South China University of Technology

The Experiment Report of Machine Learning

SCHOOL: SCHOOL OF SOFTWARE ENGINEERING

SUBJECT: SOFTWARE ENGINEERING

Author:

Ziwei Zhang, Haixu Liu and Jintao Xuan

Supervisor:

Mingkui Tan

Student ID：

201530613719, 201530612347 and 201530031377

Grade:

Undergraduate

December 28, 2017

Experiments with Matrix Factorization

Abstract— We used Matrix Factorization to build a Collaborative Filtering recommender system. We choose Stochastic Gradient Descent as optimizer to set the model parameters. In this report, we record experiments we carried out to implement MF and some study on MF.

I. Introduction

Matrix Factorization (MF) is an unsupervised learning method. It allows us to perform matrix decomposition to uncover the latent features underlying the interactions between users and items.

Unlike memory-based Collaborative Filtering (CF) algorithm, MF doesn't use the whole dataset to make prediction. It first creates a model from the information in the recommender system database. Then uses this generated model to do recommendations. In this way, it reduces the problem from a high level of sparsity to a smaller space, which containing some more useful information.

In our experiments, first implement the MF algorithm and use Stochastic Gradient Descent to set the model parameters. Then we add in bias term, taking the user bias into consideration. Finally, we compare the performance of our MF algorithm with Singular Value Decomposition (SVD).

II. Methods and Theory

A. The MF algorithm

We use two low-rank matrix P and Q to represent the latent preferences of users and the latent attributes of items from known ratings. The goal is to learn the value of P and Q so that combine them together can achieve the approximation of the original rating matrix:

$$R \approx P^T Q$$

Then we define the distance between the low-rank approximation and the original rating matrix using RMSE:

$$L = \sum_{u,i \in S} (R_{ui} - P_u^T Q_i)^2 + \beta \left(\sum_u \|P_u\|^2 + \sum_i \|Q_i\|^2 \right)$$

Then we use Gradient Descent to optimize the P and Q matrix simultaneously.

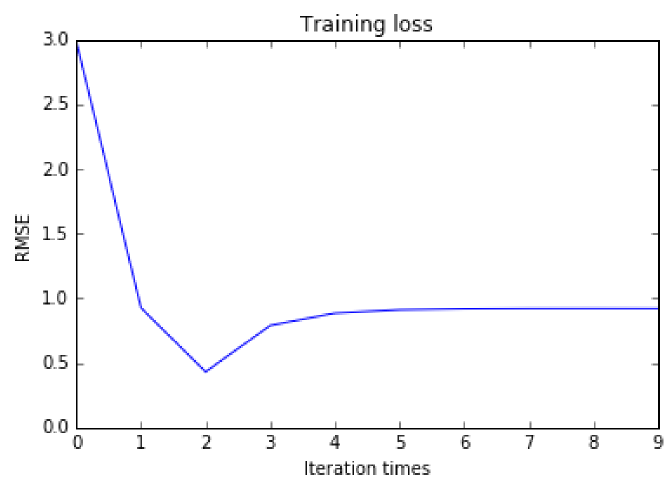
III. Experiment

A. Dataset

The experiment is based on MovieLens-100k dataset. The dataset consists 10,000 comments from 943 users out of 1682 movies. At least, each user comment 20 videos. Users and movies are numbered consecutively from number 1 respectively. We separated from original dataset with proportion of 80% and 20%.

B. Movie recommendation task

The training loss is given in Figure 1. After we optimize the model on the training set, we got a test error (RMSE loss) about 0.968.



IV. Conclusion

We have implemented Matrix Factorization and used it to build a recommender system. We optimized it using Stochastic Gradient Descent and it gave back a reasonable result.