Group 113: Flight Delay Analysis

First Name	Last Name	Monday or Tuesday class
Vighnesh Sanjay	Sawant	Monday class
Prateek Ravindra	Deshmukh	Monday class
Siddharth	Singh	Monday class

Table of Contents

1. Introduction

We have performed a **Delay Analysis** associated with **Universal Airlines** which are originating from **Washington D.C. Area to New York City Area.** The reason why we have chosen this area because we have observed that there are some frequent delays occurring in between Washington D.C. area and New York City area. In our Delay Analysis, we have provided with some statistics, visualization and meaningful outcomes, also discussing about the factors to what causes the delay and prognosticate the future delays of airline which will prove as a benefactor to the universal airlines to avoid the delays in the future to achieve a better customer satisfaction.

2. Data

To perform the Delay Analysis, we have gathered or collected data from the authentic source of **Bureau of Transportation – Research and Innovative Technology Administration (RITA)**. The link to Bureau of Transportation is https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=236 Since the data which we collected from this source was restricted only to a certain domain and that's the reason why we could not perform furthermore task or analysis. So, to perform further analysis, we have gathered the data from **Kaggle.com** too. The link to dataset from Kaggle is https://www.kaggle.com/usdot/flight-delays/data By this we have consolidated various data to achieve our complete analysis on flight delays. We also have incorporated many other factors in the data set like SCHEDULED_TIME, ELAPSED_TIME, AIR_TIME, SCHEDULED_ARRIVAL, ARRIVAL_TIME, ARRIVAL_DELAY, delaytimeweathergood, delaytimeweatherbad and DELAY_LABEL. The data have been stored in a CSV format (.csv) file. We have maintained various CSV files for various analysis. Fortunately, the data set of ours didn't have any missing values, so we didn't do the data pre-processing. In our data set we have **2202 rows of data and 26 columns or attributes**. The attributes are as follows:

SCHEDULED_DEPARTURE – time at which flight departure is scheduled
carrier – name of carrier
DEPARTURE_TIME – time at which flight departs
dest – name of destination airport
distance – total distance between origin and destination
DATE – date at which flight takes off
flightnumb – flight number
origin – name of origin airport
weather – 0 when the weather is good and 1 when the weather is bad
dayweek – day of a week (1 for Monday, 2 for Tuesday, 3 for Wednesday till 7 for Sunday)
daymonth – day of a month (1 for January, 2 for February till 12 for December)
tailnu – tail number of the flight
delay – flight status (on time = 0 and delay = 1)
delaytimeweathergood – time required to reach destination when weather is good
delaytimeweatherbad - time required to reach destination when weather is bad
DEPARTURE_DELAY – time between scheduled departure and actual departure
SCHEDULED_TIME – land time amount needed for the flight trip
ELAPSED_TIME – the time between taxi_in, taxi_off and air time.
AIR_TIME – total time between wheels off and wheels on.
SCHEDULED_ARRIVAL – time at which flight is scheduled for arrival
ARRIVAL_TIME – time at which the flight arrives at destination airport
ARRIVAL_DELAY – time between scheduled arrival time of flight and actual arrival time
YEAR – from year 1991 to year 2012
MONTH – month of a year (1 for January, 2 for February, 3 for march, 4 for April and so on till 12 for December)
DAY – day of month (1^{st} to 30^{th} of the month)
DELAY_LABEL – labelling the category of delay (high, medium, low, on time)
Most important, **we have split the data into 80 % training and 20 % testing**.

3. Problems to be Solved

- Our primary research goal is to predict the **ARRIVAL_DELAY** as a function of the other variables.
- Our secondary research goal is to determine what are the core factors which affects the schedule of the universal airlines or simply what factors causes the flights to get delayed and applying the fundamental concepts learned in the ITMD 527 course/class like performing regression analysis, building and evaluating different models with the help of R programming language.
- We have also performed some of the **descriptive statistics** to make understand this delay analysis to a common man by providing Graphs, visualization and meaningful outcomes.
- We can carry out the **Inferential Statistics like hypothesis testing** on the data like delay time when weather is bad will be more than delay time when the weather is good (two sample mean).
- We can carry out **analysis of variance (ANOVA)** to predict the arrival time.
- We can use **Time Series analysis** and determine whether it is stationary or not i.e. any visible trend over time, whether it has constant mean and variance or varying with time and by plotting auto correlation function. Also, most importantly, we will be able to predict or forecast the future ARRIVAL_TIME of the universal airlines.
- We will be also predicting the flight status i.e. whether the flight or carrier will be on time or delayed by performing different **classification techniques like K – Nearest Neighbor (KNN) and Naïve Bayes Classification** and further evaluating these two classification techniques and finding out which one is the best in terms of accuracy to predict the ARRIVAL_TIME of a flight.

4. Data Processing

Fortunately, the data set of ours **didn't have any missing values**, so we didn't do the data pre-processing.
Secondly, we have split the entire data set into training and testing. **80% of data set will be used as training data set** and **remaining 20% will be testing data set**. Training data set is used to build the model i.e. training data is passed to an algorithm which builds a model and testing data is used to test the accuracy of the model. We will observe this ahead in model selection, linear regression, classification.

5. Methods and Process

In the flight delay analysis that we performed, we have used multiple analysis technique or methods. Each analysis technique that we used involves different process. We have implemented **descriptive statistics** such as plotting graphs, Density chart, Box plot analysis, scatter plot analysis.

1. Considering the first descriptive statistics such as graphs, we have plotted a graph of Delay vs on time. From the graph that we plotted, we can see that the **no of flights which are on time are more than the no of flights which are delayed.**
2. In the density chart of arrival delay, we have plotted arrival delay in minutes and density of arrival delay, in that we can see that **there are more no of flights which are on time**, also, **there are more no of flights which have arrival delay between 1 to 50 minutes**. Another observation from the density chart is that there are few flights which are having arrival delay of 100 to 150 minutes.
3. In the box plot analysis, we have plotted box plot of different carriers and arrival delay in minutes. We have observed **that carrier MQ has the highest arrival delay as compared to other carriers like DH and DL.**
4. In the scatter plot analysis, we have plotted a scatter plot between departure delay and arrival delay for three different carriers like MQ, DH and DL. We have observed that it shows **linear relationship** since as the **departure delay increases, the arrival delay also increases.**

5. Next, we have plotted a graph between arrival delay vs no of days in a week for three different carriers like MQ, DH and DL. Here, we observed that irrespective of any carrier, **all the carriers have a maximum arrival delay on Friday** i.e. on Friday carriers are delayed the most.

HYPOTHESIS TESTING:

In our further analysis on flight delay, we have done **inferential statistics like Hypothesis Testing**. We have made the claim that delay time is more when weather is good as compared to when weather is bad. Since it involves to parameters i.e. delay time weather good and delay time weather bad, it is a two sample mean.

μ_1=Delay time when weather is good

μ_2=Delay Time when weather is bad

H_0(Null Hypothesis): $\mu_1 > \mu_2$

H_a (Alternate Hypothesis): $\mu_1 < \mu_2$

We install BSDA library and use the z.test function. We set confidence level as 95%. At 95% confidence level, the p-value is 2.2e-16 which is less than alpha i.e. 0.05, indicating that we must reject null hypothesis and accept alternate hypothesis.

Therefore, we can conclude that time of delay is less when weather is good as compared to time of delay when weather is bad

LINEAR REGRESSION:

In our further analysis on flight delays, we did **linear regression**. we have done model selection to get the best possible linear model for prediction. We performed model selection on training data set i.e. we have built the model on 80% training data set by using three approaches:

Approach I: Backward Elimination by p-value:

In this method, first we must build model using all the variables, then keep on eliminating the variables based on p-value. If the p-value is greater than 0.05, eliminate the variable which has largest p-value first.

Building the FULL model:

Full = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_TIME + distance + weather + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + AIR_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Then, we came to know that distance, weather, ELAPSED_TIME, AIR_TIME, DEPARTURE_DELAY, DEPARTURE_TIME, SCHEDULED_TIME are not having a significant effect on ARRIVAL_DELAY because the p value of all these variables or factors are greater than 0.05

so, we will start removing these variables but one by one i.e. we remove the variable which is having the highest p value first. Here we can see largest p-value is 0.9624 which is of variable weather, so we must eliminate weather and build a new model m1.

m1 = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_TIME + distance + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + AIR_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Here we can see largest p-value is 0.5075 which is of variable DEPARTURE_TIME, so we must eliminate DEPARTURE_TIME and build a new model m2.

m2 = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + AIR_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Here we can see largest p-value is 0.3969 which is of variable AIR_TIME, so we must eliminate AIR_TIME and build a new model m3.

m3 = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Here we can see largest p-value is 0.2580 which is of variable ELAPSED_TIME, so we must eliminate ELAPSED_TIME and build a new model m4.

m4 = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance + DEPARTURE_DELAY + SCHEDULED_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Here we can see largest p-value is 0.8937 which is of variable SCHEDULED_TIME, so we must eliminate SCHEDULED_TIME and build a new model m5.

m5 = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance + DEPARTURE_DELAY + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Here we can see largest p-value is 0.1616 which is of variable distance, so we must eliminate distance and build a new model m6.

So, the model m6 will have SCHEDULED_DEPARTURE, DEPARTURE_DELAY, SCHEDULED_ARRIVAL and ARRIVAL_TIME. These are the variables or factors which will have significant effect on ARRIVAL_DELAY because the p value of these variables is less than 0.05. The model has **adj-R2 value of 81.36%** which is higher enough to indicate that the model is good.

Approach II: Backward elimination by AIC:

In this method, full is the full regression model that adopts all the x variables or factors. We set trace = True or False, can help you track the steps in Backward Elimination. However, the step function above, will use AIC as the metric to drop x variables or factors. In this method it tries to minimize the AIC criterion which is like the Cp statistics.

First, we use the step function on full model and set direction as backward and trace as false

Step(full, direction="backward", trace=F)

aicmodel = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

after building the model, we have same model as the model build by Backward elimination by p-value i.e. having the same x variables or factors and adjusted R square

Approach III: Stepwise Regression:

In these method we can also use step() function. First, we must build a base model, Base is the model you start from, simply you can build a model with one x variable. The stepwise regression will try to minimize AIC too.

At first, include SCHEDULED_DEPARTURE as a x variable.

base = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE)

then use step function with direction forward and trace as false

step(base, scope=list(upper = full, lower=~1), direction=" forward", trace=F)

building the model:

stepwiseregressionmodel = lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

By this method, we have got same model as compared to the models build by other two methods.

We can say that on ARRIVAL_DELAY, there are 4 independent significant x variables: **SCHEDULED_DEPARTURE, DEPARTURE_DELAY, SCHEDULED_ARRIVAL, ARRIVAL_TIME**

Interpretation of model,

1-unit change in SCHEDULED_DEPARTURE will decrease the ARRIVAL_DELAY by 0.005882.

1-unit change in DEPARTURE_DELAY will increase the ARRIVAL_DELAY by 0.335626.

1-unit change in SCHEDULED_ARRIVAL will decrease the ARRIVAL_DELAY by 0.405357.

1-unit change in ARRIVAL_TIME will increase the ARRIVAL_DELAY by 0.404357.

ANOVA:

Next step in our analysis we have done **ANOVA**, in ANOVA, since our predicted variable is arrival delay and it is a numerical value, we have split the arrival delay into four different groups namely high, medium, low and on time. Here, for performing anova, we maintained an altogether different dataset which consists of columns like arrival delay and delay_label. The delay labels were high, medium, low and on time.

ANOVA model step by step:

We plotted the box plot on different delay_label. We showed the summary of different delay_label. We then fitted the anova model and we found out that arrival delay averages are not equal. The arrival time of a flight will differ according to various arrival delays. We then performed the residual analysis to support our model. In this, we plotted predicted vs residuals. Also, we plotted QQ plot.

TIME SERIES ANALYSIS:

Next step in our analysis, we did Time series analysis, here we have done time series analysis on arrival time to predict or forecast the future arrival time of a carrier.

In **time series analysis**, we have used different data set comprising of an additional column of date, in the date column we have included dates from 1991 to 2012 to perform a better time series analysis of past data to predict the future.

