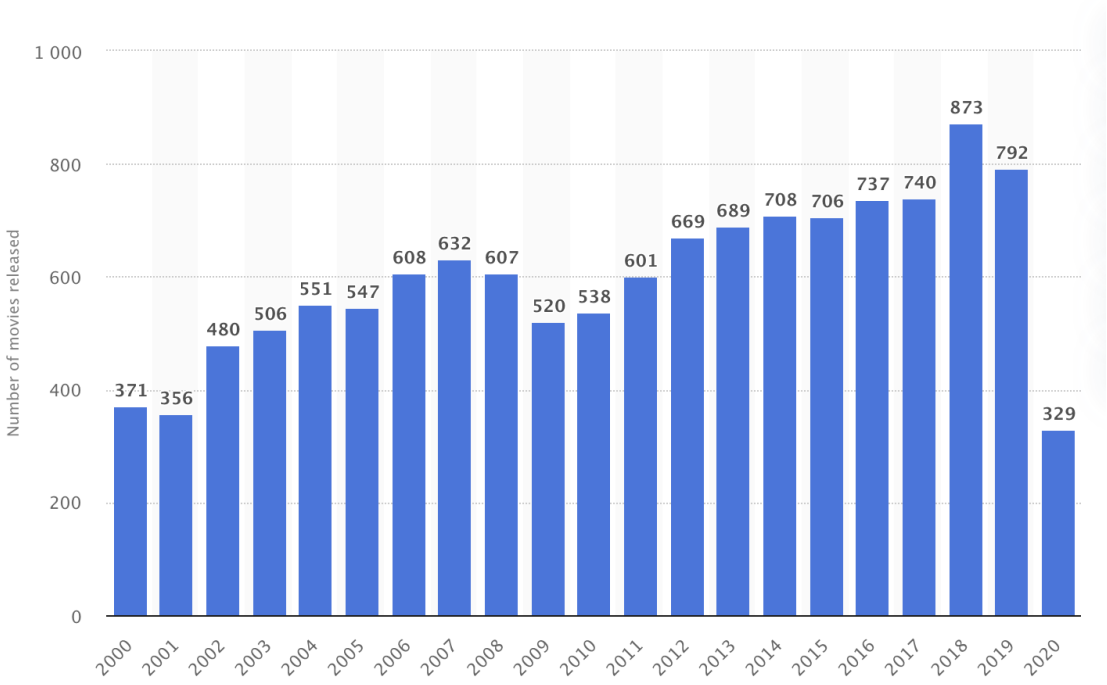
Recommendation System of Movies in MovieLens

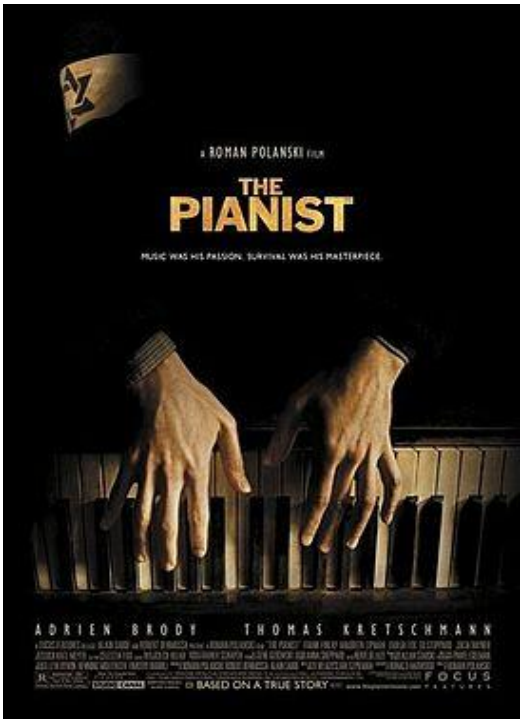
By Tianyang Shen

Since the invention of video films more than a century ago, movies have become one of the most important forms of art and a popular way of entertainment. In 2019, the total gross of movies is estimated to be $11.4 billion. In the year 2011-2019, more than 600 movies in the US and Canada have been released each year. For someone who is willing to decide which movies to watch, a recommendation system is needed for suggesting films. One may have special preference over some specific genres of movies, sometimes the actors, the actress and the directors are the important factors of which movies to watch. Another way to look at this issue is to check other users' rating records. It is reasonable to assume that people having similar taste and rating of movies tend to prefer the same movies.

