

# Práctica 2: Limpieza y validación de los datos

## 1. Detalles de la Actividad

### 1.1. Descripción

En esta práctica se elabora un caso práctico 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. 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.

### 1.3. 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.

## 2. Resolución

### 2.1. Descripción del Dataset

El conjunto de datos se ha extraído del enlace de Kaggle  
[\(https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/\)](https://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/)

Los datos originales consisten en variantes del vino portugués Vinho-Verde y tiene 1599 observaciones de vino tinto y 4898 observaciones de vino blanco. Para cada uno tenemos la calidad del vino (calificada entre 0 y 10) y once atributos químicos (cuantitativos), que son los siguientes: acidez fija, acidez volátil, ácido cítrico, azúcar residual, cloruros, dióxido de azufre libre, dióxido de azufre total, densidad, PH, Sulfatos y alcohol.

Debido a cuestiones de privacidad y logística, solo están disponibles las variables fisicoquímicas (inputs) y sensoriales (outputs) (por ejemplo, no hay datos sobre los tipos de uva, la marca del vino, el precio de venta del vino, etc.).

La definición de las variables es la siguiente:

- **fixed acidity:** Suma de los diferentes ácidos orgánicos que se encuentran en el mosto o en el vino
- **volatile acidity:** La acidez volátil (VA) es una medida de los ácidos volátiles (o gaseosos) del vino. El ácido volátil primario en el vino es el ácido acético, que también es el ácido primario asociado con el olor y el sabor del vinagre.
- **citric acid:** El ácido cítrico es uno de los tres ácidos principales que se encuentran en las uvas y se convierten mediante el proceso de vinificación. Las uvas tienen naturalmente de 0,1 a 0,7 gramos por litro de ácido cítrico, que es aproximadamente el 10% de todos los ácidos
- **residual sugar:** indica cuánta azúcar queda en el vino una vez completa la fermentación. La cantidad de azúcar residual indica qué dulce será el vino
- **chlorides:** Nivel de cloruros en el vino
- **free sulfur dioxide:** niveles de dióxido de azufre libre que muestra el vino
- **total sulfur dioxide:** niveles totales de dióxido de azufre
- **density:** La densidad es la masa por unidad de volumen de vino o mosto a 20 ° C. Se expresa en gramos por mililitro, y se indica con el símbolo p

- **pH:** es la medida del grado de acidez relativa versus la alcalinidad relativa de cualquier líquido, en una escala de 0 a 14, siendo 7 el neutro. Los enólogos usan el pH como una forma de medir la madurez en relación con la acidez. Los vinos de pH bajo tendrán un sabor ácido y crujiente, mientras que los vinos de pH más alto son más susceptibles al crecimiento bacteriano.
- **Sulphates:** Niveles de sulfatos en el vino
- **Alcohol:** Niveles de alcohol en el vino
- **Quality:** El atributo de output es quality, basada en datos sensoriales y que toma valores entre 0 y 10

En esta práctica unificaremos los datos, los limpiaremos y trataremos de estimar un modelo que a partir de los datos nos pueda predecir la calidad. Este modelo se aplicará a ambos datasets para comprobar si influyen en su calidad los mismos componentes fisicoquímicos y si la calidad del vino es distinta si el vino es tinto o blanco.

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

Una vez seleccionados los datos y antes de comenzar su tratamiento se van a cargar ambos ficheros (vino blanco y vino tinto) y se van a integrar en un único dataset, donde se añadirá una nueva variable wine\_type para diferenciar los dos tipos de vino y que se codificará con 0 para el vino tinto y con 1 para el vino blanco

El código R que se ejecuta y su resultado, es:

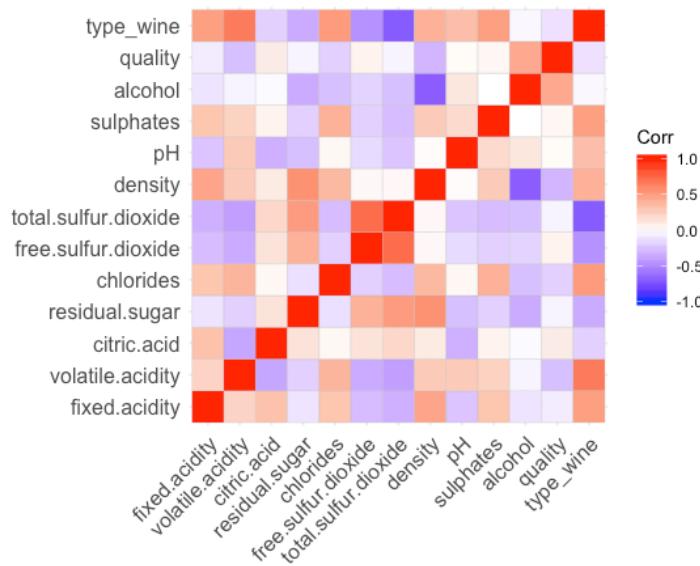
```
wine_red <- read.csv("winequality-red.csv", header=TRUE, sep =",", dec=".")  
wine_white <- read.csv("winequality-white.csv", header=TRUE, sep =";", dec=".")  
wine_red$type_wine = 1  
wine_white$type_wine = 0  
  
wine <- rbind(wine_red,wine_white)|  
  
head(wine)  
summary(wine)  
  
> head(wine)  
fixed.acidity volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide  
1 7.4 0.70 0.00 1.9 0.076 11  
2 7.8 0.88 0.00 2.6 0.098 25  
3 7.8 0.76 0.04 2.3 0.092 15  
4 11.2 0.28 0.56 1.9 0.075 17  
5 7.4 0.70 0.00 1.9 0.076 11  
6 7.4 0.66 0.00 1.8 0.075 13  
total.sulfur.dioxide density pH sulphates alcohol quality type_wine  
1 34 0.9978 3.51 0.56 9.4 5 1  
2 67 0.9968 3.20 0.68 9.8 5 1  
3 54 0.9970 3.26 0.65 9.8 5 1  
4 60 0.9980 3.16 0.58 9.8 6 1  
5 34 0.9978 3.51 0.56 9.4 5 1  
6 40 0.9978 3.51 0.56 9.4 5 1  
> summary(wine)  
fixed.acidity volatile.acidity citric.acid residual.sugar chlorides  
Min. : 3.800 Min. :0.0800 Min. :0.00000 Min. :0.600 Min. :0.00900  
1st Qu.: 6.400 1st Qu.:0.2300 1st Qu.:0.2500 1st Qu.: 1.800 1st Qu.:0.03800  
Median : 7.000 Median :0.2900 Median :0.3100 Median : 3.000 Median :0.04700  
Mean : 7.215 Mean :0.3397 Mean :0.3186 Mean : 5.443 Mean :0.05603  
3rd Qu.: 7.700 3rd Qu.:0.4000 3rd Qu.:0.3900 3rd Qu.: 8.100 3rd Qu.:0.06500  
Max. :15.900 Max. :1.5800 Max. :1.6600 Max. :65.800 Max. :0.61100  
free.sulfur.dioxide total.sulfur.dioxide density pH sulphates  
Min. : 1.00 Min. : 6.0 Min. :0.9871 Min. :2.720 Min. :0.2200  
1st Qu.: 17.00 1st Qu.: 77.0 1st Qu.:0.9923 1st Qu.:3.110 1st Qu.:0.4300  
Median : 29.00 Median :118.0 Median :0.9949 Median :3.210 Median :0.5100  
Mean : 30.53 Mean :115.7 Mean :0.9947 Mean :3.219 Mean :0.5313  
3rd Qu.: 41.00 3rd Qu.:156.0 3rd Qu.:0.9970 3rd Qu.:3.320 3rd Qu.:0.6000  
Max. :289.00 Max. :440.0 Max. :1.0390 Max. :4.010 Max. :2.0000  
alcohol quality type_wine  
Min. : 8.00 Min. :3.000 Min. :0.0000  
1st Qu.: 9.50 1st Qu.:5.000 1st Qu.:0.00000  
Median :10.30 Median :6.000 Median :0.00000  
Mean :10.49 Mean :5.818 Mean :0.2461  
3rd Qu.:11.30 3rd Qu.:6.000 3rd Qu.:0.00000  
Max. :14.90 Max. :9.000 Max. :1.0000
```

Comprobamos que la asignación de tipo de variable se ha hecho correctamente:

```
sapply(wine, class)
fixed.acidity      volatile.acidity       citric.acid      residual.sugar
"numeric"          "numeric"           "numeric"        "numeric"
chlorides   free.sulfur.dioxide total.sulfur.dioxide
"numeric"          "numeric"           "numeric"        density
pH                  sulphates          alcohol          quality
"numeric"          "numeric"           "numeric"        "integer"
type_wine
"numeric"
```

