A Project Report On

Airline Delay And Cancellation

Submitted By

Sumit Bansod(PL)
(220940325081)
Saurav Sharma
(220940325063)
Dipti Patil
(220940325028)
Keshav Yawale
(220940325036)
Gauri Pandey
(220940325033)

Guided by
Vineeta Singh,CDAC Kharghar

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

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

Year 2022-23

Abstract

The airline industry is prone to delays, which can cause significant inconvenience for passengers and incur costs for airlines. This project report aims to analyze airline delay data to identify patterns and trends and suggest strategies to improve operational efficiency and reduce delays.

The project uses a combination of regression analysis, machine learning, and network analysis to analyze a large dataset of airline delay data. The dataset includes information on flight schedules, arrival and departure times, weather conditions, and airline operations.

The analysis reveals several key findings, including the most common causes of delays, the airlines and airports with the highest delay rates, and the impact of weather on delays. Based on the analysis, the report suggests several strategies for reducing delays. The report also recommends further research to explore the impact of other factors

Overall, the project report provides valuable insights into the complex factors that contribute to airline delays and suggests strategies for improving operational efficiency and reducing delays in the airline industry.

Table of Contents

1. Introduction:

Passengers may experience difficulty and annoyance as a result of airline delays, which can also have a large financial impact on airlines and airports. So, in order to enhance their operations and cut down on delays, airlines and airports must examine delay data and find patterns and trends.

The information on flight schedules, actual departure and arrival times, the causes of delays, and other pertinent details were gathered from a major airline for inclusion in this report. The information contains both domestic and international flights throughout the course of a year..

The report will start off by giving a general overview of the data and the analysis techniques. The analysis' findings, including information on delay lengths and causes of delays, will subsequently be presented. On the basis of the analysis, the report will then offer potential options for minimising delays.

The overall goal of this project is to offer useful insights on airline delays as well as offer workable solutions for enhancing airline operations and decreasing passenger delays.

2. Research Motivation:

Airline delays can cost the company money, raise operating expenses, and harm its brand. Moreover, delays may cause customers to be less satisfied, which may ultimately harm an airline's bottom line. This airline delay and analysis project report was created with the goal of addressing an important problem in the aviation sector: airline delays. In addition to annoying passengers, flight delays have a major financial impact on both airlines and airports.

This project report intends to offer insights into potential solutions for minimising delays and enhancing the overall passenger experience by examining airline delay data and identifying factors that lead to delays. The results of this study can be used by airports and airlines to make data-driven decisions and implement operational changes that will shorten wait times and increase customer satisfaction. Also, the report's conclusions and suggestions may assist airlines in enhancing their productivity and competitiveness, which is crucial in the fiercely competitive aviation sector.

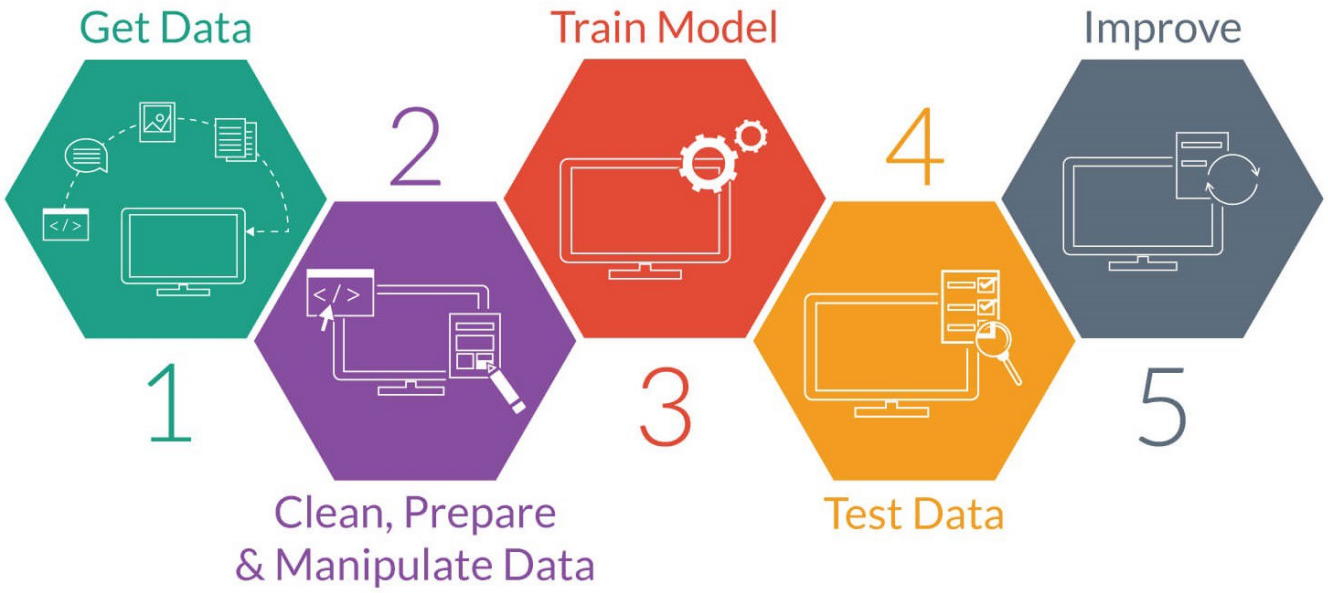
In order to address this pressing issue in the aviation sector and to offer workable solutions for minimising delays, strengthening operations, and improving the overall passenger experience, this airline delay and analysis project report was inspired.

3.Problem Statement:

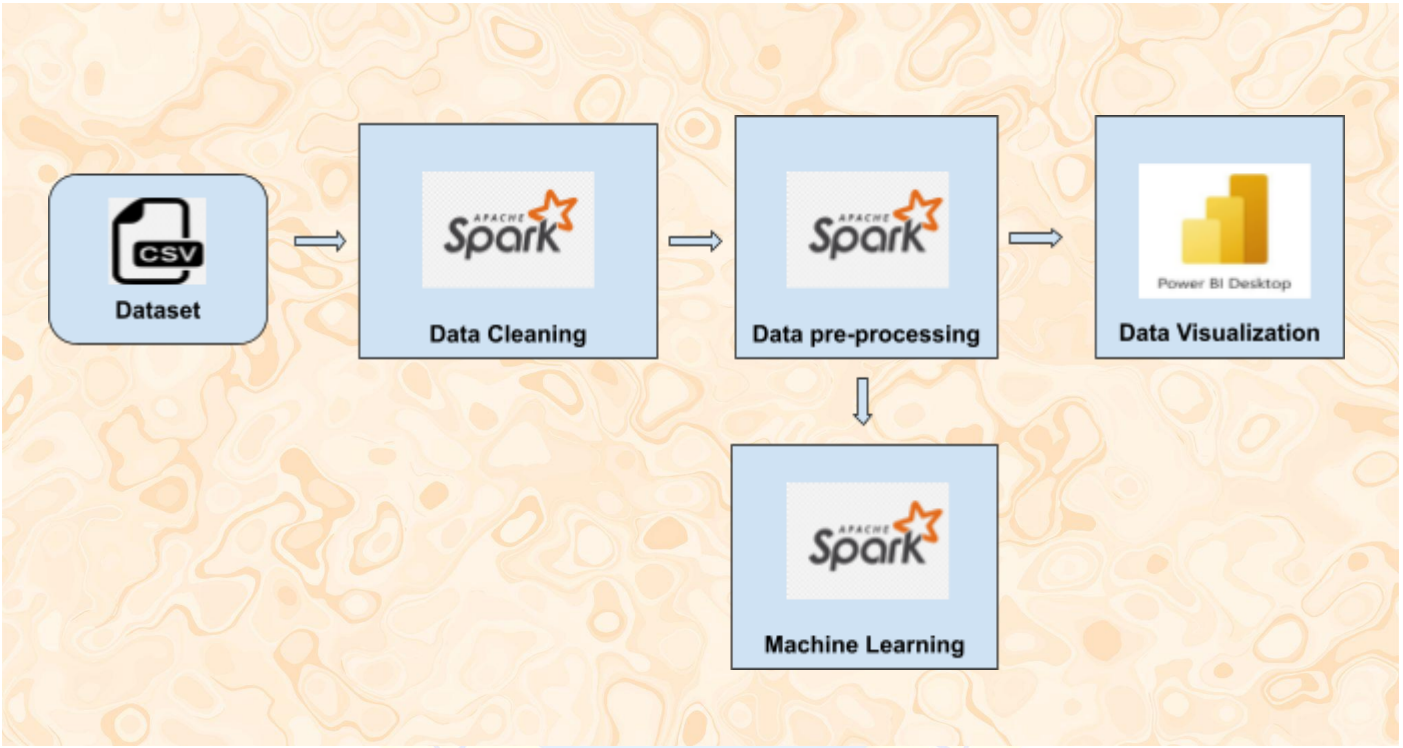
Airline Delay and Cancellation is a pervasive problem in the aviation industry that affects not only travelers, but also airlines and airport operators.. Our aim is to create an effective and informative visual representation of flight delay and cancellation data for major US airlines, to identify patterns and insights that can improve air travel. This will benefit airlines, airports, and passengers by better understanding the factors contributing to delays and cancellations.We'll be using exploratory analysis and machine learning models to predict airline departure and arrival delays.

4.Project development:

An overview of the data mining and data modelling process, from data collection to data preparation to data modelling, is provided in this section. According to numerous data scientists, data cleaning and formatting can be thought of as the most important element of the entire project. Figure below illustrates how algorithms used in data mining to extract knowledge from a dataset work.



5 . Methodology



Technologies used:

1. Apache Spark
2. Apache Spark MLlib
3. Apache Spark SQL
4. PowerBI

A.Data sources:

The data source that we have 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.

Each year's worth of data is contained in a CSV file, totaling about 7 GB for the complete dataset. Moreover, there are roughly 6 million rows in each file. We have decided to just use the data for the year 2018 because we will be conducting research on Anaconda Navigator , which has a RAM and GB constraint. For our EDA and modelling, this dataset with 7.2 million rows will be adequate; utilising a bigger combined dataset will greatly slow down the training of models and perhaps only slightly improve overall performance.

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

```
In [5]: display(airline.limit(5).toPandas())
```

	FL_DATE	OP_CARRIER	OP_CARRIER_FL_NUM	ORIGIN	DEST	CRS_DEP_TIME	DEP_TIME	DEP_DELAY	TAXI_OUT	WHEELS_OFF	...	CRS_ELAPSED_TIM
0	2018-01-01	UA	2429	EWR	DEN	1517	1512.0	-5.0	15.0	1527.0	...	268.
1	2018-01-01	UA	2427	LAS	SFO	1115	1107.0	-8.0	11.0	1118.0	...	99.
2	2018-01-01	UA	2426	SNA	DEN	1335	1330.0	-5.0	15.0	1345.0	...	134.
3	2018-01-01	UA	2425	RSW	ORD	1546	1552.0	6.0	19.0	1611.0	...	190.
4	2018-01-01	UA	2424	ORD	ALB	630	650.0	20.0	13.0	703.0	...	112.

5 rows × 28 columns

B. Data preprocessing:

Big data is a reality in our time. We've gathered a lot of data, allowing us to draw conclusions that are meaningful and guide business decisions.

Yet, unless it is processed and investigated, the raw data does not give much. We require a rigorous exploratory data analysis procedure in order to get the most out of raw data. We cannot simply throw the raw data to machine learning models, no matter how intricate or well-organized they are. The quality of the models depends on the data we give them. The analysis and exploration of the data get more challenging as the volume of data grows.

As a result, we must clean and change the data into an appropriate format.

1.Import necessary libraries and initialize a SparkSession:

```
import findspark
findspark.init()

import pyspark # only run after findspark.init()
from pyspark.sql import SparkSession

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

2.Read in the dataset

```
airline=spark.read.format("csv").option("header","true").option("inferSchema",'True').load(r'D:\Project airline\DATASET\2018.csv'
```

3.Check the schema and preview the data:

```
]: airline.printSchema()
```

```
airline.printSchema()  # getting the infomation about the schema i.e, columns and their datatypes
```

```
root
 |-- FL_DATE: timestamp (nullable = true)
 |-- OP_CARRIER: string (nullable = true)
 |-- OP_CARRIER_FL_NUM: integer (nullable = true)
 |-- ORIGIN: string (nullable = true)
 |-- DEST: string (nullable = true)
 |-- CRS_DEP_TIME: integer (nullable = true)
 |-- DEP_TIME: double (nullable = true)
 |-- DEP_DELAY: double (nullable = true)
 |-- TAXI_OUT: double (nullable = true)
 |-- WHEELS_OFF: double (nullable = true)
 |-- WHEELS_ON: double (nullable = true)
 |-- TAXI_IN: double (nullable = true)
 |-- CRS_ARR_TIME: integer (nullable = true)
 |-- ARR_TIME: double (nullable = true)
 |-- ARR_DELAY: double (nullable = true)
 |-- CANCELLED: double (nullable = true)
 |-- CANCELLATION_CODE: string (nullable = true)
 |-- DIVERTED: double (nullable = true)
 |-- CRS_ELAPSED_TIME: double (nullable = true)
 |-- ACTUAL_ELAPSED_TIME: double (nullable = true)
 |-- AIR_TIME: double (nullable = true)
 |-- DISTANCE: double (nullable = true)
 |-- CARRIER_DELAY: double (nullable = true)
 |-- WEATHER_DELAY: double (nullable = true)
 |-- NAS_DELAY: double (nullable = true)
 |-- SECURITY_DELAY: double (nullable = true)
 |-- LATE_AIRCRAFT_DELAY: double (nullable = true)
 |-- Unnamed: 27: string (nullable = true)
```

4.Convert columns to appropriate data types:

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

