Right on Time

Smart Air Travels

By: Le Gu, Jacky Zhang

Initiatives

According to a study conducted by UC Berkeley researchers, flight delays put a $32.9 billion hole in the U.S. economy in 2007, and more than half of that cost is borne by us, the passengers.

This project aims to better understand air travel experiences by analyzing flight on-time performance statistics. By exploring flight delays, the project hope to reveal delay trends by airport, airline, aircraft models and reasons for delay etc.

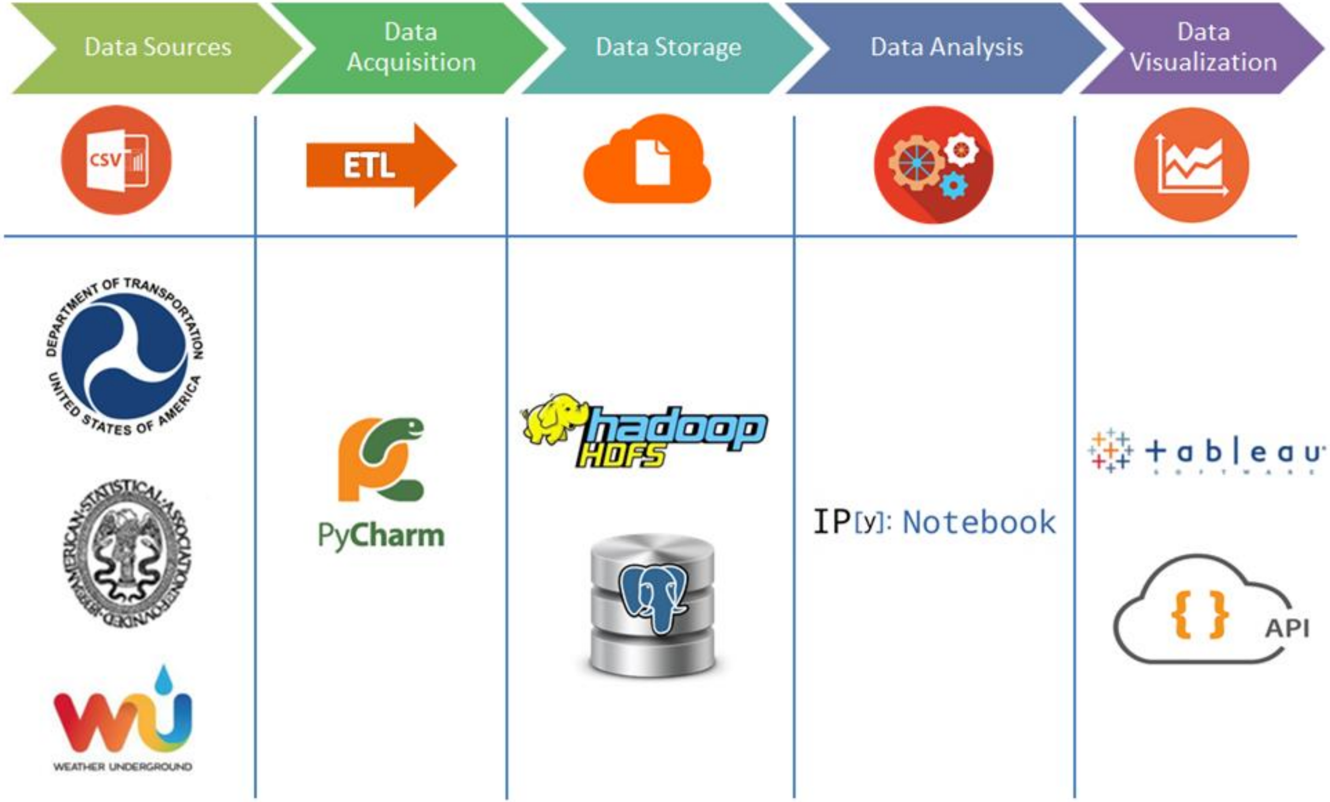
Data Sources

The United States Department of Transportation tracks on-Time performances of major US carriers. The data set contains flights departure, arrival details for US domestic flights since 1987. This project would take the most recent 2 years’ data (6 GB) for analysis.

