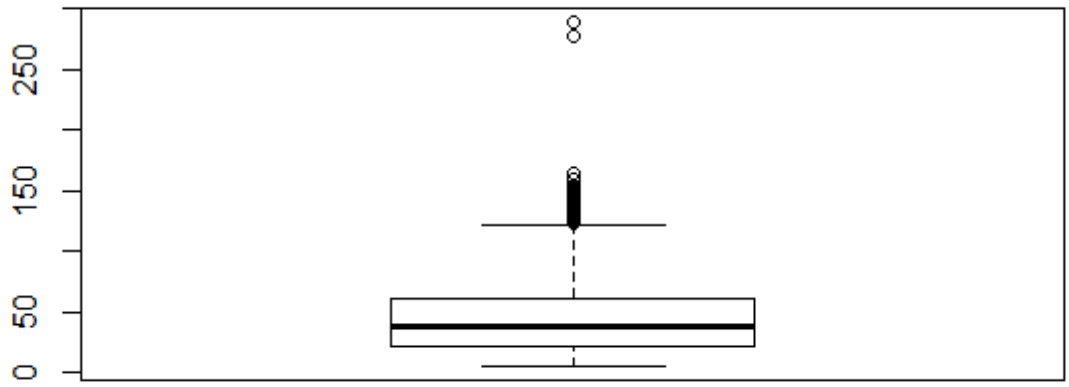

```
##   [1] 0.176 0.170 0.368 0.341 0.172 0.332 0.464 0.401 0.467 0.122 0.178 0.146 0.236
##  [14] 0.610 0.360 0.270 0.039 0.337 0.263 0.611 0.358 0.343 0.186 0.213 0.214 0.121
##  [27] 0.122 0.122 0.128 0.120 0.159 0.124 0.122 0.122 0.174 0.121 0.127 0.413 0.152
##  [40] 0.152 0.125 0.122 0.200 0.171 0.226 0.226 0.250 0.148 0.122 0.124 0.124 0.143
##  [53] 0.222 0.039 0.157 0.422 0.034 0.387 0.415 0.157 0.157 0.243 0.241 0.190 0.132
##  [66] 0.126 0.038 0.165 0.145 0.147 0.012 0.012 0.039 0.194 0.132 0.161 0.120 0.120
##  [79] 0.123 0.123 0.414 0.216 0.171 0.178 0.369 0.166 0.166 0.136 0.132 0.132 0.123
##  [92] 0.123 0.123 0.403 0.137 0.414 0.166 0.168 0.415 0.153 0.415 0.267 0.123 0.214
## [105] 0.214 0.169 0.205 0.205 0.039 0.235 0.230 0.038
```

```
#variable free.sulfur.dioxide
boxplot(vino$free.sulfur.dioxide)
boxplot.stats(vino$free.sulfur.dioxide)$out
```



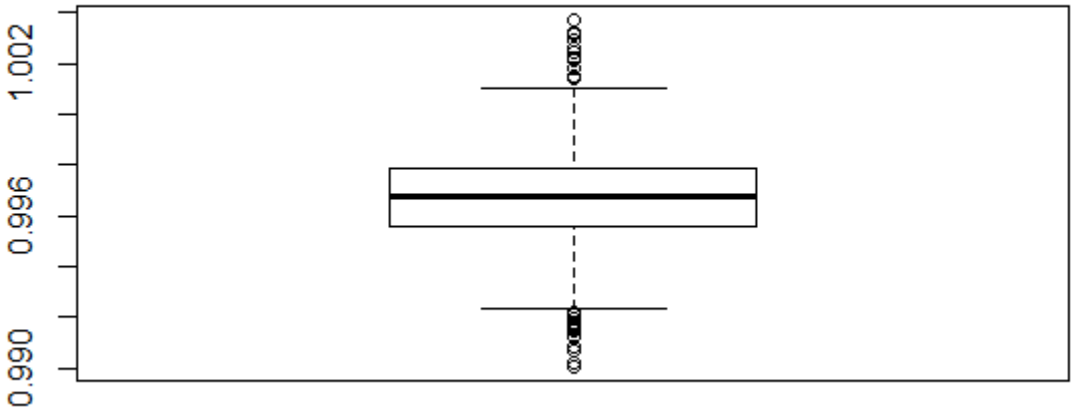

```
## [1] 52 51 50 68 68 43 47 54 46 45 53 52 51 45 57 50 45 48 43 48 72 43 51 51 52 55
## [27] 55 48 48 66
```

```
#variable total.sulfur.dioxide
boxplot(vino$total.sulfur.dioxide)
boxplot.stats(vino$total.sulfur.dioxide)$out
```



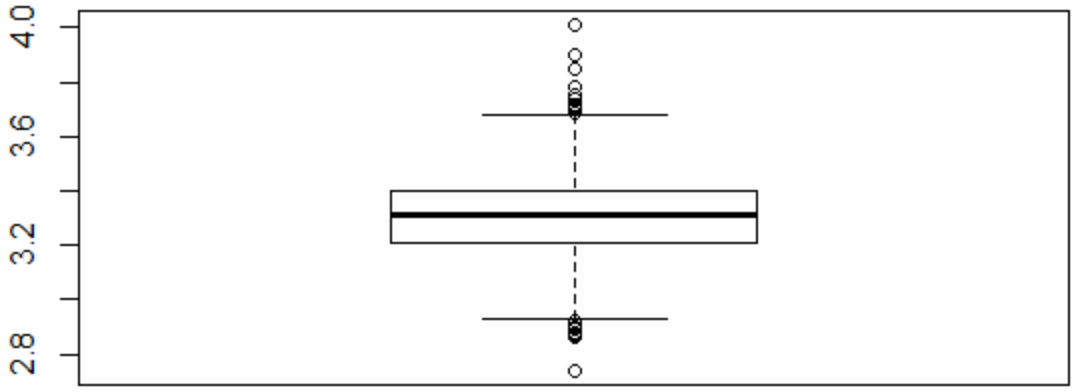

```
## [1] 145 148 136 125 140 136 133 153 134 141 129 128 129 128 143 144 127 126 145 144
## [21] 135 165 124 124 134 124 129 151 133 142 149 147 145 148 155 151 152 125 127 139
## [41] 143 144 130 278 289 135 160 141 141 133 147 147 131 131 131
```

```
#variable density
boxplot(vino$density)
boxplot.stats(vino$density)$out
```



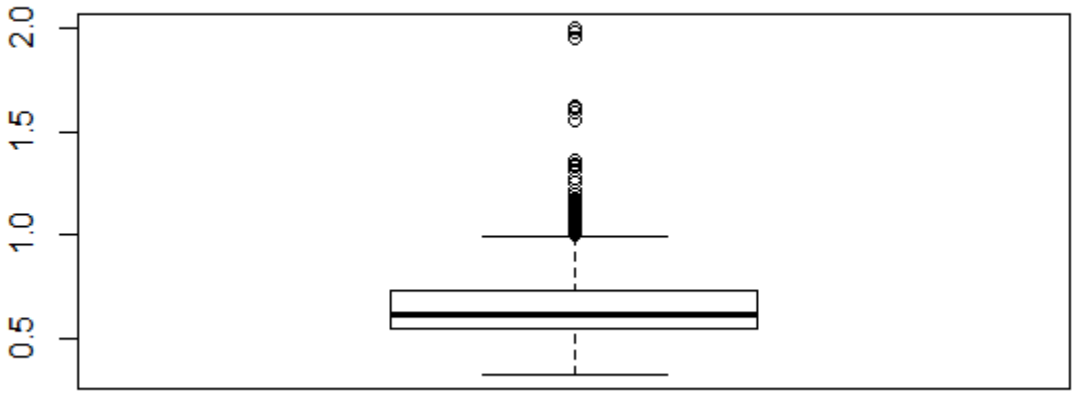

```
## [1] 0.99160 0.99160 1.00140 1.00150 1.00150 1.00180 0.99120 1.00220 1.00220 1.00140
## [11] 1.00140 1.00140 1.00140 1.00320 1.00260 1.00140 1.00315 1.00315 1.00315 1.00210
## [21] 1.00210 0.99170 0.99220 1.00260 0.99210 0.99154 0.99064 0.99064 1.00289 0.99162
## [31] 0.99007 0.99007 0.99020 0.99220 0.99150 0.99157 0.99080 0.99084 0.99191 1.00369
## [41] 1.00369 1.00242 0.99182 1.00242 0.99182
```

```
#variable pH
boxplot(vino$pH)
boxplot.stats(vino$pH)$out
```



