ECE 219 Project3

Evelyn Chen UID: 704332587
Jack Gong UID: 005025415
Jiuru Shao UID:204288539
Haoxiang Zhang UID:104278461

February 2018

1 Introduction

In this project, we implemented collaborative filtering models for recommendations. We implemented and analyzed neighborhood-based and model-based collaborative filtering.

Specifically, we built a recommendation system to predict the ratings of the movies in the MovieLens dataset. The data set was extracted from an excel sheet and was generated to a sparse matrix. Details of the implementation and further analysis and comparison can be seen in the following report.

2 Q1

The sparsity of the movie rating dataset is:

0.01644

3 Q2

Here is the plot showing the frequency of the rating values. Bin width is 0.5. We don't count ratings of 0. The shape of the histogram is skewed to the left, with more higher ratings (i.e. 4 or 5) and fewer lower ratings (i.e. 1 or 2).

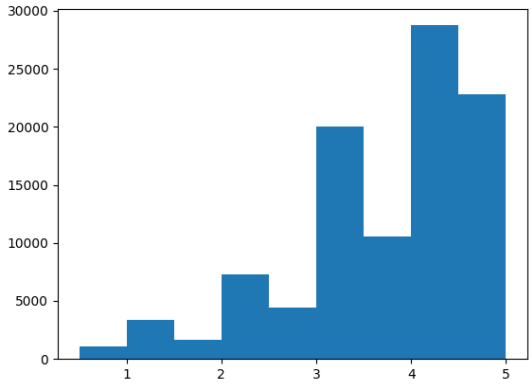


Figure 1: **Frequency of the rating values**

4 Q3

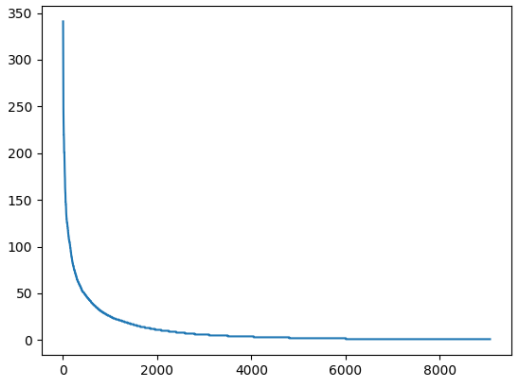


Figure 2: **Distribution of ratings among movies**

5 Q4

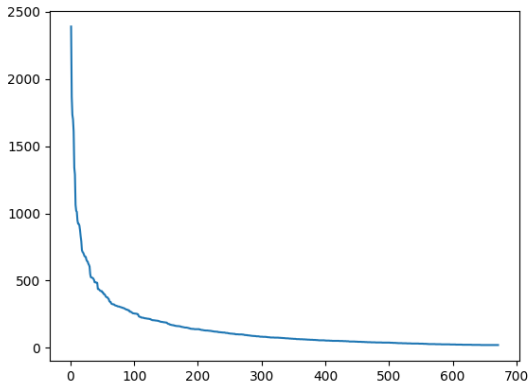


Figure 3: **Distribution of ratings among users**

6 Q5

As we can see from the figure in Question 3, very few movies have more than 50 ratings, and most of the movies have less than 10 ratings. These features imply that the rating matrix is sparse, which is the main challenge in the recommendation process.

7 Q6

Here is the plot showing the variance of the rating values received by each movie. Bin width is 0.5.

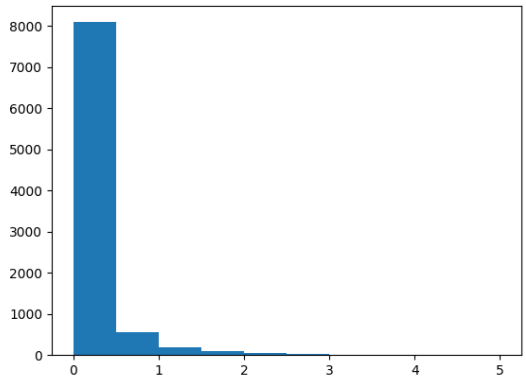


Figure 4: **Variance of the rating values received by each movie**

From the above figure, it is obvious that most of the movies receive similar ratings from users (i.e. the variance value is less than 0.5). Very few movies have high variance of ratings among different users. This means that users have similar taste for most of the movies, which will make our recommendation process work at most of the time.

8 Q7

$$\mu_u = \frac{1}{|I_u|} \sum_{k \in I_u} r_{uk} \tag{1}$$

9 Q8

$I_u \cap I_v$ means the set of item indices for which ratings have been specified by both user v and u. It can be Ø when none of the items rated by u has been rated by v and vise versa.

