# EE 219 Project 3

Yuan Tao (305033824), Yi Jia (805033204), Shuojian Ye (904946811), Fangjia Zhu (905036438)

February 22, 2018

## Introduction

In this project we used the collaborative filtering models to build a recommender system, which is based on the utilization of user data to infer customer interests. Two types of collaborative filtering methods were implemented and examined in detail: Neighborhood-based collaborative filtering and Model-based collaborative filtering. Similarity between users were calculated using Pearson-correlation coefficient, unspecified ratings in the MovieLens dataset were predicted as well as ranking of movies were performed using the predicted ratings. .

## Question 1

We first constructed a ratings matrix containing m users (rows) and n movies (columns). The (i, j) entry of the matrix is the rating of user i for movie j and is denoted by $R_{ij}$:

```
[[ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]
 [ 0.  0.  0. ...,  0.  0.  0.]
 ...,
 [ 0.  0.  0. ...,  0.  0.  0.]
 [ 4.  0.  0. ...,  0.  0.  0.]
 [ 5.  0.  0. ...,  0.  0.  0.]]
(671, 9066)
```

Then we calculated density using the ratings matrix, which is defined as the total number of available ratings over total number of possible ratings. Finally, the sparsity of the movie rating dataset was given by subtracting the density from 1. **The final result we obtained for sparsity is 0.9836**

## Question 2

The frequency of the rating values is shown below. It is clear from the figure that the distribution is left-skewed with rating values mainly follow into the interval of [3,5]. This indicates that most movies are receiving above average ratings.

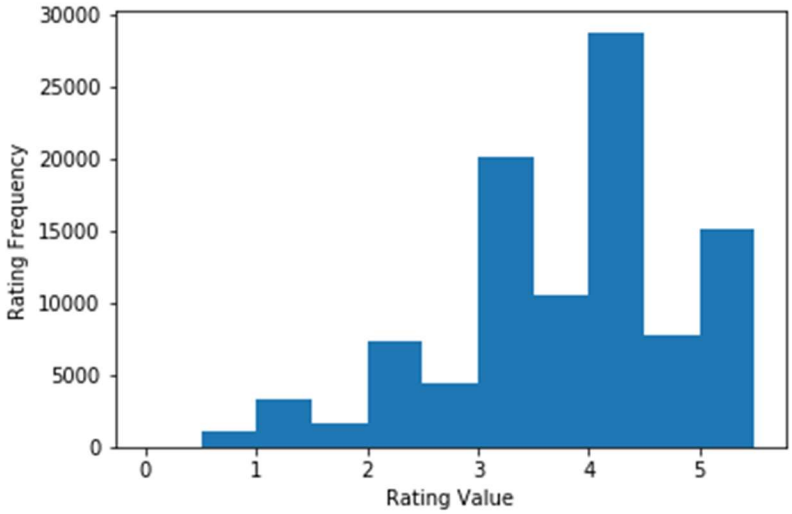



*Figure 1. The Frequency of the Rating Values*

## Question 3

For this question, we used the ratings matrix mentioned above and iterated its columns to count the number of ratings received by each movie and sort the frequency in descending order to obtain the distribution plot as shown below:

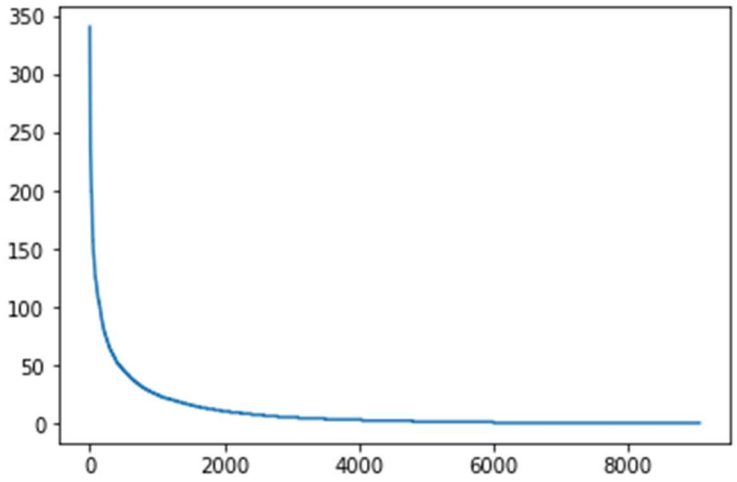



*Figure 2 . The Distribution of Ratings among Movies*

The distribution is in a manifest trend of exponential decay, with few movies receiving tremendous amount of ratings for about several hundred while the others are receiving merely around 20-30 ratings. This may result from the fact that some popular movies were paid a lot of attentions, good or bad, while some other movies that are not popular enough were relatively vanished in the filming market and thus did not receive too much ratings.

## Question 4

In this question we obtained the distribution of the number of movies rated by users in a descending order using a similar procedure in Question 3. The plot is shown below, which is also exponential decaying. This can also be similarly explained as people who like rating and writing reviews would naturally rate a lot more than people who are not a big fan of it.

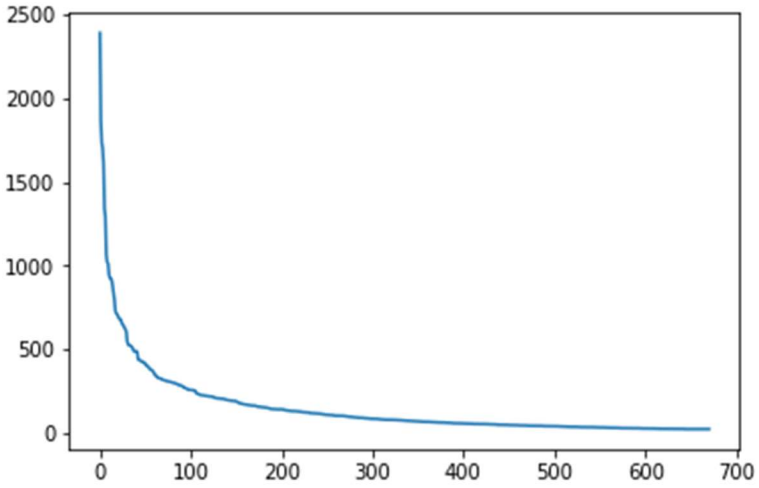



*Figure 3 . The Distribution of Ratings among Users*

## Question 5

The salient feature of the distribution of ratings among movies is extremely unbalanced. To be more specific, some movies get more than three hundred of ratings while others only get less than ten ratings. This makes the rating matrix sparse and gives challenges to designing collaborative filtering methods since most of the ratings are unspecified.

In order to solve the problem, we used the feature that the observed ratings are often highly correlated across various users and the similarity between users or items can be used to make inferences about incompletely specified values.

## Question 6

The variance of the rating values received by each movie is shown below. We can figure out that the variances are mainly located between 0 to 1.5, in other word, a large proportion of movies received the ratings which have relatively low variance and that means the users have similar ratings on these movies.

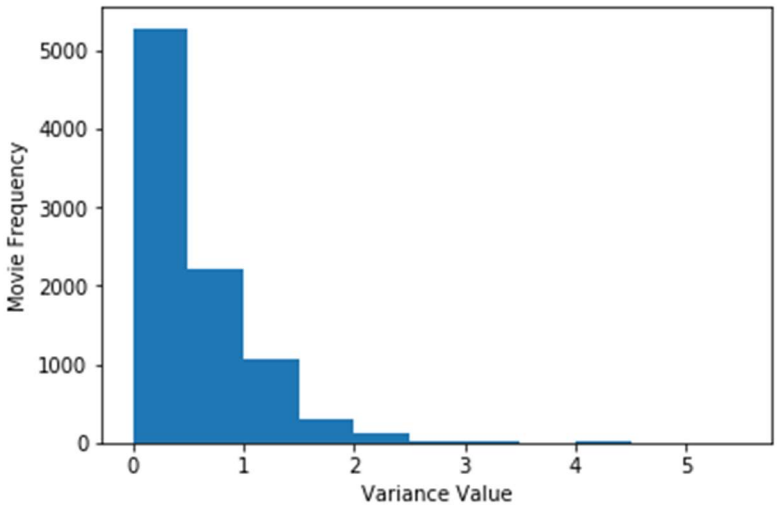



*Figure 4 . The Variance of the Rating Values received by each Movie*

## Question 7

For Pearson-correlation coefficient, the mean rating for user u $\mu_{\square}$ can be calculated using the following equation: $\mu_u = \frac{1}{|I_u|}\sum_{k \in I_u} r_{uk}$, where $|I_u|$ is the cardinality of the set $I_u$.

