Boston University Metropolitan College

Assignment 1
Big Data Analytics
MET CS777

Faculty - Farshid Alizadeh-Shabdiz, PhD, MBA

Problem 1 (20 points)

Write a python code (normal Python and not pyspark) to answer following question. "Social computing research at the university of Minnesota" has released moving rating data sets at different sizes at "gouplens.org" web site.

Load MovieLens 10M dataset, which consists of 10million movie ratings. You can down load the data by going to grouplens.org, and under the "datasets" tab, upload "movieLens 10M dataset" it is 63 MB.

a. Divide the data to 5 almost equal size files and use the five files in the rest of the assignment (2 points)
b. Sort the data from the highest rating movie to the lowest one.
   Measure how much time sorting takes. (6 points)
   - Don't use sort function, and write the sort function yourself.
   - Use sort function
c. Create histogram of the movie ratings.
   Measure how much time it takes to create the histogram. (2 points)
d. Data contains more than 10M ratings of 10681 movies by 71567 users.
   Create histogram of number of times each movie got rated.
   Measure how much time it takes to create the histogram. (4 points)
e. Choose the lowest three bins of histogram in part C and create a histogram of movie ratings for these three bins. Do the same thing for the top three bins of the histogram. (6 points)

Problem 2 (20 points)

Write a python code (normal Python and not pyspark) to answer following question.

Description

The goal is to analyze a data set consisting of New York City Taxi trip reports in the Year 2013. The dataset was released under the FOIL (The Freedom of Information

Law) and made public by Chris Whong (https://chriswhong.com/open-data/foil_nyc_taxi/).

Taxi Data set

The data set itself is a simple text file. Each taxi trip report is a different line in the file. Among other things, each trip report includes the starting point, the drop-off point, corresponding timestamps, and information related to the payment. The data are reported by the time that the trip ended, i.e., upon arrive in the order of the drop-off timestamps. The attributes present on each line of the file are in order as it was shown in table 1. The data files are in comma separated values (CSV) format.

	Attribute	Description
0	medallion	an md5sum of the identifier of the taxi - vehicle bound (Taxi ID)
1	hack_license	an md5sum of the identifier for the taxi license (Driver ID)
2	pickup_datetime	time when the passenger(s) were picked up
3	dropoff_datetime	time when the passenger(s) were dropped off
4	trip_time_in_secs	duration of the trip
5	trip_distance	trip distance in miles
6	pickup_longitude	longitude coordinate of the pickup location
7	pickup_latitude	latitude coordinate of the pickup location
8	dropoff_longitude	longitude coordinate of the drop-off location
9	dropoff_latitude	latitude coordinate of the drop-off location
10	payment_type	the payment method -credit card or cash
11	fare_amount	fare amount in dollars
12	surcharge	surcharge in dollars
13	mta_tax	tax in dollars
14	tip_amount	tip in dollars
15	tolls_amount	bridge and tunnel tolls in dollars
16	total_amount	total paid amount in dollars

Table 1: Taxi data set fields.

Obtaining the Dataset

The data set (93 MB compressed, uncompressed 384 MB) is available at the following url on Google Cloud : gs://meetcs777/taxi-data-sorted-small.csv.bz2

Assignment Tasks

Task 1 : Top-10 Active Taxis (5 points)

Many different taxis have had multiple drivers. Write and execute a Python program that computes the top ten taxis that have had the largest number of drivers. Your output should be a set of (medallion, number of drivers) pairs.

Note: You should consider that this is a real world data set that might include wrongly formatted data lines. You should clean up the data before the main processing, a line might not include all of the fields. If a data line is not correctly formatted, you should drop that line and do not consider it.

Report the processing time of the task as well.

Task 2 - Top-10 Best Drivers (7 Points)

We would like to figure out who the top 10 best drivers are in terms of their average earned money per minute spent carrying a customer. The total amount field is the total money earned on a trip. In the end, we are interested in computing a set of (driver, money per minute) pairs.

Report the processing time of the task as well.

Task 3 - Best time of the day to Work on Taxi (8 Points)

We would like to know which hour of the day is the best time for drivers that has the highest profit per miles. Consider the surcharge amount in dollar for each taxi ride (without tip amount) and the distance in miles, and sum up the rides for each hour of the day (24 hours) – consider the pickup time for your calculation. The profit ratio is the ration surcharge in dollar divided by the travel distance in miles for each specific time of the day.

Profit Ratio = (Surcharge Amount in US Dollar) / (Travel Distance in miles) We are interested to know the time of the day that has the highest profit ratio.

Report the processing time of the task as well.

Academic Misconduct Regarding Programming

In a programming class like our class, there is sometimes a very fine line between ”cheating” and acceptable and beneficial interaction between peers. Thus, it is very important that you fully understand what is and what is not allowed in terms of collaboration with your classmates. We want to be 100% precise, so that there can be no confusion.

The rule on collaboration and communication with your classmates is very simple: you cannot transmit or receive code from or to anyone in the class in any way—visually (by showing someone your code), electronically (by emailing, posting, or otherwise sending someone your code), verbally (by reading code to someone) or in any other way we have not yet imagined. Any other collaboration is acceptable.

The rule on collaboration and communication with people who are not your classmates (or your TAs or instructor) is also very simple: it is not allowed in any way, period. This disallows (for example) posting any questions of any nature to programming forums such as StackOverflow. As far as going to the web and using Google, we will apply the ”two line rule”. Go to any web page you like and do any search that you like. But you cannot take more than two lines of code from an external resource and actually include it in your assignment in any form. Note that changing variable names or otherwise transforming or obfuscating code you found on the web does not render the ” two line rule” inapplicable. It is still a violation to obtain more than two lines of code from an external resource and turn it in, whatever you do to those two lines after you first obtain them.

Furthermore, you should cite your sources. Add a comment to your code that includes the URL(s) that you consulted when constructing your solution. This turns out to be very helpful when you’re looking at something you wrote a while ago and you need to remind yourself what you were thinking.