MAS: Trabalho de Grupo

nome do representante do grupo

21 de março, 2023

André Filipe Gomes Silvestre

O Trabalho de Grupo de *Métodos de Aprendizagem Supervisionada* refere-se à análise do data set "Consumo.Jovens.csv".

Neste data set incluem-se 1523 registos e 28 atributos listados a seguir:

q0: País de residência
q1: Sexo
q2: Idade
q3: Situação estudantil
q10: Compra produtos de marca? (1-Sim; 2-Não)
q12b_a: Compra em centros comerciais? (1-Sim; 0-Não)
q12b_b: Compra em super/hipermercados? (1-Sim; 0-Não)
q12b_c: Compra no comércio local? (1-Sim; 0-Não)
q13a: Fidelidade a marcas? (1-Sim; 0-Não)
q13b: Fidelidade a lojas? (1-Sim; 0-Não)

Variáveis q14 na Escala 1-Nada Importante, 2, 3, 4, 5-Extremamente importante)
q14a: Preço
q14b: Necessidade do produto
q14c: Conveniência da localização da loja
q14d: Qualidade do produto
q14e: Imagem do produto
q14f: Imagem da loja
q14g: Características do produto
q14h: Promoção especial
q14i: Imagem da marca
q14j: Publicidade

Variáveis q19 na Escala 1-Discordo Completamente, 2, 3, 4, 5-Concordo Completamente)
q19_1: Alguns dos feitos + importantes da vida incluem adquirir bens materiais
q19_2: Não dou importância à quantidade de bens materiais
q19_3: Gosto de ter coisas para impressionar as pessoas
q19_4: Geralmente compro apenas aquilo de que preciso
q19_5: Gosto de gastar dinheiro em coisas que não são necessárias
q19_6: Comprar coisas dá-me imenso prazer

q19_7: Tenho todas as coisas de que preciso para ser feliz
q19_8: Seria mais feliz se tivesse dinheiro para comprar mais coisas

Notas:

1. Efetuar todos os Save com "Save with encoding UTF-8" de modo a manter palavras acentuadas e caracteres especiais**
2. A cotação está anexa a cada pergunta
3. **OS ALUNOS QUE NÃO SUBMETEREM PDF NO MOODLE TERÃO UMA PENALIZAÇÃO DE 1 VALOR; SE, O FICHEIRO ALTERNATIVO QUE SUBMETEREM (VIA EMAIL) REPORTAR ERROS NA COMPILAÇÃO, TERÃO UMA PENALIZAÇÃO ADICIONAL DE 1 VALOR**

```
# 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 “Consumo.Jovens.csv” e análise preliminar dos mesmos

1.1) [1 valor] Leitura dos dados; apresentação de dimensão e estrutura dos dados; verificação do número de casos com dados em falta (para todos os atributos); sumário dos dados completos (depois de eliminação dos casos/linhas com dados omissos)

```
# Leitura dos dados (Nota: verifique sep no ficheiro de origem)
CJ <- read.csv("Consumo.Jovens.csv", header=TRUE, dec=".",na.strings="",
sep=";",stringsAsFactors = TRUE)
CJ_original <- CJ

# Apresentação de dimensão e estrutura dos dados.
dim(CJ)

## [1] 1523   28

print(paste("Nº de Observações:", nrow(CJ)))

## [1] "Nº de Observações: 1523"

print(paste("Nº de Colunas:", ncol(CJ)))

## [1] "Nº de Colunas: 28"

str(CJ)

## 'data.frame':    1523 obs. of  28 variables:
##  $ q0    : Factor w/ 6 levels "Alemanha","China",..: 6 6 6 6 6 6 6 6 6
6 ...
##  $ q1    : Factor w/ 2 levels "Feminino","Masculino": 1 2 1 2 1 2 1 2
2 2 ...
##  $ q2    : int  19 19 19 20 21 19 20 21 20 22 ...
##  $ q3    : Factor w/ 3 levels "Estudante-trabalhador",..: 2 2 2 2 2 2
2 2 2 2 ...
##  $ q10   : Factor w/ 2 levels "Nao","Sim": 2 2 1 2 2 2 2 2 2 2 ...
##  $ q12b_a: int  0 1 0 1 1 1 1 0 1 1 ...
##  $ q12b_b: int  1 0 0 1 0 1 1 0 1 1 ...
##  $ q12b_c: int  0 1 1 0 0 0 0 1 0 0 ...
##  $ q13a  : int  0 1 0 1 0 0 1 1 0 1 ...
##  $ q13b  : int  0 1 1 0 0 0 0 0 1 NA ...
##  $ q14a  : int  5 4 5 3 5 3 3 5 3 3 ...
##  $ q14b  : int  3 3 5 5 3 4 5 3 5 3 ...
##  $ q14c  : int  NA 2 3 2 2 1 2 1 2 2 ...
##  $ q14d  : int  4 5 5 4 5 3 5 4 3 4 ...
##  $ q14e  : int  NA 3 3 2 4 2 3 2 3 3 ...
##  $ q14f  : int  NA 4 4 2 3 1 3 1 2 2 ...
##  $ q14g  : int  NA 5 5 3 4 4 4 3 4 2 ...
##  $ q14h  : int  NA 2 2 2 2 3 4 2 3 2 ...
```

