Flight Delay Analysis

Shubham Kulkarni, Atul Gupta, Aman Gupta, Nihar Dugade

December 2023

Abstract

In this study, we present a comprehensive analysis of the U.S. Department of Transportation's Bureau of Transportation Statistics dataset, focusing on the on-time performance of domestic flights operated by large air carriers. Spanning a decade from 2009 to 2019, the dataset provides a wealth of daily airline information, carrier details, taxing-in/out times, and generalized delay reasons. Our project aims to extract valuable insights from this extensive dataset, uncovering patterns and identifying factors that significantly influence flight delays and cancellations. The objectives include analyzing the on-time performance over the specified period, identifying trends and patterns in flight disruptions, and investigating the impact of carriers, flight information, and time-related factors on performance. The ultimate goal is to provide actionable recommendations for improving on-time performance and reducing delays in the U.S. domestic aviation sector. For the implementation details and code repository, please refer to our GitHub repository: `https://github.com/atulgupta002/flight_delay_analysis`.

1 Introduction

The aviation industry plays a pivotal role in connecting people and goods across the United States, contributing significantly to the country's economic growth and development. One critical aspect of this industry is the on-time performance of domestic flights, which directly affects passenger satisfaction, operational efficiency, and overall economic productivity. The U.S. Department of Transportation's Bureau of Transportation Statistics (BTS) has meticulously collected and maintained a dataset spanning the years 2009 to 2019, capturing a decade's worth of information related to domestic flights operated by large air carriers.

Our research focuses on leveraging this extensive dataset to conduct a thorough analysis of the on-time performance of domestic flights. By examining daily airline information, carrier details, taxing-in/out times, and generalized delay reasons, we aim to uncover underlying patterns and identify key factors contributing to flight delays, cancellations, and diversions.

Through statistical analysis and data visualization techniques, we intend to provide a nuanced understanding of the trends observed over the specified 10-year period.

The objectives of our study encompass the analysis of on-time performance, identification of trends and patterns in flight disruptions, and a comprehensive investigation into the impact of carriers, flight information, and time-related factors on overall performance. By achieving these objectives, we aspire to offer actionable recommendations that stakeholders in the aviation industry, including airlines and regulatory bodies, can implement to enhance on-time performance and reduce delays.

The remainder of this document outlines the methodology, data processing techniques, and key findings of our analysis. By delving into the intricacies of domestic flight performance, we aim to contribute valuable insights that can inform decision-making processes and facilitate improvements within the U.S. aviation sector.

2 Dataset Description

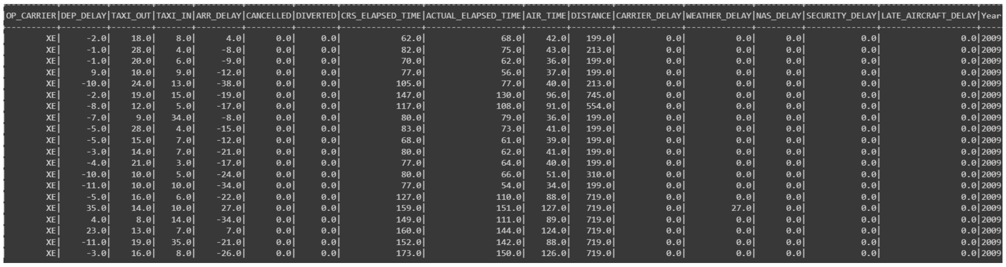
The dataset under investigation provides a comprehensive record of domestic flights operated by major air carriers in the United States. Covering a span of a decade from 2009 to 2019, this dataset incorporates essential parameters related to the on-time performance of these flights.

Among the key attributes, the 'OP_CARRIER' field signifies the operating carrier code, designating the airline responsible for the flight. 'DEP_DELAY' represents the departure delay, indicating the time difference between scheduled and actual departure. 'TAXI_OUT' and 'TAXI_IN' denote the time spent taxiing before takeoff and after landing, respectively.

The 'ARR_DELAY' attribute measures the arrival delay, highlighting the variance between scheduled and actual arrival times. Binary indicators such as 'CANCELLED' and 'DIVERTED' convey whether a flight was canceled or diverted, respectively. 'CRS_ELAPSED_TIME' reflects the planned duration of the flight, while 'ACTUAL_ELAPSED_TIME' provides the actual time taken from departure to arrival. 'AIR_TIME' accounts for the time the aircraft spends in the air, excluding taxiing time. The 'DISTANCE' column denotes the total distance traveled during the flight.

Delving into the factors influencing delays, the dataset includes categorical variables such as 'CARRIER_DELAY,' 'WEATHER_DELAY,' 'NAS_DELAY,' 'SECURITY_DELAY,' and 'LATE_AIRCRAFT_DELAY.' These variables capture delays attributed to carrier-related issues, weather conditions, National Airspace System (NAS) constraints, security concerns, and delays caused by the late arrival of the aircraft from a previous flight, respectively.