## Question 8

$I_u \cap I_v$ is the set of items rated by both users u and v. $I_u \cap I_v = \emptyset$ when the users u and v don't have common items rated.

## Question 9

Mean-centering would better reflect the features. Given that each feature should have equal probability or weight to influence the classification result, a large scale of data, in our case movies rated by users who rate all items highly, would have a much greater weight than those rated by users who rate all items poorly and thus the features would not be a true representation of the data.

Also, the proposed KNN algorithm is using Pearson-correlation coefficient as distance measure, in which the training ratings would already have their mean been subtracted. It is quite obvious that the data to be predicted must be preprocessed the exact same way as the training data.

## Question 10

The curve of RMSE and MAE against k are shown as below. We can figure out that both RMSE and MAE decrease rapidly with the increase of number of neighbors at the very beginning and then converge to a steady state.

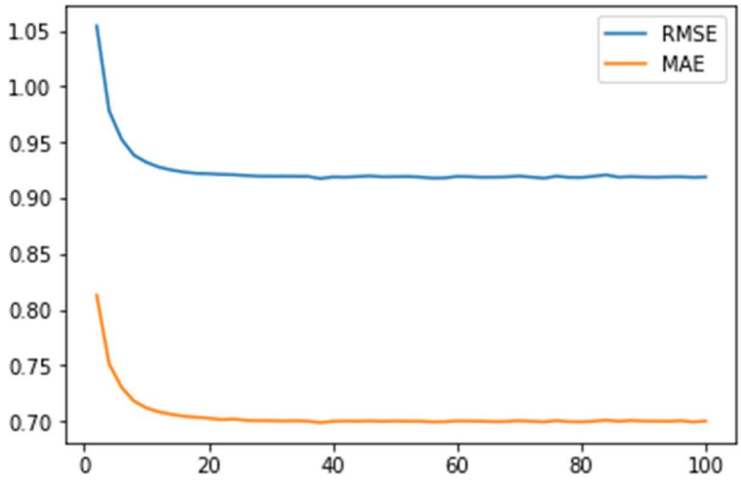



*Figure 5 . The Average RMSE and Average MAE against k*

## Question 11

We found the 'minimum k that correspond to the k value for which average RMSE and average MAE converges to a steady-state value. We used 0.001 as the threshold of “steady state”, in other words, if the average RMSE and average MAE decrease less than 0.001 with the increase of k, then we defined it as steady state. The minimum k we found with this context is 18, with corresponding RMSE to be 0.9221 and MAE to be 0.7037.

## Question 12

The curve of average RMSE against k in popular movie trimmed test set is shown as below. Compared to the curve that generated in Question 10, we can figure out that the RMSE converged to a lower value which is about 0.875. This is easy to interpret since all the movies in the popular movie trimmed test set have more than two ratings which can provide the filter with more information to make more precise predictions.

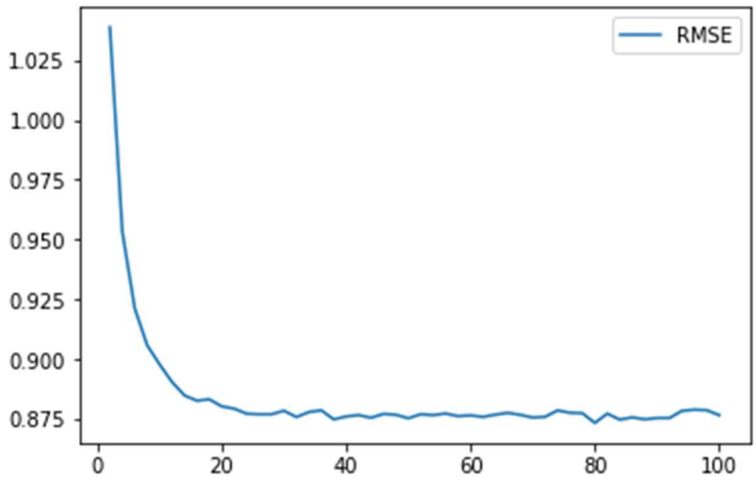



*Figure 6 . Average RMSE against k in Popular Movie Trimmed Test Set*

The minimum average RMSE is 0.8735.

## Question 13

When predicting the ratings of the movies in the unpopular movie trimmed test set, we can figure out that the RMSE is converged to a higher value comparing to untrimmed test set. This is because all the movies in the unpopular trimmed test set have less than two ratings which provide the filter with less information to make precise prediction.

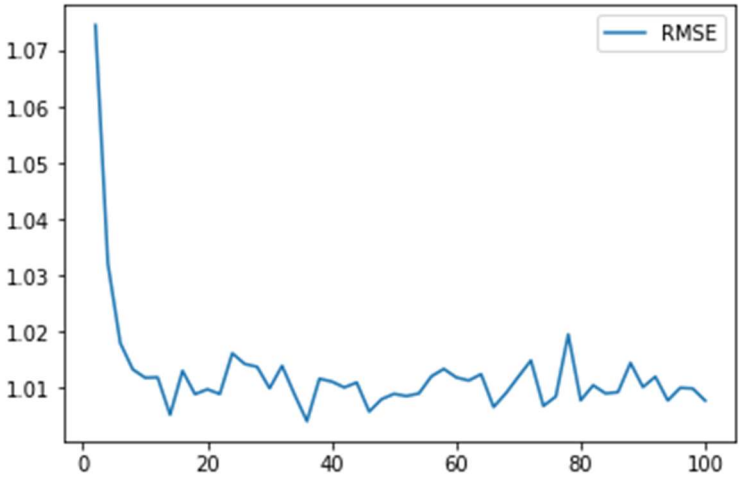



*Figure 7 . Average RMSE against k in Unpopular Movie Trimmed Test Set*

The minimum average RMSE = 1.0041

## Question 14

When predicting the ratings of the movies in the high variance movie trimmed test set, we can figure out that the RMSE is converged to an even higher value comparing to untrimmed test set. This is because all the movies in high variance trimmed test set have high variance of ratings which make the prediction more difficult so that the filter can't perform well.

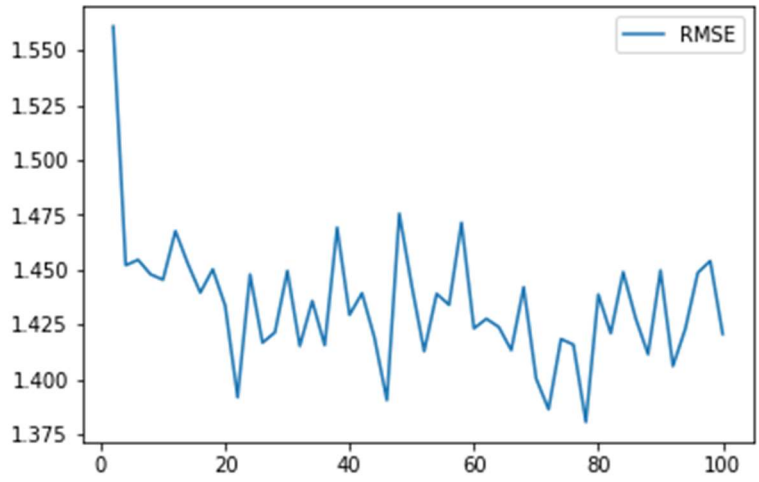



*Figure 8 . Average RMSE against k in High Variance Movie Trimmed Test Set*

The minimum average RMSE = 1.3805

## Question 15

For this problem we plot the ROC for the kNN collaborative filter designed in Question 10 with threshold values of 2.5, 3, 3.5 and 4. Here we use k = 18. As you can tell from the following figures in this section, a threshold of 2.5 is giving the largest area under curve, which could further imply it's the best threshold for the mentioned filter.

