**Prediction Of Flight Arrival and Departure Delay**

Dharun Selvan

Department of Data Science

Depaul University,

Chicago, illinois 60616

dselvan@depaul.edu

Vallabha Datta Penmetcha

Department of Data Science

Depaul University,

Chicago, illinois 60616

vpenmetc@depaul.edu

Abinaya Badhinath

Department of Data Science

Depaul University,

Chicago, illinois 60616

abadrina@depaul.edu

Shadhana Palaniswami

Department of Data Science

Depaul University,

Chicago, illinois 60616

Spalani1@depaul.edu

Word count: 5134 word + 11 Fig (11 X 250) = 7884 words.

1. **ABSTRACT**

Delays in airlines are one the major drawbacks for the travellers and in the business point of view. In this project we attempted to find whether there will be any delay in the line if so how long will be the delay in both arrival and departure. We used multiple Machine Learning techniques to find these. Classification was used to find whether there will be any delay and Regression was used to find how long the delay will be. We were able to achieve 97% accuracy overall in our project. *Keywords: Machine Learning , Classification , Regression.*

1. **INTRODUCTION**

Flight delays are a fundamental problem, which costs airlines, passengers, and the U.S. economy. A 2007 study by National Center of Excellence for Aviation Operations Research (NEXTOR) estimates that air transportation delays cost a total of $32.9 billion (about $100 per person in the US) (Ball, 2010). The same report mentions over 25 percent of flights delayed (15+ minutes) and cancelled. 43.6% of all flight delays are caused by weather-related conditions (BTS, 2019). So, in this project we developed a model that will predict the flight arrival /departure time using the weather data. Our solution will help mitigate the uncertainty of flight delays and flight cancellations.

1. **LITERATURE REVIEW**

[1] The authors did it in two parts. Part one is predicting the departure delay. Part two is predicting the Arrival Delay. They were able to achieve 86% in departure delay and 94% in arrival delay. In both parts they used Classification and regression to find whether there is any delay and if so how much is the delay.

[2] The authors took data from july 2019 to december 2019.They did this study two phase.1st phase - Is there any delay? 2nd phases - If yes , How much delay? They used classification for phase 1 and Regression for phase 2. They were able to get 94% accuracy in phase 1 using XGBoost classification and 93% accuracy for the linear regression model for phase 2.

[3] Predicting arrival delay of scheduled flights, classification models for origin-destinations were trained with airline and weather data. The data was classified into on-time and delayed flights. Their Random forest model had the highest accuracy of 83.40.

[4] Their model has a two - stage approach where the first stage predicts the daily delay status using deep RNN technique while the second stage predicts the individual flight delay using the output of the first stage. Total day-to-day accuracy for 10 different airports were recorded and except for Phoenix, the accuracy rate lied between 85% to 91% and Phoenix’s accuracy was over 71%. By applying LSTM RNN architecture to the predictive model, a dependable delay status can be obtained. By learning the day-to-day delay statuses, we can predict the delay of a specific flight accurately by 85 – 91%.

[5] According to the Federal Aviation Administration (FAA), an aircraft is deemed delayed if it is delayed for more than 15 minutes. The authors of the paper also indicated that only 25% of flights fall within that range, hence it is more necessary to consider precision or recall than accuracy when modeling. They have applied multi-class categorization and determined that more data is needed to increase recall. Complex models like CNN and RNN may be able to capture the dynamo effect.

[6] The research investigates the flight delay data by constructing visualizations in order to determine the amount to which class imbalance impacts learning performance, as well as other probable factors that may impact the model's performance. To evenly divide the target variable among the split, the authors used the t-SNE dimensionality reduction methodology. Based on the findings, they concluded that the data contained class overlaps, which should be taken into account when making predictions.

[7] The effectiveness of this successful paradigm in air traffic delay prediction tasks is investigated in this research. A precise and robust prediction model has been constructed by merging this regression model based on the machine learning paradigm, allowing for a detailed investigation of air traffic delay patterns. In representing sequential data, the Gradient Boosted Decision Tree has demonstrated to be extremely accurate. Day-to-day sequences of departure and arrival flight delays at a specific airport can be accurately predicted using this model. The model was tested on passenger flight on-time performance data from the United States in this publication.

[8] The authors examined the major components that can cause flight delay propagation, as well as their interrelationships, in this research, and provided a model and prediction algorithm for flight delay propagation. To overcome the constraints of existing prediction models, this work provides a unique prediction model and corresponding algorithm for flight delay propagation that considers both critical flight and essential airport resources. Simulations show that the model and technique may be used to give a reliable technique for quantifying flight delay propagation.

[9] Cat-boost model is used to domestic airline on-time performance data from the United States Transportation Administration, together with the model's features, to identify influencing factors and estimate flight arrival delays within the United States. The model's accuracy, precision, and other criteria are used to assess its performance on the data

[10] In this research, a machine learning-based strategy is proposed for predicting airline arrival delays by taking into account input characteristics ranging from distance to weather conditions to determine whether or not a certain flight is delayed. To calculate flight delay, it employs neural networks and deep learning techniques.The proposed method is evaluated on a large dataset of real-world flight data, yielding an accuracy of 77 percent for deep nets and 89 percent for neural nets.

1. **DATA MINING**
2. Flight Data

We started to obtain data sets of airlines from https://www.transtats.bts.gov/ONTIME/ for 5 different airlines. They are American Airlines (AA) , Alskan Airline (AS) , Delta Airline (DL) , Spirit Airlines (NK) and United Airline (UA). We took these data for 10 different cities in which we had to consider 11 airports to cover all those airlines. The air ports are O’Hare international airport (ORD - Chicago), John F. Kennedy International Airport (JFK - New York City) ,LaGuardia Airport ( LGA -New York City) , Tampa International Airport (TPA -Tampa ), George Bush Intercontinental Airport ( IAH - Houston) , Dulles International Airport (IAD - Washington DC) , Raleigh-Durham International Airport ( RDU - Raleigh ) , San Francisco International Airport ( SFO - San Francisco ) , Los Angeles International Airport (LAX - Los Angeles ) ,Hartsfield-Jackson Atlanta International Airport ( ATL - Atlanta ) and McCarran International Airport ( LAS - Lasvegas).For each airline in an airport we got one csv file containing all the flight information.We obtained 98 dataset for the airline information. 49 for arrival and 49 for departure.

