Atividade avaliativa 2 - Classificação

Jonatas Varella

30 de abril de 2019

Exercício 10

###This question should be answered using the Weekly data set, which is part of the ISLR package. This data is similar in nature to the Smarket data from this chapter's lab, except that it contains 1,089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

(a) Produce some numerical and graphical summaries of the Weekly data. Do there appear to be any patterns?

```
library(ISLR)
```

```
## Warning: package 'ISLR' was built under R version 3.5.3
```

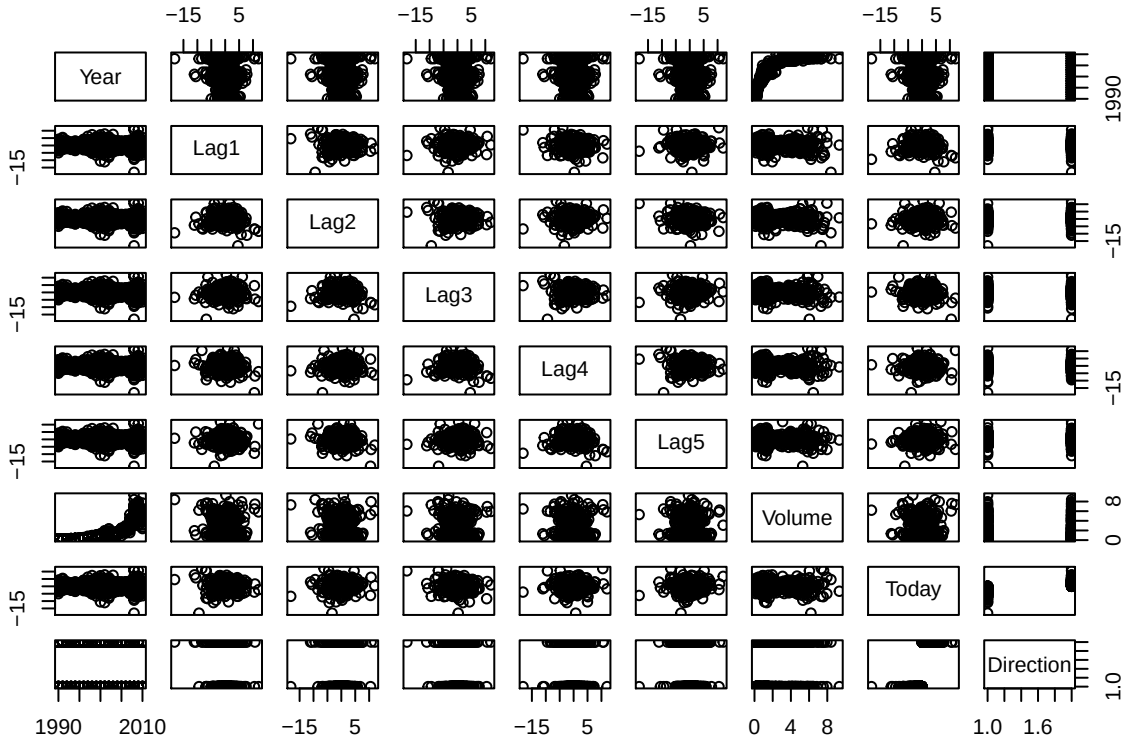
```
bd = Weekly
#help("Weekly")
names(bd)
```

```
## [1] "Year"      "Lag1"      "Lag2"      "Lag3"      "Lag4"      "Lag5"
## [7] "Volume"    "Today"     "Direction"
```

```
summary(bd)
```

```
##       Year           Lag1               Lag2               Lag3
##  Min.   :1990   Min.   :-18.1950   Min.   :-18.1950   Min.   :-18.1950
##  1st Qu.:1995   1st Qu.: -1.1540   1st Qu.: -1.1540   1st Qu.: -1.1580
##  Median :2000   Median :  0.2410   Median :  0.2410   Median :  0.2410
##  Mean   :2000   Mean   :  0.1506   Mean   :  0.1511   Mean   :  0.1472
##  3rd Qu.:2005   3rd Qu.:  1.4050   3rd Qu.:  1.4090   3rd Qu.:  1.4090
##  Max.   :2010   Max.   : 12.0260   Max.   : 12.0260   Max.   : 12.0260
##       Lag4               Lag5              Volume
##  Min.   :-18.1950   Min.   :-18.1950   Min.   :0.08747
##  1st Qu.: -1.1580   1st Qu.: -1.1660   1st Qu.:0.33202
##  Median :  0.2380   Median :  0.2340   Median :1.00268
##  Mean   :  0.1458   Mean   :  0.1399   Mean   :1.57462
##  3rd Qu.:  1.4090   3rd Qu.:  1.4050   3rd Qu.:2.05373
##  Max.   : 12.0260   Max.   : 12.0260   Max.   :9.32821
##      Today          Direction
##  Min.   :-18.1950   Down:484
##  1st Qu.: -1.1540   Up  :605
##  Median :  0.2410
##  Mean   :  0.1499
##  3rd Qu.:  1.4050
##  Max.   : 12.0260
```

```
pairs(bd)
```




```
cor(bd[-9])
```

```
##                Year         Lag1        Lag2        Lag3         Lag4
## Year    1.00000000 -0.032289274 -0.03339001 -0.03000649 -0.031127923
## Lag1   -0.03228927  1.000000000 -0.07485305  0.05863568 -0.071273876
## Lag2   -0.03339001 -0.074853051  1.00000000 -0.07572091  0.058381535
## Lag3   -0.03000649  0.058635682 -0.07572091  1.00000000 -0.075395865
## Lag4   -0.03112792 -0.071273876  0.05838153 -0.07539587  1.000000000
## Lag5   -0.03051910 -0.008183096 -0.07249948  0.06065717 -0.075675027
## Volume  0.84194162 -0.064951313 -0.08551314 -0.06928771 -0.061074617
## Today  -0.03245989 -0.075031842  0.05916672 -0.07124364 -0.007825873
##                Lag5      Volume        Today
## Year   -0.030519101  0.84194162 -0.032459894
## Lag1   -0.008183096 -0.06495131 -0.075031842
## Lag2   -0.072499482 -0.08551314  0.059166717
## Lag3    0.060657175 -0.06928771 -0.071243639
## Lag4   -0.075675027 -0.06107462 -0.007825873
## Lag5    1.000000000 -0.05851741  0.011012698
## Volume -0.058517414  1.00000000 -0.033077783
## Today   0.011012698 -0.03307778  1.000000000
```

