**Analysis Report on**

**Airline Flight Delay Prediction**

**Team Delta Force**

Brad Dyer, Sophia Lee, Jeff Menzin, Hoai Anh Nguyen,

Ferdinand Percentie, Michael Pham

Data Analytics Engineering Department, George Mason University

CS 504: Principles of Data Management and Mining

Dr. James Baldo

December 10, 2020

Table of Contents

[Abstract 4](#_Toc58446502)

[1 Introduction 5](#_Toc58446503)

[1.1 Background and Rationale 5](#_Toc58446504)

[1.2 Research 6](#_Toc58446505)

[1.3 Project Objectives 7](#_Toc58446506)

[1.4 Problem Space 8](#_Toc58446507)

[1.5 Primary User Story: 8](#_Toc58446508)

[1.6 Solution Space 9](#_Toc58446509)

[1.7 Product Vision - Sample scenarios 9](#_Toc58446510)

[*Scenario #1 9*](#_Toc58446511)

[*Scenario #2 9*](#_Toc58446512)

[1.8 Definition of Terms: 10](#_Toc58446513)

[2 Data Acquisition 10](#_Toc58446514)

[2.1 Field Descriptions 11](#_Toc58446515)

[2.2 Data Context 13](#_Toc58446516)

[2.3 Data Conditioning 13](#_Toc58446517)

[2.4 Data Quality Assessment 14](#_Toc58446518)

[2.5 Other Data Sources 15](#_Toc58446519)

[3 Analytics and Algorithms 16](#_Toc58446520)

[3.1 Introduction 16](#_Toc58446521)

[3.2 Linear Regression 16](#_Toc58446522)

[3.3 XGBoost 19](#_Toc58446523)

[3.4 Random Forest 24](#_Toc58446524)

[4 Visualization 27](#_Toc58446525)

[5 Findings 32](#_Toc58446526)

[5.1 Regression Analysis Findings 32](#_Toc58446527)

[5.2 Classification Findings 32](#_Toc58446528)

[*5.2.1 Binary Classification Findings – Label1 32*](#_Toc58446529)

[*5.2.2 Multi-classification Findings – Label2 33*](#_Toc58446530)

[*5.2.3 Multi-classification – Label3 34*](#_Toc58446531)

[*5.2.4 Conclusion 34*](#_Toc58446532)

[5.3 Feature Importance 35](#_Toc58446533)

[6 Summary 35](#_Toc58446534)

[7 Future Work 36](#_Toc58446535)

[7.1 Clustering Algorithms 36](#_Toc58446536)

[7.2 Other Factors 36](#_Toc58446537)

[Appendix A – Project Repository 39](#_Toc58446538)

[Appendix B – Risk Section 39](#_Toc58446539)

[Appendix C – Carrier Identifier Listing 39](#_Toc58446540)

[Appendix D – Airport Identifier Listing 39](#_Toc58446541)

[Appendix E – References 40](#_Toc58446542)

# Abstract

Airline flight delays significantly affect airlines and passengers, and cause airlines tens of millions of dollars in losses each year. Delays also create stress in the passengers and impact their lives when they cannot make it to their destination on time. Poor weather conditions are one of the most prevalent causes of flight delays. For this study, a combined dataset between carrier on-time performance data with hourly weather data that correlated to the time of departure from the US’s four largest airlines at 12 of the busiest US airports from January 2018 to December 2019 was analyzed by three separate algorithms – Linear Regression, XGBoost, and Random Forest. Linear Regression was used to identify the most influence variables on flight delays. Meanwhile, binary and 5-value classification problems were the approach for XGBoost and Random Forest analyses. The goal was to give airline decision-makers and passengers a tool to predict departure delays based on weather conditions.

All three models showed that hourly precipitation had the biggest impact on weather delays, while other weather conditions had meaningful, but less significant impacts. For labeled data, XGBoost and Random Forest had similar results when classifying the binary categories. The application to a binary label performed better than application to the multi-classification label. The results of this project can help airline decision-makers and passengers better determine what models will best predict weather delays and what attributes have the biggest impact on departure weather delays.

# Introduction

## Background and Rationale

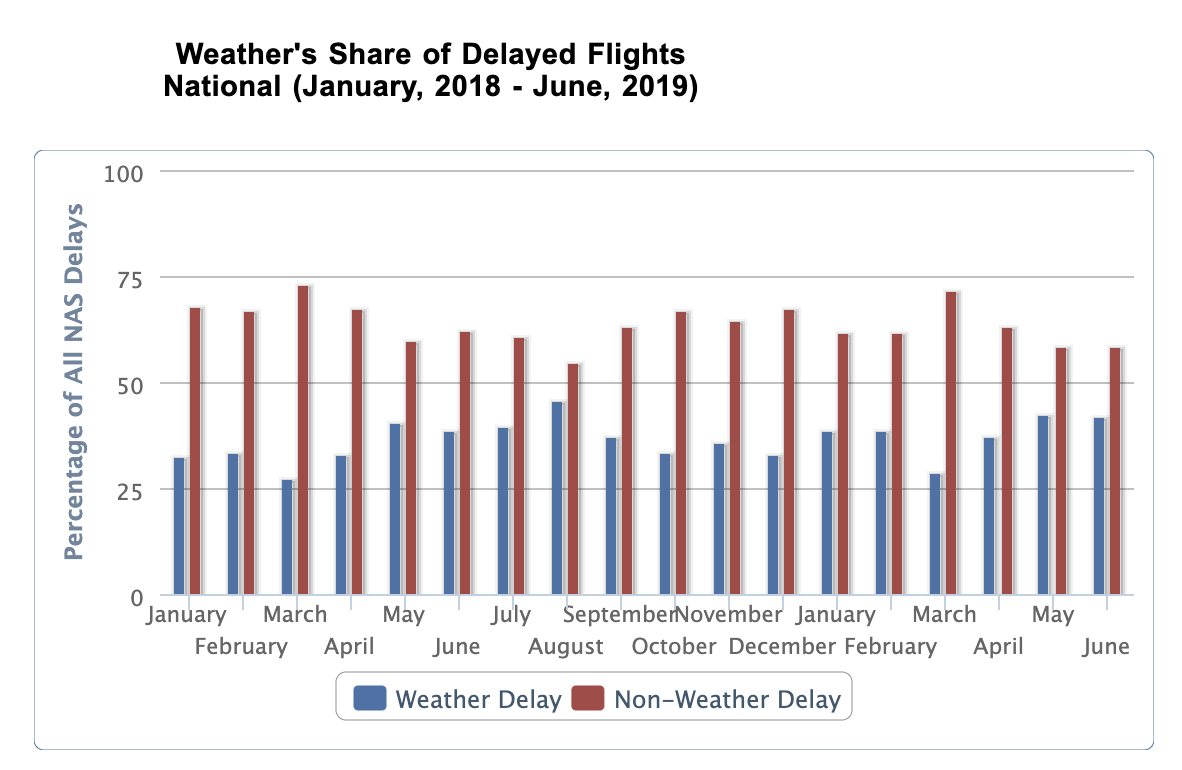
Air Traffic in the United States is the busiest of any country in the world. 2,900,000 passengers board planes every day traveling to destinations both nationally and internationally. 45,000 flights are handled by the Federal Aviation Administration (FAA) on average each day with as many as 4,500 aircraft airborne at any given time throughout the United States’ National Airspace System (“Air Traffic”, 2020).

These numbers demonstrate how critical it is for passenger-carrying aircraft to depart on time. Delays can cost the airlines tens of millions of dollars a year, especially if these delays lead to subsequent delays in the flight schedule. For example, airlines must pay employees extra during delays. This includes paying schedulers overtime for rescheduling aircrew and aircraft, customer service agents for accommodating unhappy passengers, or maintenance and baggage teams for simply waiting on delayed aircraft to arrive. This is just part of the added cost and does not account for the unknown losses of dissatisfied passengers who, because of their experience, vow to “never fly with that airline again.”

This leads us to the immediate effect these delays have on thousands, if not tens of thousands, of passengers’ lives with second and third order effects impacting many more lives (Akers et al., 2020). These second and third order effects could be caused by aircraft being delayed that would be used for subsequent flights, or aircrew – who typically fly multiple flights each day – being delayed which would also have a domino effect causing even more delays. These situations cause stress and wasted time for passengers who rely on airlines to get them to their desired destinations. Flight delays due to weather account for 24-30% of overall flight delays. With an industry topping $1.7 trillion in the U.S. alone and providing over 10 million jobs, there are huge economic ramifications for chronic flight delays and cancellations (Airlines for America, 2020).

No one – neither the airline nor the passengers – wants delays. Everyone wants airline operations to run smoothly and on time; and when delays do occur, they want them to have minimal impact on the rest of the schedule. Part of this process of keeping things on time and ensuring delays have minimal impact includes the process of predicting when a delay will occur with reasonable confidence.

The purpose of this project was to analyze historical flight and weather data in order to produce a model that can be used to predict when a flight delay will occur, based on the weather at the location of departure. The model was developed using flight data for the four US-based airlines with the largest market share - American Airlines, Delta Air lines, Southwest Airlines, and United Airlines. It included flights originating from 12 of the busiest US airports based on passenger throughput (Dean, 2020). These airports included international airports in Atlanta (ATL), Baltimore (BWI), Chicago (ORD), Dallas-Fort Worth (DFW), Denver (DEN), Dulles (IAD), Los Angeles (LAX), New York John F. Kennedy (JFK), Orlando (MCO), San Francisco (SFO), Seattle-Tacoma (SEA), Washington Reagan (DCA). This project focused on weather as an independent variable in our model because research indicates that it is frequently cited as the primary reason for many air carrier delays (Busson, 2020).



*Figure 1.1. Weather’s Share of Delayed Flights related to National Aviation System Delays*

Figure 1.1 is a graph from the Bureau of Transportation Statistics. It shows the impact weather has on flight delays. This chart shows specifically the comparison between weather delays and non-weather delays related to National Aviation System (NAS) Delays. Despite the clear fact that weather is a significant factor to delayed flights, it is unclear how significant. The development of extreme weather conditions, such as hurricanes, tornadoes, and blizzards, nearly always lead to flight delays. The reason for this is clear; these severe weather conditions make it extremely difficult for pilots to safely pilot the airplanes, due to a number of factors including poor visibility, high winds, and icy conditions. However, delays due to extreme weather only account for approximately 6% of all flight delays, meaning that oftentimes, weather is cited at the primary reason for a delay when it actually is not (Busson, 2020). Just because the weather is sub-optimal (such as mild rain, wind, or cloud cover), that does not necessarily mean that flights need to be delayed.

This knowledge only reinforced the desire to develop an accurate model for predicting flight delays based on weather. Such a model could be used by passengers to determine whether their flight could be delayed, based on the current weather conditions. If it turned out that the conditions were not severe enough to warrant a delay, but the flight was, nevertheless, delayed, then the airline may have delayed the flight for a completely different reason but cited it as a weather delay. In this situation, the passenger may be entitled to financial compensation, which is not the case for legitimate weather-related delays (Busson, 2019).

This model could also be used by pilots to determine whether they should be flying in the current weather conditions. For example, when getting ready to depart, if a pilot used this model to determine that the current weather conditions have historically led to delays 90% of the time, then the pilot knows that there is data backing-up his decision to delay based on the weather.

## Research

During this project, various sites were searched for historical flight delay data. It was ultimately determined that the sources took flight delay data from the same repository. This conclusion was based on downloading sample data from each site and determining that the labeling, format and information within the tables were identical.

The same initial research process was done with the weather data. Some sites had data only through 2017, while others did not have specific hourly data. Ultimately, the data chosen had hourly and daily weather measurements for all the stations chosen for analysis.

This project was influenced by the paper “Predictive Analytics for Airline Operations” completed by the George Mason University DAEN 690 Summer 2020 Capstone project team (Akers et al., 2020). The report’s objective was to predict airline “meltdown” days, the days when airlines have the most cumulative delays (as delays build for flight to flight) at the end of the day.

This project diverged from the “meltdown” project, but their report was informative for several reasons. First, the paper was informative regarding the models they could use to analyze the flight delay data. The report discussed various analytical tools – clustering techniques such as Support Vector Machines (SVM) and K-Means Clustering, the machine learning boosting algorithm XGBoost, the time-series machine learning algorithm Long Short-Term Memory (LSTM), as well as standard statistical analysis such as linear regression. These suggestions ultimately led to the tools chosen for this project. A logistical regression model was also considered for analysis. All these options were researched and ultimately three labeled columns of the data set were created: one binary, and two multi-classification labels. Then three algorithms were selected: linear regression, XGBoost, and random forest. Linear regression and XGBoost were applied to unlabeled data. XGBoost and random forest were applied to both the binary and multi-classification data.

Second, the paper provided significant background into previous papers delving into the same subject matter. Their paper was itself inspired by a George Mason University DAEN690 Spring 2020 capstone project, “Predictive Analytics for Airline Operations” (Mamdoohi et al., 2020). This paper sought to build a predictive model around cumulative arrival delays on any day based upon delays in the system at 12pm EST. The report also discussed several predictive modeling methods; Long Short-Term Memory (LSTM), Support Vector Machine (SVM), Random Forest, and recursive regression, to achieve an accuracy rate of 77%. The difference between the Spring 2020 paper and the Summer 2020 paper was that the former focused on cumulative delays as a whole, whereas the latter sought to break the cumulative delays down by airline.

The first outside paper mentioned was the 2016 paper “Prediction of Weather- Airline Delays Based on Machine Learning Algorithms” (Choi et al., 2016). The goal of the paper was to predict airline delays caused by weather conditions using supervised machine learning algorithms. The most successful algorithm tested in this paper was the Random Forest algorithm, which had an 80% accuracy rate.

Another paper mentioned was the 2020 paper “Flight Arrival Delay Prediction and Analysis Using Ensemble Learning” (Dou, 2020). This paper sought to improve the accuracy of predicting flight arrival delay time using the Cat-boost model. This model slightly improved on the 2016 Choi et. al paper with an 80.44% accuracy and an average prediction error of 9.733.

During analysis for this project, python was used for scraping and cleaning data as well as for the data analysis. Microsoft Power BI was used to integrate visualization since it provided seamless graphical presentation of the results. This was based on the team’s desire to familiarize themselves with these tools and become more proficient using them in situations like this.

## Project Objectives

This work seeks to accurately predict flight delays due to weather in passenger-carrying airlines based on historical flight delay data from the 4 largest airlines and 12 of the busiest airports within the United States correlated to the weather conditions at the time of departure. This prediction can then be used to predict future flight delays based on current weather conditions.

Flight delays due to weather factors was the focus of this project. Because weather conditions differ across the country, departure delays were considered from 12 airports within various regions of the country. Also, because weather patterns differ throughout the year, a continuous 2-year period was evaluated. This mitigated any discrepancies that may have arisen due to differences in regional or seasonal weather patterns. Due to the timeline of this project, weather delays within the category of NAS delays were not considered but is a recommended future project discussed later in the report. Only weather delays

As a part of the data consolidation process, data was collected, combined, and cleaned for both historical flight delays and historical weather conditions at 12 of the busiest airports within the United States from the four largest air carriers, then combined the datasets to determine how weather impacted the historical flight’s delay. This data was analyzed to determine which weather factors impacted departure delays the most. Three separate algorithms – Linear Regression, XGBoost, and Random Forest were applied for this project’s analysis. The goal was to give airline decision-makers and passengers a tool to predict departure delays based on weather conditions.

## Problem Space

Airports, airlines, and passengers are significantly affected by flight delays. Weather is one of the biggest causes of flight delays. Airlines lose tens of millions of dollars because of them. Passengers lose as well. Airline decision-makers and passengers are missing a tool to predict flight delays based on weather. This project's objective was to analyze the weather impact on flight departure at the departure airports and provide sufficient prediction models for the expected delay based on the underlying weather conditions. The results from this study could give a suggestion of models to be used by airline decision-makers, passengers, and others to predict flight delays due to weather. Flights from the top 4 United States airline companies at the 12 busiest airports in the country were considered in the analysis.

