**FINAL PROJECT**

**Project Link: http://team3flightpredication.azurewebsites.net/**

**Flight Analysis**

**INFO 7390: Advances in Data Sciences/Architecture**

**Team 3**

**Sankalp Jadon**

**Vivek Kumar**

**Vyshak Srishylappa**

**Table of Contents**

1. Introduction

* Problem description and motivation
* Business case

1. Data preprocessing and visualization

* Dataset Description
* Data wrangling and cleaning
* Data visualization

1. Price prediction Model

* Technique used and approach
* Analysis

1. Arrival Delay Time Prediction

* Process and Approach
* Prediction Analysis

1. Flight Cancellation Prediction

* Process and Approach
* Prediction Analysis

1. Recommender system for flights.

**I. INTRODUCTION**

**Problem description and Motivation**

With the increasing list of airway passengers each year, naturally the sky space is also getting crowed with the increase in the number of aircrafts. If a flight gets delayed due to any unforeseen reason, it’ll result in delay of many other flights by hindering in their schedules. The idea to predict the flight delay in advance and inform travelers beforehand is very useful in that regards.

As delays can never be predicted precisely, it is interesting to study their entire probability distributions. Many factors affect flight delays including air traffic control issues, weather conditions, equipment delays, etc.

Our objective was to predict the airline delays. More specifically, to identify if the flight is going to be delayed or not, what is the predicted delay time for a particular flight. Moreover, since we had presented on recommendation systems before in the class, we implemented a smart recommender systems to recommend the best pricing available for travellers. So, based on the analysis of our prices, we can recommend the best priced flights for a particular day.

**Business Case**

We were approached by the United States Department of Aviation to help them to sort out the problem of increasing delays in flights. Some problems faced by our clients were:

1. Crowded airspace becoming unpredictable.
2. Rescheduling of critical government air space operations because of delays.
3. Problems in liaisoning between US military and the civilian Air Traffic Control because of sudden delays.
4. Bad customer satisfaction for US residents.
5. Sudden surge/decrease in the airfare.

Based on some of these problems, the client’s requirements were clear:

a) To establish a classification model to predict the cancellation of airlines on a particular day.

b) To predict the Arrival delay in minutes of flights leaving on a particular day.

c) To help the government with a recommender system for pricings so that any irregularities can be noticed by different carriers.

a) Flight Cancellation Prediction

Our first business case was to predict whether a flight would be cancelled or not.

Using this predictive model, the government can schedule their critical operations suitably. We should not forget that the airspace traffic is the major cause of delays in many operations.

* Solution:-

We use various classification algorithms to classify weather the flight is going to be cancelled or not.

b) Arrival Delay Time Prediction

The second business case would be to predict the arrival delay time for the flights.

By having an insight of flight delays from one place to another, Aviation department can easily coordinate with the US Military so that various critical operations can be carried out without effecting the travellers. Moreover, flight delays are an inconvenience to passengers. Anger, frustration and a bad overall experience led the United States Government to step in.

* Solution:-

We have used various regression algorithms to predict the flight arrival delay. We have then chosen the best model and integrated with the website.

c) Flight Price Prediction

Our final business case would be to predict the prices for the flights for their destinations.

The government noticed that the flight carriers are charging passengers based on their whims. They decided to put a regulation in airfare prices for which, the government needed a pricing model, which takes in the average price of the flight over a long duration to set it as benchmark for those locations. This would ultimately lead to fair airfare charges by carriers. The carrier’s decision to price the ticket will be reduced to a certain degree, and will lead to more happy travelers.

* Solution: -

we have used previous years price data for each airlines to a particular destination to predict the airfare. This data can be used as a guideline for the government to set a benchmark for pricing by different carriers.

**II. DATA PROCESSING AND VISUALISATION**

**Dataset Description**

The final dataset is a result of merger from various sources like flight details from Bureau of Transportation and Statistics (BTS), Wunderground API(Weather data), price data. Final dataset contains detailed attributes of a particular flight, merged from all the above sources. This includes various features like Destination, Origin, Arrival time, and Departure time and so on.

Following is the final dataset and their sources-

a) Initial Dataset-

1. Year- Year Format (After breaking down of date attribute)
2. Month- Number Format (After breaking down of date attribute)
3. Day of month- Number Format (After breaking down of date attribute)
4. Day of week- Number Format (After breaking down of date attribute)
5. Departure Time- Number Format
6. Scheduled Departure Time- Number Format
7. Arrival Time- Number Format
8. Scheduled Arrival Time- Number Format
9. Unique carrier- Alphanumeric Format
10. Flight Number- Number Format
11. Tail Number- Alphanumeric Format
12. Actual Elapsed Time- Number Format
13. Scheduled Elapsed Time- Number Format
14. Air Time- Number Format
15. Arrival Delay- Number Format
16. Departure Delay- Number Format
17. Origin - String Format
18. Destination- String Format
19. Distance- Number Format
20. Taxi In- Number Format
21. Taxi Out- Number Format
22. Cancelled- Number Format
23. Diverted- Number Format
24. Hour- Number Format
25. Quarter- Number Format

b) Data from Wunderground API-

1. Temperature- Number Format
2. Dew Point- Number Format
3. Humidity- Number Format
4. Sea Level- Number Format
5. Visibility- Number Format
6. Wind Direction- Number Format

c) Price data from another data source-

1. Average Price- Number Format

The final dataset contained csv files which had the monthly data. Since the number of records were huge in each month (around 0.5 million), we decided to limit the scope of the dataset to only 1 year of data i.e. 2014. Based on our business case scenario, we were focused more on building an application which our client would be benefitted. For this we manually extracted 16 columns according to business context which would mostly help in making a decision about price, delay and cancellation of flights.

**Data Wrangling and Cleaning**

We used R to process the data before uploading it on Azure to build models.

* 1. The first step in the process was to analyze the data by trying to understand the relationships between various attributes.
  2. The second step was to split the columns to make it easy to pass values to our models in azure. For example, we split the date attribute into Year, Month, Day of month, Day of week.
  3. Next, we decided to exclude redundant data from our dataset. Features like tail num, which would not add importance to our project.
  4. The next step in the data wrangling process was to check for the null values, and replacing them with approximates using the mean. We noticed that there were a lot of missing and null values.
  5. Then we looked for outliers in our data and processed in R.
  6. Next step was to use wunderground API to scrape the weather data and merge the data fetched from there to our main dataset based on the date and time.
  7. Finally we used gathered the price data and merged it based on the route (Origin and Destination)

The ne asd

* 1. Cancellation code column had a lot of missing data. So we filled this data by filling the missing values as On-Time.
  2. There were missing values in the price column also. Since we had around half a million records, the amount of missing data in price column were comparatively negligible which helped us in deciding to remove the rows with missing price values.

