**Statistics 6620 : Statistical Learning**

**Project :1**

**Analyzing Airline Data using Classification and Prediction Methods**

**Group**

**Ruchita Trivedi**

**Ramnadh Ambattpudi**

**Naga Ambadipudi**

**Abstract**

Machine Learning methods are very useful in conducting data analysis and understand the patterns in the data. Both classification and prediction algorithms have their own merits in predicting labels/classes or forecast. Machine learning algorithms were used on airlines data for predicting target features

**Introduction**

Delay is very common feature in airline industry. They are several factors might influence the airline delay, which includes, weather, mechanical failures, airport congestion, airplane make and model. After September 11, 2011 terrorism threats also become one of the factors contributing for delays. The objective of this analysis is to predict delay of an airline based on features associated with the flight.

**Data Description:**

The Federal Aviation Administration of United States of America collects a comprehensive set of information on air travel. This data spanned more than 21 years starting 1987. American Statistical Associations hosting this data for research and analysis purposes. The data was downloaded from ASA’s statistical Graphics data expo symposium website. Even though, data is available for multiple years, for computation simplicity and hardware limitations, our analysis is restricted to a single year 1987.

**Methodology**

This analysis is outlined on STAT 6620 group project objectives. Here R-software was used for data manipulation, modeling and prediction.

**Analysis is divided into following parts**

1. **Importing and pre-processing the data**
2. **Generating Summary Statistics**
3. **Creating target Variable**
4. **Applying kNN Classification algorithm**
5. **Prediction using**

**5.1 Regression Tree algorithm,**

* 1. **Logistic Regression to find the output probabilities**

**5.3 Finding ROC and AUC based on results of Logistic Regression**

**6.Summary of findings**

1. **Importing and pre-processing the data.**

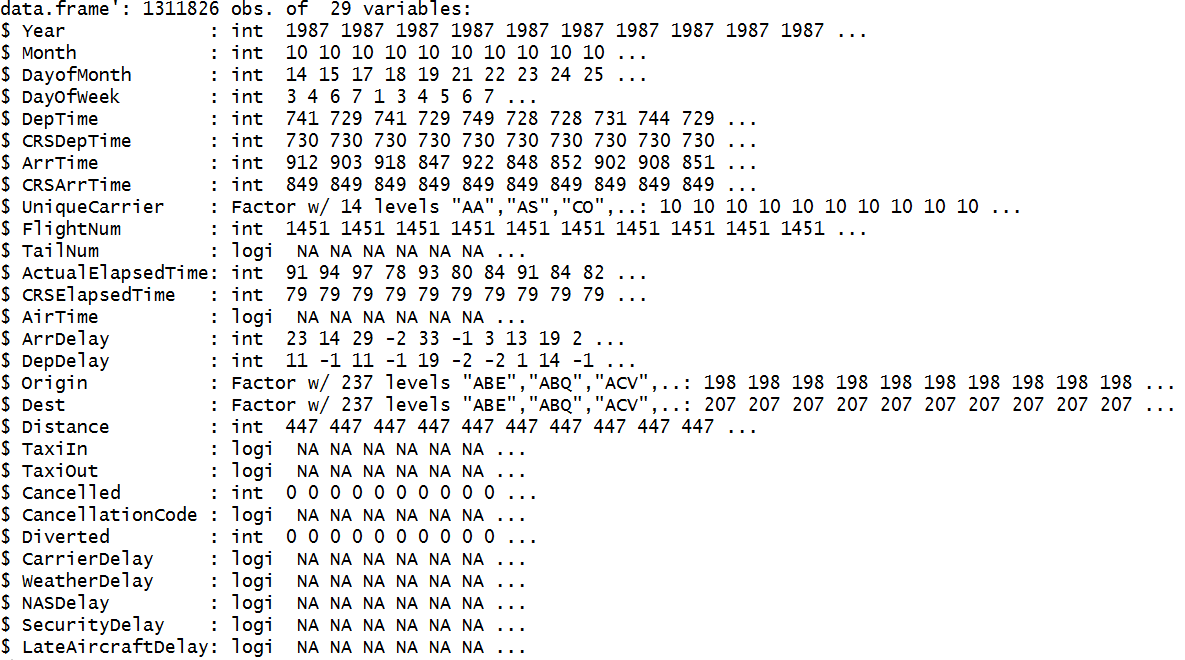
**1.1 Data Import**

Data is downloaded from the website is in a highly compressed zip format known as bzip2. To import data into r environment, read.csv function was used. An additional .bz2 was added to import the compressed data without extractio*n.*

**1.2 Pre-processing the data**

Before processing to modeling phase, data needs to be checked for missing and redundant variables. Some of the missing values will influence the outcome; this is more evident for prediction models.

