Project 3: Recommender System

Lauren Mizner (#005225768)

Question 1

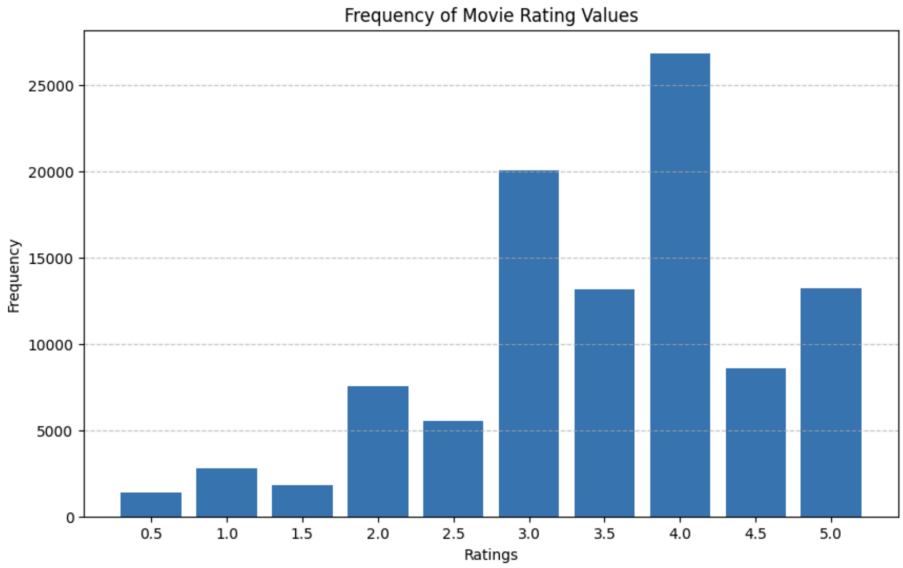
Explore the Dataset: In this question, we explore the structure of the data.

A. Compute the sparsity of the movie rating dataset

Sparsity of Movie Rating Dataset = 0.016999

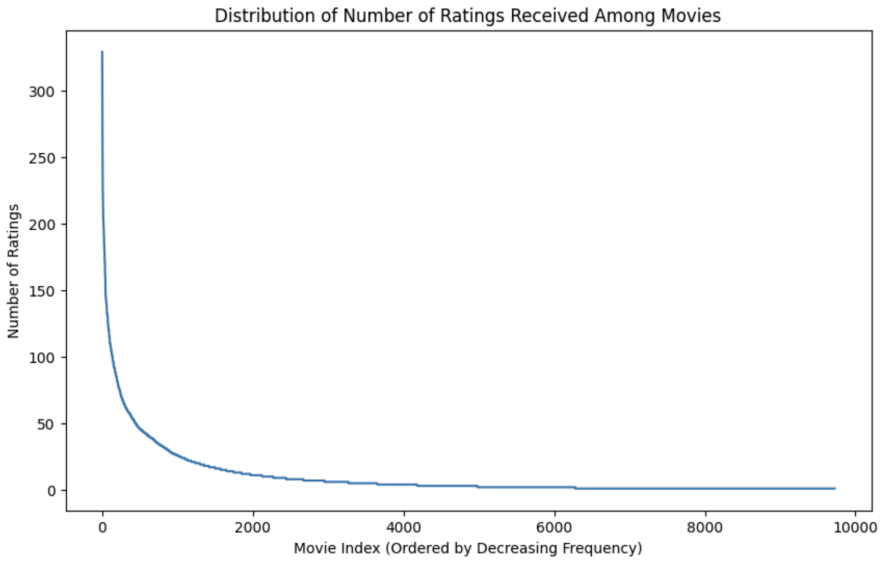
The sparsity is very low indicating that the dataset is very sparse and there are many missing ratings in the dataset.

B. Plot a histogram showing the frequency of the rating values

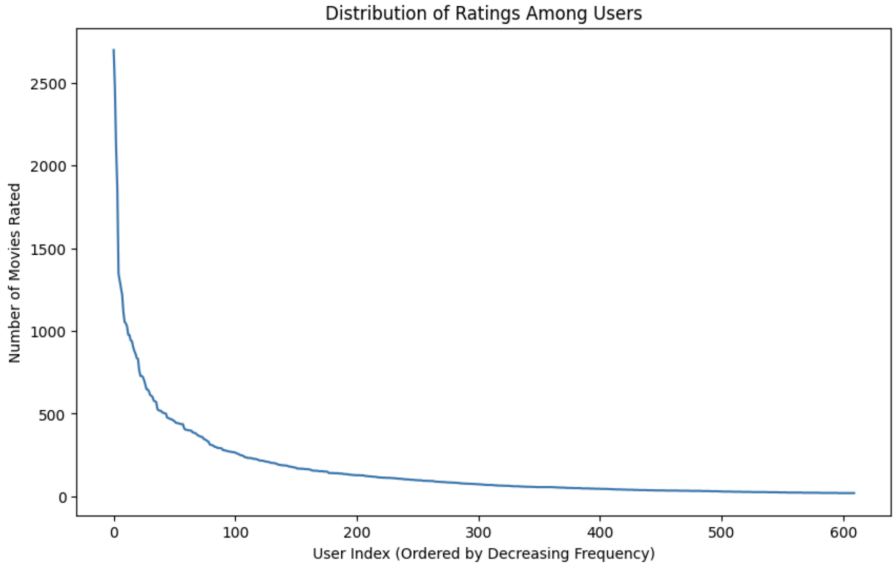


From the histogram it can be seen that the majority of ratings lie between 3.0 and 5.0, meaning that the distribution of the data is skewed to the right. This also indicates that users generally rate movies higher rather than lower, showing that users typically enjoy the movies they watch.

C. Plot the distribution of the number of ratings received among movies



D. Plot the distribution of ratings among users

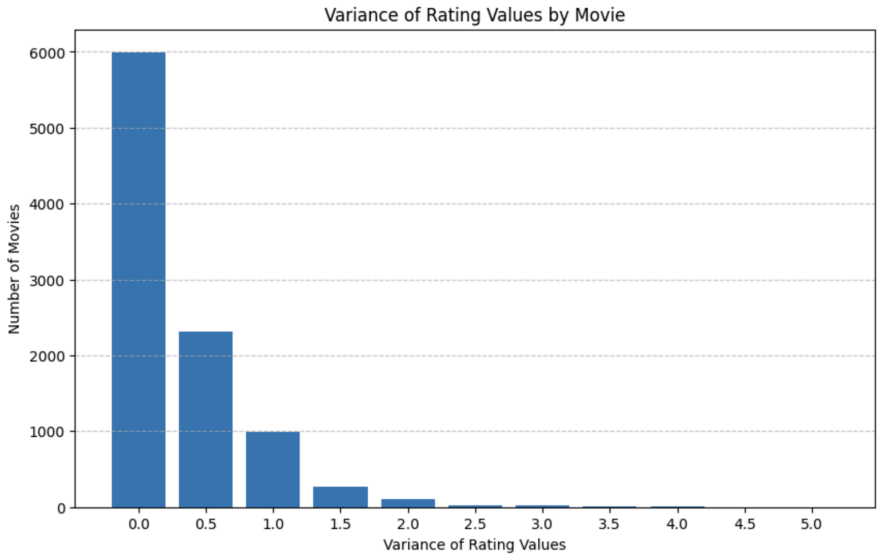


E. Discuss the salient features of the distributions from Questions C and D and their implications for the recommendation process.

Both distributions monotonically decrease in a similar nature, with the long-tail pattern. Among the distributions, we see that there is a small number of movies or users, depending on the plot, that corresponds to a large number of ratings. In other words, there are a few highly popular movies receiving a large number of ratings and, similarly, a few users who are highly active in rating a large number of movies. Both of these plots match our expectation based on the sparsity calculation indicating very sparse data. This

can lead to many challenges for the recommendation system, as it can be difficult to make accurate recommendations due to limited/missing data. The skew of the distribution can also indicate a lack of diversity in the data, leading to a lack of diversity in user recommendations as well, decreasing the overall user experience with the recommendation system.

F. Compute the variance of the rating values received by each movie.



From this histogram we can see that most movies have a very low variance in their rating, with the histogram very heavily left skewed. Since the ratings of a typical movie does not vary significantly, it points to users tending to watch movies based on existing ratings and reviews and continuing to rate accordingly. We can also take advantage of this similarity property to predict missing data in the dataset.

Question 2

Understanding the Pearson Correlation Coefficient:

A. Write down the formula for μ_u in terms of I_u and r_{uk}.

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

B. In plain words, explain the meaning of $I_u \cap I_v$. Can $I_u \cap I_v = 0$? (Hint: Rating matrix R is sparse)

$I_u \cap I_v$ represents the intersection of the set I_u and the set I_v, where each set contains the time indices that user u and user v have rated, respectively. Since it's the intersection of those two sets, it would equate to the list of movies that both users have rated. It is possible for the intersection between the sets I_u and I_v to be empty if there are no items in common that user u and user v have rated. In terms of a sparse matrix, this can occur due to users having rated different sets of items, where there is no overlap between the items that they have rated.