Table 1. Columns in Arrival Data:

|  |  |  |
| --- | --- | --- |
| S NO | Variable Name | Explanation |
| 1 | CarrierCode | Short term representation of the Airline. |
| 2 | DateMMDDYYYY | Date variable in for MMDDYYYY. |
| 3 | FlightNumber | Flight number consists of 4 digits which has 2 word for the airline and 2 to 4 numbers that mentions route irrespective of the stops. |
| 4 | TailNumber | Tail Number is the identification number where each flight will get on during their registration. |
| 5 | OriginAirport | Airport from where the flights will depart |
| 6 | ScheduledArrivalTime | The planned arrival time of the flight |
| 7 | ActualArrivalTime | The time when the flight arrived the airport |
| 8 | ScheduledElapsedTimeMinutes | Planned minutes the plane spent on air |
| 9 | ActualElapsedTimeMinutes | Actual minutes the plane spent on air |
| 10 | ArrivalDelayMinutes (Y) | Minutes delay in Arrival airport. |
| 11 | WheelsonTime | The time when wheels touched the runway |
| 12 | TaxiIntimeMinutes | Minutes the plane spends on the runway. |
| 13 | DelayCarrierMinutes | Delay happened due multiple reasons inside the carriers including delays due to cleaning, delay in connecting passengers or bags etc. |
| 14 | DelayWeatherMinutes | Delay due to weather. |
| 15 | DelayNationalAviationSystemMinutes | Delays happened in departure or due to air traffic. |
| 16 | DelaySecurityMinutes | Delay due to security clearance or queue larger than 30 mins in security check. |
| 17 | DelayLateAircraftArrivalMinutes | Delay in flight reaching the gate from the runway. |
| 18 | ArrivalAirport | Airport which flights are arriving. |
| 19 | ArrTime | Rounded off the arrival time to the nearest whole number for merging purposes. |
| 20 | zip code | Zip Code of the airport. |

Departure data have almost the same data except it is everything from the departure point of view.

1. Weather Data

Weather data was obtained from https://www.worldweatheronline.com/. We used the WorldWeatherPy package in python to access it in an API. Using this API we were able to obtain hourly weather for the airport location using it’s zip code as dataframe.

Table 2: Weather data columns:

|  |  |  |
| --- | --- | --- |
| S.No | Variable Name | Explanation |
| 1 | date | Recorded date |
| 2 | time | Recorded time - hourly |
| 3 | maxtempC | Maximum temperature recorded in that day in Celcius |
| 4 | mintempC | Minimum temperature recorded in that day in Celcius |
| 5 | totalSnow\_cm | Total Snow on centimeters |
| 6 | sunHour | Total sun hours presented in the recorded day |
| 7 | uvIndex | An average person to experience UltraViolet rays in the sky |
| 8 | moon\_illumination | Total moon hours presented in the recorded day |
| 9 | moonrise | Time when moonrise happened |
| 10 | moonset | Time when moonset happened |
| 11 | sunrise | Time when sun rise happened |
| 12 | sunset | Time when sunset happened |
| 13 | DewPointC | Measurement of Dew point in Celcius |
| 14 | FeelsLikeC | Feels like temperature in Celcius |
| 15 | HeatIndexC | An average person might take heat as a result of temperature and humidity in the air in Celcius |
| 16 | WindChillC | A still air temperature in Celcius |
| 17 | WindGustKmph | Measurement of wind Gust in Kilometer Per Hour |
| 18 | cloudcover | How much cloud is covered in the sky. |
| 19 | humidity | Humidity measured at that time. |
| 20 | precipMM | Precipitation in air measured in MilliMeter |
| 21 | pressure | Air Pressure |
| 22 | tempC | Temperature in Celcius |
| 23 | visibility | Number of kilometers visible in naked eye (1Km to 10Km) |
| 24 | winddirDegree | Wind Direction |
| 25 | windspeedKmph | Wind speed in Kilometer Per Hour |
| 26 | city | Zip Code of the city. |

