Práctica 2 - https://github.com/joserodriguezsanchezUOC/PRA2

Apellidos: Rodríguez Sánchez
Nombre: José

Descripción del dataset

Los datos tratados se han descargado de *Kaggle*, la comunidad online de científicos de datos y *machine learning* de *Google*, los cuales están recogidos en un único fichero *CSV*, separado por comas, compuesto por 12 columnas y 1.599 registros representados por los siguientes campos:

1 - ***fixed acidity***: cantidad de ácidos que no se evaporan rápidamente.
2 - ***volatile acidity***: cantidad de ácido acético, responsable del sabor avinagrado.
3 - ***citric acid****:* cantidad de ácido cítrico, el cual aporta frescura y sabor.
4 - ***residual sugar***: cantidad de azúcar residual tras la fermentación.
5 - ***chlorides***: cantidad de sal.
6 - ***free sulfur dioxide***: cantidad de dióxido de azufre no fijado a otras moléculas, el cual actúa como antioxidante y freno al crecimiento de microbios.
7 - ***total sulfur dioxide***: cantidad total de dióxido de azufre, el cual se genera de forma natural en la fermentación y actúa como antioxidante, antimicrobiano y antiséptico.
8 - ***density***: densidad del vino, la cual depende su concentración de azúcar y alcohol.
9 - ***pH***: medida de la acidez o basicidad *[0, …, 14]*
10 - ***sulphates***: sulfatos que actúan contra microbios y oxidación.
11 - ***alcohol***: porcentaje de alcohol.
12 - ***quality***: medida de calidad entre 0 y 10. Resultado de la combinación del resto de variables.

Importancia y objetivos del análisis

El conjunto de datos, recogidos y filtrados de modo que se preserva el tipo de uva, marca, precio…, fueron recogidos en 2009 y representan una oportunidad principalmente para determinar ***qué elementos y en qué proporción debemos usarlos para producir el mejor vino***, según mediciones reales podríamos ***pronosticar la calidad de la añada*** que se esté produciendo o incluso ***salvar una campaña***, al ofrecernos indicadores que nos alerten ante desequilibrios químicos en la composición que perjudiquen la calidad del vino.

Integración y selección de los datos de interés a analizar

El conjunto de datos sugerido consta de ***un fichero sencillo***, por lo que ***no tenemos la necesidad de integrar diferentes conjuntos de datos***, estudiar posibles discrepancias en formato, tipo, escala, redundancias… ni decidir un sistema de almacenamiento, puesto que contamos con un fichero separado por comas.

Por otro lado, ***no nos plantearíamos una reducción en número o dimensiones*** para obtener una representación resumida del conjunto de datos a través de técnicas como *clusters*, *sampling*,

regresión… puesto que ya contamos con un número reducido de registros y variables. Estas últimas preseleccionadas como elementos básicos de influencia en la calidad del vino.

Puesto que conocemos el formato (*CSV*), cargamos los datos en un *data frame* para su manipulación y análisis:

```
> vinos <- read.csv("C:/Users/cabopalos/Documents/UOC/TCVD/PRA2/winequality-red.csv", header=TRUE)
```

Verificamos que *los formatos interpretados por R han sido correctos*, siendo todos tipo *numeric* al tratarse de valores decimales, a excepción de *quality* establecido a entero:

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

Con el siguiente resumen comprobamos que *no tenemos valores NA (Not Available)*, lo cual indica que *no tenemos valores vacíos* —*existencia de un atributo pero que se desconoce su valor pero que puede ser hallado*— *ni nulos* —*no existencia del valor, ni siquiera de la nada*— por lo que ***no tendremos que recurrir a técnicas de valores perdidos***: *ignorar la tupla afectada,* algo que supondría pérdida de información*, cambiar por constante/media/mediana, valor más probable (regresión/árbol decisión-kNN)...* posiblemente debido a un *buen diseño del cuestionario* de recopilación de datos.

```
> str(vinos)
'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 3.51 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 ...
```

Hemos encontrado ***240 registros duplicados***, por lo que podríamos limpiar la muestra:

```
> nrow(vinos[duplicated(vinos), ])
[1] 240
> vinosDisjuntos <- vinos[!duplicated(vinos), ]
> nrow(vinosDisjuntos)
[1] 1359
```