We can understand the structure of the data from using str function in r

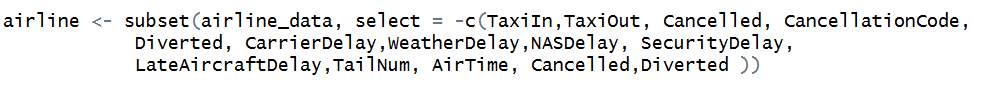


The output of the str function can be summarized as below

**Table1: Variable Classification**

|  |  |  |
| --- | --- | --- |
| **Variable Type** | **Number of Variables** | **Variable Name** |
| Factor | 3 | UniqueCarrier, Origin, Destination |
| Integer | 16 | Year, Month, DayofMonth, DayOfWeek, DepTime, CRSDepTime, ArrTime, CRSArrTime, FlightNum, ActualElapsedTime, CRSElapsedTime, ArrDelay, DepDelay, Distance, Cancelled, Diverted |
| Logical | 10 | TailNum, AirTime, TaxiIn, TaxiOut, CancellationCode, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay, LateAircraftDelay |

The data set contains 1,311,826 examples and 29 features. The features of the airline data can be categorized as Factor, Integer and Logical. There are 16 integer variables, 3 factor variables and 10 logical variables in the data set. From data frame structure we can observe that some of the features contains either contains only missing values or unique values, these variables can be excluded from the modeling data frame.

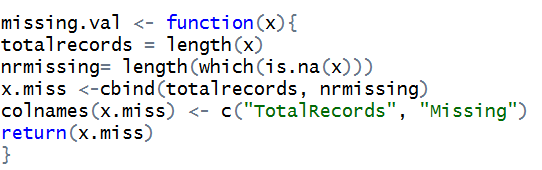


The reduced data set contains 1,311,826 examples and 17 features.

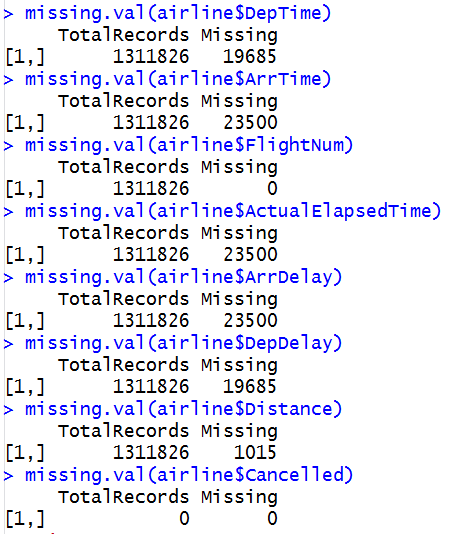
**1.3 Missing Value Treatment**

Missing values influence the model outcome, especially in prediction models such as logistic regression. Logistic regression excludes entire example, if there exist a missing value in any feature of the example. So if any feature contain majority of the missing values, it needs a special treatment or entire feature needs to be excluded from the model.

A user defined function was created to identify the missing values in the numeric data



**R- Output**



From r output, we can observe that missing values are varying from variable to variable, but maximum number of missing example in any numeric feature is 23,500. This number is very small considering to 1.3 million data available for training and modeling. So these records can be excluded from the modeling data set

To remove examples with missing values to at least one of the features we can use complete.cases function in r

The new data set contains 1,287,333 examples and 15 features. None of the features in this dataset contains missing examples