Resposta:

Observa-se correlação apenas entre as variáveis "Year" e "Volume". Aparentemente, o volume aumenta ao longo dos anos.

(b) Use the full data set to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results.

Do any of the predictors appear to be statistically significant? If so, which ones?

```
names(bd)
```

```
## [1] "Year"      "Lag1"      "Lag2"      "Lag3"      "Lag4"      "Lag5"
## [7] "Volume"    "Today"     "Direction"
```

```
reg = glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume, data = bd, family = binomial)
summary(reg)
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +
##     Volume, family = binomial, data = bd)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6949  -1.2565   0.9913   1.0849   1.4579
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.26686    0.08593   3.106   0.0019 **
## Lag1        -0.04127    0.02641  -1.563   0.1181
## Lag2         0.05844    0.02686   2.175   0.0296 *
## Lag3        -0.01606    0.02666  -0.602   0.5469
## Lag4        -0.02779    0.02646  -1.050   0.2937
## Lag5        -0.01447    0.02638  -0.549   0.5833
## Volume      -0.02274    0.03690  -0.616   0.5377
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1496.2  on 1088  degrees of freedom
## Residual deviance: 1486.4  on 1082  degrees of freedom
## AIC: 1500.4
##
## Number of Fisher Scoring iterations: 4
```

Resposta:

Apenas a variável Lag2 apresenta significância estatística.

(c) Compute the confusion matrix and overall fraction of correct predictions. Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression.

```
reg.probs = predict(reg, type = "response")
contrasts(bd$Direction)
```

```
##      Up
## Down  0
## Up    1
```

```
dim(bd)
```

```
## [1] 1089    9
```

```
reg.pred = rep("Down", 1089)
reg.pred[reg.probs > .5]= "Up"
```

```
table(reg.pred, bd$Direction)
```

```
##
## reg.pred Down  Up
##     Down   54  48
##     Up    430 557
```

Resposta:

A diagonal central representa os casos de acerto. Nesse caso, a confusion matrix demonstra que o modelo foi capaz de prever em 557 dias que a direção do mercado foi positiva, e em 54 dias foi negativo. A soma dos dois, 611, representa a quantidade de acerto do modelo. Em contrapartida, o modelo errou em 478 dos dias.

