PROJECT REPORT

UNDERSTANDING AND ANALYZING FLIGHT DELAYS

Team Members

Bolla Nithin

Durgavenkata Praveen Reddy Chappidi.

Code Path - https://github.com/bollanithin/Flight-delay

Problem Statement and Background Analysis:

The project aims to analyze and understand delays in airplane on-time performance using flight data sets from bureau of transportation statics for the year 2014.

Reaching on time is a significant customer expectation, and Flight delays hurt airports, airlines as it damages this expectation. It is very tough to explain the reason for a delay as many factors define this issue. Every year approximately 20% of airline flights are cancelled or delayed, costing passengers and airlines a lot of financial and time loss. A few factors responsible for the flight delays are

- Extreme whether
- National aviation system (Delays and cancellations caused by the national aviation system due to a variety of factors such as non-extreme weather, airport operations, excessive traffic volume, and air traffic management)
- Late arrival aircrafts (Reactionary delays due to the late arrival of the previous flight)
- Security (Delays or cancellations caused by a terminal or concourse evacuation, as well as re-boarding of aircraft owing to a security breach)
- Carrier issue (The cancellation or delay was caused by the airline's control, such as maintenance, crew issues, aircraft cleaning, luggage loading, and fueling.

Through this project, we would like to analyze and answer few problems -

- Which airports have the highest delays.
- Which air traffic routes are the most delayed.
- When is the best time of day/day of week/time to fly to minimize delays.

- Which aircrafts carriers have the most delays.
- Do older planes suffer more delays.

We can use the final analysis outcome to find what are major reasons for flight delays in US and what can be the best time to fly with minimum probability of a delay. By the end of the analysis we can come to a conclusion on what are the airports that have the highest amount of delay and which routes have the highest probability of delay.

In this project we plan to use logistic regression to find the accuracy of the analysis that was performed and the outcome of the analysis is reliable.

Data:

We will be working on the dataset that has 400k records which was taken from the Bureau of transportation statistics, the dataset has values of all the commercial flights for the year 2014. Some of the important columns that we extract from the csv file is given below.

- OP_UNIQUE_CARRIER: This is the code assigned to each carrier by IATA and used to identify each carrier.
- ORIGIN - Departure city name.
- DEST - Arrival city name.
- DEP1_Delay - This is the difference of expected and actual time of arrival.
- CRS_DEP_TIME - This is the expected departure time.
- CARRIER_DELAY - This indicates if there is a delay because of carrier in minutes.
- WEATHER_DELAY - This indicates if there is a delay because of whether in minutes.
- NAS_DELAY - This indicates if there is a delay because of National Air System in minutes.
- SECURITY_DELAY - This indicates if there is a delay because of security in minutes.
- DEPDELAYED - This the used to identify whether there is any delay. This is a Boolean value.
- LATE_AIRCRAFT_DELAY - This is used to indicate total delay in minutes.

Methods:

In the data organization part, we used pyspark to create data frames in the required form by adding few columns to the data as required. The dataset on which the analysis was performed was already