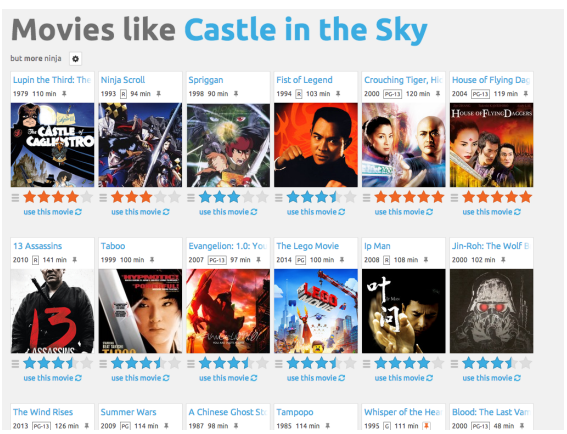


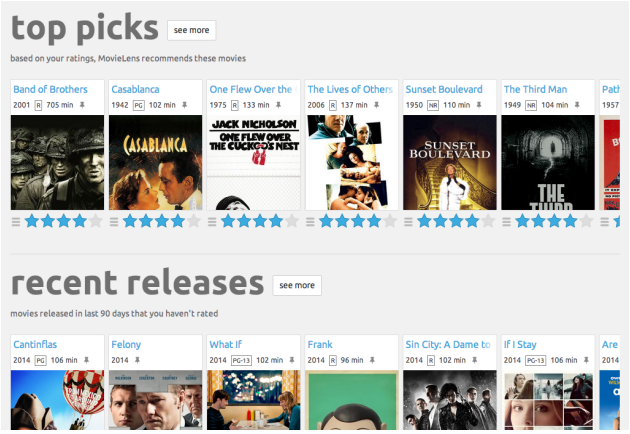
New movie released yearly in US and Canada

For movies, databases have been collecting ratings for them. This information can be used to build up a recommendation system helping someone who is interested in watching new movies which ones to choose. The goal of the system is to provide an algorithm that with the user id, it will provide the top movies the machine learning algorithm suggests.



Since for content-based information such as genres and crew members, the similarities between movies can be found in non machine learning algorithms, this project focuses mainly on collaborative-based filtering. In this method, the system will provide estimated ratings for movies unrated by certain users based on rating records from other similar users. For example, if Alice and Bob share more than 20 same movies that they both rated for 5, then for a movie Alice rated top that Bob hasn't watched, it is highly potential that this movie will be in the recommendation list of Bob. This filtering method will rely on model-based approaches, such as clustering algorithms and matrix factorization algorithms, to predict ratings for unrated movies. The sci-kit surprise package is used for building the model.



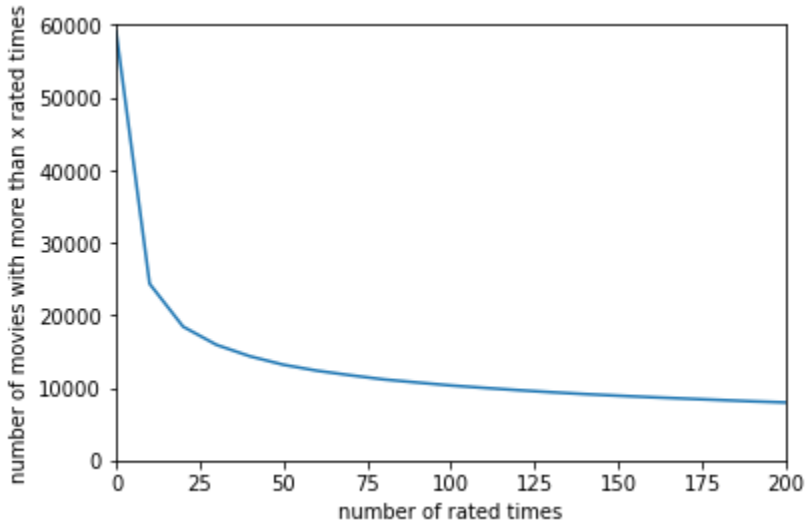


Data

The data set is from https://grouplens.org/datasets/movielens/25m/. It describes 5-star rating (lowest 0.5, highest 5.0 with step 0.5) and free-text tagging activity from MovieLens, a movie recommendation service. It contains 25000095 ratings and 1093360 tag applications across 62423 movies. These data were created by 162541 users between January 09, 1995 and November 21, 2019. This dataset was generated on November 21, 2019. Users were selected at random for inclusion. All selected users had rated at least 20 movies. No demographic information is included. Each user is represented by an id, and no other information is provided. The movie dataset has the information for the title, the year and the genre it belongs to.

IMDB provides other basic information for the movies, here we include the director, actor and actress lists for each movie. Though information is not used for building up the recommendation

system, it is important for a real movie searching engine and it is therefore kept. Some movies don’t have records in IMDB, so they are not kept



The distribution of the rating numbers for the movies shows that there are more than 30000 movies that have less than 10 reviewers. In order to have more reliable analysis, only movies with at least 10 reviewers will be studied. In all, 24430 movies are kept in the dataset. With these amount of movies, 24874543 rating records are kept in the dataset.

Statistics by genre

Each movie in the dataset can belong to multiple genres. We reshape the movie dataset that whether a movie belongs to a certain genre is represented by multiple columns of boolean parameters. In the following example, we can see that movieId 1 is Toy Story released in 1995,

with the average rate 3.893708, and it belongs to “Adventure”, “Animation”, “Children”, “Comedy” (and possibly other unshown columns of genres).

movieId	title	rating numbers	average rate	Action	Adventure	Animation	Children	Comedy	Crime	Documentary	...
1	Toy Story (1995)	57309	3.893708	False	True	True	True	True	False	False	...
2	Jumanji (1995)	24228	3.251527	False	True	False	True	False	False	False	...
3	Grumpier Old Men (1995)	11804	3.142028	False	False	False	False	True	False	False	...
4	Waiting to Exhale (1995)	2523	2.853547	False	False	False	False	True	False	False	...
5	Father of the Bride Part II (1995)	11714	3.058434	False	False	False	False	True	False	False	...

