
Time Influence on Ratings in the Netflix Prize Dataset

Timothy Ozimek

North Carolina State University
teozimek@ncsu.edu

1 Background

The online movie service, Netflix, gives a user the choice of watching amongst several thousand movie titles. While not all titles are available at all times, Netflix will rotate selections, and add new ones, during three month windows. One feature of Netflix is a recommendation system that suggests movies that suits a user's tastes. This feature works especially well after a user has recorded a rating after they watch a movie. The more ratings they score, the better the suggestion becomes.

One way of understanding the recommendation system is through everyday word-of-mouth conversation. Friends and acquaintances tend to share common opinions over a variety of topics. Likes and dislikes accordingly tend to coincide. The field of data mining and machine learning has observed this relationship and formalized the task as a recommender system (need citation). Such systems tend to employ the technique of collaborative filtering. In short, this method scours data, looking for commonalities in a request. Items deemed "close enough" to a request are filtered and used to predict. This filtered group may contain ratings not in the initial request. Based on these ratings an estimate is constructed for how a given request might rate that item. In this way, recommendation systems offer new knowledge.

2 Method

Model 1: Load data, give rating to user of average ratings for a movie.

Model 2: Employ user ratings filter via cosine metric. Average matching results.

Model 3: Explore with nearest neighbor setting.

Model 4: Combine model 1 and model 2. For cold start request, weight global rating more than filter, up to a threshold. If global rating has low variance, apply even more weight.

Model 5: Search for changing bias and use as filtering indicator. This requires time gets involved. Use cutoff to see if bias exists. Will require timestamps.

3 Experiment

4 Conclusion

References / Papers to Read

- [1] Linyuan Lu, Matus Medo, Chi Ho Yeung, Yi-Cheng Zhang, Zi-Ke Zhang, Tao Zhou. (2012) *Recommender systems*. In Physics Reports pp. 1-49, Elsevier B. V. 0370-1573.
- [2] Michael Hahsler (2017). recommenderlab: Lab for Developing and Testing Recommender Algorithms. R package version 0.2-2. <http://lyle.smu.edu/IDA/recommenderlab/>
- [3] Arkadiusz Paterek, (2007) *Improving regularized singular value decomposition for collaborative filtering*. KDDCup.07 pp. 39-42. ACM 978-1-59593-834-3/07/0008