

Práctica 2: Limpieza y validación de los datos

Waziri Ajibola Lawal, David Fernández González

5 de Xaneiro, 2021

Contents

Detalles de la actividad	2
Descripción	2
Competencias	2
Objetivos	2
Realización de la práctica	2
1. Descripción del dataset. ¿Por qué es importante y qué pregunta/problema pretende responder?	2
2. Integración y selección de los datos de interés a analizar.	4
3. Limpieza de los datos.	7
3.1. ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?	7
3.2. Identificación y tratamiento de valores extremos.	7
3.3. Exportación de los datos preprocesados	13
4. Análisis de los datos	13
4.1. Selección de los grupos de datos que se quieren analizar/comparar (planificación de los análisis a aplicar).	13
4.2. Comprobación de la normalidad y homogeneidad de la varianza.	14
4.2.1. Normalidad	14
4.2.2. Homogeneidad de la varianza	19
4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.	20
4.3.1. Correlaciones	20
4.3.2. Modelo de regresión lineal	22
4.3.2. Modelo de regresión logística	24
5. Representación de los resultados a partir de tablas y gráficas.	27
6. Resolución del problema. A partir de los resultados obtenidos, ¿cuáles son las conclusiones? ¿Los resultados permiten responder al problema?	27
Tabla de contribuciones al trabajo	28

Detalles de la actividad

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.

Realización de la práctica

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

El conjunto de datos que se va a analizar es el de Red Wine Quality y se ha obtenido a partir de este enlace en Kaggle (<https://www.kaggle.com/uciml/red-wine-quality-cortez-et-al-2009>). El conjunto de datos de vino tinto contiene 1599 observaciones, 11 predictores y 1 valor categórico que indica la calidad del vino. Entre los campos de este conjunto de datos, encontramos los siguientes:

- fixed acidity: la mayoría de los ácidos involucrados con el vino, o fijos o no volátiles (no se evaporan fácilmente).
- volatile acidity: cantidad de ácido acético en el vino.
- citric acid: cantidad de ácido cítrico en el vino.
- residual sugar: cantidad de azúcar residual en el vino.
- chlorides: cantidad de sal de potasio en el vino.
- free sulfur dioxide: El SO₂ existe en equilibrio, demasiado afectará la salud.
- total sulfur dioxide: cantidad de total de SO₂ en el vino.
- density: la densidad del vino se acerca a la del agua dependiendo del porcentaje de alcohol y del contenido de azúcar.
- pH: describe qué tan ácido o básico es un vino.
- sulphates: un aditivo para el vino que puede contribuir a los niveles de dióxido de azufre (SO₂), que actúa como antimicrobiano y antioxidante.
- alcohol: el porcentaje de contenido de alcohol del vino.
- quality: valor que describe la calidad del vino (basada en datos sensoriales, puntuación entre 0 y 10).

El objetivo principal es encontrar que variables ofrecen más información sobre la calidad del vino. También intentaremos hacer predicciones de la calidad de un vino, y comprobar si se corresponde con su calidad real.

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

Primer contacto con el conjunto de datos, visualizamos su estructura.

```
# Cargamos el fichero de datos
redWineData <- read.csv('winequality-red.csv', stringsAsFactors = FALSE, header = TRUE)
#filas=dim(redWineData)[1]

attach(redWineData)

# Verificamos la dimension del conjunto de datos
dim(redWineData)

## [1] 1599    12

# Verificamos la estructura del conjunto de datos
sapply(redWineData, 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"

# Verificamos la estructura del conjunto de datos
str(redWineData)

## '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 ...

# Verificamos la distribución de los datos
head(redWineData)

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

```

## 5      7.4      0.70      0.00      1.9      0.076
## 6      7.4      0.66      0.00      1.8      0.075
##   free.sulfur.dioxide total.sulfur.dioxide density    pH sulphates alcohol
## 1          11          34  0.9978 3.51      0.56      9.4
## 2          25          67  0.9968 3.20      0.68      9.8
## 3          15          54  0.9970 3.26      0.65      9.8
## 4          17          60  0.9980 3.16      0.58      9.8
## 5          11          34  0.9978 3.51      0.56      9.4
## 6          13          40  0.9978 3.51      0.56      9.4
##   quality
## 1      5
## 2      5
## 3      5
## 4      6
## 5      5
## 6      5

```

Estadísticas básicas, verificamos algunas métricas sobre las variables

```
summary(redWineData)
```

```

## 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

```

```
summary(redWineData$quality)
```

```

##   Min. 1st Qu. Median  Mean 3rd Qu.  Max.
## 3.000 5.000 6.000 5.636 6.000 8.000

```

```
table(redWineData$quality)
```

```

##
##   3   4   5   6   7   8
## 10  53 681 638 199  18

```

Hay 1599 observaciones de 12 variables numéricas.

Quantity es una variable categórica y discreta, con una escala de 0 a 10. Los valores varían sólo de 3 a 8, con una media de 5,6 y una mediana de 6. Todas las demás variables parecen ser cantidades continuas (con la excepción de los sufijos .sulfur.dioxide).

Todos los predictores son valores numéricos, los resultados son enteros.

3. Limpieza de los datos.

3.1. ¿Los datos contienen ceros o elementos vacíos? ¿Cómo gestionarías cada uno de estos casos?

Como podemos observar, no existen valores vacíos en nuestro conjunto de datos.

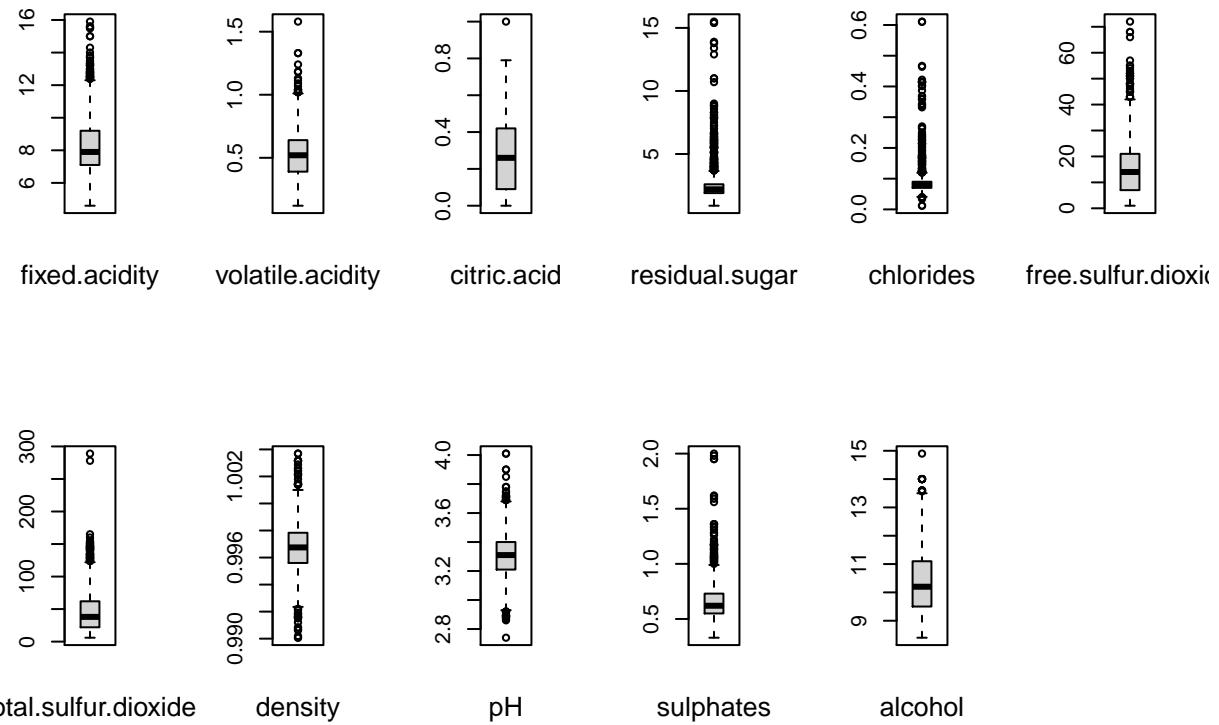
```
# Verificamos si existen valores vacíos en el conjunto de datos  
colSums(is.na(redWineData))
```

