Goel, Ajay: Nu Id (001897443)

INFO 7250
Engineering Big-Data Systems
Summer Full 2019
Project Report

Big Data Analysis
(Airline Dataset 1987 - 2008)

- Map Reduce (Hadoop)
- PIG
- HIVE
- Python (Sentimental Analysis)
- AWS EMR
- Tableau
- Neo4j

- By
Ajay Goel
(001897443)

Table of Contents

Goel, Ajay: Nu Id (001897443)

Introduction

1. About Dataset:

The Dataset of the flight records in USA is available on http://stat-computing.org/dataexpo/2009/the-data.html. The data is available from year 1987 to 2008. It has many columns which can be helpful in Map Reduction analysis. Once it is done, more analysis on the basis of aggregation or various important points can be done using Pig, Hive, HBase.

Variable descriptions		
Col. No.	**Name**	**Description**
1	Year	1987-2008
2	Month	12-Jan
3	DayofMonth	31-Jan
4	DayOfWeek	1 (Monday) - 7 (Sunday)
5	DepTime	actual departure time (local, hhmm)
6	CRSDepTime	scheduled departure time (local, hhmm)
7	ArrTime	actual arrival time (local, hhmm)
8	CRSArrTime	scheduled arrival time (local, hhmm)
9	UniqueCarrier	unique carrier code
10	FlightNum	flight number
11	TailNum	plane tail number
12	ActualElapsedTime	in minutes
13	CRSElapsedTime	in minutes
14	AirTime	in minutes
15	ArrDelay	arrival delay, in minutes
16	DepDelay	departure delay, in minutes
17	Origin	origin IATA airport code
18	Dest	destination IATA airport code
19	Distance	in miles
20	TaxiIn	taxi in time, in minutes
21	TaxiOut	taxi out time in minutes
22	Cancelled	was the flight cancelled?

23	CancellationCode	Reason for cancellation (A = carrier, B = weather, C = NAS, D = security)
24	Diverted	1 = yes, 0 = no
25	CarrierDelay	in minutes
26	WeatherDelay	in minutes
27	NASDelay	in minutes
28	SecurityDelay	in minutes
29	LateAircraftDelay	in minutes

1.1 Tools Used:

- Map-Reduce Using HDFS
- PIG
- HIVE
- Python (Sentimental Analysis)
- AWS EMR
- Tableau
- Neo4j

1.1.1 LOAD DATA:

- Shell script to run and load data is present in shell.sh
- Data is then loaded to HDFS

```
--------------------------------------------------------------------------------------------------------
echo Downloding file
max = 2008
for (( i=1987; i <= 2008; ++i ))
do
    curl -o /Users/ajaygoel/Documents/Neu/BigData/Project/Data/$i.csv.bz2 "http://stat-computing.org/da
done

echo "Unzipping the file"
for (( i=1987; i <= 2008; ++i ))
do
    bzip2 -d /Users/ajaygoel/Documents/Neu/BigData/Project/Data/$i.csv.bz2
done
hadoop fs -copyFromLocal /Users/ajaygoel/Documents/Neu/BigData/Project/Data /project/data
--------------------------------------------------------------------------------------------------------

fi
echo "Unzipping the file"
bzip2 -d /Users/ajaygoel/Documents/Neu/BigData/Project/Data/1987.csv.bz2
if [ "$?" -gt 0 ]
then
echo "Error unzipping file"
exit
```

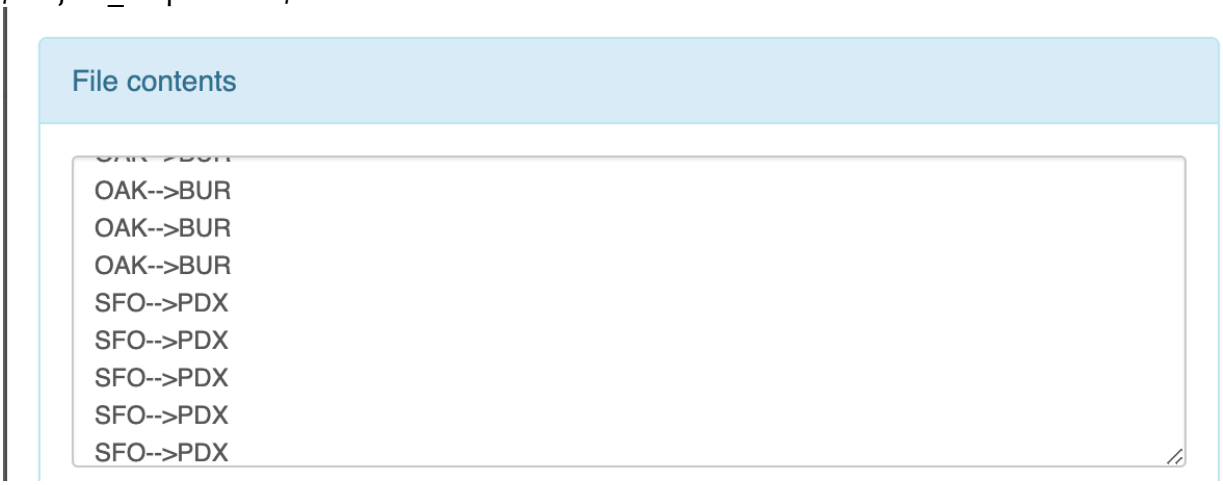
Analysis Performed:

2. Map – Reduction using Hadoop:

2.1 Bloom Filter: Source and Destination Test (Filter Pattern Algorithms)

- Chosen one source and destination, based on the probability the Sources and Destination will be filtered. False positives will be there in the o/p as probability taken is 0.3

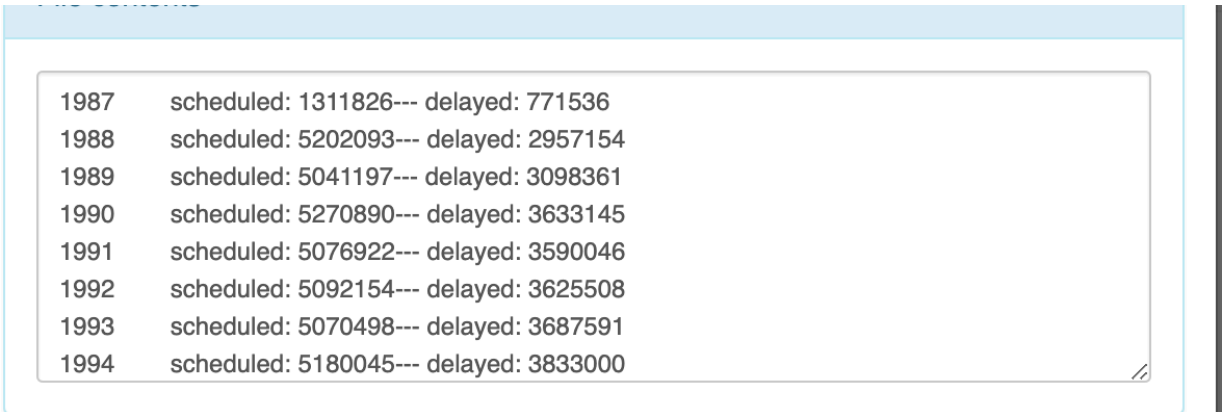
hadoop jar /Users/ajaygoel/Documents/Neu/BigData/Project/Final_Project/target/Project-1.0-SNAPSHOT.jar BloomFilterSourceDestTest.Driver /project/data/1987.csv /Project_mapReduce/BloomFilterSourceDestTest



Here, OAK → BUR is coming as false positive and SFO →PDX is the expected data.