Question 3

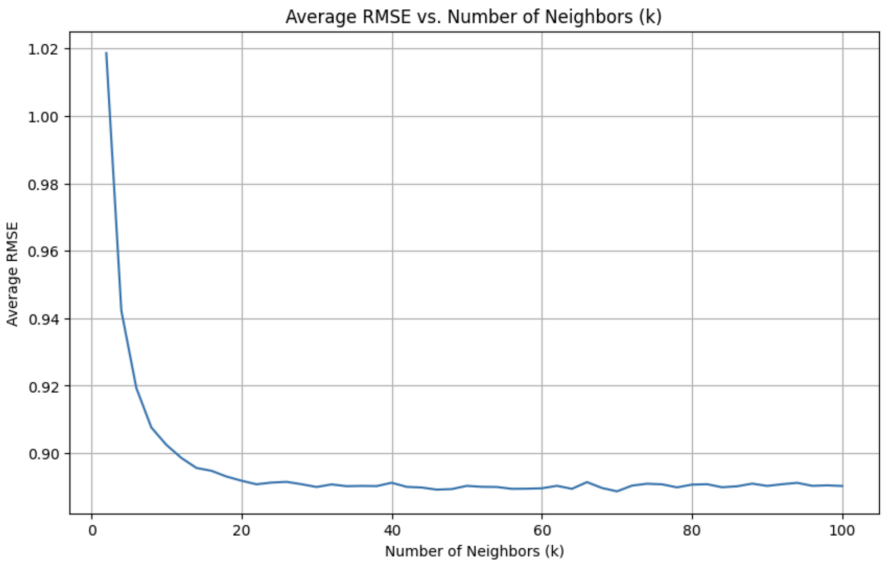
Understanding the Prediction Function:

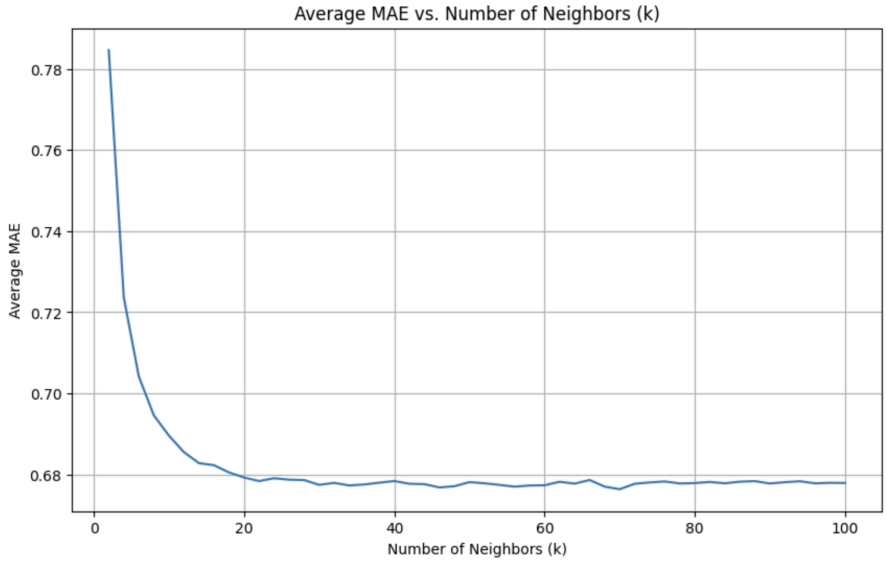
Can you explain the reason behind mean-centering the raw ratings $(r_{vj} - \mu_v)$ in the prediction function? (Hint: Consider users who either rate all items highly or rate all items poorly and the impact of these users on the prediction function.)

Mean-centering the raw ratings helps to normalize the ratings, remove user biases, and improve the stability and accuracy of the predictions. By subtracting the mean rating from each raw rating, we are able to normalize the ratings relative to the mean of user v, which subsequently removes user bias by helping to bring ratings to a common scale, specifically for the situation in which one user tends to rate items higher overall or lower overall. This process also helps improve stability by making predictions less sensitive to extreme ratings from individual users (again, because they're brought down to a common scale).

Question 4

Design a k-NN collaborative filter to predict the ratings of the movies in the original dataset and evaluate its performance using 10-fold cross validation. Sweep k (number of neighbors) from 2 to 100 in step sizes of 2, and for each k compute the average RMSE and average MAE obtained by averaging the RMSE and MAE across all 10 folds. Plot average RMSE (Y-axis) against k (X-axis) and average MAE (Y-axis) against k (X-axis).





Question 5

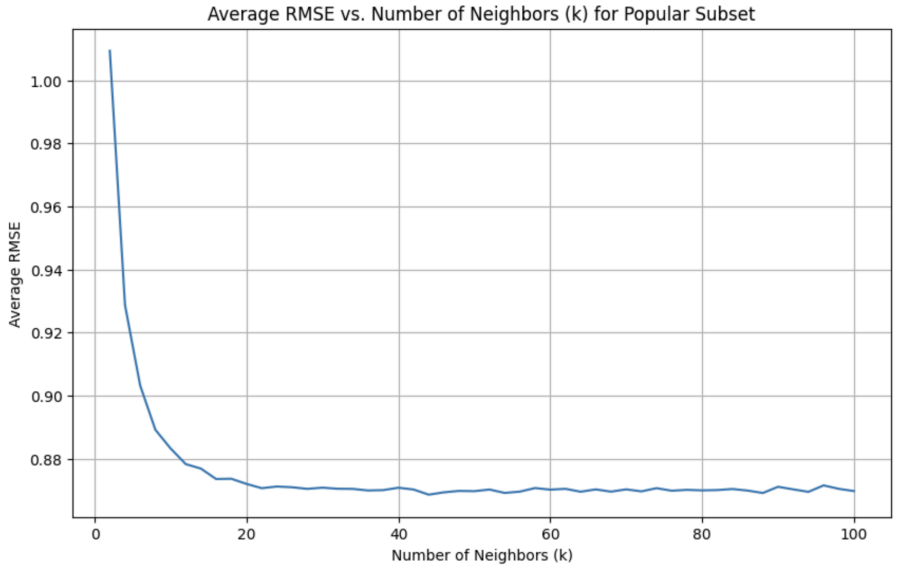
Use the plot from question 4 to find a 'minimum k'. Note: The term 'minimum k' in this context means that increasing k above the minimum value would not result in a significant decrease in average RMSE or average MAE. If you get the plot correct, then 'minimum k' would correspond to the k value for which average RMSE and average MAE converge to a steady-state value. Please report the steady state values of average RMSE and average MAE.

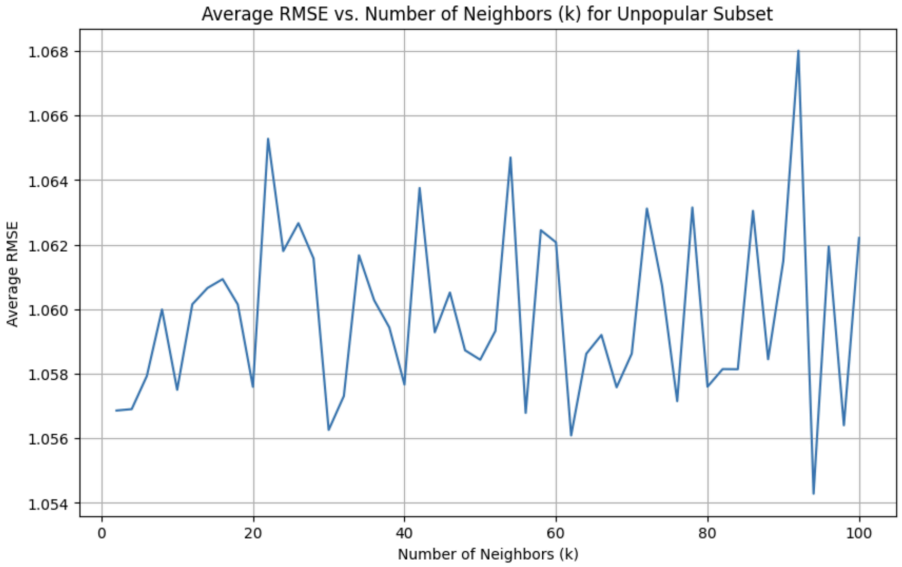
Minimum k (based on inspection of plot) = 20
Steady state value of average RMSE = 0.888639
Steady state value of average MAE = 0.676365

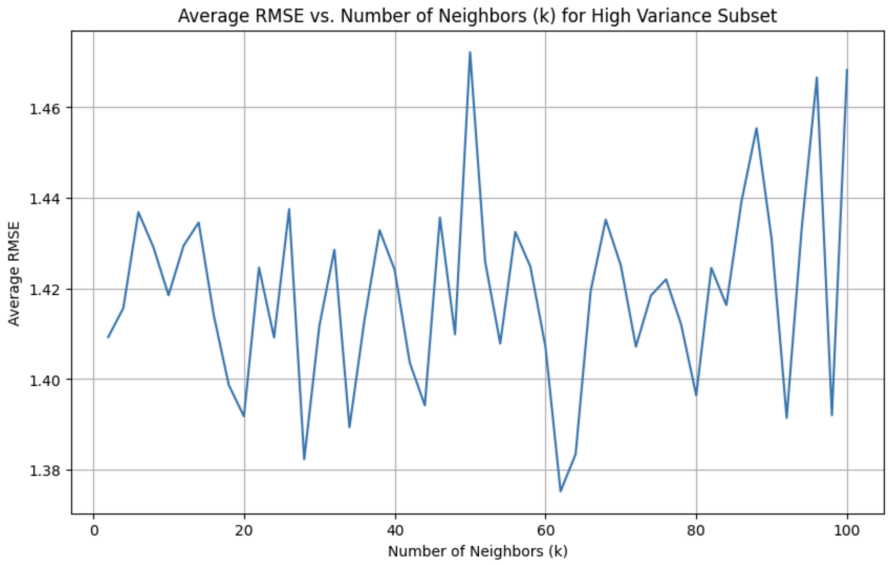
Question 6

Within EACH of the 3 trimmed subsets in the dataset, design, train, and validate:
A k-NN collaborative filter on the ratings of the movies (i.e. Popular, Unpopular, and High-Variance) and evaluate each of the three models' performance using 10-fold cross validation.

