EST171 - Aprendizado de Máquina
Departamento de Estatística
Universidade Federal de Minas Gerais

Lista 2

Henrique Aparecido Laureano Matheus Henrique Sales

Outubro de 2016

Sumário

Exercício I

Baixe o conjunto de dados `titanic.txt`. Cada observação deste banco é relativa a um passageiro do Titanic. As covariáveis indicam características destes passageiros; a variável resposta indica se o passageiro sobreviveu ou não ao naufrágio.

```
# <code r> ========================================================================= #
path <- "C:/Users/henri/Dropbox/Scripts/aprendizado de maquina/list 2/"

data <- read.table(paste0(path, "titanic.txt"))

summary(data)
# </code r> ======================================================================== #
```

```
  Class         Sex          Age       Survived
 1st :325   Female: 470   Adult:2092   No :1490
 2nd :285   Male  :1731   Child: 109   Yes: 711
 3rd :706
 Crew:885
```

Seu objetivo é criar classificadores para predizer a variável resposta com base nas covariáveis disponíveis. Para tanto, você deverá implementar os seguintes classificadores, assim como estimar seus riscos via conjunto de teste:

```
# <code r> ========================================================================= #
test.psg <- sample(size = nrow(data) * .2, x = 0:nrow(data))

train <- data[-test.psg, ] # nrow(train): 1761

test <- data[test.psg, ] # nrow(test): 440
# </code r> ======================================================================== #
```

- **Regressão Logística. Mostre os coeficientes estimados.**
- **Regressão Linear. Mostre os coeficientes estimados.**
- **Naive Bayes.**
- **Análise Discriminante Linear.**
- **Análise Discriminante Quadrática.**
- **KNN. Para isso você precisará transformar as covariáveis categóricas em numéricas. Você pode usar variáveis dummies.**

Responda ainda as seguintes perguntas:

- **Qual o melhor classificador segundo o risco estimado? Discuta.**
- **Para os classificadores baseados em estimativas de probabilidade, faça também as curvas ROC com o conjunto de teste. Faça também a tabela de confusão quando o corte usado é 0.5 e também quando o corte é aquele que maximiza sensibilidade mais especificidade. Comente.**

Regressão Logística

```
# <code r> ============================================================== #
reg.log <- glm(Survived ~ ., train, family = binomial)
# </code r> ============================================================= #
```

As características (covariáveis) são significativas?

```
# <code r> ============================================================== #
anova(reg.log, test = "Chisq")
# </code r> ============================================================= #
```

```
Analysis of Deviance Table

Model: binomial, link: logit

Response: Survived

Terms added sequentially (first to last)


      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                  1760     2188.6
Class  3  136.709      1757     2051.8 < 2.2e-16 ***
Sex    1  302.650      1756     1749.2 < 2.2e-16 ***
Age    1   14.619      1755     1734.6 0.0001316 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com as características sendo adicionadas sequencialmente, todas são estatísticamente significativas.

E quando incluímos a característica num modelo que contem as demais?

```
# <code r> ============================================================== #
car::Anova(reg.log)
# </code r> ============================================================= #
```

```
Analysis of Deviance Table (Type II tests)

Response: Survived
      LR Chisq Df Pr(>Chisq)
Class  102.029  3  < 2.2e-16 ***
Sex    295.156  1  < 2.2e-16 ***
Age     14.619  1  0.0001316 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ainda assim todas as caractetísticas são significativas.

Coeficientes estimados:

```
# <code r> ============================================================== #
cbind(Estimates = coef(reg.log), confint.default(reg.log))
# </code r> ============================================================= #
```

```
             Estimates      2.5 %     97.5 %
(Intercept)  1.9759944  1.6093164  2.3426723
Class2nd    -0.9312155 -1.3585515 -0.5038795
Class3rd    -1.8377111 -2.2219188 -1.4535034
ClassCrew   -0.7272476 -1.0722845 -0.3822108
SexMale     -2.4929589 -2.8047147 -2.1812031
AgeChild     1.0365795  0.5067773  1.5663817
```

