Project3 Report

Cheng Ma 105033453, **Jinxi Zou** 605036454, **Shuo Bai** 505032786, Xiaoxi Gong 705034355

February 22, 2018

1 Introduction

In this project, our main study is to develop a recommendation system. We use different kinds of collaborative filtering models to analyze this problem and observe the performance. In this project, two main kinds of filtering methods is:

1. **Neighborhood-based collaborative filtering**
2. **Model-based collaborative filtering**

The whole project is divided into 5 different parts.

2 MovieLens dataset

2.1 Question 1

Firstly, we should do some analysis about the dataset. The expression of rating user i to movie j can be used to generate a matrix. The sparsity of matrix **R** is defined as:

$$Sparsity = \frac{available\ rating\ number}{R\ length \times R\ height} \tag{1}$$

Finally, we get the answer $Sparsity = 0.0164391416087$

2.2 Question 2

In this question, we are asked to show the distribution of ratings over frequency, we can observe that most of the ratings are distributed over 3.5 to 5.0 and the number of very low ratings is comparatively smaller.

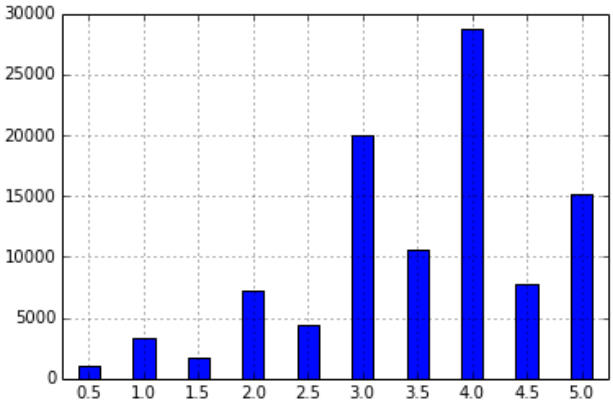


Figure 1: frequency of rating values

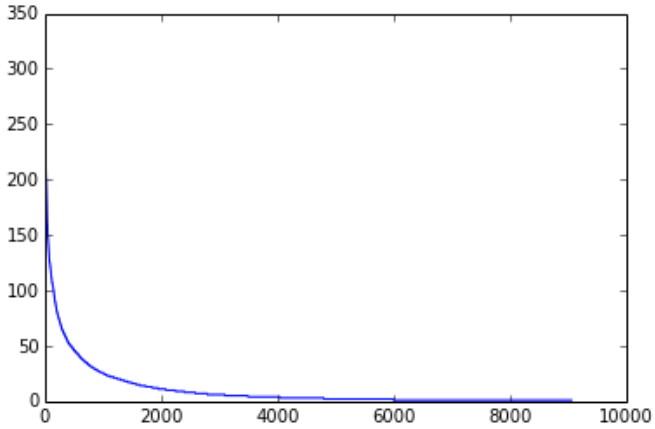


Figure 2: Distribution of ratings among movies

2.3 Question 3

In this question, the point is to show the relation between ratings number and movies index.

2.4 Question 4

In this question, the point is to show the relation between ratings number and users.

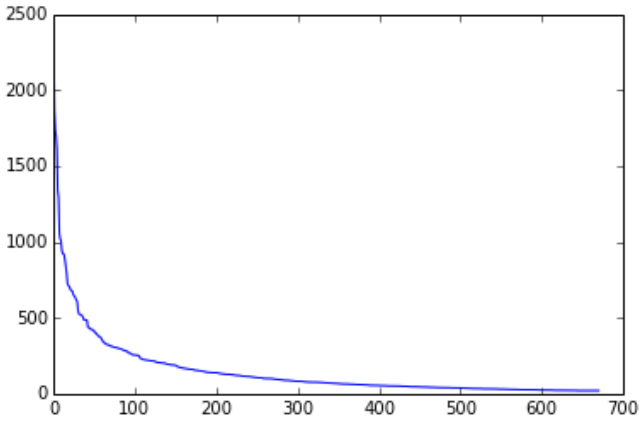


Figure 3: Distribution of ratings among users

2.5 Question 5

From the result we get from question 3, we observe that the most important feature it implicates is that the distribution is quite centralized. It means that many people have a similar preference to a group of movies. Therefore, we can use collaborative ways to use a similar preference to recommend movies to other users.

2.6 Question 6

This question is to compute the variance for each movie's ratings. From the variance, we can judge the coherence between users' view. As the result shows, most movies have a low variance. It indicates that users have similar comment on most movies. Therefore, the collaborating feature is quite high here. It is suitable to take further study.

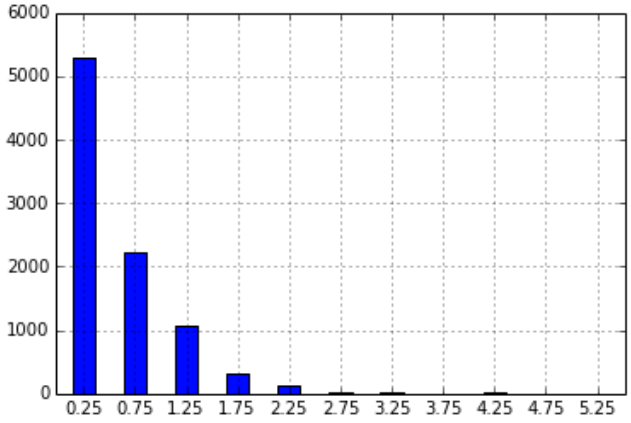


Figure 4: Distribution of each movie's rating variance

3 Neighborhood-based collaborative filtering

3.1 Question 7

The formula for μ_μ in terms of I_μ and r_{uk} is

$$\mu_\mu = \frac{\sum_{i \in I_u} r_{ui}}{len(I_u)} \tag{2}$$

3.2 Question 8

It means the set of item indices that both user u and v have rated.