**2. Summarize the data.**

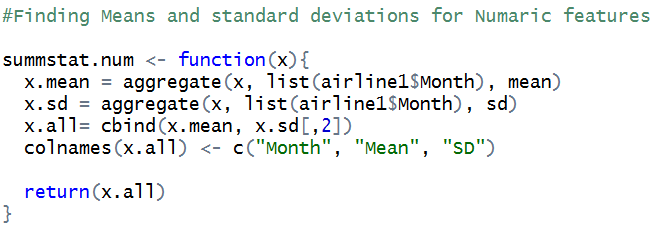
After excluding the redundant variables, there are 15 features, there are 13 numeric variables and 3 categorical variables

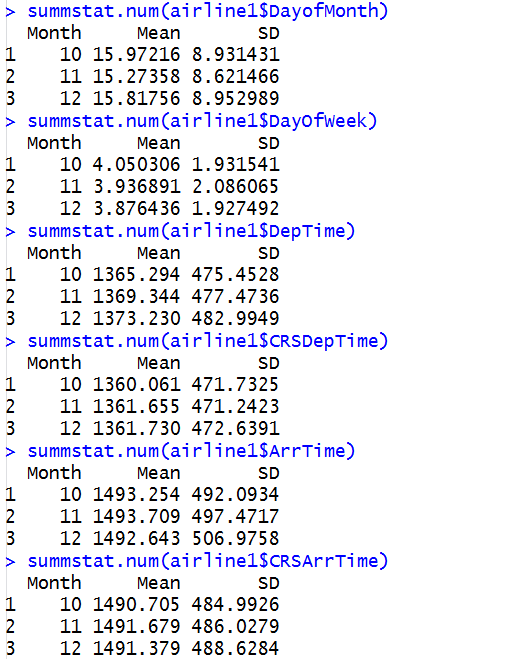
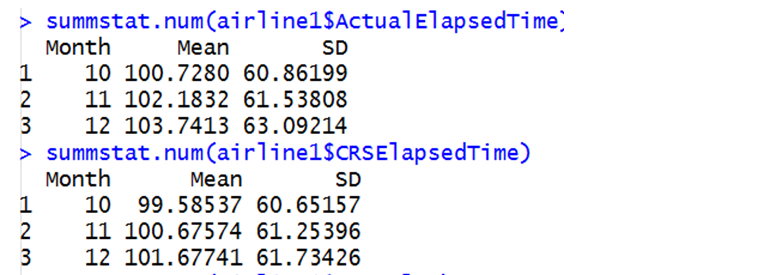
**Table2: Variable Classification after Variable Reduction**

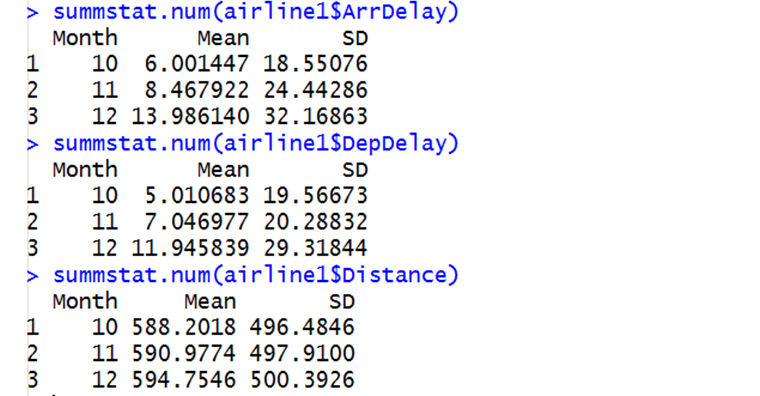
|  |  |  |
| --- | --- | --- |
| **Variable Type** | **Number of Variables** | **Variable Name** |
| Factor | 3 | UniqueCarrier, Origin, Destination |
| Integer | 16 | Year, Month, DayofMonth, DayOfWeek, DepTime, CRSDepTime, ArrTime, CRSArrTime, ActualElapsedTime, CRSElapsedTime, ArrDelay, DepDelay, Distance, Cancelled, Diverted |

In 1987 data we have information on only last quarter i.e. information is available only for the months of October, November and December.