The more specifics that the decision-maker has regarding the amount of time the flight is delayed due to weather, the better decision he or she will be able to make. In other words, a binary model of “on-time” or “delayed” would add value, but not as much value as a multi-category model that could predict a range of minutes that a flight will be delayed. Therefore, a 5-category model, for example, will be better for the decision maker. This is because it allows for a more tailored response based on delay time. If all the decision-maker knows is that there is a delay greater than 15 minutes, then only one set of actions can be taken for all delays whether it’s a 16-minute delay or a 60-minute delay. However, in a multi-classification model, if the decision-maker knows that the flight will be delayed less than 15 minutes, he or she would take a specific set of actions; but if he knew that a flight was going to be delayed between 15 and 30 minutes, then he or she would add more actions. If the delay was between 30 and 60 minutes, there would be additional steps to take, and even more if it was delayed over 60 minutes. This type of tailored response is a more desired from the decision-maker.

## Primary User Story:

As an airline decision-maker, I want to be able to reasonably predict how much a flight will be delayed based on current weather conditions so that I can decide whether or not to take action to mitigate the risk of potential delays.

As a passenger, I want to know if my flight will be delayed because of the weather. If it is delayed, I want to know how long it will be delayed so I can plan when to leave for the airport or so I know if I need to plan for alternate flights as soon as I can.

## Solution Space

This project’s model delivers value to the user when it accurately predicts whether a flight will be delayed based on weather. Airline decision-makers can start planning early to prevent fallout from delays and start notifying customers. The result will be a significant reduction in the amount of revenue lost due to flight delays. This model could also be used to help alleviate stress in passengers as well as help them plan their pre-departure travel plans or alternate travel plans if delays are predicted from this model. The objective of this project is to deliver value to the users when the analyzing models accurately predict whether a flight will be delayed based on weather.

As discussed in the problem space, a multi-classification model will be more valuable than a binary model. However, accuracy of the model is also important. If the binary model is significantly more accurate compared to a low accuracy of a multi-classification model, then it would be more reasonable to use a more accurate binary model compared to an inaccurate multi-class model. Therefore, the goal related to this is to determine if a binary or multi-classification model would be more useful based on accuracy.

## Product Vision - Sample scenarios

* *For*: decision-makers within the airlines and passengers flying commercial airlines
* *Who*:  those within the airline making decisions based on potential flight delays as well as passengers interested in gathering information on the potential of a flight delay
* *The*: Flight Delay Prediction Model
* *Is a*: Statistical Modeling Tool
* *That*: Assists airline decision-makers and passengers in determining the potential for flights being delayed based on weather
* *Unlike*: Guessing or making decisions based on lack of information.
* *Our product*: Utilizes advanced analytics and machine learning techniques to deliver a statistical analysis in order to determine whether a flight will be delayed based on weather with a range of accuracy in minutes.
* *Caveats:* The data used in this problem focused primarily on the factor of weather in delays. No other data besides the Bureau of Transportation Statistics and National Oceanic and Atmospheric Administration data was included.

### Scenario #1

As an airline decision-maker, I want to know the potential for a specific flight to be delayed based on the weather forecast. This will help me prioritize flights and proactively mitigate risks of fallout from delays. However, I do not want to make changes if I can determine that the potential for delay is low. I would like to be able to use this model to determine if I need to start making changes to the flight schedule and plan for risk-mitigation or if I should forego any action until the delay is imminent.

### Scenario #2

As a passenger, I want to determine whether or not my flight will be delayed based on potential adverse weather for my departure airport. Do I need to arrive to the gate early or can I use that time to work on my Data Analytics homework versus waiting at the airport when I anticipate that my flight will be delayed? I also want to know that if I am running late, what the potential is for my flight to be delayed? This would determine if I should start looking for other options for flights or alternative transportation. Finally, if I can anticipate ahead of other passengers that my flight is delayed, I can call the airline early or get at the front of the customer service line in order to get the best alternate flight as well as minimize the time it takes to rebook that flight.

## Definition of Terms:

Airline (or air carrier) – A single US commercial company whose purpose is carrying passengers through the air.

Airline decision-makers – Those individuals within the airline whose role is to decide whether or not to cancel a flight or one who makes decisions on the flight schedule based on information.

Delay – A flight is considered delayed when it departs after the flight’s scheduled departure time.

Flight – A single instance of a commercial aircraft with its crew for the intent of carrying passengers from one airport to another.

On time – A flight is considered on time when it leaves the gate before or right at the scheduled departure time and takes off from the departure airport.

Passenger – customers of the airlines traveling on flights. These are the primary revenue sources for commercial airlines and therefore, play a critical role in keeping the airline profitable.

Scheduled Departure Time – The time the flight was originally scheduled to depart. This time is in local time based on the flight’s departure airport.

Weather delay – a flight reported by the air carrier that departs after the flight’s scheduled departure time due strictly to weather, but outside the category associated with weather delays related to National Aviation System delays.

# Data Acquisition

This project used a combined dataset from 2 different reliable, well-known open data sources. The first source was TranStats website of the U.S. Department of Transportation’s Bureau of Transportation Statistics (www.trastats.bts.gov) which provides data about US domestic flight details. This study specifically utilized the airline data on Marketing Carrier On-Time Performance from January 2018 to December 2019. It was decided that two years was a good balance between enough data for analysis and reasonable file size for workability. They chose these years to keep the analysis as current as possible while still avoiding any potential anomalies from the COVID-19 epidemic. This data can be retrieved from <https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=237> (“Airline On-Time”, n.d.).

The second data set was sourced from the National Oceanic and Atmospheric Administration (NOAA) ([https://www.ncdc.noaa.gov](https://www.ncdc.noaa.gov/cdo-web/datatools/lcd)). This data provided hourly land-based weather observations. The same timeframe of January 2018 to December 2019 was used to retrieve data for analysis. This data can be retrieved from [https://www.ncdc.noaa.gov/cdo-web/datatools/lc](https://www.ncdc.noaa.gov/cdo-web/datatools/lcd)d (Local Climatological Data, n.d.)

The initial flight dataset contained 2.2GB of data which included all the domestic flight records in the US from 2018 to 2019 with 52 columns and around 16 million rows. However, the scope of this project focused on the impact of weather on flight performance of four airlines at 12 chosen airports. Therefore, the flight dataset was trimmed down to 187 MB with 1,720,407 records and 10 columns.

The initial weather dataset contained 312,961 rows and 124 columns with a size of 117 MB and only included the weather conditions at the 12 selected airports. Of the 124 columns in the dataset, only 12 of them contained enough valid data to be useful for our purposes. After cleaning, the final weather dataset contained 14 columns and around 312,000 rows for a total size of 72.5 MB.

After pre-processing the two datasets mentioned above, they were combined in the way that the time of departure was correlated with the time of weather observation. Three labeled attributes were created for classification analysis.

The combined dataset used for further analysis has the size of 273 MB with 25 variables and more than 1.72 million records.

## Field Descriptions

The initial combined dataset for this analysis included 37 distinct variables, with a combination of string, numeric, and boolean data types. After cleaning the data of all irrelevant variables, the analysis focused on the following descriptive variables.

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Categories** | **Data Types** | **Description** |
| 1 | DayOfWeek | int | Day of Week |
| 2 | FlightDate | date | Scheduled Flight Date and Time |
| 3 | Carrier | object | Code assigned by IATA and commonly used to identify a carrier. The list of codes for the carriers used in the report can be found in Appendix C. |
| 4 | DepAirportID | object | An identification number assigned by US DOT to identify a unique airport (the flight's origin). These codes can be found in Appendix D. |
| 5 | DepStateAbr | object | Origin Airport, State Abbreviation |
| 6 | CRSDepTime | date | The CRS departure time in local time (hh:mm) |
| 7 | ActualDepTime | date | Actual Departure Time (local time: hh:mm) |
| 8 | DepDelay | float | Difference in minutes between the scheduled and actual departure times. Early departures show negative numbers |
| 9 | DepDelayNew | float | Difference in minutes between scheduled and actual departure time. Early departures set to 0. |
| 10 | Cancelled | boolean | Cancelled Flight Indicator (1=Yes) |
| 11 | WeatherDelay | float | Delays caused by weather (in minutes) |
| 12 | WeatherDate | date | Timestamp of the weather observation |
| 13 | HourlyAltimeterSetting | float | Atmospheric pressure reduced to sea level using temperature profile of the “standard” atmosphere. Given in inches of Mercury (in Hg). |
| 14 | HourlyDewPointTemperature | int | This is the dew point temperature. It is given here in tenths of a degree Celsius. |
| 15 | HourlyDryBulbTemperature | int | This is the dry-bulb temperature and is commonly used as the standard air temperature reported. It is given here in tenths of a degree Celsius. |
| 16 | HourlyPrecipitation | float | Amount of precipitation in inches to hundredths over the past hour. For certain automated stations, precipitation will be reported at sub-hourly intervals (e.g. every 15 or 20 minutes) as an accumulated amount of all precipitation within the preceding hour. |
| 17 | HourlyRelativeHumidity | int | Relative humidity given to the nearest whole percentage. |
| 18 | HourlySeaLevelPressure | float | Sea level pressure (in inches of mercury) |
| 19 | HourlyStationPressure | float | Atmospheric pressure observed at the station during the time of observation. Given in inches of Mercury (in Hg). |
| 20 | HourlyVisibility | float | The horizontal distance an object can be seen and identified given in whole miles. |
| 21 | HourlyWetBulbTemperature | int | This is the wet-bulb temperature. It is given here in tenths of a degree Celsius. |
| 22 | HourlyWindSpeed | int | Speed of the wind at the time of observation given in miles per hour (mph). |
| 23 | Label1 | object | Binary classification label indicating whether an aircraft departed “ontime” meaning at or before departure time or “delay” meaning any time after departure time. |
| 24 | Label2 | object | 3-class label indicating whether an aircraft departed “ontime” meaning within 15 minutes of departure, “delay>15” meaning a 16-to-30-minute delay, and “delay>30” meaning a delay greater than 30 minutes. |
| 25 | Label3 | object | 5-class label indicating whether an aircraft departed “ontime “’meaning at or before departure time, “delay” meaning within 15 minutes of departure time, “delay>15” meaning a 16-to-30-minute delay, “delay>30” meaning a 31-to-60-minute delay, and “delay>60” meaning a delay greater than 60 minutes. |

## Data Context

The data for this project was collected in two separate manners. The flight data from the U.S. Bureau of Transportation (BTS) was collected from reporting data which the four airlines used in this study are federally mandated to report to the DoT on a monthly basis. The regulation specifies exactly what and how this data is to be reported so that it is standardized among all carriers required to report (Federal Register, 2002). The NOAA data was collected by the National Centers for Environmental Information (NCEI) using land-based stations surrounding the 12 major airports used for this project. These stations have instruments that collect all the listed variables used in the data in this project. These stations have imbedded quality control and remove biases associated with factors that could skew the data such as urbanization and instrumentation changes (Land-Based Station Data).

The flight data and weather data were combined into one dataset. The first 17 columns are the flight data. Every flight is separated into a single row with data for each flight listed in the first 17 columns shown in section 2.2. The remainder of the data contains the associated weather data which correlates to the rows of flights.

## Data Conditioning

In order to produce the combined dataset used to analyze this project. The two initial datasets needed to be trimmed, cleaned, and standardized in accordance with the project’s objectives.

1. ***Flight Dataset***

The initial flight dataset contained all records of US domestic flights from 2018 to 2019. As this project focused on only 4 airlines and 12 airports mentioned above, a large amount of data regarding other airlines and airports was filtered out. These records were dropped from the final dataset during the cleansing process. Moreover, only departure records and weather-related data was considered for this study. Therefore, 29 irrelevant variables were dropped. This initial cleaning reduced two-third of the original dataset size.

Furthermore, 6,494 diverted flights and 1,039,933 delayed flights caused by weather-unrelated reasons were filtered out. The cleaning process also included removing 33,880 null values in the departure time, number of minutes of delay, departure delay indicator, departure delay intervals. All the null values in weather cause of delay were replaced by 0 indicating that there was no impact of weather conditions on the flight performance.

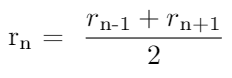
The last part of preparing the flight dataset was converting the planned departure time from integer to timestamp datatype and merging the planned departure time with the flight date to standardize datetime format.

1. ***Weather Dataset***

The initial weather dataset contained 312,961 rows and 124 columns with a size of 117 MB and only included the weather conditions at the 12 selected airports. Of the 124 columns in the initial dataset, most of them contained at least 50% null records. It was decided that any columns with too many null records should be removed from the dataset, since they would not be useful for analysis. A benchmark of 70% was used, meaning that any columns that did not have a value for at least 70% of records would be removed from the dataset. This number was chosen because there was a large drop-off between 50% and 70%, with the most important columns having over 70% of rows populated. This resulted in a final dataset with only 19 columns and a size of 72.5 MB.

Furthermore, any rows containing more than 10 missing or null values were removed from the dataset. Most of the variables in weather dataset had unconformity in datatypes. This issue was handled by removing redundant characters and converting to the correct format according to section 2.3. For variable “HourlyPrecipitation”, the value “T” indicated a trace amount of precipitation which was considered as 0.

To deal with null records scattered throughout the weather dataset, it was decided that the best approach would be to estimate the value of these cells based on the value of surrounding cells. This approach makes sense because the weather data is chronological, and the weather conditions at any point time are likely to be similar to the conditions moments earlier or later. For all columns with a numeric datatype (integers, floats, etc.), the value of null/NaN cells was calculated using the averaging formula below, where rn indicates the value of the cell at row n:



If the value rn+1 was also unknown, then the value of rn was instead set equal to the value of rn-1 for the purposes of populating missing data.

Once the dataset was cleaned, the name of the airport was mapped to each row based on the weather station ID. This was done so that the flight dataset and weather dataset could be joined by airport ID.

1. ***Combined Dataset***

After cleaning the two initial datasets, a combined data was merged using the following method:

* Weather Station IDs were mapped to the corresponding Airport IDs
* The planned flight dates and times were matched with the nearest hour of a weather observations
* Duplicate variables used for mapping 2 datasets were filtered out

## Data Quality Assessment

* Completeness: The initial flight data was extremely complete. The initial weather dataset had blocks of data within the variables that contained null values. However, after cleaning irrelevant variables, and handling missing and null values, the team considered the dataset highly complete. The details of cleaning process were described in section 2.3 showing the majority of null values were removed or replaced and the small amount of remaining null values belonging to optional variables that would not affect the overall analysis.
* Consistency: During the data cleaning process, the team made sure that the dataset was consistent throughout. The team’s subject matter expert reviewed random samples of the data to ensure the data was consistent. For example, that the scheduled versus actual departure times matched the number of minutes delayed and that temperature was consistent with the type of precipitation. There were no inconsistencies found in the data types used. The data is highly consistent.
* Uniqueness: The data within the final dataset is unique. Flight departure time, departure airport and flight carrier were considered as tracking variables used to verify the uniqueness of each record.
* Accuracy: The accuracy of the data is high. Due to the authoritative, official government sources using very reliable tracking and measuring tools this team had high confidence in the accuracy of the data it collected from its sources. In the rare instance the team did notice any values that did not make sense – or, more common, were missing – they either corrected, removed or replaced it with the closest relevant values as explained in the previous section. Therefore, this dataset maintained its accuracy throughout the cleansing process.
* Integrity: The individual dataset’s integrity was high since they were two original datasets pulled from authoritative, official open data sources, namely BTS and NOAA. The raw flight data was largely created automatically from flight trackers and the weather observations data was taken from automated stations that read the weather at specified intervals. The automatic data collecting process eliminates the potential for human error. The team focused on keeping the integrity of the dataset during the aggregation process. They ensured that the timeframe fro the weather matched the timeframe of the departures. They also made sure that the weather data variables that remained in the combined data worked together with the flight data.
* Conformity: After data cleaning process mentioned in section 2.3, each field contains values that conform to the same convention and style. The team has high confidence in the conformity of the data.
* Overall Quality: Overall, the final combined dataset is of high quality ready for further analysis.

