Flight Delay Prediction

Sideshwar J B

April 23, 2020

Abstract

Flight delay makes flight scheduling the most challenging issue in the aircraft industry. Several factors are attributed to flight delays such as weather conditions and technical errors. This project presents a two stage predictive machine learning model that uses forecasted weather data to predict whether the flight is delayed or not and the amount of delay, thereby hoping to improve flight scheduling.

1 Introduction

Over the past few years air travel has been increasingly preferred by people due to its speed and comfort, leading to a phenomenal increase in air traffic. This has resulted in massive amount of flight delays. The major factors causing these delays are weather conditions, maintenance errors, air traffic control restrictions, security and airport conditions. These delays are responsible for large economic and environmental losses. According to the U.S. Department Of Transportation (DOT), the economic impact of flight delays is estimated to be more than $41 Billion per year to the national economy. Given this figure, there is a primary concern to predict, mitigate delays and optimize flight planning.

As flight delays caused by the above mentioned factors can affect departure or arrival time of the flight, the project aims to predict and calculate the arrival delay using the flight data and the weather data of the corresponding origin and destination airports.

The project uses two ML algorithms namely Classification and Regression to predict the chance of flight delay and delay time respectively.

2 Dataset

The flight data-set comprises of details of all the flights for the years 2016 and 2017. The corresponding weather details for the years is also collected. Each data-set is filtered to contain features with maximum relevance to the given problem. The following are the airport codes of those airports under consideration.

ATL	IAH	MIA
CLT	JFK	ORD
DEN	LAS	PHX
DFW	LAX	SEA
EWR	MCO	SFO

Table 1: Airport Codes

The features considered in the flight data are tabulated as follows.

FlightDate	Quarter	Year
Month	DayofMonth	DepTime
DepDel15	CRSDepTime	DepDelayMinutes
OriginAirportID	DestAirportID	ArrTime
CRSArrTime	ArrDel15	ArrDelayMinutes

Table 2: Flight data - Features

The features considered in the weather data are tabulated as follows.

WindSpeedKmph	WindDirDegree	WeatherCode
precipMM	Visiblity	Pressure
Cloudcover	DewPointF	WindGustKmph
tempF	WindChillF	Humidity
date	time	airport

Table 3: Weather data - Features

The flights that lack data in any of the considered features are removed from the data set. The given flight and weather data is compared using date, time, and airport of the weather data and FlightDate, ArrTime and destination airport of flight data to merge the two data-sets. The final data-set contains a total of 27 features.

3 Classification

The classification model aims to predict whether the flight is delayed or not. The merged data-set that includes the flight data and weather data of corresponding departure and arrival airports is made use of in the classification task. The criteria to classify a flight to be delayed is set above a threshold of 15 minutes. *ArrDel15* denotes whether the flight is delayed or not. The class 1 values indicates that the flight is delayed and class 0 values indicates it is not.

The data-set is split into train and test sets in the ratio of 70:30 to train and evaluate the model respectively. The training data is used to train various classification models like Logistic Regression, Decision Trees, Random Forests, Extra Trees and XGBoost.

3.1 Metrics

The following metrics are used to evaluate the performance of the models.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$F - measure = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{3}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$ConfusionMatrix = \begin{bmatrix} TP & TN \\ FP & FN \end{bmatrix} \tag{5}$$

where*TP-True Positives*, *FP-False Positives*, *TN-True Negatives*, *FN-False Negatives*.

3.2 Results

The following are the scores of different classifiers.

Algorithm	Precision		Recall		F1-Score		Accuracy
	0	1	0	1	0	1	
Logistic Regression	0.92	0.89	0.98	0.68	0.95	0.77	0.92
Decision Trees	0.92	0.70	0.92	0.71	0.92	0.70	0.87
Random Forests	0.93	0.89	0.98	0.70	0.95	0.78	0.92
Extra Trees	0.93	0.82	0.96	0.74	0.95	0.78	0.91
XGBoost	0.92	0.90	0.98	0.68	0.95	0.78	0.92

Table 4: Classifier Scores

4 Data Imbalance Problem

The results shows that there is a difference between the class 1 and the class 0 scores. This is because of the non-uniform class distribution that prevails in the data-set (i.e) the number of non-delayed flights is much greater than the delayed flights.

