A Two Stage Predictive Machine Learning Engine to Forecast Flight On-Time Performance

Akash Ambashankar

KCG College of Technology
Chennai, TN, India
akashambashankar@gmail.com

Abstract. Delayed flights are a major challenge for airlines today. They are a cause of extensive turmoil and confusion for airlines as well as passengers. In order to deal with the issue of flight delay, this project proposes a Two Stage Predictive Machine Learning Engine that is able to classify delayed flights and predict the arrival delay period after takeoff, using corresponding flight information along with the relevant weather forecast.

Keywords: Flight Delay Prediction · Regression · Classification.

1 Introduction

In the last few years, air travel has become a common, easy and affordable mode of transport. Over 100,000 commercial flights operate on a daily basis. One of the few critical issues in air travel is delays in flight arrival. While flights are delayed for a number of reasons such as flight-crew delay or bird strikes, **adverse weather conditions** are the leading cause for delayed flights, as they affect the visibility and stability of the flight. Around 200,000 flights arrive delayed each year which causes huge losses for airlines as well as their customers. Unexpected delays leave many passengers stranded for long hours, hinder businesses, and cause additional expenses for airlines in the form of compensation and rescheduling of flights.

This project aims to predict flight delays by constructing a Two Stage Predictive Machine Learning Engine, consisting of a Classifier and a Regressor. Using flight and weather information, the classifier will predict which flights will be delayed and for these flights, the regressor will predict the delay period. This will enable airlines to proactively take measures to minimise the effect of these delayed flights, thereby improving efficiency, profits and customer satisfaction.

This project report discusses the dataset being used, analyses the performance of the classifier and regressor through appropriate metrics, and demonstrates a pipeline to identify delayed flights and predict their arrival delay period.

2 Dataset

The dataset uses a combination of flight and weather information of fifteen airports collected in 2016 and 2017. The airports codes of the airports considered are given in Table 1.

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

Table 1. Airport Codes

The flight data has been sourced from the "On-Time" database of the TranStats data library. The features of flight data taken into consideration are given in Table 2.

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

Table 2. Flight features

The weather data has been sourced from World Weather Online. The features of weather data taken into consideration are given in Table 3.

WindSpeedKmph	WindDirDegree	WeatherCode	precipMM	Visibility
Pressure	CloudCover	DewPointF	WindGustKmph	tempF
WindChillF	Humidity	Date	Time	Airport

Table 3. Weather features

The flight and weather data are pre-processed and merged by mapping each flight with its corresponding weather information using the features given in Table 4.

Year	Month	DayofMonth
Origin	Dest	DepTime
ArrTime		

Table 4. Features used to merge flight and weather data

Henceforth, in this report, the resulting dataset will be referred to as *flight dataset*.

3 Classification

Classification is the process of predicting the class label for a given sample based on the input features. In this project, the goal is to classify flights as *On Time* (class 0) or *Delayed* (class 1) using *ArrDel15* as the target variable along with input features from the flight dataset.

The ground truth dataset is split, with 80% of samples used as training data and the remaining 20% used as testing data. The classifier models that were considered are Logistic Regressor, Decision Trees, Support Vector Machine (SVM), Extra Trees Classifier, Extreme Gradient Boosting (XGBoost) Classifier, and Random Forest Classifier.

3.1 Evaluation Metrics

In order to evaluate the performance of the aforementioned models, the following evaluation metrics are used.

Accuracy The ratio between the number of correctly classified flights and the total number of flights.

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

Precision The ratio between the number of flights that were correctly classified as delayed and the total number of flights classified as delayed.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

Recall The ratio between the number of flights that were correctly classified as delayed and the total number of flights that were actually delayed.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

F1-Score Harmonic mean of Precision and Recall, that gives equal importance to both.

$$F1 - Score = 2 \times \frac{precision \times recall}{precision + recall} \tag{4}$$

where,
TP = True Positive, TN = True Negative,
FP = False Positive, FN = False Negative

Balanced Accuracy A modified version of Accuracy (Eqn. 1) that is used when data is imbalanced. Balanced Accuracy assigns a higher weight to the minority classes and vice versa, so that the accuracy score equally represents every class of the dataset.