Steps:

we created a time series object by using zoo package.

We computed the summary statistics by using libraries like tseries and fBasics

We plotted histogram and we found out that the histogram is symmetric.

We plotted QQ plot and found out that the QQ plot is normal distributed.

We created time plot on arrival time from the year 1991 to the year 2012 and we found out that there is no trend and no change in variance. So, no need to apply differencing.

We then computed ACF and PACF values up to a maximum lag of 15 and plotted the correlogram and we found out that p = 2 and q = 1 from PACF and ACF plots respectively.

We then did Ljung Box test (white noise testing) to check for serially correlated data and we found out that since p values in all the different lag values are less than 0.05, we conclude that the data is serially correlated and, so we can predict the future arrival time from the past arrival time data.

Moving towards the AR model that we fitted, we took p = 2 from PACF plot. The aic value of AR(2) model came out to be 29088.66

Then we fitted MA model, we took q = 1 from ACF plot. The aic value of MA(1) model came out to be 29368.1

comparing, AR(2) is better as aic is less than MA(1)'s aic value.

Further, we did residual analysis on the above models, we did Jarque bera test and we found out that p value is less than 0.05. we also checked for white noise i.e. we did Ljung Box test, we found out that for different lag values, p values are less than 0.05. this means it supports our models.

Further, we fitted ARMA model, here ARMA model can be fitted by using three different approaches. We did use all of them to find out the best model out of them.

Approach I (by selecting p and q values manually):

From the earlier plotted PACF and ACF plots, we got p = 2 and q = 1 from PACF and ACF plot respectively.

By using arima function, we plotted ARMA model of order = c(2,0,1) and got the aic value as 28811.84

Further, we did residual analysis on the above models, we did Jarque bera test and we found out that p value is less than 0.05. we also checked for white noise i.e. we did Ljung Box test, we found out that for different lag values, p values are less than 0.05. this means it supports our models.

Approach II (by automatically generating p and q values by loading EACF.R file):

We first loaded the EACF.R file in the R console and calculated on arrival time and found out that p = 2 and q = 2.

By using arima function, we plotted ARMA model of order = c(2,0,2) and got the aic value as 28828.14

Further, we did residual analysis on the above models, we did Jarque bera test and we found out that p value is less than 0.05. we also checked for white noise i.e. we did Ljung Box test, we found out that for different lag values, p values are less than 0.05. this means it supports our models.

Approach III (by minimizing AIC/BIC value):

By using auto.arima(arrivaltime, max.P=8, max.Q=8, ic="aic"), we got aic value as 28794.65

Further, we plotted ARIMA model by using auto.arima(coredata(ratets)) where ratets is the time series object and we got the aic value as 29477.47

best model is from approach III i.e. ARMA(1,1,1) i.e. m9 having aic value = 28794.65 which is the least of all the above aic values.

Then, we use this model having aic value as 28794.65 to predict or forecast the future arrival time. We used predict function from the forecast library.

pr = predict(m9, n.ahead = 10, se.fit=T) and we forecasted arrival time 10 steps ahead in the future. Also, we plotted the time series plot for the forecasted arrival time.

CLASSIFICATION:

The next analysis that we did is by using various **classification techniques** i.e. we made analysis by using **K nearest neighbor (KNN) approach and Naïve Bayes classification** approach. Before we proceed to these two approaches, let me just inform you that we have used a different data set to perform classification. We have built the classification model on 80% training data set and tested the model on 20% testing data set. Amongst all the columns in the data set, we have used only the important ones which will act as features and label. SCHEDULED_DEPARTURE, DEPARTURE_DELAY,

SCHEDULED_ARRIVAL and ARRIVAL_DELAY will act as features and delay will act as label. With the help of features (factors), we will predict the label (flight status) i.e. whether the flight or carrier will be on time or delayed.

KNN approach:

We will read the csv file and store its content in gc.

We use recode function to recode the flight status i.e. delay as on time = 0 and delay = 1

We will normalize the data for accurate results. It is better practice to always normalize your data.

```
normalize <- function(x) {return ((x - min(x)) / (max(x) - min(x)))}
```

we then choose only the above-mentioned features and normalize them by

```
gc_n <- as.data.frame(lapply(gc[15:18], normalize))
```

we then split the dataset into training and testing, where we take 80% as training and 20% as testing. To be more precise we take row 1 to row 1761 as training set and row 1762 to row 2201 as testing set.

```
gc_train <- gc_n[1:1761,]

gc_test <- gc_n[1762:2201,]
```

we also apply the same splitting on label (delay)

```
gc_train_labels <- gc[1:1761, 14]

gc_test_labels <- gc[1762:2201, 14]
```

we import library class and apply knn function to testing data set.

```
gc_test_pred <- knn(train = gc_train, test = gc_test,cl = gc_train_labels, k=20)

CrossTable(x= gc_test_labels, y=gc_test_pred,prop.chisq=FALSE)
```

we tested for various k values as k = 3, 5, 21, 10 but for k = 20 we got the highest accuracy of 62.72%

Naïve Bayes Classification approach:

Here, the splitting of data remains the same as 80% training and 20% testing.

We first install package caret, klaR and mass and naivebayes.

We will read the csv file and store its content in features.

We use recode function to recode the flight status i.e. delay as on time = 0 and delay = 1

The features used in the naïve bayes classification technique remains the same i.e. SCHEDULED_DEPARTURE, DEPARTURE_DELAY, SCHEDULED_ARRIVAL and ARRIVAL_DELAY and label also remains the same i.e. delay.

We load the features in x and label in y

```
x = features[,-5]

y = features$delay
```

we then use train function to build the model and pass x and y variables.

```
model = train(x,y,'nb',trControl=trainControl(method='cv',number=10))
```

we then predict the classified label by using predict function i.e. find the value of label “delay”. In other words, we say that we find the flight status whether it is on time or delayed.

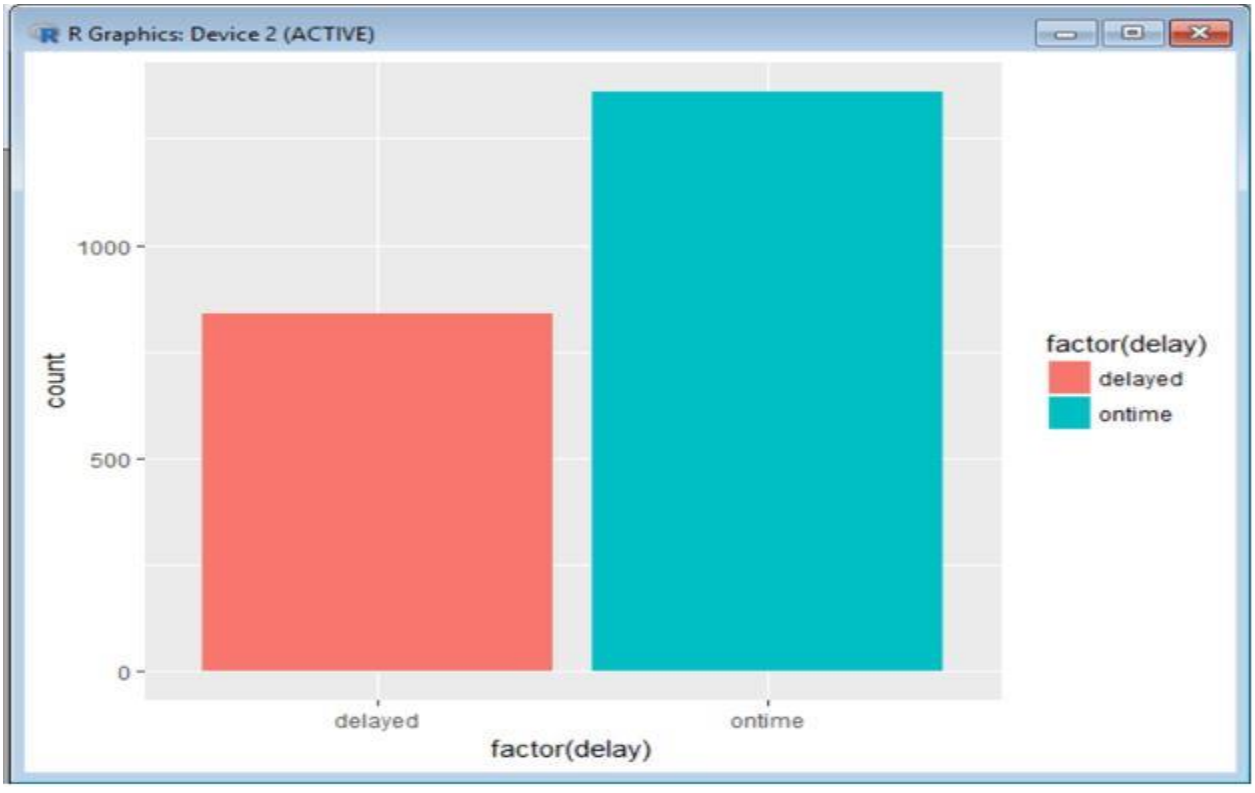
```
predict(model$finalModel,x)
```


```
predict(model$finalModel,x)
```

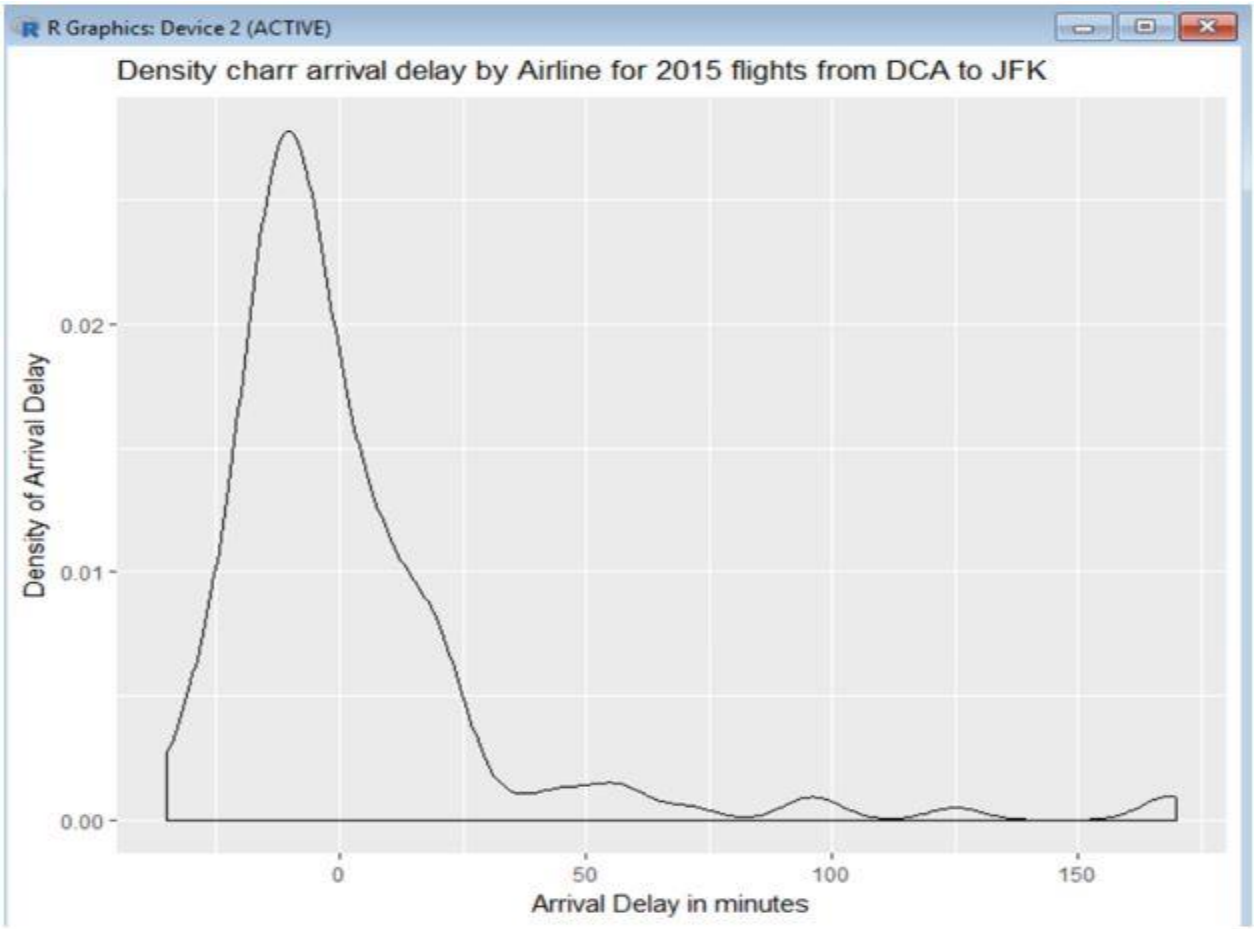

the output will consist of labels and its posterior probability. It means that for example, if you consider data 2197, then due to whatever features listed in that row 2197, the prediction will be on time and the probability of the prediction will be some log value and you must convert it to normal value by taking antilog. The normal value will be 79%. In other words, we say that it is 79% accurate that the flight status will be on time or the probability that the flight status will be on time is 0.79 which is high, and we can trust it.