3.3 Question 9

If one user will rate all items highly or poorly, then the absolute rates can not reflect the true rate for the items, and thus the relative rates matter in such case.

3.4 Question 10

In this task, we could use the function *KNNWithMeans* and *cross_ validate* to get the final result, and the result can be shown as follows.

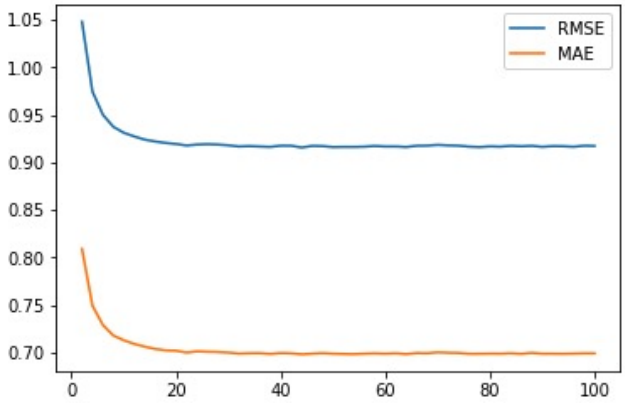


Figure 5: RMSE and MAE against k

3.5 Question 11

From the figure we can conclude that the minimum k is 22.

3.6 Question 12

The popular movie trimming is defined as trimming the test set to contain movies that has received more than 2 ratings. And thus after trimming the test data, we can use the codes in Question 10 to get the RMSE curve. And the minimum average RMSE is 0.88.

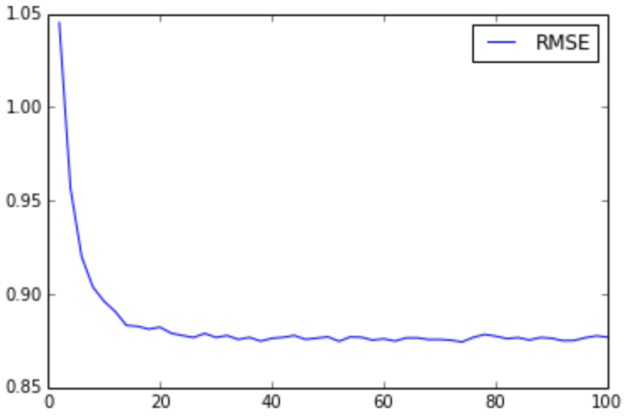


Figure 6: RMSE of the popular movie

3.7 Question 13

The unpopular movie trimming is defined as trimming the test set to contain movies that has received less than or equal to 2 ratings. And its implementation is almost the same as Question 12. The minimum average RMSE is 1.0.

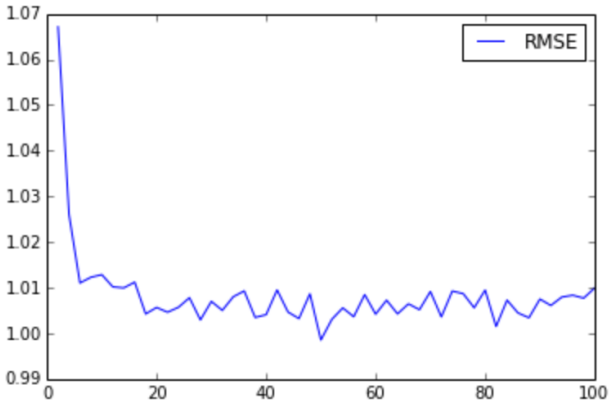


Figure 7: RMSE of the unpopular movie

3.8 Question 14

The High variance movie trimming is defined as trimming the test set to contain movies that has variance (of the rating values received) of at least 2 and has received at least 5 ratings in the entire dataset. And the minimum average RMSE is approximately 1.40.

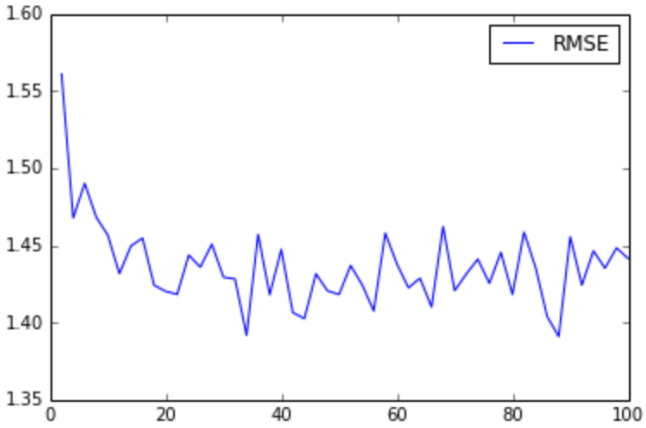


Figure 8: RMSE of the high variance movie

3.9 Question 15

After using threshold to categorize the train set, we can use the estimated values to judge this recommendation system. And we can see that the areas under the ROC curves are almost the same, which indicates that different thresholds do little help to the quality of the classifier.