A. Sweep k (number of neighbors) from 2 to 100 in step sizes of 2, and for each k compute the average RMSE obtained by averaging the RMSE across all 10 folds. Plot average RMSE (Y-axis) against k (X-axis). Also, report the minimum average RMSE.

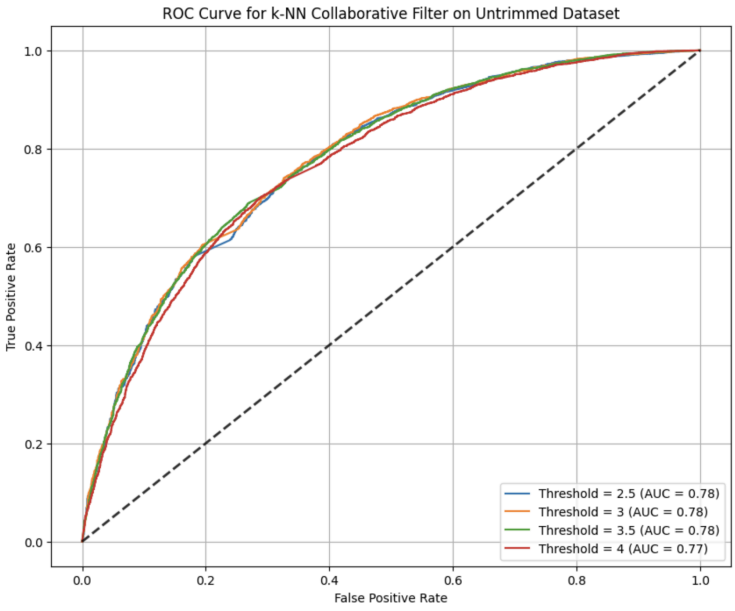




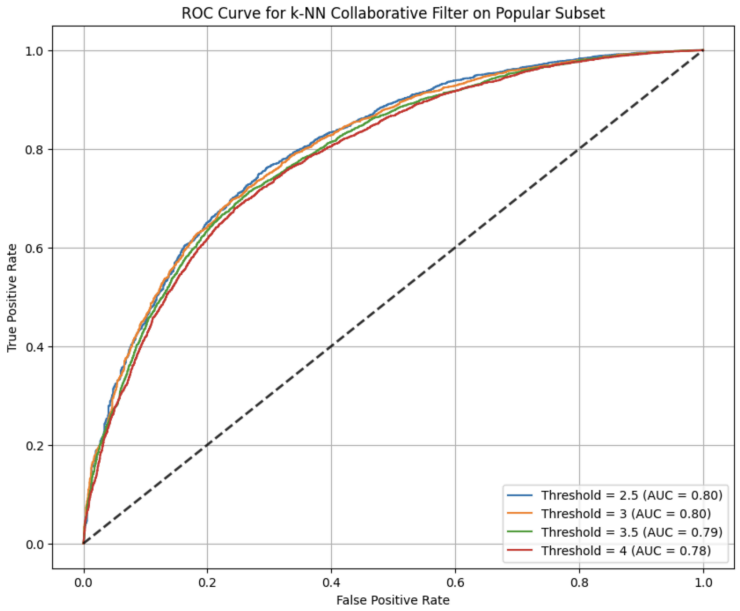


Minimum average RMSE for Popular Subset = 0.868546
Minimum average RMSE for Unpopular Subset = 1.054286
Minimum average RMSE for High Variance Subset = 1.375234

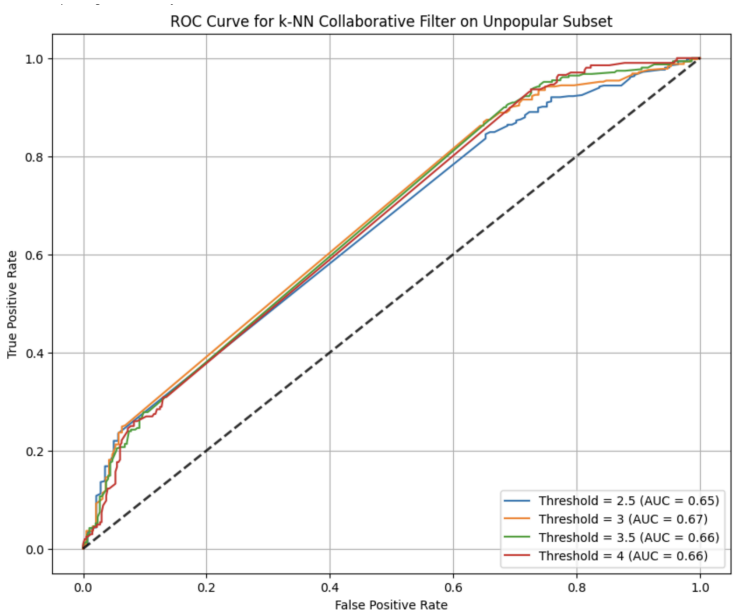
B. Plot the ROC curves for the k-NN collaborative filters for threshold values [2.5, 3, 3.5, 4]. These thresholds are applied only on the ground truth labels in the held-out validation set. For each of the plots, also report the area under the curve (AUC) value. You should have 4 x 4 plots in this section (4 trimming options - including no trimming time 4 thresholds) - all thresholds can be condensed into one plot per trimming option yielding only 4 plots.



Threshold 2.5 AUC = 0.78
Threshold 3.0 AUC = 0.78
Threshold 3.5 AUC = 0.78
Threshold 4.0 AUC = 0.77

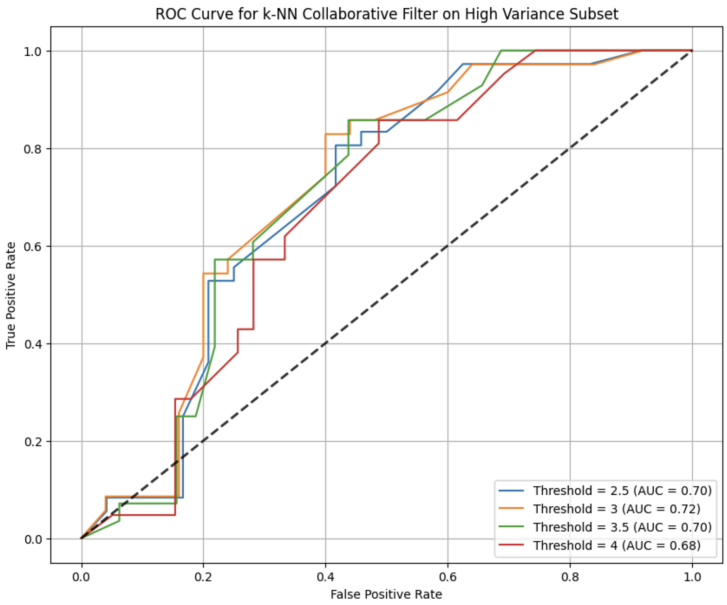


Threshold 2.5 AUC = 0.80
Threshold 3.0 AUC = 0.80
Threshold 3.5 AUC = 0.79
Threshold 4.0 AUC = 0.78



Threshold 2.5 AUC = 0.65
Threshold 3.0 AUC = 0.67

Threshold 3.5 AUC = 0.66
Threshold 4.0 AUC = 0.66



Threshold 2.5 AUC = 0.70
Threshold 3.0 AUC = 0.72
Threshold 3.5 AUC = 0.70
Threshold 4.0 AUC = 0.68

Question 7

Understanding the NMF cost function: Is the optimization problem given by equation 5 convex? Consider the optimization problem given by equation 5. For U fixed, formulate it as a least-squares problem.

The optimization problem given in equation 5 is a convex function because the objective function, which is the sum of squared difference between the observed ratings r_{ij} and the reconstructed ratings $(UV^T)_{ij}$, is a convex quadratic function. And since the summation of convex functions is still convex, then the overall optimization function is also convex.

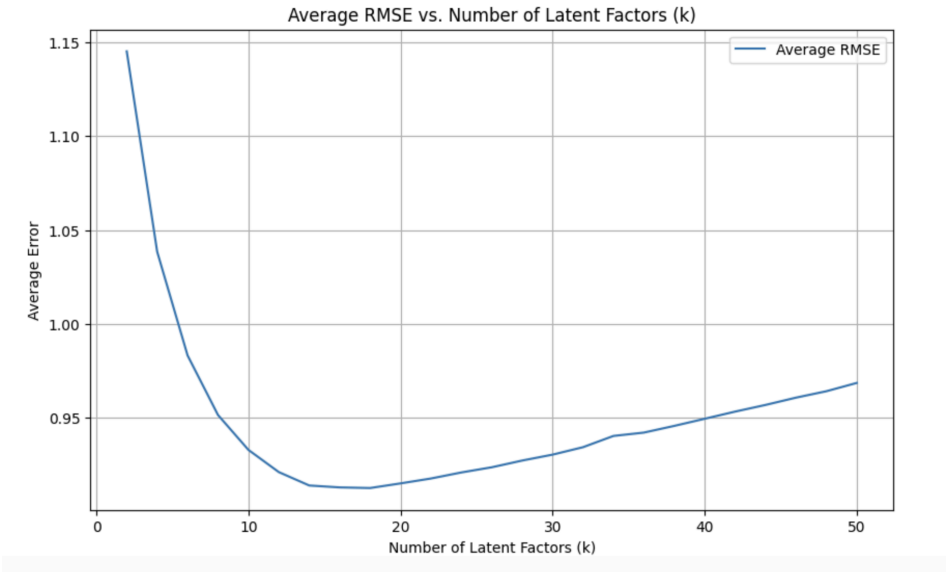
Given a fixed U, we can formulate it as a least-squares problem where we find a V that minimizes the sum of squares, as shown below.