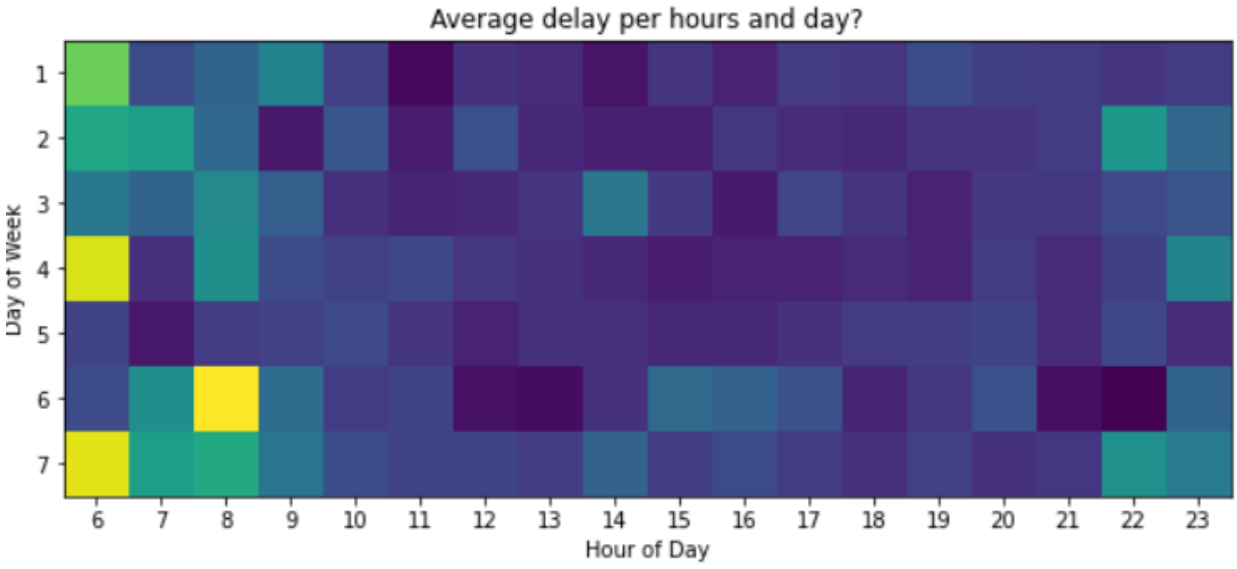
clean so we dint have to perform any cleaning on the dataset. In the data organization part,we have added new columns which stores delay of a flight which will be used in the querying part.

In the querying part we have used spark sql for reading the data. Spark sql provides a wide range of inbuilt methods, which we have used to extract the data from the data frame created in the data organization part.

We have used the logistic regression as the modeling algorithm to determine the accuracy of the analysis.

A prominent approach for predicting a categorical answer is logistic regression. It is a subset of Generalized Linear models that forecasts the likelihood of outcomes. Logistic regression in spark.ml may be used to predict a binary result using binomial logistic regression or a multiclass outcome using multinomial logistic regression.

Parameter choices has a prominent part as this can have a great impact on the performance part. We used several parameters in our project where if the data with large size and with great delays are passed to the system then it might take a longer time for execution and average delay will be increased significantly.



We have noticed a significant increase in execution time when a table has been created with pandas than when compared to pyspark. The use of data Pyspark sql functions have improved the execution time significantly.

We have tested our analysis with a base line model where we have taken a dataset with few columns and tested our analysis and the results were as expected but the system was not able to predict the delays properly as the data set did not consist large number of records. In the baseline model we got minimal values for the average, sum values and the graphs which shows the delays didn't show any major airports which had huge delays as there are few records.

We have utilized the jupyter notebook and configured Pyspark in jupyter for effective usage of methods. Some of the library's and methods we have used are pyspark, findspark, matplotlib, pyarrow and pandas. We have used the pyspark and findspark library's to use different method like findspark.init(), SparkConf(), SQLContext(), SparkSession(), LabeledPoint(), udf(), LogisticRegressionWithLBFGS(), LabeledPoint(), createDataFrame(). We use the init() and SparkConf() to initialize the spark directory and configure the spark directory, SQLContext() is used to set the properties in spark, to create data frame the dataset we use is SparkSession() , we use LabeldPoint() and UDF() for regression analysis. We use the matplotlib library to use the methods for plotting the map that shows the route that has the highest delays. Pyarrow library is used for integration with panda's environment.

We have used the methods findspark.init(), SparkConf(), SparkContext(), SparkSession(), withColumn(), createDataFrame(),textFile(), map(), SQLContext(), registerTempTable for data organization using pyspark and sql. The data set we have selected did not have any unwanted data so we did not perform any data cleaning on the dataset. We have queried the data frames created for querying using pandas, spark sql and Basemap methods. The methods we used for modeling are StringIndexer(), transform(), randomSplit, map().

