PRA2

Francisco Fernández Poyato y Javier Gallego Fernández

21/5/2022

Índice

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

El objetivo principal de nuestro estudio va a ser llevar a cabo un análisis detallado de todas las características químicas de los vinos, así como ver cómo estas están relacionadas con la calidad del mismo.

Los análisis que se van a proponer sobre este dataset serán, en primer lugar un análisis descriptivo de las variables, en el que se representará graficamente cómo están distribuidas, así como un estudio de las correlaciones que pueda haber entre ellas.

También plantearemos el estudio de la normalidad y homocedasticidad de nuestras variables, para tener claro que técnicas de análisis de datos podremos utilizar.

Se planteará una regresión lineal para estudiar cuál es la relación entre todos nuestros parámetros y la variable objeto de estudio, que para nosotros será la calidad del vino. Además, realizaremos contrastes de hipótesis dividiendo nuestra población para poder sacar conclusiones de la relación de los parámetros químicos con la calidad del vino

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

En nuestro caso, hemos seleccionado un dataset de Kaggle en el que están recogidos datos relacionados con las características químicas de los vinos tintos.

En este caso no va a ser necesario unir datasets ni seleccionar conjuntos de entrenamiento ni de test, ya que eso lo haremos directamente con nuestro conjunto de datos que vamos a utilizar.

```
datosvino =read.csv('winequality-red.csv', sep = ',')
```

```
sapply(datosvino, function(x) sum(is.na(x)))
```

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

```
shapiro.test(datosvino$fixed.acidity)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosvino$fixed.acidity
## W = 0.94203, p-value < 2.2e-16
```

```
summary(datosvino)
```

```
##  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
```

```
str(datosvino)
```

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

3. Limpieza de datos

Etapa importante dentro de un análisis de datos calidad datos calidad modelo...

3.1 ¿Los datos contiene ceros o elementos vacíos? Gestiona cada uno de estos casos

En primer lugar estudiaremos si en nuestro conjunto de datos existen elementos vacíos. Para ello vamos a estudiar si existe algún registro de alguna variable que esté sin información

Para ello utilizaremos la siguiente función

```
sapply(datosvino, function(x) sum(is.na(x)))
```

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

Como se puede observar, no tendremos ningún registro en ninguna variable que tenga valores nulos, por lo que no tendríamos que descartar ningún registro ni tendríamos que aplicar técnicas de imputación de valores vacíos

A continuación, pasaremos a estudiar la existencia de posibles valores registrados como 0 y veremos si tienen sentido dentro de la variable o se ha guardado así para representar que no se tiene información de ese registro.

Observando las variables y todos los rangos que tiene esta, podemos ver que solamente la variable citric.acid tiene al 0 dentro del rango de posibles valores. Sin embargo, haciendo un estudio de los posibles valores que puede tener esta variable, concluimos que es un valor posible para un vino en esta variable, por lo que no descartaremos aquellos registros que tengan 0 como valor en citric.acid

3.2 Registros duplicados

Otro de los análisis previos que se deben realizar con anterioridad a la implementación de técnicas de análisis de datos es el estudio de los posibles valores duplicados que puede haber en el conjunto de datos. En este conjunto de datos no tenemos una variable que actúe como identificador, por lo que no tendremos una clave primaria predefinida, sin embargo, al ser 12 variables numéricas, entendemos que es prácticamente imposible que un vino tenga exactamente los mismos valores para estos 12 campos, por lo que si tenemos algún caso que comparta los mismos valores para todas las variables, lo identificaremos como duplicado

Para ello, en primer lugar vamos a estudiar la existencia de estos posibles casos y también cuántos registros se verán afectados

```
sum(duplicated(datosvino))
```

```
## [1] 240
```

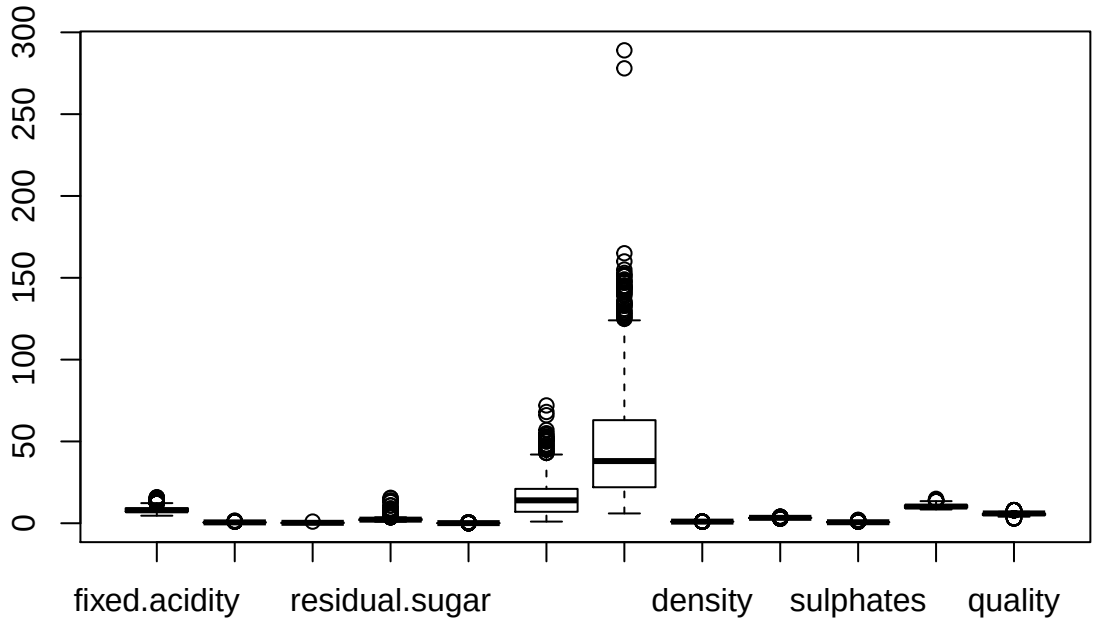
Como podemos observar, tendremos 240 registros duplicados. A continuación, pasaremos a seleccionar solamente aquellos registros que no se repiten en el conjunto de datos.

```
datosvino = unique(datosvino)
```

