Predicting delay in flights using Machine learning and Big Data

Mithun Mallick
Lovely Professional University
mithunmallick147852@gmail.com

Sourabh Kumar
Lovely Professional University
sourabh81.sk@gmail.com

[1,2]**School of Computer Science and Engineering, Lovely Professional University, Jalandhar, Punjab, India**

ABSTRACT

Delay of flight has been regarded as one of the toughest difficulties in aviation control. This study analyses data from The U.S. Department of Transportation's (DOT) Bureau of Transportation Statistics (BTS) and proposes a method to model the arriving flights and three different regression algorithms to predict delay, and checking its accuracy with certain parameters. The result testing shows that this method is convenient for calculation, and also can predict the flight delays effectively. It can provide a decision basis for airport authorities. The model aims to help the authorities to predict the flight delay in order to save as much as time of the customers and also the staffs. This model increases the accuracy of the prediction and also overcomes the limitation of large data processing.

Keywords: Flight delay prediction Exploratory data analysis Regression models

1. Introduction

With the rapid development of the financial system, the demand for all types of transport has increased dramatically. The flight delay caused by commercial flights of the aviation industry has become more and more serious, which directly causes serious damage to the image of civil aviation services. For passengers, flight delay caused the inconvenience of travel, bad mood because of the double loss of their valuable time and economy; for the airport, the delay of the flight seriously affects the traditional operation of the airport; for airline, frequent flight delay not only bring huge economic losses to the airline, but also affect the reputation of the airline. Flight delay has become the shackles of the events leading to the development of the aviation industry[24]. Today, flight delay has not only become an issue for the bulk of travellers, but also the world's civil aviation industry problems. Flight delays were caused by many reasons. The biggest factor is that the capacity of the airport and airspace is insufficient [4]. For other reasons, like the weather, the airport scheduling, the corporate plan, passengers and luggage etc. also cause flight delay. There is a sequence of reaction of the flight delay. When flight delay occurs, if the plan is compact, it'll affect the take-off or landing of the next flight on time, thereby indirectly affecting more downstream flights and airports[3]. If we could timely forecast the arrival flight delay, we will take necessary measures to scale back the economic and credit losses thanks to it. Obviously, it's important and necessary to predict the arrival flight delay in real time.

This project is developed with reference to the course Big Data. Its purpose is to fulfil the given condition that when a big amount of data is received, the underlying model will be able to correctly predict the arrival delay of commercial flights. To do so we are going to develop three different algorithms in a Spark application in Scala.

For our study we chose to work with the dataset of the year 2018 provided by the US government. In this paper we established a linear regression model using Departure Delay and Taxing Out to predict arrival delay, and presented the design and implementation of a flight delay prediction system. The rest of this paper is organized as follows: Data used in this study are presented in Sec. 2. Problem definition is presented in Sec. 3. In sec. 4, we provide information about the exploratory data analysis being performed on the available dataset. The method, implementation and techniques of the system are described in Sec. 5. Our detailed Experimental results are presented in sec. 6. Conclusion and future work are presented in Sec. 7.

2. Literature Review

Delay is one of the most important performance indicators in any travel plan. Significantly, commercial flying players understand delays as a time when the plane arrives late or postponed. Therefore, the delay can be represented by the difference between the planned and actual times of departure or arrival of the plane [21]. State regulatory authorities have a number of indicators related to air traffic control delays. Indeed, flight delays are an important issue in the context of air travel systems. In 2013, 36% of flights were delayed by more than five minutes in Europe, 31.1% flights were delayed by more than 15 minutes in the United States, and 16.3% of flights were cancelled or suffered a delay of more than 30 minutes in Brazil [8,2]. This shows how relevant this indicator is and how it does not affect it does matter the size of the aircraft fabrics.

Flight delays have a negative impact on the economy, for passengers, airports, and airport authorities. Given the uncertainty by their very nature, passengers often plan to travel several hours before their appointment, extending their journey costs, ensuring their timely arrival [4,10]. On the other hand, fines and additional performance costs, such as staff and remembered flight attendants at airports [5,20,9,11]. In addition, from the sustainability area of vision, delays can also cause environmental damage by increasing fuel consumption and emissions [16,1,17,13,3,22].