Snapshot of Descriptive Statistics:

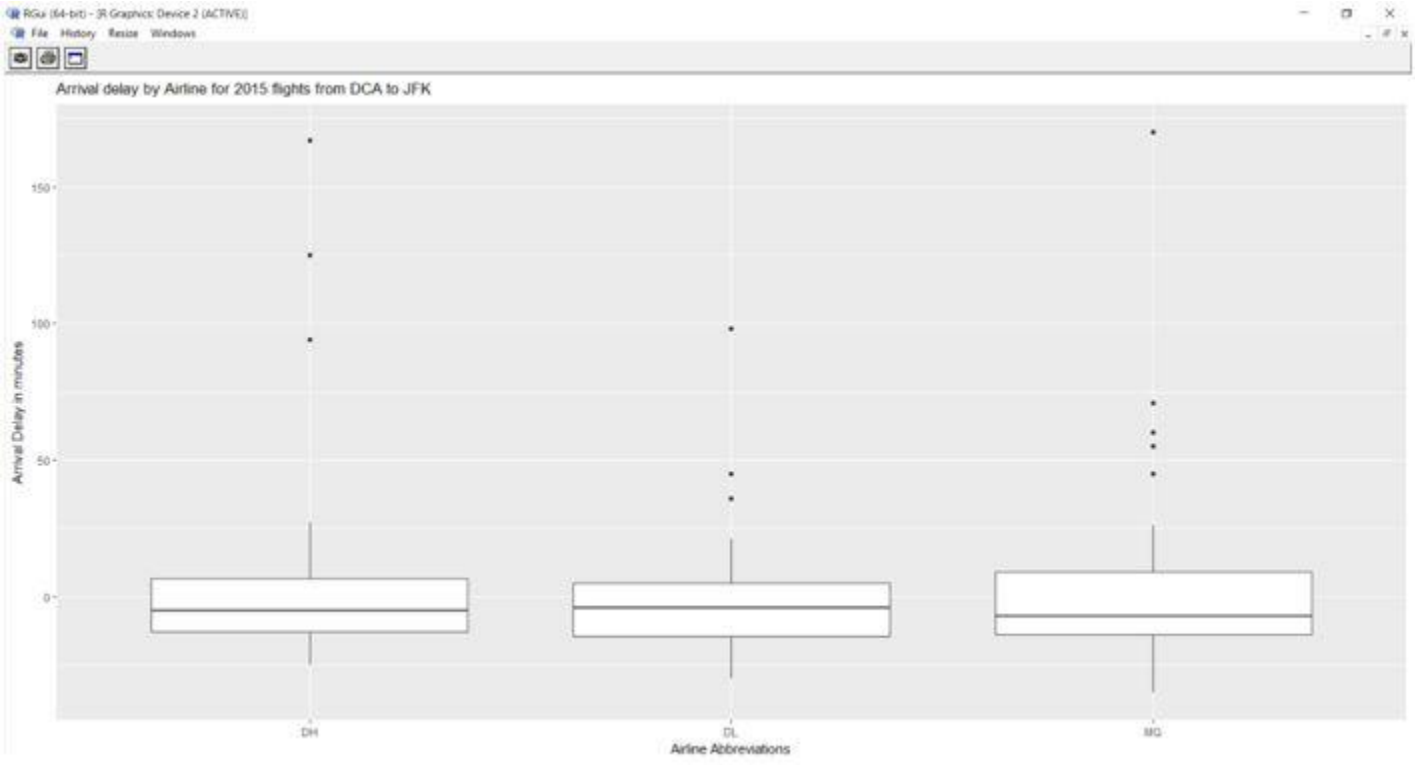
Snapshot of no of flights which were delayed and on time



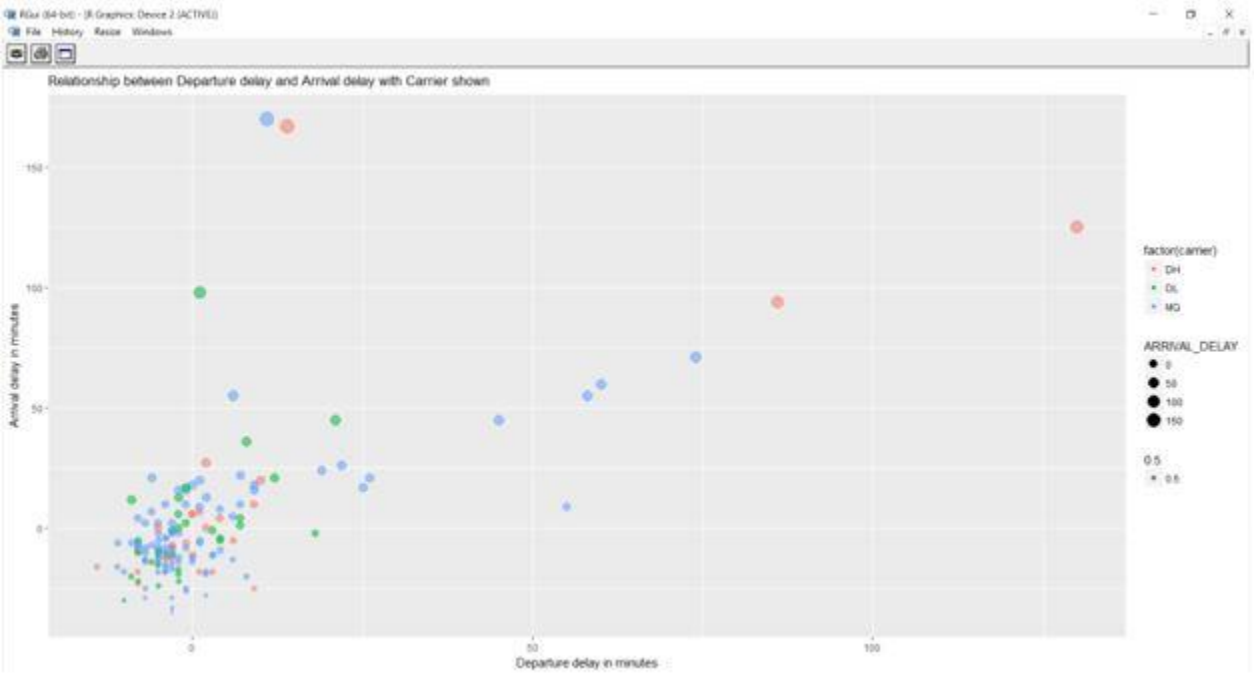
Snapshot of density chart showing no of flights which were on time and delayed for 1 min to beyond 150 mins



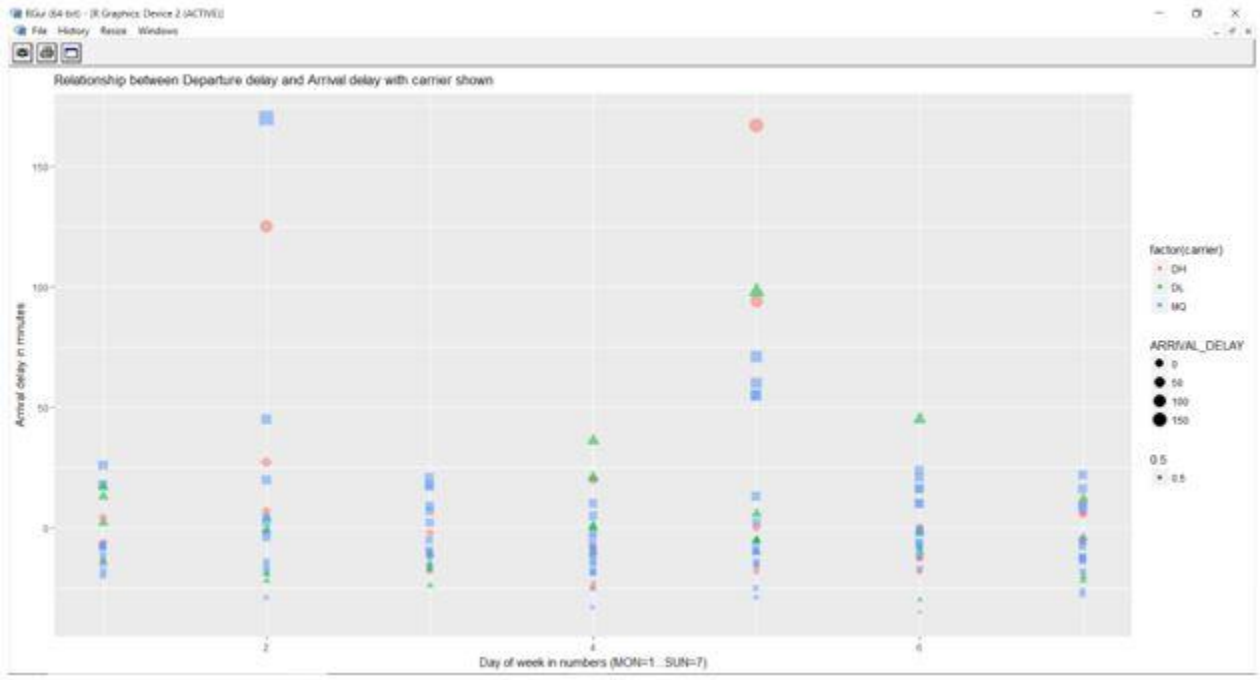
Snapshot of box plot analysis showing various carriers and its arrival delay in minutes



Snapshot of scatter plot or relationship between Arrival delay and departure delay



Snapshot of arrival delay of carriers on days of week



Providing snapshots of Hypothesis testing under inferential statistics.

```
> z.test(delaytimeweathergood,delaytimeweatherbad,alternative="less",mu=0,sigma.x=sd(delaytimeweathergood),sigma.y=sd(delaytimeweatherbad),conf.level=0.95)

        Two-sample z-Test

data:  delaytimeweathergood and delaytimeweatherbad
z = -7918.9, p-value < 2.2e-16
alternative hypothesis: true difference in means is less than 0
95 percent confidence interval:
        NA -149.6684
sample estimates:
mean of x mean of y
 13.88018 163.57968
```

Next, we performed linear regression analysis and for building a best fitted model, we did model selection by using three different approaches.

