Two Stage Predictive Machine Learning Engine that Forecasts the On-Time Performance of Flights

Nitish Kumar M

Abstract. The uncertainty in weather prediction has been creating problems in the air transportation sector. This project focuses on training a two stage machine learning model that will predict if a flight will be delayed and by how much with respect to the predicted weather of the destination and origin of the aircraft before departure.

Keywords: Classifiers · Regressors · Sampling · Metrics

1 Introduction

Flights all over the world are cancelled or delayed by hours due to the sudden changes in weather. These hold ups in flights have led to the loss of productive man hours. Apart from mechanical issues, the major factor responsible for the delay is weather. Hence it is highly desirable to predict the delay of a flight in advance with respect to the predicted weather and can be helpful for the aircraft company to schedule the flights early or late depending upon the weather, also keep the passengers informed.

2 Dataset and Preprocessing

Data required to train and test the machine learning models consists of two different data sets namely the flight data and weather data. These two data sets were merged such that it contains the flight data mapped along with the weather data of the destination and origin of the flight accordingly.

Flight Data: Data comprises of the information of the flights over a period of two years (2016-2017) for every month and has several features (Origin, Destination, Departure time & Delay, Arrival time & Delay, etc) and this data was compiled accordingly. The Flight Dataframe was generated by compiling the data taking into consideration only some of the important features.

Weather Data: Data comprises of weather details over a period of 5 years and are grouped by cities. The files of the years 2016 and 2017 was accessed to retrieve weather features for 24 hours everyday for every city code. From all the available weather data, only certain weather features (humidity, visibility, wind speed, cloud cover,etc) were considered. The retrieved data from the files were processed and stored as the final weather data.

Merging Flight and Weather Data: At first, the flight and weather data was merged upon date, time and Origin city code and merged another time with Destination city code. The generated dataset consists of the flights data as well as the weather details of the origin city as well as the destination city. The final merged dataframe will be used for the running of machine learning models.

The final dataframe was divided into training and testing datasets such that training data consists of more than 70 % of the entire data.

3 Classification

In the development of a two stage machine learning model, the first stage involves a classification model that must be trained to classify flights on whether they would be delayed or not. The dataset generated previously consists of a column "ArrDel15" that provides a binary classification for delayed flights, which serves as the ground truth. This column was considered because a delay greater than 15 minutes is significant rather than a delay by few minutes.

In this case, the models used were XGBoost Classifier, Extra Trees Classifier, AdaBoost Classifier and Logistic Classifier. These models were trained and were used to predict the output of the test data.

The metrics were calculated for every model to determine the performance with respect to the dataset as the performance of every model varies with different types of datasets. The performance evaluation of the classifiers are done using the metrics Area Under the Curve (AUC), Accuracy, Precision, Recall and F1 Score.

A receiver operating characteristic curve, or ROC curve, is a graphical plot that illustrates the ability of a binary classifier system as its discrimination threshold is varied.

Accuracy can be defined as the percentage of correctly classified instances. Accuracy is calculated as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

TP - True Positives, FN - False Negatives, TN - True Negatives and FP -False Positives

Recall is calculated as the number of true positives divided by the total number of true positives and false negatives. Similarly , Precision is calculated as the number of true positives divided by the total number of true positives and false positives.Recall is also termed as true positive rates.

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

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

The F-score, also called the F1-score, is a measure of a model's accuracy on a dataset. The F-score is a way of combining the precision and recall of the model and is calculated as:

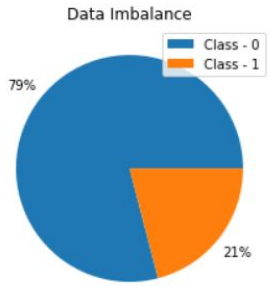
$$F1\ Score\ =\ 2\ *\ (\frac{Precision * Recall}{Precision + Recall}) \tag{4}$$

The models and their metrics are given in the table below.

Table 1. Classifier Model and Metrics

Class	Model	AUC	Accuracy	Precision	Recall	F1 Score
Class 0	XGBoost Classifier	0.578105	0.81	0.82	0.98	0.89
	Extra Trees Classifier	0.617575	0.79	0.83	0.92	0.87
	Logistic Classifier	0.506955	0.79	0.79	1.00	0.88
	AdaBoost Classifier	0.517382	0.79	0.80	0.99	0.88
Class 1	XGBoost Classifier	0.578105	0.81	0.69	0.18	0.28
	Extra Trees Classifier	0.617575	0.79	0.51	0.32	0.39
	Logistic Classifier	0.506955	0.79	0.65	0.02	0.03
	AdaBoost Classifier	0.517382	0.79	0.60	0.04	0.08

Data Imbalance Upon observing the final dataset, it can be seen that the number of delayed flights constitute less than 20 percent of the entire data. The distribution of the data is given in the figure below. Due to this, the performance of the classifier models decreases as the number of data with which it is trained of is very less in proportion. This condition of the datasets is termed as data imbalance and must be taken care using sampling.



