Practical Data Science (with Python)

COSC2670/COSC2738

Assignment 3

Semester 2, 2024

Type	Individual assignment
Due Date	Sunday, 27 October 2024 by 23:59
Weighting	30%
Marking Criteria	See the course Canvas shell >> Assignments
Mark/Feedback	Will be available in Canvas

1. Introduction

This assignment focuses on *recommender systems*, a data science application widely used in the real world. You will need to develop and implement appropriate solutions to complete the corresponding tasks, and present the results (virtually).

2. General Requirements

Please read the following requirements and information carefully before you start.

- You must do the assignment in Jupyter Notebook.
- You must choose (by yourself) and use the *appropriate* Python functions (or methods) to complete the tasks.
- You must provide *sufficient (or detailed), precise* comments and/or annotations in your code, explaining what each (key) part does.
- You must use the provided template A3code-Template.ipynb to organise your code.
- Ensure that your submission follows the file naming rules specified in this document. Replace YourStudentNumber with your Student Number (e.g. s1234567) wherever applicable.
- Responsible use of AI tools is allowed. You must appropriately acknowledge and reference the use of any AI tools and their outputs. Failure to reference the use of these tools can result in academic misconduct. For more instructions, see: https://rmit.libguides.com/referencing_AI_tools
- Plagiarism is *never* okay. Relevant tools (like Turnitin) might be used for plagiarism check for this assignment.
- There are rules in place to support you uphold the academic integrity of RMIT. See: https://www.rmit.edu.au/students/my-course/assessment-results/academic-integrity

3. Tasks

This assignment deals with movie recommendation. The dataset to be used throughout the assignment is the MovieLens 1M Dataset. This dataset ml-1m.zip is downloadable in Canvas under the Assignments >> Assignment 3 section of the course Canvas shell. The README.txt file

therein provides details about the dataset. Note that this MovieLens 1M dataset is different from what is used in course labs (which is MovieLens 100K Dataset). Besides, different users may have rated different groups of movies.

Task 1: kNN-based Collaborative Filtering

In this task, you need to implement and evaluate **user-based** (i.e., user-user) collaborative filtering that uses kNN (k-nearest neighbour), i.e., kNN-based Collaborative Filtering. Randomly choose one user (as **test set/data**) and predict this user's ratings on **all** movies. Note that in the given dataset, the user might have only rated some of the movies.

Specific tasks include:

- Use RMSE (root-mean-square error) as the metric for evaluation.
- Study the impact of the parameter *k* (of kNN), with at least 5 different values. Choose the optimal value for *k*.
- Compare the performance of at least 2 similarity metrics, and choose the (most) appropriate one.
- Summarise the results and findings concisely in slides (and presentation).

Task 2: Matrix Factorization-based Recommendation

In this task, you need to explore, implement, and evaluate a recommender system that uses a matrix factorization technique **of your choice**. Examples of matrix factorization techniques include, but not limited to, Singular Value Decomposition (SVD), Non-negative Matrix Factorization (NMF), Alternating Least Squares (ALS), and their variants.

Randomly choose 5 items (i.e., movies; as **test set**) and predict **all** users' ratings on these movies. Note that in the given dataset, it is possible that some users didn't rate a chosen movie.

Specific tasks include:

- Use RMSE as the metric for evaluation.
- Implement matrix factorization-based recommendation (by using appropriate Python functions/methods) and evaluate its performance.
- Propose an appropriate approach to improve the performance of the original/above matrix factorization-based recommendation. Validate its effectiveness (via comparison).
- In slides (and presentation), describe the ideas of both the matrix factorization technique you choose and the approach your propose for performance improvement (in your own words) clearly and precisely; explain why the proposed approach can deliver better performance in this case. Cite references wherever necessary.
- Summarise the results and findings concisely in slides (and presentation).