Una vez tenemos un resumen de como son las variables, se estudia la correlación que existe entre ellas porque si existe una alta correlación entre 2 variables habrá que eliminar una ella en los análisis futuros

```
corr<-cor(wine)
ggcorrplot(corr)
```



La gráfica muestra que el alcohol y la densidad tienen una correlación lineal negativa de 0.69 y que el dióxido de azufre libre y el dióxido de azufre total tienen una correlación lineal positiva de 0.72. Estas variables están bastante correlacionadas por lo tanto, la densidad y el dióxido de azufre libre se eliminarán del análisis.

```
wine <- wine[,-6] #eliminamos free.sulfur.dioxide
wine <- wine[,-7] #eliminamos density
```

## 2.3. Limpieza de datos

- Ceros y elementos vacíos

Se comprueba si existen variables en el fichero con algún valor vacío y qué registros son:

```
# buscamos variables y registros con valores perdidos
# la columna/s que tienen valores perdidos|
colnames(wine)[colSums(is.na(wine)) > 0]
# la fila/s y n? de columna/s que tienen valores perdidos
which(is.na(wine), arr.ind=TRUE)
```

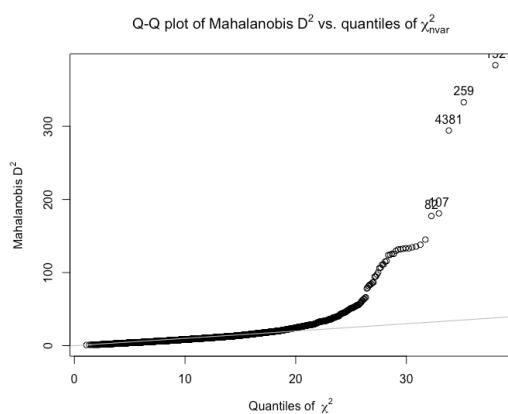
El resultado devuelve que no existen valores vacíos

```
> # la columna/s que tienen valores perdidos
> colnames(wine)[colSums(is.na(wine)) > 0]
character(0)
> # la fila/s y n? de columna/s que tienen valores perdidos
> which(is.na(wine), arr.ind=TRUE)
      row col
```

- **Valores extremos**

Los valores extremos u *outliers* son aquellos que parecen no ser congruentes si los comparamos con el resto de los datos.

Los outliers pueden complicar el análisis, ya que los modelos podrían sesgarse hacia esos valores. Se utiliza la función de outliers disponible en el paquete psych para buscar valores atípicos. La función calcula la distancia de Mahalanobis, que es  $D^2 = (x - \mu)^T \Sigma^{-1}(x - \mu)$  donde  $\Sigma$  es la covarianza de la matriz X. D2 se utiliza como una forma de detectar valores atípicos en la distribución. Los valores grandes de D2, en comparación con los valores de Chi cuadrado esperados indican un patrón de respuesta inusual. En la gráfica siguiente se ven los outliers detectados:



La ejecución de las sentencias:

```

d2 <- outlier(wine)
sat.d2 <- data.frame(wine,d2)
outlier(wine, plot = TRUE, bad = 5,na.rm = TRUE)

```

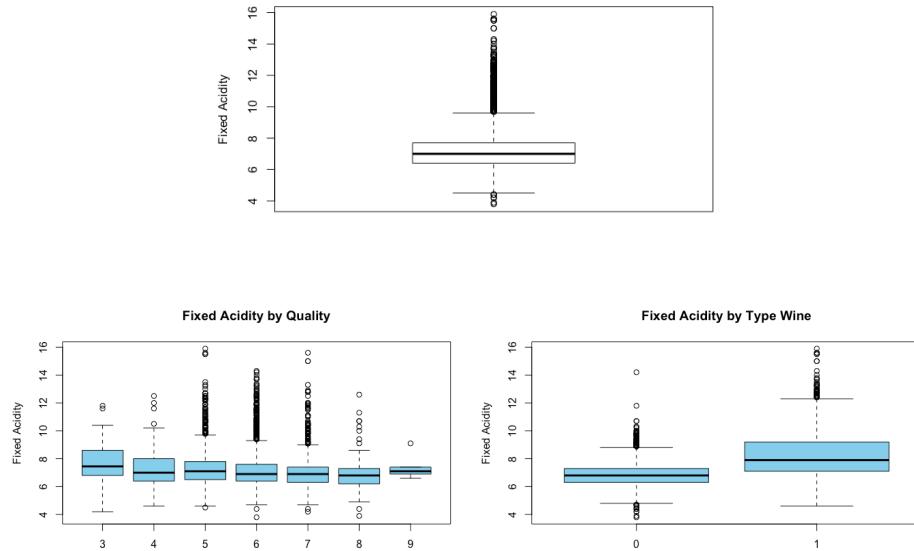
da el gráfico anterior y además nos dicta los valores outliers, en nuestro caso, los 5 principales registros outliers son los registros 4381, 152, 259, 6345 y 107, cuyos valores son:

```

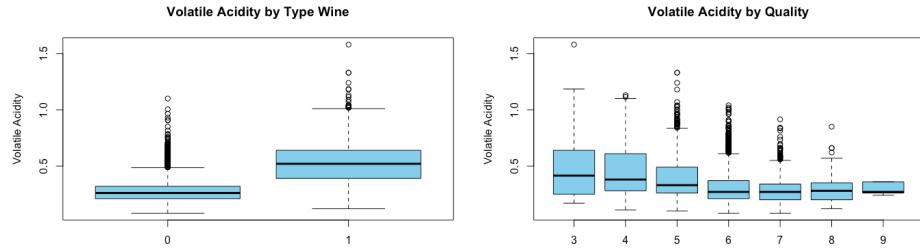
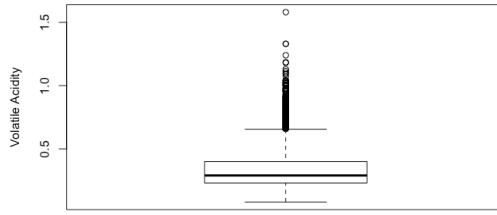
> head(d2_order,5) # para ver como va quedando.
   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides total.sulfur.dioxide
152          9.2           0.520      1.00        3.4       0.610            69
259          7.7           0.410      0.76        1.8       0.611            45
4381         7.8           0.965      0.60       65.8       0.074           160
107          7.8           0.410      0.68        1.7       0.467            69
82           7.8           0.430      0.70        1.9       0.464            67
   pH sulphates alcohol quality type_wine      d2
152 2.74     2.00    9.4      4 1 383.6270
259 3.06     1.26    9.4      5 1 332.8827
4381 3.39     0.69   11.7      6 0 294.2273
107 3.08     1.31    9.3      5 1 180.9133
82  3.13     1.28    9.4      5 1 177.3327