Approach I: Backward elimination by p value:

Snapshot of full model:

```
> full=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+DEPARTURE_TIME+distance+weather+DEPARTURE_DELAY+SCHEDULED_TIME+ELAPSED_TIME+AIR_TIME+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(full)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_TIME +
    distance + weather + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME +
    AIR_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-188.399   -5.914   -0.789    5.203  125.311

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        -1.6231340  5.4235616  -0.299   0.7648
SCHEDULED_DEPARTURE -0.0067445  0.0034350  -1.963   0.0498 *
DEPARTURE_TIME      -0.0004671  0.0007052  -0.662   0.5078
distance             0.0343960  0.0236860   1.452   0.1466
weather              0.1246815  2.6473183   0.047   0.9624
DEPARTURE_DELAY      0.3390522  0.0189038  17.936   <2e-16 ***
SCHEDULED_TIME      -0.2435289  0.2001267  -1.217   0.2238
ELAPSED_TIME         0.2441753  0.2000513   1.221   0.2224
AIR_TIME            -0.0019703  0.0022826  -0.863   0.3882
SCHEDULED_ARRIVAL   -0.4045169  0.0070195 -57.628   <2e-16 ***
ARRIVAL_TIME         0.4036899  0.0068231  59.165   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.26 on 1749 degrees of freedom
Multiple R-squared:  0.8145,    Adjusted R-squared:  0.8134
F-statistic:   768 on 10 and 1749 DF,  p-value: < 2.2e-16
```

Eliminating weather:

```
> m1=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+DEPARTURE_TIME+distance+DEPARTURE_DELAY+SCHEDULED_TIME+ELAPSED_TIME+AIR_TIME+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m1)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_TIME +
    distance + DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME +
    AIR_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-188.402   -5.911   -0.790    5.198  125.308

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -1.6452113  5.4017254  -0.305   0.7607
SCHEDULED_DEPARTURE -0.0067307  0.0034216  -1.967   0.0493 *
DEPARTURE_TIME      -0.0004673  0.0007049  -0.663   0.5075
distance             0.0344552  0.0236458   1.457   0.1453
DEPARTURE_DELAY      0.3390184  0.0188848  17.952   <2e-16 ***
SCHEDULED_TIME      -0.2432830  0.2000015  -1.216   0.2240
ELAPSED_TIME         0.2439367  0.1999301   1.220   0.2226
AIR_TIME            -0.0019719  0.0022817  -0.864   0.3876
SCHEDULED_ARRIVAL   -0.4045191  0.0070173 -57.646   <2e-16 ***
ARRIVAL_TIME         0.4036934  0.0068207  59.186   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.26 on 1750 degrees of freedom
Multiple R-squared:  0.8145,    Adjusted R-squared:  0.8136
F-statistic: 853.8 on 9 and 1750 DF,  p-value: < 2.2e-16
```

Eliminating DEPARTURE_TIME:

```
> m2=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+distance+DEPARTURE_DELAY+SCHEDULED_TIME+ELAPSED_TIME+AIR_TIME+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m2)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance +
    DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + AIR_TIME +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-188.630   -5.906   -0.837    5.146  125.229

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -2.013716   5.372187  -0.375   0.7078
SCHEDULED_DEPARTURE -0.007069   0.003383  -2.090   0.0368 *
distance             0.033935   0.023629   1.436   0.1511
DEPARTURE_DELAY      0.339029   0.018882  17.955   <2e-16 ***
SCHEDULED_TIME      -0.228585   0.198737  -1.150   0.2502
ELAPSED_TIME         0.229252   0.198667   1.154   0.2487
AIR_TIME            -0.001933   0.002281  -0.847   0.3969
SCHEDULED_ARRIVAL   -0.404318   0.007010 -57.680   <2e-16 ***
ARRIVAL_TIME         0.403564   0.006817  59.201   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1751 degrees of freedom
Multiple R-squared:  0.8145,    Adjusted R-squared:  0.8136
F-statistic: 960.8 on 8 and 1751 DF,  p-value: < 2.2e-16
```

Eliminating AIR_TIME:

```
> m3=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+distance+DEPARTURE_DELAY+SCHEDULED_TIME+ELAPSED_TIME+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m3)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance +
    DEPARTURE_DELAY + SCHEDULED_TIME + ELAPSED_TIME + SCHEDULED_ARRIVAL +
    ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.503   -5.855   -0.769    5.148  125.245

Coefficients:
                     Estimate Std. Error t value Pr(>|t|)
(Intercept)        -2.198923   5.367308  -0.410   0.6821
SCHEDULED_DEPARTURE -0.007073   0.003382  -2.091   0.0367 *
distance             0.033921   0.023627   1.436   0.1513
DEPARTURE_DELAY      0.337346   0.018776  17.967   <2e-16 ***
SCHEDULED_TIME      -0.225478   0.198687  -1.135   0.2566
ELAPSED_TIME         0.224696   0.198579   1.132   0.2580
SCHEDULED_ARRIVAL   -0.404408   0.007008 -57.704   <2e-16 ***
ARRIVAL_TIME         0.403766   0.006812  59.272   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1752 degrees of freedom
Multiple R-squared:  0.8144,    Adjusted R-squared:  0.8136
F-statistic:  1098 on 7 and 1752 DF,  p-value: < 2.2e-16
```

Eliminating ELAPSED_TIME:

```
> m4=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+distance+DEPARTURE_DELAY+SCHEDULED_TIME+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m4)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance +
    DEPARTURE_DELAY + SCHEDULED_TIME + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.682   -5.874   -0.717    5.090  124.994

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        -2.5945775  5.3563339  -0.484   0.6282
SCHEDULED_DEPARTURE -0.0060593  0.0032620  -1.858   0.0634 .
distance            0.0329892  0.0236146   1.397   0.1626
DEPARTURE_DELAY     0.3357512  0.0187241  17.932   <2e-16 ***
SCHEDULED_TIME     -0.0007493  0.0056041  -0.134   0.8937
SCHEDULED_ARRIVAL  -0.4050973  0.0069823 -58.017   <2e-16 ***
ARRIVAL_TIME        0.4043161  0.0067953  59.500   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1753 degrees of freedom
Multiple R-squared:  0.8143,    Adjusted R-squared:  0.8136
F-statistic:  1281 on 6 and 1753 DF,  p-value: < 2.2e-16
```

Eliminating SCHEDULED_TIME:

```
> m5=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+distance+DEPARTURE_DELAY+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m5)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + distance +
    DEPARTURE_DELAY + SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.641   -5.876   -0.719    5.115  125.042

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)        -2.636374   5.345707  -0.493   0.6220
SCHEDULED_DEPARTURE -0.005870   0.002939  -1.998   0.0459 *
distance            0.033051   0.023603   1.400   0.1616
DEPARTURE_DELAY     0.335752   0.018719  17.937   <2e-16 ***
SCHEDULED_ARRIVAL  -0.405305   0.006806 -59.554   <2e-16 ***
ARRIVAL_TIME        0.404293   0.006791  59.532   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1754 degrees of freedom
Multiple R-squared:  0.8142,    Adjusted R-squared:  0.8137
F-statistic:  1538 on 5 and 1754 DF,  p-value: < 2.2e-16
```

Eliminating distance:

```
> m6=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+DEPARTURE_DELAY+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(m6)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.600   -5.798   -0.744    5.123  125.538

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         4.362504   1.896480   2.300   0.0215 *
SCHEDULED_DEPARTURE -0.005882   0.002940  -2.001   0.0456 *
DEPARTURE_DELAY     0.335626   0.018724  17.925   <2e-16 ***
SCHEDULED_ARRIVAL  -0.405357   0.006807 -59.546   <2e-16 ***
ARRIVAL_TIME        0.404357   0.006793  59.526   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1755 degrees of freedom
Multiple R-squared:  0.814,     Adjusted R-squared:  0.8136
F-statistic:  1921 on 4 and 1755 DF,  p-value: < 2.2e-16
```

Here, we got the model having **adj-R2 value as 81.36%**

Approach II: Backward elimination by AIC:

```
> step(full, direction="backward", trace=F)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Coefficients:
        (Intercept)  SCHEDULED_DEPARTURE      DEPARTURE_DELAY    SCHEDULED_ARRIVAL         ARRIVAL_TIME
           4.362504            -0.005882             0.335626            -0.405357             0.404357

> aicmodel=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE+DEPARTURE_DELAY+SCHEDULED_ARRIVAL+ARRIVAL_TIME)
> summary(aicmodel)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.600   -5.798   -0.744    5.123  125.538

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)          4.362504   1.896480   2.300   0.0215 *
SCHEDULED_DEPARTURE -0.005882   0.002940  -2.001   0.0456 *
DEPARTURE_DELAY      0.335626   0.018724  17.925   <2e-16 ***
SCHEDULED_ARRIVAL   -0.405357   0.006807 -59.546   <2e-16 ***
ARRIVAL_TIME         0.404357   0.006793  59.526   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1755 degrees of freedom
Multiple R-squared:  0.814,     Adjusted R-squared:  0.8136
F-statistic:  1921 on 4 and 1755 DF,  p-value: < 2.2e-16
```

Here, we have got same model as that of the approach I.

Approach III: Step Wise Regression model:

```
> base=lm(ARRIVAL_DELAY~SCHEDULED_DEPARTURE)
> step(base, scope=list(upper=full, lower=~1), direction="forward", trace=F)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Coefficients:
        (Intercept)  SCHEDULED_DEPARTURE      DEPARTURE_DELAY    SCHEDULED_ARRIVAL         ARRIVAL_TIME
           4.362504            -0.005882             0.335626            -0.405357             0.404357

> stepwiseRegressionModel=lm(ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +SCHEDULED_ARRIVAL + ARRIVAL_TIME)
> summary(stepwiseRegressionModel)

Call:
lm(formula = ARRIVAL_DELAY ~ SCHEDULED_DEPARTURE + DEPARTURE_DELAY +
    SCHEDULED_ARRIVAL + ARRIVAL_TIME)

Residuals:
     Min       1Q   Median       3Q      Max
-189.600   -5.798   -0.744    5.123  125.538

Coefficients:
                    Estimate Std. Error t value Pr(>|t|)
(Intercept)         4.362504   1.896480   2.300   0.0215 *
SCHEDULED_DEPARTURE -0.005882   0.002940  -2.001   0.0456 *
DEPARTURE_DELAY      0.335626   0.018724  17.925   <2e-16 ***
SCHEDULED_ARRIVAL   -0.405357   0.006807 -59.546   <2e-16 ***
ARRIVAL_TIME         0.404357   0.006793  59.526   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 13.25 on 1755 degrees of freedom
Multiple R-squared:  0.814,     Adjusted R-squared:  0.8136
F-statistic:  1921 on 4 and 1755 DF,  p-value: < 2.2e-16
```

Again, we have got the same model as that of approach I and II.

Now, to determine the accuracy of the model, we calculate the **RMSE values** of every above model that we got.

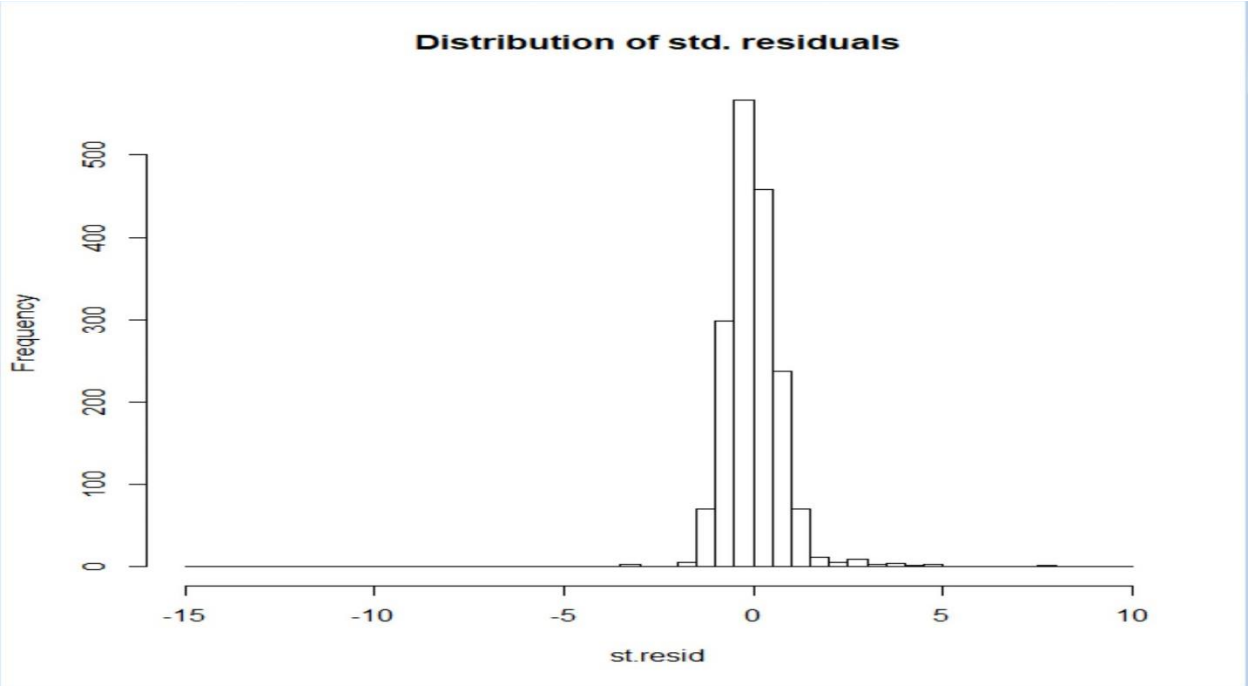
Snapshot of RMSE values of every model:

```
> y1=predict.glm(m6,test.data)
> y=test.data[,22]
> rmse_1=sqrt((y-y1)%*%(y-y1))/nrow(test.data)
> rmse_1
          [,1]
[1,] 0.6621076
> y2=predict.glm(aicmodel,test.data)
> y=test.data[,22]
> rmse_2=sqrt((y-y2)%*%(y-y2))/nrow(test.data)
> rmse_2
          [,1]
[1,] 0.6621076
> y3=predict.glm(stepwiseRegressionModel,test.data)
> y=test.data[,22]
> rmse_3=sqrt((y-y3)%*%(y-y3))/nrow(test.data)
> rmse_3
          [,1]
[1,] 0.6621076
```

Here, we found out that every model has same RMSE value, so we have the liberty to choose anyone one of above model. We decide to go with Step Wise Regression.

