KULLIYYAH OF INFORMATION & COMMUNICATION TECHNOLOGY

CSC 3303 BIG DATA ANALYTICS
SECTION 01

TECHNICAL REPORT

"Airlines On-Time Performance Predictive Analysis"

PREPARED BY:
AHMAD (1526703)
MAHFUZEALAHI NOMAN (1515803)
ABID ENBA SAIF UTSHA (1433527)
MOHAMMAD NAFEES BIN ZAMAN (1616357)

LECTURER
DR. RAINI BINTI HASSAN

DUE

7th May 2019

Airlines on-Time Performance Predictive Analysis

Abid Ebna Saif Utsha, Ahmad, Mahfuzealahi Noman, Mohammad Nafees Bin Zaman
Dept. of Computer Science, IIUM, Gombak, Malaysia.
abidebnasaifutsha@gmail.com

Abstract— Airline delays are existent in all parts of the world and some people need to reach their destinations in a punctual manner especially when it relates to business related activities. This research aims to identify whether a flight will experience delay depending on a number of features. A Two-Class Boosted Decision Tree Algorithm has been implemented as the model to acquire the desired results. The results show that the model has an accuracy of 81.7%. The data product formed could be used for airline agency companies so that they can recommend customers that are in a hurry, to choose a suitable flight that would have less chances of having delays.
.

Keywords— Big Data Analytics, Machine Learning, Flight delay prediction, RStudio, Microsoft Azure, Two-Class Boosted Decision Trees.

I. Introduction

In today's globalized world, aviation airlines are used almost everywhere each day. Although technology has improved drastically throughout these recent years, flight delay issues are inevitable and are still quite common every week. Customers may have important meetings to attend do or may have some other reason to be on time, thus, it creates problems for their schedule as well as dampens the image of the specific airline business. Customer satisfaction should be one of the main priorities for these airlines and reducing delays or dealing with them by recommending a suitable flight could help to improve the situation.

This research focuses on analysing the airline delays that occurred through the year of 2015 for local flights in the USA and aims to provide results that relates to the business-domain of the aviation industry. The dataset acquired contains over 5.8 million entries and 31 columns. This data was collected from a total of 14 different airlines and their various flights. Our main goal is to apply predictive analytics and deduce whether a flight (in USA) would likely experience delay or not, depending on a number of features which include: Month, Airline, Day of week, Flight number, Origin Airport, Destination Airport and Scheduled Departure. The Two-Class Boosted Decision Tree Algorithm has been implemented to process this dataset and produce results for our classification. This analysis would then aid the information gathered from the results to be used as a recommendation for customers to choose the right flight to have better chances of avoiding delays if they need to reach their destination at a specific prescribed time. According to the Federal Aviation Administration (FAA), a flight is considered to have delay if it is 15 minutes later than its scheduled time, and this has been incorporated for our project.

II. Background

The Two-Class Boosted Decision Tree is supervised machine learning model that is based on the boosted decision trees algorithm, whereas in this case the predicted outcome can only be of two classes. When a decision tree is boosted, it is applying an ensemble learning method in which multiple trees are created. Each tree tries to correct the errors of the preceding tree in order to improve the prediction accuracy. The final prediction is based on all of the ensemble of trees combined together [1]. In many cases, if boosted decision trees are configured in the right manner, they can easily produce great performance and prediction accuracies for a wide variety of machine learning tasks. However, they do have a drawback in the fact that they tend to consume a lot of processing memory compared to other machine learning models. Due to this setback, a boosted decision tree model may not be capable of processing very large datasets on normal computers and would instead require it to be done on a special server or an online cloud.

III. Research question

Would a specific flight experience delay or not?

IV. Research hypothesis

Delays can be identified through the month, airline, day of week, flight number, origin airport, destination airport and scheduled departure of a flight.

V. RESEARCH OBJECTIVES

- Operate on RStudio and Microsoft Azure
- Perform Data Analytics
- Identify whether a flight would experience delay or not
- Evaluate outcome of results
- Provide recommendations

VI. RESEARCH SIGNIFICANCES

This research would help airline agency companies to use the model for recommending customers that need to arrive on time, to choose a suitable flight that would have less chances of having delays.

VII. LITERATURE REVIEW

Nigam and Govinda applied logistic regression on Microsoft Azure to predict flight delay [2]. They split the dataset into 70:30 ratios. The accuracy was 80.6 percent though their model were incapable of predicting flight delay with both precision and recall greater than 50 percent. For future work, the authors were planning to use other supervised model to see whether the accuracy will be improved or not.

Manna et al., made a model using gradient boosted decision tree to predict flight delay and got the accuracy above 94 percent [3]. They collected data from the U.S. Department of transformation which contained flight delay data for the period of April – October 2013. The model was limited because it was not able to predict flight delays with both precision and recall greater than 50 percent. Also, the limitation of the model was that it can predict the flight delay only for the 70 airports it had been trained with. For the next step, the authors proposed to increase the number of airports to train their model better.

Qianya, Lei, Rong, Bin and Xinhong applied Bayesian network model by series analysis on actual airline data to analyse flight delays [4]. The results showed the accuracy is high and the solution has good analysis as well as prediction reliability.

Liang and Li optimized a flight scheduling problem using combination of ant colony optimization and genetic algorithm [5]. The proposed model can effectively optimize flight departure. The simulation done by Liang and Li showed that the model can effectively reduce flight delay. The experimental result showed that compared to the First Come First Served (FCFS) algorithm and ant colony algorithm, the combination of ant colony algorithm and genetic algorithm can effectively shorten the total flight time consuming of departure and can get the best flight departure sequence. For future work, they proposed to use the model in real - life application and add multi-track departure sequence in their model.

Gopalakrishnan and Balakrishnan predicted delays in air traffic networks using three different models and compared which model will give the best results [6]. The three different models were Markov Jump Linear System (MJLS), Classification and Regression Trees (CART), Artificial Neural Network (ANN). For the classification problem at a two-hour prediction horizon and 60 min threshold, for the balanced dataset, the accuracy achieved by using ANN is 70 percent. However, by using other network delay states and Random Forest (RF), a much higher accuracy was achieved. For the future work, they were planning to use ensemble methods in ANN.

Khanmohammadi, Tutun and Kucuk predicted flight delays at JFK airport using Artificial Neural Network [7]. The performance measured was calculated using Root Mean Square Error (RMSE). In their dataset, they used inbound flights of JFK airport in January 2012, there were 1099 flights from 53 airports. The authors implemented ANN and got the RMSE around 0.13. In the future, they will consider the integration of the proposed method with fuzzy logic to expand the real-world application of the proposed method.

Li, Chen, Ge and Ning used deep learning technique to predict delay in airport. The technique they used was Long-Short Term Memory (LSTM) [8]. The dataset of experiment was based on actual operation data of Baiyun Airport in 2017. Experimental results show that compared with the traditional Neural Network model whose accuracy is 70.45 percent, the proposed prediction model has higher accuracy of 88.04 percent. In future, more information about airport data will be captured, more appropriate models will be created, and this method will be applied to data with greater magnitude of delays.