2.2 Delay and Scheduled flights based on Year

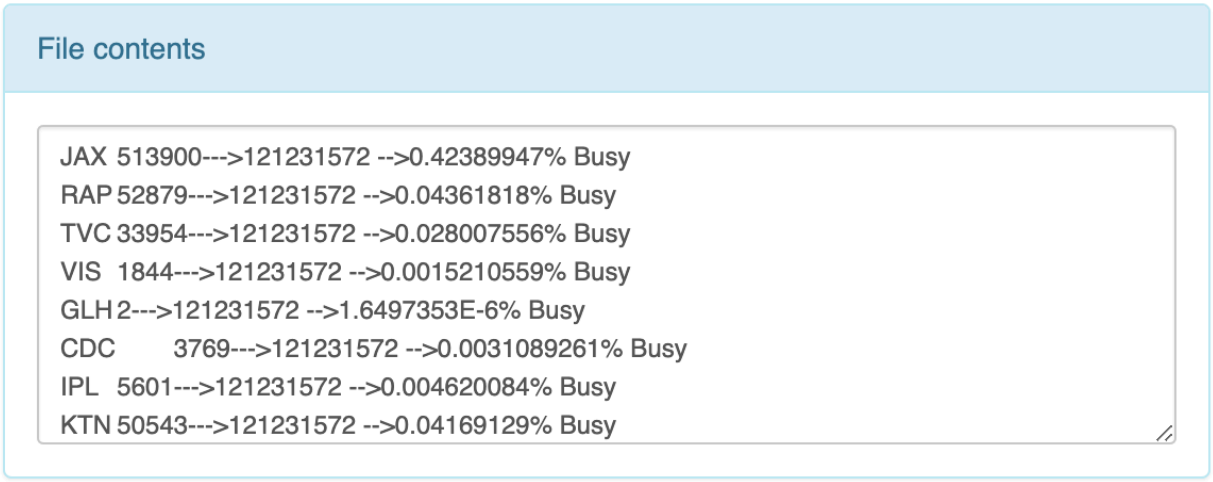
hadoop jar /Users/ajaygoel/Documents/Neu/BigData/Project/Final_Project/target/Project-1.0-SNAPSHOT.jar YrDelayAndScheduledFlights.DelayDriver /project/data /Project_mapReduce/YrDelayAndScheduledFlights



In a particular year, how many scheduled and delayed flights are there.

2.3 Busiest Airport in10 years & percentage using (Recommendation System):

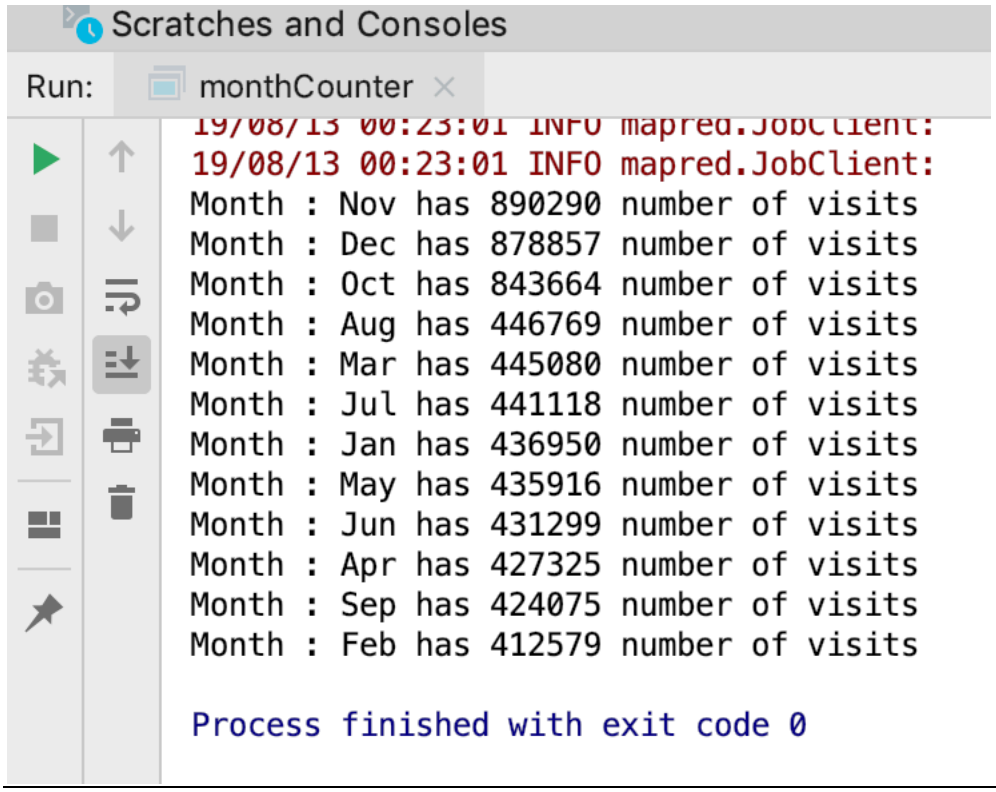
hadoop jar /Users/ajaygoel/.m2/repository/Final_Project/Project/1.0-SNAPSHOT/Project-1.0-SNAPSHOT.jar busyAirport.Driver /project/data /Project_mapReduce/busyAirport



2.4 Number of flights visits each month (Counter Algorithm)

This will return the number of flights that were not cancelled in a particular year, sorted by count decreasingly.

hadoop jar /Users/ajaygoel/Documents/Neu/BigData/Project/Final_Project/target/Project-1.0-SNAPSHOT.jar monthCounter.monthDriver /project/data /Project_mapReduce/monthCounter



2.5 From each source, destination is indexed (Inverted Index Algorithm)

It will show all the destinations which have a direct flight from a source using Inverted index.

hadoop jar /Users/ajaygoel/Documents/Neu/BigData/Project/Final_Project/target/Project-1.0-SNAPSHOT.jar srcDstInvertedIndex.srcDstDriver /project/data/1987.csv /Project_mapReduce/srcDstInvertedIndex