```
from pyspark.sql.types import StringType

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

```
df.select('OP_CARRIER_FL_NUM').dtypes
```

```
[('OP_CARRIER_FL_NUM', 'string')]
```

5. Feature Transformation

```
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).transform(bd2)

bd3.groupBy('OP_CARRIER','INDEX_CARRIER').count().sort('INDEX_CARRIER').show()
```

6.Feature Selection

```
bd4=bd3.select('DEP_DELAY',
               'DISTANCE',
               'FL_DAYOFWEEK',
               'INDEX_CARRIER',
               'TimeSlot',
               'Delayed',
               'FL_MONTH',
               'ACTUAL_ELAPSED_TIME',
               'INDEX_ORIGIN')

bd4.limit(10).toPandas()
```

7. Imputing Null values

```
null_values(bd4)
```

```
DEP_DELAY ----------- > 0.047574571252558574
DISTANCE ------------ > 0.0
FL_DAYOFWEEK -------- > 0.0
INDEX_CARRIER ------- > 0.0
TimeSlot ------------ > 0.0
Delayed ------------- > 0.0
FL_MONTH ------------ > 0.0
ACTUAL_ELAPSED_TIME - > 3.974483813914668e-05
INDEX_ORIGIN -------- > 0.0
```

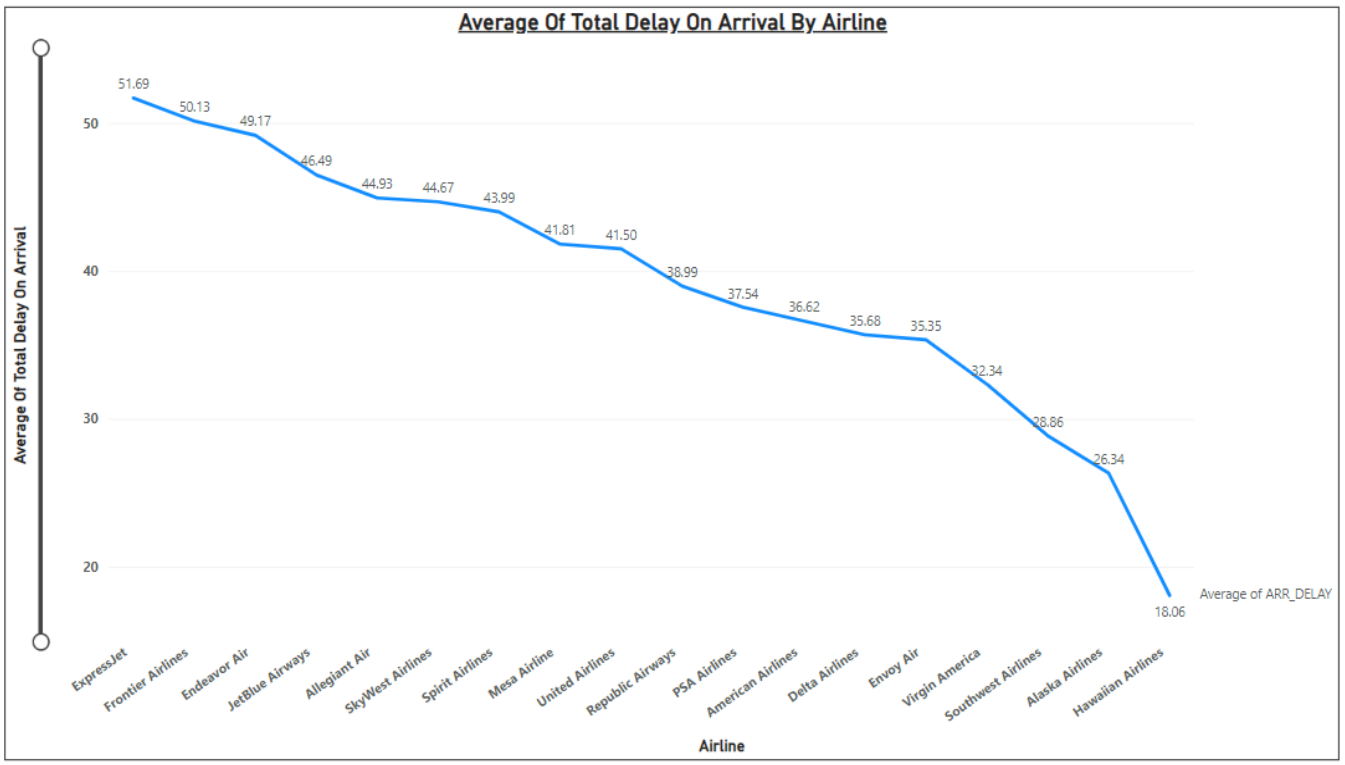
```
from pyspark.ml.feature import Imputer
```

```
imputer = Imputer(
    inputCols = bd4.columns,
    outputCols = ["{}".format(a) for a in bd4.columns]
).setStrategy("median")
```

C. Data Visualization:

After our data has been cleansed, we are prepared to visualise it in order to acquire some important insights and effectively communicate our project to non-technical stakeholders..