**2.1 Mean and Standard Deviation of Numeric Variables by Month**

A user defined function was defined in r to calculate Mean and Standard deviation of numeric features by month

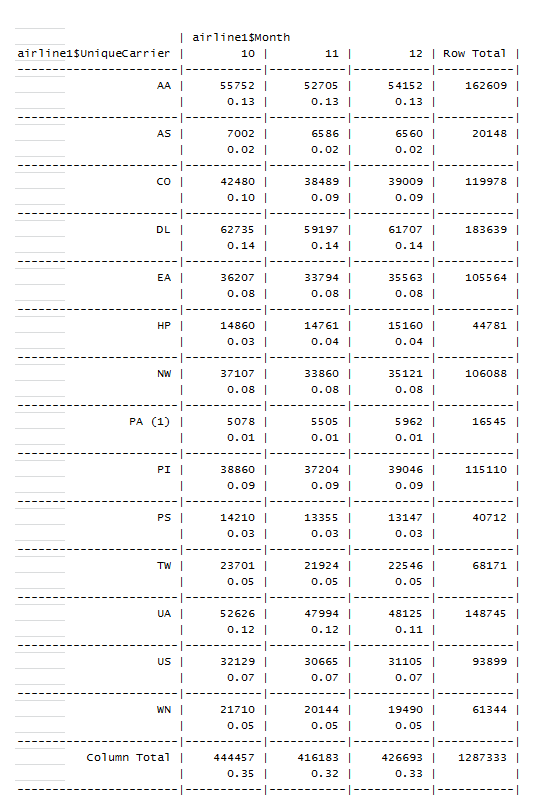




**2.2 Counts and Relative Frequencies of Categorical Variables by Month**

Cross table function is used to calculate relative frequencies. The factors such as Origin and Destination have 237 levels, so we are going to list only partial output.

**Frequencies of Airline Name by Month**

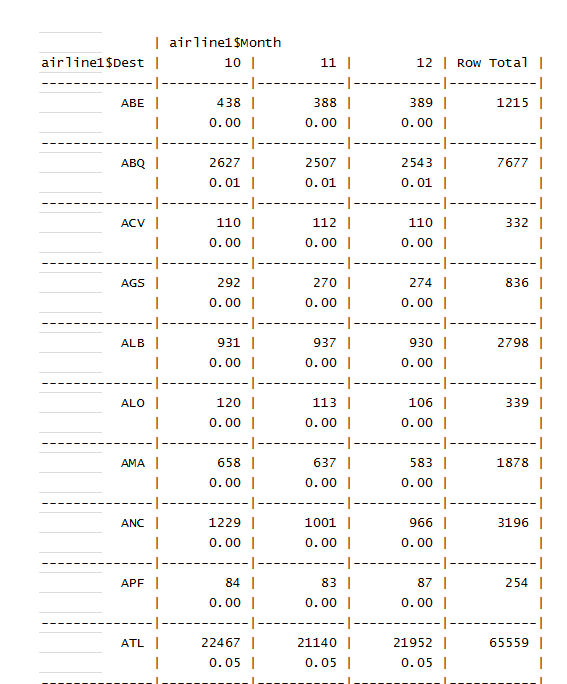


To consider the cataegoriacal variables in our analysis, we converted them to numerical values.

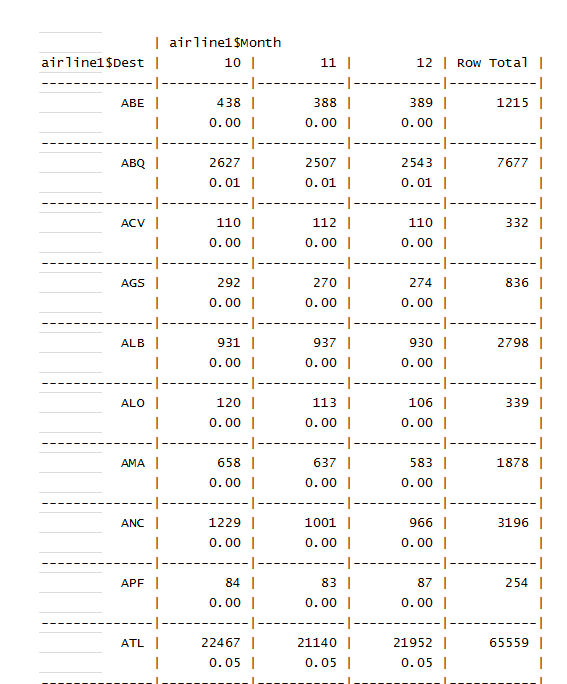
airline1$Origin<-as.numeric(airline1$Origin)

airline1$Dest<-as.numeric(airline1$Dest)

**Frequencies of Destination Airport by Month**



**Frequencies of Origen Airport by Month**



**3.1: Creating Variable ArrivedLate**

If an aircraft missed its scheduled arrival time (CRSArrTime) is treated as delayed. According to instrument flight rules (IFR) Federal Aviation Administration, any airline delayed for 15 minutes or more is considered reportable delay.