Delays also impact air marketing strategies, as companies rely on customer loyalty to support their common fly plans and consumer preferences are also reflected in reliable performance. There is an identified relationship between rates of delays and fares, aircraft sizes, aircraft frequency and air service complaints [7,14,6,15,23]. Flight delay estimates can improve strategic and operational decisions for airports and airports managers and warn passengers so they can reschedule their plans.

Commercial flights are a complex transportation system. It works with essential resources, a complex matrix from the destination that requires orchestration to provide smooth operation and safety. In addition, each passenger follows his or her own route while flights plan different flight schedules, pilots and flight attendants. Figure 1 shows the normal operation of a commercial aircraft. Stages can happen in the end borders, airports, runways, and airports, which are found in various types of delays. Other examples include equipment problems, weather conditions, ground delays, flight control, flight lines and power issues [18,12,1].

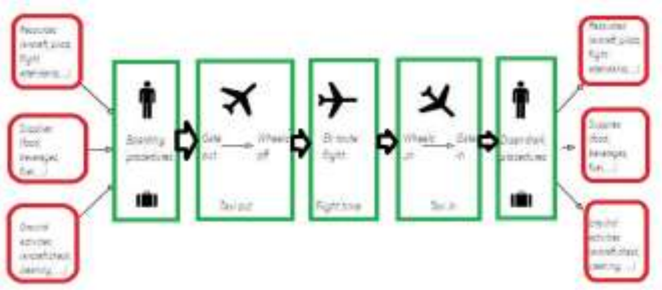

Figure 1: A typical operation of a commercial flight

2.1. Problems to address in a flight delay prediction model

Serious problems related to flight delay forecasts are identified and taxed. Including scales, models, and troubleshooting methods for predicting flight delays. It considers flight domain features, as a problem scope, and data Science concepts, such as data and methods. Fig 2 shows all the tax revenue for the next period The clauses define each tax component and related activity.

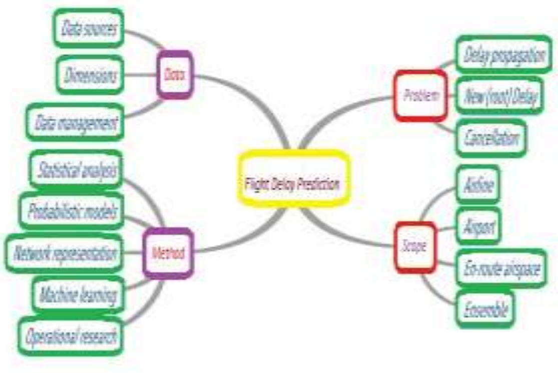

Figure 2: Taxonomy of the flight delay prediction problem

2.2. Overview

Since flight delays create economic consequences for passengers and planes, treating them correctly is essential to improve marketing decisions. As a result, several weather models have been developed over the past two decades. These models want to understand how delays are spread across a network of airports or airports, predicting root delays in the process or understanding of the cancellation process. Apart from these three points to treat flight delays and predictive problems, models can also vary in their application scope, data issues and methods.

3. Case study

In this study, domestic flights data between Jan 2008 and Dec 2008 are used as a case study. Note that the extracted dataset does not contain the international flights which have a relatively low delay ratio. The collected dataset, which is owned by The U.S. Department of Transportation's (DOT) Bureau of Transportation Statistics (BTS) , contains a combined data of flights, origin of flight, destination and distance, etc.

4. Data source

The results presented in this paper were obtained using data from http://statcomputing.org/dataexpo/2009/the-data.html. We had access to a large dataset with 1048576 entries of data of commercial flights in the year 2008. The dataset provides information about flights by date of flight, the airport the flight will take off from, flight number, distance, departure, and arrival time. There are 29 fields in the dataset, and the main fields are shown in Table 1.

Table 1. Fields and description of dataset.