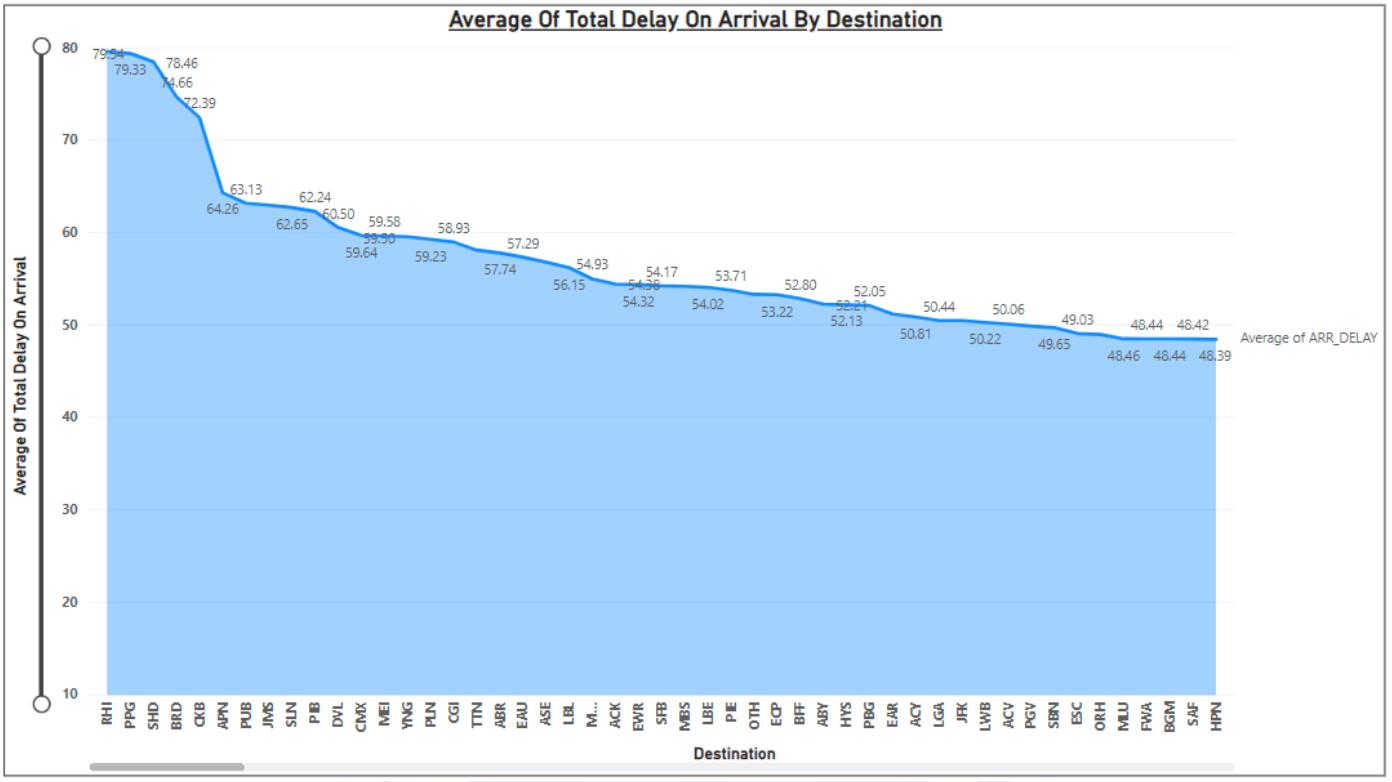
A.Average total Delay on Arrival by Airline



Observation:

Expressjet has an average of 51.69 which is the highest amount of average delay.

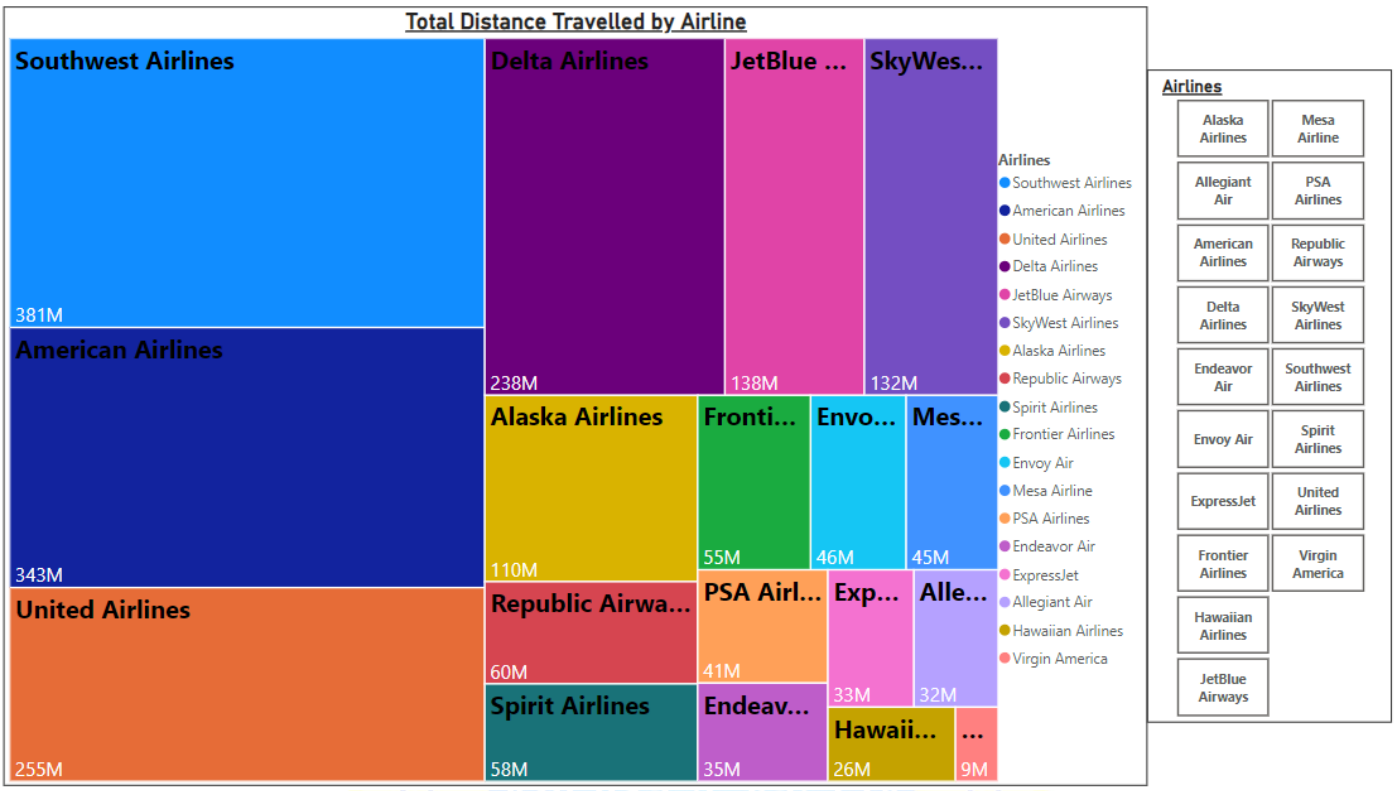
B. Average of total delay on arrival in minutes by destination



Observation:

(RHI) Rhinelander–Oneida Country Airport has highest average of total delay on arrival in minute