The temporal aspect is captured by the 'Year' column, facilitating a longitudinal analysis over the specified decade. This dataset serves as a valuable resource for a comprehensive understanding of the intricacies of domestic flight operations, offering insights into on-time performance, identifying patterns in delays and cancellations, and exploring the impact of various factors on the aviation sector during the specified timeframe.



```
----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----
OP_CARRIER|DEP_DELAY|TAXI_OUT|TAXI_IN|ARR_DELAY|CANCELLED|DIVERTED|CRS_ELAPSED_TIME|ACTUAL_ELAPSED_TIME|AIR_TIME|DISTANCE|CARRIER_DELAY|WEATHER_DELAY|NAS_DELAY|SECURITY_DELAY|LATE_AIRCRAFT_DELAY|Year
----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----
        XE|     -2.0|    18.0|    8.0|      4.0|      0.0|     0.0|            62.0|               68.0|    42.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -1.0|    28.0|    4.0|     -8.0|      0.0|     0.0|            82.0|               75.0|    43.0|   213.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -1.0|    20.0|    6.0|     -9.0|      0.0|     0.0|            70.0|               62.0|    36.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|      9.0|    10.0|    9.0|    -12.0|      0.0|     0.0|            77.0|               56.0|    37.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|    -10.0|    24.0|   13.0|    -38.0|      0.0|     0.0|           105.0|               77.0|    40.0|   213.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -2.0|    19.0|   15.0|    -19.0|      0.0|     0.0|           147.0|              130.0|    96.0|   745.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -8.0|    12.0|    5.0|    -17.0|      0.0|     0.0|           117.0|              108.0|    91.0|   554.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -7.0|     9.0|   34.0|     -8.0|      0.0|     0.0|            80.0|               79.0|    36.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -5.0|    28.0|    4.0|    -15.0|      0.0|     0.0|            83.0|               73.0|    41.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -5.0|    15.0|    7.0|    -12.0|      0.0|     0.0|            68.0|               61.0|    39.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -3.0|    14.0|    7.0|    -21.0|      0.0|     0.0|            80.0|               62.0|    41.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -4.0|    21.0|    3.0|    -17.0|      0.0|     0.0|            77.0|               64.0|    40.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|    -10.0|    10.0|    5.0|    -24.0|      0.0|     0.0|            80.0|               66.0|    51.0|   310.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|    -11.0|    10.0|   10.0|    -34.0|      0.0|     0.0|            77.0|               54.0|    34.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -5.0|    16.0|    6.0|    -22.0|      0.0|     0.0|           127.0|              110.0|    88.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     35.0|    14.0|   10.0|     27.0|      0.0|     0.0|           159.0|              151.0|   127.0|   719.0|          0.0|         27.0|      0.0|           0.0|                0.0|2009
        XE|      4.0|     8.0|   14.0|    -34.0|      0.0|     0.0|           149.0|              111.0|    89.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     23.0|    13.0|    7.0|      7.0|      0.0|     0.0|           160.0|              144.0|   124.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|    -11.0|    19.0|   35.0|    -21.0|      0.0|     0.0|           152.0|              142.0|    88.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
        XE|     -3.0|    16.0|    8.0|    -26.0|      0.0|     0.0|           173.0|              150.0|   126.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009
```

Figure 1: Sample Representation of the Dataset

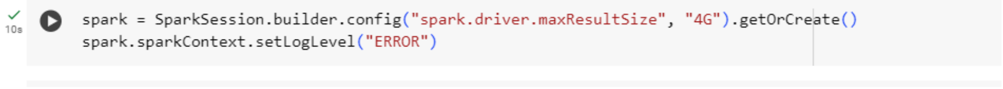
3 Implementation

3.1 Importing Libraries

The code begins by importing essential Python libraries for various tasks, including data processing, machine learning, and visualization. Noteworthy libraries include PySpark for big data processing, Pandas for data manipulation, and scikit-learn for machine learning.

3.2 Spark Session Initialization

This section initializes a Spark session, a fundamental component of PySpark, used for distributed data processing. The log level is set to ”ERROR” to reduce unnecessary log outputs.



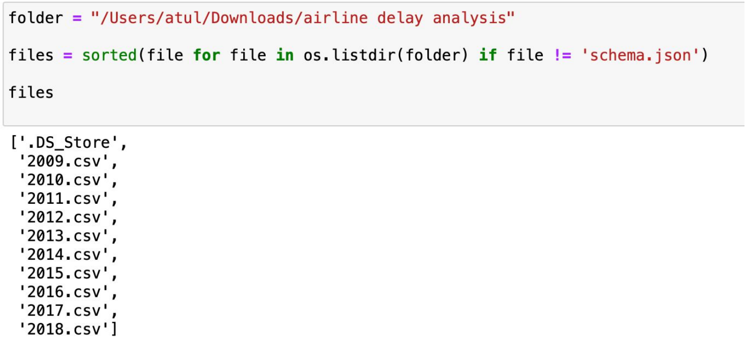
```
spark = SparkSession.builder.config("spark.driver.maxResultSize", "4G").getOrCreate()
spark.sparkContext.setLogLevel("ERROR")
```