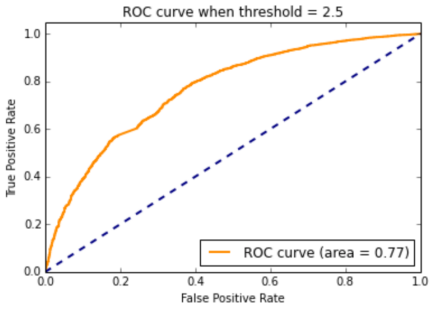


Figure 9: ROC curve when threshold = 2.5

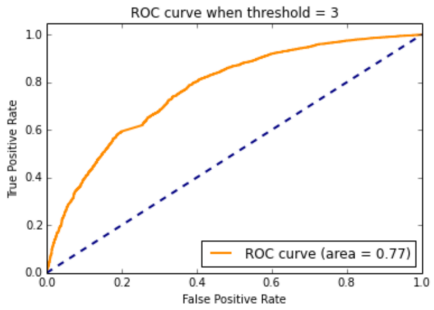


Figure 10: ROC curve when threshold = 3

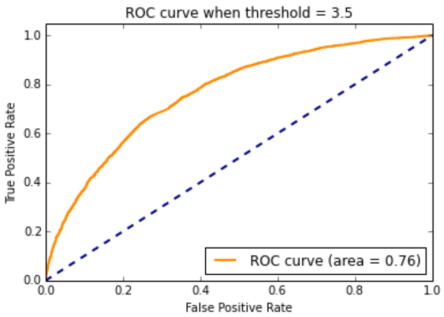


Figure 11: ROC curve when threshold = 3.5

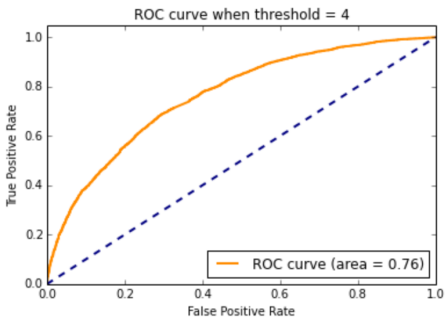


Figure 12: ROC curve when threshold = 4

4 Model-based collaborative filtering

4.1 Question 16

Yes, the optimization function is convex. For U fixed, the derivative of equation is

$$\frac{\partial W_{ij}(r_{ij} - (UV^T)_{ij})^2}{\partial U} = \frac{\partial(||R||^2 - 2RUV^T + ||UV^T||^2)}{\partial U} = -2RV^T + 2U^TVV^T \quad (3)$$

$$\frac{\partial(-2RV^T + 2U^TVV^T)}{\partial U^T} = 2VV^T \tag{4}$$

The derivative result of equation (5) is semidefinite, so it's convex.

4.2 Question 17

We designed the NNMF-based filter and used cross-validation to evaluate its performance. The plot of RMSE and MAE are following.

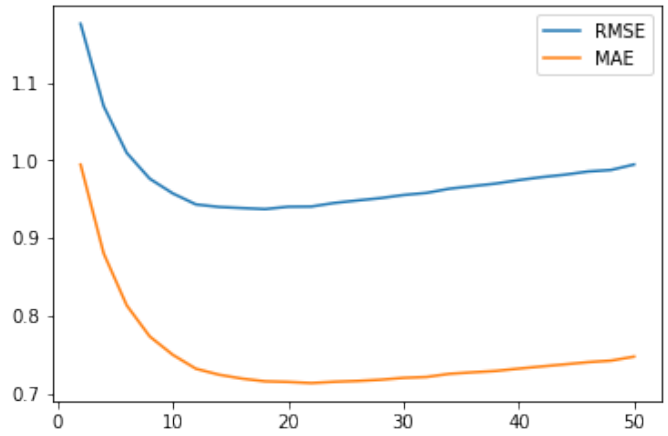


Figure 13: RMSE and MAE of the NNMF-based filter

4.3 Question 18

From the plot in question 17, the k that gave the lowest RMSE and MAE is $k = 18$. The minimum RMSE is 0.9374, the minimum MAE is 0.7148. From the instruction file in Movielens dataset, we knew that the number of genres of movies in this dataset is 18, which is the same with optimal $k = 18$. This is because filter is based on NNMF, when number of latent factor is the same with that of movie genres, the filter can predict by considering all genres of movies.

4.4 Question 19

We designed the filter base on NNMF and evaluate its performance on popular movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.9177.

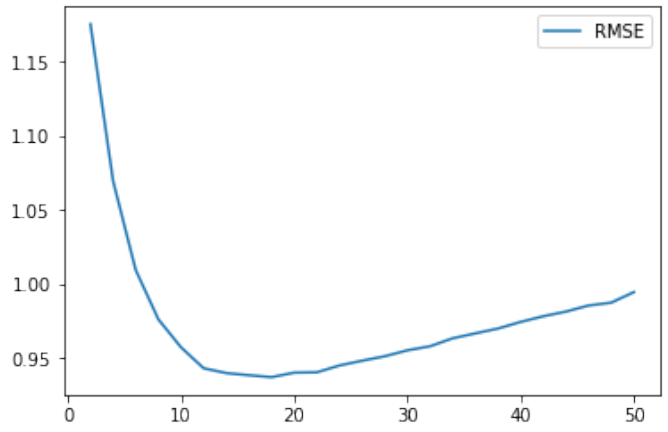


Figure 14: RMSE of the NNMF-based filter on popular movie trimmed set

4.5 Question 20

We designed the filter base on NNMF and evaluate its performance on unpopular movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.9243.