```
##  $ q14i  : int  1 3 3 3 3 2 3 2 3 2 ...
##  $ q14j  : int  2 3 3 2 2 2 4 1 2 2 ...
##  $ q19_1 : int  NA 2 4 NA 3 1 NA 2 2 2 ...
##  $ q19_2 : int  NA 4 4 NA 5 4 NA 4 2 4 ...
##  $ q19_3 : int  NA 1 1 NA 2 1 NA 2 3 3 ...
##  $ q19_4 : int  NA 3 5 NA 3 4 NA 5 3 3 ...
##  $ q19_5 : int  NA 3 1 NA 3 1 NA 2 3 2 ...
##  $ q19_6 : int  NA 3 3 NA 3 3 NA 3 3 1 ...
##  $ q19_7 : int  NA 4 3 NA 5 2 NA 3 5 3 ...
##  $ q19_8 : int  NA 3 5 NA 3 4 NA 4 4 4 ...
```

```
# Verificação do número de casos com dados em falta (para todos os atributos)
colSums(is.na(CJ)) # NAs por atributo
```

```
##     q0     q1     q2     q3    q10 q12b_a q12b_b q12b_c   q13a   q13b   q14a
##      0      5      0     21     44      4      5      7     60     70     13
##   q14b   q14c   q14d   q14e   q14f   q14g   q14h   q14i   q14j  q19_1  q19_2
##     19     24     14     20     23     23     21     19     23     46     48
##  q19_3  q19_4  q19_5  q19_6  q19_7  q19_8
##     46     44     52     47     52     53
```

```
paste("No total, existem", nrow(is.na(CJ)), "NAs.")
```

```
## [1] "No total, existem 1523 NAs."
```

```
# Eliminação dos casos/linhas com dados omissos
CJ<-na.omit(CJ)
```

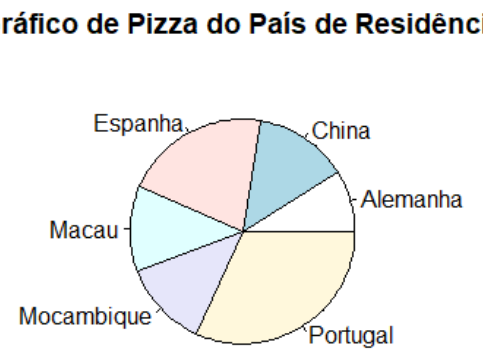
```
# Sumário dos dados completos
summary(CJ)
```

```
##           q0              q1           q2
##  Alemanha  :113   Feminino :727   Min.   :17.00
##  China     :170   Masculino:538   1st Qu.:20.00
##  Espanha   :266                   Median :21.00
##  Macau     :156                   Mean   :21.19
##  Mocambique:158                   3rd Qu.:23.00
##  Portugal  :402                   Max.   :25.00
##                          q3        q10          q12b_a           q12b_b
##  Estudante-trabalhador    : 116   Nao:556   Min.   :0.0000   Min.   :0.0000
##  Estudante a tempo inteiro:1044   Sim:709   1st Qu.:0.0000   1st Qu.:0.0000
##  Outra                    : 105             Median :1.0000   Median :0.0000
##                                             Mean   :0.5209   Mean   :0.3621
##                                             3rd Qu.:1.0000   3rd Qu.:1.0000
##                                             Max.   :1.0000   Max.   :1.0000
##      q12b_c            q13a             q13b             q14a
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :1.000
##  1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:3.000
##  Median :0.0000   Median :0.0000   Median :0.0000   Median :4.000
##  Mean   :0.4791   Mean   :0.4198   Mean   :0.4806   Mean   :3.696
##  3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:1.0000   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :5.000
```