Results:

The Final results that are expected is a prediction model that can show what are the main reasons for a flight getting delayed and what the probabilities that a flight can be delayed based on the

given input parameter like origin, destination, flight departure and arrival time, reasons for the delay with minutes if there is a delay.

We have tested our study with logistic regression which gave us a model accuracy of 83.7 , which was good and as expected so we did not use any other regression models apart from logistic regression

```
TP=TN=FP=FN=0.0
for x in acc:
    if x[0]=='TP': TP= x[1]
    if x[0]=='TN': TN= x[1]
    if x[0]=='FP': FP= x[1]
    if x[0]=='FN': FN= x[1]
eps = sys.float_info.epsilon
Accuracy = (TP+TN) / (TP + TN+ FP+FN+eps)
print("Model Accuracy for SJC:")
print(float(Accuracy*100))
```

```
Model Accuracy for SJC:
83.65019011406845
```

Primary cause for flight delays: Upon analyzing the dataset, we came to a prediction that the primary cause for flight delays is because of late arrival of a previous flight and security delays has the least possibility that a flight will be delayed. We have used the spark SQL methods to extract the information about the main reason for flight delay on a particular day.

```
In [9]: cause_delay = sqlContext.sql("SELECT sum(WeatherDelay) Weather,sum(NASDelay) NAS,sum(SecurityDelay) Security,sum(Late
                                    FROM airlineDF ")

In [10]: df_cause_delay = cause_delay.toPandas()
         df_cause_delay.head()
```

Out[10]:

	Weather	NAS	Security	lateAircraft	Carrier
0	578051	1476657	5353	3243417	2384451

Airports with most delays: From the data set we tried analyzing the data to find the airport which has the highest delays so that any passenger who travels through the airport can expect a delay. The SQL functions in spark and pandas are used to find which airport has the highest delay by grouping the data from same airport.

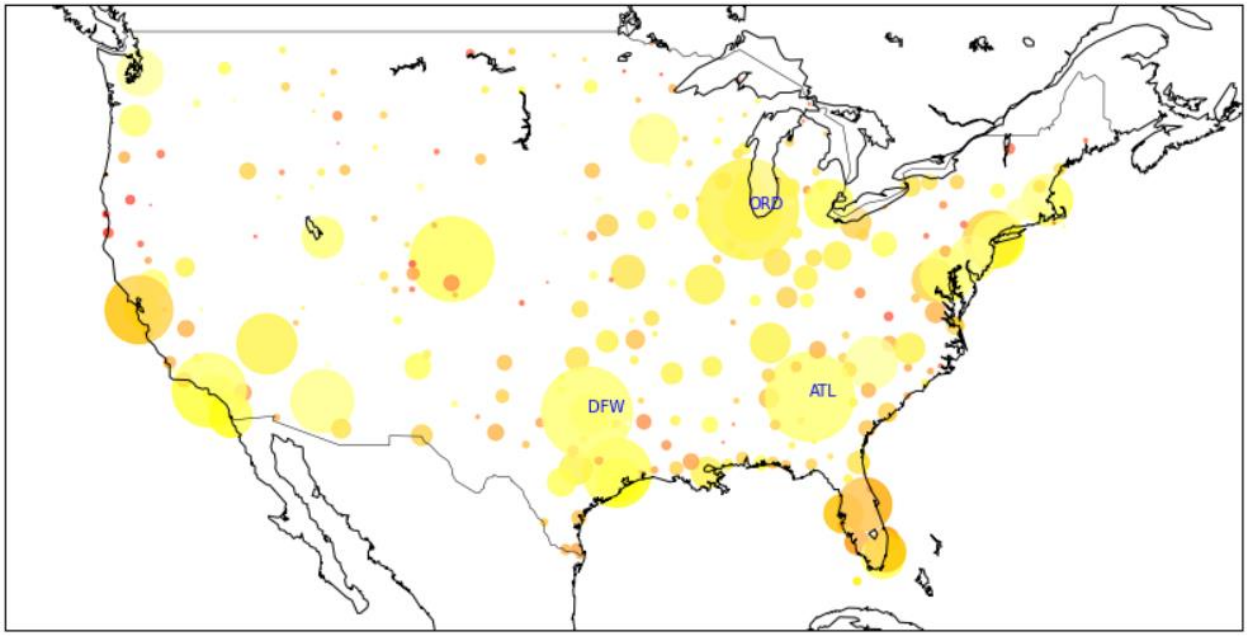
```
In [11]: groupedDelay = sqlContext.sql("SELECT Origin, count(*) conFlight,avg(DepDelay) delay \
                                        FROM airlineDF \
                                        GROUP BY Origin")

         df_origin = groupedDelay.toPandas()
         df_origin.sort_values('delay',ascending=0).head()
```