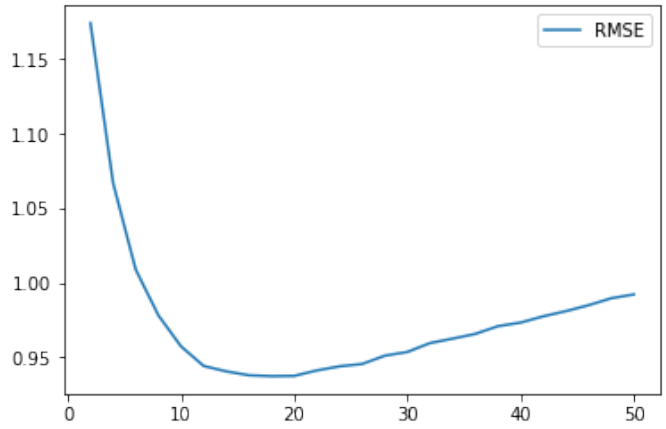


Figure 15: RMSE of the NNMF-based filter on popular movie trimmed set

4.6 Question 21

We designed the filter base on NNMF and evaluate its performance on high variance movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.9243.

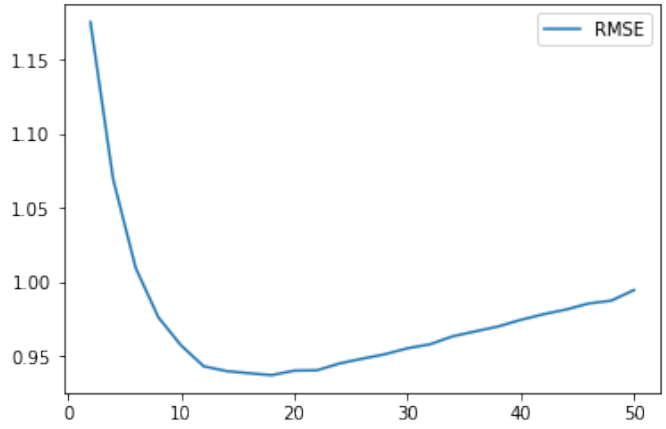


Figure 16: RMSE of the NNMF-based filter on high variance movie trimmed set

4.7 Question 22

The ROC curves for NNMF-based filter with different threshold numbers are shown in Following. The AUC value are shown in figures.

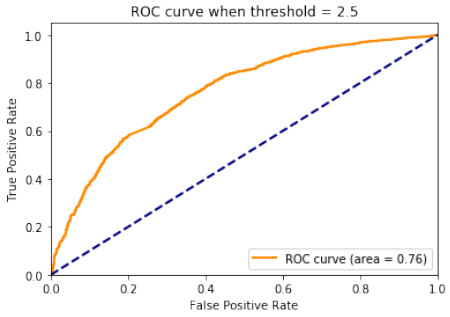


Figure 17: ROC curve when threshold = 2.5

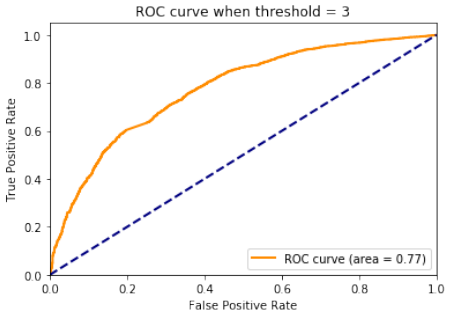


Figure 18: ROC curve when threshold = 3

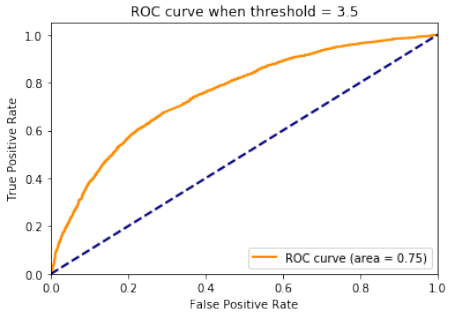


Figure 19: ROC curve when threshold = 3.5

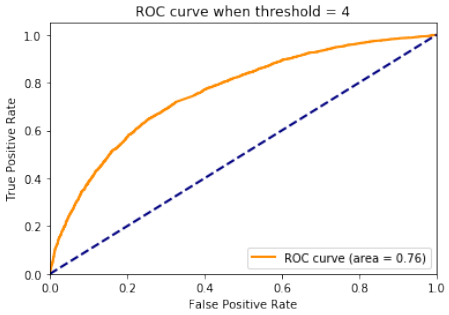


Figure 20: ROC curve when threshold = 4

5 MF with bias filter performance on trimmed test set

5.1 Question 23

We sorted the column of V matrix after MF factorization and got the top 10 movies' genres. The top 10 movies genres of column are shown in following Table. From the table, we found that genres within each column are similar. For example, there are five comedy movies in column 3 top 10 movies. There are five action movies in column 6 top 10 movies. It looks like each latent factor represents a kind of genres. The value of i movie row, j genre column represents the "probability" of movie i is belong to genre j.

