Practical Machine Learning and Deep Learning - Assignment 2 - Movie Recommender System

Author: Grigorii Fil (BS21-DS02) g.fil@innopolis.university

Introduction

Recommender system is a system that can recommend some items (eg. movies) for some users. Recommender systems can be used in various scenarios, for example, on online movie platforms to recommend movies that are most probably to be liked by users. The number of movies on such platforms can grow large, so users can benefit from good recommender systems. In this assignment, I will try to develop the recommender system using Machine Learning techniques.

Data Analysis

The dataset (MovieLens 100K dataset) was provided with the task description. It consists of data about 943 users and 1682 movies, with their interactions (rating that users gave to the movies). Though the dataset contains demographic information about users (such as gender, age, and occupation) and some information about movies (such as genres), I am going to utilize only the rating information. The rating values are integers from 1 to 5 (Figure 1). I interpret them in such a way that if the rating value is greater than 3, then the user liked the movie. Based on that, I made a matrix with rows corresponding to users and columns corresponding to movies. The values of this matrix are binary: 1 if the user liked the movie, 0 - otherwise. I have performed a train/test split of this dataset with ratio 0,2/0,8.

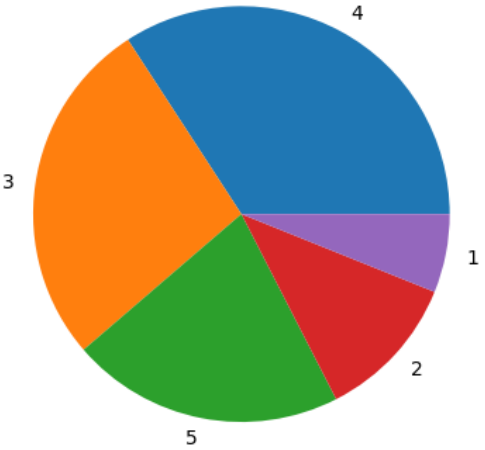

Figure 1. Ratings distribution.

Model Implementation

For this task, I will use the Alternating Least Squares (ALS) model. This model was described in this paper. Its main goal is to find latent factors for users and items, that is, a vector representation of users and items. This is done by factorizing a matrix of user preferences (such as described in the previous section). The user preference to the given item can be found by inner product of user factors and item factors. For a given user, we can calculate preferences for each item and then take the items with the highest preferences scores as a recommendation.

Model Advantages and Disadvantages

As stated here, there are some advantages and disadvantages of the ALS model.

Advantages:

- Simplicity: It is very easy to explain and to understand
- Applicability: There are hardly any applications where least squares doesn't make sense
- Theoretical Underpinning: It is the maximum-likelihood solution and, if the Gauss-Markov conditions apply, the best linear unbiased estimator

Disadvantages:

- Sensitivity to outliers
- Test statistics might be unreliable when the data is not normally distributed (but with many datapoints that problem gets mitigated)
- Tendency to overfit data (LASSO or Ridge Regression might be advantageous)

Training Process

For the training, I have used the implicit library, specifically, its implementation of the Alternating Least Squares model. The model has several hyperparameters. The number of iterations was set to 200. I have trained several models with different numbers of factors and regularization parameters. Specifically, the number of factors were chosen from 50, 100, 200, 300, and the regularization parameter was chosen from 0.01, 0.05, and 0.1. So, 12 models were trained in total.

Evaluation

For evaluation, I have used two metrics - precision@K and AUC_score@K. The K parameter was chosen to be 10, so the metrics are precision@10 and AUC_score@10. Their implementation was also provided by the implicit library.

Results

The results of evaluation can be seen in the table below. The model with 50 factors and 0.01 regularization parameter has the best scores.

Number of factors	Regularization parameter	AUC_score@10	precision@10
50	0.01	**0.600991**	**0.252167**
50	0.05	0.600667	0.250591
50	0.1	0.600927	0.250749
100	0.01	0.582258	0.196217
100	0.05	0.582636	0.194799
100	0.1	0.580806	0.192750
200	0.01	0.543909	0.104807
200	0.05	0.543765	0.107171
200	0.1	0.543991	0.105910
300	0.01	0.525364	0.062411
300	0.05	0.525682	0.062727
300	0.1	0.526565	0.062727