## Other Data Sources

The group considered using the TranStats website, but the source of the data is Bureau of Transportation Statistics - U.S. Department of Transportation which is the official source of all domestic flight data in the US, no other data source was considered.

For the weather data source, a number of unofficial datasets were discovered on websites such as Kaggle. It had a dataset for all hourly weather over the United States from 2012-2017, but the data set didn’t fit the time period needed by the project. The NOAA dataset was found on Google Cloud and because the necessary data existed there, an attempt at gathering the data using BigQuery was made but it became quite complicated to narrow down the airports by airport code and to understand which columns to query without the proper documentation available. We chose to go with the official option directly from the NOAA website and chose the hourly Local Climatological Data, downloading each airport individually.

# Analytics and Algorithms

## Introduction

The objective of this project was to predict if the departing flight was delayed based on the weather at the departure airport and how long the delay would be. To achieve this objective, three Machine Learning models were considered for analysis: Linear Regression, XGBoost, and Random Forest.

10 weather-related Independent Variables were used in the analysis for all three models: "HourlyAltimeterSetting", "HourlyDewPointTemperature", "HourlyDryBulbTemperature", "HourlyPrecipitation", "HourlyRelativeHumidity", "HourlySeaLevelPressure", "HourlyStationPressure", "HourlyVisibility", "HourlyWetBulbTemperature", and "HourlyWindSpeed".

## Linear Regression

**3.2.1 Linear Regression Description**

Linear Regression is a term used for a category of analysis techniques that seek to determine how accurately one or more Independent Variables (IVs) can predict changes to a Dependent Variable (DV). Using this method, we can also determine which IVs (if there are more than one) are the strongest predictors of the DV. This project focused primarily on Multiple Linear Regression (MLR), which is used for cases with multiple continuous or interval Independent Variables.

When examining the results of the linear regression analysis, we focused on several important summary statistics. The Square-Root of the Mean-Squared Error (RMSE) approximates the error associated with the model. RMSE is the square root of the average of squared differences between the predicted values and actual observations. It provides a gauge of the accuracy of the model, with a higher weight given to more severe error values.

Another statistic we examined is R-Squared (R2). The R-Squared value measures the proportion of the variance of the dependent variable that can be explained by changes to the independent variables. It can range from 0.0 to 1.0, with higher values meaning a greater correlation. A low R-Squared value could mean that there is a minimal correlation between the IVs and DV, but it could also just indicate noisy data. To determine which is the case, we examined the P-Values of each IV as well.

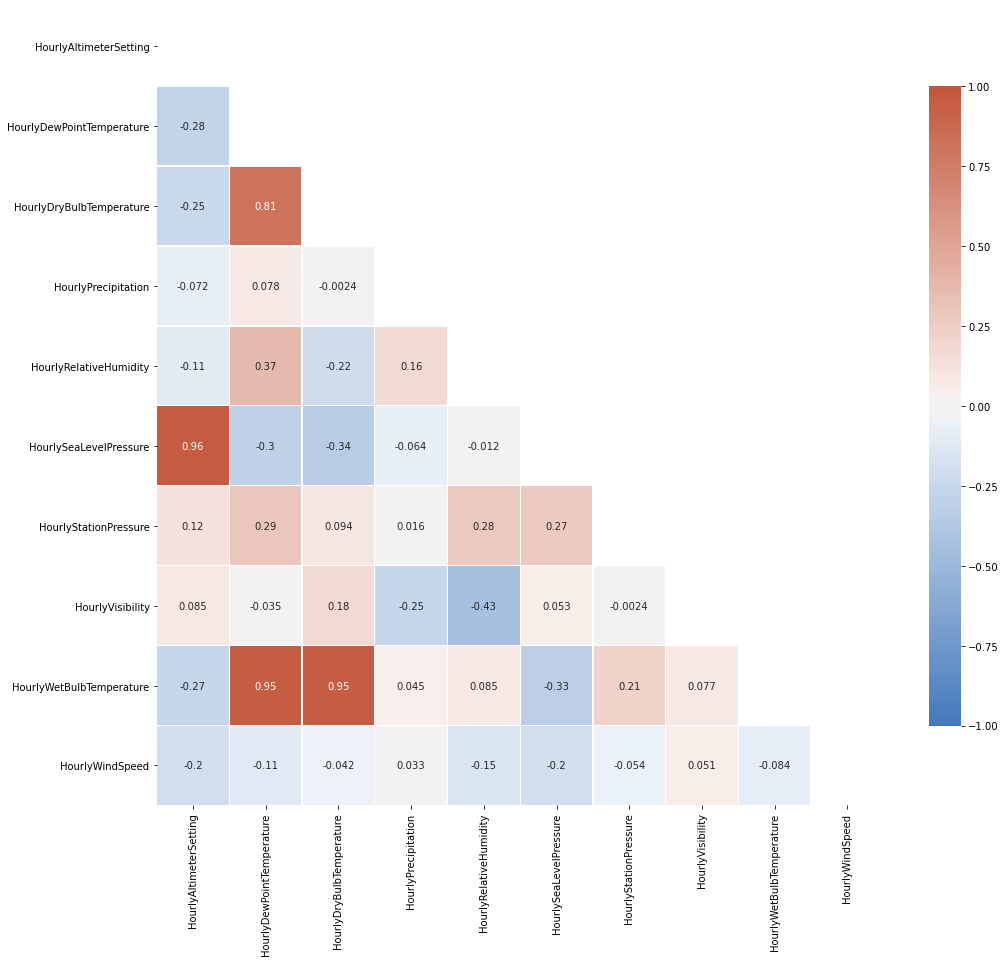
The P-Value represents the statistical significance of the correlation between each Independent Variable and the Dependent Variable. Smaller P-Values indicate a more statistically significant correlation. P-Values aren't as accurate in a multiple regression model than they are in a single variable linear regression model, because of the difficulty associated with disentangling the correlations between numerous variables. Nonetheless, they still provided a good general idea of which variables are worth using in our model, and which would not provide much value.

**3.2.2 Linear Regression Application**

The Linear Regression analysis for this data focused on a single Dependent Variable, “WeatherDelay,” which contained the number of minutes of flight delay that was attributed to weather conditions. A total of 4 variable selection iterations were performed. The dataset was first partitioned into a training set (75%) and a test set (25%) and the Ordinary Least Squares (OLS) Multiple Linear Regression model was fitted using the training data. The generated model was then evaluated using the test dataset. This process was repeated 10 times for each iteration, and the results were averaged across the 10 trials to produce the final results for that iteration.

At the end of each iteration, the results were examined to determine whether any variables should be removed from the dataset due to high correlation to other variables or low statistical significance. A combination of correlation matrices (pictured below) and the Variance Inflation Factor (VIF) score were used to determine multicollinearity. The correlation plot shows the degree and direction of correlation between each pair of IVs, ranging from -1.0 (full negative correlation, where increase in x results in a proportional decrease in y) to 1.0 (full positive correlation, where increase in x results in a proportional increase in y). The VIF is a measure of the strength of each Independent Variable's multicollinearity with all other IVs, with a VIF of 1.0 indicating no correlation to any other variables.

The benchmarks for what is considered “too much correlation” varies greatly across the industry, with recommended thresholds ranging from 0.3 (30%) to 0.9 (90%) for correlation matrices, and 2.5 to 10.0 for VIFs. This team chose to err on the conservative side by choosing to use 0.3 (30%) and 2.5 thresholds respectively. The results of our first iteration can be viewed in the correlation plot below:



*Figure 3.1. Linear Regression Correlation Matrix*

Based on Figure 3.1, two sets of weather data Independent Variables were deemed highly correlated. “HourlyAltimeterSetting” and “HourlySeaLevelPressure” comprised one set, while "HourlyDewPointTemperature,” “HourlyWetBulbTemperature,” and "HourlyDryBulbTemperature” comprised the other. For each set of correlated variables, we analyzed the P-Values and VIF scores of each variable to determine which one would be the most useful to include in the model. The other variables from that set were removed from the model.

Since the job of selecting variables for a model is more of an art than a science, It was decided to remove only 1 or 2 variables at a time, then re-ran the regression analysis, and used the results, and our best judgment, to determine which variables to remove with each iteration. After 4 iterations of this process, a total of 4 Independent Variables remained: HourlyPrecipitation, HourlyWindSpeed, HourlyStationPressure, and HourlyVisibility. A linear regression model was generated using these variables, and the model was tested using the partitioned test data subset. The summary statistics of the final 4 Independent Variables and the overall model are as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| **Independent Variable Name** | **Coefficient** | **P-Value** | **VIF** |
| HourlyPrecipitation | 69.748524 | 0.000000e+00 (virtually 0) | 1.069558 |
| HourlyWindSpeed | 0.078467 | 3.053695e-181 | 1.007899 |
| HourlyStationPressure | -0.059895 | 1.233015e-09 | 1.003287 |
| HourlyVisibility | -0.287497 | 3.823092e-302 | 1.070909 |

**Intercept (Constant):** 4.609745

**Final R-Squared:** 0.025982

**Final Square-Root of Mean-Squared-Error:** 15.807739

Each of the final variables had extremely low P-Values and VIF scores, meaning that they all had a statistically significant correlation to the weather delay time, and none of them had a substantial correlation to each other. Of all the variables, “HourlyPrecipitation” clearly had the biggest impact on “WeatherDelay.” With a coefficient of nearly 70, a single unit increase of “HourlyPrecipitation” (equivalent to 1 inch of rain) should be expected to cause an increased weather delay of approximately 70 minutes, or over an hour on average.

Other variables, such as “HourlyStationPressure,” had a smaller, but still meaningful correlation to “WeatherDelay.” Due to the negative correlation, a single unit increase of “StationPressure” (equivalent to 1 mmHg, an industry standard unit) should be expected to cause a decrease in weather delay of approximately 0.04 minutes (approximately 24 seconds) on average. This made sense, because high air pressure is associated with what most people consider “good” weather. Low pressure systems, on the other hand, typically lead to severe storms, which are a common reason for flight delays.

The correlation coefficients of “HourlyWindSpeed” and “HourlyVisibility” are similarly straightforward. As the wind speed increases, the delay also increases. High winds are a very common reason for flight delays, since they make take-offs and landings unsafe. Decreasing visibility also increases flight delays for the same reason.

The Independent Variable coefficients combined with their respective values, plus the intercept/constant, comprised the equation at the core of the linear regression model. Plugging values for each of the IVs into this equation will generate the prediction of the “WeatherDelay” (in minutes) for those conditions.

The final R-Squared value ended up being extremely low, at approximately 0.026, meaning that less than 3% of all variation in the dataset could be explained by the variables in the model. While this could theoretically mean that the variables included in the model were not statistically significantly correlated with the overall weather delay, this seemed unlikely given the extremely low p-values of all 4 variables. The more likely explanation was that there *was* a statistically significant correlation, but the dataset was extremely noisy due to the multitude of other confounding factors at play.

The finalSquare-Root of Mean-Squared-Error (RMSE) was 15.807739 minutes. Due to the way that RMSE is calculated, larger error values are penalized to a higher degree than smaller errors and contribute to a higher overall RMSE value. It was determined that this approach was ideal since a delay that is twice as long would normally be seen by the passenger as *more* than twice as inconvenient (i.e. 30-minute delay vs 1 hour delay). Therefore, it should be penalized more than twice as heavily. The final RMSE value was slightly lower than the initial value, meaning that the model comprised of the final set of only 4 variables was likely a more accurate predictor than the initial 10-variable model.

**3.2.3 Linear Regression Risks**

One of the challenges of Multiple Linear Regression is determining which Independent Variables to include in the model. Incorporating additional variables typically leads to an increase in the R-squared value but can also result in overfitting of the data and a less flexible model.

Another challenge of MLR is multicollinearity, or overly correlated Independent Variables. The correlation coefficient assigned to each IV in the model represents the magnitude of change to the DV that should, on average, result from a single unit increase in that specific IV. Independent Variables are assumed to be *independent* of each other. When this assumption is broken, and the value of one IV is in fact affected by changes in another, it becomes difficult to determine the effect of each. In this case, several irregularities can occur. These include an artificially inflated R-squared value, inaccurate correlation coefficient values, overfitting of the model, and incorrect rejection of the null hypothesis. Because of this, there are several steps that we need to take to ensure that all the variables used in the model are sufficiently independent of each other.

It is important to keep in mind that the linear regression model generated was nowhere near perfect. There could have been any number of additional factors that impacted weather delays, such as pilot experience, airline resources, or government regulations. Additionally, some of these variables may have contributed to a non-linear correlation, such as an exponential or logarithmic correlation. For example, one could reason that once the wind reaches a certain speed, each additional MPH increase in wind speed causes a smaller increase in weather delays. This could potentially be represented by a logarithmic regression, or possibly a 3rd order polynomial regression. Linear regression may not be the best fit for every variable.

Even so, linear regression analysis can provide useful insights about the effects of weather on airline delays. While the predictions that result from the model may not be extremely accurate, knowing how each individual Independent Variable affects Weather Delays could prove to be useful in developing future models and experiments.

## XGBoost

**3.3.1 Description**

XGBoost is a machine learning algorithm which stands for eXtreme Gradient Boosted trees. It uses an ensemble method. XGBoost is used for supervised learning problems, where we use the training data with multiple features to predict a target variable. The ensemble method applies multiple versions of a model chained together so that ever subsequent tree in the boosting scheme boosts the attributes that led to misclassifications and corrects the errors of the previous tree. The regularized boosting prevents overfitting. It can handle missing values and can run parallel across multiple threads. It supports incremental training. It also cross-validates at each iteration to evaluate the performance of the algorithm at each step of its training to find the optimal number of iterations. It is also very flexible in that it allows customization of optimization objectives. Finally, its tree pruning feature results in a deeper, but optimized analysis. It uses an interface, a DMatrix structure to hold features and labels. All paramaters are passed in via a dictionary (XGBoost, 2020).

**3.3.2 XGBoost Application**

XGBoost is an open-source software library designed to be highly efficient, flexible and portable. It implements machine learning algorithms under the Gradient Boosting framework. In order to use XGBoost, the library was installed into python and executed on Jupyter Notebook. Python syntax is clear, logical and easy to understand with the support from its comprehensive standard libraries. XGBoost application on the analysis was performed and tested by the project’s developers. In this case, there was no need to place the code under any kind of version control, it was simply put on a shared 365 drive for the direct purpose of storing the project.

In this project, XGBoost was used as the Machine Learning algorithm for classification and regression problems. The objective of this project was to determine whether a flight was delayed and how many minutes it would be delayed based on weather conditions.

To analyze the performance of XGBoost on solving classification predicting model problems, labels for a binary classification and a quintuplet classification were used. A labeled dataset, which was a subset from final dataset was produced which included a binary class variable (Label 1) and a quintuplet class variable (Label3):

* + ‘Label1’ - binary classification with 2 values
    - ‘ontime’: at or before scheduled departure time
    - ‘delay’: any departure after scheduled departure time
  + ‘Label3’ - quintuplet classification with 5 values
    - ‘ontime’: at or before scheduled departure time
    - ‘delay’: delay less than or equal to 15 minutes
    - ‘delay>15’: delay more than 15 minutes but less than or equal to 30 minutes
    - ‘delay>30’: delay more than 30 minutes but less than or equal to 60 minutes
    - ‘delay>60’: delay more than 60 minutes

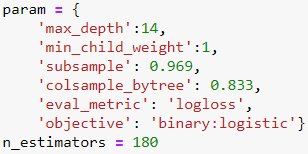
Before running the XGBoost model, pre-processing the labeled dataset needed to be done. First, the target variable was converted to categorical features in numeric format. Second, in order to evaluate the performance of the model, the labeled dataset was split into training and testing data with an 80:20 ratio.

Additionally, since more than 90% of the initial dataset consisted of “ontime” flights, it is suspected that the imbalance between the two labels would affect the model accuracy. To prove that, the XGBoost model was performed on another derived training dataset with the minority class increased to balance the weight between “ontime” and “delay”. The testing dataset remained the same in both iterations.