Sampling Sampling is a method used on a data imbalanced dataset to improve the performance of a classifier model by changing the training dataset contents according to the model's performance requirements. In this case, sampling can be done in two ways to achieve the required data set which are oversampling and under sampling.

Oversampling involves increasing the number of 1s present in the dataset or in other words increasing the number of "true" ground truths in the dataset. The dataset produced from oversampling was used to train the models again and their metrics were calculated, which are mentioned below.

Under Sampling involves reducing the number of 0s present in the datasets in other words decreasing the number of "false" ground truths in the dataset. The training data distribution after sampling is visualised in the figure below. The dataset produced from under sampling was used to train the classifier models and their metrics were calculated, which are mentioned below.

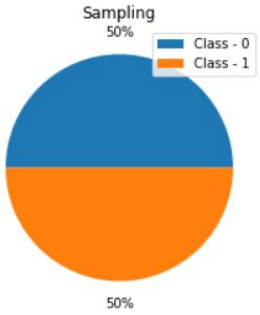


Table 2. Over Sampled Classifier Models and Metrics

Class	Model	AUC	Accuracy	Precision	Recall	F1 Score
Class 0	XGBoost Classifier	0.678735	0.69	0.88	0.70	0.78
	Extra Trees Classifier	0.620328	0.78	0.84	0.90	0.87
	Logistic Classifier	0.6080403	0.61	0.85	0.60	0.71
	AdaBoost Classifier	0.630610	0.63	0.87	0.63	0.73
Class 1	XGBoost Classifier	0.678735	0.69	0.37	0.65	0.47
	Extra Trees Classifier	0.620328	0.78	0.48	0.34	0.40
	Logistic Classifier	0.6080403	0.61	0.29	0.61	0.40
	AdaBoost Classifie	0.630610	0.63	0.31	0.63	0.42

Table 3. Under Sampled Classifier Models and Metrics

Class	Model	AUC	Accuracy	Precision	Recall	F1 Score
Class 0	XGBoost Classifier	0.676409	0.69	0.88	0.69	0.78
	Extra Trees Classifier	0.667277	0.68	0.88	0.70	0.78
	Logistic Classifier	0.607951	0.61	0.85	0.60	0.71
	AdaBoost Classifier	0.630891	0.63	0.87	0.63	0.73
Class 1	XGBoost Classifier	0.676409	0.69	0.36	0.66	0.47
	Extra Trees Classifier	0.667277	0.68	0.36	0.64	0.46
	Logistic Classifier	0.607951	0.61	0.29	0.61	0.39
	AdaBoost Classifier	0.630891	0.63	0.31	0.63	0.42

Sampling does not necessarily result in improving the quality of the dataset for training. Hence the all the classifier models were compared. Sampling can only be done to improve the performance of a classifier but not a regressor as absolute values cannot be sampled.

Upon observing the metrics of different models, it can be seen that Extra Trees Classifier trained with an over sampled training data has the best performance. The ground truth in test data is replaced with the predicted output of the Extra Trees classifier and the dataframe will have been later used in the two-stage model.

4 Regression

After the classification, the second stage of the machine learning engine is a regressor which determines the minutes by which an aircraft would be delayed, is predicted for the flights with respect to the weather using regression models. The datasets are initially changed such that it contains data of "ArrDel15"=1 to make sure that the model is not trained with redundant data values. In this case, the ground truth is"ArrDelayMinutes". The models used in this case were XGBoost Regressor, Decision Tree Regressor, Linear Regressor and AdaBoost Regressor. These models were trained and tested.

The metrics were calculated for every model to compare the performance of models with respect to the dataset. The performance of the models are determined using metrics like Root Mean Squared Error (RMSE) , Mean Absolute Error (MAE) and R Squared (R2). The corresponding equations to determine the respective metrics are as follows.