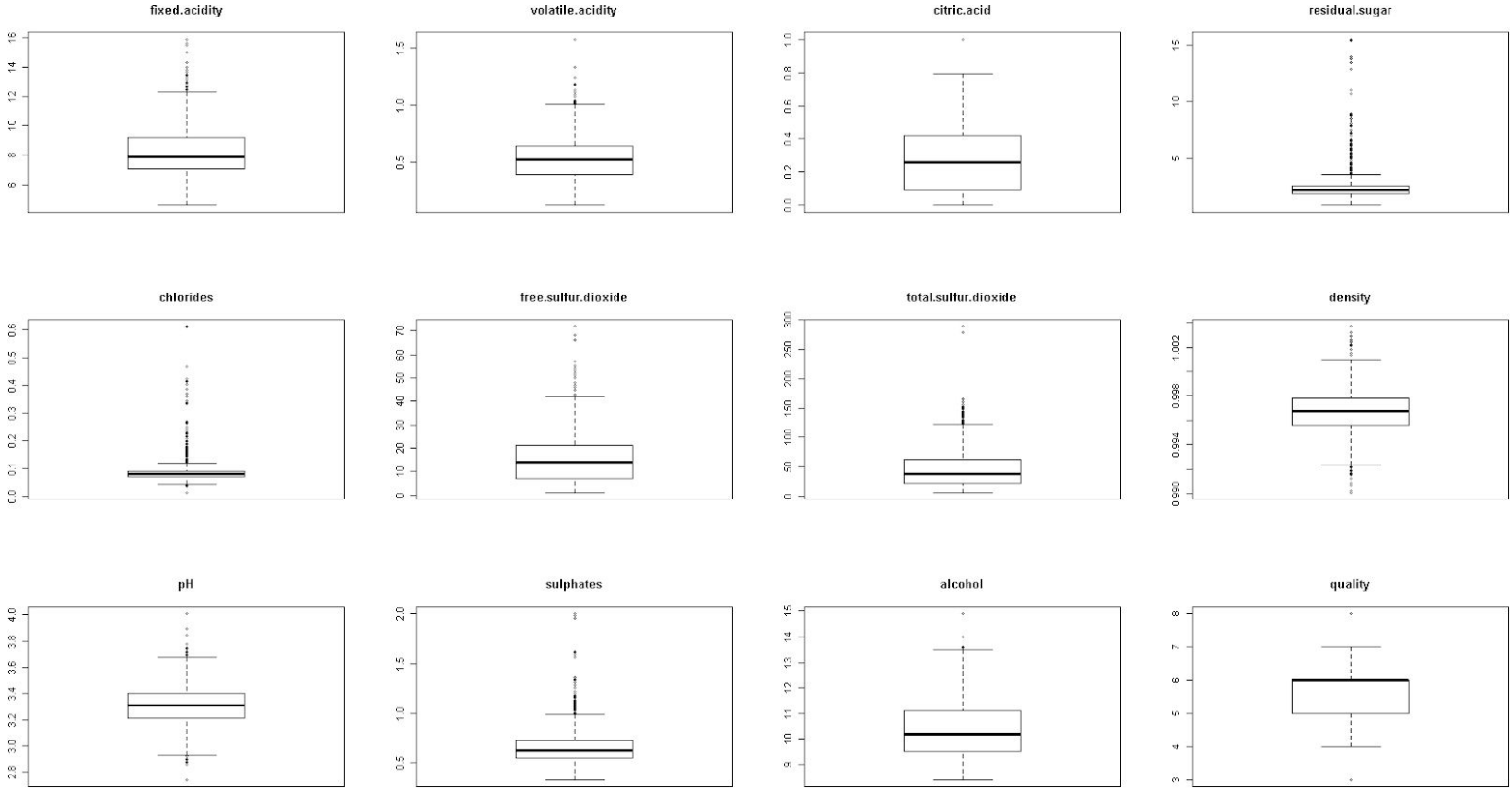
```
> summary(vinos)
 fixed.acidity   volatile.acidity  citric.acid    residual.sugar     chlorides
 Min.   : 4.60   Min.   :0.1200   Min.   :0.000   Min.   : 0.900   Min.   :0.01200
 1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090   1st Qu.: 1.900   1st Qu.:0.07000
 Median : 7.90   Median :0.5200   Median :0.260   Median : 2.200   Median :0.07900
 Mean   : 8.32   Mean   :0.5278   Mean   :0.271   Mean   : 2.539   Mean   :0.08747
 3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420   3rd Qu.: 2.600   3rd Qu.:0.09000
 Max.   :15.90   Max.   :1.5800   Max.   :1.000   Max.   :15.500   Max.   :0.61100
 free.sulfur.dioxide total.sulfur.dioxide    density             pH
 Min.   : 1.00       Min.   :  6.00       Min.   :0.9901   Min.   :2.740
 1st Qu.: 7.00       1st Qu.: 22.00       1st Qu.:0.9956   1st Qu.:3.210
 Median :14.00       Median : 38.00       Median :0.9968   Median :3.310
 Mean   :15.87       Mean   : 46.47       Mean   :0.9967   Mean   :3.311
 3rd Qu.:21.00       3rd Qu.: 62.00       3rd Qu.:0.9978   3rd Qu.:3.400
 Max.   :72.00       Max.   :289.00       Max.   :1.0037   Max.   :4.010
   sulphates         alcohol          quality
 Min.   :0.3300   Min.   : 8.40   Min.   :3.000
 1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000
 Median :0.6200   Median :10.20   Median :6.000
 Mean   :0.6581   Mean   :10.42   Mean   :5.636
 3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000
 Max.   :2.0000   Max.   :14.90   Max.   :8.000
```

Identificación y tratamiento de valores extremos

Para identificar posibles *outliers*, recurriremos a ***boxplots***, mediante diagramas de "*caja y bigotes*" a través de:

```
> par(mfrow=c(3,4))
> boxplot(vinos$fixed.acidity, main="fixed.acidity")
> boxplot(vinos$volatile.acidity, main="volatile.acidity")
> boxplot(vinos$citric.acid , main="citric.acid")
> boxplot(vinos$residual.sugar, main="residual.sugar")
> boxplot(vinos$chlorides, main="chlorides")
> boxplot(vinos$free.sulfur.dioxide, main="free.sulfur.dioxide")
> boxplot(vinos$total.sulfur.dioxide, main="total.sulfur.dioxide")
> boxplot(vinos$density, main="density")
> boxplot(vinos$pH, main="pH")
> boxplot(vinos$sulphates, main="sulphates")
> boxplot(vinos$alcohol, main="alcohol")
> boxplot(vinos$quality, main="quality")
```

