**An Approach to Accurately Predicting Flight Delays**

Bryan Meriel  
*Humber College*  
*Group 2   
Toronto, Ontario*  
N01496545

Mariana Betancourt  
*Humber College*  
*Group 2  
Toronto, Ontario*  
N01498484

Sandra Maldonado  
*Humber College*  
*Group 2  
Toronto, Ontario*  
N01468706Steven Kinnunen  
*Humber College*  
*Group 2  
Toronto, Ontario*  
N01501178

*Abstract*— *As a result of growth in the aviation industry, air-traffic congestion has caused flight delays, which have negative impacts on the economy and the airline industry. Delays in air transportation are not only detrimental to the economy, but also to the environment. This has caused management of air traffic to become more and more difficult. In this project we apply machine learning algorithms like Naïve Bayes, Classification Tree, and Logistic Regression to predict if a given flight’s arrival will be delayed or not. Both the unbalanced logistic regression and unbalanced classification tree had the best overall accuracy. However, decision tree interpretability makes it a more attractive model.*

***Keywords—Naïve Bayes, Logistic Regression, Classification Tree, Ontime, Delayed, Confusion Matrix, Accuracy***

# INTRODUCTION

There has been a noticeable increase in the popularity of air travel, primarily due to its speed and, in some cases, its comfort. There has also been a dramatic increase in aircraft delays on the ground and in the air as a result of growing air traffic.[5] Consequently, the environment and the economy suffer significant losses [5]. Due to uncertainties, human factors, and competing stakeholder interests, aircraft delays are inevitable. As a result of the networked nature of the air transportation system, delays propagate from one end of the system to the other. Balakrishnan argues that new analysis techniques and operational strategies are needed to prevent cascading delays and even collapses of air transportation systems [2].

In the design of algorithms for air transportation, as in the case of most other real-world infrastructures, a wide range of multi-objective optimization problems need to be addressed [2].  It is our hope that by using Naïve Bayes, Classification and Regression Trees, as well as Logistic Regression models, we will improve the efficiency and robustness of the air transportation system while still preserving its safety and security [2]. For the purpose of our study, we will be analyzing a Flight Delays dataset containing details on the following factors: flight date, departure time, distance between airports, flight status, the flight carrier, etc. Using the above-mentioned algorithms and dataset, we aim to predict if a flight will arrive on time or delayed. In order for a flight to be considered delayed, it must arrive at least 15 minutes later than scheduled. Furthermore, we will compare all three algorithms and determine which algorithm we recommend to be used for future prediction.

# LITERATURE REVIEW

Predicting delays for airlines can be a complicated problem that requires consideration for varying factors such as day of the week, weather and airline, amongst others. Moreover, established literature on the topic has pointed out that this issue often involves imbalanced data. Delayed records tend to occur with less frequency than on time records and this can impact the accuracy of the resulting classification model, particularly in respect to predicting delayed records [3][4].

In addition to considering the class imbalance issue, there are a number of algorithms that may be used in classification problems that may predict the status of arriving flights. Notably, random forests and perceptron algorithms have been applied on balanced data to improve the accuracy of predictions [3]. Yazdi et al propose that traditional approaches to predicting flight status have been less accurate due to the sheer volume of the data in conjunction with the number of features. Therefore, they propose and utilize a deep learning model to predict flight status [11]. Tang, however, used more traditional methods, such as logistic regression, naive bayes, decision tree, random forest, support vector machine, gradient boosted tree and k-nearest neighbours. The decision tree model produced the best result [9].

# METHODS

To find the most accurate predicting flight delays model, we analyze the following three supervised machine learning algorithms: Naïve Bayes, Classification and Regression Tree, and Logistic Regression. To build and measure those models our methodology included data preprocessing, model building, and predicting new values with the models.

3.1. Methodology

3.1.1. Data preprocessing

The dataset was analyzed and cleaned by using data reduction and dimension reduction.