10 Q9

Mean-centering effectively produces user v's preference on item j compared to his or her average rating. For instance, if user v tends to give high ratings, we could produce an unbiased rating by subtracting v's mean rating μ_u from the actual rating. ‘

11 Q10

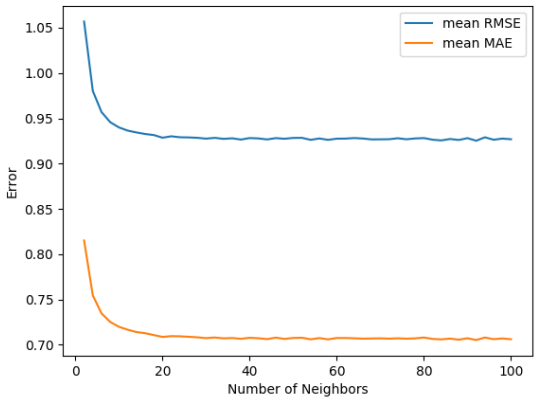


Figure 5: **The plot of the number of neighbors k vs average MAE and RMSE, for k = 2 to 100.**

12 Q11

For average RMSE, the minimum $k = 22$ and RMSE $= 0.928$, and for average MAE, the minimum $k = 26$ and MAE $= 0.708$.

13 Q12

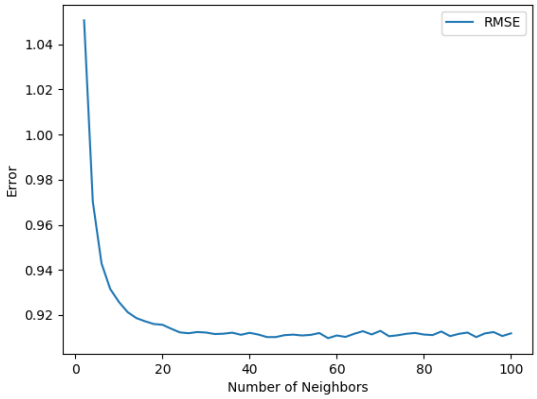


Figure 6: **The plot of the number of neighbors k vs average RMSE for prediction on popular movies, for $k = 2$ to 100.**

The smallest RMSE is 0.911 achieved at $k \geq 32$. In other words, RSME converges to 0.911.

14 Q13

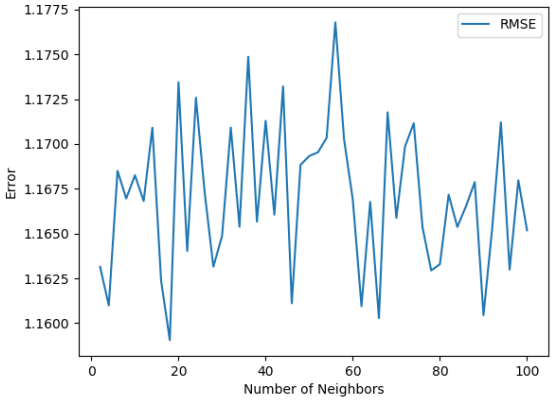


Figure 7: **The plot of the number of neighbors k vs average RMSE for prediction on unpopular movies, for $k = 2$ to 100.**

The smallest RMSE is 1.159 achieved at $k = 19$.

15 Q14

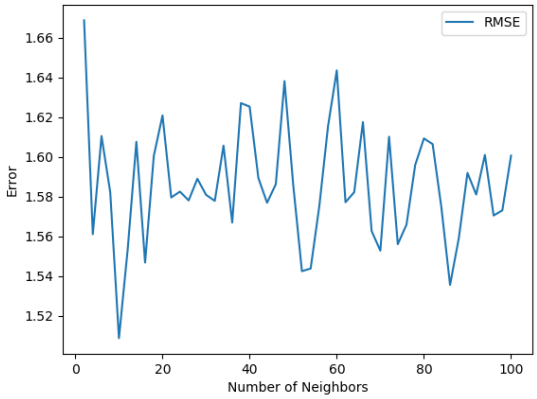
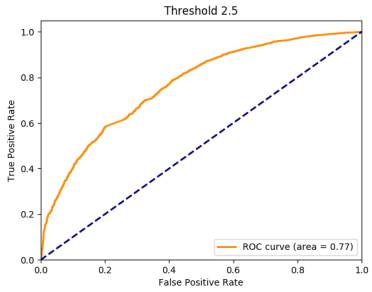


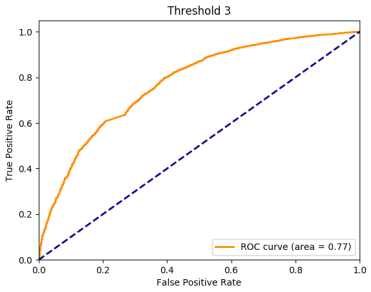
Figure 8: **The plot of the number of neighbors k vs average RMSE for prediction on high variance movies, for $k = 2$ to 100.**

The smallest RMSE is 1.509 achieved at $k = 10$.