O percentual de acerto é de: 56,1%

(d) Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).

```
train = (bd$Year <= 2008)
test = bd[!train, ]
bd.Direction = bd$Direction[!train]
reg2 = glm(Direction ~  Lag2, data = bd, family = binomial, subset = train)
reg2.prob = predict(reg2, test, type = "response")
reg2.pred = rep("Down", 104)
reg2.pred[reg2.prob >.5] = "Up"
table(reg2.pred, bd.Direction)
```

```
##          bd.Direction
## reg2.pred Down Up
##      Down    9  5
##      Up     34 56
```

Resposta:

Conforme demonstrado na confusion matrix, o modelo acertou 9 vezes onde o mercado teve a direção "Down" e "56" vezes quando o mercado teve a direção "Up". O modelo acertou em 62,5% das vezes.

(e) Repeat (d) using LDA.

```
library(MASS)
```

```
## Warning: package 'MASS' was built under R version 3.5.3
```

```
##
## Attaching package: 'MASS'
```

```
## The following object is masked from 'package:dplyr':
##
##     select
```

```
reg.lda = lda(Direction ~  Lag2, data = bd, family = binomial, subset = train)
reg.lda
```

```
## Call:
## lda(Direction ~ Lag2, data = bd, family = binomial, subset = train)
##
## Prior probabilities of groups:
##      Down        Up
## 0.4477157 0.5522843
```

```
##
## Group means:
##             Lag2
## Down -0.03568254
## Up    0.26036581
##
## Coefficients of linear discriminants:
##            LD1
## Lag2 0.4414162
```

```
reg.lda.pred = predict(reg.lda, test)
names(reg.lda.pred)
```

```
## [1] "class"     "posterior" "x"
```

```
lda.class = reg.lda.pred$class
table(lda.class, bd.Direction)
```

```
##          bd.Direction
## lda.class Down Up
##      Down    9  5
##      Up     34 56
```

```
mean(lda.class == bd.Direction)
```

```
## [1] 0.625
```

Resposta:

O modelo manteve o percentual de acerto em 62,5%.

(f) Repeat (d) using QDA

```
reg.qda = qda(Direction ~ Lag2, data = bd, family = binomial, subset = train)
reg.qda
```

```
## Call:
## qda(Direction ~ Lag2, data = bd, family = binomial, subset = train)
##
## Prior probabilities of groups:
##      Down        Up
## 0.4477157 0.5522843
##
## Group means:
##             Lag2
## Down -0.03568254
## Up    0.26036581
```

```
qda.class = predict(reg.qda, test)$class
table(qda.class, bd.Direction)
```

```
##          bd.Direction
## qda.class Down Up
##      Down    0  0
##      Up     43 61
```

```
mean(qda.class==bd.Direction)
```

```
## [1] 0.5865385
```

Resposta:

Usando o QDA, a acurácia diminuiu. O resultado foi de 58,6%.

(h) Which of these methods appears to provide the best results on this data?

Resposta:

Os dois melhores modelos foram a regressão logística e o LDA. Ambos acertam 62,5% das vezes.

Exercício 11

###In this problem, you will develop a model to predict wheter a given car gets high or low gas mileage based on the Auto data set

(a) Create a binary variable, mpg01, that contains a 1 if mpg contains a value aboce its median, and a 0 if mpg contains a value below its median. You can compute the median using the median() function. Note tou may find it helpful to use the data.frame() function to create a single data set containing both mpg01 and the other Auto variables