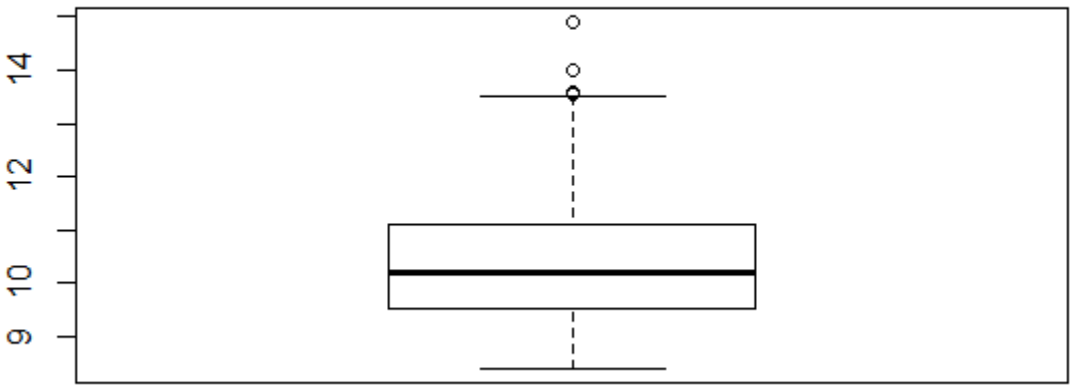

```
## [1] 3.90 3.75 3.85 2.74 3.69 3.69 2.88 2.86 3.74 2.92 2.92 2.92 3.72 2.87 2.89 2.89
## [17] 2.92 3.90 3.71 3.69 3.69 3.71 3.71 2.89 2.89 3.78 3.70 3.78 4.01 2.90 4.01 3.71
## [33] 2.88 3.72 3.72
```

```
#variable sulphates
boxplot(vino$sulphates)
boxplot.stats(vino$sulphates)$out
```



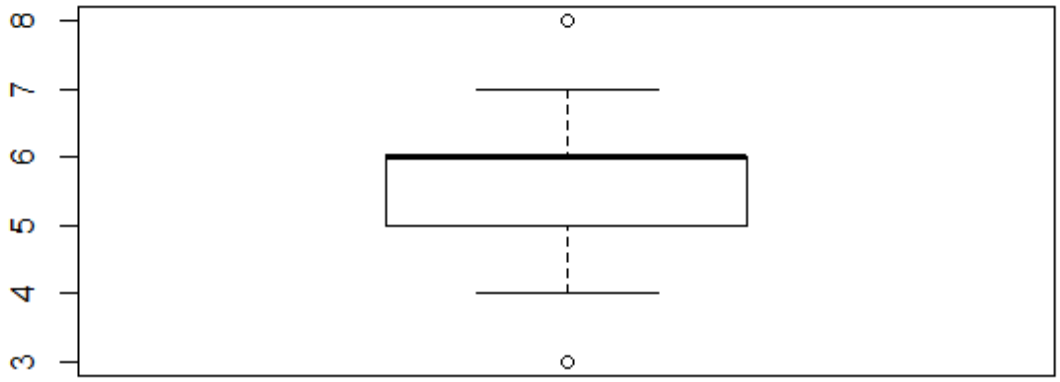

```
## [1] 1.56 1.28 1.08 1.20 1.12 1.28 1.14 1.95 1.22 1.95 1.98 1.31 2.00 1.08 1.59 1.02
## [17] 1.03 1.61 1.09 1.26 1.08 1.00 1.36 1.18 1.13 1.04 1.11 1.13 1.07 1.06 1.06 1.05
## [33] 1.06 1.04 1.05 1.02 1.14 1.02 1.36 1.36 1.05 1.17 1.62 1.06 1.18 1.07 1.34 1.16
## [49] 1.10 1.15 1.17 1.17 1.33 1.18 1.17 1.03 1.17 1.10 1.01
```

```
#variable alcohol
boxplot(vino$alcohol)
boxplot.stats(vino$alcohol)$out
```




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

```
#variable quality
boxplot(vino$quality)
boxplot.stats(vino$quality)$out
```



[1] 8 8 8 8 8 3 8 8 8 3 8 3 8 3 3 8 8 8 8 3 3 8 8 3 3 3 8

Después de analizar los valores extremos, llego a la conclusión de que no es necesario hacer ningún ajuste con ellos, como reemplazarlos por la media o la mediana o eliminarlos, por dos razones. La primera, porque tratándose de un repositorio de datos de casi 1.600 vinos, puede haber algunos que registren valores diferentes a la media, sin que por ello los datos sean erróneos. En segundo lugar, porque se trata de un repositorio que fue creado para una investigación de ciencia de datos con expertos en ese campo, con lo cual es muy posible que los datos ya hayan sido revisados y se hayan descartado los erróneos.

Por otro lado, también es curioso observar que a pesar de que la escala de la calidad se sitúa entre cero y diez, los outliers de esta variable son las observaciones(vinos) que toman como valores ocho y tres, con lo que la escala efectiva está entre cuatro y siete, con 28 vinos que toman valores diferentes.

Análisis de los datos

Selección de los grupos de datos que se quieren analizar/comparar. A priori, o por lo menos, con el conocimiento que tengo sobre la materia, no parece que haya grupos de valores dentro de las variables que nos permitan segmentar para analizar. Podríamos, por ejemplo, sabiendo los niveles de pH, ver si los vinos con un nivel de pH superior a 3,5 son superiores a los inferiores a ese nivel, pero estaríamos dejando de lado la influencia de otras variables.

En este caso, si hubiéramos hecho el estudio con los dos datasets que hay en la UCI, el del vino blanco y tinto, podríamos haber hecho una diferenciación en base al color de los vinos. Pero como sólo estamos trabajando con el color tinto, no es posible realizar dicha diferenciación.

En ese caso habríamos hecho lo siguiente, suponiendo que tuviéramos una variable que recogiera el color de cada vino,

```
# Agrupación por tipo de vino
vino.tinto <- vino[vino$tipo.vino == "tinto",]
vino.blanco <- vino[vino$tipo.vino == "blanco",]
```

Una posible diferenciación sería la de partir la escala de calidad por la mitad, por ejemplo, y ver cuáles son las características o en que niveles se sitúan las variables para esos dos grupos. En ese caso, podríamos haberlo hecho de dos maneras. La primera es separando los valores de la variable calidad en buenos o malos, convirtiendo dichos valores en valoraciones, por ejemplo, con la siguiente escala:

Puntos	**Valoración**
0-3	Malo
4-6	Intermedio
7-8	Bueno

Aunque como hemos visto en el apartado anterior, la mayoría de nuestros vinos estarían ene l nivel intermedio y solo los outliers (y los puntuados con siete) estarían en malo o bueno. Y luego agruparíamos por,

```
# Agrupación por calidad de vino
vino.malo <- vino[vino$quality == "malo",]
vino.medio <- vino[vino$quality == "medio",]
vino.bueno <- vino[vino$quality == "bueno",]
```

La segunda opción sería fijar una nueva variable, sin convertir la variable calidad, que hiciera la misma escala numérica y que se llamase, por ejemplo, quality2, con lo cual, la agrupación sería parecida a la anterior,