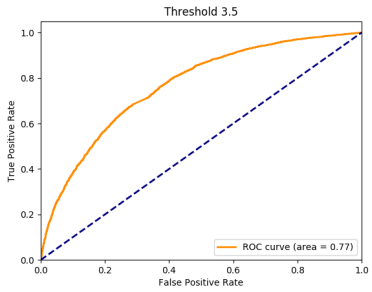
16 Q15



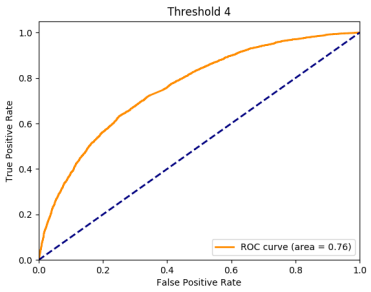
(a) Threshold 2.5



(b) Threshold 3



(c) Threshold 3.5



(d) Threshold 4

Figure 9: **ROC curves for kNN where $k = 22$ as found in question 11**

Threshold	2.5	3	3.5	4
AUC value	0.77	0.77	0.76	0.75

Table 1: **AUC value for kNN where $k = 22$ as found in question 11**

Above plots of ROC cureves and the area under the curve (AUC) value is the result of using $k = 22$ found in question 11.

17 Q16

Yes. By taking second order partial derivative with respect to V, when U is fixed, we are able to derive the Hessian matrix, which describes the local curvature of this optimization equation(5).

18 Q17

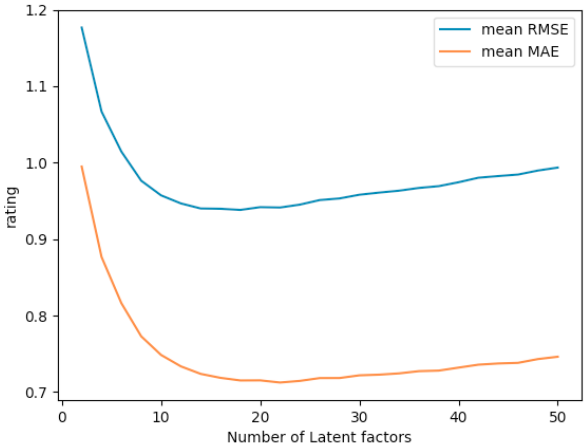


Figure 10: **The plot of the number of latent factors k vs average MAE and RMSE, for k = 2 to 50 for NNMF-collaborate based filter**

19 Q18

min average MAE = 0.711: k = 22; min average RMSE = 0.935: k = 18; There are 18 genres as listed on the readme of dataset. Thus, the optimal number of latent factors of the minimum average RMSE is same as the number of movie genres, but that of the minimum average MAE is not same as the number of movie genre.

20 Q19

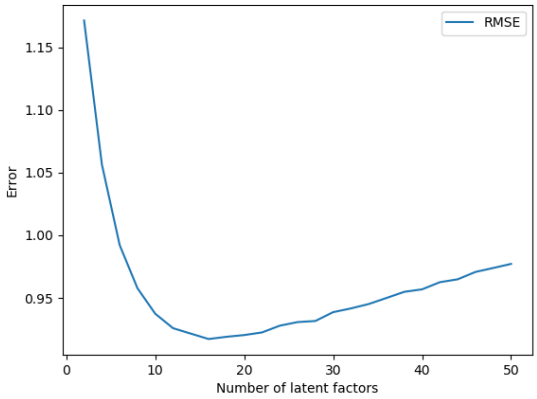


Figure 11: **The plot of the number of latent factors k vs average RMSE for prediction on popular movies, for $k = 2$ to 50 for NNMF-collaborate based filter.**

The minimum averge RMSE is 0.917 at $k = 16$.

21 Q20

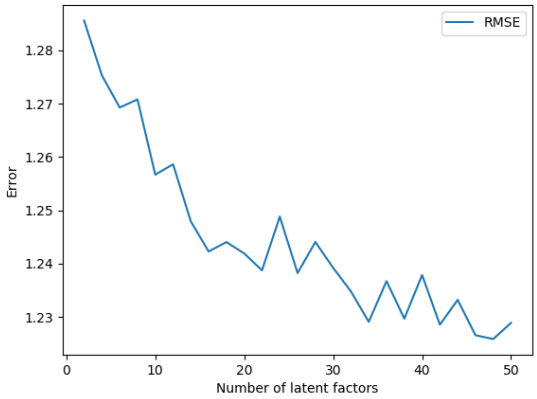


Figure 12: **The plot of the number of latent factors k vs average RMSE using NN for prediction on unpopular movies, for $k = 2$ to 50 for NNMF-collaborate based filter.**

The minimum averge RMSE is 1.225 at $k = 48$.

22 Q21

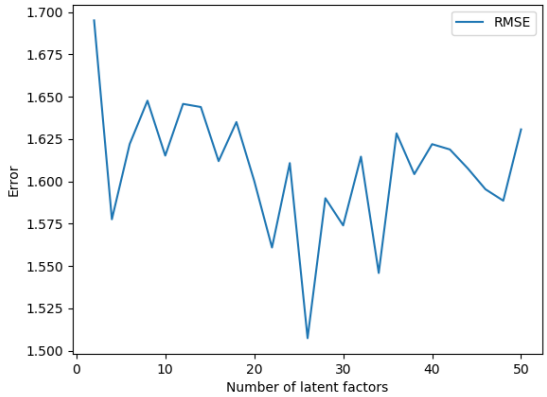
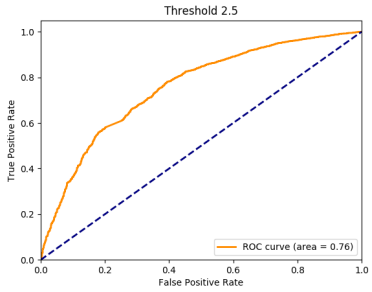


