MAS: Exame 1ª Época - Parte II (Prática do R)

André Filipe Gomes Silvestre | Nº104532

28 de março, 2023

Nota:

Deve efetuar todos os Save com "Save with encoding UTF-8" de modo a manter palavras acentuadas e caracteres especiais**

Base de dados water_potability

A base de dados water_potability é composta por 2011 observações e é definida pelas seguintes variáveis: ✓ ph: indicador de acidez/alcalinidade da água ✓ Hardness: dureza da água ✓ Solids: total de sólidos dissolvidos ✓ Chloramines: nível de desinfetantes presentes na água ✓ Sulfates: nível de sulfato na água ✓ Conductivity: grau de condutividade elétrica da água ✓ Organic_carbon: quantidade de carbono orgânicos da água ✓ Trihalomethanes: quantidade de químicos THMs encontrados na água ✓ Turbidity: turbidez da água ✓ Potability: assume o valor 1 se a água for potável, caso contrário assume o valor 0

```
# Remover tudo!
rm(list=ls(all=TRUE))

# Incluir as libraries de que necessita
library(MASS)        # The MASS library contains the Boston data set
library(Metrics)     # To help calculating metrics
library(ggplot2)     # To provide graphics
library(lsr)         # For ETA and Cramer's V measure of association
library(caret)       # Cross-validation + Metrics for classification
library(e1071)       # For classification with Naïve Bayes
library(FNN)         # Implementing KNN - K-Nearest Neighbour
library(car)         # To verify multicolinearity
library(psych)       # For some descriptives
library(nnet)        # For Multinomial Logistic Regression
library(knitr)       # To pretty outputs
library(tree)        # For Classification Tree
```

1. Leitura dos dados wat_pot e análise preliminar dos mesmos

1.1) Leitura dos dados usando wat_pot<-read.csv("water_potability.csv", header=TRUE, stringsAsFactors = TRUE, dec=".", sep=",").Nota: verifique sep no ficheiro de origem.

```
# Leitura do dataset
wat_pot<-read.csv("water_potability.csv", header=TRUE, stringsAsFactors =
TRUE, dec=".", sep=",")

# Dimensão do dataset
dim(wat_pot)

## [1] 2011   10
```

1.2) [1 valor] Apresentação de head(wat_pot) e apresentação da tabela de frequências relativas da variável Potability.

```
# Definir a variável Potability como factor
wat_pot$Potability<-factor(wat_pot$Potability,levels = c(0,1), labels =
c("No", "Yes"))

# Primeiras observações da base de dados
head(wat_pot)

##          ph Hardness   Solids Chloramines  Sulfate Conductivity Organic_carbon
## 1  8.316766 214.3734 22018.42    8.059332 356.8861     363.2665      18.436524
## 2  9.092223 181.1015 17978.99    6.546600 310.1357     398.4108      11.558279
## 3  5.584087 188.3133 28748.69    7.544869 326.6784     280.4679       8.399735
## 4 10.223862 248.0717 28749.72    7.513408 393.6634     283.6516      13.789695
## 5  8.635849 203.3615 13672.09    4.563009 303.3098     474.6076      12.363817
## 6 11.180284 227.2315 25484.51    9.077200 404.0416     563.8855      17.927806
##   Trihalomethanes Turbidity Potability
## 1       100.34167  4.628771         No
## 2        31.99799  4.075075         No
## 3        54.91786  2.559708         No
## 4        84.60356  2.672989         No
## 5        62.79831  4.401425         No
## 6        71.97660  4.370562         No

# Tabela de frequências relativas da variável Potability
prop.table(table(wat_pot$Potability))

##
##        No       Yes
## 0.5967181 0.4032819
```