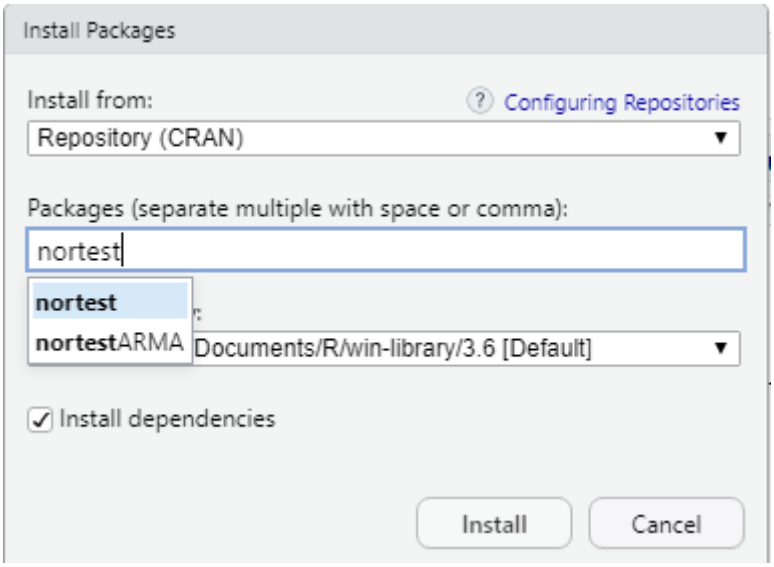
```
# Agrupación 2 por calidad de vino
vino.malo <- vino[vino$quality2 == "malo",]
vino.medio <- vino[vino$quality2 == "medio",]
vino.bueno <- vino[vino$quality2 == "bueno",]
```

Aunque tal vez, lo más interesante será observar si hay variables que tienen una mayor incidencia en la calidad sobre otras y así poder centrarnos en su estudio, para ahorrar tiempo y esfuerzo. Así que un primer paso sería la comprobación de las relaciones entre la variable output calidad y las relaciones con cada una de las variables input.

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 α = 0, 05. Si esto se cumple, entonces se considera que variable en cuestión sigue una distribución normal.

Para ello necesitamos instalar la librería nortest, así que vamos a Tools en el menú superior del RStudio, a la opción 'Install Packages' y descargamos la librería,




```
# Lo que nos saldrá en el intérprete de órdenes
install.packages("nortest")
```

Y nos saldrá el siguiente aviso,

```
> install.packages("nortest")
WARNING: Rtools is required to build R packages but is not currently installed. Please
 download and install the appropriate version of Rtools before proceeding:

https://cran.rstudio.com/bin/windows/Rtools/
Installing package into 'C:/Users/josej/Documents/R/win-library/3.6'
(as 'lib' is unspecified)
probando la URL 'https://cran.rstudio.com/bin/windows/contrib/3.6/nortest_1.0-4.zip'
Content type 'application/zip' length 39063 bytes (38 KB)
downloaded 38 KB

package 'nortest' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\josej\AppData\Local\Temp\Rtmpqggy1L\downloaded_packages
```

Así que buscamos la página de RTools (Using Rtools40 on Windows (r-project.org), los descargamos y lo instalamos.

A continuación, podemos aplicar la librería nortest,

```
# Carga librería nortest
library(nortest)

# prueba de normalidad de Anderson-Darling
alpha = 0.05
col.names = colnames(vino)

for (i in 1:ncol(vino)) {
  if (i == 1) cat("Variables que no siguen una distribución normal:\n")
  if (is.integer(vino[,i]) | is.numeric(vino[,i])) {
    p_val = ad.test(vino[,i])$p.value
```

```
    if (p_val < alpha) {
      cat(col.names[i])

    if (i < ncol(vino) - 1) 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, alcoholquality
```

Seguidamente, pasamos a estudiar la homogeneidad de varianzas mediante la aplicación de un test de Fligner-Killeen ya que como hemos visto en el apartado anterior las variables no siguen una distribución normal y esta es la mejor opción para comprobar la homocedasticidad. Comprobaremos todas las variables.

A modo de ejemplo, calculamos primero la de la acidez fija en relación con la calidad.

```
# Aplicación test Fligner-Killeen
fligner.test(quality ~ fixed.acidity, data = vino)
```

```
##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by fixed.acidity
##Fligner-Killeen:med chi-squared = 68.457, df = 95, p-value = 0.9818
```

Puesto que obtenemos un p-valor superior a 0,05, aceptamos la hipótesis de que las varianzas de ambas muestras son homogéneas.

Para no ir comentando todas las variables, primero hayamos el test y luego hago un comentario general sobre los resultados.

```
# Aplicación del test sobre el resto de las variables del dataset
fligner.test(quality ~ volatile.acidity, data = vino)
```

```
##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by volatile.acidity
##Fligner-Killeen:med chi-squared = 147.35, df = 142, p-value = 0.3621
```

```
fligner.test(quality ~ citric.acid, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by citric.acid
##Fligner-Killeen:med chi-squared = 87.67, df = 79, p-value = 0.2362


fligner.test(quality ~ chlorides, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by chlorides
##Fligner-Killeen:med chi-squared = 148.54, df = 152, p-value = 0.5642


fligner.test(quality ~ free.sulfur.dioxide, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by free.sulfur.dioxide
##Fligner-Killeen:med chi-squared = 52.989, df = 59, p-value = 0.6955


fligner.test(quality ~ total.sulfur.dioxide, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by total.sulfur.dioxide
##Fligner-Killeen:med chi-squared = 180.55, df = 143, p-value = 0.01832


fligner.test(quality ~ density, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by density
##Fligner-Killeen:med chi-squared = 364.74, df = 435, p-value = 0.9938


fligner.test(quality ~ pH, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by pH
##Fligner-Killeen:med chi-squared = 86.558, df = 88, p-value = 0.5235
```

```
fligner.test(quality ~ sulphates, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by sulphates
##Fligner-Killeen:med chi-squared = 120.2, df = 95, p-value = 0.04138


fligner.test(quality ~ alcohol, data = vino)

##        Fligner-Killeen test of homogeneity of variances
##
##data:  quality by alcohol
##Fligner-Killeen:med chi-squared = 135.98, df = 64, p-value = 4.157e-07
```

Como conclusión parece que hay tres variables que no cumplen la homocedasticidad: total.sulfur.dioxide, sulphates y alcohol, ya que sus p-value son menores que 0,05.

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

Correlación. 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 del vino. Para ello, se utilizará el coeficiente de correlación de Spearman, puesto que hemos visto que tenemos datos que no siguen una distribución normal.

```
# Cálculo de las correlaciones de las variables input
corr_matrix <- matrix(nc = 2, nr = 0)
colnames(corr_matrix) <- c("estimate", "p-value")

for (i in 1:(ncol(vino) - 1)) {
  if (is.integer(vino[,i]) | is.numeric(vino[,i])) {
    spearman_test = cor.test(vino[,i], vino[,length(vino)], method = "spearman",exact=FALSE)
    corr_coef = spearman_test$estimate
    p_val = spearman_test$p.value
    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(vino)[i]
    }
}
```

```
print(corr_matrix)
##                   estimate     p-value
## fixed.acidity      0.11408367 4.801220e-06
## volatile.acidity  -0.38064651 2.734944e-56
## citric.acid        0.21348091 6.158952e-18
## residual.sugar     0.03204817 2.002454e-01
## chlorides         -0.18992234 1.882858e-14
## free.sulfur.dioxide  -0.05690065 2.288322e-02
## total.sulfur.dioxide -0.19673508 2.046488e-15
## density           -0.17707407 9.918139e-13
## pH                -0.04367193 8.084594e-02
## sulphates          0.37706020 3.477695e-55
## alcohol            0.47853169 2.726838e-92
```

Como al copiar y pegar del Rstudio los valores no están alineados, hago una captura de pantalla para que se puedan observar mejor.

```
> print(corr_matrix)
                      estimate      p-value
fixed.acidity        0.11408367 4.801220e-06
volatile.acidity    -0.38064651 2.734944e-56
citric.acid          0.21348091 6.158952e-18
residual.sugar       0.03204817 2.002454e-01
chlorides           -0.18992234 1.882858e-14
free.sulfur.dioxide -0.05690065 2.288322e-02
total.sulfur.dioxide -0.19673508 2.046488e-15
density             -0.17707407 9.918139e-13
pH                  -0.04367193 8.084594e-02
sulphates            0.37706020 3.477695e-55
alcohol              0.47853169 2.726838e-92
```

Así, identificamos cuáles son las variables más correlacionadas con el precio en función de su proximidad con los valores -1 y +1. En nuestro caso las correlaciones no son muy elevadas y es el alcohol la variable más correlacionada con la calidad, seguida de la acidez volátil y los sulfatos.

