A Project Report On

Airline Delay Analysis and Prediction

Submitted By

Abubakar Vasi Shaikh (PL)
(220340325002)

Sanket Sanjay Desai
(220340325044)

Amitendra Veer Shyam Singh
(220340325006)

Krushna Ramlu Chitepwad
(220340325021)

Kaustubh Pradip Upadhye
(220340325020)

In partial fulfilment of
the requirements for the award of the degree of

Post Graduate Diploma
In
Big Data Analytics
(PG-DBDA)

Year 2022

Abstract

Nowadays, the aviation industry plays a crucial role in the world's transportation sector, and a lot of businesses rely on various airlines to connect them with other parts of the world. But, extreme weather conditions may directly affect the airline services by means of flight delays.

To solve this issue, accurately predicting these flight delays allows passengers to be well prepared for the deterrent caused to their journey and enables airlines to respond to the potential causes of the flight delays in advance to diminish the negative impact.

Table of Contents

1. Introduction:

For anyone who has travelled on an airplane, you may have experience with one of the inevitable pains of flying: the delays. Sometimes your plane arrives late, other times there may be a queue for take-off, occasionally the weather forces hour-long delays (or even cancellation); regardless of the reason for the delay, they pose a huge inconvenience for travellers.

Hence, given the vast amount of data on flight travels (16+ million flights annually in the U.S. alone), valuable insights can be drawn from this data to allow us to gain a better understanding of flight delays. Moreover, with the abundance in data, machine learning models can be trained to possibly predict these delays — something that may prove very valuable in for individual travellers and businesses alike. Thus, this is a topic that I hope to explore further by extracting potentially meaningful insights from available data and constructing and comparing machine learning models to hopefully predict flight delays.

2. Research Motivation:

Average aircraft delay is regularly referred to as an indication of airport capacity. Flight delay is a prevailing problem in this world. It's very tough to explain the reason for a delay. A few factors responsible for the flight delays like runway construction to excessive traffic are rare, but bad weather seems to be a common cause. Some flights are delayed because of the reactionary delays, due to the late arrival of the previous flight. It hurts airports, airlines, and affects a company's marketing strategies as companies rely on customer loyalty to support their frequent flying programs.

3. Problem Statement:

Flight delay is a serious and widespread problem across the world. Increasing flight delays place a significant strain on the air travel system. Every year 20% of airline flights are cancelled or delayed, costing passengers more than 20 billion dollars in money and their time. Our goal is to use exploratory analysis and machine learning models to predict airline departure and arrival delays by building a data pipeline on AWS.

4. Project Development:

In this section, there is an overview of the process of data mining and data modeling, from collecting the data, through the data preparation and finally the data modeling. Data cleaning and formatting can be considered as the most critical part of the whole project according to several data scientists . Figure 1 shows how the process of data mining works to extract knowledge using algorithms Loading the Dataset