Resposta

```
library(ISLR)
summary(Auto)
```

```
##       mpg          cylinders      displacement     horsepower
##  Min.   : 9.00   Min.   :3.000   Min.   : 68.0   Min.   : 46.0
##  1st Qu.:17.00   1st Qu.:4.000   1st Qu.:105.0   1st Qu.: 75.0
##  Median :22.75   Median :4.000   Median :151.0   Median : 93.5
##  Mean   :23.45   Mean   :5.472   Mean   :194.4   Mean   :104.5
##  3rd Qu.:29.00   3rd Qu.:8.000   3rd Qu.:275.8   3rd Qu.:126.0
##  Max.   :46.60   Max.   :8.000   Max.   :455.0   Max.   :230.0
##
##      weight      acceleration        year           origin
##  Min.   :1613   Min.   : 8.00   Min.   :70.00   Min.   :1.000
##  1st Qu.:2225   1st Qu.:13.78   1st Qu.:73.00   1st Qu.:1.000
##  Median :2804   Median :15.50   Median :76.00   Median :1.000
##  Mean   :2978   Mean   :15.54   Mean   :75.98   Mean   :1.577
##  3rd Qu.:3615   3rd Qu.:17.02   3rd Qu.:79.00   3rd Qu.:2.000
##  Max.   :5140   Max.   :24.80   Max.   :82.00   Max.   :3.000
##
##                  name
##  amc matador       :  5
##  ford pinto        :  5
##  toyota corolla    :  5
##  amc gremlin       :  4
##  amc hornet        :  4
##  chevrolet chevette:  4
##  (Other)           :365
```

```
bd = Auto
class(bd$mpg)
```

```
## [1] "numeric"
```

```
median(bd$mpg)
```

```
## [1] 22.75
```

```
bd = bd %>% mutate(mpg01 = case_when(
  mpg > median(mpg) ~ 1,
  mpg <= median(mpg) ~ 0
))
```

b) Explore the data graphically in order to investigate the association between mpg01 and the other features. Which of the other features seem most likely to be useful in predicting mpg01? Scatterplots and boxplots may be useful tools to answer this question. Describe your findings

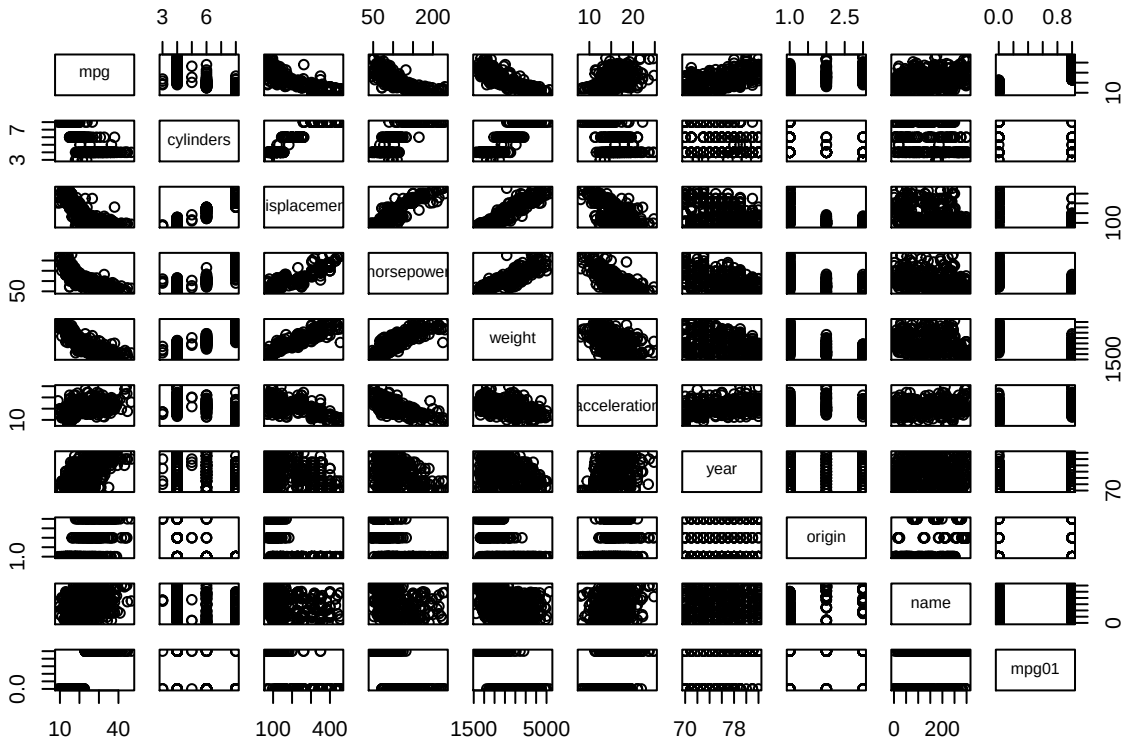
```
names(bd)
```

```
##  [1] "mpg"          "cylinders"    "displacement" "horsepower"
##  [5] "weight"       "acceleration" "year"         "origin"
##  [9] "name"         "mpg01"
```

```
cor(bd[-9])
```

```
##                     mpg  cylinders displacement horsepower     weight
## mpg           1.0000000 -0.7776175   -0.8051269 -0.7784268 -0.8322442
## cylinders    -0.7776175  1.0000000    0.9508233  0.8429834  0.8975273
## displacement -0.8051269  0.9508233    1.0000000  0.8972570  0.9329944
## horsepower   -0.7784268  0.8429834    0.8972570  1.0000000  0.8645377
## weight       -0.8322442  0.8975273    0.9329944  0.8645377  1.0000000
## acceleration  0.4233285 -0.5046834   -0.5438005 -0.6891955 -0.4168392
## year          0.5805410 -0.3456474   -0.3698552 -0.4163615 -0.3091199
## origin        0.5652088 -0.5689316   -0.6145351 -0.4551715 -0.5850054
## mpg01         0.8369392 -0.7591939   -0.7534766 -0.6670526 -0.7577566
##              acceleration       year     origin      mpg01
## mpg             0.4233285  0.5805410  0.5652088  0.8369392
## cylinders      -0.5046834 -0.3456474 -0.5689316 -0.7591939
## displacement   -0.5438005 -0.3698552 -0.6145351 -0.7534766
## horsepower     -0.6891955 -0.4163615 -0.4551715 -0.6670526
## weight         -0.4168392 -0.3091199 -0.5850054 -0.7577566
## acceleration    1.0000000  0.2903161  0.2127458  0.3468215
## year            0.2903161  1.0000000  0.1815277  0.4299042
## origin          0.2127458  0.1815277  1.0000000  0.5136984
## mpg01           0.3468215  0.4299042  0.5136984  1.0000000
```

```
pairs(bd)
```