Contraste de hipótesis. Como hemos visto que la variable que está más actualizada con la calidad es el alcohol, vamos a hacer un contraste de hipótesis basándonos en dicha variable. Para ello, dividimos primero los vinos en alcohólicos (si superan la mediana de 10.2 que hemos hallado en los primeros apartados de la práctica) o en no alcohólicos (si no superan la mediana) y almacenamos el resultado en la variable alcohol2.

Antes, duplicaremos el fichero vino, para guardar los cambios en otro fichero denominado vino2, por si hay algún problema con las transformaciones, evitar sobre escribir el fichero original.

```
# Duplicamos el fichero
vino2<-vino
```

```
#creamos la nueva variable con los valores comentados
vino2<-vino2 %>%
  mutate(alcohol2= case_when(
  alcohol>10.2~"alcohólico",
  TRUE~"no alcohólico")
)
```

Ahora que tenemos la nueva variable, ya podemos hacer el contraste de hipótesis. Para ello, tendremos dos muestras: la primera de ellas se corresponderá con la calidad de los vinos alcohólicos y, la segunda, con aquellos vinos que tienen un menor grado de alcohol.

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.

```
vino2.alcoholico.calidad <- vino2[vino2$alcohol2 == "alcohólico",]$quality
vino2.noalcoholico.calidad <- vino2[vino2$alcohol2 == "no alcohólico",]$quality
```

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 $\alpha = 0, 05$.

```
# Aplicamos el método de contraste
t.test(vino2.alcoholico.calidad, vino2.noalcoholico.calidad, alternative = "less")
```

```
##        Welch Two Sample t-test
##
## data:  vino2.alcoholico.calidad and vino2.noalcoholico.calidad
## t = 17.584, df = 1417.3, p-value = 1
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##      -Inf 0.7202451
## sample estimates:
## mean of x mean of y
## 5.982827  5.324228
```

El p-valor (p-value = 1) es mayor que el nivel de significación fijado ($\alpha = 0, 05$), por lo tanto, aceptamos la hipótesis nula por la cual las dos medias son iguales y no podemos decir que la calidad de un vino sea mejor en función de si es más alcohólico o no lo es.

Modelo de regresión lineal. Cuando analizábamos los objetivos de la práctica y las posibles variables relacionadas, mencionaba que al no saber que variables eran las más importantes, sería bueno construir un modelo en el que se pudiera comprobar como influyen las diferentes variables para el cálculo de la calidad del vino, de cara a la elaboración de vinos de mayor calidad.

Con el modelo de regresión que vamos a construir lo que podremos hacer es saber la calidad de un vino en base a los valores de las características que intervienen. Como sería muy costoso elaborar un modelo con 11 variables, lo mejor será construir varios modelos de acuerdo a los resultados que hemos obtenido en el apartado en el que hemos visto la correlación entre las variables con la calidad y compararlos con ese modelo de 11 variables para ver cuál explica mejor la calidad.

Primero, recuperamos, visualmente, los valores calculados antes,

```
> print(corr_matrix)
                       estimate      p-value
fixed.acidity         0.11408367 4.801220e-06
volatile.acidity     -0.38064651 2.734944e-56
citric.acid           0.21348091 6.158952e-18
residual.sugar        0.03204817 2.002454e-01
chlorides            -0.18992234 1.882858e-14
free.sulfur.dioxide  -0.05690065 2.288322e-02
total.sulfur.dioxide -0.19673508 2.046488e-15
density              -0.17707407 9.918139e-13
pH                   -0.04367193 8.084594e-02
sulphates             0.37706020 3.477695e-55
alcohol               0.47853169 2.726838e-92
```

Segundo, preparamos los regresores cuantitativos con mayor coeficiente de correlación respecto a la calidad, aunque los prepararemos todos para ver si el modelo de 11 variables es mejor que el resto de menos variables,

```
# Regresores cuantitativos respecto a la calidad
afija = vino$fixed.acidity
avolatil = vino$volatile.acidity
acitrica = vino$citric.acid
azucar = vino$residual.sugar
cloruro = vino$chlorides
fazufre = vino$free.sulfur.dioxide
tazufre = vino$total.sulfur.dioxide
densidad = vino$density
ph = vino$pH
sulfato = vino$sulphates
alcohol = vino$alcohol

# Variable a predecir
calidad = vino$quality
```

Tercero, generamos los modelos,

```
# Generación de los modelos
modelo1<-lm(calidad~afija-avolatil+acitrica+azucar-cloruro-fazufre-tazufre-densidad-
ph+sulfato+alcohol, data=vino)
modelo2<-lm(calidad~afija-avolatil+acitrica-cloruro-tazufre-densidad+sulfato+alcohol,
data=vino)
modelo3<-lm(calidad~afija-avolatil+azucar-cloruro-fazufre-tazufre+sulfato+alcohol, data=vino)
modelo4<-lm(calidad~avolatil-fazufre-tazufre-densidad-ph+sulfato+alcohol, data=vino)
modelo5<-lm(calidad~avolatil-fazufre-ph+sulfato+alcohol, data=vino)
```

Finalmente, con el coeficiente de determinación mediremos la bondad de los ajustes y nos quedaremos con aquel modelo que mejor coeficiente presente.

```
# Tabla con los coeficientes de determinación de cada modelo
tabla.coeficientes <- matrix(c(1, summary(modelo1)$r.squared,
2, summary(modelo2)$r.squared,
3, summary(modelo3)$r.squared,
4, summary(modelo4)$r.squared),
5, summary(modelo5)$r.squared),
ncol = 2, byrow = TRUE)

colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
```

```
##    Modelo      R^2
## [1,]      1 0.2868853
## [2,]      2 0.2862314
## [3,]      3 0.2838601
## [4,]      4 0.3358973
## [5,]      5 0.3358973
```

Del análisis de la tabla podemos ver que ninguno de los modelos tiene mucha precisión, pero que los dos que presentan mejor precisión son los modelos 4 y 5, que tendrían la siguiente forma,

modelo4: calidad=avolatil-fazufre-tazufre-densidad-ph+sulfato+alcohol
modelo5: calidad=avolatil-fazufre-ph+sulfato+alcohol

Para ver como de efectivo es el modelo, hacemos una prueba con un sexto modelo en el que la única variable explicativa sea el alcohol, que era la que tenía la R2 más elevada en los anteriores apartados y el resultado que obtenemos es el siguiente,

```
# Creamos un nuevo modelo y volvemos a comparar
modelo6<-lm(calidad~alcohol, data=vino)
tabla.coeficientes <- matrix(c(1, summary(modelo1)$r.squared, 2, summary(modelo2)$r.squared, 3, summary(modelo3)$r.squared, 4, summary(modelo4)$r.squared, 5, summary(modelo5)$r.squared, 6, summary(modelo6)$r.squared), ncol = 2, byrow = TRUE)
colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
```

```
##      Modelo      R^2
## [1,]      1 0.2868853
## [2,]      2 0.2862314
## [3,]      3 0.2838601
## [4,]      4 0.3358973
## [5,]      5 0.3358973
## [6,]      6 0.2267344
```

Con lo cual podríamos ofrecer como explicación a demostrar, porque tendríamos que probar con todas las combinaciones de variables que, primero, las combinaciones de las variables que tienen mejor R2 parece que no son las que mejor explican el modelo, y segundo que parece que un modelo con más variables no necesariamente explica mejor la calidad de los vinos.

Otra posible opción de cálculo del modelo de regresión sería usar la función lm() con varias variables y ver la combinación de aquellas que presentan una mejor covarianza y poseen una mayor representatividad de los datos. Es decir, sería usar el camino inverso a la primera opción, primero lm y luego correlación, pero teniendo la correlación calculada parece más lógico y rápido la primera opción.