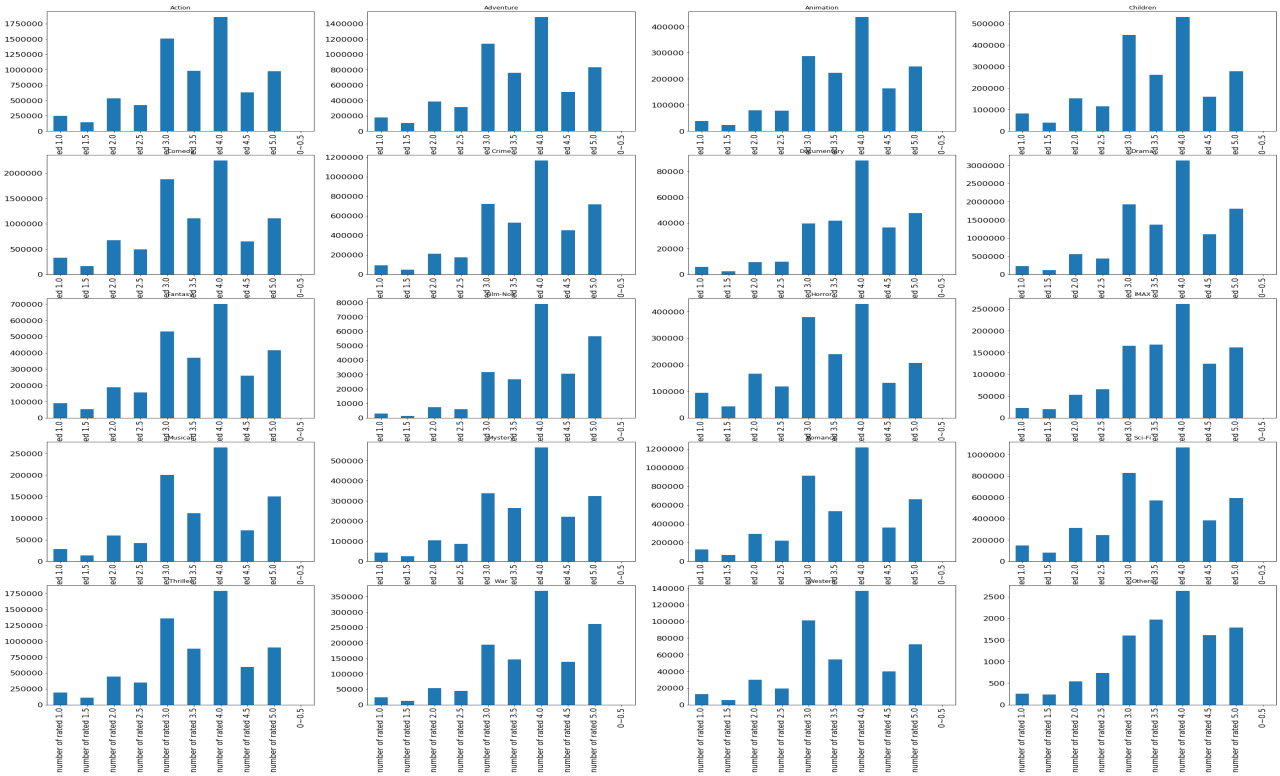
The dataset of movies, with genres in boolean parameters

In order to find how the genre may affect the rating tends, we plot the distribution of ratings in a certain genre, and the distribution of the average ratings of movies in a certain genre.
The first group of histograms in the next page shows how many users rated movies belong to a certain genre in a certain genre. For example, there are more than 1400,000 records of “Adventure” movies rated between 3.5 and 4.0 shown in the second plot.