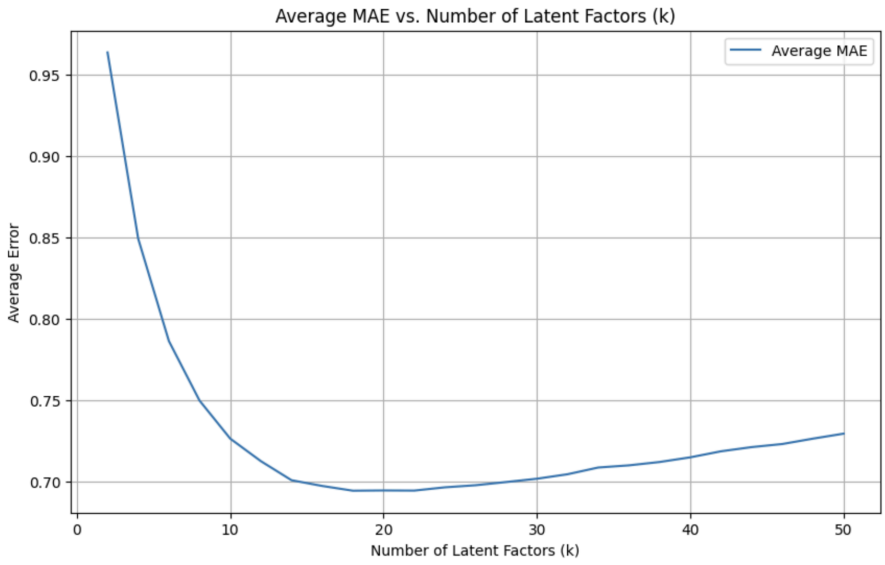
$$min_v \sum_{i,j} W_{ij}(r_{ij} - (UV^T)_{ij})^2$$

Question 8

Designing the NMF Collaborative Filter:

A. Design a NMF-based collaborative filter to predict the ratings of the movies in the original dataset and evaluate its performance using 10-fold cross-validation. Sweep k (number of latent factors) from 2 to 50 in step sizes of 2, and for each k compute the average RMSE and average MAE obtained by averaging the RMSE and MAE across all 10 folds. If NMF takes too long, you can increase the step size. Increasing it too much will result in poorer granularity in your results. Plot the average RMSE (Y-axis) against k (X-axis) and the average MAE (Y-axis) against k (X-axis). For solving this question, use the default value for the regularization parameter.



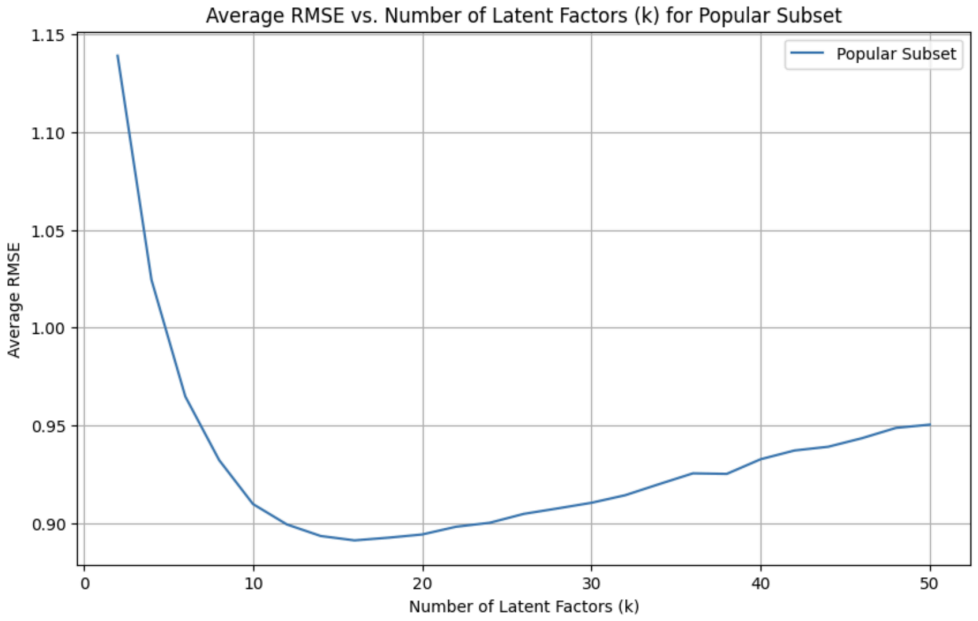


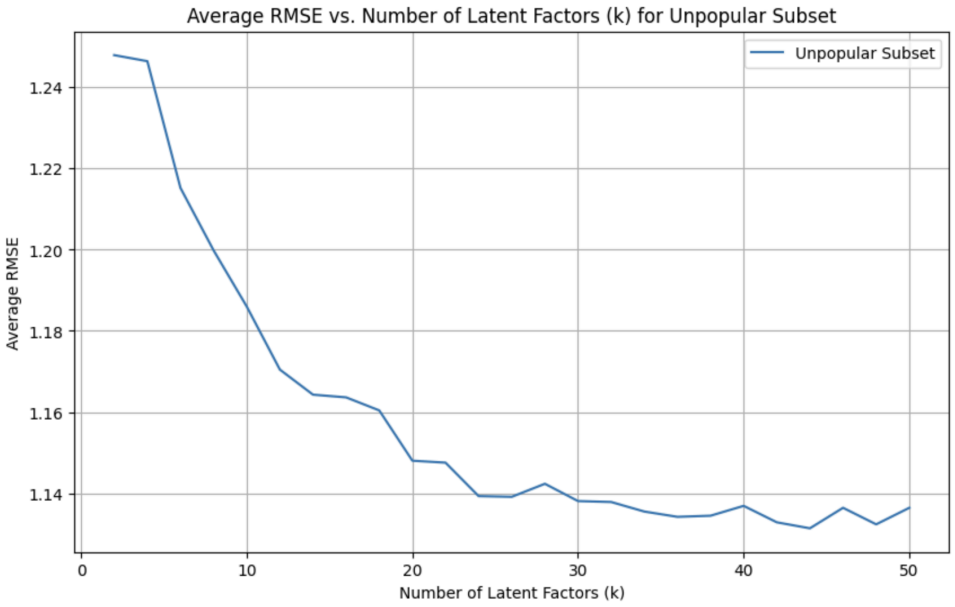
B. Use the plot from the previous part to find the optimal number of latent factors. The optimal number of latent factors is the value of k that gives the minimum average RMSE or the minimum average MAE. Please report the minimum average RMSE and MAE. Is the optimal number of latent factors the same as the number of movie genres?

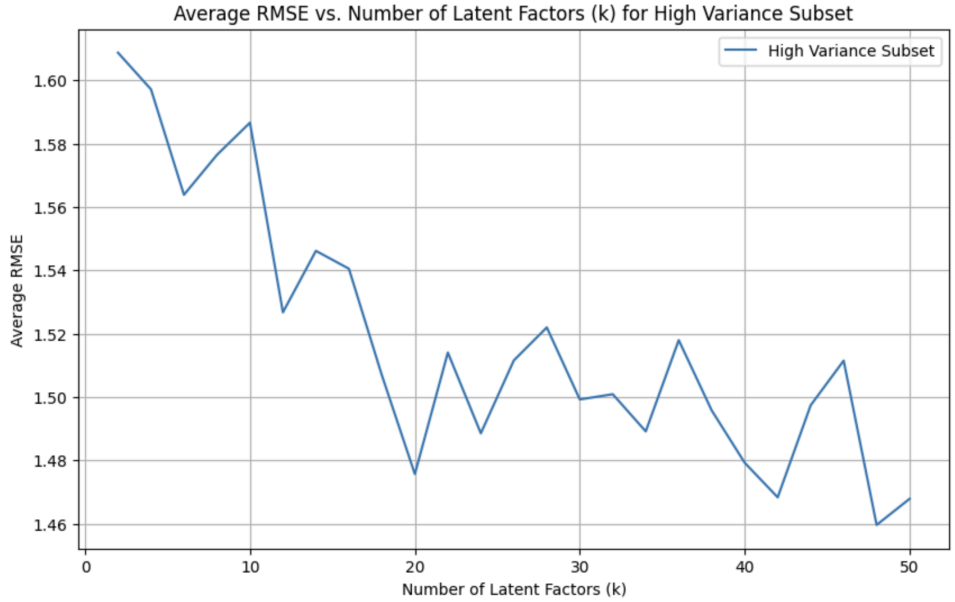
Minimum average RMSE = 0.912499
Minimum average MAE = 0.694186
Optimal Number of Latent Factors = ~20

Yes, the number of latent factors is roughly equal to the number of unique movie genres.

