PRA2 - Limpieza y validación de datos

Antonio Caparrini Lopez

26/12/2019

Contents

Detalles de la actividad

Descripción

Como parte de la asignatura *tipología de datos y ciclo de vida de los datos* dentro del máster en ciencia de datos de la UOC este documento elabora un caso práctico que consiste en el tratamiento de un conjunto de datos utilizando las herramientas disponibles para su limpieza, validación y 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.

Competencias

En esta práctica se desarrollan las siguientes competencias del **Master 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.

Resolución

Descripción del dataset

```
winedataset <- read.csv("data/winequality-red.csv")
```

El conjunto de datos elegido para el análisis es el *Red Wine Quality* disponible en **kaggle**. Este conjunto de datos tiene un total de 1599 registros y un total de 12 columnas que pasamos a comentar a continuación:

1. **fixed acidity**: La no-volatilidad de los ácidos presentes en el vino (que no se evaporan con facilidad).
2. **volatile acidity**: Cantidad de ácido acético en el vino. Niveles muy altos llevan a un sabor desagradable a vinagre.
3. **citric acid**: En pequeñas cantidades puede dar frescura a un vino.
4. **residual sugar**: El nivel de azúcar después de que pare la fermentación del vino. Es raro encontrar vinos con una menor a 1g/litro y los vinos con más de 45g/litro se consideran dulces.
5. **chlorides**: Cantidad de sal en el vino.
6. **free sulfur dioxide**: Cantidad de dióxido de azufre libre.
7. **total sulfur dioxide**: Cantidad de dióxido de azufre libre y ligado.
8. **density**: Suele ser cercana a la densidad del agua dependiendo de los niveles de alcohol o azúcar.
9. **pH**: Describe lo ácido o básico que es el vino. Desde 0 (muy ácido) a 14 (muy básico). Suele encontrarse en torno a 3-4.
10. **sulphates**: Aditivos al vino que pueden contribuir al gas dióxido de azufre que tiene efectos antimicrobianos y antioxidantes.
11. **alcohol**: Porcentaje de alcohol.
12. **quality**: Variable de salida, medida entre 0 y 10.

Para más detalles sobre los datos referimos al trabajo original Cortez et al., 2009.

Importancia y objetivos de los análisis

A partir del dataset se pretender determinar las variables más relevantes a la hora de valorar la calidad de un vino. También podremos generar un modelo de regresión que partiendo de las variables del vino estime la calidad.

Este análisis es relevante para el campo de la enología (estudio del vino). Un soumiller (por ejemplo) que deba hacer una cata de vinos necesitaría una cantidad de tiempo elevada para catar una gran cantidad de vinos mientras que con un modelo previo que estime los mejores podría dedicar su esfuerzo y criterio experto en los que previamente van a ser mejores debido a sus características.

Limpieza de los datos

Los datos los habíamos leído previamente en el apartado anterior, y mostramos un sumario de su contenido.

```r
summary(winedataset)
```

```
##  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
##  Min.   :0.01200   Min.   : 1.00       Min.   :  6.00
##  1st Qu.:0.07000   1st Qu.: 7.00       1st Qu.: 22.00
##  Median :0.07900   Median :14.00       Median : 38.00
##  Mean   :0.08747   Mean   :15.87       Mean   : 46.47
##  3rd Qu.:0.09000   3rd Qu.:21.00       3rd Qu.: 62.00
##  Max.   :0.61100   Max.   :72.00       Max.   :289.00
##     density             pH          sulphates         alcohol
##  Min.   :0.9901   Min.   :2.740   Min.   :0.3300   Min.   : 8.40
##  1st Qu.:0.9956   1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50
##  Median :0.9968   Median :3.310   Median :0.6200   Median :10.20
##  Mean   :0.9967   Mean   :3.311   Mean   :0.6581   Mean   :10.42
##  3rd Qu.:0.9978   3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10
##  Max.   :1.0037   Max.   :4.010   Max.   :2.0000   Max.   :14.90
##     quality
##  Min.   :3.000
##  1st Qu.:5.000
##  Median :6.000
##  Mean   :5.636
##  3rd Qu.:6.000
##  Max.   :8.000
```

No vamos a descartar ninguna de las variables por el momento. En primera instancia vamos a considerarlas todas como potencialmente relevantes.

Es importante ver el tipo de variables que tenemos.

```r
str(winedataset)
```

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

Vemos que las 11 variables que utilizamos para predecir son variables numéricas continuas y la variable *quality* que es el objetivo es una variable numérica entera.