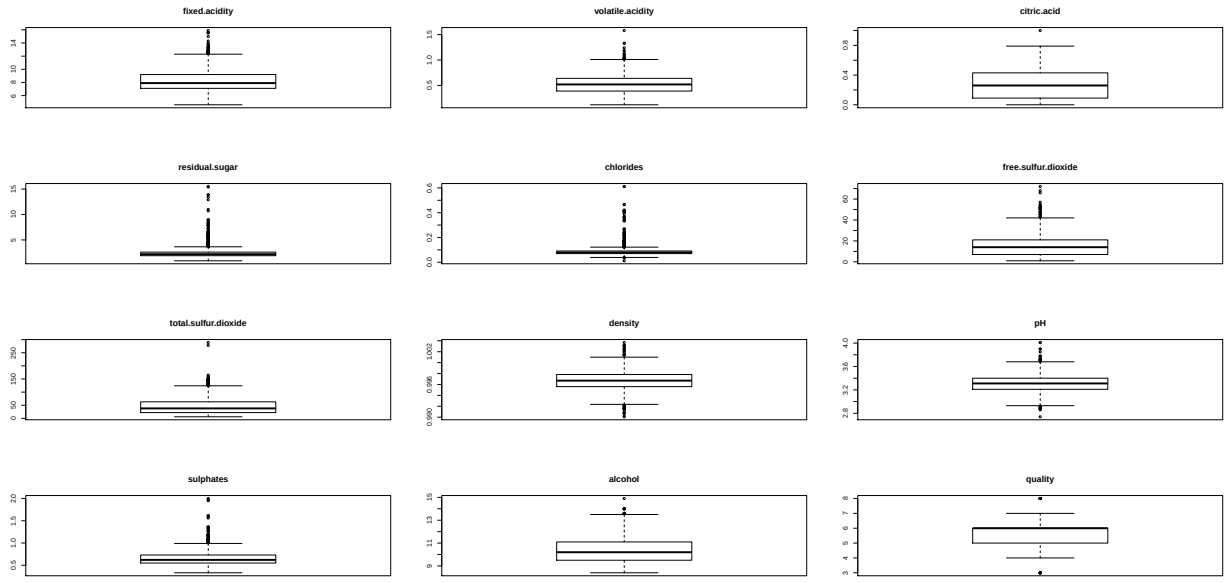
3.3 Identifica y gestiona los valores extremos

Para poder identificar cuáles son los valores extremos de las variables y ver si tiene sentido dentro de nuestro conjunto de datos, pasaremos a realizar un diagrama de caja y bigotes para ver que registros nos generan valores **outliers**.

```
boxplot(datosvino)
```




```
par(mfrow=c(4,3))
for(i in 1:ncol(datosvino)) {
boxplot(datosvino[,i], main = colnames(datosvino)[i])
}
```



Haciendo un estudio de los posibles valores que tienen todas estas variables en los vinos, concluimos que no

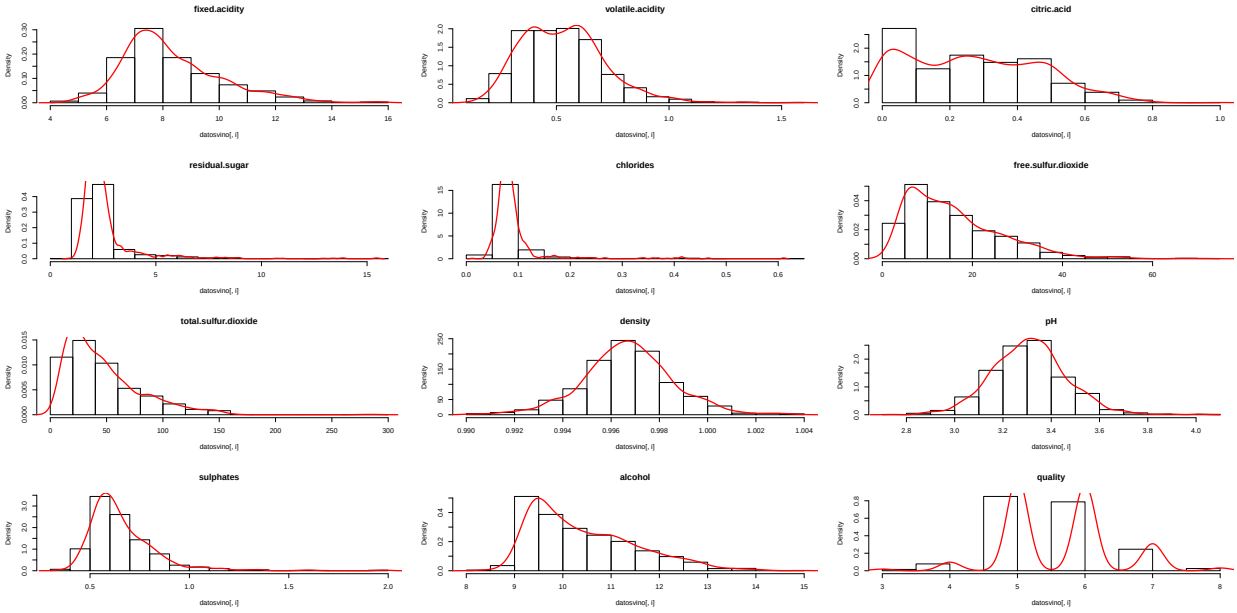
hay ningún valor que podamos descartar, aunque este sea muy elevado y esté lejano a la gran mayoría de los valores de las variables, ya que no tenemos evidencia de que un vino no pueda tener esos valores. Por poner un ejemplo, el caso que más nos ha llamado la atención es el del total.sulfur.dioxide, ya que tiene valores cercanos a los 250 mg/L, muy lejanos del grueso de la distribución de la variable, sin embargo, hemos encontrado la siguiente información relativa a esta variable, y es que los valores pueden llegar a ser de 300 mg / L o incluso 400 mg / L en el caso de determinadas denominaciones geográficas de vinos dulces.