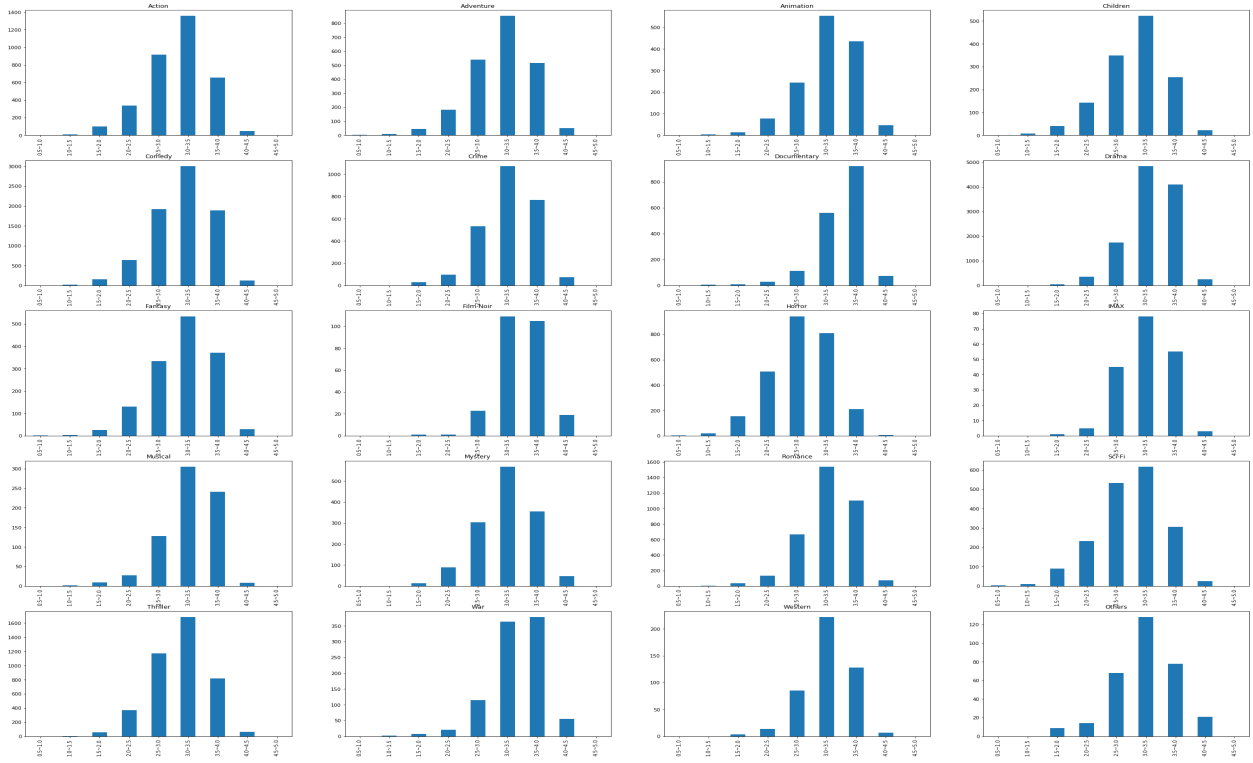
The second plot of the next page shows the distribution of the average ratings of a movie by genre. For example, there are about 900 movies in the genre “action” with average rate 3.0~3.5 shown in the first plot.

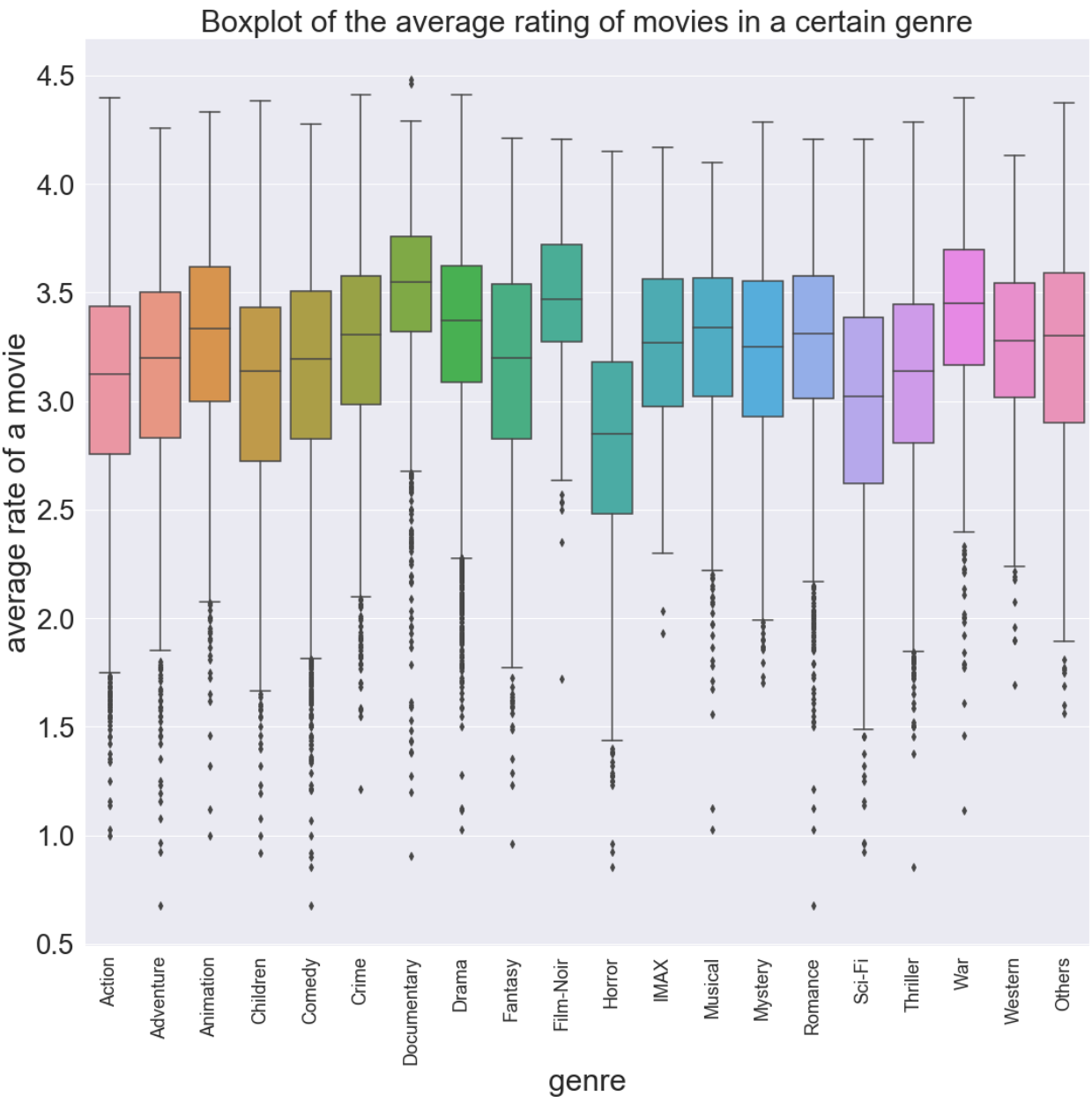
We also plot the box diagrams of the average rating of movies by genre. The plots suggest that almost every genre has the centre value rate between 3.0 and 3.5, and the most centred 75% rates are within 1 point of maximum difference. It also suggests that the extreme ratings are most from low ratings.