Resposta

Observa que mpg01 possui uma correlação negativa com as variáveis: “Cylinders”, “Displacement”, “Horsepower” e “Weigth”.

c) Split the data into a training set and a test set.

Resposta

```
train = sample(1:nrow(bd), nrow(bd)*.8)
bd.train = bd[train, ]
bd.test = bd[-train, ]
```

d)Perform LDA on the training data in order to predict mpg01 using the variables that seemed most associated with mpg01 in (b). What is the test error of the model obtained?

```
reg.lda = lda(mpg01 ~ cylinders + displacement + horsepower + weight, data = bd.train,
              family = binomial)
reg.lda
```

```
## Call:
## lda(mpg01 ~ cylinders + displacement + horsepower + weight, data = bd.train,
##     family = binomial)
##
## Prior probabilities of groups:
##         0         1
## 0.4984026 0.5015974
##
## Group means:
```

```
##   cylinders displacement horsepower   weight
## 0  6.750000     272.4167  128.91026 3604.622
## 1  4.171975     115.7484   78.99363 2335.611
##
## Coefficients of linear discriminants:
##                         LD1
## cylinders    -0.4709854286
## displacement -0.0025444385
## horsepower    0.0055798993
## weight       -0.0009079676
```

```
lda.class = predict(reg.lda, bd.test)$class
table(lda.class, bd.test$mpg01)
```

```
##
## lda.class  0  1
##         0 34  2
##         1  6 37
```

```
mean(lda.class==bd.test$mpg01)
```

```
## [1] 0.8987342
```

Resposta

O percentual de acerto é de 88%

e) Perform QDA on the training data in order to predict mpg01 using the variables that seemed most associated with mpg01 in (b). What is the test error of the model obtained?

```
reg.qda = qda(mpg01 ~ cylinders + displacement + horsepower + weight, data = bd.train,
              family = binomial)
reg.qda
```

```
## Call:
## qda(mpg01 ~ cylinders + displacement + horsepower + weight, data = bd.train,
##     family = binomial)
##
## Prior probabilities of groups:
##         0         1
## 0.4984026 0.5015974
##
## Group means:
##   cylinders displacement horsepower   weight
## 0  6.750000     272.4167  128.91026 3604.622
## 1  4.171975     115.7484   78.99363 2335.611
```

```
qda.class = predict(reg.qda, bd.test)$class
table(qda.class, bd.test$mpg01)
```

```
##
## qda.class  0  1
##         0 34  4
##         1  6 35
```

```
mean(qda.class==bd.test$mpg01)
```

```
## [1] 0.8734177
```