Para hallar el mejor modelo podríamos utilizar la estrategia de stepwise mixto, consistente en reducir las variables para conseguir el mejor modelo de regresión. La calidad del modelo la determinaremos por el AIC. Para ello utilizamos la función step().

```
# Calcular el mejor modelo
step(object = modelo1, direction = "both", trace = 1)
```

```
## Start:  AIC=-1213.14
## calidad ~ afija - avolatil + acitrica + azucar - cloruro - fazufre -
##     tazufre - densidad - ph + sulfato + alcohol
##
##          Df Sum of Sq    RSS     AIC
## - azucar    1     0.681 743.86 -1213.67
## <none>                  743.18 -1213.14
## - afija     1     2.789 745.97 -1209.15
## - acitrica  1     3.153 746.34 -1208.37
## - sulfato   1    27.373 770.56 -1157.30
## - alcohol   1   206.821 950.00  -822.55
##
## Step:  AIC=-1213.67
```

```
## calidad ~ afija + acitrica + sulfato + alcohol
##
##          Df Sum of Sq    RSS      AIC
## <none>                 743.86 -1213.67
## + azucar    1     0.681 743.18 -1213.14
## - afija     1     2.712 746.58 -1209.86
## - acitrica  1     2.918 746.78 -1209.41
## - sulfato   1    27.795 771.66 -1157.01
## - alcohol   1   206.250 950.11  -824.36
##
## Call:
## lm(formula = calidad ~ afija + acitrica + sulfato + alcohol,
##    data = vino)
##
## Coefficients:
## (Intercept)       afija     acitrica      sulfato      alcohol
##      1.1382       0.0325       0.3121       0.8211       0.3456
```

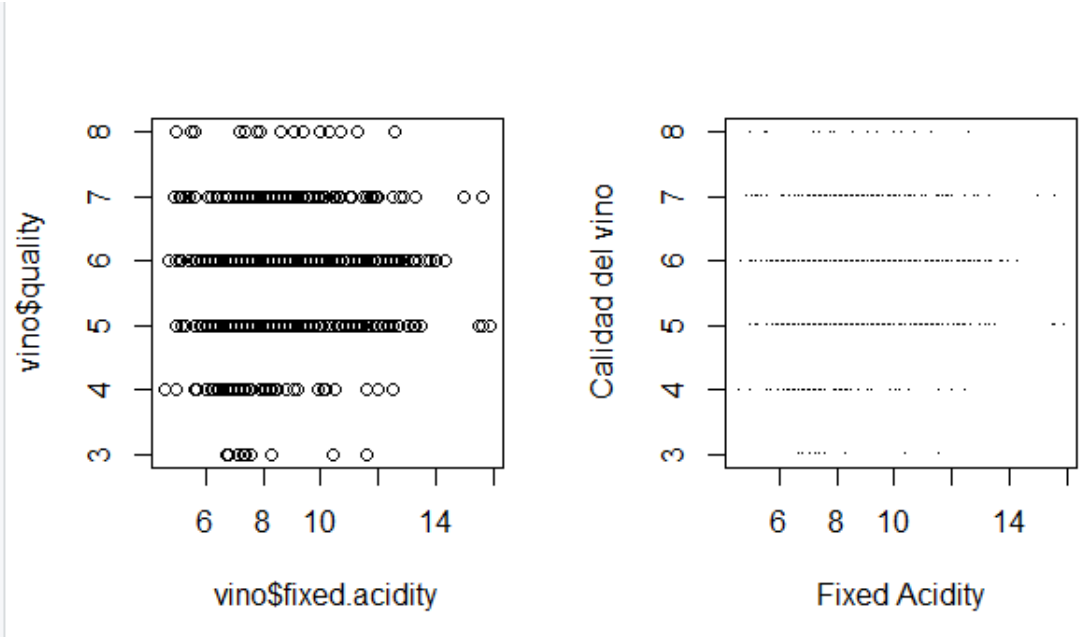
Por lo tanto, el modelo óptimo será: calidad en función de las variables afija + acitrica + sulfato + alcohol (moderados por los coeficientes que se ofrecen al final de la salida del rstudio.

Representación de los resultados a partir de tablas y gráficas

En la primera parte ya hemos hecho la representación de los outliers, mediante los diagramas de caja, así que no repetiremos esa parte aquí.

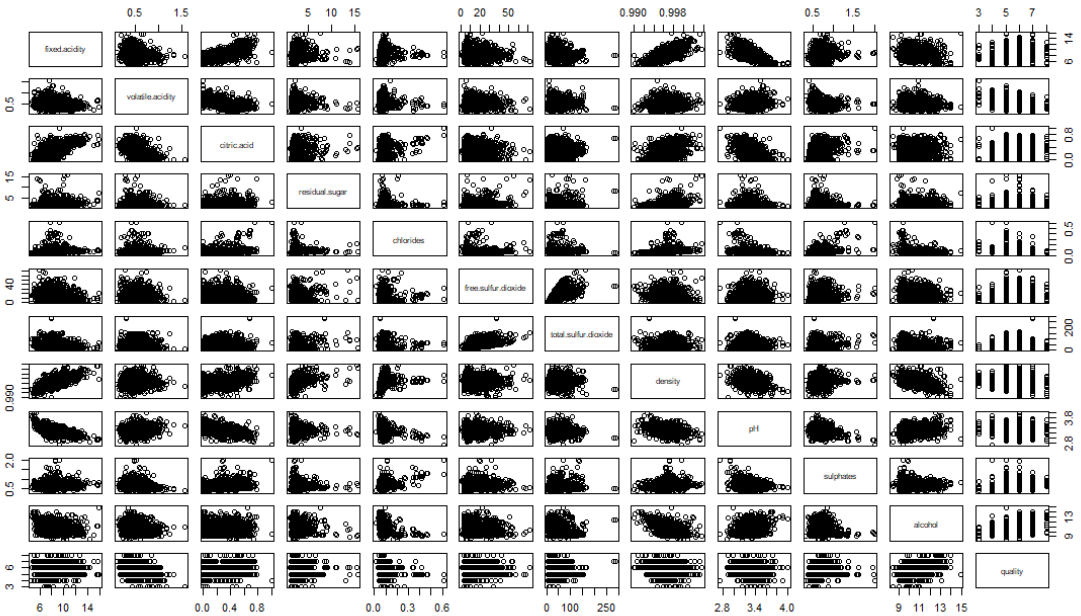
Si podemos ver las relaciones de la variable quality con el resto de los inputs del grupo de datos, por ejemplo, con la primera variable la acidez fija,

```
# Gráfico que relaciona acidez fija y calidad
par(mfrow=c(1, 2))
plot(x=vino$fixed.acidity, y=vino$quality)
plot(x=vino$fixed.acidity, y=vino$quality, pch='.', xlab='Fixed Acidity', ylab='Calidad del vino')
```