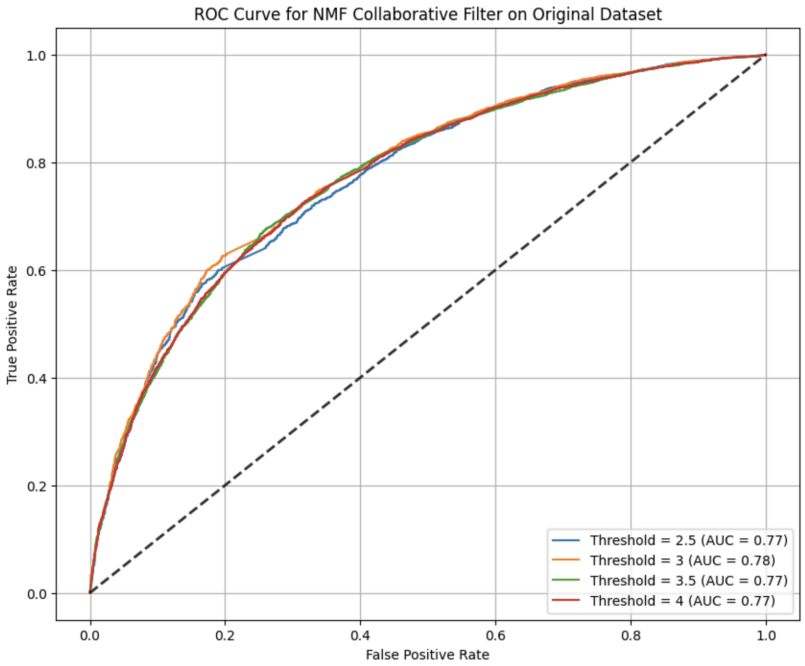
C. Performance on trimmed dataset subsets: For each of the Popular, Unpopular, and High-Variance subsets -
   - Design a NMF collaborative filter for each trimmed subset and evaluate its performance using 10-fold cross validation. Sweep k (number of latent factors) from 2 to 50 in step sizes of 2, and for each k compute the average RMSE obtained by averaging the RMSE across all 10 folds.
   - Plot average RMSE (Y-axis) against k (X-axis). Report the minimum average RMSE.
   - Plot the ROC curves for the NMF-based collaborative filter and also report the area under the curve (AUC) value.



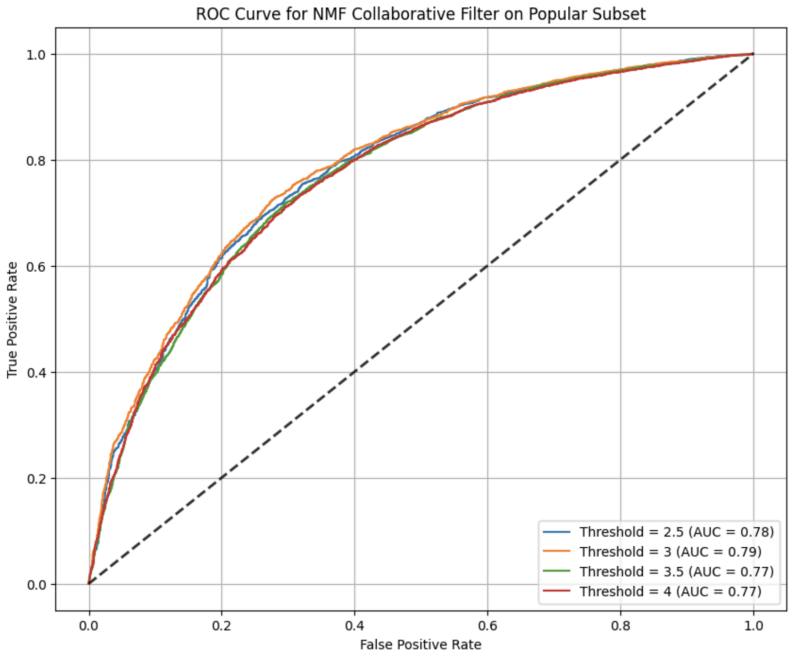




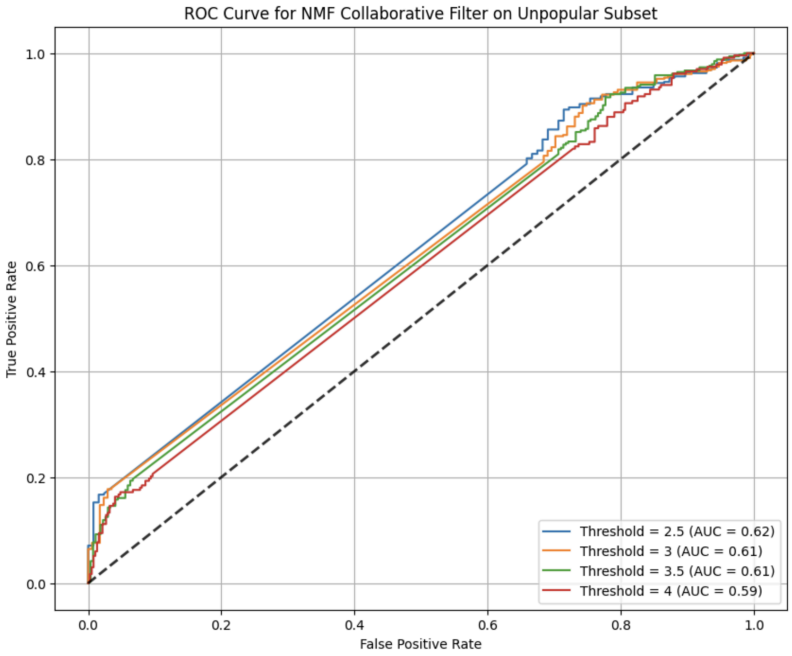
Minimum average RMSE for Popular Subset = 0.891261
Minimum average RMSE for Unpopular Subset = 1.131359
Minimum average RMSE for High Variance Subset = 1.459604



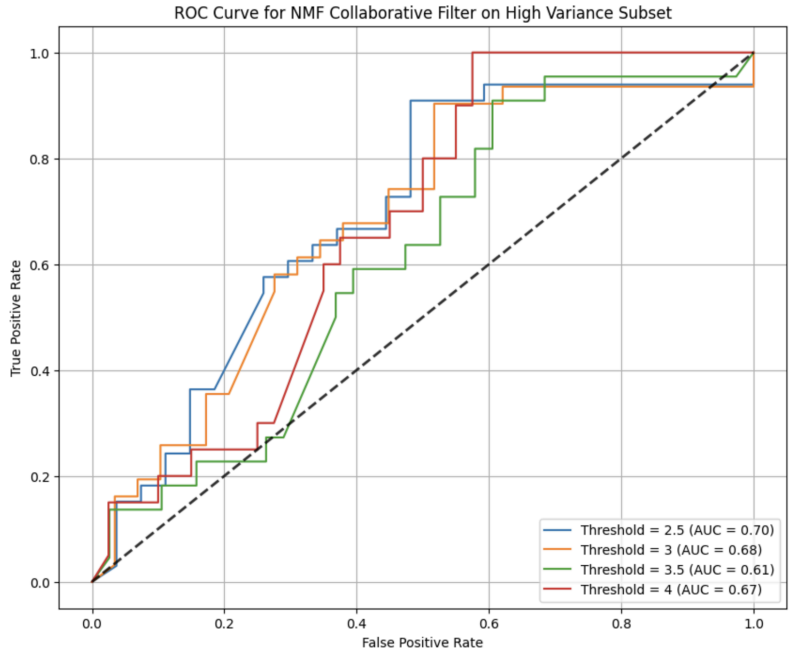
Threshold 2.5 AUC = 0.77
Threshold 3.0 AUC = 0.78
Threshold 3.5 AUC = 0.77
Threshold 4.0 AUC = 0.77



Threshold 2.5 AUC = 0.78
Threshold 3.0 AUC = 0.79
Threshold 3.5 AUC = 0.77
Threshold 4.0 AUC = 0.77



Threshold 2.5 AUC = 0.62
Threshold 3.0 AUC = 0.61
Threshold 3.5 AUC = 0.61
Threshold 4.0 AUC = 0.59



Threshold 2.5 AUC = 0.70
Threshold 3.0 AUC = 0.68
Threshold 3.5 AUC = 0.61
Threshold 4.0 AUC = 0.67

Question 9

Interpreting the NMF model: Perform non-negative matrix factorization on the ratings matrix R to obtain the factor matrices U and V, where U represents the user-latent factors interaction and V represents the movie-latent factors interaction (use k = 20). For each column of V, sort the movies in descending order and report the genres of the top 10 movies. Do the top 10 movies belong to a particular or a small collection of genres? Is there a connection between latent factors and the movie genres?

From the results given below, we can tell that there is a connection between the latent factors and movie genres and that for each group, the top 10 movies do belong to a small collection of genres. For example, in V:0 we see that the group is strongly correlated with comedy, drama, and romance, while if we look at V:3 we see that the group is strongly correlated with action and comedy.

