Airline Arrival Delay Prediction

Venkata Satya Sri Ram Giduthuri  
*Masters in Computer Science*  
*Syracuse University*Syracuse, United States  
vgiduthu@syr.edu

Rahul Ravikanti  
Masters in Computer Science *Syracuse University*Syracuse, United States  
vgiduthu@syr.edu Pranathi Nallala  
Masters in Computer Science *Syracuse University*Syracuse, United States  
vgiduthu@syr.edu

Nikhilesh Gunnam  
Masters in Computer Science *Syracuse University*Syracuse, United States  
vgiduthu@syr.edu

*Abstract*—This electronic document is a “live” template and already defines the components of your paper [title, text, heads, etc.] in its style sheet. *\*CRITICAL: Do Not Use Symbols, Special Characters, Footnotes, or Math in Paper Title or Abstract*. (*Abstract*)

Keywords—Machine Learning, Flight Delay Prediction, Syracuse, XGBoost Classifier, Random Forest Classifier, Decision Trees Classifier, Exploratory Data Analysis, Classification, Bureau of Transportation Statistics

# Introduction

Airline arrival delays can have significant implications for passenger experience and airline operations. Predicting these delays is crucial for minimizing disruptions and optimizing flight schedules. In this project, we leverage historical flight data and weather information to develop accurate models for predicting arrival delays at Syracuse airport. By analyzing patterns and trends in flight data, we aim to provide actionable insights for stakeholders in the aviation industry.

# Data Collection

## What is Relevant Data?

The initial step is to gather relevant data for analysis and model training. For our predictive analysis of airline arrival delays, historical data of arrivals, departures and weather is essential to train the model effectively. It is important to note that there might be other real-world events are influenced by numerous factors beyond past occurrences in the historical data.

## Historical Flight Data

To collect historical flight data for various airlines arriving at Syracuse airport, we utilize the Bureau of Transportation Statistics website. We focus on data from the year 2019 to ensure an adequate historical context for our analysis. Also, we identified the combinations of airlines and origin airports in the arrivals data and downloaded the corresponding data from the same site.

## Weather Data

For weather data collection, we employ Weatherbit APIs to fetch historical weather data. Each team member utilizes their individual token to access the API and retrieve hourly weather data from January 1 2019, to April 1 2024. By collecting historical weather data for all relevant airports, including SYR, JFK, ORD and MCO, we ensure comprehensive coverage of weather-related factors influencing flight delays.

## Weather Forecast Data

We also collect forecast weather data using the Weatherbit Forecast API for the period between April 19, 2023, and April 24, 2023. This forecast data will serve as additional features in our prediction dataset, enriching our analysis with future weather predictions.

With historical flight and weather data collected for all pertinent airports, we proceed to the next phase of refining and understanding the data to facilitate optimal model training and predictive accuracy.

# Exploratory Data Analysis (EDA)

## Data Cleaning

Before you begin to train any model, we need to clean the data. Since some of the column names have spaces and some of them have special characters in them, we converted them into snake case format.

We also checked if any columns have null values in the data, and observed that “tail number” is empty in 457 rows and it is irrelevant for our model, so, we won’t consider it for our model.

The data types of some columns such as arrival delay (minutes), etc are parsed as object / string. So, we are converting them into integers.

Also, we are creating a categorical dependent variable for categorizing arrival delay as “delay”, “on-time” and “early” if flight is more than 5 minutes late, between 5 min late and 5 min early and more than 5 min early respectively.

We extract additional features such as the time block (morning/afternoon/evening/night) which can give insights to identify the impact of it on flight delay, month of scheduled arrival, weekday of scheduled arrival. We can further classify the days into weekdays and weekends which can help in identifying the behavior of flights during weekdays and weekends and months as snowy month to identify the months with snow at Syracuse as flights tend to operate differently during Syracuse winters.

Since the flights operate in a different fashion at the time of covid i.e, 2020, we exclude it from the data. We also eliminated 2019 as the travel rules are different back then i.e, pre-covid.

It is important to consider the federal holidays as there might be a difference in frequency of travel at lot of airports during holiday periods. We used USFederalHolidayCalendar module (form pandas.tseries.holiday), to identify the federal holidays during those years. We also gave a buffer of 3 days if the holiday is thanksgiving, 7 days if it is Christmas, as these are more famous holidays in US.

We want to identify the previous flight based on the actual arrival time. A lot of times, when the arrival time is after midnight, the date hasn’t been changed. So, we introduced a new datetime column to correct it and identified the previous flight if it is delayed or not and created a new column “prev\_flight\_time\_difference” which is binary, and it is 1 if the time difference between these flights is less than 1 hour. This row might help in the model to decide to consider the previous flight delay status.