On our dimension reduction analysis and based on our research, we have determined TAIL\_NUM, corresponding to the tail number, can be dropped since it is merely a unique identifier and not a meaningful category or numeric value that measures a quantity. Similarly, flight numbers (FL\_NUM) are unique identifiers for specific trips and do not represent meaningful categories or measurable quantities. CRS departure times (CRS\_DEP\_TIME) appear to correspond to scheduled rather than actual departure times. Since there is another column with actual departure times (DEP\_TIME), this column is redundant and was also removed. Moreover, the flight date (FL\_DATE) column is redundant since all records take place in the same month. Since a day of the month column already exists, we dropped the date column.[1][7]

Converting numerical variables to categorical variables, we would convert the departure time to a DateTime format. However, we intended to bin all integer values to ensure that the Naïve Bayes algorithm will work. Therefore, this would be redundant. With respect to the Distance column, since there are only 7 unique values, we can simply convert the integers to categorical values. However, the departure times are too varied to retain each unique value as categorical so we will bin the values. We will additionally bin DAY\_OF\_MONTH as well. Then, the original columns were deleted.   The reference table for our categorical data is available on Table 1: Reference Table for Numerical to Categorical Values.

Table 1: Reference Table for Numerical to Categorical Values

|  |  |  |
| --- | --- | --- |
| **Variable** | **Numerical value** | **Categorical value given** |
| DEPT\_TIME  Rename as Departure\_Time\_  Interval | 0 to 559 | Early Morning |
| 600 to 1159 | Morning |
| 1200 to 1759 | Afternoon |
| 1800 to 2400 | Evening |
| DAY\_OF\_MONTH  Rename as  Days\_of\_month | 1 - 7 | 1 to 7 |
| 8 - 14 | 8 to 14 |
| 15 - 21 | 15 to 21 |
| 22 - 31 | 22 onwards |

In addition, to ensure the correct performance of the categorical models that we will use, we create dummy variables for those that require them.

After analyzing, cleaning, categorizing, and creating dummy variables we split the dataset (60% training data, 40% validation data), then we train the models.

Finally, our analysis discovered that there is a class imbalance amongst the target variable, flight status. There are significantly more “on time” records than “delayed” records. To help deal with this issue, we have created a secondary dataset that oversamples from the “delayed” records to create more balanced classes. This dataset will be used to run another iteration of each model to assess whether a balanced dataset outperforms the original dataset.

3.1.2. Model building

By using a training set of 2126 commercial flights that departed from the Washington, DC area and arrived in New York in the year 2004. The outcome variable is Flight Status, with ‘on time’ defined as the success and ‘delayed’ defined as the failure. This would be equivalent to setting the outcome variables to 1 for ‘on time’ and 0 for ‘delayed. We built the following models:

**Naïve Bayes**

The first model we built was the Naïve Bayes (NB) model. This is a classification model based on probability. This model works with categorical data type, and it uses the concept of conditional probability to classify a record [8].  After training the model and validating in the training and validation dataset respectively, we evaluate the accuracy of the model by using a confusion matrix.

**Classification Tree**

Second in our methodology is the decision tree classifier. As we are required to predict the categorical outcomes of future observations, the model will first learn from the patterns and outcomes in the training dataset. Thereafter, the model will be validated using another set of data. Lastly, it will come across five instances of new data and in which the model will attempt to accurately classify. In our project we have decided to run two iterations of our models using balanced and unbalanced data to measure the difference of accuracy.

**Logistic Regression Model**

Last in our methodology is the Logistic Regression Model. After preprocessing the dataset, we ensured the data is also categorical. The Logistic Regression algorithm does not comply with numerical data. Having multiple predictor variables, we converted them all into sets of dummy variables followed by partitioning the data into training and validation datasets, and later on, creating a 5-record testing dataset. By this point, the logistic regression model is able to be fitted to all datasets, and we are now able to evaluate its classification performance by constructing a confusion matrix.