```
##       q14b            q14c            q14d            q14e      
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:3.000   1st Qu.:2.000   1st Qu.:4.000   1st Qu.:2.000  
##  Median :4.000   Median :3.000   Median :4.000   Median :3.000  
##  Mean   :3.704   Mean   :2.553   Mean   :4.029   Mean   :2.952  
##  3rd Qu.:4.000   3rd Qu.:3.000   3rd Qu.:5.000   3rd Qu.:4.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000  
##       q14f            q14g            q14h            q14i      
##  Min.   :1.000   Min.   :1.000   Min.   :1.000   Min.   :1.000  
##  1st Qu.:2.000   1st Qu.:3.000   1st Qu.:2.000   1st Qu.:2.000  
##  Median :2.000   Median :4.000   Median :3.000   Median :3.000  
##  Mean   :2.544   Mean   :3.496   Mean   :2.651   Mean   :2.675  
##  3rd Qu.:3.000   3rd Qu.:4.000   3rd Qu.:3.000   3rd Qu.:3.000  
##  Max.   :5.000   Max.   :5.000   Max.   :5.000   Max.   :5.000  
##       q14j           q19_1           q19_2           q19_3           q19_4      
##  Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.000   Min.   :1.0  
##  1st Qu.:2.000   1st Qu.:2.00   1st Qu.:3.000   1st Qu.:1.000   1st Qu.:2.0  
##  Median :2.000   Median :3.00   Median :4.000   Median :2.000   Median :3.0  
##  Mean   :2.192   Mean   :2.91   Mean   :3.404   Mean   :2.436   Mean   :3.3  
##  3rd Qu.:3.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:3.000   3rd Qu.:4.0  
##  Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.000   Max.   :5.0  
##      q19_5           q19_6           q19_7           q19_8      
##  Min.   :1.000   Min.   :1.00   Min.   :1.000   Min.   :1.00  
##  1st Qu.:2.000   1st Qu.:3.00   1st Qu.:2.000   1st Qu.:3.00  
##  Median :2.000   Median :3.00   Median :3.000   Median :3.00  
##  Mean   :2.387   Mean   :3.27   Mean   :2.947   Mean   :3.24  
##  3rd Qu.:3.000   3rd Qu.:4.00   3rd Qu.:4.000   3rd Qu.:4.00  
##  Max.   :5.000   Max.   :5.00   Max.   :5.000   Max.   :5.00
```

1.2) [1.5 valores] Breve análise descritiva de q0, q1, q2 e q3.

```
#q0: País de residência - Tabela de Frequências Absolutas e Relativas e Gráfico de Pizza
table(CJ[,1])
##   Alemanha      China    Espanha      Macau Mocambique   Portugal 
##        113        170        266        156        158        402 

prop.table(table(CJ[,1]))
##   Alemanha      China    Espanha      Macau Mocambique   Portugal 
## 0.08932806 0.13438735 0.21027668 0.12332016 0.12490119 0.31778656 

pie(table(CJ[,1]), main = "Gráfico de Pizza do País de Residência")
```



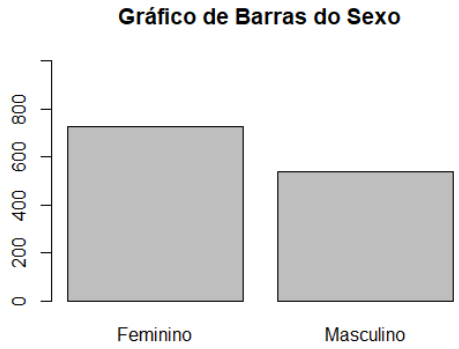
```r
#q1: Sexo - Tabela de Frequências Absolutas e Relativas e BarPlot
table(CJ[,2])
```

```
##
##  Feminino Masculino
##       727       538
```

```r
prop.table(table(CJ[,2]))
```

```
##
##  Feminino Masculino
## 0.5747036 0.4252964
```

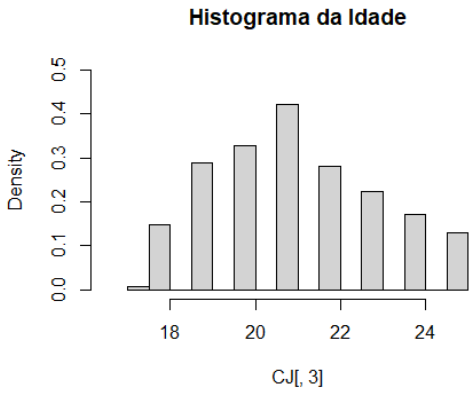
```r
barplot(table(CJ[,2]), ylim=c(0,1000), main = "Gráfico de Barras do Sexo")
```




```r
#q2: Idade - Métricas para variaveis quantitativas e Histograma
describe(CJ[,3])
```

```
##    vars    n  mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 1265 21.19 1.96     21   21.13 1.48  17  25     8 0.23    -0.77 0.06
```