Odds-ratios:

```
# <code r> ============================================================== #
exp(cbind(OR = coef(reg.log), confint.default(reg.log)))
# </code r> ============================================================= #
```

```
                   OR      2.5 %     97.5 %
(Intercept) 7.21378916 4.99939231 10.4090159
Class2nd    0.39407441 0.25703281  0.6041822
Class3rd    0.15918136 0.10840090  0.2337499
ClassCrew   0.48323720 0.34222580  0.6823512
SexMale     0.08266501 0.06052404  0.1129056
AgeChild    2.81955620 1.65993313  4.7892876
```

Regressão Linear

```
# <code r> ========================================================================== #
reg.lin <- lm(as.numeric(Survived) ~ ., train)
# </code r> ========================================================================= #
```

As características são significativas?

```
# <code r> ========================================================================== #
anova(reg.lin)
# </code r> ========================================================================= #
```

```
Analysis of Variance Table

Response: as.numeric(Survived)
            Df  Sum Sq Mean Sq F value    Pr(>F)
Class        3  31.041  10.347  64.529 < 2.2e-16 ***
Sex          1  63.831  63.831 398.088 < 2.2e-16 ***
Age          1   2.322   2.322  14.481 0.0001464 ***
Residuals 1755 281.404   0.160
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Com as características sendo adicionadas sequencialmente, todas são estatísticamente significativas.

E quando incluímos a característica num modelo que contem as demais?

```
# <code r> ========================================================================== #
car::Anova(reg.lin)
# </code r> ========================================================================= #
```

```
Anova Table (Type II tests)

Response: as.numeric(Survived)
           Sum Sq   Df F value    Pr(>F)
Class      17.096    3  35.541 < 2.2e-16 ***
Sex        61.478    1 383.410 < 2.2e-16 ***
Age         2.322    1  14.481 0.0001464 ***
Residuals 281.404 1755
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Ainda assim todas as caractetísticas são significativas.

Coeficientes estimados:

```r
# <code r> ========================================================================= #
cbind(Estimates = coef(reg.lin), confint(reg.lin))
# </code r> ========================================================================= #
```

```
             Estimates       2.5 %      97.5 %
(Intercept)  1.8747836  1.81888956  1.93067768
Class2nd    -0.1671427 -0.23797279 -0.09631268
Class3rd    -0.3046091 -0.36459061 -0.24462767
ClassCrew   -0.1523171 -0.21267124 -0.09196303
SexMale     -0.4974034 -0.54722583 -0.44758103
AgeChild     0.1682078  0.08151267  0.25490288
```

Naive Bayes

```r
# <code r> ========================================================================= #
library(e1071)

nb <- naiveBayes(Survived ~ ., train)

nb$tables
# </code r> ========================================================================= #
```

```
$Class
     Class
Y            1st        2nd        3rd       Crew
  No  0.08760331 0.11487603 0.35619835 0.44132231
  Yes 0.28675136 0.17241379 0.23956443 0.30127042

$Sex
     Sex
Y         Female       Male
  No  0.08760331 0.91239669
  Yes 0.49364791 0.50635209

$Age
     Age
Y          Adult      Child
  No  0.96115702 0.03884298
  Yes 0.91651543 0.08348457
```

Análise Discriminante Linear

```r
# <code r> ========================================================================== #
library(MASS)

dl <- lda(Survived ~ ., train)

dl$scaling
# </code r> ========================================================================== #
```

```
                LD1
Class2nd  -0.8247560
Class3rd  -1.5030759
ClassCrew -0.7516000
SexMale   -2.4544080
AgeChild   0.8300114
```

