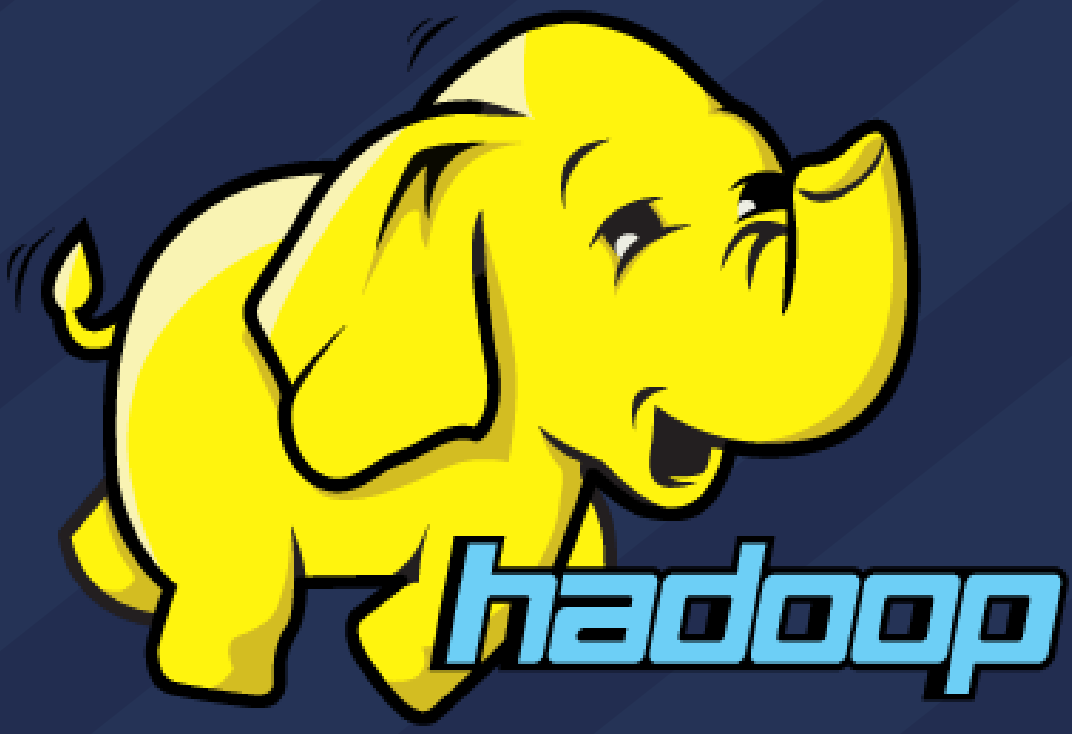
Thiago Panini

2020-Feb-08

PEER-GRADED EXERCISE

ANALYSIS FOR BUSINESS DECISION



schema/database:

fly

tables:

flights, planes

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.

Recomendation

I recommend the following tunnel route:

	First Direction	Second Direction
Three-letter airport code for origin	LAX	SFO
Three-letter airport code for destination	SFO	LAX
Average flight distance in miles	337	337
Average number of flights per year	14,540	14,712
Average anual passenger capacity	1,981,059	1,996,597
Average arrival delay in minutes	13.98	10.50

Method

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

```
SELECT

    f.origin,
    f.dest,
    Max(p.seats)                                    AS max_seats,
    Round(Count(*)/Count(DISTINCT f.year), 0)       AS flights_per_year,
    Round(Avg(f.distance), 2)                       AS avg_distance,
    Round(Sum(p.seats)/Count(DISTINCT f.year),2) AS avg_seats_per_year,
    Round(Avg(f.arr_delay), 2)                      AS avg_arr_delay
FROM   fly.flights f
       LEFT JOIN fly.planes p
              ON f.tailnum = p.tailnum
GROUP  BY origin,
          dest
HAVING avg_distance BETWEEN 300 AND 400
       AND flights_per_year > 5000
ORDER  BY max_seats DESC
LIMIT  2;
```

Notes

- The query was built to be consistent for many years: Instead of dividing the averages per year by ten (as long as we have 10 years of data in the flights table), I use the COUNT(DISTINCT) function to count the total of years presented by the table. With this, even if the database manager or the owner of the table inserts data from another year, the averages will still consistent.