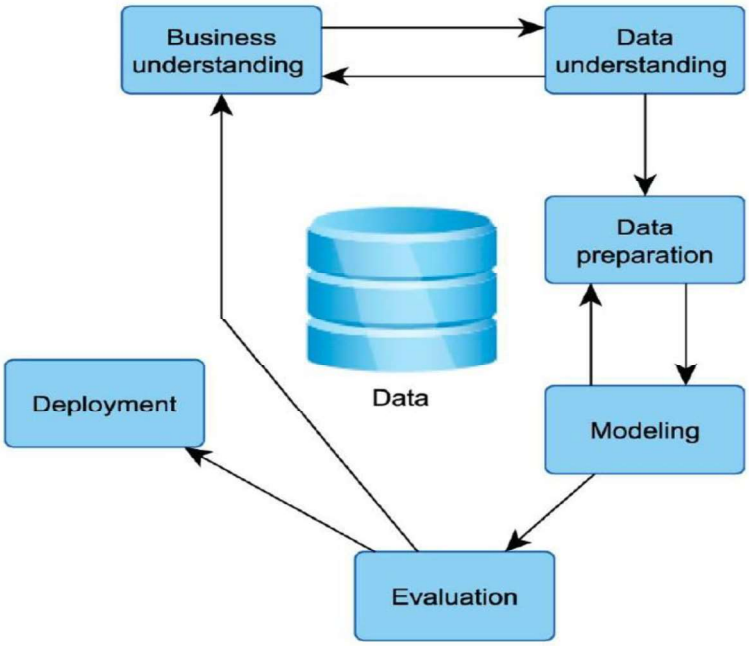

Figure 1: Project Development

5. Methodology and Implementation

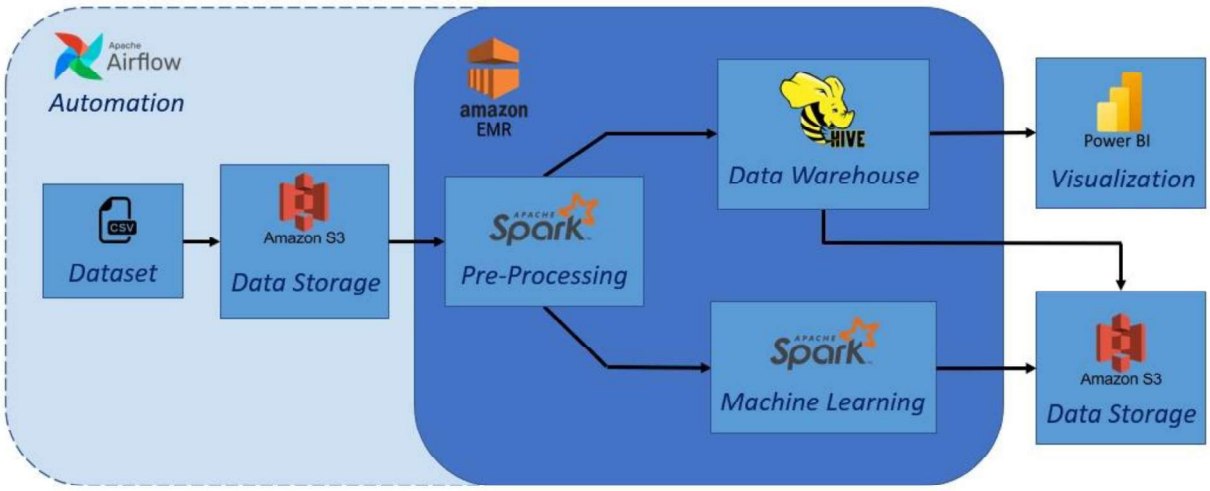

Figure 2: Project Architecture

Technologies used:

1. Apache Spark
2. Apache Hive
3. Apache Airflow
4. PowerBI
5. AWS (S3 and EMR)

A. Data Source:

The data source that we will be using in the is analysis is a dataset from Kaggle which contains U.S. flight data from 2009–2018. The rows of the dataset represent specific flights from that year, while the columns contain extensive information on the flight such as airline, flight date, departure delay, arrival delay, etc.

The entire dataset contains CSV files for each year, which in total amass to ~7 GB. Moreover, each file has approximately 6 million rows. As we will be running analysis on Visual Studio Code, which has a RAM and GB limit, we have chosen to only use the data for the year 2018. This dataset with 7.2 million rows will be sufficient for our EDA and modelling; using a larger combined dataset will significantly slow down the training of models and will likely only increase overall performance marginally.

Below is an image of what the dataframe looks like after being read.

	FL_DATE	OP_CARRIER	OP_CARRIER_FL_NUM	ORIGIN	DEST	CRS_DEP_TIME	DEP_TIME	DEP_DELAY	TAXI_OUT	WHEELS_OFF	WHEELS_ON	TAXI_IN	CRS_ARR_TIME	ARR_TIME	ARR_DELAY
0	2018-01-01	UA	2429	EWR	DEN	1517	1512.0	-5.0	15.0	1527.0	1712.0	10.0	1745	1722.0	-23.0
1	2018-01-01	UA	2427	LAS	SFO	1115	1107.0	-8.0	11.0	1118.0	1223.0	7.0	1254	1230.0	-24.0
2	2018-01-01	UA	2426	SNA	DEN	1335	1330.0	-5.0	15.0	1345.0	1631.0	5.0	1649	1636.0	-13.0
3	2018-01-01	UA	2425	RSW	ORD	1546	1552.0	6.0	19.0	1611.0	1748.0	6.0	1756	1754.0	-2.0
4	2018-01-01	UA	2424	ORD	ALB	630	650.0	20.0	13.0	703.0	926.0	10.0	922	936.0	14.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5674616	2017-12-31	UA	2421	IAH	LAS	750	744.0	-6.0	14.0	758.0	849.0	4.0	916	853.0	-23.0
5674617	2017-12-31	UA	2425	RSW	ORD	1611	1602.0	-9.0	12.0	1614.0	1753.0	12.0	1821	1805.0	-16.0
5674618	2017-12-31	UA	2426	SNA	DEN	1335	1334.0	-1.0	9.0	1343.0	1627.0	10.0	1649	1637.0	-12.0
5674619	2017-12-31	UA	2427	LAS	SFO	1115	1107.0	-8.0	11.0	1118.0	1224.0	15.0	1254	1239.0	-15.0
5674620	2017-12-31	UA	2429	EWR	DEN	1510	1612.0	62.0	28.0	1640.0	1827.0	11.0	1740	1838.0	58.0