```r
hist(CJ[,3], freq = F, ylim = c(0,.5), xlim = c(16.5, 25.5), main = "Histograma da Idade")
```



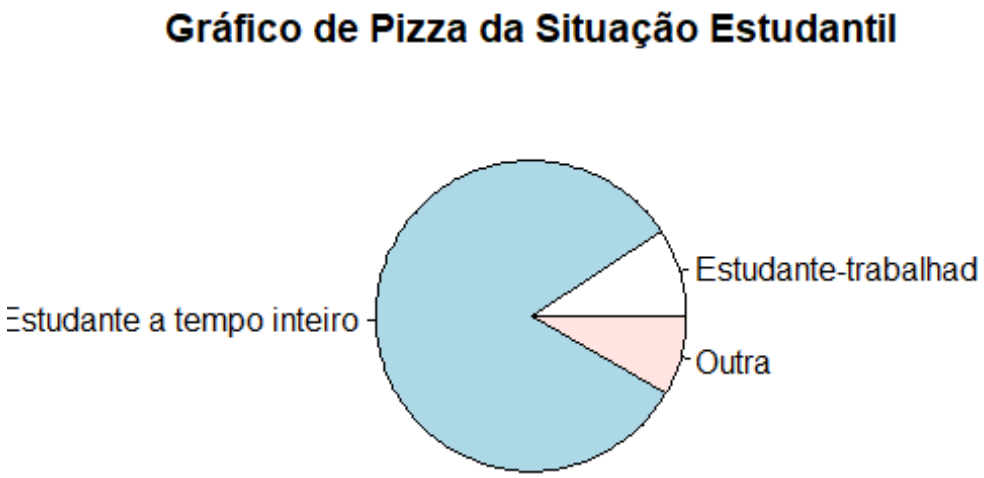
```r
#q3: Situação Estudantil - Tabela de Frequências Absolutas e Relativas e BarPlot
table(CJ[,4])
```

```
##
##     Estudante-trabalhador Estudante a tempo inteiro
Outra
##                       116                      1044
105
```

```r
prop.table(table(CJ[,4]))
```

```
##
##     Estudante-trabalhador Estudante a tempo inteiro
Outra
##                0.09169960               0.82529644
0.08300395
```

```r
pie(table(CJ[,4]), main = "Gráfico de Pizza da Situação Estudantil")
```



1.3) [1.5 valores] Cálculo (e apresentação) de medidas de associação entre as variáveis:

a) q14a...q14j;
b) q0 e as variáveis q19_1...q19_8;
c) q10 e q1

```
#a) q14a…q14j - Correlação de Pearson
# Medir a correlação dos preditores métricos
(corr <- round(cor(CJ[, 11:20], method = "pearson"), 2))
```

```
##       q14a  q14b q14c  q14d  q14e  q14f  q14g  q14h  q14i  q14j
## q14a  1.00  0.08 0.05 -0.05 -0.07 -0.12 -0.01  0.15 -0.11 -0.06
## q14b  0.08  1.00 0.07  0.15  0.00 -0.01  0.22  0.15  0.01 -0.09
## q14c  0.05  0.07 1.00  0.10  0.13  0.20  0.05  0.11  0.12  0.19
## q14d -0.05  0.15 0.10  1.00  0.24  0.20  0.22 -0.04  0.19  0.10
## q14e -0.07  0.00 0.13  0.24  1.00  0.50  0.27  0.13  0.51  0.27
## q14f -0.12 -0.01 0.20  0.20  0.50  1.00  0.27  0.14  0.40  0.32
## q14g -0.01  0.22 0.05  0.22  0.27  0.27  1.00  0.20  0.18  0.08
## q14h  0.15  0.15 0.11 -0.04  0.13  0.14  0.20  1.00  0.20  0.22
## q14i -0.11  0.01 0.12  0.19  0.51  0.40  0.18  0.20  1.00  0.44
## q14j -0.06 -0.09 0.19  0.10  0.27  0.32  0.08  0.22  0.44  1.00
```

```
#b) q0 e as variáveis q19_1…q19_8 - ETA
# Associação entre o target categórico e os preditores métricos
eta<- matrix(0,8,1)
rownames(eta)<-colnames(CJ[,21:28])

for (i in 21:28) {
  anova_ <- aov(CJ[,i] ~ q0, CJ)
  eta[i-20]<-sqrt(etaSquared(anova_ )[,1])
}
eta
```

```
##           [,1]
## q19_1 0.3405109
## q19_2 0.2373190
## q19_3 0.6471752
## q19_4 0.2867009
## q19_5 0.3524935
## q19_6 0.2231179
## q19_7 0.4309111
## q19_8 0.2231995
```