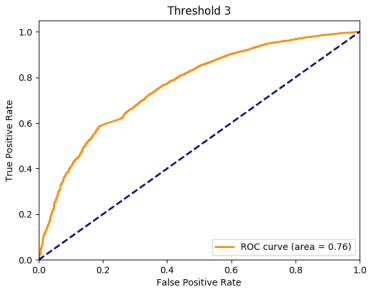
Figure 13: **The plot of the number of latent factors k vs average RMSE using NN for prediction on high variance movies, for $k = 2$ to 50 for NNMF-collaborate based filter.**

The minimum RMSE is 1.507 achieved at $k = 26$.

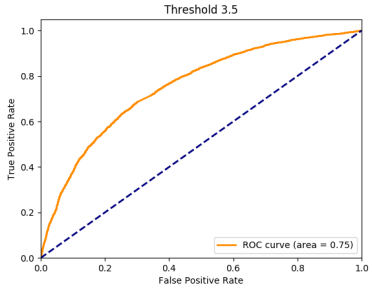
23 Q22



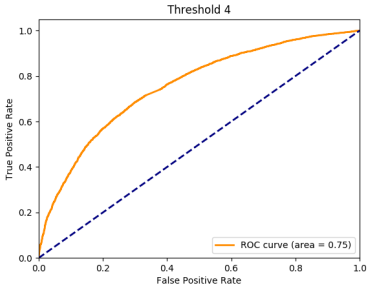
(a) Threshold 2.5



(b) Threshold 3



(c) Threshold 3.5



(d) Threshold 4

Figure 14: **ROC curves for NNMF where optimal number of latent factors is 18 as found in question 18**

Threshold	2.5	3	3.5	4
AUC value	0.77	0.78	0.76	0.76

Table 2: **AUC value for NNMF where optimal number of latent factors is 18 as found in question 18**

Above plots of ROC cureves and the area under the curve (AUC) value is the result of using 18 as the optimal number of latent factors found in question 18.

24 Q23

Basically, question 23 is designed to explore the interpretability of NNMF. Here is the table showing the genres of the top 10 movies of chosen columns.

Column 14	Column 18
Documentary	Comedy—Musical—Romance
Horror—Thriller	Drama
Drama—Thriller	Drama
Documentary	Documentary
Drama	Comedy
Action—Adventure—Sci-Fi	Comedy
Crime—Drama—Thriller	Comedy—Mystery
Crime—Drama—Mystery—Thriller	Comedy—Horror—Sci-Fi—Thriller
Comedy—Crime	Comedy
Horror—Thriller	Action—Adventure

Table 3: **Genres of top 10 movies of chosen columns**

Genres of column 14 can be concluded as **Thriller or Crime** since most of top 10 movies of column 14 belong to these two categories. Similarly, genres of column 18 can be concluded as **Comedy or Drama**.

Although top 10 movies of each column belong to a small collection of genres instead of a particular genre, there is a connection between latent factors and movie genres. Latent factors (20 here) group all movies into 20 collections of genres.

25 Q24

Here, we design a matrix factorization with bias collaborate filter to predict the ratings of the movies.

This figure plots the average RMSE/MAE against k. For this question, we use the default value for the regularization parameter.

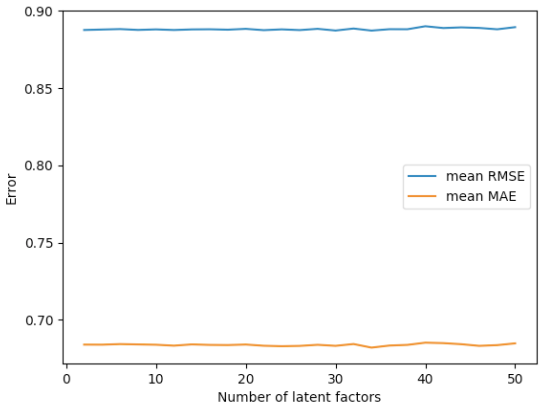


Figure 15: **The plot of the number of latent factors k vs average MAE and RMSE, for k = 2 to 50**

However, it is difficult to see the trend for RMSE and MAE from the combined graph. Therefore, we also plot them separately. Note that the y-axis range changed, and thus the bigger different in y-values in these graphs.