donde tenemos una visión clara de cómo se distribuyen las variables



pudiendo recoger directamente los valores “sospechosos” en todas gracias a los siguientes comandos:

```
> boxplot.stats(vinos$fixed.acidity)$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 14.0 13.7 13.7 12.7 12.5 12.8 12.6
[23] 15.6 12.5 13.0 12.5 13.3 12.4 12.5 12.9 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
[45] 15.9 13.3 12.9 12.6 12.6
> boxplot.stats(vinos$volatile.acidity)$out
 [1] 1.130 1.020 1.070 1.330 1.330 1.040 1.090 1.040 1.240 1.185 1.020 1.035 1.025 1.115 1.020 1.020 1.580 1.180 1.040
> boxplot.stats(vinos$citric.acid)$out
[1] 1
> boxplot.stats(vinos$residual.sugar)$out
  [1]  6.10  6.10  3.80  3.90  4.40 10.70  5.50  5.90  5.90  3.80  5.10  4.65  4.65  5.50  5.50  5.50  5.50  7.30
 [19]  7.20  3.80  5.60  4.00  4.00  4.00  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  5.20 15.50  4.10  8.30  6.55  6.55
 [55]  4.60  6.10  4.30  5.80  5.15  6.30  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  6.00  6.00  3.80  9.00  4.60  8.80
 [91]  8.80  5.00  3.80  4.10  5.90  4.10  6.20  8.90  4.00  3.90  4.00  8.10  8.10  6.40  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  4.30 13.40  4.80  6.30  4.50  4.50
[127]  4.30  4.30  3.90  3.80  5.40  3.80  6.10  3.90  5.10  5.10  3.90 15.40 15.40  4.80  5.20  5.20  3.75 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
> boxplot.stats(vinos$chlorides)$out
  [1] 0.176 0.170 0.368 0.341 0.172 0.332 0.464 0.401 0.467 0.122 0.178 0.146 0.236 0.610 0.360 0.270 0.039 0.337
 [19] 0.263 0.611 0.358 0.343 0.186 0.213 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 0.122 0.124 0.124 0.143 0.222 0.039
 [55] 0.157 0.422 0.034 0.387 0.415 0.157 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 0.369 0.166 0.166 0.136 0.132 0.132
 [91] 0.123 0.123 0.123 0.403 0.137 0.414 0.166 0.168 0.415 0.153 0.415 0.267 0.123 0.214 0.214 0.169 0.205 0.205
[109] 0.039 0.235 0.230 0.038
> boxplot.stats(vinos$free.sulfur.dioxide)$out
 [1] 52 51 50 68 68 43 47 54 46 45 53 52 51 45 57 50 45 48 43 48 72 43 51 51 52 55 55 48 48 66
```

```
> boxplot.stats(vinos$total.sulfur.dioxide)$out
 [1] 145 148 136 125 140 136 133 153 134 141 129 128 129 128 143 144 127 126 145 144 135 165 124 124 134 124 129 151
[29] 133 142 149 147 145 148 155 151 152 125 127 139 143 144 130 278 289 135 160 141 141 133 147 147 131 131 131
> boxplot.stats(vinos$density)$out
 [1] 0.99160 0.99160 1.00140 1.00150 1.00150 1.00180 0.99120 1.00220 1.00220 1.00140 1.00140 1.00140 1.00140 1.00320
[15] 1.00260 1.00140 1.00315 1.00315 1.00315 1.00210 1.00210 0.99170 0.99220 1.00260 0.99210 0.99154 0.99064 0.99064
[29] 1.00289 0.99162 0.99007 0.99007 0.99020 0.99220 0.99150 0.99157 0.99080 0.99084 0.99191 1.00369 1.00369 1.00242
[43] 0.99182 1.00242 0.99182
> boxplot.stats(vinos$pH)$out
 [1] 3.90 3.75 3.85 2.74 3.69 3.69 2.88 2.86 3.74 2.92 2.92 2.92 3.72 2.87 2.89 2.89 2.92 3.90 3.71 3.69 3.69 3.71
[23] 3.71 2.89 2.89 3.78 3.70 3.78 4.01 2.90 4.01 3.71 2.88 3.72 3.72
> boxplot.stats(vinos$sulphates)$out
 [1] 1.56 1.28 1.08 1.20 1.12 1.28 1.14 1.95 1.22 1.95 1.98 1.31 2.00 1.08 1.59 1.02 1.03 1.61 1.09 1.26 1.08 1.00
[23] 1.36 1.18 1.13 1.04 1.11 1.13 1.07 1.06 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
[45] 1.18 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
> boxplot.stats(vinos$alcohol)$out
 [1] 14.00000 14.00000 14.00000 14.00000 14.90000 14.00000 13.60000 13.60000 13.60000 14.00000 14.00000 13.56667
[13] 13.60000
> boxplot.stats(vinos$quality)$out
 [1] 8 8 8 8 8 3 8 8 8 3 8 3 8 3 3 8 8 8 8 8 3 3 8 8 3 3 3 8
```

Como puede verse, *encontramos valores atípicos* examinando las variables de forma independiente, aunque podríamos haber recurrido por ejemplo a la *distancia de Cook* para un análisis multivariante pero, sin una base adecuada sobre el dominio —en un caso real consultaríamos a un experto en la materia— ***optaríamos por mantener el conjunto de datos original*** para no perder la variabilidad de la muestra.