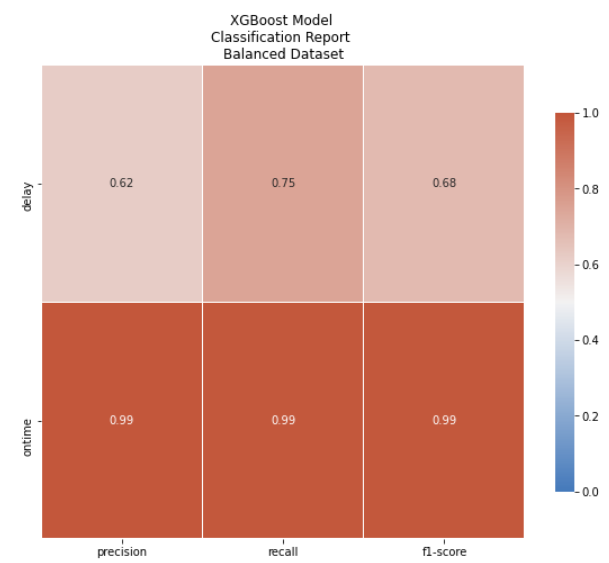
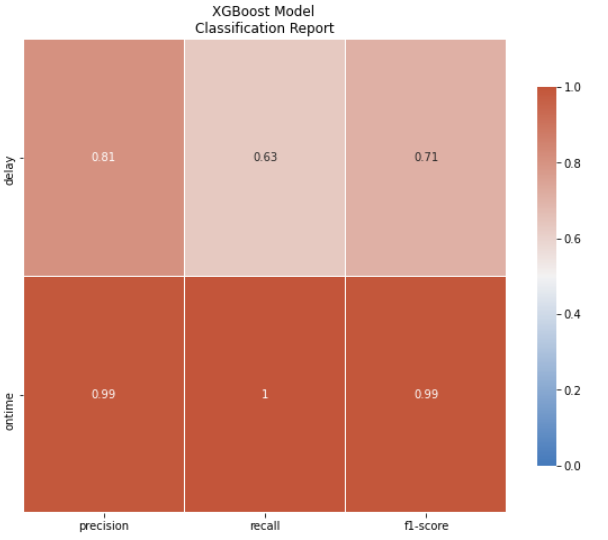
One more important task needed to be done with the XGBoost. That was tuning the hyperparameters prior to the algorithm’s execution to achieve optimal performance. The tuning approach used in this project is Bayesian optimization using Hypeopt. Hypeopt is a powerful python library that search through a hyperparameter space of values and find the best possible values that yield the minimum of the loss function (Bergstra, 2013). Initially, GridSearchCV, a library function that is a member of sklearn's model\_selection package, was used for the hyperparameters tuning process. However, grid search method was extremely time consuming. It ran on one teammember’s laptop for more than 24 hours with no results. In contrast, the Bayesian optimization returned the optimal hyperparameter set in less than an hour for the binary classification and in around 3 hours for the multiclass classification. The tuned hyperparameter set increased the training process performance significantly in terms of speed and accuracy.

* 1. **Binary Classification**

To find the optimal hyperparameter sets for binary classification, Bayesian optimization using Hypeopt will find the minimum value of the validation error of a machine learning model with respect to the hyperparameters. In this case, validation error of binary classification is logloss. The tuned hyperparameters for XGBoost binary classification model shows in Figure 3.2.

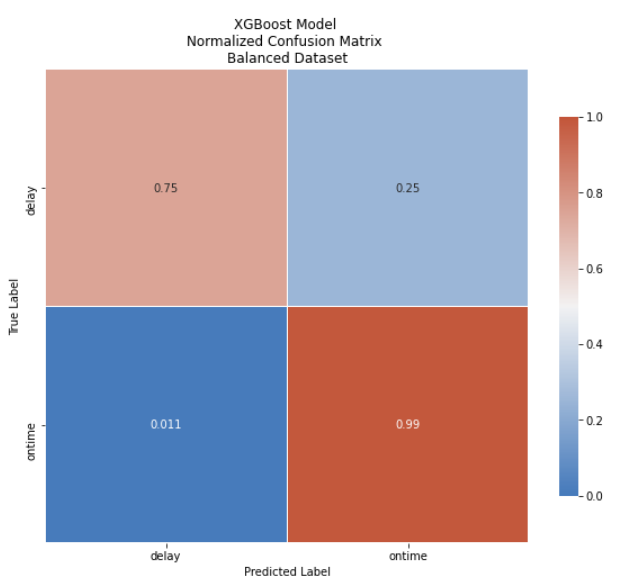


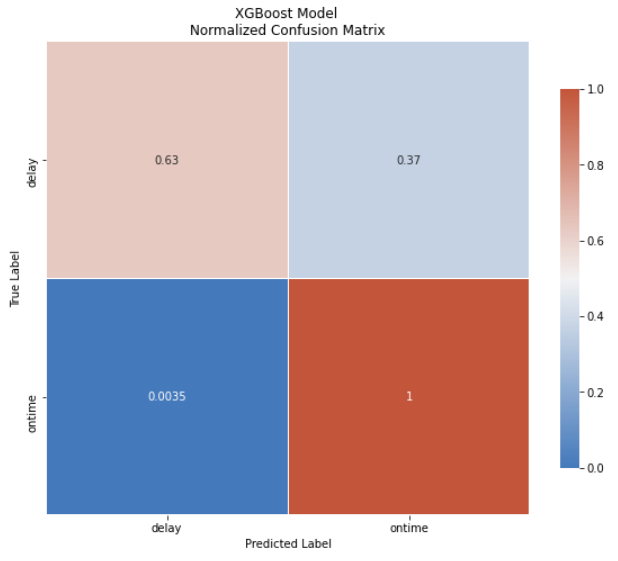
*Figure 3.2. Tuned hyperparameters for binary classification model*

Two iterations were performed. The first one was to run the model on training dataset extracted from the labeled dataset. The second one executed the same procedure, but with a training dataset that had been balanced between the two labels. As shown in Figure 3.3. and Figure 3.4., the classification reports from both iterations show similar results. For this particular problem, recall score would tell the how many actual “delay” flight records can be caught by the model and it was considered the evaluation score for XGBoost classification model. According to the below heatmaps, it showed the higher recall score (0.75) in “delay” label when taking the imbalance of the classes into account.

*Figure 3.3. Classification Reports of XGBoost binary classification model from*

*Initial Labeled Dataset vs. Balanced Labeled Dataset*

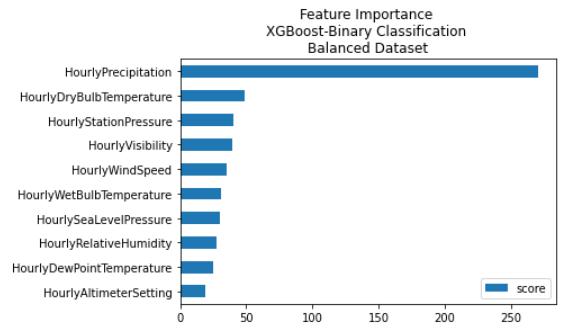
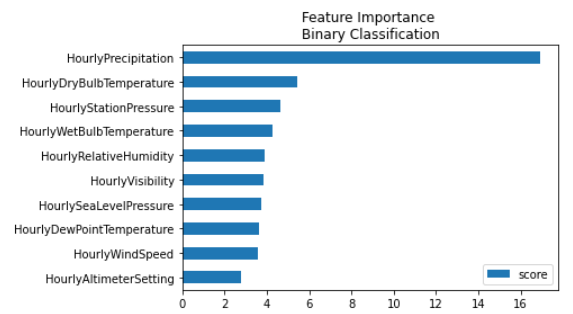




*Figure 3.4. Confusion Matrix of XGBoost binary classification model from*

*Initial Labeled Dataset vs. Balanced Labeled Dataset*

The feature importance type used in this study for XGBoost was “Gain”. It is the improvement in accuracy brought by a feature to the decision tree branches it is on. There were some differences between the two iterations when it came to feature importance. However, the top 4 independent variables that have great influence on flight performance are still "HourlyPrecipitation", "HourlyDryBulbTemperature", and "HourlyStationPressure" with slight difference in ranking. These can be seen in figure 3.5. below.

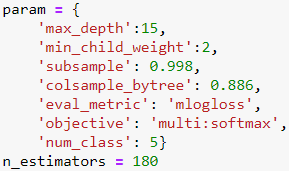


*Figure 3.5. Feature Importance*

*Initial Labeled Dataset vs. Balanced Labeled Dataset*

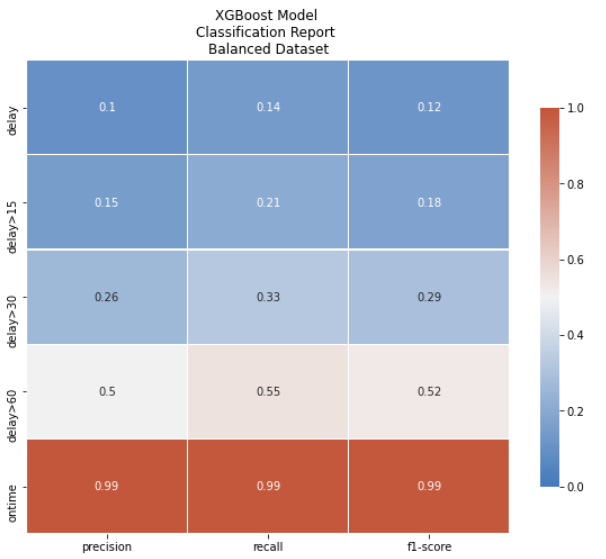
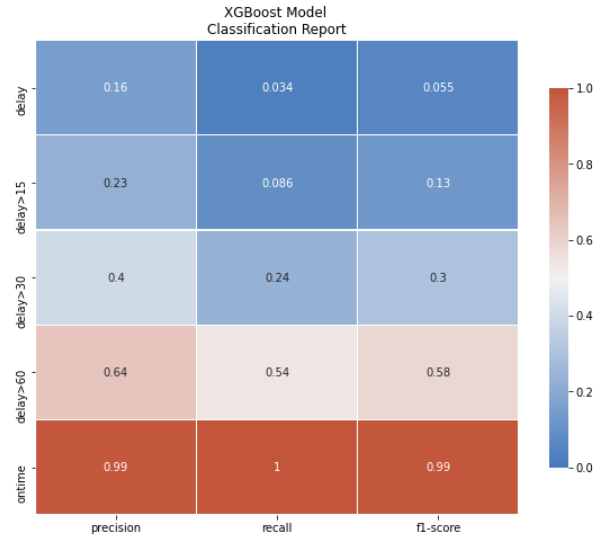
* 1. **Multiclass Classification**

Similar procedure to binary classification was performed with multiclass classification on XGBoost model. The validation error for multiclass classification was mlogloss. The tuned hyperparameters for XGBoost multiclass classification model is shown in Figure 3.6.



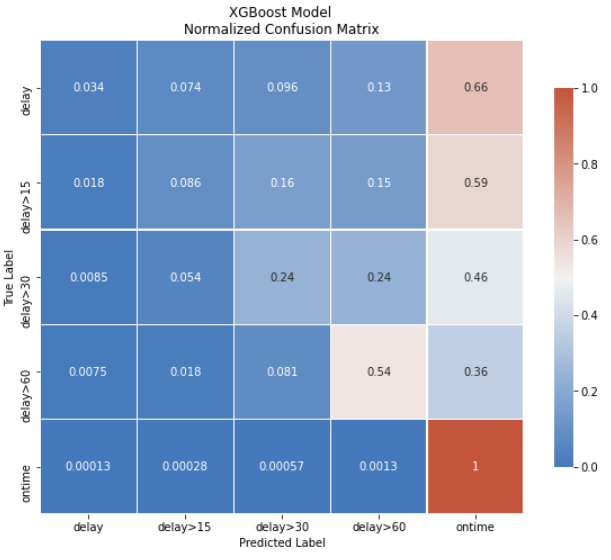
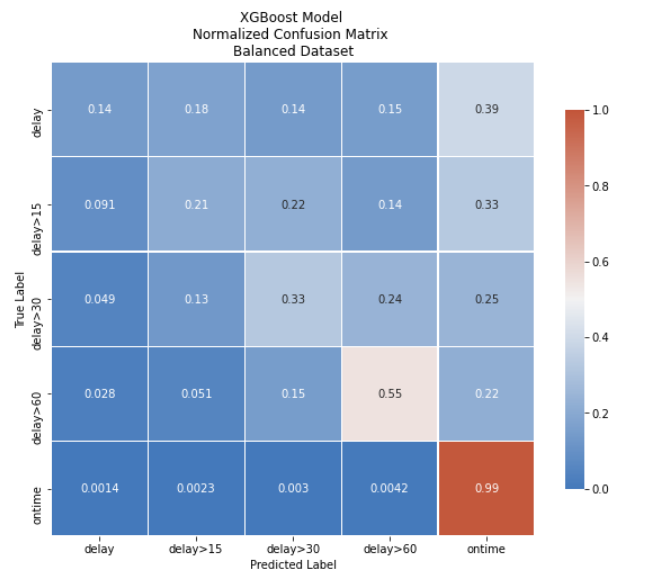
*Figure 3.6. Tuned hyperparameters for binary classification model*

The classification reports (Figure 3.7.) from iterations of imbalanced and balanced training data show similar results. Yet, one can see the prediction of “delay”, “delay>15”, delay>30” have poor performance with low recall scores, less than 0.5. Model prediction on “delay>60” label had recall score at 0.5-0.6 scores which was the highest among the delay labels but it could not be considered as decent performance. Prediction of “ontime” flights still showed high accuracy. In accordance with the classification reports, it was not surprising to see low scores on all delay labels. However, the performance of “ontime” prediction improved noticeably when running the model on the balanced training dataset. The false prediction of “ontime” was less than the iteration of the imbalanced training dataset.



*Figure 3.7. Classification Reports of XGBoost multiclass classification model from Initial Labeled Dataset vs. Balanced Labeled Dataset*

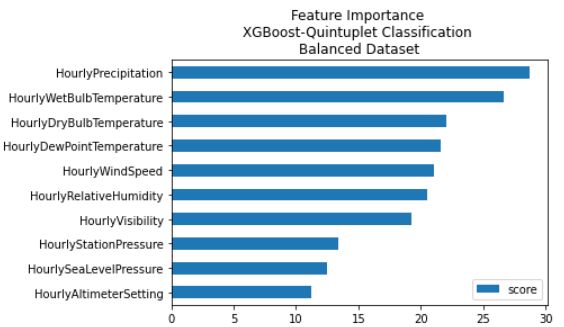
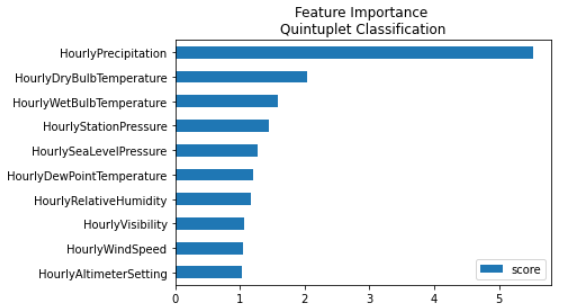
*Initial Labeled Dataset vs. Balanced Labeled Dataset*



*Figure 3.8. Confusion Matrix of XGBoost multiclass classification model from*

*Initial Labeled Dataset vs. Balanced Labeled Dataset*

For multiclass classification, the three most influential weather variables on flight performance were "HourlyPrecipitation", "HourlyDryBulbTemperature", and " HourlyWetBulbTemperature " with a slight difference in ranking. This can be seen in figure 3.9. below.