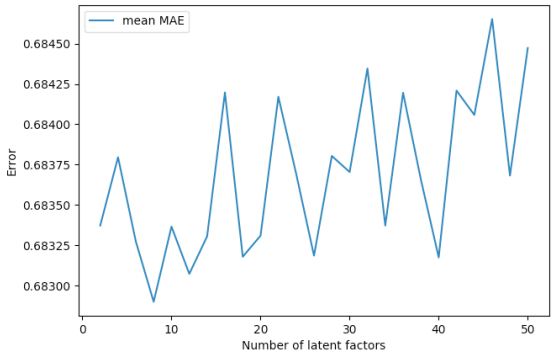


Figure 16: **The plot of the number of latent factors k vs average MAE, for k = 2 to 50**

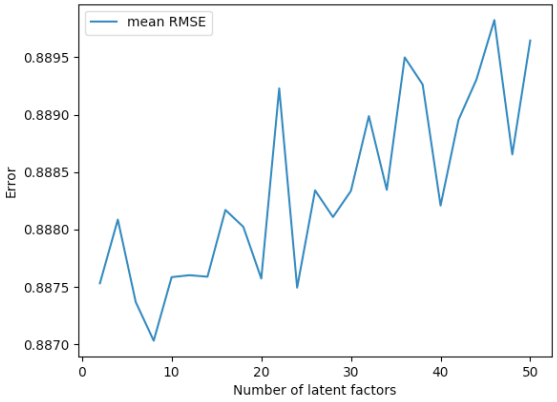


Figure 17: **The plot of the number of latent factors k vs average RMSE, for k = 2 to 50**

26 Q25

We want to find the optimal number of latent factors from the plot in question 24. Optimal number is the value of k that gives minimum average RMSE/MAE. From the separately plotted MAE and RMSE from problem 24, we see that around k = 8 for latent factor is the optimal number for both MAE and RMSE for MF with bias.

Optimal number: k=8 for both MAE and RMSE
Min average RMSE: 0.8870
Min average MAE: 0.68289

27 Q26

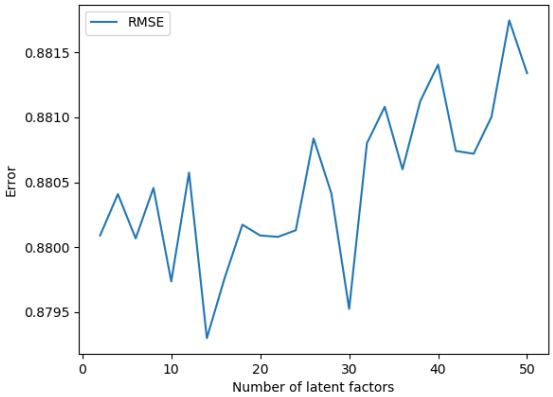


Figure 18: **The plot of the number of latent factors k vs average RMSE for prediction on popular movies, for k = 2 to 50.**

The minimum RMSE is 0.879 achieved at $k = 14$.

28 Q27

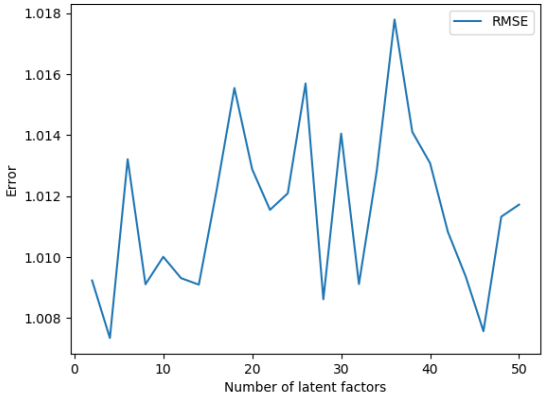


Figure 19: **The plot of the number of latent factors k vs average RMSE for prediction on unpopular movies, for k = 2 to 50.**

The minimum RMSE is 0.879 achieved at $k = 4$.

29 Q28

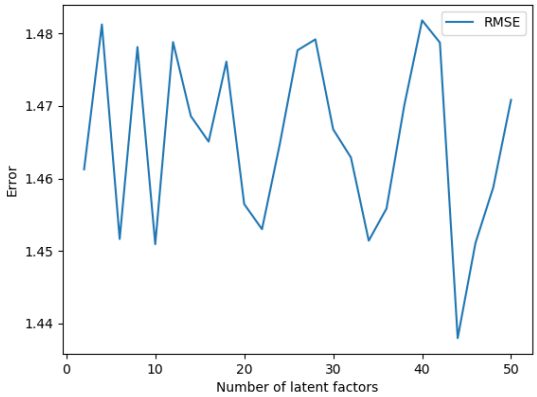
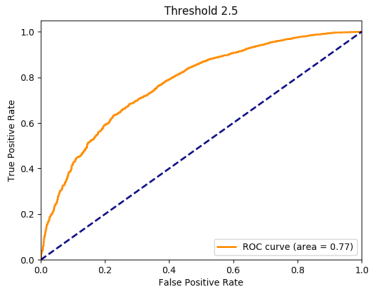


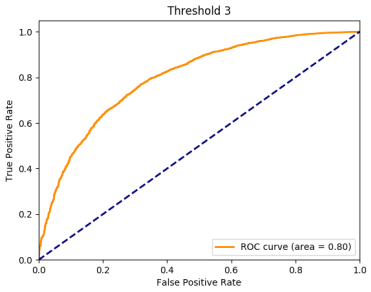
Figure 20: **The plot of the number of latent factors k vs average RMSE for prediction on high variance movies, for k = 2 to 50.**

The minimum RMSE is 1.438 achieved at $k = 44$.