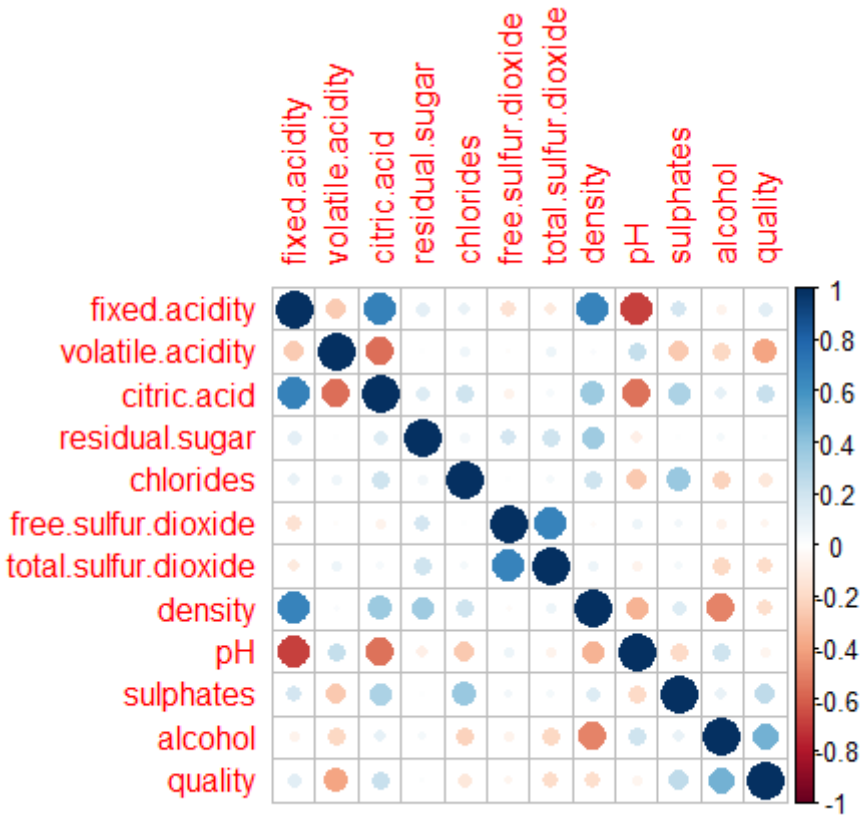
En el gráfico lo que podemos ver es que al ser la calidad una variable discreta y, por lo tanto, lo que recoge es la puntuación que se le ha dado a ese vino en función de la acidez fija que posee. Obtendríamos resultados parecidos con el resto de las variables.

También podríamos crear una matriz de dispersión con la función pairs(), pero al tener tantas variables en los datos es probable que gráficamente no pudiésemos apreciar nada.



Para la correlación, podemos usar un diagrama de círculos en el que se exprese la matriz de correlación y recoja la misma entre las diferentes variables

```
# Grafo 1 de la matriz de correlación
install.packages("corrplot")
library("corrplot")
corrplot(cor(vino), method = "circle")
```



En azul podemos observar las variables que están más fuertemente correlacionadas, lo cual nos podría servir para mejorar nuestros modelos de regresión. Así vemos que las variables que están más fuertemente correlacionadas son las siguientes,

- la acidez fija con el ácido cítrico – esta relación es curiosa puesto que podría ser más normal en vinos blancos que contienen notas más cítricas, pero no es tan clara en los vinos tintos. Habría que ver las razones e investigar más.
- el dióxido de azufre libre con el total – esta relación es más lógica, puesto que el total incluye al libre.
- La acidez fija y la densidad – puede ser algo lógico, teniendo en cuenta que la acidez fija es la base de muchas de las características del vino, entre ellas el cuerpo, que está relacionada con la densidad del líquido.
- La acidez fija y el pH – en este caso la relación es negativa y puede deberse a, como hemos dicho al principio, si el nivel de pH es elevado, el vino es menos ácido.

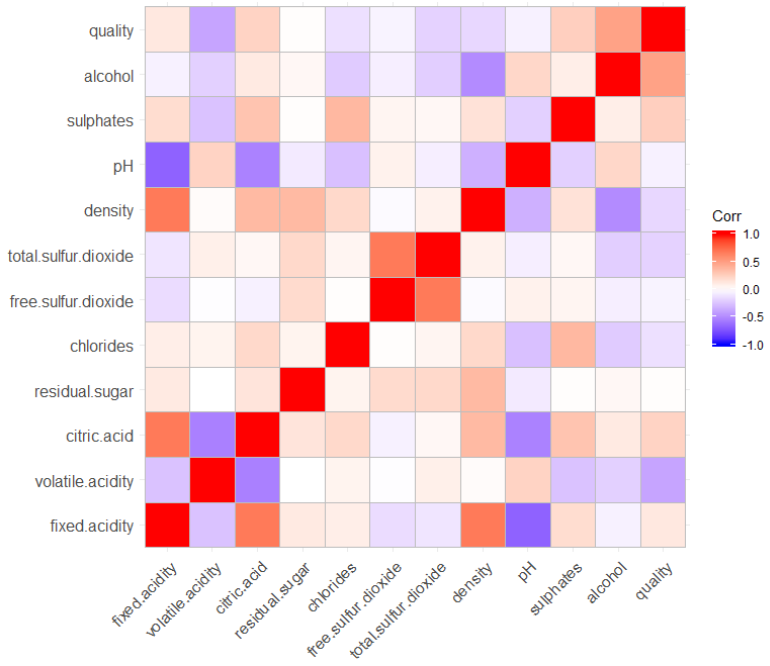
- La acidez volátil y la calidad – es una relación lógica, como se ha explicado en la caracterización de las variables.

Es curioso, por ejemplo, que la acidez y que el alcohol que, en teoría, deberían estar relacionados inversamente, no tengan casi relación. O que la calidad esté estrechamente relacionada con el alcohol, aunque como hemos visto son vinos que no son muy alcohólicos.

También hay que tener en cuenta que estamos trabajando con la variedad menos frecuente en el vinho verde, donde la uva predominante es la blanca. Habría que replicar los métodos de investigación que hemos seguido con el vino tinto en ese tipo uvas blancas y contemplar los resultados obtenidos. Incluso realizar un estudio conjunto

Utilizamos también otro tipo de gráfico de correlación para ver si obtenemos los mismos resultados (que deberían serlos, porque sólo cambiamos el diseño)

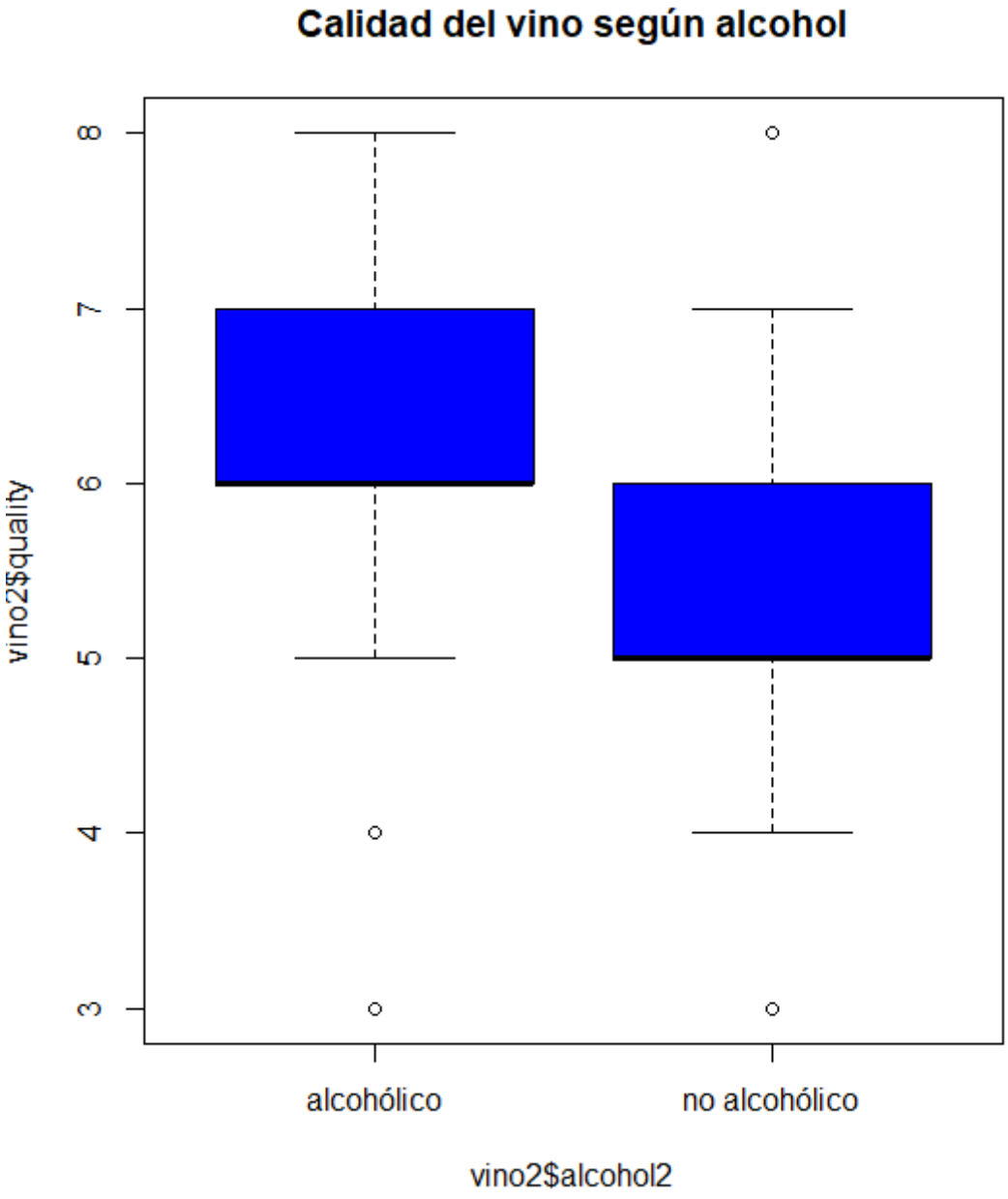
```
# Grafo 2 de la matriz de correlación
install.packages("ggcorrplot")
library("ggcorrplot")
ggcorrplot(cor(vino))
```



