Assignment 4: Frequent Itemsets, Advertising and Recommendation Systems

Formative, Weight (15%), Learning objectives $(1, 2, 3)$,
Abstraction (4), Design (4), Communication (4), Data (5), Programming (5)

Due date: $11:59$ **pm,** 15 **April,** 2022

1 Overview

This assignment must be done **individually**. This means all the rules regarding individual submission will apply and the submission must be solely your own work. Therefore, we will not use the groups on MyUni. You will need to submit on the assignment page as an individual.

2 Assignment

Exercise 1 Collaborative Filtering (Exercise 9.3.1) (30 points)

	a	**b**	**c**	**d**	**e**	**f**	**g**	**h**
A	4	5		5	1		3	2
B		3	4	3	1	2	1	
C	2		1	3		4	5	3

Table 1: Utility matrix for exercise 1.

Table 1 is a utility matrix, representing the ratings, on a 1–5 star scale, of eight items, a through h, by three users A, B, and C. Compute the following from the data of this matrix:

1. Treating the utility matrix as boolean, compute the Jaccard distance between each pair of users.

2. Repeat Part (1), but use the cosine distance this time.

3. Treat ratings of 3, 4, and 5 as 1 and 1, 2, and blank as 0. Compute the Jaccard distance between each pair of users.

4. Repeat Part (3), but use the cosine distance this time.

5. Normalize the matrix by subtracting from each non-blank entry the average value for its user.

6. Using the normalized matrix from Part (5), compute the cosine distance between each pair of users.

Exercise 2 Frequent Itemsets (50 points)

For this exercise, you will need to first study the Section 6.4 (up to 6.4.3) of the text-book, Mining Massive Datasets, Leskovec, Rajaraman, Ullman (third edition, 2020), and then follow the instructions below:

1. Implement the simple, randomized algorithm given in Section 6.4.1 of the text-book.

2. Implement the algorithm of Savasere, Omiecinski, and Navathe (SON algorithm), as explained in Section 6.4.3.

3. Compare the two algorithms implemented above on ALL the following datasets: T10I4D100K, T40I10D100K, chess, connect, mushroom, pumsb, pumsb star; those datasets are available at:

   http://fimi.ua.ac.be/data/

   Report the observed outcomes and comparisons on individual datasets and in overall.

4. Perform different experiments on the simple randomized algorithm, using the following sample sizes: 1%, 2%, 5% and 10%. Compare your experimental results. Additionally compare the results of the above experiments with the results produced by the SON algorithm.

   Your approach should be as efficient as possible in terms of runtime and memory requirements.
   Report on challenges that you might have come across during the implementation and running the experiments.

Exercise 3 Advertising (Exercise 8.4.1 and more) (10+5+5 points)

Part 1 Explain both the *Greedy Algorithm* (Section 8.2.2 of the textbook) and *Balance Algorithm* (Section 8.4.4 of the textbook) and explain what *Competitive Ratio* is.

Part 2 Consider Example 8.7. Suppose that there are three advertisers A, B, and C. There are three queries x, y, and z. Each advertiser has a budget of 2.

Advertiser A only bids on x, B bids on x and y, and C bids on x, y, and z. Note that on the query sequence $xxyyzz$, the optimal offline algorithm would yield a revenue of 6, since all queries can be assigned.

1. Show that the greedy algorithm will assign at least 4 of the 6 queries $xxyyzz$.
2. Find another sequence of queries such that the greedy algorithm can assign as few as half the queries that the optimal offline algorithm would assign to that sequence.

3 General assignment submission guidelines

Your submissions will need to include the following, at minimum:

- a PDF file of your solutions for any theoretical exercises. The solutions should contain a detailed description of how to obtain the results, not just the final results.
- PDF or txt file with a brief descriptions of your implementations to understand your code.
- Files containing the output results of running your algorithms on the provided datasets.
- PDF or txt file of your computation times of the algorithms on the provided datasets.
- All source files, logs, and all the project files.
- a README.txt file containing instructions to run the code, student ID, and email address.
- the submissions that do not follow the above guidlines may lose points accordingly.

Please do not hesitate to reach out using the discussion forum, workshops, or the contact details of the teaching assistants on the home page of MyUni, should you have any questions or concerns.