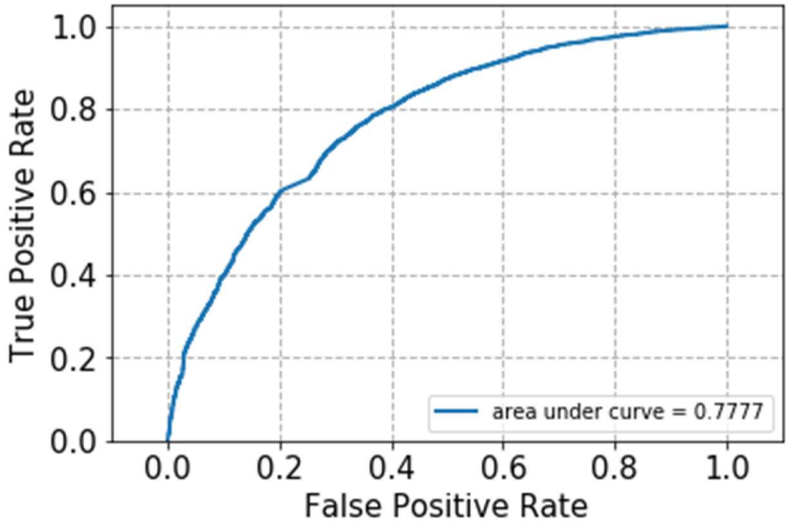



*Figure 9(a). ROC Curve with Threshold Value=2.5*

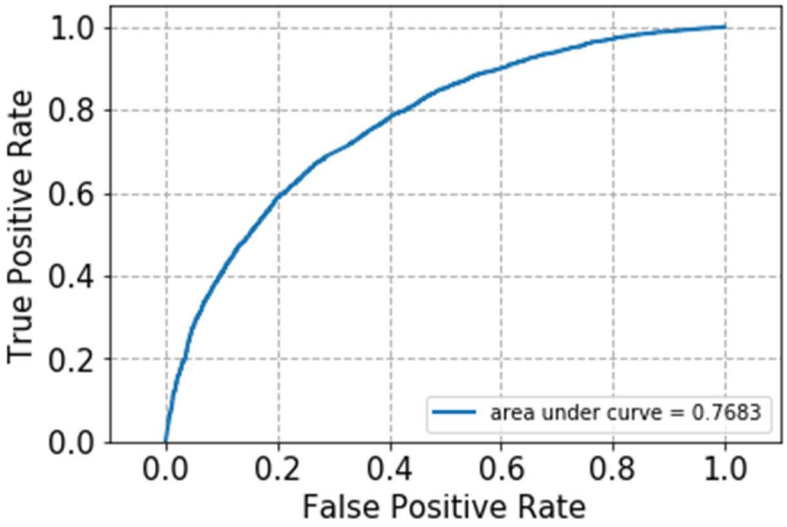



*Figure 9(b). ROC Curve with Threshold Value=3*

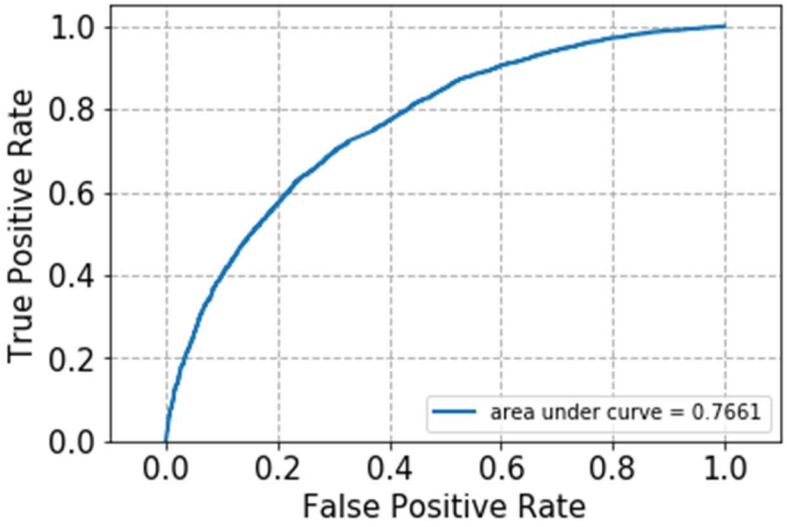



*Figure 9(c). ROC Curve with Threshold Value=3.5*

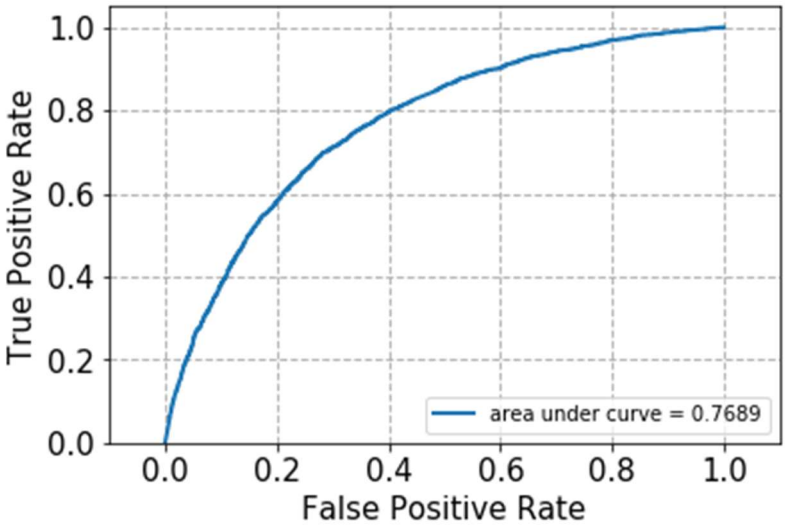



*Figure 9(d). ROC Curve with Threshold Value=4*

## Question 16

**The optimization problem by equation 5 is convex.**

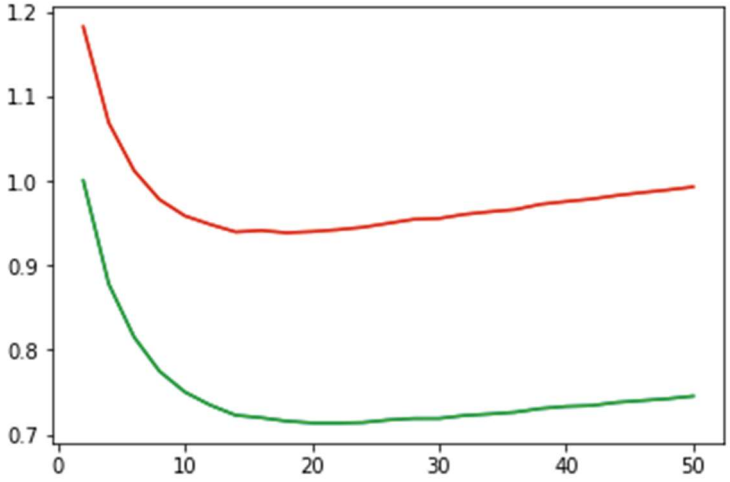
When U is fixed, Equation 5 in the problem statement becomes:

$$\min V \sum_{i=1}^{m}\sum_{j=1}^{n} W_{ij}\left(r_{ij} - (UV^T)_{ij}\right)^2 =$$

$$\min V \sum_{i=1}^{m}\sum_{j=1}^{n} W_{ij}\left(r_{ij} - U_i V_j^T\right)^2 =$$

$$\min V \sum_{i=1}^{m}\left(\sum_{j=1}^{n} W_{ij} r_{ij}^2 - V_j^T \sum_{j=1}^{n} 2W_{ij} r_{ij} U_i + (V_j^T)^2 \sum_{j=1}^{n} W_{ij} U_i^2\right)$$

## Question 17

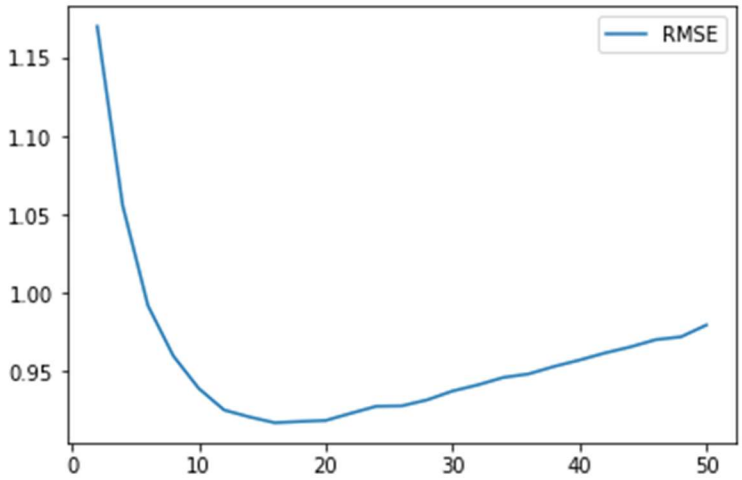



The red line represents RMSE while the green line represents MAE. The curve of RMSE and MAE against k are shown above. We can figure out that both RMSE and MAE first decrease and then increase with the increase of latent factors.