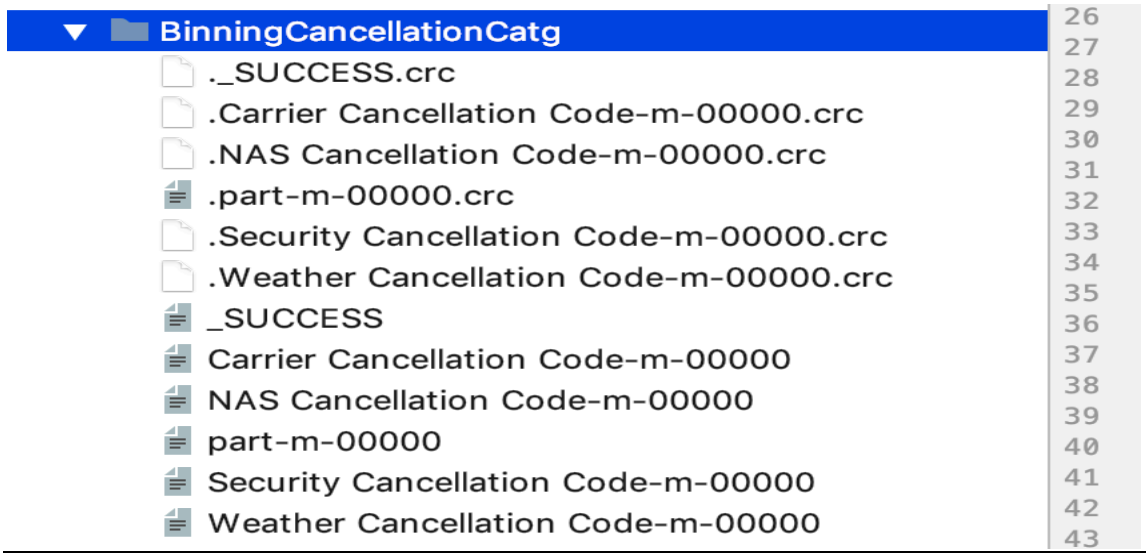
File contents

```
ABE --->ORD ,PHL ,DTW ,ATL ,PIT ,AVP ,MDT
ABQ      --->ORD ,SAN ,LAX ,DTW ,SAT ,ELP ,STL ,SLC ,LBB ,DFW ,PHL ,AMA ,MDT
,SFO ,PHX ,TUL ,DEN ,DAL ,IAH ,MCI ,MAF ,TUS ,ATL ,PIT ,AVP ,LAS
ACV --->ORD ,SAN ,LAX ,DTW ,SAT ,ELP ,STL ,SLC ,LBB ,RDD ,DFW ,PHL ,AMA
,MDT ,SFO ,PHX ,TUL ,DEN ,DAL ,IAH ,MCI ,MAF ,TUS ,ATL ,PIT ,AVP ,LAS
AGS      --->ORD ,SAN ,LAX ,DTW ,SAT ,ELP ,STL ,CLT ,SLC ,LBB ,RDD ,DFW ,PHL
,AMA ,MDT ,SFO ,PHX ,CAE ,TUL ,DEN ,DAL ,IAH ,MCI ,MAF ,TUS ,ATL ,PIT ,AVP
,LAS
```

2.6 Binning Algorithm: Creating bins on the bases of codes (Data Organisation Algorithm)

Creating a Cancellation log based on the types of cancellation
Creating different bins based on the types of cancellation. Here, 4 types are mentioned as below:

hadoop jar /Users/ajaygoel/.m2/repository/Final_Project/Project/1.0-SNAPSHOT/Project-1.0-SNAPSHOT.jar BinningCancellationCategories.Binning /project/data/1987.csv /Project_mapReduce/BinningCancellationCategories



2.7 Most Visited Destination (TOP 10 Algorithm)

Finding the most visited destination from a source and its count.

hadoop jar /Users/ajaygoel/.m2/repository/Final_Project/Project/1.0-SNAPSHOT/Project-1.0-SNAPSHOT.jar top10srcDstFlights.Driver /project/data /Project_mapReduce/top10srcDstFlights

```
SFO,LAX 338471
LAX,SFO 336938
LAX,LAS 292125
LAS,LAX 286328
PHX,LAX 279716
LAX,PHX 279115
ORD,MSP 249960
MSP,ORD 249250
```

2.8 Carrieer and Source Join (Joins: Inner Join)

Joining the Carrier code in one file with other file to display the source, destination, distance and the name of the airline carrier.

```
-->SAN-->OAK-->446  -->American Airlines Inc.
-->SAN-->OAK-->446  -->American Airlines Inc.
-->SAN-->OAK-->446  -->American Airlines Inc.
-->SAN-->OAK-->446  -->American Airlines Inc.
-->SAN-->OAK-->446  -->American Airlines Inc.
```

```
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
-->DFW-->ORD-->802  -->American Airlines Inc.
```

2.9 Hierarchy Algorithm (XML Creation using Data Organization Pattern Algorithm)

Creating a hierarchy in form of XML of States and cities present in it.

part-r-00000

```
<State_City><state>"MS"</state><city_names><city_name>"Bay Springs"</city_name></city_names></State_City>
<State_City><state>"TX"</state><city_names><city_name>"Livingston"</city_name></city_names></State_City>
<State_City><state>"CO"</state><city_names><city_name>"Colorado Springs"</city_name></city_names></State_City>
<State_City><state>"NY"</state><city_names><city_name>"Perry"</city_name></city_names></State_City>
<State_City><state>"FL"</state><city_names><city_name>"Hilliard"</city_name></city_names></State_City>
<State_City><state>"MS"</state><city_names><city_name>"Belmont"</city_name></city_names></State_City>
<State_City><state>"AL"</state><city_names><city_name>"Clanton"</city_name></city_names></State_City>
<State_City><state>"WI"</state><city_names><city_name>"Brookfield"</city_name></city_names></State_City>
<State_City><state>"OH"</state><city_names><city_name>"East Liverpool"</city_name></city_names></State_City>
<State_City><state>"MO"</state><city_names><city_name>"Memphis"</city_name></city_names></State_City>
<State_City><state>"MS"</state><city_names><city_name>"Pittsboro"</city_name></city_names></State_City>
<State_City><state>"MN"</state><city_names><city_name>"Hawley"</city_name></city_names></State_City>
<State_City><state>"IN"</state><city_names><city_name>"Griffith"</city_name></city_names></State_City>
<State_City><state>"TX"</state><city_names><city_name>"Gatesville"</city_name></city_names></State_City>
```

File contents

<State_City><state>"MS"</state><city_names><city_name>"Bay Springs"</city_name>
</city_names></State_City>
<State_City><state>"TX"</state><city_names><city_name>"Livingston"</city_name>
</city_names></State_City>
<State_City><state>"CO"</state><city_names><city_name>"Colorado Springs"</city_name>
</city_names></State_City>
<State_City><state>"NY"</state><city_names><city_name>"Perry"</city_name></city_names>
</State_City>

2.10 WORST 20 Flights (Filter Pattern Algorithm):

Unique maximum delays worst top 20.