Resposta

Utilizando o QDA, a taxa de acerto aumentou para 92%.

f) Perform logistic regression on the training data in order to predict mpg01 using the variables that seemed most associated with mpg01 in (b). What is the test error of the model obtained?

```
reg.lm = glm(mpg01 ~ cylinders + displacement + horsepower + weight, data = bd.train,
             family = binomial)
reg.lm
```

```
##
## Call:  glm(formula = mpg01 ~ cylinders + displacement + horsepower +
##     weight, family = binomial, data = bd.train)
##
## Coefficients:
##  (Intercept)     cylinders  displacement    horsepower        weight
##    11.524481     -0.066421     -0.014726     -0.040363     -0.001717
##
## Degrees of Freedom: 312 Total (i.e. Null);  308 Residual
## Null Deviance:       433.9
## Residual Deviance: 165.8     AIC: 175.8
```

```
reg.probs = predict(reg.lm, type = "response", newdata = bd.test)
dim(bd.test)
```

```
## [1] 79 10
```

```
reg.pred = rep(0, 79)
reg.pred[reg.probs > .5]= 1
table(reg.pred, bd.test$mpg01)
```

```
##
## reg.pred  0  1
##        0 34  3
##        1  6 36
```

```
mean(reg.pred==bd.test$mpg01)
```

```
## [1] 0.8860759
```

Resposta

Assim como no modelo QDA, a regressão logística também obteve 92% de acerto.

Exercício 13

###Using the Boston data set, fit classification models in order to predict whether a given suburb has a crime rate above or below the median. Explore logistic regression, LDA, and KNN models using various subsets of the predictors. Describe your findings.

```
bd = Boston
summary(bd)
```

```
##       crim                zn              indus            chas
##  Min.   : 0.00632   Min.   :  0.00   Min.   : 0.46   Min.   :0.00000
##  1st Qu.: 0.08204   1st Qu.:  0.00   1st Qu.: 5.19   1st Qu.:0.00000
##  Median : 0.25651   Median :  0.00   Median : 9.69   Median :0.00000
##  Mean   : 3.61352   Mean   : 11.36   Mean   :11.14   Mean   :0.06917
##  3rd Qu.: 3.67708   3rd Qu.: 12.50   3rd Qu.:18.10   3rd Qu.:0.00000
##  Max.   :88.97620   Max.   :100.00   Max.   :27.74   Max.   :1.00000
```

```
##       nox               rm             age              dis
##  Min.   :0.3850   Min.   :3.561   Min.   :  2.90   Min.   : 1.130
##  1st Qu.:0.4490   1st Qu.:5.886   1st Qu.: 45.02   1st Qu.: 2.100
##  Median :0.5380   Median :6.208   Median : 77.50   Median : 3.207
##  Mean   :0.5547   Mean   :6.285   Mean   : 68.57   Mean   : 3.795
##  3rd Qu.:0.6240   3rd Qu.:6.623   3rd Qu.: 94.08   3rd Qu.: 5.188
##  Max.   :0.8710   Max.   :8.780   Max.   :100.00   Max.   :12.127
##       rad              tax           ptratio          black
##  Min.   : 1.000   Min.   :187.0   Min.   :12.60   Min.   :  0.32
##  1st Qu.: 4.000   1st Qu.:279.0   1st Qu.:17.40   1st Qu.:375.38
##  Median : 5.000   Median :330.0   Median :19.05   Median :391.44
##  Mean   : 9.549   Mean   :408.2   Mean   :18.46   Mean   :356.67
##  3rd Qu.:24.000   3rd Qu.:666.0   3rd Qu.:20.20   3rd Qu.:396.23
##  Max.   :24.000   Max.   :711.0   Max.   :22.00   Max.   :396.90
##      lstat            medv
##  Min.   : 1.73   Min.   : 5.00
##  1st Qu.: 6.95   1st Qu.:17.02
##  Median :11.36   Median :21.20
##  Mean   :12.65   Mean   :22.53
##  3rd Qu.:16.95   3rd Qu.:25.00
##  Max.   :37.97   Max.   :50.00
```

```
names(bd)
```

```
##  [1] "crim"    "zn"      "indus"   "chas"    "nox"     "rm"      "age"
##  [8] "dis"     "rad"     "tax"     "ptratio" "black"   "lstat"   "medv"
```

```
median(bd$crim)
```

```
## [1] 0.25651
```

```
bd = bd %>% mutate(crim01 = case_when(
  crim > median(crim) ~ 1,
  crim <= median(crim) ~ 0
))
train = sample(1:nrow(bd), nrow(bd)*.8)
bd.train = bd[train, ]
bd.test = bd[-train, ]
```