## Question 18

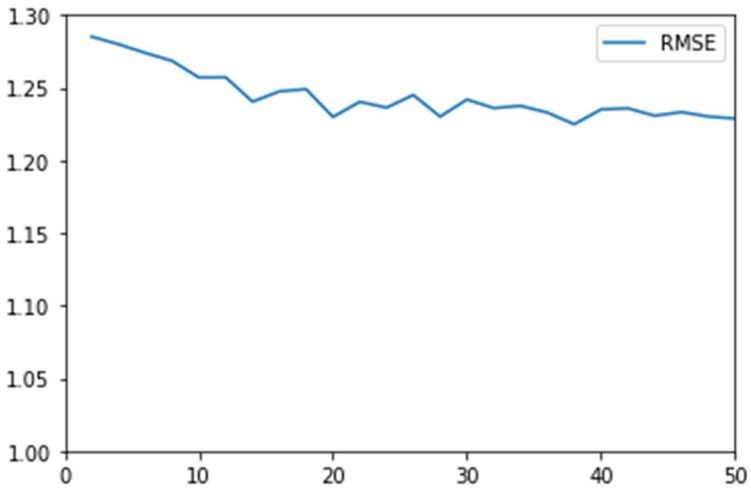
We find that the optimal number of latent factor is 18. At k=18, the minimum average MAE is 0.7158, the minimum average RMSE is 0.9388. It is interesting to note that the number of movie genres is 18, which is the same as the optimal number of latent factor.

## Question 19




We can see that after we constructed trimmed testset on popular movies, the testing result of RMSE is better than that in Question 17. The reason is that popular movies have more than 2 training rating that the model can refer to during the training. What's more, the min RMSE is smaller than that in Question 18.
Best K: 16
The min RMSE=0.9175

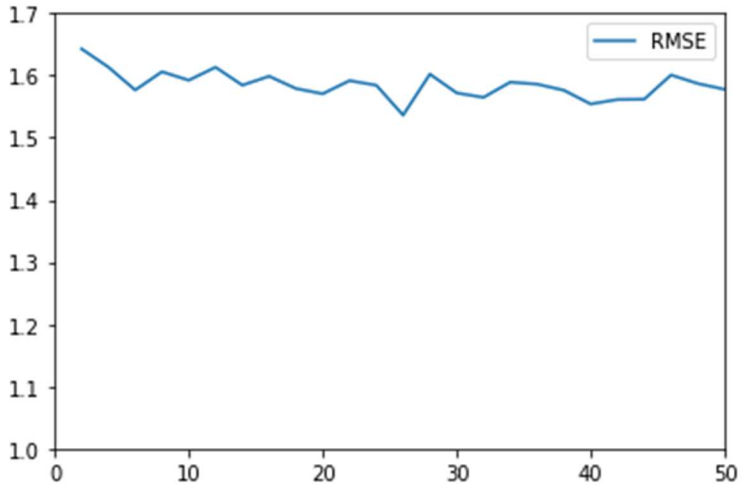
## Question 20




When predicting the ratings of the movies in the unpopular movie trimmed test set, we can figure out that the RMSE does not decrease much when it reaches about 1.25. The min RMSE is higher than that of untrimmed test set. This is because all the movies in the unpopular trimmed test set

have less than two ratings which provide the model with less information to make precise prediction.
Best k= 20
The min RMSE=1.229

## Question 21

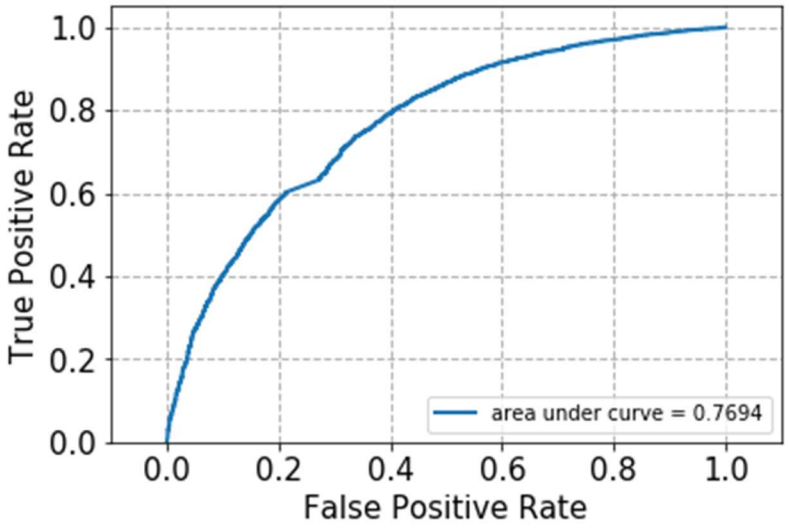



When predicting the ratings of the movies in the high variance movie trimmed test set, we can figure out that the RMSE is not decreasing a lot comparing to untrimmed test set. The lowest RMSE is still very high compared to untrimmed test set. This is because all the movies in high variance trimmed test set have high variance of ratings which make the prediction more difficult so that the filter can't perform well.
Best k=26
The min RMSE=1.5363

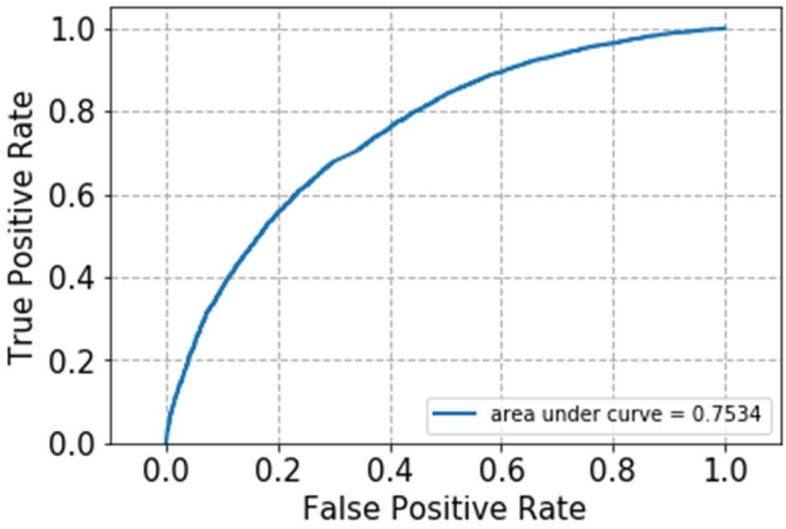
## Question 22

Below are the ROC curves at best latent factor k=18. We can see the four ROC curve under thresholds 2.5, 3, 3,5 4. From the four figures, we can conclude that the threshold 2.5 is the best because it has the highest auc value 0.7694.

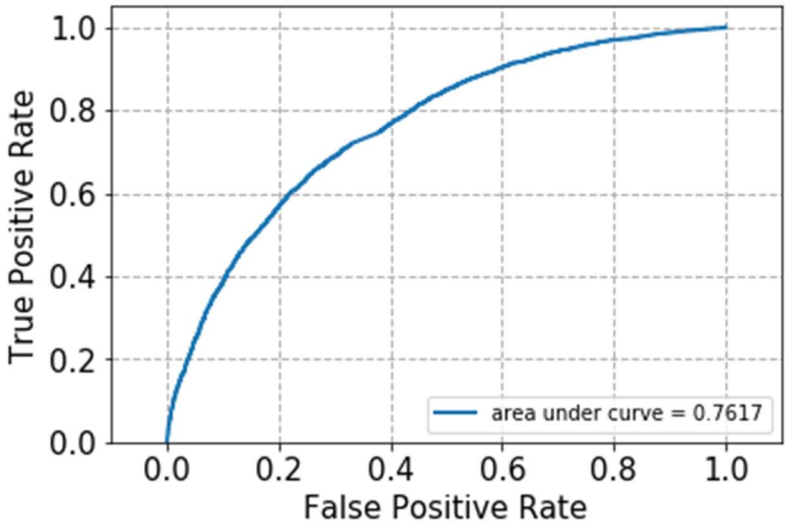
Threshold=2.5
AUC=0.7694



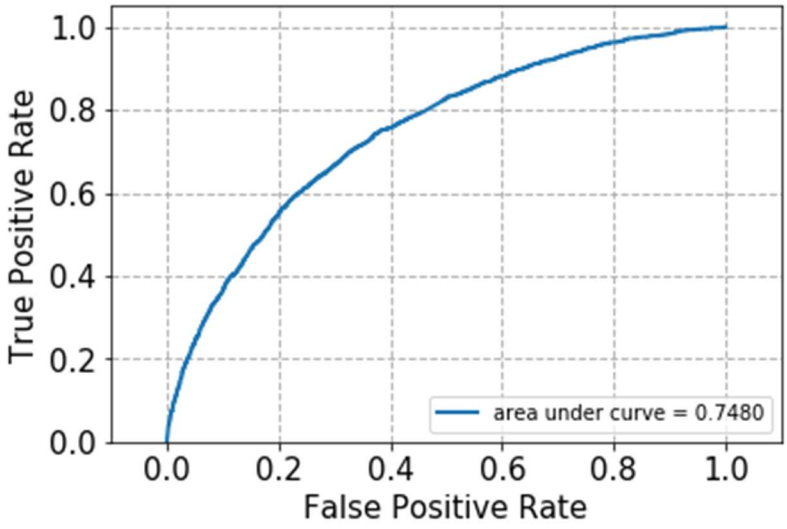

Threshold=3
AUC=0.7534




Threshold=3.5
AUC=0.7617




Threshold=4
AUC=0.7480



## Question 23

**Column 1:**

We can see that that the top 10 movies of column 1 are mostly in genres: drama and comedy.