No.	Forbidden	Name	Description
1		Year	2008
2		Month	1-12
3		DayofMonth	1-31
4		DayOfWeek	1 (Monday) - 7 (Sunday)
5		DepTime	actual departure time (local, hhmm)
6		CRSDepTime	scheduled departure time (local, hhmm)
7	x	ArrTime	actual arrival time (local, hhmm)
8		CRSArrTime	scheduled arrival time (local, hhmm)
9		UniqueCarrier	unique carrier code
10		FlightNum	flight number
11		TailNum	plane tail number

12	x	ActualElapsedTime	in minutes
13		CRSElapsedTime	in minutes
14	x	AirTime	in minutes
15		ArrDelay	arrival delay, in minutes
16		DepDelay	departure delay, in minutes
17		Origin	origin IATA airport code
18		Dest	destination IATA airport code
19		Distance	in miles
20	x	TaxiIn	taxi in time, in minutes
21		TaxiOut	taxi out time in minutes
22		Cancelled	Was the flight cancelled?
23		CancellationCode	reason for cancellation (A = carrier, B = weather, C = NAS, D = security)
24	x	Diverted	1 = yes, 0 = no
25	x	CarrierDelay	in minutes
26	x	WeatherDelay	in minutes
27	x	NASDelay	in minutes
28	x	SecurityDelay	in minutes
29	x	LateAircraftDelay	in minutes

The forbidden column in the table represents those variables from the dataset that can't be included since their values are only known once the plane has already taken off. Hence, if we use the forbidden variables, all the algorithms would return unreliable results. The forbidden columns include ArrTime, ActualElapsedTime, AirTime, TaxiIn, Diverted, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay and LateAircraftDelay.

5. Methodology

In this section, we review the dataset available to us and after proper cleaning of the data we run different analytical methods to find the insights of the data like the various variables responsible for the arrival delay of flights and after the exploratory analysis of the available data, we make the data run across certain predictive analysis methods like Linear Regression model and try to predict the flight arrival delay with the highest precision and accuracy and conclude which regression model among the three models available produces the better result.

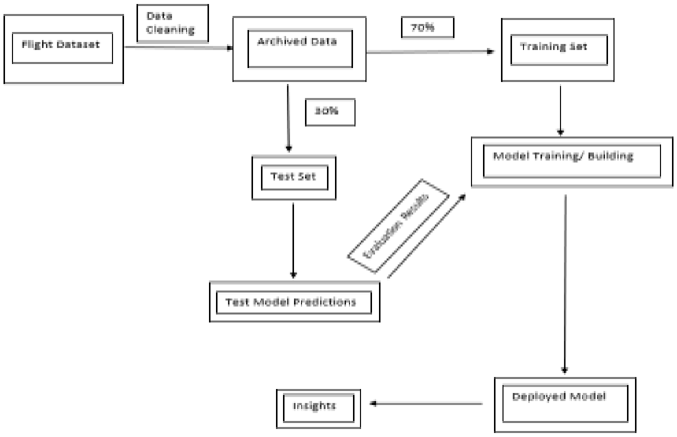

Figure 3. Schematic representation of the model

5.1. Problem definition

The decisions taken within the management of an airport are often not subtle or discernible and influence several variables, like flight delay. Reducing this delay presents the advantage of decreasing costs and increasing the standard of the service provided to the passengers. So, it is important to search out which variables influence flight delay and use them to predict it. A number of them treat flight delay prediction as a regression problem, predicting the delay by the minute, whereas others treat them as a classification problem, predicting a quantity where the delay will fall[2]. The problem considered in this paper is to predict the correlated dependency of the various factors with flight arrival delay . So, in this paper, the prediction mechanisms adapted to accomplish the desired result is : regression. For better understanding and prediction three different regression models: Linear , Random Forest Trees and Gradient-Boosted Trees were used.

5.2. Exploratory Data Analysis

As described in the previous section, our focus is to predict the value of ArrDelay (Arrival Delay). In order to perform the required study properly, there are some variables from the dataset that can't be included (forbidden variables) since their values are only known once the plane has already taken off. If we used the forbidden variables, all the algorithms would return unreliable results. For this reason, all the columns ArrTime, ActualElapsedTime, AirTime, TaxiIn, Diverted, CarrierDelay, WeatherDelay, NASDelay, SecurityDelay and LateAircraftDelay were dropped.

Once the available dataset was properly cleaned, we decided to perform some exploratory data analysis not only to understand our data, but also to try to gather as many insights before starting working with it. All the code used for this is in the parser eda.py.