Figure 2: Code Snippet for Spark initialization

3.3 Loading Data

3.3.1 Folder and File Handling

Specifies the folder path containing airline delay data, lists, and sorts the files in the folder, excluding ”schema.json.”



```
folder = "/Users/atul/Downloads/airline delay analysis"

files = sorted(file for file in os.listdir(folder) if file != 'schema.json')

files
```

```
['.DS_Store',
 '2009.csv',
 '2010.csv',
 '2011.csv',
 '2012.csv',
 '2013.csv',
 '2014.csv',
 '2015.csv',
 '2016.csv',
 '2017.csv',
 '2018.csv']
```

Figure 3: Input Dataset

3.3.2 Schema Loading

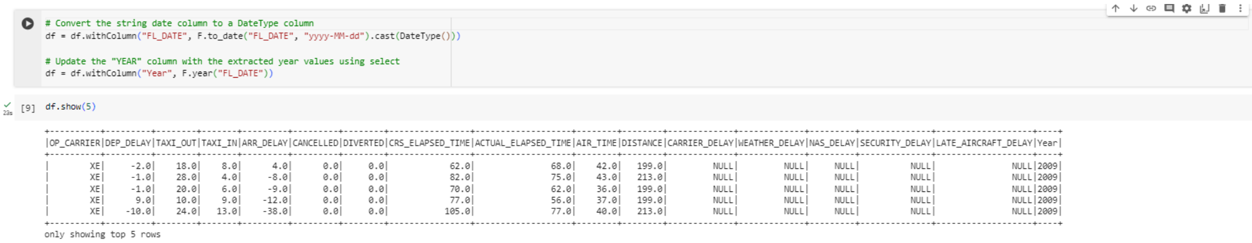
Reads the schema from the ”schema.json” file. Schema specifies the data type associated with each column and the overall structure of the data. We will use schema to merge our yearly datasets into a spark dataframe.

3.3.3 Data Loading and Cleaning

Reads the first CSV file into a PySpark DataFrame using the specified schema. Iterates over the remaining files, reads them, and appends them to the existing DataFrame. Removes unwanted columns like flight dates, flight numbers, and blank columns.

3.3.4 Date and Year Processing

Converts the "FL_DATE" column to a DateType column. Extracts the year from the date and adds a "Year" column to the DataFrame.



```
# Convert the string date column to a DateType column
df = df.withColumn("FL_DATE", F.to_date("FL_DATE", "yyyy-MM-dd").cast(DateType()))

# Update the "YEAR" column with the extracted year values using select
df = df.withColumn("Year", F.year("FL_DATE"))
```

```
[9] df.show(5)
```

```
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+
|OP_CARRIER|DEP_DELAY|TAXI_OUT|TAXI_IN|ARR_DELAY|CANCELLED|DIVERTED|CRS_ELAPSED_TIME|ACTUAL_ELAPSED_TIME|AIR_TIME|DISTANCE|CARRIER_DELAY|WEATHER_DELAY|NAS_DELAY|SECURITY_DELAY|LATE_AIRCRAFT_DELAY|Year|
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+
|        XE|     -2.0|    18.0|    8.0|      4.0|      0.0|     0.0|            62.0|               68.0|    42.0|   199.0|         NULL|         NULL|     NULL|          NULL|               NULL|2009|
|        XE|     -1.0|    28.0|    4.0|     -8.0|      0.0|     0.0|            82.0|               75.0|    43.0|   213.0|         NULL|         NULL|     NULL|          NULL|               NULL|2009|
|        XE|     -1.0|    20.0|    6.0|     -9.0|      0.0|     0.0|            70.0|               62.0|    36.0|   199.0|         NULL|         NULL|     NULL|          NULL|               NULL|2009|
|        XE|      9.0|    10.0|    9.0|    -12.0|      0.0|     0.0|            77.0|               56.0|    37.0|   199.0|         NULL|         NULL|     NULL|          NULL|               NULL|2009|
|        XE|    -10.0|    24.0|   13.0|    -38.0|      0.0|     0.0|           105.0|               77.0|    40.0|   213.0|         NULL|         NULL|     NULL|          NULL|               NULL|2009|
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+
only showing top 5 rows
```

Figure 4: Data Cleaning

3.4 Data Preprocessing

3.4.1 Unique Values Extraction

Defines a function (`get_unique`) to retrieve unique values from a specific column. We have also defined lists for delay types and years.

