Tipología y Ciclo de Vida de los Datos - Práctica 2

Autores: José Castillo Alba - Hubner Janampa Patilla

Diciembre 2020

Contents

Presentación

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.

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 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.

Desarrollo

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

¿Por qué es importante y qué pregunta?

> El **conjunto de datos** está compuesto por variables físico/químicas de un vino Portugués. Se torna muy interesante para los fabricantes y productores vinicolas conocer que tan bueno o malo es el vino que ofrecen al mercado consumidor.
>
> La finalidad que se persigue al realizar el análisis de las características del vino es conocer las cantidades y elementos físicos/químicos que influyen a la hora determinar su calidad. Se trata de que técnicos especialistas en la elaboración del vino apoyen sus decisiones basado en criterios estadísticos y análiticos.

¿Cual es el problema pretende responder?

> El problema que pretende responder es el de determinar la calidad del vino, sobre las cantidades y elementos físicos/químicos que lo influyen, ver que elementos fisicos-químicos son mas sensibles a la hora de determinar su calidad.

En la práctica se va realizar un análisis exploratorio de los datos que iniciará con la limpieza de datos para conocer si existen valores nulos o vacios, identificar outliers, entre otros. En el caso de ser necesario se determinara las variables de importancia en el estudio, talvez normalizar y/o categorizar, luego se procede con el propio análisis de datos para descubrir patrones que nos lleve a una o varias conclusiones.

Realizamos la lectura inicial del conjunto de datos, el mismo que esta compuesto 12 variables y 1599 observaciones:

```
df <- read.csv('winequality-red.csv',header=T, sep=",", stringsAsFactors = FALSE)
str(df)
```

```
## '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 ...
```

El fichero esta compuesto por variables de tipo númericos y enteros:

```
sapply(df,class)
```

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

Las variables que forman parte del conjunto de datos, encontramos las siguientes:

- **Fixed acidity/acidez fija:** *La mayoría de los ácidos involucrados con el vino o fijos o no volátiles (no se evaporan fácilmente).*
- **Volatile acidity/acidez volátil:** *Cantidad de ácido acético en el vino, que en niveles demasiado altos puede provocar un sabor desagradable a vinagre.*
- **Citric acid/ácido cítrico:** *Encontrado en pequeñas cantidades, el ácido cítrico puede agregar 'frescura' y sabor a los vinos.*
- **Residual sugar/azúcar residual:** *Cantidad de azúcar que queda después de que se detiene la fermentación.*
- **Chlorides/cloruros:** *Cantidad de sal en el vino.*
- **Free sulfur dioxide/dióxido de azufre libre:** *La forma libre del SO2 existe en equilibrio entre el SO2 molecular (como gas disuelto) y el ion bisulfito.*
- **Total sulfur dioxide/dióxido de azufre total:** *Cantidad de formas libres y unidas de SO2.*
- **Density/densidad:** *La densidad del agua es cercana a la del agua dependiendo del porcentaje de alcohol y contenido de azúcar.*
- **pH:** *Describe qué tan ácido o básico es un vino en una escala de 0 (muy ácido) a 14 (muy básico); la mayoría de los vinos están entre 3-4.*

- **Sulphates/sulfatos:** *Un aditivo para el vino que puede contribuir a los niveles de dióxido de azufre (SO2), que actúa como antimicrobiano.*
- **Alcohol/alcohol:** *Porcentaje de alcohol contenido en el vino.*
- **Quality/calidad:** *Basado en datos sensoriales, cuya puntuación es entre 0 y 10.*

Integración y selección de los datos de interés a analizar

Al tratarse de elementos físico/químicos son variables que de alguna forma u otra determinan la calidad del vino, pero se pondrá especial atención a las variables *total.sulfur.dioxide, density y alcohol* ya que considero desde mi punto de vista que son factores relevantes en el vino.

Limpieza de los datos, ¿Los datos contienen ceros o elementos vacíos?,¿Cómo gestionarías cada uno de estos casos?

Partimos realizamos un detalle estadístico de las variables para conocer como están distribuidas:

```
summary(df)
```

```
##  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
```

Por lo regular cuando se trata de datos faltantes, siempre se denotan por campos vacios o cero. Para este caso particular no se puede asegurar al 100% en el caso de valores con cero que se trate precisamente de valores perdidos ya que se puede tratar de elementos fisico/químicos que realmente pueden ser cero, pero se debe descartar la existencia valores vacios.

A continuación se verifica que campos tienen valores vacíos:

```
colSums(is.na(df))
```

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

```
##                   0                   0                   0
```

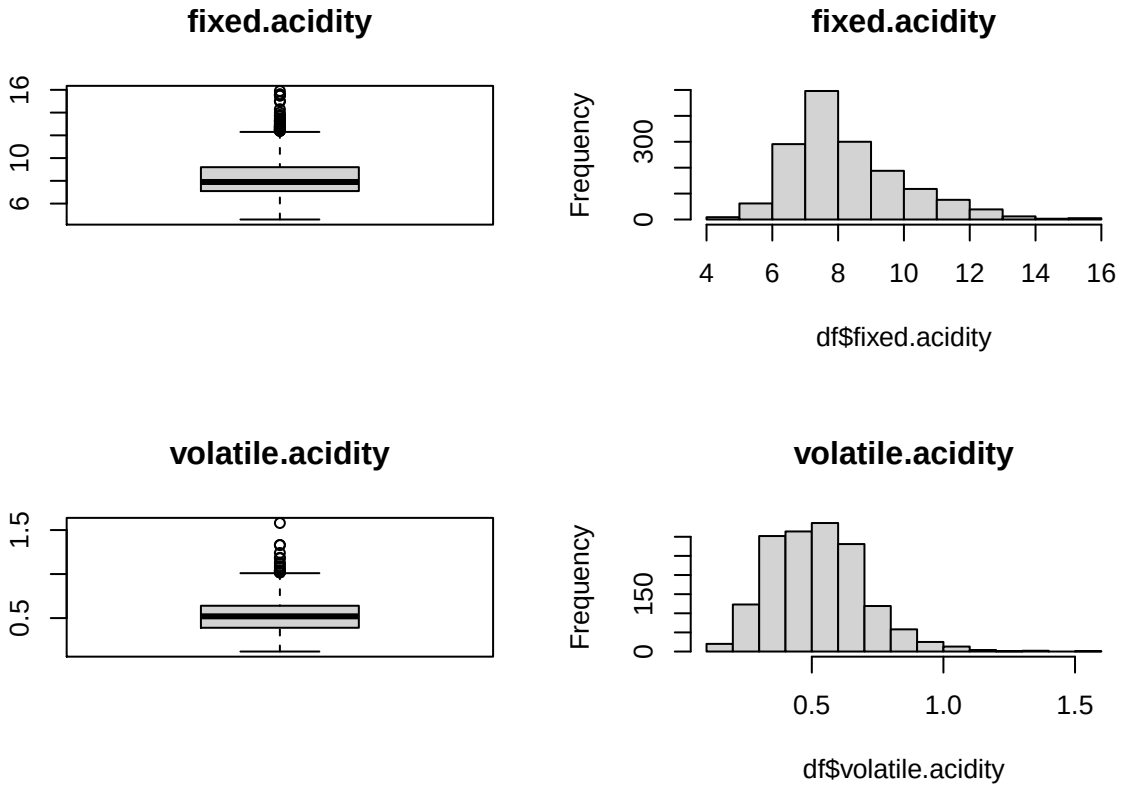
Podemos notar que no existen valores vacios, esto es una bueña señal que permite avanzar con en el análisis de los datos.

Identificación y tratamiento de valores extremos

Los valores extremos tambien conocidos como outliers, se trata de datos muy alejados de la ditribución normal de una variable (datos anómalos). Al momento de realizar el análisis, la presencia de datos extremos pueden generar estimaciones y/o resultados equivocados, por tal razón se procede a graficar mediante diagrama de cajas para corroborar o descartar la existencia.

```
par(mfrow=c(2,2))
boxplot(df$fixed.acidity,main="fixed.acidity")
hist(df$fixed.acidity,main="fixed.acidity")

boxplot(df$volatile.acidity,main="volatile.acidity")
hist(df$volatile.acidity,main="volatile.acidity")
```




```
boxplot.stats(df$fixed.acidity)$out
```

```
##  [1] 12.8 12.8 15.0 15.0 12.5 13.3 13.4 12.4 12.5 13.8 13.5 12.6 12.5 12.8 12.8
## [16] 14.0 13.7 13.7 12.7 12.5 12.8 12.6 15.6 12.5 13.0 12.5 13.3 12.4 12.5 12.9
## [31] 14.3 12.4 15.5 15.5 15.6 13.0 12.7 13.0 12.7 12.4 12.7 13.2 13.2 13.2 15.9
## [46] 13.3 12.9 12.6 12.6
```