| | idx | movieId | userId | rating | timestamp | title | genres |
|---|---|---|---|---|---|---|---|
| **0** | 5558 | 8857 | 1 | 1 | 1 | Lilith (1964) | Drama |
| **1** | 4206 | 5577 | 12 | 12 | 12 | Igby Goes Down (2002) | Comedy\|Drama |
| **2** | 2282 | 2851 | 3 | 3 | 3 | Saturn 3 (1980) | Adventure\|Sci-Fi\|Thriller |
| **3** | 6180 | 34523 | 2 | 2 | 2 | The Chumscrubber (2005) | Comedy\|Drama |
| **4** | 5533 | 8808 | 8 | 8 | 8 | Princess Diaries 2: Royal Engagement, The (2004) | Comedy\|Romance |
| **5** | 976 | 1220 | 94 | 94 | 94 | Blues Brothers, The (1980) | Action\|Comedy\|Musical |
| **6** | 5519 | 8777 | 1 | 1 | 1 | Roadkill (a.k.a. Roadkill: Move or Die) (1989) | Drama |
| **7** | 4083 | 5353 | 1 | 1 | 1 | Butterflies Are Free (1972) | Comedy\|Drama |
| **8** | 4828 | 6832 | 4 | 4 | 4 | Regarding Henry (1991) | Drama |
| **9** | 4684 | 6537 | 38 | 38 | 38 | Terminator 3: Rise of the Machines (2003) | Action\|Adventure\|Sci-Fi |

**Column 14:**

We can see that that the top 10 movies of column 14 are mostly in genres: Romance, adventure and drama.

| | idx | movieId | userId | rating | timestamp | title | genres |
|---|---|---|---|---|---|---|---|
| **0** | 8468 | 108090 | 1 | 1 | 1 | Dragon Ball: The Path to Power (Doragon bôru: ... | Action\|Adventure\|Animation\|Children |
| **1** | 759 | 939 | 1 | 1 | 1 | Reluctant Debutante, The (1958) | Comedy\|Drama |
| **2** | 8467 | 108076 | 1 | 1 | 1 | Other Shore, The (2013) | Adventure\|Documentary |
| **3** | 4592 | 6350 | 22 | 22 | 22 | Laputa: Castle in the Sky (Tenkû no shiro Rapy... | Action\|Adventure\|Animation\|Children\|Fantasy\|Sc... |
| **4** | 5887 | 27351 | 1 | 1 | 1 | Spiral (2000) | Horror |
| **5** | 1446 | 1873 | 13 | 13 | 13 | Misérables, Les (1998) | Crime\|Drama\|Romance\|War |
| **6** | 4759 | 6711 | 85 | 85 | 85 | Lost in Translation (2003) | Comedy\|Drama\|Romance |
| **7** | 7245 | 69757 | 45 | 45 | 45 | (500) Days of Summer (2009) | Comedy\|Drama\|Romance |
| **8** | 911 | 1135 | 23 | 23 | 23 | Private Benjamin (1980) | Comedy |
| **9** | 2066 | 2575 | 9 | 9 | 9 | Dreamlife of Angels, The (Vie rêvée des anges,... | Drama |

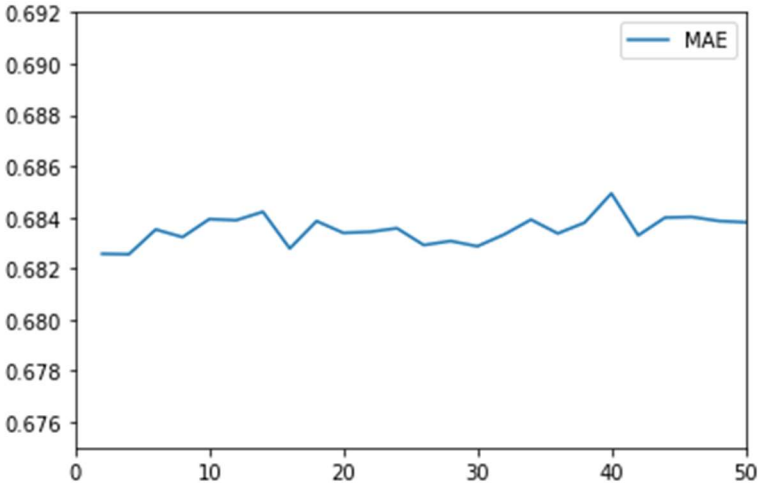
**Column 16:**

We can see that that the top 10 movies of column 16 are mostly in genres: Thriller and Horror.

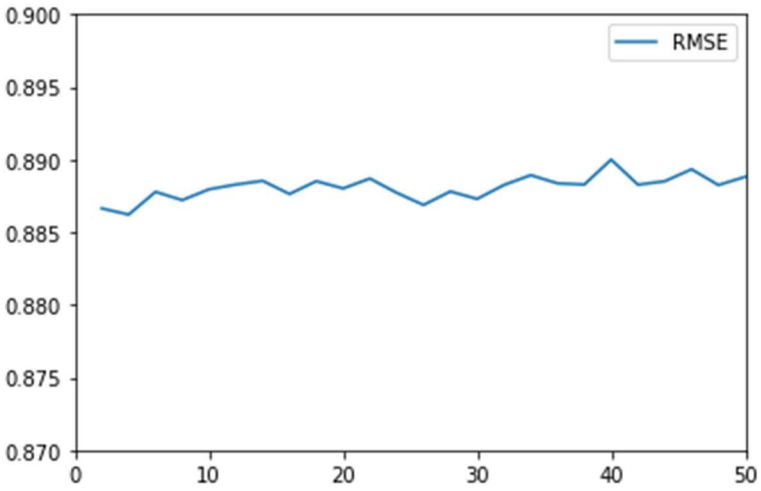
| | idx | movieId | userId | rating | timestamp | title | genres |
|---|---|---|---|---|---|---|---|
| **0** | 5393 | 8376 | 45 | 45 | 45 | Napoleon Dynamite (2004) | Comedy |
| **1** | 1010 | 1256 | 34 | 34 | 34 | Duck Soup (1933) | Comedy\|Musical\|War |
| **2** | 2716 | 3404 | 12 | 12 | 12 | Titanic (1953) | Action\|Drama |
| **3** | 1094 | 1350 | 25 | 25 | 25 | Omen, The (1976) | Horror\|Mystery\|Thriller |
| **4** | 4400 | 5962 | 2 | 2 | 2 | Body of Evidence (1993) | Drama\|Thriller |
| **5** | 4882 | 6953 | 30 | 30 | 30 | 21 Grams (2003) | Crime\|Drama\|Mystery\|Romance\|Thriller |
| **6** | 4288 | 5736 | 1 | 1 | 1 | Faces of Death 3 (1985) | Documentary\|Horror |
| **7** | 4852 | 6883 | 1 | 1 | 1 | Sylvia (2003) | Drama\|Romance |
| **8** | 4737 | 6658 | 5 | 5 | 5 | 10 (1979) | Comedy\|Romance |
| **9** | 3188 | 3984 | 23 | 23 | 23 | Diamonds Are Forever (1971) | Action\|Adventure\|Thriller |

Since we choose latent factor k=20 and there are 18 movie genres. From the above columns, we can see that each column mainly represents a small set of movie genres. We can conclude that when the latent factor is close to the number of movie genres, the collaboration filter will work better. The latent factor is highly related to the number of movie genres.