1.3)[1 valor] Realização de summary dos dados wat_pot e apresentação, para as variáveis Hardness, Chloramines, Conductivity e Trihalomethanes, dos valores mínimo, máximo, média e desvio padrão.

```
# Summary dos dados wat_pot
summary(wat_pot)
```

```
##        ph              Hardness          Solids         Chloramines     
##  Min.   : 0.2275   Min.   : 73.49   Min.   :  320.9   Min.   : 1.391  
##  1st Qu.: 6.0897   1st Qu.:176.74   1st Qu.:15615.7   1st Qu.: 6.139  
##  Median : 7.0273   Median :197.19   Median :20933.5   Median : 7.144  
##  Mean   : 7.0860   Mean   :195.97   Mean   :21917.4   Mean   : 7.134  
##  3rd Qu.: 8.0530   3rd Qu.:216.44   3rd Qu.:27182.6   3rd Qu.: 8.110  
##  Max.   :14.0000   Max.   :317.34   Max.   :56488.7   Max.   :13.127  
##     Sulfate       Conductivity   Organic_carbon  Trihalomethanes  
##  Min.   :129.0   Min.   :201.6   Min.   : 2.20   Min.   :  8.577  
##  1st Qu.:307.6   1st Qu.:366.7   1st Qu.:12.12   1st Qu.: 55.953  
##  Median :332.2   Median :423.5   Median :14.32   Median : 66.542  
##  Mean   :333.2   Mean   :426.5   Mean   :14.36   Mean   : 66.401  
##  3rd Qu.:359.3   3rd Qu.:482.4   3rd Qu.:16.68   3rd Qu.: 77.292  
##  Max.   :481.0   Max.   :753.3   Max.   :27.01   Max.   :124.000  
##    Turbidity     Potability
##  Min.   :1.450   No :1200  
##  1st Qu.:3.443   Yes: 811  
##  Median :3.968             
##  Mean   :3.970             
##  3rd Qu.:4.514             
##  Max.   :6.495
```

```
# Valores mínimo, máximo, média e desvio padrão para as variáveis
Hardness, Chloramines, Conductivity e Trihalomethanes
describe(wat_pot[,c("Hardness", "Chloramines", "Conductivity",
"Trihalomethanes")])[c(8,9,3,4)]
```

```
##                     min    max   mean    sd
## Hardness          73.49 317.34 195.97 32.64
## Chloramines        1.39  13.13   7.13  1.58
## Conductivity     201.62 753.34 426.53 80.71
## Trihalomethanes    8.58 124.00  66.40 16.08
```

1.4) [1 valor] Divisão dos dados em amostra de treino (70%) e de teste (30%) usando set.seed(434) e apresentação de summary e tabela de frequências relativas da variável Potability em cada amostra.

```
# Definir o set.seed para permitir reprodutibilidade dos resultados
set.seed(434)

# Divisão em Conjunto Treino/Teste (70/30)
ind_train <- sample(nrow(wat_pot),0.7*nrow(wat_pot))

# Conjunto Treino (wat_pot_train)
wat_pot_train <- wat_pot[ind_train,]
paste("O Conjunto de Treino tem", nrow(wat_pot_train),"observações.")
```

```
## [1] "O Conjunto de Treino tem 1407 observações."
```

```
summary(wat_pot_train)
```

```
##        ph              Hardness          Solids         Chloramines    
##  Min.   : 0.2275   Min.   : 73.49   Min.   :  320.9   Min.   : 1.391  
##  1st Qu.: 6.1079   1st Qu.:177.01   1st Qu.:15762.3   1st Qu.: 6.131  
##  Median : 7.0779   Median :197.52   Median :21153.3   Median : 7.109  
##  Mean   : 7.1243   Mean   :196.08   Mean   :22072.1   Mean   : 7.119  
##  3rd Qu.: 8.0650   3rd Qu.:215.88   3rd Qu.:27416.5   3rd Qu.: 8.089  
##  Max.   :14.0000   Max.   :306.63   Max.   :56488.7   Max.   :13.044  
##     Sulfate       Conductivity   Organic_carbon  Trihalomethanes  
##  Min.   :129.0   Min.   :201.6   Min.   : 2.20   Min.   :  8.577  
##  1st Qu.:307.6   1st Qu.:364.9   1st Qu.:12.11   1st Qu.: 55.900  
##  Median :331.8   Median :421.9   Median :14.35   Median : 66.189  
##  Mean   :332.8   Mean   :424.5   Mean   :14.38   Mean   : 66.242  
##  3rd Qu.:358.1   3rd Qu.:478.6   3rd Qu.:16.75   3rd Qu.: 77.162  
##  Max.   :481.0   Max.   :753.3   Max.   :27.01   Max.   :124.000  
##    Turbidity     Potability  
##  Min.   :1.450   No :847     
##  1st Qu.:3.446   Yes:560     
##  Median :3.962               
##  Mean   :3.967               
##  3rd Qu.:4.510               
##  Max.   :6.494
```

```
# Tabela de frequências relativas da variável Potability - conjunto treino
prop.table(table(wat_pot_train$Potability))
```

```
##      No     Yes 
## 0.60199 0.39801
```

```
# Conjunto Teste (wat_pot_test)
wat_pot_test <- wat_pot[-ind_train,]
paste("O Conjunto de Teste tem", nrow(wat_pot_test),"observações.")
```

```
## [1] "O Conjunto de Teste tem 604 observações."
```

```
summary(wat_pot_test)
```

```
##        ph            Hardness          Solids       Chloramines    
##  Min.   : 2.129   Min.   : 94.91   Min.   : 1352   Min.   : 1.920  
##  1st Qu.: 6.033   1st Qu.:175.82   1st Qu.:15324   1st Qu.: 6.149  
##  Median : 6.918   Median :196.01   Median :20249   Median : 7.216  
##  Mean   : 6.997   Mean   :195.71   Mean   :21557   Mean   : 7.171  
##  3rd Qu.: 7.995   3rd Qu.:217.37   3rd Qu.:26808   3rd Qu.: 8.173  
##  Max.   :11.898   Max.   :317.34   Max.   :55335   Max.   :13.127  
##     Sulfate       Conductivity   Organic_carbon   Trihalomethanes  
##  Min.   :182.4   Min.   :253.0   Min.   : 4.467   Min.   : 23.79  
##  1st Qu.:307.7   1st Qu.:369.6   1st Qu.:12.226   1st Qu.: 56.34  
##  Median :334.1   Median :425.7   Median :14.198   Median : 66.58  
##  Mean   :334.1   Mean   :431.3   Mean   :14.305   Mean   : 66.77  
##  3rd Qu.:362.4   3rd Qu.:491.8   3rd Qu.:16.537   3rd Qu.: 77.53  
##  Max.   :458.4   Max.   :666.7   Max.   :24.755   Max.   :113.05
```

```
##    Turbidity     Potability
##  Min.   :1.813   No :353
##  1st Qu.:3.426   Yes:251
##  Median :3.994
##  Mean   :3.975
##  3rd Qu.:4.524
##  Max.   :6.495

# Tabela de frequências relativas da variável Potability - conjunto teste
prop.table(table(wat_pot_test$Potability))

##
##        No       Yes
## 0.5844371 0.4155629
```