```

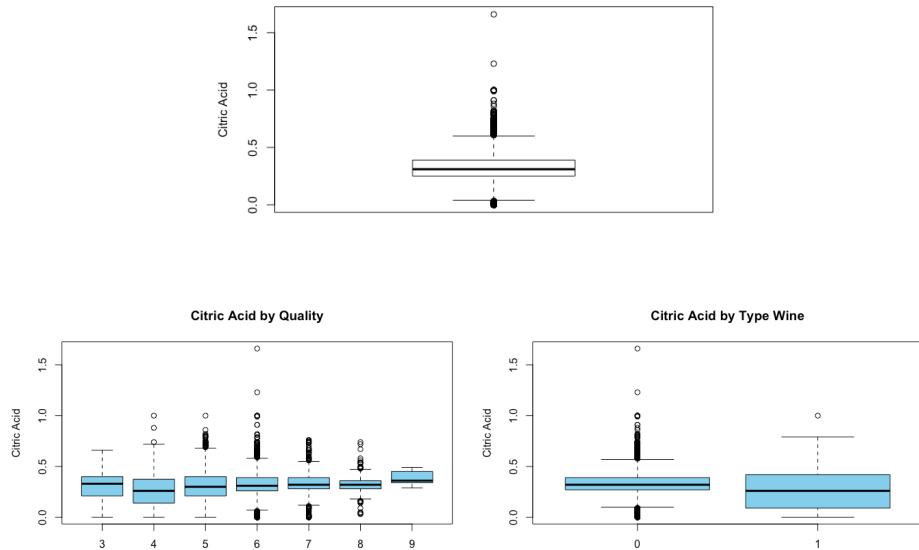
Por otro lado, se estudia los outliers que tiene cada una de las variables y además se hace un análisis de outliers por la variable Quality. Para identificarlos visualmente, se hace uso de los diagramas de caja para cada una de las variables y se utiliza la función boxplots.stats() de R para identificarlos numéricamente



```
> boxplot.stats(wine$fixed.acidity)$out
[1] 11.2 10.2 9.7 10.1 11.5 12.8 12.8 11.0 9.7 11.6 12.0 15.0 15.0 10.8 11.1 10.0 12.5
[18] 11.8 11.5 11.5 10.9 11.5 10.3 11.4 9.9 9.9 12.0 11.6 11.0 10.4 13.3 10.8 10.6 11.1
[35] 10.3 10.3 10.3 10.3 9.8 9.8 10.3 10.0 10.0 11.6 10.3 13.4 10.7 10.2 10.2 11.9 12.4
[52] 12.5 12.2 10.6 10.9 10.9 11.9 13.8 10.7 13.5 11.5 10.5 11.9 12.6 11.9 12.5 12.8 10.0
[69] 12.8 10.4 10.3 14.0 11.5 11.5 11.4 13.7 13.7 12.7 12.0 11.5 11.5 12.2 11.4 9.8 12.0
[86] 10.4 12.5 9.9 10.6 11.9 10.6 12.8 10.5 11.9 12.3 10.4 12.3 11.1 10.4 12.6 11.9 15.6
[103] 10.0 12.5 11.9 11.9 10.4 11.3 10.4 11.6 11.0 11.5 10.0 10.3 11.4 13.0 12.5 9.9 10.5
[120] 10.4 10.6 10.6 10.6 10.6 10.2 10.2 10.2 11.6 10.7 10.7 10.4 10.5 10.5 10.5 10.2 10.4
[137] 11.2 10.0 13.3 12.4 10.0 10.7 10.5 12.5 10.4 10.9 9.8 10.4 9.9 11.9 11.9 10.3
[154] 10.0 9.9 12.9 11.2 11.2 14.3 10.6 12.4 15.5 10.9 15.6 10.9 13.0 12.7 13.0 12.7
[171] 9.8 11.5 10.2 10.5 10.6 12.3 9.9 10.6 12.3 12.3 11.7 12.0 11.8 11.1 10.2 9.9 12.4
[188] 11.9 12.7 13.2 13.2 10.1 13.2 11.5 11.4 11.3 10.0 10.4 10.1 9.9 9.9 10.7 9.8
[205] 15.9 9.7 10.7 12.0 10.1 12.1 11.3 10.0 11.3 9.8 10.8 10.8 10.8 13.3 9.8 11.8 10.6
[222] 9.7 10.6 9.9 11.6 11.1 9.9 9.9 10.0 10.0 10.1 10.8 12.9 10.8 12.6 10.8 9.8 10.8
[239] 10.4 11.6 10.1 11.1 10.6 9.9 11.7 10.4 10.7 10.7 10.1 10.0 12.0 9.9 10.1 9.8 10.2
[256] 10.2 10.2 10.4 10.4 10.1 12.2 12.2 9.8 9.7 10.0 9.9 10.5 11.3 11.3 10.1 9.9 11.6
[273] 10.2 11.1 11.1 9.9 10.3 11.6 11.6 10.0 10.8 10.7 10.0 10.5 10.4 10.4 10.0 10.0 10.2
[290] 10.6 9.9 9.7 9.8 10.2 9.9 9.9 10.2 10.9 10.9 10.5 12.6 10.2 9.8 9.8 10.4 9.8
[307] 11.3 9.7 9.7 9.7 11.5 11.6 11.6 11.6 11.6 9.9 10.0 10.0 10.2 10.2 10.0 11.7 10.0 9.9
[324] 9.9 11.1 11.2 9.8 9.8 10.2 10.0 10.3 9.8 9.8 9.7 10.3 9.7 10.7 10.7 9.8 14.2
[341] 9.8 10.0 10.0 9.9 11.8 9.8 9.9 9.8 4.2 9.7 9.7 4.2 3.8 4.4 4.4 3.9 4.4
```

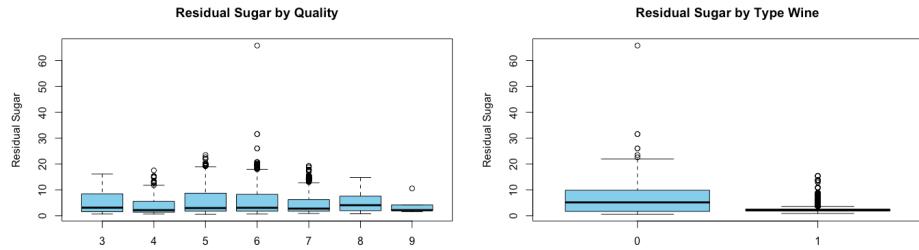
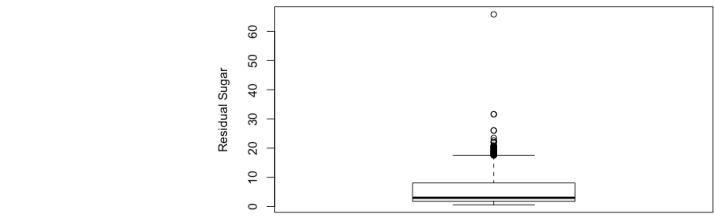


```
> boxplot.stats(wine$volatile.acidity)$out
[1] 0.700 0.880 0.760 0.700 0.660 0.710 0.675 0.685 1.130 0.660 0.670 0.935 0.660 0.690
[15] 0.735 0.725 0.725 0.705 0.705 0.670 0.690 0.675 0.785 0.750 0.670 1.020 0.775 0.900
[29] 0.785 0.690 1.070 0.695 0.710 1.330 1.330 0.745 1.040 0.745 0.715 0.745 0.715 0.670
[43] 0.680 0.680 0.950 0.680 0.705 0.885 0.805 0.730 0.705 0.835 1.090 0.725 0.735 0.725
[57] 0.820 1.000 1.000 0.660 0.680 0.660 0.775 0.695 0.975 0.870 0.715 0.830 0.670 0.705
[71] 0.670 0.770 0.660 0.770 0.660 0.695 0.685 0.815 0.785 0.670 0.795 0.665 0.750 0.700
[85] 0.765 0.660 0.850 0.665 0.735 0.765 0.735 0.735 0.735 0.725 0.770 0.840 0.960 0.960
[99] 0.840 0.670 0.780 0.840 0.670 0.685 0.735 0.660 0.820 0.680 0.680 0.685 0.670 0.670
[113] 0.775 0.690 0.690 0.670 0.825 0.715 0.660 1.040 0.700 0.700 0.730 0.720 0.915 0.835
[127] 0.755 0.690 0.690 0.685 0.935 0.840 0.880 0.885 0.915 0.670 0.845 0.840 0.840 0.730
[141] 0.730 1.240 0.730 0.800 0.780 0.780 0.980 0.780 0.660 1.185 0.920 1.020 0.765 1.035
[155] 0.780 1.025 0.690 0.740 0.660 1.115 0.660 0.720 0.865 0.875 0.835 0.965 0.965 0.690
[169] 0.760 0.910 0.980 0.870 0.870 0.700 0.680 0.670 0.720 0.685 1.000 0.765 0.820 0.890
[183] 0.715 0.660 0.685 0.685 0.665 0.665 0.660 0.685 0.680 0.735 1.010 0.715 0.715 0.715
[197] 0.750 0.800 0.900 0.660 0.660 1.020 0.715 0.670 0.660 0.860 0.660 0.660 0.840 0.690
[211] 1.005 0.710 0.830 0.795 0.820 0.745 0.910 0.965 0.780 0.740 0.770 0.770 0.725 0.725
[225] 0.780 0.690 0.690 0.690 0.660 0.660 0.760 0.785 0.785 0.880 0.660 0.915 0.660 0.800 0.955
[239] 0.885 0.885 0.745 0.740 0.720 0.815 0.750 0.750 0.730 0.670 0.740 0.720 0.660 0.700
[253] 1.020 0.780 0.715 0.690 0.715 0.765 0.740 0.755 1.580 0.860 0.790 0.690 0.680 0.690
[267] 1.180 0.740 0.760 0.660 0.740 0.870 0.775 0.835 0.850 0.830 0.740 0.770 0.780 0.780
[281] 0.750 0.815 0.885 0.830 0.755 0.810 0.685 0.675 0.670 0.690 0.690 0.690 0.915 0.670
[295] 0.785 0.670 0.785 0.670 0.900 0.785 1.040 0.980 0.690 0.700 0.875 0.910 0.680 0.740
[309] 0.895 0.740 0.725 0.820 0.760 0.810 0.690 0.790 0.840 0.840 0.700 0.690 0.705 0.855
[323] 0.680 0.670 0.680 0.735 0.855 0.880 0.855 0.695 0.695 0.670 0.695 0.695 0.690 0.700
[337] 0.670 0.715 0.660 0.725 0.740 0.660 0.660 0.670 0.670 0.685 0.905 0.670 0.705 0.680
[351] 0.660 0.670 0.850 0.910 0.710 1.005 0.760 0.930 0.695 0.705 0.815 0.680 0.965 0.740
[365] 0.780 0.680 0.750 0.670 0.730 1.100 0.660 0.695 0.695 0.690 0.690 0.785 0.760
```