*Figure 3.9. Feature Importance*

*Initial Labeled Dataset vs. Balanced Labeled Dataset*

* 1. **Regression on XGBoost**

To compare the performance of XGBoost on regression problem of this project, which was to predict the amount of minutes flight delay based on weather conditions, to Linear Regression analysis in section 3.2, same evaluation metric (RMSE) and accuracy rate (R-Squared) were used when running the model. XGBoost Regression would have the same setting and applied procedure with its classification problems. The only difference was the objectives parameter to set as “Reg: linear”. The result was shown the table below:

|  |  |  |
| --- | --- | --- |
|  | **XGBoost Regression** | **Linear Regression** |
| **R-Squared** | 0.1175 | 0.025982 |
| **RMSE** | 14.95536 | 15.807739 |

XGBoost had a slight better performance compared with Linear Regression. However, both algorithms showed low accuracy rate for predicting the amount of delay time in minutes.

**3.3.3 Risks Associated with the XGBoost**

XGBoost is one of the leading algorithms in data science giving unparalleled performance which can solve a variety of problems; from regression, classification, and ranking to any user-defined prediction. However, XGBoost has a lot of hyperparameters that must be tuned to achieve optimal performance. The tuning process can take from half an hour to many hours or even days to find the best hyperparameter set. After the optimal hyperparameter set is determined, the XGBoost model performance improves significantly. Therefore, the tuning process cannot be excluded when running the XGBoost model.

## Random Forest

**3.4.1 Description**

The Random Forest algorithm is an ensemble learning method. Its name comes from the concept of its creation of a “forest” of multiple decision trees which is combined in order to create a prediction. It is often celebrated for its simplicity and versatility, as it can be used for both regression and classification models. The “random” in random forest comes from the way it grows the classification trees. The model does not select the most important or strongest feature. It instead takes a random sampling of the features and selects the best one from that subset. This, in turn, increases the range of potential outcomes (Donges, 2019).

**3.4.2 Random Forest Application on the dataset**

Python and Jupyter Notebook were used to develop the random forest model and analyze the combined flight and weather data. The various libraries and modules in Python greatly reduced the complexity of the code. Jupyter Notebook provided us with an interactive environment to easily test and edit code. It allows files to be duplicated for version control and to easily build multiple models in which minor tweaks could be made.

The same 10 variables listed in the introduction of this section were initially used. The focus for this model was the dependent variable Label1, the binary outcome scenario. The dataset was split into a 70/30 training to testing split to evaluate the performance of the model and to prevent overfitting.

To analyze the performance of the Random Forest model, 5 different approaches were taken into account:

**Round 1 – No Parameter Adjustment:** This is the initial round using default parameters of RandomForestClassifier function to test the performance of Random Forest model without any modification in the parameters

**Round 2 – Balanced Class Weight:** Since more than 97% in the dataset was “ontime” flight records, the imbalance between classes can impact the model performance. This round aimed to examine the class imbalance influence by two different methods: changing the class\_weight hyperparameter of the RandomForestClassifier from class\_weight=None to class\_weight = ‘balanced’ and using SMOTE (Synthetic Minority Oversampling Technique) function from the imblearn library. The function will synthesize new examples for the minority class which is the delay label in this case.

**Round 3 – Removing Redundant Columns:** Collecting from Section 3.2.2, there were several pairs of weather variables that had a high degree of correlation (over 0.9). This included HourlyAltimeterSetting and HourlySealLevelPressure, HourlyDewPointTemperature, HourlyDryBulbTemperature and HourlyWetBulbTemperature. Those 5 variables were removed from the analyzed dataset to determine if variables with high correlation have any impact on the model run in Round 1.

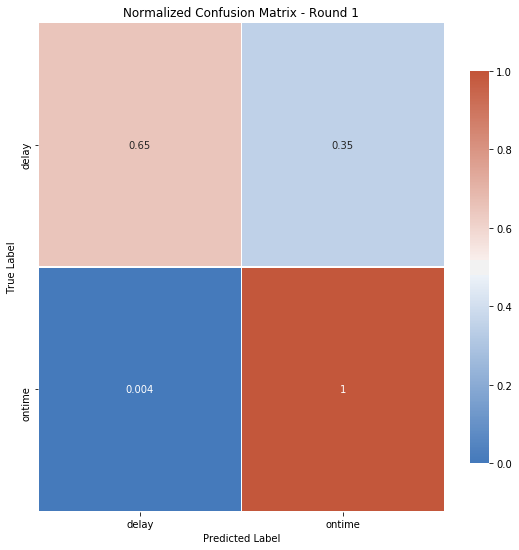
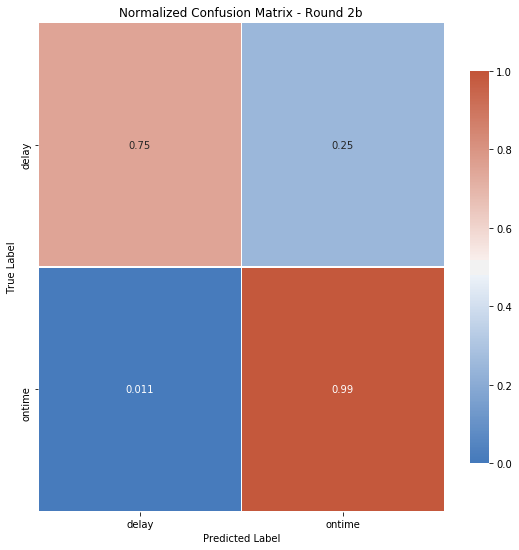
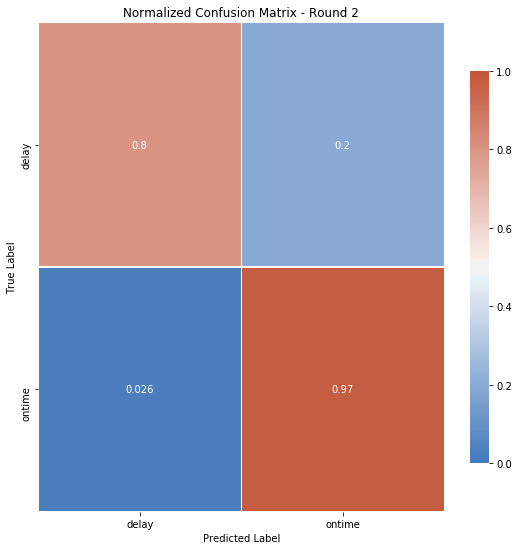
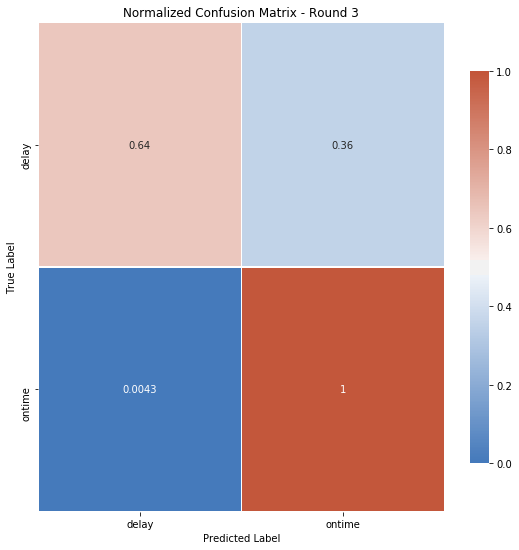
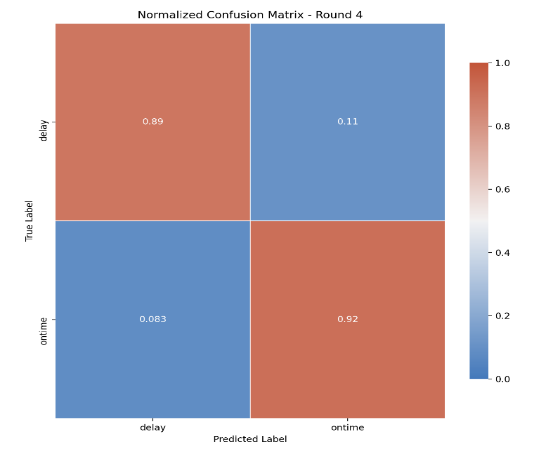
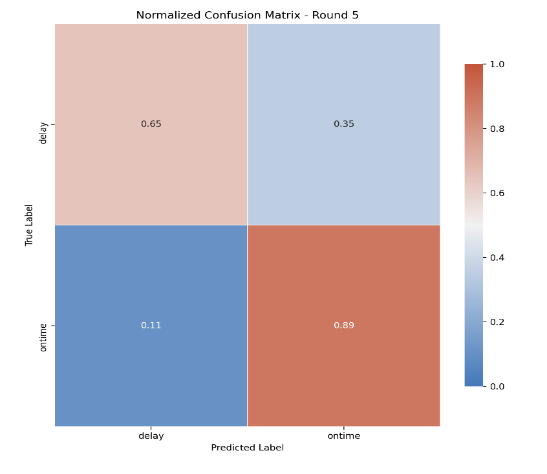
**Round 4 – Balanced Random Forest Classifier:** As taking the imbalance of the classes into account, the model parameters needed tuning in such way to deal with it. A different random forest algorithm named Balanced Random Forest Classifier was found. It resamples the data on the bootstrap to explicitly change the class distribution for the purpose of creating a synthetic balance in the dataset, which down-sampled the majority class. The first run of the Balanced Random Forest used the default parameters to see how the performance of the model compared to the Random Forest.

**Round 5 – Tuning Balanced Random Forest Parameters:** The randomized search cross validation method was chosen over the grid search to search for optimal parameters within given bounds for a few parameters because of its performance and the opportunity to fit the parameters over a larger range. Performance is important because the team had limited computing power and time to run these algorithms. A grid search would cycle through each possible combination, which would take away more time from performing other analysis tasks. Using the randomized search allowed for a larger range of values to iterate over while also limiting the number of model iterations.

The classification results from running Random Forest models in each iteration were shown in the below table.

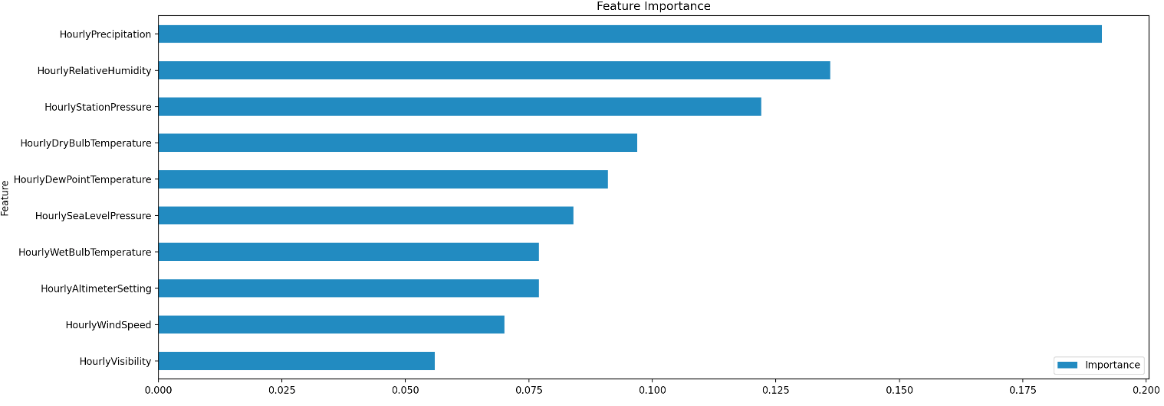
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Round 1** | **Round 2** | **Round 2b** | **Round 3** | **Round 4** | **Round 5** |
| **Accuracy Score** | 0.98789 | 0.96942 | 0.98332 | 0.98781 | 0.91819 | 0.88781 |
| **“Ontime” Precision score** | 0.99 | 1.00 | 0.99 | 0.99 | 1.00 | 0.99 |
| **“Ontime” Recall score** | 1.00 | 0.97 | 0.99 | 1.00 | 0.92 | 0.89 |
| **“Ontime” f1 score** | 0.99 | 0.98 | 0.99 | 0.99 | 0.96 | 0.94 |
| **“Delay” Precision score** | 0.78 | 0.42 | 0.61 | 0.79 | 0.2 | 0.13 |
| **“Delay” Recall score** | 0.65 | 0.80 | 0.76 | 0.65 | 0.89 | 0.65 |
| **“Delay” f1 score** | 0.71 | 0.55 | 0.68 | 0.71 | 0.33 | 0.21 |

Random Forest Model shows high accuracy score for all 5 rounds. It can be explained by the major dominance of the majority “ontime” class in the dataset as it made up more than 90% if the data. The precision, recall and f1 scores for “ontime” flights are consistent throughout all the trials, regardless of the imbalance influence between classes. However, there was a fluctuation in those scores of “delay” flight records. The higher the recall scores were, the better performance for the Random Forest model in predicting true positives. Therefore, round 4 iteration with the highest “delay” recall score (0.89) gave the best performance. The next iteration that also had a high recall score was round 2 (0.8 and 0.76). It implied that the imbalance classes can have a significant impact on model performance since those trials had taken imbalance between classes into consideration.



*Figure 3.10. Random Forest Confustion Matrices per round*

In Figure 3.10. results show that in both round 2 and round 4 iterations, the four most important features having great influence on flight performance were ‘HourlyPrecipitation’, ‘HourlyRelativeHumidity’, ‘HourlyStationPressure’, HourlyDryBulkTemperature’ in respective order.



*Figure 3.11. Random Forest Feature Importance Results*

**3.4.3 Random Forest Risk**

The primary risk with the Random Forest algorithm is overfitting of data. As we saw in our models, the ontime departures were often predicted 95 - 100% of the time. Due to the discrepancy in the number of ontime departures and delayed departures, the bootstrap samples the algorithm will be trained on will lean towards the label that is in the majority. As we continued to fine tune the hyperparameters of the model, the performance of the delay label prediction tended to decrease as we tried to account for the imbalance.

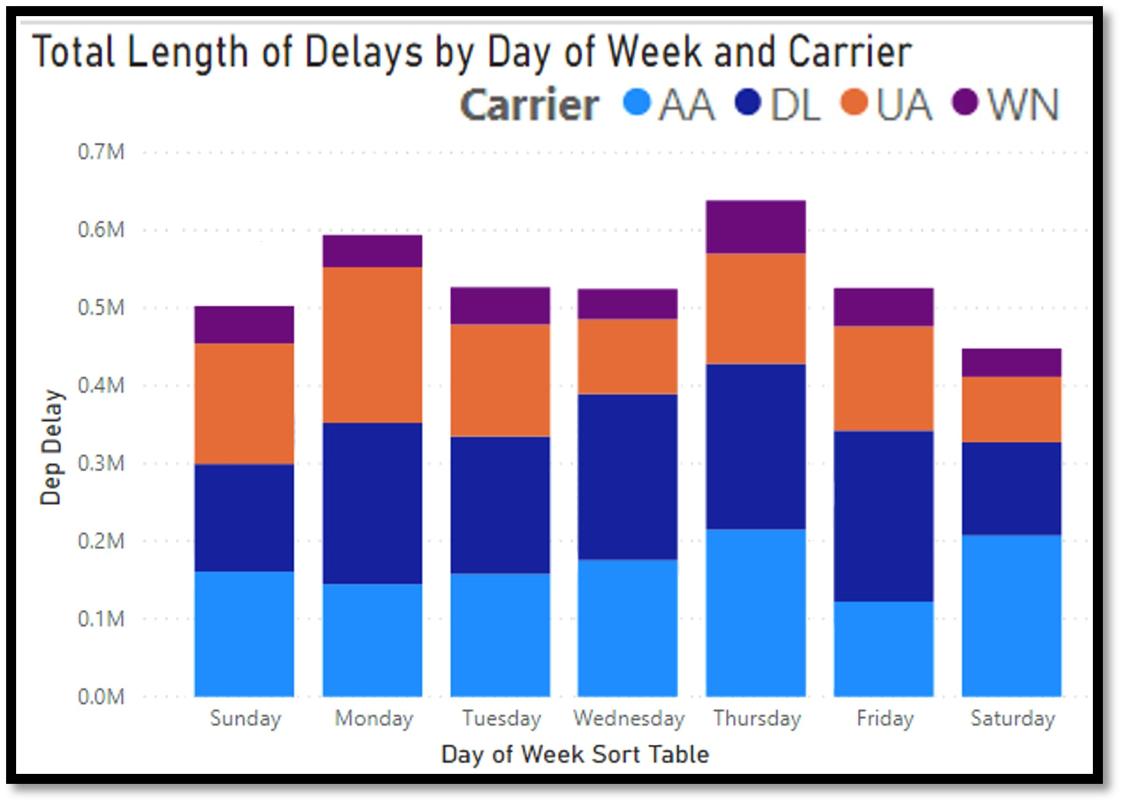
**3.4.4 Random Forest Conclusion**

In terms of accuracy, the 6 models were mostly accurate, routinely performing in the high 90s in terms of accuracy. But again, this was mostly due to the great prediction for on-time departures. In terms of performance for the delayed departures (by f1 score), the team was unable to improve their results from the default model, which had the highest f1 score, at 0.71. The highest recall score for came from the default Balanced Random Forest algorithm, at 0.89. Overall, the team’s original model proved the most accurate and achieved the best f1 score for predicting delays with the default Balanced Random Forest being the most accurate for the minority class prediction.

