INF 551 – Fall 2017

Homework #5

Due: November 27, Monday
100 points

In this homework, you are asked to analyze the flight and passenger data of LAX using Hadoop MapReduce and Spark. There are two data sets:

- lax_flights.csv contains the number of flights for each month from January 2006 to August 2017 for different types of flights (charter, commuter, and scheduled), whether it is arrival or departure, and domestic or international.
  More details about the lax flight data set can be found here: https://catalog.data.gov/dataset/los-angeles-international-airport-flight-operations-by-month-afb2a
- lax_passengers.csv contains the number passengers in the same time period for different terminals, arrival or departure, domestic or international.
  More details are available here: https://catalog.data.gov/dataset/los-angeles-international-airport-passenger-traffic-by-terminal-756ee

Note that both files are comma separated. For your reference, data sets with header are also provided as lax_flights_header.csv and lax_passengers_header.csv. Note that you can ignore the DataExtractDate in both data sets. The time relevant to the homework is the ReportPeriod. Note that 01/01/2006 is the data for January of 2006, and so on.

1. [Hadoop MapReduce, 50 points] Use lax_passengers.csv. Write a MapReduce program "BusyMonth.java" in Java that finds each month when the number of passengers travelling through Terminals 1 to 8 and Tom Bradley International Terminal collectively exceeds (i.e., greater than) 5 millions in the month. Report the months and the number of passengers in each month as follows:

   01/2006 5,001,008
   08/2006 6,000,134
   …

   Submission: BusyMonth.java
   Note: You can assume that the data set file is located under an "input" directory.

2. [Spark, 50 points]
   a. [15 points] Answer the same question as in Question 1, but instead write a Spark script in Python, named "BusyMonth.py".

      Execution: spark-submit BusyMonth.py lax_passengers.csv

b. [15 points] Use lax_flights.csv. Write a Spark script "AvgNumFlights.py" that reports, for each year, the average number of flights (remove the fractional part) per month in the year, for the different directions of flights (arrival or departure). Example output:

    2006, arrival, 207
    2006, departure, 302
    …

    Execution: spark-submit AvgNumFlights.py lax_flights.csv

c. [20 points] Use both lax_flights.csv and lax_passengers.csv. Write a Spark script "FlightAndPassenger.py" that reports for each month, the number of flights and passengers in that month, broken down by arrival/department, domestic/international. For example:

    01/2006, Arrival, Domestic, 51, 19748
    01/2006, Arrival, International, 75, 20383
    01/2006, Departure, Domestic, 79, 19388
    …

    Execution: spark-submit FlightAndPassenger.py lax_flights.csv lax_passengers.csv

Submission: BusyMonth.py, AvgNumFlights.py, FlightAndPassenger.py