4. Análisis de los datos.

4.1 Análisis descriptivo del conjunto de datos

En primer lugar, para llevar a cabo nuestro análisis descriptivo visualizaremos un histograma de todas las variables junto con la gráfica de su distribución, para poder ver como se comporta la variable

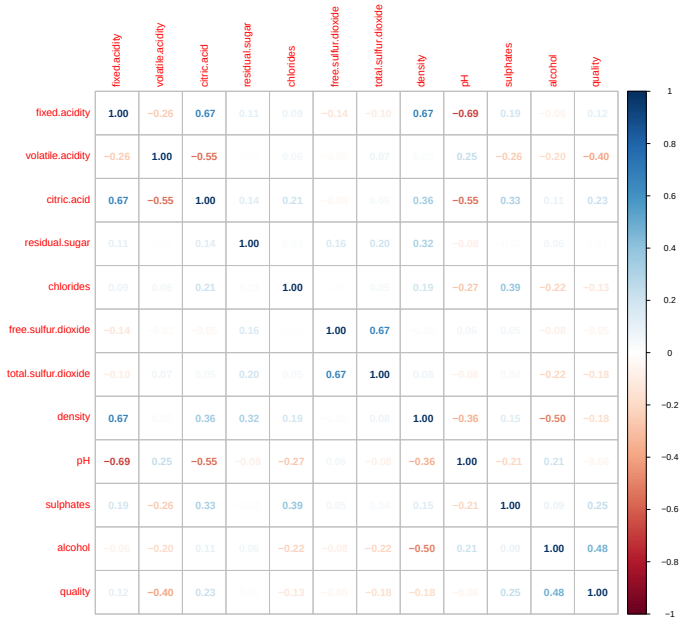
```
par(mfrow=c(4,3))
for(i in 1:ncol(datosvino)) {
hist(datosvino[,i], main = colnames(datosvino)[i], probability = TRUE)
lines(density(datosvino[,i]),
 lwd = 2, # thickness of line
 col = "red")
}
```



Observando estas gráficas, podríamos decir que las variables densidad y pH tendrán una distribución similar a una normal, aunque nos aseguraremos posteriormente mediante los test de normalidad.

Una vez vistas las distribuciones de las variables, pasaremos a estudiar las correlaciones que tendrán nuestras variables entre ellas, para así poder ver posibles relaciones y comportamientos similares, poniendo el foco en la variable objeto de estudio que será la calidad del vino.

```
matrixcor = cor(datosvino)
corrplot::corrplot(matrixcor, method = "number")
```

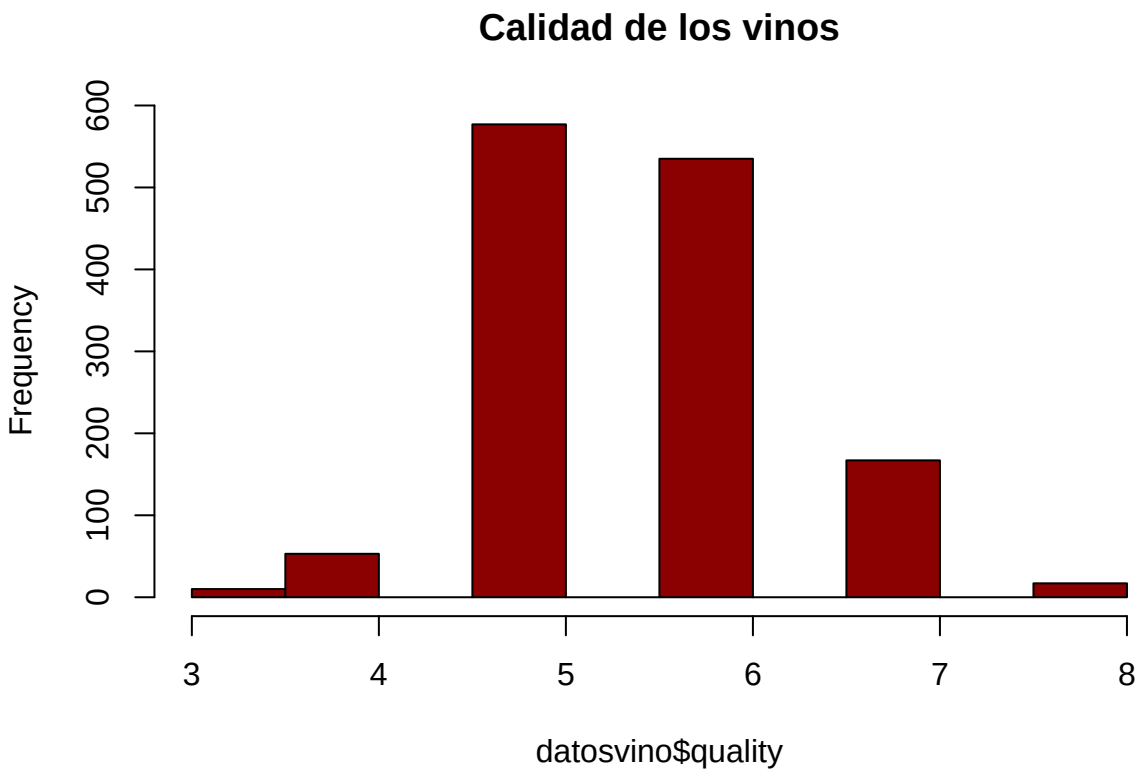


Observando las correlaciones obtenidas entre las variables, destacamos que la variable fixed.acidity estará correlacionada con las variables ph (correlación negativa), teniendo esto sentido ya que un menor pH estará relacionado con una mayor acidez, y densidad (correlación positiva). También cabe destacar que estarán relacionados los valores de sulfuro libre y sulfuro total, observando que es algo que tiene sentido debido a que los sulfuros libres están incluidos en el sulfuro total.