In the historical weather data and forecast data we obtained from weatherbit APIs; we observed that only few columns are common. So, we will filter both these datasets for the features we have in common as we need the columns we used at the time of prediction.

## Arrivals – EDA

We identified that the trends in any day of the week are similar in nature at Syracuse. But we’ll consider the “day\_of\_week” and “weekend” columns since it might be a different case at the origin airports.

A screenshot of a computer

Description automatically generated

But there’s a different trend in fed holidays.

A screenshot of a computer

Description automatically generated

By looking at the median arrival delay and count of flights (as per status) for each hour, we identified the peak hours at Syracuse - 0, 10, 11, 16, 17, 18, 22, 23 (in 24-hour format). We created a new column which identifies if the scheduled arrival time is in peak hour or not.

## Departures - EDA

We downloaded datasets from the origin airport to render our analysis on departures at airports. We began our analysis by loading the flight departure data from a CSV file. Initial exploration included the creation of key features that summarize the sources of delays into three main categories: delays caused by airlines ('Delay\_by\_airlines') and delays caused by airport operations ('Delay\_by\_airport') and also a combination of both. These features were computed by summing specific delay-related columns, providing a consolidated view of the factors contributing to overall flight delays.

**Airline based analysis**: We aggregated delay data by 'Carrier Code' to analyze the frequency and average of delays per airline. This involved calculating the total, average delays and percentage of delays, and determining the busiest times for each carrier. The analysis helped identify which airlines frequently face delays and the typical severity of these delays, essential for operational planning and customer service enhancements

**Airport based analysis**: Further, we focused on airport operations by grouping data by 'Origin Airport'. Here, we computed the average, count percentage of delays caused by airport factors, allowing us to pinpoint airports with higher incidences of delays and assess the impact of airport operations on overall travel times. This insight is crucial for airport authorities aiming to improve efficiency and reduce delay times.

**Departure based analysis**: Additionally, we performed a departure-based analysis by combining 'Carrier Code' and 'Origin Airport' to study delays at the granularity of specific flights. This included counting delays and calculating average and percentage delay times for each departure. This detailed level of analysis aids airlines and airports in identifying specific flights that are consistently delayed and may require operational adjustments.

This analysis helps identify which airlines and airports frequently encounter delays and the typical magnitude of these delays. These insights are critical for optimizing flight schedules and improving airline and airport management strategies.

## Merging

All our features such as arrivals features, we created newly weather at Syracuse and origin airports are scattered across different files. To prepare the dataset for training, we need all the features in a single tabular format / dataframe in this case. So, we merge them accordingly.

To merge the weather data at Syracuse with arrivals data or weather data at origin airports with the respective departures data, we use pandas “merge\_asof” function which maps the flight with the weather as per the weather details identified at the closest time to its scheduled departure time. In this way, we merge weather data at Syracuse to flights arrival data and weather data at origin airports to the departure data.

Now, we merge both datasets along with the statistical information of airports and airlines, which will give us information about the flight carrier, origin airport, destination airport, and their weather information. Let’s add 15 minutes buffer to the busiest time to decide if the flight’s scheduled departure is in the busiest time at the origin airport or not.

# Feature Selection

Since we have a lot of features in the data after merging, we wish to remove some of them from the data which we think have no real importance at the prediction time. These include tail number, flight number, date, scheduled arrival time, carrier code, origin airport, etc. Also, we don’t have access to some of the features in real-time, such as wheels-on-time, delay weather, etc. We further refine the weather related features by conducting a correlation test between the numerical features in the weather and the delay weather column and manually selected few features from them. We removed the wind direction column since we don’t have the information regarding weather it supports or opposes the flight landing process at the runway as we have no idea about runway alignment.

Given the categorical nature of many features in our dataset, such as weather\_description, day\_of\_week, and month etc it was necessary to transform these categories into a format suitable for machine learning algorithms. One-hot encoding was utilized to convert these categorical variables into a series of binary columns, each representing a unique category. This process helped us to prevent the machine learning algorithm from misinterpreting the ordinal relationship between categories.

We employed OneHotEncoder from sklearn.preprocessing to transform these categories. This method ensures that each category value is converted into a new binary column, enhancing the interpretability and effectiveness of the model.

# Model Development

## Data Standardization

Now that we selected the features and one-hot encoded the categorical features, we can standardize the data before training. The first step is to split the dataset into train and test sets using test\_train\_split from sklearn.model\_selection. Since we are dealing with classification model, it is important to use stratify variable while splitting the data into train and test sets in-order to uniformly distribute the data among various classes in dependent variable among them.