```
# c) q10 e q1 - V de Cramer
# Medir a associação entre preditores qualitativos e o target
cramersV(CJ$q1,CJ$q10)
```

```
## [1] 0.07398348
```

1.4) [1 valor] Divisão dos dados em amostra de treino (60%)- CJ.train - e de teste (40%) – CJ.test - usando set.seed(444);apresentação de tabela de frequências relativas de q1 em cada amostra

```
# Definir o set.seed para permitir reprodutibilidade dos resultados
set.seed(444)

# Divisão em Conjunto Treino/Teste
ind_train <- sample(nrow(CJ),0.6*nrow(CJ))

# Conjunto Treino (CJ.train)
CJ.train <- CJ[ind_train,]
paste("O Conjunto de Treino tem", nrow(CJ.train),"observações.")

## [1] "O Conjunto de Treino tem 759 observações."

# Tabela de frequências relativas da variável q1 - Conjunto de Treino
prop.table(table(CJ.train$q1))

##
##  Feminino Masculino
## 0.5770751 0.4229249

# Conjunto Teste (CJ.test)
CJ.test <- CJ[-ind_train,]
paste("O Conjunto de Teste tem", nrow(CJ.test),"observações.")

## [1] "O Conjunto de Teste tem 506 observações."

# Tabela de frequências relativas da variável q1 - Conjunto de Teste
prop.table(table(CJ.test$q1))

##
##  Feminino Masculino
## 0.5711462 0.4288538
```

1.5) [1 valor] Completação das frases seguintes:

Inicialmente, o número de casos omissos na variável q1 era **`r sum(is.na(CJ_original$q1))`**. No conjunto de dados em análise (depois de eliminar os registos com observações omissas) o número de estudantes trabalhadores é igual a **116**. A correlação mais elevada entre o pares de variáveis q14 tem o valor **0.51**. A correlação maior entre a variável q0 e as variáveis q19_ regista-se para a variável **q19_3**

```
# 1 - sum(is.na(CJ_original$q1))
# 2 - as.vector(table(CJ[,4])[1])
# 3 - max(abs(corr[corr!=1]))
# 4 - q19_3
```

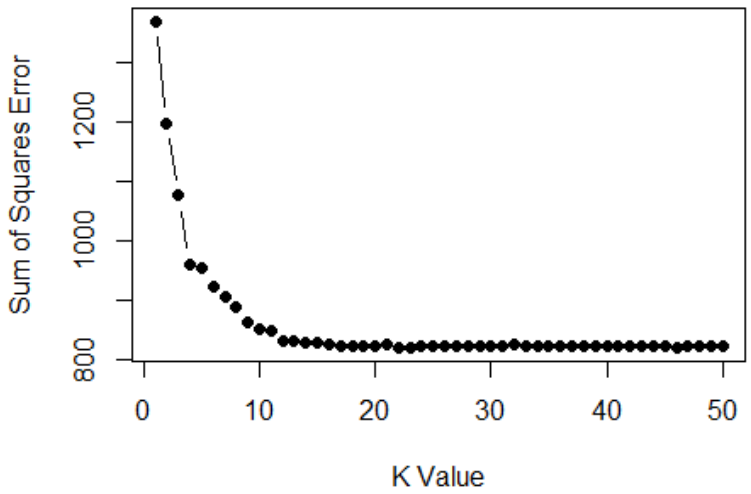
2. Regressão: utilização do K-Nearest Neighbour para prever q19_8 com base nas variáveis q12b_a , q12b_b, q12b_c, q13a e q13b.

2.1) [2 valores] Aprendizagem sobre CJ.train[,c(6:10)] e considerando y=y=CJ.train$q19_8 recorrendo a one-hold-out validation; determinação de um "melhor" valor de K atendendo ao Sum of Squares Error

Fórmula do Sum of Squares Error (SSE)

$$\sum(y_i - \hat{y}_i)^2$$

```
# Modelo de KNN com o target "q19_8" e preditor "q12b_a", "q12b_b",
"q12b_c", "q13a" e "q13b"
# Seleção do "melhor" k de acordo com o SSE

k.sse<-matrix(NA,50,2)

for (i in 1:50){
  knn.CJ <- knn.reg(CJ.train[,c(6:10)], y=CJ.train$q19_8, k=i)
  k.sse[i,1]<-i
  k.sse[i,2] <- sse(knn.CJ$pred, CJ.train$q19_8)
}

# Representação Gráfica da SSE
plot(k.sse[,2], type = "b", pch = 19, xlab = "K Value", ylab = "Sum of
Squares Error")
```



```r
# Ordenar o SSE
k.sse<-k.sse[order(k.sse[,2],decreasing=FALSE),]