In order to use Linear Regression, it's necessary to remove correlated variables to improve our model. For this reason, we did a correlation matrix with all the numerical variables of the dataset.

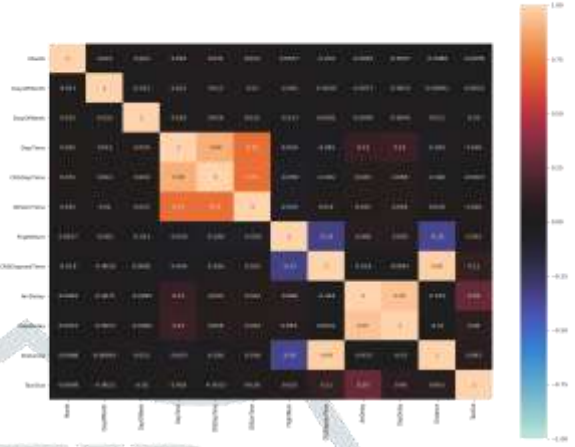

Figure 4. Correlation matrix of all numerical variables of 2008.csv

As we can see from Fig. 4, light shades represent positive correlation while darker shades represent negative correlation. Besides the colours, we also used the values of each correlation to get a better idea of the relation between the variables.

Since our target variable is ArrDelay, we are going to take especially attention to the relations with that variable. We can clearly see that the strongest correlations are with DepDelay (0.95) and TaxiOut (0.29). It also shows some, but much less, correlation with DepTime (0.13), CRSDepTime (0.043) and CRSArrTime (0.043).

To get a better understanding of these features we draw scatter plots between our target variable and all the variables that it showed some correlation with.

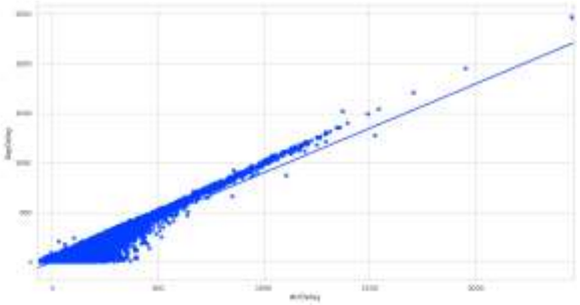

Figure 5. Scatterplot between DepDelay and ArrDelay

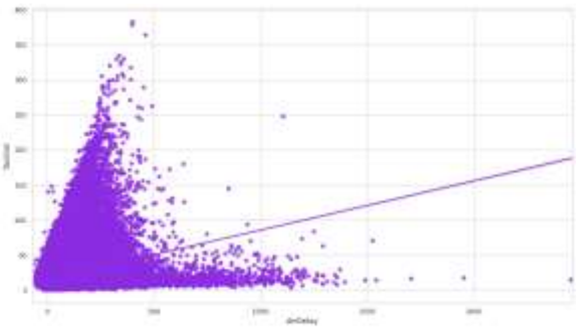

Figure 6. Scatterplot between TaxiOut and ArrDelay

As we can conclude from the images above, in Fig. 5 there's a much stronger correlation than in Fig. 6. We chose to run our model once with these variables since these are the ones with the strongest correlation, therefore, should provide the best results.