Takeichi, Kaida, Shimomura and Yamauchi predicted flight delay due to air traffic control using Artificial Neural Network (ANN) [9]. The performance measurement used in this article was Root Mean Square Error (RMSE) which was around 119. In the article, it was revealed that the ANN is able to predict the delay average value accurately in addition to predicting the tendency of increase and decrease. However, the ANN is unable to exactly predict drastic increase of the delay arising around the peak traffic volume time zones. ANN is unable to explicitly learn the propagation of the delay. In future, Recurrent Neural Network will be used which has the ability to learn the propagation of delay.

Belcastro, Marozzo, Talia and Trunfio predicted flight delays using data mining technique [10]. Belcastro et al., used MapReduce and Random Forest (RF) and got the accuracy of around 85 percent. The main goal of this article was to predict several days in advance the arrival delay of a scheduled flight due to weather conditions. Airline flight and weather observation datasets have been analysed and

TABLE I
LITERATURE REVIEW SUMMARY

No.	Year	Authors	Research problem	Main techniques applied	Results	Future Works (if any)
1.	2017	Nigam, R. & Govinda, K.	Cloud Based Flight Delay Prediction	Logistic Regression	Prediction accuracy: 80.6%	To use other supervised models
2.	2017	Manna, S., Biswas, S., Kundu, R., Rakshit, S., Gupta, P. & Barman, S.	Predicting Flight Delay	Gradient Boosted Decision Tree	Prediction accuracy: 94.8523%	Increasing the no. of airports
3.	2015	Qianya, L., Lei, W., Rong, F., Bin, W. & Xinhong, H.	Analysis for Flight Delays	Bayesian Network Model	Prediction Accuracy: High	
4.	2014	Liang, W. & Li, Y.	Optimization of Flight Scheduling Problem	Ant Colony Optimization and Genetic Algorithm	Model can effectively optimize flight departure	Practical application and multi-track departure sequencing
5.	2017	Gopalakrishnan , K. & Balakrishnan, H.	Predicting delays in air traffic networks	Markov Jump Linear System, Classification and Regression Trees, ANN	ANN accuracy – 70%	Using ensemble methods in ANN
6.	2016	Khanmohamm adi, S., Tutun, S. & Kucuk, Y.	Predicting flight delays at JFK airport	ANN	RMSE – 0.1366	Integration of fuzzy logic
7.	2018	Li, Z., Chen, H., Ge, J. & Ning, K.	Airport delay prediction Method	Long-Short Term Memory	Accuracy: LSTM – 88.04%, traditional NN – 70.45%	Apply this method with greater magnitude of delays
8.	2017	Takeichi, N.. Kaida, R.. Shimomura, A. & Yamauchi, T.	Prediction of delay due to air traffic control	ANN	RMSE – 119.9	Implement Recurrent Neural Network (RNN)
9.	2016	Belcastro, L., Marozzo, F., Talia, D. & Trunfio, P.	Scalable data mining for predicting flight delays	Random Forest, MapReduce	Accuracy – 85.8%	

mined using parallel algorithms implemented as MapReduce programs executed on a cloud platform. The results showed a high accuracy, for instance, with a delay threshold of 15 min, they achieved an accuracy of 74.2 percent and recall of 71.8 percent. While a threshold of 60 min, the accuracy is 85.8 percent and the delay recall is 86.9 percent. Moreover, if weather conditions were not considered, the model achieved an accuracy of 69.1 percent.

Most of the work that has been done on this area makes use of various machine learning techniques to get better results. Some of these techniques include Logistic Regression, Gradient Boosted Decision Tree, Bayesian Network Model, Genetic Algorithm etc. Among these algorithms, Gradient Boosted Decision Tree performed the best, with a prediction accuracy of 94.8523%. Many of those researches could be improved further, such as by using other supervised models, increasing the number of airports and practical applications, etc. Apart from that, another approach would be to apply a modified machine learning algorithm to get an improved accuracy.

In this project, we implement the two-class boosted decision tree algorithm. From the previous research, we have seen that implementing the gradient boosted decision

tree algorithm has yielded great prediction accuracy results, thus we would like to discover if a two-class boosted decision tree model could match the results of the gradient boosted decision tree model or even achieve a better result compared to that previous research. In addition to that, another key difference of our project compared to the previous researches conducted, would be that our main objective is to identify and predict the nature of delays depending on a number of different features.

VIII. METHODOLOGY

A. *Data Source*

Kaggle - 2015 Flight Delays and Cancellations [11]

- Contains 5.8million records and 31 features

B. *Tools Used*

- RStudio – for data pre-processing
- Microsoft Azure – for machine learning model

C. *Workflow*

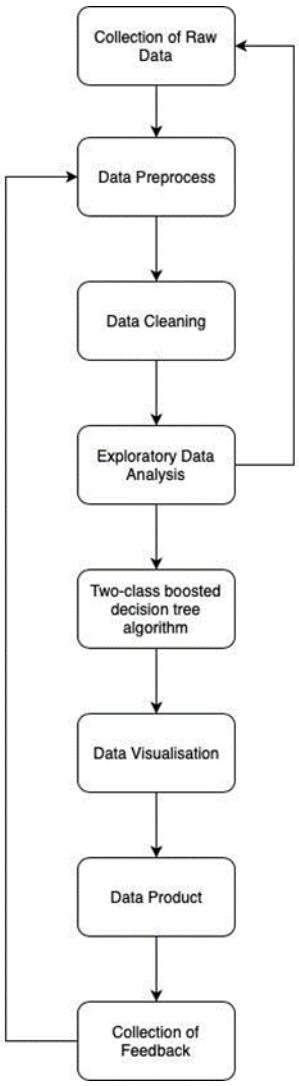

Fig 1: Diagram of this project's workflow

D. *Pre-processing and Visualization:*

- Converting 'ARRIVAL_DELAY' into 'DELAY_RESULT'
- Converting 'DELAY_RESULT' into factors
- Check for NULL values
- Remove cancelled flights and diverted flight records
- Selecting necessary features only
- Joining acronyms for both airlines and airports

```
> names(dataset)
 [1] "YEAR"                "MONTH"               "DAY"                 "DAY_OF_WEEK"         "AIRLINE"
 [6] "FLIGHT_NUMBER"       "TAIL_NUMBER"         "ORIGIN_AIRPORT"      "DESTINATION_AIRPORT" "SCHEDULED_DEPARTURE"
[11] "DEPARTURE_TIME"      "DEPARTURE_DELAY"     "TAXI_OUT"            "WHEELS_OFF"          "SCHEDULED_TIME"
[16] "ELAPSED_TIME"        "AIR_TIME"            "DISTANCE"            "WHEELS_ON"           "TAXI_IN"
[21] "SCHEDULED_ARRIVAL"   "ARRIVAL_TIME"        "ARRIVAL_DELAY"       "DIVERTED"            "CANCELLED"
[26] "CANCELLATION_REASON" "AIR_SYSTEM_DELAY"    "SECURITY_DELAY"      "AIRLINE_DELAY"       "LATE_AIRCRAFT_DELAY"
[31] "WEATHER_DELAY"       "DELAY_RESULT"
```

