ML bpalazzo_3

Brandon Palazzo

10/12/2020

```r
library(caret)
```

```
## Loading required package: lattice

## Loading required package: ggplot2
```

```r
library(lattice)
library(ggplot2)
library(e1071)
library(gmodels)
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following object is masked from 'package:gmodels':
##
##     ci

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```r
data<- read.csv("FlightDelays.csv")
str(data)
```

```
## 'data.frame':    2201 obs. of  13 variables:
##  $ CRS_DEP_TIME : int  1455 1640 1245 1715 1039 840 1240 1645 1715 2120 ...
##  $ CARRIER      : chr  "OH" "DH" "DH" "DH" ...
##  $ DEP_TIME     : int  1455 1640 1245 1709 1035 839 1243 1644 1710 2129 ...
##  $ DEST         : chr  "JFK" "JFK" "LGA" "LGA" ...
##  $ DISTANCE     : int  184 213 229 229 229 228 228 228 228 228 ...
##  $ FL_DATE      : chr  "01/01/2004" "01/01/2004" "01/01/2004" "01/01/2004" ...
##  $ FL_NUM       : int  5935 6155 7208 7215 7792 7800 7806 7810 7812 7814 ...
##  $ ORIGIN       : chr  "BWI" "DCA" "IAD" "IAD" ...
##  $ Weather      : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ DAY_WEEK     : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ DAY_OF_MONTH : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ TAIL_NUM     : chr  "N940CA" "N405FJ" "N695BR" "N662BR" ...
##  $ Flight.Status: chr  "ontime" "ontime" "ontime" "ontime" ...
```

Making Flight.Status Categorical

```
data$Flight.Status2[data$Flight.Status == "ontime"] = 0
data$Flight.Status2[data$Flight.Status == "delayed"] = 1
str(data)
```

```
## 'data.frame':    2201 obs. of  14 variables:
##  $ CRS_DEP_TIME  : int  1455 1640 1245 1715 1039 840 1240 1645 1715 2120 ...
##  $ CARRIER       : chr  "OH" "DH" "DH" "DH" ...
##  $ DEP_TIME      : int  1455 1640 1245 1709 1035 839 1243 1644 1710 2129 ...
##  $ DEST          : chr  "JFK" "JFK" "LGA" "LGA" ...
##  $ DISTANCE      : int  184 213 229 229 229 228 228 228 228 228 ...
##  $ FL_DATE       : chr  "01/01/2004" "01/01/2004" "01/01/2004" "01/01/2004" ...
##  $ FL_NUM        : int  5935 6155 7208 7215 7792 7800 7806 7810 7812 7814 ...
##  $ ORIGIN        : chr  "BWI" "DCA" "IAD" "IAD" ...
##  $ Weather       : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ DAY_WEEK      : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ DAY_OF_MONTH  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ TAIL_NUM      : chr  "N940CA" "N405FJ" "N695BR" "N662BR" ...
##  $ Flight.Status : chr  "ontime" "ontime" "ontime" "ontime" ...
##  $ Flight.Status2: num  0 0 0 0 0 0 0 0 0 0 ...
```

Making predictor variables as factors

```
data$DAY_WEEK <- as.factor(data$DAY_WEEK)
data$CRS_DEP_TIME <- as.factor(data$CRS_DEP_TIME)
data$ORIGIN <- as.factor(data$ORIGIN)
data$DEST <- as.factor(data$DEST)
data$CARRIER <- as.factor(data$CARRIER)
data$Flight.Status2 <- as.factor(data$Flight.Status2)
```

Parition the data

```
data2 <- data[, c(1,2,4,8,10,14)]

str(data2)
```

```
## 'data.frame':    2201 obs. of  6 variables:
##  $ CRS_DEP_TIME  : Factor w/ 59 levels "600","630","640",..: 33 43 26 47 19 11 25 44 47 58 ...
##  $ CARRIER       : Factor w/ 8 levels "CO","DH","DL",..: 5 2 2 2 2 2 2 2 2 2 ...
##  $ DEST          : Factor w/ 3 levels "EWR","JFK","LGA": 2 2 3 3 3 2 2 2 2 2 ...
##  $ ORIGIN        : Factor w/ 3 levels "BWI","DCA","IAD": 1 2 3 3 3 3 3 3 3 3 ...
##  $ DAY_WEEK      : Factor w/ 7 levels "1","2","3","4",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Flight.Status2: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
set.seed(123)
Index_Train<-createDataPartition(data2$Flight.Status2, p=0.6, list=FALSE)
Train <-data2[Index_Train,]
Test  <-data2[-Index_Train,]

str(Train)
```

```
## 'data.frame':    1321 obs. of  6 variables:
##  $ CRS_DEP_TIME  : Factor w/ 59 levels "600","630","640",..: 43 47 19 11 44 47 58 58 16 50 ...
##  $ CARRIER       : Factor w/ 8 levels "CO","DH","DL",..: 2 2 2 2 2 2 2 2 3 3 ...
##  $ DEST          : Factor w/ 3 levels "EWR","JFK","LGA": 2 3 3 2 2 2 2 3 3 3 ...
##  $ ORIGIN        : Factor w/ 3 levels "BWI","DCA","IAD": 2 3 3 3 3 3 3 3 2 2 ...
##  $ DAY_WEEK      : Factor w/ 7 levels "1","2","3","4",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Flight.Status2: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