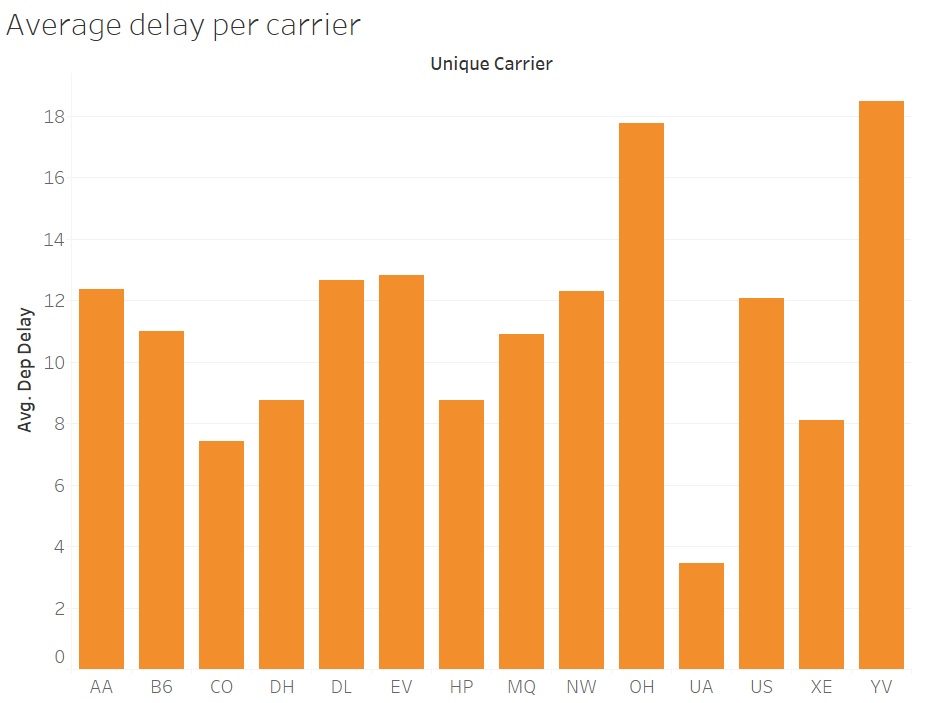
1. **Data Visualization**

The visualization was done using Tableau. We imported our entire dataset on tableau and tried to plot different graphs to make sense of how different features are related to each other.

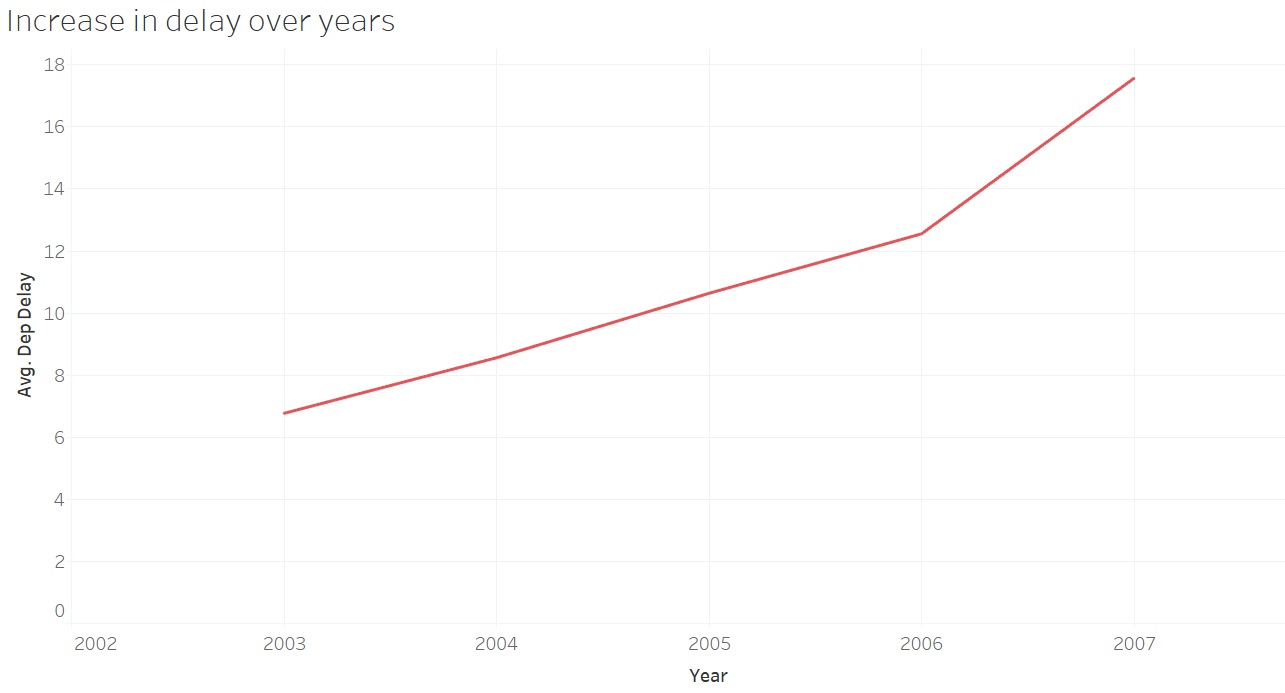
Following are the graphs-

1. The first graph relates average departure delay and the unique carriers.

We notice that average delay for carrier- UA (Code for United Airlines is very low), whereas the average delay for YV( Code for Mesa airlines is very high).