# Visualization

All of the following visuals were created in PowerBI. PowerBI is a Microsoft-owned analytics service that allows users to create interactive dashboards and reports with data visualizations, share them on the web, and perform business intelligence analytics using a graphical user interface (GUI). PowerBI was released in 2014 but has undergone major updates, including the addition of virtual communities that allow for the sharing of visualization templates and dashboards that users have created. There are free trials and free versions of PowerBI, but Microsoft also offers personal and enterprise licenses at various price points. PowerBI visuals are based in the coding language SQL, and SQL queries can be embedded into PowerBI queries to sort, filter, and transform the data. PowerBI is a fairly simple tool to use, but has several options for increasing the complexity, drilling down into datasets, and using the tool to its full potential.

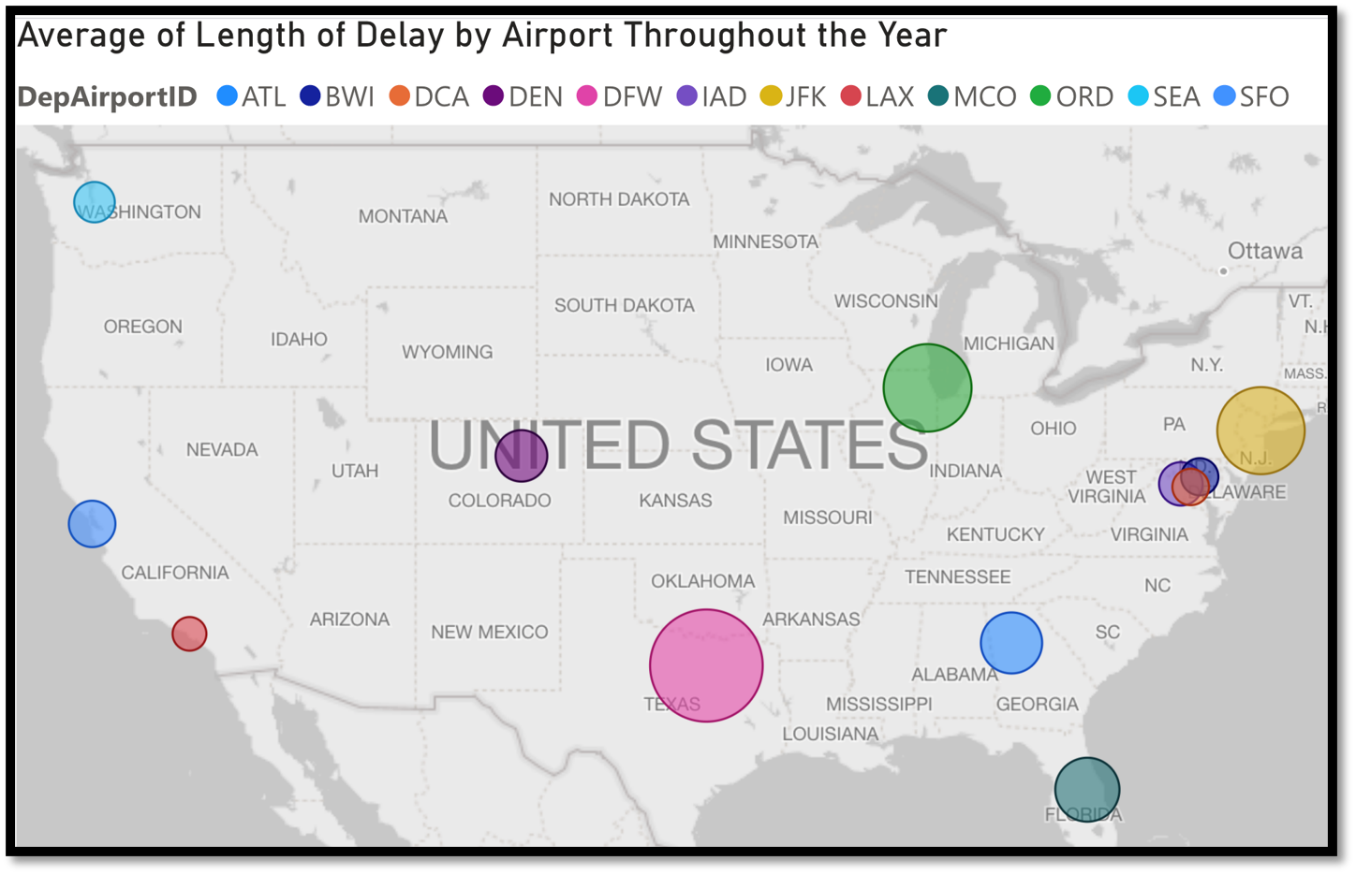
**Total Length of Delay by Day of Week and Carrier**



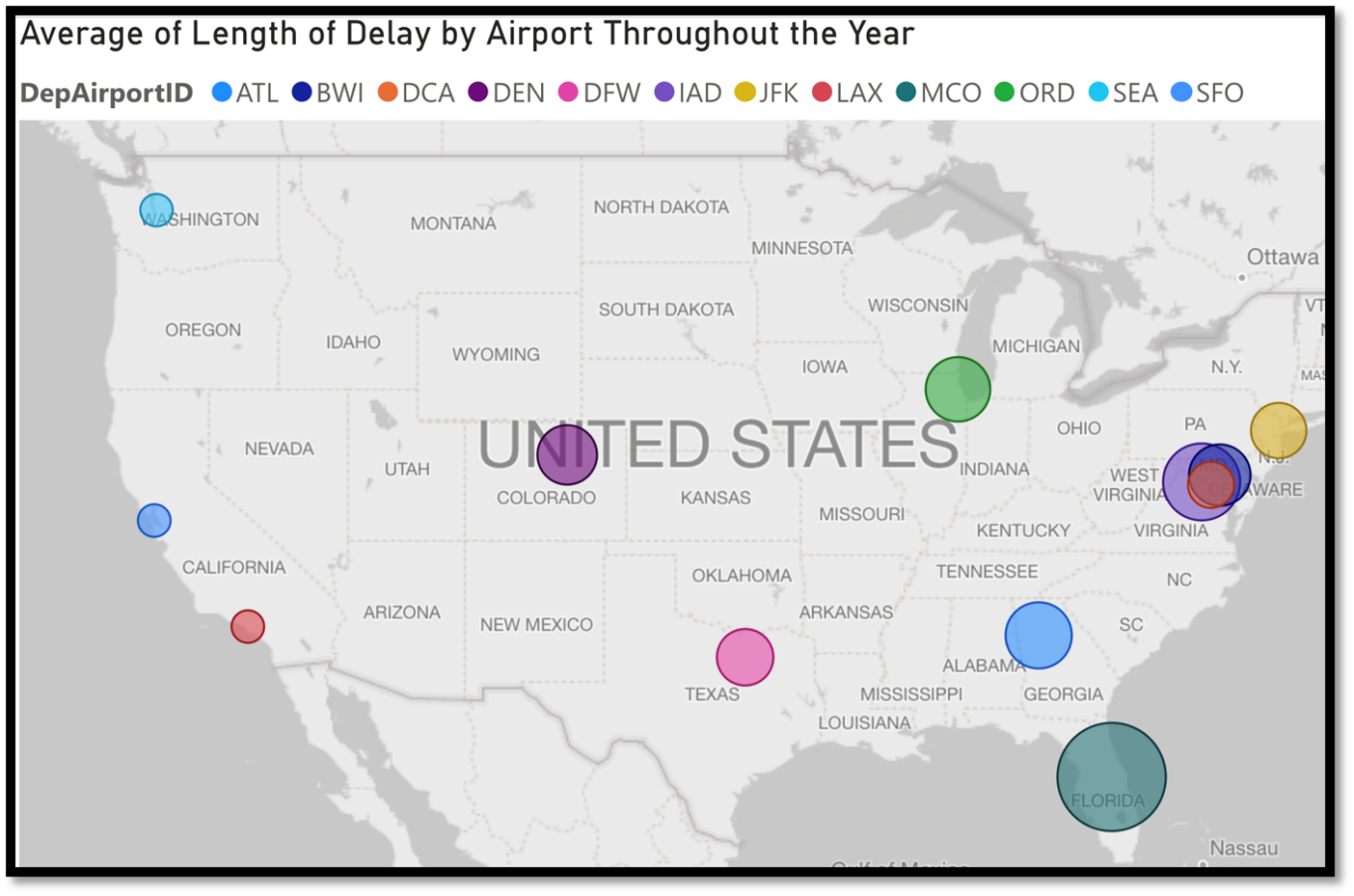
*Figure 4.1. Total Length of Delays by Day of Week and Carrier*

Figure 4.1. shows the sum of all delays for each carrier separated by the day of the week. This includes all delayed flights, and their total delay time has been added together for the entirety of the two-year period of analysis.

Risks associated with this visualization would be that this visual could be misinterpreted to be representing the number of the delays by day of week, which actually shows much less variation from day to day than the visual above. This would cause the viewer to assume that they are simply more likely to experience a flight delay of any kind on Thursdays than on Fridays. Instead, by looking at the sum of the length of delays, this shows how dramatically a flight may be delayed and what the likelihood of having an extremely delayed flight might be. Although the probability of this misinterpretation may be reduced with proper labeling, many viewers may take it at face value and see the columns as representations of likelihood instead of a combination of likelihood and intensity of delay. This risk is of low status and has been mitigated by proper titling.



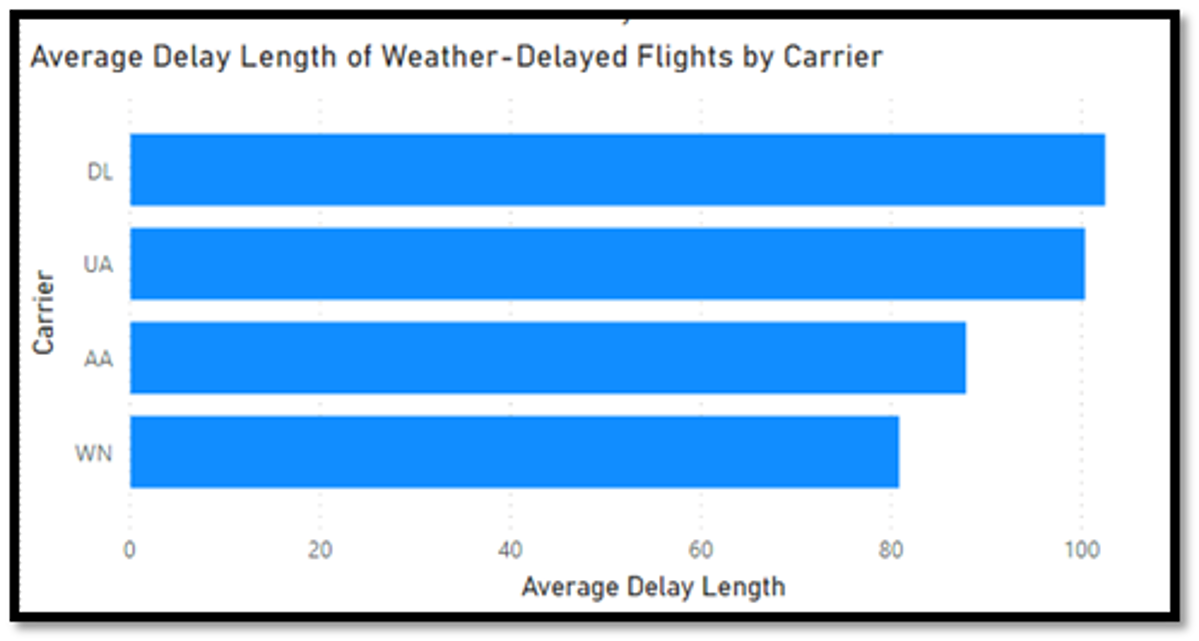
*Figure 4.2. Average Length of Delay by Airport Throughout the Year (July)*



*Figure 4.3. Average Length of Delay by Airport Throughout the Year (December)*

Figures 4.2. and 4.3. show the average length of delay by airports across the country, first in the month of July and second in the month of December. As shown, flights in northern states (JFK-New York, SEA-Washington, ORD-Illinois) experienced increase delays in the month of December. MCO-Florida had a noticeable increase in delays in July, likely due to hurricane season and stronger winds.

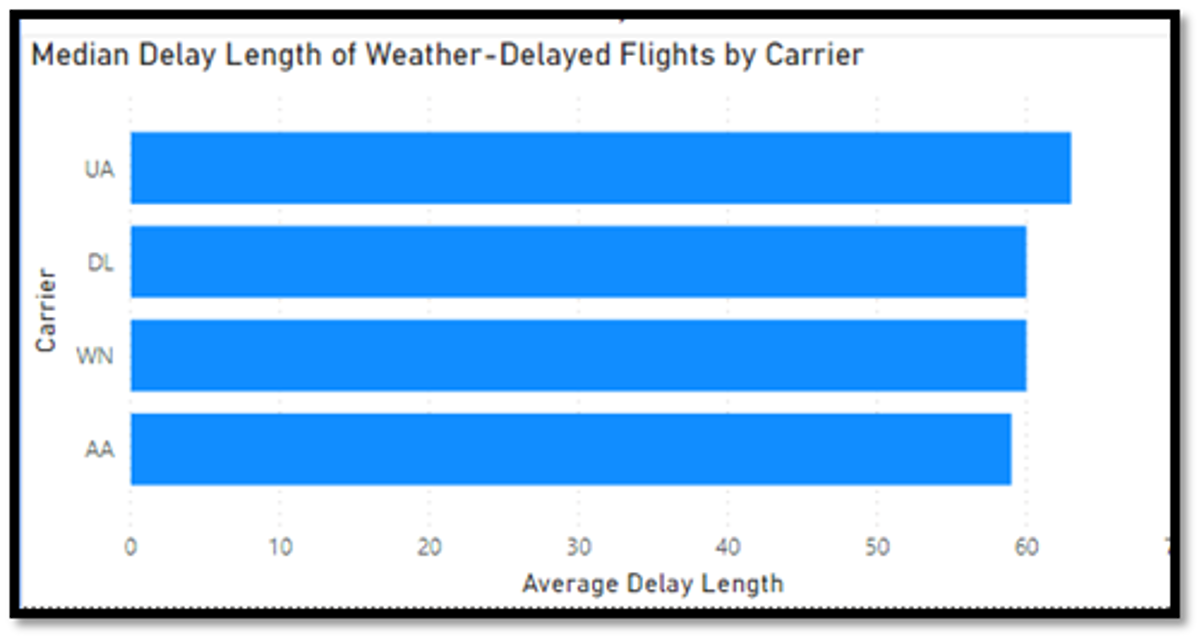
A risk involved with the visuals displayed in this comparison is that the differences from the summer to winter months could be misconstrued and given too much weight in the analysis. Flights are not necessarily more delayed during the winter and summer due to winds and precipitation or snow. Having a packed flight schedule could possibly lead to an increase in delays and delay length, and if certain airports are busier during different seasons, this could drastically impact flight timeliness if only one flight is delayed due to weather and sets off a chain of delayed flights. However, if people attribute these statistical differences to being strictly due to weather to being general seasonal trends, they could try to make predictions that are not based in reality. The probability of this risk is moderate, as human nature is likely to lead people to find causation where there is correlation. This risk is mitigated with labeling “Average Length of Delay” and not “Average Delays” or “Amount of Delays.”