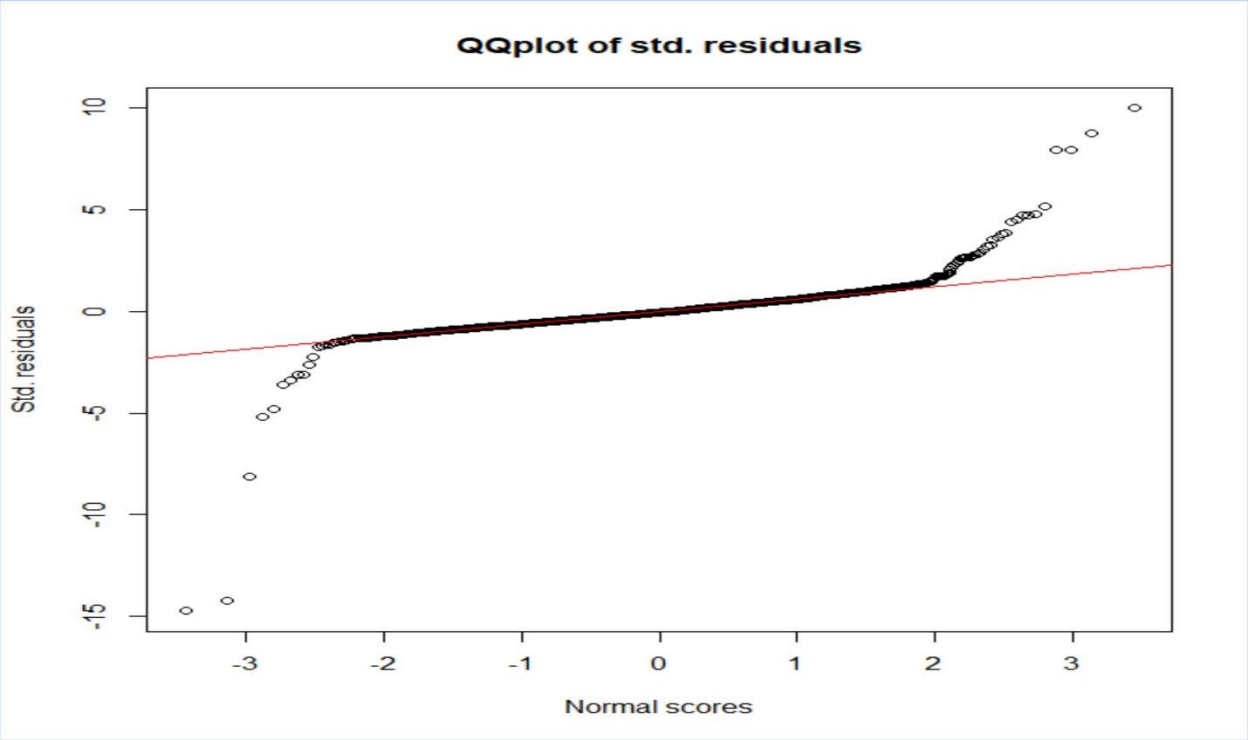
Snapshot of Residual analysis performed to support out model:

Snapshot of histogram:



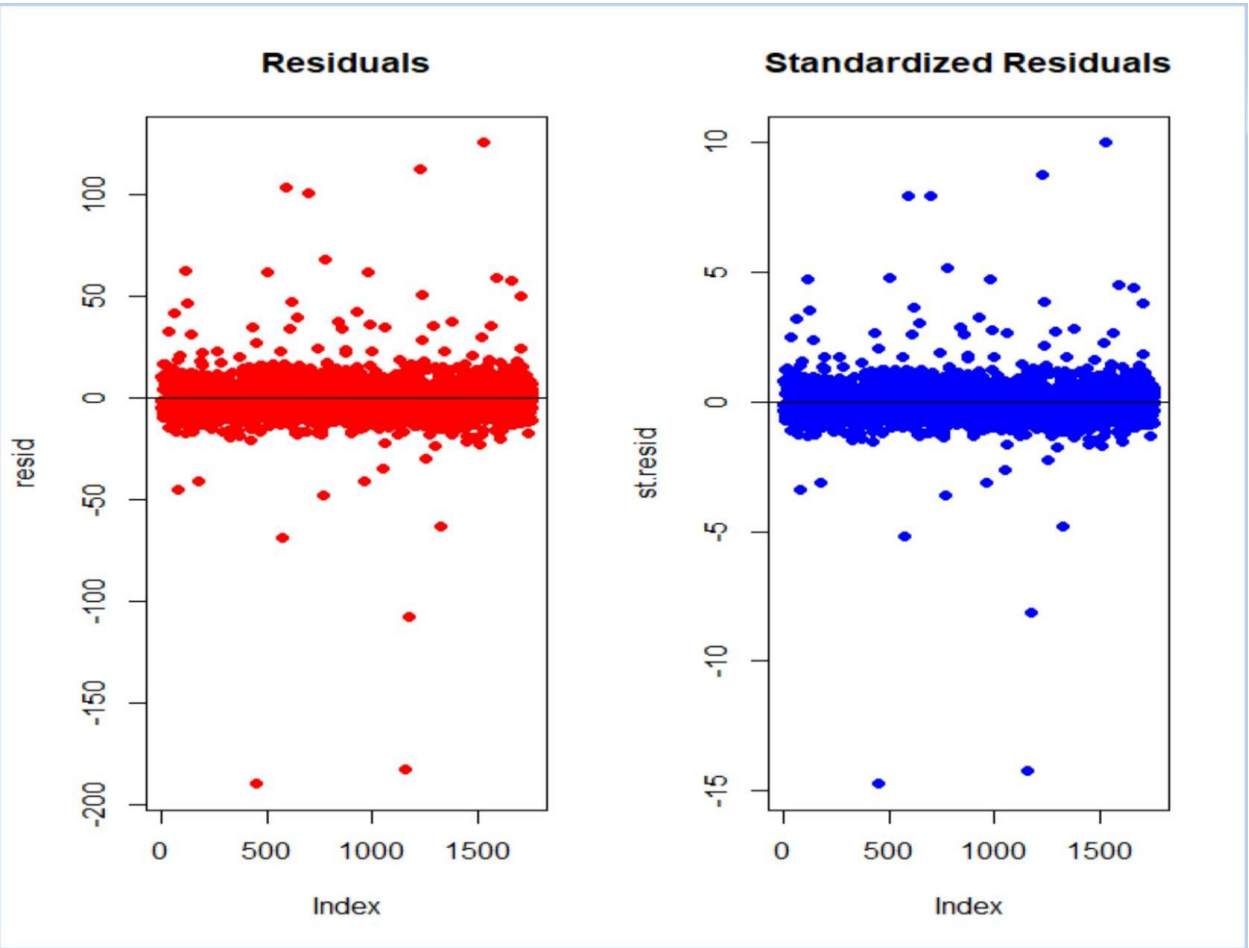
We found that histogram is s**ymmetric.**

Snapshot of QQ plot:



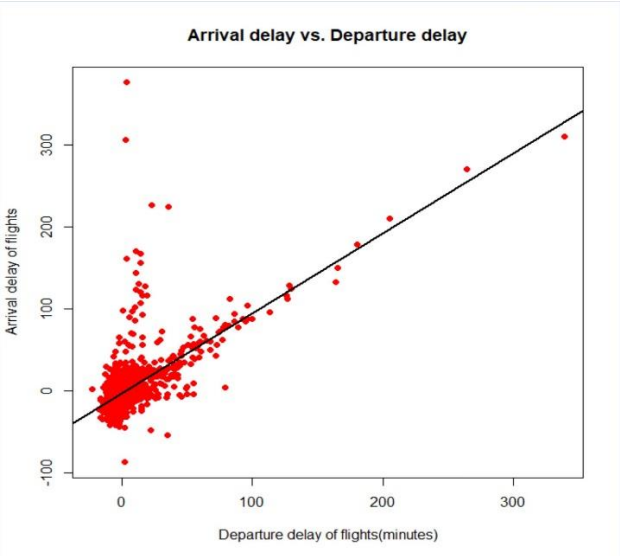
We found out that **QQ plot is normally distributed.**

Snapshot of checking constant variance:



We found out that we have **constant variance.**

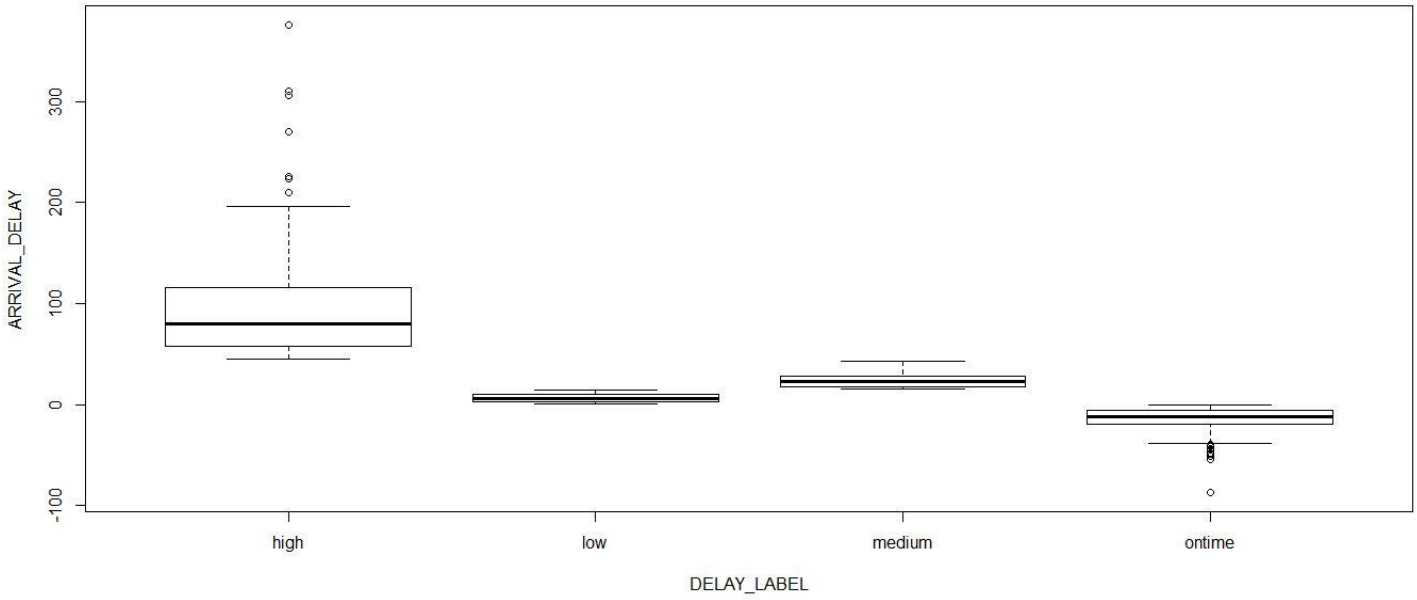
Snapshot of scatterplot:



Interpretation of Plot: For arrival delays vs departure delay, we can see that an increase in departure delay will also increase the arrival delay.