Balanced Accuracy can also be defined as the arithmetic mean of Sensitivity and Specificity, where Sensitivity is the True Positive Rate (Recall for class 1) and Specificity is the True Negative Rate (Recall for class 0).

$$Balanced\ Accuracy = \frac{Sensitivity\ +\ Specificity}{2} \tag{5}$$

3.2 Results

The objective of the classifier is to predict delayed flights correctly. Thus, Recall must be maximised. But from Eqn 5, it is seen that Balanced Accuracy is the arithmetic mean of Recall 0 and Recall 1. Thus, it is considered as the primary metric for Classification.

The results for each model are shown in Table 5.

Classifier	Precision		Recall		F1-Score		Accuracy	Balanced Accuracy
	0	**1**	**0**	**1**	**0**	**1**		
Logistic Regressor	0.92	**0.89**	**0.98**	0.69	**0.95**	0.77	0.9166	0.8315
Decision Trees	0.92	0.68	0.91	**0.71**	0.92	0.70	0.8704	0.8102
Random Forest	**0.93**	0.88	0.97	0.70	**0.95**	**0.78**	**0.9178**	**0.8390**
XGBoost	0.92	**0.89**	**0.98**	0.69	**0.95**	**0.78**	0.9176	0.8312
Extra Trees	0.92	0.87	0.97	0.67	**0.95**	0.76	0.9103	0.8221
SVM	0.92	**0.89**	**0.98**	0.68	**0.95**	0.77	0.9167	0.8302

Table 5. Classification Results

From Table 5, it is inferred that Random Forest Classifier and XGBoost Classifier have similar Recall scores. But the former has a better Balanced Accuracy score. Thus *Random Forest Classifier* is noted as the best performing classifier.

From the overall results, it is inferred that the scores for class 1 predictions are much lower than that of class 0. Moreover, there is significant difference between the Accuracy and Balanced Accuracy scores for all models. *This implies that the dataset is imbalanced and resampling is required to overcome this imbalance.*

4 Data Imbalance

Data imbalance is a phenomenon encountered while classifying data wherein the distribution of data among the various classes is disproportionate. Due to imbalance, the scores of minority classes are lower than the other classes.

From Table 5, it is observed that the scores for On Time flights (class 0) are much better than Delayed flights (class 1). This is because the former is the majority class of the flight dataset, as shown in Figure 1. This results in a lower score for class 1 in the classification metrics.

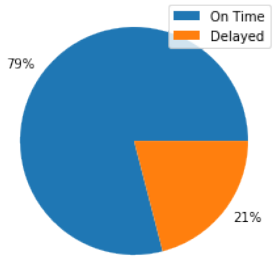

Fig. 1. Imbalanced distribution between class 0 and class 1

In order to eliminate Data Imbalance from the flight dataset, Resampling is used. Resampling is a method used to balance data and improve minority class scores. The two types of resampling are *Oversampling* and *Undersampling*.

Oversampling Artificially generating data points of the minority class to increase its population is known as Oversampling. *Random Oversampling* is a technique wherein random samples of the minority class are duplicated until the distribution of data is equal.

Undersampling Removing data points from the majority class to reduce its population is known as Undersampling. *Random Undersampling* is a technique wherein random samples of the majority class are deleted until the distribution of data is equal.

4.1 Results

Table 6 shows the results of performing *Random Oversampling* and Table 7 shows the results of performing *Random Undersampling*.

Classifier	**Precision**		**Recall**		**F1-Score**		**Accuracy**	**Balanced Accuracy**
	0	**1**	**0**	**1**	**0**	**1**		
Logistic Regression	**0.94**	0.74	0.93	0.78	0.93	0.76	0.8954	0.8530
Decision Trees	0.92	0.69	0.92	0.70	0.92	0.70	0.8713	0.8072
Random Forest	0.93	0.84	0.96	0.73	**0.95**	**0.78**	**0.9142**	0.8474
XGBoost	**0.94**	0.73	0.92	**0.79**	0.93	0.76	0.8951	**0.8553**
Extra Trees	0.91	**0.88**	**0.98**	0.65	0.94	0.74	0.9063	0.8106
SVM	**0.94**	0.76	0.93	0.77	0.94	0.76	0.8997	0.8522

Table 6. Classification with Random Oversampling Results

Classifier	**Precision**		**Recall**		**F1-Score**		**Accuracy**	**Balanced Accuracy**
	0	**1**	**0**	**1**	**0**	**1**		
Logistic Regression	0.94	0.74	**0.93**	0.78	0.93	**0.76**	0.8958	0.8534
Decision Trees	0.94	0.51	0.79	0.80	0.86	0.62	0.7955	0.7988
Random Forest	**0.95**	0.72	0.91	0.81	0.93	**0.76**	0.8930	**0.8635**
XGBoost	0.94	0.73	0.92	**0.79**	0.93	**0.76**	0.8952	0.8557
Extra Trees	**0.95**	0.65	0.88	0.83	0.91	0.73	0.8699	0.8540
SVM	0.94	**0.76**	**0.93**	0.77	**0.94**	**0.76**	**0.8996**	0.8525