# "Melhor" k segundo o SSE
best_k <- k.sse[1,1]
paste("O 'melhor' K utilizando esta metedologia é", best_k)
```

```
## [1] "O 'melhor' K utilizando esta metedologia é 22"
```

2.2) [2 valores] Considerando o "melhor" valor de K (v. 2.1), obtenção de estimativas do alvo e listagem dos 6 primeiros valores estimados nos conjuntos CJ.train e CJ.test

```r
# Estimativas sobre CJ.train
knn.CJ_train <- knn.reg(CJ.train[,c(6:10)], y = CJ.train$q19_8, k =
best_k)
head(knn.CJ_train$pred, 6)
```

```
## [1] 3.181818 3.000000 3.136364 3.090909 3.500000 3.181818
```

```r
# Estimativas sobre CJ.test
knn.CJ_test <- knn.reg(CJ.train[,c(6:10)], CJ.test[,c(6:10)], y =
CJ.train$q19_8, k = best_k)
head(knn.CJ_test$pred, 6)
```

```
## [1] 3.227273 3.045455 3.227273 3.409091 3.090909 3.136364
```

2.3) [2 valores] Determinação de Sum of Squares Error e de Root Mean Squared Error (RMSE) correspondentes às estimativas obtidas pelo KNN em 2.2) para as amostras CJ.train e CJ.test

Fórmula do Sum of Squares error (SSE)

$$\sum(y_i - \hat{y}_i)^2$$

Fórmula do Root Mean Squared Error (RMSE)

$$\sqrt{\frac{\sum(y_i - \hat{y}_i)^2}{n}}$$

```
# Métricas sobre CJ.train
train_sse <- sse(knn.CJ_train$pred, CJ.train$q19_8)
train_rmse <- rmse(knn.CJ_train$pred, CJ.train$q19_8)
cat("SSE para CJ.train:", round(train_sse,2), "\n")

## SSE para CJ.train: 820.72

cat("RMSE para CJ.train:", train_rmse, "\n")

## RMSE para CJ.train: 1.039862

# Métricas sobre CJ.test
test_sse <- sse(knn.CJ_test$pred, CJ.test$q19_8)
test_rmse <- rmse(knn.CJ_test$pred, CJ.test$q19_8)
cat("SSE para CJ.test:", round(test_sse,2), "\n")

## SSE para CJ.test: 570.33

cat("RMSE para CJ.test:", test_rmse, "\n")

## RMSE para CJ.test: 1.061664
```

2.4) [1 valor] Completação das frases seguintes:

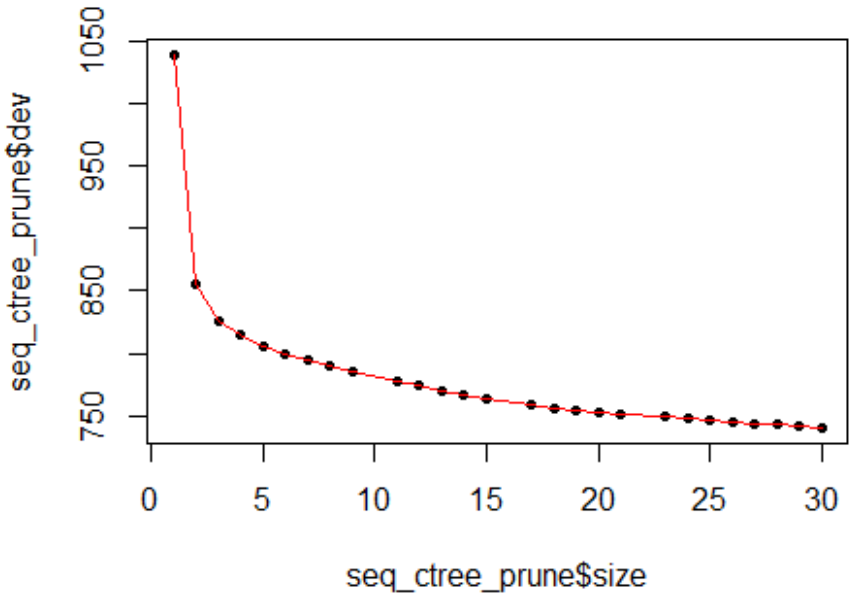
O "melhor" valor de K, para K-NN, obtido segundo validação hold-one-out sobre a amostra de treino é **22**; o valor estimado do alvo para a 1ª observação do conjunto de teste é **3.2272727**; neste conjunto (teste) obtém-se um RMSE de **1.06** e um SSE de **570.33**.