Con respecto a la variable objeto de estudio, observamos que la variable alcohol será la que esté más relacionada con la calidad del vino, aunque no es muy elevada (0.48)

También cabe destacar que en la varible calidad, nuestra muestra no va a estar balanceada, concentrándose la gran mayoría de los valores entre el rango de 5 y 6

```
hist(datosvino$quality, main = 'Calidad de los vinos', col = "red4")
```



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

En nuestro caso hemos considerado interesante crear tres variables categóricas para dividir nuestra población y así poder sacar conclusiones en función de las características que tengan los vinos.

En primer lugar vamos a dividir la muestra en función de la graduación alcohólica que tengan los vinos, creando una variable que sea Alta graduación alcohólica para aquellos registros que estén por encima de la mediana y Baja graduación alcohólica para aquellos registros que estén por debajo de la mediana. Así, tendremos la población dividida en dos y podremos aplicar contrastes de hipótesis para ver si esto afecta o no a la calidad del vino. También haremos lo mismo con la variable pH, donde seguiremos el mismo criterio para dividir la población.

Por último, también crearemos una variable categórica a partir de la calidad de los vinos, siendo baja cuando la puntuación de este sea 3 o 4, media cuando sea 5 o 6 y alta cuando la puntuación llegue a 7 u 8.

Pasamos ahora a crear estas variables

```
datosvinocats<-datosvino
datosvinocats$pHcat <- cut(datosvino$pH,
                      breaks=c(min(datosvino$pH)-1,median(datosvino$pH), max(datosvino$pH)+1),
                      labels=c('pH bajo', 'pH alto'))

datosvinocats$alcoholcat <- cut(datosvino$alcohol,
                      breaks=c(min(datosvino$alcohol)-1,median(datosvino$alcohol), max(datosvino$alcoho
                      labels=c('Graduación alcohólica baja', 'Graduación alcohólica alta'))


datosvinocats$calidadcat <- cut(datosvino$quality,
```

```
                    breaks=c(2,4,6,9),
                    labels=c('Calidad baja','Calidad media','Calidad alta'))
```

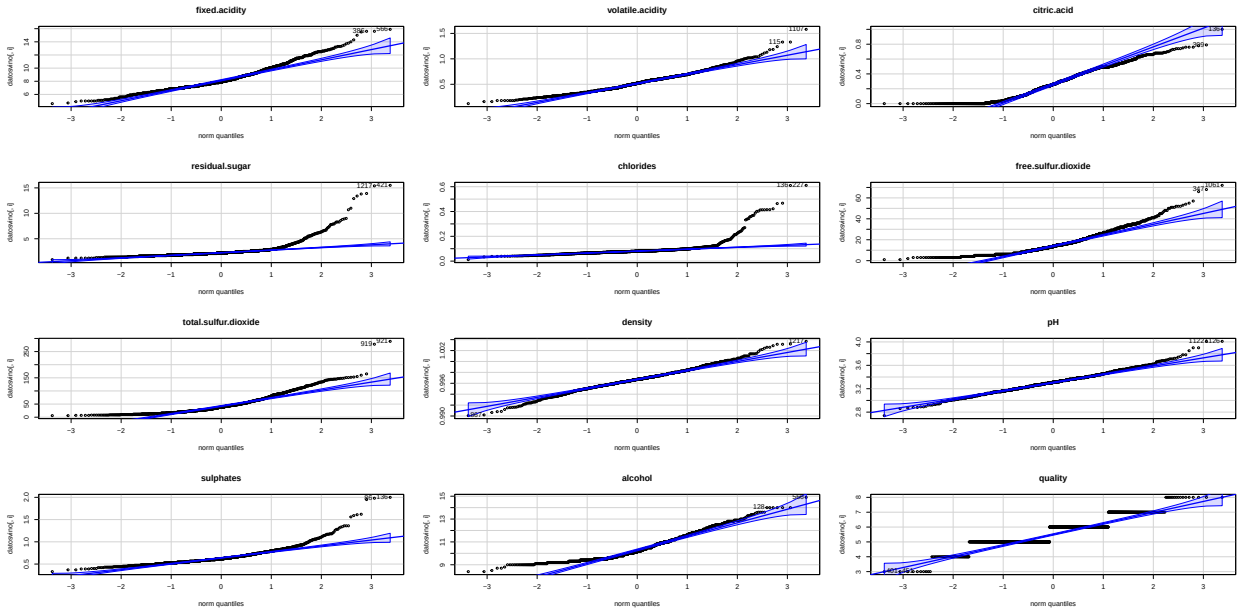
4.3 Comprobación de la normalidad y homogeneidad de la varianza.

```
library("car")
```

```
## Loading required package: carData
```

```
par(mfrow=c(4,3))
for(i in 1:ncol(datosvino)) {
if (is.numeric(datosvino[,i])){

qqPlot(datosvino[,i], main = colnames(datosvino)[i])}
}
```



