Spendylove Apaloo

Dr. Chou

CSC – 485

March 8, 2024

Movie Recommendation System

1. **Introduction**

Modern technology has expanded the range of choices available to people for virtually any decision they have to make. From selecting a movie to watch on streaming platforms to choosing a restaurant for dinner, the abundance of choices can often lead to a phenomenon known as analysis paralysis. This phenomenon occurs when a person is faced with too many choices, making the decision making process very overwhelming. Recommendations systems have been created as a tool used to alleviate the burden of choice. These systems use algorithms and machine learning techniques in order to assess a user’s historical record in order to create recommendations that would accurately align with their organic choices. Recommendation systems are powerful tools used by many platforms. E-commerce platforms use these systems to recommend clothes that users will like based on their search history, social media platforms use them to recommend friends, posts and articles based on the users’ past interactions. Movie and music streaming platforms use recommendation systems to help users discover new movies and artists similar to the ones they previously interacted with. There are many types of recommendation systems implemented by these platforms in order to produce the most accurate prediction, however, many prefer a hybrid recommendation system which . The purpose of this project is to create a movie recommendation system using the collaborative filtering system.

1. **Data Overview**

The data used to build this recommendation system The 100k MovieLense ratings data set found in R using this the recommenderlab package. The data was collected from the MovieLens web site from September 19th, 1997, through April 22nd, 1998. The data set contains about 100,000 ratings (1-5) from 943 users on 1664 movies. The format of MovieLense is an object of class realRatingMatrix. There are only three variables used from this data set and they are items – the title of the movie, user – id of the viewer of the movie, and ratings – movie ratings on a scale of 1-5: one being the lowest and 5 being the highest. Figure 1 displays the top 10 movies in the data set. Figure 2 and Figure 3 shows the number of ratings per user and the number of ratings per movie. They indicate that not all of the movies have been rated and not all the users have watched a large number of movies. This can suggest that there is some sparsity in the data since there are some movies that do not have ratings. Finally, Figure 4 is a distribution of the ratings given by the users. The table indicates that generally, users in this dataset do not give very high or very low ratings. The average rating in about 3.5 for this data set, however, the ratings are also spread evenly across the data set. A diverse range of ratings allows the recommendation system to learn from various user preferences and behaviors. This leads to a more comprehensive understanding of user-item interactions and helps in training a more accurate model. Additionally, the recommendation system can better personalize recommendations to individual users by identifying niche preferences and catering to them effectively, leading to a higher likelihood of user satisfaction.

An issue very common when creating recommendation systems is the problem of sparsity. Sparsity refers to the phenomenon where the majority of users have rated only a small fraction of items in the dataset and as a result, the user-item interaction matrix is sparse, containing mostly missing values. The percentage of sparsity in this data set is 93%, leaving only 7% of the cells available for analysis. There are various ways to deal with sparsity among data, including data preprocessing techniques, neighborhood-based methods, and matrix factorization techniques. This project employs data preprocessing to remove the empty cells from the data set, in order to mitigate the effects of sparsity. The empty cells are the cells with zero as a rating value. As a result, the analysis uses 99,392 user-item matrix to generate a recommendation system. Figures 5 displays the sparsity matrix. Figures 6 and 7 display the data before and after removing the empty cells.

