EE3801 Data Engineering

Laboratory Exercise (LAB-II)

Assignment release date: Sept 09, 2020 Briefing: Sept 10, 2020

Date submission due: Sept 16, 2020

Grading: Your ASSIGNMENT will be graded out of 100 marks and the final weight of this assignment is 10%. The guidelines explained in the template will also carry marks. So please adhere to the guidelines.

NOTE: This is an individual lab exercise and NO DISCUSSIONS AND EXCHANGE OF SOLUTION IDEAS BETWEEN ANY STUDENTS ARE ALLOWED. If we come to know of this in any form, we will be taking relevant disciplinary actions. Please follow this guideline strictly.

Concepts used: Reading and writing data from/to csv files, generating new data frames, data wrangling - extraction, handling missing data, transforming a DF to a target DF, results visualization by plotting (different types – bar charts, pie charts, etc), and interpretation of results.

Data required for this assignment: taxis.csv

Consider the taxis rides dataset (***taxis.csv***) given to you. Note that the data is for the month of **March**. If you encounter issues reading the file, consider using: pd.read_csv(filename, encoding= 'unicode_escape')

Nomenclature: Dataframe – **DF**

Read the questions carefully. Answer all the following.

1) Cleaning the data:
    a. Separate date and time for both pickup and dropoff, creating four new columns, 'pickup_date', 'pickup_time', 'dropoff_date', 'dropoff_time'. For the dates, keep only the day value as the Year

and month remains the same. Add the new columns to the existing DF. Display the first 5 rows from your DF. [4 marks]

b. Compute the following for each car type- Green and Yellow. [10 marks]
   i. Total fare, use the fare column in the DF
   ii. Total number of passengers travelled
   iii. Total distance travelled
   iv. Total time of travel.
   Hint: Look up the datetime python library.

2) Extract the details of the trip with the **longest** distance travelled for each of color of cars (green and yellow). Do this for each of the pickup dates shown below where a customer paid using **cash**. **Dates**: 10, 15, 20, 25, and 30. If none for any case, return a 0. Display your output in a new dataframe named **GY_cash**, which should only have the following columns: Pickup_date, Color of the cab, distance travelled, pickup, pickup_date, pickup_time, dropoff, dropoff_date, dropoff_time, and the fare. [10 marks]

3) From your GY_cash, identify by extracting the information on which color cab travelled larger distance on each pickup date with the corresponding distance information. Display your result in a new dataframe named GY_maxDist. A typical row of this DF should be [Pickup_date, Color of the cab, distance travelled, pickup, pickup_date, pickup_time, dropoff, dropoff_date, dropoff_time, fare] [6 marks]

4) Write a function to calculate the actual speed in meters per second of the vehicle. Given the distance, start time and end time. Assume that the distances given are in kilometers. Use the ***apply*** method in pandas to apply the function to GY_maxDist and create a new column with this information called speed. Display your GY_maxDist DF. [7 marks]

5) Using the DF from Q1, starting from 'Brooklyn' borough and dropping off in 'Manhattan' borough, how many trips were made with the pickup date between the dates 10th March and 25th March? For each of these trips, compute the actual speed of the cars and output the mean speed of green cars and mean speed of yellow cars. [5 marks]

6) Between 2.30pm and 4pm on March 17th, which color cars had more pick ups?[4 marks]

7) Use your original taxis.csv dataset. Compute the following and report in a single dataframe: Compute the minimum, maximum, and the total number of passengers served from each of the pickup zones. Also compute the respective minimum, maximum, and the total fare for each zone. Display the first 5 rows of the dataframe. Sample row in your DF must be: [3 marks]

	passangers			fare		
pickup_zone	sum	min	max	sum	min	max
XXXXXX	2	1	1	74.69	10.50	54.16

8) Using the original data, for each pickup_zone , compute the following for each color of the cab and store this as two DFs, one for each color of the car (Yellow_stats_df, Green_stats_df) [10 marks]:

   a. Number of trips starting from a pickup_zone
   b. Total number of passengers travelled from that zone
   c. Total distance travelled from that zone
   d. Total fare (captured in the 'fare' col) each zone yields to that car company (green or yellow)
   e. Total fuel cost for each zone that a company needs to invest.
      i. To compute the total fuel cost, **assume** that a car (Yellow and Green) on an average consumes fuel around 5.5L per 100 kms and cost per litre is, say, $3.
      ii. Example: Actual dist travelled in a trip = 39.5 kms; Then the total fuel consumed in litres = (39.5 * 5.5 / 100) = 2.1725 L; Thus, the total fuel cost for this trip = 2.1725 * 3 = $6.5175

9) Then using your DFs in Q8 above, compute the following for Green and Yellow car: [8 Marks]
   a. Total number of passengers travelled
   b. Total distance travelled

c. Total fare
d. Total fuel cost for the month for that company

10) Plot the top 10 pickup zones by total fare using the following statistics from your Dfs in Q8.

a. Sort the pickup zones by total fare in descending order. Determine the top 10 pickup zones by total fare for each of the DFs in Q8. [5 marks]
b. **Bar Charts:** for each color car, using the top 10 pickup zones, plot a bar chart for each of the following statistics. You may decide if you wish to plot group bar charts or w.r.t zones, etc. [12 marks]
    i. Total number of passengers travelled
    ii. Total distance travelled
    iii. Total fare
    iv. Number of trips made
c. **Pie Chart:** for each color car, plot a pie chart of the total cost fare each zone yields and another pie chart of the total fuel cost for each zone [8 marks]