*Figure 4.4. Average Length of Delays by Carrier*

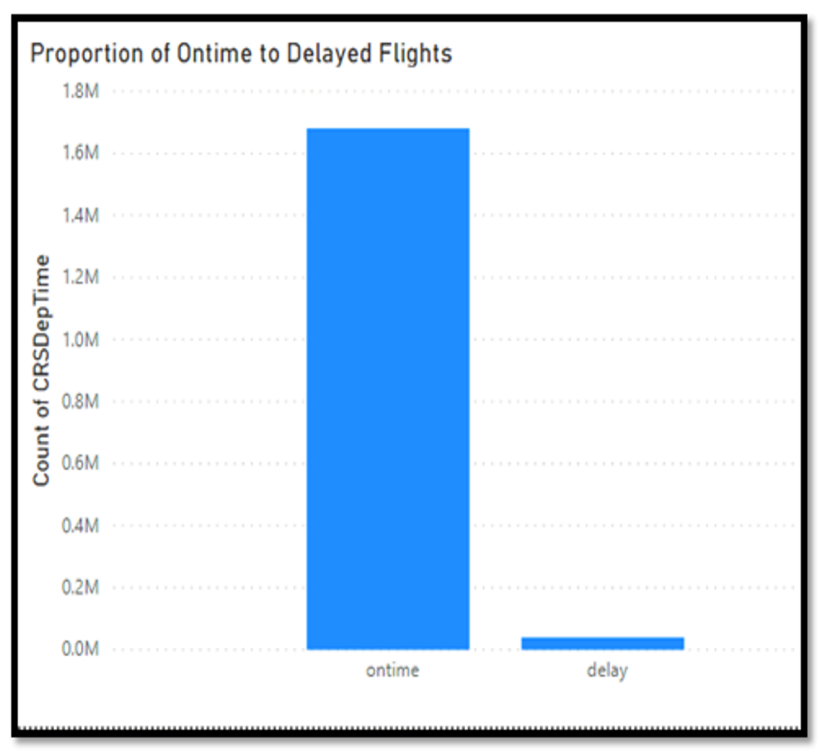
Figure 4.4. shows the average length of weather-delayed flights separated by carrier throughout the 2-year period of analysis. As shown above, when Delta flights experienced weather delays, those delays had the longest average delay length of over 100 minutes. Close behind was United Airlines, with an average weather-delay length of 100 minutes. Southwest had the lowest average flight delay due to weather at just over 80 minutes, with American Airlines at just under 90 minutes.

Risks associated with this visual are that there were significant outliers with extremely long delays due to weather in flight departure that skew the average length of delays. When the median value was used instead of the mean, the visual changed to show the graph below.



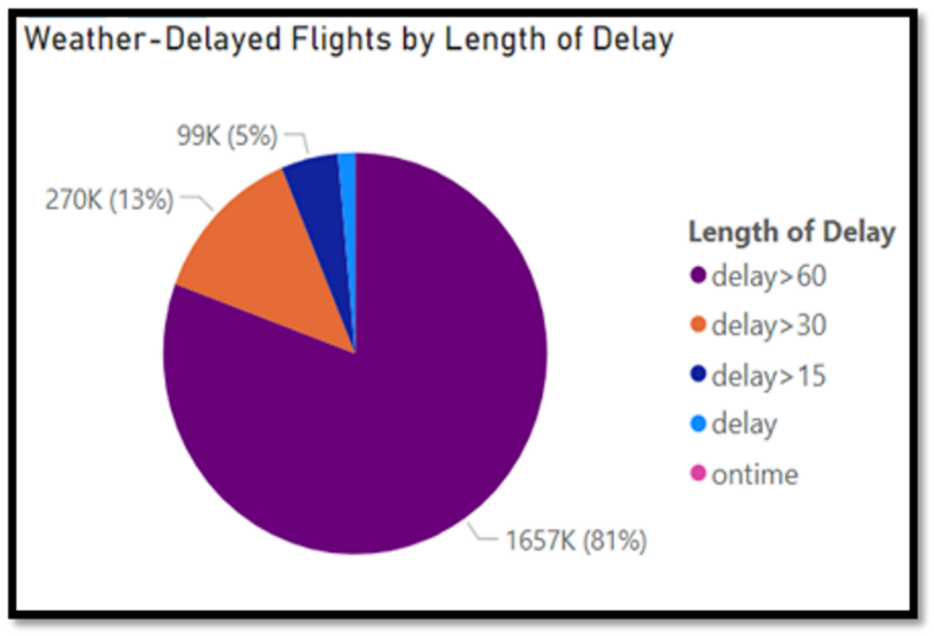
*Figure 4.5. Median Delay Length of Weather-Delayed Flights by Carrier*

Figure 4.5. shows the median length of weather delay by carrier. This change results in United Airlines having the longest delays due to weather, at a median value of 63 minutes. Using the median value, American Airlines was shown to have the shortest median length of delays, although all 4 carriers had very close median values. These visuals should not be used to draw the conclusion that some carriers experience longer delays due to weather for any management or systemic reasons. The risk of these assumptions is high, however, as many viewers may be looking to choose an airline based on low delays. As the two visuals above show, there is not a strong correlation between length of weather-caused delays and different airline carriers.



*Figure 4.6. Proportion of On-time to Delayed Flights in Original Dataset*

Figure 4.6. shows the original proportion of on time to delayed flights in the original dataset. It is obvious that the dataset was extremely unbalanced, which caused the need for the team to balance the dataset before beginning analysis. If analysis were to be performed on predicting flight delays with this unbalanced dataset, any models that were created could easily have high accuracy by simply predicting that all flights will be on time regardless of weather conditions, because the vast majority of flights in this dataset departed the airport on time. In the analysis process, these risks were mitigated by using XGBoost and Random Forest and by assigning entries class weights so that the models would account for the delayed flight conditions equally to the on time flight conditions. The probability of having an unbalanced dataset to analyze is quite high in these types of scenarios, as the “normal” condition is likely to have many more instances than the “abnormal” condition. For example, if an analysis that aimed to predict whether or not students would fail a class, the dataset would likely be unbalanced because most students would typically earn a passing grade. By assigning the failing student data extra weight in the model, a more accurate prediction model can be created.



*Figure 4.7. Weather-Delayed Flights by Length of Delay*

Figure 4.7. is a pie chart of all weather-delayed flights, and the breakdown of the length of those delays. As shown above, a large majority of delays caused an over 60-minute hindrance in take-off time. 84% of weather-delayed flights were delayed by over 30 minutes. No on time flights are present in this visual in order to focus specifically on flights that had weather noted as the cause of the delay, and all on time entries were excluded from the pie chart as they would overwhelm the delayed flight instances and would not show enough detail. These flights were separated by the previously established Label 3, which separated the delayed flights by intensity of the delay.

Risks associated with this visualization are that it may be misinterpreted to represent the average length of all flight delays instead of only representing flights delayed due to poor weather. This visual also does not tell the user the likelihood of having a delayed flight and should not be interpreted to represent causation of longer or shorter delays due to any specific weather conditions. If this chart is misinterpreted, it could cause viewers to make faulty assumptions about correlations in the data that do not exist. Not all flight delays would result in this long of a delay, as other types of delays (flight crew, takeoff scheduling, etc.) could be much shorter. This risk is mitigated by proper labeling and is best avoided by providing an explanation along with the visual explaining its purpose and shortcomings.

# Findings

This study focused on three types of models for analysis: Linear Regression, XGBoost and Random Forest. The objectives of this project were to determine the performance of each algorithm on predicting whether a departure flight would be delayed and the amount of delay time in minutes based on weather conditions at the departure airport. Each method was also used to determine which weather variables had the greatest impact on flight delays.

Based on these objectives, there were two types of issues to be analyzed by the three models: regression and classification. For the regression problem which was to predict the amount of delay time in minutes, this was addressed by the analysis of Linear Regression and Regression on XGBoost. Random Forest and XGBoost were used to analyze the binary classification problem about the prediction of “ontime” or “delay” on flight performance. More than that, a multiclass classification was examined by XGBoost to determine whether XGBoost could give better prediction results on the multiclass classification versus the binary one.

## Regression Analysis Findings

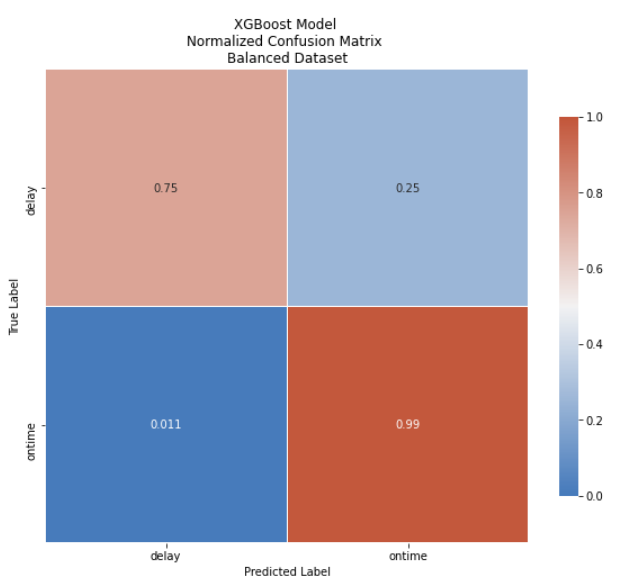
In terms of linear regression algorithms, XGBoost gave slightly better prediction than traditional linear regression. However, both models showed poor overall performance when it came to predicting the amount of delay time in minutes, as indicated by the low R-Squared values. This may have been caused by the dependent variable used in the models, “WeatherDelay.” This variable represented the amount of flight delay time caused solely by weather conditions, as opposed to “DepDelay,” which represented the total amount of delay time. The way that the “WeatherDelay” was computed is still unknown and may have contributed to the wide variance of values. Alternatively, the low R-Squared values may have been due to other noise in the dataset unrelated to weather conditions. For example, individual pilots, companies, or airports may have different standards when it comes to delaying flights due to weather.

## Classification Findings

### **5.2.1 Binary Classification Findings – Label1**

Both XGBoost and Random Forest gave decent prediction on the binary class where there were only two labels: “ontime” and “delay”. Without taking the imbalance between the two classes into consideration, both models could predict accurately around 65% of the time. There was a huge imbalance in the dataset since the number of “ontime” flights accounted for 97% of the records. When this issue was addressed and solved, the performance of both models improved up to 76%-89%. The method used to solve the imbalance issue in the dataset leading to 89% of the accuracy score was called Balanced Random Forest Classifier. It resampled the data on the bootstrap sample to explicitly change the class distribution for the purpose of creating a synthetic balance in the dataset. It outperformed other methods used in the analysis for handling the imbalanced classes, such as: changing the hyperparameter class\_weight to ‘balanced’ or Synthetic Minority Oversampling Technique (SMOTE).

**Chart, treemap chart

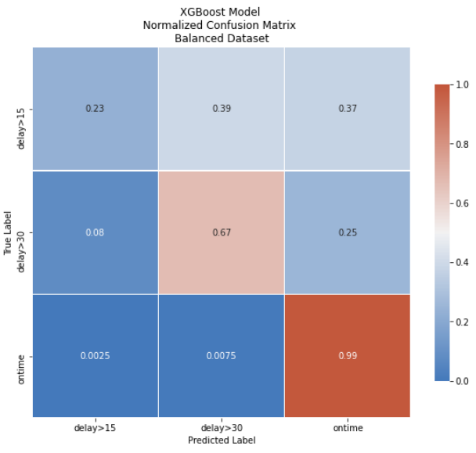
Description automatically generated**

*Figure 5.1. XGBoost and Random Forest confusion matrices on ‘Label1’*

### **5.2.2 Multi-classification Findings – Label2**

For label 2, predictions of “ontime” flights tended to be around 99% for the XGBoost and 86% for the Balanced Random Forest. The Balanced Random Forest algorithm took an expected hit on the “ontime” class based on the bootstrap method in that algorithm where it will balance out the samples of each class, although down-sampling the majority class where accuracy took a hit. On the other hand, prediction of “delay” flights was challenging. Using XGBoost, the accuracy for “delay>15” was only 23% and the accuracy for “delay>30” was only 63% compared to 62% and 58% from the Balanced Random Forest. The balanced boot strap sample from the Balanced Random Forest helped the “delay>15” class perform significantly better.

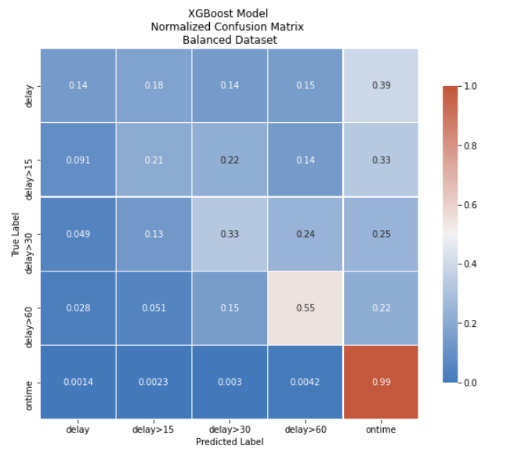
Chart, treemap chart

Description automatically generated

*Figure 5.2. XGBoost and Random Forest confusion matrices on ‘Label2’*

### **5.2.3 Multi-classification – Label3**

For the 5-category multiclass label where delay times were separated into “delay”, “delay>15”, “delay>30”, “delay>60”, XGBoost was only able to predict “delay>60” with 55% accuracy compared to the results of the Random Forest analysis, these were considered a poor performance.



Chart, treemap chart

Description automatically generated

*Figure 5.3. XGBoost and Random Forest confusion matrices on ‘Label3’*

### **5.2.4 Conclusion**

It was determined that the best model for the classification prediction was the Balanced Random Forest Classifier. This model performs random under-sampling of the majority class in reach bootstrap sample. Since it was determined that this model worked best, this model model on all the labeled variables we have which are label1, label2, label3. As you can see in confusion matrices, label2 actually had the best result in term of accuracy and helpful meanings. The Balanced Random Forest Classifier can predict delay more than 15min, more than 30 min and the on-time flight with 62, 58 82% accuracy of the time respectively.

**Chart, treemap chart

Description automatically generated**

**Chart, treemap chart

Description automatically generatedChart, treemap chart

Description automatically generated**

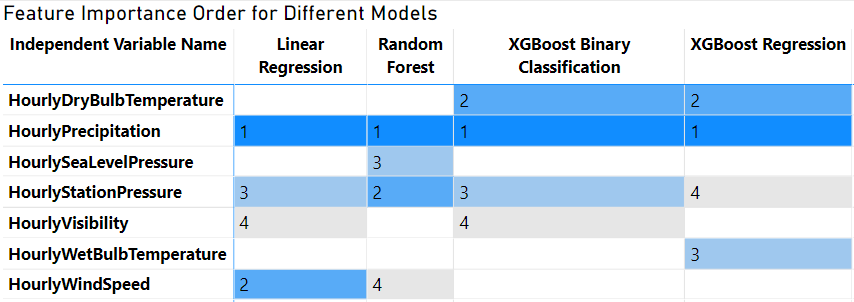
*Figure 5.4. Random Forest confusion matrices on all 3 labels: ‘Label1’, ‘Label2’, ‘Label3’*

## Feature Importance

Finally, the three models could determine which dependent variables had significant impact on the flight performance. There were some differences about the four most influence weather conditions on flight delay between Linear Regression, XGBoost and Random Forest. However, all three models showed the same result that ‘HourlyPrecipitation’ was stand out compared with other variables in its impact of flight performance.

|  |  |  |  |
| --- | --- | --- | --- |
| **Linear Regression** | **XGBoost**  **Binary Classification** | **XGBoost**  **Regression** | **Random Forest** |
| 1. HourlyPrecipitation  2. HourlyWindSpeed  3. HourlyStationPressure  4. HourlyVisibility | 1. HourlyPrecipitation  2. HourlyDryBulbTemperature  3. HourlyStationPressure  4. HourlyVisibility | 1. HourlyPrecipitation  2. HourlyWetBulbTemperature  3. HourlyDryBulbTemperature  4. HourlyStationPressure | 1. HourlyPrecipitation  2. HourlyRelativeHumidity  3. HourlyStationPressure  4. HourlyDryBulbTemperature |

*Figure 5.1. Top 4 Feature Importance of four models*



*Figure 5.2 Feature Importance Comparison Matrix*

# Summary

The team’s Linear Regression analysis proved that several weather conditions *are* in fact significantly correlated to flight weather delays. The variables Hourly Precipitation, Hourly Wind Speed, Hourly Station Pressure, and Hourly Visibility showed undeniably statistically significant correlations to the weather delay, with very little correlation between them. This analysis proved that high levels of precipitation, high wind speeds, low air pressure, and low visibility levels all contribute to flight delays in a meaningful way. High precipitation levels are a particularly potent source of delays, with a single additional inch of precipitation correlating to an average of approximately 70 additional minutes of delay. Airline decision-makers and passengers could find this information useful knowing that if precipitation is high, there is a high probability for a long departure delay. This information could be used to help predict departure delays.