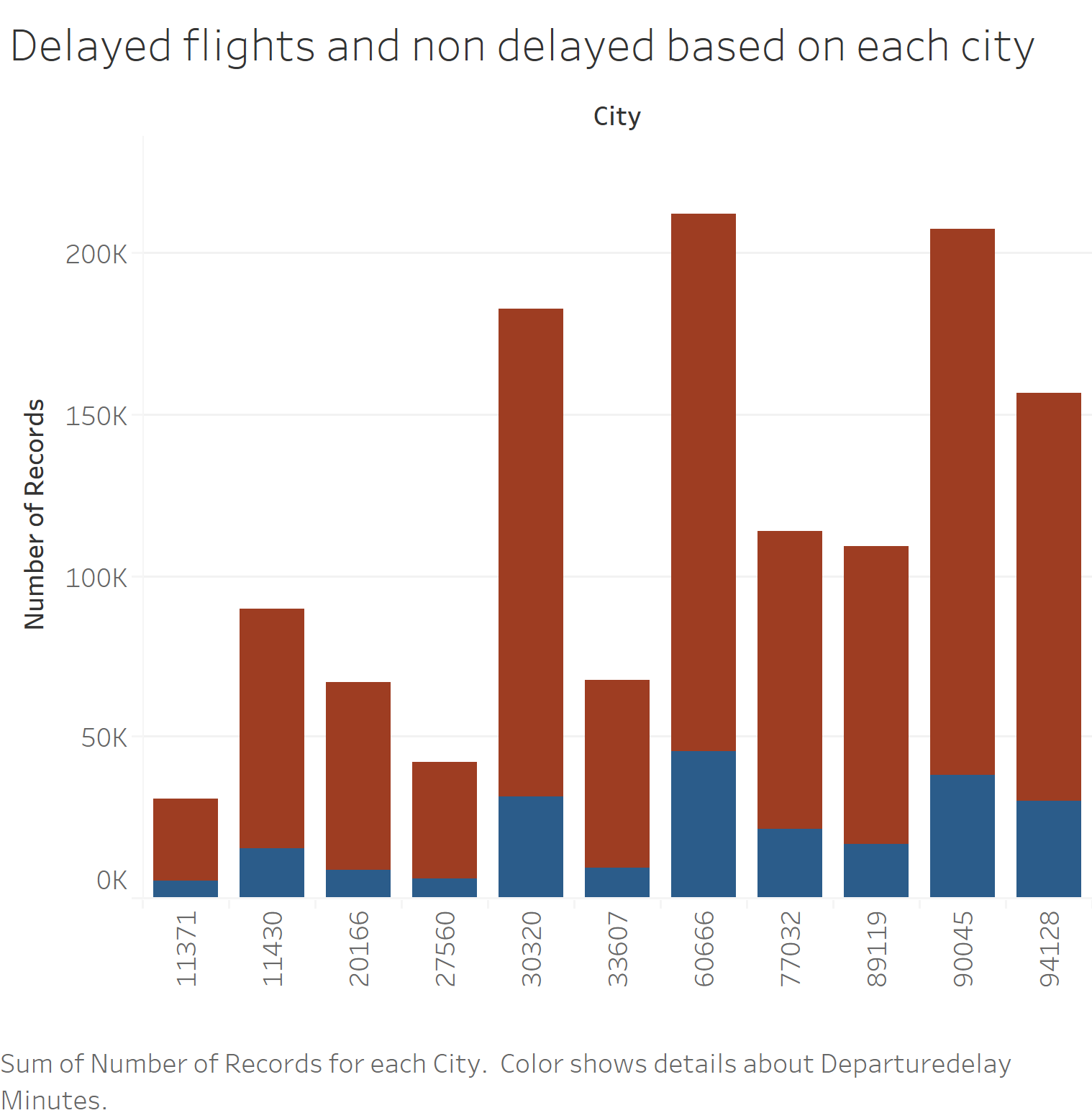
1. **DATA PRE-PROCESSING**

We started pre-processing by merging the files we got. We looked for the missing values in the data set, but we didn't find any missing values in all the columns of the airline data except flight tail number. Later we found that the flight tail number missing from the data set is not an error, it is missing because the flights were cancelled. We removed the cancelled flights from our data set. All 49 departure airlines data was combined into one file similarly the arrival data was also done. Filters were applied such that we took only data for the airports which we selected above.

All the weather data into one file to merge it with arrival data and departure data. we used pandas inner joint function to merge weather data and the online data we used a date, time and the zip code of the airport as the key to merge it. The zip code of the airport was not in the airline data set and was added manually. Using domain knowledge, we removed all the unwanted columns such as origin airport, destination airport, flight tail number, airline code, flight number etc.

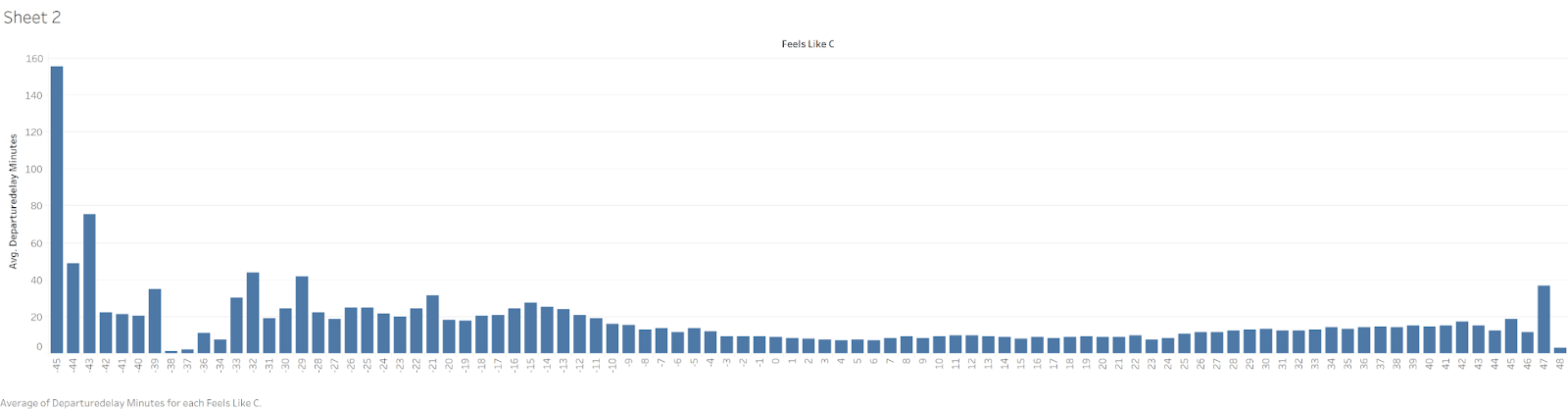
1. **EXPLORATORY DATA ANALYSIS:**

Fig 1: City vs Delay



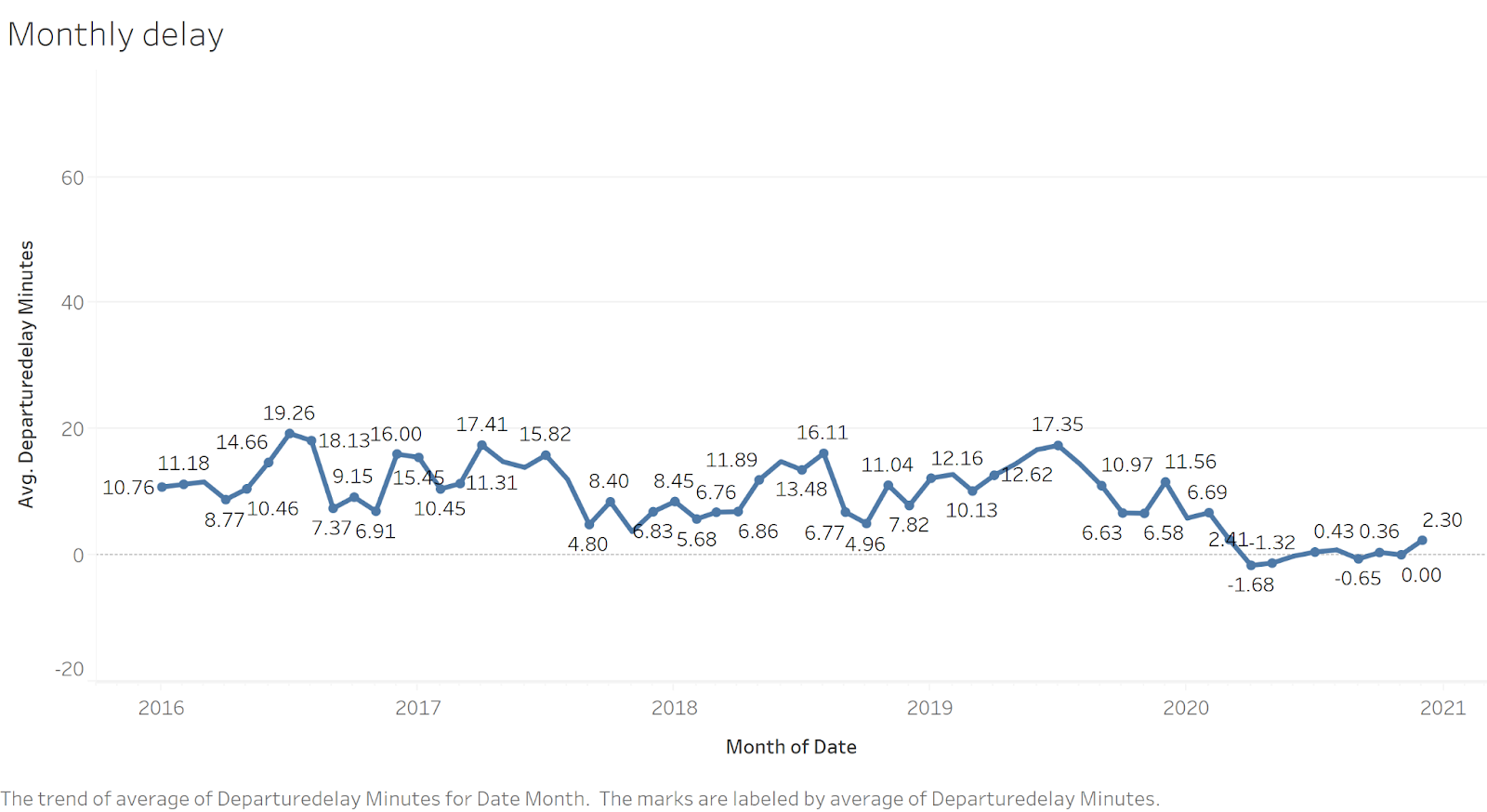
Red ones in the graph were on time flights and blue ones were delayed flights. Highest delayed was happened in 60666 i.e. ORD ( Chicago ). Lowest delayed was happened in 27560 i.e. RDU ( Raleigh ). Based on percentage delay. Almost all airports are equally delayed in percentage wise.