Out[11]:

	Origin	conFlight	delay
273	BGR	9	161.000000
42	CDV	6	125.166667
282	YAK	8	125.125000
271	SIT	13	110.615385
253	ALO	17	105.823529

Plotting map for the airports with their respective delays: We have used the methods in basemap and matplotlib library's to get the desired output. Where in the below graph each marker is an airport and size of traffic is represented by the size of the marker, larger the marker indicates there is more traffic in the airport. The color of the marker is represented by the expected delays, When the marker is red it means there is longer delays.



Routes with highest delays: We use the spark SQL and pandas to find the routes which have the highest delays. We use pandas on the data after grouping the datasets which gives us the routes

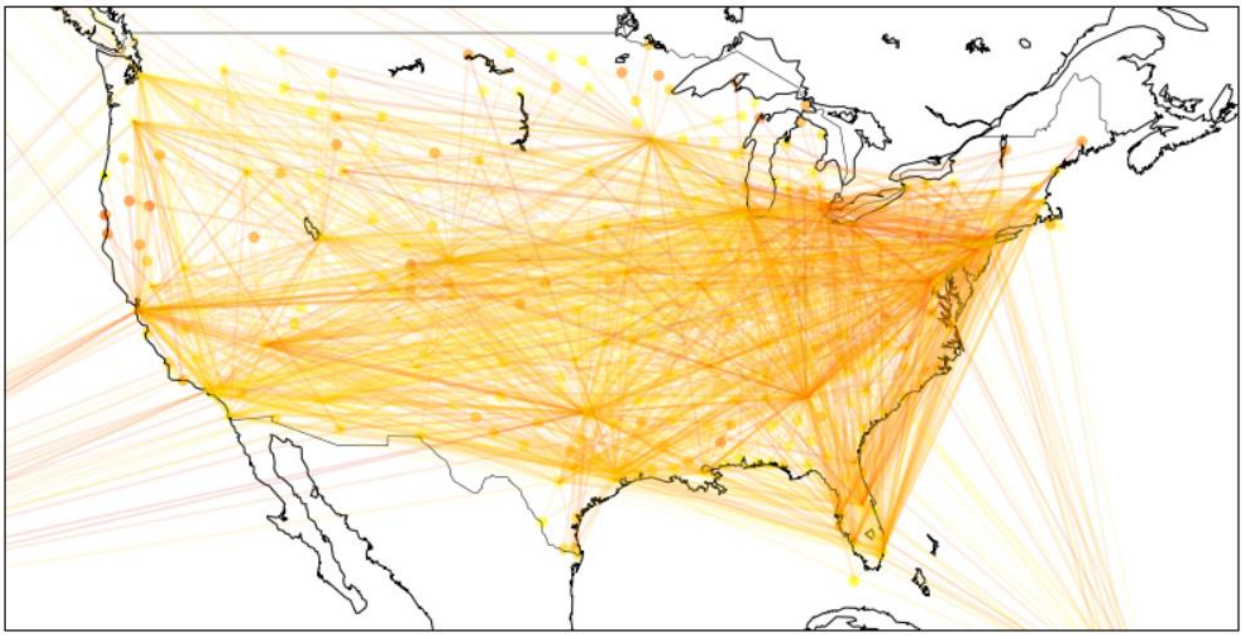
with the highest delays. From the below output we can see that BNA and PIT have the highest delays when compared to all other airports.

```
grp_rout_Delay = sqlContext.sql("SELECT Origin, Dest, count(*) traffic,avg(Distance) avgDist,\
                                 avg(DepDelay) avgDelay\
                                FROM airlineDF \
                                GROUP BY Origin,Dest")
rout_Delay = grp_rout_Delay.toPandas()
```

```
df_airport_rout1 = pd.merge(rout_Delay, df, left_on = 'Origin', right_on = 'IATA')
df_airport_rout2 = pd.merge(df_airport_rout1, df, left_on = 'Dest', right_on = 'IATA')
df_airport_rout = df_airport_rout2[["Origin","lat_x","lng_x","Dest","lat_y","lng_y",\
                                    "avgDelay", "traffic"]]
df_airport_rout.sort_values('avgDelay',ascending=0).head()
```