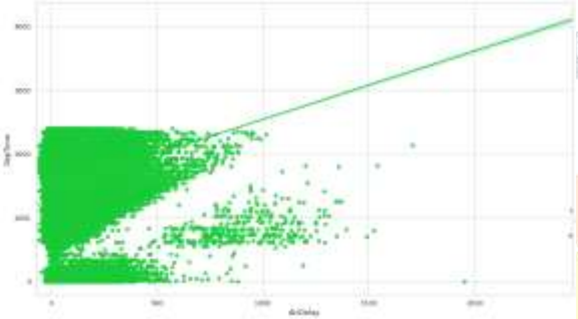

Figure 7. Scatterplot between DepTime and ArrDelay

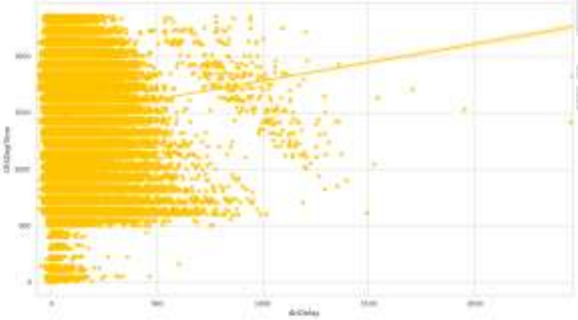

Figure 8. Scatterplot between CRSDepTime and ArrDelay

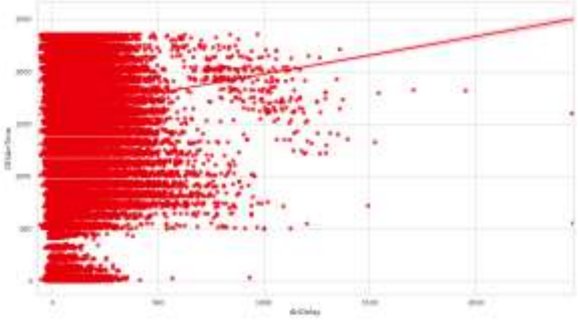

Figure 9. Scatterplot between CRSArrTime and ArrDelay

From the images above, we can also see a positive correlation, yet not so strong if we compare with Fig. 5 and Fig. 6. For this reason, we also thought it would be a good idea to try out our model with these 5 variables plus the categorical variables and check if the results would be better.

5.3. Machine Learning Techniques

Regression algorithms are machine learning techniques for predicting and estimating the relationship between variables. Whereas classification models identify which category an observation belongs to, regression models will return a numeric value. Therefore, for the purpose of this project, to predict our continuous numerical target variable ArrDelay, we need to use regression algorithms.

Once the exploratory data analysis phase was completed, we were able to conclude that the dependent variable showed linear relations with independent variables of the data-set. Hence, a suitable model to our needs is Linear Regression.

Although linear regression presented great results we decided to test other machine learning algorithms, such as Random Forest Trees and Gradient-Boosted Trees.

Random Forest Trees and Gradient-Boosted Trees are techniques that are able to discover more complex dependencies at the cost of more time for fitting and therefore were worth being put to the test.

5.4. Loading and processing the data

The provided data is a .csv file and is being loaded by providing the full path of the file from the local data storage. Then the nominal features need to be transformed into something that the model can use by converting them into a numerical representation. A StringIndexer is used to map a string column of labels into a vector of numbers of length of the total unique string in that column and we apply that to every categorical feature. Then OneHotEncoder is used to wrap all unique values in a single vector in a new column of the dataset. Finally, we use a VectorAssembler, to assemble all the features into one vector called "features" to be used by the machine learning algorithms.

5.5. Creating and validating the model

As mentioned in 5.1, our application supports three different machine learning techniques for creating a model. The approach is very similar but some considerations were taken. All the data processing transformations as well as the machine learning training method is put into a pipeline combining all the steps into one transformation and making it easy to repeat all the transformations on new data.

Parameter optimization for building a model using linear regression, we perform a grid search to optimize the parameters of training, specifically we try different variations of elastic net regularization.

Since we are running on a single machine with limited resources, we only used grid search for parameter optimization in the linear regression method since it was not possible to do the same for the other two algorithms in feasible time.

Training and validating from the initial exploratory data analysis we created 5 feature vectors to train our model. The feature vectors are as follows:

{ x1: [DepDelay, TaxiOut, UniqueCarrier, DepTime, CRSArrTime, CRSElapsedTime, Distance, FlightNum, CRSDepTime, Year, Month, DayofMonth, DayOfWeek, TailNum, Origin, Dest]}

{ x2: [DepDelay, TaxiOut, Distance, FlightNum, CRSDepTime, Year, Month, DayofMonth, DayOfWeek, TailNum, Origin, Dest]}

{ x3: [DepDelay, TaxiOut, TailNum, Origin, Dest]}

{ x4: [DepDelay, TaxiOut, Origin, Dest]}

{ x5: [DepDelay, TaxiOut]}

The final version of the application only considers x4 and x5 since these two seem to provide the essential information for a good prediction and allow to run the three machine learning algorithms in a single machine for a full year worth of data. For the purpose of testing, we used the five vectors in linear regression since it is a fast algorithm and to get a grasp on the insights of using each vector. Then we drilled down to more useful features according to our exploratory data analysis.

