Web Science Assignment 1: Movie Recommendation
Due: 26-02-2018, 23:55

1 Movie Recommendation

Netflix is the world's leading content provider for online movies. With a library containing more than 20.000 movies, it becomes important to recommend to users, the movies that they will most likely want to watch. In this project, you will be implementing a simple movie recommender system that might help companies such as Netflix.

1.1 Academic Code of Conduct

You are welcome to discuss the project with other students. However, any sharing of code and/or text is *not* permitted, and each person must submit their own project. Plagiarism tools will be used in your submissions. **If you have questions regarding the project, post them on the discussion forum.**

1.2 Preliminaries

You may use any programming language, any libraries such as NumPy or Apache IO and external programs, etc., to solve this project. Make sure that we can run your code with these dependencies when you submit.

2 Project Task

You are given a small version of the MovieLens data. Your task is to implement and evaluate an item-to-item collaborative filtering algorithm that can predict user ratings to unseen movies. This section provides details on how to do this.

To help you in solving this project, I *strongly recommend* you keep "Collaborative Filtering Recommender Systems" by Ekstrand, Riedl and Konstan at hand. You can find it here: http://files.grouplens.org/papers/FnT%20CF%20Recsys%20Survey.pdf.

2.1 Data

Download the *small* version of the dataset from https://grouplens.org/datasets/movielens/latest/. The dataset consists of 100,000 ratings from 700 users on 9,000 movies. Read the README so you understand the format.

2.2 Calculating similarity

Given two items (or movies) i and j compute the similarity using adjusted cosine similarity:

$$\text{sim}(i,j) = \frac{\sum_{u \in U} (R_{u,i} - \overline{R}_u)(R_{u,j} - \overline{R}_u)}{\sqrt{\sum_{u \in U} (R_{u,i} - \overline{R}_u)^2} \sqrt{\sum_{u \in U} (R_{u,j} - \overline{R}_u)^2}}$$

where R(u) is the average of the u'th users ratings, $R_{u,i}$ and $R_{u,j}$ are the ratings given by user u to items i and j, U is the set of users which have rated both item i and j.

You can then compute the prediction of an item i for user u as follows:

$$P_{u,i} = \frac{\sum_{n \in N} (R_{u,n} \cdot \text{sim}(i,n))}{\sum_{n \in N} |\text{sim}(i,n)|}$$

where the summations are over all other rated items n ∈ N for user u, sim (i, j) is the similarity between items i and n, $R_{u,n}$ is the rating for user u on item n and $|\cdot|$ in the denominator denotes the absolute value.

2.3 Evaluation

Evaluate your recommender system using x-fold cross validation, and using MAE and RMSE as the evaluation measures.
MAE is the Mean Absolute Error; it computes the deviation between computed ratings and actual ratings.

$$MAE = \frac{1}{n} \sum_{i=1}^{n} |p_i - r_i|$$

where n is the total number of ratings over all users, p are the predicted ratings, and r are the actual ratings.
RMSE is the Root Mean Square Error; it is similar to MAE but places emphasis on larger deviations.

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (p_i - r_i)^2}$$

where the notation is the same as for MAE.

X-fold cross validation requires that you split the dataset randomly into x folds, and use some of them for training and some of them for testing, as follows:

1. Split your dataset into x equal-sized disjunct fractions of known ratings *randomly*, i.e. you do not have to split your dataset in a user-stratified way.
2. Treat x-1 of them as training folds (i.e. use them to build and optimise the method), and the remaining single fold as the test fold (i.e. treat it as ground truth to evaluate the model's quality measured separately in MAE and RMSE)
3. Iterate point 2 above by choosing a different fold as test fold each time, until all 10 folds have been used as test folds exactly once
4. Report the mean MAE and RMSE of your methods across the x test folds.

Set x=10.

3 Project Questions

You must answer all of the following questions. If there is anything in the project that is not clear, be sure to (i) state what was unclear in your report and (ii) how you addressed it. Make sure you substantiate your writing with results from the recommender system, whenever possible.

Question 1: Repeat the x-fold validation by setting x=3. How does your recommendation system's performance vary as you change x from 10 to 3? Explain this variation in performance.

Question 2: Are the observed differences between MAE and RMSE for x=10 and for x=3 statistically significant, or are they due to chance? To answer this, perform pairwise analysis of variance (ANOVA) between x=10 and x=3, first for MAE and then for RMSE. The Null Hypothesis H0 is that the observed differences are due to chance. If the outcome of the statistical test rejects H0, then the differences are statistically significant. Report the level of significance.

Question 3: Whenever a new item enters the database on top of which the recommender system runs, it has no ratings and therefore it is difficult to recommend the product. Discuss in your own words how a recommender system could alleviate this problem. Insert 3 new items (that you make up) to the dataset with no ratings and implement a solution that makes recommendations for these 3 items.

Question 4: MovieLens has datasets that are much larger than the one used here. Does your solution scale to large datasets? Why or why not? Discuss.

4 Submitting your project

You should focus on answering all questions in the project. In the event you are unable to do so, you must detail how you would have solved the remainder of the project have you been given more time.

4.1 What to hand in

You **must submit a single tar.gz or zip file** that contains:

1. Your report in pdf *including* the latex sources or original Word document

2. The source code that you developed including any appendices describing how to run your code for each subpart of the assignment.

Do not include the datasets or any subset of this. Any shortcomings in your report such as results reported using an arbitrary subset of the data, or choices that are not justified will affect your grade.

4.2 How to hand in

The tar.gz or zip file must be uploaded to Absalon before the deadline. **The name of your file must be your KU username and the project identifier P1**. For example, your submission might look like abc123-P1.zip or pkn877- P1.tar.gz. No late submissions are accepted and will count as a used attempt at completing the exam. If Absalon is down, send your submission to your TA. Extensions for submissions can only be given on the grounds of exceptional reasons (e.g. medical reasons, with proof) and only by the course responsible.