```
str(Test)
```

```
## 'data.frame':    880 obs. of  6 variables:
##  $ CRS_DEP_TIME  : Factor w/ 59 levels "600","630","640",..: 33 26 25 33 24 32 52 14 27 31 ...
##  $ CARRIER       : Factor w/ 8 levels "CO","DH","DL",..: 5 2 2 3 3 3 4 4 4 4 ...
##  $ DEST          : Factor w/ 3 levels "EWR","JFK","LGA": 2 3 2 2 3 3 2 3 3 3 ...
##  $ ORIGIN        : Factor w/ 3 levels "BWI","DCA","IAD": 1 3 3 2 2 2 2 2 2 2 ...
##  $ DAY_WEEK      : Factor w/ 7 levels "1","2","3","4",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Flight.Status2: Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 1 1 1 ...
```

Building a naïve Bayes classifier

```
nb_model <-naiveBayes(Flight.Status2~CRS_DEP_TIME+CARRIER+DEST+ORIGIN+DAY_WEEK,data = Train)
nb_model
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##         0         1
## 0.8054504 0.1945496
##
## Conditional probabilities:
##    CRS_DEP_TIME
## Y            600          630          640          645          700
##   0 0.0140977444 0.0300751880 0.0075187970 0.0131578947 0.0479323308
##   1 0.0077821012 0.0116731518 0.0272373541 0.0038910506 0.0428015564
##    CRS_DEP_TIME
## Y            730          735          759          800          830
##   0 0.0112781955 0.0075187970 0.0009398496 0.0159774436 0.0122180451
##   1 0.0038910506 0.0038910506 0.0000000000 0.0116731518 0.0116731518
##    CRS_DEP_TIME
## Y            840          845          850          900          925
##   0 0.0300751880 0.0028195489 0.0159774436 0.0375939850 0.0018796992
##   1 0.0116731518 0.0000000000 0.0116731518 0.0233463035 0.0000000000
##    CRS_DEP_TIME
## Y            930         1000         1030         1039         1040
##   0 0.0159774436 0.0112781955 0.0206766917 0.0028195489 0.0075187970
##   1 0.0000000000 0.0000000000 0.0155642023 0.0000000000 0.0038910506
##    CRS_DEP_TIME
## Y           1100         1130         1200         1230         1240
```

```
##   0 0.0253759398 0.0112781955 0.0122180451 0.0122180451 0.0169172932
##   1 0.0155642023 0.0038910506 0.0000000000 0.0038910506 0.0194552529
##    CRS_DEP_TIME
## Y          1245         1300         1315         1330         1359
##   0 0.0216165414 0.0545112782 0.0000000000 0.0140977444 0.0112781955
##   1 0.0466926070 0.0272373541 0.0077821012 0.0000000000 0.0038910506
##    CRS_DEP_TIME
## Y          1400         1430         1455         1500         1515
##   0 0.0244360902 0.0244360902 0.0479323308 0.0338345865 0.0018796992
##   1 0.0155642023 0.0311284047 0.0856031128 0.0350194553 0.0116731518
##    CRS_DEP_TIME
## Y          1520         1525         1530         1600         1605
##   0 0.0009398496 0.0093984962 0.0216165414 0.0216165414 0.0000000000
##   1 0.0000000000 0.0194552529 0.0233463035 0.0311284047 0.0038910506
##    CRS_DEP_TIME
## Y          1610         1630         1640         1645         1700
##   0 0.0131578947 0.0234962406 0.0150375940 0.0150375940 0.0328947368
##   1 0.0077821012 0.0194552529 0.0077821012 0.0038910506 0.0311284047
##    CRS_DEP_TIME
## Y          1710         1715         1720         1725         1730
##   0 0.0150375940 0.0225563910 0.0084586466 0.0009398496 0.0178571429
##   1 0.0116731518 0.0505836576 0.0233463035 0.0000000000 0.0466926070
##    CRS_DEP_TIME
## Y          1800         1830         1900         1930         2000
##   0 0.0131578947 0.0281954887 0.0310150376 0.0084586466 0.0103383459
##   1 0.0000000000 0.0272373541 0.0739299611 0.0077821012 0.0077821012
##    CRS_DEP_TIME
## Y          2030         2100         2120         2130
##   0 0.0140977444 0.0187969925 0.0385338346 0.0000000000
##   1 0.0038910506 0.0233463035 0.0778210117 0.0000000000
##
##    CARRIER
## Y           CO         DH         DL         MQ         OH         RU
##   0 0.04135338 0.24530075 0.18703008 0.11748120 0.01315789 0.17763158
##   1 0.07392996 0.33852140 0.07392996 0.18677043 0.01167315 0.20233463
##    CARRIER
## Y           UA         US
##   0 0.01597744 0.20206767
##   1 0.01167315 0.10116732
##
##    DEST
## Y         EWR       JFK       LGA
##   0 0.2998120 0.1701128 0.5300752
##   1 0.3813230 0.1712062 0.4474708
##
##    ORIGIN
## Y          BWI        DCA        IAD
##   0 0.05263158 0.64285714 0.30451128
##   1 0.09727626 0.50583658 0.39688716
##
##    DAY_WEEK
## Y            1          2          3          4          5          6
##   0 0.13909774 0.14473684 0.14003759 0.17669173 0.18045113 0.10996241
##   1 0.17509728 0.16731518 0.14785992 0.15175097 0.17898833 0.04669261
```