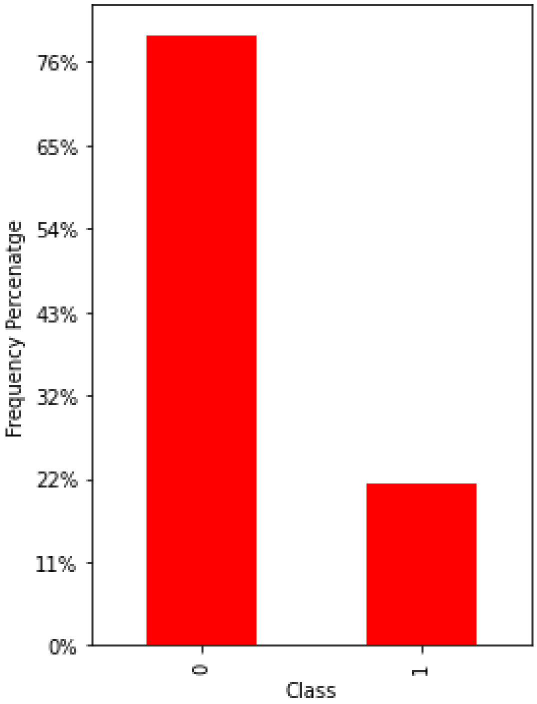

Figure 1: Class Distribution

The uneven class distribution makes classification algorithms to predict the majority class data (non-delayed flights). It treats the features of minority class data (delayed flights) as noise and ignores it. Thus, there is a high chance of misclassification of the minority class as compared to majority class. This imbalance causes the classifier to not perform properly.

4.1 Solution

The problem is addressed by the use of sampling methods. Sampling balances the class distribution by adding instances of class 1 or deleting class 0 instances known as oversampling and undersampling respectively. Oversampling compensates for the imbalanced class distribution by duplicating samples from minority class in the training set, whereas undersampling tends to delete the samples of majority class from the training set. Due to the risk of loss of important data, undersampling is often not preferred.

The following are the results of the classifiers after various sampling techniques applied.

Algorithm	Precision		Recall		F1-Score		Accuracy
	0	1	0	1	0	1	
Logistic Regression	0.92	0.89	0.98	0.68	0.95	0.77	0.92
Decision Trees	0.92	0.68	0.91	0.71	0.92	0.69	0.87
Random Forests	0.92	0.89	0.98	0.70	0.95	0.78	0.92
Extra Trees	0.93	0.82	0.96	0.74	0.95	0.78	0.91
XGBoost	0.93	0.88	0.97	0.71	0.95	0.78	0.92

Table 5: Classifier Scores - SMOTE Oversampling

Algorithm	Precision		Recall		F1-Score		Accuracy
	0	1	0	1	0	1	
Logistic Regression	0.94	0.74	0.93	0.78	0.93	0.76	0.90
Decision Trees	0.94	0.51	0.80	0.81	0.86	0.63	0.80
Random Forests	0.95	0.73	0.92	0.81	0.93	0.77	0.90
Extra Trees	0.95	0.72	0.92	0.81	0.93	0.76	0.89
XGBoost	0.94	0.73	0.92	0.79	0.93	0.76	0.90

Table 6: Classifier Scores - Random Undersampling

As F1-score is a function of both precision and recall, it is a suitable metric to measure the performance of classifiers, along with accuracy.

From Table 5 and Table 6 we observe that SMOTE oversampling gives the best F1-Scores and accuracy values. Hence, it outperforms other sampling methods in achieving better class 1 scores.

As the F1-Scores and accuracy are nearly the same for both sampled and un-sampled data, sampling does not seem to improve the performance of the classifier for the given data-set. Hence proves to be an ineffective solution to solve the class imbalance problem.

Hence, the XGBoost classifier trained using un-sampled data is found to have the best overall scores and predicts the chance of flight delay to a better extent than the other classifiers.

5 Regression

The Regression model aims to predict the amount of delay of the flights in case of delay. The *ArrDel15* feature is used to filter the delayed flights which makes the train and test set for the model. The train and test set is split in the ratio 70:30. Various regression models such as Linear Regression, Random forest Regression, Extra Trees, Support Vector Regression (SVR) and XGBoost Regression are trained and evaluated to find the best model based on their performance metrics.

5.1 Metrics

$$\text{Root mean squared error } \mathbf{RMSE} = \sqrt{\frac{1}{n}\sum_{t=1}^{n} e_i^2}$$

$$\text{Mean absolute error } \mathbf{MAE} = \frac{1}{n}\sum_{t=1}^{n} |e_i|$$

$$\text{R Squared } \mathbf{R2} = 1 - \frac{SSE_{residual}}{SSE_{total}}$$

where e_i is the difference between the actual and predicted values (Error) and SSE is the sum of squared errors.

5.2 Results

The results of the various regression algorithms are tabulated as follows.

Algorithm	MAE	RMSE	R2
Linear Regression	12.13	17.45	0.93
Random Forest	11.69	16.64	0.94
SGD Regressor(SVR)	12.20	17.71	0.94
Extra Trees	11.70	16.68	0.94
XGBoost	11.62	16.63	0.94

Table 7: Regression Models - Metrics

The results shows that XGBoost regression model have the least MAE and RMSE scores and the highest R2-Score. This implies that the difference in actual and predicted delay values is relatively lesser compared to other models. Thus the conclusion is that XGBoost regressor predicts the flight delay more accurately than the other regression models.