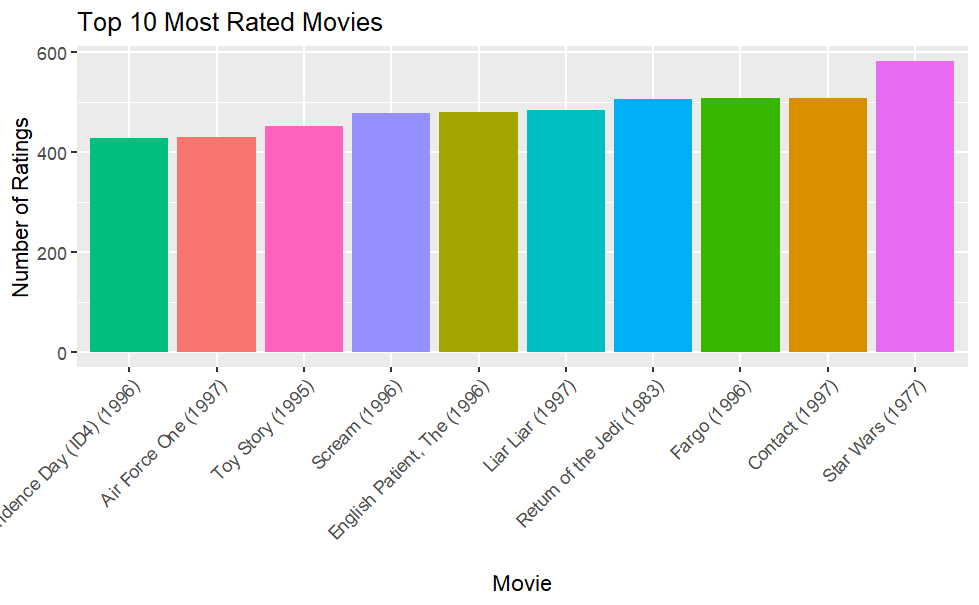


Figure 1 Top 10 Rated Movie

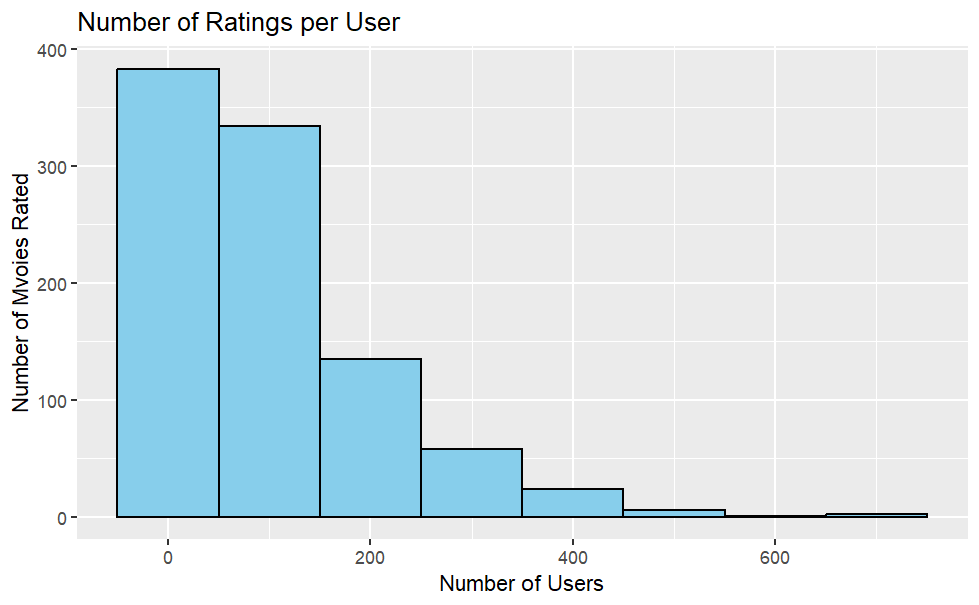
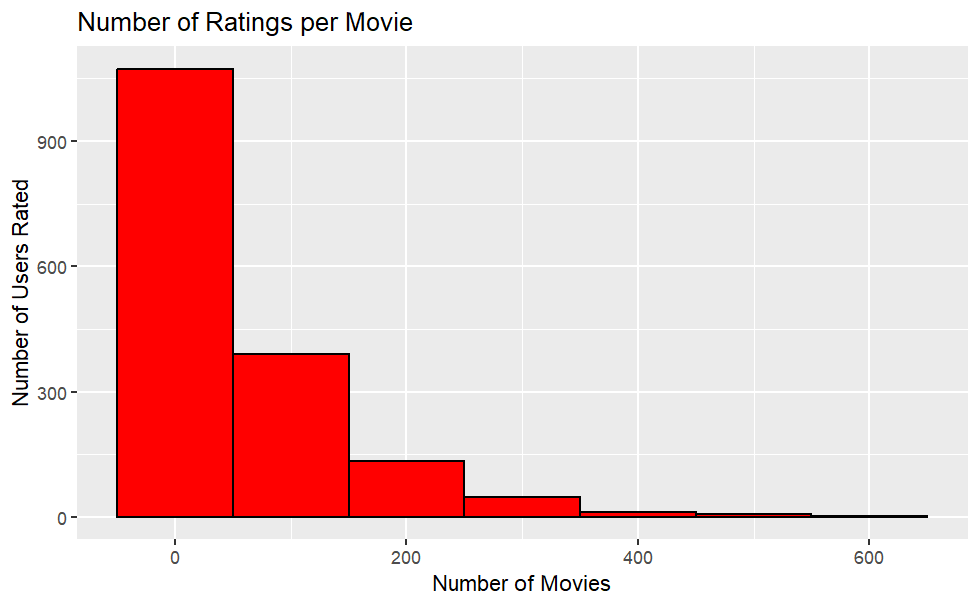