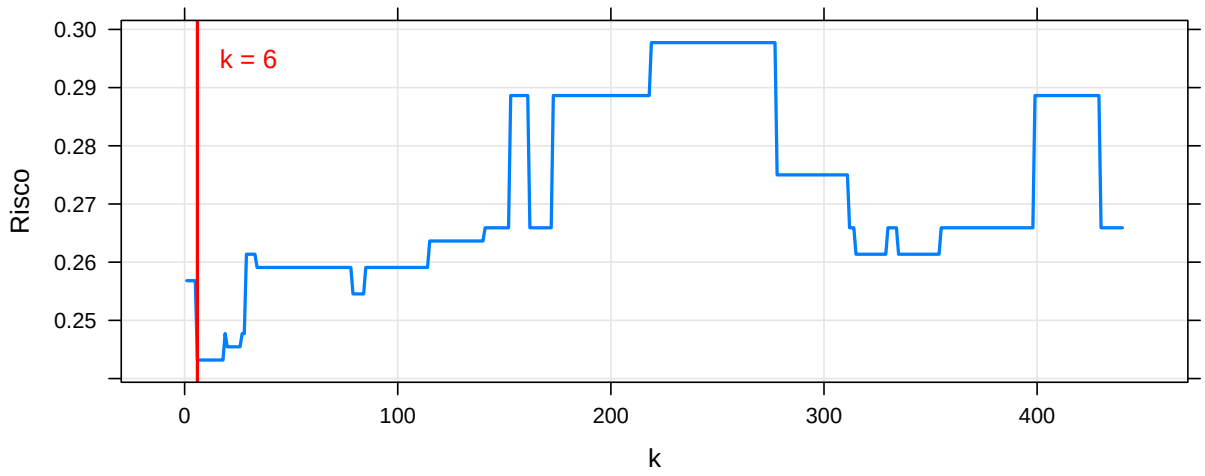
Análise Discriminante Quadrática

```r
# <code r> ========================================================================== #
dq <- qda(Survived ~ ., train)

dq$scaling
# </code r> ========================================================================== #
```

```
, , No

                1          2         3          4           5
Class2nd  3.134756 -0.8719008 -3.084898 -0.1277106 -0.01557274
Class3rd  0.000000 -2.1666026 -3.084898 -0.6300692  0.51345720
ClassCrew 0.000000  0.0000000 -3.697256  0.1199123  0.01462184
SexMale   0.000000  0.0000000  0.000000 -3.7335072 -0.45525544
AgeChild  0.000000  0.0000000  0.000000  0.0000000 -5.41305198

, , Yes

                1         2        3          4          5
Class2nd  2.644922 0.7009925 1.335471 -0.2210384  0.6537404
Class3rd  0.000000 2.4216105 1.335471  0.4330530  0.5476621
ClassCrew 0.000000 0.0000000 2.606581  1.5146459 -0.2746628
SexMale   0.000000 0.0000000 0.000000 -2.4252828  0.2847691
AgeChild  0.000000 0.0000000 0.000000  0.0000000 -3.8243018
```

KNN

```r
# <code r> ========================================================================== #
library(FNN)

train.knn <- train
train.knn$Class <- as.numeric(train.knn$Class)
train.knn$Sex <- as.numeric(train.knn$Sex)
train.knn$Age <- as.numeric(train.knn$Age)
train.knn$Survived <- as.numeric(train.knn$Survived)

test.knn <- test
test.knn$Class <- as.numeric(test.knn$Class)
test.knn$Sex <- as.numeric(test.knn$Sex)
test.knn$Age <- as.numeric(test.knn$Age)

risco <- 0
for (i in 1:nrow(test.knn)){
  knn <- knn.reg(train.knn[ , -4], test.knn[ , -4], train.knn[ , 4], k = i)
  risco[i] <- mean(test.knn$Survived != ifelse(knn$pred < 1.5, "No", "Yes"))}

library(latticeExtra)
xyplot(risco ~ 1:nrow(test.knn)
       , type = c("l", "g")
       , xlab = "k"
       , ylab = "Risco"
       , lwd = 2
       , panel= function(...){
         panel.xyplot(...)
         panel.abline(v = which.min(risco), col = 2, lwd = 2)
         panel.text(30, .295, labels = paste("k =", which.min(risco)), col = 2)
         })
# </code r> ========================================================================== #
```



Qual o melhor classificador segundo o risco estimado?

Regressão Logística:

```
# <code r> ========================================================== #
mean(
  test$Survived != ifelse(predict(reg.log, test, type = "response") < .5
                          , "No", "Yes"))
# </code r> ========================================================== #
```

```
[1] 0.2409091
```

Regressão Linear:

```
# <code r> ========================================================== #
mean(test$Survived != ifelse(predict(reg.lin, test) < 1.5, "No", "Yes"))
# </code r> ========================================================== #
```

```
[1] 0.2409091
```

Naive Bayes:

```
# <code r> ========================================================== #
mean(test$Survived != predict(nb, test))
# </code r> ========================================================== #
```

```
[1] 0.2409091
```

Análise Discriminante Linear:

```
# <code r> ========================================================== #
mean(test$Survived != predict(dl, test)$class)
# </code r> ========================================================== #
```

```
[1] 0.2409091
```