Task 3: Ranking-based Evaluation and Comparison

In this task, you will conduct ranking-based evaluation and comparison of the two solutions implemented in Task 1 and Task 2 respectively. Randomly choose 10 users (from who have rated more than 100 movies each; as **test set**) and recommend Top-20 movies to each of them. Use **AP** (Average Precision) and **NDCG** (Normalized Discounted Cumulative Gain) as evaluation metrics. Compare two solutions:

- kNN-based Collaborative Filtering: The solution developed in Task 1. Use the optimal parameters (based on results of Task 1). Name it as KNNCF wherever needed.
- Improved Matrix Factorization-based Recommendation: The solution developed in Task 2. Use the approach proposed in Task 2 which leads to improved performance. Name it as IMFR wherever needed.

Specific tasks include:

- Use an appropriate graph/chart to visualise the results of each metric, respectively (i.e., one graph for AP and another for NDCG).
- In slides (and presentation), summarise the results and findings concisely. Discuss the limitations of KNNCF in this case and how to improve its performance. Explain why IMFR can deliver better performance.

It is required that IMFR must achieve **better** (overall) performance; if this is not the case, your whole Task 3 will be rated "Poor" (as per the rubric).

Task 4: Presentation

In this assignment, you need to design slides and record a presentation (for the above tasks). Your slides and presentation could include, e.g., but not limited to:

- a cover page/slide containing e.g. your name and student ID, Assignment info, etc.
- a concise outline, key results, and findings of Task 1,
- description and explanation of approaches used in Task 2 (with proper citations),
- a concise outline, key results, and findings of Task 2,
- a concise outline, key results, visulisation, and findings of Task 3,
- discussions and explanation of Task 3,
- a list of references.

The following requirements must be strictly met; otherwise, your submission will be considered invalid.

- The slides must be no more than 10 pages in total (i.e., no more than 10 slides).
- The presentation (recording) must be no more than 5 minutes.
- The recording video file must be in MP4 format.
- The recording video file must be less than 50MB in size.

There is no template for slides. Code is not needed in slides. Key results and findings must be included in slides. Use any referencing style recommended by RMIT Library (https://www.lib.rmit.edu.au/easy-cite/).

You may use any (appropriate) tool to record your presentation. Some tips for making a presentation recording video:

- Record a presentation with Microsoft PowerPoint (Record - From Beginning): https://support.microsoft.com/en-us/office/record-a-slide-show-with-narration-and-slide-timings-0b9502c6-5f6c-40ae-b1e7-e47d8741161c
- Save a presentation as a video (Record - Export to Video, Customize Export; or File - Export - Create a Video): https://support.microsoft.com/en-us/office/turn-your-presentation-into-a-video-c140551f-cb37-4818-b5d4-3e30815c3e83
- Select the quality of the video as Standard (480p) (to minimise the file size). Note that video resolution won't affect marking. HD (720p) is acceptable only when the file size doesn't exceed the limit.

4. Submission

The following files must be submitted (uploaded one by one, in one submission) in Canvas:

- **The Jupyter Notebook file** named as YourStudentNumber-A3code.ipynb: containing your Python code (and comments).
    - For the Jupyter Notebook code, please make sure to clean them and remove any unnecessary lines of code (cells). Comments are required. Follow these steps before submission:
        - Main menu → Kernel → Restart & Run All

        - Wait till you see the output displayed correctly. You should see all the data outputs printed and graphs displayed.
    - Note that you might obtain different results after each run. It is okay if you avoid unnecessary re-sampling in every run by e.g. disabling some code.
- **Your clean slides file in PDF**, named as YourStudentNumber-A3Slides.pdf.
    - This PDF file should not contain any recordings (e.g. narration) therein.
- **Your presentation recording video file in MP4,** named as: YourStudentNumber-A3Presentation.mp4.

Resubmission without penalty is allowed until the deadline. Please do NOT submit any unnecessary files. Invalid/incomplete submissions will receive 0 marks.