Statistical Computing Organization provides aircraft-related information such as manufacturer, issue year, model type etc and airport geographical information.

www.wunderground.com provides historical airport weather data.

Data Architecture

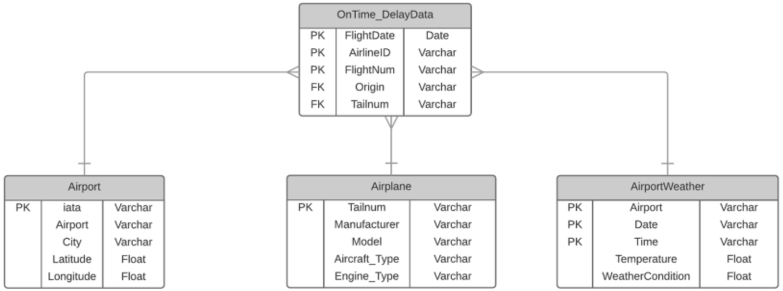


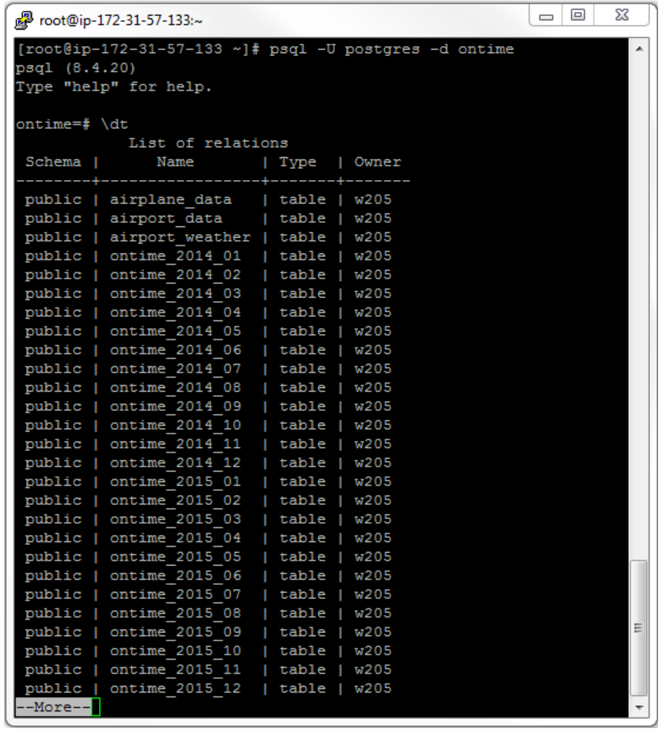
PostgreSQL

Structural and tabular data

Data retrieval optimization:

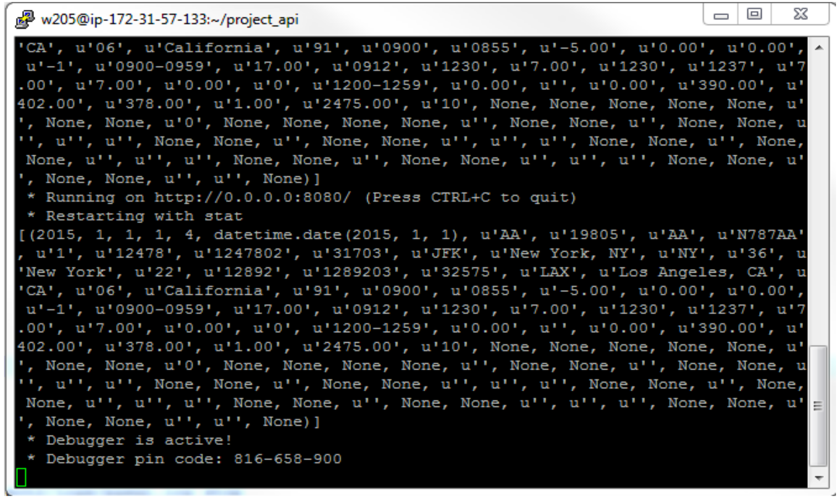
- Primary & Foreign Keys, Indexes
- Table partitionings