## Question 24




*24(1) Figure for MAE graph*




*24(2) Figure for RMSE graph*

In this problem, as we have observed that both of the RMSE and MAE are relatively stable compare to the K-NN and NNMF, which means that MF performs better in small values of k. Further, the curve does not vary too much across all k, so we can pick a relatively small k for our latent factor.

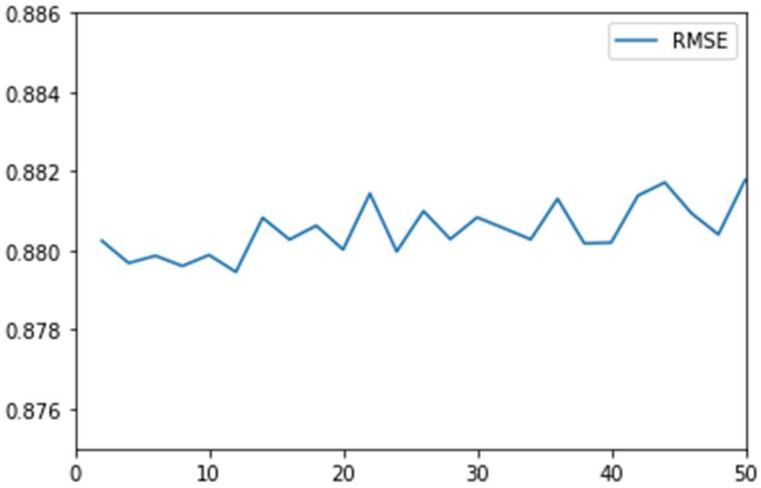
## Question 25

The overall pattern of the MAE and RMSE shows does not change too much crossing the whole k values from 2 to 50, while there is a drop of RMSE on k =24, so we choose to use k=24 as latent factor.

Minimum average RMSE: 0.8877

Minimum average MAE: 0.6836

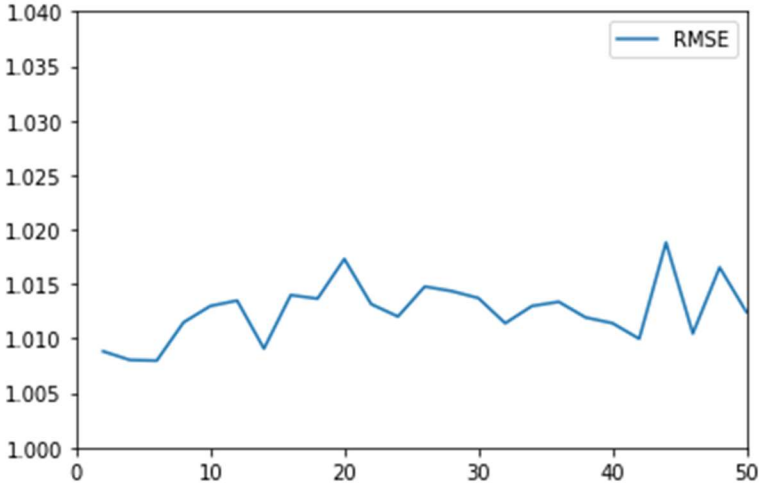
## Question 26




*26 Figure for RMSE from k=2~50, popular set*

The above graph is for the analysis of popular set, as we know that biased MF has less error in small k, so the graph is relatively flat for the whole plot. Further, the average RMSE is the smallest among the three sets: popular, unpopular, high variance, since this data set contains the user which has more than 2 ratings and thus the data are more reliable. Minimum RMSE occurs at k =12 since the value tends to increase after k=12. And the value of average RMSE = 0.879461736726

## Question 27

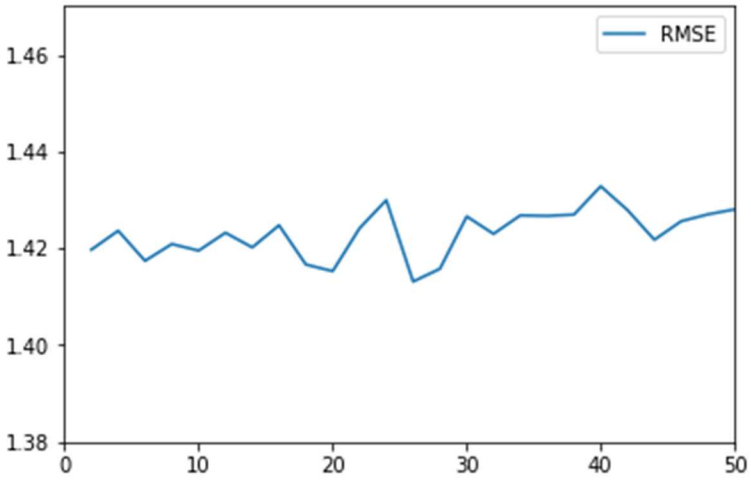



*27 Figure for RMSE from k = 2~50, unpopular set*

For the unpopular set, the graph is similar to that of popular set, which is plat over all k, while the average error tends to be larger. The reason for the larger error is that for the unpopular

movie, only few ratings occurs, which makes the data not so reliable compare to that of popular set. Minimum RMSE occurs at k = 14 since the value tends to increase after k=14. And the value of average RMSE = 1.00907569146

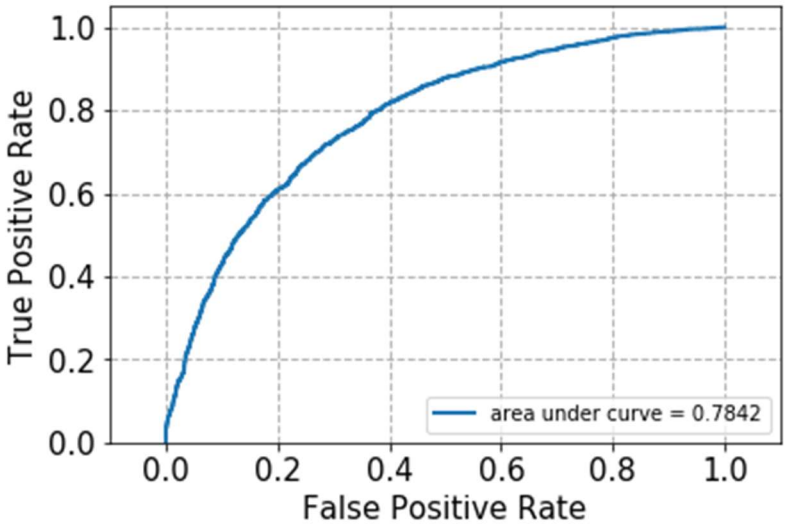
## Question 28



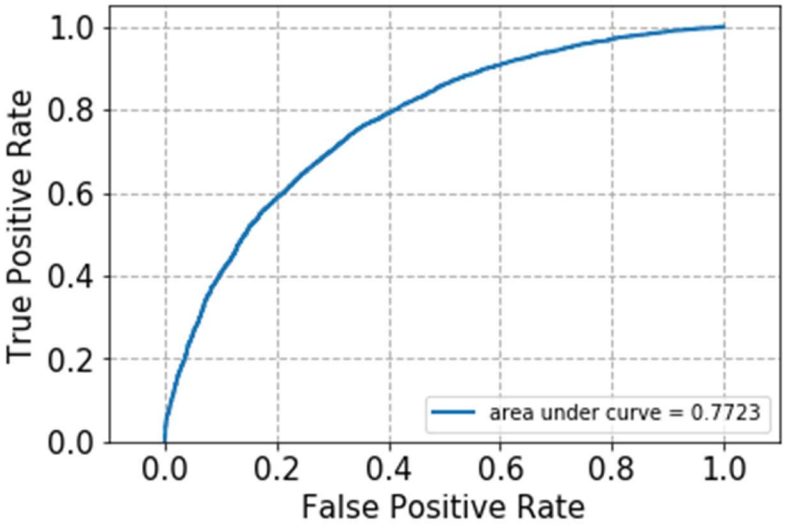

*28 Figure for RMSE from k = 2~50, high variance*

For high variance set, there are more ratings per movie among all users, but the ratings themselves have high variance, which leads to a higher error for all k in the plot. The whole graph has the similar pattern as the previous two graphs. Minimum RMSE occurs at k = 20 since the value tends to increase after k=20. And the value of average RMSE = 1.41527817595