Challenge: find unique in 10 years: Used a nested TreeMap in this.

```
LAX
STL
DTW
MSP
ATL
MSP
DFW
EWR
RDU
GRR
MEM
BIL
MIA
ORD
ATL
DFW
EWR
MEM
BNA
ORD
```

2.11 Sampling using low scoring probability (Filter Pattern Algorithm)

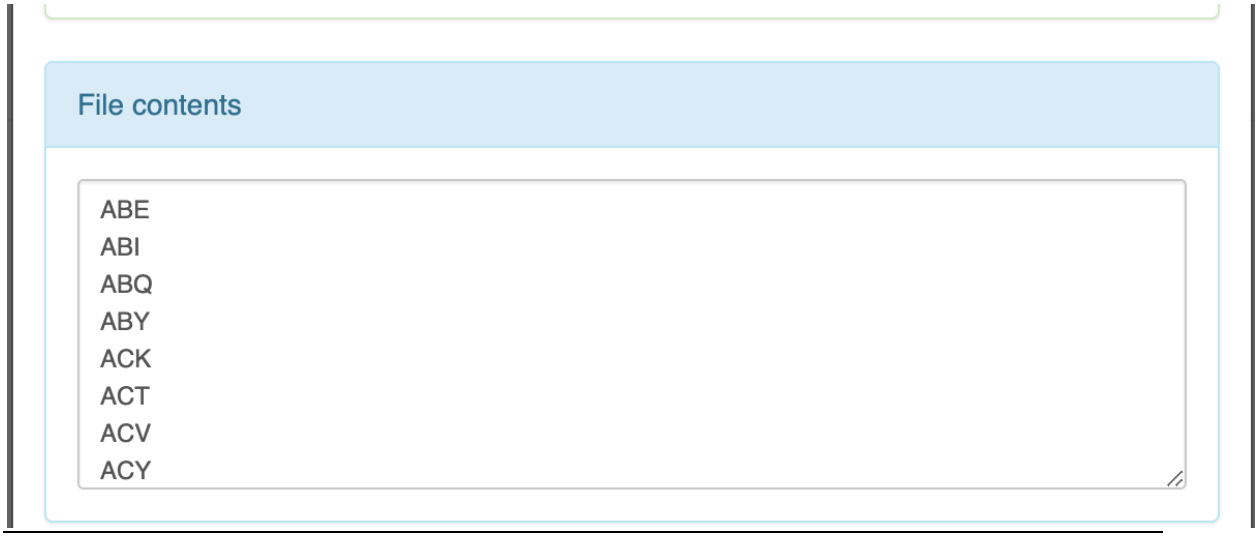
Sampling:

Based the percentage of the filter it will emit data randomly.

```
1987,12,25,5,1238,1215,1354,1323,NW,963,NA,76,68,NA,31,23,MEM,MSY,349,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,27,7,1244,1215,1410,1323,NW,963,NA,86,68,NA,47,29,MEM,MSY,349,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,28,1,1215,1215,1333,1323,NW,963,NA,78,68,NA,10,0,MEM,MSY,349,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,1,2,1620,1620,1935,1910,NW,964,NA,135,110,NA,25,0,MEM,MCO,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,2,3,1620,1620,1910,1910,NW,964,NA,110,110,NA,0,0,MEM,MCO,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,3,4,1619,1620,1859,1910,NW,964,NA,100,110,NA,-11,-1,MEM,MCO,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,4,5,1632,1620,1937,1910,NW,964,NA,125,110,NA,27,12,MEM,MCO,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,5,6,1620,1620,1904,1910,NW,964,NA,104,110,NA,-6,0,MEM,MCO,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
```

2.12 Finding unique sources:

hadoop jar /Users/ajaygoel/.m2/repository/Final_Project/Project/1.0-SNAPSHOT/Project-1.0-SNAPSHOT.jar uniqueSourceAirport.Driver /project/data /Project_mapReduce/uniqueSourceAirport



2.13 Total Number of flights in each year (Min Max Tuple):

hadoop jar /Users/ajaygoel/.m2/repository/Final_Project/Project/1.0-SNAPSHOT/Project-1.0-SNAPSHOT.jar totalDataByYear.Driver /project/data /Project_mapReduce/totalDataByYear

File contents

```
1987    1311826
1988    5202093
1989    5041197
1990    5270890
1991    5076922
1992    5092154
1993    5070498
1994    5180045
```

2.14 Standard deviation, Mean Distance in each year:

hadoop jar /Users/ajaygoel/Documents/Neu/BigData/Project/Final_Project/target/Project-1.0-SNAPSHOT.jar flightDistanceSDMed.DistanceDriver /project/data /Project_mapReduce/flightDistanceSDMed

File contents

```
1987    Standard Deviation : 497.96115 --- Median_Distance: 416.0
1988    Standard Deviation : 500.31335 --- Median_Distance: 432.0
1989    Standard Deviation : 513.059 --- Median_Distance: 446.0
1990    Standard Deviation : 519.819 --- Median_Distance: 453.0
1991    Standard Deviation : 521.1285 --- Median_Distance: 475.0
1992    Standard Deviation : 522.54407 --- Median_Distance: 487.0
1993    Standard Deviation : 520.97186 --- Median_Distance: 508.0
1994    Standard Deviation : 521.2502 --- Median_Distance: 508.0
```

2.15 Distributed Grep (Filter Pattern Algorithm)

Searching with regular expression as 848.

```
1987,12,7,1,1750,1750,1848,1851,NW,967,NA,118,121,NA,-3,0,MCO,MEM,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,28,1,1750,1750,1848,1851,NW,967,NA,118,121,NA,-3,0,MCO,MEM,683,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,19,6,1648,1555,1848,1658,NW,974,NA,120,63,NA,110,53,MEM,STL,256,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,24,4,1848,1740,2105,2008,NW,974,NA,77,88,NA,57,68,STL,DTW,440,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,3,4,1510,1510,1848,1853,NW,986,NA,158,163,NA,-5,0,TUS,MEM,1224,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,6,7,1848,1900,1930,1935,NW,997,NA,42,35,NA,-5,-12,SNA,LAX,36,NA,NA,0,NA,0,NA,NA,NA,NA,NA
1987,12,1,2,810,815,848,855,NW,998,NA,38,40,NA,-7,-5,LAX,SNA,36,NA,NA,0,NA,0,NA,NA,NA,NA,NA
```

3. PIG

3.1 Create schema from the script Schema.pig

```
grunt>
grunt> flightData = LOAD '/project/data/1987.csv' USING PigStorage(',') AS (
>> Year:int,
>> Month:int,
>> DayofMonth:int,
>> DayOfWeek:int,
>> DepTime :int,
>> CRSDepTime:int,
>> ArrTime:int,
>> CRSArrTime:int,
>> UniqueCarrier:chararray,
>> FlightNum:int,
>> TailNum:chararray,
>> ActualElapsedTime:int,
>> CRSElapsedTime:int,
>> AirTime:int,
>> ArrDelay:int,
>> DepDelay:int,
>> Origin:chararray,
>> Dest:chararray,
>> Distance:int,
>> TaxiIn:int,
>> TaxiOut:int,
>> Cancelled:int,
>> CancellationCode :chararray,
>> Diverted:chararray,
>> CarrierDelay:int,
>> WeatherDelay:int,
>> NASDelay:int,
>> SecurityDelay:int,
>> LateAircraftDelay:int);
grunt> carrierData = LOAD '/project/data2/carriers.csv' USING PigStorage(',') AS
>> (code:chararray, description:chararray);
grunt> airportData = LOAD '/project/data2/airports.csv' USING PigStorage(',') AS
>> (Iata:chararray,airport:chararray, city:chararray, state:chararray
```