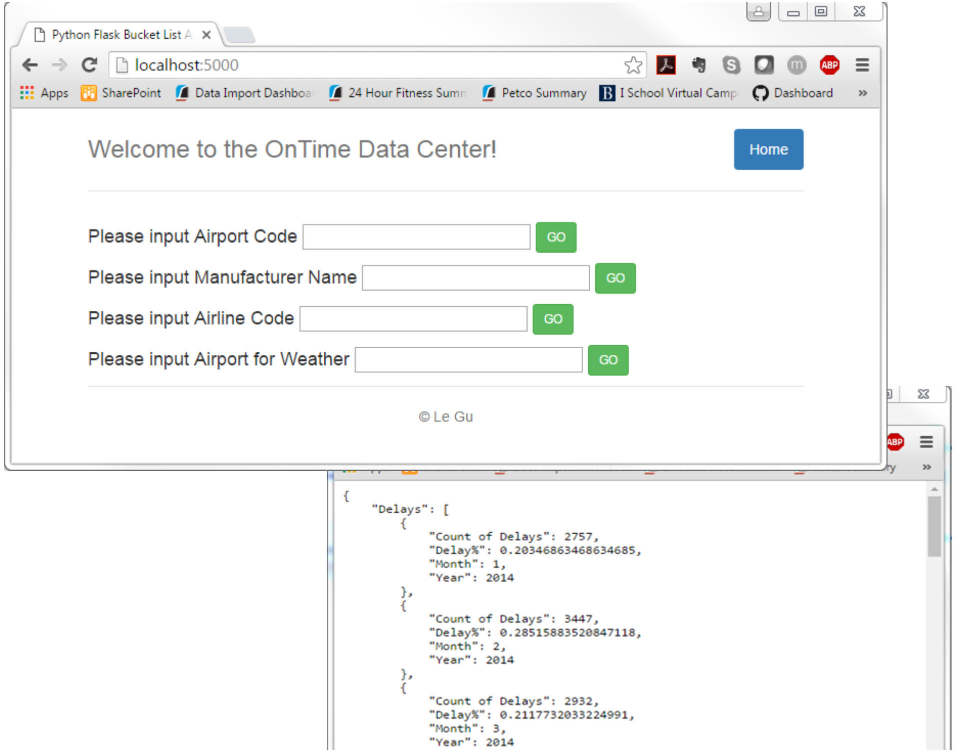


REST API & Flask Web Framework

- Serve Data from PostgreSQL Database through REST API using Python.