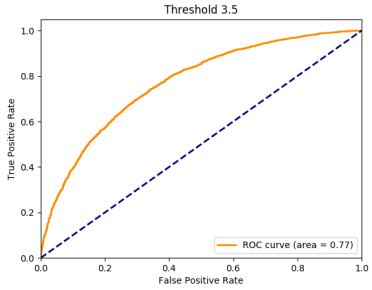
30 Q29



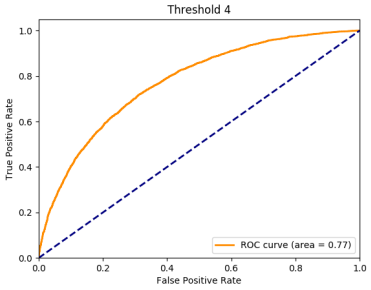
(a) Threshold 2.5



(b) Threshold 3



(c) Threshold 3.5



(d) Threshold 4

Figure 21: **ROC curves for SVD where optimal number of latent factors is 8 as found in question 25**

Threshold	2.5	3	3.5	4
AUC value	0.79	0.80	0.77	0.78

Table 4: **AUC value for SVD where optimal number of latent factors is 8 as found in question 25**

Above plots of ROC curves and the area under the curve (AUC) value is the result of using 8 as the optimal number of latent factors found in question 25.

31 Q30

The average RMSE for naive collaborative filter using 10-fold cross validation is 0.9554.

32 Q31

The average RMSE for the popular trimmed test set using 10-fold cross validation is 0.9521. This value is slightly lower than that of the non trimmed test set. This makes sense because people usually have similar tastes for popular movies.

33 Q32

The average RMSE for the unpopular trimmed test set using 10-fold cross validation is 1.0096. This value is slightly higher than that of the non trimmed test set. This makes sense because people usually have different tastes for unpopular movies.

34 Q33

The average RMSE for the high variance trimmed test set using 10-fold cross validation is 1.4986. This values is the highest one among results of question 30 to question 33. This makes sense because we are using the high variance trimmed test set.

35 Q34

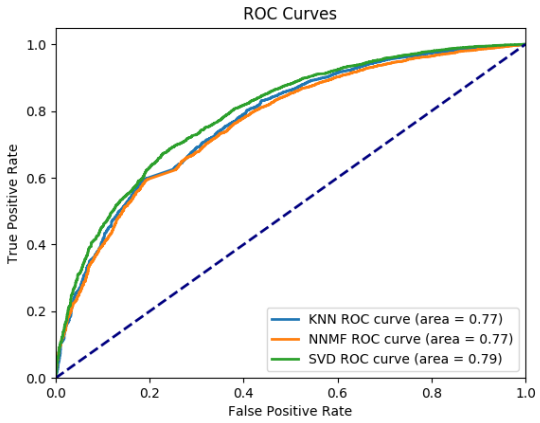


Figure 22: **ROC curves for the KNN, NNMF and SVD in the same figure**

	kNN	NNMF	SVD
AUC value	0.77	0.77	0.79

Table 5: **AUC value for different collaborative filters**

From above figure and table, all best kNN, NNMF and SVD with bias based collaborative filgres have **similar performances**.

36 Q35

Precision:
The fraction of recommended items that are liked by the user.

Recall:
The fraction of items liked by the user that are in the recommended set.

37 Q36

For problem 36, we plot precision vs t, recall vs t, and precision vs recall for the ranking from k-NN collaborative filter predictions.

The number of neighbor k used here is from Q11. We use k = 30 here.

The graph for precision vs t has a downward trend. It shows that higher t, lower precision. The bigger the recommended set, the smaller portion of ground truth it has. This is because with bigger t, the ratings are lower, and less ground truth the recommended set contains.

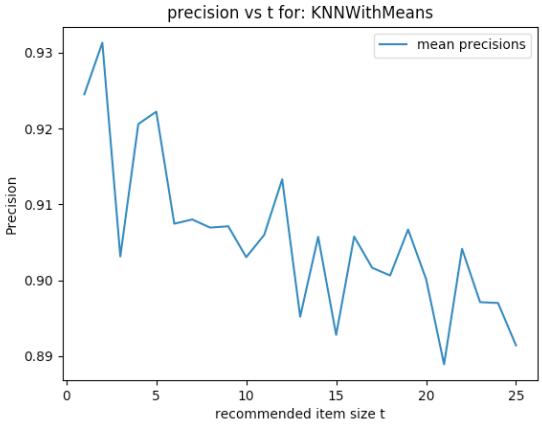


Figure 23: **The plot of the precision vs number of recommended item t, for t = 1 to 25.**

The graph for recall vs t has a upward but curved trend. It shows that higher t, higher recall, but the incremental increase decreases with higher t. The bigger the recommended set, the more ground truth it can find.