distributions of different ratings in each genre



distributions of numbers of movie in different range of ratings for each genre





User Rating Tend

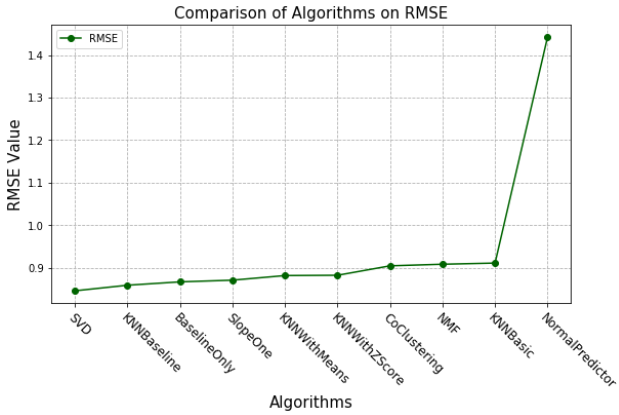
We analyze the features of ratings by users, and it is shown that only about 0.2% users have the rating range less than 1pt, so that they will not have a significant influence on the recommendation system built. We just simply do not consider the range of ratings of a certain user as a factor.

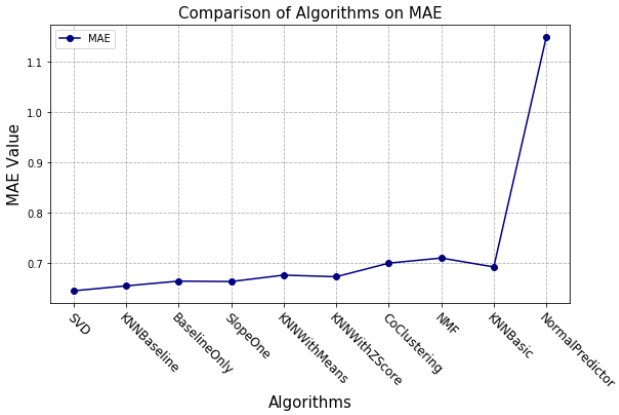
The model

The sci-kit surprise package is used to build the collaborative-based method model. The algorithms of SVD,SVDpp, SlopeOne, NMF, NormalPredictor, KNNBaseline, KNNBasic, KNNWithMeans, KNNWithZScore, BaselineOnly, CoClustering are tested in the first round of selection with 10k rating records. It turns out that SVDpp is a very time consuming algorithm, so that it is not suitable for a dataset of 24.88 million rows.

Algorithm	test_rmse	test_mae	fit_time	test_time
BaselineOnly	0.900157	0.701801	0.020949	0.020054
SVDpp	0.907694	0.702333	28.881190	0.896235
SVD	0.908462	0.704477	0.389803	0.018576

In the second round, we test models with 1M data by using the rest of the algorithms. It is shown that the SVD algorithm result in both the minimum of RMSE and minimum MAE.

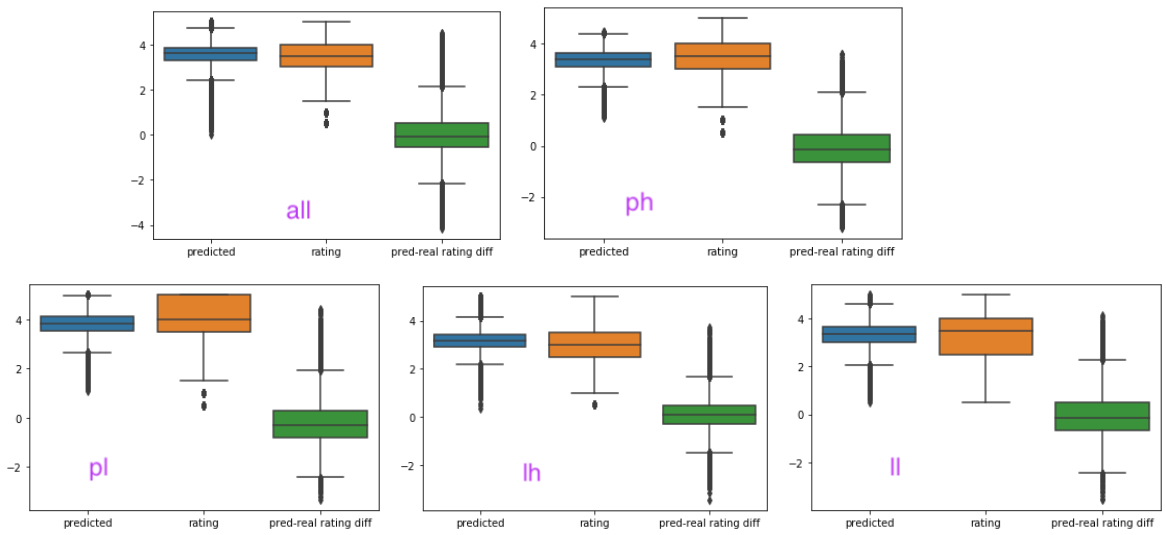




In the third round, we conduct a grid search of hyperparameter space for the SVD algorithm and determine to use the best performing parameters: 'n_epochs': 25, 'lr_all': 0.007, 'reg_all': 0.4. For a 3-fold cross validation with 75% data being the train set, and 25% being the test set, this model has the final MAE of 0.6801 and RMSE of 0.8846.

