Peer-Graded Assignment: Analyzing Big Data with SQL
Name: Jian Carlo CALAUNAN
Date: 2020-09-08

Assignment

Recommend which pair of United States airports should be connected with a high-speed passenger rail tunnel. To do this, write and run a SELECT statement to return pairs of airports that are between **300** and **400** miles apart and that had at least **5,000** (five thousand) flights per year on average *in each direction* between them. Arrange the rows to identify which one of these pairs of airports has largest total number of seats on the planes that flew between them. Your SELECT statement must return all the information required to fill in the table below.

Recommendation

I recommend the following tunnel route:

	First Direction	Second Direction
Three-letter airport code for origin	SFO	LAX
Three-letter airport code for destination	LAX	SFO
Average flight distance in miles	337	337
Average number of flights per year	14712	14540
Average annual passenger capacity	1996597	1981059
Average arrival delay in minutes	10	14

Method

I identified this route by running the following SELECT statement using Impala on the VM:

```
/*  Abbreviations:
   AFD - avg. flight dist
   ANFY - avg. num. flights per year
   AAPC - avg. annual passenger capacity
   AADM - avg. arrival delay in minutes
*/
SELECT f.origin, f.dest,
      round(avg(f.distance)) AS AFD,
      round(sum(p.seats)/count(DISTINCT f.year)) as AAPC,
      round(count(f.flight)/count(DISTINCT f.year)) as ANFY
   FROM flights as f
      LEFT OUTER JOIN planes as p
      ON f.tailnum = p.tailnum
```

WHERE (distance BETWEEN 300 AND 400) AND (f.origin IS NOT NULL AND f.dest IS NOT NULL)
GROUP BY f.origin, f.dest
HAVING ANFY > 5000
ORDER BY AAPC DESC, ANFY DESC, AFD DESC
LIMIT 20;

Notes

Based on final table provided, there appear to be 4 aggregated columns and two main columns to be queried.

1. Analyze all the tables within fly database
    a. Look for relationships between the tables, required in the querying process.
    b. Take note of units for conversions where required.

2. Breakdown each query into smaller groups and test functionality
    a. Find average distance between two airports
        i. Use avg function on distance column

    b. Determine average number of flights per year
        i. Confirm the number of years of flights recorded first
            1. Using *COUNT * and GROUP BY year* OR *SELECT DISTINCT year*
        ii. Obtain total number of rows and divide by number of unique years
            1. SELECT COUNT(*)/SUM(DISTINCT year) as avg_num_flights FROM flights;
            2. Alter * to specific column in the main query i.e. flights

    c. Find average aircraft passenger capacity per year by querying planes table
        i. Same process as b.
            1. SELECT SUM(seats) as FROM planes WHERE year BETWEEN 2008 AND 2017;
            Careful to note years do not match those in flights database, i.e, quantity, so explicitly need to state 10 or refer to flights year column
    d. Average arrival delay in minutes
        i. Same process as a)

3. Filter per the requirements:
    a. Columns needed for presentation, to reduce compute and networking resources
        i. Origin and dest are all that's required, as others are aggregations
    b. JOIN tables flights and planes
        i. Left outer Join to incorporate all rows relating to Origin and Dest
    c. FILTER BY
        i. Distance to be between 300 and 400
    d. ORDER BY seats to be from highest to lowest to obtain the highest amount
        i. Remove nulls as impala will sort them at the top

    e. Slowly integrate individual queries in Step 2.
        i. Starting with avg no. of flights (ANFY) greater then or equal to 5000, to add to the SELECT and HAVING statement
        ii. Repeat for remaining

4. Confirm final order within ORDER BY based on requirements:
    a. Largest number of passengers capacity annually [AAPC]
    b. Average number flights per year
    c. Average flight delay
    d. Average distance

5. Limit by 10 or in this case 20 as they come in pairs.