Column number of V: 0
Comedy|Drama|Romance
Comedy|Drama
Comedy|Romance
Comedy|Horror
Drama|Romance|War
Drama|Romance
Comedy|Musical|Romance
Comedy|Documentary
Comedy|Drama
Comedy|Drama

Column number of V: 1
Action
Drama
Documentary
Drama|War
Crime|Drama
Comedy
Drama|Western
Adventure|Comedy|Mystery|Romance
Drama
Animation|Children

Column number of V: 2
Comedy|Crime|Drama|Thriller
Comedy|Crime
Comedy
Comedy|Sci-Fi
Comedy|Drama|Romance

Drama|Fantasy
Action|Comedy|Crime|Thriller
Action|Adventure|Animation|Comedy|Fantasy|Mystery|Sci-Fi
Action|Drama|Thriller
Action|Adventure|Animation|Children|Fantasy

Column number of V: 3
Crime|Thriller
Drama
Adventure|Children|Drama|Fantasy
Comedy
Adventure|Comedy|Mystery|Romance
Horror|Sci-Fi
Comedy|Romance
Adventure|Children|Fantasy|Sci-Fi|Thriller
Mystery|Thriller
Comedy|Romance

Column number of V: 4
Action|Adventure|Drama|War
Drama
Comedy|Sci-Fi
Drama
Action|Fantasy|Horror|Sci-Fi|Thriller
Adventure|Animation|Sci-Fi
Drama
Comedy
Comedy|Crime
Drama

Column number of V: 5
Comedy|Crime|Horror|Thriller
Comedy
Comedy|Drama|Romance
Action|Adventure|Fantasy|War
Comedy|Drama|Romance
Documentary
Comedy|Romance
Action|Crime|Mystery|Sci-Fi|Thriller
Drama|Thriller
Action|Sci-Fi

Column number of V: 6
Comedy|Crime
Horror

Crime|Drama
Action|Adventure|Sci-Fi|Thriller|War
Comedy|Romance
Adventure|Animation|Children|Comedy
Comedy|Horror
Crime|Drama
Drama|Musical
Comedy|Romance

Column number of V: 7
Action|Adventure|Comedy|Fantasy
Drama
Comedy|Fantasy|Horror|Thriller
Drama
Adventure|Comedy
Drama|Mystery|Romance|Thriller
Comedy|Romance
Action|Sci-Fi|Thriller|IMAX
Children|Comedy
Drama

Column number of V: 8
Comedy
Adventure|Drama|Romance
Drama|Romance
Comedy|Romance
Action|Crime|Drama|Thriller
Horror|Sci-Fi|Thriller
Action|Drama|Romance|Sci-Fi
Crime|Drama
Comedy|Horror
Action|Sci-Fi|Thriller|IMAX

Column number of V: 9
Thriller
Action|Adventure|Drama|Thriller
Comedy|Horror
Crime|Drama|Film-Noir
Crime|Film-Noir|Mystery
Horror|Sci-Fi|Thriller
Comedy|Drama|Sci-Fi|Thriller
Horror
Comedy|Drama|Romance
Drama

Column number of V: 10
Drama|Thriller
Comedy|War
Adventure|Animation|Comedy
Drama|Thriller
Documentary
Documentary|Drama
Drama
Adventure|Drama|Fantasy|Romance
Comedy|Drama|Romance
Crime|Drama|Film-Noir|Thriller

Column number of V: 11
Thriller
Children|Fantasy
Adventure|Animation|Fantasy|Romance
Sci-Fi
Drama|Musical
Adventure|Animation|Children|Comedy|Fantasy
Action|Adventure|Sci-Fi
Drama|Western
Action|Drama|Western
Crime|Drama|Romance|Thriller

Column number of V: 12
Horror|Mystery|Thriller
Drama|Romance
Action|Comedy
Action|Adventure|Drama|Thriller
Comedy|Drama
Crime|Thriller
Horror|Thriller
Comedy|Crime
Comedy|Sci-Fi
Comedy

Column number of V: 13
Comedy|Drama|Romance
Comedy|Fantasy|Sci-Fi
Drama|Mystery
Comedy|Documentary
Action|Sci-Fi|Thriller
Action|Adventure|Animation|Children|Fantasy|Sci-Fi
Comedy|Crime|Drama|Thriller
Action|Adventure|Drama|Romance|War
Adventure|Drama

Drama

Column number of V: 14
Action|Crime|Drama
Drama|Romance
Comedy|Horror
Drama|War
Adventure|Drama
Comedy|Drama
Comedy|Drama|War
Comedy|Romance
Comedy|Romance
Comedy|Documentary

Column number of V: 15
Animation|Drama|Romance
Comedy|Crime
Drama
Crime|Drama
Comedy|Drama
Action|Crime|Thriller
Drama|Mystery|Thriller
Comedy|Drama
Drama|Horror|Thriller
Action|Comedy|Romance

Column number of V: 16
Comedy|Crime
Action|Comedy
Adventure|Children|Sci-Fi
Drama
Sci-Fi
Musical|Romance|Western
Adventure|Drama|War|Western
Comedy|Drama
Comedy|Drama|Romance
Comedy|Drama

Column number of V: 17
Comedy
Documentary
Sci-Fi
Action|Animation|Crime
Horror|Sci-Fi|Thriller
Drama|Romance|Thriller

Adventure|Drama
Horror|Thriller
Drama|Horror|Musical|Thriller
Action|Romance|Thriller

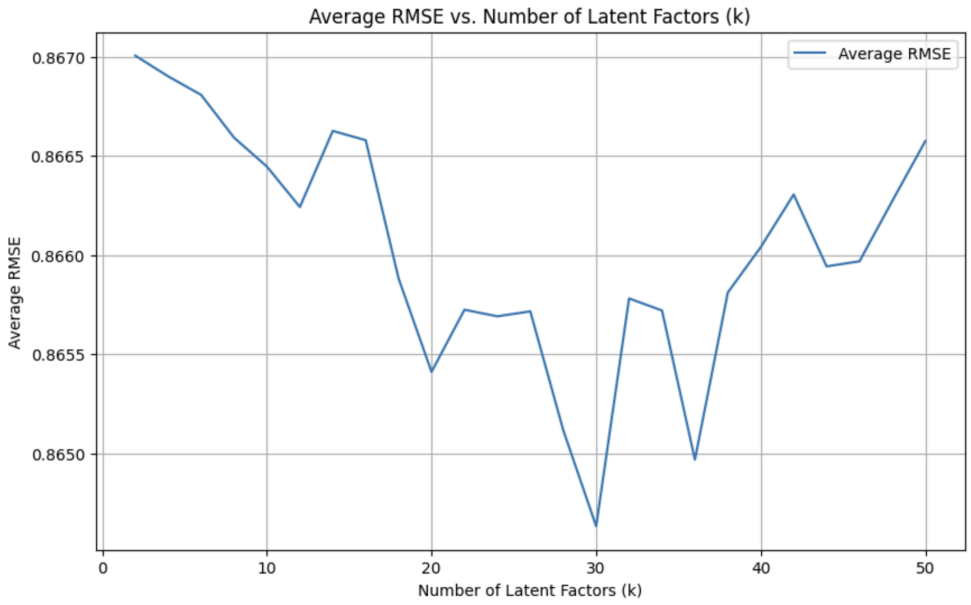
Column number of V: 18
Documentary|Drama
Comedy
Comedy|Horror|Sci-Fi
Comedy|Horror
Comedy|Romance
Action|Adventure|Fantasy
Comedy|Drama
Comedy
Adventure|Romance
Drama|War

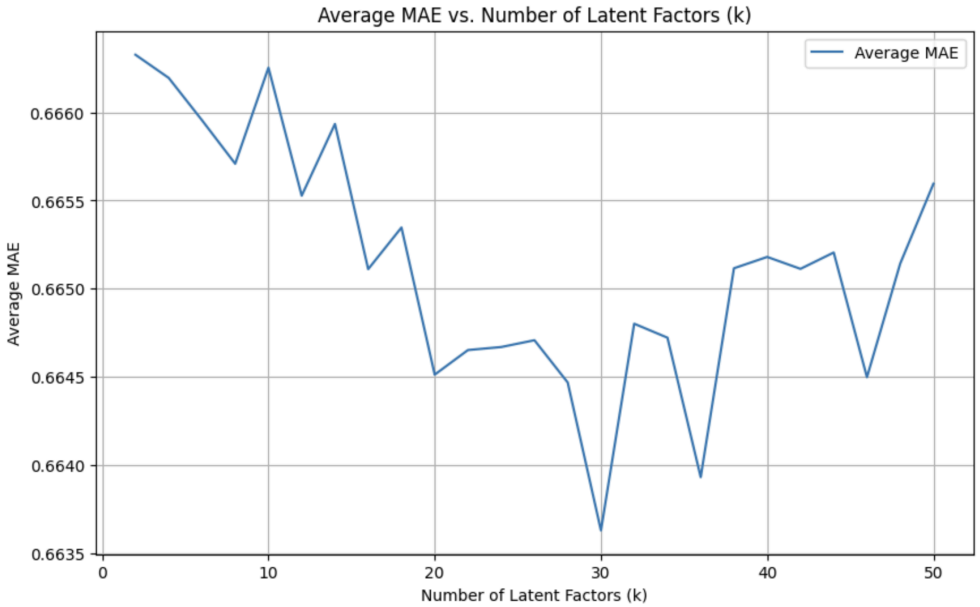
Column number of V: 19
Comedy|Romance
Action|Sci-Fi
Documentary
Western
Comedy|Drama
Comedy
Action|Adventure|Sci-Fi
Drama|Romance
Action|Crime|Drama|Thriller
Mystery|Sci-Fi|Thriller

Question 10

Designing the MF Collaborative Filter:

A. Design a MF-based collaborative filter to predict the ratings of the movies in the original dataset and evaluate its performance using 10-fold cross-validation. Sweep k (number of latent factors) from 2 to 50 in step sizes of 2, and for each k compute the average RMSE and average MAE obtained by averaging the RMSE and MAE across all 10 folds. Plot the average RMSE (Y-axis) against k (X-axis) and the average MAE (Y-axis) against k (X-axis). For solving this question, use the default value for the regularization parameter.