```
# 1 - k.sse[1,1]
# 2 - knn.CJ_test$pred[1]
# 3 - test_rmse
# 4 - test_sse
```

3. Classificação: utilização de uma Árvore para prever q10 (Compra ou não compra produtos de marca) considerando 4 preditores: q12b_a, q13a, q14e e q14i.

3.1) [2 valores] Construção de uma Árvore de classificação sobre CJ.train efetuando a sua poda de modo a fixar 15 nós folha (para prever q10 com base nos preditores q12b_a, q13a, q14e e q14i)

```
# ======================== Árvore de Classificação
========================

# Nomes das Variáveis
colnames(CJ.train)

##  [1] "q0"     "q1"     "q2"     "q3"     "q10"    "q12b_a" "q12b_b"
"q12b_c"
##  [9] "q13a"   "q13b"   "q14a"   "q14b"   "q14c"   "q14d"   "q14e"
"q14f"
## [17] "q14g"   "q14h"   "q14i"   "q14j"   "q19_1"  "q19_2"  "q19_3"
"q19_4"
## [25] "q19_5"  "q19_6"  "q19_7"  "q19_8"

# Começamos por criar uma Árvore Grande
ctree_large <-tree(q10~q12b_a+q13a+q14e+q14i, data = CJ.train,
                   control=tree.control(nrow(CJ.train),
                                        mincut = 1, minsize = 2,
                                        mindev = 0.001),
                   split = "deviance")

# Resultados da Árvore
summary(ctree_large)

##
## Classification tree:
## tree(formula = q10 ~ q12b_a + q13a + q14e + q14i, data = CJ.train,
##     control = tree.control(nrow(CJ.train), mincut = 1, minsize = 2,
##         mindev = 0.001), split = "deviance")
## Number of terminal nodes:  30
## Residual mean deviance:  1.015 = 740.1 / 729
## Misclassification error rate: 0.2437 = 185 / 759

### Custo-Complexidade - Poda da Árvore

# Gráfico de Custo/Complexidade
seq_ctree_prune <- prune.tree(ctree_large)
plot(seq_ctree_prune$size,seq_ctree_prune$dev,pch =20)
lines(seq_ctree_prune$size,seq_ctree_prune$dev, col = "red")
```

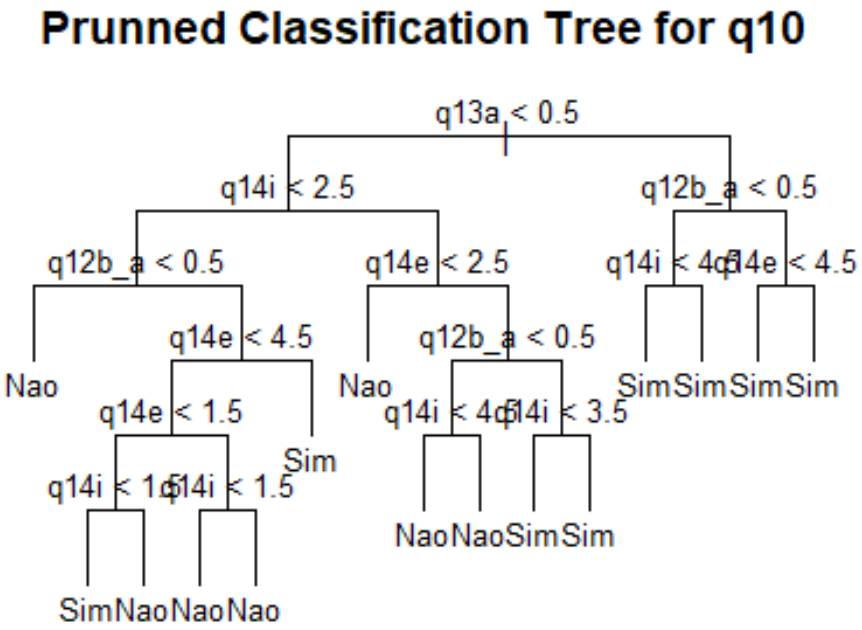



```
# Utilizando o "melhor" tamanho de 15 como referido no enunciado,
obtermos a seguinte Árvore Podada
ctree.CJ<-prune.tree(ctree_large, best=15)
```

3.2) [2 valores] Representações da Árvore de Classificação: a) Lista indentada; b) Gráfico da Árvore