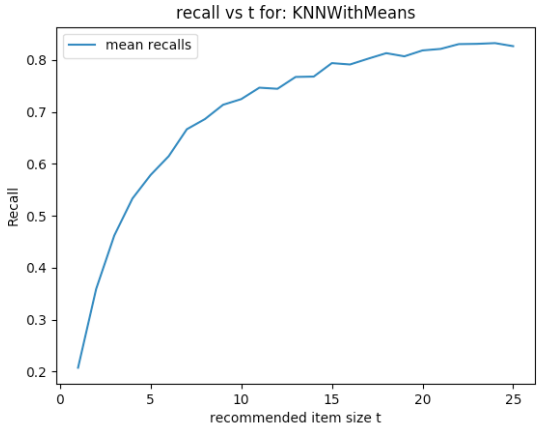


Figure 24: **The plot of the recall vs number of recommended item t, for t = 1 to 25.**

The graph for precision vs recall has a downward trend. Higher the recall, lower the precision.

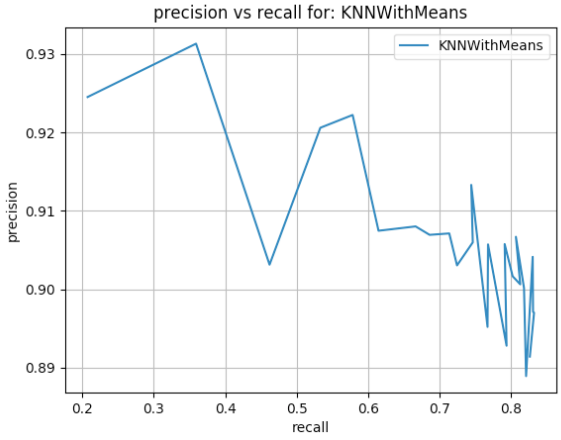


Figure 25: **The plot of the precision vs recall**

38 Q37

For problem 37, we plot precision vs t, recall vs t, and precision vs recall for the ranking from NNMF collaborative filter predictions.

The number of factors k used here is from Q18. We use k = 18 here.

The graph for precision vs t has a downward trend. It shows that higher t, lower precision. The analysis of the trend is same as problem 36.

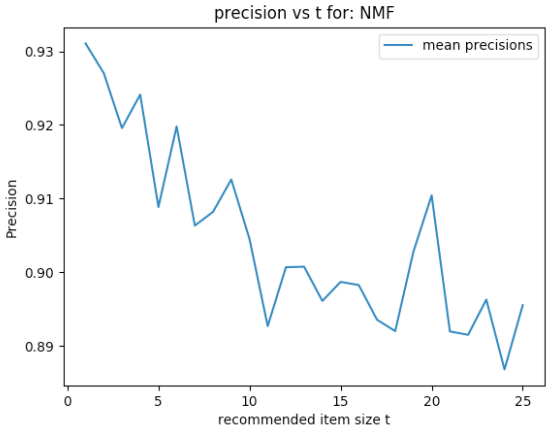


Figure 26: **The plot of the precision vs number of recommended item t, for t = 1 to 25.**

The graph for recall vs t has a upward but curved trend. It shows that higher t, higher recall, but the incremental increase decreases with higher t. The analysis of the trend is same as problem 36.

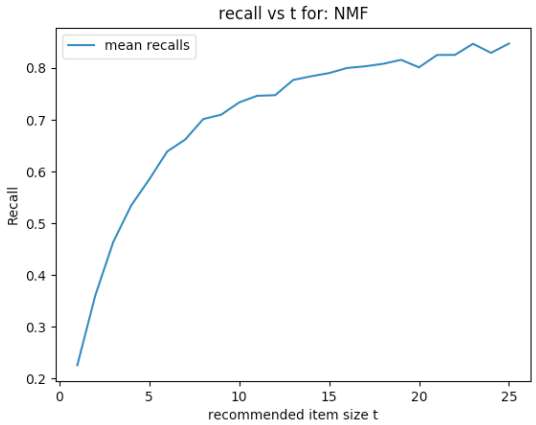


Figure 27: **The plot of the recall vs number of recommended item t, for t = 1 to 25.**

The graph for precision vs recall has a downward trend. Higher the recall, lower the precision.

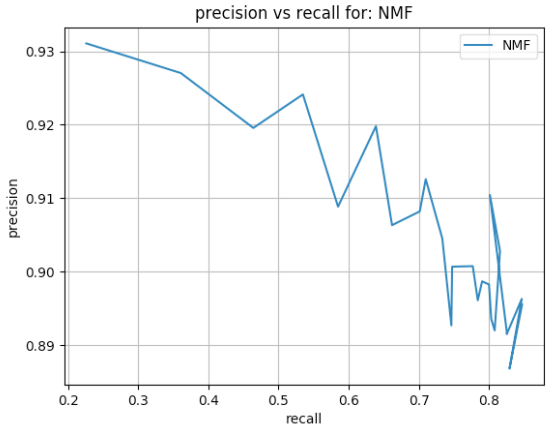


Figure 28: **The plot of the precision vs recall**

39 Q38

For problem 38, we plot precision vs t, recall vs t, and precision vs recall for the ranking from MF with bias collaborative filter predictions.

The number of factors k used here is from Q25. We use k = 8 here.