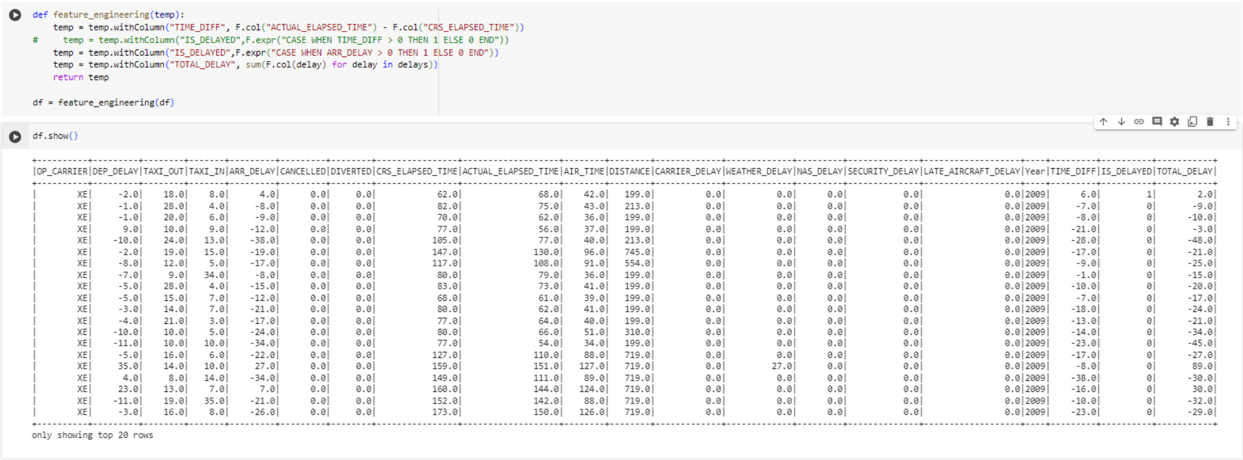
3.4.2 Null Value Handling

Defines columns to fill with 0.0 for null values. We have defined a function (`preprocessing`) to fill null values in specified columns with 0.0 and drop the rows with null values. These columns are CARRIER_DELAY, WEATHER_DELAY, NAS_DELAY, SECURITY_DELAY, LATE_AIRCRAFT_DELAY. These columns are mostly empty but can be used to calculate the overall delay for every flight wherever the data is available

3.5 Feature Engineering

3.5.1 Time and Delay Features

Defines a function (`feature_engineering`) to create new columns: "TIME_DIFF," "IS_DELAYED," and "TOTAL_DELAY." Here TIME_DIFF is the difference in minutes between the actual total time taken by the flight to operate and the scheduled total time the flight should have taken. We then use the ARR_DELAY column to create a binary indicator "IS_DELAYED" where 0 corresponds to "NOT DELAYED" and 1 corresponds to "DELAYED". TOTAL_DELAY is the sum of all different delays available in the dataset.



```
def feature_engineering(temp):
    temp = temp.withColumn("TIME_DIFF", F.col("ACTUAL_ELAPSED_TIME") - F.col("CRS_ELAPSED_TIME"))
#     temp = temp.withColumn("IS_DELAYED",F.expr("CASE WHEN TIME_DIFF > 0 THEN 1 ELSE 0 END"))
    temp = temp.withColumn("IS_DELAYED",F.expr("CASE WHEN ARR_DELAY > 0 THEN 1 ELSE 0 END"))
    temp = temp.withColumn("TOTAL_DELAY", sum(F.col(delay) for delay in delays))
    return temp

df = feature_engineering(df)
```

```
df.show()
```

```
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+---------+----------+-----------+
|OP_CARRIER|DEP_DELAY|TAXI_OUT|TAXI_IN|ARR_DELAY|CANCELLED|DIVERTED|CRS_ELAPSED_TIME|ACTUAL_ELAPSED_TIME|AIR_TIME|DISTANCE|CARRIER_DELAY|WEATHER_DELAY|NAS_DELAY|SECURITY_DELAY|LATE_AIRCRAFT_DELAY|Year|TIME_DIFF|IS_DELAYED|TOTAL_DELAY|
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+---------+----------+-----------+
|        XE|     -2.0|    18.0|    8.0|      4.0|      0.0|     0.0|            62.0|               68.0|    42.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|      6.0|         1|        2.0|
|        XE|     -1.0|    28.0|    4.0|     -8.0|      0.0|     0.0|            82.0|               75.0|    43.0|   213.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|     -7.0|         0|       -9.0|
|        XE|     -1.0|    20.0|    6.0|     -9.0|      0.0|     0.0|            70.0|               62.0|    36.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|     -8.0|         0|      -10.0|
|        XE|      9.0|    10.0|    9.0|    -12.0|      0.0|     0.0|            77.0|               56.0|    37.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -21.0|         0|       -3.0|
|        XE|    -10.0|    24.0|   13.0|    -38.0|      0.0|     0.0|           105.0|               77.0|    40.0|   213.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -28.0|         0|      -48.0|
|        XE|     -2.0|    19.0|   15.0|    -19.0|      0.0|     0.0|           147.0|              130.0|    96.0|   745.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -17.0|         0|      -21.0|
|        XE|     -8.0|    12.0|    5.0|    -17.0|      0.0|     0.0|           117.0|              108.0|    91.0|   554.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|     -9.0|         0|      -25.0|
|        XE|     -7.0|     9.0|   34.0|     -8.0|      0.0|     0.0|            80.0|               79.0|    36.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|     -1.0|         0|      -15.0|
|        XE|     -5.0|    28.0|    4.0|    -15.0|      0.0|     0.0|            83.0|               73.0|    41.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -10.0|         0|      -20.0|
|        XE|     -5.0|    15.0|    7.0|    -12.0|      0.0|     0.0|            68.0|               61.0|    39.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|     -7.0|         0|      -17.0|
|        XE|     -3.0|    14.0|    7.0|    -21.0|      0.0|     0.0|            80.0|               62.0|    41.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -18.0|         0|      -24.0|
|        XE|     -4.0|    21.0|    3.0|    -17.0|      0.0|     0.0|            77.0|               64.0|    40.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -13.0|         0|      -21.0|
|        XE|    -10.0|    10.0|    5.0|    -24.0|      0.0|     0.0|            80.0|               66.0|    51.0|   310.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -14.0|         0|      -34.0|
|        XE|    -11.0|    10.0|   10.0|    -34.0|      0.0|     0.0|            77.0|               54.0|    34.0|   199.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -23.0|         0|      -45.0|
|        XE|     -5.0|    16.0|    6.0|    -22.0|      0.0|     0.0|           127.0|              110.0|    88.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -17.0|         0|      -27.0|
|        XE|     35.0|    14.0|   10.0|     27.0|      0.0|     0.0|           159.0|              151.0|   127.0|   719.0|          0.0|         27.0|      0.0|           0.0|                0.0|2009|     -8.0|         0|       89.0|
|        XE|      4.0|     8.0|   14.0|    -34.0|      0.0|     0.0|           149.0|              111.0|    89.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -38.0|         0|      -30.0|
|        XE|     23.0|    13.0|    7.0|      7.0|      0.0|     0.0|           160.0|              144.0|   124.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -16.0|         0|       30.0|
|        XE|    -11.0|    19.0|   35.0|    -21.0|      0.0|     0.0|           152.0|              142.0|    88.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -10.0|         0|      -32.0|
|        XE|     -3.0|    16.0|    8.0|    -26.0|      0.0|     0.0|           173.0|              150.0|   126.0|   719.0|          0.0|          0.0|      0.0|           0.0|                0.0|2009|    -23.0|         0|      -29.0|
+----------+---------+--------+-------+---------+---------+--------+----------------+-------------------+--------+--------+-------------+-------------+---------+--------------+-------------------+----+---------+----------+-----------+
only showing top 20 rows
```