Análise Discriminante Quadrática:

```r
# <code r> ========================================================================== #
mean(test$Survived != predict(dq, test)$class)
# </code r> ========================================================================= #
```

```
[1] 0.2477273
```

KNN:

```r
# <code r> ========================================================================== #
knn <- knn.reg(
  train.knn[ , -4], test.knn[ , -4], train.knn[ , 4], k = which.min(risco))

mean(test.knn$Survived != ifelse(knn$pred < 1.5, "No", "Yes"))
# </code r> ========================================================================= #
```

```
[1] 0.2431818
```

Com a Regressão Logística, Regressão Linear, Naive Bayes e Análise Discriminante Linear, o risco estimado é o mesmo, 0.2409091.

Com o KNN o risco estimado é um pouco maior, 0.2431818.

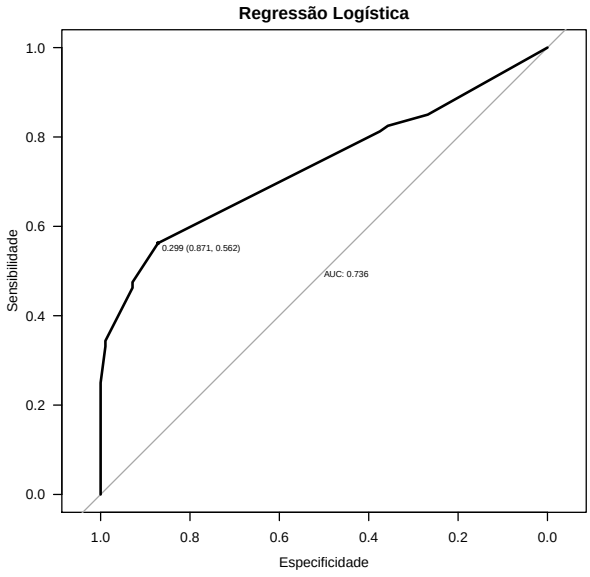
Com a Análise Discriminante Quadrática o maior risco foi estimado, 0.2477273.

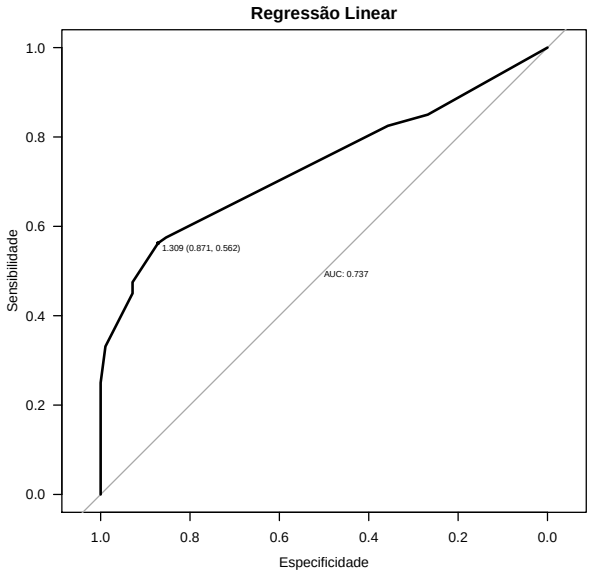
Para os classificadores baseados em estimativas de probabilidade, faça também as curvas ROC com o conjunto de teste. Faça também a tabela de confusão quando o corte usado é 0.5 e também quando o corte é aquele que maximiza sensibilidade mais especificidade