Observando los qqPlot de nuestas variables, podemos deducir que las variables density y pH se pueden comportar como una distribución normal. Aún así, de todas formas a continuación aplicaremos el test de Shapiro-Wilk para comprobar si efectivamente se pueden asumir como distribuciones normales

```
for(i in 1:ncol(datosvino)) {
if (is.numeric(datosvino[,i])){
a = shapiro.test(datosvino[,i])
print(colnames(datosvino)[i])
print(a)
}}
```

```
## [1] "fixed.acidity"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.94684, p-value < 2.2e-16
##
## [1] "volatile.acidity"
```

```
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.97018, p-value = 3.931e-16
##
## [1] "citric.acid"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.95552, p-value < 2.2e-16
##
## [1] "residual.sugar"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.57673, p-value < 2.2e-16
##
## [1] "chlorides"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.48448, p-value < 2.2e-16
##
## [1] "free.sulfur.dioxide"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.90323, p-value < 2.2e-16
##
## [1] "total.sulfur.dioxide"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.87169, p-value < 2.2e-16
##
## [1] "density"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.99239, p-value = 1.804e-06
##
## [1] "pH"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.99272, p-value = 3.082e-06
```

```
##
## [1] "sulphates"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.83024, p-value < 2.2e-16
##
## [1] "alcohol"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.9268, p-value < 2.2e-16
##
## [1] "quality"
##
##  Shapiro-Wilk normality test
##
## data:  datosvino[, i]
## W = 0.86398, p-value < 2.2e-16
```

Sin embargo, realizando el test de Shapiro-Wilk a todas nuestras variables, no tendremos ninguna que podamos asumir como variable normal, aunque aquellas variables que introduzcamos posteriormente en el modelo se podrán asumir como tal por el Teorema Central del Límite.

Una vez hemos hecho el estudio de la homocedasticidad, lo haremos a partir de dividir la muestra en función de las variables categóricas que hemos creado.

```
datosalcoholbajo<-datosvino[datosvino$alcoholcat=='Graduación alcohólica baja',]
datosalcoholalto<-datosvino[datosvino$alcoholcat=='Graduación alcohólica alta',]
datosphbajo<-datosvino[datosvino$pHcat=='pH bajo',]
datosphalto<-datosvino[datosvino$pHcat=='pH alto',]
```

Aunque no hayamos obtenido normalidad, pasaremos a calcular la homocedasticidad de la variable calidad

```
library(stats)
```

(Se dejan comentadas las funciones porque al pasar a pdF genera error, sin embargo en R si que compila correctamente la función, de la que se adjunta la salida como texto)

```
#var.test(datosalcoholalto$quality, datosalcoholbajo$quality)
```

```
F test to compare two variances
```

data: datosalcoholalto*qualityanddatosalcoholbajo*quality F = 1.6884, num df = 640, denom df = 717, p-value = 1.011e-11 alternative hypothesis: true ratio of variances is not equal to 1 95 percent confidence interval: 1.452482 1.964017 sample estimates: ratio of variances 1.688441

```
# var.test(datosalcoholalto$pH, datosalcoholbajo$pH)
```

F test to compare two variances

data: datosalcoholalto*pHanddatosalcoholbajo*pH F = 1.2013, num df = 640, denom df = 717, p-value = 0.01695 alternative hypothesis: true ratio of variances is not equal to 1 95 percent confidence interval: 1.033406 1.397350 sample estimates: ratio of variances 1.201285

Como podemos observar, en ambos casos obtenemos que no se puede asumir igualdad de varianzas.

4.4 Aplicación de pruebas estadísticas para comparar los grupos de datos.

4.4.1 Regresión lineal múltiple En primer lugar, comenzaremos realizando una regresión lineal múltiple para intentar predecir la calidad del vino en función de las variables con los parámetros químicos. Para ello implementaremos la función lm, con la cuál pondremos la variable calidad en función de las demás.

```
attach(datosvino)
reg1<-lm(quality~., data = datosvino)
```

```
summary(reg1)
```

```
##
## Call:
## lm(formula = quality ~ ., data = datosvino)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.64913 -0.36259 -0.03834  0.46208  1.99662
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          13.2379410 23.5218706   0.563 0.573670
## fixed.acidity         0.0125661  0.0289274   0.434 0.664067
## volatile.acidity     -1.1204370  0.1303899  -8.593  < 2e-16 ***
## citric.acid          -0.1642423  0.1618053  -1.015 0.310259
## residual.sugar        0.0071080  0.0169600   0.419 0.675207
## chlorides            -1.9302567  0.4484864  -4.304 1.80e-05 ***
## free.sulfur.dioxide   0.0033430  0.0023930   1.397 0.162645
## total.sulfur.dioxide -0.0027073  0.0007976  -3.394 0.000709 ***
## density              -8.9904296 24.0017287  -0.375 0.708036
## pH                   -0.4584869  0.2127934  -2.155 0.031369 *
## sulphates             0.9147023  0.1270138   7.202 9.87e-13 ***
## alcohol               0.2895307  0.0293153   9.876  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6596 on 1347 degrees of freedom
## Multiple R-squared:  0.3638, Adjusted R-squared:  0.3586
## F-statistic: 70.02 on 11 and 1347 DF,  p-value: < 2.2e-16
```

Realizando la regresión lineal múltiple, obtenemos un estadístico R^2 de 0.358, con lo que podemos concluir que nuestro modelo no será muy bueno a la hora de predecir la calidad del vino.