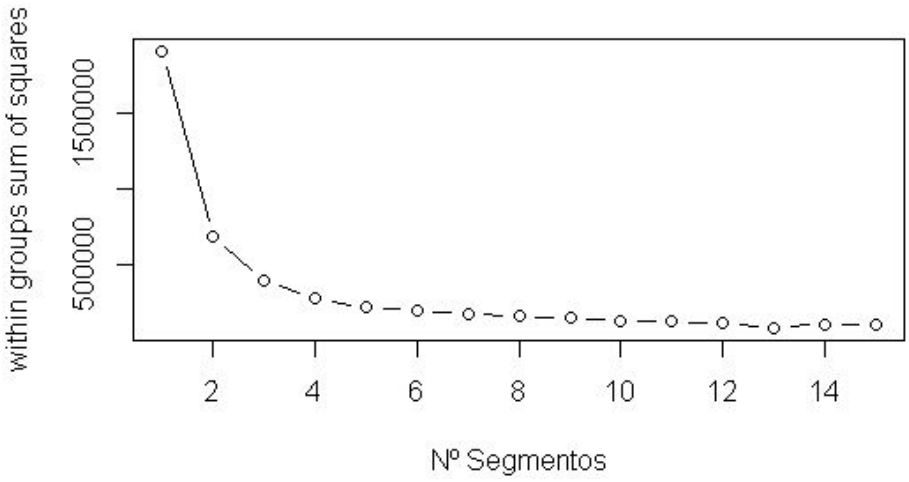
Análisis de los datos

En esta fase incipiente del análisis, en la que todavía no se tiene demasiado conocimiento de los datos, podríamos intentar descubrir si la información está estructurada en grupos o subpoblaciones homogéneas, en cuanto a la semejanza de sus elementos, dentro de los datos globales.

Para ello recurrimos a técnicas de *clustering* (clasificación no supervisada, aprendizaje no supervisado)cuyo objetivo es particionar o segmentar el conjunto de datos en grupos basados en la similaridad de los individuos en ciertas variables. Como los grupos no son dados a priori, utilizaremos el método comúnmente llamado ***Elbow***, el cual se basa en la representación gráfica de la suma de los cuadrados de los errores (SSE)

```
> wss <- (nrow(vinos)-1) * sum(apply(vinos, 2, var))
> for (i in 2:15) wss[i] <- sum(kmeans(vinos, centers=i,iter.max=30)$withinss)
> plot(1:15, wss, type="b", xlab="N° Segmentos", ylab="within groups sum of squares")
```

en la que elegiremos el número de clusters que coincida con una reducción brusca de SSE donde, a partir de la misma, no aparecerán cambios sustanciales



En nuestro caso particular, apreciamos que a partir del "*codo*" 4 no se producen cambios significativos, y también por simplicidad de interpretación, lo escogeremos como número apropiado de cluster para construir nuestro modelo.

```
> kmvinos <- kmeans(vinos, 4, 15)
> print(kmvinos)
K-means clustering with 4 clusters of sizes 516, 265, 102, 716

Cluster means:
  fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides free.sulfur.dioxide
1      8.194380        0.5267829   0.2521512       2.400969 0.09007558           19.045543
2      8.070943        0.5495283   0.2777358       2.906226 0.09103396           24.739623
3      8.026471        0.5494608   0.3193137       3.345588 0.08958824           29.897059
4      8.543715        0.5174511   0.2751536       2.387221 0.08396369            8.311453
  total.sulfur.dioxide   density       pH sulphates   alcohol  quality
1             47.25775 0.9967347 3.330581 0.6727519 10.412274 5.670543
2             82.77358 0.9970061 3.322792 0.6445283 10.153019 5.471698
3            130.07843 0.9970477 3.228824 0.6890196  9.856863 5.117647
4             20.55028 0.9966164 3.304483 0.6482682 10.611266 5.745810

Clustering vector:
   [1] 4 2 1 1 4 1 1 4 4 2 1 2 1 4 3 3 2 1 4 1 1 2 1 1 1 4 4 1 1 4 2 1 3 2 1 4 4 4 4 2 2 1 4
...
[1592] 4 1 1 1 1 1 1 1

Within cluster sum of squares by cluster:
[1] 76237.23 69215.13 85570.89 53124.72
 (between_SS / total_SS =  85.2 %)

Available components:

[1] "cluster"      "centers"      "totss"        "withinss"     "tot.withinss"
[6] "betweenss"    "size"         "iter"         "ifault"
```

El valor "*between_SS / total_SS*" la cual *nos orienta sobre la bondad del modelo* en el base a la distancia de los elementos intra-grupo y entre diferentes grupos, y cuyo valor ideal sería 1, podríamos confirmar que el modelo es realmente adecuado, muy cercano al 100% (*85,2%*).

```
> aggregate(vinos, by=list(kmvinos$cluster),FUN=mean)
  Group.1 fixed.acidity volatile.acidity citric.acid residual.sugar  chlorides
1       1      8.194380        0.5267829   0.2521512       2.400969 0.09007558
2       2      8.070943        0.5495283   0.2777358       2.906226 0.09103396
3       3      8.026471        0.5494608   0.3193137       3.345588 0.08958824
4       4      8.543715        0.5174511   0.2751536       2.387221 0.08396369
  free.sulfur.dioxide total.sulfur.dioxide   density       pH sulphates   alcohol  quality
1           19.045543              47.25775 0.9967347 3.330581 0.6727519 10.412274 5.670543
2           24.739623              82.77358 0.9970061 3.322792 0.6445283 10.153019 5.471698
3           29.897059             130.07843 0.9970477 3.228824 0.6890196  9.856863 5.117647
4            8.311453              20.55028 0.9966164 3.304483 0.6482682 10.611266 5.745810


> vinosKM <- data.frame(vinos, kmvinos$cluster)
> head(vinosKM)
  fixed.acidity volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide
1           7.4             0.70        0.00            1.9     0.076                  11
2           7.8             0.88        0.00            2.6     0.098                  25
3           7.8             0.76        0.04            2.3     0.092                  15
4          11.2             0.28        0.56            1.9     0.075                  17
5           7.4             0.70        0.00            1.9     0.076                  11
6           7.4             0.66        0.00            1.8     0.075                  13
  total.sulfur.dioxide density   pH sulphates alcohol quality kmvinos.cluster
1                   34  0.9978 3.51      0.56     9.4       5               4
2                   67  0.9968 3.20      0.68     9.8       5               2
3                   54  0.9970 3.26      0.65     9.8       5               1
4                   60  0.9980 3.16      0.58     9.8       6               1
5                   34  0.9978 3.51      0.56     9.4       5               4
6                   40  0.9978 3.51      0.56     9.4       5               1

> vinosKM$kmvinos.cluster <- factor(vinosKM$kmvinos.cluster)

scatterplotMatrix(~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+chlorides+free.sulfur.dioxide+

total.sulfur.dioxide+density+pH+sulphates+alcohol | kmvinos.cluster, reg.line=FALSE, smooth=FALS$
```