Figure 5: Feature Engineering

3.6 Visualizations

Let's delve into the visual representation of our dataset to unravel the underlying trends it encapsulates. The ensuing plots offer a nuanced perspective, facilitating a comprehensive understanding of the dataset dynamics. Over the years, the dataset exhibits a consistent pattern in the number of flights. The objective here is to meticulously examine and visually depict how flights are distributed across various carriers. This analysis proves instrumental in deciphering the collective impact of each airline on the dataset. Noteworthy is the observation of a surge in flight counts in the year 2018. However, it is imperative to acknowledge that this uptick might be influenced by the dataset's composition, potentially containing a higher volume of records from that particular year. As we interpret these visualizations, it is crucial to consider the dataset's temporal distribution and potential variations in data density across different years.

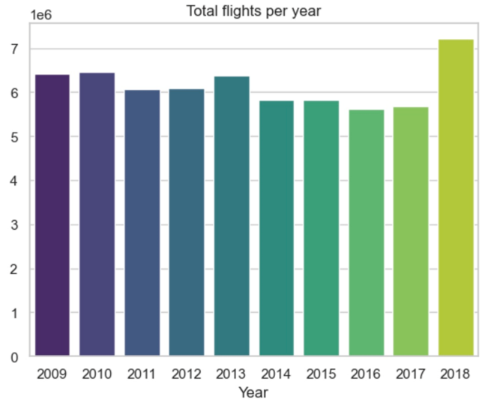


Figure 6: Total Flights per year

3.6.1 Total Flights by Carrier

Groups the data by carrier to compute the count of flights. By grouping data based on carriers, we calculate the frequency of flights for each airline. This graphical representation enhances our analysis of the distribution of flights across different airline operators.

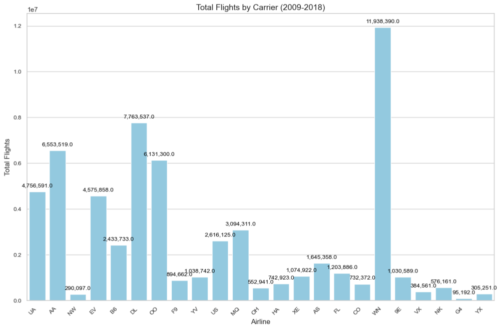


Figure 7: Total Flights by Carrier (2009-2018)

3.6.2 Percentage Delay by Carrier

Calculates the total delay for each carrier. Merges the total delay and actual air time DataFrames. Computes the ratio of delay time to actual elapsed time. Creates a bar plot to show the percentage delay by carrier.

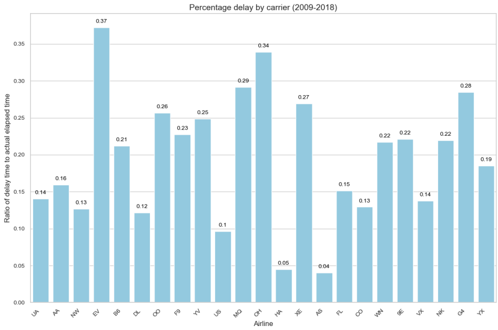


Figure 8: Percentage delay by carrier (2009-2018)

3.6.3 Total Delayed Flights by Carrier

Groups the data by carrier to count the total delayed flights. Merges this information with previous DataFrames. Creates a bar plot to show the total delayed flights by carrier.