Fig 2: Temperature feels like vs Average Delay



Average delay based on temperature (Feels like in C) shows that delays are happening when temperatures are moving towards extremes. As expected, colder temperatures have more impact than normal temperature.

Fig 3: Monthly Avg delay in Departure



Average delay with regard to every month was displayed to find the delay trend and we can see that higher average delay was happening during the month July and August in the first 4 years. In 2020, maybe because of the Pandemic the peak didn’t happen in summer.

1. **METHODOLOGY**
2. **Classification**

The classification process's goal is to predict whether the flight will be delayed or not, as well as to identify what aviation and weather factors influence the prediction. Firstly, classification was performed on both arrival and departure datasets. Later, the hourly weather data for the chosen areas was combined with airline data to identify the features and the extent to which they influence flight on-time performance.

The target variables (ArrivalDelayMinutes and DepartureDelayMinutes) are converted into binary format (0 = not delayed, 1 = delayed) to be used in the classification methods. A flight is considered delayed if the delay minutes exceed 15 minutes, as defined by the Federal Aviation Administration.

The dataset was divided into train and test sets, with the train set accounting for 70% of the data and the test set accounting for 30% of the data. Six different classification techniques were used to predict whether or not a flight would be delayed using the training set, and the performance of the models was evaluated using the test set. We initially fit the full models with the default parameters, and it was observed that they performed well on train data but poorly on test data, implying overfitting. Taking this into consideration, GridSearchCV is used, which is a method of determining the optimal parameters by performing hyperparameter tuning for the model's specified metric. In the project the GridSearchCV is implemented using precision as a metric because, in addition to predicting whether the flight would arrive on time or not, we also wanted to reduce the total number of false positives.

Although hyperparameter tuning improved the model's performance, there are still issues in the data, such as too many insignificant features. As a result, feature selection methods such as Mlxtend, backward elimination, and dimensionality reduction techniques such as PCA have been implemented to improve training speed while also lowering modeling costs. Table 3 shows a comparison of the three techniques. The Mlxtend forward selection is identified as the optimal technique which gives the best results with minimum number of features. Table 4 shows the cumulative score which defines the percentage of variance explained by the features chosen by the Mlxtend method.

Table 3: Comparison of Feature Reduction Techniques

|  |  |  |  |
| --- | --- | --- | --- |
| Method | Number of Features | Accuracy (%)  Weighted Average | Precision (%)  Weighted Average |
| Full model | 44 | 100 | 100 |
| Mlxtend | 8 | 95 | 95 |
| Backward Elimination | 25 | 95 | 95 |
| PCA | 5 | 89 | 89 |

Table 4: Cumulative score of the features

|  |  |
| --- | --- |
| Feature | Score |
| Delay Carrier Minutes | 0.8925 |
| Delay National Aviation System Minutes | 0.9291 |
| Taxi In-time minutes | 0.9429 |
| Delay Late Aircraft Arrival Minutes | 0.9467 |
| PrecipitationMM | 0.9491 |
| Humidity | 0.9492 |
| Visibility | 0.9492 |
| Pressure | 0.9493 |

1. **RESULTS AND DISCUSSION**

A. Classification

We used six alternative classification model strategies, as indicated in Table III, to predict whether the flight will be delayed or not (arrival and departure delays). Performance methods like accuracy, precision, recall, and f1-score are used to evaluate each performance on a test set. On the basis of the confusion matrix and classification report, the various models are evaluated, and the best model with the best predictions is chosen.

(A) Departure Data With Inclusion Of Weather Parameters

To understand what weather parameters actually affect the prediction of whether a departure flight is delayed, we fit and compare the six classification models as displayed in Table V on the departure data combined with weather data. The values shown are the % of the weighted average of the predicted points, therefore they are whole numbers. Although the weighted averages of decision tree, random forest, gradient boosting and XGBoost are comparable in the table, the random forest was chosen as the final model because it produced the best confusion matrices, as seen in Fig. 4. According to the confusion matrix, the final random forest model was able to predict the departure flight on time performance with an accuracy and precision of 95%.

Table 5: Comparison of The Classification Models on Departure Data Combined with Weather Data

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| Naïve Bayes | 93 | 93 | 93 | 93 |
| Decision Tree | 95 | 95 | 95 | 95 |
| Random forest | 95 | 95 | 95 | 95 |
| Adaboost | 94 | 94 | 94 | 94 |
| Gradient Boosting | 95 | 95 | 95 | 95 |
| XGBoost | 95 | 95 | 95 | 95 |

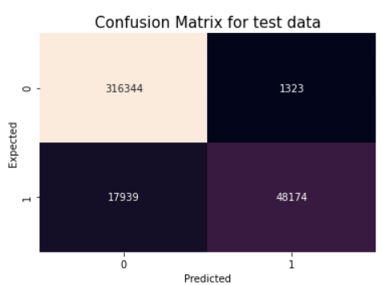


Fig. 4. Confusion matrix of Random Forest Model

Fig. 5. displays the feature importance of the random forest model's variables. The feature importance plot's purpose is to discover which airline and weather elements influence departure timings. The metric DelayLateAircraftArrivalMinutes accounts for around 45% of variance in departure delays followed by DelayCarrier and DelayNationalAviationSystem minutes. As a result, if the previous flight of the same airline arrives late, the current flight's departure time is delayed. It was also discovered that the relative importance of weather parameters is less than 1%, implying that weather conditions have little influence on departure times. Departing flights will be canceled if there is severe weather or airport system damage. It may be inferred that only airline data can be used to estimate flight departure times, whereas both airline and weather data should be utilized to estimate flight arrival times.