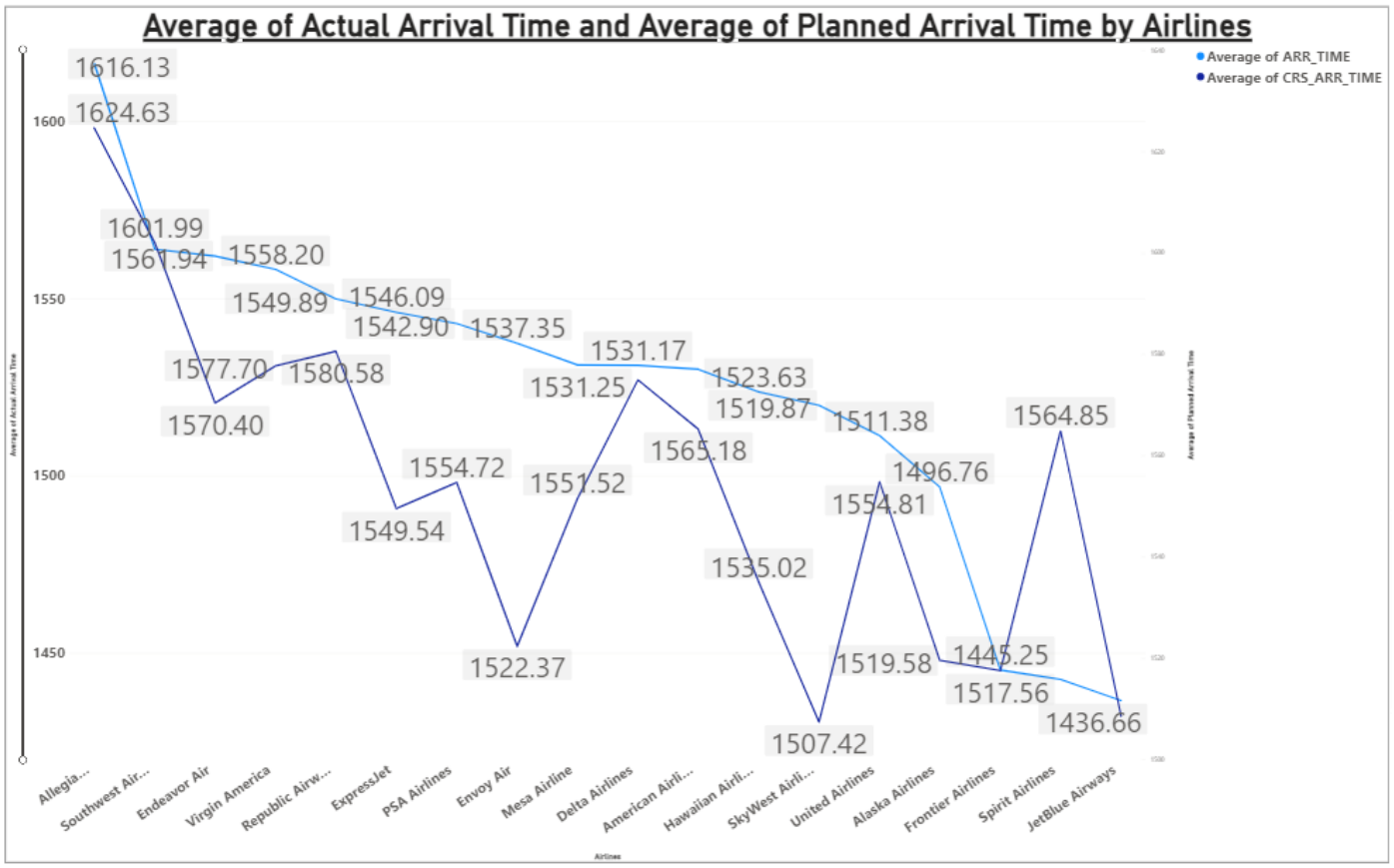
C. Total Distance Travelled by Airline



Observation:

Southwest airline cover highest distance so it is busiest airline

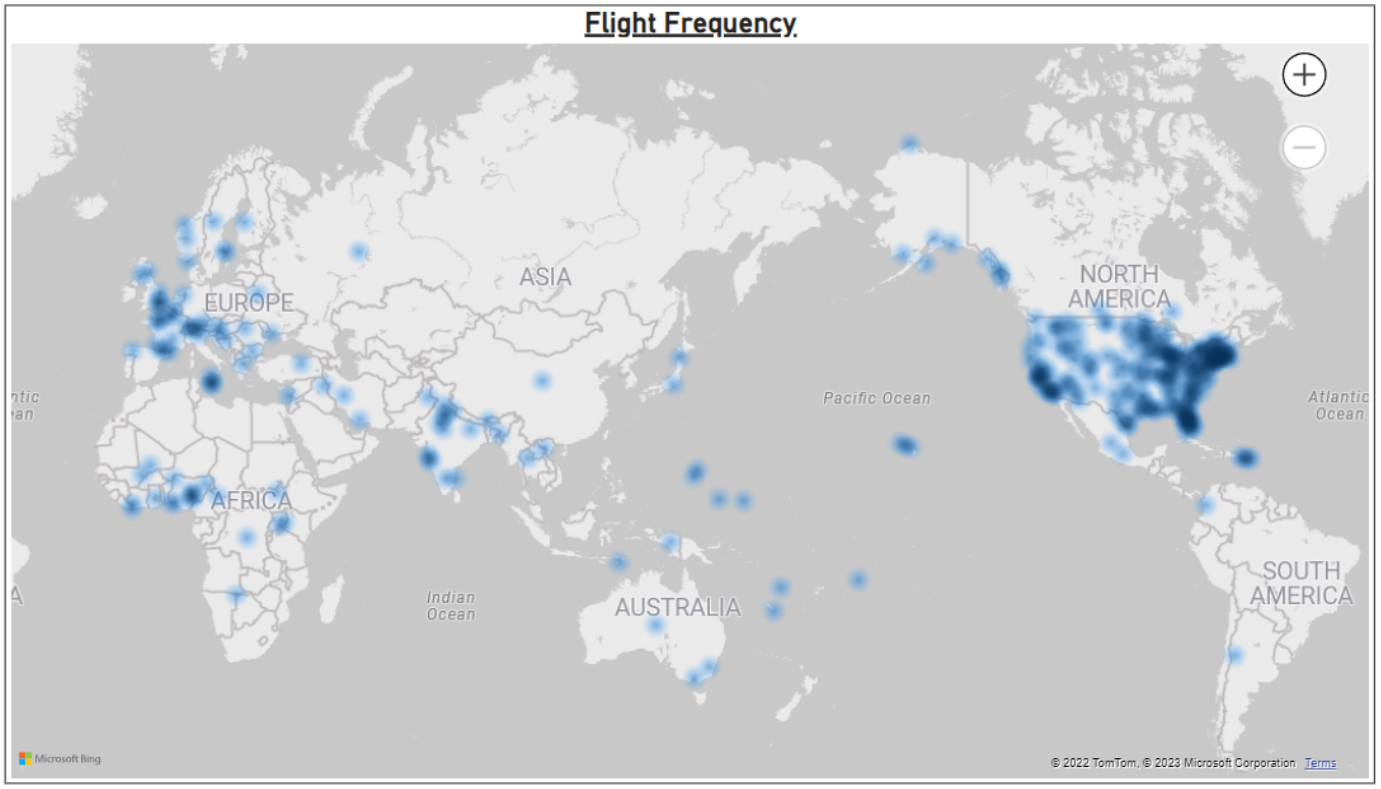
D. Average of actual arrival time by Airline