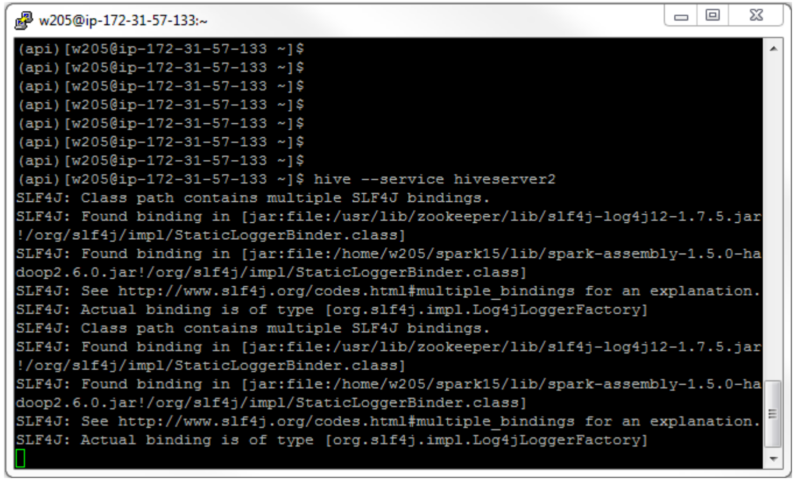


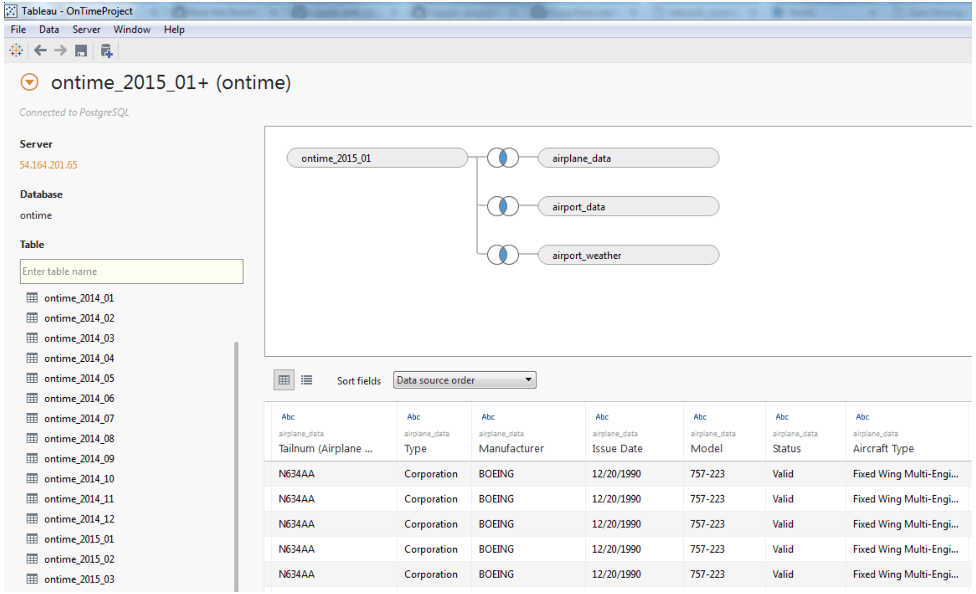
- Web interface based on Python Flask



Tableau

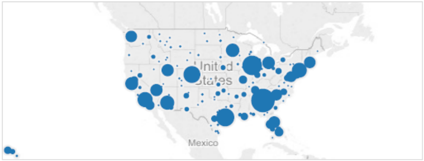
- Access remote PostgreSQL database from Hive Thrift Server
- Join different data sources in Tableau for analysis and visualization



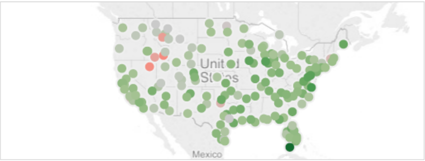


Tableau

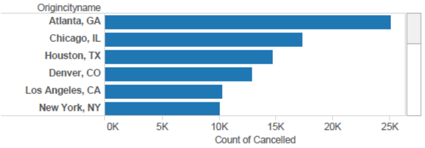
Total Flight by Airport



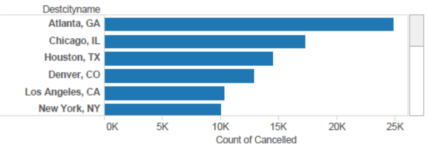
Avg Delay by Airport (minute)



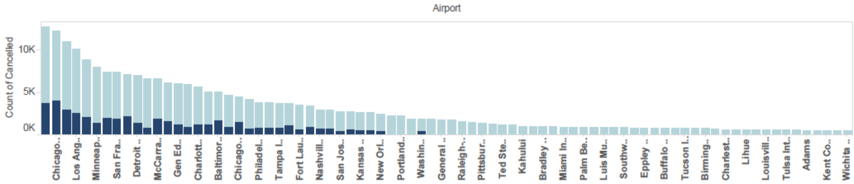
Top Origin City



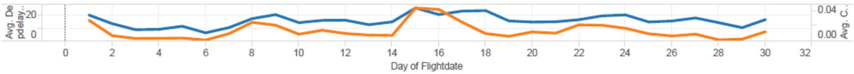
Top Destination City



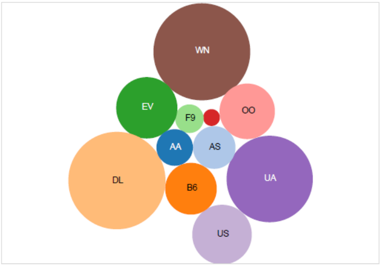
Total Airport Delay vs Total Flights



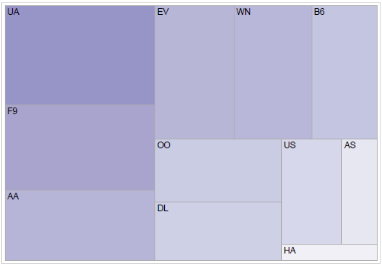
Delay and Cancellation in a Month (percentage)