This project also found some great information from the exploratory data analysis. The PowerBI interactive visualizations could be very useful to future users for comparisons between the four different airlines within this analysis or to compare delays between regions from the airports this project used.

From our analysis, we determined that

XGBoost and Linear Regression algorithms could not solve the regression problem which was the prediction of how long the flight delay would be. For predicting whether a flight would be delayed, Balance Random Forest Classifier showed the most accuracy results.

# Future Work

## Clustering Algorithms

One potential improvement for this project would be to incorporate clustering algorithms into the analysis. A clustering algorithm could be used to categorize delays by the severity of the delay, rather than only using a binary delay/no-delay categorization. If most delay times cluster around a set of distinct time increments, the level of delay could be defined based on those clusters. For example, a *slight delay* may be defined as approximately 15 minutes or less, a *moderate delay* as 15-60 minutes, and a major delay as greater than 60 minutes.

The same approach could be applied to categorizing weather data. Various weather variables could be analyzed using a clustering algorithm to determine what defines *good* weather, *mild* weather, *extreme* weather, etc.

These detailed classifications could be applied to each record in the dataset and analyzed in a similar manner to our existing binary labels. This would create a model that is much more useful for the end-user by providing a more precise value for the duration of the expected delay based on general weather conditions.

## Other Factors

**Deeper Dive into Weather**

Three weather aspects that this project did not include were cloud coverage/ceiling , REM (Hourly Remarks), and wind direction. These attributes were removed from this project’s data for multiple reasons. First, the large size of the aircraft being flown and the high qualifications of the pilots flying these aircraft made them low value attributes because there are few occasions when these factors would cause delays. Second, the format of the three attributes made them time consuming to change into a useful format. This team determined that these attributes were correlated to other attributes that were kept for analysis. Specifically, the ceiling attribute was directly correlated to the visibility. This is because when the ceiling is low enough to be a factor for the departures in this scenario, the visibility would be decreased as well. The REM attribute was determined to be redundant data because it was the transcript of the coded Automatic Terminal Information Service (ATIS) recording that is broadcast on specified ATIS radio frequencies. The data it provides was already within other attributes within the data set. Finally, the wind direction was not as much of a factor as the wind speed because delays for crosswinds were determined to be such a small percentage of weather delays for this type of flying and if there are delays due to crosswinds, then it would be because of the wind speed as well. Of course, there are still some occasions when crosswinds will delay even these aircraft and pilots at lower wind speeds. However, determining when this is the case would require comparing the wind direction to the runway direction and the time limitations of this project kept this from being achievable.

**Location Variation of Weather Delays**

This paper looked at the impact of the weather at the departure destination and its effect on flight delays at the departure destination. However, there are other locations where the weather might be a factor. One location would be the arrival airport. One variation of our paper would be investigating how the weather at the arrival airport predicts weather delays for a flight. A second location would be the points en route to an arrival airport. Another variation would be investigating how the weather at points in the flight path of the airplane would affect flight delays. This would require incorporating a lot more data, both data points for the location of the flight path of each flight and data for the weather variables at those locations along the flight path. This analysis could be helpful for airlines that may want to optimize their flight scheduling and increase their ability to plan ahead for potential flight delays.

**Seasonal Variation of Weather Delays**

This paper looks at the impact of weather delays in aggregate. However, weather has seasonal variations. While flights departing from the Orlando airport in July may be delayed due to seasonal thunderstorms, a flight departing in December has less of a risk of being delayed. Similarly, a flight departing Chicago’s O’Hare airport in December will be more likely to be delayed by a blizzard than a similar flight in July. An analysis that could identify the potential delay based on the season or calendar month would also be very helpful in helping to adjust predictions seasonally, by incorporating seasonal weather patterns.

**Expansion of Airlines**

This paper just used flight data from the four major airlines, American, Delta, Southwest, and United. The scope could be expanded by increasing the airlines involved and comparing the delays for different airlines to see if some airlines are more likely to delay flights due to weather. This could be a helpful analysis for airlines that may want to see how they compare to other airlines and for consumers that want to understand, which airlines have a higher risk of flight delay due to weather.

**Expansion of Airports**

This paper also only incorporates the 12 busiest airports in the US. Another possible scope expansion would be increasing the number of airports or even changing the scope to include other countries. Expanding the number of airports could be avenue to compare airports within a region (e.g. if weather delay cancellations are longer at the Ronald Reagan Washington National Airport than the Dulles International Airport). This would be helpful for airlines and airports to see if processes for delays at certain airports are leading to longer delays for airport-specific reasons.

**Expansion of Time Period**

The data set in this paper covers the time period 2018-2019. One possible time-related scope expansion would be expanding the time period to cover more years. Another would be to cover a different point of time in the past and see if the same results hold for a period 10 or 20 years in the past. It would also be interesting to see if the results still hold today, as COVID-19 has drastically changed travel operations in the US.

**Incorporating Other Types of Delays**

Value could be added if data for other types of delays could be collected and reasonably predicted for future flights. For example, a team could gather aircraft maintenance delays from airlines and use that data to predict when an aircraft will be delayed. To go a step further, if the data that was collected included aircraft time when parts failed, one could also use that data to predict when a part needed to be replaced in order to maximize its usability while still avoiding operational delays from waiting until part failure to replace it.

**Compare models with additional algorithms applied to the dataset**:

The paper used several methods to find results from the data set: Linear Regression, XGBoost, and Random Forest. However, there are many different algorithms available that have the possibility of producing better models than the ones used for the project. Some examples of these are Long Short-Term Memory (LSTM), Support Vector Machines (SVM), Cat-Boost, logistic regression, and the clustering algorithms mentioned above.

In addition, depending on the model in this projects’ analysis, different attributes floated to the top when it came to their impact on departure delays. An ensemble of these models, or others, could lead to a more complete result than just one model. Therefore, future work could include combining algorithms in an ensemble method.

**Improve weather delay categorization:**

While the scope of this paper covered the delays that were marked as weather delays by the Bureau of Transportation Statistics (BTS), that is not all of the delay that may have been caused by weather. The BTS will allocate various portions of a delay into different types of delay. That makes calculating the exact amount of weather delay more difficult as it can be spread across multiple categories of delay. Per the BTS website, the extreme weather delays are combined with an allocation of the National Aviation System (NAS) delay that is due to weather. This is then added to an allocation that is estimated for the proportion of late-arriving aircraft due to weather.

# Appendix A – Project Repository

For more information, code, and supporting data please visit the Github repository linked below.

<https://github.com/brdyer/CS504-DeltaForce-Team-Project-Fall-2020.git>

# Appendix B – Risk Section

**Scope versus Timeline:** Can we finish the project in the remaining timeline?

* Probability of occurring – 20%.
* Impacts if it Occurs – limited learning on the analysis and visualization aspects of the project.
* Mitigation – hold close to the business rhythm and if we start getting behind, discuss options on scope with the professor.

**Analysis Tools / Models:** Besides time series analysis, a ML model (regression or XGBoost) will be applied for future prediction. However, the concept of ML is a challenge for the team.

* Probability of occurring – 30%.
* Impacts if it Occurs – The learning process could delay analysis.
* Mitigation – if ML becomes too difficult, the team will stick with the time series analysis findings and visualization.

**Hardware Limitation**: The team's personal computers for analysis have limited capability which may make it difficult to execute all the models we would like to for our project in a reasonable amount of time.

* Probability of occurring – 80%.
* Impacts if it Occurs – Limits the results of the team's analysis to more basic tools and algorithms
* Mitigation – Work with Professor Baldo to use AWS/SageMaker

# Appendix C – Carrier Identifier Listing

|  |  |
| --- | --- |
| American Airlines | AA |
| Delta Air Lines | DL |
| Southwest Airlines | WN |
| United Airlines | UA |

# Appendix D – Airport Identifier Listing

|  |  |
| --- | --- |
| Atlanta Hartsfield-Jackson International Airport | ATL |
| Baltimore/Washington International Thurgood Marshall Airport | BWI |
| Chicago O’Hare International Airport | ORD |
| Dallas/Fort Worth International Airport | DFW |
| Denver International Airport | DEN |
| Dulles International Airport | IAD |
| New York John F. Kennedy International Airport | JFK |
| Los Angeles International Airport | LAX |
| Orlando International Airport | MCO |
| San Francisco International Airport | SFO |
| Seattle-Tacoma International Airport | SEA |
| Washington National/Ronald Reagan Airport | DCA |

# Appendix E – References

Air Traffic by the Numbers. (2020, March 19). Federal Aviation Administration. Retrieved October 28,

2020 from https://www.faa.gov/air\_traffic/by\_the\_numbers/

Airline On-Time Performance Data. (n.d.). Bureau of Transportation Statistics. Retrieved October 27,

2020 from <https://www.transtats.bts.gov/DL_SelectFields.asp?Table_ID=237>

Akers, C., Anderson, L.J., Cantrell, C., Haag, M., Sutor, J., Winters, P. (2020). Predictive Analytics for

Airline Operations.

Bergstra, J., Yamins, D., Cox, D. D. (2013) Making a Science of Model Search: Hyperparameter

Optimization in Hundreds of Dimensions for Vision Architectures. To appear in Proc. of the 30th International Conference on Machine Learning (ICML 2013).

Bureau of Transportation Statistics. “Airline On-Time Statistics and Delay Causes.” Department of Transportation, [www.transtats.bts.gov/OT\_Delay/ot\_delaycause1.asp?type=4](http://www.transtats.bts.gov/OT_Delay/ot_delaycause1.asp?type=4).

Busson, T. Why is My Flight Delayed? Retrieved October 28, 2020 from

<https://www.claimcompass.eu/blog/why-is-my-flight-delayed/>

Busson, T. Can I Get a Compensation When my Flight was Delayed or Canceled because of Bad

Weather? Maybe. Retrieved November 7, 2020 from <https://www.claimcompass.eu/blog/bad-weather-flight-compensation/>

Cheng-Lung Wu, Kristie Law. Modelling the delay propagation effects of multiple resource connections

in an airline network using a Bayesian network model, Transportation Research Part E: Logistics and Transportation Review, Volume 122, 2019, Pages 62-77, https://doi.org/10.1016/j.tre.2018.11.004.

Choi, S., Kim, Y., Briceno, S., & Mavris, D. (2016, December 12). Prediction of weather-induced airline

delays based on machine learning algorithms. The Institute of Electrical and Electronics

Engineers, Inc. https://doi.org/10.1109/ITNEC48623.2020.9084929

Dean, S. (September 4, 2020). 200 Busiest US Airports List [Sept 2020 Update]. Retrieved October 24

2020 from <https://www.stratosjets.com/blog/busiest-us-airports/>

Donges, Niklas (2019, June 16, 2019, Updated 2020, September 3) A Complete Guide to the Random Forest Algorithm https://builtin.com/data-science/random-forest-algorithm

Dou, X. (2020). Flight Arrival Delay Prediction And Analysis Using Ensemble Learning. The Institute of

Electrical and Electronics Engineers, Inc. https://doi.org/ 10.1109/ITNEC48623.2020.9084929

Economic Impact of Commercial Aviation By State. Airlines for America. Retrieved November 21, 2020

from <https://www.airlines.org/data/>.

FAA TV: How Does the National Airspace System (NAS) Work. (2011, September 15). Federal Aviation

Administration. Retrieved November 23, 2020 from https://www.faa.gov/tv/?mediaId=370

FAQ: Weather Delay. Federal Aviation Administration. Retrieved November 7, 2020

from https://www.faa.gov/nextgen/programs/weather/faq/

Federal Register, Vol. 67, No. 227. Rules and Regulations. (2002, November 25). Retrieved 3 December,

2020 from https://www.regulations.gov/document?D=DOT-OST-2000-8164-0059

How NextGen Works. (2020, August 10). Federal Aviation Administration. Retrieved October 24, 2020

from https://www.faa.gov/nextgen/how\_nextgen\_works/new\_technology/

IATA Airline and Location Codes. (n.d.). International Air Transport Association. Retrieved October 24, 2020 from https://www.iata.org/en/services/codes/

Jain, A. (2016, March 1). Complete Guide to Parameter Tuning in XGBoost with codes in Python.

Analytics Vidhya. Retrieved November 23, 2020 from [https://www.analyticsvidhya.com/blog/ 2016/03/](https://www.analyticsvidhya.com/blog/%202016/03/)complete-guide-parameter-tuning-xgboost-with-codes-python/

Khanmohammadi, S., Tutun, S., & Kucuk, Y. (2016). A New Multilevel Input Layer Artificial Neural

Network for Predicting Flight Delays at JFK Airport. Procedia Computer Science.

https://doi.org/10.1016/j.procs.2016.09.321

Land-Based Station Data. Retrieved December 3, 2020 from <https://www.ncdc.noaa.gov/data->

access/land-based-station-data

Local Climatological Data (LCD). Retrieved October 28, 2020 from

<https://www.ncdc.noaa.gov/cdo-web/datatools/lcd>

Local Climatological Data (LCD) Dataset Documentation. Retrieved November 3, 2020 from

https://www1.ncdc.noaa.gov/pub/data/cdo/documentation/LCD\_documentation.pdf

Mamdoohi, S., Fotouhi, H., Aguilar, A., Jabeen, A., & Tan, Y. (2020). Predictive Analytics for

Airline Operations.

Mazareanu, E. (2020, Aug) Countries with the highest number of airline passengers globally 2018.

Retrieved October 28, 2020. https://www.statista.com/statistics/537002/airline-passengers-worldwide-by-country/worldwide-by-country/

Seif, G. (2019, May 29). A Beginner’s guide to XGBoost. Towards Data Science. Retrieved November 23,

2020 from https://towardsdatascience.com/a-beginners-guide-to-xgboost-87f5d4c30ed7

The Difference Between an Air Dispatcher and an Air Traffic Controller. (2020, August 10).

Sheffield School of Aeronautics. Retrieved October 28, 2020 from https://www.sheffield.com/articles/the-difference-between-an-aircraft-dispatcher-and-an-air-traffic-controller.

Wu, C. & Law, K. (2019, February). Modelling the delay propagation effects of multiple resource

connections in an airline network using a Bayesian network model. Transportation Research Part

XGBoost: How it works, with an example. *YouTube*, uploaded by Sundog Education with Frank Kane, 11

February, 2020, https://www.youtube.com/watch?v=OQKQHNCVf5k

Zarina, I. (2017, March). A Simple XGBoost Tutorial Using the Iris Dataset. KDNuggets.

Retrieved November 21, 2020 from https://www.kdnuggets.com/2017/03/simple-xgboost-tutorial-iris-dataset.html

Zhang, Z. (2019, June 26). Boosting Algorithms Explained. Towards Data Science. Retrieved

November 21, 2020 from https://towardsdatascience.com/boosting-algorithms-explained-d38f56ef3f30