mbruner3_3

Mark Bruner

10/4/2020

Libraries needed for this assignment.

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union

## Loading required package: lattice

##
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':
##
##     lift

## naivebayes 0.9.7 loaded
```

Imported dataset.

```
## # A tibble: 6 x 13
##   CRS_DEP_TIME CARRIER DEP_TIME DEST  DISTANCE FL_DATE    FL_NUM ORIGIN Weather
##          <dbl> <fct>      <dbl> <fct>    <int> <date>     <fct>  <fct>  <fct>
## 1         1455 OH          1455 JFK        184 2004-01-01 5935   BWI    0
```

```
## 2         1640 DH          1640 JFK        213 2004-01-01 6155   DCA    0
## 3         1245 DH          1245 LGA        229 2004-01-01 7208   IAD    0
## 4         1715 DH          1709 LGA        229 2004-01-01 7215   IAD    0
## 5         1039 DH          1035 LGA        229 2004-01-01 7792   IAD    0
## 6          840 DH           839 JFK        228 2004-01-01 7800   IAD    0
## # ... with 4 more variables: DAY_WEEK <fct>, DAY_OF_MONTH <fct>,
## #   TAIL_NUM <fct>, `Flight Status` <fct>
```

Getting to know data.

```
summary(flight_delays)
```

```
##   CRS_DEP_TIME     CARRIER       DEP_TIME     DEST         DISTANCE
##  Min.   : 600   DH     :551   Min.   :  10   JFK: 386   Min.   :169.0
##  1st Qu.:1000   RU     :408   1st Qu.:1004   LGA:1150   1st Qu.:213.0
##  Median :1455   US     :404   Median :1450   EWR: 665   Median :214.0
##  Mean   :1372   DL     :388   Mean   :1369              Mean   :211.9
##  3rd Qu.:1710   MQ     :295   3rd Qu.:1709              3rd Qu.:214.0
##  Max.   :2130   CO     : 94   Max.   :2330              Max.   :229.0
##                 (Other): 61
##     FL_DATE               FL_NUM     ORIGIN     Weather  DAY_WEEK
##  Min.   :2004-01-01   7800   :  31   BWI: 145   0:2169   4:372
##  1st Qu.:2004-01-08   7806   :  31   DCA:1370   1:  32   5:391
##  Median :2004-01-16   7812   :  31   IAD: 686            6:250
##  Mean   :2004-01-16   7814   :  31                       7:253
##  3rd Qu.:2004-01-23   746    :  31                       1:308
##  Max.   :2004-01-31   1768   :  31                       2:307
##                       (Other):2015                       3:320
##   DAY_OF_MONTH     TAIL_NUM    Flight Status
##  22     :  86   N225DL :  65   ontime :1773
##  6      :  85   N242DL :  56   delayed: 428
##  8      :  85   N223DZ :  50
##  13     :  85   N221DL :  45
##  20     :  85   N241DL :  36
##  21     :  85   N722UW :  36
##  (Other):1690   (Other):1913
```

The five main carriers for the flights are DH, RU, US, DL, and MQ. Over half of the flights are to LGA (La Guardia, NY). The mean and median a fairly close together for the distance variable which means that the distribution is symmetrical. Most of the flights originated out of DCA (Washington DC). The weather only impacted 32/2201 flights and caused delays from January 25 to January 27th mostly. More flights occurred on days 4 and 5. Lastly, about 19% of all flights were delayed in January.

Checked for missing values and created dummy variables.