Vector x1, represents all the features except the forbidden ones. Vector x2, encompasses vector x1 minus features that are highly correlated between themselves and are therefore redundant, such as, DepTime and CRSArrTime which are highly correlated to CRSDeptime and CRSElapsedTime which is highly correlated to distance. Vector x3 includes vector x2 minus uncorrelated variables to the target, arrDelay. Vector x4 includes x3 minus TailNum. Vector x5 removes the categorical variables and is composed only of the variables that are highly correlated to the target. Before training we split the data into a training and test set, 70% and 30% respectively.

For the linear regression method, since we specified several different variations of the model, a regression evaluator as well as a train test split were implemented to train the several models, with different sets of parameters in the specified grid, and evaluate them. To do so we used Spark's TrainValidationSplit class. Then applied the fit function using the train dataset to search the potential model space and returning the best performing model. In the case of Random Forest and Gradient-Boosted Trees we train directly the pipeline since it would take too much time and space to run and evaluate in all the possible configurations of these methods and therefore, we use the default ones.

Finally, we use the model to transform the test dataset and use a RegressionMetrics class to obtain useful information regarding the accuracy of the final model.

5.6. Implementation Environment

In order to predict flight delay, we develop a system. The system includes the predictor who is responsible for training, predicting and testing and the system architecture. The system was developed by Scala language, the development and implementation environment are shown in Table 2.

Table 2. The development and implementation environment.

Software Environment

Development Language	**Scala**
Development Tool	**Spark**
Operating System	**Ubuntu 18.04.3**

Hardware Environment

Processor	**Intel(R)Core (TM) i5-7200U CPU @2.5GHz 2.7 GHz**
Memory	**8.00 GB**
System Type	**64-bit**

6. Experimental Results

To analyse and compare the results of our experiments we analysed three metrics, the Root Mean Square Error (RMSE), Mean Absolute Error and R squared. The results for the three machine learning algorithms are as follows.

Table 3. Regression metrics for linear regression method.

Features Vector	X1	X2	X3	X4	X5
RMSE	9.7967	9.8506	9.8583	9.9618	10.8789
MAE	6.8871	6.8917	6.9304	7.0254	7.7834
RSquared	0.9351	0.9347	0.9344	0.9330	0.9201

Table 4. Regression metrics for random forests method.

Features Vector	X4	X5
RMSE	18.2667	18.3545
MAE	9.6023	9.3950
RSquared	0.7747	0.7726

Table 5. Regression metrics for gradient-boosted trees method.

Features Vector	X4	X5
RMSE	17.8073	17.9833
MAE	9.2140	9.3390
RSquared	0.7859	0.7817