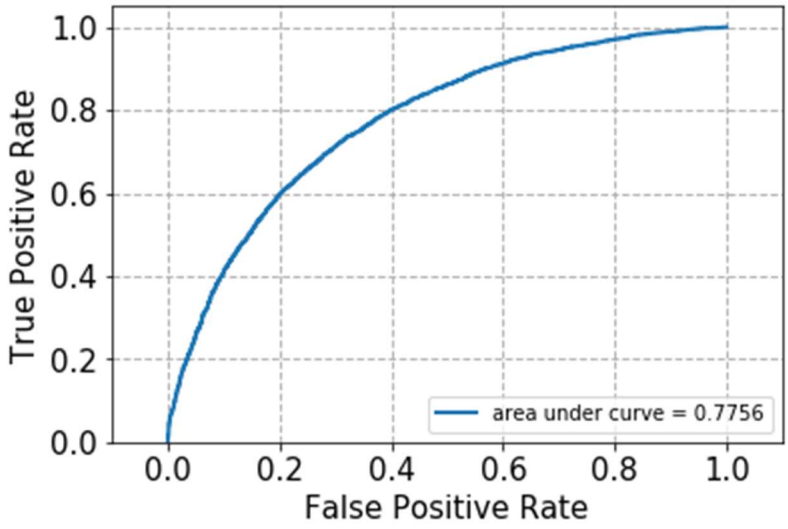
## Question 29



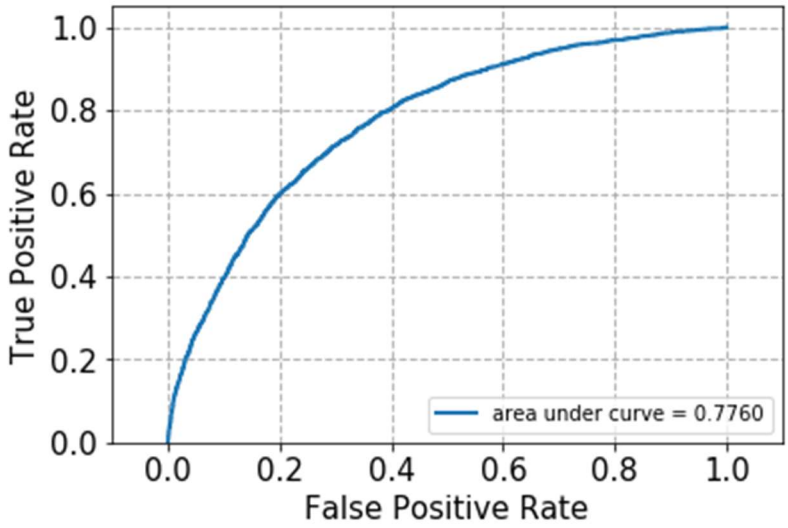

*29(1) Threshold = 2.5*




*29(2) Threshold = 3*




*29(3) Threshold = 3.5*




*29(4) Threshold = 4*

For the four graphs as above, we want to know which threshold makes the filter performs better. The standard is that, as the graph reach to stable states faster, the graph performs better, so the factor we care is the area under curve, where the rating of 2.5 tends to be better than all of the rest threshold, which means the score more than 2.5 over 5 is the illustration of "like".

## Question 30

The naive collaborative filter is based on the mean value ui of all the ratings for each user i. To do this, we first compute all the mean values of each user on the whole dataset. Then we apply 10-fold cross validation to deal with 10 different testset and find the mean rmse as follows.

**rmse_mean=0.9554**

## Question 31

After trimmed testset to popular movies, the mean rmse is 0.9521, which is decreased. The reason is that popular movies have more possibility to contribute to user's average rating.

**rmse_mean=0.9521**

## Question 32

After trimmed testset to unpopular movies, the mean rmse is 1.0092, which is higher than untrimmed testset. The reason is that unpopular movies have less possibility to contribute to user's average rating.

**rmse_mean=1.0092**

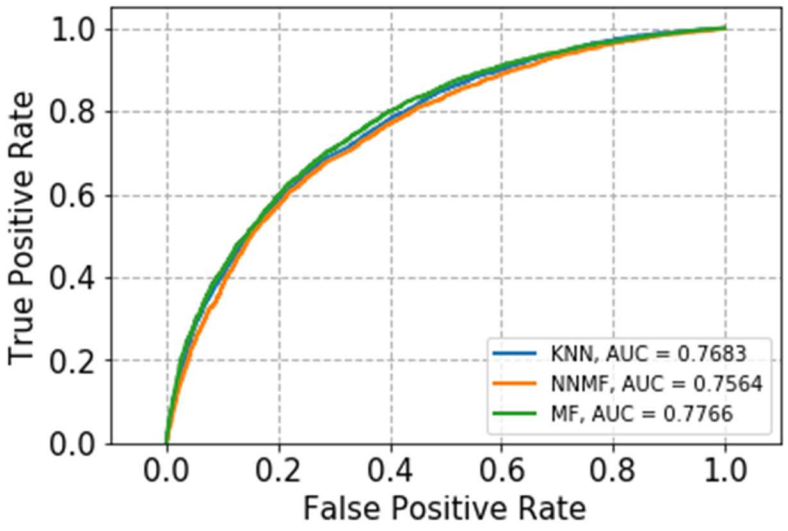
## Question 33

After trimmed testset to high variance movies, the mean rmse is 1.4403, which is higher than untrimmed testset. The reason is that high variance movies may have ratings that are far from user's average rating.

**rmse_mean=1.4403**

## Question 34

The performance of the three filters in predicting the ratings of the movies is shown as below. From the figure we can tell that the three filters have similar performance and MF with bias-based collaborative filter performs slightly better than the other two.

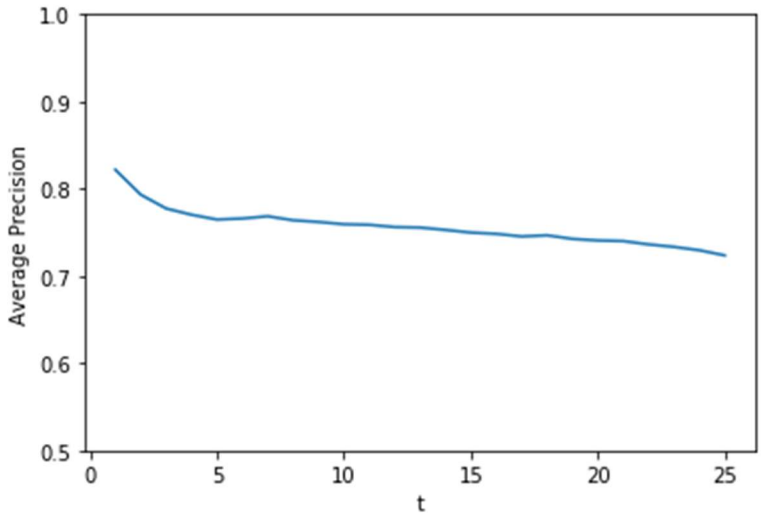


## Question 35

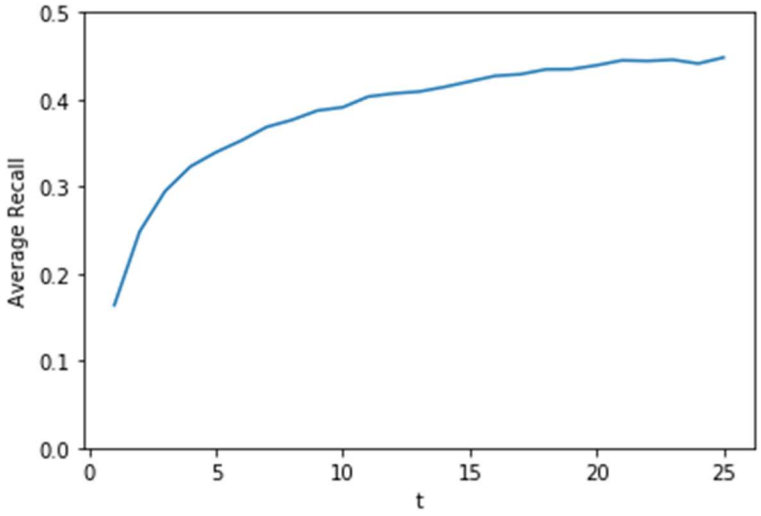
Precision is the ratio of the number of movies that the recommendation system has correctly predicted to the total number of movies that the recommendation system predicts (mix of correct and wrong predictions). In other words, it is how precise of the prediction. Recall is the ratio of the number of movies that the recommendation system has correctly predicted to the number of movies that a user actually likes.