```
colMeans(is.na(flight_delays)) # no missing data.
```

```
##  CRS_DEP_TIME       CARRIER      DEP_TIME          DEST      DISTANCE
##             0             0             0             0             0
##       FL_DATE       FL_NUM        ORIGIN       Weather      DAY_WEEK
##             0             0             0             0             0
##  DAY_OF_MONTH      TAIL_NUM Flight Status
##             0             0             0
```

```
flight_delays <- dummy_cols(flight_delays, select_columns = "Flight Status", remove_selected_columns = T

flight_delays <- flight_delays %>%
  rename("On Time" = "Flight Status_ontime")

flight_delays <- flight_delays %>%
  rename("Delayed" = "Flight Status_delayed")

flight_delays$`On Time` <- as.factor(flight_delays$`On Time`)
flight_delays$Delayed <- as.factor(flight_delays$Delayed)
```

Since all variables have column means of 0, that implies that there are no missing values in this dataset.

Created "bins" for CRS_DEP_TIME (Scheduled Departure Time).

```
library(OneR)
flight_delays_time <- as.data.frame(bin(flight_delays$CRS_DEP_TIME, nbins = 18, labels = c(1:18)))

flight_delays <- cbind(flight_delays_time, flight_delays)

flight_delays <- flight_delays %>% rename("CRS_DEP_TIME GROUP NO" = "bin(flight_delays$CRS_DEP_TIME, nbi

flight_delays %>%
  group_by(`CRS_DEP_TIME GROUP NO`) %>%
  summarise(CRS_DEP_TIME = length(`CRS_DEP_TIME GROUP NO`))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 18 x 2
##    `CRS_DEP_TIME GROUP NO` CRS_DEP_TIME
##    <fct>                             <int>
##  1 1                                   126
##  2 2                                   135
##  3 3                                   162
##  4 4                                   108
##  5 5                                    23
##  6 6                                   125
##  7 7                                    42
##  8 8                                   120
##  9 9                                   157
## 10 10                                   98
```

```
## 11 11                              292
## 12 12                               70
## 13 13                              182
## 14 14                              167
## 15 15                               85
## 16 16                              119
## 17 17                               53
## 18 18                              137
```

```
flight_delays <- flight_delays[, -2]
head(flight_delays)
```

```
##   CRS_DEP_TIME GROUP NO CARRIER DEP_TIME DEST DISTANCE    FL_DATE FL_NUM ORIGIN
## 1                    11      OH     1455  JFK      184 2004-01-01   5935    BWI
## 2                    13      DH     1640  JFK      213 2004-01-01   6155    DCA
## 3                     8      DH     1245  LGA      229 2004-01-01   7208    IAD
## 4                    14      DH     1709  LGA      229 2004-01-01   7215    IAD
## 5                     6      DH     1035  LGA      229 2004-01-01   7792    IAD
## 6                     3      DH      839  JFK      228 2004-01-01   7800    IAD
##   Weather DAY_WEEK DAY_OF_MONTH TAIL_NUM On Time Delayed
## 1       0        4            1   N940CA       1       0
## 2       0        4            1   N405FJ       1       0
## 3       0        4            1   N695BR       1       0
## 4       0        4            1   N662BR       1       0
## 5       0        4            1   N698BR       1       0
## 6       0        4            1   N687BR       1       0
```

Separated dataset into only predictors and then partitioned it into training and vaildation sets.

```
flight_delays_predictors <- flight_delays[, c(1, 2, 4, 8, 10, 14)] # created df with predictors.

set.seed(15)
index_train <- createDataPartition(flight_delays_predictors$DAY_WEEK, p = .6, list = FALSE)
flight_train <- flight_delays_predictors[index_train, ]
flight_valid <- flight_delays_predictors[-index_train, ]
head(flight_train)
```

```
##   CRS_DEP_TIME GROUP NO CARRIER DEST ORIGIN DAY_WEEK Delayed
## 1                    11      OH  JFK    BWI        4       0
## 2                    13      DH  JFK    DCA        4       0
## 6                     3      DH  JFK    IAD        4       0
## 7                     8      DH  JFK    IAD        4       0
## 8                    13      DH  JFK    IAD        4       0
## 9                    14      DH  JFK    IAD        4       0
```

Naive Bayes Model with training data.

```
naive_model <- naive_bayes(flight_train[, 1:5], flight_train[, 6], laplace = 1)
```

Predicting delayed/on-time flights on validation dataset.

