FLIGHT DELAY PREDICTION

By,

Vykunth. P,

S.V.C.E 3rd year

May 2020

Abstract

Flight Delays play a major role in the aviation field. The reason for such flight delays are mainly due to weather factors. These delays impact both the airline entities as well as the passengers which invariably lead to unpleasant and confusing situations. To avoid these situations, this project was constructed to forecast the flight delays in the arrival by using a two-stage predictive model.

1 Introduction

Flight Delays are a major operational problem for Airline corporations. They lead to critical economic repercussions such as losing customers who shift to a different airline, operational costs fines and penalties. Delay is defined as the period in which a flight is late or postponed. Thus, a delay may be represented by the difference between scheduled and real times of departure or arrival of a plane. Some examples of delays include mechanical problems, ground delays, busy air traffic, runway queues and security reasons. In this Project we take the arrival delay due to weather conditions of the flight. We use a two-stage predictive model in this project. The first stage is classification where we classify the occurrence of flight delays. The second stage involves regression to predict the value of delay in minutes of the classified flights. The data comprises of weather data at 15 different airports and the flight data of all the flights that flew inside the USA between 2016 and 2017.

2 Data Pre-Processing

Data Preprocessing and cleaning is required to convert the data into a proper dataset which can be used in our model to give required results.

2.1 Flight Data

We Preprocess the data only for the airports mentioned below in table 1 for the years 2016 and 2017 only inside the United States of America. From the flight csv, we select flights only from the required airports (Table 1). Also, the features considered for the flight are mentioned in Table 2.

Table 1: Recommended Airports

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

Table 2: Recommended flight features

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

2.2 Weather Data

The weather data is of JSON format for the years 2013-2017 for various airports. We load the json file and filter it only for the recommended weather features (Table 3) in the recommended airports (Table 1) for the years 2016 and 2017.

Table 3: Recommended Weather features

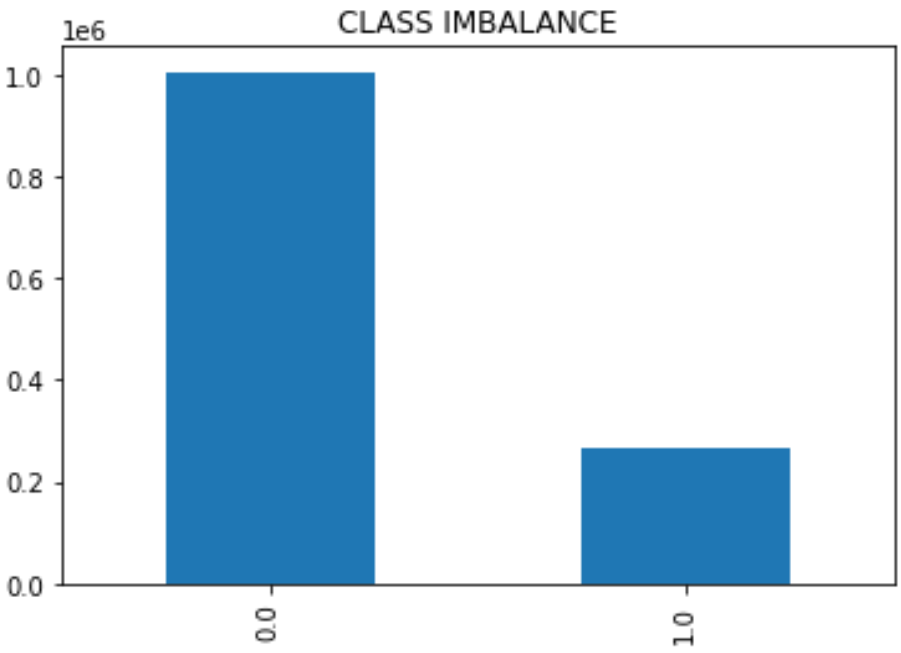
WindSpeedKmph	WindDirDegree	WeatherCode	precipMM
Visibilty	Pressure	Cloudcover	DewPointF
WindGustKmph	tempF	WindChillF	Humidity
date	time	airport	

2.3 Merging Flight and Weather Data

We merge the weather data and flight data into a single DataFrame. The features used to merge are Date, Time and Airport. The merged DataFrame is now saved as a csv file for further use. The csv is then cleaned to drop unwanted columns.