Fig 2: Exploring features

```
> #selecting necessary features only
> dataset <- dataset %>% select("MONTH", "AIRLINE", "DAY_OF_WEEK", "FLIGHT_NUMBER",
+           "ORIGIN_AIRPORT", "DESTINATION_AIRPORT", "SCHEDULED_DEPARTURE", "DELAY_RESULT")
> #checking dataset
> head(dataset)
  MONTH AIRLINE DAY_OF_WEEK FLIGHT_NUMBER ORIGIN_AIRPORT DESTINATION_AIRPORT SCHEDULED_DEPARTURE DELAY_RESULT
1     1      AS           4            98            ANC                 SEA                   5           No
2     1      AA           4          2336            LAX                 PBI                  10           No
3     1      US           4           840            SFO                 CLT                  20           No
4     1      AA           4           258            LAX                 MIA                  20           No
5     1      AS           4           135            SEA                 ANC                  25           No
6     1      DL           4           806            SFO                 MSP                  25           No
```

Fig 3: selecting features

```
> head(dataset)
  MONTH DAY_OF_WEEK FLIGHT_NUMBER SCHEDULED_DEPARTURE DELAY_RESULT                    AIRLINE
1     9           2          6168                2102          No Atlantic Southeast Airlines
2     7           4          5055                1412         Yes Atlantic Southeast Airlines
3    11           7          2582                2100          No        Delta Air Lines Inc.
4     8           4          5051                1008          No Atlantic Southeast Airlines
5    12           1          5969                1353         Yes Atlantic Southeast Airlines
6     9           7          6168                2102         Yes Atlantic Southeast Airlines
                                ORIGIN_AIRPORT                DESTINATION_AIRPORT
1            Chicago O'Hare International Airport Lehigh Valley International Airport
2                 Detroit Metropolitan Airport Lehigh Valley International Airport
3 Hartsfield-Jackson Atlanta International Airport Lehigh Valley International Airport
4                 Detroit Metropolitan Airport Lehigh Valley International Airport
5            Chicago O'Hare International Airport Lehigh Valley International Airport
6            Chicago O'Hare International Airport Lehigh Valley International Airport
>
```

Fig 4: Joint Acronyms for airline and airport

```
> prop.table(table(dataset$DELAY_RESULT))

       No       Yes
0.8138891 0.1861109
```

Fig 5: Probability of yes and no for 'DELAY_RESULT'

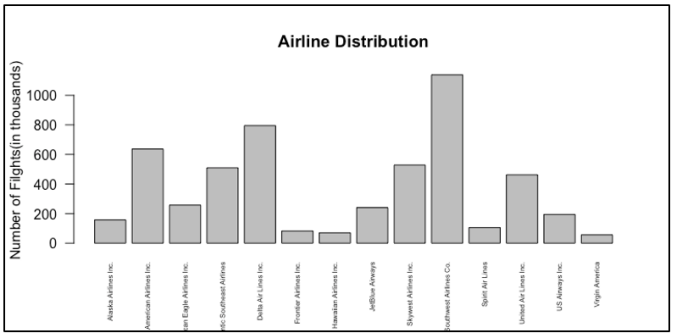

Fig 6: Airline Distribution (for clear picture please refer to Appendix A)

From the graph, it can be seen that Southwest Airlines Co. has been one of the most active airlines in 2015. While Hawaiian Airlines Inc., Virgin America and Frontier Airlines Inc. have had the least flights in that year.

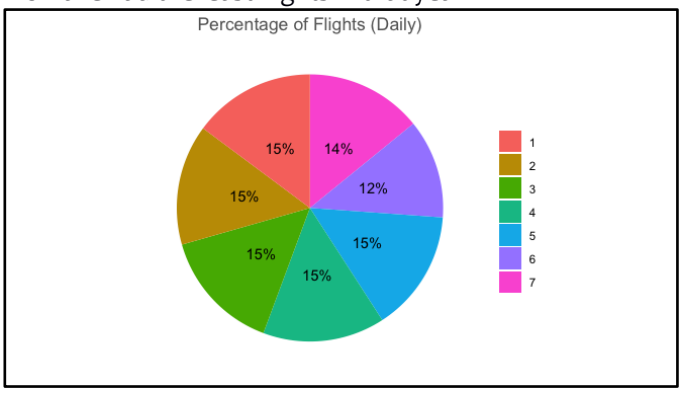

Fig 7: Percentage of flights (daily)

In Fig 7 (above), we look into how the percentage of flights that have occurred each day where Day 1 starts from Sunday. It can be seen that generally most of the days had a similar amount of flights. Saturday (Day 6) was the only day that experienced less flights compared to the other parts of the week.

From Fig 8, it can be seen that throughout the year of 2015, there are at least 400 thousand flights or more for each individual month. July (Month 7) topped the list having more than 500 thousand flights and this may be due to many US citizens having holiday during the summer season. Meanwhile, February (Month 2) had the least amount of flights with about 400 thousand showing that less people opted to travel during that month.

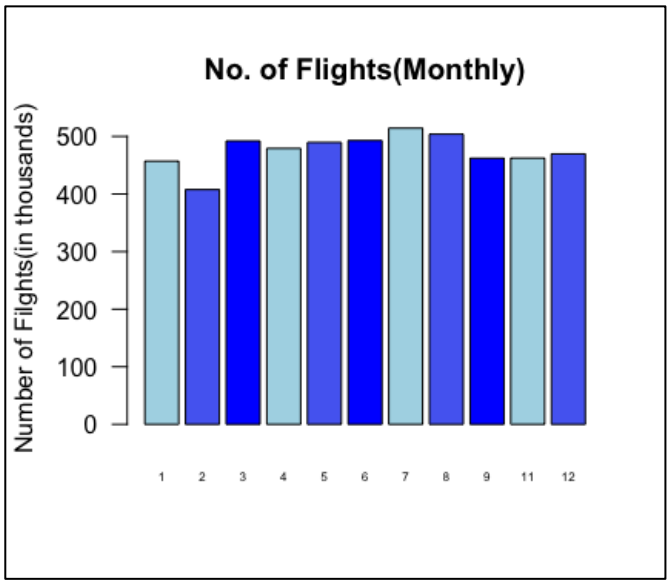

Fig 8: number of flights (monthly)

E. Machine Learning Model:

After pre-processing and EDA in RStudio, a Two-Class Boosted Decision Tree was implemented in Microsoft Azure. The model was set up with a total of 100 trees for the ensemble method and a learning rate of 0.2. Each tree until the 100th were used to improve the accuracy and reduce the errors of its preceding tree.