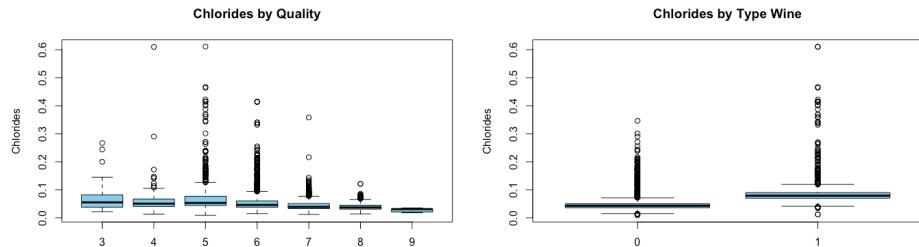
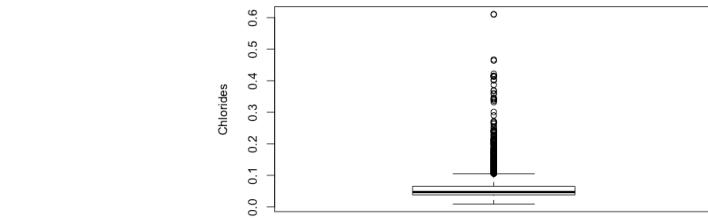
```
> boxplot.stats(wine$citric.acid)$out
```

```
[1] 0.00 0.00 0.00 0.00 0.00 0.02 0.00 0.00 0.00 0.00 0.00 0.02 0.64 0.64 0.00 0.70 0.00
[18] 0.68 0.00 0.00 0.00 0.00 0.01 0.00 0.00 0.02 1.00 0.03 0.03 0.00 0.03 0.02 0.03 0.03
[35] 0.00 0.02 0.74 0.74 0.64 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.76 0.03 0.00
[52] 0.68 0.02 0.66 0.62 0.64 0.64 0.01 0.00 0.02 0.67 0.00 0.00 0.00 0.00 0.00 0.79 0.01 0.66 0.66
[69] 0.66 0.66 0.63 0.66 0.66 0.61 0.02 0.63 0.71 0.66 0.68 0.68 0.68 0.61 0.65 0.76 0.02 0.02
[86] 0.66 0.64 0.00 0.03 0.03 0.63 0.69 0.63 0.63 0.73 0.72 0.65 0.76 0.01 0.69 0.69 0.03
[103] 0.63 0.67 0.63 0.66 0.68 0.70 0.69 0.02 0.65 0.68 0.73 0.73 0.63 0.75 0.64 0.64 0.74
[120] 0.74 0.01 0.00 0.00 0.00 0.00 0.00 0.01 0.65 0.02 0.00 0.02 0.02 0.02 0.02 0.02 0.02 0.00
[137] 0.00 0.02 0.02 0.03 0.66 0.03 0.00 0.00 0.00 0.68 0.01 0.00 0.00 0.00 0.02 0.01 0.02 0.02
[154] 0.00 0.03 0.00 0.01 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.02 0.66 0.03 0.01 0.03 0.00
[171] 0.01 0.00 0.00 0.03 0.03 0.03 0.01 0.01 0.02 0.00 0.01 0.01 0.01 0.01 0.01 0.02 0.00 0.00
[188] 0.00 0.00 0.00 0.66 0.66 0.01 0.00 0.00 0.01 0.00 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.65 0.65
[205] 0.65 0.68 0.68 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.03 0.00 0.00 0.03
[222] 0.00 0.00 0.02 0.02 0.03 0.00 0.01 0.01 0.00 0.01 0.00 0.00 0.00 0.00 0.01 0.01 0.00 0.00 0.02
[239] 0.00 0.00 0.68 0.00 0.00 0.03 0.01 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00
[256] 0.00 0.68 0.00 0.00 0.02 0.00 0.00 0.00 0.02 0.02 0.02 0.02 0.02 0.01 0.03 0.03 0.00
[273] 0.01 0.03 0.03 0.03 0.02 0.00 0.00 0.02 0.02 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.01
[290] 0.01 0.00 0.00 0.00 0.00 0.02 0.02 0.02 0.02 0.00 0.01 0.02 0.63 0.00 0.00 0.03 0.01
[307] 0.03 0.02 0.03 0.03 0.02 0.02 0.00 0.00 0.01 0.00 0.00 0.02 0.02 0.02 0.02 0.64 0.78 0.63
[324] 0.62 0.03 0.61 0.62 0.63 0.61 0.62 0.63 0.66 0.66 0.00 0.67 0.67 0.88 0.70 0.00 0.00
[341] 0.62 0.62 0.70 0.62 0.62 0.02 0.65 0.65 0.71 0.66 0.66 0.68 0.68 0.68 0.68 0.72 0.69
[358] 0.70 1.66 0.63 0.00 0.00 0.00 0.65 0.00 0.62 0.62 1.00 0.01 0.71 0.71 0.74 0.81 0.69
[375] 0.69 0.00 0.64 0.72 0.73 0.65 0.68 0.65 0.74 0.71 0.68 0.72 0.64 0.02 0.74 0.74 0.74 0.74
[392] 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74
[409] 0.74 0.99 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.74 0.01
[426] 0.74 0.01 0.74 0.74 1.00 1.00 0.00 0.61 0.61 0.61 0.02 0.67 0.67 0.67 0.65 0.71 0.71 0.03
[443] 0.64 0.64 0.81 0.61 0.62 0.00 0.63 0.73 0.68 0.78 0.79 0.64 0.65 0.65 0.00 0.73 0.73
[460] 0.64 0.71 0.72 0.82 1.00 0.66 0.80 0.80 1.23 0.02 0.00 1.00 0.62 0.00 0.71 0.71 0.71 0.71
[477] 0.61 0.61 0.00 0.72 0.62 0.62 0.79 0.82 0.67 0.01 0.01 0.86 0.61 0.02 0.00 0.69 0.69
[494] 0.01 0.66 0.66 0.78 0.00 0.91 0.91 0.74 0.62 0.73 0.00 0.00 0.67 0.01 0.00 0.02
```

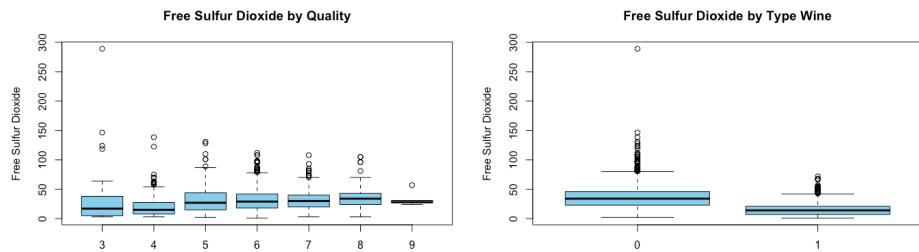
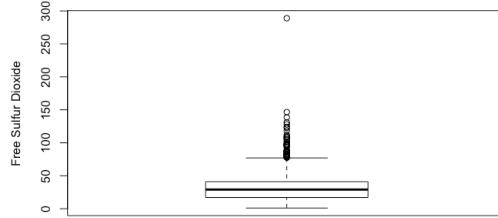


```
> boxplot.stats(wine$residual.sugar)$out
```

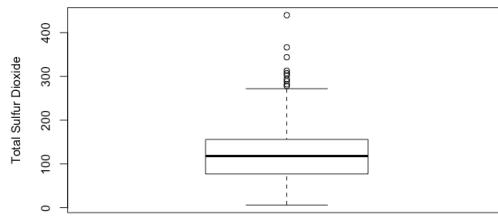
```
[1] 20.70 20.70 19.25 17.95 17.95 18.90 18.75 17.70 22.00 22.00 17.80 17.80 19.45 19.45  
[15] 19.80 18.20 18.20 18.95 18.95 20.80 18.15 17.75 18.05 18.05 18.75 19.30 18.60 18.20  
[29] 19.45 17.55 17.85 19.80 19.40 17.85 18.15 18.15 18.05 23.50 31.60 31.60 18.35 17.85  
[43] 17.85 19.50 19.60 17.95 17.80 17.85 18.30 18.35 18.35 18.15 18.15 18.15 18.15 18.80  
[57] 17.80 17.75 19.10 19.35 19.95 19.95 18.10 18.10 18.10 18.95 20.40 65.80 20.20 20.20  
[71] 18.10 19.80 19.80 17.60 18.75 20.15 18.50 18.75 19.95 19.50 19.90 26.05 26.05 17.60  
[85] 17.75 17.75 18.80 18.80 18.15 18.15 20.80 17.80 17.80 17.80 17.80 17.90 17.90 18.00  
[99] 18.00 17.80 17.80 17.55 17.55 20.30 18.10 18.10 17.80 17.80 17.80 19.30 19.30 19.30  
[113] 22.60 19.25 19.25 18.35 18.40 19.40
```