For the purpose of this analysis, target variable ArrivedLate is created based on IFR rules of Federal Aviation Administration. Any flight delayed more than 15 minutes or higher was classified as arrived late using if-else condition.

Summary Statistics of ArrivedLate Variable



**4: Classification:**

Classification methods relay on distance methods, so all non-numeric features needs to be excluded from the model. The numeric variables need to be normalized in order to reduce scale impacts on the model.

**4.2 Dividing data into Train and Test**

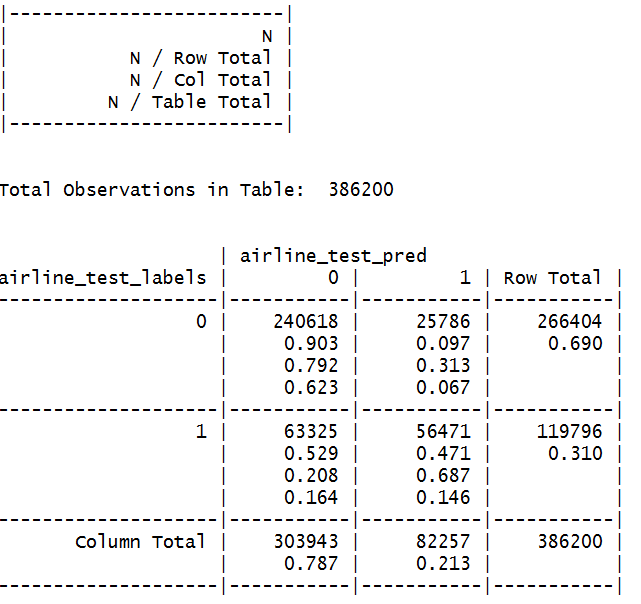
In order to evaluate the performance of the model, data was divided between train and test data sets in 70:30 ratios. Model training was performed on 70% data and validated the classification on test data set

**4.3 Applying kNN algorithm**

kNN algorithm was applied on train data set and predicting the outcome of test data. Here we have taken 5 nearest neighbors for calculation of distances.

**4.4 Validating kNN algorithm on test data set**

Labels of predicted data were compared with actual labels of test data using cross table function.



The outcome of confusion matrix gives an accuracy of 83%. This accuracy can be improved by increasing the number of neighbors or trying different k values. It can also be improved by adding additional information such as airline . But these variables are defined as character, so variable binning needs to be performed based on information value. We are keeping this analysis for subsequent improvements.

**Part 5: Prediction**

**5.1 Correlation Matrix:**A great way to explore is to use a pairwise correlation matrix. This will measure the correlation between every combination of your variables.. The measure value can range between **1** and **-1**, where **1** is perfectly correlated, -**1** is perfectly inversely correlated, and **0** is not correlated at all.

Cor.prob and flatten.square.matrix functions were borrowed. cor. prob returns a corelation matrix and also returns p values,so it makes a large matrix ## of all variables. flattenmatrix simply takes the huge matrix from above and break them into four colouns.

ie row names, coloun names, correlation and p values

selectedSub <- subset(corList, (abs(cor)& j == 'ArrDelay'))

