Assignment [100 marks, weight: 30%]

BEE2041: Data Science in Economics

In this assignment, you will demonstrate your understanding and mastery of using different data science tools.

What you will have learnt by the end of Week 7 should cover almost everything you will need, and what you learnt is already enough to start working on some problems. If you are stuck then read through the notebooks again. If you are still unsure, then have a look online. *Google* and *Stack OverFlow* are your friends!

The grade of this assignment contributes 30% towards your overall grade in the course. The following aspects need to be shown:

- Using the Bash
- Designing relational DBMS
- Writing queries in SQL
- Writing queries in MongoDB
- Using Python and its libraries, especially Pandas and Matplotlib
- Improving and extending analysis.

Your submission will be a compressed .zip file which includes a PDF, **Answers.pdf**, that contains your designs, scripts and screenshots of executing commands and queries for all problems P1-P5. In addition, your submission will include a text/Word file that contains the commands that you run in order to answer problems P1, P3, and P4 as well as a Jupyter Notebook file for P5. Failure to include any of the code files may result in deducting some marks.

Your scripts must be sufficient to reproduce your answers to all questions and plots. You are responsible for making sure that your files (including the Jupyter Notebook file) will open without errors. Submissions that do not open may receive a zero.

The deadline is **Monday 20th March at 15:00 (GMT)**

Collaboration & Misconduct: You are encouraged to think about this assignment in groups or ask each other for help. If you do, you should do the following: 1) write your own code (no code copying from others), 2) Report the names of all people that you worked with in your submission, 3) if you received help from someone, write that explicitly, 4) plagiarism of code or writeup will not be tolerated; do not copy blocks of code in your answers, and 5) do not post your solutions online (even after the release of your marks). For those who want to evidence your experience to recruiters, make sure you share a private link to your project/work (or undiscoverable link). **If I can find your answers online anytime until September this year, you will be reported for misconduct.**

The University takes poor academic practice and academic misconduct very seriously and expects all students to behave in a manner which upholds the principles of academic honesty. Please make sure you familiarise yourself with the general guidelines and rules from this link[1] and this link[2].

[1] http://as.exeter.ac.uk/academic-policy-standards/tqa-manual/aph/managingacademicmisconduct/

[2] https://vle.exeter.ac.uk/pluginfile.php/1794/course/section/27399/A%20Guide%20to%20Citing%2C%20Referencing%20and%20Avoiding%20Plagiarism%20V.2.0%202014.pdf

Problem 1 [15 marks]

For this problem, you need to use the Bash to do the following:

A. Using "echo", print your "Candidate Number" (find it on BART) to screen. Take a screenshot and include it in your submission.
B. Move inside the directory *DATE_FILES*
C. Count the number of files in this directory
D. Print the names of the first 8 files in this directory, along with information about owner, date, and size.
E. Move to the parent directory
F. Create a new directory there, named *second_10_days*
G. Copy from the *DATE_FILES* directory the files that are related to the days 10-19 of every month (140 files) to the newly created directory
H. Move inside *second_10_days* directory, and append the line "This is the last Line" to the end of file *2011_04_10*
J. Write a one-liner command to append the line "This is the last Line of X", where X is the name of the file, to the end of every file in the directory *second_10_days*
K. Using Bash: create a bash file *P1K.sh*. Write your code from J to it. Run the file *P1K.sh* file.

Problem 2 [20 marks]

For this problem, you can use Powerpoint or Keynote to do your designs (you may want to reuse boxes and lines from the Week 3 Powerpoint slides or Week 3 Exercises slides). Feel free to use any other fancy tools. Consider the following text:

> "The management of the ridesharing company *Uber* wants to create a database system that allows them to store and handle data for two services: ridesharing (*Uber Ride*) and food delivery (*Uber Eats*). For the former, they want to store information on drivers, passengers, trips (from source to destination). For the latter, they want to store information about restaurants, menus, customers (who can be passengers in the ridesharing service), drivers (who can be the same drivers from the ridesharing service), and food orders. You are asked to help design the database."