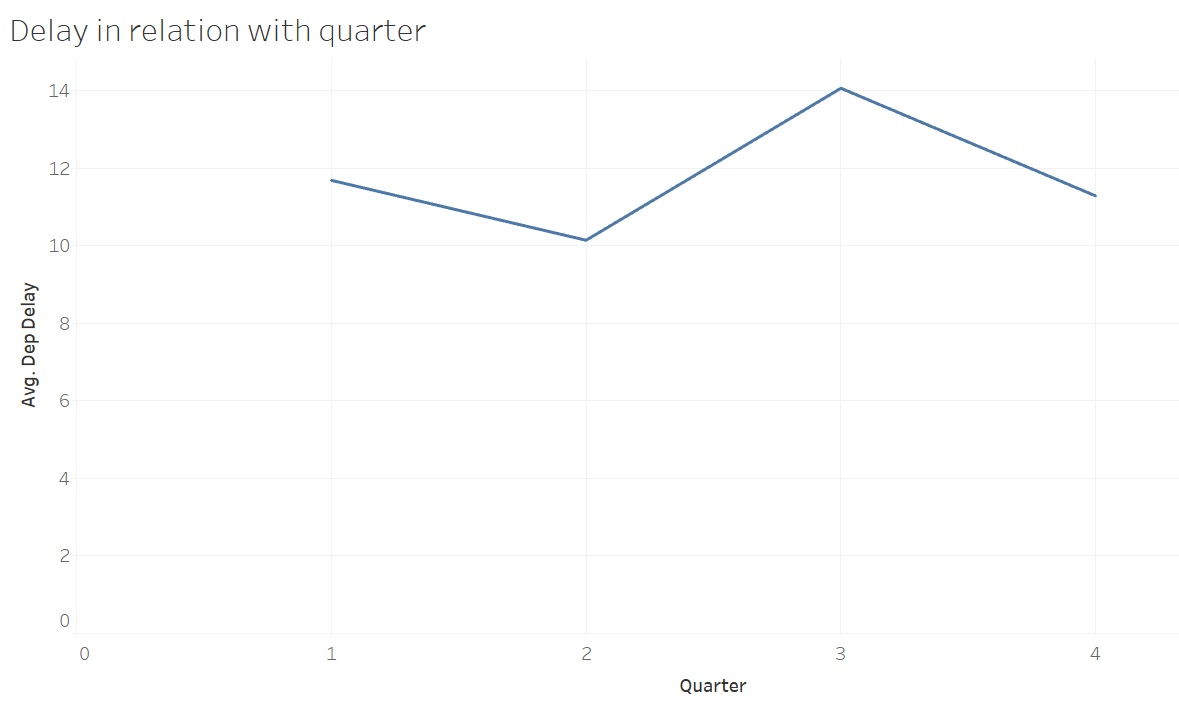


1. Following graph shows the problem of increasing number of delays over years. One possible reason is that the air space has become considerably crowded.



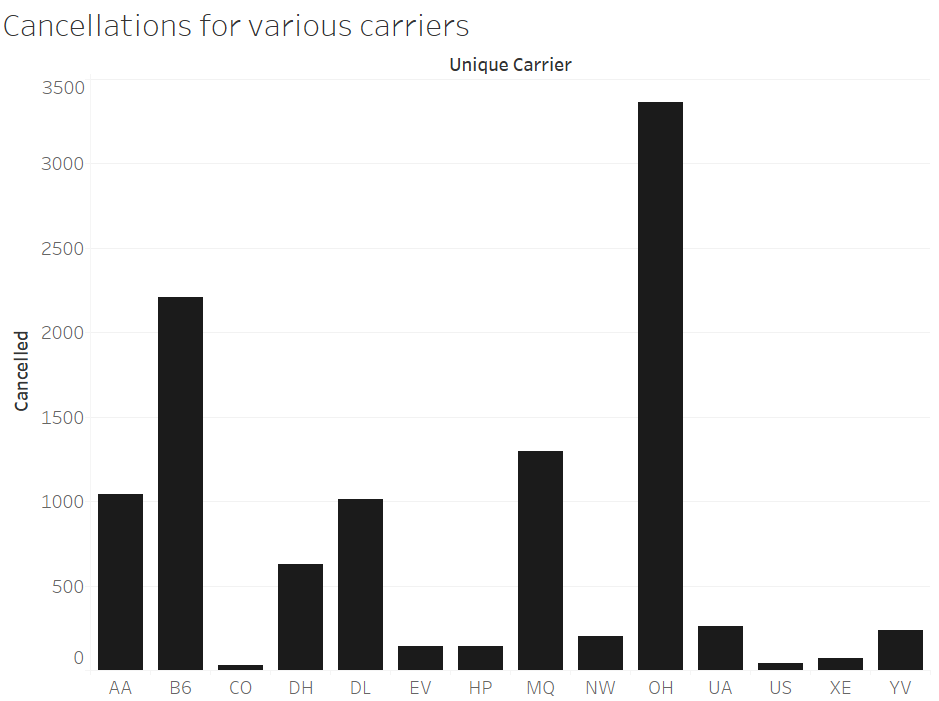
1. Average departure delay for each quarter.

We notice here that the average departure delay is high for winters, which usually have adverse weather conditions. Whereas, the delay is least in summers.



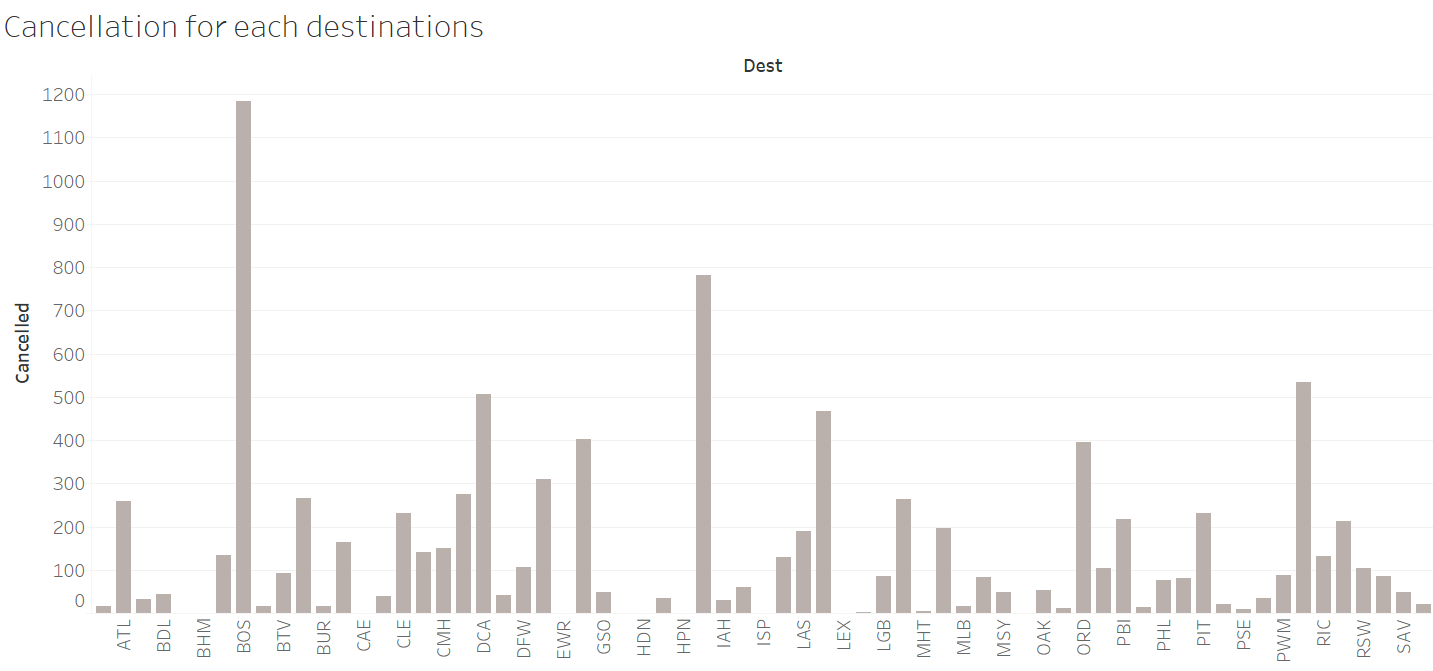
1. Cancellations for various carriers

We notice that there are certain airlines which have very high number of cancellations versus those with very small number of cancellations. This counts towards user satisfaction, and ultimately towards the recommendation system that we have built based on user ratings.