> head(selectedSub,20)

i j cor p

44 ActualElapsedTime ArrDelay 0.12003688 0

39 DepTime ArrDelay 0.11682889 0

40 CRSDepTime ArrDelay 0.08853443 0

42 CRSArrTime ArrDelay 0.08663669 0

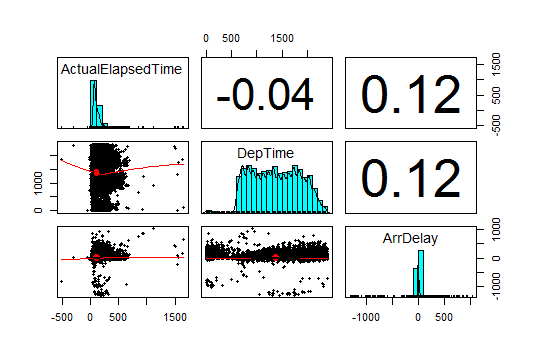
37 DayofMonth ArrDelay 0.06933089 0

41 ArrTime ArrDelay 0.06430147 0

43 FlightNum ArrDelay 0.03963065 0

38 DayOfWeek ArrDelay -0.03896971 0

45 CRSElapsedTime ArrDelay 0.03559619 0



We can see that the highest correlation is that of ActualElapsedTime and DepTime with the target class.

**5.2 Regression Tree :**Fit a tee model based on training data using all other variables against ArrDelay

m.rpart1 <- rpart(DepDelay[1:1000] ~DayofMonth[1:1000]+DayOfWeek[1:1000]+DepTime[1:1000], data = airline\_reg\_train)

m.rpart1

|  |
| --- |
| m.rpart1  n=992 (900141 observations deleted due to missingness)  node), split, n, deviance, yval  \* denotes terminal node  1) root 1000 255481.100 7.930000  2) DepTime[1:1000]< 2035.5 922 170234.500 6.395879  4) DepTime[1:1000]< 1649.5 607 67558.360 4.678748 \*  5) DepTime[1:1000]>=1649.5 315 97437.540 9.704762  10) DayOfWeek[1:1000]< 2.5 95 8087.937 4.147368 \*  11) DayOfWeek[1:1000]>=2.5 220 85148.600 12.104550  22) DayofMonth[1:1000]< 21.5 148 42271.860 9.087838 \*  23) DayofMonth[1:1000]>=21.5 72 38761.280 18.305560  46) DayOfWeek[1:1000]>=5.5 21 3669.238 7.476190 \*  47) DayOfWeek[1:1000]< 5.5 51 31615.180 22.764710  94) DayofMonth[1:1000]>=29.5 11 496.000 7.000000 \*  95) DayofMonth[1:1000]< 29.5 40 27633.600 27.100000 \*  3) DepTime[1:1000]>=2035.5 78 57426.680 26.064100  6) DepTime[1:1000]< 2122.5 57 34282.000 22.000000  12) DayofMonth[1:1000]< 21.5 36 13775.000 18.833330 \*  13) DayofMonth[1:1000]>=21.5 21 19527.140 27.428570  26) DayofMonth[1:1000]>=24 14 3694.857 15.714290 \*  27) DayofMonth[1:1000]< 24 7 10068.860 50.857140 \*  7) DepTime[1:1000]>=2122.5 21 19647.810 37.095240  14) DepTime[1:1000]>=2194 9 2263.556 7.777778 \*  15) DepTime[1:1000]< 2194 12 3846.917 59.083330 \*  We already know from correlation matrix that DepDelay was most correlated with the target class  Class which obviously makes scence. Now, how do we predict what factors lead to DepDelay,  To make our model helpull to passengers and airline companys.  From the output of regression tree we conclude that:  1-If the DepTime is less than 10:30 pm , the delays will be less.  2-If the DayofWeek is later than Wednesday the delays will be more.  3-If the flight is in the last week of month(after 21st ) the delays will be maximum. |
|  |
| |  | | --- | | rpart.plot(m.rpart1, digits = 3)    a few adjustments to the diagram, prunnig etc:  rpart.plot(m.rpart1, digits = 4, fallen.leaves = TRUE, type = 3, extra = 101)      From output of the model, DepDelay turns out to be the most important variable in the analysis,  scheduled arrival and scheduled departure timings, actual arrived time  and time spent on travel were other key variables in predicting the arrival delay.  **Model Validation:** Regression tree model was validated by apply model on train dataset. The  values of actual delay with predicted delay was compared using correlation check.  mean(p.rpart1-airline\_reg\_test$ArrDelay)^2  [1] 0.9910168  > cor(p.rpart1, airline\_reg\_test$ArrDelay)  [1] 0.8851464  > summary(p.rpart1)  Min. 1st Qu. Median Mean 3rd Qu. Max.  -648.200 1.545 1.545 12.810 14.310 205.500  *>* summary(airline\_reg\_test$ArrDelay)  Min. 1st Qu. Median Mean 3rd Qu. Max.  -1019.00 -2.00 6.00 13.81 19.00 940.00  From correlation check and summary comparison, we can infer that, the regression tree model  did a reasonable job to predict the Arrival Delay, as we got a high correlation of .88 between predicted and actual values. Also Mean values are quiet close .We tried to include maximum number of variables in our analysis by converted converting categorical variables like Origin, Dest . into numeric. Also we included variables like DayOfMonth, DayOfWeek. | |