Observation:

Here we have compared Average of actual arrival time with planned arrival time

E. Busiest routes

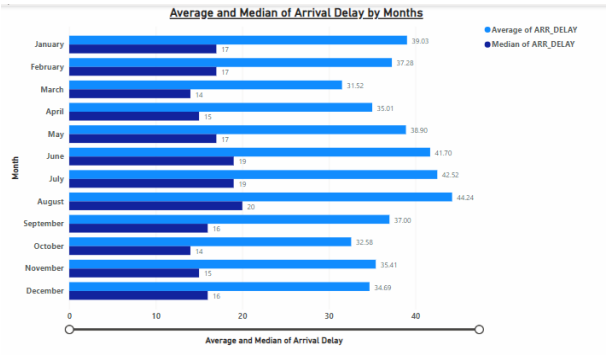


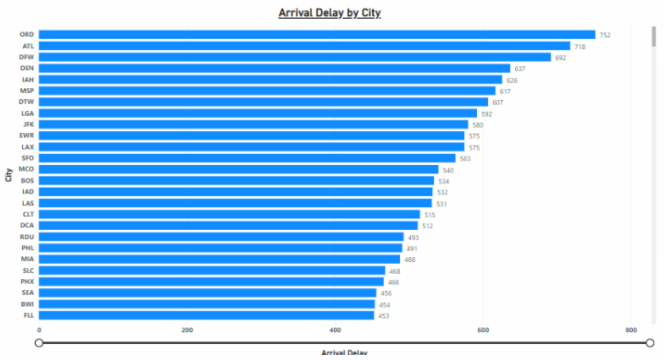
Observations:

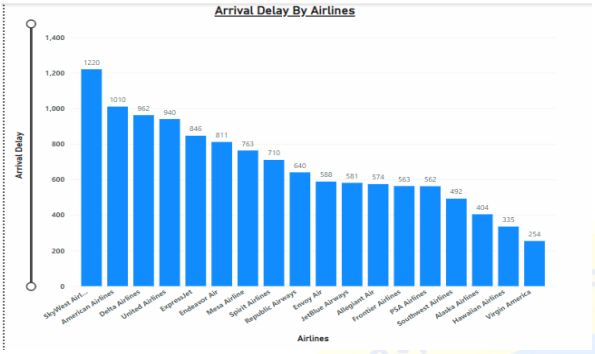
New York, florida, London, new delhi, paris, malta, hawaii has most number of flights running (so they are busiest route)

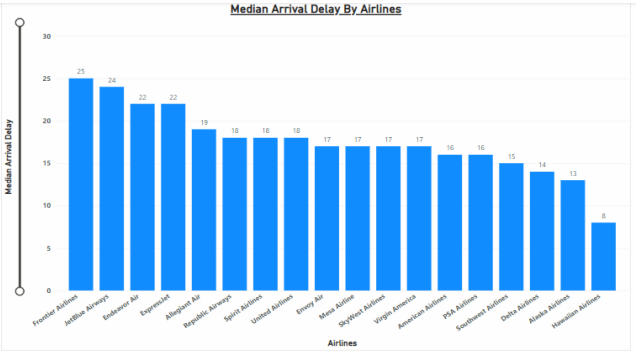
F. Arrival Delays

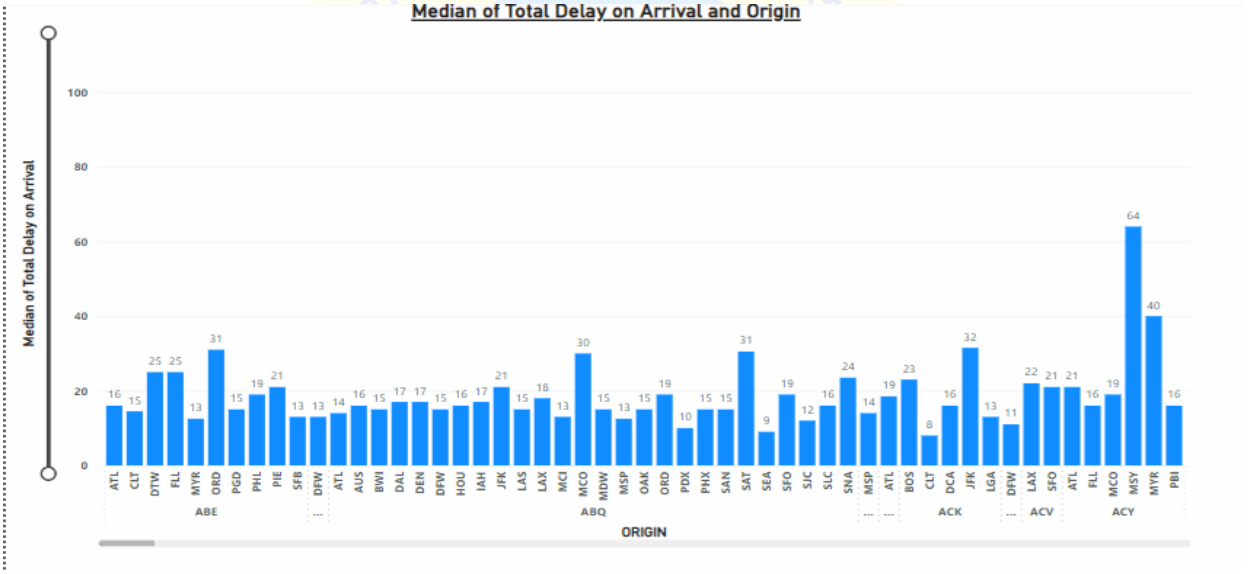
- Arrival delay month
- Arrival delay by airlines
- Arrival Delay by City
- Arrival Delay in Airlines by Time
- Median Arrival Delay By Airlines
- Median of Total Delay on Arrival and ORIGIN