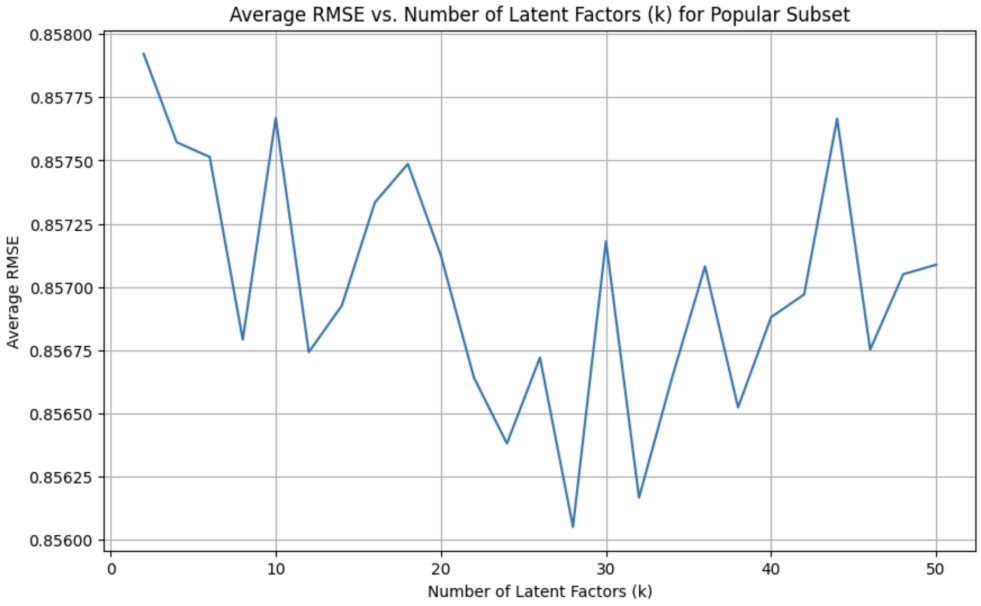


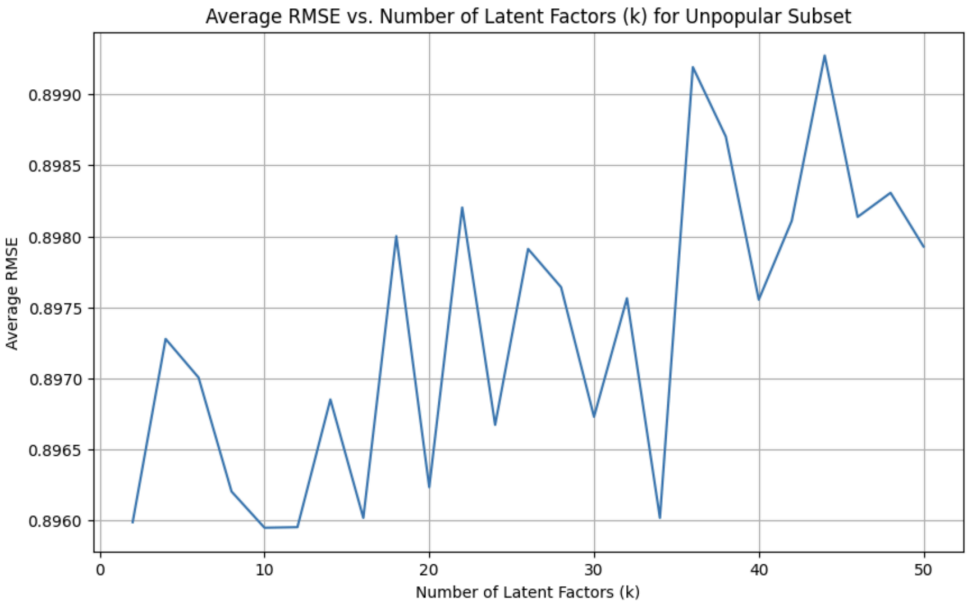


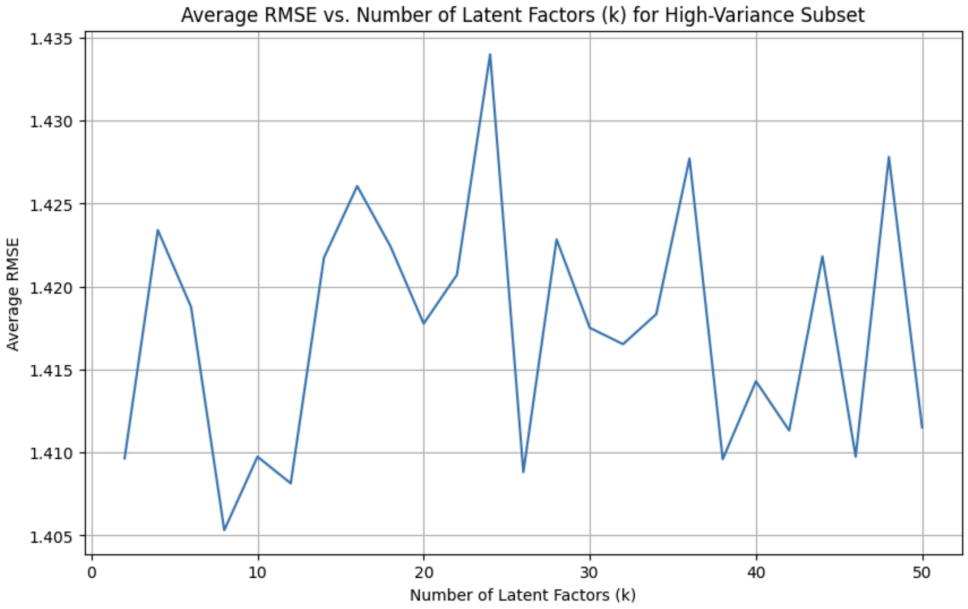
B. Use the plot from the previous part to find the optimal number of latent factors. The optimal number of latent factors is the value of k that gives the minimum average RMSE or the minimum average MAE. Please report the minimum average RMSE and MAE. Is the optimal number of latent factors the same as the number of movie genres?

Minimum average RMSE = 0.864631
Minimum average MAE = 0.663630
Optimal latent factors = 30,
which does not match the number of movie genres of 20

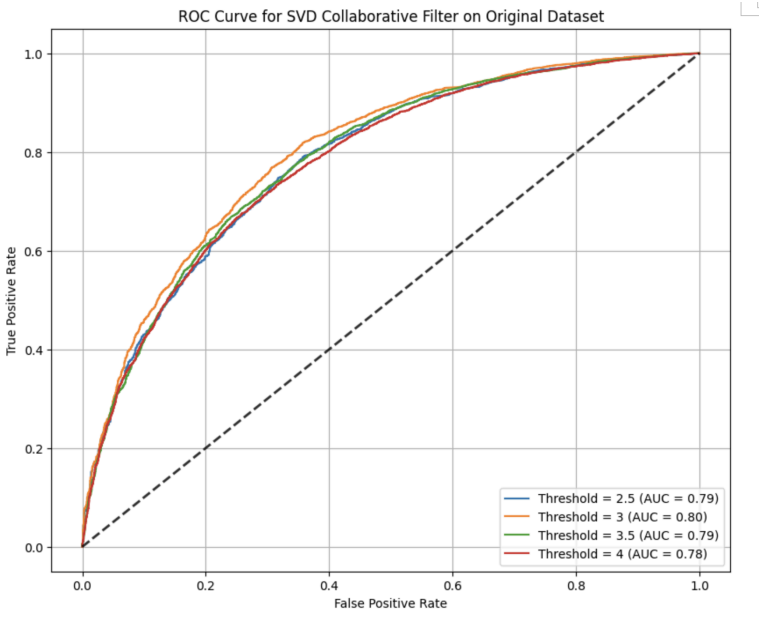
C. Performance on dataset subsets: For each of the Popular, Unpopular, and High-Variance subsets -
   - Design a MF collaborative filter for each trimmed subset and evaluate its performance using 10-fold cross validation. Sweep k (number of latent factors) from 2 to 50 in step sizes of 2, and for each k compute the average RMSE obtained by averaging the RMSE across all 10 folds.
   - Plot average RMSE (Y-axis) against k (X-axis). Report the minimum average RMSE.
   - Plot the ROC curves for the MF-based collaborative filter and also report the area under the curve (AUC) values.