**5.3: Fitting ROC using Logistic Regression:** For finding the ROC curve ,we need a classification algorithm with binary prediction of target class ie Delay.So we have fitted the logistics regression as follows,

Slicing our data into testing and training data

d1\_train <- d1[1:800, ]

d1\_test <- d1[801:999, ]

d1\_train\_labels <- d1[1:800, 1]

d1\_test\_labels <- d1[801:999, 1]

Delay\_model<-glm(Delay~DayofMonth+DayOfWeek+DepTime+CRSDepTime+ ArrTime+CRSArrTime+

FlightNum+ActualElapsedTime+CRSElapsedTime+ArrDelay+DepDelay+

Origin+Dest+Distance, data=d1\_train,family=binomial)

summary(Delay\_model)

model\_pred\_probs<-predict(Delay\_model,d1\_train,type="response")

model\_pred\_Delay<-rep("0",800)

model\_pred\_Delay[model\_pred\_probs>.5]<-"1"

table(model\_pred\_Delay,d1\_train\_labels)

d1\_train\_labels

model\_pred\_Delay 0 1

0 304 76

1 70 350

mean(model\_pred\_Delay!=d1\_train\_labels)

[1] 0.1825

The Above lines predict a categorical variable from the fitted model using an "unseen" testing data. And create the confusion matrix to compute the miss-classification error rate.We see that we get a

Error rate of .182

install.packages("ROCR")

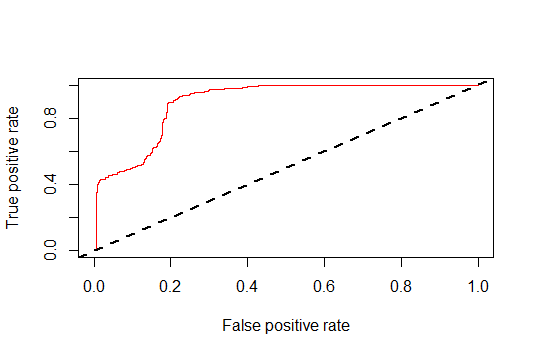
library(ROCR)

.pred <- prediction(model\_pred\_probs,d1\_train\_labels)

perf <- performance(pred, measure = "tpr", x.measure = "fpr")

plot(perf, col=rainbow(10))

abline(a = 0, b = 1, lwd = 2, lty = 2)



The predictions are your continuous predictions of the classification, the labels are

the binary truth for each variable to confirm the performance quantitatively, we need AUC, To do so, we first need to create another performance object, this time specifying measure = "auc",

perf.auc <- performance(pred, measure = "auc")

unlist([perf.auc@y.values](mailto:perf.auc@y.values))

[1] 0.8976802

**6.Summary of findings:** Overall, we did some useful results in our analysis. With the help of correlation matrix, we found several variables that were statistically significant, DepDelay, ArrTime

Etc. We found out some reasons behind the U.S Air Fights Arrival Delay.

The fastest and easiest way to make decision about our dataset is to apply Decision tree mechanism where we organized our data hierarchically. So, we used Rpart algorithm for making decision tree. From the diagram, we conclude that normally at night after 10:30PM the delays are rela-

tively higher. And normally on days after wednesday like Saturday and Sunday delays are

possibly higher compared with weekdays, which makes sense. Also towards the end of month the delays are higher.

In classi\_cation we used knn techniques. In k-nearest neighbor tech-

nique we found that approximately 83 percent of data are correctly classified.

Same thing we found in our confusion matrix