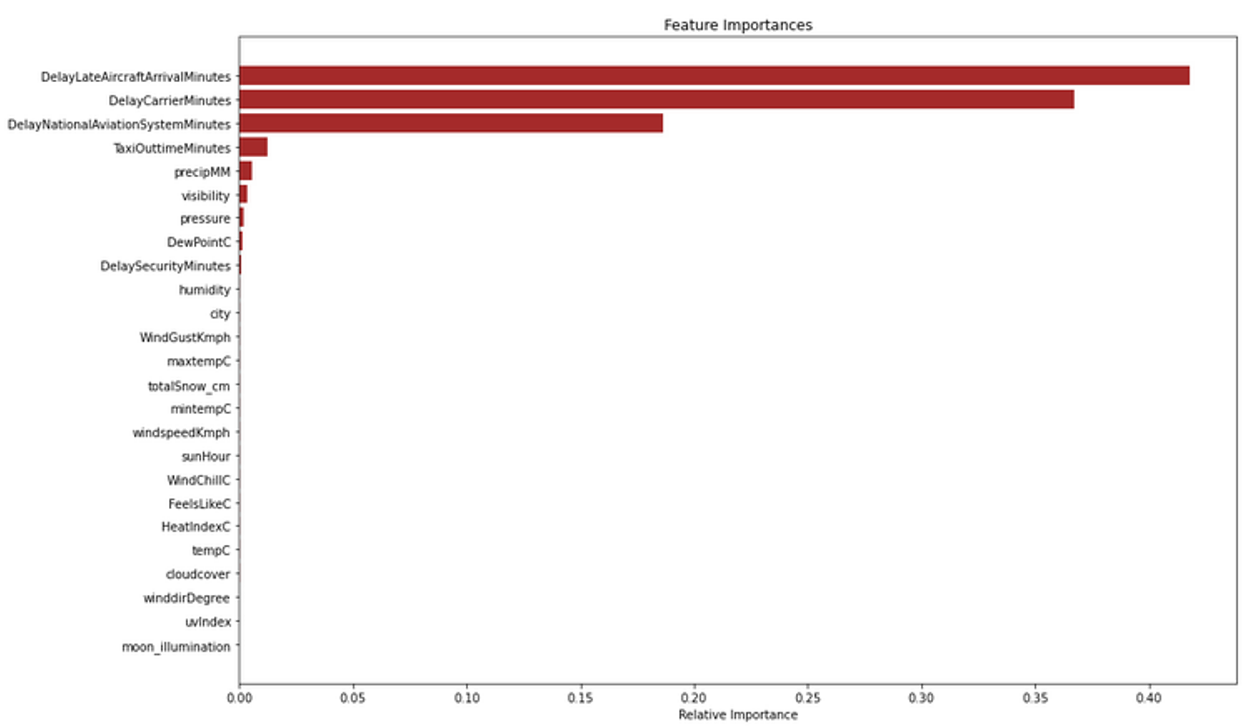


Fig. 5. Percentage of Variance Explained by the Features

(B) Arrival Data Without Inclusion Of Weather Parameters

The comparison of all six models is shown in Table 6. The values shown are the % of the weighted average of the predicted points, therefore they are whole numbers. Although the weighted averages of decision tree, random forest, and XGBoost are comparable in the table, the decision tree was chosen as the final model because it produced the best confusion matrices, as seen in Fig. 1. According to the confusion matrix, the final decision tree model predicted all actual non-delayed arrival flights as non-delayed and real delayed arrival flights as delayed with an accuracy of 98 percent.

Table 6: Comparison of The Classification Models

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| Naïve Bayes | 100 | 100 | 100 | 100 |
| Decision Tree | 98 | 98 | 98 | 98 |
| Random forest | 98 | 98 | 98 | 98 |
| Adaboost | 93 | 94 | 93 | 93 |
| Gradient Boosting | 93 | 94 | 93 | 93 |
| XGBoost | 98 | 98 | 98 | 98 |

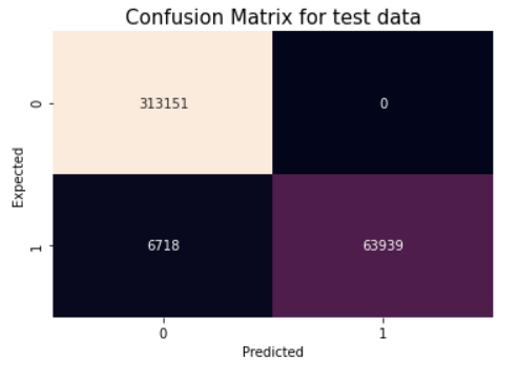


Fig. 6. Confusion matrix of Decision Tree Model

Fig. 6. depicts the feature importance of the variables chosen by the decision tree model. The goal of the feature importance plot is to determine which airline factors impact arrival times. According to the plot, the parameter DelayNationalAviationSystemMinutes (airport operations and heavy traffic volume) accounts for around 40% of the variance in arrival delays, followed by carrier delays. It was also observed that DelayWeatherMinutes may explain 10% of the variation. This suggests that weather has some impact on arrival flights, and in order to determine which weather elements (for example, temperature, precipitation, and pressure) truly influence arrival timings, we excluded DelayWeatherMinutes feature and then integrated the arrival dataset with weather data and performed prediction. The results of this are shown in later sections.

