ManaliThe data are taken from Shmueli et al. (2010). The data set consists of 2201 airplane flights in January 2004 from the Washington DC area into the NYC area. The characteristic of interest (the response) is whether or not a flight has been delayed by more than 15 min.

The explanatory variables include three different arrival airports (Kennedy, Newark, and LaGuardia); three different departure airports (Reagan, Dulles, and Baltimore); eight carriers; a categorical variable for 16 different hours of departure (6 am to 10 pm); weather condition (0=good and 1 = bad); day of week (1 = Monday, 2 = Tuesday, 3 = Wednesday, … , 6 = Saturday and 7 = Sunday);

Here the objective is to identify flights that are likely to be delayed.

Follow the steps and answer the questions:

1. Import the csv file and check the data frame of the dataset. How many flights included in this data?

data <- read.csv(file.choose(), stringsAsFactor = F)

str(data)

1. Create a new schedule variable sched, where sched is the original schedule time variable divided by 100. Round the number using floor() function. How many flights scheduled to fly during “9” and “10”?

sched <- floor(data$schedtime / 100)

table(sched)

“9” has 108 flights

“10” has 100 flights

1. Recode the dayweek variable to dummy variable. If dayweek larger than 5, we recode it as 1, otherwise we code it as 0. Inspect the distribution of the new dayweek variable. How many flights travel during Saturday and Sunday?

data$dayweek <- factor(data$dayweek)

levels(data$dayweek) <- c("0", "0", "0", "0", "0", "1", "1")

summary(data$dayweek)

503 flights travel during Saturday and Sunday

1. Omit unused variables (variable 1, 3, 5, 6, 7, 11 and 12)

data <- data[-12]

data <- data[-11]

data <- data[-7]

data <- data[-6]

data <- data[-5]

data <- data[-3]

data <- data[-1]

1. If weather is good, how many flights were delayed in total?

table(data$delay, data$weather)

396 flights are delayed while the weather is good

1. Which origin airport operates the most flights? Which destination airport manages the most flights? Which origin airport has the highest delay rate (delay flights in this airport/total flights in this airport)? Which flight carrier has the most flights traveling?

table(data$origin)

DCA operates most flights

table(data$dest)

LGA manages the most flights

prop.table(table(data$origin, data$delay)\*2201)\*100

DCA has the highest delay rate at roughly 10%

table(data$carrier)

DH carrier has the most flights traveling

1. Recode origin: if origin airport is “DCA”, recode value to 1, 0 otherwise;  
   Recode destination airport: if destination is “LGA”, recode value as 1, else 0;

Recode carrier: if carrier is “DH”, recode carrier as 1, else recode it as 0.

Recoded origin:

data$origin <- factor(data$origin)

str(data$origin)

levels(data$origin) <- c("0", "1", "0")

Recoded destination:

data$dest <- factor(data$dest)

str(data$dest)

levels(data$dest) <- c("0", "0", "1")

Recoded carrier

data$carrier <- factor(data$carrier)

str(data$carrier)

levels(data$carrier) <- c("0", "1", "0", "0", "0", "0", "0", "0")

1. Randomize the sample. Generate a normalization function and normalize the variables except the class variable delay from randomized sample. Why do you need to normalize features?

minmax <- function(x){

ifelse(is.numeric(x),

return( ( x - min(x) ) / ( max(x) - min(x) ) ),

return (x) )

}

data <- data.frame(lapply(data, minmax))

str(data)

We normalize the features so that the algorithm isn’t dominated by large variables.

1. Create training and testing sample. Use the first 60% of flights as training sample and the rest as testing sample. Round the number of training sample using floor(). How many flights do you include in your testing sample?

nRows <- nrow(data)

nRows

trainSize <- floor(nRows \* 0.6)

trainSize

set.seed(1234)

train.index <- sample(1:nRows, trainSize, replace = F)

train.index[1:10]

data.train <- data[train.index,]

data.test <- data[-train.index,]

881 flights are included in the testing sample

1. Build a knn model with k = 10 to predict the class delay of testing sample. How many flights have been misclassified as on time but they actually belong to delay?

library(class)

predicted.out <- knn(data.train[-6], data.test[-6], data.train$delay, k = 10)

predicted.out[1:10]

library(gmodels)

CrossTable(data.test$delay, predicted.out, dnn = c("Actual", "Predicted"))

174 flights have been misclassified (False Positives) as on time but were actually delayed

Naïve Bayes

1. Build a Naïve Bayes model with laplace = 1 to predict the class delay of testing sample. How many flights have been misclassified as on time but they actually belong to delay?

model <- naiveBayes(data.train[-6], data.train$delay, laplace = 1)

predicted.test.bayes <- predict(model, data.test[-6])

CrossTable(data.test$delay, predicted.test.bayes, dnn=c("Actual", "Predicted"))