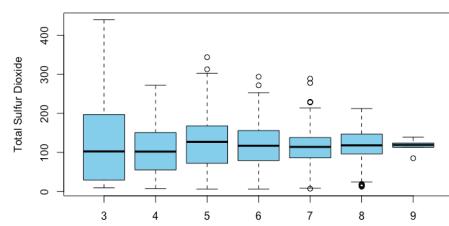
```
> boxplot.stats(wine$chlorides)$out
[1] 0.114 0.176 0.170 0.368 0.341 0.106 0.106 0.172 0.332 0.114 0.113 0.110 0.111 0.115
[15] 0.464 0.401 0.110 0.107 0.110 0.110 0.467 0.122 0.119 0.178 0.119 0.146 0.118 0.117
[29] 0.236 0.610 0.360 0.270 0.111 0.337 0.263 0.611 0.358 0.343 0.186 0.112 0.213 0.214
[43] 0.121 0.107 0.122 0.122 0.128 0.118 0.118 0.107 0.120 0.116 0.109 0.107 0.159 0.106
[57] 0.124 0.112 0.122 0.122 0.174 0.117 0.111 0.121 0.127 0.413 0.152 0.152 0.114 0.125
[71] 0.107 0.107 0.122 0.200 0.171 0.118 0.118 0.226 0.226 0.250 0.108 0.110 0.148 0.122
[85] 0.124 0.124 0.111 0.143 0.115 0.107 0.106 0.106 0.222 0.157 0.422 0.387 0.112 0.112
[99] 0.415 0.157 0.157 0.243 0.241 0.190 0.114 0.114 0.132 0.126 0.114 0.165 0.145 0.147
[113] 0.115 0.119 0.109 0.194 0.112 0.132 0.114 0.161 0.111 0.120 0.115 0.109 0.110 0.116
[127] 0.120 0.123 0.123 0.414 0.117 0.117 0.216 0.171 0.178 0.118 0.118 0.110 0.369 0.111
[141] 0.166 0.166 0.114 0.111 0.136 0.132 0.132 0.123 0.123 0.403 0.114 0.137 0.414
[155] 0.107 0.166 0.168 0.112 0.115 0.415 0.153 0.415 0.110 0.267 0.107 0.123 0.214 0.214
[169] 0.169 0.205 0.205 0.114 0.106 0.106 0.114 0.106 0.235 0.230 0.118 0.172 0.173 0.147
[183] 0.200 0.197 0.197 0.132 0.108 0.346 0.114 0.186 0.180 0.240 0.290 0.185 0.110 0.130
[197] 0.135 0.115 0.170 0.119 0.126 0.150 0.152 0.244 0.137 0.201 0.201 0.301 0.138 0.169
[211] 0.168 0.122 0.172 0.167 0.239 0.138 0.137 0.123 0.123 0.133 0.211 0.123 0.123 0.255
[225] 0.204 0.208 0.168 0.160 0.179 0.217 0.157 0.157 0.148 0.158 0.157 0.168 0.157 0.147
[239] 0.142 0.121 0.121 0.156 0.119 0.119 0.170 0.171 0.152 0.169 0.112 0.154 0.126 0.126
[253] 0.142 0.184 0.184 0.146 0.117 0.117 0.118 0.160 0.167 0.194 0.144 0.149 0.185 0.175
[267] 0.110 0.110 0.174 0.142 0.145 0.208 0.209 0.176 0.176 0.108 0.271 0.120 0.212 0.117
[281] 0.173 0.175 0.174 0.127 0.127 0.136
```



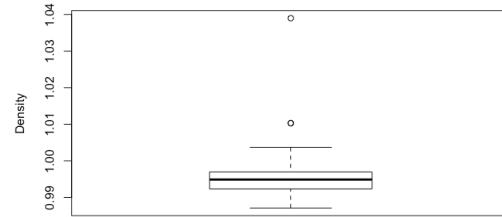
```
> boxplot.stats(wine$free.sulfur.dioxide)$out
[1] 81.0 82.0 131.0 82.5 87.0 87.0 83.0 79.0 122.5 78.0 78.0 83.0 81.0 80.0 88.0
[16] 77.5 82.0 118.5 81.0 96.0 83.0 83.0 78.0 146.5 128.0 110.0 85.0 89.0 86.0 86.0
[31] 96.0 96.0 93.0 85.0 81.0 138.5 78.0 95.0 124.0 87.0 87.0 105.0 105.0 101.0 101.0
[46] 108.0 79.5 79.5 79.5 108.0 98.0 98.0 112.0 108.0 98.0 81.0 81.0 81.0 79.0
[61] 289.0 97.0
```



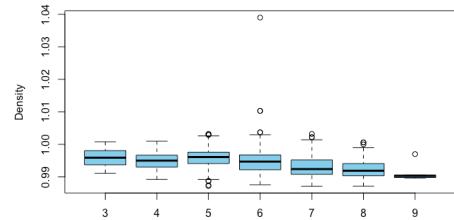
Total Sulfur Dioxide by Quality



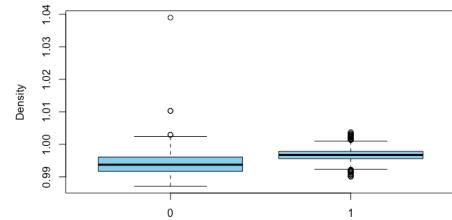
```
> boxplot.stats(wine$total.sulfur.dioxide)$out  
[1] 278.0 289.0 313.0 366.5 307.5 344.0 282.0 303.0 294.0 440.0
```



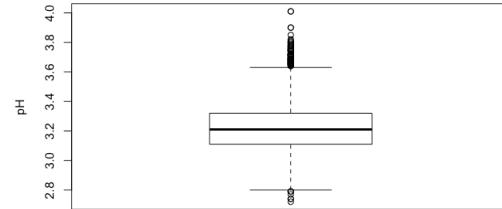
Density by Quality



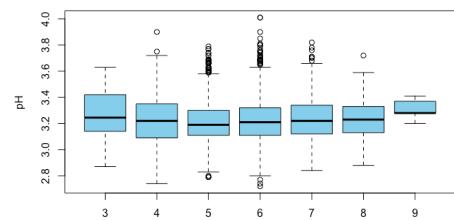
Density by Type Wine



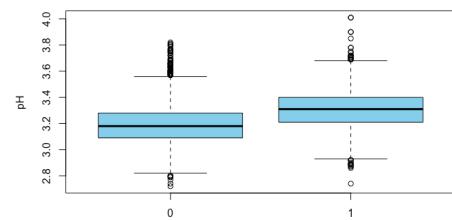
```
> boxplot.stats(wine$density)$out  
[1] 1.01030 1.01030 1.03898
```



pH by Quality

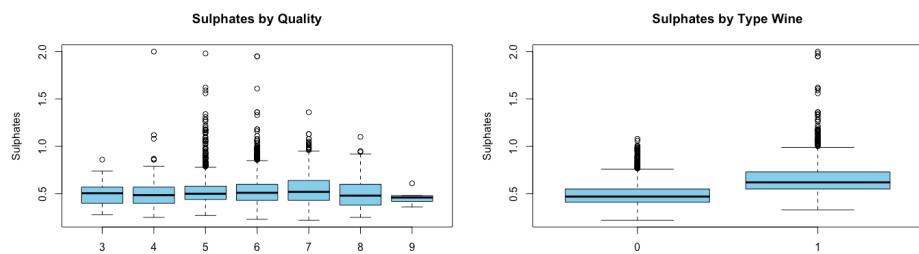
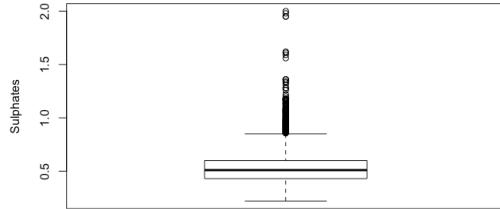


pH by Type Wine



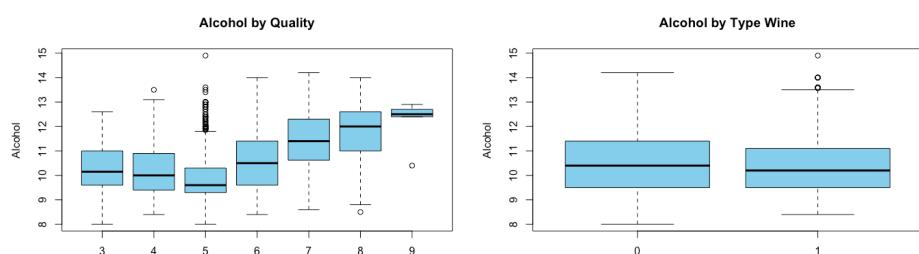
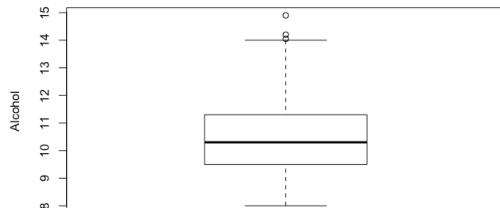
```
> boxplot.stats(wine$pH)$out
```

```
[1] 3.90 3.75 3.85 3.68 3.68 2.74 3.69 3.69 3.67 3.67 3.68 3.74 3.72 3.90 3.66 3.66 3.71 3.66  
[19] 3.69 3.69 3.71 3.71 3.78 3.68 3.68 3.70 3.78 4.01 4.01 3.71 3.66 3.72 3.72 3.67 3.69 3.72  
[37] 3.64 3.64 3.72 3.72 3.66 2.74 3.82 3.81 3.65 3.65 3.77 3.77 3.74 2.72 2.79 2.79 3.80 3.68  
[55] 2.77 3.79 3.68 3.66 3.70 3.74 3.80 3.65 3.77 3.76 3.69 3.66 2.79 3.75 3.75 3.76 3.66 3.66  
[73] 3.67
```