Para intentar mejorar esta regresión, utilizaremos la función step, cuya utilidad consiste en encontrar las variables que generan el modelo de regresión óptimo.

```
 regopt<-step(reg1, direction = 'both')
```

```
## Start:  AIC=-1119.15
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + sulphates + alcohol
##
##                       Df Sum of Sq    RSS     AIC
## - density              1     0.061 586.07 -1121.0
## - residual.sugar       1     0.076 586.08 -1121.0
## - fixed.acidity        1     0.082 586.09 -1121.0
```

```
## - citric.acid           1     0.448 586.46 -1120.1
## - free.sulfur.dioxide   1     0.849 586.86 -1119.2
## <none>                              586.01 -1119.2
## - pH                    1     2.020 588.03 -1116.5
## - total.sulfur.dioxide  1     5.012 591.02 -1109.6
## - chlorides             1     8.059 594.07 -1102.6
## - sulphates             1    22.563 608.57 -1069.8
## - volatile.acidity      1    32.123 618.13 -1048.6
## - alcohol               1    42.436 628.44 -1026.1
##
## Step:  AIC=-1121.01
## quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     pH + sulphates + alcohol
##
##                         Df Sum of Sq    RSS      AIC
## - fixed.acidity          1     0.022 586.09 -1122.96
## - residual.sugar         1     0.027 586.10 -1122.95
## - citric.acid            1     0.444 586.51 -1121.98
## <none>                              586.07 -1121.01
## - free.sulfur.dioxide    1     0.891 586.96 -1120.95
## + density                1     0.061 586.01 -1119.15
## - pH                     1     3.577 589.65 -1114.74
## - total.sulfur.dioxide   1     5.125 591.19 -1111.18
## - chlorides              1     8.300 594.37 -1103.90
## - sulphates              1    23.395 609.46 -1069.82
## - volatile.acidity       1    32.834 618.90 -1048.93
## - alcohol                1   106.741 692.81  -895.63
##
## Step:  AIC=-1122.96
## quality ~ volatile.acidity + citric.acid + residual.sugar + chlorides +
##     free.sulfur.dioxide + total.sulfur.dioxide + pH + sulphates +
##     alcohol
##
##                         Df Sum of Sq    RSS      AIC
## - residual.sugar         1     0.030 586.12 -1124.89
## - citric.acid            1     0.486 586.58 -1123.83
## <none>                              586.09 -1122.96
## - free.sulfur.dioxide    1     0.918 587.01 -1122.83
## + fixed.acidity          1     0.022 586.07 -1121.01
## + density                1     0.001 586.09 -1120.96
## - pH                     1     5.478 591.57 -1112.32
## - total.sulfur.dioxide   1     5.777 591.87 -1111.63
## - chlorides              1     9.248 595.34 -1103.68
## - sulphates              1    23.614 609.71 -1071.28
## - volatile.acidity       1    34.649 620.74 -1046.90
## - alcohol                1   107.901 693.99  -895.31
##
## Step:  AIC=-1124.89
## quality ~ volatile.acidity + citric.acid + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol
##
##                         Df Sum of Sq    RSS      AIC
## - citric.acid            1     0.463 586.58 -1125.82
```

```
## <none>                              586.12 -1124.89
## - free.sulfur.dioxide   1     0.944 587.07 -1124.70
## + residual.sugar        1     0.030 586.09 -1122.96
## + fixed.acidity         1     0.026 586.10 -1122.95
## + density               1     0.002 586.12 -1122.89
## - pH                    1     5.507 591.63 -1114.18
## - total.sulfur.dioxide  1     5.756 591.88 -1113.61
## - chlorides             1     9.228 595.35 -1105.66
## - sulphates             1    23.625 609.75 -1073.19
## - volatile.acidity      1    34.714 620.84 -1048.69
## - alcohol               1   109.784 695.91  -893.57
##
## Step:  AIC=-1125.82
## quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol
##
##                        Df Sum of Sq    RSS      AIC
## <none>                              586.58 -1125.82
## - free.sulfur.dioxide   1     1.157 587.74 -1125.14
## + citric.acid           1     0.463 586.12 -1124.89
## + density               1     0.073 586.51 -1123.98
## + fixed.acidity         1     0.058 586.53 -1123.95
## + residual.sugar        1     0.007 586.58 -1123.83
## - pH                    1     5.279 591.86 -1115.64
## - total.sulfur.dioxide  1     6.475 593.06 -1112.90
## - chlorides             1    10.103 596.69 -1104.61
## - sulphates             1    23.230 609.81 -1075.03
## - volatile.acidity      1    40.487 627.07 -1037.11
## - alcohol               1   111.409 697.99  -891.49
```

```
summary(regopt)
```

```
##
## Call:
## lm(formula = quality ~ volatile.acidity + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + pH + sulphates + alcohol, data = datosvino)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.65850 -0.36113 -0.04136  0.47413  2.00412
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.2825144  0.4437336   9.651  < 2e-16 ***
## volatile.acidity     -1.0530341  0.1090493  -9.656  < 2e-16 ***
## chlorides            -2.0404731  0.4230025  -4.824 1.57e-06 ***
## free.sulfur.dioxide   0.0038336  0.0023483   1.632 0.102808
## total.sulfur.dioxide -0.0028856  0.0007472  -3.862 0.000118 ***
## pH                   -0.4540960  0.1302355  -3.487 0.000505 ***
## sulphates             0.8916069  0.1218939   7.315 4.41e-13 ***
## alcohol               0.2940819  0.0183589  16.019  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6589 on 1351 degrees of freedom
```

```
## Multiple R-squared:  0.3632, Adjusted R-squared:  0.3599
## F-statistic: 110.1 on 7 and 1351 DF,  p-value: < 2.2e-16
```

Observando el modelo generado, tenemos que se quedará con las variables chlorides, volatile.acidity, free.sulfur.dioxide, total.sulfur.dioxide, pH, sulphates y alcohol.

Sin embargo, no observamos una gran mejoría, por lo que tampoco será este un modelo que nos ayude a predecir la calidad del vino.

4.4.2 Contrastes de hipótesis A continuación, vamos a realizar un estudio para comprobar si la calidad está relacionada con la graduación alcohólica de los vinos y con su pH, y ver si podemos inferir que un vino tiene una mayor o peor calidad en función de estos parámetros.