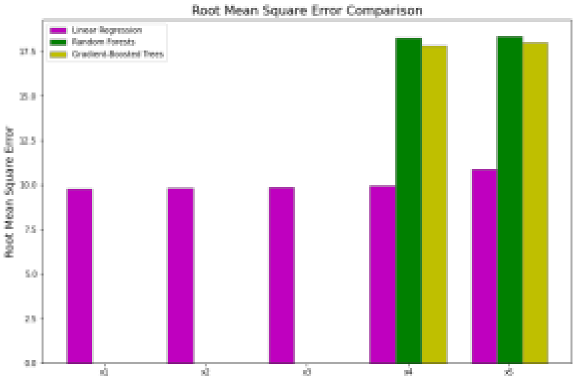

Figure 10. Root Mean Squared Error Comparison

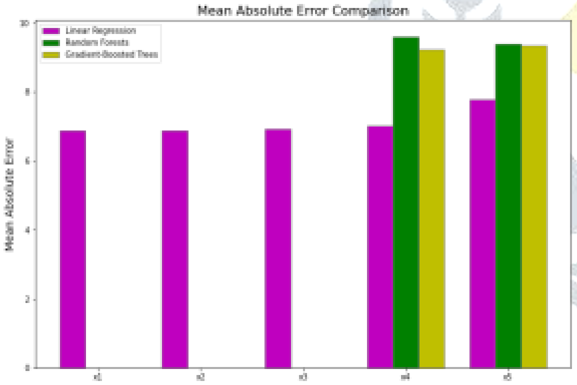

Figure 11. Mean Absolute Error Comparison

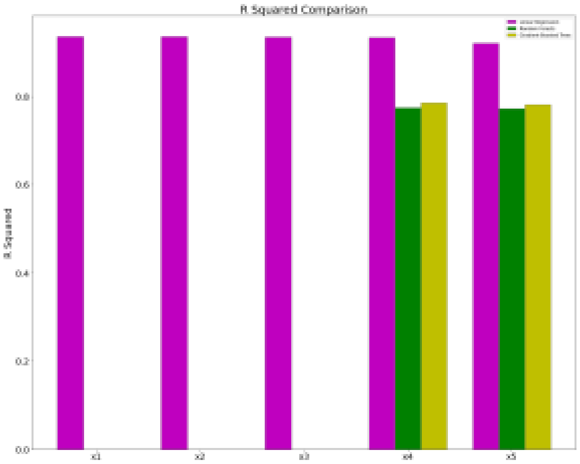

Figure 12. R Squared Comparison

7. Conclusion and Future Work

Before taking any solid conclusions, we will first analyze the metrics we used to evaluate the algorithms.

1. Root Mean Square Error(RMSE): It measures the difference between values predicted by a model and the values observed. Considering this, the lower RMSE value is, the better is the model.

Taking a look at the RMSE Comparison between the three models, Fig. 10, we can plainly see that Linear Regression provided low values with all the five vectors used. Besides, by comparing it with the values of Random Forests and Gradient-Boosted Trees in the vectors x4 and x5, Linear Regression provided the lowest in both scenarios.

2. Mean Absolute Error(MAE): It refers to the mean of the absolute values of each prediction error on all instances of the test data-set, where prediction error is the difference between the actual value and the predicted value for that instance. Once again, the lower the MAE value is, the better is the model.

By examining the Mean Absolute Error Comparison, Fig. 11, we are able to verify that Linear Regression returned low values with all the vectors whereas Random Forests and Gradient-Boosted Trees MAE are round 9, Linear Regression has a MAE of round 6.

3. R Squared: It is also known as the coefficient of determination, it is a statistical measure of how close the data is to the fitted regression line. It assesses the goodness of fit of our regression model. The closer the value is to 1, the better is the model.

Considering R Squared Comparison, Fig. 12, Linear Regression produced values between the 0.9 and 0.95 to all of the five vectors studied. On the other hand, both Random Forests and Gradient-Boosted Trees returned values between 0.75 and 0.8. Although these results are not bad, the former are better.

All in all, after comparing the values of each of the used metrics in the three algorithms, we can conclude that the model which proved to provide the best predictions was Linear Regression.

References

[1] S. AhmadBeygi, A. Cohn, Y. Guan, and P. Belobaba. Analysis of the potential for delay propagation in passenger airline networks. Journal of Air Transport Management, 14(5):221–236, Sept. 2008. ISSN 0969-6997.

[2] ANAC. Agˆencia Nacional de Aviac¸ ˜ao Civil. Technical report, http://www.anac.gov.br/, 2017.

[3] E. Balaban, I. Roychoudhury, L. Spirkovska, S. Sankararaman, C. Kulkarni, and T. Arnon. Dynamic routing of aircraft in the presence of adverse weather using a POMDP framework. In 17th AIAA Aviation Technology, Integration, and Operations Conference, 2017, 2017.

[4] P. Balakrishna, R. Ganesan, and L. Sherry. Accuracy of reinforcement learning algorithms for predicting aircraft taxi-out times: A case-study of Tampa Bay departures. Transportation Research Part C: Emerging Technologies,

18(6):950–962, Dec. 2010. ISSN 0968-090X.

[5] R. Britto, M. Dresner, and A. Voltes. The impact of flight delays on passenger demand and societal welfare. Transportation Research Part E: Logistics and Transportation Review, 48(2):460–469, Mar. 2012. ISSN 1366-

5545.