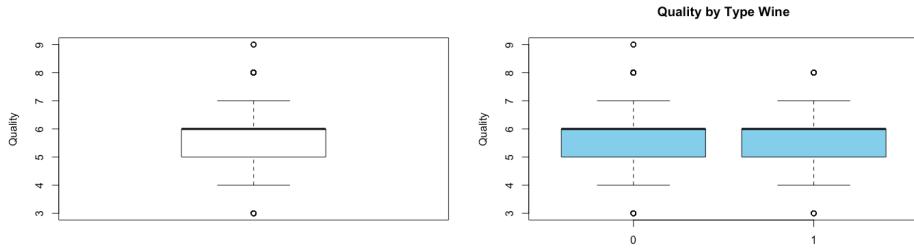
```
> boxplot.stats(wine$sulphates)$out
```

```
[1] 1.56 0.88 0.93 1.28 1.08 0.91 0.91 0.90 1.20 0.95 1.12 1.28 1.14 1.95 1.22 1.95 1.98  
[18] 1.31 0.93 0.93 0.92 2.00 1.08 1.59 1.02 0.97 1.03 0.88 0.86 1.61 1.09 0.96 0.96 1.26  
[35] 0.87 0.86 0.91 0.97 0.97 0.91 0.97 1.08 0.86 0.95 0.86 1.00 1.36 1.18 0.87 0.89 0.93  
[52] 0.92 0.86 0.98 0.88 0.91 0.87 0.93 1.13 0.87 1.04 1.11 1.13 0.99 1.07 0.90 0.90 0.89  
[69] 0.89 1.06 0.91 0.89 1.06 0.92 1.05 1.06 0.92 0.90 1.04 1.05 1.02 1.14 0.90 0.99 0.87  
[86] 0.87 0.86 0.91 1.02 1.36 0.93 0.96 1.36 1.05 1.17 1.62 1.06 0.92 0.91 1.18 0.94 0.86  
[103] 0.86 0.86 1.07 0.89 0.89 0.87 0.90 0.99 0.86 0.87 0.87 1.34 0.89 0.86 0.86 0.88 0.87  
[120] 0.87 1.16 1.10 0.98 0.88 0.86 0.94 0.87 1.15 0.87 1.17 1.17 1.33 1.18 1.17 1.03 1.17  
[137] 1.10 0.90 0.94 0.93 1.01 0.93 0.94 0.90 0.93 0.88 0.88 0.97 0.97 0.93 0.96 0.97 0.95  
[154] 0.95 0.95 0.90 0.88 0.88 0.87 0.86 0.90 0.90 0.92 0.98 1.06 0.88 0.88 0.88 1.00 0.90  
[171] 0.90 0.89 0.94 0.99 0.86 0.95 0.87 0.88 0.88 0.98 0.98 0.98 0.98 0.98 0.98 0.96 1.01 0.96  
[188] 0.92 0.94 0.95 1.08
```



```
> boxplot.stats(wine$alcohol)$out
```

```
[1] 14.90 14.20 14.05
```



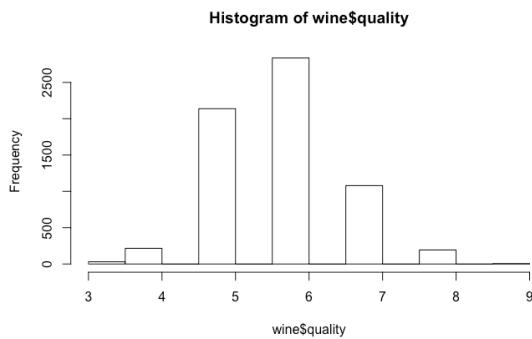
```
> boxplot.stats(wine$quality)$out  
[1] 8 8 8 8 8 3 8 8 8 3 8 3 3 8 8 8 8 8 3 3 8 8 3 3 3 8 8 8 8 8 8 8 8 3 3 8 8 3 8 8 8  
[45] 8 8 8 3 8 8 8 8 8 8 3 9 8 8 8 9 9 8 8 8 8 8 8 8 8 8 3 9 8 8 8 8 8 3 8 8 8 8 8 8 8 8 8 8 8 8  
[89] 3 8 8 8 8 8 8 8 8 8 8 8 8 8 3 8 3 8 8 8 9 8 8 8 3 8 8 8 8 8 3 8 8 8 8 8 3 8 8 8 8 8 3 8 8 8 8  
[133] 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 3 8 8 8 8 8 8 8 8 3 8 8  
[177] 8 3 8 8 8 3 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 3 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8  
[221] 8 8 3 8 8 8 8 8
```

## 2.4. Análisis de los Datos

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

El análisis que se quiere hacer es ver como contribuyen los atributos químicos del vino en su calidad.

Sabemos que la calidad varía de 0 a 10 en incrementos de 1. Sin embargo, las observaciones en el conjunto de datos solo van de 3 a 9. Sólo existen 7 grupos de calidad y además la mayor parte de los datos de calidad se concentran en las calificaciones 5,6,y 7 mientras que 3 y 9 son las que menos observaciones tienen, como se ve en el histograma



- Comprobación de la normalidad y homogeneidad de la varianza.

Para comprobar la homogeneidad de la varianza, vamos a aplicar la prueba de Bartlett que proporciona R

```

> bartlett.test(wine$fixed.acidity,wine$quality)
Bartlett test of homogeneity of variances

data: wine$fixed.acidity and wine$quality
Bartlett's K-squared = 22.586, df = 6, p-value = 0.0009476

> bartlett.test(wine$volatile.acidity,wine$quality)
Bartlett test of homogeneity of variances

data: wine$volatile.acidity and wine$quality
Bartlett's K-squared = 426.38, df = 6, p-value < 2.2e-16

> bartlett.test(wine$citric.acid,wine$quality)
Bartlett test of homogeneity of variances

data: wine$citric.acid and wine$quality
Bartlett's K-squared = 225.96, df = 6, p-value < 2.2e-16

> bartlett.test(wine$residual.sugar,wine$quality)
Bartlett test of homogeneity of variances

data: wine$residual.sugar and wine$quality
Bartlett's K-squared = 81.907, df = 6, p-value = 1.442e-15

> bartlett.test(wine$chlorides,wine$quality)
Bartlett test of homogeneity of variances

data: wine$chlorides and wine$quality
Bartlett's K-squared = 592.85, df = 6, p-value < 2.2e-16

> #bartlett.test(wine$free.sulfur.dioxide,wine$quality)
> bartlett.test(wine$total.sulfur.dioxide,wine$quality)

Bartlett test of homogeneity of variances

data: wine$total.sulfur.dioxide and wine$quality
Bartlett's K-squared = 156.36, df = 6, p-value < 2.2e-16

> #bartlett.test(wine$density,wine$quality)
> bartlett.test(wine$pH,wine$quality)

Bartlett test of homogeneity of variances

data: wine$pH and wine$quality
Bartlett's K-squared = 19.712, df = 6, p-value = 0.003116

> bartlett.test(wine$sulphates,wine$quality)
Bartlett test of homogeneity of variances

data: wine$sulphates and wine$quality
Bartlett's K-squared = 45.051, df = 6, p-value = 4.572e-08

> bartlett.test(wine$alcohol,wine$quality)
Bartlett test of homogeneity of variances

data: wine$alcohol and wine$quality
Bartlett's K-squared = 318.17, df = 6, p-value < 2.2e-16

> bartlett.test(wine$type_wine,wine$quality)
Bartlett test of homogeneity of variances

data: wine$type_wine and wine$quality
Bartlett's K-squared = Inf, df = 6, p-value < 2.2e-16

```

Los resultados del test indican que ninguna variable tiene homogeneidad en la varianza respecto a variable *quality*