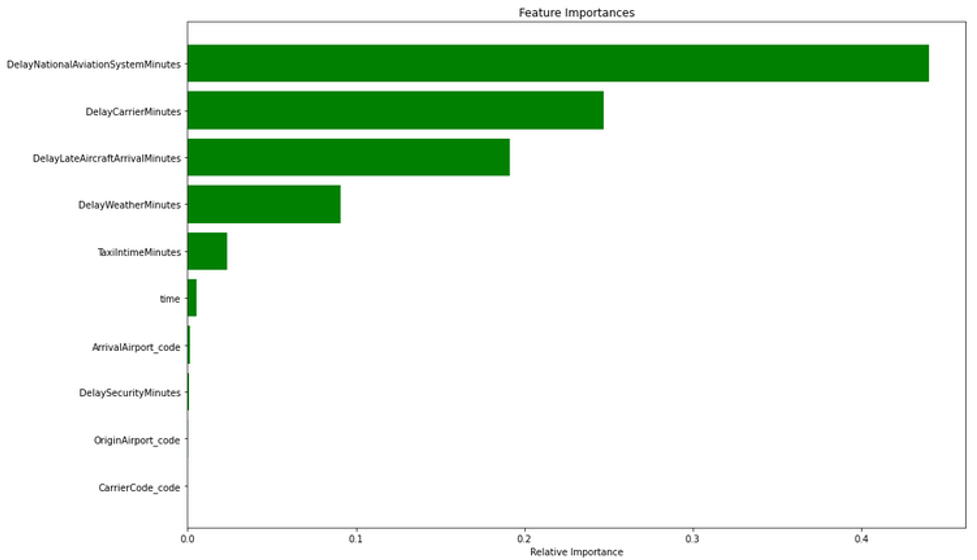


Fig. 7. Percentage of Variance Explained by the Features

(C) Arrival Data With Inclusion Of Weather Parameters

We fit and compared the results of the six different models on the combined dataset (Arrival merged with weather) as given in Table 7 to see if the weather data had any effect on the model’s performance. Despite the fact that the decision tree and XGBoost weighted averages are equal in Table IV, the decision tree was chosen as the final model based on two considerations. One, as shown in Fig. 3, it provided the finest confusion matrices. Second, XGBoost increases the model's complexity, which necessitates a longer run time. According to the confusion matrix, the final decision tree model predicted all actual non-delayed arrival flights as non-delayed and real delayed arrival flights as delayed with an accuracy of 98 percent.

When Table 6 and Table 7 are compared, it is clear that the inclusion of weather parameters has not much improvement on the best model's predictions, but rather increases the total number of false negatives. This means that the current weather characteristics in the dataset have little or no importance on whether the arriving flight will be delayed. To further comprehend this, we used the final decision tree model to build a feature importance plot, as shown in Fig.8.

Table 7. Comparison of The Classification Models on Combined Dataset

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1-score** |
| Naïve Bayes | 100 | 100 | 100 | 100 |
| Decision Tree | 98 | 98 | 98 | 98 |
| Random forest | 95 | 95 | 95 | 95 |
| Adaboost | 93 | 94 | 93 | 93 |
| Gradient Boosting | 93 | 94 | 93 | 93 |
| XGBoost | 98 | 98 | 98 | 98 |

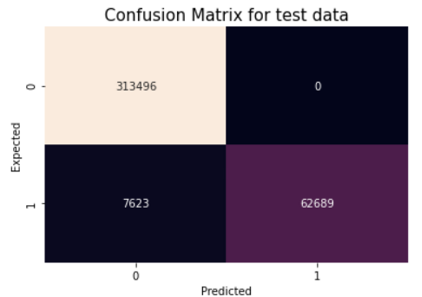


Fig. 8. Confusion matrix of Decision Tree Model Fitted on Combined Dataset

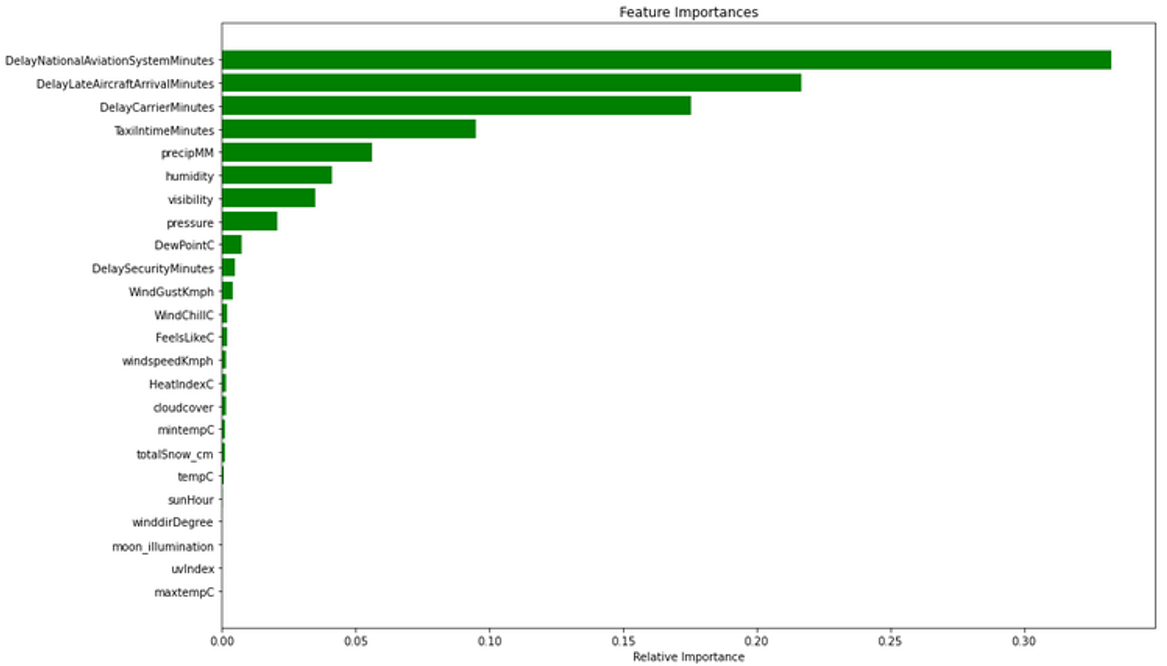


Fig. 9. Percentage of Variance Explained by the Features

According to the feature importance plot in Fig. 4, the parameter DelayNationalAviationSystemMinutes (airport operations and high traffic volume) accounts for about 35% of the variation in arrival delays, followed by DelayLateAircraftArrival, DelayCarrier, and TaxiInTime minutes. Weather characteristics such as precipitation, humidity, visibility, and pressure were found to impact airplane arrival times. Despite contributing 5-10% of the variation, we can claim that these characteristics have some influence on arrival delays.

**B. Regression:**

1. **Regression on Departure Data:**

Unlike classification, In the Regression part we took only the delayed rows using the condition Departure delay >14. This is because the US aviation will consider a flight as delayed only if their delay is more than or equal to 15 mins. We didn't use the datasets we considered for the classification since the target variable was converted as binary. After filtration in our dataset, we did normalization in order to have the same measurement in all variables. For normalization we used standardized scaling options in sklearn as we did in classification. We started our modeling with linear modeling followed by various advanced techniques.

**Linear Regression:**

In linear Regression, We included all the variables we had. We used SPSS for linear Regression. Results were very poor since the model had very accuracy. This is because the variables in the data set were either highly correlated or very poorly correlated with the target variables. Forward selecting, backward elimination were also tried. But the results haven't changed much. I was able to achieve 35% accuracy.