###Logistic Regression

```
reg.lm = glm(crim01 ~ zn + indus + chas + nox + rm + age + dis + rad + tax + ptratio +
               black + lstat + medv, family = binomial, data = bd.train)
```

```
## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

```
reg.probs = predict(reg.lm, bd.test, type = "response")
dim(bd.test)
```

```
## [1] 102  15
```

```
reg.pred = rep(0,102)
reg.pred[reg.probs > .5] = 1
table(reg.pred, bd.test$crim01)
```

```
##
## reg.pred  0  1
##        0 51  4
##        1  8 39
```

```
mean(reg.pred == bd.test$crim01)
```

```
## [1] 0.8823529
```

LDA

```
reg.lda = lda(crim01 ~ zn + indus + chas + nox + rm + age + dis + rad + tax + ptratio +
              black + lstat + medv, family = binomial, data = bd.train)
reg.lda
```

```
## Call:
## lda(crim01 ~ zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat + medv, data = bd.train, family = binomial)
##
## Prior probabilities of groups:
##        0        1
## 0.480198 0.519802
##
## Group means:
##          zn     indus       chas       nox       rm      age      dis
## 0 22.693299  6.829072 0.04123711 0.4673907 6.413881 49.69588 5.186696
## 1  1.352381 15.215143 0.09047619 0.6401714 6.198619 84.85810 2.500454
##         rad      tax  ptratio    black     lstat     medv
## 0  4.025773 300.6701 17.85000 390.6811  9.071804 25.30928
## 1 15.123810 511.7476 18.90857 323.3002 15.668095 20.50619
##
## Coefficients of linear discriminants:
##                   LD1
## zn      -0.0070326049
## indus    0.0024688147
## chas     0.1613223076
## nox      7.4259189682
## rm       0.0937475508
## age      0.0091012996
## dis      0.0018934046
## rad      0.0688703861
## tax     -0.0002609018
## ptratio  0.0243487793
## black   -0.0008549628
## lstat    0.0290568409
## medv     0.0401036279
```

```
lda.class = predict(reg.lda, bd.test)$class
table(lda.class, bd.test$crim01)
```

```
##
## lda.class  0  1
##         0 54 11
##         1  5 32
```

```
mean(lda.class==bd.test$crim01)
```

```
## [1] 0.8431373
```

QDA

```
reg.qda = qda(crim01 ~ zn + indus + chas + nox + rm + age + dis + rad + tax + ptratio +
              black + lstat + medv, family = binomial, data = bd.train)
reg.qda
```

```
## Call:
## qda(crim01 ~ zn + indus + chas + nox + rm + age + dis + rad +
##     tax + ptratio + black + lstat + medv, data = bd.train, family = binomial)
##
## Prior probabilities of groups:
##        0        1
## 0.480198 0.519802
##
## Group means:
##          zn     indus       chas       nox       rm      age      dis
## 0 22.693299  6.829072 0.04123711 0.4673907 6.413881 49.69588 5.186696
## 1  1.352381 15.215143 0.09047619 0.6401714 6.198619 84.85810 2.500454
##         rad      tax  ptratio    black     lstat     medv
## 0  4.025773 300.6701 17.85000 390.6811  9.071804 25.30928
## 1 15.123810 511.7476 18.90857 323.3002 15.668095 20.50619
```

```
qda.class = predict(reg.qda, bd.test)$class
table(qda.class, bd.test$crim01)
```

```
##
## qda.class  0  1
##         0 55 12
##         1  4 31
```

```
mean(qda.class==bd.test$crim01)
```

```
## [1] 0.8431373
```

Resposta

Os modelos apresentaram o seguinte resultados:

Regressão Logística: 88% de acerto; QDA: 88% de acerto; LDA: 83% de acerto.

Com base nesses dados, podemos dizer que a regressão logística e o QDA obtiveram os melhores resultados e o melhor ajuste.