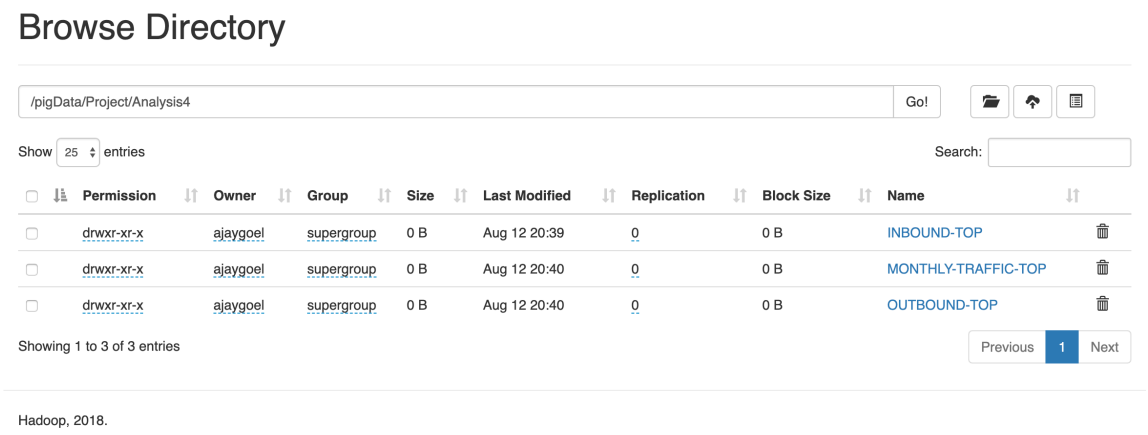
3.2 Top 20 cities by total volume of flights

What are the busiest cities by total flight traffic. JFK will feature, but what are the others? For each airport code compute the number of inbound, outbound and all flights. Variation on the theme: compute the above by day, week, month, and over the years.(month,iata,trafic) .

exec /Users/ajaygoel/Documents/Neu/BigData/Project/PIG/Analysis4.pig

File contents

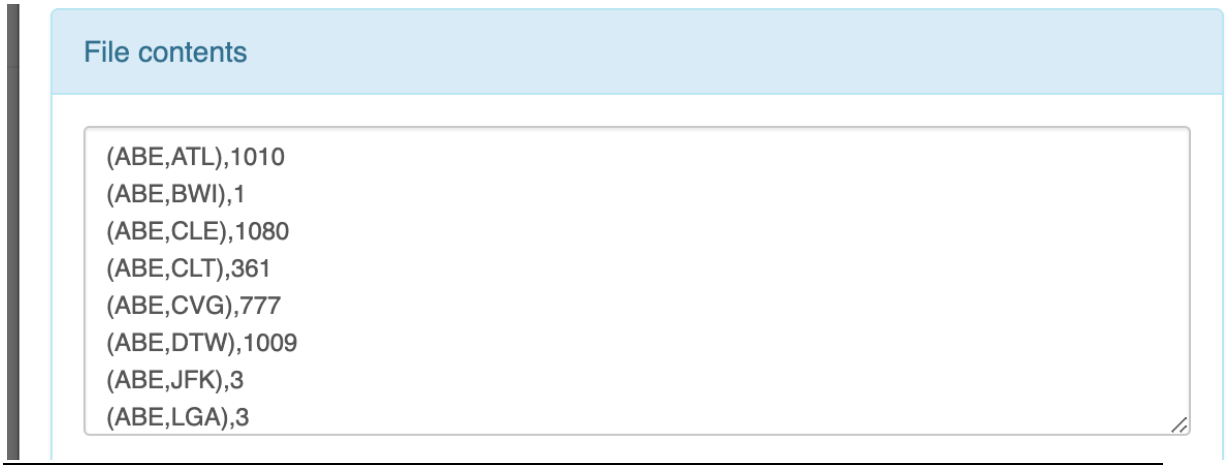
```
1,BWI,8860
1,CVG,9576
1,JFK,10492
1,CLT,10402
1,MCO,10282
1,LGA,10620
1,LAS,15063
1,MSP,12934
```



3.3 Busy Routes:

Which are busy the routes? A simple first approach is to create a frequency table for the unordered pair (i,j) where i and j are distinct airport codes. (Source, Destination, Count)

exec /Users/ajaygoel/Documents/Neu/BigData/Project/PIG/Analysis5.pig



3.4 Proportion of Flights Delayed

A flight is delayed if the delay is greater than 15 minutes. Compute the fraction of delayed flights per different time granularities (hour, day, week, month, year): m,dow,count(tot),count(del), count(tot)/count(del).

File contents

```
(1,1),102646,30501,0.29714748
(1,2),101698,22154,0.21784106
(1,3),102465,20666,0.20168838
(1,4),83417,21902,0.2625604
(1,5),83547,25298,0.3027996
(1,6),69351,13494,0.19457541
(1,7),78435,24311,0.30995092
(2,1),83765,25955,0.30985495
```

3.5 Carrier Popularity

Some carriers come and go, others demonstrate regular growth. Compute the (log base 10) volume -- total flights -- over each year, by carrier. The carriers are ranked by their median volume (over the 10 year span): Month, C.Name, log(count(C.Data))

File contents

```
1,9E,4.32349960843175
1,AA,4.741340724046487
1,AQ,3.588271706842329
1,AS,4.108158951256403
1,B6,4.186589091272407
1,CO,4.415757710035791
1,DL,4.594127251355629
1,EV,4.3539931244194845
```

3.6 Flights belonging to a particular carrier:

```
joinFlightData = join flightData by UniqueCarrier, carrierData by code;
filterFlightData = filter joinFlightData by description matches 'Southwest Airlines Co.';
store filterFlightData INTO '/pigData/Project/Analysis3' USING PigStorage(',');
```

File contents

```
30
```

4. HIVE

4.1 Schema creation:

CREATE Schema from HIVE_SCHEMA

4.2 Flights that started late but reached on time:

```
INSERT OVERWRITE DIRECTORY '/Output/HiveOutput/Analysis1'
select
Year,Month,DayofMonth,Origin,Dest,AirTime,Distance,TaxiIn,TaxiOut
from ontimeperf where DepTime>CRSDepTime and
ArrTime<=CRSArrTime;
```

4.3 Flights that travel less than 1000 miles:

```
INSERT OVERWRITE DIRECTORY '/Output/HiveOutput/Analysis2.0'
select count(*) from onTimePerf where Distance > 1000;
INSERT OVERWRITE DIRECTORY '/Output/HiveOutput/Analysis2.1'
select count(*) from onTimePerf where Distance < 1000;
```

4.4 Count of flights for each Carrier:

```
INSERT OVERWRITE DIRECTORY '/Output/HiveOutput/Analysis3'
Select carriers.description, uniqCount.countCancelled,
uniqCount.countCarrier from
(Select UniqueCarrier, sum(cancelled) as countCancelled, count(*) as
countCarrier from onTimePerf group by UniqueCarrier)
AS uniqCount, carriers
where carriers.code = uniqCount.UniqueCarrier;
HIVE ANALYSIS 4: FLIGHTS THAT TOOK MORE THAN 15 MINS
TO TAXIIN AND TAXOUT
select count(*) from ontimeperf where TaxiIn+ TaxiOut>15;
```

5. Sentiment Analysis

5.1 Analyzing Tweets of POTUS from twitter using vader lexicon dictionary.

Trump Sentiment Analysis

Analyzing and draw conclusions about the current US President's tweet behavior using data from the Twitter API.