Pasamos a comprobar valores nulos o vacíos.

```
sapply(winedataset, 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
```

Ninguna de las variables tiene registros nulos o vacíos.

Análisis de los datos

Comprobamos si alguna de las variables cumple las características de una distribución normal. Para ello aplicamos el test de Shaphiro-Wilk. Aplicando este test la hipótesis nula es que la distribución cumple normalidad, por ello, para no rechazar la hipótesis nula el p-valor tiene que ser mayor que el nivel de significación elegido (alpha=0.05).

```
alpha <- 0.05
for(n in names(winedataset)){
  pvalue <- shapiro.test(winedataset[,n])$p.value
  if(pvalue>alpha){
    cat("Variable ", n, " SI es normal\n")
    print(shapiro.test(winedataset[,n]))
  }else{
    cat("Variable ", n, " NO cumple una distribución normal\n")
  }
}
```

```
## Variable  fixed.acidity  NO cumple una distribución normal
## Variable  volatile.acidity  NO cumple una distribución normal
## Variable  citric.acid  NO cumple una distribución normal
## Variable  residual.sugar  NO cumple una distribución normal
## Variable  chlorides  NO cumple una distribución normal
## Variable  free.sulfur.dioxide  NO cumple una distribución normal
## Variable  total.sulfur.dioxide  NO cumple una distribución normal
## Variable  density  NO cumple una distribución normal
## Variable  pH  NO cumple una distribución normal
## Variable  sulphates  NO cumple una distribución normal
## Variable  alcohol  NO cumple una distribución normal
## Variable  quality  NO cumple una distribución normal
```

Vemos que según este test ninguna variable cumple una distribución normal, probamos con el test de Anderson-Darling.

```
alpha <- 0.05
for(n in names(winedataset)){
  pvalue <- ad.test(winedataset[,n])$p.value
  if(pvalue>alpha){
    cat("Variable ", n, " SI es normal\n")
    print(shapiro.test(winedataset[,n]))
  }else{
    cat("Variable ", n, " NO cumple una distribución normal\n")
```

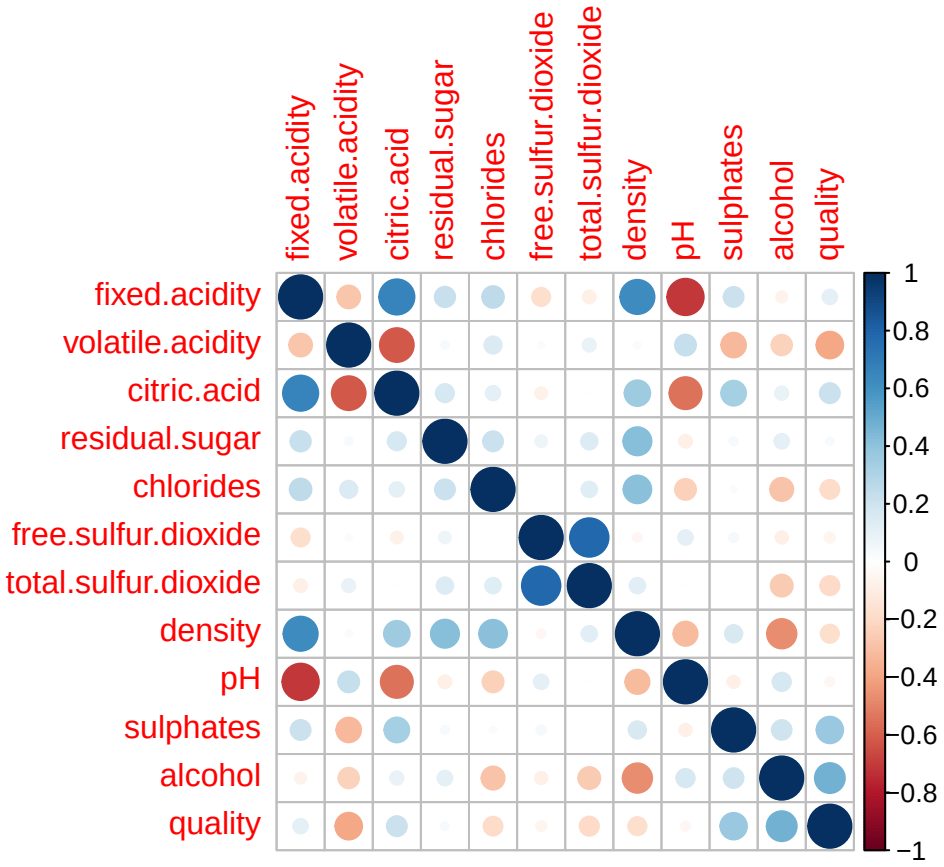
```
  }
}
```

```
## Variable  fixed.acidity  NO cumple una distribución normal
## Variable  volatile.acidity  NO cumple una distribución normal
## Variable  citric.acid  NO cumple una distribución normal
## Variable  residual.sugar  NO cumple una distribución normal
## Variable  chlorides  NO cumple una distribución normal
## Variable  free.sulfur.dioxide  NO cumple una distribución normal
## Variable  total.sulfur.dioxide  NO cumple una distribución normal
## Variable  density  NO cumple una distribución normal
## Variable  pH  NO cumple una distribución normal
## Variable  sulphates  NO cumple una distribución normal
## Variable  alcohol  NO cumple una distribución normal
## Variable  quality  NO cumple una distribución normal
```