1. Cancellation for each destination

Following is a graph showing the cancellations per destination city.



1. Average price of each carrier

The following graph shows average price of each airline and the unique carrier code.

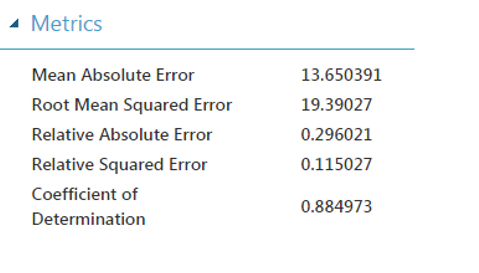


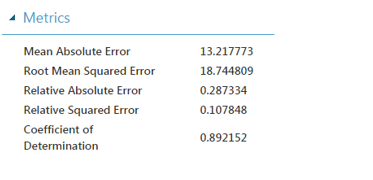
**III. PRICE PREDICTION MODEL**

1. Technique used and approach
2. We upload our cleansed dataset on azure first.
3. We then used filter based feature selection to select the features based on the weights of features on the dependent variable (Price).
4. On evaluation, the data we found that the main predictors were- Unique Carrier, destination, month, Origin and destination end date, carrier to predict the price.
5. Next, we split our data into test and train based on randomized split.
6. We used the following models for price prediction-
7. Boosted Decision Tree
8. Baysian regression
9. Decision Forest Regression
10. Neural Networks Regression
11. While applying and training each model, we improved the performance matrix by adding and removing a few features that we initially thought were meaningful, but instead had negative impact on our performance.
12. Next, we used the performance metrics to compare each model, and found that Neural Network was our best model.
13. We then deployed Neural Network as a webservice on Azure and integrated the API endpoint with our application.

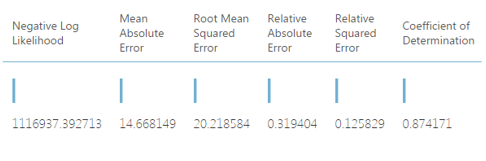
Model Analysis

i) Linear Regression

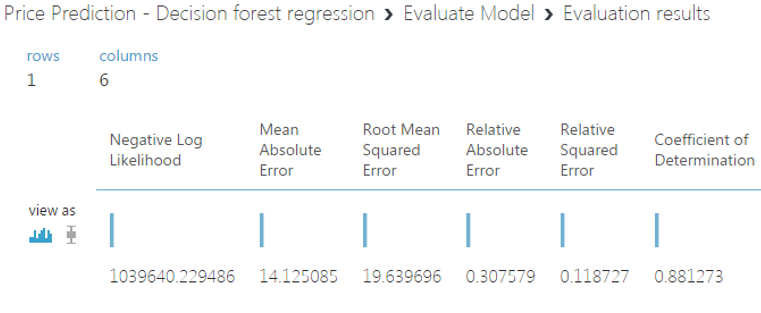


ii) Boosted Decision Tree****

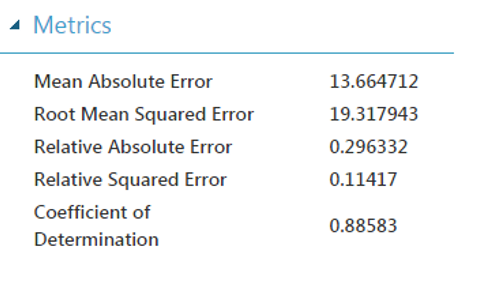
iii) Bayesian Linear Regression

****

iv) Decision Forest Regression

****

v) Neural Networks

****

Model Comparision:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | **Linear Regression** | **Neural Networks** | **Bayesian Linear Regression** | **Decision Forest Regression** | **Boosted Decision Tree** |
| **MAE** | 13.650 | 13.6647 | 14.668 | 14.125 | 13.2177 |
| **RMSE** | 19.390 | 19.317 | 20.218 | 19.369 | 18.744 |
| **Relative Absolute Error** | 0.2960 | 0.2963 | 0.3194 | 0.3075 | 0.2873 |
| **Relative Squared Error** | 0.1150 | 0.1141 | 0.1258 | 0.1187 | 0.1078 |
| **Coefficient** | 0.8849 | 0.8858 | 0.8741 | 0.8812 | 0.8921 |

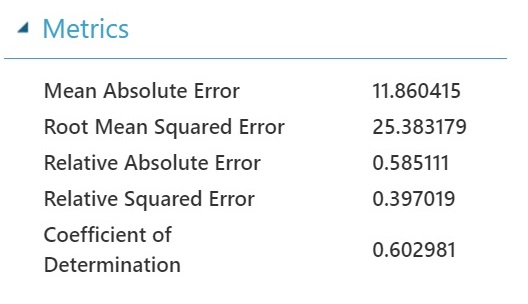
**Review:**

1. After comparing the models, we saw that Boosted decision tree has the best performance matrix.
2. For each score models, we saw the predicted results for price, and boosted decision tree is the closest of all the predictions.
3. We compared all the models and found that Boosted decision tree predicts best on all the models.