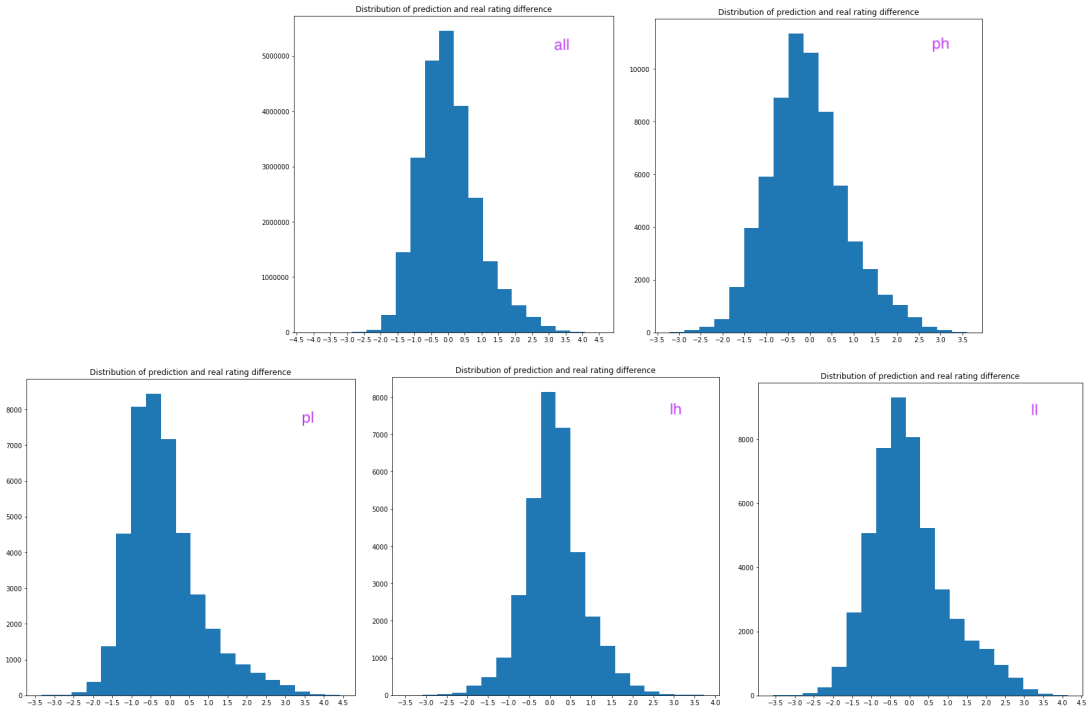
Model Performance Analysis

In order to understand how the recommendation system model works, we consider two special groups of movies: popular movies with more than 10,000 rating records, and less-known movies with less than 50 reviews; as well as two groups of users: heavy users with more than 500 movies rated, and light users with less than 10 movies rated. To simplify the test, we use “p” and “l” for popular and less-unknown movies, “h” and “l” for heavy and light users. For example “ph” stands for statistics of popular movies in heavy users, “ll” stands for less-known movies in light users.