```
##    DAY_WEEK
## Y            7
##   0 0.10902256
##   1 0.13229572
```

Counts and Proportion Table for DEST

```r
table(data2$Flight.Status2, data2$DEST)
```

```
##
##     EWR JFK LGA
##   0 504 302 967
##   1 161  84 183
```

```r
prop.table(table(data2$Flight.Status2, data2$DEST))
```

```
##
##            EWR        JFK        LGA
##   0 0.22898682 0.13721036 0.43934575
##   1 0.07314857 0.03816447 0.08314403
```

Model the test set

```r
Predicted_Test_labels <-predict(nb_model,Test)

CrossTable(x=Test$Flight.Status2,y=Predicted_Test_labels, prop.chisq = FALSE)
```

```
##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  880
##
##
##                     | Predicted_Test_labels
## Test$Flight.Status2 |         0 |         1 | Row Total |
## --------------------|-----------|-----------|-----------|
##                   0 |       668 |        41 |       709 |
##                     |     0.942 |     0.058 |     0.806 |
##                     |     0.824 |     0.594 |           |
##                     |     0.759 |     0.047 |           |
## --------------------|-----------|-----------|-----------|
##                   1 |       143 |        28 |       171 |
##                     |     0.836 |     0.164 |     0.194 |
```

```
##                      |     0.176 |     0.406 |           |
##                      |     0.163 |     0.032 |           |
## ---------------------|-----------|-----------|-----------|
##         Column Total |       811 |        69 |       880 |
##                      |     0.922 |     0.078 |           |
## ---------------------|-----------|-----------|-----------|
##
##
```

```
confusionMatrix(table(Predicted_Test_labels, Test$Flight.Status2))
```

```
## Confusion Matrix and Statistics
##
##
## Predicted_Test_labels   0   1
##                     0 668 143
##                     1  41  28
##
##                Accuracy : 0.7909
##                  95% CI : (0.7625, 0.8173)
##     No Information Rate : 0.8057
##     P-Value [Acc > NIR] : 0.8744
##
##                   Kappa : 0.1369
##
##  Mcnemar's Test P-Value : 9.634e-14
##
##             Sensitivity : 0.9422
##             Specificity : 0.1637
##          Pos Pred Value : 0.8237
##          Neg Pred Value : 0.4058
##              Prevalence : 0.8057
##          Detection Rate : 0.7591
##    Detection Prevalence : 0.9216
##       Balanced Accuracy : 0.5530
##
##        'Positive' Class : 0
##
```

Raw Prediction Probablities

```
nb_model <- naiveBayes(Flight.Status2~CRS_DEP_TIME+CARRIER+DEST+ORIGIN+DAY_WEEK,data = Train)

Predicted_Test_labels <-predict(nb_model,Test, type = "raw")
```

ROC Curve

```
roc(Test$Flight.Status2, Predicted_Test_labels[,2])
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```
##
## Call:
## roc.default(response = Test$Flight.Status2, predictor = Predicted_Test_labels[,     2])
##
## Data: Predicted_Test_labels[, 2] in 709 controls (Test$Flight.Status2 0) < 171 cases (Test$Flight.Sta
## Area under the curve: 0.6676
```

```
plot.roc(Test$Flight.Status2,Predicted_Test_labels[,2])
```

```
## Setting levels: control = 0, case = 1
## Setting direction: controls < cases
```