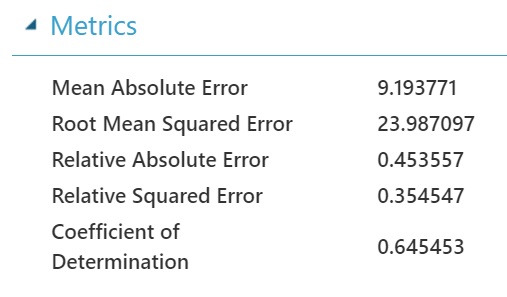
**III. ARRIVAL DELAY PREDICTION MODEL**

1. Technique used and approach
2. We upload our cleansed dataset on azure.
3. We then used filter based feature selection to select the features based on the weights of features on the dependent variable (ArrDelay).
4. On evaluation, the data we found that the main predictors were- Year, Month, Day of month, Day of week, Taxi in, Hour, Flight number, Temperature and Visibility.
5. Next, we split our data into test and train based on 80:20 split.
6. We used the following regression models for price prediction-
7. Linear regression
8. Boosted decision tree regression
9. Decision Forest Regression
10. Neural Networks Regression
11. While applying and training each model, we improved the performance matrix by adding and removing a few features that we initially thought were meaningful, but instead had negative impact on our performance.
12. All the regression models that we generated performed average. We realized that the features in our dataset are not very accurate predictors of delay in flight. We concluded that many other features like mechanical failure probability, airport congestion, etc. are more responsible for a more accurate predictive model, which we were not able to find on internet.
13. We used the performance metrics to compare each model, and found that Linear regression was our best model.
14. We then deployed Boosted decision tree as a webservice on Azure and integrated the API endpoint with our application.

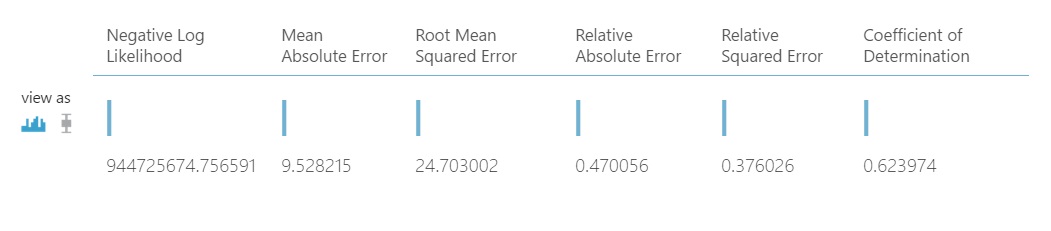
i) Linear Regression



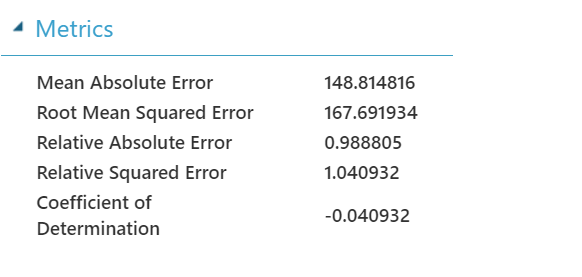
ii) Boosted Decision Tree



iii) Decision forest regression

****

iv) Neural Networks

****

Model Comparision:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Linear Regression** | **Neural Networks** | **Decision Forest Regression** | **Boosted Decision Tree** |
| **MAE** | 11.86 | 148.81 | 9.52 | 9.19 |
| **RMSE** | 25.38 | 160.93 | 24.70 | 23.98 |
| **Relative Absolute Error** | 0.58 | 0.98 | 0.47 | 0.45 |
| **Relative Squared Error** | 0.39 | 1.04 | 0.37 | 0.35 |
| **Coefficient** | 0.60 | -0.04 | 0.62 | 0.64 |

**Review :**

1. After comparing the models, we saw that Boosted decision tree has the best performance matrix.
2. For each score models, we saw the predicted results for price, and boosted decision tree is the closest of all the predictions.

**V. FLIGHT CANCELLATION CLASSIFICATION:**

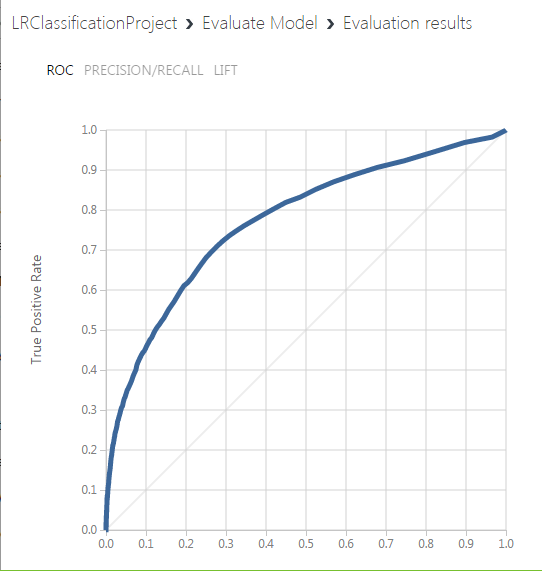
**Techniques used and approach-**

1. We import dataset in azure .
2. By using the various algorithm modules in azure, we ran our model to predict the cancellation of flights for future data.
3. We used the following algorithms:
4. Two class boosted decision tree
5. Two class logistic regression
6. Two class decision forest
7. Two class decision jungle

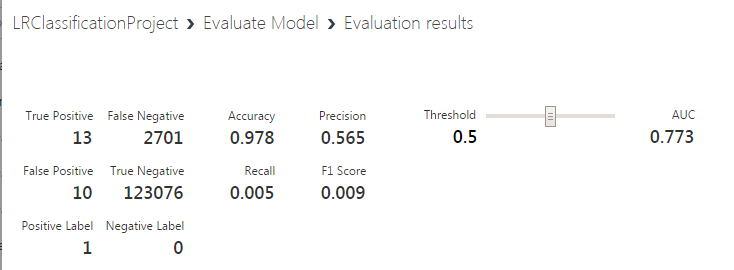
**Graphs and interpretations**

Two Class Logistic Regression:

ROC Curve

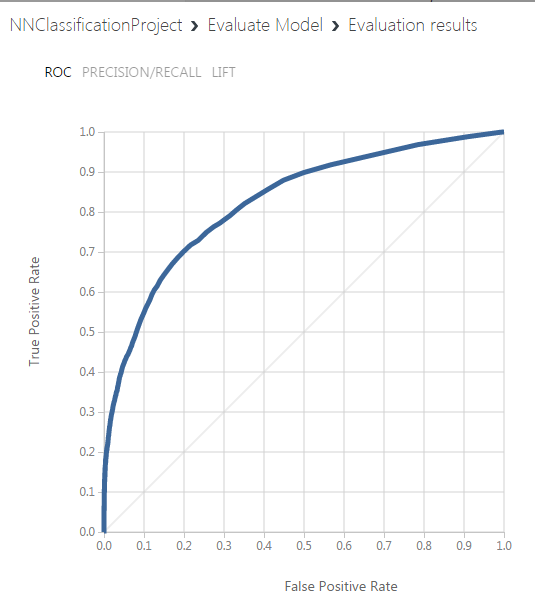


Confusion Matrix

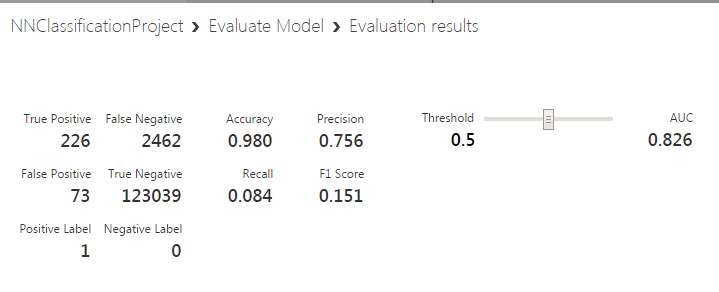


**Two Class Neural Network**

**ROC Curve**

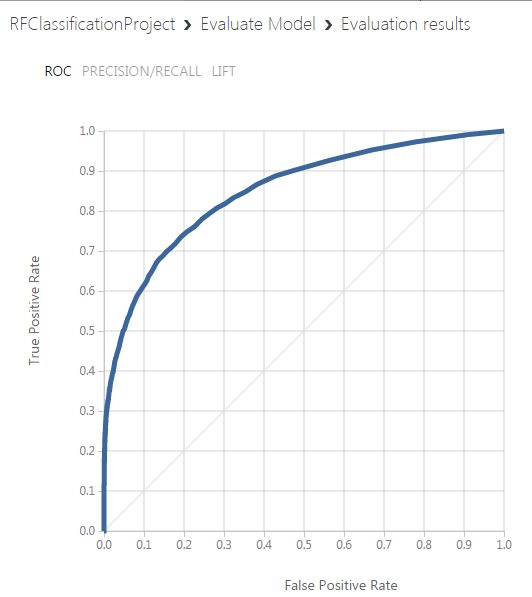


Confusion Matrix:

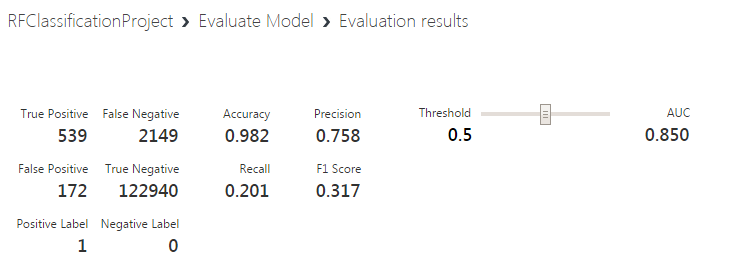


**Two Class Boosted Decision Tree**

**ROC Curve**

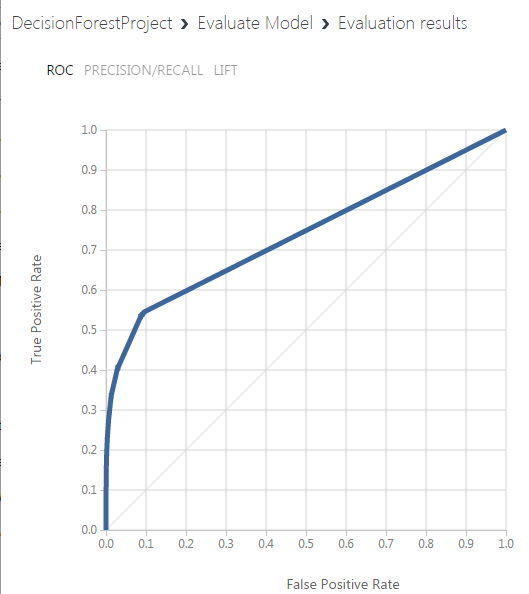


**Confusion Matrix**

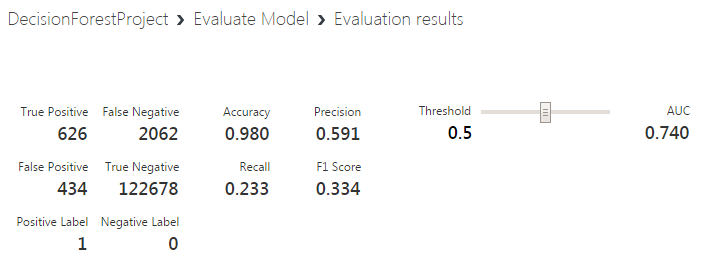


Two Class Decision Forest:

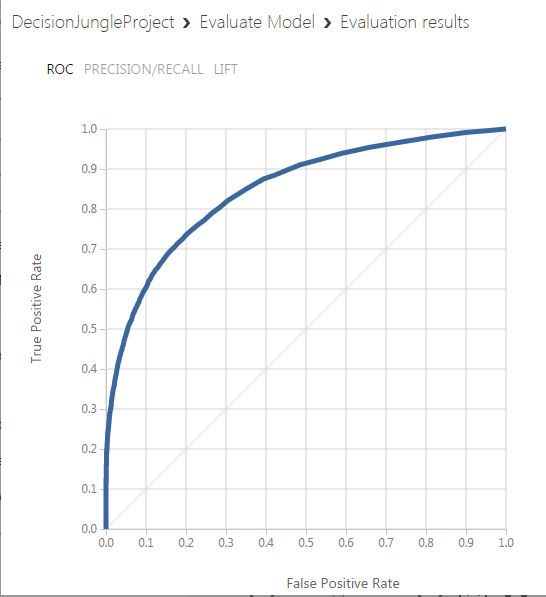
**ROC Curve:**

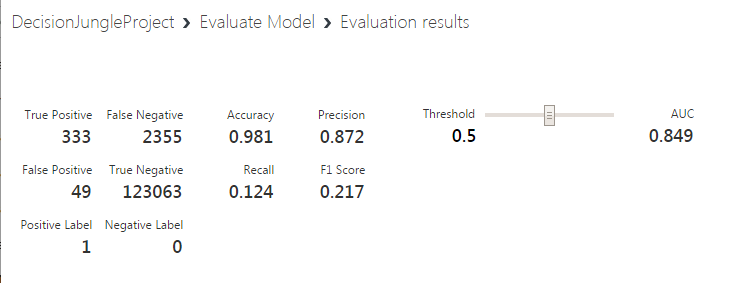


**Confusion Matrix:**



Two Class Decision Jungle





**Model comparison:**

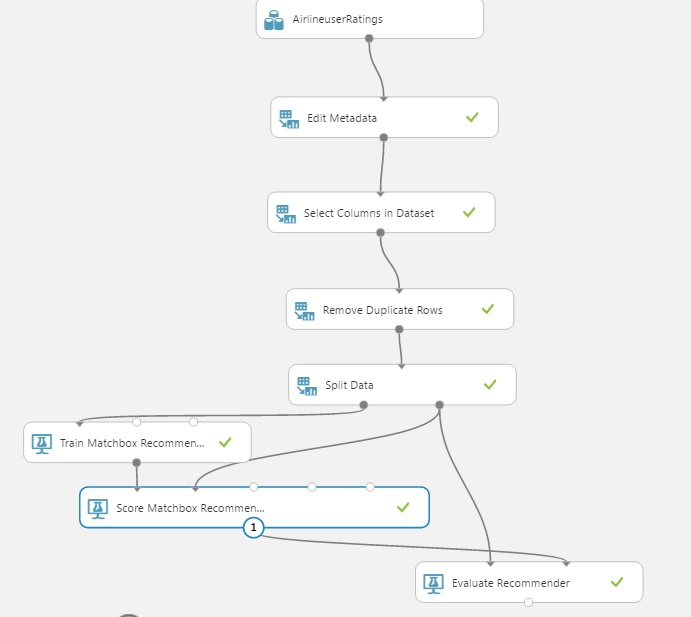
|  |  |  |
| --- | --- | --- |
| Model | Accuracy | Precision |
| Two Class Logistic Regression | 0.978 | 0.565 |
| Two Class Neural Network | 0.980 | 0.756 |
| **Two Class Boosted DecisionTree** | **0.982** | **0.758** |
| Two Class Decision Forest | 0.980 | 0.591 |
| Two Class Decision Jungle | 0.981 | 0.872 |

From the analysis of different classification Modules it can be found that Two Class Boosted Decision Tree has the predictions with the accuracy of 98.2%.

1. **RECOMMENDER SYSTEM**

We have used another dataset which contains the reviews of various different airlines by different users.

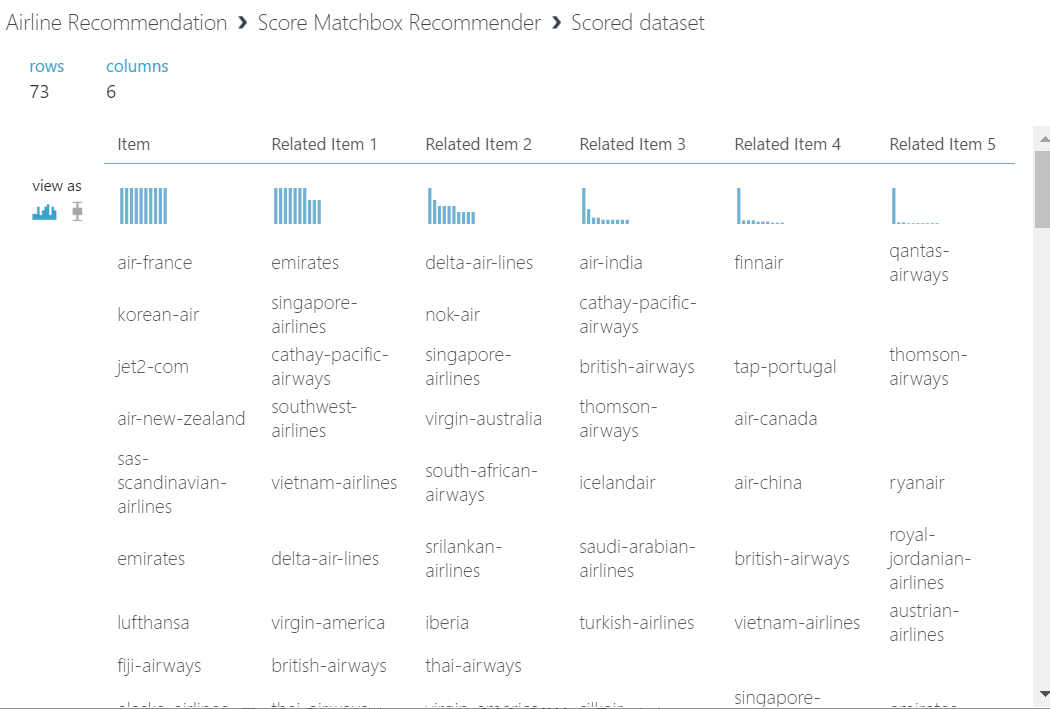
We have used matchbox recommender module in azure to deploy our model.



Steps:

1. We upload the dataset.
2. We change the column names by using edit metadata module
3. Next, we select only relevant columns from our dataset.
4. We noticed that our dataset had a few duplicate rows. We used ‘Remove duplicate rows module’ to remove all the duplicate rows from the data set.
5. Next, we split the data in train and test.
6. We apply matchbox recommendation module and finally evaluate our recommendations by using evaluate recommender module

Output of matchbox recommender:



We then created a web service of this recommender model, and integrate with our application.

The recommender basically uses collaborative filtering to recommend airlines based on user ratings. When a new user request the rating, new set of airlines will be displayed as there will be no way to associate new users preferences with previous users.

**Web Application ::**

We used ASP.net embedded with JQuery and Bootstrap to develop our front end.

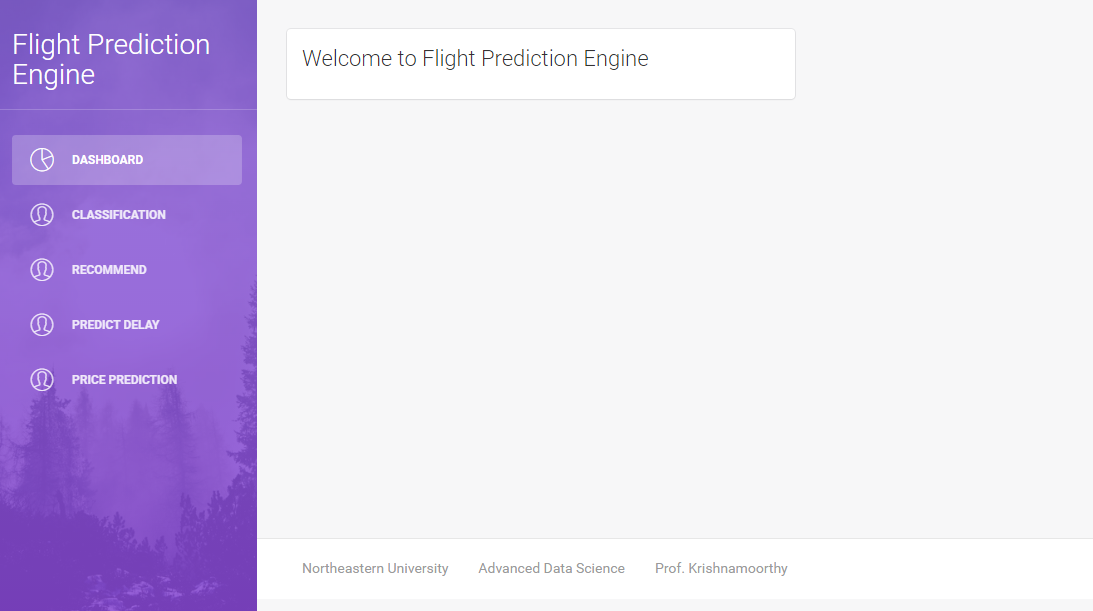
Technologies used:

* C#
* Python for scraping
* ASP.Net
* HTML, CSS, Bootstrap, Javascript, JQuery, Ajax
* Azure Machine Learning

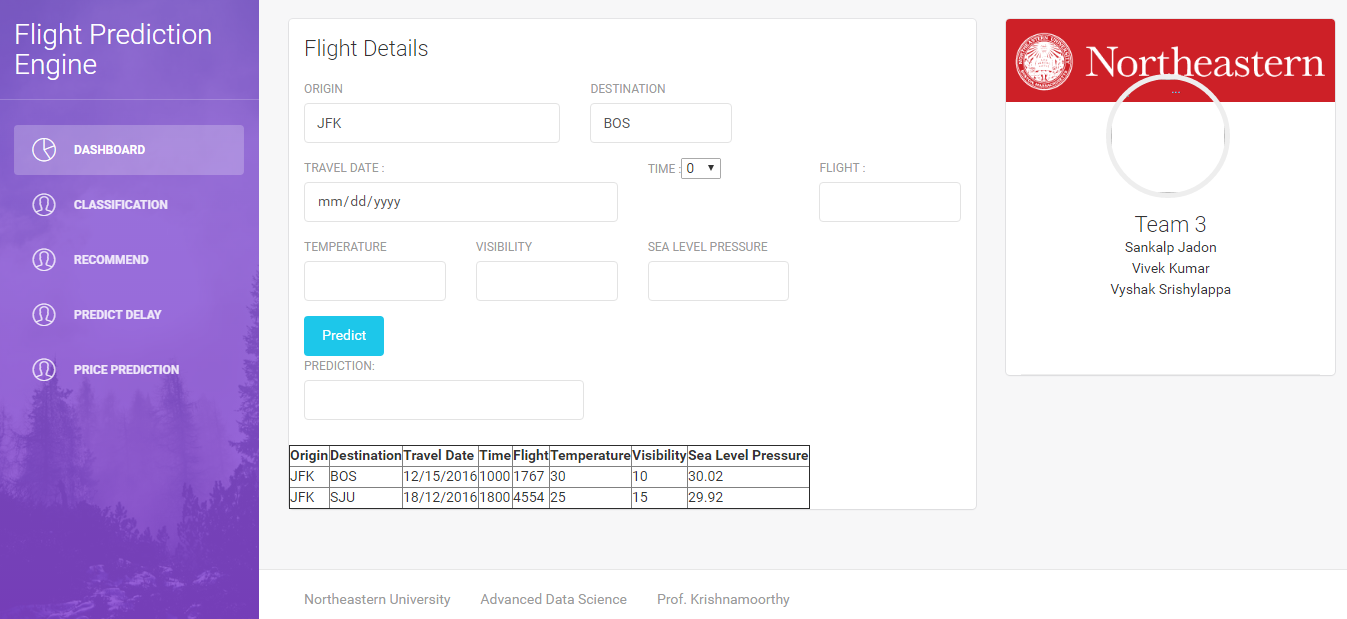
We have three pages to show Flight Price prediction, Flight delay prediction and Flight cancellation. To predict cancellation, we are merging the weather data with the received flight data.

**Screenshots**

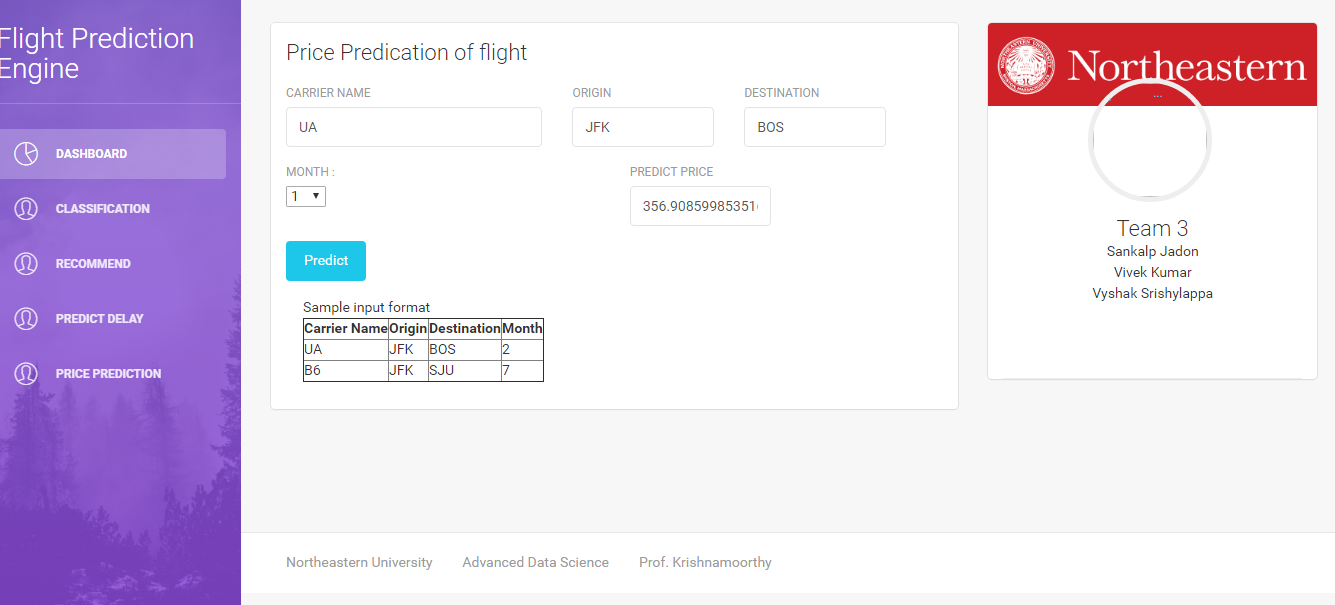
Home Page – Dashboard



Classification



**Flight Price Prediction**



References ::

<http://www.flightstats.com/company/products/flight-data-services/>

<https://docs.microsoft.com/en-us/azure/machinelearning/machine-learning-create-experiment>