3 Classification

In this section, we classify whether the flight has been delayed or not. In the x-axis, we take the feature columns (table 2) and y-axis as ArrDel15. The data is split into training data and test data in the 70:30 ratio. The training set contains a known output and the model learns on this data whereas we have the testing set to test our model's prediction on this subset. There occurs a problem where there is a huge difference in flights which have been delayed and flights that have not been delayed. This is known as class imbalance (refer below picture).



Class 0 depicts the flights that have not been delayed and class 1 depicts the number of flights that have been delayed. There is an imbalance in these two classes. To overcome this, Under-sampling or Over-sampling is done.

3.2 CLASSIFICATION METRICS

Confusion-Matrix

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

Accuracy
Accuracy is the proportion of true results among the total number of cases examined.

```
Accuracy = (TP+TN)/(TP+FP+FN+TN)
```

Precision
Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives.

```
Precision = (TP)/(TP+FP)
```

Recall
Recall is defined as the number of true positives divided by the number of true positives plus the number of false negatives.

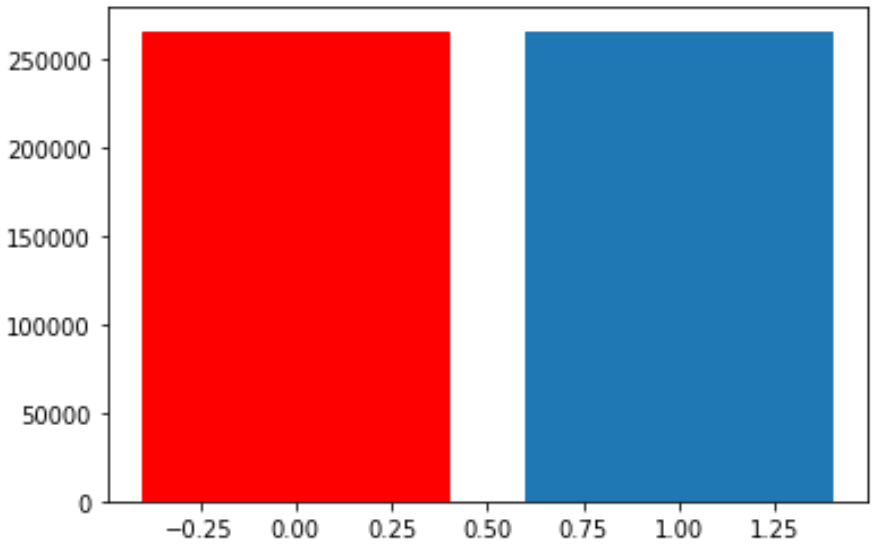
```
Recall = (TP)/(TP+FN)
```

F1 score
The F1 score is a number between 0 and 1 and is the harmonic mean of precision and recall.

```
F1 score = 2*[(Precision*recall)/(Precision+recall)]
```

3.2 UNDER-SAMPLING

Under-sampling techniques remove examples from the training dataset that belong to the majority class in order to better balance the class distribution. With under-sampling you drop a lot of information to balance the sample. For under-sampling we use Random under-sampling method and the data was fit into several classification algorithms.



EXTRA-TREES CLASSIFIER

	precision	recall	f1-score	support
Class 0	1.00	0.97	0.98	430732
Class 1	0.89	0.99	0.94	113890
accuracy			0.97	544622

DECISION-TREE CLASSIFIER

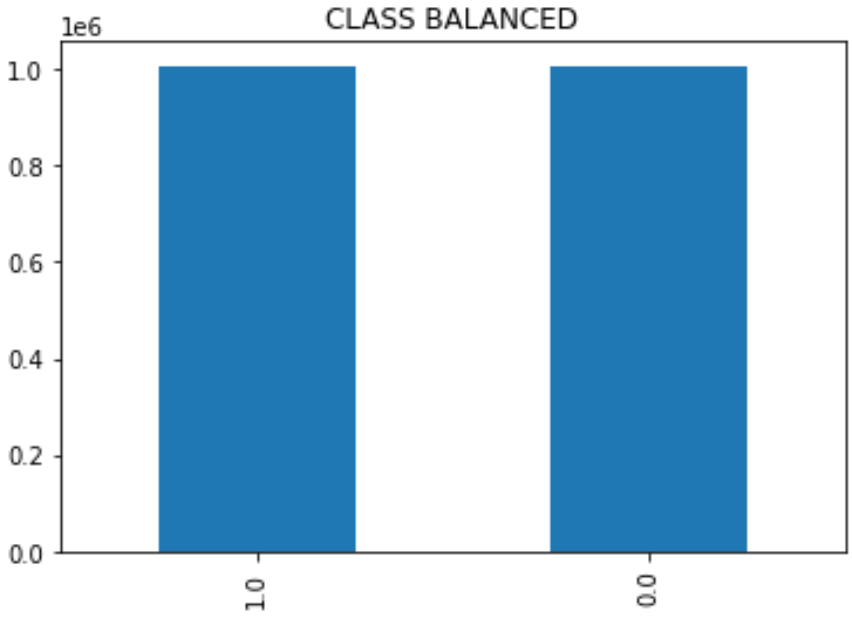
	precision	recall	f1-score	support
Class 0	0.79	0.79	0.79	430732
Class 1	0.21	0.21	0.21	113890
accuracy			0.67	544622