Como vemos, lo resultados son los mismos que en el caso anterior, aunque en este gráfico se pueden observar más matices cromáticos respecto a las correlaciones, lo cual nos puede dar más pistas para los modelos de regresión.

Para el contraste de hipótesis creo que no hay un gráfico determinado para la función t.test, pero si podemos recurrir al gráfico de cajas para ver si la calidad dentro de la variable alcohol2, que creamos para aplicar el t.test, presenta diferencias en las dos opciones de caracterización que fijamos (alcohólico y no alcohólico).

```
# Gráfico boxplot de la variable calidad en función de la variable alcohol2
boxplot(vino2$quality ~ vino2$alcohol2, col = "blue", main = "Calidad del vino según alcohol")
```



Según el resultado presentan mejor calidad los vinos alcohólicos, lo cual puede presentar diferencias con lo visto en el t.test, sin embargo, hay que tener en cuenta que en el contraste de hipótesis se estaba midiendo las medias y aquí se enseñan las medianas, así que habría que analizar con más calma como juegan los outliers en el cálculo de la media.

Por otro lado, si comparamos el resultado con el obtenido al dibujar la matriz de correlaciones, en ese caso si puede parecer que tenga sentido que los alcohólicos tengan mayor calidad, a tenor de la alta correlación que existía entre esas dos variables (alcohol y calidad).

Para la regresión múltiple, partimos del modelo ideal que hemos calculado al final del apartado y sobre él vamos a analizar la relación entre la calidad y cada uno de los inputs seleccionados con diagramas de dispersión entre cada uno de los predictores y los residuos del modelo.

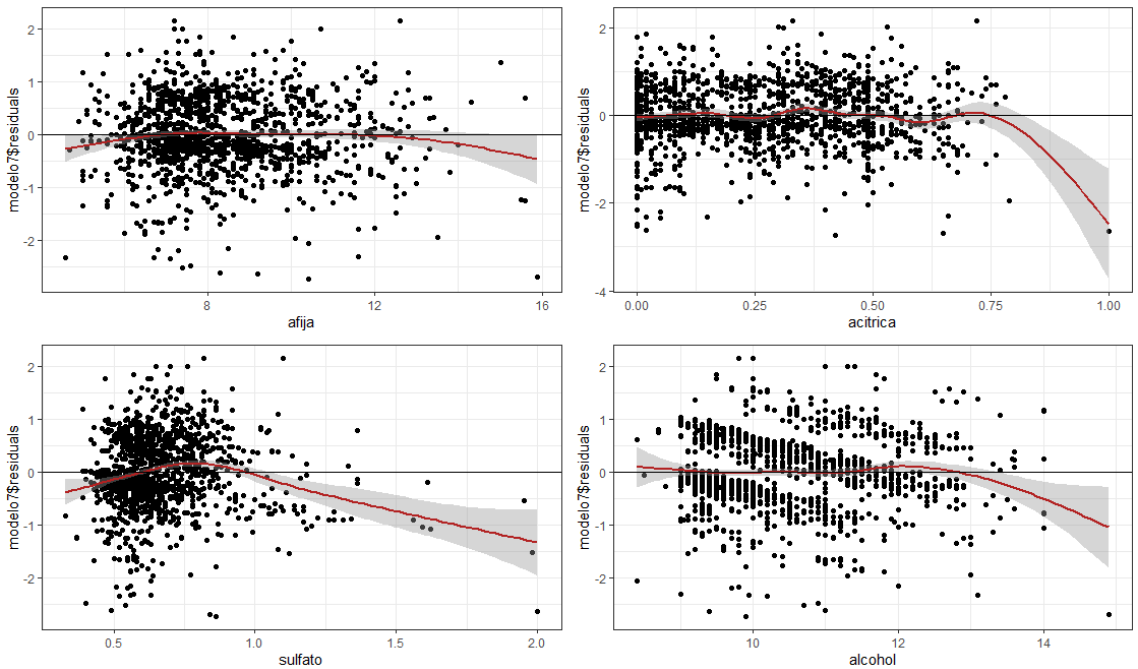
```
# Carga del modelo 7
modelo7<-lm(calidad~afija + acitrica + sulfato + alcohol, data=vino)

# Dibujo y gráficos de los residuos del modelo 7
library(ggplot2)
library(gridExtra)
```

```
## Attaching package: 'gridExtra'
##
## The following object is masked from 'package:dplyr':
##
##    combine
```

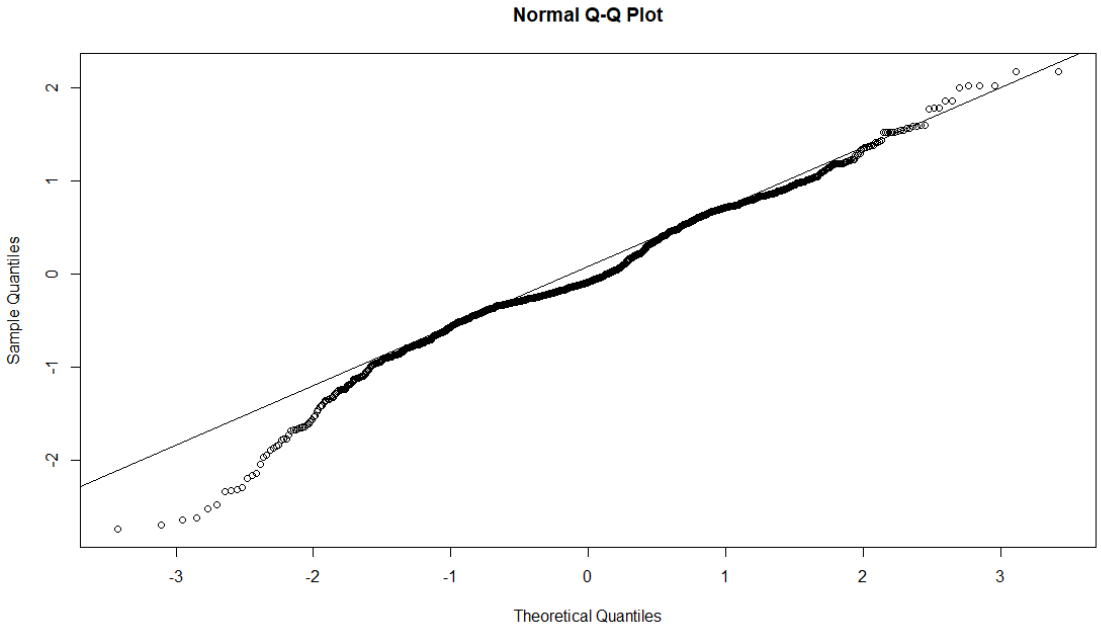
```
library(dplyr)

plot1 <- ggplot(data = vino, aes(afija, modelo7$residuals)) + geom_point() + geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
plot2 <- ggplot(data = vino, aes(acitrica, modelo7$residuals)) + geom_point() + geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
plot3 <- ggplot(data = vino, aes(sulfato, modelo7$residuals)) + geom_point() + geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
plot4 <- ggplot(data = vino, aes(alcohol, modelo7$residuals)) + geom_point() + geom_smooth(color = "firebrick") + geom_hline(yintercept = 0) + theme_bw()
grid.arrange(plot1, plot2, plot3, plot4)
```