De los datos anteriores y con el gráfico representación a continuación podremos apreciar la *relación entre cada par de variables* de manera detallada y su *agrupación creada en base a la similitud de los individuos*, dejando a la vista las características de los diferentes vinos en grupos y donde *empezamos a vislumbrar la importancia de determinadas variables* que prácticamente se mantienen iguales en los diferentes grupos, como *fixed.acidity*, *volatile.acidity*, *density* o *pH* y otras que realmente cambian entre ellos: *alcohol*, *free.sulfur.dioxide*.

Grupos *cluster*, niveles significativos:

1 - **azul**:

Alta cantidad de *alcohol*, *sulphates*, bajos *sulfure dioxides*, altos *chlorides* → ***Calidad media-alta***

2 - **magenta**

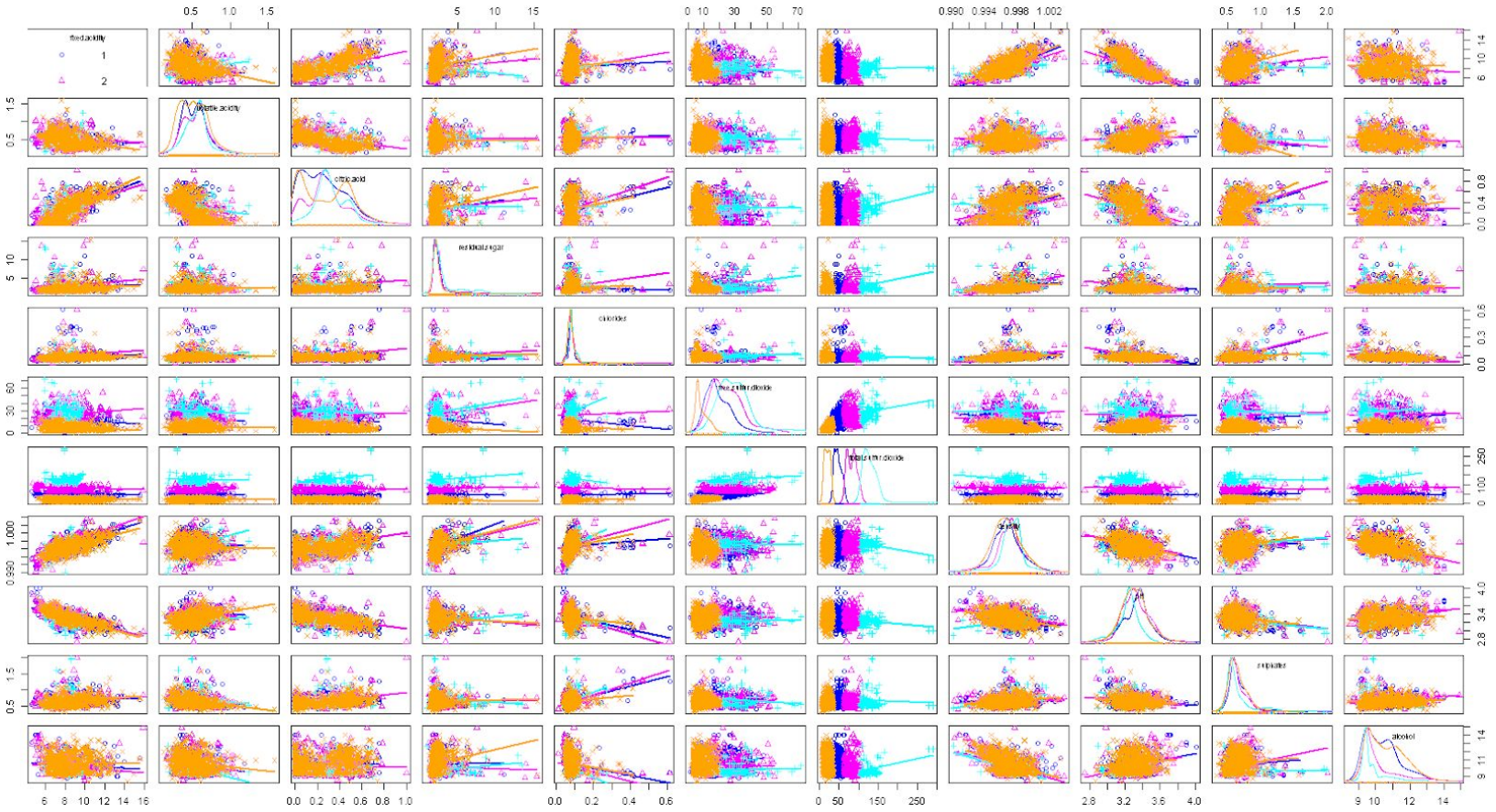
bajo free.sulfur.dioxide, medio *alcohol* → ***Calidad media***

3 - **cyan**

Alto *citric acid*, bajo *alcohol*, altos *sulfure dioxides* → ***Calidad media-baja***

4 - **naranja**,

Alta cantidad de *alcohol*, *sulphates*, muy bajos *sulfure dioxides*, bajos *chlorides* → ***Calidad alta***



Comprobación de la normalidad y homogeneidad de la varianza

Para comprobar la normalidad en la distribución de las variables que estamos manejando, usaremos el *test de normalidad de Anderson-Darling* tomando como valor de $\alpha = 0,05$ y, por tanto, considerando que la variable sigue una distribución normal en caso de que su *p-valor* sea superior.