- Independent Variables:
  Month, Day of week, Flight Number, Scheduled Departure, Airline, Origin Airport, Destination Airport
- Dependent/Target Variable:
  Delay Result

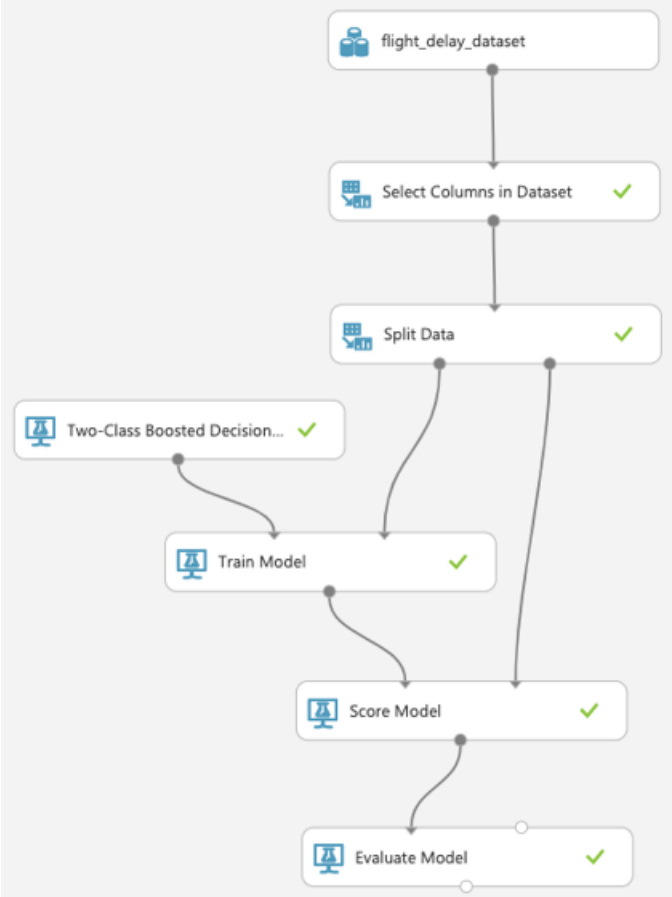

Fig 9: Two-class boosted decision tree model

IX. Results

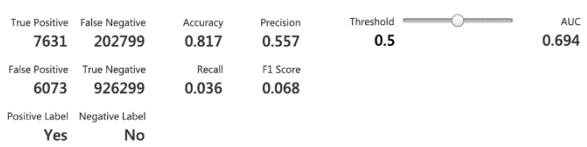

Fig 10: Results

From the figure above, it can be seen that the model achieved an accuracy of 81.7% with a precision rate of 55.7%. The recall is only 3.6%, which indicates that the model performed very poorly in terms of predicting the positive values ('Yes'). This is due to the data being overfitted with the negative values ('No') covering 81.4% of the dataset while positive values are only 18.6%, so the model is over trained for negative values. This results in a biased prediction for positive values. From the confusion matrix, it shows that it managed to predict 7631/13704 (55.6%) positive values correctly, while it predicted 926299/1129098 (82%) negative values correctly. Thus, the model can predict the negative values much better compared to positive values.

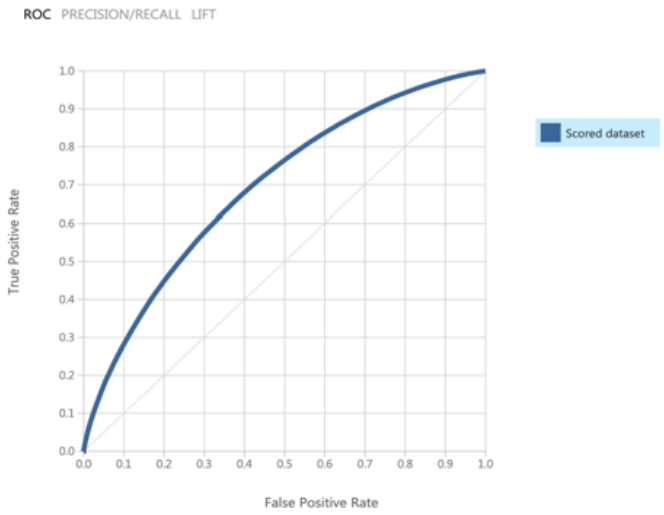

Fig 11: ROC curve

The ROC curve which is plotted from the False Positive Rate against the True Positive Rate, got an AUC score 69.4% which can be considered as an average result. That means, our model can distinguish between negative and positive class.

X. Data Product

Using the Azure ML Web Service (Fig 13), we can input the necessary features of a customer's flight to identify whether it will experience delay or not which achieved by the executed model. Airline agency companies could use the system to recommend customers on a suitable flight. The delay result field is not inputted as it is the feature that will be predicted by the model.

A. *Demo outcome of recommendation system*:

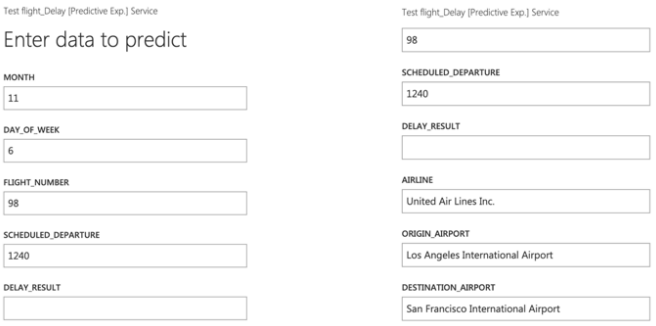

Fig 13: Recommendation System

From the example of values that we have inputted, it provided us with an answer 'No' (Fig 12) which indicates that that flight would not delay.

XI. Discussion

From a business perspective this type of recommendation system could prove to be very useful for airline agency companies for recommending a suitable flight to their customers if they require to reach their destination at a very specific time. Although the model has achieved an accuracy of 81.7%, it is mainly due to its capability of predicting correctly for those flights that will not be delayed. As for flights that could experience delay, the model only manages to predict approximately 50% of the time correctly due to the dataset being overfitted with the negative value compared to positive values. For this model to improve, it requires the dataset to be unbiased to ensure a better training for the machine learning model, hence providing better predictions. In short, the present recommendation system that we have acquired, would not be ideal since it only manages to predict the 'No Delay' correctly for most cases.

B. *Considering Big Data:*

Assuming that we are dealing with Big Data, we can't perform preprocessing and execution of our model in the traditional way. Instead, our first approach was to solve this issue by carrying out our work on RStudio Open Source Server which would be able to handle our very large data. After looking into that option, we found out that the server only runs on Linux machines, so we decided to look elsewhere since none of us are using Linux OS on our laptops. Next, we searched more options and finally came up with a solution. The solution was to instead use Google BigQuery as the data warehouse. This is a suitable option for us because it provides 10GB free space for each month as mentioned in the website, allowing us to make use of it

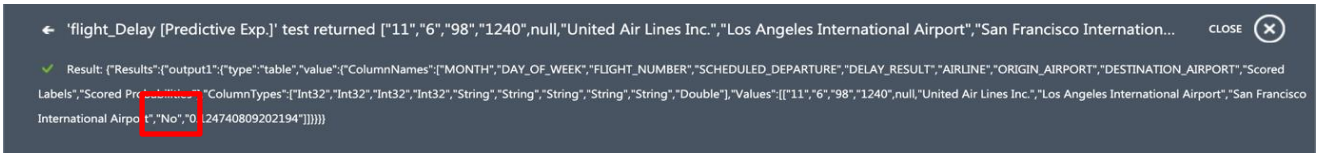

Fig 12: Outcome of prediction

without having to pay any charges. In addition to that, the 'dplyr' package in RStudio allows us to directly connect to the Google BigQuery server where our dataset would be available. We can then perform pre-processing and develop our model directly in RStudio by connecting it to Google BigQuery to retrieve our dataset.

