Assignment 5: Collaborative Filtering

Apala Guha

Spring 2017

Provided Code and Data In this assignment, we will use the MovieLens dataset which contains 100K movie.

1 Introduction

In order to train and test your models, you are provided with train and test splits extracted from the 100K movie ratings of the MovieLens dataset. These splits can be found in (`MovieLens100K_train.txt, MovieLens100K_test.txt`). The dataset is organized as follows: `"UserID \t MovieID \t Rating \t Timestamp"` where UserIDs range between 1 and 943, MovieIDs range between 1 and 1682, and, Ratings are based on a 5-star scale (whole-star ratings only). You can ignore the timestamp data for this assignment. Movie names are in the file `u.item`.

2 Matrix Factorization-based Collaborative Filtering

Using the `PySpark ML` API train a ALS recommendation model. Use the values (2, 4, 8, ..., 256) as matrix rank and maximum iterations of 20. Train the model using 5-fold cross validation. Plot the RMSE error for different matrix ranks on the test set.

Also, extract the movie factors from the ALS models. Train a K-means model of 50 clusters on these factors. For every rank size, list the movie names contained in any two clusters.

3 Item-item collaborative Filtering

Use the training dataset to compute the average rating of each user and the deviation from the average rating for each user item pair. Also compute the item-item correlation. Remove the item-item pairs with correlation below a certain threshold. Use thresholds of (0.5, 0.6, 0.7, 0.8, 0.9).

Use the data computed on the training set to evaluate predictions on the test set and plot the RMSE error for different similarity cutoff thresholds. Note that some item-item correlations will be missing because they do not have any common users. Ensure that the weight for such cases is zero.