6 Regression Analysis

The regression model chosen is tested in different intervals of arrival delay minutes to analyze the performance of the model in each of the intervals. The scores obtained in the different intervals of delay minutes are tabulated as follows.

Arrival Delay Minutes - Range	MAE	RMSE
0-100	10.97	14.63
100-200	17.74	26.77
200-500	19.43	31.48
500-1000	22.40	30.58
1000-2000	33.35	38.96

Table 8: Regression Analysis - Scores

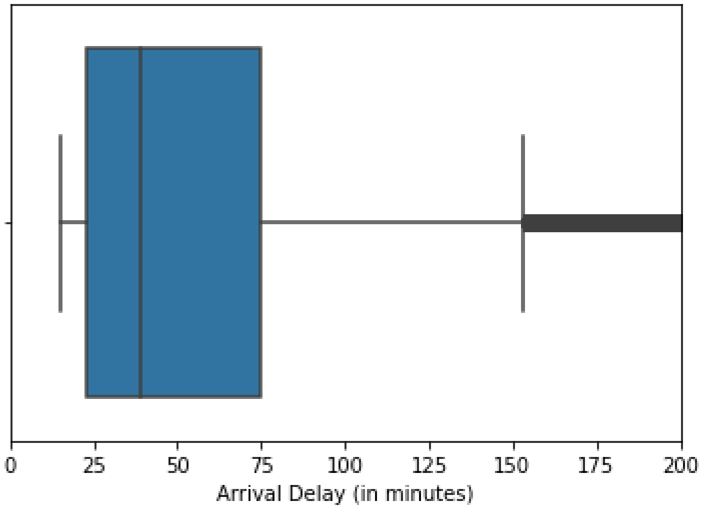

Figure 2: Box Plot Distribution of Arrival Delay Minutes

From the box plot distribution, the Inter-Quartile range appears to lie between 23 and 75 minutes. This is a clear indication that the dataset is largely populated with values of arrival delay in the range of 23 and 75 minutes. Although, Table 8 shows the least MAE and RMSE scores for this range of delay values, the model also seems to predict the outliers in the range of 500 to 2000 minutes with relatively low percent errors. Thus, the conclusion is that the model performs well over the entire domain of arrival delay as it predicts the delays precisely with acceptable values of percentage errors.

7 Pipelining

The original data-set is split into train and test set which is used to classify the flights to be delayed or not. For this purpose, the XGBoost classification algorithm seems to be the most suitable one as it provides better scores than the other classifiers. For the pipe-line output, the delayed flights as predicted by the classifier is given to the regression model that predicts the arrival delay minutes. The most appropriate model for this purpose is XGBoost regressor because of its least MAE and RMSE scores among the other regression models.

The following figure depicts the flow of the pipe-line model.

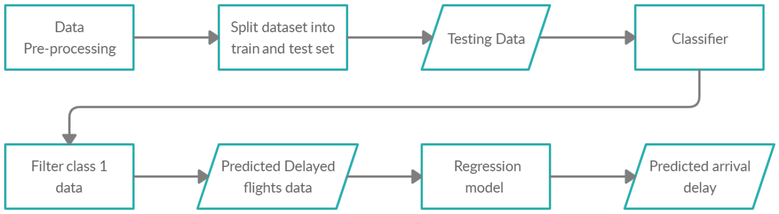

Figure 3: Flow Chart - Pipe-line model

The following are the scores of the pipe-line regression model.

Metrics	Values
MAE	13.41
MSE	335.62
RMSE	18.32
R2	0.94

Table 9: Regression model scores - Pipe-line output

Table 9 shows that the model predicts the delay minutes accurately with relatively low errors.

8 Conclusion

The first part of the project is to retrieve and process the flight and weather data. The following part uses the processed data to classify the flights as delayed or not.

Due to the data imbalance problem, different sampling methods are employed to make up for the improper class distribution. But sampling did not result in any improvement in the class 1 scores and is ineffective to predict the delay for the given dataset. Among the various classifiers trained, XGBoost classifier using un-sampled data achieved the best overall class-0 and class-1 scores and accuracy. Hence, it classifies the delayed and non-delayed flights accurately.

Among the various regression models, XGBoost regressor achieved the least MAE(11.62) and RMSE(16.63) scores. As the flight delay values lie prominently between the range of 20 to 100 minutes, MAE of 11.62 is acceptable as the percent error is only between 10% and 50%. Hence, the model is effective in predicting the flight delay.

The delayed flights as predicted by the classifier is pipe-lined to the regression model. Thus the two stage machine learning model is implemented successfully and works efficiently for the given problem statement.