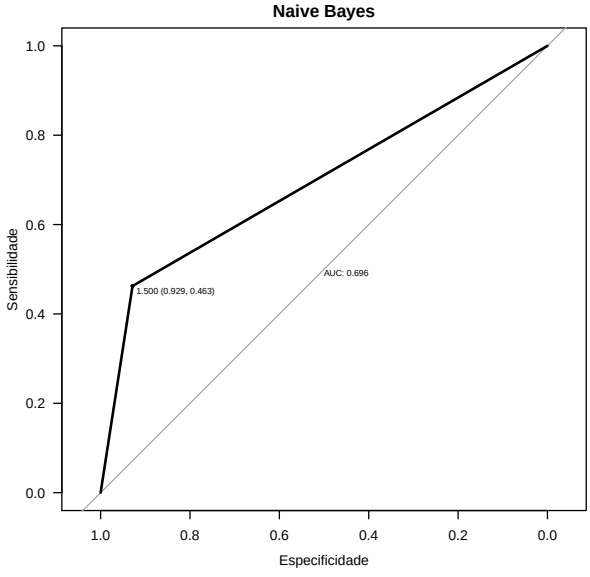
---

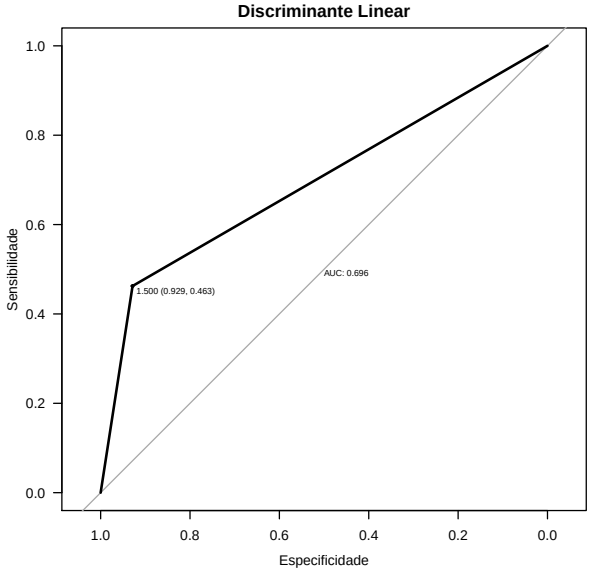
```r
# <code r> ========================================================================== #
library(pROC)

par(mfrow = c(3, 2))

plot.roc(
  roc(test$Survived, predict(reg.log, test, type = "response"))
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "Regressão Logística")
```

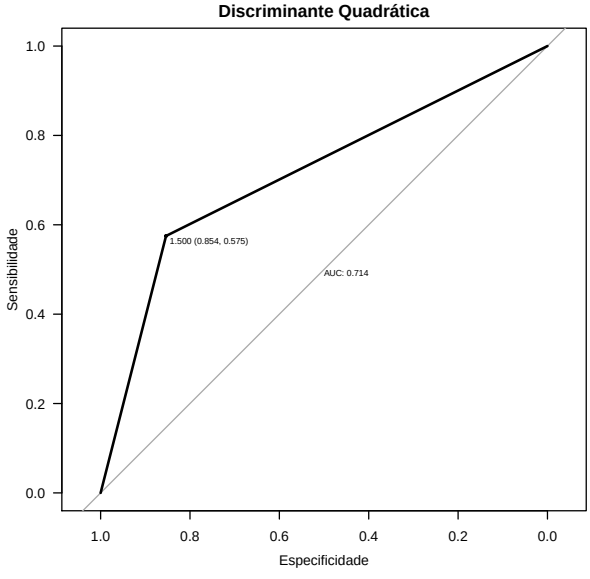
```r
plot.roc(
  roc(test$Survived, predict(reg.lin, test))
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "Regressão Linear")

plot.roc(
  roc(test$Survived, as.numeric(predict(nb, test)))
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "Naive Bayes")

plot.roc(
  roc(test$Survived, as.numeric(predict(dl, test)$class))
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "Discriminante Linear")

plot.roc(
  roc(test$Survived, as.numeric(predict(dq, test)$class))
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "Discriminante Quadrática")

plot.roc(
  roc(test$Survived, knn$pred)
  , print.auc = TRUE
  , print.thres = TRUE
  , las = 1
  , xlab = "Especificidade"
  , ylab = "Sensibilidade"
  , main = "KNN")
# </code r> ============================================================== #
```