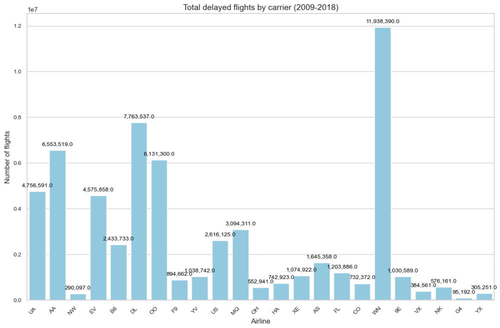


Figure 9: Total Delayed Flights by Carrier (2009-2018)

3.7 Total unique destinations by carrier (2009-2018)

The graph illustrates the distribution of total unique destinations served by each carrier within the dataset spanning the years 2009 to 2018. The primary objective is to discern the breadth of each airline's reach and identify patterns in the variety of destinations they connect.

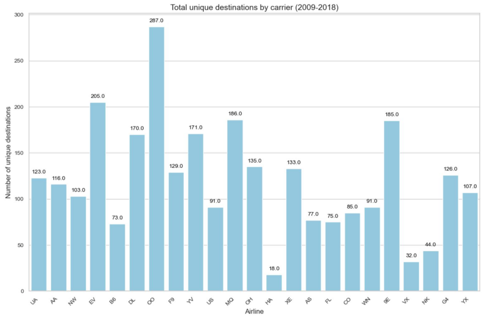


Figure 10: Total unique destinations by Carrier (2009-2018)

3.8 Modelling

3.8.1 Columns Selection

Defines columns to remove and keeps the relevant ones.

3.8.2 Class Distribution and Undersampling

Determines the class distribution before undersampling. Identifies the majority class and separates majority and minority classes. Performs undersampling on the majority class.

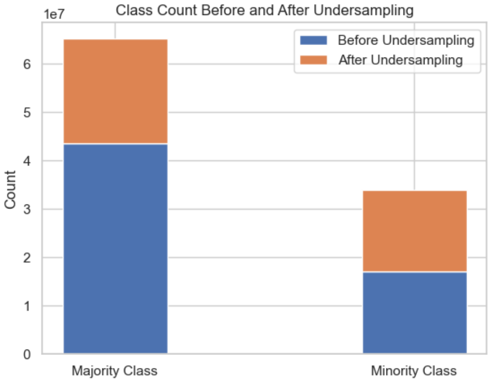


Figure 11: Class Count Before and After Undersampling

3.9 Model Training and Evaluation

The project employs scikit-learn models for binary classification, specifically Logistic Regression, XGBoost, Bagging Classifier, and Random Forest. The performance of each model is evaluated using key metrics such as accuracy, precision, recall, F1 score, and confusion matrix.

3.9.1 Logistic Regression

Description: Logistic Regression is a linear model suitable for binary classification tasks. It models the probability of an instance belonging to a particular class. In the context of the airline delay analysis, Logistic Regression is used to predict whether a flight will be delayed or not based on various features.

Performance Metrics:

- Accuracy: 83.05%.
- Precision: 87.27%
- Recall: 80.55%
- F1 Score: 83.78%

Confusion Matrix:

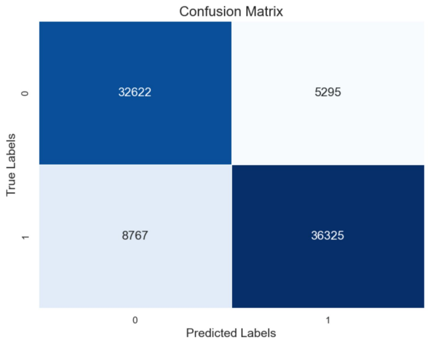


Figure 12: Confusion Matrix for Logistic Regression

3.9.2 XGBoost

Description: XGBoost, short for Extreme Gradient Boosting, stands out as a powerful ensemble learning technique rooted in the gradient boosting framework. Renowned for its exceptional performance, efficiency, and adeptness at managing substantial datasets, XGBoost has become a cornerstone in various machine learning applications.

Performance Metrics:

- Accuracy: 83.62%.
- Precision: 87.02%.
- Recall: 82.10%.
- F1 Score: 84.49%.

Confusion Matrix:

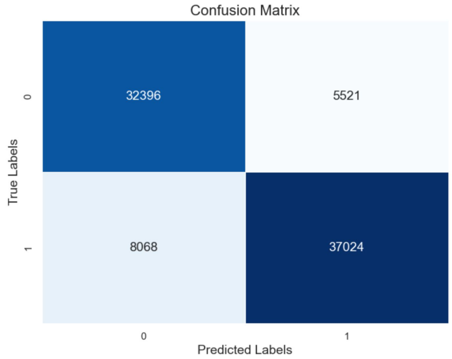


Figure 13: Confusion Matrix for XGBoost Classifier

3.9.3 Bagging Classifier

Description: The Bagging Classifier is an ensemble meta-estimator that fits base classifiers each on random subsets of the original dataset and then aggregates their individual predictions. Bagging helps reduce overfitting and variance by averaging the predictions of multiple models.