12651227 rows × 20 columns

Figure 3: Airline Dataset

B. Data Storage:

For data storage, we have used AWS service i.e. S3. Amazon Simple Storage Service (Amazon S3) is an object storage service offering industry-leading scalability, data availability, security, and performance.

C. Data Pre-processing:

We live in the era of big data. We collect lots of data which allows to infer meaningful results and make informed business decisions. However, the raw data does not offer much unless it is processed and explored. In order to make the most out of raw data, we need a thorough exploratory data analysis process. Even if we build complex, well-structured machine learning models, we cannot just dump the raw data to them. The models get as good as the data we feed to them. As the amount of data increases, it gets trickier to analyze and explore the data. Thus we need to clean and transform the data to a suitable form. Firstly, we extracted our data in the form of csv file from S3 to HDFS in-order to clean and transform our using pyspark.

```
spark = SparkSession.builder.enableHiveSupport().getOrCreate()

df=spark.read.format("csv").option("header","true").option("inferSchema",'True').load(f's3://dbda-prj10-trigger/2017.csv')
```

Figure 4: Data Extraction

```
c=df.count()
for i in df.columns:
    if (((df.filter(df[i].isNull()).count())/c)*100)>80:
        df=df.drop(i)

df=df.withColumn("FL_DATE",to_date(col("FL_DATE"),"yyyy-MM-dd"))

df = df.withColumn("OP_CARRIER_FL_NUM",df["OP_CARRIER_FL_NUM"].cast(StringType()))

l={'UA':'United Airlines',
    'AS':'Alaska Airlines',
    '9E':'Endeavor Air',
    'B6':'JetBlue Airways',
    'EV':'ExpressJet',
    'F9':'Frontier Airlines',
    'G4':'Allegiant Air',
    'HA':'Hawaiian Airlines',
    'MQ':'Envoy Air',
    'NK':'Spirit Airlines',
    'OH':'PSA Airlines',
    'OO':'SkyWest Airlines',
    'VX':'Virgin America',
    'WN':'Southwest Airlines',
    'YV':'Mesa Airline',
    'YX':'Republic Airways',
    'AA':'American Airlines',
    'DL':'Delta Airlines'}
l=list(l.items())
for i in range(18):
    df=df.withColumn('OP_CARRIER', regexp_replace('OP_CARRIER', l[i][0], l[i][1]))

df1=df.filter(df['ARR_DELAY']>0)

df1=df1.withColumn('FL_YEAR',year(df1.FL_DATE))
df1=df1.withColumn('FL_DAYOFWEEK',dayofweek(df1.FL_DATE))
df1=df1.withColumn('FL_MONTH',month(df1.FL_DATE))
```

Figure 5: Data Pre-processing

D. Data Loading:

Comma-separated value (CSV) files and, by extension, other text files with separators can be imported into a Spark DataFrame and then stored as a Hive table.

```
s3_location = 's3://dbda-final-project10/hive/2016/'
df1.write.mode('overwrite').saveAsTable(f'default.airline_2016', path=s3_location)
```

Figure 6: Data Loading to Hive

E. Data Visualization:

After our data is cleaned, we are ready for visualizing our data to gain some valuable insights as well as present our project to non-technical stake holders in a better way.