column 9	column 3	column 6
'Horror|Thriller'	'Action|Comedy'	'Horror'
'Action|Drama|Thriller'	'Horror|Mystery'	'Action|Comedy|Crime|Fantasy'
'Comedy|Drama|Musical'	'Comedy'	'Action|Crime|Fantasy'
'Horror|Thriller'	'Comedy|Romance'	'Action|Adventure|Comedy|Sci-Fi|Thriller'
'Adventure|Comedy'	'Comedy|Musical|Romance'	'Horror|Thriller'
'Children|Comedy'	'Action|Drama|Thriller'	'Adventure|Comedy'
'Adventure|Animation|Children|Comedy	'Drama|War'	'Adventure|Animation|Children|Comedy'
'Action|Horror|Sci-Fi|Thriller'	'Action|Comedy|Crime|Fantasy'	'Adventure|Drama|Fantasy|Romance'
'Action|Adventure|Romance'	'Crime|Drama'	'Action|Horror|Sci-Fi|Thriller'
'Action|Adventure|Sci-Fi'	'Drama|Romance'	'Action|Adventure|Romance'

5.2 Question 24

We designed the MF-based filter and used cross-validation to evaluate its performance. The plot of RMSE and MAE are following.

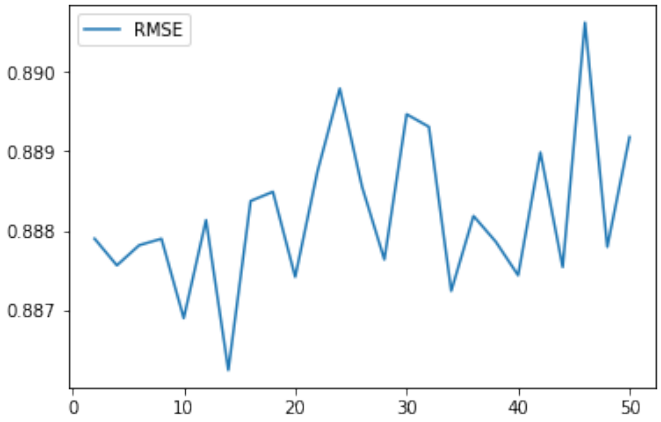


Figure 21: RMSE of the MF-based filter

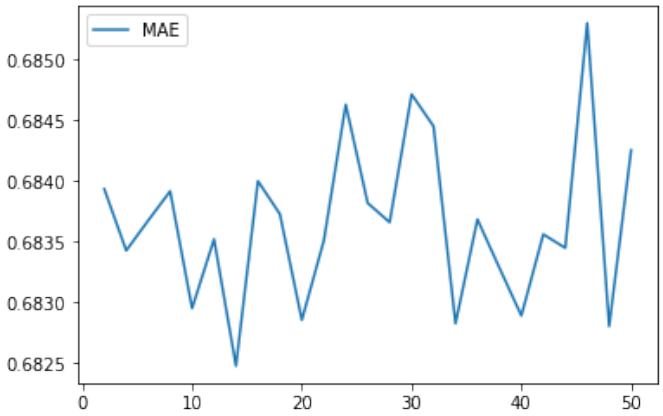


Figure 22: MAE of the MF-based filter

5.3 Question 25

From the plot in question 24, the k that gave the lowest RMSE and MAE is $k = 14$. The minimum RMSE is 0.8862, the minimum MAE is 0.6824.

5.4 Question 26

We designed the filter base on MF and evaluate its performance on popular movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.8736.

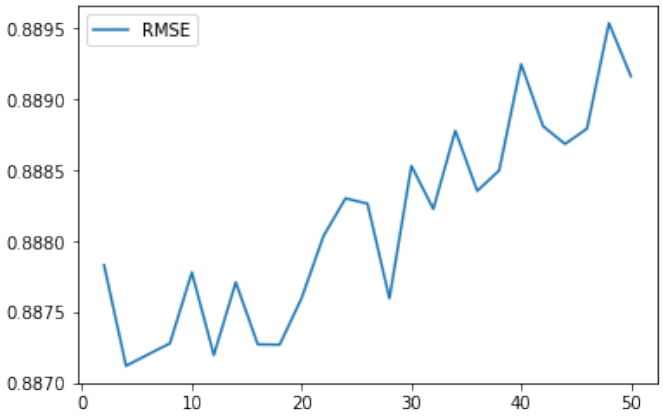


Figure 23: RMSE of the MF-based filter on popular movie trimmed set

5.5 Question 27

We designed the filter base on MF and evaluate its performance on unpopular movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.8741.

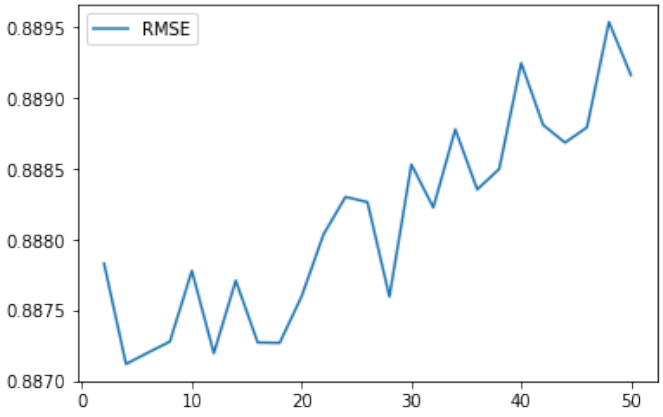


Figure 24: RMSE of the MF-based filter on popular movie trimmed set

5.6 Question 28

We designed the filter base on MF and evaluate its performance on high variance movie trimmed set with respect different number of latent factors. The results are in following figures. The minimum RMSE is 0.8643.