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

Por otro lado, para comprobar la *homogeneidad de la varianza* (*homocedasticidad*) recurriremos al *test de Fligner-Killeen* particularmente sobre los *grupos definidos sobre la puntuación de la calidad del vino* (*quality*), pero comprobamos que el *p-valor* es mucho menor que nuestro umbral de 0,05 por lo que *rechazamos la hipótesis de que las varianzas de los diferentes grupos que definiría la calidad del vino sean homogéneas*.

```
> fligner.test(quality ~ alcohol, data = vinos)

        Fligner-Killeen test of homogeneity of variances

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

Pruebas estadísticas

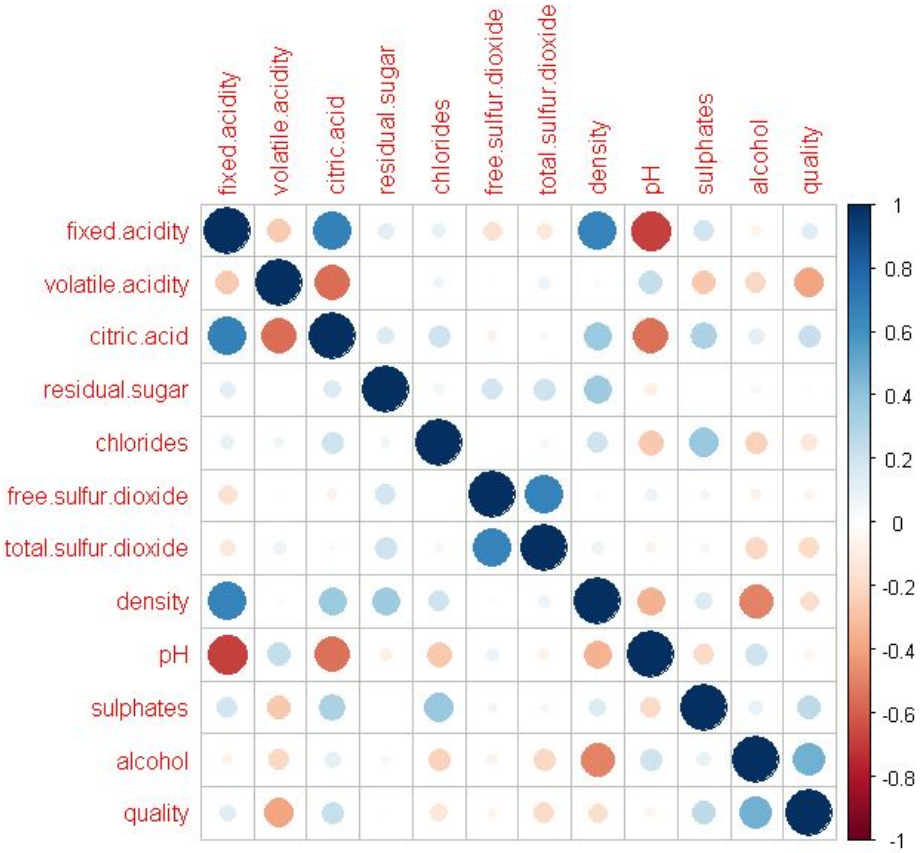
Podemos comenzar por estudiar qué variables cuantitativas influyen más en la calidad de los vinos a través de la siguiente matriz de ***correlación***:

```
> corr_matrix <- matrix(nc = 2, nr = 0)
> colnames(corr_matrix) <- c("estimate", "p-value")
> for (i in 1:ncol(vinos) - 1) {
+ if (is.integer(vinos[,i]) | is.numeric(vinos[,i])) {
+ spearman_test = cor.test(vinos[,i], vinos[,length(vinos)], method = "spearman")
+ corr_coef = spearman_test$estimate
+ p_val = spearman_test$p.value
+
+ pair = matrix(ncol = 2, nrow = 1)
+ pair[1][1] = corr_coef
+ pair[2][1] = p_val
+ corr_matrix <- rbind(corr_matrix, pair)
+ rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(vinos)[i]
+ }
+ }
> print(corr_matrix)
                       estimate      p-value
fixed.acidity         0.11408367 4.801220e-06
volatile.acidity     -0.38064651 2.734944e-56
citric.acid           0.21348091 6.158952e-18
residual.sugar        0.03204817 2.002454e-01
chlorides            -0.18992234 1.882858e-14
free.sulfur.dioxide  -0.05690065 2.288322e-02
total.sulfur.dioxide -0.19673508 2.046488e-15
density              -0.17707407 9.918139e-13
pH                   -0.04367193 8.084594e-02
sulphates             0.37706020 3.477695e-55
alcohol               0.47853169 2.726838e-92
```

donde los más cercanos a -1 y 1: *alcohol, volatile.acidity, sulphates, citric.acid…* son *los que más efecto tienen sobre la calidad* y los que menos, aquellos más alejados de estos valores: *residual.sugar, pH, free.sulfur.dioxide*…

Gracias al paquete ***corrplot*** podemos aclararnos de forma visual:

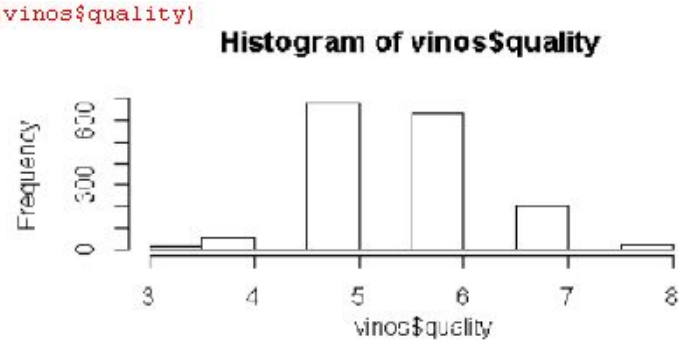
```
> install.packages("corrplot")
Installing package into 'C:/Users/cabopalos/Documents/R/win-library/3.4'
(as 'lib' is unspecified)
--- Please select a CRAN mirror for use in this session ---
probando la URL 'https://cloud.r-project.org/bin/windows/contrib/3.4/corrplot_0.84.zip'
Content type 'application/zip' length 5417997 bytes (5.2 MB)
downloaded 5.2 MB