XGBOOST CLASSIFIER

	precision	recall	f1-score	support
Class 0	0.94	0.95	0.94	430732
Class 1	0.79	0.75	0.77	113890
accuracy			0.91	544622

3.3 OVER-SAMPLING

Oversampling increases the weight of the minority class by replicating the minority class examples. For oversampling, we use Random oversampling or Synthetic Minority Oversampling Technique (SMOTE) and use the better performing algorithm.



3.3.1 Random Over-sampling Vs Synthetic Minority Oversampling Technique

	Algorithm	Precision		Recall		F1-score		Accuracy
		0	1	0	1	0	1	
Random oversampling	Logistic Regression	0.94	0.74	0.93	0.78	0.93	0.76	0.90
	Decision Tree	0.92	0.71	0.92	0.70	0.92	0.70	0.87
	Extra Trees	0.93	0.80	0.95	0.74	0.95	0.74	0.91
	Xg-boost	0.94	0.73	0.92	0.79	0.93	0.76	0.90
S.M.O.T.E.	Logistic Regression	0.94	0.74	0.93	0.78	0.93	0.76	0.90
	Decision tree	0.92	0.68	0.91	0.70	0.92	0.69	0.87
	Extra Trees	0.94	0.77	0.94	0.76	0.94	0.76	0.90
	Xg-boost	0.94	0.79	0.95	0.75	0.94	0.77	0.90

Extra Trees under random over-sampling was the best performing model as it has the most accuracy and better F1 score.

4 Regression

Regression predicts continuous values. In this second stage, we use regression algorithms to predict the amount of time in which the flight has been delayed. We take X as Feature columns (table 2) and the target variable y as ArrayDelayMinutes. We use various algorithms such as Linear regressor, Extra Trees Regressor and XGBoost Regressor to predict the values.

4.1 Regression Metrics

RMSE

Root mean square error is the square root of the average of squared differences between prediction and actual observation.

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

MAE

MAE measures the average magnitude of the errors in a set of predictions, without considering their direction.

$$MAE = \frac{1}{N}\sum_{i=1}^{N} |y_i - \hat{y}|$$

R-Squared Error

R2 represents the coefficient of how well the values fit compared to the original values.

$$R^2 = 1 - \frac{\sum(y_i - \hat{y})^2}{\sum(y_i - \bar{y})^2}$$

4.2 Regression Scores

Algorithm	MAE	RMSE	R-SQUARED
Linear Regressor	10.7654	32.9720	0.3643
Extra Trees Regressor	10.8291	32.9045	0.3665
XGBoost Regressor	10.8290	32.9046	0.3677

4.3 PIPELINE

In this module, the result from the best performing classifier (Extra-trees from oversampling) was taken and the amount of delay was predicted by using a pre-trained regressor for the resultant delayed flights of the classifier.

4.3.1 PIPELINE SCORES

Algorithm	MAE	RMSE	R-SQUARED
Linear Regressor	13.9127	18.6743	0.9391
Extra Trees Regressor	14.6292	19.3198	0.9348
XGBoost Regressor	14.3726	19.0106	0.9370

5 CONCLUSION

Thus, a two-stage predictive machine learning model was constructed, where-in the first stage dealt with classification of flight whether delayed or not. The data had to be over-sampled to balance the two data sets. Extra-Trees classifier was the best performing classifier after over-sampling. The second stage dealt with regression. Pipelining was done to improve the scores and the model, so the results from classification were taken and pipelined. Linear regressor had the best scores compared to extra trees and Xg-boost. The predicted arrival delay had a Mean Absolute Error of 13.9 minutes with the actual arrival delay. The RMSE was 18.67 minutes with the actual arrival delay.