# RESULTS AND DISCUSSION

**Naïve Bayes**

Table 2: NB - Summary Confusion Matrix Results – Validation Dataset

|  |  |  |
| --- | --- | --- |
| **Measure** | **Balanced**  **Data Model** | **Unbalanced**  **Data Model** |
| **Error** | 33.85% | 19.89% |
| **Accuracy** | 66.15% | 80.11% |
| **Specificity** | 68.87% | 24.57% |
| **Sensitivity** | 63.42% | 93.90% |

After running two iterations of our model, one for the denominated unbalanced data and the second one for the balanced data. The performance of the NB model in predicting whether a flight is on-time or delayed is represented in accuracy by using a confusion matrix. According to the model’s accuracy and based on the results shown in table 2, comparing the validation dataset for the unbalanced and balanced data, the former gave us the highest value which was around 80% while the second one has an accuracy of roughly 64%. In terms of sensitivity, the model shows the highest sensitivity value in the unbalanced dataset, more than 93%, which means that the model has a great ability to predict true on-time flights. However, the specificity value is low, slightly under 25%.

Related to the balanced data, the model shows poor performance for both the training and validation dataset; the measures of accuracy, sensitivity, and specificity were between 62% to 66%.

**Classification Tree**

Table 3: CART - Summary Confusion Matrix Results - Validation Dataset (Pruned)

|  |  |  |
| --- | --- | --- |
| **Measure** | **Balanced**  **Data Model** | **Unbalanced**  **Data Model** |
| **Error** | 25.68% | 18.18% |
| **Accuracy** | 74.32% | 81.82% |
| **Specificity** | 81.71% | 8.57% |
| **Sensitivity** | 72.48% | 100% |

As we mentioned before, in our project we have decided to run two iterations of our models using balanced and unbalanced data to measure the difference of accuracy. Firstly, using the balanced dataset and a confusion matrix to measure the error rate, the model had an accuracy percentage of 91.5 while using the training set. On the other hand, the validation set scored 81.8%. Thereafter, we initiated the Pruning process where we did an initial grid search and found that a max depth of 30 was the most optimal. Rerunning the grid search with a max depth 30, we are able to get an actual score of 74.32%.

In our second iteration using an unbalanced dataset, the training set scored an accuracy percentage of 92.4%. Whereas the validation set scored 78.1%. Upon initiating the Pruning process, we were able to do a grid search and found that a max depth of 10 was optimal. Rerunning the grid search with a max depth of 10, we were able to get an actual score of 81.8%

Comparing the figures between the training set and the validation set in both iterations, it is evident that the training set holds a higher accuracy percentage. This is due to the nature of overfitting, which is common when making CART models. The condition arises when the model fits the training data but fails to generalize the unseen validation data. This could suggest that the model captures the noise of the training data rather than the important patterns.[8] In order to combat this, a method called Pruning is implemented in order to prevent overfitting.

By tuning hyperparameters of the decision tree, the model can remove parts of the decision tree to combat this.[8] In our project, we specifically implemented a grid search and used a combination of different parameter values. This in turn led us to use an exhaustive system to find a tree with the smallest error.[8] Upon finding the most optimal tree, we used cross-validation on the validation set in order to evaluate actual performance.[8]

**Logistic Regression**

Table 4: Logistic Regression - Summary Confusion Matrix Results - Validation dataset

|  |  |  |
| --- | --- | --- |
| **Measure** | **Balanced**  **Data Model** | **Unbalanced**  **Data Model** |
| **Error** | 32.39% | 18.18% |
| **Accuracy** | 67.61% | 81.82% |
| **Specificity** | 62.86% | 12.57% |
| **Sensitivity** | 68.79% | 99.01% |

As previously mentioned, to evaluate the classification performance of the logistic regression model we constructed a confusion matrix. The summary of its results is in table 4. Using the validation and testing datasets, we compared the classification accuracy for each one. Beginning with the balanced validation dataset, it appears to have a 67.6% accuracy. The same was done for the unbalanced validation dataset, resulting in an accuracy of 81.8%.

As for the testing dataset consisting of only five records, the accuracy for each model is available in the following table:

Table 5: Summary Confusion Matrix Results – Testing dataset

|  |  |  |
| --- | --- | --- |
| **Model** | **Balanced Data Model - Accuracy** | **Unbalanced Data Model - Accuracy** |
| **Naive Bayes** | 60% | 80% |
| **CART** | 80% | 80% |
| **Logistic Regression** | 60% | 80% |

Comparing both balanced and unbalanced validation datasets, in table 5. We can conclude that the balanced data model does not seem to classify ‘delays’ (0) and ‘on time’ (1) as well as the unbalanced. Having such a small dataset, there is very little opportunity for error, as compared to the larger balanced and unbalanced datasets.