Table 7. Classification with Random Undersampling Results

Upon comparison of the scores from Tables 5, 6, and 7, it is seen that Resampling does not have a significant effect on the scores. The goal of introducing Resampling was to improve class 1 scores, but resampling was unsuccessful in doing so. Hence, **unsampled data** is used for classification.

5 Regression

Regression is the process of predicting a continuous value based on the input features. Having classified the delayed flights, regression is used to predict the Arrival delay period for each of these flights. *ArrDelayMinutes* is used as the target variable.

The ground truth dataset is split, with 80% of samples used as training data and the remaining 20% used as testing data. The regression models that were considered are Linear Regressor, Support Vector Machine (SVM), Extra Trees Regressor, Extreme Gradient Boosting (XGBoost) Regressor, and Random Forest Regressor.

5.1 Evaluation Metrics

In order to evaluate the performance of the aforementioned models, the following evaluation metrics are used

R-Squared (R2) The measure of dependence of the target variable, *ArrDelayMinutes* on the independent features of the flight dataset.

$$R2 = 1 - \frac{\sum(y - y_{pred})^2}{\sum(y - y_{mean})^2} \tag{6}$$

Mean Absolute Error (MAE) Average absolute difference between the predicted delay period and actual delay period across the dataset.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y - y_{pred}| \tag{7}$$

Root Mean Squared Error (RMSE) Root Squared Average of the difference between the predicted delay period and actual delay period across the dataset.

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y - y_{pred})^2} \tag{8}$$

where,
$y = Actual\ Arrival\ Delay$
$y_{pred} = Predicted\ Arrival\ Delay$
$y_{mean} = Average\ Arrival\ Delay$
$N = Total\ Number\ of\ Delayed\ Flights$

5.2 Results

The results for each model are shown in Table 8.

Regressor	R2	MAE	RMSE
Linear Regression	0.9394	12.1859	17.5426
Extra Trees	0.9454	11.7460	16.6479
XGBoost	0.9443	**11.6326**	16.8081
Random Forest	**0.9461**	11.6896	**16.5420**
SDG Regressor	0.9391	12.2712	17.5830

Table 8. Regression Results

The objective of the regressor is to predict flight delay period for each delayed flight with minimum error. Hence, **R2** and **RMSE** are considered as the primary metrics. On this basis, *Random Forest Regressor* is noted as the best performing regressor.

6 Regression Analysis

The performance of the Random Forest Regressor is evaluated across different Arrival delay period intervals. The result for each interval is shown in Table 9.

Arrival Delay Minutes - Range	MAE	RMSE	No. of Samples
0 - 100	11.0098	14.6363	97035
100 - 200	17.5633	26.3952	14761
200 - 500	18.6736	29.4391	4228
500 - 1000	20.5524	29.7369	338
1000 - 2000	26.7076	33.1749	56

Table 9. Regression Analysis

From Table 9, the MAE and RMSE scores indicate that regression is effective only at lower delay intervals. This is a result of the distribution of the dataset.

The Interquartile range of Figure 2 indicates that majority of samples of flight dataset occur between 25 to 75 minutes, which lies within the 0-100 interval. Since majority of the data used to train the regressor is from this interval, it has the least MAE and RMSE score.

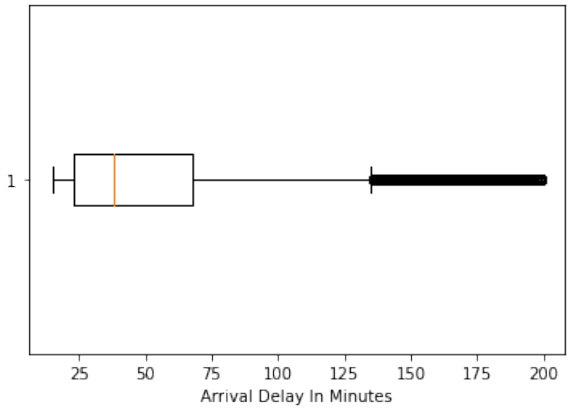

Fig. 2. Distribution of dataset

However, the Error Percentage in the predictions of the outlier delay intervals must also be taken into consideration.

$$Error\ Percentage = \frac{\frac{RMSE}{lower\ limit\ of\ interval} + \frac{RMSE}{upper\ limit\ of\ interval}}{2} \times 100 \quad (9)$$

Error Percentage uses the RMSE score, as well as the lower and upper limits of the interval considered, to provide a more comprehensive measure of error in predicted values. In higher delay intervals, a small percentage of error is acceptable.