Load Libraries and Setup

In [2]:
```
import csv
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import zipfile

# Ensure that Pandas shows at least 280 characters in columns, so we can see full tweets
pd.set_option('max_colwidth', 280)

%matplotlib inline
plt.style.use('fivethirtyeight')
import seaborn as sns
sns.set()
sns.set_context("talk")
import re
import tweepy
```

In [8]:
```
def load_keys(path):
    """Loads your Twitter authentication keys from a file on disk.

    Args:
        path (str): The path to your key file.  The file should
          be in JSON format and look like this (but filled in):
            {
                "consumer_key": "4itQ0urqREG2aAJNg6Eyzajgk",
                "consumer_secret":  "CSXRjWOzEgrG7ClfwLNJcyxmmuLIFPQgWjuk5UNsQi8W6TDZkn"
                "access_token": "2974941042-wO8homuub4QlhSiU4LOdWR2fSjq2V1MTQ3UsxWs",
                "access_token_secret": "zaubrsUNIBzsYWaVibCVTAQuAMUJxpXj9DMPEQ963Yifj"
            }

    Returns:
        dict: A dictionary mapping key names (like "consumer_key") to
          key values."""

    with open(path) as f:
        keys = json.load(f)
    return keys
```

```
trump['est_time'] = (
    trump['time'].dt.tz_localize("UTC") # Set initial timezone to UTC
                 .dt.tz_convert("EST") # Convert to Eastern Time
)
trump.head()
```

id	time	source	text
1159473911748317189	2019-08-08 14:38:17	Twitter for iPhone	....John Deere, our car companies, & others, to compete on a level playing field. With substantial Fed Cuts (there is no inflation) and no quantitative tightening, the dollar will make it possible for our companies to win against any competition. We have the greatest comp...
1159473912616501254	2019-08-08 14:38:17	Twitter for iPhone	....in the world, there is nobody even close, but unfortunately the same cannot be said about our Federal Reserve. They have called it wrong at every step of the way, and we are still winning. Can you imagine what would happen if they actually called it right?
1159522492324687877	2019-08-08 17:51:19	Twitter for iPhone	Iran is in serious financial trouble. They want desperately to talk to the U.S., but are given mixed signals from all of those purporting to represent us, including President Macron of France....
1159522493801123840	2019-08-08 17:51:20	Twitter for iPhone	....I know Emmanuel means well, as do all others, but nobody speaks for the United States but the United States itself. No one is authorized in any way, shape, or form, to represent us!
1159590032514134016	2019-08-08 22:19:42	Twitter for iPhone	Sue Gordon is a great professional with a long and distinguished career. I have gotten to know Sue over the past 2 years and have developed great respect for her. Sue has announced she will be leaving on August 15, which....

```
#Trumps most negative tweets
print('Most negative tweets:')
for t in trump.sort_values('polarity').head()['text']:
    print('\n  ', t)
```

```
Most negative tweets:

   ....to inflame and cause chaos. they create their own violence, and then try to blame others. they are the true ra
cists, and are very bad for our country!

   rt @w_terrence: died of suicide on 24/7 suicide watch ? yeah right! how does that happen

#jefferyepstein had information on bill clinton &amp;amp;…

   maggie haberman of the failing @nytimes reported that i was annoyed by the lack of cameras inside the hospitals in
dayton &amp;amp; el paso, when in fact i was the one who stated, very strongly, that i didn't want the fake news inside &
amp; told my people not to let them in. fake reporting!

   ..tv talent. then, during the 2016 election, i would watch as joe scarborough &amp;amp; his very angry psycho wife(?)
would push donny to the point of total humiliation. he would never fight back because he wanted to stay on tv, even o
n a very low rated show, all in the name of ambition!

   never has the press been more inaccurate, unfair or corrupt! we are not fighting the democrats, they are easy, we
are fighting the seriously dishonest and unhinged lamestream media. they have gone totally crazy. make america great
again!
```

n [45]:

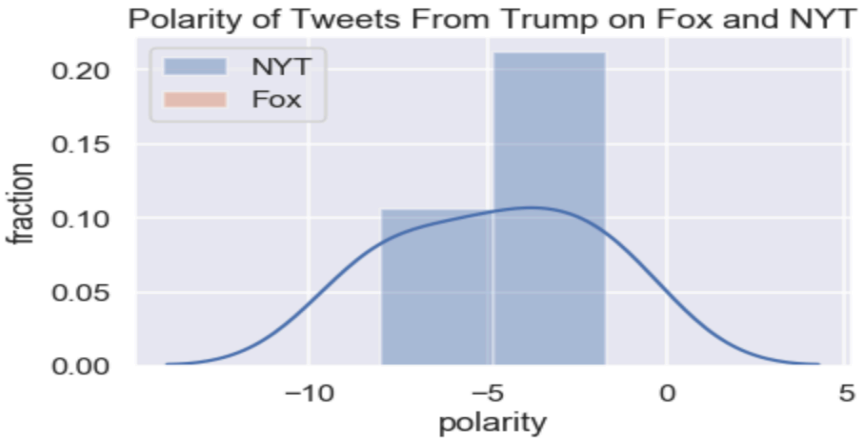
```
#Trumps most positive Tweets
print('Most positive tweets:')
for t in trump.sort_values('polarity', ascending=False).head()['text']:
    print('\n  ', t)
```

```
Most positive tweets:

   i am pleased to inform you that the honorable joseph maguire, current director of the national counterterrorism ce
nter, will be named acting director of national intelligence, effective august 15th. admiral maguire has a long and d
istinguished....
```

Goel, Ajay: Nu Id (001897443)

In [46]:
```
nyt = trump[trump["no_punc"].str.contains('nyt')]['polarity']
fox =  trump[trump["no_punc"].str.contains('fox')]['polarity']

sns.distplot(trump.loc[nyt.index.values]["polarity"], label = "NYT")
sns.distplot(trump.loc[fox.index.values]["polarity"], label = "Fox")
plt.ylabel("fraction")
plt.legend()
plt.title("Polarity of Tweets From Trump on Fox and NYT")
plt.show()
```