Principal Component Analysis to overcome this issue. SPSS suggests going with 10 components such that those 10 components will account for more than 75% of variability in the data. But the results haven't changed much. I was able to achieve 39% accuracy in the model.

**Extra Tree regression:**

Extra tree regression was tried. We were able to archive 99% accuracy in training data and 95% accuracy in test data. Hyperparameter tuning didn’t make much difference in departure data in extra tree regression. The best parameter for extra tree regression is bootstrap = True, criterion = 'mse' , max\_depth = None, max\_features = 'auto', max\_leaf\_nodes = None, max\_samples = None, min\_samples\_leaf = 1, min\_samples\_split = 2 n\_estimators = 100.

**Random Regression regression:**

Random forest was tried next and I got almost the same accuracy in both training and test set. 'n\_estimators': [5, 50, 100, 200, 400], 'max\_features': ['auto', 'sqrt'], 'max\_depth': [2, 6, 8, 10] was set for random search. The best parameter we got is n\_estimators:100 , max\_features: auto , max\_depth : None.

Table 8: Regression Comparison of Departure Delay

|  |  |
| --- | --- |
| **Type** | **Accuracy** |
| Linear Regression | 35% |
| Linear Regression with PCA | 39% |
| RandomForestRegressor | 95% |
| ExtraTreesRegressor | 95% |

1. **Regression on Arrival Data**

Regression was done to predict the delays in arrival of a flight at a particular airport .We took the weather some of factor in account for developing the models.We filtered the data for the delayed flights alone .So any flight that arrived 15 mins late was considered delayed and we filtered the data ,by extracting the data points for which the delay in minutes was greater than 14 . Models like linear regression ,lasso and elastic were built but none of them gave us a good accuracy ,so we tried ensemble regression models .were developed but the following three are best models we built .Later all these three models were hypertuned to receive the best accuracy .We selected accuracy as the metric to evaluate our model .

**Extra Tree regression:**

This class implements a meta estimator that employs averaging to increase predicted accuracy and control over-fitting by fitting a number of randomized decision trees (a.k.a. extra-trees) on various sub-samples of the dataset. Initially an extra tree regression model was built without hypertuning .The training accuracy of the model was 99.9% and the test accuracy of the model was 99.8% .The R^2 value for training and testing was 99.9% and 99.5%. Later GridSearch was performed using 5 fold cross validation with 100 different combinations. The best parameter set obtained using the GridSearch method was min\_samples\_leaf= 2, min\_samples\_split= 2 and n\_estimators= 5.Later the model was fit using the best parameters and the training accuracy was 99.9% and the testing accuracy was 99.8%.The R^2 for training and test data was 99.9% and 99.6%.

**Random Forest regression:**

A random forest is a meta estimator that employs averaging to increase predicted accuracy and control over-fitting by fitting a number of classification decision trees on various sub-samples of the dataset.The model developed before hypertuning has a training accuracy of 99.8%and the test accuracy was 99.9%.The R^2 for training and test data was 99% and 99.5%.Randomized search was used to hypertune the model with 3 fold cross validation method across 100 different combination .The best model developed using the best parameter results of the Randomized Search had n\_estimators= 200, max\_features=auto and max\_depth= 10.The training accuracy was 98.9% and the testing accuracy was 98.8%.The R^2 for training and test are 98% and 97.6%.

**Gradient Boost Regression:**

The Gradient model developed without the hypertuning had a training accuracy of and test accuracy of 98.6% and 98.7%.The R^2 was 99.6% for training and 99.5% .Gradient Search method with 5 fold cross validation was performed to get the best parameters with we can fit the model .The best parameters received was learning\_rate=1, max\_depth= 2, n\_estimators=200.The training and testing accuracies that we got after hyper tuning were 99.9 %and 99.8% respectively .The R^2 for both training and test data was almost 99.8%.

Table 9:Accuracy Before Hypertuning

|  |  |  |
| --- | --- | --- |
| **Model** | **Training Accuracy** | **Test Accuracy** |
| Extra Tree Regression | 99.9% | 99.8% |
| Random Forest Regression | 99.8% | 99.9% |
| Gradient Boost Regression | 98.6% | 98.7% |

Table 10: Accuracy After Hypertuning

|  |  |  |
| --- | --- | --- |
|  | **Training Accuracy** | **Test Accuracy** |
| Extra Tree Regression | 99.8% | 99.9% |
| Random Forest Regression | 98.9% | 98.8% |
| Gradient Boost Regression | 99.9% | 99.8% |

1. **CONCLUSION AND FUTURE WORK**

In this project, we implemented a machine learning model for predicting the departure and arrival flight delay. We built classification and regression models to predict the departure and arrival delay by using airline data and weather data. Different classification and regression models were applied and the results were compared with and without weather data to determine the influence of weather in departure/arrival delay. It was found that inclusion of weather data did not play a major role in departure and arrival delays. The most significant factors contributed to flight delay are Delay National Aviation System Minutes, Delay Carrier Minutes, Delay Late aircraft arrival minutes while precipitation, visibility and pressure had a minor impact on predicting the delays. Furthermore, regression models gave best accuracy for both arrival and departure data. Overall, we achieved 97% accuracy in predicting the Departure and Arrival flight delay. The overall model of our flow is shown in fig 10.