Pruebas estadísticas

Es interesante comprobar la capacidad explicativa de la *quality* para cada variable. Para ello vamos a calcular los coeficientes de correlación de Spearman. Es necesario utilizar los de Spearman ya que nos encontramos con variables que no siguen una distribución normal.

```
corrplot(cor(winedataset, method = c("spearman")))
```



Nos encontramos con dos variables que estan positivamente correlacionadas con la calidad que son *alcohol* y *sulphates*. Estas dos variables tienen la mayor correlación positiva por lo que a mayor nivel de estas dos

mayor calidad en el vino.

Por otro lado la variable *volatile.acidity* tiene la mayor correlación negativa indicando que a mayor nivel de esta la calidad es menor. Como veíamos en la descripción de las variables esta es la cantidad de ácido acético en el vino que a mayor cantidad, mayor sabor a vinagre. Este relación es por tanto razonable.

Correlación positiva algo inferior tenemos *fixed.acidity* y *citric.acid*.

Y con una correlación negativa pero también pequeña *chlorides*, *total.sulfur.dioxide* y *density*.

Vamos a crear un modelo de regresión lineal que estime la calidad del vino. Para ello vamos a crear 11 modelos distintos y quedarnos con el mejor. Como criterio de construcción vamos a ir añadiendo las variables con el coeficiente de correlación mayor en valor absoluto.

```
cor_data <- cor(winedataset, method = c("spearman"))

quality_cor <- sort(abs(cor_data[, "quality"]))
vars_names <- names(quality_cor)

model11 <- lm(quality ~ residual.sugar + pH + free.sulfur.dioxide
              + fixed.acidity + density +
                chlorides + total.sulfur.dioxide + citric.acid
              + sulphates + volatile.acidity + alcohol,
              data=winedataset)
model10 <- lm(quality ~ pH + free.sulfur.dioxide + fixed.acidity + density +
                chlorides + total.sulfur.dioxide + citric.acid
              + sulphates + volatile.acidity + alcohol,
              data=winedataset)
model9 <- lm(quality ~ free.sulfur.dioxide + fixed.acidity + density +
                chlorides + total.sulfur.dioxide + citric.acid
             + sulphates + volatile.acidity + alcohol,
              data=winedataset)
model8 <- lm(quality ~ fixed.acidity + density +
                chlorides + total.sulfur.dioxide + citric.acid
             + sulphates + volatile.acidity + alcohol,
              data=winedataset)
model7 <- lm(quality ~ density +
                chlorides + total.sulfur.dioxide + citric.acid
             + sulphates + volatile.acidity + alcohol,
              data=winedataset)
model6 <- lm(quality ~ chlorides + total.sulfur.dioxide +
               citric.acid + sulphates + volatile.acidity +
               alcohol, data=winedataset)
model5 <- lm(quality ~ total.sulfur.dioxide + citric.acid +
               sulphates + volatile.acidity + alcohol, data=winedataset)
model4 <- lm(quality ~ citric.acid + sulphates +
               volatile.acidity + alcohol, data=winedataset)
model3 <- lm(quality ~ sulphates + volatile.acidity + alcohol, data=winedataset)
model2 <- lm(quality ~ volatile.acidity + alcohol, data=winedataset)
model1 <- lm(quality ~ alcohol, data=winedataset)
```