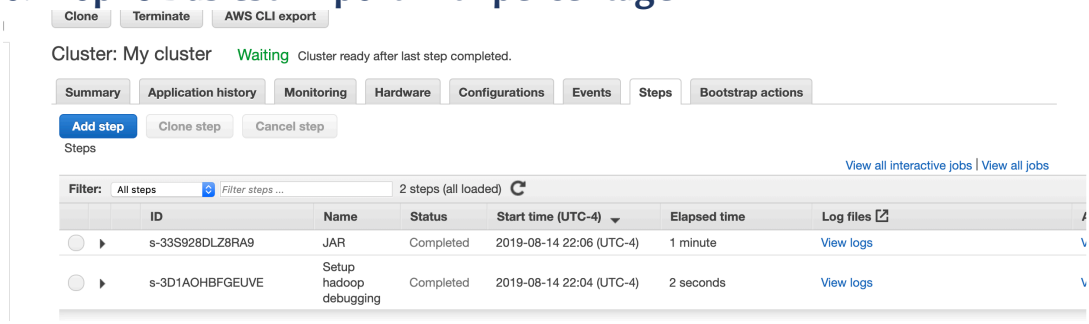
```
//anaconda3/lib/python3.7/site-packages/seaborn/distributions.py:198
  line, = ax.plot(a.mean(), 0)
//anaconda3/lib/python3.7/site-packages/numpy/core/_methods.py:85: R
e_scalars
  ret = ret.dtype.type(ret / rcount)
//anaconda3/lib/python3.7/site-packages/numpy/lib/histograms.py:893:
e_divide
  return n/db/n.sum(), bin_edges
```



Goel, Ajay: Nu Id (001897443)

6. AMAZON EMR

6.1 Top 10 Busiest Airport with percentage



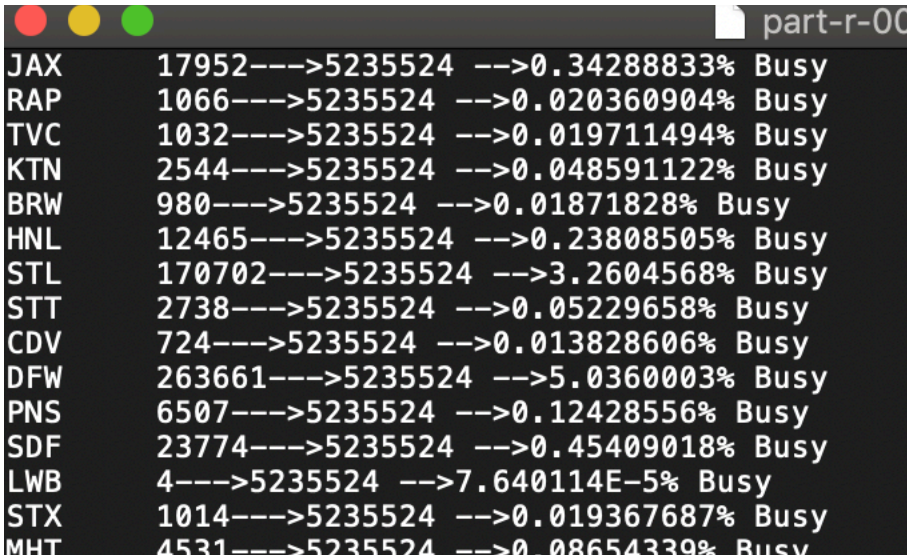
Last modified
Aug 14, 2019 10:07:57 PM GMT-0400

Etag
59e18dbe076e1bb7cf23c0e128c482ca

Storage class
Standard

Server-side encryption
None

Size
8.6 KB

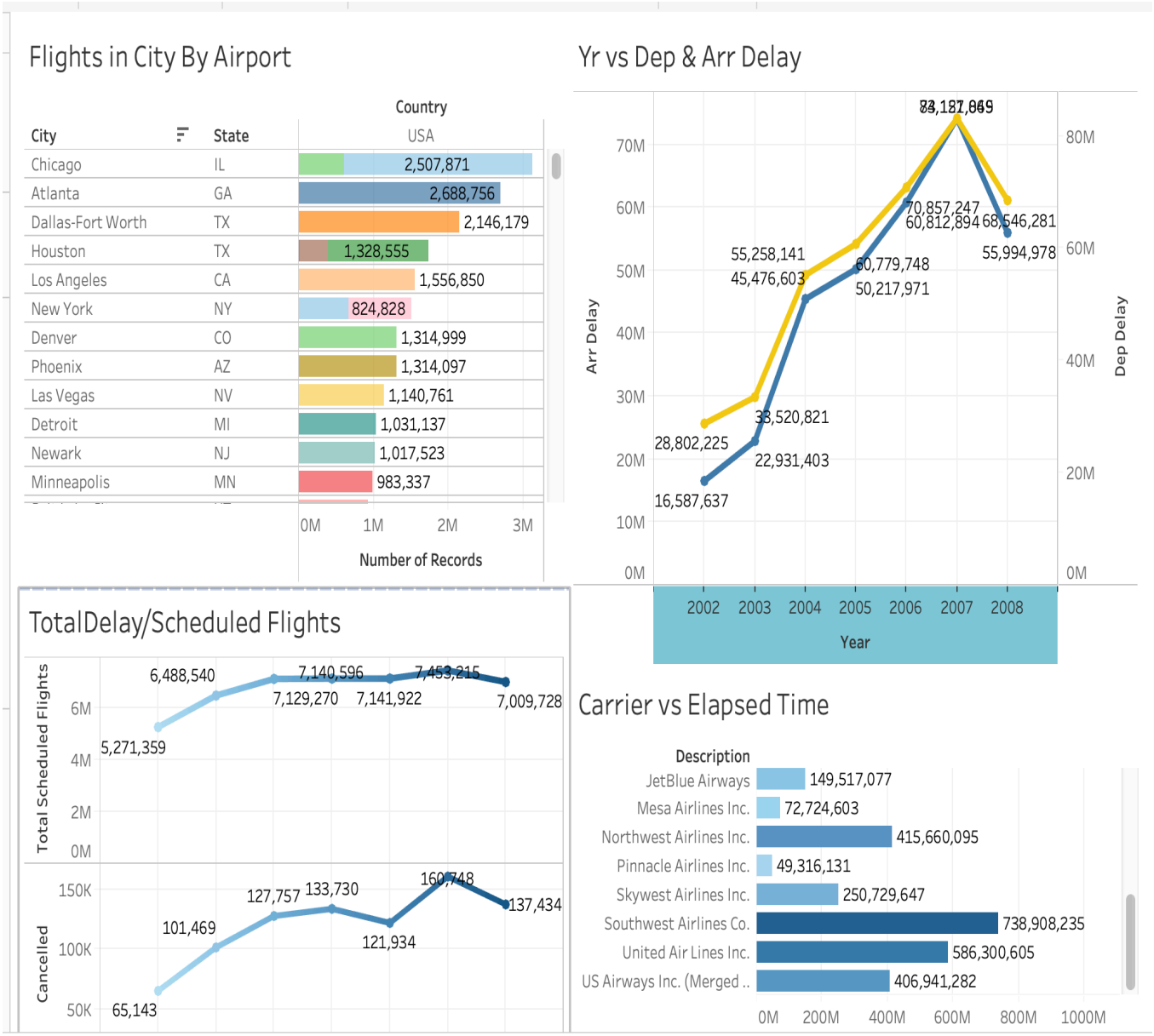


Goel, Ajay: Nu Id (001897443)