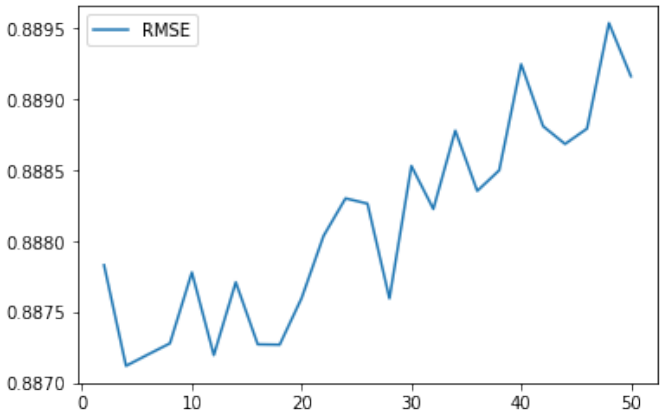


Figure 25: RMSE of the MF-based filter on high variance movie trimmed set

5.7 Question 29

The ROC curves for MF-based filter with different threshold numbers are shown in Following. The AUC value are shown in figures.

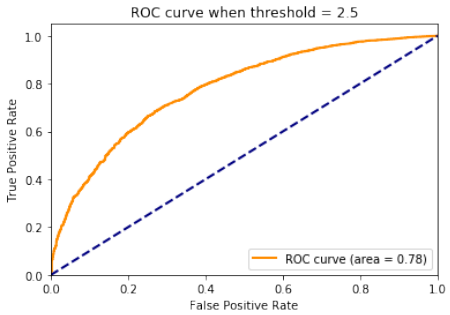


Figure 26: ROC curve when threshold = 2.5

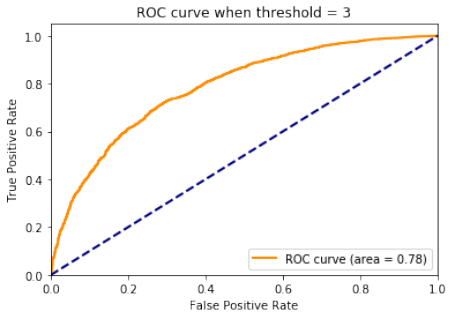


Figure 27: ROC curve when threshold = 3

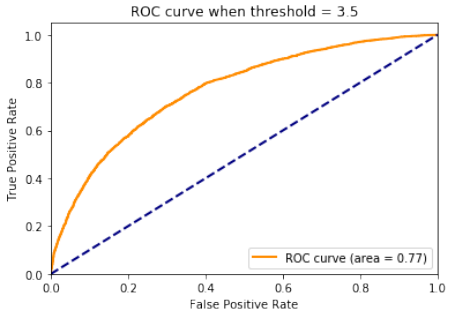


Figure 28: ROC curve when threshold = 3.5

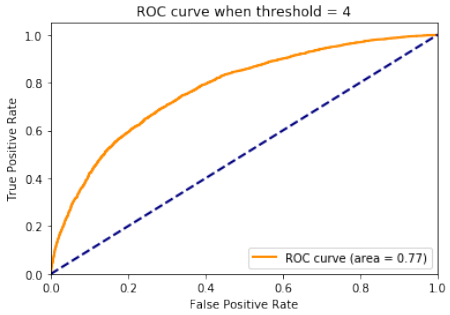


Figure 29: ROC curve when threshold = 4

6 Naive collaborative filtering

In this part, we consider a naive collaborative filtering which predict the rating in the following way:

$$\hat{r}_r j = u_i \tag{5}$$

u_i is the mean of all $user_i$ rating, in this case, we use a different way and reconstruction a predict algorithm to take the task.

6.1 Question 30

In this question we use a new 10-fold validation. We cut the data into ten pieces and use predict function to get the average RMSE

```
RMSE: 0.9623
RMSE: 0.9553
RMSE: 0.9642
RMSE: 0.9684
RMSE: 0.9610
RMSE: 0.9536
RMSE: 0.9767
RMSE: 0.9665
RMSE: 0.9539
RMSE: 0.9608
avg: 0.9622719245124223
```

Figure 30: RMSE of Naive collaborative fitering over all data

6.2 Question 31&32&33

This question is the same with above question except we use naive collaborative filter in this case.

```
RMSE: 0.9584
RMSE: 0.9363
RMSE: 0.9391
RMSE: 0.9411
RMSE: 0.9418
RMSE: 0.9409
RMSE: 0.9516
RMSE: 0.9426
RMSE: 0.9518
RMSE: 0.9611
0.9464811276282006
```

(a)

```
RMSE: 1.0140
RMSE: 0.9813
RMSE: 0.9884
RMSE: 1.0113
RMSE: 0.9985
RMSE: 0.9852
RMSE: 0.9858
RMSE: 0.9964
RMSE: 1.0005
RMSE: 1.0190
0.9980208386428572
```

(b)

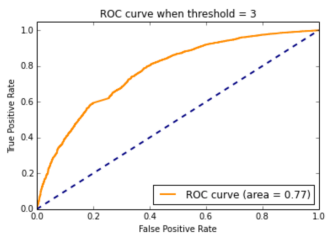
```
RMSE: 0.9611
RMSE: 0.9683
RMSE: 0.9624
RMSE: 0.9617
RMSE: 0.9602
RMSE: 0.9691
RMSE: 0.9734
RMSE: 0.9552
RMSE: 0.9573
RMSE: 0.9557
0.9624426162629863
```