```
## 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
```

3.2. Identificación y tratamiento de valores extremos.

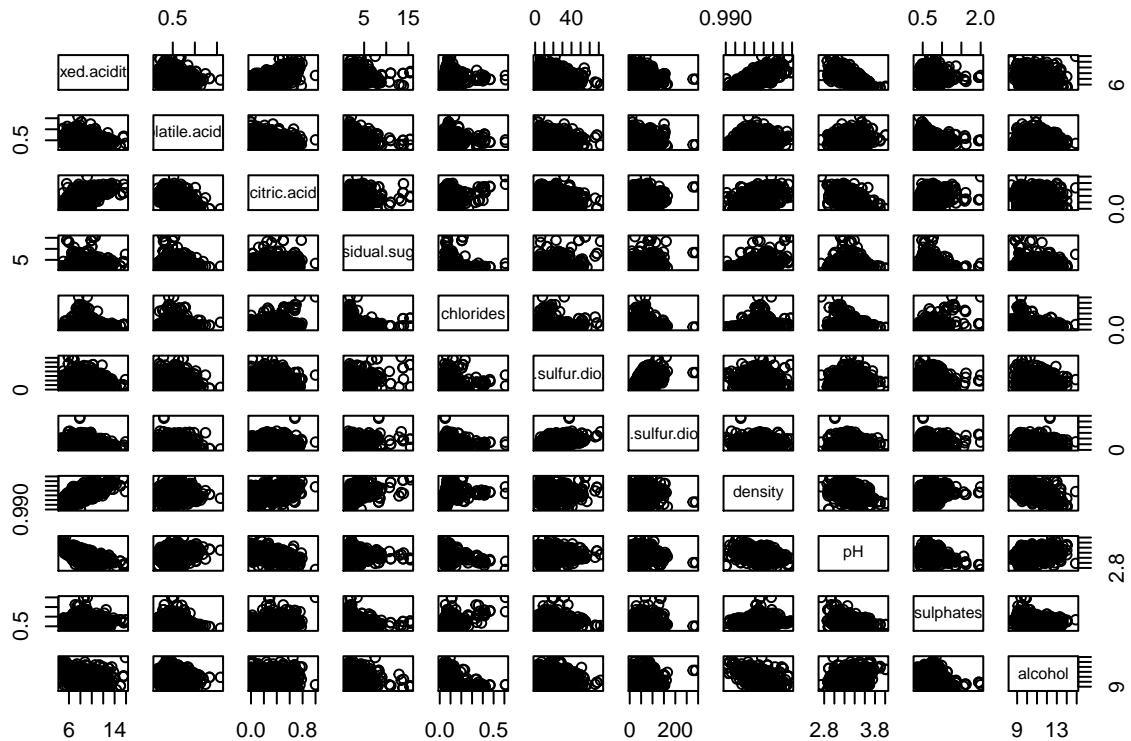
Las métricas del conjunto de datos nos muestran que la mayoría de las variables tienen un rango amplio en comparación con el rango intercuartil, lo que puede indicar una dispersión en los datos y la presencia de valores atípicos. Investigamos más a fondo produciendo diagramas de caja para cada una de las variables:

```
oldpar = par(mfrow = c(2,6))  
for ( i in 1:11 ) {  
  boxplot(redWineData[[i]])  
  mtext(names(redWineData)[i], cex = 0.8, side = 1, line = 2)  
}  
par(oldpar)
```



Para obtener mas información sobre la posición de los valores atípicos, podemos usar la función pairs(), con la que obtendremos una matriz de gráfico de dispersión.

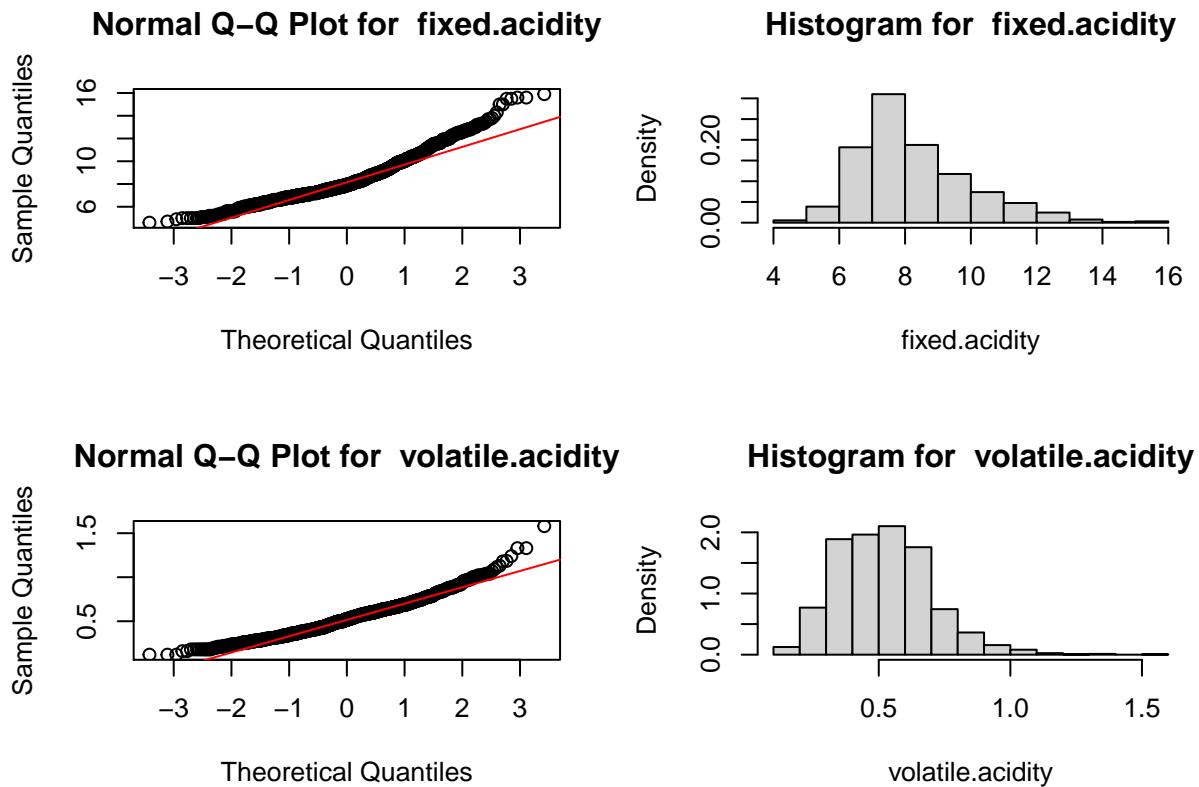
```
pairs(redWineData[, -grep("quality", colnames(redWineData))])
```

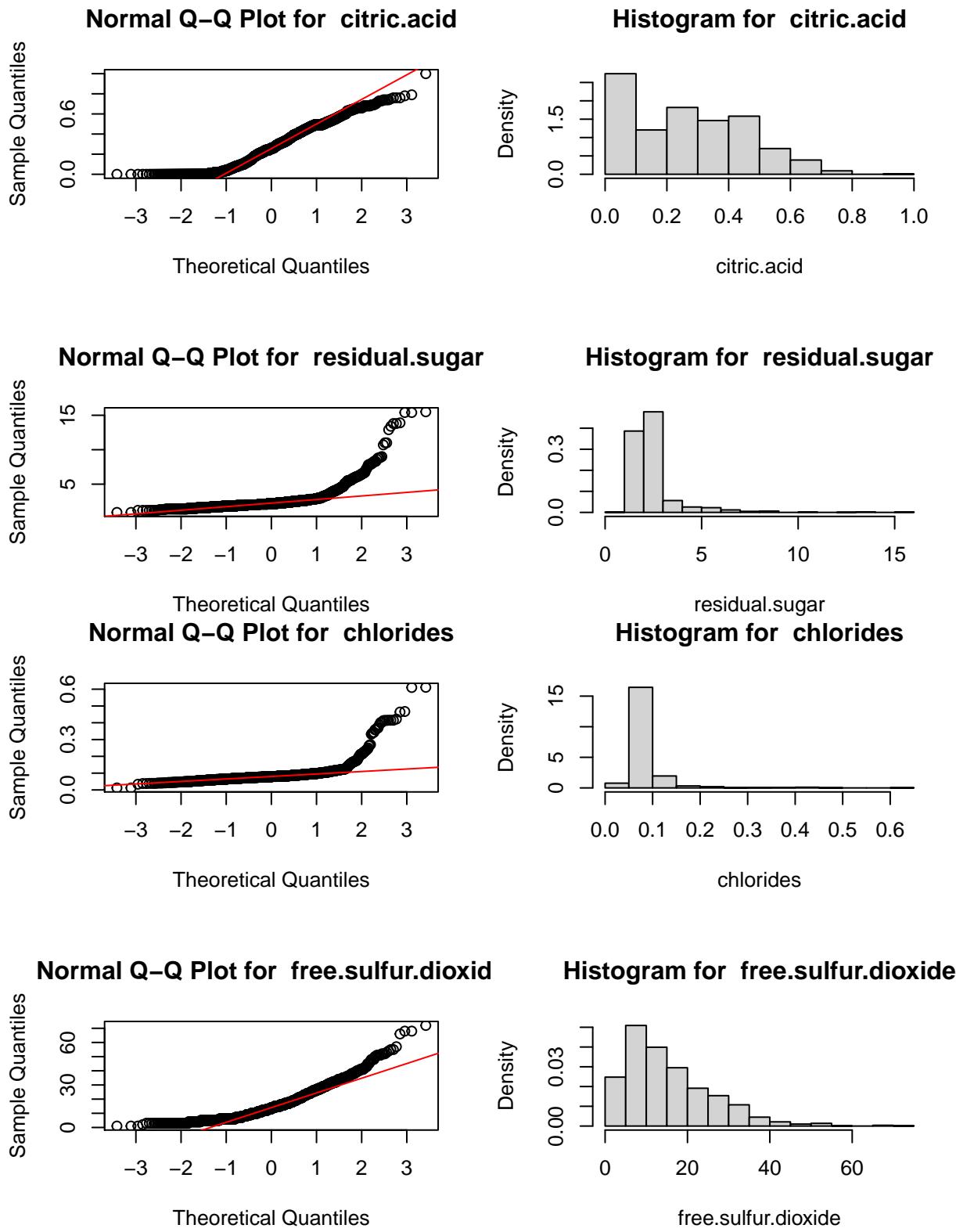


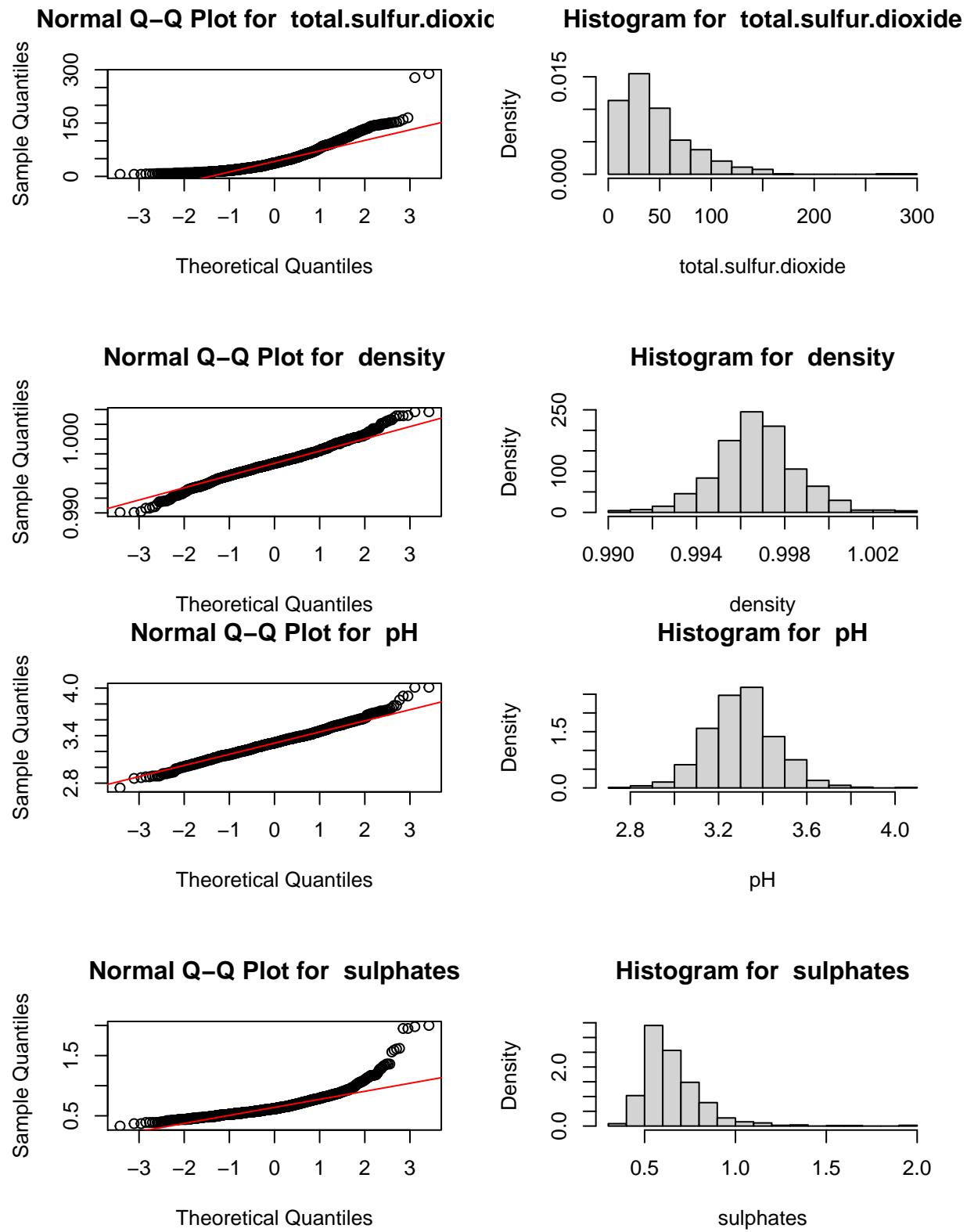
Podemos ver que todas las variables contienen valores atípicos. Estos valores atípicos se encuentran en los extremos superiores.

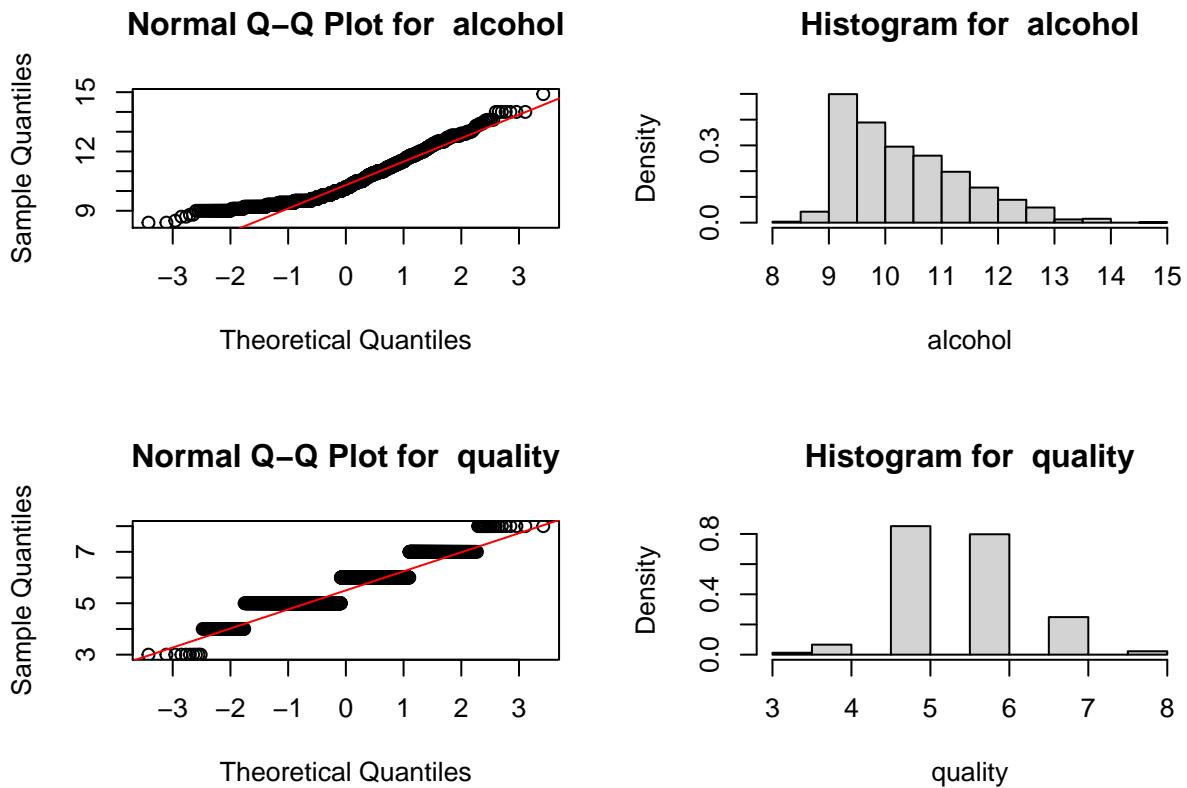
Procedemos a la generación de histogramas y de las gráficas quantile-quantile para entender la distribución de cada variable (predictor), sin eliminar los valores atípicos.

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









Observamos que casi todas las distribuciones están sesgadas positivamente. La calidad, el pH y la densidad se aproximan a una distribución normal.

A continuación, se eliminaran los valores extremos de las variables con el objetivo de ver si podemos mejorar la distribución de las variables.

