COM6012 Assignment 1 - Deadline: 5:00PM, Wed 06 March 2019

Assignment Brief

How and what to submit

A. Create a .zip file containing the following:

1) **AS1_report.pdf**: A report in PDF containing answers to all questions. The report should be concise. You may include appendices/references to make it self-contained.
2) **Code, script, and output files:** All files used to generate the answers for individual questions above, **except the data**. These files should be named properly starting with the question number: e.g., your python code as **Q1**_xxx**.py** (**one each question**), your script for HPC as **Q1**_HPC.sh, and your output files on HPC such as **Q1**_output.txt or **Q1**_figB.jpg. Alternatively, now with the support of **Jupyter Hub** (Lab 1 1.1b), you can also develop your answers in **Jupyter Notebook** and submit the as **AS1_xxx.ipynb** and anything else needed to reproduce the results in your report, with Question numbers clearly labeled (e.g., in **bold** or bigger font) in the notebook (note the results should be from HPC, not your local machine for verification purposes).

B. Upload your .zip file to MOLE before the deadline above. Name your .zip file as **USERNAME_STUDENTID_AS1.zip**, where USERNAME is your username such as **abc18de**, and STUDENTID is your student ID such as 18xxxxxxx.

C. **NO DATA UPLOAD**: Please do not upload the data files used. We have a copy already. Instead, please use **relative file path in your code (data files under folder 'Data')**, as in the lab notebook so that we can run your code smoothly.

D. **Code and output**. 1) Use **PySpark** as covered in the lecture and lab sessions to complete the tasks; 2) **Submit your PySpark job to HPC** with **qsub** to obtain the output.

Assessment Criteria (Scope: Session 1-4; Total marks: 20)

1. Being able to use PySpark to analyse big data to answer questions.
2. Being able to perform log mining tasks on large log files.
3. Being able to perform movie recommendation with scalable collaborative filtering.
4. Being able to use scalable k-means to analyse big data.

Late submissions: We follow Department's guidelines about late submissions, i.e., a deduction of 5% of the mark each working day the work is late after the deadline, but **NO late submission will be marked one week after the deadline** because we will release a solution by then. Please see this link.

Use of unfair means: "*Any form of unfair means is treated as a serious academic offence and action may be taken under the Discipline Regulations.*" (from the MSc Handbook). Please carefully read this link on what constitutes Unfair Means if not sure.

Question 1. Log Mining [10 marks]

Please finish critical & essential tasks in Lab 1 and Lab 2 before solving this question.

Data: the NASA access log July 1995 data (click to download).

Please read the dataset description to understand the data and complete the following four tasks. A PDF of this description is also uploaded under Assignment 1.

A. Find out the average number of requests on each of the seven days in a week (i.e., Monday, Tuesday, Wednesday, Thursday, Friday, Saturday, and Sunday) during July 1995. You need to report **SEVEN** numbers, one for each day of the week. **Hint**: see pyspark sql API related to data format. [3 marks]

B. Visualise the results in A above as a figure (e.g., bar graph or pie chart) and discuss your **observations** (e.g., any trend, contrast, something expected, unexpected or interesting) using 1 to 3 sentences. To plot on HPC, you need to activate your environment and install matplotlib via **conda install -c conda-forge matplotlib** [2 marks]

C. Find out the top 20 most requested **.gif** images. Report the file name and number of requests for each of these 20 images. [3 marks]

D. Visualise the results in C above as a figure (e.g., bar graph or pie chart) and discuss your observations (e.g., anything interesting) using 1 to 3 sentences. [2 marks]

Question 2. Movie Recommendation [10 marks]

Please finish essential tasks in Lab 3 before solving A/ B, and in Lab 4 before solving C/D.

Data: the MovieLens 20M Dataset (click to go to the download page).

Please read the dataset description to understand the data & complete the following tasks.

A. Perform a **five-fold cross validation** of ALS-based recommendation on the rating data **ratings.csv**. Study two versions of ALS: one with the ALS setting in Lab 3 notebook with "drop" as the coldStartStartegy, and another different setting decided by you. For each of the five splits, report the Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) for the two versions of ALS. Then compute and report the mean and standard deviation (std) of RMSE and MAE. Put all results (these numbers for RMSEs and MAEs **in a Table in the report**. **Hint**: check out the RegressionEvaluator API. [4 marks]

B. Briefly discuss your observations on results in A with 3 to 5 sentences. [1 mark]

C. After ALS, each movie is modelled with some factors. Use *k*-means with ***k=20*** to cluster the movie factors (**hint**: see **itemFactors** in ALS API, id=movieid in this problem) learned with the ALS setting in Lab 3 notebook in A for each of the five splits. For each of the five split, use **genome-scores.csv** to find the top five tags for **each of the top three largest clusters (i.e., 15 tags in total for each split)** and report the names of these top tags using **genome-tags.csv**. **Hint**: For each cluster, sum up tag scores for all movies in it; find the largest five scores and their indexes; go to genome-tags to find their names. [4 marks]

D. Briefly discuss your observations on results in C with 3 to 5 sentences. [1 mark]

Appendix: Clarifications collected from MOLE.

All points have been summarised above. These are kept here for reference.

Q1A. We expect **seven numbers**, one for each day of the week.

Q1B/D: To plot on HPC, you need to activate your environment and install matplotlib via **conda install -c conda-forge matplotlib**

Q2: The **default ALS setting in earlier version** refers to the **ALS setting in the Lab 3 notebook** that uses coldStartStartegy= '**drop**' (The API default is 'NaN'). Please use drop.

Q2A. We expect a table in the report (not necessarily in the code output) based on five fold cross validation (dividing the data into five folds; split x with fold x as test set and the remaining four folds as training set, x=1,...,5) :

Split 1: RSME for ALS1, RSME for ALS2, MAE for ALS1, MAE for ALS2

Split 2: the same above

Split 3: the same above

Split 4: the same above

Split 5: the same above

Average: from the five splits above, compute their mean & std of RSME for ALS1, mean & std of RSME for ALS2, mean & std of MAE for ALS1, mean & std of MAE for ALS2

Q2C: We expect for the largest three clusters

Cluster #1: top five tags for all movies in this cluster

Cluster #2: the same as above

Cluster #3: the same as above

For each cluster: sum up tag scores for all movies in; find the largest five scores and their indexes; go to genome-tags to find their names.