(c)

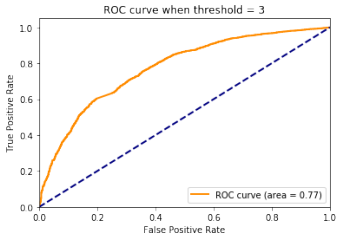
Figure 31: RMSE of popular, unpopular, high variance movie

7 Performance comparison

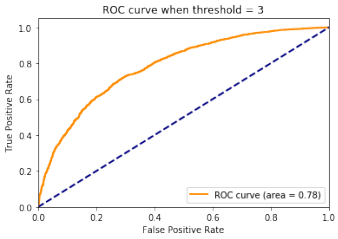
7.1 Question 34



(a)



(b)



(c)

Figure 32: Comparison between K-NN, NNMF, MF filter

According to the comparison figure shows, ROC curve of KNN and NNMF look similar and the area is the same. MF ROC plot is more smooth and the area is 0.01 larger than previous two plots. Therefore, we can get the conclusion that MF collaborative filtering will gain higher performance.

7.2 Question 35

Precision calculate the ratio of intersection of recommended set and set liked by users. Briefly, Precision is a measure of result relevancy, while recall is a measure of how many truly relevant results are returned.

7.3 Question 36

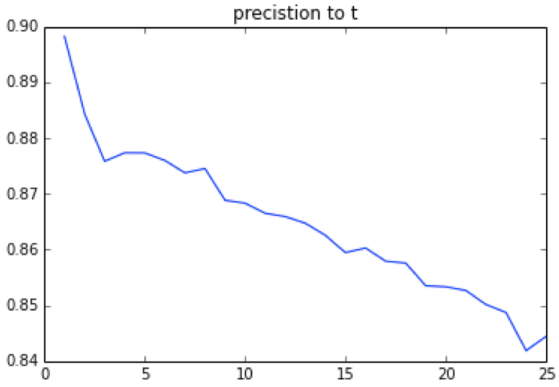


Figure 33: precision to t

The precision to t is decreasing with t increasing. It is almost Monotonic curve. The precision not drop too much. It just drop less than 0.1.

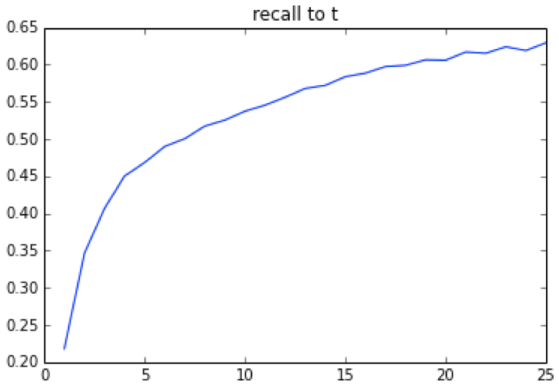


Figure 34: recall to t

The recall to t is increasing with t increasing. It is almost a monotonic curve.The recall to t increases a lot with t increasing. Recall increase fast at the beginning of the curve. Recall to t increases more than 0.4.

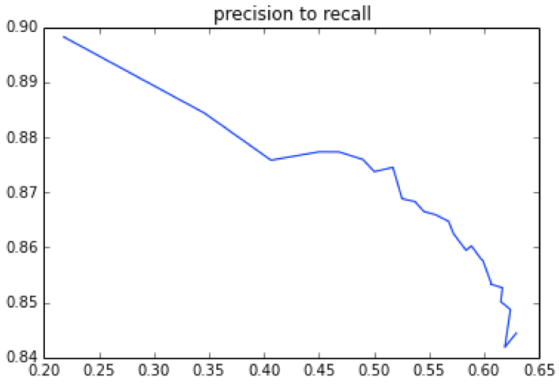


Figure 35: precision to recall

The precision to recall is decreasing with recall increasing. It is almost a Monotonic curve. The precision not drop too much. The precision drops fast when recall is larger than 0.5. It just drop less than 0.05.

7.4 Question 37

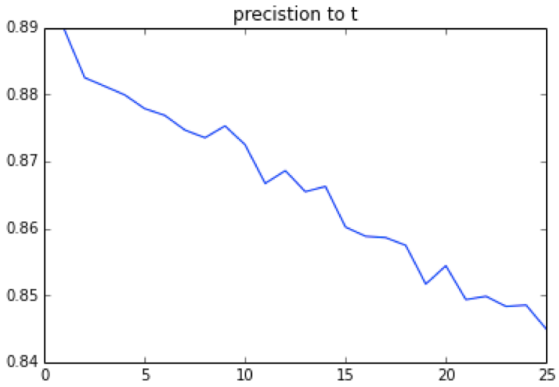


Figure 36: precision to t

The precision to t is decreasing with t increasing. The curve looks like a linear curve. It is almost Monotonic curve. The precision not drop too much. It just drop less than 0.05.