A. Think about the different entities in this system, how they relate to each other, and what attributes you may want to store on each. Create an Entity-Relationship Diagram (ERD) to represent all of this.
B. Create a relational database schema. Try to make sure that your design satisfies the first three normal forms

Note: There's no one right answer. The design depends on what they want to store data on, so you will need to make assumptions about their requirement. Please be clear about your assumptions and explain your logic.

Problem 3 [25 marks]

In this problem, you will use the files *US_codes.txt* (data on US Zip Codes) and *US_population.csv* (data on population numbers per gender, age group, and zip code). Answer the following questions below by using the command line (SQLite environment). Copy the commands to a text/Word file, and take screenshots and include in your PDF.

A. Create a new database in SQLite named *P3.db*

B. Have a look at the two data files *US_codes.txt* and *US_population.csv*. Create two tables *US_Code* and *US_Pop* with column headings that match these two data frames. The former does not have headings. Use the following headings:

‘CountryCode','ZipCode','City','StateFull','State2','CountyFull','FIPSCountyCode',
'MunicipalityFull','MunicipalityCode', ‘Latitude’, ' Longitude', 'Accuracy'

The latter has headings, but you need to use the following headings:

’Geo_ID’,’Zip’,’Gender’,’AgeRange’,‘Population’

Make sure you choose appropriate types and constraints for columns, and appropriate primary keys and foreign keys, if any. Use ‘ZipCode’ as a Primary Key for *US_Code.* For *US_Pop*, add a column called ‘ID’ which would be the Primary Key of that table. ‘ID’ should have an auto-increment feature (it starts with 1 and increases automatically for each record).

C. Insert the data from the two files into the two tables. Make sure you don’t insert the column heading from the file *US_population.csv*. Explain what you did about that

D. Write an SQL query to print the total population per gender (use the *US_Pop* table only)

E. Write an SQL query to print the total population per gender, but join the two tables. Explain why you see different numbers from part D

F. Write an SQL query to print the total population per age group (use the *US_Pop* table only)

G. Write an SQL query to print the Top 10 largest states (full name) in terms of population size

H. Write an SQL query to print the number of existing counties (not countries) in the data base

J. Write an SQL query to print the total population per gender and age group for the county named “Middlesex”.

Problem 4 [25 marks]

In this problem, you will use the JSON file *tvshows.json*. This file contains data about TV shows; one document/row for each show.

Answer the following questions below by using the command line (mongo environment). Copy the commands to a text/Word file, and take screenshots and include in your PDF. For queries that print full documents (C, D, and H), use *pretty()* at the end for a nicer presentation.

A. Import the *tvshows.json* file in your local MongoDB under the database name *TV_Shows* with a collection name *shows*

B. Write a query that counts the number of documents in the *shows* collection

C. Write a query that prints all information of the show named *"How I Met Your Mother"*

D. Write a query that prints shows that are stored in the collection in the order 27th -28th

E. Write a query that counts the number of shows that were still running at the time the data was collected

F. Write a query that counts the number of shows that have an indicated web channel.

G. Write a query that counts the number of “Comedy” shows in the *shows* collection

H. Write a query that prints the one show that has the following three genres: 'Drama', 'Thriller', and 'Comedy'

J. Write a query that prints only the name and genres of shows that have (at least) both of the following two genres: 'Drama' and 'Comedy'

K. Write a query that prints the average rating of ‘Comedy’ shows

L. Write a query that prints a unique list of all genres

Problem 5 [15 marks]

For this problem, use the same data from Problem 4 to perform compelling extra analysis and produce a plot in Python (*Matplotlib*). There are several other features in this data set, you will get marks if you find a compelling and interesting visualisation of these (one plot is enough, but you may produce as many as you want as long as they are all tied into one main idea). Make sure you provide textual description or analysis of the plot. You need to use the Python *Matplotlib* library for plotting, but you can also use other Python libraries (*PyMongo* and *Pandas*) to perform the queries and analysis (rather than mongo environment) for this problem. Include the resulting .png figure in your PDF, and include your code as a screenshot and as a textual script in the PDF. Include the Jupyter Notebook file as well.