```
eliminated_outliers <- redWineData
for (i in 1:11) {
  # Q <- quantile(redWineData[[i]], probs=c(.25, .75), na.rm = FALSE)
  # iqr <- IQR(redWineData[[i]])
  # up <- Q[2]+1.5*iqr # Upper Range
  # low <- Q[1]-1.5*iqr # Lower Range
  # eliminated_outliers <- subset(redWineData, redWineData[[i]] > (Q[1] - 1.5*iqr) & redWineData[[i]] <
  # ggbetweenstats(eliminated_outliers, quality, redWineData[[i]], outlier.tagging = TRUE)
  boxplot(redWineData[[i]], plot = FALSE)$out
  outliers <- boxplot(redWineData[[i]], plot = FALSE)$out
  eliminated_outliers <- eliminated_outliers[-which(eliminated_outliers[[i]] %in% outliers), ]
}

dim(eliminated_outliers)
```

```
## [1] 1194 12
```

Una vez eliminados los valores extremos, el tamaño del conjunto de datos se ha reducido a 1194 observaciones. Más adelante se volverá a comprobar la normalidad de las variables.

3.3. Exportación de los datos preprocesados

```
write.csv(eliminated_outliers, "winequality-red_data_clean.csv")
```

4. Análisis de los datos

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

Siendo nuestro objetivo principal definir cuales son los parámetros (variables) importantes para la definición de la calidad del vino, procederemos a utilizar todas las variables en nuestro conjunto de datos para los análisis que se efectuaran en esta práctica.

```
lm_modelo_1 <- lm(quality ~ ., data = eliminated_outliers)
summary(lm_modelo_1)