# CONCLUSION

Ultimately, it appears that amongst Naïve Bayes, CART, and the logistic regression model, the decision tree and logistic regression will produce the same results for predicting whether a flight is on-time. However, the classification tree model’s specificity and sensitivity values were higher and decision trees are easier to explain and interpret. Therefore, we recommend using the decision tree model to predict flight delays in the future.

# REFERENCES

1. Author, I. E. (2022, December 2). *How to find flight number: A guide for first-time flyers*. Travel Blog | Travel Inspiration, Tips and News | Travel Diary. Retrieved December 7, 2022, from https://www.indianeagle.com/traveldiary/how-to-find-flight-number/
2. Balakrishnan, H. (2016, April 28). Control and optimization algorithms for Air Transportation Systems. Annual Reviews in Control. Retrieved December 9, 2022, from <https://www.sciencedirect.com/science/article/abs/pii/S1367578816300220>
3. Hendrickx, R., Zoutendijk, M., Mitici, M., & Schafer, J. (2021). Considering airport planners’ preferences and imbalanced datasets when predicting flight delays and cancellations. *2021 IEEE/AIAA 40th Digital Avionics Systems Conference (DASC)*. https://doi.org/10.1109/dasc52595.2021.9594367
4. Khan, W. A., Ma, H.-L., Chung, S.-H., & Wen, X. (2021). Hierarchical Integrated Machine Learning Model for predicting flight departure delays and duration in series. *Transportation Research Part C: Emerging Technologies*, *129*. <https://doi.org/10.1016/j.trc.2021.103225>
5. Kuhn, N., & Jamadagni, N. (n.d.). *Application of Machine Learning Algorithms to Predict Flight Arrival Delays*. stanford.edu. Retrieved December 9, 2022, from http://cs229.stanford.edu/proj2017/final-reports/5243248.pdf
6. Kumar, Satyam. (2021). 3 Techniques to Avoid Overfitting of Decision Trees. *Towards Data Science.* Retrieved December 8, 2022, from <https://towardsdatascience.com/3-techniques-to-avoid-overfitting-of-decision-trees-1e7d3d985a09>
7. Purdue University (Ed.). (n.d.). *Airline 2008 Dataset Definition*. Purdue University - Department of Statistics. Retrieved December 5, 2022, from https://www.stat.purdue.edu/~lfindsen/stat350/airline2008\_dataset\_definition.pdf
8. Shmueli, G., Bruce, P. C., Gedeck, P., & Patel, N. R. (2020). *Data mining for Business Analytics: Concepts, techniques and applications in Python*. John Wiley & Sons, Inc.
9. Tang, Y. (2021). Airline flight delay prediction using machine learning models. *2021 5th International Conference on E-Business and Internet*. <https://doi.org/10.1145/3497701.3497725>
10. Truong, D. (2020, December 10). *Using causal machine learning for predicting the risk of flight delays in Air Transportation*. Journal of Air Transport Management. Retrieved December 9, 2022, from https://www.sciencedirect.com/science/article/abs/pii/S0969699720305755
11. [Yazdi, M. F., Kamel, S. R., Chabok, S. J., & Kheirabadi, M. (2020). Flight delay prediction based on Deep Learning and Levenberg-Marquart algorithm. *Journal of Big Data*, *7*(1). <https://doi.org/10.1186/s40537-020-00380-z>

# APENDIX

Table 1: Reference Table for Numerical to Categorical Values

Table 2: NB - Summary Confusion Matrix Results - Validation Dataset

Table 3: CART - Summary Confusion Matrix Results - Validation Dataset

Table 4: Logistic Regression - Summary Confusion Matrix Results - Validation Dataset

Table 5: Summary Confusion Matrix Results – Testing dataset

\*\*Tables 2,3,4 were created using Excel based on the confusion matrix of each model.