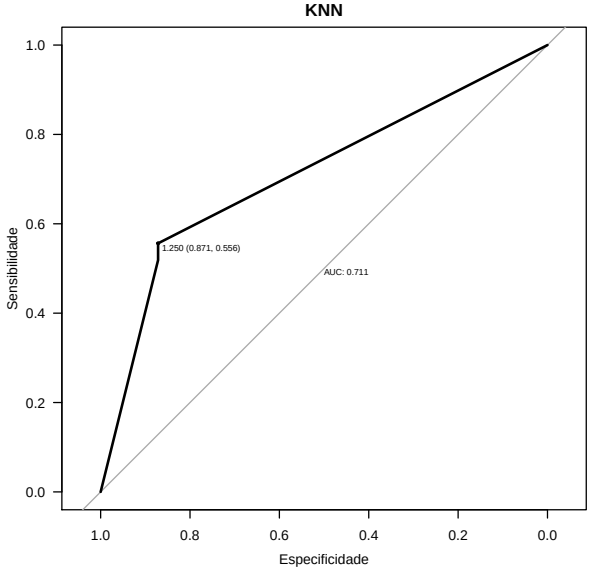

Regressão Logística
Sensibilidade
Especificidade
0.299 (0.871, 0.562)
AUC: 0.736


Regressão Linear
Sensibilidade
Especificidade
1.309 (0.871, 0.562)
AUC: 0.737


Naive Bayes
Sensibilidade
Especificidade
1.500 (0.929, 0.463)
AUC: 0.696


Discriminante Linear
Sensibilidade
Especificidade
1.500 (0.929, 0.463)
AUC: 0.696


Discriminante Quadrática
Sensibilidade
Especificidade
1.500 (0.854, 0.575)
AUC: 0.714


KNN
Sensibilidade
Especificidade
1.250 (0.871, 0.556)
AUC: 0.711

Tabelas de confusão:

Regressão Logística:

Ponto de corte 0.5:

```
# <code r> ============================================================== #
table(test$Survived, ifelse(predict(reg.log, test, type = "response") < .5
                            , "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ============================================================== #
```

```
          Predito
Observado  No Yes
      No  260  20
      Yes  86  74
```

Ponto de corte 0.299:

```
# <code r> ============================================================== #
table(test$Survived, ifelse(predict(reg.log, test, type = "response") < .299
                            , "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ============================================================== #
```

```
          Predito
Observado  No Yes
      No  244  36
      Yes  70  90
```

Regressão Linear:

Ponto de corte 1.5:

```
# <code r> ============================================================== #
table(test$Survived, ifelse(predict(reg.lin, test) < 1.5, "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ============================================================== #
```

```
          Predito
Observado  No Yes
      No  260  20
      Yes  86  74
```

Ponto de corte 1.309:

```r
# <code r> ========================================================================= #
table(test$Survived, ifelse(predict(reg.lin, test) < 1.309, "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ========================================================================= #
```

```
         Predito
Observado  No Yes
      No  244  36
      Yes  70  90
```

Naive Bayes:

Ponto de corte 1.5:

```r
# <code r> ========================================================================= #
table(test$Survived, predict(nb, test), dnn = list("Observado", "Predito"))
# </code r> ========================================================================= #
```

```
         Predito
Observado  No Yes
      No  260  20
      Yes  86  74
```

Análise Discriminante Linear:

Ponto de corte 1.5:

```r
# <code r> ========================================================================= #
table(test$Survived, predict(dl, test)$class
      , dnn = list("Observado", "Predito"))
# </code r> ========================================================================= #
```

```
         Predito
Observado  No Yes
      No  260  20
      Yes  86  74
```

Análise Discriminante Quadrática:

Ponto de corte 1.5:

```r
# <code r> ========================================================================= #
table(test$Survived, predict(dq, test)$class
      , dnn = list("Observado", "Predito"))
# </code r> ========================================================================= #
```

```
         Predito
Observado  No Yes
      No  239  41
      Yes  68  92
```

KNN:

Ponto de corte 1.5:

```
# <code r> ========================================================================== #
table(test$Survived, ifelse(knn$pred < 1.5, "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ========================================================================== #
```

```
         Predito
Observado  No Yes
      No  244  36
      Yes  71  89
```

Ponto de corte 1.25:

```
# <code r> ========================================================================== #
table(test$Survived, ifelse(knn$pred < 1.25, "No", "Yes")
      , dnn = list("Observado", "Predito"))
# </code r> ========================================================================== #
```

```
         Predito
Observado  No Yes
      No  244  36
      Yes  71  89
```

Sensibilidade: quantos foram corretamente classificados como sobreviventes

Especificidade: quantos foram corretamente classificados como não sobreviventes

A maior sensibilidade, 92, é obtida com a Análise Discriminante Quadrática.

A maior especificidade, 260, é obtida com o:

- ponto de corte 0.5 da Regressão Logística,
- ponto de corte 1.5 da Regressão Linear,
- ponto de corte 1.5 do Naive Bayes,
- ponto de corte 1.5 da Análise Discriminante Linear.