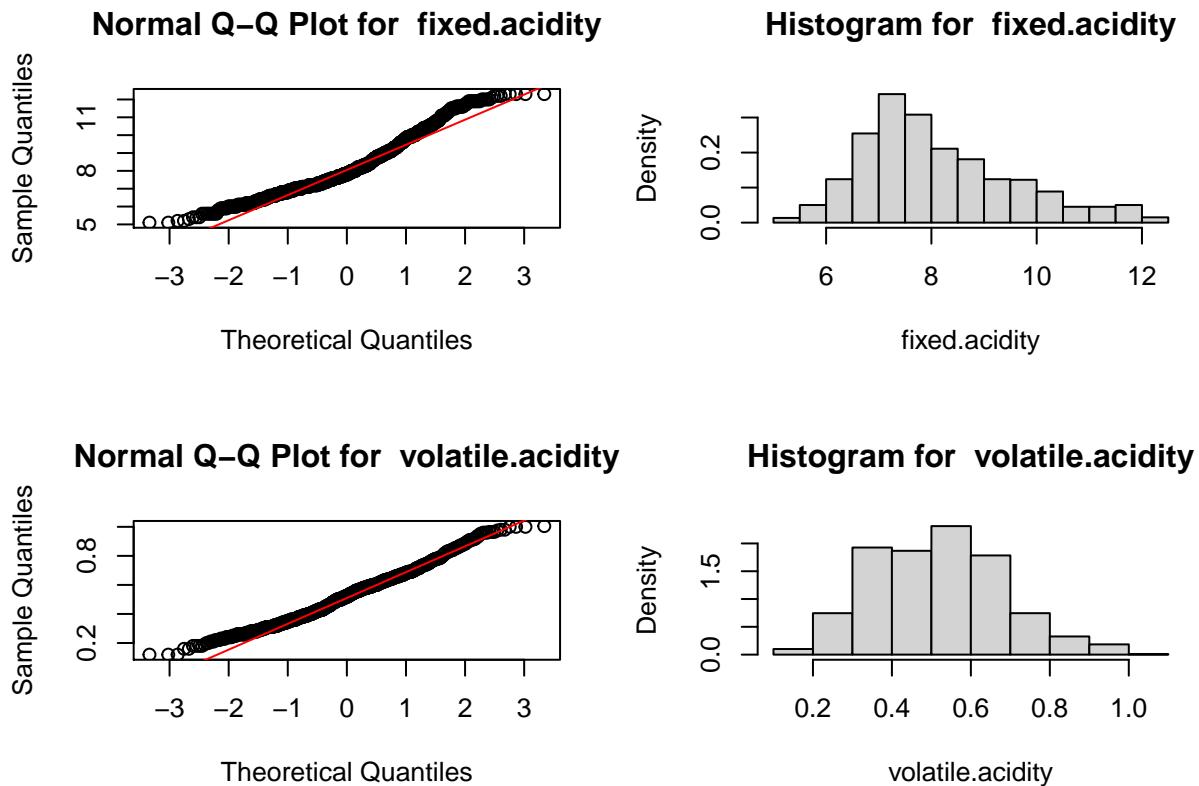
##
## Call:
## lm(formula = quality ~ ., data = eliminated_outliers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -2.26580 -0.36791 -0.04936  0.41792  1.89284 
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)    
## (Intercept) 2.684e+01 2.748e+01  0.977  0.32893  
## fixed.acidity 1.898e-02 3.100e-02  0.612  0.54061  
## volatile.acidity -8.504e-01 1.503e-01 -5.660  1.9e-08 *** 
## citric.acid -3.136e-01 1.708e-01 -1.836  0.06664 .  
## residual.sugar 1.238e-02 5.152e-02  0.240  0.81012  
## chlorides     -1.444e+00 1.383e+00 -1.044  0.29675  
## free.sulfur.dioxide 3.921e-03 2.653e-03  1.478  0.13972  
## total.sulfur.dioxide -2.639e-03 9.512e-04 -2.774  0.00563 ** 
## density        -2.272e+01 2.804e+01 -0.810  0.41799  
## pH            -6.461e-01 2.312e-01 -2.795  0.00528 ** 
## sulphates      1.735e+00 1.721e-01 10.079 < 