Performance Metrics:

- Accuracy: 79.70%.
- Precision: 81.11%.
- Recall: 81.63%.
- F1 Score: 81.37%.

Confusion Matrix:

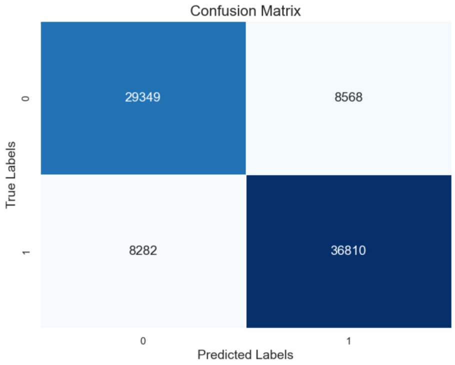


Figure 14: Confusion Matrix for Bagging Classifier

3.9.4 Random Forest

Description: Random Forest is an ensemble learning method that constructs a multitude of decision trees at training time and outputs the class that is the mode of the classes. It builds each tree independently and combines their predictions to improve accuracy and control overfitting.

Performance Metrics:

- Accuracy: 79.70%.
- Precision: 81.11%.
- Recall: 81.63%.

- F1 Score: 81.37%.

Confusion Matrix:

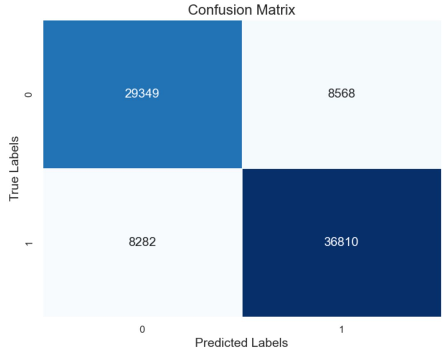


Figure 15: Confusion Matrix for Random Forest Classifier

3.10 Models Comparison

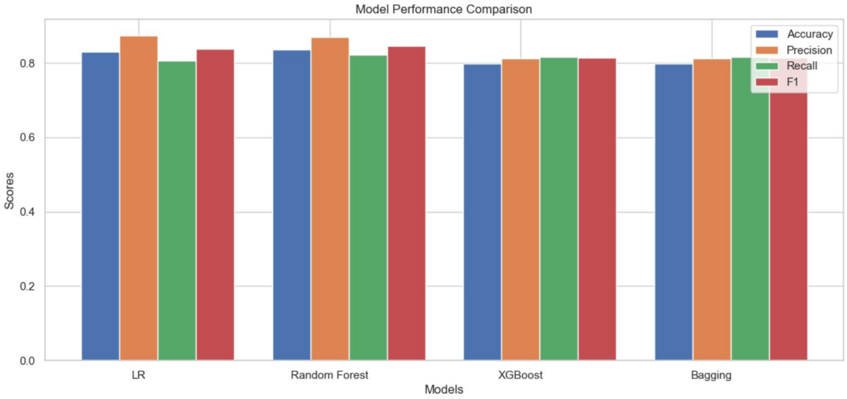


Figure 16: All 4 classification models comparison

4 Tools and Technologies

4.1 Technology

The magnitude of the dataset necessitated the utilization of various big data technologies for this project. The technologies employed are as follows:

- Coding Framework and Libraries: PySpark, Python 3.0, NumPy, Pandas, Scikit-learn

- Coding Platforms: Jupyter, Google Colab, and Databricks

- Visualizations: Seaborn, Matplotlib, Plotly

- Storage: Amazon S3, Google Cloud, and Drive

- Machine Learning Algorithms: Random Forest, Bagging, XGBoost, Decision Trees, Logistic Regression

4.1.1 Spark

Apache Spark stands as an open-source distributed computing system tailored for big data processing and analytics. It is renowned for its speed, versatility, and ease of implementation for distributed computing tasks. Initially based on Scala, PySpark provides a Python-based interface to Spark, which is the framework we adopted for our project.

5 Time Scale Analysis

LR (Small)	XGBoost (Small)	RF (Small)	Bagging (Small)	LR	XGBoost	RF	Bagging
0.054	0.525	2.99	2.42	0.26	3.41	32.25	26.78

Table 1: Time Scale Analysis (in seconds).

We found that increasing the sample size by 10 increases the time taken for model training by a factor of almost 100!

6 Insights

The analysis has offered a multitude of insights. We can observe:

- The number of flights have remained consistent over the years. That is, the demand for aviation has been steady for ten years.

- SouthWest Airlines, Delta Airlines and American Airlines are the top 3 flight carriers in the United States. SouthWest Airlines operates nearly twice the number of flights than Delta Airlines. It operated 11 million flights over a decade.