1.5) [1 valor] Completação das frases seguintes em comentário do script:

```
# Os dados wat_pot_train são compostos por ________ observações e por
_______ variáveis métricas; neste conjunto, a média da variável ph é
________ enquanto a mediana de Sulfate é ________.

# 1 - 1407      nrow()
# 2 - 9         ncol(wat_pot_train[, sapply(wat_pot_train, is.numeric)])
# 3 - 7.12434   mean(wat_pot_train$ph)`
# 4 - 331.8346  median(wat_pot_train$Sulfate)
```

Os dados wat_pot_train são compostos por **1407** observações e por **9** variáveis métricas; neste conjunto, a média da variável ph é **7.1243401** enquanto a mediana de Sulfate é **331.8346323**.

2. Aprendizagem, sobre a amostra de treino, do modelo de Regressão Logística, baseado nos preditores Solids and Turbidity, para prever Potability e avaliação do seu desempenho.

2.1) [2 valores] Determine a associação entre os preditores e o alvo.

```
# Associação entre o target categórico e os preditores métricos Solids e
Turbidity
eta<- matrix(0,2,1)
rownames(eta)<-colnames(wat_pot_train[,c(3,9)])

# Função Eta_ mede a associação (preditores, target)
Eta_<-function(y,x){
  freqk<-as.vector(table(x))
  l<-nlevels(x)
  m<-rep(NA, l)
  qual<-as.numeric(x)
  for (k in 1:l) {m[k]<-mean(y[qual == k])}
  return(sqrt(sum(freqk*(m-mean(y))^2)/sum((y-mean(y))^2)))
}

eta[1] <- Eta_(wat_pot_train$Solids,wat_pot_train$Potability)
eta[2] <- Eta_(wat_pot_train$Turbidity,wat_pot_train$Potability)
eta
```

```
##                 [,1]
## Solids    0.03923864
## Turbidity 0.03151408
```

2.2) [2.5 valores] Obtenção do modelo e das correspondentes estimativas de Potability sobre amostra de teste.

```
# Modelo da Regressão Logística
rlog.Potability <- glm(Potability ~ Solids+Turbidity,
                       data=wat_pot_train,
                       family=binomial)
summary(rlog.Potability)
```

```
##
## Call:
## glm(formula = Potability ~ Solids + Turbidity, family = binomial,
##     data = wat_pot_train)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -9.509e-01  3.180e-01  -2.990  0.00279 **
## Solids       9.309e-06  6.313e-06   1.475  0.14030
## Turbidity    8.334e-02  7.023e-02   1.187  0.23538
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1891.6  on 1406  degrees of freedom
## Residual deviance: 1888.0  on 1404  degrees of freedom
## AIC: 1894
##
## Number of Fisher Scoring iterations: 4
```

```
# Obtendo as estimativas de Potability sobre amostra de teste
rlog.pred_test<-round(predict(rlog.Potability, wat_pot_test,
type="response"))
```