2e-16 *** 
## alcohol        2.894e-01 3.401e-02  8.511  < 2e-16 *** 
## ---      
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6002 on 1182 degrees of freedom
## Multiple R-squared:  0.3925, Adjusted R-squared:  0.3869 
## F-statistic: 69.43 on 11 and 1182 DF,  p-value: < 2.2e-16
```

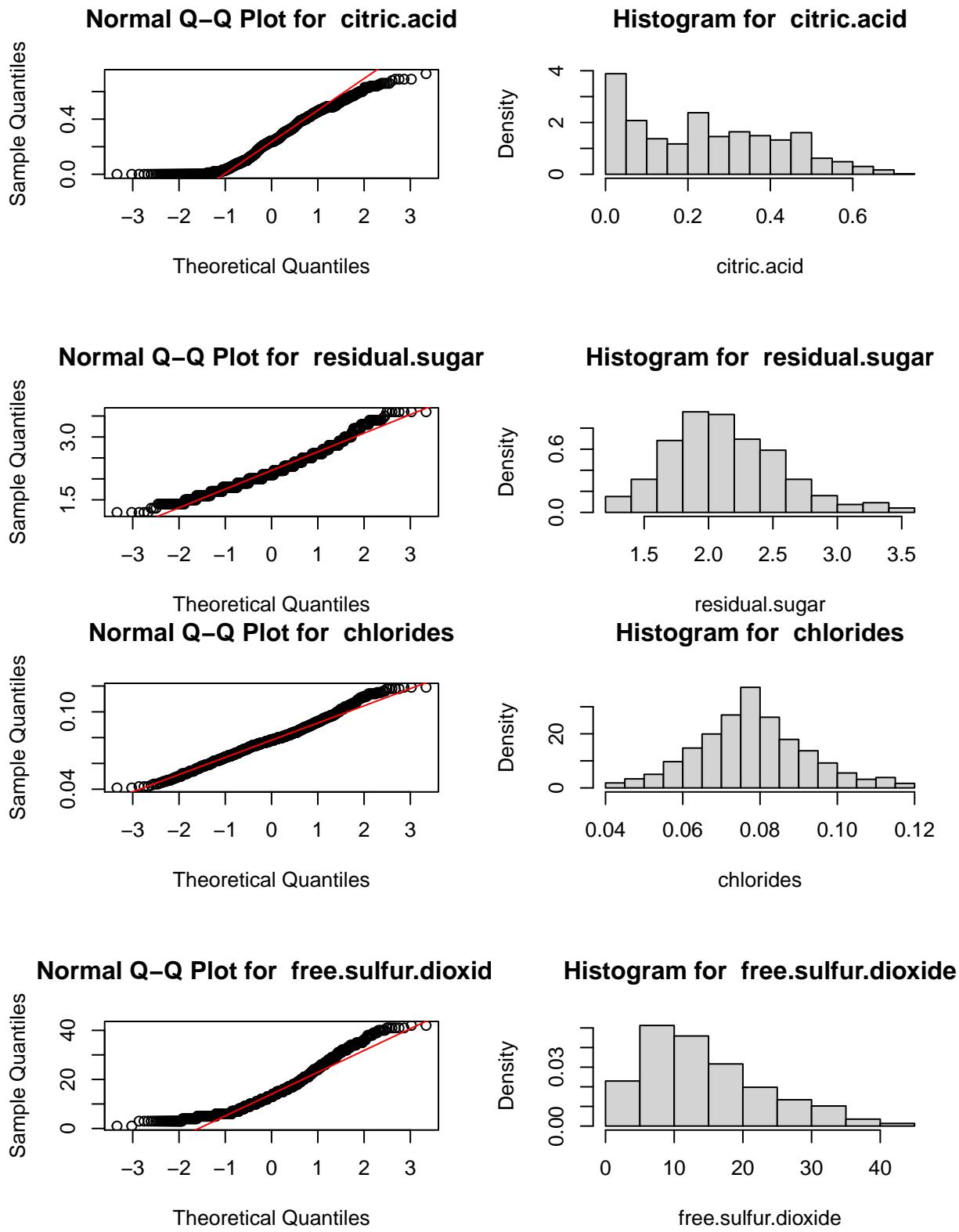
4.2. Comprobación de la normalidad y homogeneidad de la varianza.

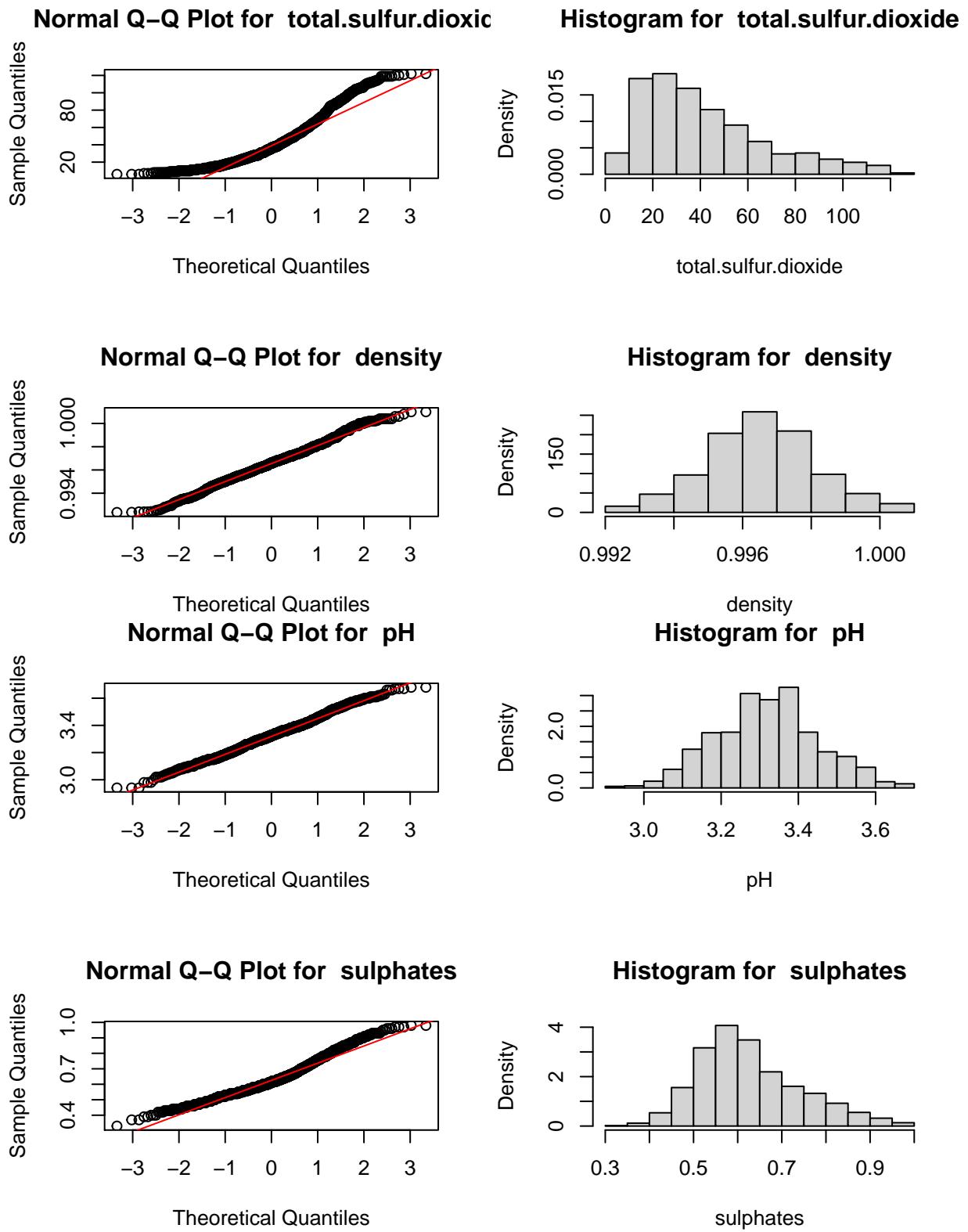
Procedemos a la generación de histogramas y de las gráficas quantile-quantile para entender la distribución de cada variable (predictor) utilizando el nuevo conjunto de datos con los valores atípicos eliminados.

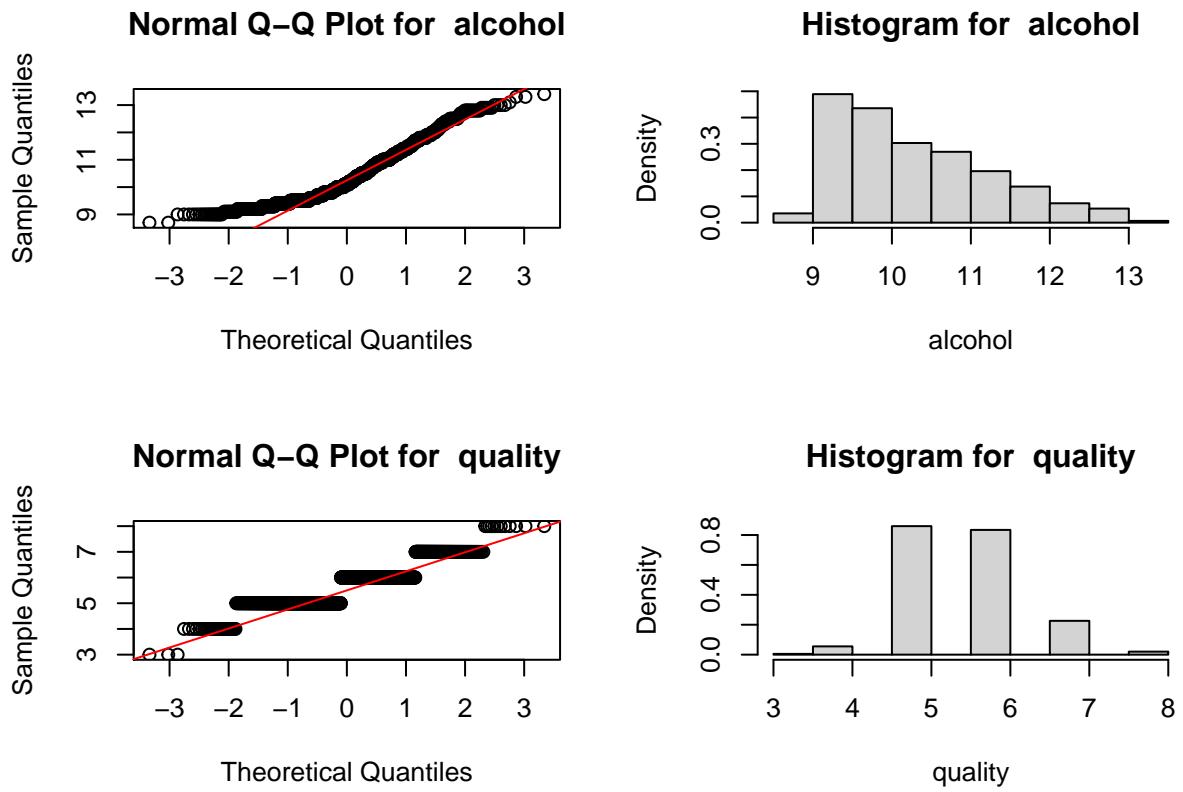
4.2.1. Normalidad

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









Una vez eliminados los valores extremos, podemos ver que las variables se aproximan a una distribucion normal.