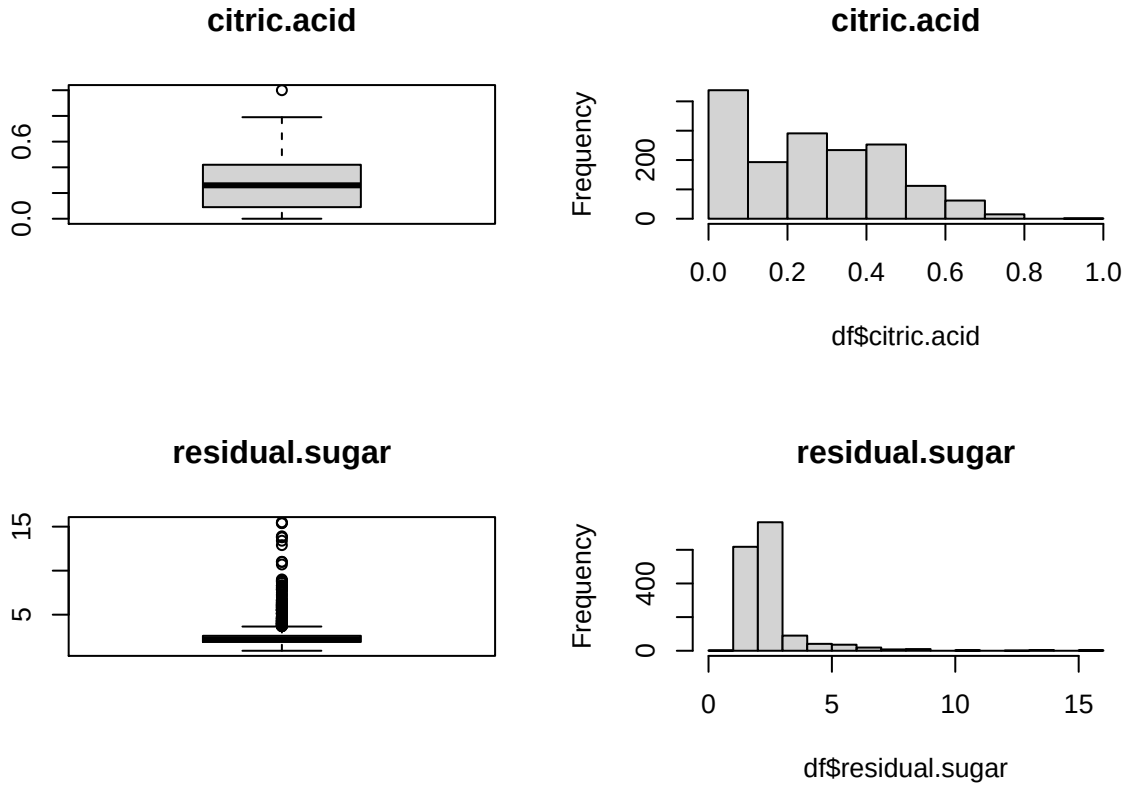
```
boxplot.stats(df$volatile.acidity)$out
```

```
##  [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
```

```
## [13] 1.025 1.115 1.020 1.020 1.580 1.180 1.040
```

```
par(mfrow=c(2,2))
boxplot(df$citric.acid,main="citric.acid")
hist(df$citric.acid,main="citric.acid")

boxplot(df$residual.sugar,main="residual.sugar")
hist(df$residual.sugar,main="residual.sugar")
```




```
boxplot.stats(df$citric.acid)$out
```

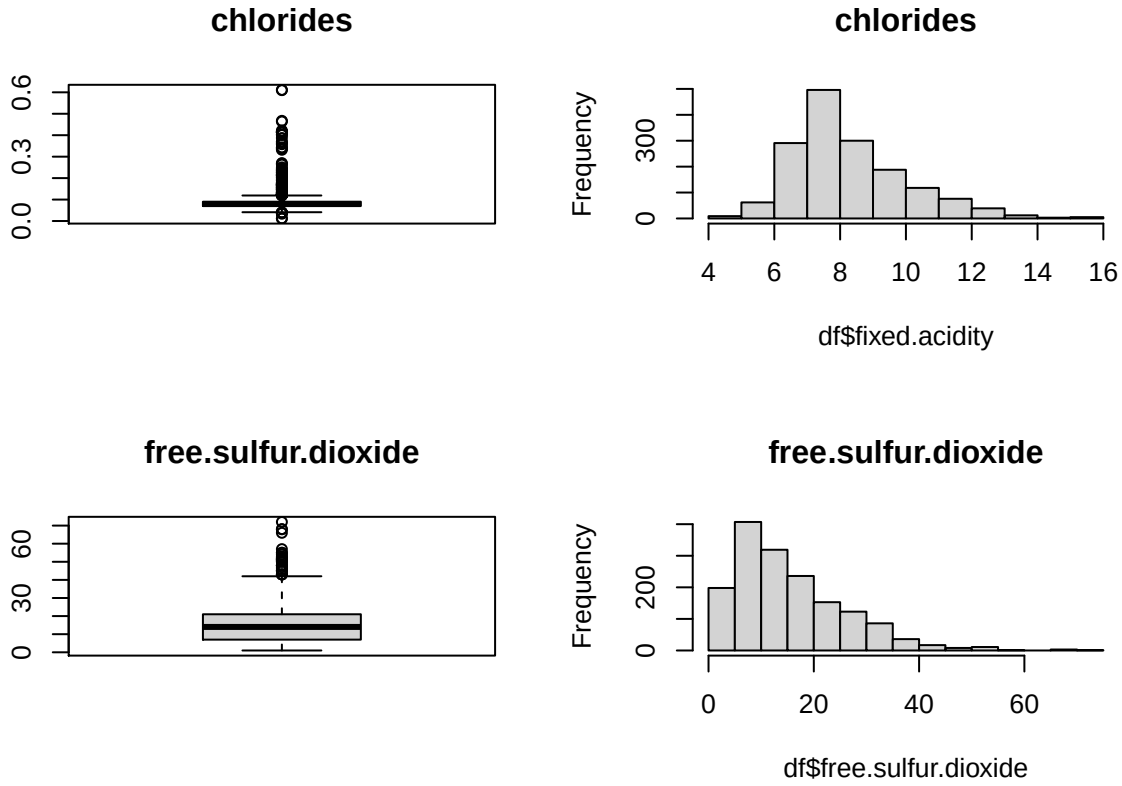
```
## [1] 1
```

```
boxplot.stats(df$residual.sugar)$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
##  [13]  4.65  5.50  5.50  5.50  5.50  7.30  7.20  3.80  5.60  4.00  4.00  4.00
##  [25]  4.00  7.00  4.00  4.00  6.40  5.60  5.60 11.00 11.00  4.50  4.80  5.80
##  [37]  5.80  3.80  4.40  6.20  4.20  7.90  7.90  3.70  4.50  6.70  6.60  3.70
##  [49]  5.20 15.50  4.10  8.30  6.55  6.55  4.60  6.10  4.30  5.80  5.15  6.30
##  [61]  4.20  4.20  4.60  4.20  4.60  4.30  4.30  7.90  4.60  5.10  5.60  5.60
##  [73]  6.00  8.60  7.50  4.40  4.25  6.00  3.90  4.20  4.00  4.00  4.00  6.60
##  [85]  6.00  6.00  3.80  9.00  4.60  8.80  8.80  5.00  3.80  4.10  5.90  4.10
##  [97]  6.20  8.90  4.00  3.90  4.00  8.10  8.10  6.40  6.40  8.30  8.30  4.70
## [109]  5.50  5.50  4.30  5.50  3.70  6.20  5.60  7.80  4.60  5.80  4.10 12.90
## [121]  4.30 13.40  4.80  6.30  4.50  4.50  4.30  4.30  3.90  3.80  5.40  3.80
## [133]  6.10  3.90  5.10  5.10  3.90 15.40 15.40  4.80  5.20  5.20  3.75 13.80
## [145] 13.80  5.70  4.30  4.10  4.10  4.40  3.70  6.70 13.90  5.10  7.80
```

```
par(mfrow=c(2,2))
boxplot(df$chlorides,main="chlorides")
hist(df$fixed.acidity,main="chlorides")

boxplot(df$free.sulfur.dioxide,main="free.sulfur.dioxide")
hist(df$free.sulfur.dioxide,main="free.sulfur.dioxide")
```




```
boxplot.stats(df$chlorides)$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
##  [13] 0.236 0.610 0.360 0.270 0.039 0.337 0.263 0.611 0.358 0.343 0.186 0.213
##  [25] 0.214 0.121 0.122 0.122 0.128 0.120 0.159 0.124 0.122 0.122 0.174 0.121
##  [37] 0.127 0.413 0.152 0.152 0.125 0.122 0.200 0.171 0.226 0.226 0.250 0.148
##  [49] 0.122 0.124 0.124 0.143 0.222 0.039 0.157 0.422 0.034 0.387 0.415 0.157
##  [61] 0.157 0.243 0.241 0.190 0.132 0.126 0.038 0.165 0.145 0.147 0.012 0.012
##  [73] 0.039 0.194 0.132 0.161 0.120 0.120 0.123 0.123 0.414 0.216 0.171 0.178
##  [85] 0.369 0.166 0.166 0.136 0.132 0.132 0.123 0.123 0.123 0.403 0.137 0.414
##  [97] 0.166 0.168 0.415 0.153 0.415 0.267 0.123 0.214 0.214 0.169 0.205 0.205
## [109] 0.039 0.235 0.230 0.038
```