Carrier Total Distance



Carrier Avg Delays (minute)



Manufacturer, Plan Model vd Delays (minute)

	Type				
Manufacturer	Co-Owner	Corporation	Foreign Corp..	Individual	Partnership
AIRBUS		15.4	12.5		
AIRBUS INDUS..		15.0	14.9		
AERO COMMA..		77.6		25.9	
AEROSPATIALE		22.3			
AVIAT AIRCRA..		25.5			
BEECH		12.9			
BOEING		16.8			
BOMBARDIER ..		14.8			
CANADAIR		12.9			
CESSNA	22.6	20.6		19.3	
DEHAVILLAND				10.2	
DOUGLAS		130.0			
EMBRAER		18.0			
GULFSTREAM ..		25.5			
HELIO		23.5			
MCDONNELL ..		16.3		18.0	
MCDONNELL ..		14.6			
MCDONNELL ..		15.6			
PAIR MIKE E				14.4	
PIPER	59.1	10.9			22.6
SIKORSKY		46.5			
STINNETT RON				13.4	

Manufucturer, Engine vs Delays (percentage)

	Engine Type					
Manufacturer	4 Cycle	Reciproca..	Turbo-Fan	Turbo-Jet	Turbo-Pro..	Turbo-Sh..
SIKORSKY						0.7500
AEROSPATIALE						0.3333
AERO COMMANDER				0.4390	0.5200	
AIRBUS			0.2098	0.2323		
AIRBUS INDUSTRIE			0.1601	0.2294		
AVIAT AIRCRAFT INC		0.3529				
BEECH					0.1818	
BOEING			0.2785	0.1995		
BOMBARDIER INC			0.2044			
CANADAIR				0.1848		
CESSNA	0.2667	0.2584				
DEHAVILLAND		0.1385				
DOUGLAS		0.5556				
EMBRAER			0.2383			
GULFSTREAM AER..			0.3077			
HELIO					0.2273	
MCDONNELL DOUG..			0.2279	0.2177		
MCDONNELL DOUG..			0.1941			
MCDONNELL DOUG..				0.2034		
PAIR MIKE E		0.2222				
PIPER		0.2555				
STINNETT RON		0.1838				