```
par(mfrow=c(1,1))

shapiro.test(eliminated_outliers$fixed.acidity);

##  
##  Shapiro-Wilk normality test  
##  
## data:  eliminated_outliers$fixed.acidity  
## W = 0.95365, p-value < 2.2e-16

shapiro.test(eliminated_outliers$citric.acid);

##  
##  Shapiro-Wilk normality test  
##  
## data:  eliminated_outliers$citric.acid  
## W = 0.95138, p-value < 2.2e-16

shapiro.test(eliminated_outliers$sulphates);

##  
##  Shapiro-Wilk normality test  
##  
## data:  eliminated_outliers$sulphates  
## W = 0.9689, p-value = 2.517e-15

shapiro.test(eliminated_outliers$alcohol);

##  
##  Shapiro-Wilk normality test  
##  
## data:  eliminated_outliers$alcohol  
## W = 0.92964, p-value < 2.2e-16
```

4.2.2. Homogeneidad de la varianza

```
bptest(lm_modelo_1)

##
## studentized Breusch-Pagan test
##
## data: lm_modelo_1
## BP = 22.882, df = 11, p-value = 0.01837
```

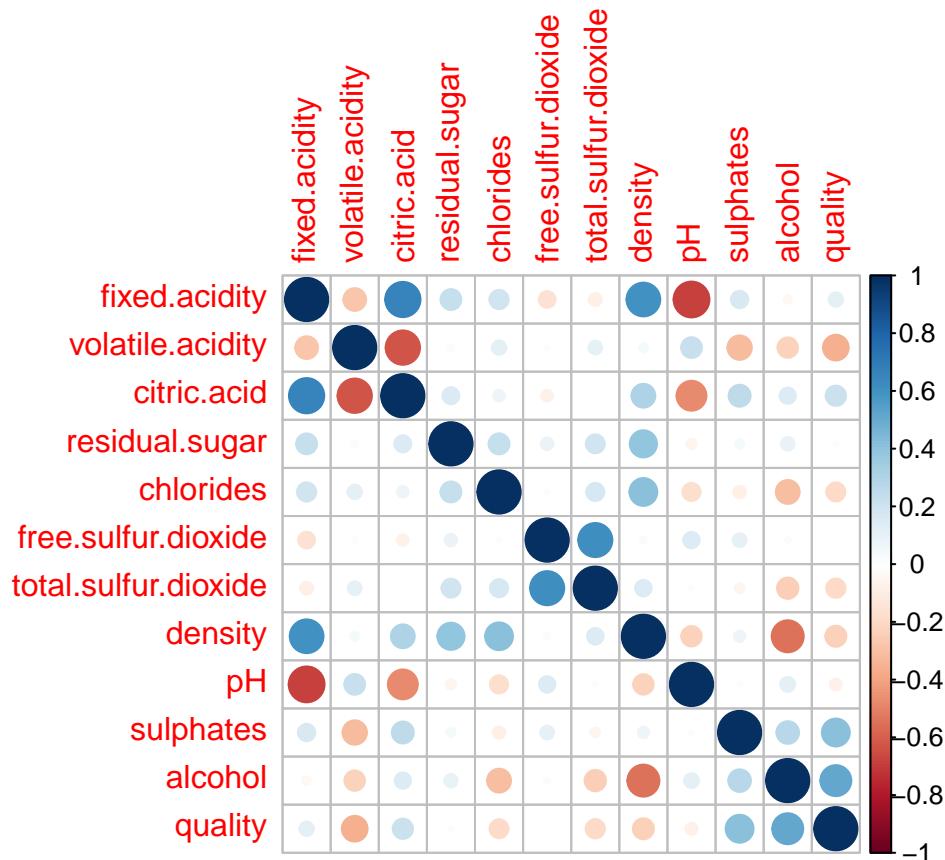
4.3. Aplicación de pruebas estadísticas para comparar los grupos de datos. En función de los datos y el objetivo del estudio, aplicar pruebas de contraste de hipótesis, correlaciones, regresiones, etc. Aplicar al menos tres métodos de análisis diferentes.

4.3.1. Correlaciones

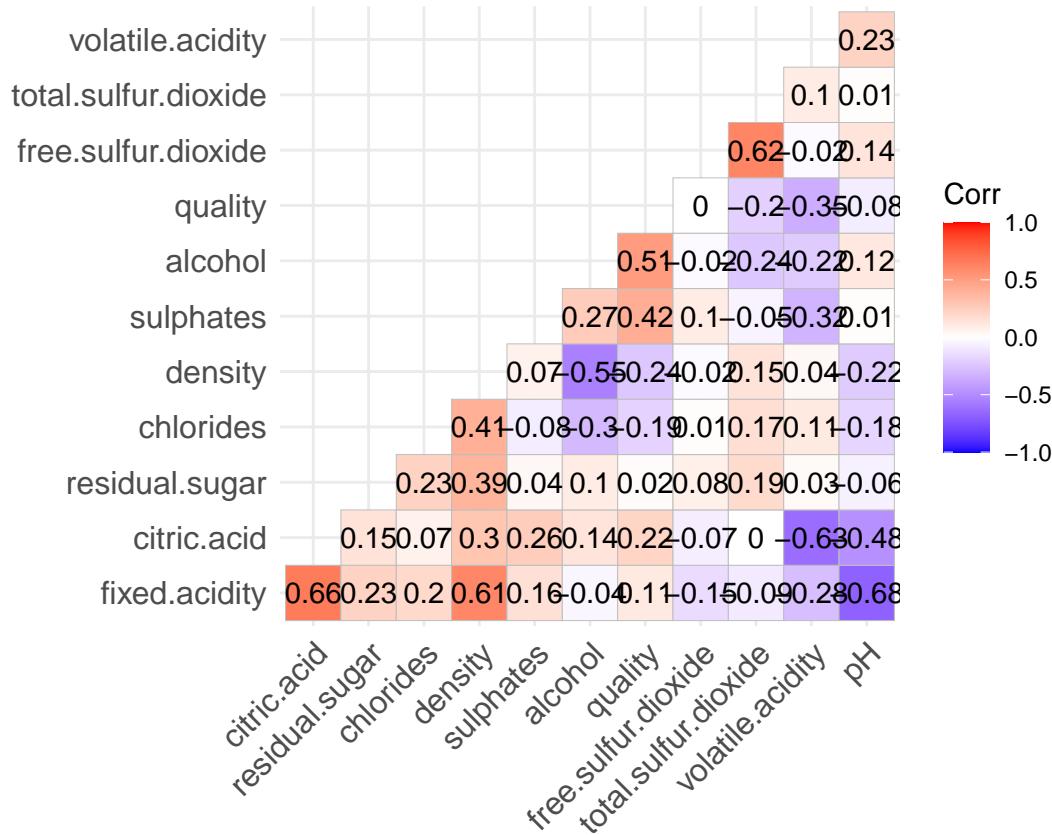
En primer lugar, procedemos a realizar un análisis de correlación entre las distintas variables para determinar cuáles de ellas ejercen una mayor influencia sobre la calidad del vino.

Para ello, se creará una matriz de correlación, puesto que una vez eliminados los valores atípicos, nuestros datos siguen una distribución normal.

```
corrplot(cor(eliminated_outliers), method = "circle")
```



```
ggcorrplot(cor(eliminated_outliers), hc.order = TRUE, type = "lower", lab = TRUE, insig = "blank")
```



Mediante la matriz de correlación, podemos ver que la calidad de los vinos está relacionada en gran medida con las variables sulphates y alcohol.

Tambien podemos observar las correlaciones entre las variables fixed.acidity - density (correlación fuerte), fixed.acidity - pH (correlación fuerte), fixed.acidity - citric.acid, volatlie.acidity - citric.acid y total.sulfur.dioxide - free.sulfur.dioxide.

```
cor(x=eliminated_outliers[1:11], y=eliminated_outliers$quality)
```

```
##                                     [,1]
## fixed.acidity          0.11067147
## volatile.acidity      -0.35221458
## citric.acid            0.21989725
## residual.sugar         0.01543850
## chlorides              -0.19227105
## free.sulfur.dioxide   -0.00278825
## total.sulfur.dioxide  -0.19853858
## density                -0.23633307
## pH                      -0.07572362
## sulphates              0.41535589
## alcohol                 0.51137367
```

Podemos ver que las variables están correlacionadas con quality de la siguiente manera

- alcohol (+++)
- sulphates (+++)
- volatlie.acidity (-)

- citric.acid (++)
- fixed.acidity (+)
- total.sulfur.dioxide (-)
- density (-)
- chlorides (-)

4.3.2. Modelo de regresión lineal

```

lm_modelo_2 <- lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide
+ total.sulfur.dioxide + pH + sulphates + alcohol, data = eliminated_outliers)

summary(lm_modelo_2)

##
## Call:
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol, data = eliminated_outliers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.25972 -0.34998 -0.06607  0.42237  1.86518
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 4.0407519  0.5199155  7.772 1.67e-14 ***
## volatile.acidity -0.6826454  0.1177287 -5.798 8.57e-09 ***
## chlorides     -2.0917450  1.3061996 -1.601 0.109555
## free.sulfur.dioxide  0.0049367  0.0026175  1.886 0.059539 .
## total.sulfur.dioxide -0.0030438  0.0009026 -3.372 0.000769 ***
## pH            -0.6080011  0.1417338 -4.290 1.93e-05 ***
## sulphates      1.6503392  0.1631210 10.117 < 2e-16 ***
## alcohol        0.3041744  0.0201237 15.115 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6007 on 1186 degrees of freedom
## Multiple R-squared:  0.3895, Adjusted R-squared:  0.3859
## F-statistic: 108.1 on 7 and 1186 DF,  p-value: < 2.2e-16

lm_modelo_3 <- lm(quality ~ volatile.acidity + chlorides + free.sulfur.dioxide
+ total.sulfur.dioxide + pH + sulphates + alcohol
+ total.sulfur.dioxide:sulphates + sulphates:alcohol, data = eliminated_outliers)
summary(lm_modelo_3)

##
## Call:
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol + total.sulfur.dioxide:sulphates +
##     sulphates:alcohol, data = eliminated_outliers)
##
## Residuals:

```

