Flight Delay Prediction

15.04.2019

Naveen Narayanan
2nd Year CSE
SSN College Of Engineering
Chennai

Introduction

It is extremely essential for flights to arrive and depart on the time scheduled. Otherwise a lot of difficulties will be created for both the customers as well as the airline company. But then, it is not always possible to do so. There might be days where the delay is unavoidable. But what is avoidable is the difficulties which arise because of these delays. If we are able to know if there is going to be a delay, then we can change our plans accordingly and sort out the issue.

So the aim of the machine learning engine is to predict the arrival delay of a particular flight after the flight has taken off. The data considered for solving this problem is weather data of the airport from which the flight is departing and the airport at which the flight is landing, and also the flight data.

Specifications

The datasets used for this problem are:

- Weather data of 15 airports from 2014 to 2017 (Refer TABLE: 1)
- Flight data of all flights travelling in USA during 2016 and 2017 (Refer TABLE: 3)

DATA PREPROCESSING

The first step is to preprocess the given data and take only the necessary features. We will first be working on the weather data. We will be taking the following weather features into consideration:

TABLE: 1

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

The weather data is in JSON format. So we will take the necessary features during the time period of 2016 and 2017. After that we merge the weather data in one single dataframe and store it as a CSV file.

The second step is to merge all the flight data. We will be working on 15 airports, hence we will choose only those flights which have travelled from or to the below mentioned list of airports:

TABLE: 2

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

The features that we will be considering for a flight are:

TABLE: 3

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

Next we will merge the weather data and flight data into a single dataframe. Each flight data is merged with its corresponding departure time weather and arrival time weather data.
Flight and weather data is merged based on these 3 features :

- Date
- Time
- Airport

After we finish merging and make it into a single CSV file, we will classify the data.

CLASSIFICATION

We will classify whether a flight is going to arrive on time or not. Some of the features used to fit the classifier are ORIGIN AIRPORT, DESTINATION AIRPORT, DEPARTURE DELAY, and then weather features of both destination and origin airports. There is heavy class imbalance, i.e: there is more number of flights which are not delayed whereas there is less number of flights which are delayed. Hence we have to manipulate the training data. Data can be manipulated either by oversampling or undersampling the data. Classification results were better for oversampled data than undersampled data. Under oversampling, SMOTE (Synthetic minority oversampling technique) was used to oversample as it gave a

better resultant dataset than the random oversampler. The Classifiers considered are: Extra Trees Classifier, XGBoost, Decision Tree, Gaussian Naive Bayes.

NO SAMPLING:

XGBOOST:

	PRECISION	RECALL	F1 SCORE
Class 0	0.8	0.99	0.88
Class 1	0.68	0.05	0.09

EXTRA TREES CLASSIFIER:

	PRECISION	RECALL	F1 SCORE
Class 0	0.83	0.92	0.87
Class 1	0.48	0.29	0.37

DECISION TREE:

	PRECISION	RECALL	F1 SCORE
Class 0	0.83	0.82	0.83
Class 1	0.36	0.39	0.38

GAUSSIAN NAIVE BAYES:

	PRECISION	RECALL	F1 SCORE
Class 0	0.82	0.88	0.85
Class 1	0.36	0.26	0.3

UNDER-SAMPLING:

XGBOOST:

	PRECISION	RECALL	F1 SCORE
Class 0	0.87	0.65	0.74
Class 1	0.32	0.63	0.43

EXTRA TREES CLASSIFIER:

	PRECISION	RECALL	F1 SCORE
Class 0	0.86	0.71	0.78
Class 1	0.35	0.58	0.43

DECISION TREE:

	PRECISION	RECALL	F1 SCORE
Class 0	0.85	0.59	0.7
Class 1	0.29	0.61	0.39

GAUSSIAN NAIVE BAYES:

	PRECISION	RECALL	F1 SCORE
Class 0	0.83	0.79	0.81
Class 1	0.32	0.38	0.35

OVER-SAMPLING:

XGBOOST:

	PRECISION	RECALL	F1 SCORE
Class 0	0.94	0.92	0.93
Class 1	0.72	0.79	0.75
Weighted Avg	0.89	0.89	0.89

EXTRA TREES CLASSIFIER:

	PRECISION	RECALL	F1 SCORE
Class 0	0.9	0.95	0.93
Class 1	0.78	0.62	0.69
Weighted Avg	0.88	0.88	0.88

DECISION TREE:

	PRECISION	RECALL	F1 SCORE
Class 0	0.92	0.91	0.92
Class 1	0.68	0.71	0.69
Weighted Avg	0.87	0.87	0.87

GAUSSIAN NAIVE BAYES:

	PRECISION	RECALL	F1 SCORE
Class 0	0.94	0.89	0.91
Class 1	0.66	0.77	0.71

Weighted Avg	0.88	0.87	0.87

Precision is the frequency with which a model was correct for a particular class. Recall can be defined as, for a particular class, how many did the model correctly identify?

CLASS 0: FLIGHT NOT DELAYED

CLASS 1: FLIGHT IS DELAYED

P(for class 1)= TN(TN + FN)

P(for class 0)= TP/(TP + FP)

R(for class 1)= TN/(TN + FP)

R(for class 0)= TP/(TP + FN)

Where P is precision, TP is true positive, FP is false positive, FN is false negative and R is recall. There are two cases in which our model can give a wrong result. One is FLIGHT NOT DELAYED, BUT PREDICTED TO BE DELAYED(FN). The second is FLIGHT IS DELAYED, BUT PREDICTED TO BE NOT DELAYED(FP). The second case is far more dangerous for the passenger. Hence we want the recall of the class: flights delayed (CLASS 1) to be high.

Along with that the F1 score is also taken into account. XGBoost classifier after oversampling was the best performing classifier because it had a good F1 score and recall for both the majority class as well as the minority class, unlike other classifiers which was biased towards the majority class.

REGRESSION

Now we will predict the amount of time the flight has been delayed. We will be taking only those flight records which have been predicted to be arrive late by the classifier. The same number of features which was used in the classifier have been used to train the regressor as well. The regressors considered are : Extra Trees Regressor, Linear Regressor, XGBoost Regressor. The RMSE and MAE are taken into account for finding the right regressor. RMSE is ROOT MEAN SQUARE ERROR while MAE is MEAN ABSOLUTE ERROR. Both RMSE and MAE gives the error in the predicted value when compared to the actual value. So lesser the error, the better is the model built.

	MAE	RMSE
XGBoost Regressor	11.6177	16.4693
Extra Trees Regressor	12.3742	17.2427

Linear Regressor	11.8314	16.7774

From the following table, we see that XGBoost Regressor has given a MAE of 11.61 and RMSE of 16.46, which is considered to be less because the actual arrival delay is pretty high when compared to these values, and it is better than the other regressors.

Results and Conclusions

This machine learning model was able to predict the arrival delay of a flight with a decent accuracy. XGBoost Classifier was able to classify the data into delayed or non delayed flights, and XGBoost Regressor was able to predict the arrival delay in minutes. The XGBoost classifier was able to give an F1 score of 0.93 for the majority class and 0.75 for the minority class after SMOTE oversampling. The predicted arrival delay had a Mean Absolute Error of close to 11 minutes with the actual arrival delay. The RMSE was close to 16 minutes with the actual arrival delay.