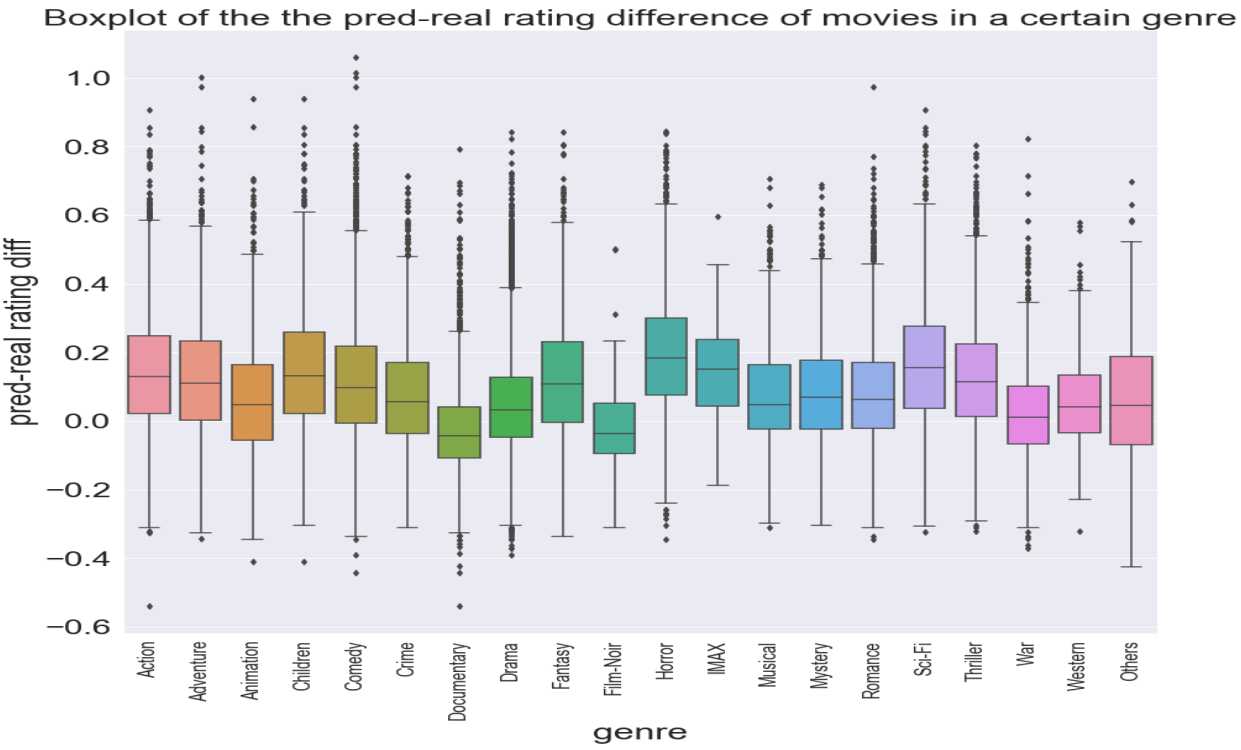


The box diagrams of the predicted ratings of each case suggest that our model tends to provide a narrower range of ratings than the real ratings.

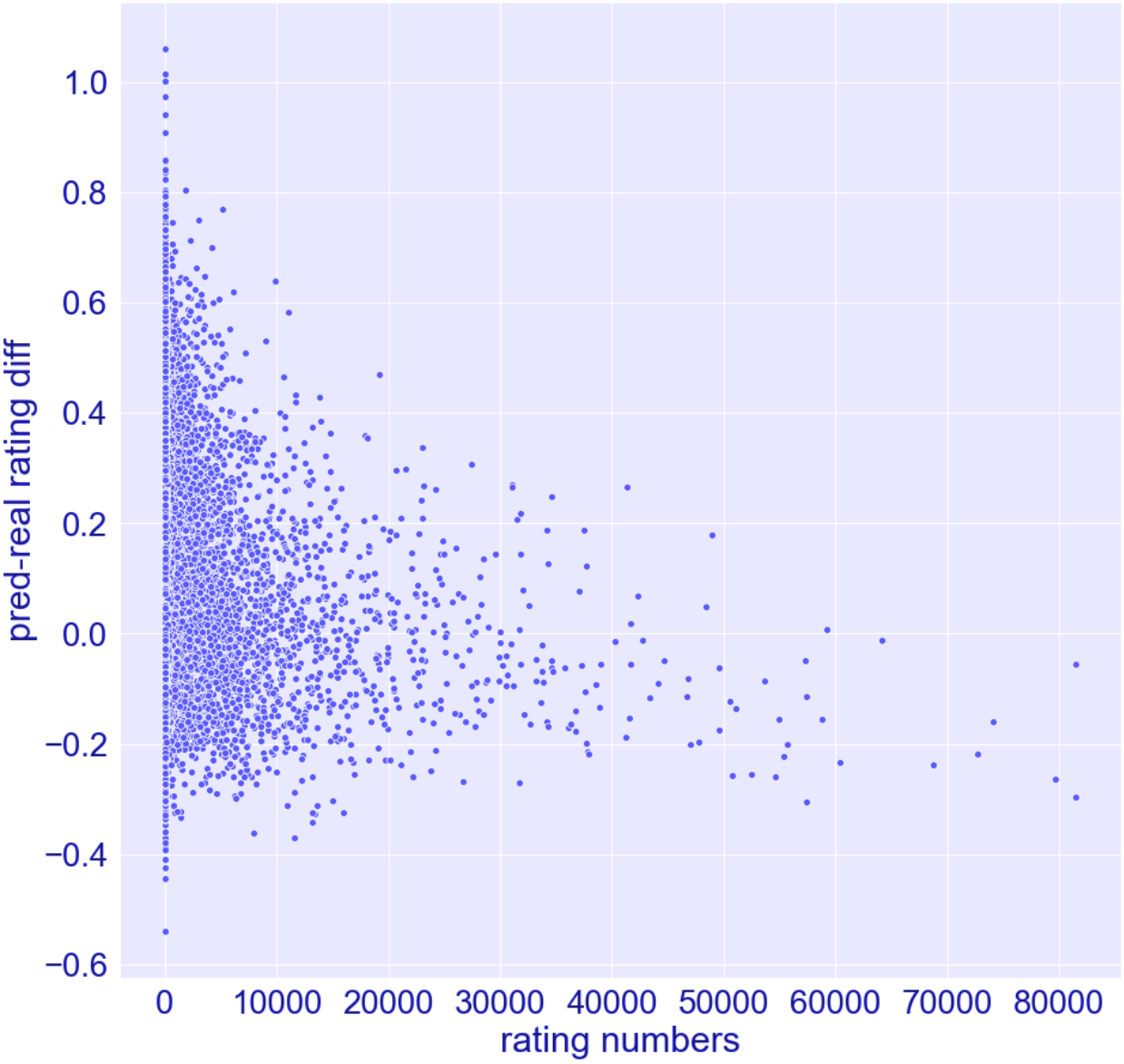
The histograms of the distribution of prediction-real rating difference suggest that in all categories, the most frequent case is that the rating difference is around -0.5 to 0.0.