The error percentage at various intervals is shown in Table 10

Arrival Delay Minutes - Range	**Error Percentage**
0 - 100	14.6363
100 - 200	13.1976
200 - 500	5.8879
500 - 1000	2.9737
1000 - 2000	1.6587

Table 10. Error Percentage at each Delay Interval

It is observed that the error percentage is relatively low at higher delay intervals. This is shown in Figure 3.

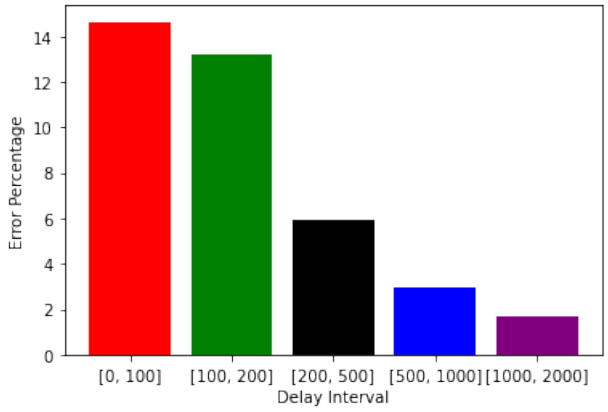

Fig. 3. Error Percentage vs Delay Interval

In summary, a low RMSE value is observed at lower intervals, while low error percentage is observed at higher intervals.

7 Pipeline

In Sections 3 and 5, it is noted that *Random Forest* is the best performing classifier and regressor. In this Section, the two are combined to form a pipeline, which will identify delayed flights, and then predict the arrival delay period for each delayed flight.

A Random Forest Classifier is trained to classify flights as On Time or Delayed. The flights classified as delayed are fed to a Random Forest Regressor, which has been trained on the Ground Truth Delayed flights. The regressor predicts the Arrival delay in minutes. The pipeline has been illustrated in Figure 4.

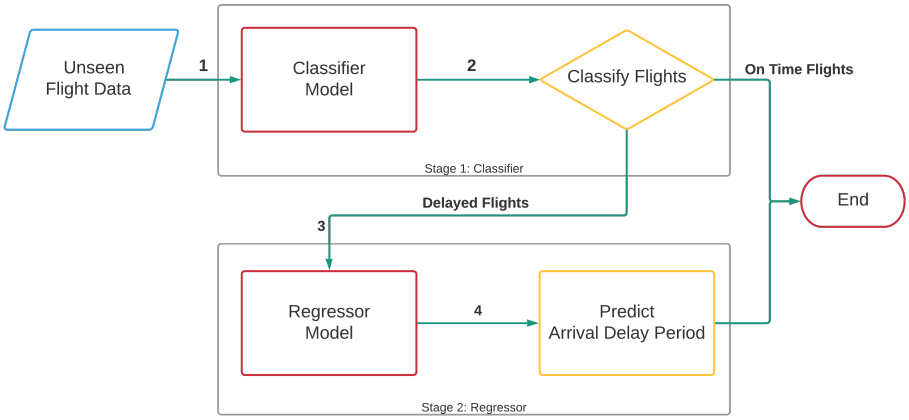

Fig. 4. The Pipelined Two Stage Predictive Engine

7.1 Results

The results of the pipeline are shown in Table 11.

Regressor	R2	MAE	RMSE
Random Forest	0.9469	13.7386	18.6446

Table 11. Pipeline Results

From Table 11 it is seen that pipelining does not improve the performance of the regressor.

8 Conclusion

This project addresses the problem of Flight Delay Prediction through the use of flight information with corresponding weather details. After analysing the performance of every model, Extra Trees Classifier and Random Forest Regressor were determined as the most accurate classifier and regressor.

Using a pipeline, this project has demonstrated a Two Stage Predictive Machine Learning Engine that is able to classify delayed flights and predict arrival delay period of each flight using the given data. This will be greatly beneficial for airlines and their passengers as it can improve the efficiency and reliability of air travel.