Assignment 3

Jared Stadden

3/1/2020

Preliminary data setup:

```
#Reading data into R
original = read.csv("C:\\Users\\jared\\Desktop\\FlightDelays.csv")

#loading libraries
library(caret)

## Loading required package: lattice

## Loading required package: ggplot2

library(ISLR)
library(e1071)

## Warning: package 'e1071' was built under R version 3.6.2

#create working dataset
flight<-original

#transforming some variables to facor
flight$DAY_WEEK <- factor(flight$DAY_WEEK)
flight$SCH_DEP_TIME <- factor(round(flight$CRS_DEP_TIME/100))

#keep only categorical predictors and target variable
flight <- flight[,c(10,14,8,4,2,13)]
```

Partitioning the data into Training and Validation sets

```
set.seed(123)

Index_Train<-createDataPartition(flight$Flight.Status, p=0.6, list = FALSE)
Train<-flight[Index_Train,]
Validation<-flight[-Index_Train,]
```

Creating Naive Bayes classifier with tuning

```
nb_model <- naiveBayes(Flight.Status~DAY_WEEK+SCH_DEP_TIME+ORIGIN+DEST+CARRIE
R,data = Train, preProc=c("BoxCox","center","scale"))

#predict delay status in validation set
Predicted_Validation_labels <- predict(nb_model,Validation)
```

Counts Table:

```
table(Train$Flight.Status,Train$DEST)

##
##           EWR JFK LGA
##   delayed 100  57 100
##   ontime  300 194 570
```

Proportion Table:

```
prop.table(table(Train$Flight.Status,Train$DEST),margin=1)

##
##                 EWR       JFK       LGA
##   delayed 0.3891051 0.2217899 0.3891051
##   ontime  0.2819549 0.1823308 0.5357143
```

Confusion Matrix:

```
library("gmodels")

## Warning: package 'gmodels' was built under R version 3.6.2

CrossTable(x=Validation$Flight.Status,y=Predicted_Validation_labels,prop.chisq = FALSE)

##
##
##    Cell Contents
## |-------------------------|
## |                       N |
## |           N / Row Total |
## |           N / Col Total |
## |         N / Table Total |
## |-------------------------|
##
##
## Total Observations in Table:  880
##
##
##                            | Predicted_Validation_labels
## Validation$Flight.Status |   delayed |    ontime | Row Total |
## -------------------------|-----------|-----------|-----------|
##                  delayed |        17 |       154 |       171 |
##                          |     0.099 |     0.901 |     0.194 |
##                          |     0.333 |     0.186 |           |
##                          |     0.019 |     0.175 |           |
## -------------------------|-----------|-----------|-----------|
##                   ontime |        34 |       675 |       709 |
##                          |     0.048 |     0.952 |     0.806 |
##                          |     0.667 |     0.814 |           |
##                          |     0.039 |     0.767 |           |
## -------------------------|-----------|-----------|-----------|
##             Column Total |        51 |       829 |       880 |
##                          |     0.058 |     0.942 |           |
## -------------------------|-----------|-----------|-----------|
##
##
```

ROC and plot of ROC curve:

```
Predicted_Validation_labels <- predict(nb_model,Validation,type="raw")

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

roc(Validation$Flight.Status, Predicted_Validation_labels[,2])

## Setting levels: control = delayed, case = ontime

## Setting direction: controls < cases
## Call:
## roc.default(response = Validation$Flight.Status, predictor = Predicted_Val
idation_labels[,     2])
##
## Data: Predicted_Validation_labels[, 2] in 171 controls (Validation$Flight.
Status delayed) < 709 cases (Validation$Flight.Status ontime).
## Area under the curve: 0.6383

plot.roc(Validation$Flight.Status,Predicted_Validation_labels[,2])

## Setting levels: control = delayed, case = ontime
## Setting direction: controls < cases
```