package 'corrplot' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
        C:\Users\cabopalos\AppData\Local\Temp\RtmpUvGtVp\downloaded_packages

> corrplot(cor(vinos), method = "circle")
Error in corrplot(cor(vinos), method = "circle") :
  no se pudo encontrar la función "corrplot"
> library(corrplot)
corrplot 0.84 loaded
Warning message:
package 'corrplot' was built under R version 3.4.4
> corrplot(cor(vinos), method = "circle")
```



En cualquier caso, para confirmar la relación de la calidad con las variables anteriormente mencionadas, podemos realizar un ***Contraste hipótesis*** sobre vinos selectos creando una nueva variable booleana "*premium*" que tomará valor 1 en caso de que *quality* > *6* y 0 en el resto de casos, quedando segregados los vinos *Premium* del resto.

```
> table(vinos$quality)
  3   4   5   6   7   8
 10  53 681 638 199  18
```

```
> hist(vinos$quality)
```



Afortunadamente tenemos una *muestra de gran tamaño* para utilizar esta prueba paramétrica a pesar de la falta de normalidad en los datos.

Nos plantearemos la ***hipótesis*** de *igualdad de medias de alcohol entre los vinos*, Premium y el resto:

```
> vinos$premium <- ifelse(vinos$quality > 6, 1, 0)
> vinos.standard.alcohol <- vinos[vinos$premium == 0,]$alcohol
> vinos.premium.alcohol <- vinos[vinos$premium == 1,]$alcohol

> t.test(vinos.standard.alcohol, vinos.premium.alcohol, alternative = "less")

        Welch Two Sample t-test

data:  vinos.standard.alcohol and vinos.premium.alcohol
t = -17.45, df = 283.78, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
      -Inf -1.147195
sample estimates:
mean of x mean of y
 10.25104  11.51805
```

y como *p-value* es claramente menor que el valor de significación fijado (*0,05*), ***se confirma la relación entre la cantidad de alcohol del vino con su calidad***, es decir, a mayor graduación de alcohol el vino será de mayor calidad y, por tanto, entraría en una clasificación *Premium*.

Modelo de regresión lineal

Como mencionamos en la introducción, resultaría de especial interés realizar predicciones sobre la calidad del vino dadas su composición, de modo que podríamos diseñar una producción de calidad e incluso corregir en el mismo proceso de fermentación para ajustar la calidad en función de los intereses de la bodega.

Para dar respuesta a estas necesidades podemos crear algunos modelo de regresión lineal con todas las variables del conjunto de datos

```
modelo <- lm(quality ~ fixed.acidity + volatile.acidity + citric.acid + residual.sugar + chlorides + free.sulfur.dioxide + total.sulfur.dioxide + density + pH + sulphates + alcohol, data = vinos)
```

y con las variables que hemos considerado más determinantes en la calidad, obtenidas en el análisis de correlación realizado anteriormente:

```
> modelo2 <- lm(quality ~ volatile.acidity + citric.acid + total.sulfur.dioxide + sulphates + alcohol, data = vinos)
> summary(modelo2)$r.squared
```

incluso podríamos crear alguno adaptándonos a las medidas que tengamos disponibles en el procesos de producción ya que quizá con un par de variables ya podríamos tener una predicción aceptable.

Para determinar la *calidad del ajuste* de los modelos nos fijamos en el valor de R^2 o *coeficiente de determinación*:

```
> summary(modelo)$r.squared
[1] 0.3605517
> summary(modelo2)$r.squared
[1] 0.3438525
```

a mayor valor mejor será la bondad del ajuste del modelo.

De este modo, podríamos comprobar el ajuste de los modelos a través de muestras con valores de nuestro *dataset* y comprobando que la predicción del valor de calidad se corresponde con el valor real de nuestro conjunto de datos, como en las siguientes muestra:

```
> muestra2 <- data.frame(volatile.acidity = 0.70, citric.acid = 0, total.sulfur.dioxide = 34, sulphates = 0.56, alcohol = 9.4)
> predict(modelo2, muestra2)
       1
5.091925
```

donde se “acierta” en un valor muy cercano a su valor real (*5*).

Conclusiones

Tras la imprescindible comprobación de los datos previa al análisis de datos, en nuestro caso basada en la constatación de la ausencia de datos vacíos o ceros y la revisión de *outliers*, los cuales se han decidido mantener por sus valores realmente posibles dentro de la variabilidad de la composición de los vinos, manteniendo de este modo la riqueza de la información, hemos detectado qué componentes resultan esenciales en la definición de la calidad final de los vinos, gracias a los análisis de correlación junto al contraste de hipótesis.

Finalmente, a través de la construcción de un modelo de regresión lineal, con mayor o menor número de medidas, se ha conseguido obtener una herramienta para lograr los objetivos del análisis: ser capaz de diseñar una campaña para conseguir el nivel de calidad más adecuado y, por otro lado, controlar o corregir el proceso de producción mediante controles de los principales componentes para evitar caldos con una calidad fuera de lo esperado.

Código en R