Si la relación fuera lineal, los residuos deberían de distribuirse aleatoriamente en torno a 0 en el eje Y, con una variabilidad constante a lo largo del eje X. En nuestro caso, vemos que hay cierta linealidad, pero que los valores extremos que no hemos limpiado, por las razones ya comentadas, hacen que el modelo sea peor y que el mismo es probable que se pudiera mejorar eliminando dichos outliers.

```
# Comprobamos la distribución normal de los residuos,
qqnorm(modelo7$residuals)
qqline(modelo7$residuals)
```



Y lo que vemos es que no es normal, como se confirma por el test de Shapiro

```
# Aplicación test de Shapiro
shapiro.test(modelo7$residuals)
```

```
##        Shapiro-Wilk normality test
##
## data:  modelo7$residuals
## W = 0.9819, p-value = 2.58e-13
```

Resolución del problema. Conclusión

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

Aunque varias de las conclusiones ya las hemos adelantado antes, en los apartados respectivos, vamos a tratar de volverlas a recopilar en este apartado a modo de resumen. Hemos partido de un data set que reunía 11 variables input de características fisicoquímicas y una variable output sobre la calidad del vino verde tinto portugués.

Sobre esa base hemos hecho un primer análisis sobre la calidad de los datos para ver las características de las variables y si había que hacer una limpieza de los valores extremos. Hemos optado, por razones ya explicadas en el primer apartado, por usar los datos tal y como estaban en el dataset, considerando los outliers como observaciones adicionales que no había que eliminar porque podían ser datos lícitos. Aunque, a pesar de ello, hemos podido ver en el último apartado que, para obtener mejores modelos de regresión, tal vez, sería bueno prescindir de esos outliers, o por lo menos de las variables que parecen mejores para elaborar un modelo de regresión.

Una vez visto que las variables no seguían distribuciones normales, hemos estudiado cuáles de esas variables ejercían una mayor influencia a la hora de determinar la calidad del vino, viendo que había varias de las cuales que parecían que no cumplían la homocedasticidad, siendo sin embargo uno de esos casos, el alcohol, el que presentaba una mayor relación con la calidad en los gráficos de correlación y en la aplicación del coeficiente de Spearman.

También hemos hecho un contraste de hipótesis para ver si dentro de esa variable, el alcohol, que parece que ejerce una mayor influencia, había diferencia entre vinos que presentaban un mayor grado de alcohol, frente a los que tienen una menor cantidad de alcohol. El resultado ha sido algo contradictorio, ya que en base al contraste de medias parecía que no había diferencias, mientras que en el análisis gráfico parecía que los vinos alcohólicos si podían tener mayor calidad.

Por último, hemos intentado ver si podíamos obtener un modelo que nos permitiera predecir la calidad de los vinos. Después de estudiar varios de ellos, hemos encontrado el que parecía óptimo, que no tenía en cuenta todas las variables de las cuales teníamos datos en el input, sino que con sólo controlar cuatro de ellas podíamos obtener vinos de calidad media-alta (o lo más alta posible en el escalafón de los tintos).

Recursos

Paulo Cortez, University of Minho, Guimarães, Portugal, http://www3.dsi.uminho.pt/pcortez
A. Cerdeira, F. Almeida, T. Matos and J. Reis, Viticulture Commission of the Vinho Verde Region(CVRVV), Porto, Portugal @2009
Using Data Mining for Wine Quality Assessment | SpringerLink
Modeling wine preferences by data mining from physicochemical properties - ScienceDirect
P. Cortez, A. Cerdeira, F. Almeida, T. Matos and J. Reis. Modeling wine preferences by data mining from physicochemical properties. In Decision Support Systems, Elsevier, 47(4):547-553, 2009.

Vino verde en la Wikipedia Vinho verde - Wikipedia, la enciclopedia libre

Vino verde Vinhos Verdes. Vinos sí, verdes, no (elbaranda.com)

Ácidos en el vino Ácidos en el vino - Wikipedia, la enciclopedia libre

Acidez volátil ¿Sabes que es la acidez volátil de un vino?. - CataDelVino.com

Dióxido de azufre so2.pdf (morewinemaking.com)
difference between free and total sulfur dioxide - Bing
Free, Bound, and Total Sulfur Dioxide (SO2) during Oxidation of Wines | American Journal of Enology and Viticulture (ajevonline.org)

Potasio Sulfato Potásico en Uva | Revista HortiCultivos
Sulfito ácido de potasio - Wikipedia, la enciclopedia libre

General Encuentra aquí información de Análisis de vino para tu escuela ¡Entra ya! | Rincón del Vago (rincondelvago.com)

Características generales What Makes Great Wine… Great?. Using Machine Learning and Partial… | by Travis Tang (Voon Hao) | Towards Data Science

Valores cero y outliers Truco R. Valores perdidos a 0, ejemplo de uso de sapply - Análisis y Decisión (analisisydecision.es)
Detección y reemplazo de outliers con R - Adictos al trabajo

Dibujar varios boxplots
¿Cómo pongo múltiples boxplots en el mismo gráfico en R? (stackoverrun.com)
Detección y reemplazo de outliers con R - Adictos al trabajo

Función boxplot.stats() boxplot.stats function | R Documentation

Librería nortest
CRAN - Package nortest (r-project.org)
nortest package | R Documentation
nortest.pdf (r-project.org)

Instalar librerías
R Tips - Instalación y carga simultánea de librerías (rusersgroup.com)
Instalando librerias - paquetes en R facilmente - YouTube

Instalar RTools
Tutorial instalación R y RStudio. Este tutorial tiene como propósito… | by Pablo Casas | Ciencia y Datos | Medium
Using Rtools40 on Windows (r-project.org)

test de Fligner-Killeen
RPubs - Análisis de la homogeneidad de varianza (homocedasticidad)
Homocedasticidad - Wikipedia, la enciclopedia libre
R: Fligner-Killeen Test of Homogeneity of Variances (ethz.ch)

Warnings ver
r - Warning messages keep appearing in RStudio notebooks in chunks unrelated to the warnings - Stack Overflow
How to avoid the warning “Cannot compute exact p-value with ties” while perform correlation test for Spearman’s correlation in R? (tutorialspoint.com)

T-Test
Using t-tests in R | Department of Statistics (berkeley.edu)
t.test function | R Documentation
T-test in R: The Ultimate Guide - Datanovia
How to Perform T-tests in R | DataScience+ (datascienceplus.com)

Creación nueva variable
dataframe - ¿Cómo crear una nueva variable condicionada a otras en R? - Stack Overflow en español
Funcion %>% no encontrada - General - RStudio Community
r - Asignar valores a una columna según valores en otra columna - Stack Overflow en español
Creating a new variable under conditions of other two variables - tidyverse - RStudio Community

Regresión
RPubs - Regresión Lineal Múltiple en R
Shapiro-Wilk Test for Normality in R | R-bloggers (r-bloggers.com)

Gráficos
GRÁFICOS en R ☑ [TUTORIALES de todos los tipos de GRÁFICAS] (r-coder.com)
Análisis de correlación: guía rápida en R | Máxima Formación (maximaformacion.es)
34 - Dibujar una matriz de correlación en RStudio - YouTube
3 Gráficos para varias variables cuantitativas | Gráficos con R (fhernanb.github.io)
R para profesionales de los datos: una introducción (datanalytics.com)
Correlation Matrix in R (3 Examples) | Compute & Plot Cor Coefficient (statisticsglobe.com)
Tema 1: Introducción a la Informática y al lenguaje R (us.es)
Regresión Lineal Simple en R con Ejemplo - Rafa González Gouveia (gonzalezgouveia.com)

General
Ejemplos presentados en el enunciado de la práctica
Apuntes de la asignatura