[6] D. Bhadra. You (expect to) get what you pay for: A system approach to delay, fare, and complaints. Transportation

Research Part A: Policy and Practice, 43(9):829–843, Nov. 2009. ISSN 0965-8564.

[7] J. I. Daniel and K. T. Harback. (When) Do hub airlines internalize their self-imposed congestion delays?

Journal of Urban Economics, 63(2):583–612, Mar. 2008. ISSN 0094-1190.

[8] EUROCONTROL. CODA Digest - Delays to Air Transport in Europe. Technical report,

https://www.eurocontrol.int/articles/coda-publications, 2017.

[9] J. Evans, S. Allan, and M. Robinson. Quantifying delay reduction benefits for aviation convective weather decision support systems. In Conference on Aviation, Range, and Aerospace Meteorology, pages 39–70, 2004.

[10] P. Fleurquin, B. Campanelli, V. Eguiluz, and J. Ramasco. Trees of reactionary delay: Addressing the dynamical

[11] C.-Y. Hsiao and M. Hansen. Air transportation network flows: Equilibrium model. Transportation Research Record, (1915):12–19, 2005. robustness of the US air transportation network. In SIDs 2014 - Proceedings of the SESAR Innovation Days, 2014.

[12] G. Hunter, B. Boisvert, and K. Ramamoorthy. Advanced national airspace tra_c flow management simulation experiments and vlidation. In 2007 Winter Simulation Conference, pages 1261–1267, Dec. 2007.

[13] T. Krsti´c Simi´c and O. Babi´c. Airport tra_c complexity and environment e_ciency metrics for evaluation of ATM measures. Journal of Air Transport Management,

42(Supplement C):260–271, Jan. 2015. ISSN 0969- 6997.

[14] B. Manley and L. Sherry. Impact of ground delay program rationing rules on passenger and airline equity.

In IMETI 2008 - International Multi-Conference on Engineering and Technological Innovation, Proceedings, volume 1, pages 325–330, 2008.

[15] V. Pai. On the factors that a_ect airline flight frequency and aircraft size. Journal of Air Transport Management, 16(4):169–177, July 2010. ISSN 0969-6997.

[16] T. Pejovic, R. B. Noland, V. Williams, and R. Toumi. A tentative analysis of the impacts of an airport closure. Journal of Air Transport Management, 15(5):241–248, Sept. 2009. ISSN 0969-6997.

[17] J. J. Rebollo and H. Balakrishnan. Characterization and prediction of air tra_c delays. Transportation Research Part C: Emerging Technologies, 44(Supplement C):231–241, July 2014. ISSN 0968-090X.

[18] A. J. Reynolds-Feighan and K. J. Button. An assessment of the capacity and congestion levels at European airports. Journal of Air Transport Management, 5(3):113–134, July 1999. ISSN 0969-6997.

[19] M. S. Ryerson, M. Hansen, and J. Bonn. Time to burn: Flight delay, terminal e_ciency, and fuel consumption

in the National Airspace System. Transportation Research Part A: Policy and Practice, 69(Supplement C): 286–298, Nov. 2014. ISSN 0965-8564.

[20] Y. Tu, M. O. Ball, and W. S. Jank. Estimating flight departure delay distributions—a statistical approach with long-term trend and short-term pattern. Journal of the American Statistical Association, 103(481):112–125, 2008.

[21] F. Wieland. Limits to growth: results from the detailed policy assessment tool [air tra_c congestion]. In 16th

DASC. AIAA/IEEE Digital Avionics Systems Conference. Reflections to the Future. Proceedings, volume 2, pages 9.2–1–9.2–8 vol.2, Oct. 1997.

[22] Y. Xu, R. Dalmau, and X. Prats. Maximizing airborne delay at no extra fuel cost by means of linear holding. Transportation Research Part C: Emerging Technologies, 81:137–152, 2017.

[23] B. Zou and M. Hansen. Flight delay impact on airfare and flight frequency: A comprehensive assessment. Transportation Research Part E: Logistics and Transportation Review, 69(0):54 – 74, 2014. ISSN 1366-5545.

[24] Ding, Yi. "Predicting flight delay based on multiple linear regression." *IOP conference series: Earth and environmental science*. Vol. 81. No. 1. IOP Publishing, 2017.