Figure 3 Ratings per movie

Figure 2 Number of Ratings per user.

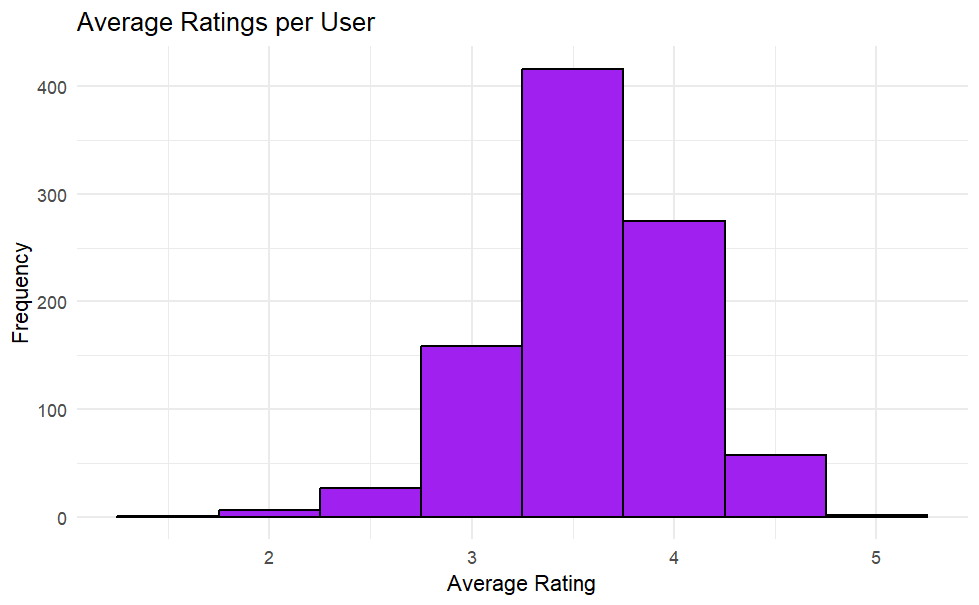
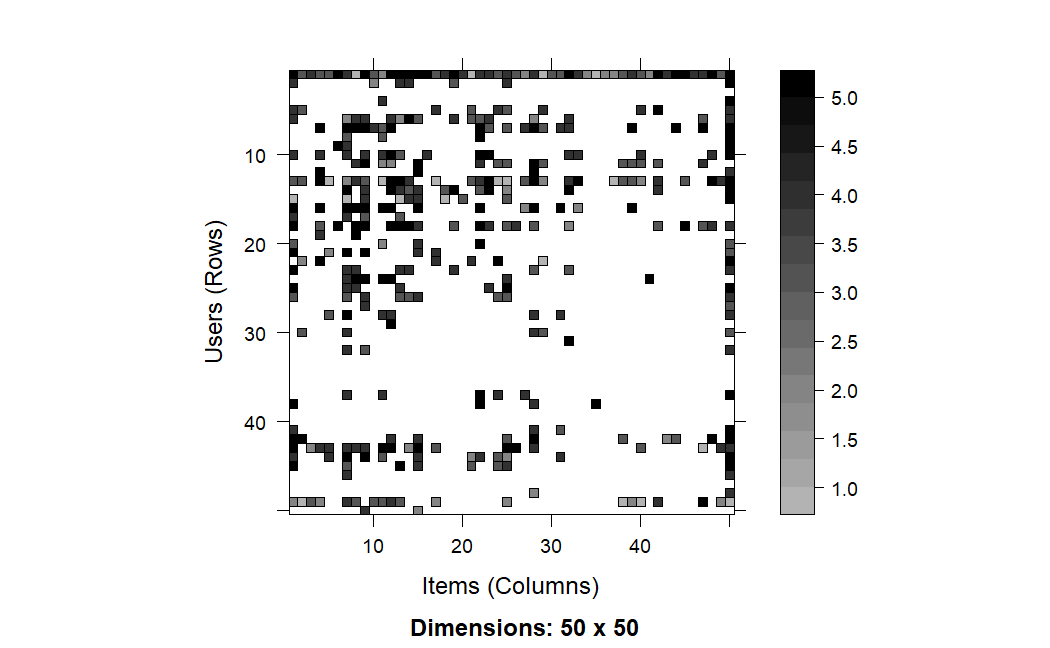


