STATS 507 Project Proposal:

MovieLens datasets—Predicting and analyzing user ratings of movies

TRONG DAT DO [1,*] and SIMON FONTAINE [1,**]

[1] *University of Michigan, Department of Statistics. West Hall, 1085 South University, Ann Arbor, MI, U.S.A., 48109. E-mail:* [*]dodat@umich.edu; [**]simfont@umich.edu

1. The MovieLens dataset

The MovieLens datasets (Harper and Konstan, 2015) contains user ratings of a variety of movies continuously collected starting from 1998. In addition to the user-movie-rating pairings, the datasets contains information about movie genres, word tagging of movies provided by users and user demographic information.

We will consider the MovieLens 100K Dataset[1], which is one of the multiple datasets provided by GroupLens[2]. We will be interested in this particular dataset because it contains additional demographic information about the users in the dataset. To include tagging information, we also consider the MovieLens Tag Genome Dataset[3]. Here is a summary of the contents of the datasets that will be used[4]:

MovieLens 100K Dataset The dataset was collected from the MovieLens website (movielens.umn.edu) between September 19th, 1997 through April 22nd, 1998. It has been preprocessed and cleaned to include only examples where the users have made at least 20 ratings during the collection period and where demographic information are complete. In the u.data file, there are 100,000 ratings on the scale of 1 to 5, taking only integer values. It contains the following entries: user id, item id, rating, timestamp. In the u.item file, there are 1.682 movies with the following information: movie id, movie title, release date, IMDb URL and 19 columns indicating movie genre with 0-1 encoding where 1 denotes that the movie is of the corresponding genre. In the u.user file, there are 943 users with the following information: user id, age, gender, occupation (see u.occupation file for details) and zip code.

MovieLens Tag Genome Dataset This dataset contains tagging information of 9.734 movies and 1.128 tags. In particular, the tag_relevance file contains the rele-

[1] Available at https://grouplens.org/datasets/movielens/100k/

[2] Organization website: https://grouplens.org/

[3] Available at https://grouplens.org/datasets/movielens/tag-genome/

[4] From the README.txt file attached to the datasets (http://files.grouplens.org/datasets/movielens/ml-100k-README.txt, http://files.grouplens.org/datasets/tag-genome/README.html)

vance of all tags for all movies reported on a continuous scale from 0 to 1, where 1 indicates strong relevance.

2. Research questions

2.1. Prediction modeling

Our first research question is to construct a predictive model for the user ratings using the available information. In particular, we wish to produce a model that is able to accurately predict the movie rating (for some movie already in the dataset) by a given user (also in the dataset). This model could then be part of a *recommendation system* where the predicted rating could be used as input to produce the recommendations.

2.2. Analysis

A secondary research question we are interested in is to analyze the effect of the available information on the user ratings. For example, we could look for genres and tags that are related to movies with better ratings. Then, we can perform more granular analyses using the demographic data: this could allow to extract correlations between population groups and movie interests. The insights recovered from such analyses could be relevant for decision-making such as identifying which movies to produce and which population groups to target with advertisement.

3. Methodology

We will investigate the first question using four different approaches and the best predictive model will be selected to make inference and answer our secondary research question. The four methods that we consider are k-Nearest Neighbors (k-NN), Neural Network, Matrix Completion and Restricted Boltzmann Machine.

Our first approach is using a k-NN algorithm. The idea is to define some distance measure between users as well as between movies, which allows to predict ratings using user-movie pairs that are similar to it. We will implement k-NN algorithm with different distances such as Euclidian distance and cosine dissimilarity. This is the most simple model that we consider, and it is pointed out in Töscher, Jahrer and Bell (2009) that kNN can be outperformed by simple factor models.

Secondly, we will use Neural Networks to predict the movie ratings. The input layer is information about each user and movie genres and tags and the output is the rating of the user for the movie. The architecture of the network (number of players, number of nodes and learning rate) will be adjusted to find a good model. Similar idea could be found in Tang et al. (2015), where they build the model with the input layers containing information about user-word instead of user-genre.

The third approach, Matrix Completion, became famous from the Netflix movie-rating challenge Bennett et al. (2007). This competition was held by Netflix, a movie-rental company, in effort to improve the recommendation system for their customers. The winner of this competition used many statistical techniques, where the Singular Value Decomposition (SVD) was the most important. The idea here is that our user-movie matrix is a missing-valued matrix and, by assuming it is low-rank, we can use SVD iteratively until our matrix converges to a completed matrix, which gives us the predicted rating of any movie by any user. Despite of being efficient, this method often overfit the data, so we will also consider a penalized version.

Finally, we will implement Restricted Boltzmann Machine (RBM) Salakhutdinov, Mnih and Hinton (2007). RBM is a probabilistic model, where we assume that there are hidden layers of variables affecting the visible users' ratings, and come up with the update rule to learn the distribution of these hidden variables. It is claimed by Salakhutdinov, Mnih and Hinton (2007) that RBM can outperform SVD models.

4. Preliminary and Final report

In our Preliminary report, we wish to address the first question: that is, applying each approach to the `MovieLens` dataset and compare them. We will present the advantages and disadvantages of each approach and interpret the result. In the Final report, it is expected to have a comprehensible answer for both questions. One more potential question we will address if time allows is to combine the approaches above to derive the best recommendetion algorithm.

References

Bennett, J., Lanning, S. et al. (2007). The netflix prize. In *Proceedings of KDD cup and workshop* **2007** 35. Citeseer.

Harper, F. M. and Konstan, J. A. (2015). The MovieLens Datasets: History and Context. *ACM Transactions on Interactive Intelligent Systems* **5** 1–19.

Salakhutdinov, R., Mnih, A. and Hinton, G. (2007). Restricted Boltzmann Machines for Collaborative Filtering. In *Proceedings of the 24th International Conference on Machine Learning. ICML '07* 791–798. Association for Computing Machinery, New York, NY, USA.

Tang, D., Qin, B., Liu, T. and Yang, Y. (2015). User modeling with neural network for review rating prediction. In *Twenty-Fourth International Joint Conference on Artificial Intelligence*.

Töscher, A., Jahrer, M. and Bell, R. M. (2009). The bigchaos solution to the netflix grand prize. *Netflix prize documentation* 1–52.