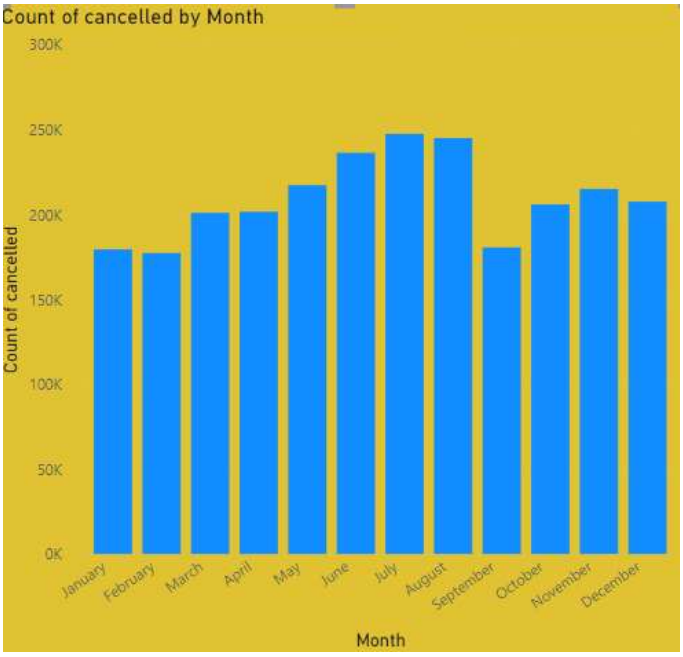

Figure 7: Count of cancelled flights by month

Figure 8: Count of cancelled flights by day

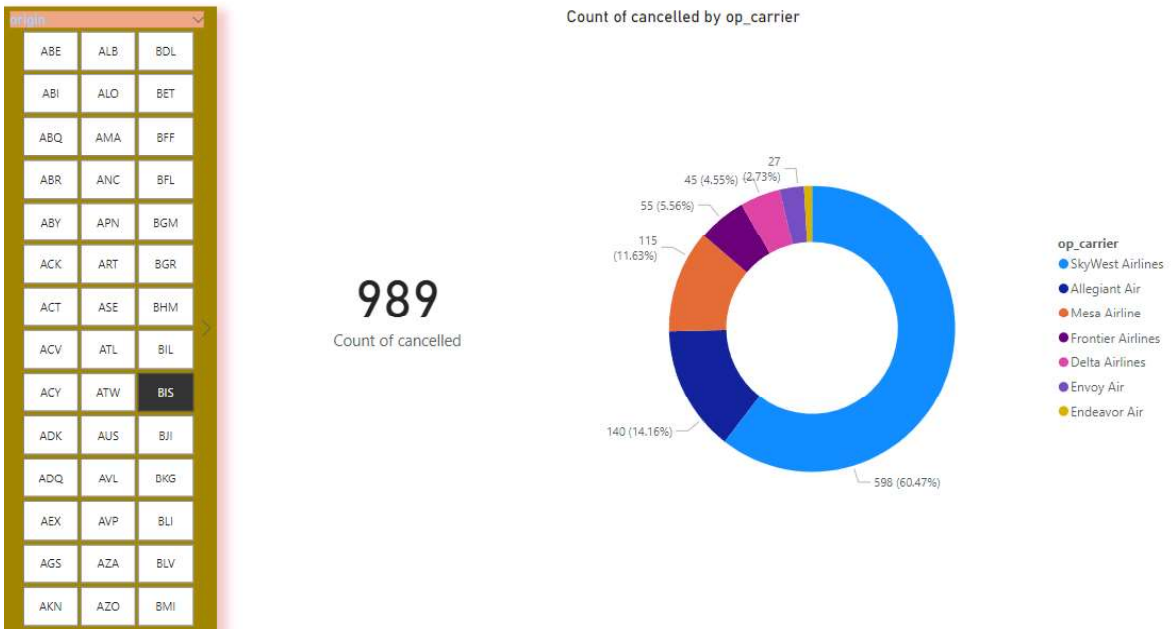

Figure 9: Count of cancelled flights by Airlines

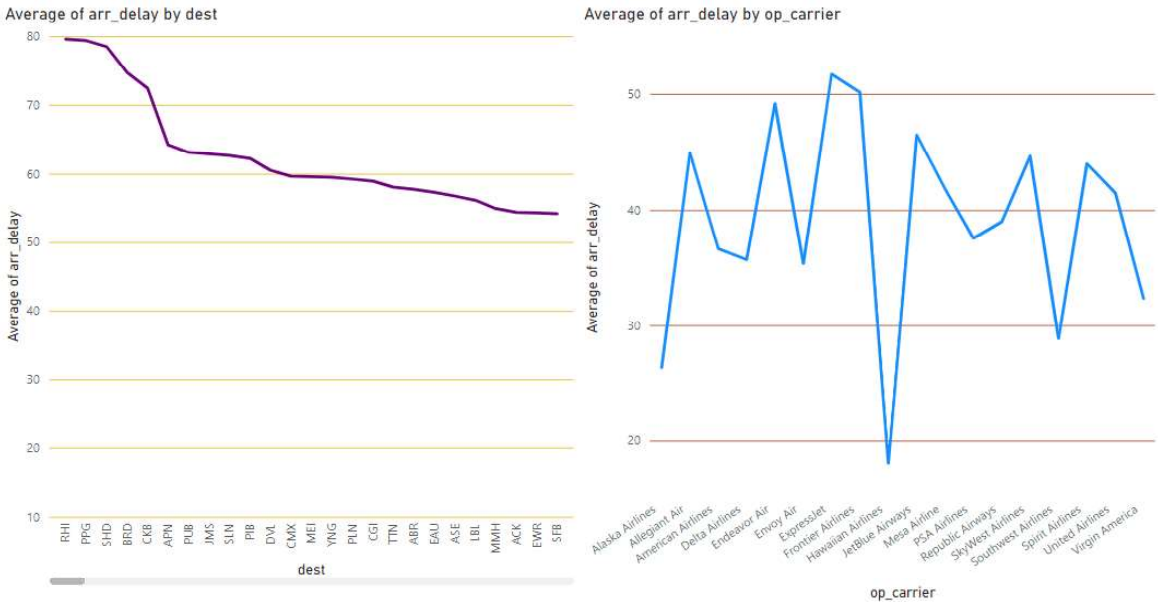

Figure 10: Average Delay by Airlines

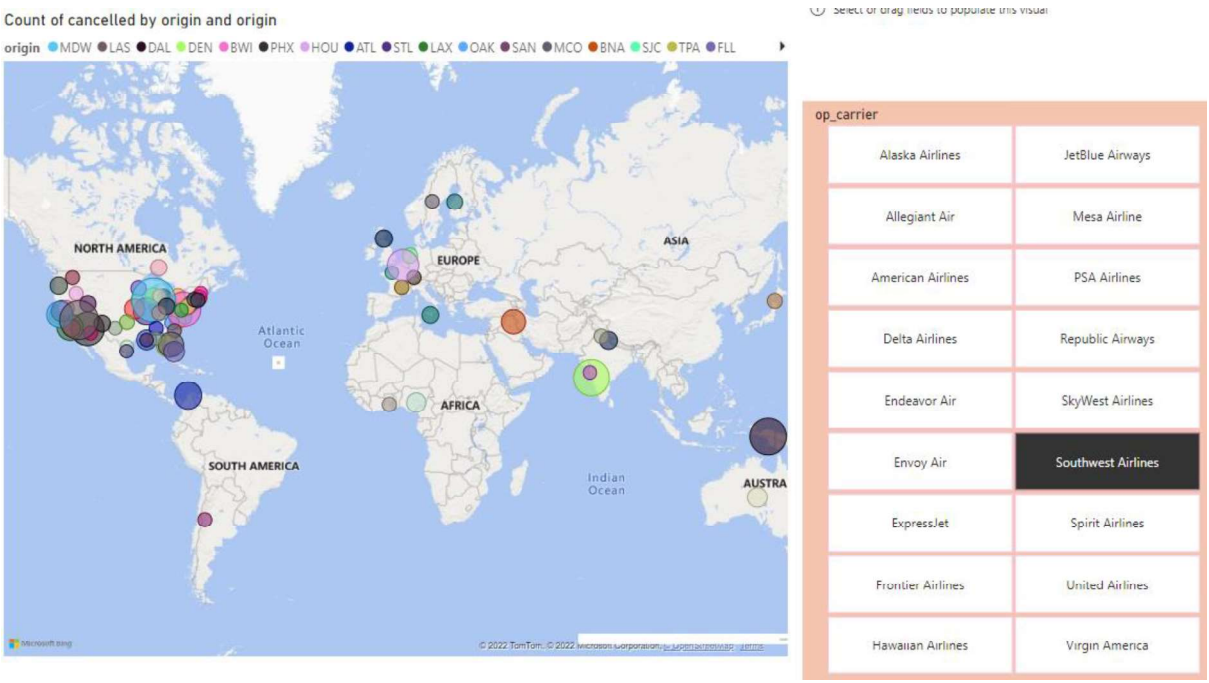

Figure 11: Count of cancelled airlines by origin

F. Machine Learning:

E.1.: Transformation:

In-order for prediction using different machine learning algorithm, we will have to transform our data accordingly. We converted continuous values of DEP_TIME into a categorical for that helps to improve our modelling. We also casted few features to double as the modelling considers values with double data type only.