Providing snapshots of ANOVA:

Snapshot of box plot analysis on various delay labels:



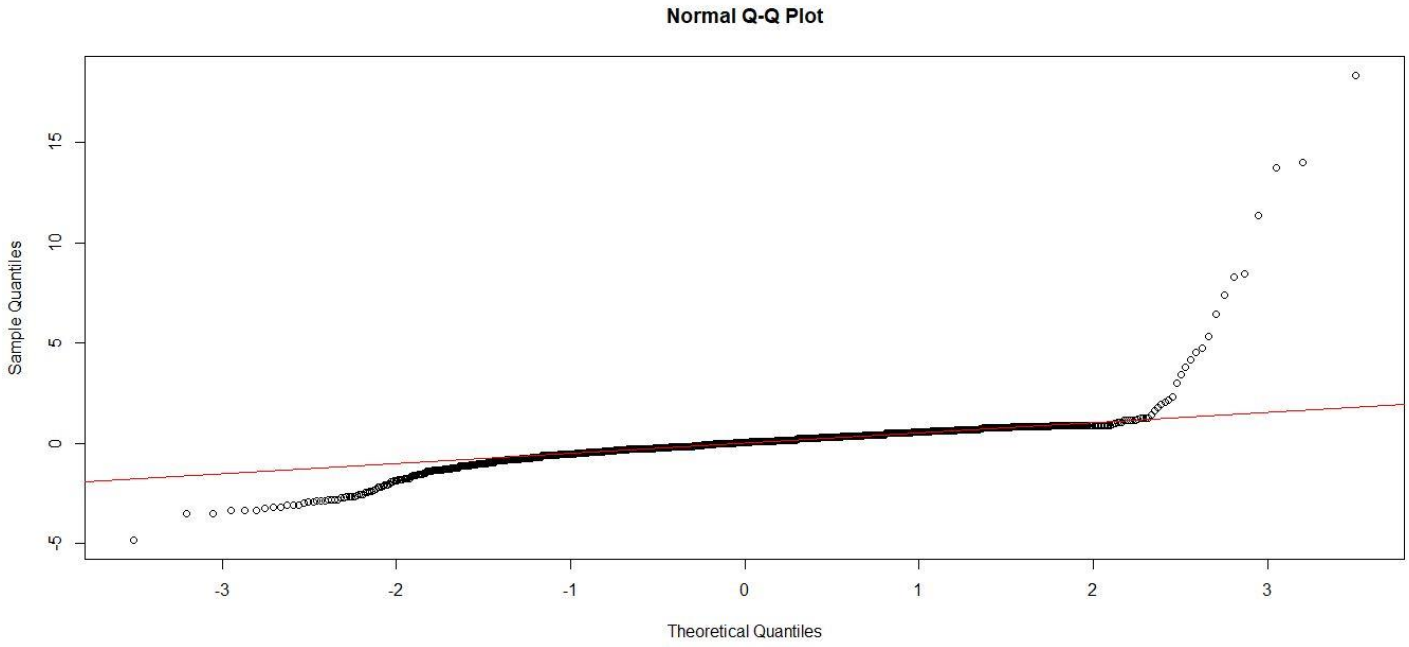
Snapshot of fitted model of ANOVA:

```
> anov = lm(ARRIVAL_DELAY~DELAY_LABEL)
> summary(anov)

Call:
lm(formula = ARRIVAL_DELAY ~ DELAY_LABEL)

Residuals:
    Min      1Q  Median      3Q     Max
-73.571  -5.033   0.429   5.429 277.505

Coefficients:
                 Estimate Std. Error t value Pr(>|t|)
(Intercept)        98.495      1.513   65.12   <2e-16 ***
DELAY_LABELlow    -91.896      1.674  -54.90   <2e-16 ***
DELAY_LABELmedium -74.462      1.838  -40.52   <2e-16 ***
DELAY_LABELontime -111.925     1.565  -71.53   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 15.2 on 2197 degrees of freedom
Multiple R-squared:  0.7311,    Adjusted R-squared:  0.7308
F-statistic:  1992 on 3 and 2197 DF,  p-value: < 2.2e-16
```

from the F test, we get mu1 = 98.495
mu2 = 6.599
mu3 = 24.033
mu4 = -13.43
therefore, we reject null hypothesis and accept alternate hypothesis
null hypothesis is the averages are equal
Alternate Hypothesis is not all averages are equal or at least one average is different
From the above R output, we can infer that the F statistics is high and, we have got **adj-r2 value of 73.08%** which tells us that the model is better. All the variables or labels are having a significant effect on predicted variable (arrival delay).

Performing residual analysis on fitted ANOVA model:

Snapshot of QQ plot:



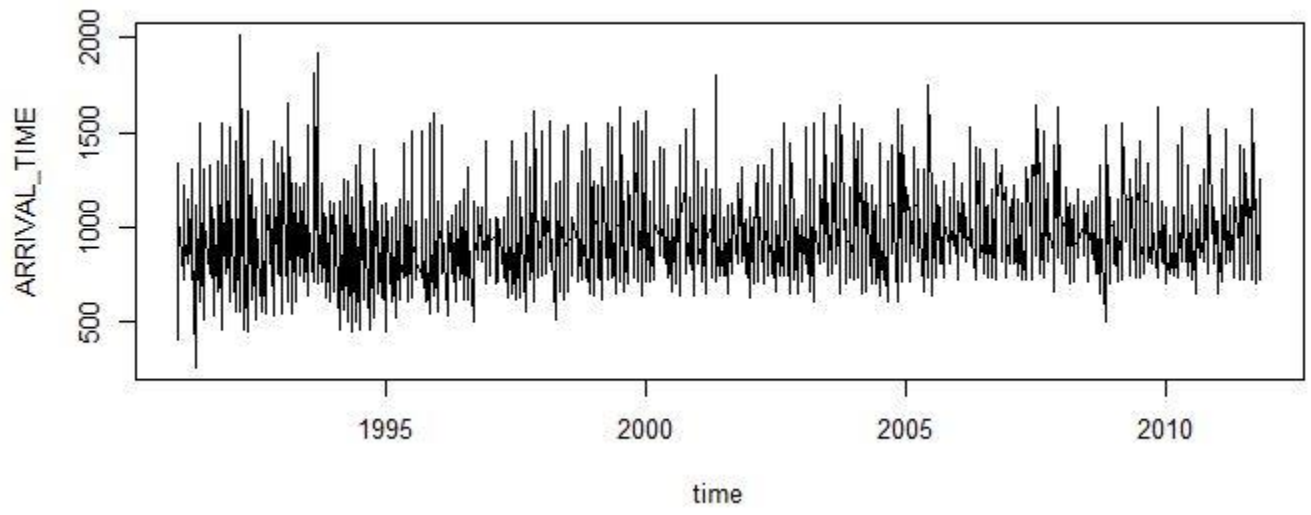
We found out that **QQ plot is normally distributed**.

Snapshot of statistics of arrival delay labels:

```
> summary(high)
 ARRIVAL_DELAY    DELAY_LABEL
 Min.   : 45.0   high  :101
 1st Qu.: 58.0   low   :  0
 Median : 80.0   medium:  0
 Mean   : 98.5   ontime:  0
 3rd Qu.:116.0
 Max.   :376.0
> summary(medium)
 ARRIVAL_DELAY    DELAY_LABEL
 Min.   :15.00   high  :  0
 1st Qu.:18.00   low   :  0
 Median :23.00   medium:212
 Mean   :24.03   ontime:  0
 3rd Qu.:28.00
 Max.   :43.00
> summary(low)
 ARRIVAL_DELAY     DELAY_LABEL
 Min.   : 1.000   high  :  0
 1st Qu.: 3.000   low   :449
 Median : 6.000   medium:  0
 Mean   : 6.599   ontime:  0
 3rd Qu.:10.000
 Max.   :14.000
> summary(ontime)
 ARRIVAL_DELAY     DELAY_LABEL
 Min.   :-87.00   high  :   0
 1st Qu.:-19.00   low   :   0
 Median :-12.00   medium:   0
 Mean   :-13.43   ontime:1439
 3rd Qu.: -6.00
 Max.   :  0.00
>
```

Providing snapshot of Time series analysis:

Snapshot of Time series Plot:

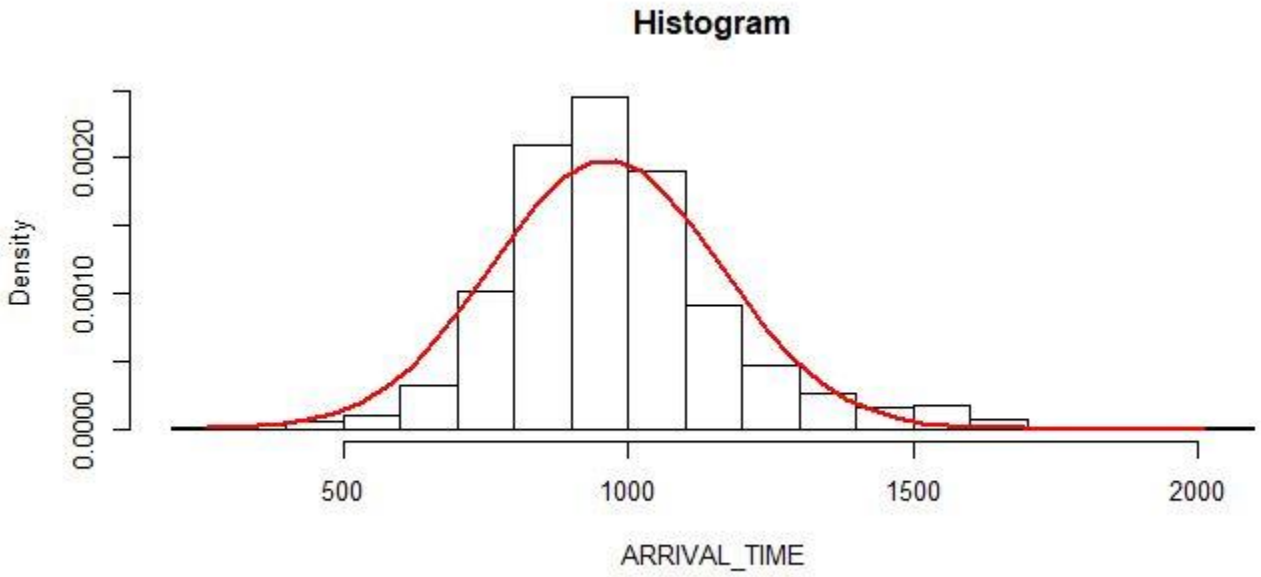


We found out that there is **no trend and has a constant variance.**

Snapshot of statistics of arrival time:

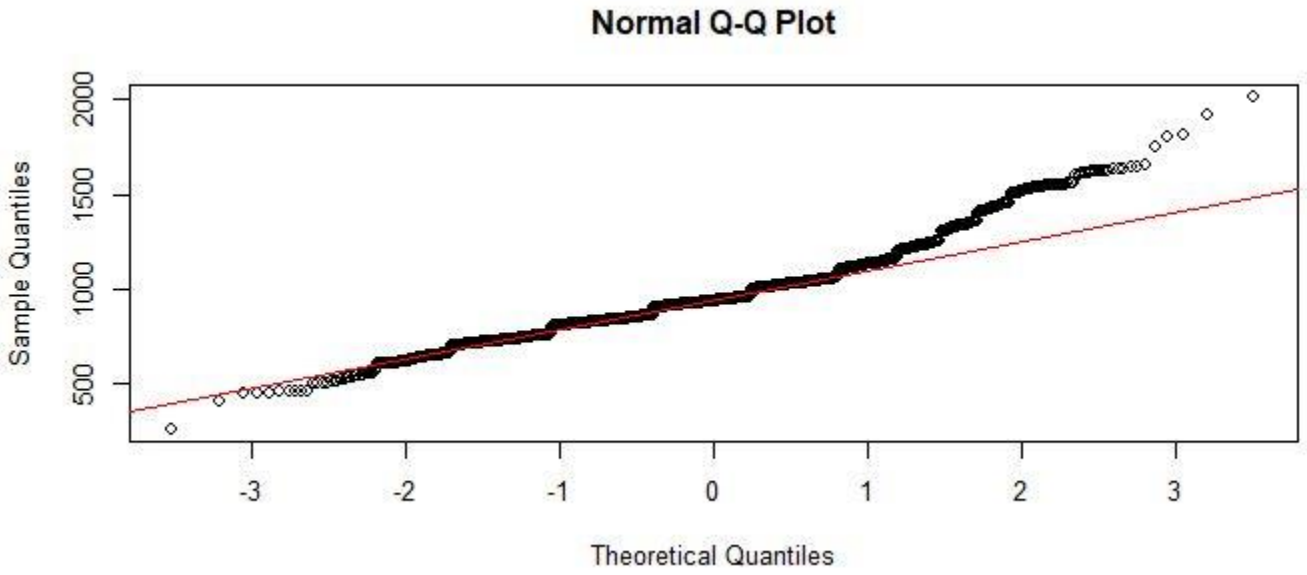
```
> basicStats(arrivaltime)
             arrivaltime
nobs        2.201000e+03
NAs         0.000000e+00
Minimum     2.590000e+02
Maximum     2.010000e+03
1. Quartile 8.330000e+02
3. Quartile 1.043000e+03
Mean        9.617874e+02
Median      9.350000e+02
Sum         2.116894e+06
SE Mean     4.276620e+00
LCL Mean    9.534007e+02
UCL Mean    9.701740e+02
Variance    4.025515e+04
Stdev       2.006369e+02
Skewness    8.786970e-01
Kurtosis    1.868710e+00
>
```

Snapshot of Histogram:



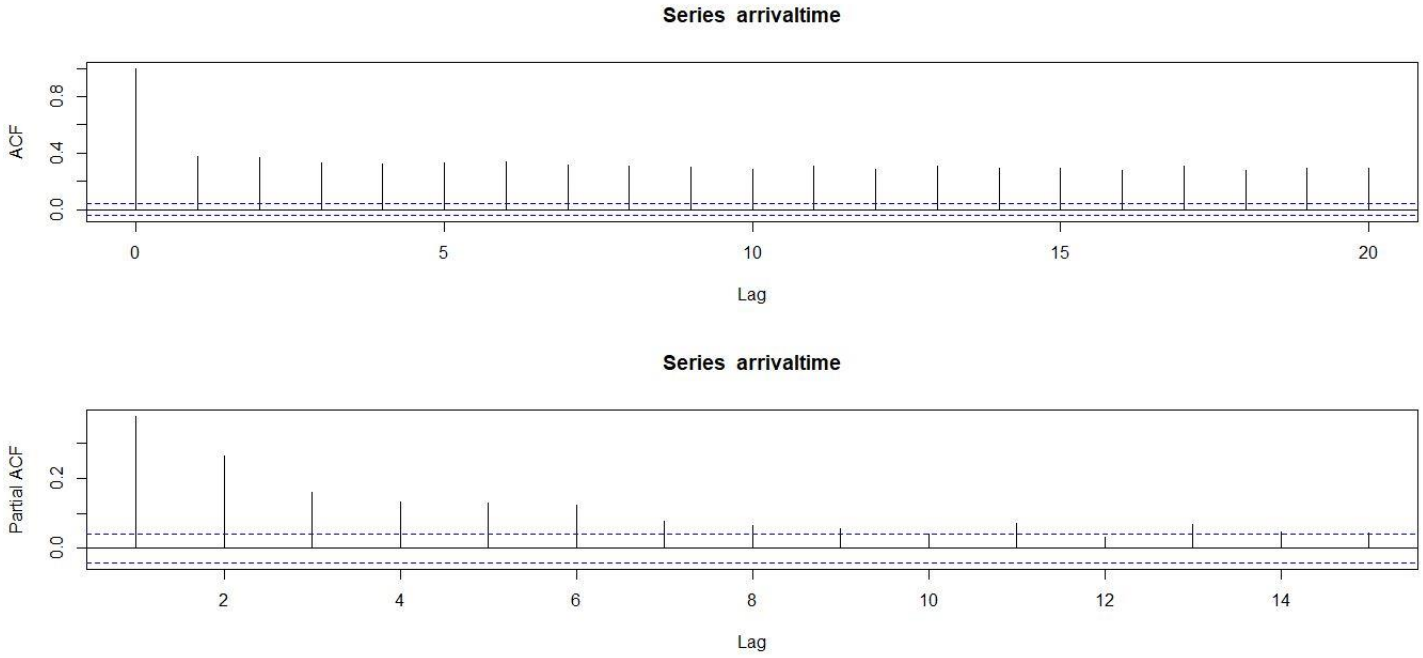
We found out that **histogram is symmetric.**

Snapshot of QQ plot:



We found out that **QQ plot is normally distributed.**

Snapshot of computing ACF and PACF plots:



We found out that **p value is 2 and q value is 1** from PACF and ACF plot respectively.

Snapshot of Ljung Box test:

```
> Box.test(arrivaltime,lag=6,type='Ljung')

        Box-Ljung test

data:  arrivaltime
X-squared = 1582.5, df = 6, p-value < 2.2e-16

> Box.test(arrivaltime,lag=12,type='Ljung')

        Box-Ljung test

data:  arrivaltime
X-squared = 2778.2, df = 12, p-value < 2.2e-16

> Box.test(arrivaltime,lag=15,type='Ljung')

        Box-Ljung test

data:  arrivaltime
X-squared = 3376.4, df = 15, p-value < 2.2e-16

>
```

We found out that all the p values are less than 0.05, so **the data is serially correlated**, and we can predict the future by the past.

Snapshot of AR(2) model:

```
> m1= arima(arrivaltime, order=c(2,0,0), method='ML', include.mean=T)
> m1

Call:
arima(x = arrivaltime, order = c(2, 0, 0), include.mean = T, method = "ML")

Coefficients:
         ar1     ar2  intercept
      0.2795  0.2650   961.7605
s.e.  0.0206  0.0206     8.3685

sigma^2 estimated as 32031:  log likelihood = -14540.33,  aic = 29088.66
>
```

We found out that aic value of AR(2) model is 29088.66

Snapshot of MA(1) model:

```
> m5= arima(arrivaltime, order=c(0,0,1), method='ML', include.mean=T)
> m5

Call:
arima(x = arrivaltime, order = c(0, 0, 1), include.mean = T, method = "ML")

Coefficients:
         ma1  intercept
      0.2613   961.7789
s.e.  0.0171     5.1291

sigma^2 estimated as 36404:  log likelihood = -14681.05,  aic = 29368.1
```

We found out that AIC value of MA(1) model is 29368.1

So, the better model is AR(2) having less AIC value.

Providing residual analysis:

Snapshot of Ljung box test:

```
> Box.test(m1$residuals,lag=6,type='Ljung')

        Box-Ljung test

data:  m1$residuals
X-squared = 83.599, df = 6, p-value = 6.661e-16

> Box.test(m1$residuals,lag=12,type='Ljung')

        Box-Ljung test

data:  m1$residuals
X-squared = 155.83, df = 12, p-value < 2.2e-16

>
```

We found out all the p values are less than 0.05 and hence the data is serially corelated and we can predict the future by the past.

Snapshot of Jarque bera test:

```
> jarque.bera.test(m1$residual)

        Jarque Bera Test

data:  m1$residual
X-squared = 1065.5, df = 2, p-value < 2.2e-16

>
```

Providing ARMA model snapshots:

Snapshots of building ARMA model by manually selecting p and q values:

```
> m7= arima(arrivaltime, order=c(2,0,1), method='ML', include.mean=T)
> m7

Call:
arima(x = arrivaltime, order = c(2, 0, 1), include.mean = T, method = "ML")

Coefficients:
         ar1      ar2      ma1  intercept
      1.0878  -0.0886  -0.9617   958.8134
s.e.  0.0232   0.0230   0.0089   122.7630

sigma^2 estimated as 28193:  log likelihood = -14400.92,  aic = 28811.84
>
```

We found out that AIC value is 28811.24

Snapshot of residual analysis on above model:

```
data:  m7$residual
X-squared = 1560.2, df = 2, p-value < 2.2e-16

> Box.test(m7$residuals,lag=6,type='Ljung')

        Box-Ljung test

data:  m7$residuals
X-squared = 13.329, df = 6, p-value = 0.03809
```

Data is serially correlated, and we can predict the future from the past.

Snapshots of building ARMA model by automatically selecting p and q values by EACF approach:

```
> source("E:/Data Analytics/EACF.R")
> EACF(arrivaltime)
[1] "EACF table"
      [,1]   [,2]   [,3]   [,4]     [,5]   [,6]
[1,]  0.38  0.370  0.330  0.324  0.33080 0.3371
[2,] -0.49  0.028 -0.027 -0.013  0.00218 0.0238
[3,] -0.45 -0.454 -0.022 -0.022  0.00125 0.0311
[4,] -0.50  0.115 -0.095  0.006 -0.01427 0.0283
[5,] -0.50 -0.341 -0.057 -0.390  0.00031 0.0293
[6,] -0.50  0.111 -0.245 -0.383  0.00093 0.0087
[1] " "
[1] "Simplified EACF: 2 denotes significance"
     [,1] [,2] [,3] [,4] [,5] [,6]
[1,]    2    2    2    2    2    2
[2,]    2    0    0    0    0    0
[3,]    2    2    0    0    0    0
[4,]    2    2    2    0    0    0
[5,]    2    2    2    2    0    0
[6,]    2    2    2    2    0    0
>
```

Building model:

```
> m8= arima(arrivaltime, order=c(2,0,2), method='ML', include.mean=T)
> m8

Call:
arima(x = arrivaltime, order = c(2, 0, 2), include.mean = T, method = "ML")

Coefficients:
         ar1     ar2     ma1      ma2  intercept
      0.0528  0.9425  0.0077  -0.8933   960.7850
s.e.     NaN     NaN     NaN      NaN    77.5529

sigma^2 estimated as 28383:  log likelihood = -14408.07,  aic = 28828.14
```

We found out that **AIC value of above model is 28828.14**

Residual analysis of above model:

```
> jarque.bera.test(m8$residual)

        Jarque Bera Test

data:  m8$residual
X-squared = 1588.4, df = 2, p-value < 2.2e-16

> Box.test(m8$residuals,lag=6,type='Ljung')

        Box-Ljung test

data:  m8$residuals
X-squared = 18.771, df = 6, p-value = 0.004569

> Box.test(m8$residuals,lag=12,type='Ljung')

        Box-Ljung test

data:  m8$residuals
X-squared = 27.202, df = 12, p-value = 0.007227

> Box.test(m8$residuals,lag=15,type='Ljung')

        Box-Ljung test

data:  m8$residuals
X-squared = 27.887, df = 15, p-value = 0.02229

>
```

Data is serially correlated, and we can predict the future from the past.