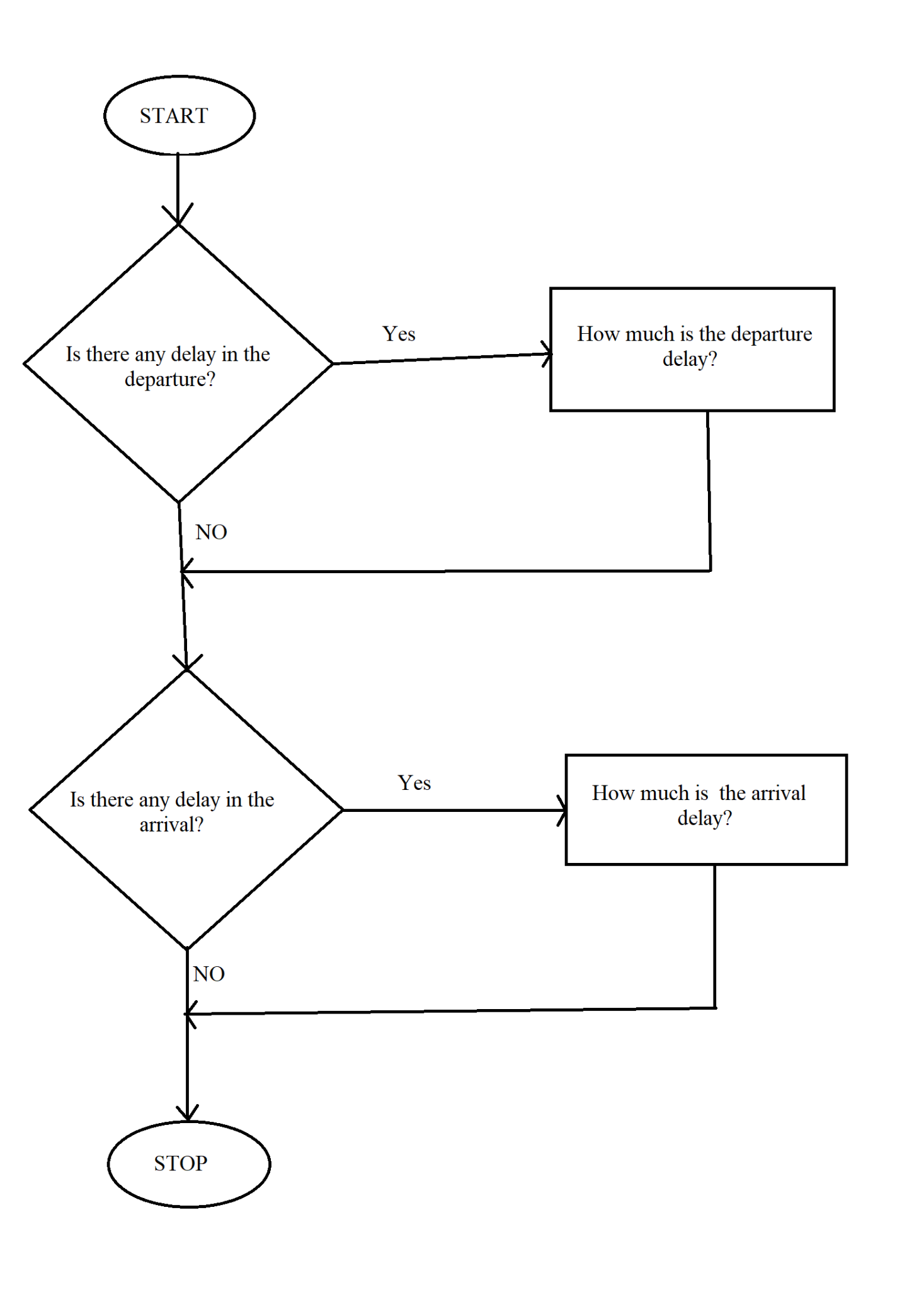


Fig 10. Overall Flow Chart

In the future, Neural networks can be applied in place of regression which improves the predictability of our model and we can combine both departure and arrival data for analyzing and managing the air traffic for specific airports such as the problem around carrier and aviation system minutes since they play a significant role in predicting departure/arrival delays.

1. **AUTHOR CONTRIBUTIONS**

Table 11: The author confirms contribution to the paper.

|  |  |
| --- | --- |
| Topic | Contribution |
| Study Conception and Design | Dharun Selvan, Vallabha Datta, Abinaya Badrinath, Shadhana Palaniswami |
| Data Collection | Dharun Selvan, Vallabha Datta, Abinaya Badrinath, Shadhana Palaniswami |
| Data Preprocessing | Dharun Selvan, Vallabha Datta, Abinaya Badrinath, Shadhana Palaniswami |
| Classification | Vallabha Datta, Abinaya Badrinath |
| Regression Analysis | Dharun Selvan, Shadhana Palaniswami |
| Report Preparation | Dharun Selvan, Vallabha Datta, Abinaya Badrinath, Shadhana Palaniswami |

All authors reviewed results and approved the final version of the report.

1. **REFERENCES:**

[1]. N. L. Kalyani, G. Jeshmitha, B. S. Sai U., M. Samanvitha, J. Mahesh and B. V. Kiranmayee, "Machine Learning Model - based Prediction of Flight Delay," 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2020, pp. 577-581, doi: 10.1109/I-SMAC49090.2020.9243339.

[2].B. Thiagarajan, L. Srinivasan, A. V. Sharma, D. Sreekanthan and V. Vijayaraghavan, "A machine learning approach for prediction of on-time performance of flights," 2017 IEEE/AIAA 36th Digital Avionics Systems Conference (DASC), 2017, pp. 1-6, doi: 10.1109/DASC.2017.8102138.

[3]. S. Choi, Y. J. Kim, S. Briceno and D. Mavris, "Prediction of weather-induced airline delays based on machine learning algorithms," 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), 2016, pp. 1-6, doi: 10.1109/DASC.2016.7777956.

[4]. Y. J. Kim, S. Choi, S. Briceno and D. Mavris, "A deep learning approach to flight delay prediction," 2016 IEEE/AIAA 35th Digital Avionics Systems Conference (DASC), 2016, pp. 1-6, doi: 10.1109/DASC.2016.7778092.

[5]. Arjun Mathur, Aaron Nagao, Kenny Ng. “Predicting flight on-time performance”, http://cs229.stanford.edu/proj2013/MathurNagaoNg-PredictingFlightOnTimePerformance.pdf

[6]. I. B. Mustapha, S. M. Shamsuddin, and S. Hasan, “A Preliminary Study on Learning Challenges in Machine Learning-based Flight Delay Prediction,” *International Journal of Innovative Computing*. [Online]. Available: https://ijic.utm.my/index.php/ijic/article/view/204.

[7]. S. Manna, S. Biswas, R. Kundu, S. Rakshit, P. Gupta and S. Barman, "A statistical approach to predict flight delay using gradient boosted decision tree," 2017 International Conference on Computational Intelligence in Data Science(ICCIDS), 2017, pp. 1-5, doi: 10.1109/ICCIDS.2017.8272656.

[8]. R. Yao, W. Jiandong and X. Tao, "Prediction model and algorithm of flight delay propagation based on integrated consideration of critical flight resources," 2009 ISECS International Colloquium on Computing, Communication, Control, and Management, 2009, pp. 98-102, doi: 10.1109/CCCM.2009.5267970.

[9]. X. Dou, "Flight Arrival Delay Prediction And Analysis Using Ensemble Learning," 2020 IEEE 4th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), 2020, pp. 836-840, doi: 10.1109/ITNEC48623.2020.9084929.

[10] .V. Venkatesh, A. Arya, P. Agarwal, S. Lakshmi and S. Balana, "Iterative machine and deep learning approach for aviation delay prediction," 2017 4th IEEE Uttar Pradesh Section International Conference on Electrical, Computer and Electronics (UPCON), 2017, pp. 562-567, doi: 10.1109/UPCON.2017.8251111.