```
predicted_flight_labels <- predict(naive_model, flight_valid, type = "class")
```

```
## Warning: predict.naive_bayes(): more features in the newdata are provided as
## there are probability tables in the object. Calculation is performed based on
## features to be found in the tables.
```

Destination and Origin Proportion and Count Tables

```
table(predicted_flight_labels, flight_valid$Delayed, dnn = c("Prediction", "Actual"))
```

```
##            Actual
## Prediction   0   1
##          0 684 148
##          1  25  21
```

```
dest_prop <- as.data.frame(naive_model$tables$DEST)
dest_count_delay <- as.data.frame(dest_prop[1:3, 3]*832)
dest_count_delay <- rename(dest_count_delay, Count = `dest_prop[1:3, 3] * 832`)
dest_count_ontime <- as.data.frame(dest_prop[4:6, 3]*46)
dest_count_ontime <- rename(dest_count_ontime, Count = `dest_prop[4:6, 3] * 46`)

dest_count <- rbind(dest_count_delay, dest_count_ontime)
dest_count_prop <- cbind(dest_prop, dest_count)

origin_prop <- as.data.frame(naive_model$tables$ORIGIN)
origin_count_delay <- as.data.frame(origin_prop[1:3, 3]*832)
origin_count_delay <- rename(origin_count_delay, Count = `origin_prop[1:3, 3] * 832`)
origin_count_ontime <- as.data.frame(origin_prop[4:6, 3]*46)
origin_count_ontime <- rename(origin_count_ontime, Count = `origin_prop[4:6, 3] * 46`)

origin_count <- rbind(origin_count_delay, origin_count_ontime)
origin_count_prop <- cbind(origin_prop, origin_count)

dest_count_prop <- rename(dest_count_prop, "Ontime/Delayed" = Var2, Proportion = "Freq")
dest_count_prop
```

```
##   DEST Ontime/Delayed Proportion      Count
## 1  JFK              0  0.1733833 144.254920
## 2  LGA              0  0.5501406 457.716963
```

```
## 3  EWR              0  0.2764761 230.028116
## 4  JFK              1  0.1755725   8.076336
## 5  LGA              1  0.4351145  20.015267
## 6  EWR              1  0.3893130  17.908397
```

```
origin_count_prop <- rename(origin_count_prop, "Ontime/Delayed" = Var2, Proportion = "Freq")
origin_count_prop
```

```
##   ORIGIN Ontime/Delayed Proportion      Count
## 1    BWI              0 0.05810684  48.344892
## 2    DCA              0 0.64292409 534.912840
## 3    IAD              0 0.29896907 248.742268
## 4    BWI              1 0.09541985   4.389313
## 5    DCA              1 0.50763359  23.351145
## 6    IAD              1 0.39694656  18.259542
```

This models predicts that JFK will have 8 delays, LGA will have 20 delays, and that EWR will have 18 delays. Also, that BWI will have 4 delays, DCA will have 23 delays, and that IAD will have 18 delays.

```
library(gmodels)
confusionMatrix(flight_valid$Delayed, predicted_flight_labels, dnn = c("Actual", "Prediction"))
```

```
## Confusion Matrix and Statistics
##
##       Prediction
## Actual   0   1
##      0 684  25
##      1 148  21
##
##                Accuracy : 0.803
##                  95% CI : (0.7751, 0.8288)
##     No Information Rate : 0.9476
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1231
##
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.8221
##             Specificity : 0.4565
##          Pos Pred Value : 0.9647
##          Neg Pred Value : 0.1243
##              Prevalence : 0.9476
##          Detection Rate : 0.7790
##    Detection Prevalence : 0.8075
##       Balanced Accuracy : 0.6393
##
##        'Positive' Class : 0
##
```

This model needs refining since incorrectly predicted that 148 flights would be delayed but were actually on-time giving it an "okay" sensitivity of 80% but still needs optimization. More

importantly, the specificity (predicting a flight will be delayed and is actually delayed) is 46% which is not good since we want this model to predict if a flight will be delayed.