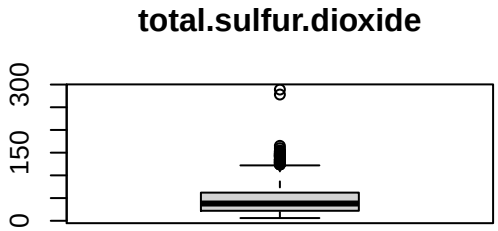
```
boxplot.stats(df$free.sulfur.dioxide)$out
```

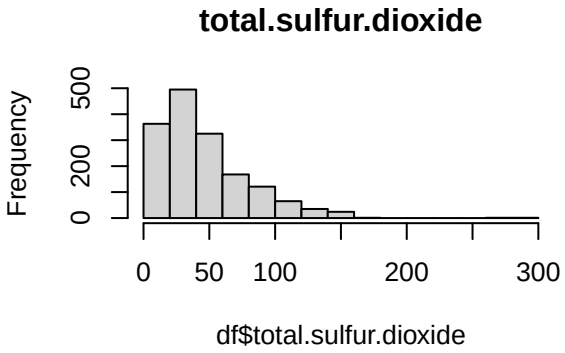
```
##  [1] 52 51 50 68 68 43 47 54 46 45 53 52 51 45 57 50 45 48 43 48 72 43 51 51 52
## [26] 55 55 48 48 66
```

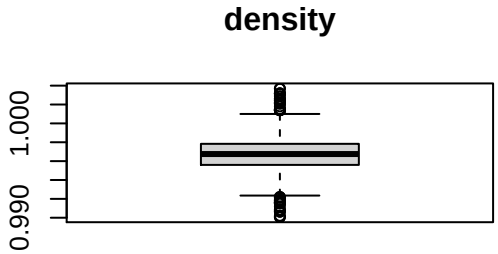
```
par(mfrow=c(2,2))
boxplot(df$total.sulfur.dioxide,main="total.sulfur.dioxide")
```

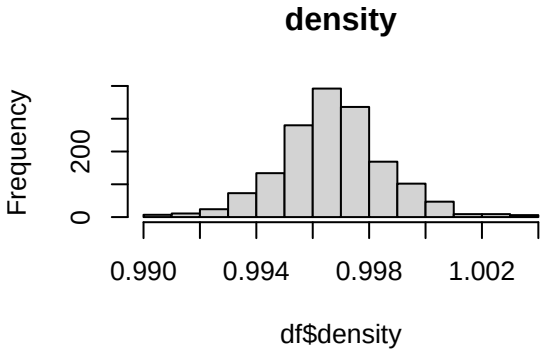
```
hist(df$total.sulfur.dioxide,main="total.sulfur.dioxide")

boxplot(df$density,main="density")
hist(df$density,main="density")
```










```
boxplot.stats(df$total.sulfur.dioxide)$out
```

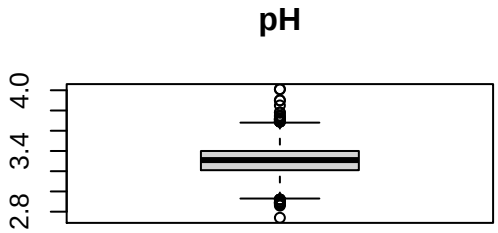
```
##  [1] 145 148 136 125 140 136 133 153 134 141 129 128 129 128 143 144 127 126 145
## [20] 144 135 165 124 124 134 124 129 151 133 142 149 147 145 148 155 151 152 125
## [39] 127 139 143 144 130 278 289 135 160 141 141 133 147 147 131 131 131
```

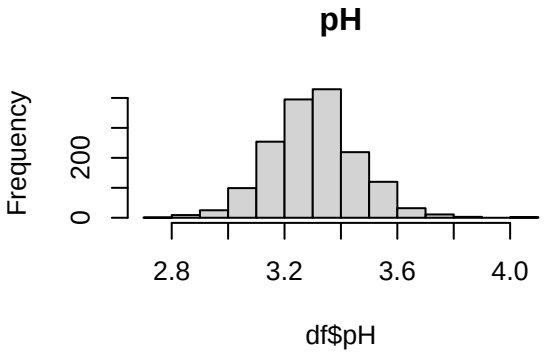
```
boxplot.stats(df$density)$out
```

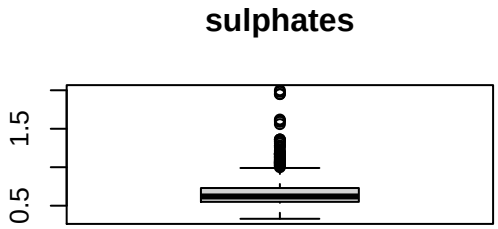
```
##  [1] 0.99160 0.99160 1.00140 1.00150 1.00150 1.00180 0.99120 1.00220 1.00220
## [10] 1.00140 1.00140 1.00140 1.00140 1.00320 1.00260 1.00140 1.00315 1.00315
## [19] 1.00315 1.00210 1.00210 0.99170 0.99220 1.00260 0.99210 0.99154 0.99064
## [28] 0.99064 1.00289 0.99162 0.99007 0.99007 0.99020 0.99220 0.99150 0.99157
## [37] 0.99080 0.99084 0.99191 1.00369 1.00369 1.00242 0.99182 1.00242 0.99182
```

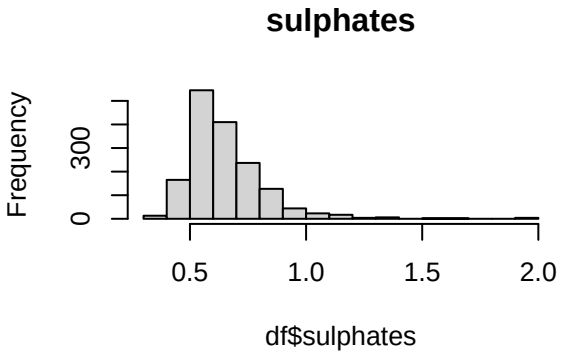
```
par(mfrow=c(2,2))
boxplot(df$pH,main="pH")
hist(df$pH,main="pH")

boxplot(df$sulphates,main="sulphates")
hist(df$sulphates,main="sulphates")
```









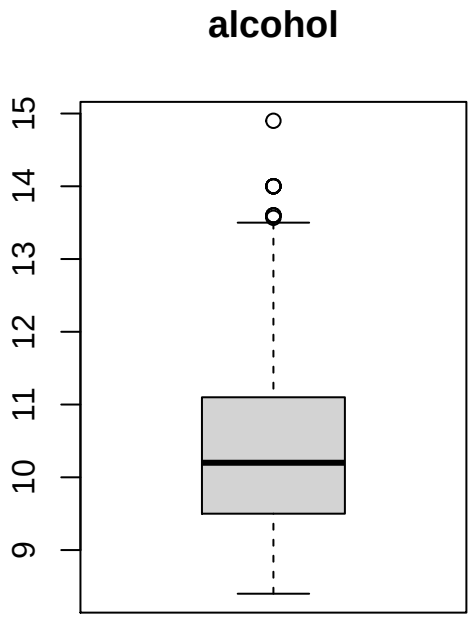

```
boxplot.stats(df$pH)$out
```

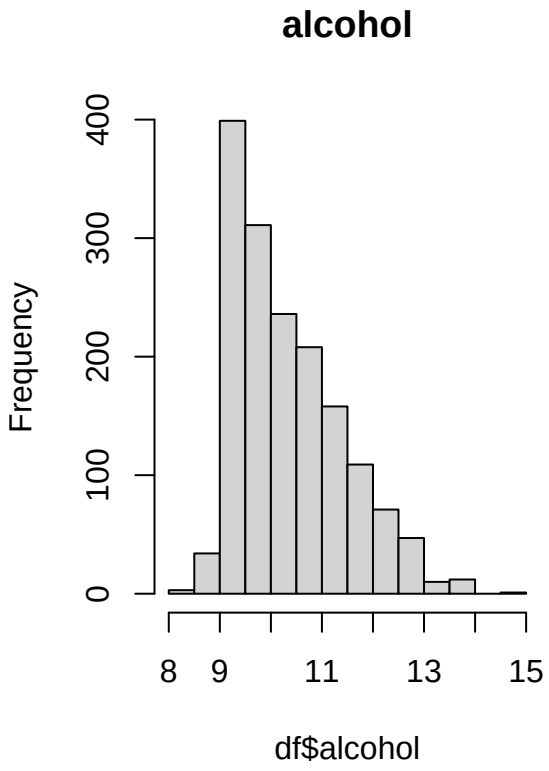
```
##  [1] 3.90 3.75 3.85 2.74 3.69 3.69 2.88 2.86 3.74 2.92 2.92 2.92 3.72 2.87 2.89
## [16] 2.89 2.92 3.90 3.71 3.69 3.69 3.71 3.71 2.89 2.89 3.78 3.70 3.78 4.01 2.90
## [31] 4.01 3.71 2.88 3.72 3.72
```

```
boxplot.stats(df$sulphates)$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
## [16] 1.02 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
## [31] 1.06 1.05 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
## [46] 1.07 1.34 1.16 1.10 1.15 1.17 1.17 1.33 1.18 1.17 1.03 1.17 1.10 1.01
```

```
par(mfrow=c(1,2))
boxplot(df$alcohol,main="alcohol")
hist(df$alcohol,main="alcohol")
```






```
boxplot.stats(df$alcohol)$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
```

Revisando los diagramas de caja, notamos presencia de valores extremos en todas las variables, pero se detalla algunas particularidades en relación al histograma.