XII. CONCLUSION

Due to overfitted data, this model is not completely unbiased and may not always provide the best prediction of whether a flight will delay or not. In general terms, the Two-Class Boosted Decision Tree itself worked well, but due to the negative values being too much, the model was not trained very effectively. The accuracy achieved was 81.7% along with a precision of 55.7%. Although the accuracy is high, the recall is only 3.6% which indicates that the model struggles with the positive values in the dataset.

XIII. FUTURE WORK

For future improvements, we would like to use a dataset that is not overfitted to ensure an unbiased model. Furthermore, to improve scalability we must add more records of international airlines so that customers from many parts of the world could engage with this recommendation system. Additionally, from previous works, we have found, a lot of researchers are moving towards implementing deep learning in their existing model. We would also like to implement deep learning technique to improve the accuracy and reliability of our model.

ACKNOWLEDGEMENT

We would like to thank dr. Raini Hassan for her constant guiding throughout this project.

REFERENCE

[1] Xiaoharper. (2018). Two-Class Boosted Decision Tree - Azure Machine Learning Studio. Available: https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/two-class boosted-decision-tree

[2] Nigam, R., & Govinda, K. (2017, December). Cloud based flight delay prediction using logistic regression. In 2017 International Conference on Intelligent Sustainable Systems (ICISS) (pp. 662-667). IEEE.

[3] Manna, S., Biswas, S., Kundu, R., Rakshit, S., Gupta, P., & Barman, S. (2017, June). A statistical approach to predict flight delay using gradient boosted decision tree. In 2017 International Conference on Computational Intelligence in Data Science (ICCIDS) (pp. 1-5). IEEE.

[4] Qianya, L., Lei, W., Rong, F., Bin, W., & Xinhong, H. (2015, May). An analysis method for flight delays based on Bayesian network. In The 27th Chinese Control and Decision Conference (2015 CCDC) (pp. 2561-2565). IEEE.

[5] Liang, W. & Li, Y. (2014, June). Research on optimization of flight scheduling problem based on the combination of ant colony optimization and genetic algorithm. In 2014 IEEE 5th International Conference on Software Engineering and Service Science (pp. 296 299). IEEE.

[6] Gopalakrishnan, K., & Balakrishnan, H. (2017). A comparative analysis of models for predicting delays in air traffic networks. ATM Seminar.

[7] Khanmohammadi, S., Tutun, S., & Kucuk, Y. (2016). A new multilevel input layer artificial neural network for predicting flight delays at JFK airport. Procedia Computer Science, 95, 237-244.

[8] Li, Z., Chen, H., Ge, J., & Ning, K. (2018, November). An Airport Scene Delay Prediction Method Based on LSTM. In International Conference on Advanced Data Mining and Applications (pp. 160-169). Springer, Cham.

[9] Takeichi, N., Kaida, R., Shimomura, A., & Yamauchi, T. (2017). Prediction of delay due to air traffic control by machine learning. In AIAA Modeling and Simulation Technologies Conference (p. 1323).

[10] Belcastro, L., Marozzo, F., Talia, D., & Trunfio, P. (2016). Using scalable data mining for predicting flight delays. ACM Transactions on Intelligent Systems and Technology (TIST), 8(1), 5.

[11] Department of Transportation. "2015 Flight Delays and Cancellations." Kaggle. February 09, 2017. https://www.kaggle.com/usdot/flight-delays#flights.csv.

APPENDIX A: (FIGURES)

```
> names(dataset)
 [1] "YEAR"                "MONTH"               "DAY"                 "DAY_OF_WEEK"         "AIRLINE"
 [6] "FLIGHT_NUMBER"       "TAIL_NUMBER"         "ORIGIN_AIRPORT"      "DESTINATION_AIRPORT" "SCHEDULED_DEPARTURE"
[11] "DEPARTURE_TIME"      "DEPARTURE_DELAY"     "TAXI_OUT"            "WHEELS_OFF"          "SCHEDULED_TIME"
[16] "ELAPSED_TIME"        "AIR_TIME"            "DISTANCE"            "WHEELS_ON"           "TAXI_IN"
[21] "SCHEDULED_ARRIVAL"   "ARRIVAL_TIME"        "ARRIVAL_DELAY"       "DIVERTED"            "CANCELLED"
[26] "CANCELLATION_REASON" "AIR_SYSTEM_DELAY"    "SECURITY_DELAY"      "AIRLINE_DELAY"       "LATE_AIRCRAFT_DELAY"
[31] "WEATHER_DELAY"       "DELAY_RESULT"
```

Fig 2: Exploring features

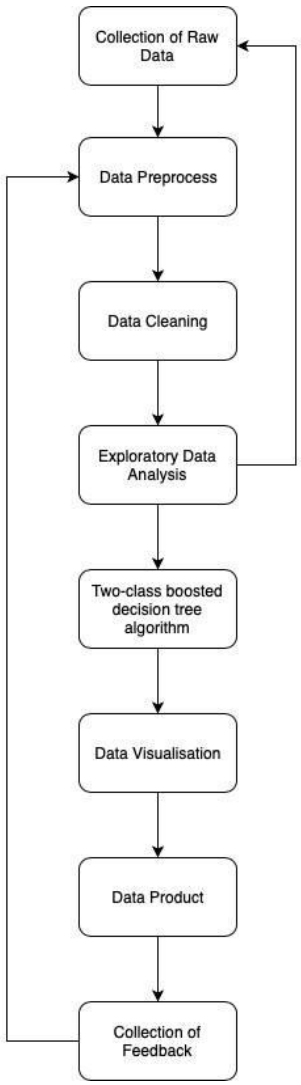

Fig 1: Diagram of this project's workflow

```
> #selecting necessary features only
> dataset <- dataset %>% select("MONTH", "AIRLINE", "DAY_OF_WEEK", "FLIGHT_NUMBER",
+             "ORIGIN_AIRPORT", "DESTINATION_AIRPORT", "SCHEDULED_DEPARTURE", "DELAY_RESULT")
> #checking dataset
> head(dataset)
  MONTH AIRLINE DAY_OF_WEEK FLIGHT_NUMBER ORIGIN_AIRPORT DESTINATION_AIRPORT SCHEDULED_DEPARTURE DELAY_RESULT
1     1      AS           4            98            ANC                 SEA                   5           No
2     1      AA           4          2336            LAX                 PBI                  10           No
3     1      US           4           840            SFO                 CLT                  20           No
4     1      AA           4           258            LAX                 MIA                  20           No
5     1      AS           4           135            SEA                 ANC                  25           No
6                                                      SFO                 MSP                  25           No
```