RMSE is the standard deviation of the errors which occur when a prediction is made on a dataset. This is the same as MSE (Mean Squared Error) but the root of the value is considered while determining the accuracy of the model and is calculated as:

$$RMSE = \sqrt{\frac{1}{n}\Sigma_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{5}$$

Mean Absolute Error refers to the mean of the absolute values of each prediction error on all instances of the test data-set. Prediction error is the difference between the actual value and the predicted value for that instance. MAE is calculated as:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{6}$$

R^2 is a statistical method that represents the proportion of variance for a dependent variable with respect to independent variables in a regression model and is calculated as:

$$R^2 = 1 - \frac{\Sigma(y_i - \hat{y}_i)^2}{\Sigma(y_i - \bar{y})^2} \tag{7}$$

The models and their metrics are given in the table below.

Table 4. Regression Models and Metrics

Model	RMSE	MAE	R^2
XGBoost Regressor	80.847	44.15276	-0.2500846
Decision Tree Regressor	92.14008	54.74519	-0.623711
Linear Regressor	85.12767	46.88885	-0.3859676
AdaBoost Regressor	97.038	79.718585	-0.80094

Upon observing the metrics, the XGBoost Regressor has the best performance when compared with other models for this dataset.

5 Pipeline

The problem statements involves classification of the flights when the necessary features are given as an input, whether they would be delayed or not and by how many minutes will it be delayed. This shows that the output of the classifier is linked to the input of the regressor. Hence this is a pipeline process. In this case, the output of the classifier that was stored as the classified flights will now act as the input to the regressor models. This classified data was passed into the trained regressor models and the delay in minutes was predicted. The output of every model was predicted and metrics were calculated, which are mentioned below.

From the metrics, it can be seen that the performance of the regression models when pipelined is much better than the regressors with the original test data. This assumption is false on all grounds. The test data that is passed to the regressor models is the output of the classifier and this classifier output data would contain errors as the accuracy of the classifier is neither 1.0 nor nearly equal to one. Hence there will be a deviation of the output of the two stage engine when compared with the actual output.

Table 5. Pipeline Regression Models and Metrics

Model	RMSE	MAE	R^2
XGBoost Regressor	56.6688	31.6266	0.140134
Decision Tree Regressor	84.8679	49.3996	-0.928544
Linear Regressor	62.4815	33.88891	-0.0453104
AdaBoost Regressor	150.36891	132.30207	-5.054216

6 Regression Testing

For further evaluation of the regression models, the trained models were passed with grouped values of the test data. The test data was divided into 5 different categories (15-100, 100-200, 200-500, 500-1000, 1000+) and passed into the models for prediction after which their metrics were evaluated. The purpose of this is to evaluate the performance of the models in it's prediction of a particular range. As the XGBoost model has the best performance for the dataset, comparing its metrics on different ranges are mentioned below.

Table 6. XGBoost Regression Metrics in different ranges

Arrival Delay in Minutes	RMSE	MAE	R^2
15-100	31.5312	23.9467	-1.0708817
100-200	108.156	102.2913	-15.306257
200-500	246.851	232.6989	-11.28709
500-1000	702.0748	683.6350	-19.127661
1000+	1290.4224	1279.2172	-61.674175

By the definition of MAE, it shows the variation of the prediction from the ground truth. From the table, the MAE values suggests that the model works better in 15 - 100 and 1000+ range than the other ranges because the value of MAE is within the given range and a variation of that magnitude is considered to be accurate. The range 1000+ is considered accurate as the range of 1000+ itself is very large and equivalent values of variation is accepted as accurate. The value of MAE for the other ranges is greater than its actual range, which means that the error of prediction will be very high and is undesirable.

The amount of training examples in the range of 15 - 100 is more when compared to the other ranges due to which the learning process of the model in those ranges diminish. From the metrics, we can say that the regressors have the a better performance in the range of 15-100 and 1000+.

7 Conclusion

The prediction in the delay of flights before departure is approached with a two step process which involves in the classification of the flights at first and then

passing the it's output to predict the delay in minutes. It initially involves in the processing of the data into a suitable format for the purpose of training and testing models. The data must be sampled due to a data imbalance and the classifiers trained and tested accordingly to determine the best fit for the data whose output is then fed as an input to the trained regression models to predict the delay in minutes as the solution is a pipeline process. It is seen that the models have a better performance in the range of 15-100 as the data available in the other ranges are very less in number and also in the range of 1000+ since a variation in such a wide range is negligible.

The project involves methods to improve the performance of the models by improving the data (such as sampling) and also several ways to evaluate the performance of the models to realise areas which needed improvements. As we know the machine learning models are data driven, the quality of data available determines the performance of the models. The outputs from the machine learning models are accurate to a great extent but with a better and larger dataset as this, the models can be trained to higher level of perfection.