	Origin	lat_x	lng_x	Dest	lat_y	lng_y	avgDelay	traffic
2679	BNA	36.124500	-86.678200	BHM	33.562901	-86.753502	402.0	1
2242	PIT	40.491501	-80.232903	PBI	26.683201	-80.095596	330.0	1
3776	JFK	40.639801	-73.778900	JAC	43.607300	-110.737999	322.0	1
3983	SYR	43.111198	-76.106300	BTV	44.471901	-73.153297	257.0	1
1936	GTF	47.481998	-111.371002	MSP	44.882000	-93.221802	256.5	2

Plotting routes with highest delays: We will be using the basemap and matplotlib library's to plot the map which shows the highest delayed routes. Each line in the below lot represents a route which was delayed. The redder the line the higher the probability of a flight getting delayed.



Average delay time for each carrier in a month : As used before we use spark SQL and panda to extract the data and we use matplotlib to plot the findings in a graph. Based on the given dataset airline aa has the highest amount of delay average in month.

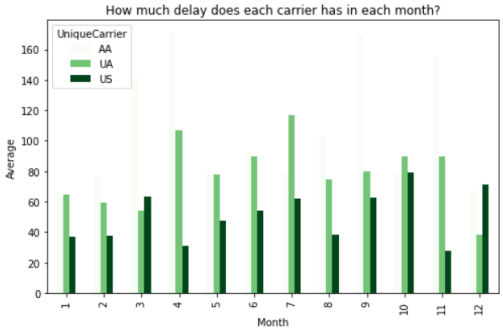
```
print("total flights from this ariport: " + str(df_ORG.count()))
```

```
total flights from this ariport: 7830
```

```
grp_carr = sqlContext.sql("SELECT  UniqueCarrier,month, avg(DepDelay) avgDelay from df_ORG \
                           WHERE DepDelayed=True \
                           GROUP BY UniqueCarrier,month")
s = grp_carr.toPandas()
```

```
ps = s.pivot(index='month', columns='UniqueCarrier', values='avgDelay')[['AA','UA','US']]
```

```
rcParams['figure.figsize'] = (8,5)
ps.plot(kind='bar', colormap='Greens');
plt.xlabel('Month')
plt.ylabel('Average')
plt.title('How much delay does each carrier has in each month?')
```



Logistic regression: We have the used the logistic regression method to find the accuracy of the analysis and the findings show that we have 83.7% of accuracy which is considered to a very good accuracy. Since we achieved an accuracy of 83.7% we did not use any other regression method to find better accuracy. Below is a part of the screenshot which shows that the model accuracy of the analysis.

```
TP=TN=FP=FN=0.0
for x in acc:
    if x[0]=='TP': TP= x[1]
    if x[0]=='TN': TN= x[1]
    if x[0]=='FP': FP= x[1]
    if x[0]=='FN': FN= x[1]
eps = sys.float_info.epsilon
Accuracy = (TP+TN) / (TP + TN+ FP+FN+eps)
print("Model Accuracy for SJC:")
print(float(Accuracy*100))
```

