The 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?

Ans: 2201 flights are included in this dataset

Code:

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

str(data)

2. 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”?

Ans:

flights scheduled to fly during “9”  à 108 flights

flights scheduled to fly during “10” à 100 flights

Code:

data$sched <- floor( data$schedtime / 100 )

summary(data$sched)

CrossTable(data$sched)

3. 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?

Ans: 503 flights travelled during sat and sun

Code:

data$dummy = ifelse(data$dayweek > 5,1,0)

data$dummy

CrossTable(data$dummy)

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

Ans:

'data.frame': 2201 obs. of 8 variables:

$ carrier: chr "OH" "DH" "DH" "DH" ... $ dest : chr "JFK" "JFK" "LGA" "LGA" ... $ origin : chr "BWI" "DCA" "IAD" "IAD" ... $ weather: int 0 0 0 0 0 0 0 0 0 0 ... $ dayweek: int 4 4 4 4 4 4 4 4 4 4 ... $ delay : chr "ontime" "ontime" "ontime" "ontime" ... $ sched : num 14 16 12 17 10 8 12 16 17 21 ... $ dummy : num 0 0 0 0 0 0 0 0 0 0 ...

Code: data\_flights<- subset(data, select = -c(1,3, 5, 6, 7, 11,12) ) str(data\_flights)

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

Ans: 396 flights were delayed in total

Code:

CrossTable(data\_flights$weather,data\_flights$delay)

6. 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?

Ans:

DCA operates most number of flights with 1370 flights

LGA airport manages the most flights with 1150 flights

BWI has highest number of delays with ( 37/145=25.51%)

DH has the most flights travelling  with 551 flights

Code:

CrossTable(data\_flights$origin)

CrossTable(data\_flights$dest)

CrossTable(data\_flights$origin,data\_flights$delay)

CrossTable(data\_flights$carrier)

7. 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.

Ans:

'data.frame':   2201 obs. of  8 variables:

 $ carrier: num  0 1 1 1 1 1 1 1 1 1 ...

 $ dest   : num  0 0 1 1 1 0 0 0 0 0 ...

 $ origin : num  0 0 0 0 0 0 0 0 0 0 ...

 $ weather: int  0 0 0 0 0 0 0 0 0 0 ...

 $ dayweek: int  4 4 4 4 4 4 4 4 4 4 ...

 $ delay  : chr  "ontime" "ontime" "ontime" "ontime" ...

 $ sched  : num  14 16 12 17 10 8 12 16 17 21 ...

 $ dummy  : num  0 0 0 0 0 0 0 0 0 0 ...

Code:

data\_flights$origin<-ifelse(data\_flights$origin=="DCA",1,0)

data\_flights$dest<-ifelse(data\_flights$dest=="LGA",1,0)

data\_flights$carrier<-ifelse(data\_flights$carrier=="DH",1,0)

str(data\_flights)

8. 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?

Ans : We need normalization so that all features can contribute equally.

Randomized sample:

     carrier dest origin weather dayweek   delay sched dummy  
798        0    1      1       0       1  ontime    15     0  
1816       0    1      1       1       1 delayed     7     0  
1971       1    0      0       0       3  ontime     8     0  
2169       0    1      1       0       6  ontime    16     1  
1289       0    1      1       0       1  ontime    18     0  
1891       0    1      1       1       2 delayed    14     0

Randomized normalized sample:

 carrier dest origin weather   dayweek      sched dummy  
1       0    1      1       0 0.0000000 0.60000000     0  
2       0    1      1       1 0.0000000 0.06666667     0  
3       1    0      0       0 0.3333333 0.13333333     0  
4       0    1      1       0 0.8333333 0.66666667     1  
5       0    1      1       0 0.0000000 0.80000000     0  
6       0    1      1       1 0.1666667 0.53333333     0

Code:

dt1<-data\_flights

set.seed(125)

randomize <- sample(1:nrow(dt1),nrow(dt1), replace = F)

dt2<- dt1[randomize,]

head(dt2)

rmd <- function(x){  
  ifelse(is.numeric(x), return ((x-min(x))/(max(x)-min(x))), return(x))  
}  
dt3<- data.frame(lapply(dt2[,c(-6)], rmd))  
summary(dt3)  
head(dt3)

9. 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?

Ans: 881 testing sample

Code:

nrows <- nrow(dt3)

nrows

train.size <- floor(nrows \* 0.6)

train.size

#training set

set.seed(45678)

train.index <- sample(1:nrows, train.size, replace = F)

dt.train <- dt3[train.index,]

str(dt.train)

#testing set

set.seed(45678)

test.size<-nrows-train.size

test.index<-sample(1:nrows,test.size,replace = F)

dt.test<-dt3[test.index,]

str(dt.test)

10. 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?

Ans: 8 flight have been misclassified as on time but they actually belong to delay

Code:

library(class)

library(gmodels)

predicted.test <- knn(train=dt.train[-6], test=dt.test[-6],

                       cl=dt.train$delay, k=10)

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

11. Naïve Bayes

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?

Ans: 0 flight have been misclassified as on time but they actually belong to delay

Code:

library(e1071)

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

predicted.test.naive <- predict(model, dt.test[-6])  
CrossTable(dt.test[[6]], predicted.test.naive, dnn=c("Predicted", "Actual"))