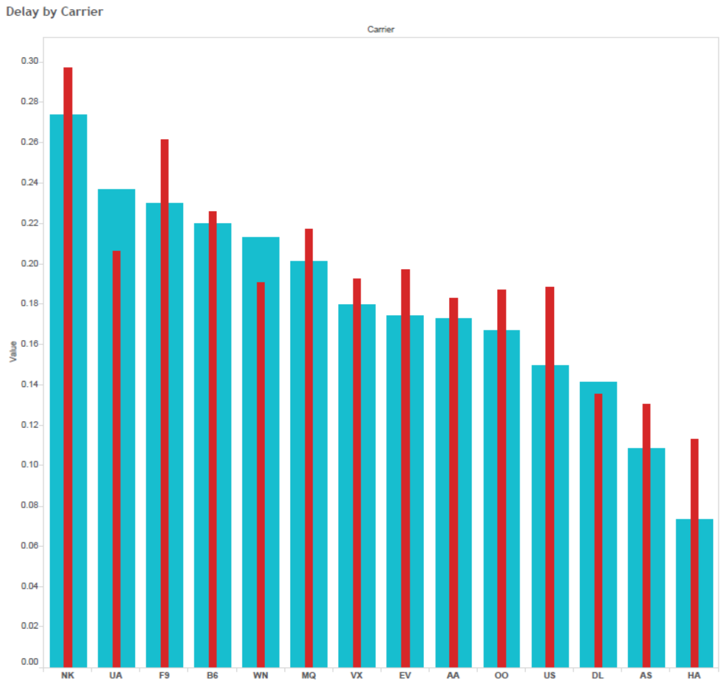
Data Analysis

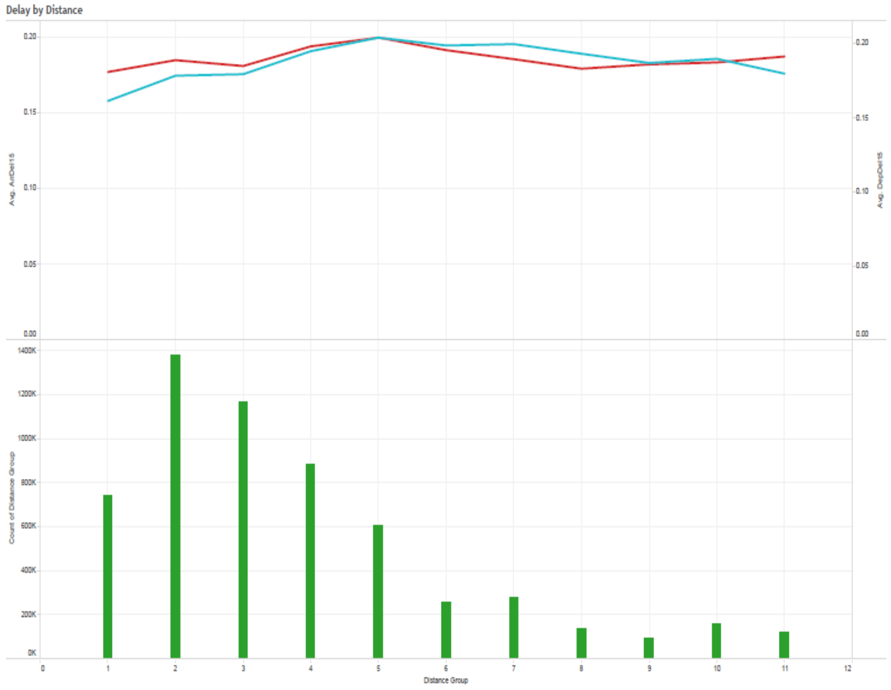
1. Which airlines have the most delays?
2. Do longer flights have more delays?
3. When is the worst time to travel, in terms of delays expected?
4. Which are the worst months the travel, in terms of delays expected?

Predictive Model:

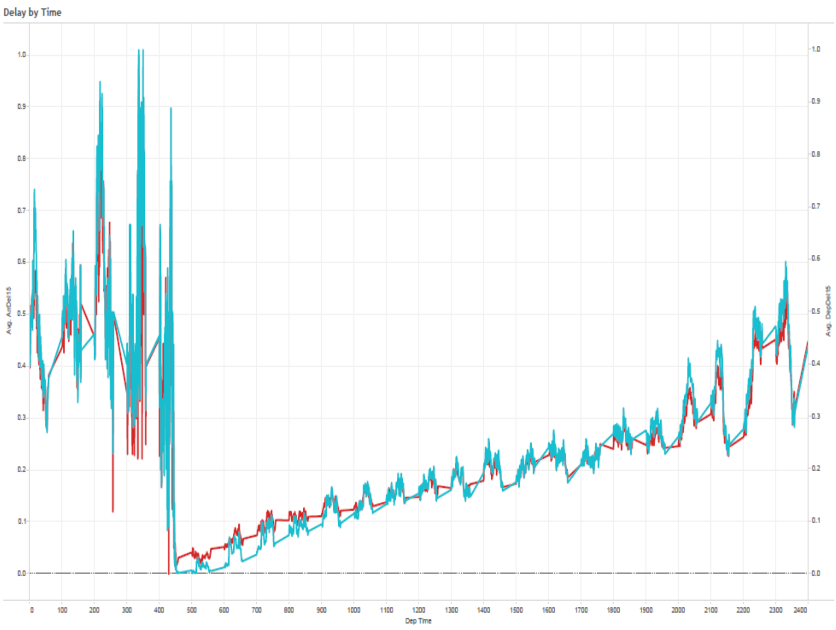
We seek to build a prediction model that gives the users an estimate of expected delay given information about their flight.