En primer lugar, plantearemos un contraste de hipótesis de comparación de medias de la calidad de dos poblaciones, donde en primer lugar tendremos los vinos de baja graduación alcohólica y por otro lado tendremos aquellos vinos que tendrán una graduación más elevada.

$$\begin{cases} H_0 : \mu_{bajo} = \mu_{alto} \\ H_1 : \mu_{bajo} \neq \mu_{alto} \end{cases}$$

Una vez hemos planteado el contraste de hipótesis, pasamos a implementar el test de Student's

```
#t.test(datosalcoholbajo$quality, datosalcoholalto$quality)
```

Welch Two Sample t-test

data: datosalcoholbajo*qualityanddatosalcoholalto*quality t = -16.107, df = 1197, p-value < 2.2e-16 alternative hypothesis: true difference in means is not equal to 0 95 percent confidence interval: -0.7502629 -0.5873363 sample estimates: mean of x mean of y 5.307799 5.976599

Realizando el contraste de hipótesis, rechazaremos la hipótesis nula de que la calidad media de los vinos es igual para aquellos con graduación alcohólica alta y baja.

A continuación, haremos un contraste para ver si podemos afirmar que los vinos con mayor graduación alcohólica tienen mayor calidad que los que tienen baja graduación.

$$\begin{cases} H_0 : \mu_{bajo} \geq \mu_{alto} \\ H_1 : \mu_{bajo} < \mu_{alto} \end{cases}$$

```
#t.test(datosalcoholbajo$quality, datosalcoholalto$quality, alternative ="less")
```

```
Welch Two Sample t-test
```

data: datosalcoholbajo*qualityanddatosalcoholalto*quality t = -16.107, df = 1197, p-value < 2.2e-16 alternative hypothesis: true difference in means is less than 0 95 percent confidence interval: -Inf -0.6004497 sample estimates: mean of x mean of y 5.307799 5.976599

Se rechaza la hipótesis nula y por lo tanto podemos afirmar que la calidad de los vinos será mayor para aquellos vinos con una graduación alcohólica alta.

Por último, haremos lo mismo para el pH.

Plantearemos un contraste de hipótesis de comparación de medias de la calidad de dos poblaciones, donde en primer lugar tendremos los vinos con un pH bajo y por otro lado tendremos aquellos vinos que tengan un pH elevado.

$$\begin{cases} H_0 : \mu_{bajo} = \mu_{alto} \\ H_1 : \mu_{bajo} \neq \mu_{alto} \end{cases}$$

Una vez hemos planteado el contraste de hipótesis, pasamos a implementar el test de Student's

```
# t.test(datosphbajo$quality, datosphalto$quality)
```

Welch Two Sample t-test

data: datosphbajo*quality*y*anddatosphalto*quality t = 1.9781, df = 1346.8, p-value = 0.04812 alternative hypothesis: true difference in means is not equal to 0 95 percent confidence interval: 0.0007317131 0.1760422957 sample estimates: mean of x mean of y 5.665722 5.577335

Realizando el contraste de hipótesis, rechazaremos la hipótesis nula de que la calidad media de los vinos es igual para aquellos con graduación alcohólica alta y baja, aunque por muy poco, ya que el pvalor obtenido es de 0.04812, teniendo en cuenta que vamos a tener un nivel de significación del 0.05, tenemos que rechazar el contraste.

A continuación, haremos un contraste para ver si podemos afirmar que los vinos con un pH bajo tiene mayor calidad que los que tienen un pH alto.

$$\begin{cases} H_0 : \mu_{bajo} \leq \mu_{alto} \\ H_1 : \mu_{bajo} > \mu_{alto} \end{cases}$$

```
#t.test(datosphbajo$quality, datosphalto$quality, alternative ="greater")
```

```
Welch Two Sample t-test
```

data: datosphbajo*quality*y*anddatosphalto*quality t = 1.9781, df = 1346.8, p-value = 0.02406 alternative hypothesis: true difference in means is greater than 0 95 percent confidence interval: 0.01483989 Inf sample estimates: mean of x mean of y 5.665722 5.577335

En este caso, tendremos que rechazar la hipótesis nula y por lo tanto podremos afirmar que los vinos con un pH bajo tienen una mayor calidad que los vinos con un pH elevado.

4.4.3 Regresión logística Por último, con la variable categórica que creamos anteriormente a partir del campo calidad, vamos a implementar una regresión logística para intentar predecir la variable.

Para ello, utilizaremos la función glm, en la que se implementará esta técnica estadística

```
attach(datosvinocats)
```

```
## The following objects are masked from datosvino:
##
##     alcohol, chlorides, citric.acid, density, fixed.acidity,
##     free.sulfur.dioxide, pH, quality, residual.sugar, sulphates,
##     total.sulfur.dioxide, volatile.acidity
```

```
reglog<-glm(calidadcat~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+chlorides+free.sulfur.
summary(reglog)
```

```
##
## Call:
## glm(formula = calidadcat ~ fixed.acidity + volatile.acidity +
##     citric.acid + residual.sugar + chlorides + free.sulfur.dioxide +
##     total.sulfur.dioxide + density + pH + sulphates + alcohol,
##     family = "binomial", data = datosvinocats)
```

```
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.3649   0.1359   0.2007   0.3021   1.7258
##
## Coefficients:
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)          -3.731e+02  1.919e+02  -1.944 0.051861 .
## fixed.acidity        -4.819e-01  2.480e-01  -1.943 0.051967 .
## volatile.acidity     -4.390e+00  8.344e-01  -5.261 1.43e-07 ***
## citric.acid          -8.849e-01  1.249e+00  -0.708 0.478773
## residual.sugar       -3.386e-01  1.110e-01  -3.050 0.002289 **
## chlorides            -6.194e+00  3.069e+00  -2.018 0.043554 *
## free.sulfur.dioxide   1.711e-02  2.307e-02   0.742 0.458235
## total.sulfur.dioxide  1.564e-02  8.328e-03   1.878 0.060343 .
## density               3.953e+02  1.956e+02   2.021 0.043286 *
## pH                   -5.671e+00  1.676e+00  -3.384 0.000714 ***
## sulphates             1.149e+00  1.294e+00   0.888 0.374450
## alcohol               7.416e-01  2.515e-01   2.948 0.003194 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 510.03  on 1358  degrees of freedom
## Residual deviance: 412.93  on 1347  degrees of freedom
## AIC: 436.93
##
## Number of Fisher Scoring iterations: 7
```

