IDS 572 Assignment 5 – Recommender system

Due: Monday, April 11th

The MovieLens data shows user ratings for movies, collected over a period of time. The dataset also includes additional movie and user attributes (grouplens.org/datasets/movielens). The data has been widely used for examining recommender systems. The ratings data is of the form:

userId	itemId	rating	time (unix secs since 1/1/1970)
196	242	3	881250949
186	302	3	891717742
22	377	1	878887116

In addition, movie attributes available are: movieTitle, releaseDate, imdbURL, genre, and user demographics are: age, gender, occupation, zip code. The movie attributes are in the file u_item.csv and the user attributes are in the file u_user.csv.

For this assignment we will use the MovieLens dataset of 100,000 ratings arising from 943 users and 1682 movies; the data been cleaned to include users who have rated at least 20 movies. We will not use any movie or user information here.

This assignment develops and evaluates recommender systems based on the MovieLens data. RapidMiner has a Recommendation Systems extension that eases our task.

We will not be using the 'time' attribute for this assignment. The 'rating' attribute should be set the 'label' role (use the Set Role operator). We will also need to set user-specified role of 'user identification' for the 'userId' attribute and 'item identification' for the itemId attribute.

1. (a)Explore the data to obtain an understanding of users, movies and how users have rated movies.
- what is the overall distribution of ratings
- on average, how do users rate movies; what ratings do movies have on average ? (you may want to plot the distribution of average ratings for users, movie. Can you show this on a single plot?)
- how many movies do users rate, and how many ratings do movies get? (consider the distribution of rating counts)
- how are rating levels distributed, do many people have high/low ratings?

(b) Consider the movie attributes in the file u_item.csv and the user attributes in the file u_user.csv. How do ratings differ by genre, by user age (group) , gender and occupation? You can analyze this in various ways – please describe what you do and any interesting findings.

2. Consider collaborative filtering based rating prediction.
We will evaluate performance of different approaches for predicting ratings. What measures will you use for assessing performance (why)? And what relationships will you examine -- for example, error (or accuracy) at different levels of ratings; are errors distributed equally across movies, users? etc.

[Remember - in the different operators that you will experiment with below, the regularization parameters can help reduce overfit]

(a) Use the Global Average method and User-Item Baseline methods. Do you find any performance differences? Do parameter changes for the user-item baseline operator make any difference?

(b) Use the Matrix factorization operator. Explore performance with varying number of factors. Does learning rate make a difference to performance?

(c) Use the User-knn and Item-knn operators. Explore performance with varying the number of nearest neighbors k? Also do you notice any differences between using the cosine similarity measure and the Pearson measure? Are the neighborhood sizes, k, that give good performance, comparable across the two operators (why?)?

Comparing performance across the different operators, which would you prefer to use (why)?

3. Consider the decision support objective of recommending movies to users. Movies predicted to receive high ratings will be recommended for a user. We then need to determine a cutoff rating for 'high' (for example, any rating >=4 is 'high'). To access performance for this, we can consider a confusion matrix and related measures like precision, sensitivity etc (or, how many predicted highs correspond to actual high, etc.). Using the predicted ratings for the test data, determi ne such decision support performance using the operators in Question 2.
Comparing performance across the different operators, which would you prefer to use (why)? What value of 'cutoff' will you use? Are errors distributed equally across movies and across users ?