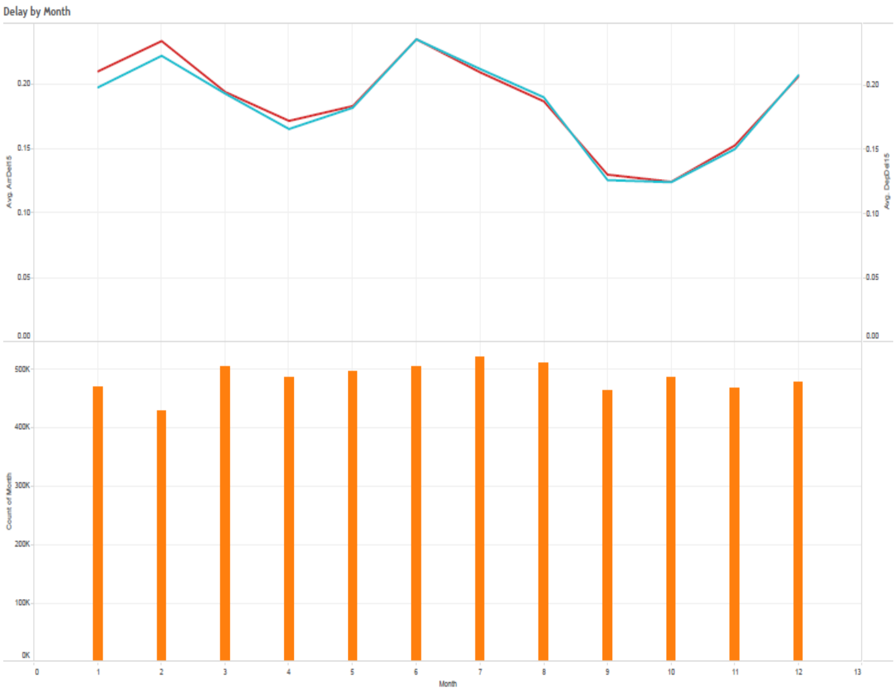
Exploratory Analysis - Tableau





Exploratory Analysis - Tableau





Prediction Model

```
In [3]: conn = psycopg2.connect(database="ontime", user="postgres",  host="localhost", port="5432")

In [4]: delay = pd.read_sql_query("SELECT year, quarter, month, dayofmonth, dayofweek, flightdate, deptime, carrier, tailnum, \n
                                  origin, dest, distance, distancegroup, depdelayminutes, depdel15, arrdelayminutes, \n
                                  arrdel15 FROM ontime_data LIMIT 2000000", conn)
```

Used IPython notebook to connect to Postgres database
Used features identified in exploratory analysis
Random forest classifier

- Data size: 2 million records
- Input variables: 10; Year, Month, DayOfMonth, DayOfWeek, DepTime, Dest, Origin, Carrier, DistancegGroup, Holiday
- Output variable: DepDel15 (Binary)
- Accuracy to beat: ~80%
- Accuracy: 83.5%
- F1-Score: 30.7

	Feature Importance
DepTime	0.373418
Origin	0.147665
Dest	0.139933
DayOfMonth	0.092986
Carrier	0.073735
DistanceGroup	0.072669
Month	0.052429
DayOfWeek	0.044109
Holiday	0.003054
Year	0.000000

```
In [40]: predict = np.asarray([2015, 3, 4, 3, 1220, 217, 83, 10, 11, 0])

         print ("Probability of delay:" , rf.predict_proba(predict.reshape(1,10))[0][1])

         ('Probability of delay:', 0.72004257916726822)
```

Next Steps

Solution Scalability:

- In order to serve concurrent users, we will need to build EC2 clusters that allow the parallel processing and offer higher IOPS.
- As data get accumulated in the Postgres database, the query optimizer would be able to leverage the table partitioning and the performance should not be greatly affected.

Solution Evolution:

- Twitter data sentiment analysis: introduce real-time stream processing with Twitter developer API, apply sentiment analysis to track public opinions on flight delays.
- Data Pipeline Scheduler: incorporate AWS data pipeline API to schedule runs in a automated fashion, such as real-time twitter data collection, nightly weather data web scraping, and monthly data feeds acquisition and model retrain.