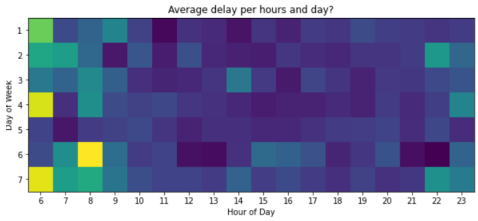
```
Model Accuracy for SJC:
83.65019011406845
```

We have analyzed the derived outcome with the baseline by considering few records in the data set. We can see that the outcome of the data from the baseline model is as expected but, since the number of records used for the evaluation is less the prediction does not show the accurate time when a flight can be delayed. The plots which show the airports and routes with high delays does not have any red or highlighted routes which show that are high delays. This is because there is not enough data for the system to predict that there has been high delays in the region.

The setback that we faced is that, we planned to predict whether the age of an aircraft is responsible for the delay. We have tried to find datasets that have the flight age data along with the delay information from different sites and data sets, But since we were not able to find the dataset which has this data we had to leave this prediction out.

Lessons Learned:

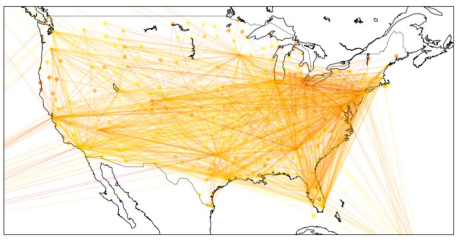
From this project we were able to predict, what day and time of the week has the highest probability of a flight getting delayed. After the prediction we were able to come to an ending that there is more probability of a flight getting delayed on morning hours of Monday, Saturday and Sunday.



By analyzing the data, we found that the major reasons for an airline delay is weather, NAS, security, late aircraft and carrier delay. Out of which Late aircraft delays stands in the top position followed by weather and the least chances that an aircraft getting delayed goes to security.

By querying the datasets, we were also able to come to a conclusion on the list of airports that has the highest probability of an airline getting delayed. By this prediction the passengers who are travelling through that airports can expect a delay. We have also designed a plot which shows what are the airports that have heavy traffic and heavy delays.

Using the same datasets, we have also predicted on what are the routes that can have heavy delays so that the passengers flying in those routes can expect delay in their flight. We have designed the plot by extracting the data from dataset using pyspark SQL and used basemap and matplotlib libraries to design the plot. Below is the plot which represents the routes which have higher delays.



So finally from this project we have got the knowledge regarding the real time issue which is delay of flights as we are predicting that why the flight is getting delayed we can easily know the possibilities of choosing other option or not getting tensed why it is getting delayed. So now we have some solution or an answer for this challenge we are facing frequently.

Big Data Systems and Tools

Spark – This technology was used to handle large amounts of data in a short amount of time. It is used to partition data in blocks which can be used for easy data processing.

Jupyter notebook – This is used to deploy bigdata and machine learning algorithms in an easy and efficient manner.

Python – We used python as we are handling large amounts of data and python efficiently handles large amount of data.

Pandas – Pandas is a data analysis toolkit which is easy to use and has high performance when used with data structures.

Excell – This is used to store the raw data and convert it into more easy forms so that useful and efficient information can be extracted.

Virtual Box – We have used virtual box to run the VM, we have tried working with jupyter notebook in local but because we kept receiving spark directory issue, we used jupyter notebook in VM which resolved the issue.

Team Contributions:

We are team consisting of two named Durgavenkata Praveen Reddy Chappidi and Nithin Bolla. From the starting of the project we had a good idea about what we have to implement or what should this project exactly need to give a certain solution for the issues or challenges faced in delay of flights. So we divided the work equally most of the data organization, querying part was handled by nithin bolla and the data modeling and regression part was handled by Durgavenkata Praveen Reddy Chappidi.