En la tabla a continuación visualizamos el valor de R^2 de todos los modelos. EL mejor modelo sería el que tenga el valor mayor que en este caso es el que utiliza todas las variables.

```
tabla.coeficientes <- matrix(c(
  1, summary(model1)$r.squared,
  2, summary(model2)$r.squared,
```

```
  3, summary(model3)$r.squared,
  4, summary(model4)$r.squared,
  5, summary(model5)$r.squared,
  6, summary(model6)$r.squared,
  7, summary(model7)$r.squared,
  8, summary(model8)$r.squared,
  9, summary(model9)$r.squared,
  10, summary(model10)$r.squared,
  11, summary(model11)$r.squared), ncol = 2, byrow = TRUE)

colnames(tabla.coeficientes) <- c("Modelo", "R^2")
tabla.coeficientes
```

```
##       Modelo       R^2
##  [1,]      1 0.2267344
##  [2,]      2 0.3170024
##  [3,]      3 0.3358973
##  [4,]      4 0.3361393
##  [5,]      5 0.3438525
##  [6,]      6 0.3516493
##  [7,]      7 0.3516932
##  [8,]      8 0.3557559
##  [9,]      9 0.3572112
## [10,]     10 0.3600742
## [11,]     11 0.3605517
```

El anterior modelo era de regresión, ahora vamos a crear un modelo de regresión pero de clasificación multilabel.

En la siguiente tabla vemos el valor de AIC de todos los modelos creados, el que tenga una AIC menor será el mejor modelo.

```
tabla.coeficientes <- matrix(c(
  1, summary(mmodel1)$AIC,
  2, summary(mmodel2)$AIC,
  3, summary(mmodel3)$AIC,
  4, summary(mmodel4)$AIC,
  5, summary(mmodel5)$AIC,
  6, summary(mmodel6)$AIC,
  7, summary(mmodel7)$AIC,
  8, summary(mmodel8)$AIC,
  9, summary(mmodel9)$AIC,
  10, summary(mmodel10)$AIC,
  11, summary(mmodel11)$AIC), ncol = 2, byrow = TRUE)

colnames(tabla.coeficientes) <- c("Modelo", "AIC")
tabla.coeficientes
```

```
##       Modelo      AIC
##  [1,]      1 3345.841
##  [2,]      2 3161.866
##  [3,]      3 3123.199
##  [4,]      4 3119.950
##  [5,]      5 3059.948
##  [6,]      6 3048.674
##  [7,]      7 3056.520
```

```
##  [8,]      8 3058.063
##  [9,]      9 3056.238
## [10,]     10 3050.772
## [11,]     11 3051.106
```

Con el proposito de tener una comparación visual de estos dos modelos seleccionamos aleatoriamente 10 registros del conjunto de datos y mostramos una tabla con la calidad real, la detectada por el modelo de regresión y por el modelo de clasificación.

```
my_index <- sample(1:1599,10)
elements <- winedataset[my_index,]
pred_model <- predict(model11, elements)
pred_mmodel <- predict(mmodel6, elements)
cbind(my_index, pred_model, pred_mmodel, label=winedataset$quality[my_index])
```

```
##      my_index pred_model pred_mmodel label
## 974       974   5.861585           4     5
## 758       758   5.046693           3     5
## 1236     1236   5.985615           4     4
## 639       639   5.077906           3     7
## 1547     1547   5.586964           4     5
## 434       434   5.446812           3     5
## 968       968   4.862559           3     5
## 367       367   5.571249           3     7
## 309       309   5.308262           3     6
## 1097     1097   5.409131           3     6
```

A pesar de que las etiquetas reales son enteras el modelo de regresión nos da una información más detallada ya que existen valores continuos entre una clase y otra (que podría relacionarse con la probabilidad de pertenecer a una clase que el modelo de clasificación acaba redondeando).

Conclusiones

En primer lugar se ha verificado que las variables del dataset no contenían valores nulos y que todas eran candidatas a aportar información en el análisis. No ha sido necesaria ninguna transformación en el conjunto de datos, aunque se podría categorizar alguna de las variables o disminuir la cantidad de outliers.

Se ha confirmado mediante test estadísticos la no normalidad de las variables y se ha empleado los coeficientes de correlación de Spearman para ver las variables más influyentes en la calidad. De esta manera hemos visto que el grado de alcohol y los sulfitos tienes un efecto positivo en la calidad, mientras que la cantidad de ácido acético influye en la disminución de la calidad.

Por otro lado se han creado dos modelos de regressión para estimar la calidad. Estos modelos no tienen una gran capacidad predictiva pero como se menciono en la introducción pueden ser utilizados para disminuir la cantidad de vinos que ha de catar un experto a los inicialmente estimados por el modelo.