We further used String Indexer to transform categorical string values into numerical ones. We selected the most relevant features out of our dataframe.

We then imputed null values in the dataset with an imputer library with median values of the feature.

```
bd1 = df1.withColumn('Delayed', (df.ARR_DELAY >=15).cast('int'))
bd1.createOrReplaceTempView("bd1")
bd1 = spark.sql("select *, case \
            when DEP_TIME <= 800 then 1 \
            when 800 < DEP_TIME and DEP_TIME <= 1200 then 2 \
            when 1200 < DEP_TIME and DEP_TIME <= 1600 then 3 \
            when 1600 < DEP_TIME and DEP_TIME <= 2100 then 4 \
            else 1 end as TimeSlot \
            from bd1")

bd1=bd1.withColumn("FL_DAYOFWEEK",col("FL_DAYOFWEEK").cast('double'))
bd1=bd1.withColumn("FL_MONTH",col("FL_MONTH").cast('double'))
bd1=bd1.withColumn("TimeSlot",col("TimeSlot").cast('double'))
bd1=bd1.withColumn("Delayed",col("Delayed").cast('double'))

from pyspark.ml.feature import StringIndexer
indexer1 = StringIndexer(inputCol='OP_CARRIER',outputCol='INDEX_CARRIER')
bd2=indexer1.fit(bd1).transform(bd1)
indexer2 = StringIndexer(inputCol='ORIGIN',outputCol='INDEX_ORIGIN')
bd3=indexer2.fit(bd2)
bd4=bd3.transform(bd2)
bd5=bd4.select('DEP_DELAY',
            'DISTANCE',
            'FL_DAYOFWEEK',
            'INDEX_CARRIER',
            'TimeSlot',
            'FL_MONTH',
            'ACTUAL_ELAPSED_TIME',
            'INDEX_ORIGIN',
            'Delayed')


from pyspark.ml.feature import Imputer
imputer = Imputer(
    inputCols = bd5.columns,
    outputCols = ["{}".format(a) for a in bd5.columns]
).setStrategy("median")
bd6=imputer.fit(bd5)
bd7=bd6.transform(bd5)
bd7=bd7.toPandas()
```

Figure 12: ML Transformations

E.2.: Prediction:

The transformed is can now be applied machine learning algorithm for prediction. After trying out several algorithm, the most suitable algorithm for our dataset the Logistic Regression which is basic but excellent algorithm for Binary Classification.

```
from sklearn import model_selection
from sklearn.linear_model import LogisticRegression
X = df.iloc[:,:-1].values
Y= df.iloc[:,-1].values
test_size = 0.33
seed = 7
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X, Y, test_size=test_size, random_state=seed)
model = LogisticRegression()
model.fit(X_train, Y_train)
```

Figure 13: ML Algoritthm

6. Conclusion:

We first made sure to understand the airline dataset, our task, and the metric by which our submissions will be judged. Then, we performed a fairly simple EDA to try and identify relationships and trends that may help our modelling. Along the way, we performed necessary pre-processing steps such as indexing categorical variables, imputing missing values, and scaling features to a range. Then, we constructed new features out of the existing data to see if doing so could help our model.

Once the data exploration and feature engineering were complete, transferred our Data to Hive Warehouse and further moved to PowerBI through Amazon Hive ODBC. Through visualization, the highest average delay is by Sky West Airline with 79.54 min. We also concluded that the most delayed flight are during monsoon season which itself could be cause of flight delay.

We then implemented a Logistic Regression Model to predict whether Airline will be delayed for years 2009-2017. We then deployed the model on Amazon Web Service. As soon as we run Pyspark script on AWS EMR, we get to know corresponding predictions in a csv file and visualizations in PowerBI.