- La variable **citric.acid** si nos fijamos en el histograma tiene varias entradas en cero que se puede tratar de valores reales o incompletos, por lo que el diagrama de caja muestra 1 outliers mayor a cero.
- Las variables **fixed.acidity, chlorides, residual.sugar, density y pH** a pesar que presentan outlier visualmente (histograma), se puede deducir que tienden a una distrbución normal.
- Las variables **total.sulfur.dioxide, sulphates, alcohol** es de cola larga hacia la derecha lo que produce los valores extremos presentes.

En **conclusión** la elaboración de los vinos se ve influenciado por los elementos físico/químico que lo componen de esa manera se puede obtener una gran variedad, por tal razón no se trata de valores extremos sino datos reales que se presentan.

Análisis de los datos.

Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar)

A continuación de determina los grupos del conjunto de datos que pueden resultar interesantes para su posterior análisis:

```
TotalResidualSugar <- quantile(df$residual.sugar,probs =0.75)
df.altoResidualSugar <- df[df$residual.sugar>=TotalResidualSugar,]$quality
df.bajoResidualSugar <- df[df$residual.sugar<TotalResidualSugar,]$quality
```

```
totalSulfurDioxide <- quantile(df$total.sulfur.dioxide,probs =0.75)
df.altoTotalSulfurDioxide <- df[df$total.sulfur.dioxide>=totalSulfurDioxide,]$quality
df.bajoTotalSulfurDioxide <- df[df$total.sulfur.dioxide<totalSulfurDioxide,]$quality
```

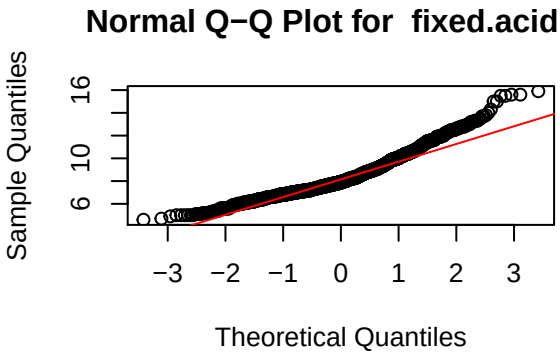
```
TotalDensity <- quantile(df$density,probs =0.75)
df.altoAlcohol <- df[df$density>=TotalDensity,]$quality
df.bajoAlcohol <- df[df$density<TotalDensity,]$quality
```

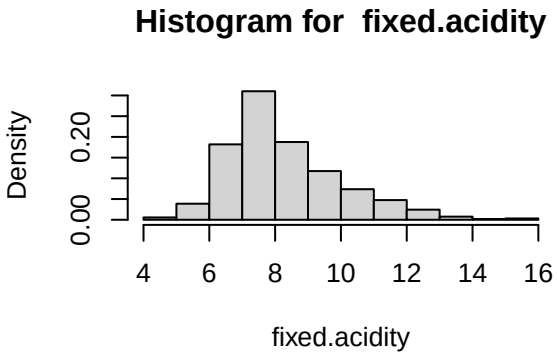
```
Totalalcohol <- quantile(df$alcohol,probs =0.75)
df.altoAlcohol <- df[df$alcohol>=Totalalcohol,]$quality
df.bajoAlcohol <- df[df$alcohol<Totalalcohol,]$quality
```

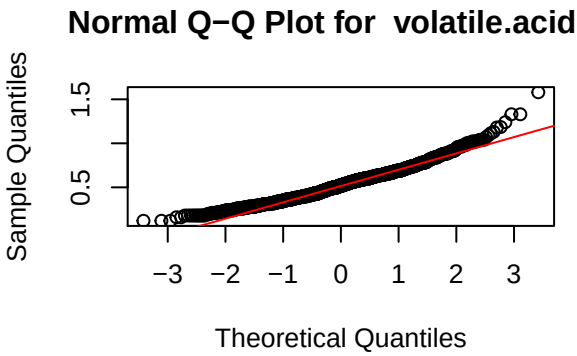
Comprobación de la normalidad y homogeneidad de la varianza.

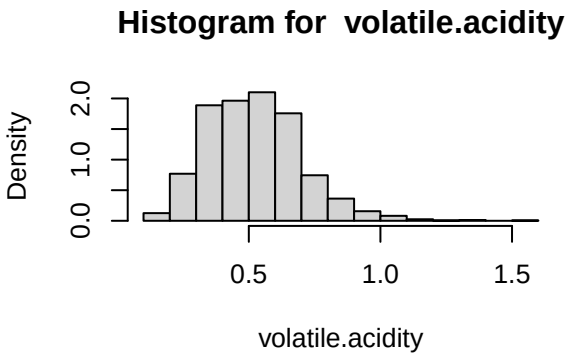
Verificamos si las variables presentan normalidad, aunque en los apartados anteriores ya se hizo un análisis previo de la distribución. Esta vez vamos a graficar mediante quartile e histograma.

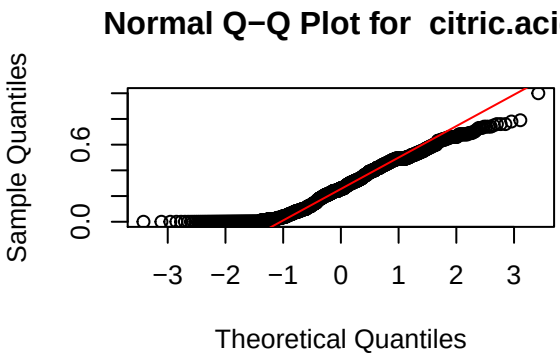
```
par(mfrow=c(2,2))
for(i in 1:ncol(df)-1) {
if (is.numeric(df[,i])){
qqnorm(df[,i],main = paste("Normal Q-Q Plot for ",colnames(df)[i]))
qqline(df[,i],col="red")
hist(df[,i],
main=paste("Histogram for ", colnames(df)[i]),
xlab=colnames(df)[i],freq =FALSE)
}
}
```

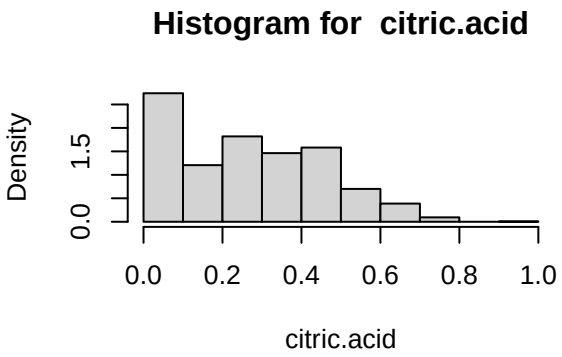

Normal Q−Q Plot for fixed.acidity
Sample Quantiles
6
10
16
−3
−2
−1
0
1
2
3
Theoretical Quantiles

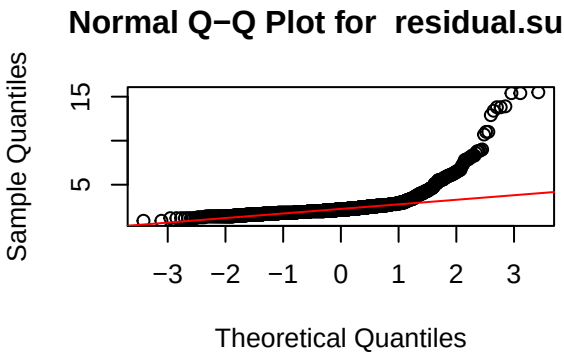

Histogram for fixed.acidity
Density
0.00
0.20
4
6
8
10
12
14
16
fixed.acidity

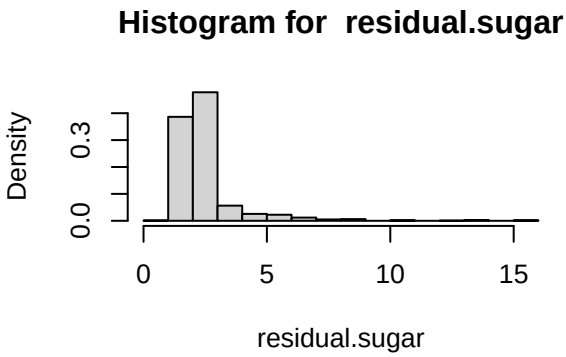

Normal Q−Q Plot for volatile.acidity
Sample Quantiles
0.5
1.5
−3
−2
−1
0
1
2
3
Theoretical Quantiles