Fig 3: selecting features

Fig 14: Probability of yes and no for 'DELAY_RESULT'


```
> head(dataset)
  MONTH DAY_OF_WEEK FLIGHT_NUMBER SCHEDULED_DEPARTURE DELAY_RESULT                   AIRLINE
1     9           2          6168                2102          No Atlantic Southeast Airlines
2     7           4          5055                1412         Yes Atlantic Southeast Airlines
3    11           7          2582                2100          No        Delta Air Lines Inc.
4     8           4          5051                1008          No Atlantic Southeast Airlines
5    12           1          5969                1353         Yes Atlantic Southeast Airlines
6     9           7          6168                2102         Yes Atlantic Southeast Airlines
                              ORIGIN_AIRPORT                DESTINATION_AIRPORT
1            Chicago O'Hare International Airport Lehigh Valley International Airport
2                  Detroit Metropolitan Airport Lehigh Valley International Airport
3 Hartsfield-Jackson Atlanta International Airport Lehigh Valley International Airport
4                  Detroit Metropolitan Airport Lehigh Valley International Airport
5            Chicago O'Hare International Airport Lehigh Valley International Airport
6            Chicago O'Hare International Airport Lehigh Valley International Airport
>
```

Fig 4: Joint Acronyms for airline and airport

```
> prop.table(table(dataset$DELAY_RESULT))

       No       Yes
0.8138891 0.1861109
```

Fig 5: Probability of yes and no for 'DELAY_RESULT'

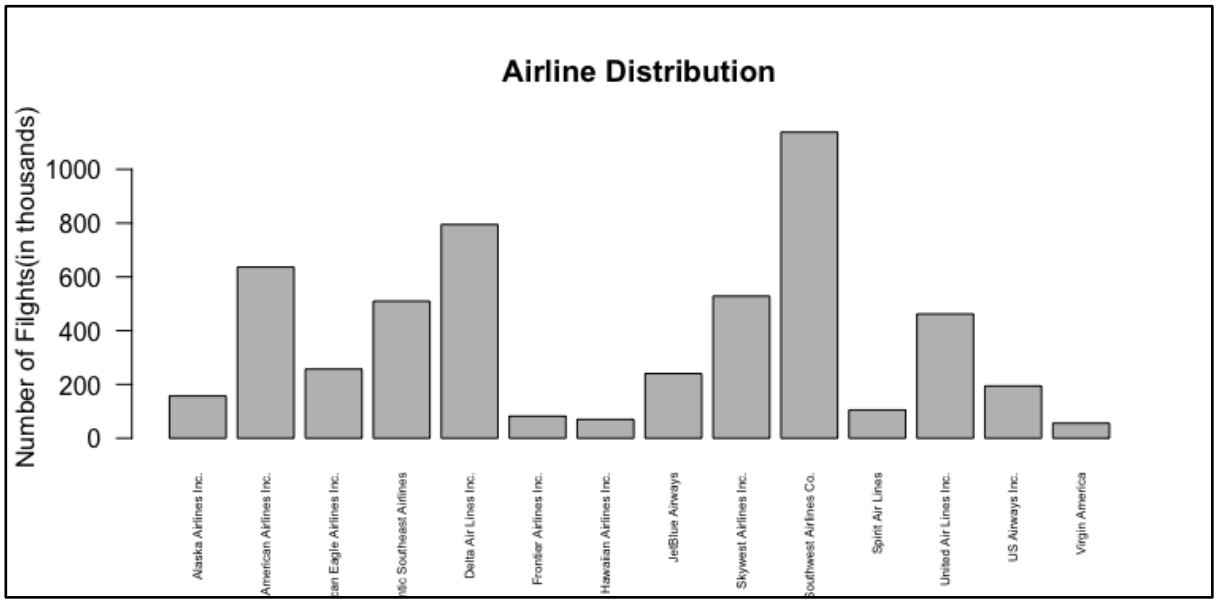

Fig 6: Airline Distribution

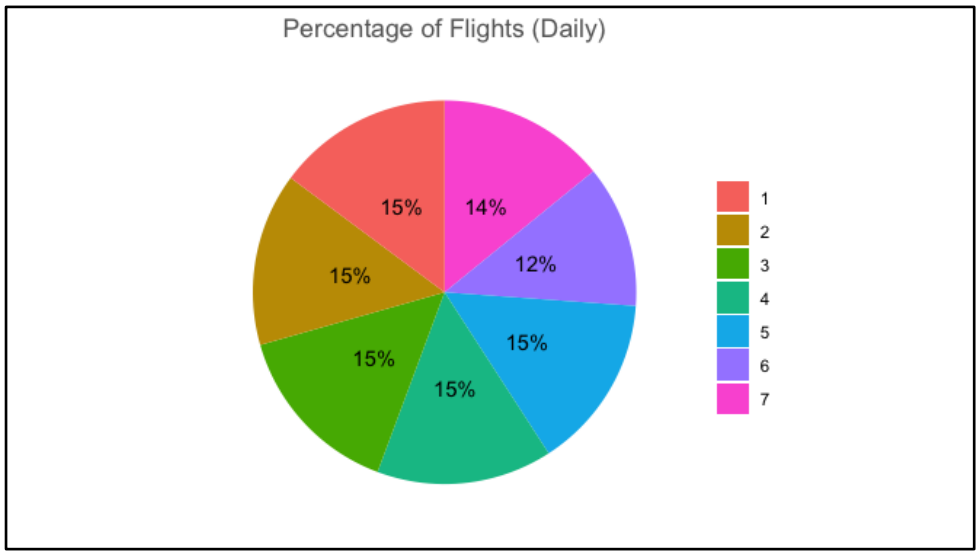

Fig 7: Percentage of flights (daily)