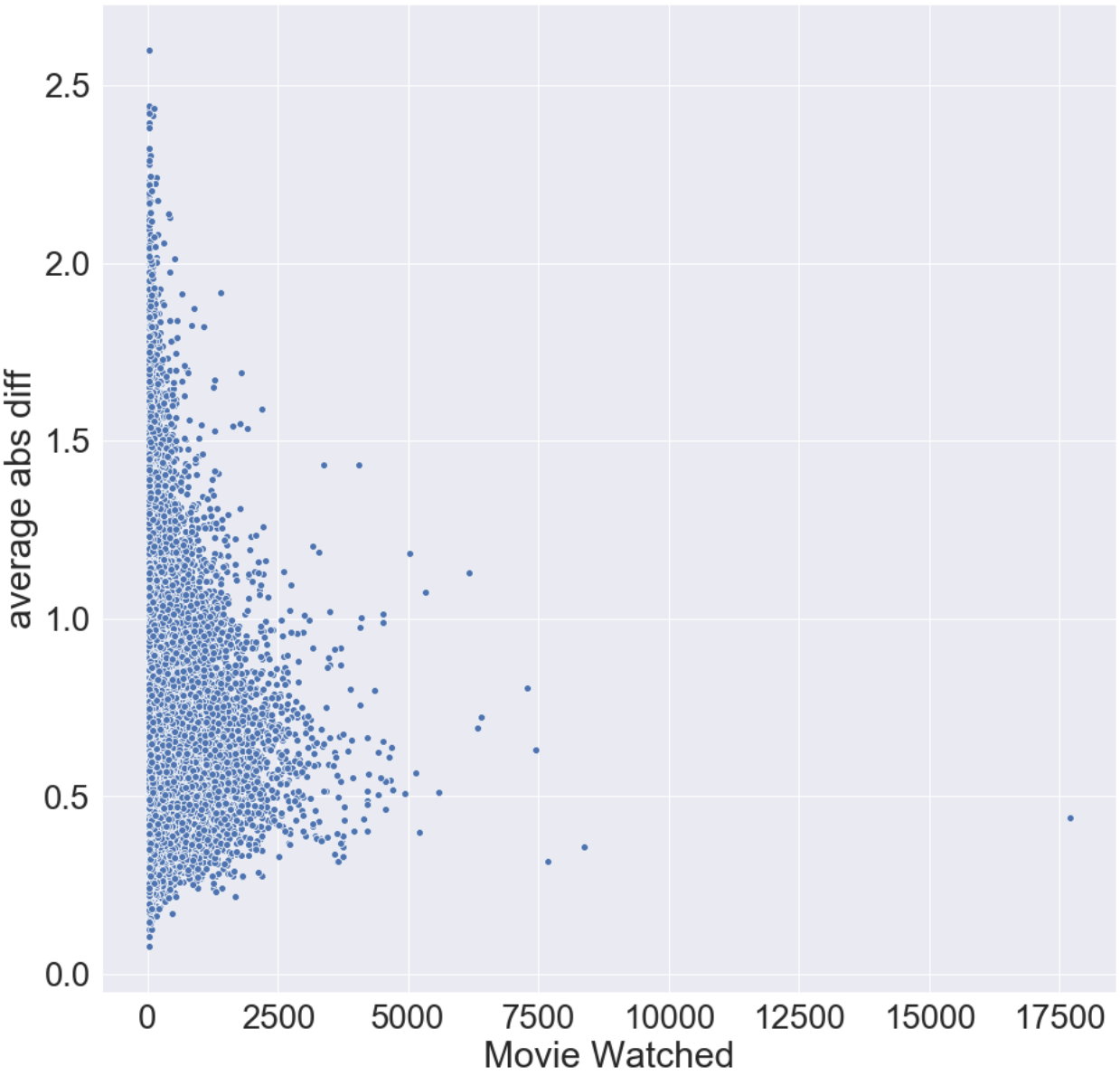
Histograms the distribution of the prediction-real rating difference in each category



We provide the box plot of the difference of the average predicted and real ratings of movies in certain genres. It suggests that for almost every genre 75% of the centre values are within -0.1 and 0.3. It also suggests that averagely speaking for a movie, the recommendation system tends to overpredict the ratings. However, if we plot the scatter plot of the prediction-real difference and the rating record numbers of movies in the following graph, it is shown that movies with many large rating records (more than 50k)are not so that overpredicted, they are slightly underpredicted.



The following scatter plot shows the relation between the MAE of ratings by a certain user and his total number of rated movies. It suggests that for a typical heavy user, the MAE of the recommendation system is around 0.6



User Interaction

We write a function that if provided with a user id, it will give a recommendation list of movies. It is also possible to recommend a movie to a list of potential interested users, but due to the huge number of users causing long computing time , this function would not be very practical.