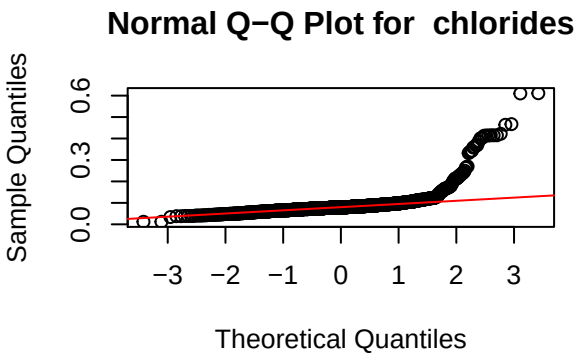

Histogram for volatile.acidity
Density
0.0
1.0
2.0
0.5
1.0
1.5
volatile.acidity

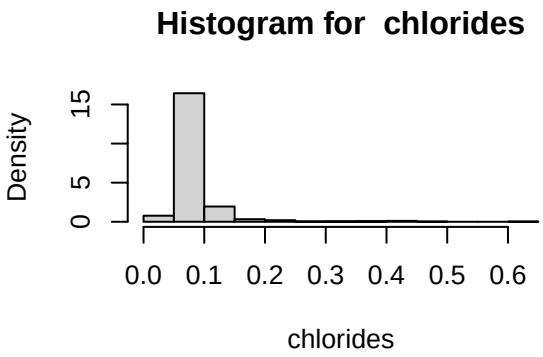

Normal Q−Q Plot for citric.acid
Sample Quantiles
0.0
0.6
−3
−2
−1
0
1
2
3
Theoretical Quantiles

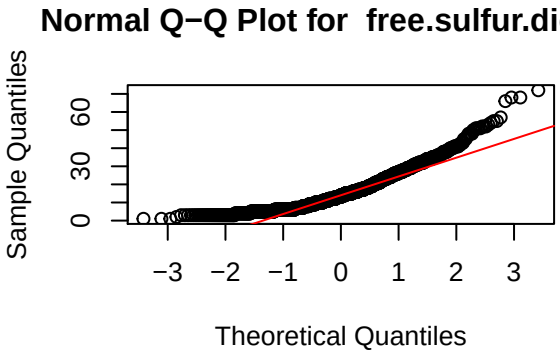

Histogram for citric.acid
Density
0.0
1.5
0.0
0.2
0.4
0.6
0.8
1.0
citric.acid

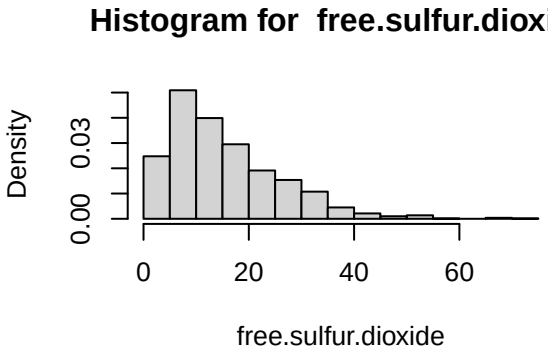

Normal Q−Q Plot for residual.sugar
Sample Quantiles
5
15
−3
−2
−1
0
1
2
3
Theoretical Quantiles

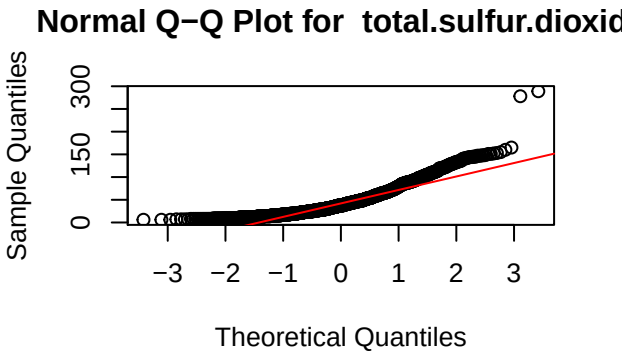

Histogram for residual.sugar
Density
0.0
0.3
0
5
10
15
residual.sugar

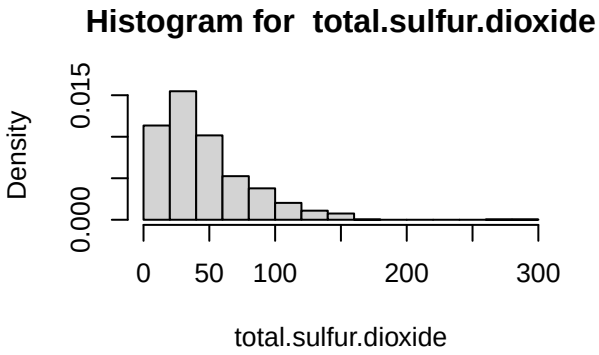

Normal Q−Q Plot for chlorides
Sample Quantiles
0.0
0.3
0.6
−3
−2
−1
0
1
2
3
Theoretical Quantiles

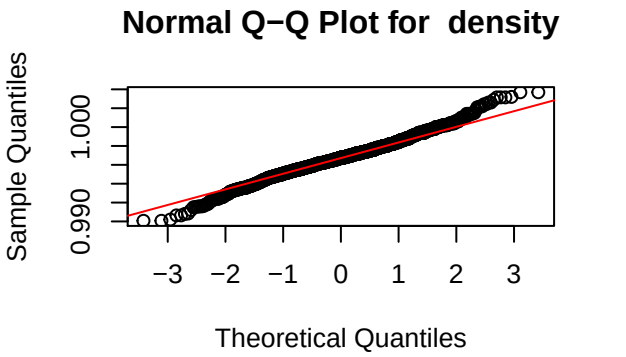

Histogram for chlorides
Density
0
5
15
0.0
0.1
0.2
0.3
0.4
0.5
0.6
chlorides

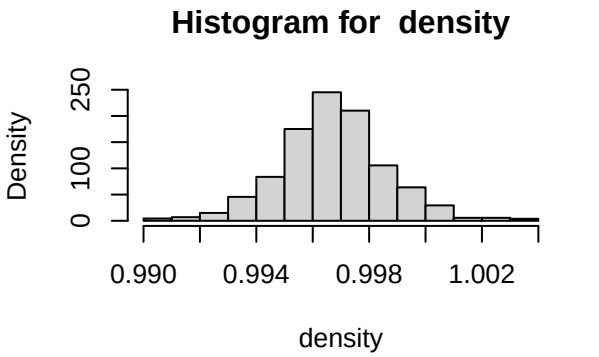

Normal Q−Q Plot for free.sulfur.dioxid
Sample Quantiles
0
30
60
−3
−2
−1
0
1
2
3
Theoretical Quantiles

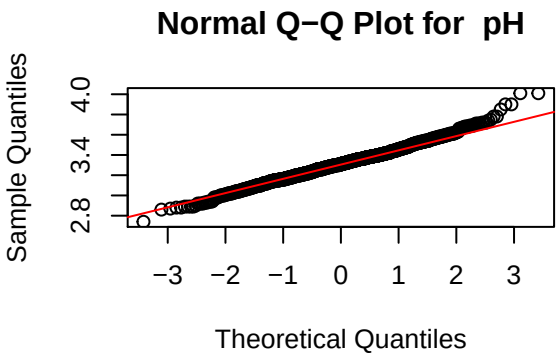

Histogram for free.sulfur.dioxide
Density
0.00
0.03
0
20
40
60
free.sulfur.dioxide

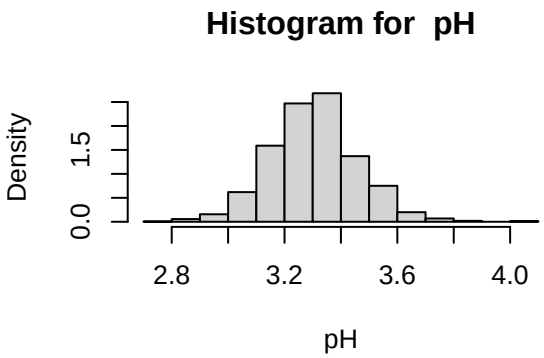

Normal Q−Q Plot for total.sulfur.dioxid
Sample Quantiles
0
150
300
−3
−2
−1
0
1
2
3
Theoretical Quantiles

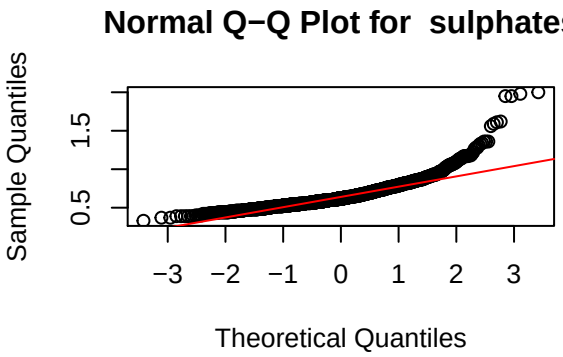

Histogram for total.sulfur.dioxide
Density
0.000
0.015
0
50
100
200
300
total.sulfur.dioxide

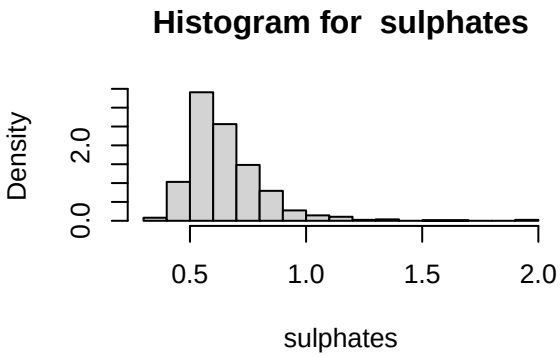

Normal Q−Q Plot for density
Sample Quantiles
0.990
1.000
−3
−2
−1
0
1
2
3
Theoretical Quantiles