```
# a) Representações da Árvore de Classificação - Lista indentada
ctree.CJ

## node), split, n, deviance, yval, (yprob)
##       * denotes terminal node
##
##  1) root 759 1040.000 Sim ( 0.43610 0.56390 )
##    2) q13a < 0.5 450  592.500 Nao ( 0.63111 0.36889 )
##      4) q14i < 2.5 246  279.900 Nao ( 0.74390 0.25610 )
##        8) q12b_a < 0.5 133  122.500 Nao ( 0.82707 0.17293 ) *
##        9) q12b_a > 0.5 113  146.900 Nao ( 0.64602 0.35398 )
##         18) q14e < 4.5 108  137.500 Nao ( 0.66667 0.33333 )
##           36) q14e < 1.5 10   13.860 Nao ( 0.50000 0.50000 )
##             72) q14i < 1.5 8   10.590 Sim ( 0.37500 0.62500 ) *
##             73) q14i > 1.5 2    0.000 Nao ( 1.00000 0.00000 ) *
##           37) q14e > 1.5 98  122.300 Nao ( 0.68367 0.31633 )
##             74) q14i < 1.5 20   13.000 Nao ( 0.90000 0.10000 ) *
##             75) q14i > 1.5 78  102.900 Nao ( 0.62821 0.37179 ) *
##         19) q14e > 4.5 5    5.004 Sim ( 0.20000 0.80000 ) *
##      5) q14i > 2.5 204  282.800 Sim ( 0.49510 0.50490 )
##       10) q14e < 2.5 41   52.640 Nao ( 0.65854 0.34146 ) *
##       11) q14e > 2.5 163  224.600 Sim ( 0.45399 0.54601 )
##         22) q12b_a < 0.5 72   98.920 Nao ( 0.55556 0.44444 )
##           44) q14i < 4.5 69   95.290 Nao ( 0.53623 0.46377 ) *
##           45) q14i > 4.5 3    0.000 Nao ( 1.00000 0.00000 ) *
##         23) q12b_a > 0.5 91  120.300 Sim ( 0.37363 0.62637 )
##           46) q14i < 3.5 68   92.790 Sim ( 0.42647 0.57353 ) *
##           47) q14i > 3.5 23   24.080 Sim ( 0.21739 0.78261 ) *
##    3) q13a > 0.5 309  263.500 Sim ( 0.15210 0.84790 )
##      6) q12b_a < 0.5 117  128.800 Sim ( 0.23932 0.76068 )
##       12) q14i < 4.5 112  126.000 Sim ( 0.25000 0.75000 ) *
##       13) q14i > 4.5 5    0.000 Sim ( 0.00000 1.00000 ) *
##      7) q12b_a > 0.5 192  124.000 Sim ( 0.09896 0.90104 )
##       14) q14e < 4.5 171  119.300 Sim ( 0.11111 0.88889 ) *
##       15) q14e > 4.5 21    0.000 Sim ( 0.00000 1.00000 ) *
```

```r
# b) Representações da Árvore de Classificação - Gráfico da Árvore
plot(ctree.CJ, type="uniform")
text(ctree.CJ, pretty =0, cex=0.8)
title(main = "Prunned Classification Tree for q10")
```



3.3) [2 valores] Obtenção, sobre as amostras CJ.train e CJ.test, das "Matrizes de Confusão" e correspondentes medidas Accuracy associadas à Árvore de Classificação

Accuracy

$$\frac{TP + TN}{TP + FN + FP + TN}$$

```
# "Matriz de Confusão" sobre CJ.train
pred_ctree.CJ_train<-predict(ctree.CJ, CJ.train, type = "class")

confusion_mat_tree_train <- table(CJ.train$q10, pred_ctree.CJ_train)
confusion_mat_tree_train
```

```
##      pred_ctree.CJ_train
##       Nao Sim
##   Nao 246  85
##   Sim 100 328
```

```
# Accuracy sobre CJ.train
(accuracy.train <-
sum(diag(confusion_mat_tree_train))/sum(confusion_mat_tree_train))
```

```
## [1] 0.7562582
```

```
# "Matriz de Confusão" sobre CJ.test
pred_ctree.CJ_test<-predict(ctree.CJ , CJ.test, type = "class")

confusion_mat_tree_test <- table(CJ.test$q10, pred_ctree.CJ_test)
confusion_mat_tree_test
```

```
##      pred_ctree.CJ_test
##       Nao Sim
##   Nao 158  67
##   Sim  78 203
```

```
# Accuracy sobre CJ.test
(accuracy.test <-
sum(diag(confusion_mat_tree_test))/sum(confusion_mat_tree_test))
```

```
## [1] 0.7134387
```

3.4) [1 valor] Completação das frases seguintes:

A árvore obtida, classifica as observações do nó folha 73) na classe **Não**; o nó folha com o maior número de observações de treino é o nó **14)**; no conjunto de teste o número de observações corretamente classificadas nas classes "Não" e "Sim" é **158** e **203**. respetivamente.