The graph for precision vs t has a downward trend. It shows that higher t, lower precision. The analysis of the trend is same as problem 36.

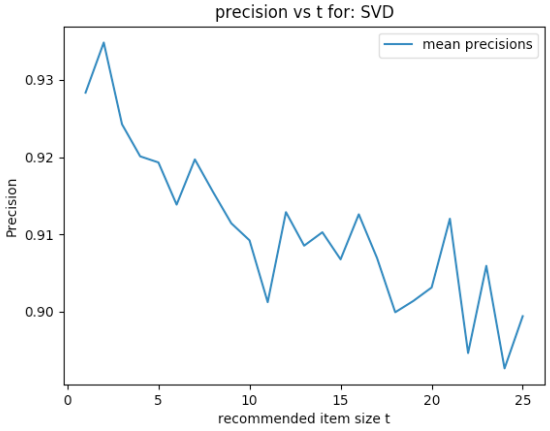


Figure 29: **The plot of the precision vs number of recommended item t, for t = 1 to 25.**

The graph for recall vs t has a upward but curved trend. It shows that higher t, higher recall, but the incremental increase decreases with higher t. The analysis of the trend is same as problem 36.

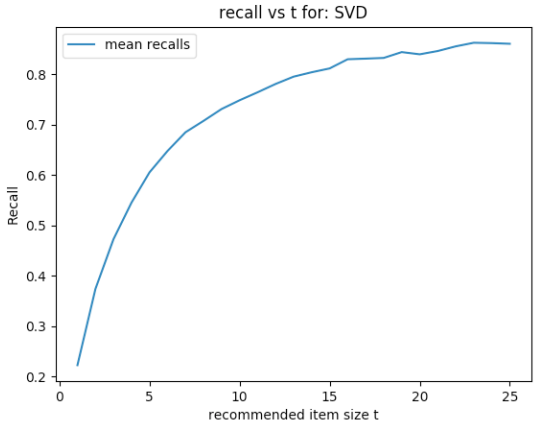


Figure 30: **The plot of the recall vs number of recommended item t, for t = 1 to 25.**

The graph for precision vs recall has a downward trend. Higher the recall, lower the precision.

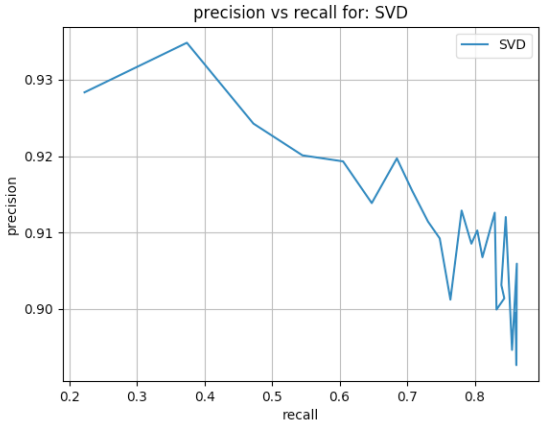


Figure 31: **The plot of precision vs recall**

40 Q39

Figure below plots precision vs recall for KNN, NMF, and MF with bias. As the figure shows, all three exhibit downward trend. Higher the recall, lower the precision. All three have more obvious downward trend from 0.2 to 0.5 of recall, and with above 0.5 recall, there's a lot of ups and downs. SVD has a little bit

of increase of precision from 0.2 to 0.4 of recall.

Generally, the downward trend shows that precision and recall have inverse relations. This makes sense because of how we define these two measures. The bigger the recommended set, the smaller portion of ground truth recommended set contains, so smaller precision, but higher ground truth recommended set can contain out of size of ground truth, so higher recall.

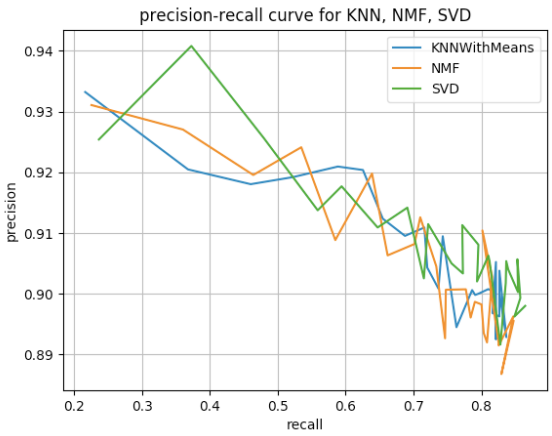


Figure 32: **The plot of precision vs recall**

41 Conclusion

For this project, we implemented a recommendation system via collaborative filtering, specifically, the neighbourhood -based collaborative filtering and model-based collaborative filtering. The data set we analyzed is a matrix comprised of users and movies, representing the ratings for different movies. The performances of these methods were analyzed and compared. Finally, a ranking mechanism was implemented through two primary approaches: prediction and ranking. Overall, this is an interesting project, from which the performances of two major model of filtering methods were manifested through the testing data set. In addition, the ranking mechanism is fairly useful and could be applied as an effective recommendation algorithm to nowadays software application development.