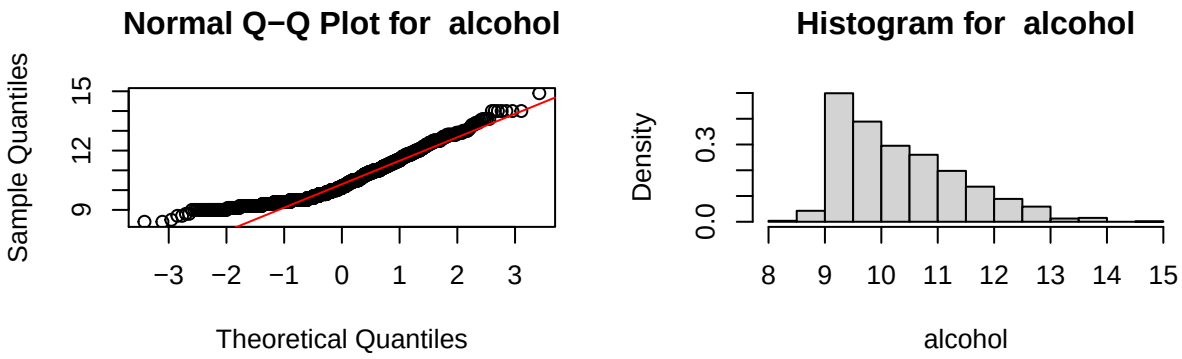

Histogram for density
Density
0
100
250
0.990
0.994
0.998
1.002
density


Normal Q−Q Plot for pH
Sample Quantiles
2.8
3.4
4.0
−3
−2
−1
0
1
2
3
Theoretical Quantiles


Histogram for pH
Density
0.0
1.5
2.8
3.2
3.6
4.0
pH


Normal Q−Q Plot for sulphates
Sample Quantiles
0.5
1.5
−3
−2
−1
0
1
2
3
Theoretical Quantiles


Histogram for sulphates
Density
0.0
2.0
0.5
1.0
1.5
2.0
sulphates



Analizando los quantile-quantile podemos suponer una normalización de las variables. Para afirmar o descartar las gráficas se aplica el test Lilliefors.

```
library("nortest")
alpha =0.05
col.names = colnames(df)
for (i in 1:ncol(df)) {
  if (i == 1) cat("Variables que no siguen una distribución normal:\n")
    if (is.integer(df[,i]) |is.numeric(df[,i])) {
        p_val = lillie.test(df[,i])$p.value
    if (p_val < alpha) {
        cat(col.names[i])
        # Format output
    if (i <ncol(df) - 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
```

El test Lilliefors, nos indica que las variables realmente no siguen una distribución normal, este suceso se da principalmente por los valores extremos que se analizaron en los apartados anteriores,

pero como lo habiamos manifestado no se pueden eliminar porque son parte del análisis.

Aplicando el ***Teorema del Límite Central*** el mismo que manifiesta, dada una muestra suficientemente grande (>30), la distribución de las medias muestrales seguirá una distribución normal.

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

Correlaciones:

Se procede a realizar un análisis de correlaciones con las distintas variables con la finalidad de conocer cuales presentan mayor peso a la hora de determinar la calidad del vino (variable **quality**):

```
corr_matrix <- matrix(nc =2,nr =0)
colnames(corr_matrix) <- c("estimate","p-value")

# Calcular el coeficiente de correlación para cada variable cuantitativa
# con respecto al campo "quality"
for (i in 1:(ncol(df) - 1)) {
  if (is.integer(df[,i]) |is.numeric(df[,i])) {
    spearman_test = cor.test(df[,i],df[,length(df)],method ="spearman")
    corr_coef =spearman_test$estimate
    p_val =spearman_test$p.value
# Add row to matrix
    pair = matrix(ncol =2,nrow =1)
    pair[1][1] =corr_coef
    pair[2][1] =p_val
    corr_matrix <- rbind(corr_matrix, pair)
    rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(df)[i]
  }
}
```

```
## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
```

```
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties

## Warning in cor.test.default(df[, i], df[, length(df)], method = "spearman"):
## Cannot compute exact p-value with ties
```

```
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
```

De acuerdo a los resultados obtenidos las 3 principales variables que tienen mayor relevancia son: _alcohol, volatile.acidity y sulphates.

Contraste de Hipótesis:

Realizamos algunos contrastes de hipótesis para algunas variables.

Contraste de hipótesis unilateral sobre la cantidad de azúcar (df.bajoResidualSugar):

Hipótesis nula $H_0 : {}_1 = {}_2$

Hipótesis alternativa $H_1 : {}_1 < {}_2$

```
t.test(df.bajoResidualSugar, df$quality,alternative ="less")
```

```
##
##  Welch Two Sample t-test
##
## data:  df.bajoResidualSugar and df$quality
## t = -0.026921, df = 2532, p-value = 0.4893
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##        -Inf 0.04986301
## sample estimates:
## mean of x mean of y
##  5.635193  5.636023
```

Contraste de hipótesis unilateral sobre la cantidad total de Dióxido de sulfuro (df.altoTotalSulfurDioxide):

Hipótesis nula $H_0 : {}_1 = {}_2$

Hipótesis alternativa $H_1 : {}_1 < {}_2$

```
t.test(df.altoTotalSulfurDioxide, df$quality,alternative ="less")
```

```
##
##  Welch Two Sample t-test
##
## data:  df.altoTotalSulfurDioxide and df$quality
## t = -6.677, df = 752.79, p-value = 2.368e-11
## alternative hypothesis: true difference in means is less than 0
## 95 percent confidence interval:
##        -Inf -0.1922452
## sample estimates:
## mean of x mean of y
##  5.380835  5.636023
```

Contraste de hipótesis unilateral sobre la cantidad de alcohol (df.altoAlcohol):

Hipótesis nula $H_0 : {}_1 = {}_2$

Hipótesis alternativa $H_1 : {}_1 > {}_2$

```
t.test(df.altoAlcohol, df$quality,alternative ="greater")
```

```
##
##  Welch Two Sample t-test
##
## data:  df.altoAlcohol and df$quality
## t = 12.555, df = 628.55, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.4891266       Inf
## sample estimates:
## mean of x mean of y
##  6.199017  5.636023
```

Regresión

Regresión Logística

Transformación de la variable quality en categórica:

```
quality_2 <- ifelse(df$quality>6,yes=1,no=0)
df$quality2 <- quality_2
```

Regresión logística con todas las variables:

```
rlg1 <- glm(quality2 ~ . -quality ,family=binomial(logit),data=df)
summary(rlg1)
```

```
##
## Call:
## glm(formula = quality2 ~ . - quality, family = binomial(logit),
##     data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.9878  -0.4351  -0.2207  -0.1222   2.9869
##
```

```
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            2.428e+02  1.081e+02   2.247 0.024660 *
## fixed.acidity          2.750e-01  1.253e-01   2.195 0.028183 *
## volatile.acidity      -2.581e+00  7.843e-01  -3.291 0.000999 ***
## citric.acid            5.678e-01  8.385e-01   0.677 0.498313
## residual.sugar         2.395e-01  7.373e-02   3.248 0.001163 **
## chlorides             -8.816e+00  3.365e+00  -2.620 0.008788 **
## free.sulfur.dioxide    1.082e-02  1.223e-02   0.884 0.376469
## total.sulfur.dioxide -1.653e-02  4.894e-03  -3.378 0.000731 ***
## density               -2.578e+02  1.104e+02  -2.335 0.019536 *
## pH                     2.242e-01  9.984e-01   0.225 0.822327
## sulphates              3.750e+00  5.416e-01   6.924 4.39e-12 ***
## alcohol                7.533e-01  1.316e-01   5.724 1.04e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1269.92  on 1598  degrees of freedom
## Residual deviance:  870.86  on 1587  degrees of freedom
## AIC: 894.86
##
## Number of Fisher Scoring iterations: 6
```

```
summary(rlg1)$aic
```

```
## [1] 894.8644
```

Regresión logística con las variables significativas de acuerdo al análisis anterior con un p-value menor o igual a 0.05:

```
rlg2 <- glm(quality2 ~ fixed.acidity + volatile.acidity + residual.sugar + chlorides + total.sulfur.dio
summary(rlg2)
```

```
##
## Call:
## glm(formula = quality2 ~ fixed.acidity + volatile.acidity + residual.sugar +
##     chlorides + total.sulfur.dioxide + density + sulphates +
##     alcohol, family = binomial(logit), data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.0158  -0.4314  -0.2220  -0.1255   2.9883
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)            2.268e+02  9.163e+01   2.475 0.013336 *
## fixed.acidity          2.812e-01  8.029e-02   3.502 0.000462 ***
## volatile.acidity      -2.913e+00  6.467e-01  -4.504 6.66e-06 ***
## residual.sugar         2.328e-01  7.009e-02   3.322 0.000893 ***
## chlorides             -8.441e+00  3.259e+00  -2.590 0.009593 **
## total.sulfur.dioxide -1.360e-02  3.447e-03  -3.946 7.95e-05 ***
## density               -2.409e+02  9.202e+01  -2.618 0.008835 **
## sulphates              3.699e+00  5.287e-01   6.997 2.62e-12 ***
## alcohol                7.823e-01  1.120e-01   6.983 2.88e-12 ***
```

```
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1269.92  on 1598  degrees of freedom
## Residual deviance:  872.08  on 1590  degrees of freedom
## AIC: 890.08
##
## Number of Fisher Scoring iterations: 6
```

```
summary(rlg2)$aic
```

```
## [1] 890.076
```