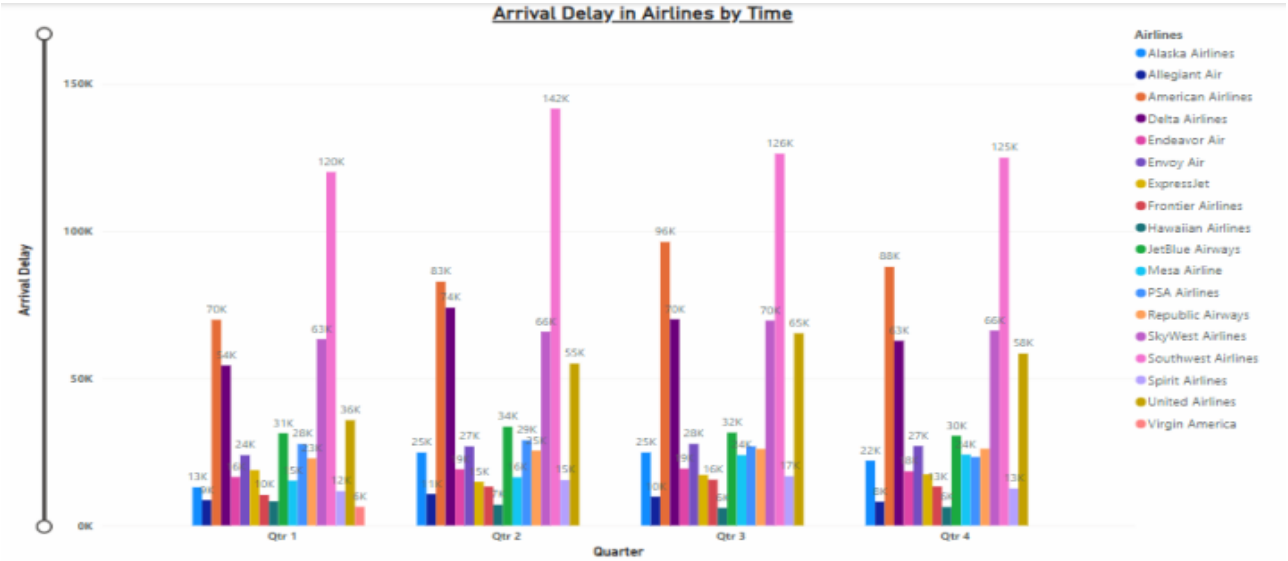

Average and Median of Arrival Delay by Months
Average of ARR_DELAY
Median of ARR_DELAY
Month
January
February
March
April
May
June
July
August
September
October
November
December
39.03
17
37.28
17
31.52
14
35.01
15
38.90
17
41.70
19
42.52
19
44.24
20
37.00
16
32.58
14
35.41
15
34.69
16
0
10
20
30
40
Average and Median of Arrival Delay


Arrival Delay by City
City
ORD
ATL
DFW
DEN
IAH
MSP
DTW
LGA
JFK
EWR
LAX
SFO
MCO
BOS
IAD
LAS
CLT
DCA
RDU
PHL
MIA
SLC
PHX
SEA
BWI
FLL
752
718
682
637
626
617
607
592
580
575
575
563
540
534
532
531
515
512
493
491
488
468
466
456
454
453
0
200
400
600
800
Arrival Delay


Arrival Delay By Airlines
Arrival Delay
1,400
1,200
1,000
800
600
400
200
0
1220
1010
962
940
846
811
763
710
640
588
581
574
563
562
492
404
335
254
Airlines


Median Arrival Delay By Airlines
Median Arrival Delay
30
25
20
15
10
5
0
25
24
22
22
19
18
18
18
17
17
17
17
16
16
15
14
13
8
Airlines


Median of Total Delay on Arrival and Origin
Median of Total Delay on Arrival
100
80
60
40
20
0
ABE
ABQ
ACK
ACV
ACY
ORIGIN

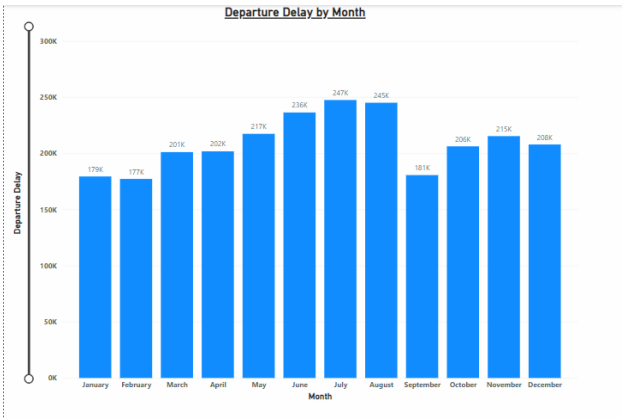


Observations:

- The month of July saw the longest arrival delays, followed by SkyWest Airlines, Ord City, and Southwest Airlines in the second quarter.The longest average arrival delay is with Frontier Airlines.

G.Departure Delay

- Dep Delay by month
- Count of total delay on departure by Airlines

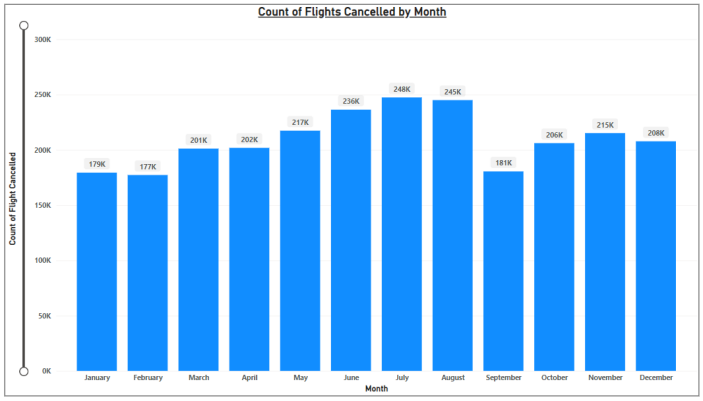


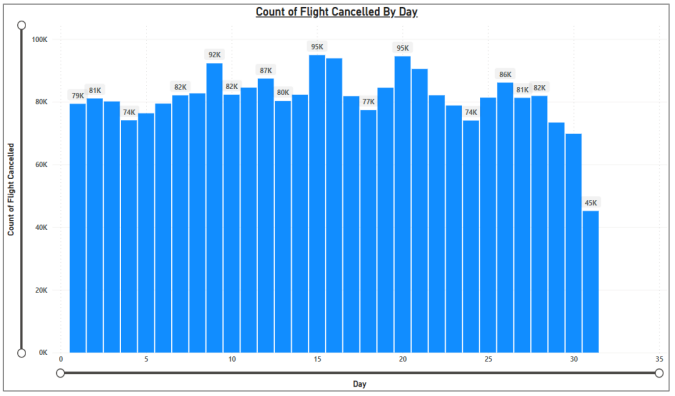
Observations:

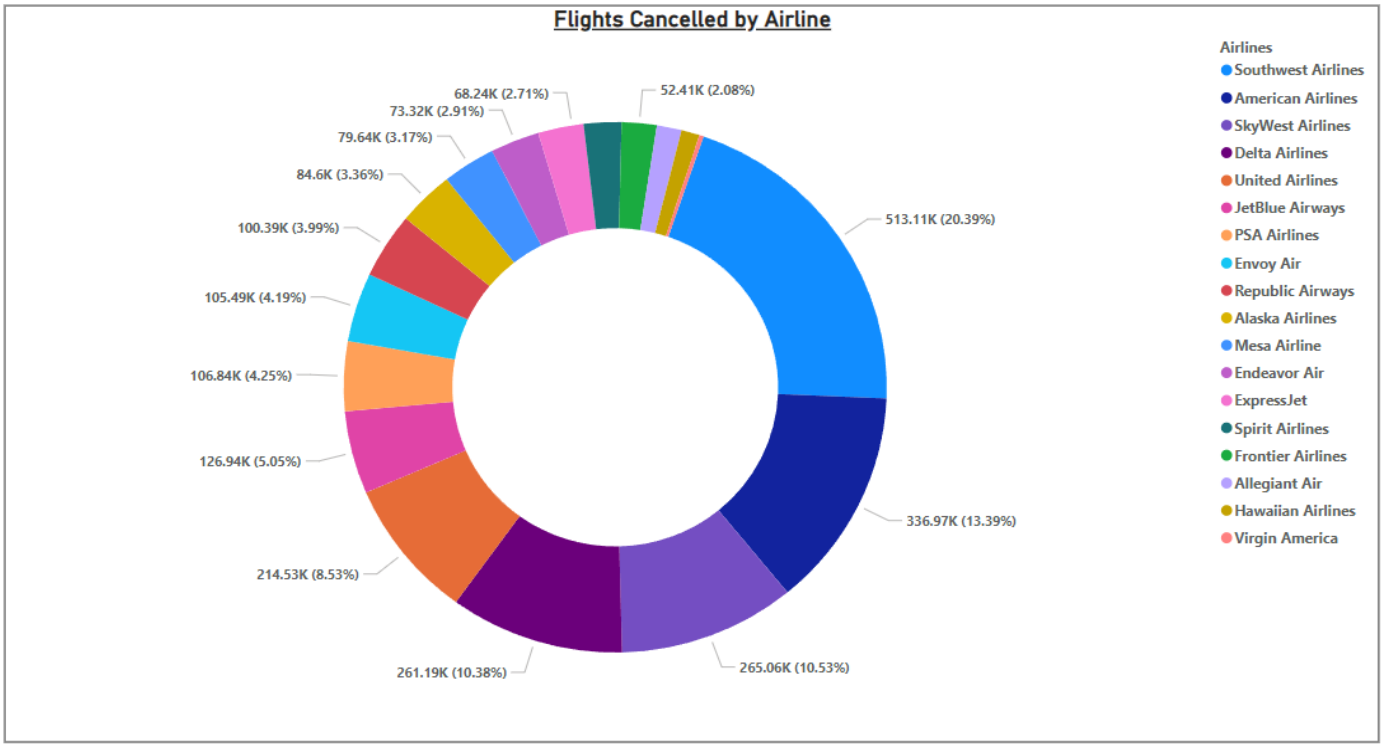
- In July highest departure delay happen
- Southwest Airline have Highest Count of total delay on departure

H. Cancelled Flight

- Count of flight cancelled by month
- Count of flight cancelled by Day
- Count of flight cancelled by airline







Observation:

- We are getting count of flight cancelled by month ,during July and august most amount of flights were cancelled
- We are getting count of flight cancelled by day wise , variations are in middle of month
- Southwest airline has highest % 20.39 for flight cancellation

D. Machine learning:

A number of machine learning algorithms, such as LOGISTIC REGRESSION, DECISION TREE CLASSIFIER, GRADIENT BOOSTING, RANDOM FOREST CLASSIFIER, and SVM, have been tested.

Each model 's F1 Score and accuracy score is calculated :

- **LOGISTIC REGRESSION = 0.7644**
- **DECISION TREE CLASSIFIER = 0.7822**
- **GRADIENT BOOSTING = 0.8215**
- **RANDOM FOREST CLASSIFIER = 0.7809**
- **SVM = 0.5000**

After trying out several algorithms, the most suitable algorithm for our dataset is the Gradient Boosting which is a basic but excellent algorithm for Binary Classification.

1. Split the preprocessed dataset into training and testing datasets

```
train, test = bd9.randomSplit([0.8, 0.2], seed = 123)

print("Train Dataset Count: " + str(train.count()))
print("Test Dataset Count: " + str(test.count()))
```

```
Train Dataset Count: 2014288
Test Dataset Count: 501762
```

2. Import libraries which are necessary for gradient boosting

Gradient Boosting

```
]: from pyspark.ml.classification import GBTClassifier
   from pyspark.ml.evaluation import BinaryClassificationEvaluator
   from pyspark.ml.feature import VectorAssembler
   from pyspark.ml.tuning import CrossValidator, ParamGridBuilder
```

3. Applying gradient boosting model

```
gbt = GBTClassifier()
```

```
param_grid = ParamGridBuilder() \
    .addGrid(gbt.maxDepth, [2, 4, 6]) \
    .addGrid(gbt.maxBins, [20, 30, 40]) \
    .addGrid(gbt.maxIter, [10, 20, 30]) \
    .build()
```

```
evaluator = BinaryClassificationEvaluator()
cv = CrossValidator(estimator=gbt, estimatorParamMaps=param_grid, evaluator=evaluator, numFolds=5)
cv_model = cv.fit(train)
```

```
predictions_gbt = cv_model.transform(test)
```

4. Check accuracy of model

```
In [98]: accuracy = evaluator.evaluate(predictions_gbt)
         print('Accuracy:', accuracy)
```

```
Accuracy: 0.9155762309847115
```

```
In [99]: from pyspark.ml.evaluation import BinaryClassificationEvaluator

         evaluator = BinaryClassificationEvaluator(labelCol="label", rawPredictionCol="prediction")

         f1_score = evaluator.evaluate(predictions_gbt)
         print("F1 score:", f1_score)
```

```
F1 score: 0.8215611660892628
```

6. Conclusion:

Before anything else, we made sure that we comprehended the airline dataset, the work at hand, and the standard by which our contributions will be evaluated. Then, we carried out a rather straightforward EDA to look for connections and patterns that would support our modelling. We took the required preprocessing steps along the way, including indexing categorical variables, imputed missing values, and scaled features to a range. Then, in an effort to improve our model, we built new features using the data that was already available.

After that, we have visualised the data using preprocessed data. Expressjet has the largest average delay, at 51.69, in the industry. (RHI) The average total delay on arrival at Rhinelander-Oneida County Airport is the greatest per minute. The most flights are operating in New York, Florida, London, New Delhi, Paris, Malta, and Hawaii (so they are the busiest routes). The majority of flights were cancelled in July and August, according to data on cancelled flights by month. According to data on cancelled flights by day, changes in cancellation rates occur in the middle of a month. Southwest Airlines has the highest cancellation rate (20.39%). Also, we came to the conclusion that monsoon season, which itself may cause aircraft delays, is when flights are most frequently delayed.

Then, in order to determine if flights will be delayed or not in 2018, we implemented several classification models i.e, Logistic Regression Model, Decision tree classifier, Gradient Boosting, Random forest and SVM model where Gradient boosting is the most appropriate model and has an accuracy rate of 91.55% .