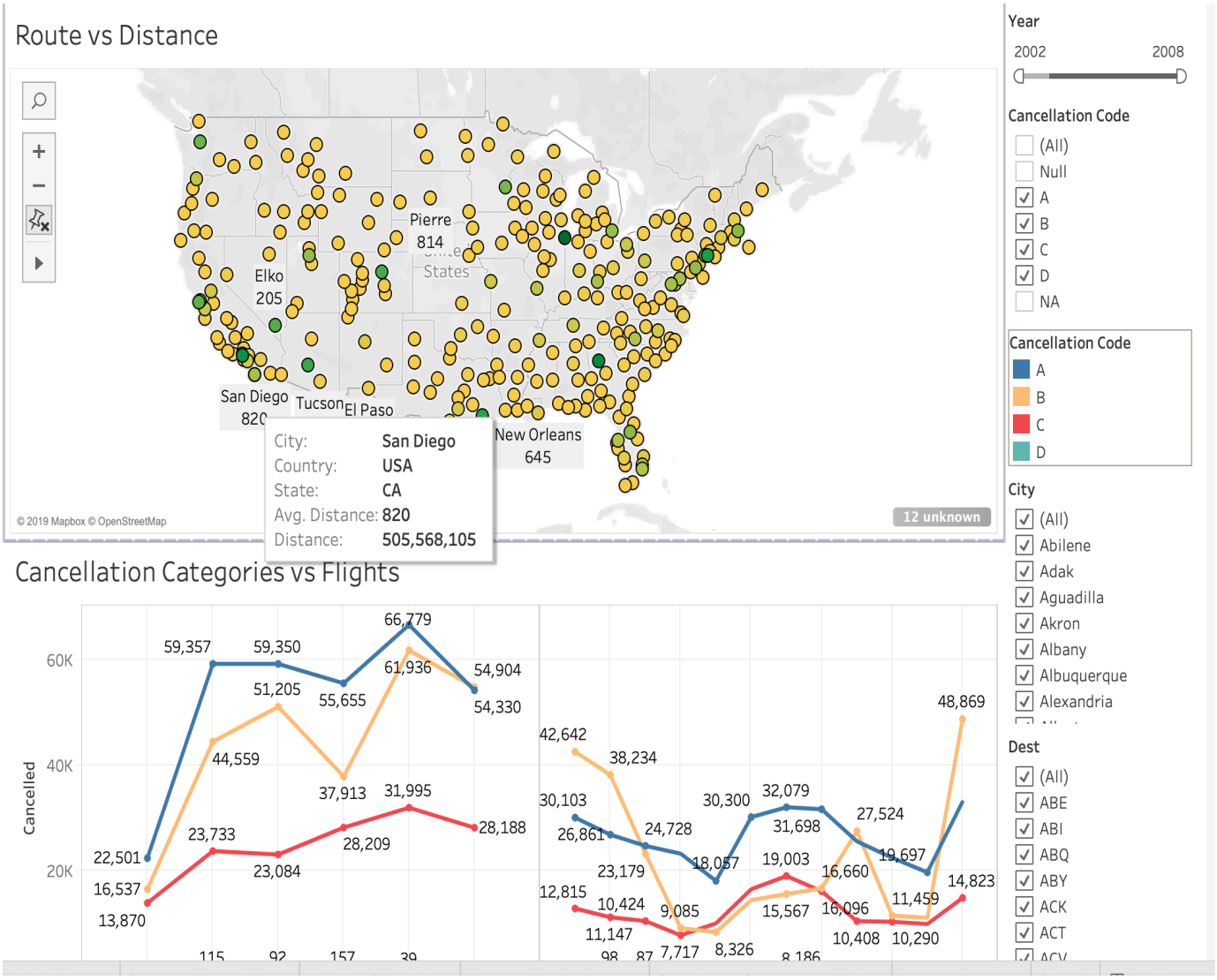
Business Visualization:

7. TABLEAU

7.1 Dashboard 1:



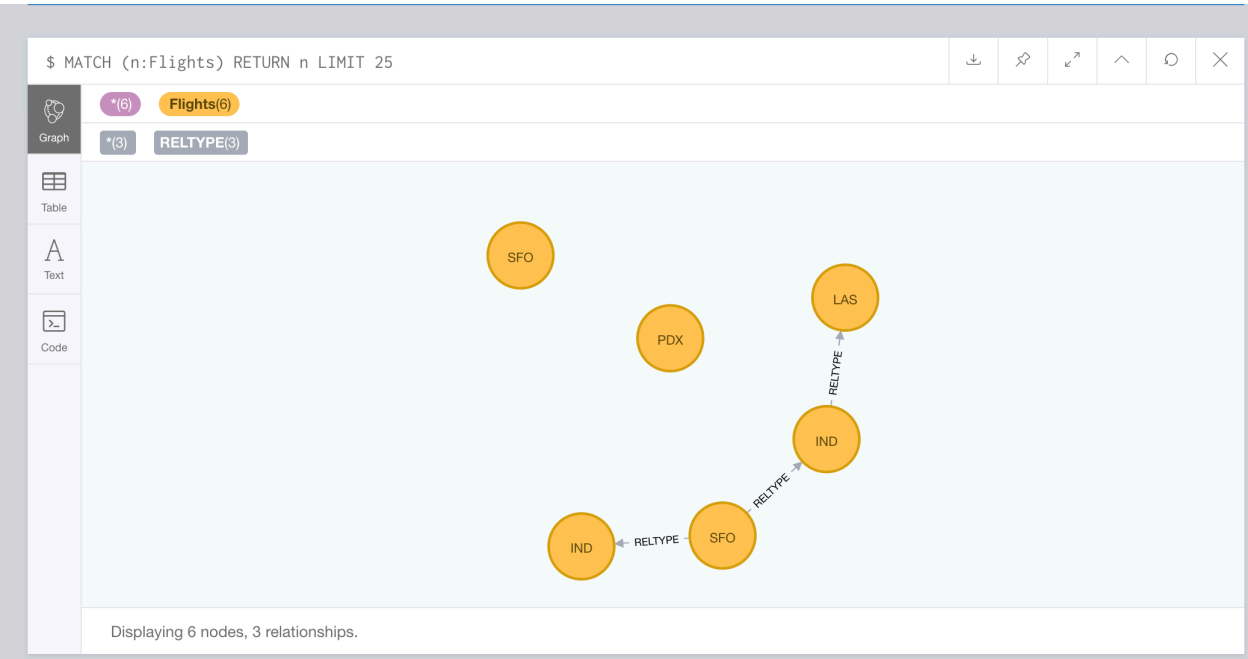
7.2 Dashboard 2:



8. Graph Database:

8.1 Neo4j

- Created 6 Nodes having origin and Destination.
- Load Script is in commands_neo4j.
- Creating mapping from origin to destination flights.



REFERENCES:

- All Year Data : http://stat-computing.org/dataexpo/2009/the-data.html
- Carrier Code csv : http://stat-computing.org/dataexpo/2009/supplemental-data.html
- Airports csv : http://stat-computing.org/dataexpo/2009/supplemental-data.html
- Plane-data csv : http://stat-computing.org/dataexpo/2009/supplemental-data.html
- MapReduce Design Patterns: https://learning.oreilly.com/library/view/mapreduce-design-patterns/
- Neo4j Beginner tutorial: https://www.youtube.com/watch?v=G03P73-KV30&t=504s
- Neo4j: https://neo4j.com/docs/cypher-manual/current/clauses/create/
- Neo4j: https://neo4j.com/sandbox-v2/