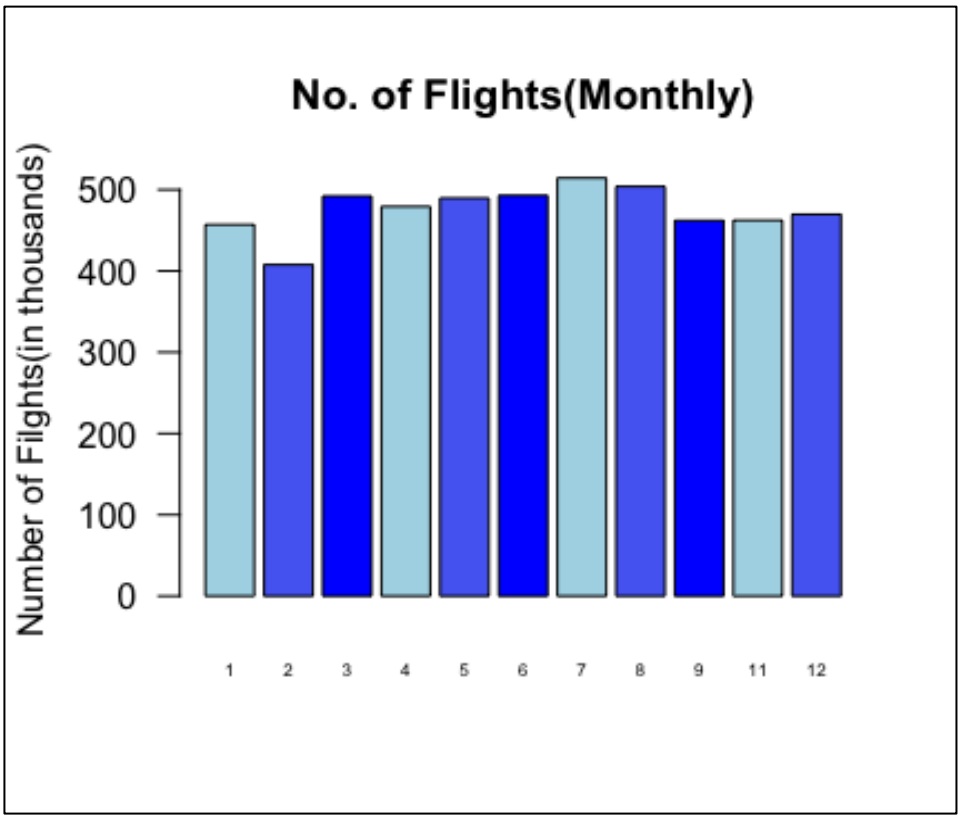

Fig 815: number of flights (monthly)

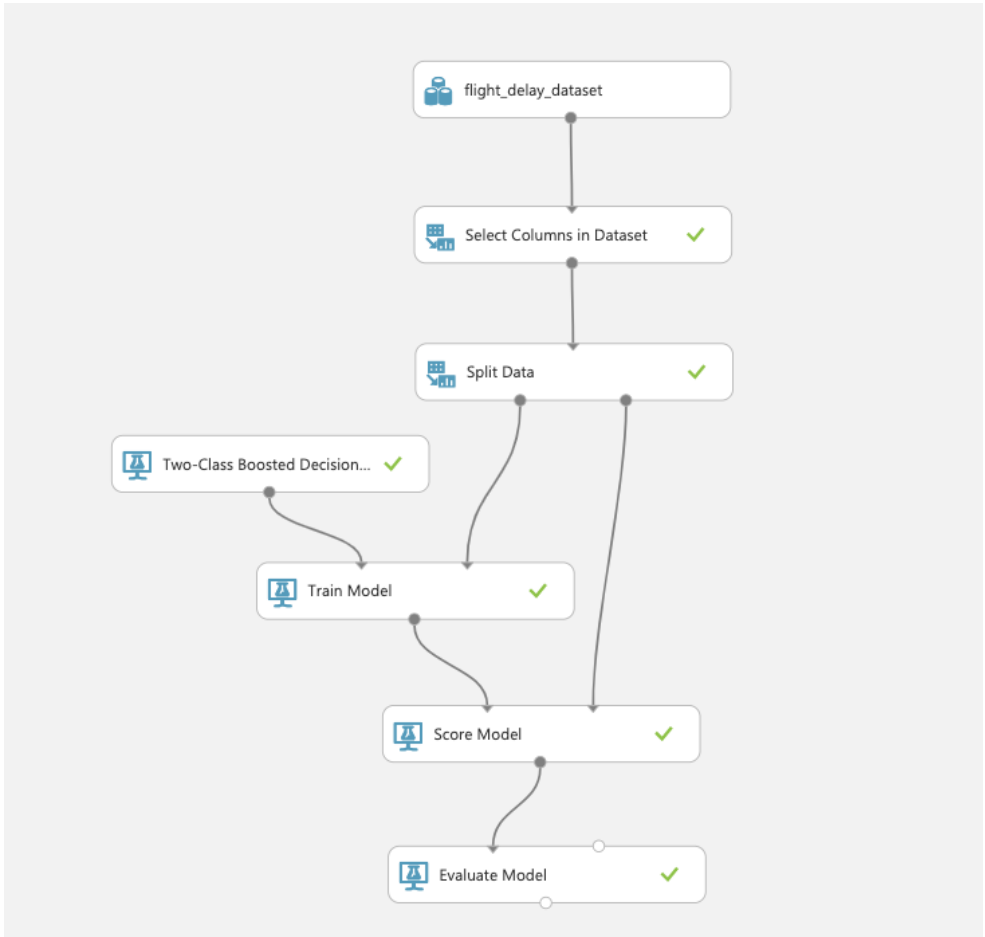

Fig 9: Two-class boosted decision tree model

True Positive	False Negative	Accuracy	Precision	Threshold	AUC
7631	202799	0.817	0.557	0.5	0.694