- ExpressJet Airlines (EV) has the highest delay to total time ratio. EV suffers the highest delay percentage of all flight carriers in the United States at 37%. However, they have operated 4.5 million flights. That means 37% of the time spent on 4.5 million flights was lost to delay.

- PSA Airlines (OH) has a delay of 34% for a total of 550k flights. For a smaller airline, this can add significant operational costs and severely impact revenue.

- The largest airlines (WN, DL, AA) have delays of 22%,12%, and 16% respectively. It is significantly lower than many other smaller airlines. However, due to their size this too can add significant operational costs to the airline.

- Alaska Airlines (AS) has been the most efficient with a delay of about 4%.

- Skywest Airlines (OO) has visited the maximum number of unique destinations. They have visited 287 different airports.

- Most other airlines have visited between 100-200 unique destinations over the span of ten years.

- Hawaiian Airlines (HA) has visited the least at 18 unique destinations indicating they are a small, focused aviation company.

- For passengers, they are least likely to suffer a delay in their flight schedule if they travel by Alaska Airlines, Hawaiian Airlines, and US Airways as they have the lowest delay percentage.

7 Challenges and Future Advancements

7.1 Challenges

- **Data Quality:**
  - Incomplete or inconsistent data could affect the accuracy of delay predictions. Ensuring data quality and handling missing values effectively is crucial.
- **Imbalanced Classes:**
  - The dataset may have imbalanced classes where the number of delayed flights significantly differs from on-time flights. Addressing this imbalance is essential for model training and evaluation.
- **Feature Selection:**
  - Choosing relevant features for predicting flight delays is challenging. Identifying the most influential factors requires domain knowledge and continuous refinement.
- **Scalability:**
  - As the dataset grows, scalability becomes an issue. Ensuring that the analysis is scalable for large datasets, especially in a distributed computing environment like PySpark, is important.
- **Model Interpretability:**
  - Some machine learning models, especially complex ones like XGBoost, may lack interpretability. Interpretable models are essential, especially in critical applications like airline operations.

7.2 Future Advancements

- **Real-Time Predictions:**
  - Implementing real-time delay predictions would be valuable for airlines to make instant decisions and enhance passenger experience.
- **Enhanced Feature Engineering:**
  - Continuously improving feature engineering by incorporating more relevant factors, such as weather patterns, air traffic, and airport conditions, could enhance prediction accuracy.
- **Anomaly Detection:**
  - Integrating anomaly detection techniques could help identify unusual patterns and potential disruptions in the flight schedule that may not be captured by traditional classification models.
- **Predictive Maintenance:**
  - Extending the analysis to predict potential maintenance issues based on historical delays can contribute to better resource management and reduce unexpected aircraft downtime.
- **Human Factors Integration:**
  - Incorporating human factors, such as air traffic controller strikes or pilot scheduling, could provide a more comprehensive view of the factors influencing flight delays.
- **Advanced Machine Learning Models:**
  - Exploring advanced machine learning models, including ensemble methods and deep learning architectures, may improve prediction accuracy.

- **Collaboration with Airlines:**
  - Collaborating with airlines to incorporate their operational insights and feedback can lead to more practical and applicable solutions.
- **Regulatory Compliance:**
  - Considering and incorporating regulatory compliance factors in delay predictions to assist airlines in adhering to regulations and optimizing operations.
- **User-Friendly Dashboards:**
  - Developing user-friendly dashboards or interfaces for airline staff to interact with and understand the delay predictions easily.
- **Environmental Impact Assessment:**
  - Evaluating the environmental impact of flight delays and incorporating sustainability considerations into decision-making processes.

8 Conclusion

In summary, this project extensively analyzed domestic flight punctuality to provide valuable insights for carriers and stakeholders. Through detailed examination of flight delays, the study aimed to enhance operational efficiency and customer satisfaction in air travel. The findings offer nuanced insights into dataset trends, challenges, and patterns, leveraging advanced data processing, visualization, and machine learning. We anticipate a positive impact on decision-making processes, fostering a travel environment with improved efficiency and satisfaction.

9 References

1. Sherry. Airline Delay Analysis. Kaggle. https://www.kaggle.com/datasets/sherrytp/airline-delay-analysis

2. Adrian Vera. Flight Delay EDA - Exploratory Data Analysis. Kaggle. https://www.kaggle.com/code/adveros/flight-delay-eda-exploratory-data-analysis

3. Yu Yaning, Hai Mo, Li Haifeng. A Classification Prediction Analysis of Flight Cancellation.

4. Maryam Farshchian Yazdi, Seyed Reza Kamel, Seyyed Javad Mahdavi Chabok, Maryam Kheirabadi. Flight delay prediction based on deep learning and Levenberg-Marquart algorithm.

5. Ahmed Elsayed, Mohamed Shaheen, Osama Badawy. Caching Techniques for Flight Delays Prediction in Big Data Using SparkR.

6. Kerim and Hose. Study of Delay Prediction in the US Airport Network.

7. Ripon Patgiri. Empirical Study on Airline Delay Analysis and Prediction.

8. Jun Chen, Meng Li. Chained Predictions of Flight Delay Using Machine Learning.