2.3) [2.5 valores] Apresentação da Confusion matrix sobre amostra de teste e da métrica accuracy correspondente.

```
# Confusion Matrix sobre amostra de teste
(confusion_matrix <- table(wat_pot_test$Potability,
factor(rlog.pred_test, levels=c(0,1), labels=c("No", "Yes"))))
```

```
##
##        No Yes
##   No  353   0
##   Yes 251   0
```

```
# Cálculo da Accuracy
(accuracy <- sum(diag(confusion_matrix))/sum(confusion_matrix))
```

```
## [1] 0.5844371
```

2.4) [1 valor] Completação das frases seguintes em comentário do script:

```
# A Residual Deviance do modelo de Regressão Logística é ________; no
conjunto teste, a proporção de observações de água não potável
corretamente classificadas é _________; a probabilidade da última
observação do conjunto de teste se referir a água potável, estimada pelo
modelo, é ________.

# 1 - 1888
# 2 - 353
# 3 - 0
```

A Residual Deviance do modelo de Regressão Logística é **1888**; no conjunto teste, a proporção de observações de água não potável corretamente classificadas é **353**; a probabilidade da última observação do conjunto de teste se referir a água potável, estimada pelo modelo, é **0**.

3. Aprendizagem, sobre a amostra de treino, de uma Árvore de Regressão para prever Conductivity e avaliação do seu desempenho

3.1) [2.5 valores] Obtenção do modelo, sobre a amostra de treino, sem utilizar poda, considerando os preditores métricos e mindev=0.006; summary da árvore correspondente.

```
# Modelo em Árvore (preditores métricos e mindev=0.006)
rtree_large <-tree(Conductivity~ ph+Hardness+Solids+Chloramines+Sulfate+
Organic_carbon+Trihalomethanes+Turbidity,
                   data=wat_pot_train,
                   control=tree.control(nrow(wat_pot_train),
                                        mincut = 1,
                                        minsize = 2,
                                        mindev = 0.006),
                   split = "deviance")

# Summary do Modelo em Árvore
summary(rtree_large)

##
## Regression tree:
## tree(formula = Conductivity ~ ph + Hardness + Solids + Chloramines +
##     Sulfate + Organic_carbon + Trihalomethanes + Turbidity, data = wat_pot_train,
##     control = tree.control(nrow(wat_pot_train), mincut = 1, minsize = 2,
##         mindev = 0.006), split = "deviance")
## Variables actually used in tree construction:
## [1] "Solids"
## Number of terminal nodes:  3
## Residual mean deviance:  6275 = 8811000 / 1404
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -234.500  -58.470   -3.235    0.000   53.910  330.200



rtree_large

## node), split, n, deviance, yval
##       * denotes terminal node
##
## 1) root 1407 8941000 424.5
##   2) Solids < 9423.72 61  369500 393.8 *
##   3) Solids > 9423.72 1346 8511000 425.9
##     6) Solids < 13642.9 171 1044000 444.8 *
##     7) Solids > 13642.9 1175 7397000 423.1 *
```

3.2) [2.5 valores] Estimação de Conductivity sobre amostra de teste, a partir da árvore obtida, e apresentação das estimativas correspondentes às 10 primeiras observações desta amostra.

```
# Estimação de Conductivity sobre amostra de teste
pred.rtree <- predict(rtree_large, wat_pot_test)


# Estimativas correspondentes às 10 primeiras observações da amostra de
teste
head(pred.rtree, 10) # OU pred.rtree[1:10]
```

```
##        2        3       22       24       25       27       28
30
## 423.1307 423.1307 423.1307 444.7780 444.7780 423.1307 444.7780
444.7780
##       32       39
## 423.1307 423.1307
```

3.3) [2 valores] Apresentação do valor da métrica RMSE (Square Root of Mean Squared Error) associado ao modelo aplicado sobre a amostra de teste.

```
# Cálculo do RMSE (Square Root of Mean Squared Error)
residuals <- wat_pot_test$Conductivity - pred.rtree
(rmse <- sqrt(mean(residuals^2)))
```

```
## [1] 83.67876
```

3.4) [1 valor] Completação das frases seguintes em comentário do script:

```
# A Árvore de Regressão é constituída por  ________ nós folha;  a
Residual Deviance associada ao modelo sobre o conjunto teste é
_____________; o erro quadrático de previsão, relativo a Conductivity,
para a primeira observação do conjunto teste é _____________
residuals[1]^2.

# 1 - 3
# 2 - 8811000
# 3 - 611.071
```

A Árvore de Regressão é constituída por **3** nós folha; a Residual Deviance associada ao modelo sobre o conjunto teste é **8811000**; o erro quadrático de previsão, relativo a Conductivity, para a primeira observação do conjunto teste é **611.071**