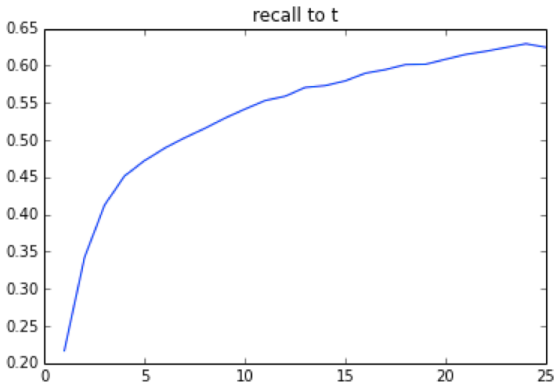


Figure 37: recall to t

The recall to t is increasing with t increasing. It is almost a monotonic curve.The recall to t increases a lot with t increasing. Recall increase fast when t is small. Recall to t increases more than 0.4.

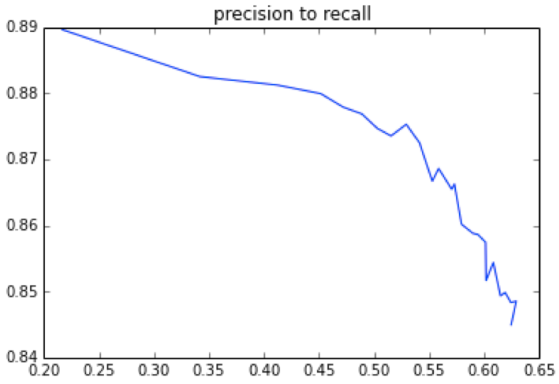


Figure 38: precision to recall

The precision to recall is decreasing with recall increasing. It is almost a Monotonic curve. It drops fast when recall is larger than 0.5. The precision not drop too much. It just drop less than 0.05.

7.5 Question 38

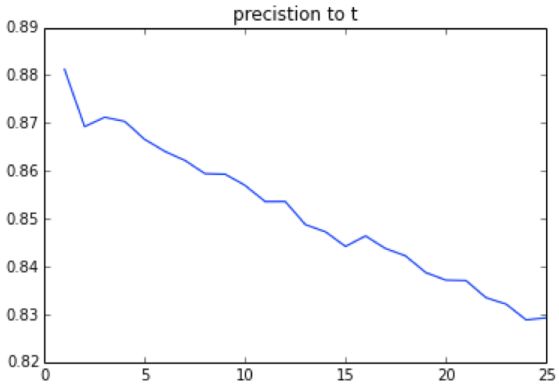


Figure 39: precision to t

The precision to t is decreasing with t increasing. The curve looks like a linear curve. It is almost Monotonic curve. The precision not drop too much. It just drop less than 0.05.

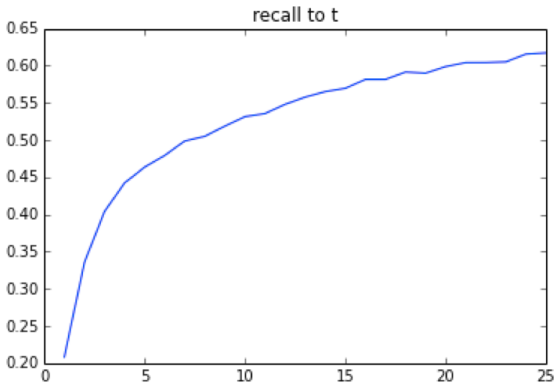


Figure 40: recall to t

The recall to t is increasing with t increasing. It is almost a monotonic curve.It increase fast when t is small. The recall to t increases a lot with t increasing. Recall to t increases more than 0.4.

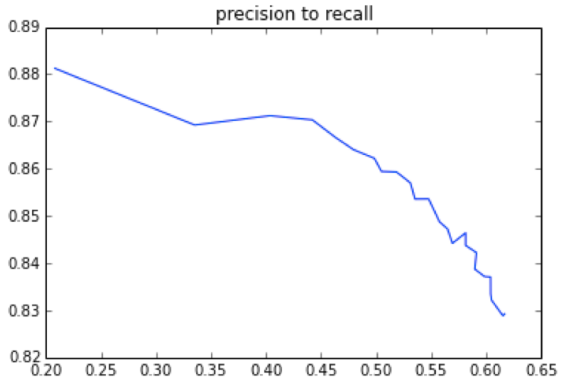


Figure 41: precision to recall

The precision to recall is decreasing with recall increasing. It is almost a Monotonic curve. It drop fast when recall is very small and when the recall is larger than 0.45. The precision not drop too much. It just drop less than 0.05.

7.6 Question 39

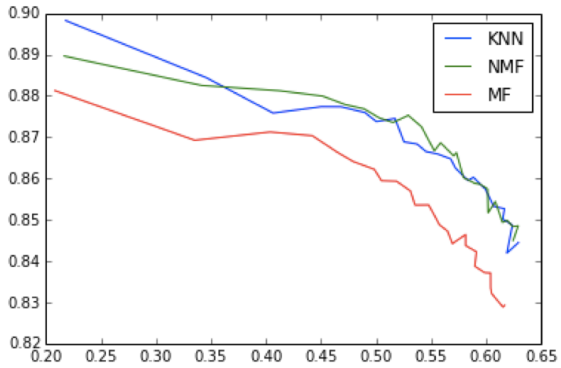


Figure 42: KNN,NNMF,MF

Relevance of recommendation list generated by MF is lowest. When the recall is small, the relevance of recommendation list generated by KNN is higher than NNMF. When recall reach around 0.35, the relevance of recommendation list generated by KNN is lower than NNMF. After recall equals 0.48, the relevance of recommendation list generated by KNN and NNMF is almost same.