```
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
```

```
predicted_flight_labels <- predict(naive_model, flight_valid, type = "prob")
```

```
## Warning: predict.naive_bayes(): more features in the newdata are provided as
## there are probability tables in the object. Calculation is performed based on
## features to be found in the tables.
```

```
roc(flight_valid$Delayed, predicted_flight_labels[, 2])
```

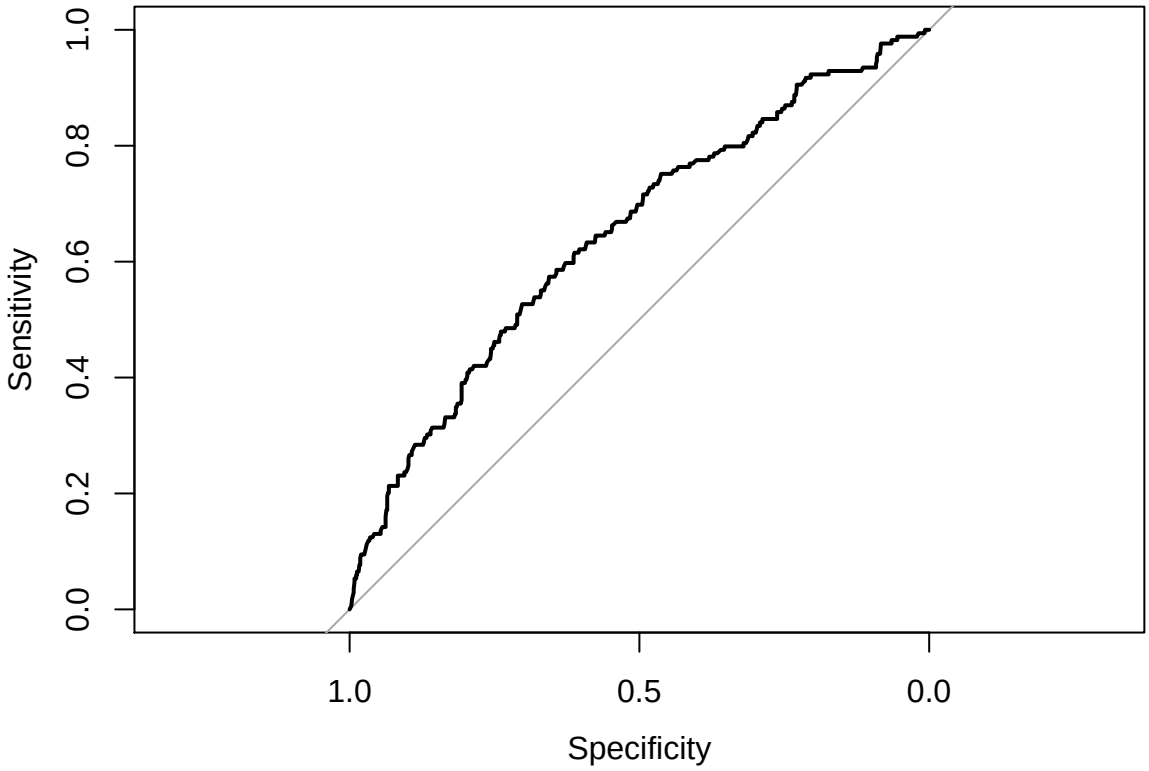
```
## Setting levels: control = 0, case = 1

## Setting direction: controls < cases

##
## Call:
## roc.default(response = flight_valid$Delayed, predictor = predicted_flight_labels[,     2])
##
## Data: predicted_flight_labels[, 2] in 709 controls (flight_valid$Delayed 0) < 169 cases (flight_valid
## Area under the curve: 0.646
```

```
plot.roc(flight_valid$Delayed, predicted_flight_labels[, 2])
```

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



The area under this curve is 64.6% which tells us that this model can be further improved since we want area to be reasonably close to 100%.