Regresión logística con las variables significativas de acuerdo a la clasificación realizada en el apartado de correlaciones:

```
rlg3 <- glm(quality2 ~  alcohol+ volatile.acidity + sulphates +  citric.acid  +  total.sulfur.dioxide +
summary(rlg3)
```

```
##
## Call:
## glm(formula = quality2 ~ alcohol + volatile.acidity + sulphates +
##     citric.acid + total.sulfur.dioxide + chlorides + density +
##     fixed.acidity, family = binomial(logit), data = df)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.6174  -0.4367  -0.2277  -0.1278   2.9855
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           9.584e+01  7.987e+01   1.200 0.230111
## alcohol               8.918e-01  1.073e-01   8.315  < 2e-16 ***
## volatile.acidity     -2.732e+00  7.757e-01  -3.521 0.000429 ***
## sulphates             3.413e+00  5.196e-01   6.570 5.04e-11 ***
## citric.acid           7.148e-01  8.215e-01   0.870 0.384235
## total.sulfur.dioxide -1.223e-02  3.553e-03  -3.443 0.000576 ***
## chlorides            -8.796e+00  3.379e+00  -2.603 0.009236 **
## density              -1.094e+02  8.017e+01  -1.364 0.172531
## fixed.acidity         1.716e-01  8.666e-02   1.981 0.047620 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1269.92  on 1598  degrees of freedom
## Residual deviance:  880.85  on 1590  degrees of freedom
## AIC: 898.85
##
## Number of Fisher Scoring iterations: 6
```

```
summary(rlg3)$aic
```

```
## [1] 898.8508
```

Regresión logística ajustando el modelo con las 4 variables mejor correlacionadas tomando el criterio de correlaciones:

```
rlg4 <- glm(quality2 ~  alcohol+ volatile.acidity + sulphates +  citric.acid ,family=binomial(logit),da
summary(rlg4)$aic
```

```
## [1] 924.1143
```

Tomando en cuenta el criterio de información de Akaike (AIC), el último modelo presenta una mejor calidad del modelo en comparación a los anteriores, pero consideramos que estaria sobreajustado limitando demasiado el conjunto, por tal motivo se considera como un modelo bueno al **rlg3** tomando en cuenta lo criterios de correlaciones.

Datos de entrenamiento y prueba

División del conjuntos de datos en datos de entrenamiento y prueba, lo dividiremos en un 80% de datos para conjuntos de datos de entrenamiento y un 20% de datos para conjuntos de datos de prueba.

```
set.seed(1)
sampleSize <- round(nrow(df)*0.8)
idx <- sample(seq_len(sampleSize), size = sampleSize)

X.train_red <- df[idx,]
X.test_red <- df[-idx,]

rownames(X.train_red) <- NULL
rownames(X.test_red) <- NULL
```

Regresión multivaribale para predicción a partir de datos de entrenamiento.

Crear el modelo de regresión multivariable

Como se mencionó en el análisis exploratorio de datos, emplearemos todas las variables predictoras, excepto el azúcar residual (residual.sugar), para el modelo. Creemos un modelo multivariable a partir de esas variables (multivariable).

De los resultados que muestra la función *summary*(), se puede ver que aproximadamente la mitad de todas las variables predictoras exhiben insignificancia. Además, el $R - cuadradoajustado = 0.3645$ es un resultado pobre. Antes de abordar este problema, deberimos verificar algunos supuestos para el modelado.

```
model_red1 <- lm(quality ~ fixed.acidity + volatile.acidity + citric.acid + chlorides + free.sulfur
                 data = X.train_red)
summary(model_red1)
```

```
##
## Call:
## lm(formula = quality ~ fixed.acidity + volatile.acidity + citric.acid +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + sulphates + alcohol, data = X.train_red)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.67536 -0.38553 -0.06879  0.45454  1.97578
##
## Coefficients:
```

```
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           1.510e+01  1.912e+01   0.790 0.429935
## fixed.acidity         1.547e-02  2.696e-02   0.574 0.566286
## volatile.acidity     -1.057e+00  1.358e-01  -7.780 1.49e-14 ***
## citric.acid          -1.774e-01  1.657e-01  -1.070 0.284621
## chlorides            -1.779e+00  4.627e-01  -3.845 0.000126 ***
## free.sulfur.dioxide   3.392e-03  2.480e-03   1.368 0.171691
## total.sulfur.dioxide -3.645e-03  8.201e-04  -4.444 9.58e-06 ***
## density              -1.102e+01  1.954e+01  -0.564 0.572664
## pH                   -3.835e-01  2.010e-01  -1.908 0.056598 .
## sulphates             7.945e-01  1.217e-01   6.527 9.67e-11 ***
## alcohol               2.924e-01  2.462e-02  11.878  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6476 on 1268 degrees of freedom
## Multiple R-squared:  0.3695, Adjusted R-squared:  0.3645
## F-statistic:  74.3 on 10 and 1268 DF,  p-value: < 2.2e-16
```

En este capítulo, analizaremos el mejor modelo hasta ahora y lo usaremos para predecir el conjunto de datos de prueba. En primer lugar, debemos interpretar el modelo seleccionado. Posteriormente, se discute el desempeño del modelo y las predicciones se realizarán posteriormente.

El modelo que se determino tiene todas las variables preditoras (11 en total) para el conjunto de datos de entrenamiento, a partir del cual determinamos la siguiente ecuación:

Ahora construimos el modelo multivariable para determinar la calidad de vino en función de todas las variables de clase:

$$Y_{estimado} = \beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+\beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_9+\beta_{10}X_{10}+\beta_{11}X_{11}$$

, donde β_1 a β_{11} son los pesos para las variables X_1 a X_{11}, y β_0 es la intersección con el eje Y.

```r
model_redAll <- lm(quality ~ ., data = X.train_red)
summary(model_redAll)
```

```
##
## Call:
## lm(formula = quality ~ ., data = X.train_red)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.4926 -0.3100 -0.0546  0.3957  1.1842
##
## Coefficients:
##                         Estimate Std. Error t value Pr(>|t|)
## (Intercept)           -3.150e+01  1.836e+01  -1.716 0.086467 .
## fixed.acidity         -2.598e-02  2.302e-02  -1.129 0.259227
## volatile.acidity      -8.414e-01  1.056e-01  -7.967 3.61e-15 ***
## citric.acid           -3.352e-01  1.285e-01  -2.609 0.009185 **
## residual.sugar        -3.119e-02  1.385e-02  -2.253 0.024460 *
## chlorides             -9.645e-01  3.604e-01  -2.676 0.007536 **
## free.sulfur.dioxide    4.712e-03  1.924e-03   2.449 0.014442 *
## total.sulfur.dioxide -2.711e-03  6.383e-04  -4.247 2.33e-05 ***
## density                3.763e+01  1.876e+01   2.006 0.045065 *
## pH                    -4.981e-01  1.650e-01  -3.019 0.002590 **
## sulphates              3.453e-01  9.702e-02   3.559 0.000386 ***
```

```
## alcohol               1.769e-01  2.270e-02   7.793 1.35e-14 ***
## quality2              1.331e+00  4.573e-02  29.112  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5016 on 1266 degrees of freedom
## Multiple R-squared:  0.6224, Adjusted R-squared:  0.6188
## F-statistic: 173.9 on 12 and 1266 DF,  p-value: < 2.2e-16
```

La función multivariable para determinar la calidad de vino:

$$Y_{estimado} = \beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_5X_5+\beta_6X_6+\beta_7X_7+\beta_8X_8+\beta_9X_9+\beta_{10}X_{10}+\beta_{11}X_{11}$$

, con los siguientes valores calculados :$\beta_0 = -3.150e+01; \beta_1 = -2.598e-02; ....; \beta_{11} = 1.769e-0$

```
model_redAll$coefficients
```