Snapshots of building ARMA model by minimizing AIC/BIC approach:

```
> auto.arima(arrivaltime, max.P=8, max.Q=8, ic="aic")
Series: arrivaltime
ARIMA(1,1,1)

Coefficients:
         ar1      ma1
      0.0909  -0.9645
s.e.  0.0227   0.0076

sigma^2 estimated as 28216:  log likelihood=-14394.33
AIC=28794.65   AICc=28794.66   BIC=28811.74
```

We found out that **AIC value of above model is 28794.66**

Snapshot of ARIMA model:

```
> auto.arima(coredata(ratets))
Series: coredata(ratets)
ARIMA(4,1,3)

Coefficients:
          ar1     ar2      ar3      ar4      ma1      ma2     ma3
      -0.1839  0.4308  -0.1688  -0.1225  -0.7012  -0.7596  0.4802
s.e.   0.0689  0.0562   0.0239   0.0234   0.0670   0.0677  0.0558

sigma^2 estimated as 38377:  log likelihood=-14730.74
AIC=29477.47   AICc=29477.54   BIC=29523.04
>
```

We found out that **aic value of above model is 29477.54**

Comparing above models, **best model is from approach III i.e. ARMA(1,1,1) i.e. m9 having aic = 28794.65**

Snapshot of predicting/forecasting the future values of arrival time of universal airline:

```
> pr = predict(m9, n.ahead = 10, se.fit=T)
> pr
$pred
Time Series:
Start = 2202
End = 2211
Frequency = 1
 [1] 1093.980 1106.607 1107.754 1107.858 1107.868 1107.868 1107.869 1107.869
 [9] 1107.869 1107.869

$se
Time Series:
Start = 2202
End = 2211
Frequency = 1
 [1] 167.9768 169.3120 169.4957 169.6272 169.7543 169.8809 170.0074 170.1337
 [9] 170.2600 170.3862

>
```

Snapshot of K-nearest neighbor:

```
> gc_test_pred <- knn(train = gc_train, test = gc_test,cl = gc_train_labels, k=20)
> CrossTable(x= gc_test_labels, y=gc_test_pred,prop.chisq=FALSE)


   Cell Contents
|-------------------------|
|                       N |
|           N / Row Total |
|           N / Col Total |
|         N / Table Total |
|-------------------------|


Total Observations in Table:  440


                | gc_test_pred
gc_test_labels |         0 |         1 | Row Total |
---------------|-----------|-----------|-----------|
             0 |       242 |        42 |       284 |
               |     0.852 |     0.148 |     0.645 |
               |     0.665 |     0.553 |           |
               |     0.550 |     0.095 |           |
---------------|-----------|-----------|-----------|
             1 |       122 |        34 |       156 |
               |     0.782 |     0.218 |     0.355 |
               |     0.335 |     0.447 |           |
               |     0.277 |     0.077 |           |
---------------|-----------|-----------|-----------|
  Column Total |       364 |        76 |       440 |
               |     0.827 |     0.173 |           |
---------------|-----------|-----------|-----------|
```

We found out that for K = 20, we got the highest accuracy of 62.72% considering the true positives and true negatives.

Here, true positive is 242 that means 242 out of 440 (test data set) are predicted accurately as on time.

Similarly, true negative is 34 that means 34 out of 440 are predicted accurately as delayed.

Here, false positive means 42 out of 440 are predicted to be on time but they are delayed.

Similarly, false negative means 122 out of 440 are predicted to be delayed but they are on time.

Snapshot of Naïve Bayes Classification:

The below snapshot is of predicted labels (on time or delayed) for a given features

```
[1873] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1882] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1891] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1900] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1909] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1918] ontime  ontime  ontime  ontime  ontime  delayed ontime  ontime  ontime
[1927] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1936] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1945] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1954] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1963] delayed ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1972] delayed ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1981] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1990] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[1999] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2008] ontime  ontime  ontime  ontime  ontime  ontime  delayed ontime  ontime
[2017] ontime  ontime  ontime  ontime  ontime  ontime  delayed ontime  ontime
[2026] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2035] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2044] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2053] ontime  delayed ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2062] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2071] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2080] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2089] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2098] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2107] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2116] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2125] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2134] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2143] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2152] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2161] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2170] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2179] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2188] ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime  ontime
[2197] ontime  ontime  ontime  ontime  ontime
Levels: delayed ontime
```

The below is snap shot of probabilities of those labels:

```
[2164,] 1.635688e-01 8.364312e-01
[2165,] 3.026967e-01 6.973033e-01
[2166,] 1.909538e-01 8.090462e-01
[2167,] 1.905018e-01 8.094982e-01
[2168,] 3.134776e-01 6.865224e-01
[2169,] 3.476044e-01 6.523956e-01
[2170,] 3.365041e-01 6.634959e-01
[2171,] 3.479320e-01 6.520680e-01
[2172,] 2.763502e-01 7.236498e-01
[2173,] 1.817551e-01 8.182449e-01
[2174,] 3.573267e-01 6.426733e-01
[2175,] 7.890358e-02 9.210964e-01
[2176,] 2.324980e-01 7.675020e-01
[2177,] 2.367422e-01 7.632578e-01
[2178,] 1.996071e-01 8.003929e-01
[2179,] 1.908380e-01 8.091620e-01
[2180,] 1.881974e-01 8.118026e-01
[2181,] 1.455941e-01 8.544059e-01
[2182,] 1.508292e-02 9.849171e-01
[2183,] 3.128145e-01 6.871855e-01
[2184,] 4.411876e-01 5.588124e-01
[2185,] 1.405001e-01 8.594999e-01
[2186,] 2.393942e-01 7.606058e-01
[2187,] 2.489948e-01 7.510052e-01
[2188,] 1.422598e-01 8.577402e-01
[2189,] 2.130427e-01 7.869573e-01
[2190,] 2.047660e-01 7.952340e-01
[2191,] 2.658526e-01 7.341474e-01
[2192,] 1.314183e-01 8.685817e-01
[2193,] 3.467169e-01 6.532831e-01
[2194,] 2.634594e-01 7.365406e-01
[2195,] 1.874276e-01 8.125724e-01
[2196,] 1.403947e-01 8.596053e-01
[2197,] 2.174206e-01 7.825794e-01
[2198,] 2.832476e-01 7.167524e-01
[2199,] 3.335937e-01 6.664063e-01
[2200,] 2.115691e-01 7.884309e-01
[2201,] 2.894409e-01 7.105591e-01
```

It means that for example, if you consider data 2197, then due to whatever features listed in that row 2197, the prediction will be on time and the probability of the prediction will be log value 2.174206e-01 and you must convert it to normal value by taking antilog. The normal value will be 79%. In other words, we say that it is 79% accurate that the flight status will be on time or the probability that the flight status will be on time is 0.79 which is high, and we can trust it.

6. Evaluations and Results

6.1. Evaluation Methods

We are using **hold out evaluation** method since our data is very large.

Since, we have built up the Linear regression model (see section 5) and we further want to evaluate the models which we have built. We can calculate the accuracy of model based of **RMSE (root mean square error)**. We have used RMSE as a metric to evaluate our model. Lower the RMSE value better is the model. We must evaluate these models on testing data. We got RMSE value of 0.6621076, which is same for all three models. Therefore, we can choose any models as the accuracy is same for all the above models.

We have performed analysis by performing classification techniques like Naïve Bayes and K nearest neighbor. Since, we have built up the model (see section 5) and we further want to evaluate the models which we have built, we have tested our two models on Testing data set.

KNN classification technique brought us to conclusion that it produces an overall accuracy of 62.72% on the testing data set and Naïve Bayes Classification technique brought us to conclusion that it produces an accuracy 79% for a feature i.e. on the selected row in testing data set.

Further, we also evaluated these two classification techniques to find out which one is more better. Since KNN approach is used when the features are numeric and Naïve Bayes is used when the features are nominal and in our data set, the features are numeric, so we would trust KNN or we can say that KNN is better model/technique as compared to Naive Bayes model/technique at least in our case.

6.2. Results and Findings

In the Linear Regression analysis and model selection we performed, after evaluating and testing the model on testing data set, we can find that RMSE value for every model is same, so we have a liberty to choose any of the models built. We are going with model built from Step Wise Regression.

In the Time Series analysis, we were able to forecast the future arrival time for the universal airlines.

KNN classification technique brought us to conclusion that it produces an overall accuracy of 62.72% and Naïve Bayes Classification technique brought us to conclusion that it produces an accuracy 79% for a feature i.e. on the selected row in testing data set. As our features in the data set are numeric, we would trust accuracy of KNN.

7. Conclusions and Future Work

7.1. Conclusions

The following conclusions are based on the above research problems mentioned.

Research problem 1: Regression analysis

Conclusion: From the Regression analysis we have done, we have predicted the arrival delay of universal airlines and from that we can infer that the arrival delay is dependent on the following factors:

1. SCHEDULED_DEPARTURE

2. DEPARTURE_DELAY

3. SCHEDULED_ARRIVAL

4. ARRIVAL_TIME

We have got the above model by performing model selection and we found out that by every model selection method (explained in above sections) we have got same result and therefore we moved on to evaluate our model based on testing

data set where we found out that every model has the same RMSE value of **0.6621076** and hence we have liberty to choose any model, by that means we have chosen the best model as Step Wise Regression.

Research problem 2: providing descriptive statistics

Conclusion: we have provided descriptive statistics and in that we had performed plotting of graphs, density chart, box plot, scatter plots.

First, we have got graph of flights which are on time and flights which are delayed. We can infer that the **no of flights which are on time are more than the no of flights which are delayed** which is a good sign for the universal airlines.

Second, we have got the density chart of flights and arrival delay. We can infer that there are more **no of flights which were/are on time and, there are no of flights which were/are delayed for about 1 to 50 minutes**. There are less no of flights which were/are delayed for about 100 to 150 minutes.

Third, we have done box plot analysis of three chosen carriers and their arrival delays. We can infer that the **carrier MQ is having the highest arrival delay as compared to the arrival delays of DL and DH carriers.** Further, if we look at remaining two carriers i.e. DH and DL, DH is having the second highest arrival delay as compared to DL and DL is having the lowest arrival delay. So, if I want to rate these carrier on basis of their punctuality, I would rate MQ carrier as 3/10 and DH carrier as 5/10 and DL carrier as 7/10.

Forth, we have plotted a scatter plot between arrival delay and departure delay for three chosen carriers like MQ, DL and DH. As we know that we have been predicting arrival delay on the function of other factors which affects the arrival delay. From the linear regression, we came to know that departure delay is indirectly a factor which influences arrival delay. Indirectly because departure delay is calculated from scheduled departure and departure time and these two have a direct effect on arrival delay. We can infer that as the **departure delay increases, the arrival delay also increases**. Because, departure delay will increase when the flight departs lately from the origin airport and hence it will arrive at the destination airport lately, so the arrival delay will also increase.

Fifth, we have plotted a plot between arrival delay and days in week for three chosen carriers like MQ, DH and DL. We can infer that irrespective of any carrier, there was a **maximum delay on Friday**.

Research problem 3: Hypothesis testing

Conclusion: The delay time when the weather is bad is greater than the delay time when the weather is good. In other words, we can say that there is more delay when the weather is bad as compared to when the weather is good.

Research problem 4: ANOVA

Conclusion: We can conclude that the arrival delay averages are not equal. The arrival time of a flight will differ according to various arrival delays.

Research problem 5: Time series analysis and predicting or forecasting the future arrival time.

Conclusion: by performing the time series analysis, we predicted or forecasted the future arrival time of carriers or flights. From the computed R output, we came to know that, or we forecasted that the future arrival time of flights will be 11 hours and 7 mins (1107) for at least next 10 rows in the data set. Also, we plotted the time plot of forecasted arrival time and we came to know that the time plot has no trend and there is no change in the variance.

Research problem 6: predicting the flight status, whether the flight will be on time or delayed.

Conclusion: to achieve the above research problem, we had to perform classification like KNN and Naïve Bayes to predict the flight status.

By using KNN approach: Using KNN classification technique, we can conclude that for any given set of features, there will be an **accuracy of 62.72%** in predicting the label or we can say that in predicting the status of a flight.

By using Naïve Bayes approach: Using Naïve Bayes Classification technique, we can conclude that if we take any set of features, we have predicted the label or flight status and it also shows up to what accuracy or probability the prediction is made on the flight status. As mentioned earlier, if you consider data 2197, then due to whatever features listed in that row 2197, the prediction will be on time and the probability of the prediction will be some log value and you must convert it to normal value by taking antilog. The normal value will be 79%. In other words, **we say that it is 79% accurate that the flight status will be on time or the probability that the flight status will be on time is 0.79** which is high, and we can trust it. As our features are numeric, so we would trust KNN or we can say that KNN is better model/technique as compared to Naive Bayes model/technique

7.2. Limitations

Model accuracy rate for regression analysis is 81%, for ANOVA model is 73.08% which is average, for KKN we have accuracy of 62.72% which is again moderate. we have got these accuracies of our models because we have limited no of rows in our data set i.e. we have less data. The above-mentioned accuracies of our models can be increases by adding more rows in the data set i.e. adding more data.

7.3. Potential Improvements or Future Work

There is certainly a potential in improving this analysis. For this in future, some new data should be added and along it some more new factors (columns) must be added to improve the prediction task and make the analysis more strong and accurate and reliable. We would like to add other factors like delay caused du to security issues and technical issues. This will help the analysis to cover some additional factors which have an important role too in the delay of a flight.