Para comprobar la normalidad de las variables se aplica el test de Kolgomorov-Smirnov con la corrección de Lillefors

```
> lillie.test(wine$fixed.acidity)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$fixed.acidity
D = 0.13047, p-value < 2.2e-16

> lillie.test(wine$volatile.acidity)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$volatile.acidity
D = 0.14991, p-value < 2.2e-16

> lillie.test(wine$citric.acid)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$citric.acid
D = 0.081503, p-value < 2.2e-16

> lillie.test(wine$residual.sugar)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$residual.sugar
D = 0.20172, p-value < 2.2e-16

> lillie.test(wine$chlorides)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$chlorides
D = 0.18069, p-value < 2.2e-16

> #lillie.test(wine$free.sulfur.dioxide)
> lillie.test(wine$total.sulfur.dioxide)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$total.sulfur.dioxide
D = 0.048687, p-value < 2.2e-16

> #lillie.test(wine$density)
> lillie.test(wine$pH)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$pH
D = 0.04301, p-value < 2.2e-16

> lillie.test(wine$sulphates)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$sulphates
D = 0.093277, p-value < 2.2e-16

> lillie.test(wine$alcohol)
Lilliefors (Kolmogorov-Smirnov) normality test

data: wine$alcohol
D = 0.095716, p-value < 2.2e-16
```

El resultado de aplicar este test es que ninguna de las variables sigue una distribución normal, ya que todas tienen un p-valor < 0.05, esto no significa que no sean normalizables ya que por el teorema del límite central al tener un n > 30 podríamos aproximar las variables a una distribución normal de media 0 y desviación normal 1.

## 2.5. Aplicación de pruebas estadísticas

- ¿Qué variables fisicoquímicas influyen más en la calidad?

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.

```
> corr_matrix <- matrix(nc = 1, nr = 0)
> colnames(corr_matrix) <- c("estimate")
> # Calcular el coeficiente de correlación para cada variable
> #cuantitativa con respecto al campo "quality"
> for (i in 1:(ncol(wine) - 2)) {
+   if (is.integer(wine[,i]) | is.numeric(wine[,i])) {
+     spearman_test = cor(wine[,i],
+                           wine$quality,
+                           method = "spearman")
+     corr_coef = spearman_test
+     #p_val = spearman_test$p.value
+     # Add row to matrix
+     pair = matrix(ncol = 1, nrow = 1)
+     pair[1][1] = corr_coef
+     #pair[2][1] = 1#p_val
+     corr_matrix <- rbind(corr_matrix, pair)
+     rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(wine)[i]
+   }
+ }
> corr_matrix
           estimate
fixed.acidity -0.09742011
volatile.acidity -0.25731184
citric.acid 0.10714870
residual.sugar -0.01745245
chlorides -0.29409989
total.sulfur.dioxide -0.05575348
pH 0.03155852
sulphates 0.03114437
alcohol 0.44656864
quality 1.00000000
```

Así, identificamos cuáles son las variables más correlacionadas con la calidad del vino en función de su proximidad con los valores -1 y +1. Teniendo esto en cuenta, queda patente cómo la variable más relevante en la calidad es el alcohol, seguido de los chlorides y volatile.acidity

- ¿La calidad del vino es mayor si el vino es tinto?

La segunda prueba estadística que se aplicará consistirá en un contraste de hipótesis sobre dos muestras para determinar si la calidad del vino es superior dependiendo del tipo de vino del que se trate (tinto o blanco). Para ello, tendremos dos muestras: la primera de ellas se corresponderá con la calidad de los vinos tintos y, la segunda, con la calidad de los vinos blancos.

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:

$$H_0 : \mu_1 - \mu_2 = 0$$

$$H_1 : \mu_1 - \mu_2 < 0$$

donde  $\mu_1$  es la media de la población de la que se extrae la primera muestra y  $\mu_2$  es la media de la población de la que extrae la segunda. Así, tomaremos  $\alpha = 0, 05$ .

```
> wine.red.quality  <-  
+  wine[wine$type_wine == 0,]$quality  
> wine.white.quality  <-  
+  wine[wine$type_wine == 1,]$quality  
> t.test(wine.red.quality, wine.white.quality, alternative = "less")  
  
Welch Two Sample t-test  
  
data: wine.red.quality and wine.white.quality  
t = 10.05, df = 2942.5, p-value = 1  
alternative hypothesis: true difference in means is less than 0  
95 percent confidence interval:  
 -Inf 0.2788736  
sample estimates:  
mean of x mean of y  
3.877884 3.638245
```

Puesto que obtenemos un p-valor mayor que el valor de significación fijado, no rechazamos la hipótesis nula. Por tanto, podemos concluir que, efectivamente, la calidad de un vino no es superior si en lugar de vino blanco es vino tinto.

- [Modelo de Regresión Lineal](#)

Vamos a aplicar un modelo que nos explique qué variables son las que más importancia tienen en la variable quality, para ello vamos a utilizar un modelo de regresión múltiple que use como variables explicativas de la calidad las variables fisicoquímicas.

Para ello hacemos un conjunto de train, con más o menos el mismo nº de registros de vino blanco y vino rojo e igualmente distribuidos por la variable a predecir quality, y otro conjunto de test para comprobar la bondad del modelo

```

set.seed(92)

trainIndexRed <- createDataPartition(wine_red$quality, p = .7,
                                      list = FALSE,
                                      times = 1)
trainIndexWhite <- createDataPartition(wine_white$quality, p = .23,
                                       list = FALSE,
                                       times = 1)
trainRed <- wine_red[trainIndexRed,]
testRed <- wine_red[-trainIndexRed,]

trainWhite <- wine_white[trainIndexWhite,]
testWhite <- wine_white[-trainIndexWhite,]

train <- rbind(trainWhite, trainRed)
test <- rbind(testWhite, testRed)

```

Al aplicar el modelo encontramos:

```

> Model_lm<- lm(quality~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+chlorides+total.sulfur.dioxide+pH+alcohol+sulphates, data=train)
> summary(Model_lm)

Call:
lm(formula = quality ~ fixed.acidity + volatile.acidity + citric.acid +
    residual.sugar + chlorides + total.sulfur.dioxide + pH +
    alcohol + sulphates, data = train)

Residuals:
    Min      1Q      Median      3Q      Max 
-3.4385 -0.4320 -0.0335  0.4488  2.4146 

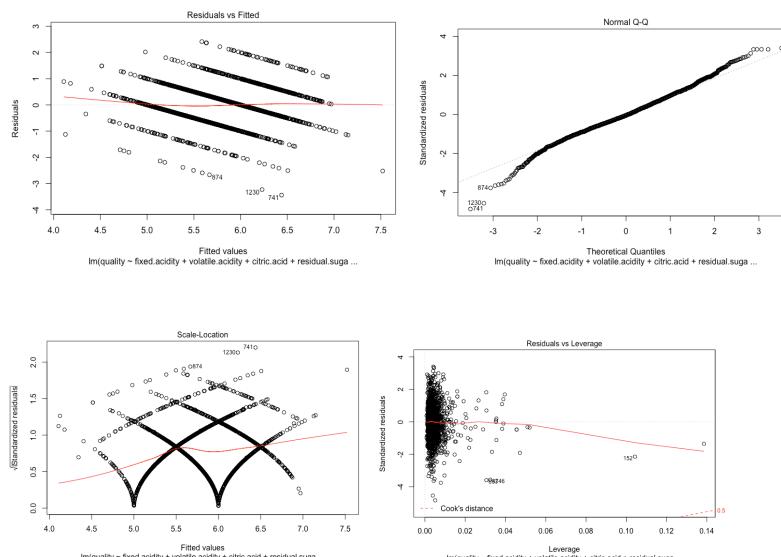
Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 3.1066814  0.4569579  6.799 1.35e-11 ***
fixed.acidity 0.0092664  0.0141490  0.655 0.512592    
volatile.acidity -1.4000966  0.1079440 -12.971 < 2e-16 ***
citric.acid   -0.2610672  0.1322004 -1.975 0.048415 *  
residual.sugar 0.0267030  0.0041600  6.419 1.67e-10 ***
chlorides     -1.3900565  0.4645211 -2.999 0.002798 ** 
total.sulfur.dioxide -0.0013939  0.0003689 -3.779 0.000162 *** 
pH           -0.1309240  0.1140115 -1.148 0.250951    
alcohol       0.3186860  0.0153019  20.827 < 2e-16 ***
sulphates     0.7311436  0.1063340  6.875 7.96e-12 *** 
--- 
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 

Residual standard error: 0.7123 on 2238 degrees of freedom
Multiple R-squared:  0.3094, Adjusted R-squared:  0.3067 
F-statistic: 111.4 on 9 and 2238 DF, p-value: < 2.2e-16

```

El modelo sólo es capaz de explicar el 30% de los casos y las variables de mayor importancia son volatile.acidity, residual.sugar, total.sulfur.dioxide, alcohol y sulphates.