```

##      Min      1Q   Median      3Q      Max
## -2.28187 -0.33986 -0.08613  0.41967  1.86794
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               5.980188  1.142829  5.233 1.97e-07 ***
## volatile.acidity          -0.627604  0.116957 -5.366 9.67e-08 ***
## chlorides                 -1.482778  1.297880 -1.142 0.253493
## free.sulfur.dioxide       0.006361  0.002599  2.447 0.014530 *
## total.sulfur.dioxide      0.014871  0.004027  3.693 0.000232 ***
## pH                         -0.569235  0.140366 -4.055 5.33e-05 ***
## sulphates                -1.819280  1.659269 -1.096 0.273112
## alcohol                   0.027976  0.100856  0.277 0.781531
## total.sulfur.dioxide:sulphates -0.030205  0.006487 -4.656 3.58e-06 ***
## sulphates:alcohol         0.445818  0.153838  2.898 0.003825 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.5931 on 1184 degrees of freedom
## Multiple R-squared:  0.4057, Adjusted R-squared:  0.4012
## F-statistic: 89.82 on 9 and 1184 DF,  p-value: < 2.2e-16

```

```

lm_modelo_4 <- lm(quality ~ volatile.acidity + total.sulfur.dioxide + pH
+ total.sulfur.dioxide:sulphates + sulphates:alcohol, data = eliminated_outliers)
summary(lm_modelo_4)

```

```

##
## Call:
## lm(formula = quality ~ volatile.acidity + total.sulfur.dioxide +
##     pH + total.sulfur.dioxide:sulphates + sulphates:alcohol,
##     data = eliminated_outliers)
##
## Residuals:
##      Min      1Q   Median      3Q      Max
## -2.38597 -0.36467 -0.08739  0.43149  2.09432
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)               4.945907  0.455005 10.870 < 2e-16 ***
## volatile.acidity          -0.642233  0.117530 -5.464 5.65e-08 ***
## total.sulfur.dioxide      0.025963  0.003529  7.357 3.49e-13 ***
## pH                         -0.403954  0.137002 -2.949 0.00326 **
## total.sulfur.dioxide:sulphates -0.047160  0.005551 -8.495 < 2e-16 ***
## sulphates:alcohol         0.384010  0.020651 18.595 < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6017 on 1188 degrees of freedom
## Multiple R-squared:  0.3864, Adjusted R-squared:  0.3838
## F-statistic: 149.6 on 5 and 1188 DF,  p-value: < 2.2e-16

```

```

tabla_coeficientes <- matrix(c(1, summary(lm_modelo_1)$r.squared,
                                2, summary(lm_modelo_2)$r.squared,

```

```

            3, summary(lm_modelo_3)$r.squared,
            4, summary(lm_modelo_4)$r.squared),
            ncol = 2, byrow = TRUE)

colnames(tabla_coeficientes) <- c("Modelo", "R^2")
rownames(tabla_coeficientes) <- c("modelo 1", "model_linear_sig", "model_linear_inter", "modelo 4")

tabla_coeficientes <- as.table(tabla_coeficientes)

tabla_coeficientes

##                               Modelo      R^2
## modelo 1           1.0000000 0.3925216
## model_linear_sig  2.0000000 0.3894791
## model_linear_inter 3.0000000 0.4057255
## modelo 4           4.0000000 0.3864144

```

4.3.2. Modelo de regresión logística

```

eliminated_outliers_mutated <- mutate(eliminated_outliers, quality = ifelse(quality > 5, 1, 0))

model_logit <- glm(quality ~ ., family = binomial(link = 'logit'), data = eliminated_outliers_mutated)
summary(model_logit)

##
## Call:
## glm(formula = quality ~ ., family = binomial(link = "logit"),
##      data = eliminated_outliers_mutated)
##
## Deviance Residuals:
##    Min      1Q      Median      3Q      Max
## -2.8175 -0.8403   0.2863   0.8266   2.2948
##
## Coefficients:
##                               Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -33.265781 112.097340 -0.297 0.766651
## fixed.acidity          0.087689  0.127577  0.687 0.491870
## volatile.acidity       -2.856240  0.632485 -4.516 6.30e-06 ***
## citric.acid           -1.789308  0.699816 -2.557 0.010563 *
## residual.sugar        -0.107844  0.208261 -0.518 0.604576
## chlorides              -8.742472  5.615873 -1.557 0.119531
## free.sulfur.dioxide   0.023869  0.011050  2.160 0.030760 *
## total.sulfur.dioxide -0.013439  0.004058 -3.311 0.000928 ***
## density                27.732633 114.373623  0.242 0.808413
## pH                     -1.578549  0.937586 -1.684 0.092253 .
## sulphates              5.417175  0.754320  7.182 6.89e-13 ***
## alcohol                 0.974233  0.143460  6.791 1.11e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##

```

```

## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 1647.5 on 1193 degrees of freedom
## Residual deviance: 1225.8 on 1182 degrees of freedom
## AIC: 1249.8
##
## Number of Fisher Scoring iterations: 4

exp(cbind(coef(model_logit),confint(model_logit)))

## Waiting for profiling to be done...

##          2.5 %      97.5 %
## (Intercept) 3.571534e-15 6.042836e-111 7.022591e+80
## fixed.acidity 1.091648e+00 8.504004e-01 1.403017e+00
## volatile.acidity 5.748450e-02 1.635898e-02 1.956320e-01
## citric.acid 1.670757e-01 4.200137e-02 6.542240e-01
## residual.sugar 8.977677e-01 5.966206e-01 1.351191e+00
## chlorides 1.596587e-04 2.561779e-09 9.553742e+00
## free.sulfur.dioxide 1.024156e+00 1.002308e+00 1.046735e+00
## total.sulfur.dioxide 9.866505e-01 9.787295e-01 9.944467e-01
## density 1.106954e+12 6.315347e-86 5.557153e+109
## pH 2.062741e-01 3.264608e-02 1.293157e+00
## sulphates 2.252419e+02 5.269195e+01 1.017304e+03
## alcohol 2.649135e+00 2.006511e+00 3.523123e+00

confusionMatrix(table(predict(model_logit, type="response") >= 0.5,eliminated_outliers_mutated$quality))

## Confusion Matrix and Statistics
##
##          FALSE TRUE
## FALSE    402 152
## TRUE     147 493
##
##          Accuracy : 0.7496
##                 95% CI : (0.724, 0.7739)
## No Information Rate : 0.5402
## P-Value [Acc > NIR] : <2e-16
##
##          Kappa : 0.4962
##
## Mcnemar's Test P-Value : 0.8171
##
##          Sensitivity : 0.7322
##          Specificity : 0.7643
## Pos Pred Value : 0.7256
## Neg Pred Value : 0.7703
##          Prevalence : 0.4598
## Detection Rate : 0.3367
## Detection Prevalence : 0.4640
## Balanced Accuracy : 0.7483

```

```
##  
##      'Positive' Class : FALSE  
##
```

Con este modelo de regresión obtenemos una estimación de la precisión de 0.7496 (pero hemos clasificado el vino en bueno o malo).

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

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

Basandonos en el análisis del conjunto de datos, estamos convencidos que los parametros más importantes para definir la calidad del vino son: alcohol, sulphates, citric acid y fixed acidity. Siendo el factor más importante la concentración del alcohol. Mientras que los otros parámetros no están muy relacionados con la calidad del vino tinto y, por lo tanto, no serán adecuados para el análisis.

Tabla de contribuciones al trabajo

```
contribuciones <- matrix(c("Investigación previa","WAjibolaL,DFdezGlez","Redacción de las respuestas","WAjibolaL,DFdezGlez"),nrow=2,ncol=2)
colnames(contribuciones) <- c("Contribuciones","Firmas")
rownames(contribuciones) <- c("", "", "")
contribuciones <- as.table(contribuciones)
contribuciones

## Contribuciones           Firmas
## Investigación previa   WAjibolaL,DFdezGlez
## Redacción de las respuestas WAjibolaL,DFdezGlez
## Desarrollo código       WAjibolaL,DFdezGlez
```