False Positive	True Negative	Recall	F1 Score
6073	926299	0.036	0.068

Positive Label	Negative Label
Yes	No

Fig 10: Results

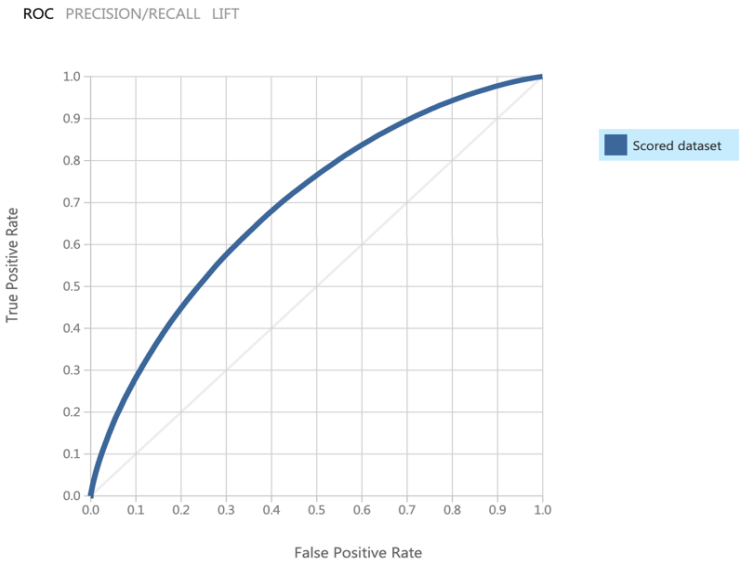

Fig 11: ROC curve

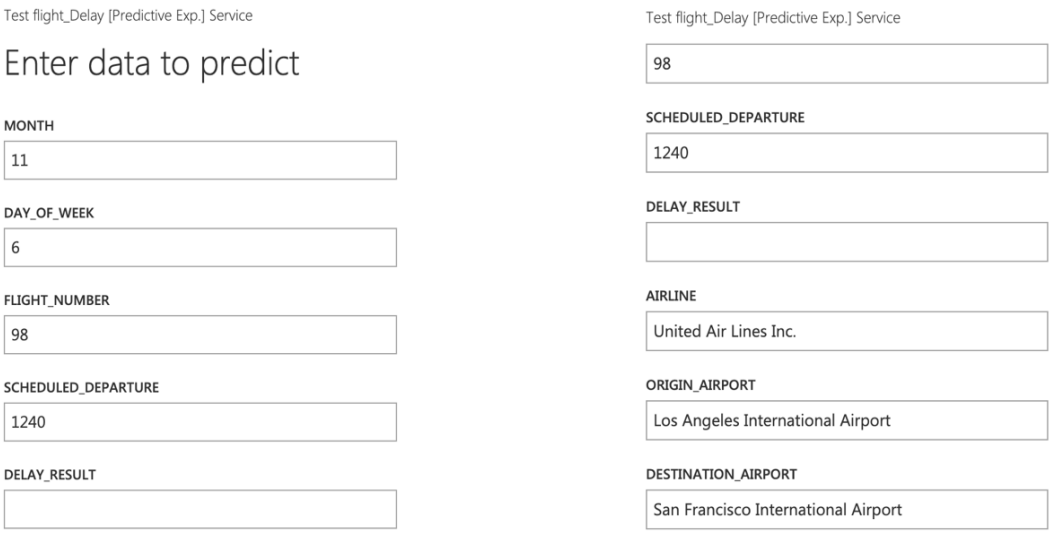

Fig 13: Recommendation System

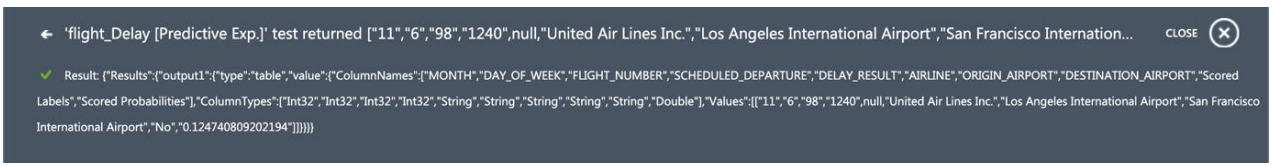

Fig 12: Outcome of prediction

APPENDIX B (SOME PARTS OF R CODE)

```
library(tidyverse)
library(dplyr)
library(caret)
library(rpart)
library(rpart.plot)
library(caTools)
library(tree)
library(hms)
library(party)
library(gbm)
library(ggplot2)
library(plyr)

#reading the flights dataset
dataset <- read.csv("flights.csv")
airline <- read.csv("airlines.csv")
airport <- read.csv("airports.csv")

#deducting all rows which have the records of cancellation and diversion
dataset <- dataset %>% filter(CANCELLED == 0) %>% filter(DIVERTED == 0)

#adding new column of delay result
dataset <- mutate(dataset, DELAY_RESULT = ARRIVAL_DELAY)

#categorizing delay result from numeric to yes or no
dataset$DELAY_RESULT <-ifelse(dataset$DELAY_RESULT<15, "No", "Yes")

#converting delay result to factor
dataset$DELAY_RESULT <- as.factor(dataset$DELAY_RESULT)

#explore dataset
head(dataset$DELAY_RESULT)
names(dataset)
class(dataset$DELAY_RESULT)
typeof(dataset$DELAY_RESULT)
prop.table(table(dataset$DELAY_RESULT))
nrow(dataset)


#selecting necessary features only
dataset <- dataset %>% select("MONTH", "AIRLINE", "DAY_OF_WEEK", "FLIGHT_NUMBER",
        "ORIGIN_AIRPORT", "DESTINATION_AIRPORT", "SCHEDULED_DEPARTURE", "DELAY_RESULT")

#checking dataset
head(dataset)
head(airline)
head(airport)

#changing column airline's value
colnames(dataset)[colnames(dataset) == "AIRLINE"] <- "IATA_CODE"
dataset <- merge(dataset, airline, by.x="IATA_CODE", by.y="IATA_CODE")
dataset <- dataset %>% select(- "IATA_CODE")


#changing column origin_airport's value
colnames(dataset)[colnames(dataset) == "ORIGIN_AIRPORT"] <- "IATA_CODE"
```

```
dataset <- merge(dataset, airport, by.x="IATA_CODE", by.y="IATA_CODE")
dataset <- dataset %>% select(- c("IATA_CODE", "CITY", "STATE", "COUNTRY", "LATITUDE", "LONGITUDE"))
colnames(dataset)[colnames(dataset) == "AIRPORT"] <- "ORIGIN_AIRPORT"


#changing column DESTINATION_AIRPORT's value
colnames(dataset)[colnames(dataset) == "DESTINATION_AIRPORT"] <- "IATA_CODE"
dataset <- merge(dataset, airport, by.x="IATA_CODE", by.y="IATA_CODE")
dataset <- dataset %>% select(- c("IATA_CODE", "CITY", "STATE", "COUNTRY", "LATITUDE", "LONGITUDE"))
colnames(dataset)[colnames(dataset) == "AIRPORT"] <- "DESTINATION_AIRPORT"



#EDA
counts <- table(dataset$AIRLINE)/1000
barplot(counts, main="Airline Distribution",
      ylab="Number of Filghts(in thousands)", las=2, cex.names=.5)


pal <- colorRampPalette(colors = c("lightblue", "blue"))(3)
counts <- table(dataset$MONTH)/1000
barplot(counts, main="No. of Flights(Monthly)", col = pal,
      ylab="Number of Filghts(in thousands)", las=1, cex.names=.5)


pal <- colorRampPalette(colors = c("lightgreen", "green"))(3)
counts <- table(dataset$DAY_OF_WEEK)/1000
barplot(counts, main="No. of Flights(Daily)", col = pal,ylim=c(0,800),
      ylab="Number of Filghts(in thousands)", las=1, cex.names=.5)


w <- table(dataset$DAY_OF_WEEK)
t = as.data.frame(w)
t$Freq <- t$Freq/5231130

pie = ggplot(t, aes(x="", y=Freq, fill=Var1)) + geom_bar(stat="identity", width=1)
pie = pie + coord_polar("y", start=0) + geom_text(aes(label = paste0(round(t$Freq*100), "%")), position = position_stack(vjust
= 0.5))
pie = pie + labs(x = NULL, y = NULL, fill = NULL, title = "Percentage of Flights (Daily)")
pie = pie + theme_classic() + theme(axis.line = element_blank(),
                                    axis.text = element_blank(),
                                    axis.ticks = element_blank(),
                                    plot.title = element_text(hjust = 0.5, color = "#666666"))

pie


#find index to suffle all rows
shuffle_index <- sample(1:nrow(dataset))
head(shuffle_index)

#suffle all rows
dataset <- dataset[shuffle_index, ]
head(dataset)

#creating traing an testing dataset
training_index <- createDataPartition(dataset$DELAY_RESULT, p=0.80, list=FALSE)
```

```
training_data <- dataset[training_index,]
testing_data <- dataset[-training_index,]

head(training_data)
head(testing_data)

#probability of YES and NO
prop.table(table(dataset$DELAY_RESULT))


#tree
tree <- rpart(DELAY_RESULT~., data = training_data, method = 'class')
rpart.plot(tree, extra = 106)

predict_unseen <-predict(tree, testing_data, type = 'class')

table_mat <- table(testing_data$DELAY_RESULT, predict_unseen)
table_mat

accuracy_Test <- sum(diag(table_mat)) / sum(table_mat)
accuracy_Test
```