Los gráficos del modelo:



Los gráficos del modelo nos indican:

- Residuals vs Fitted: Hay ningún patrón en el gráfico, ya que los residuos no se encuentran igualmente distribuidos alrededor de la línea roja, esto es un buen indicio de que hay relaciones no lineales.
- Normal Q-Q: Esta gráfica muestra si los residuos se distribuyen normalmente. En nuestro caso se observa que los residuos están bien alineados en la línea discontinua, excepto las observaciones 874, 1230 y 741 que habría que estudiarlas
- Scale-Location: Esta gráfica muestra si los residuos se reparten equitativamente a lo largo de los rangos de los predictores. Así es como puede verificar el supuesto de varianza igual (homoscedasticidad). Es bueno si ve una línea horizontal con puntos de propagación iguales (al azar). En nuestro caso prácticamente tenemos más o menos una línea con pendiente positiva, en lugar de una línea horizontal, además se vuelven a remarcar los puntos 874, 1230 y 741
- Residuals vs Leverage:Este gráfico nos ayuda a encontrar casos influyentes si los hay. No todos los valores atípicos son influyentes en el análisis de regresión lineal (lo que significan valores atípicos). A pesar de que los datos tienen valores extremos, pueden no ser influyentes para determinar una línea de regresión. En nuestro caso sí hay datos influyentes.

Como el modelo da valores numéricos continuos y quality son números enteros, vamos a redondear a 0 decimales el valor fit que devuelve el modelo y recalcularmos la bondad del modelo

```
> Model_lm_wine <- predict(Model_lm, test, interval= c("confidence"))
> Mwine <- data.frame(test,Model_lm_wine)
> Mwine$fit <- round(Mwine$fit)
>
> positiveMwine <- nrow(Mwine[Mwine$quality==Mwine$fit,])
> total <- nrow(Mwine)
>
> positiveMwine/total
[1] 0.5165921
```

Al hacer esto la bondad del modelo sube al 52%. Ahora vamos a aplicar el modelo tanto a los datos de test de vino blanco como a los datos de vino rojo, y comprobamos como se comporta el modelo

```
> Model_lm_red <- predict(Model_lm, testRed, interval= c("confidence"))
>
> Mred <- data.frame(testRed,Model_lm_red)
> Mred$fit <- round(Mred$fit)
>
> positiveMred <- nrow(Mred[Mred$quality==Mred$fit,])
> total <- nrow(Mred)
> positiveMred/total
[1] 0.5866388
>
> Model_lm_white <- predict(Model_lm, testWhite, interval= c("confidence"))
> Mwhite <- data.frame(testWhite,Model_lm_white)
> Mwhite$fit <- round(Mwhite$fit)
>
> positiveMwhite <- nrow(Mwhite[Mwhite$quality==Mwhite$fit,])
> total <- nrow(Mwhite)
> positiveMwhite/total
[1] 0.5076923
```

Encontramos que para los vinos rojos se llega a predecir correctamente el 58% de los casos de test mientras que para el vino blanco se predice bien el 51%, casi el mismo dato de predicción que daba el modelo con todos los datos de train.

Lo que nos lleva a pensar que probablemente sean otras variables las que expliquen mejor la calidad en el vino blanco, creamos otro modelo de regresión multivariante exclusivo para el dataset de vino blanco y encontramos:

```
> Model_lmWhite2<- lm(quality~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+chlorides+
  total.sulfur.dioxide+pH+alcohol+sulphates,
  +                               data=trainWhite )
> summary(Model_lmWhite)

Call:
lm(formula = quality ~ fixed.acidity + volatile.acidity + citric.acid +
    residual.sugar + chlorides + total.sulfur.dioxide + pH +
    alcohol + sulphates, data = trainWhite)

Residuals:
    Min      1Q  Median      3Q     Max 
-3.3058 -0.5034 -0.0088  0.4656  2.3620 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept) 3.0202923  0.7038310  4.291 1.93e-05 ***
fixed.acidity -0.0774880  0.0303621 -2.552  0.0108 *  
volatile.acidity -2.1766933  0.2266524 -9.604 < 2e-16 ***
citric.acid   -0.2724357  0.2013213 -1.353  0.1763    
residual.sugar 0.0317971  0.0049502  6.423 1.97e-10 ***
chlorides     -0.5447387  1.1392711 -0.478  0.6326    
total.sulfur.dioxide 0.0000919  0.0005984  0.154  0.8780    
pH             -0.1109679  0.1646972 -0.674  0.5006    
alcohol        0.3834370  0.0246049 15.584 < 2e-16 ***
sulphates      0.4812322  0.2077082  2.317  0.0207 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7556 on 1118 degrees of freedom
Multiple R-squared:  0.2713,   Adjusted R-squared:  0.2654 
F-statistic: 46.24 on 9 and 1118 DF,  p-value: < 2.2e-16
```

Que el modelo tiene una bondad de ajuste del 27%, valor bajo, y que las variables más explicativas son volatile.acidity, residual\_sugar y alcohol, eliminando sulphates y total.sulfur.dioxide que se encontraban en el general.

Si aplicamos el modelo al set de test y además hacemos el ajuste a número entero, llegamos a un acierto del 51%, que era el mismo porcentaje de acierto que teníamos con el modelo general.

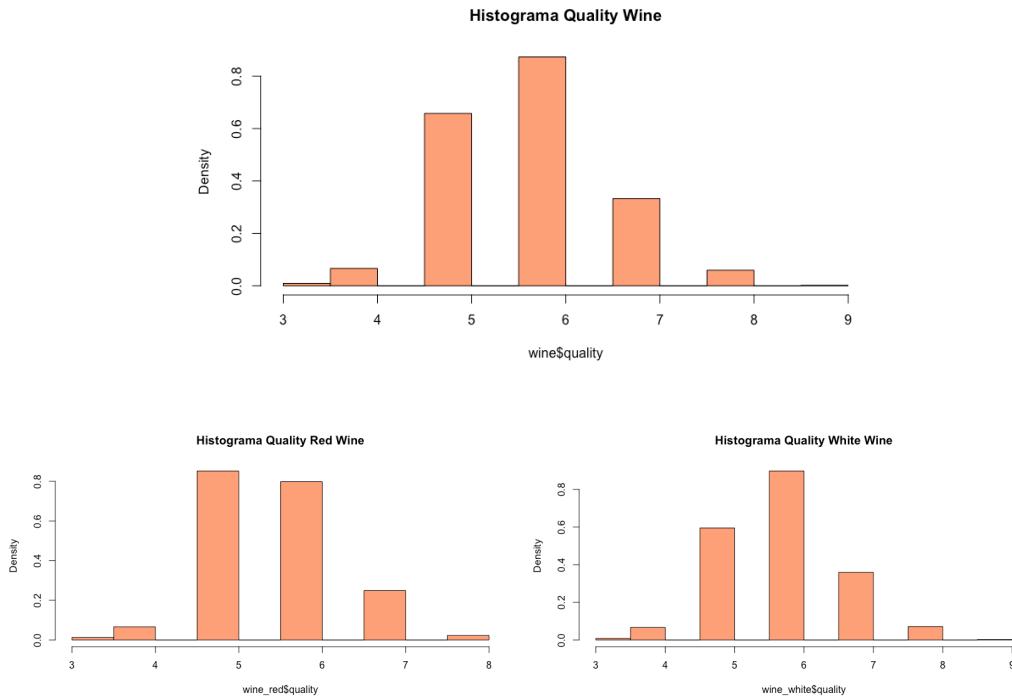
```
>
> Model_lm_white <- predict(Model_lmWhite2, testWhite, interval= c("confidence"))
> Mwhite <- data.frame(testWhite,Model_lm_white)
> Mwhite$fit <- round(Mwhite$fit)
>
> positiveMwhite <- nrow(Mwhite[Mwhite$quality==Mwhite$fit,])
> total <- nrow(Mwhite)
> positiveMwhite/total
[1] 0.5153846
```

Como el modelo no mejora la predicción del primero, decidimos que se puede utilizar el primer modelo para predecir tanto la calidad del vino tinto como la calidad del vino blanco.

Con todas estas pruebas, se concluye que la calidad del vino no depende de si es tinto o blanco, ambos tipos de vino se comportan igual en cuanto a la calidad.

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

En los histogramas de quality tanto para el vino tinto como para el blanco y para ambos tipos de vino vemos que se comportan de la misma manera, el grueso se sitúa en los valores medios de quality 5, 6 y 7 y el resto en las colas, por lo tanto, se puede determinar que la calidad del vino no depende de si el vino es tinto o blanco si no de otros factores



#### 4. Resolución del problema

Con los análisis tanto descriptivos como analíticos se determina que la calidad del vino no depende de si es blanco o tinto, sino de sus características físicoquímicas. Ambos se comporta de manera similar.