De nuevo, buscaremos el modelo óptimo a partir de la función step

```
reglogopt<-step(reglog)
```

```
## Start:  AIC=436.93
## calidadcat ~ fixed.acidity + volatile.acidity + citric.acid +
##     residual.sugar + chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + sulphates + alcohol
##
##                        Df Deviance    AIC
## - citric.acid           1   413.43 435.43
## - free.sulfur.dioxide   1   413.50 435.50
## - sulphates             1   413.77 435.77
## <none>                      412.93 436.93
## - chlorides             1   416.46 438.46
## - fixed.acidity         1   416.57 438.57
## - total.sulfur.dioxide  1   416.95 438.95
## - density               1   417.04 439.04
## - residual.sugar        1   420.26 442.26
## - alcohol               1   422.28 444.28
## - pH                    1   423.94 445.94
## - volatile.acidity      1   440.83 462.83
##
## Step:  AIC=435.43
## calidadcat ~ fixed.acidity + volatile.acidity + residual.sugar +
```

```
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + sulphates + alcohol
##
##                        Df Deviance    AIC
## - sulphates             1   414.13 434.13
## - free.sulfur.dioxide   1   414.26 434.26
## <none>                      413.43 435.43
## - total.sulfur.dioxide  1   416.96 436.96
## - chlorides             1   417.68 437.68
## - density               1   418.20 438.20
## - fixed.acidity         1   419.31 439.31
## - residual.sugar        1   421.29 441.29
## - alcohol               1   422.55 442.55
## - pH                    1   424.63 444.63
## - volatile.acidity      1   444.83 464.83
##
## Step:  AIC=434.13
## calidadcat ~ fixed.acidity + volatile.acidity + residual.sugar +
##     chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
##     density + pH + alcohol
##
##                        Df Deviance    AIC
## - free.sulfur.dioxide   1   415.07 433.07
## <none>                      414.13 434.13
## - chlorides             1   417.70 435.70
## - total.sulfur.dioxide  1   417.81 435.81
## - density               1   420.10 438.10
## - fixed.acidity         1   420.47 438.47
## - residual.sugar        1   423.12 441.12
## - alcohol               1   425.42 443.42
## - pH                    1   425.71 443.71
## - volatile.acidity      1   454.34 472.34
##
## Step:  AIC=433.07
## calidadcat ~ fixed.acidity + volatile.acidity + residual.sugar +
##     chlorides + total.sulfur.dioxide + density + pH + alcohol
##
##                        Df Deviance    AIC
## <none>                      415.07 433.07
## - chlorides             1   418.57 434.57
## - density               1   420.64 436.64
## - fixed.acidity         1   421.23 437.23
## - residual.sugar        1   423.45 439.45
## - pH                    1   426.09 442.09
## - alcohol               1   426.17 442.17
## - total.sulfur.dioxide  1   429.10 445.10
## - volatile.acidity      1   456.89 472.89
```

```
library(ResourceSelection)
```

```
## Warning: package 'ResourceSelection' was built under R version 3.5.3
```

```
## ResourceSelection 0.3-5   2019-07-22
```

```
hoslem.test(datosvinocats$calidadcat, fitted(reglogopt))
```

```
## Warning in Ops.factor(1, y): '-' not meaningful for factors
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  datosvinocats$calidadcat, fitted(reglogopt)
## X-squared = 1359, df = 8, p-value < 2.2e-16
```

Obtenemos un pvalor muy cercano a 0, por lo que tendremos que determinar que nuestro modelo no está bien ajustado, sin embargo este test es muy sensible a muestras grandes, como es el caso de nuestro modelo de estudio, por lo que tampoco tomaremos este contraste como la mejor herramienta para determinar si nuestro modelo es bueno o no.

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

Una vez terminados los análisis que hemos realizado (que son regresión lineal múltiple, regresión logística y contrastes de hipótesis), podremos sacar las siguientes conclusiones:

· De la regresión lineal múltiple no podemos sacar ninguna conclusión importante, ya que como hemos visto, el estadístico R^2 que obtenemos es muy bajo. Además se ha intentado optimizar a partir de la función step y no hemos obtenido ninguna mejoría, por lo que no nos ha aportado ninguna información

· Con la regresión logística nos ocurre algo similar al caso anterior, ya que no obtenemos un modelo bueno con el que predecir la calidad de los vinos. De igual forma intentamos optimizarlo sin conseguir mejoría alguna

· En el caso de los contrastes de hipótesis, si que hemos obtenido información relevante, ya que gracias a ellos podemos asegurar que los vinos con mayor graduación alcohólica tendrán una calidad mayor, así como que los vinos con un ph bajo también tendrán una mayor calidad.

6. Exportación de ficheros utilizados y enlace a GitHub

```
# Fichero tras limpieza de datos
write.csv(datosvino, file = "datosvino.csv")

# Fichero con variables categóricas creadas

write.csv(datosvinocats, file = "datosvinocats.csv")
```

Enlace a GitHub

7. Tabla de aportaciones

Contribuciones	Firma
Investigación previa	FFP, JGF
Redacción de las respuestas	FFP, JGF
Desarrollo del código	FFP, JGF