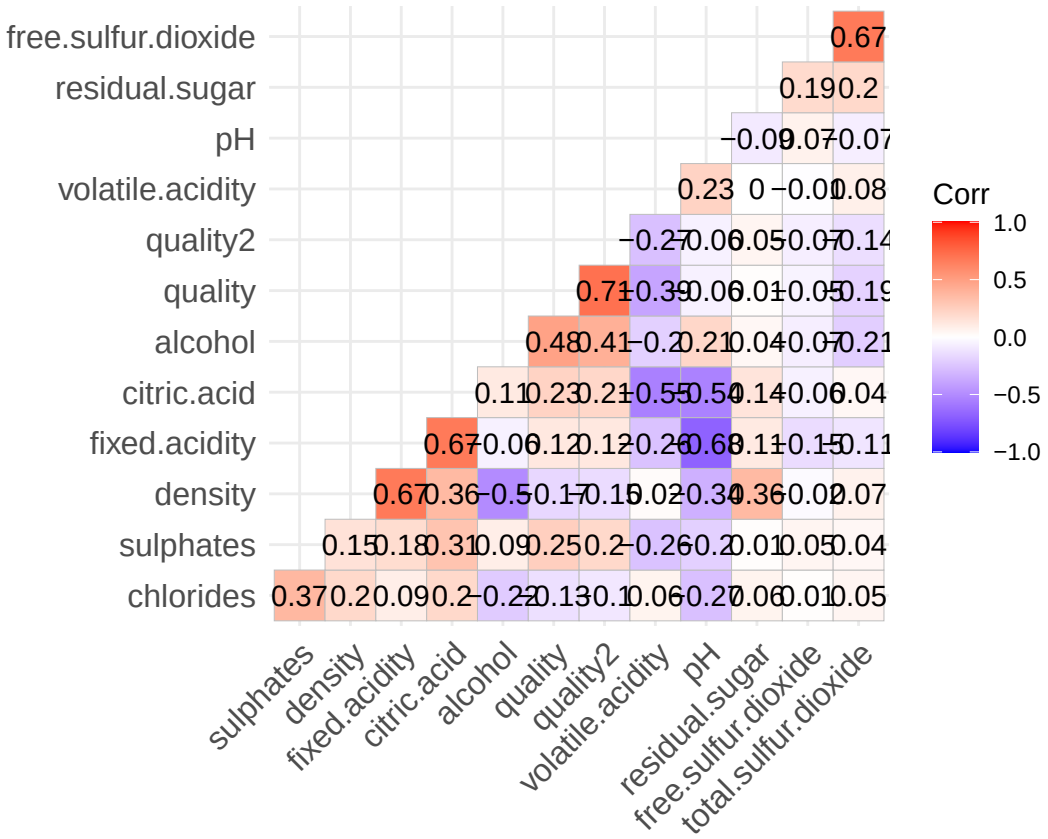
```
##          (Intercept)        fixed.acidity     volatile.acidity
##        -31.504824061         -0.025984080         -0.841354486
##          citric.acid       residual.sugar            chlorides
##         -0.335195275         -0.031189904         -0.964477497
##  free.sulfur.dioxide total.sulfur.dioxide              density
##          0.004712294         -0.002710746         37.634022974
##                   pH            sulphates              alcohol
##         -0.498149888          0.345303565          0.176887081
##            quality2
##          1.331424495
```

#Representación de los resultados a partir de tablas y gráficas

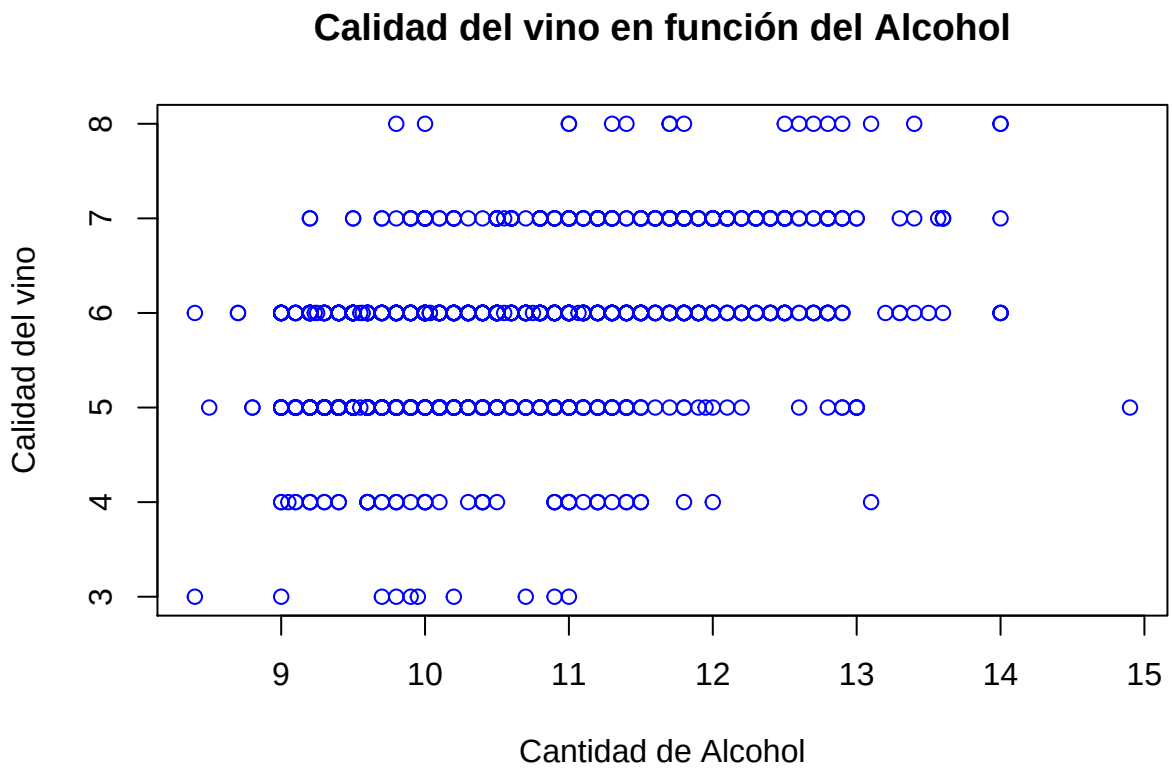
```
library(ggcorrplot)
```

```
## Loading required package: ggplot2
```

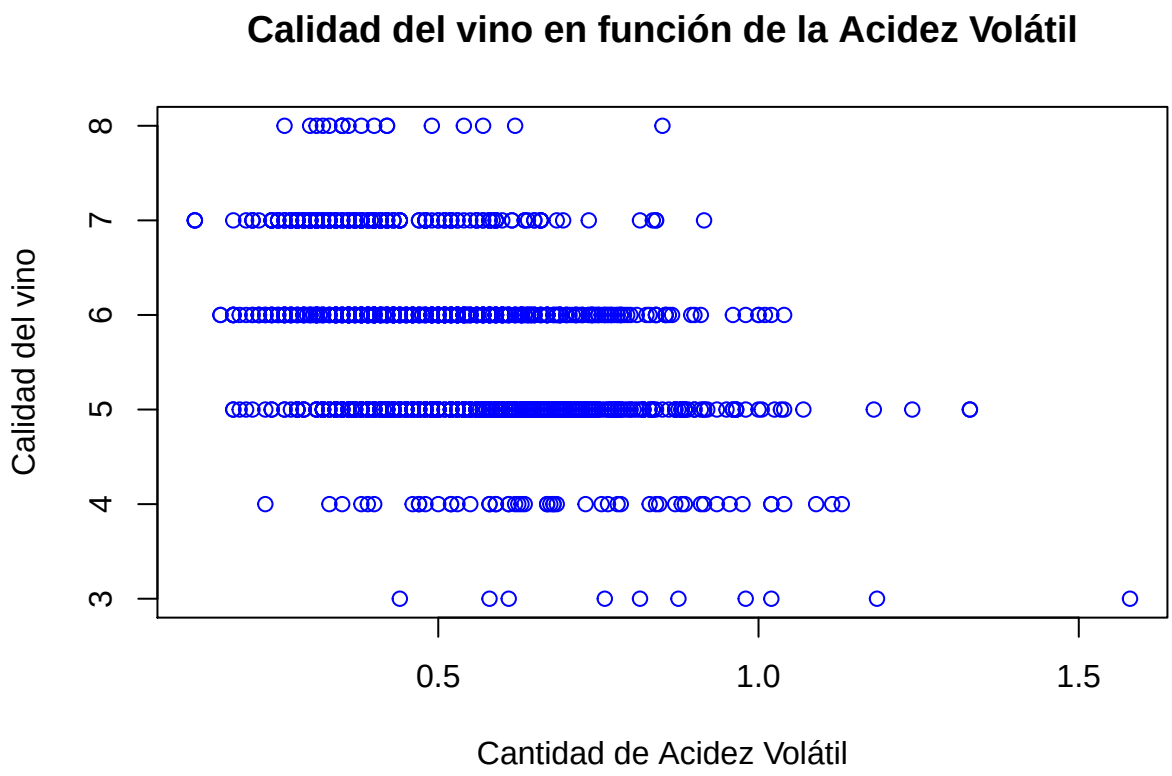
```
#ggcorrplot(cor(df),insig = "blank", lab = TRUE)
ggcorrplot(round(cor(df),digits = 2), hc.order = TRUE, type = "lower", lab = TRUE, insig = "blank")
```




```
plot(df$alcohol,df$quality,col="blue",xlab = "Cantidad de Alcohol",ylab = "Calidad del vino",main = "Ca
```




```
plot(df$volatile.acidity,df$quality,col="blue",xlab = "Cantidad de Acidez Volátil",ylab = "Calidad del
```



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

A través del desarrollo del presente trabajo de investigación se justificó la importancia del conjunto de datos seleccionado, se describió cada una de sus variables de clase, se verificó el tipo de dato, su dominio de valores; para que a través de sus variables de clase determinar la variable objetivo o target, que sería la calidad del vino. Se realizó la limpieza de datos, tratamiento de los valores extremos, valores ausentes, vacíos, y graficas descriptivas para las variables. Se selecciono grupos de datos para las variables, tal como se muestra en el desarrollo del trabajo, se analizó su distribución del mismo; y como **resultado** de la limpieza de datos aplicamos técnicas para verificar si siguen una distribución normal, las variables seleccionadas. Se realizó la correlación de las variables que intervienen en la calidad de vino, la prueba de hipótesis, se trabajo la correlación logística, la correlación multivariable, que a través de esta última podemos estimar la calidad de vino, en función de las demás variables de clase.