## Question 36

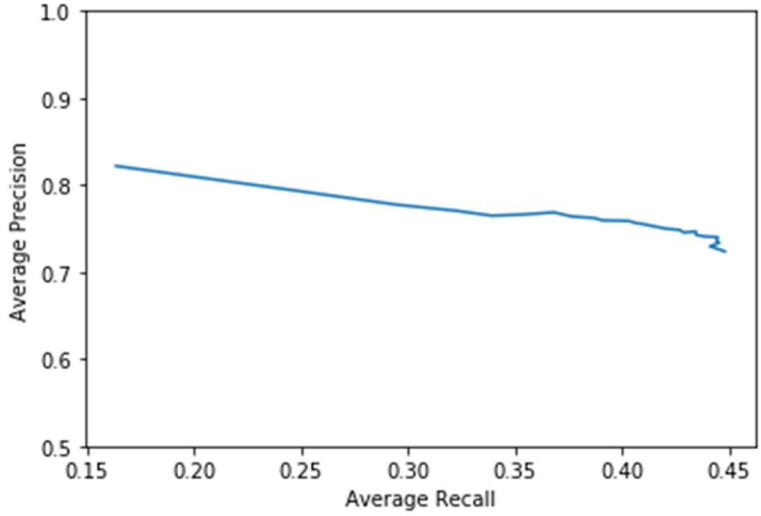
Average precision against t (the size of the recommended list) for the ranking obtained using k-NN collaborative filter predictions is shown as below. We can figure out that the curve is of descending trend. This can be explained as follows: since the number of movies that a user actually likes is fixed, the precision of the filter's prediction will slightly decrease with the increase of the number of recommended movies, which is also the denominator.



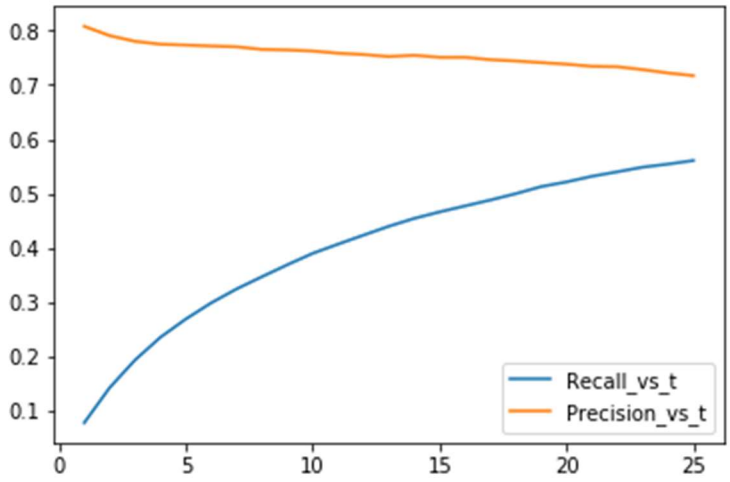

Average recall against t for the ranking obtained using k-NN collaborative filter predictions is shown as below. The curve is of ascending trend because the denominator which corresponds to the number of movies that a user actually likes is fixed and the numerator which corresponds to the number of movies that the filter has correctly predicted is increasing when t (the size of recommended list) increases.



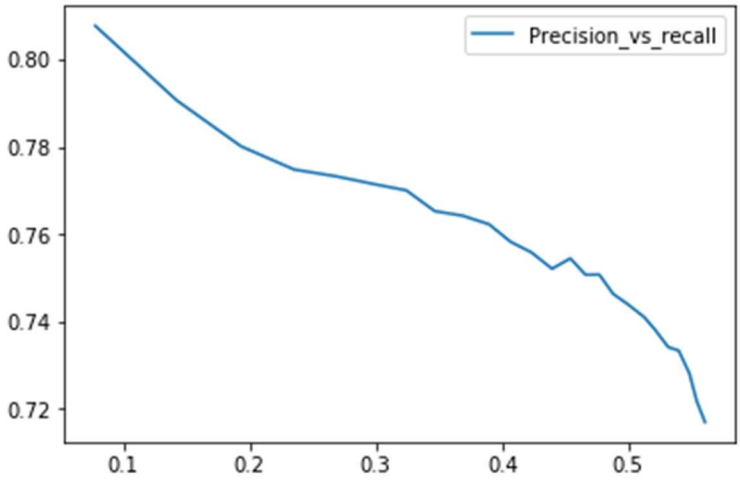

The curve of average precision against average recall is of descending trend. It shows the tradeoff between precision and recall for different threshold.



## Question 37



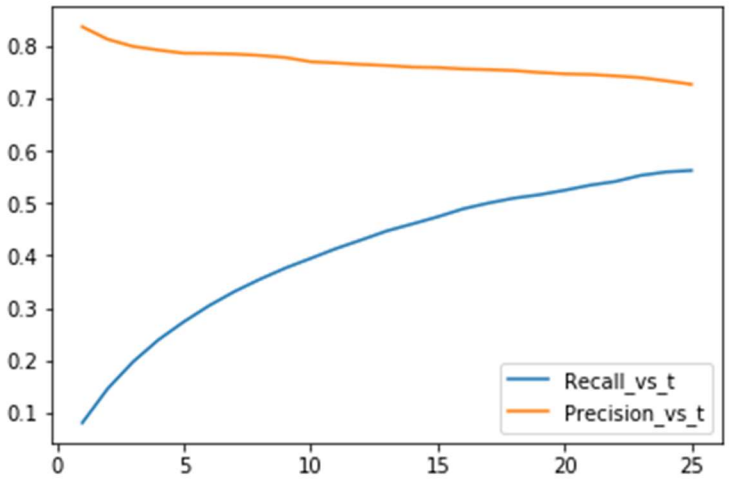

*37(1) Figure that represent the recall verse t and precision verse t*



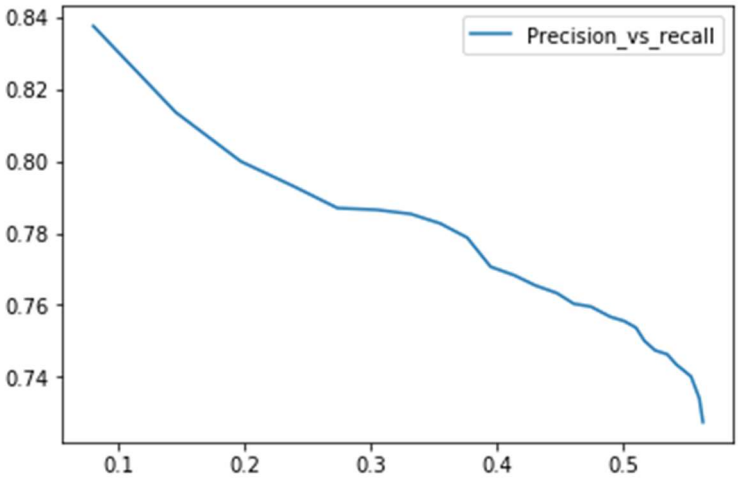

*37(2) Figure represent the Precision versus Recall*

The precision verse t tends to show a pattern that decrease with the increasing value of t, where the average precision decreasing from about 0.8 to 0.7. Moreover, the recall rate shows an increasing pattern as t increases. The pattern is reasonable for those two. As we increase the number of recommendation list by one, we cannot guarantee a movie marked “like” has been recommended, so the precision should decrease, while the recall should increase since the number of liked movie in the increasing recommendation list is always increasing but the number of ground truth is fixed. For the precision versus recall, as average recall increases, the average precision decreases.

## Question 38




*38(1)Figure that represent the recall verse t and precision verse t*




*38(2) Figure represent the Precision versus Recall*

The precision verse t tends to show a pattern that decrease with the increasing value of t, where the average precision decreasing from about 0.86 to 0.73. In addition, the recall rate shows an increasing pattern as t increases. The pattern is reasonable for those two. As we increase the number of recommendation list by one, we cannot guarantee a movie marked “like” has been recommended, so the precision should decrease, while the recall should increase since the number of liked movie in the increasing recommendation list is always increasing but the number of ground truth is fixed. For the precision versus recall, as average recall increases, the average precision decreases.

## Question 39

In this final part of the project, we plot the precision-recall curve obtained in the previous questions in the same figure as shown below. It seems like from the figure that the most relevant recommendation list generated was by NNMF algorithm with relatively higher values both in precision and recall. For the comparison between kNN and MF with bias, it is hard to tell from this figure which one is better. Maybe some other metrics should be introduced for a more accurate measure of the robustness of the predict algorithms.