```
vinos <-
read.csv("C:/Users/cabopalos/Documents/UOC/TCVD/PRA2/winequality-red.csv",
header=TRUE)

sapply(vinos, function(x) class(x))

str(vinos)

nrow(vinos[duplicated(vinos), ])
vinosDisjuntos <- vinos[!duplicated(vinos), ]
nrow(vinosDisjuntos)

summary(vinos)

par(mfrow=c(3,4))
boxplot(vinos$fixed.acidity, main="fixed.acidity")
boxplot(vinos$volatile.acidity, main="volatile.acidity")
boxplot(vinos$citric.acid, main="citric.acid")
boxplot(vinos$residual.sugar, main="residual.sugar")
boxplot(vinos$chlorides, main="chlorides")
boxplot(vinos$free.sulfur.dioxide, main="free.sulfur.dioxide")
boxplot(vinos$total.sulfur.dioxide, main="total.sulfur.dioxide")
boxplot(vinos$density, main="density")
boxplot(vinos$pH, main="pH")
boxplot(vinos$sulphates, main="sulphates")
boxplot(vinos$alcohol, main="alcohol")
boxplot(vinos$quality, main="quality")

boxplot.stats(vinos$fixed.acidity)$out
boxplot.stats(vinos$volatile.acidity)$out
boxplot.stats(vinos$citric.acid)$out
boxplot.stats(vinos$residual.sugar)$out
boxplot.stats(vinos$chlorides)$out
boxplot.stats(vinos$free.sulfur.dioxide)$out
boxplot.stats(vinos$total.sulfur.dioxide)$out
boxplot.stats(vinos$density)$out
boxplot.stats(vinos$pH)$out
boxplot.stats(vinos$sulphates)$out
boxplot.stats(vinos$alcohol)$out
boxplot.stats(vinos$quality)$out
```

```
wss <- (nrow(vinos)-1) * sum(apply(vinos, 2, var))
for (i in 2:15) wss[i] <- sum(kmeans(vinos, centers=i,iter.max=30)$withinss)
plot(1:15, wss, type="b", xlab="N° Segmentos", ylab="within groups sum of
squares")

kmvinos <- kmeans(vinos, 4, 15)
print(kmvinos)

aggregate(vinos, by=list(kmvinos$cluster),FUN=mean)
vinosKM <- data.frame(vinos, kmvinos$cluster)
head(vinosKM)

vinosKM$kmvinos.cluster <- factor(vinosKM$kmvinos.cluster)

scatterplotMatrix(~fixed.acidity+volatile.acidity+citric.acid+residual.sugar+ch
lorides+free.sulfur.dioxide+total.sulfur.dioxide+density+pH+sulphates+alcohol |
kmvinos.cluster, reg.line=FALSE, smooth=FALSE, spread=FALSE, span=0.5,
ellipse=FALSE, levels=c(.5, .9), id.n=0, diagonal= 'density', by.groups=TRUE,
data=vinosKM)

library(nortest)
alpha = 0.05
col.names = colnames(vinos)

for (i in 1:ncol(vinos)) {
      if (i == 1) cat("Variables que no siguen una distribución normal:\n")
      if (is.integer(vinos[,i]) | is.numeric(vinos[,i])) {
            p_val = ad.test(vinos[,i])$p.value
            if (p_val < alpha) {
                  cat(col.names[i])
                  # Format output
                  if (i < ncol(vinos) - 1) cat(", ")
                  if (i %% 3 == 0) cat("\n")
            }
      }
}

fligner.test(quality ~ alcohol, data = vinos)
```

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

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

            pair = matrix(ncol = 2, nrow = 1)
            pair[1][1] = corr_coef
            pair[2][1] = p_val
            corr_matrix <- rbind(corr_matrix, pair)
            rownames(corr_matrix)[nrow(corr_matrix)] <- colnames(vinos)[i]
      }
}

install.packages("corrplot")
library(corrplot)
corrplot(cor(vinos), method = "circle")

table(vinos$quality)
hist(vinos$quality)

vinos$premium <- ifelse(vinos$quality > 6, 1, 0)
vinos.standard.alcohol <- vinos[vinos$premium == 0,]$alcohol
vinos.premium.alcohol <- vinos[vinos$premium == 1,]$alcohol

t.test(vinos.standard.alcohol, vinos.premium.alcohol, alternative = "less")


modelo <- lm(quality ~  fixed.acidity + volatile.acidity + citric.acid +
residual.sugar + chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
density + pH + sulphates + alcohol, data = vinos)
summary(modelo)$r.squared

modelo2 <- lm(quality ~ volatile.acidity + citric.acid + total.sulfur.dioxide +
sulphates + alcohol, data = vinos)
summary(modelo2)$r.squared

muestra <- data.frame(volatile.acidity = 7.4, citric.acid = 0,
total.sulfur.dioxide = 34, sulphates = 0.56, alcohol = 9.4)
```

```
muestra2 <- data.frame(volatile.acidity = 0.70, citric.acid = 0,
total.sulfur.dioxide = 34, sulphates = 0.56, alcohol = 9.4)
predict(modelo2, muestra2)
```