Since we have a lot of columns after one-hot encoding, we wish to reduce the dimensions and eliminate the noise as far as possible. So, we did a Principal Component Analysis (PCA) with 20 components & random state = 42 (taken from Hitchhiker’s Guide to Galaxy). These components are linear combinations of original features and explains the variability in the data in a better way.

Now that we have reduced representation of the features, we can scale the data using StandardSclaer from sklearn.preprocessing, to scale the data with mean = 0 and standard deviation = 1. This ensures each feature to be given equal importance during the model training.

## Model Training

Headings, or heads, are organizational devices that guide the reader through your paper. There are two types: component heads and text heads.

Component heads identify the different components of your paper and are not topically subordinate to each other. Examples include Acknowledgments and References and, for these, the correct style to use is “Heading 5”. Use “figure caption” for your Figure captions, and “table head” for your table title. Run-in heads, such as “Abstract”, will require you to apply a style (in this case, italic) in addition to the style provided by the drop down menu to differentiate the head from the text.

Text heads organize the topics on a relational, hierarchical basis. For example, the paper title is the primary text head because all subsequent material relates and elaborates on this one topic. If there are two or more sub-topics, the next level head (uppercase Roman numerals) should be used and, conversely, if there are not at least two sub-topics, then no subheads should be introduced. Styles named “Heading 1”, “Heading 2”, “Heading 3”, and “Heading 4” are prescribed.

## Metrics

#### Positioning Figures and Tables: Place figures and tables at the top and bottom of columns. Avoid placing them in the middle of columns. Large figures and tables may span across both columns. Figure captions should be below the figures; table heads should appear above the tables. Insert figures and tables after they are cited in the text. Use the abbreviation “Fig. 1”, even at the beginning of a sentence.

1. Table Type Styles

| Table Head | Table Column Head | | |
| --- | --- | --- | --- |
| Table column subhead | Subhead | Subhead |
| copy | More table copya |  |  |

1. Sample of a Table footnote. (*Table footnote*)
2. Example of a figure caption. (*figure caption*)

Figure Labels: Use 8 point Times New Roman for Figure labels. Use words rather than symbols or abbreviations when writing Figure axis labels to avoid confusing the reader. As an example, write the quantity “Magnetization”, or “Magnetization, M”, not just “M”. If including units in the label, present them within parentheses. Do not label axes only with units. In the example, write “Magnetization (A/m)” or “Magnetization {A[m(1)]}”, not just “A/m”. Do not label axes with a ratio of quantities and units. For example, write “Temperature (K)”, not “Temperature/K”.

# Predictions

Write about predictions.

# Conclusion

Write conclusion

##### Acknowledgment *(Heading 5)*

The preferred spelling of the word “acknowledgment” in America is without an “e” after the “g”. Avoid the stilted expression “one of us (R. B. G.) thanks ...”. Instead, try “R. B. G. thanks...”. Put sponsor acknowledgments in the unnumbered footnote on the first page.

##### References

The template will number citations consecutively within brackets [1]. The sentence punctuation follows the bracket [2]. Refer simply to the reference number, as in [3]—do not use “Ref. [3]” or “reference [3]” except at the beginning of a sentence: “Reference [3] was the first ...”

Number footnotes separately in superscripts. Place the actual footnote at the bottom of the column in which it was cited. Do not put footnotes in the abstract or reference list. Use letters for table footnotes.

Unless there are six authors or more give all authors’ names; do not use “et al.”. Papers that have not been published, even if they have been submitted for publication, should be cited as “unpublished” [4]. Papers that have been accepted for publication should be cited as “in press” [5]. Capitalize only the first word in a paper title, except for proper nouns and element symbols.

For papers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

1. G. Eason, B. Noble, and I. N. Sneddon, “On certain integrals of Lipschitz-Hankel type involving products of Bessel functions,” Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*
2. J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
3. I. S. Jacobs and C. P. Bean, “Fine particles, thin films and exchange anisotropy,” in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
4. K. Elissa, “Title of paper if known,” unpublished.
5. R. Nicole, “Title of paper with only first word capitalized,” J. Name Stand. Abbrev., in press.
6. Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].
7. M. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

**IEEE conference templates contain guidance text for composing and formatting conference papers. Please ensure that all template text is removed from your conference paper prior to submission to the conference. Failure to remove template text from your paper may result in your paper not being published.**

We suggest that you use a text box to insert a graphic (which is ideally a 300 dpi TIFF or EPS file, with all fonts embedded) because, in an MSW document, this method is somewhat more stable than directly inserting a picture.

To have non-visible rules on your frame, use the MSWord “Format” pull-down menu, select Text Box > Colors and Lines to choose No Fill and No Line.