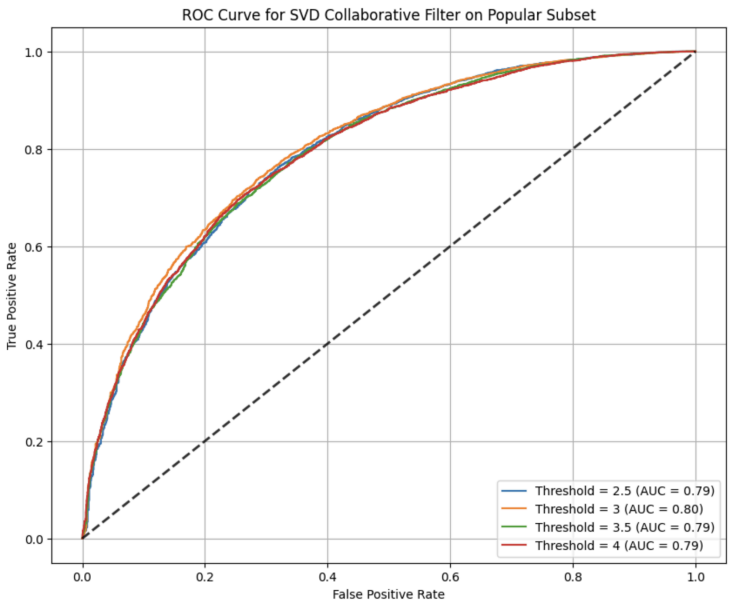




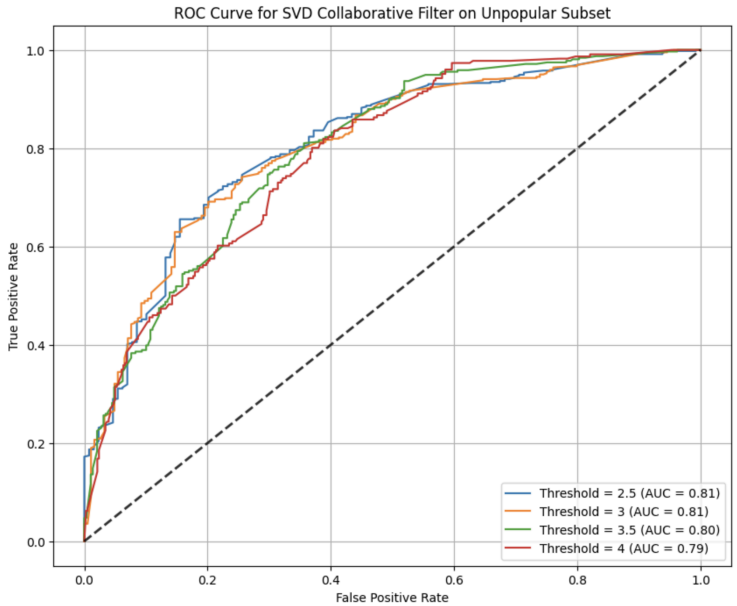
Minimum average RMSE for Popular Subset = 0.856050
Minimum average RMSE for Unpopular Subset = 0.895948
Minimum average RMSE for High Variance Subset = 1.405320



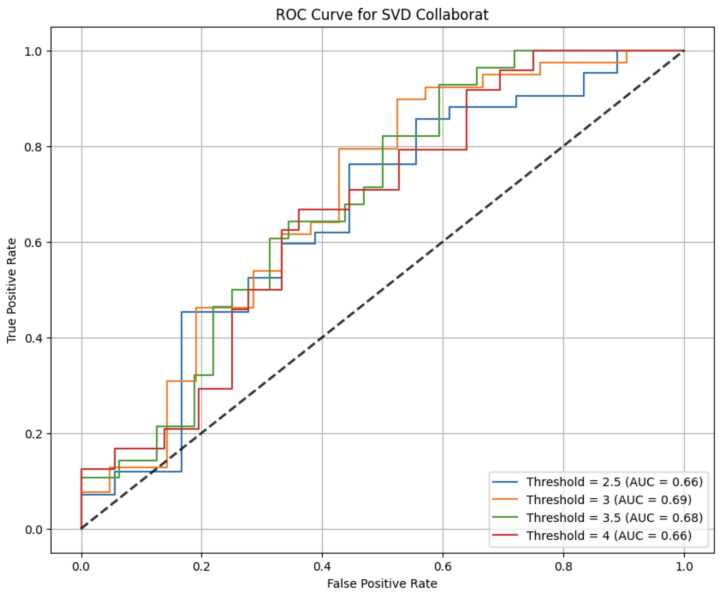
Threshold 2.5 AUC = 0.79
Threshold 3.0 AUC = 0.80
Threshold 3.5 AUC = 0.79
Threshold 4.0 AUC = 0.78



Threshold 2.5 AUC = 0.79
Threshold 3.0 AUC = 0.80
Threshold 3.5 AUC = 0.79
Threshold 4.0 AUC = 0.79



Threshold 2.5 AUC = 0.81
Threshold 3.0 AUC = 0.81
Threshold 3.5 AUC = 0.80
Threshold 4.0 AUC = 0.79



Threshold 2.5 AUC = 0.66
Threshold 3.0 AUC = 0.69
Threshold 3.5 AUC = 0.68
Threshold 4.0 AUC = 0.66

Question 11

Designing a Naive Collaborative Filter:

A. Design a naive collaborative filter to predict the ratings of the movies in the original dataset and evaluate its performance using 10-fold cross validation. Compute the average RMSE by averaging the RMSE across all 10 folds. Report the average RMSE.

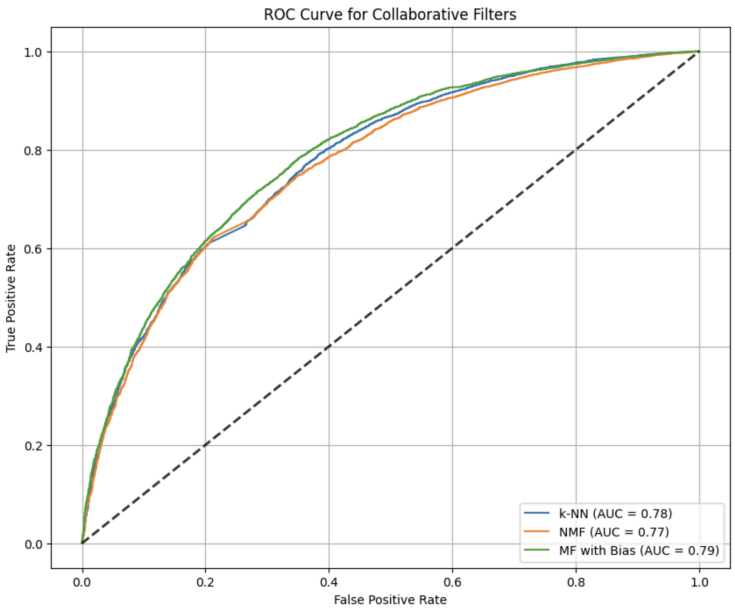
   Average RMSE = 1.426729

B. Performance on dataset subsets: For each of the Popular, Unpopular, and High-Variance subsets -
   - Design a naive collaborative filter for each trimmed set and evaluate its performance using 10-fold cross validation
   - Compute the average RMSE by averaging the RMSE across all 10 folds. Report the average RMSE.

     Average RMSE for Popular Subset = 1.017820
     Average RMSE for Unpopular Subset = 1.098369
     Average RMSE for High Variance Subset = 0.781156

Question 12

Comparing the most performant models across architecture: Plot the best ROC curves (threshold = 3) for the k-NN, NMF, and MF with bias based collaborative filters in the same figure. Use the figure to compare the performance of the filters in predicting the ratings of the movies.



k-NN AUC = 0.78
NMF AUC = 0.77
MF with Bias AUC = 0.79

Question 13

Data Understanding and Preprocessing:

- Use the provided helper code for loading and pre-processing Web10k data.
- Print out the number of unique queries in total and show distribution of relevance labels.

  Total number of unique queries: 40000

  Total distribution of relevance labels in training data across all folds:
  [1872789. 1158840. 478353.   63951.   26643.]

  Total distribution of relevance labels in test data across all folds:
  [624263. 386280. 159451. 21317.   8881.]

Question 14

LightGBM Model Training:

For each of the five provided folds, train a LightGBM model using the 'lambdarank' objective. After training, evaluate and report the model's performance on the test set using nDCG@3, nDCG@5, and nDCG@10.

Average nDCG@3 across all folds: 0.4696344288396136
Average nDCG@5 across all folds: 0.4714315145908389
Average nDCG@10 across all folds: 0.49035928048966515

Question 15

Result and Analysis Interpretation:

For each of the five provided folds, list the top 5 most important features of the model based on the importance score. Please use model.booster…feature_importance(importance_type = 'gain') as demonstrated for retrieving importance score per feature.

Fold 1: Top 5 most important features
1. Feature 133
2. Feature 7
3. Feature 107
4. Feature 54

5. Feature 129

Fold 2: Top 5 most important features
1. Feature 133
2. Feature 7
3. Feature 54
4. Feature 107
5. Feature 129

Fold 3: Top 5 most important features
1. Feature 133
2. Feature 54
3. Feature 107
4. Feature 129
5. Feature 7

Fold 4: Top 5 most important features
1. Feature 133
2. Feature 7
3. Feature 54
4. Feature 129
5. Feature 128

Fold 5: Top 5 most important features
1. Feature 133
2. Feature 7
3. Feature 54
4. Feature 107
5. Feature 129

Question 16

Experiments with Subset of Features:

For each of the five provide folds:

- Remove the top 20 most important features according to the computed importance score in question 15. Then train a new LightGBM model on the resulting 116 dimensional query-url data. Evaluate the performance of this new model on the test set using nDCG. Does the outcome align with your expectations? If not, please share your hypothesis regarding the potential reasons for this discrepancy.

  Average nDCG score across all folds after removing top 20 features:

0.522838100527449

The outcome does not align with expectations. It would be expected that we would see a decrease in score after removing the top 20 features, but instead we are seeing an increase in the score. This may occur for a few different reasons. It may be that the important features may be irrelevant or noisy, so removing them leads to better generalization across the model. Another reason could be that the top 20 features contain redundant information, so removing them allows the model to focus on more informative features, which leads to better performance. Lastly, it may be that the original model may be overfitting to the training data, and that removing the top 20 features allows the model to better generalize to unseen data, resulting in improved performance.

- Remove the 60 least important features according to the computed importance score in question 15. Then train a new LightGBM model on the resulting 76 dimensional query-url data. Evaluate the performance of this new model on the test set using nDCG. Does the outcome align with your expectations? If not, please share your hypothesis regarding the potential reasons for this discrepancy.

  Average nDCG score across all folds after removing the least important 60 features: 0.6634693900898593

  This outcome better aligns with expectations, that we would see an increase in performance occur after removing the 60 least important features. It is expected that the 60 least important features would not be contributing significantly to the model's overall performance, and that their removal would lead to a better performing model.