Figure 5 Sparsity Matrix of 50 users and movies.

Figure 4 Average Rating per movie



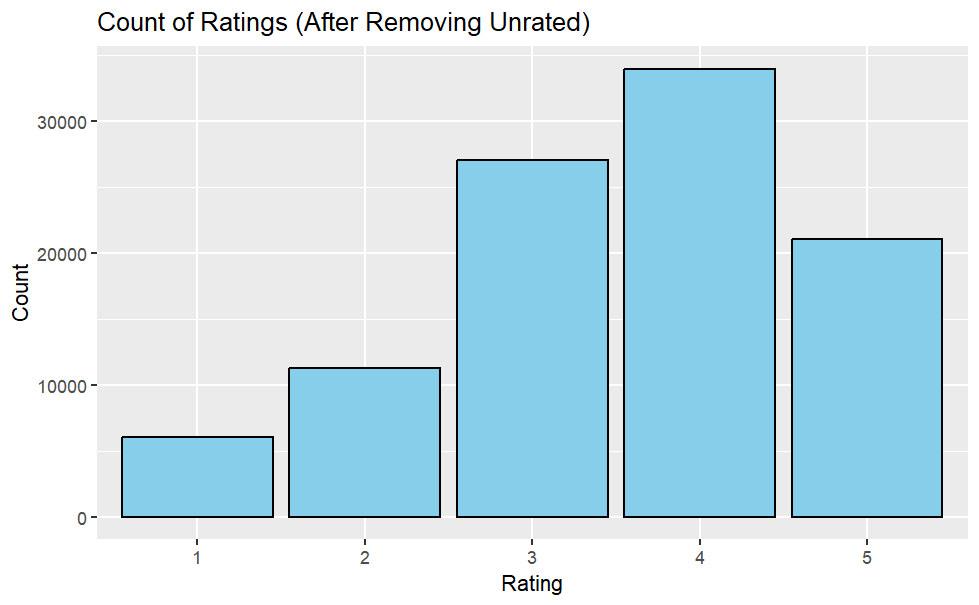
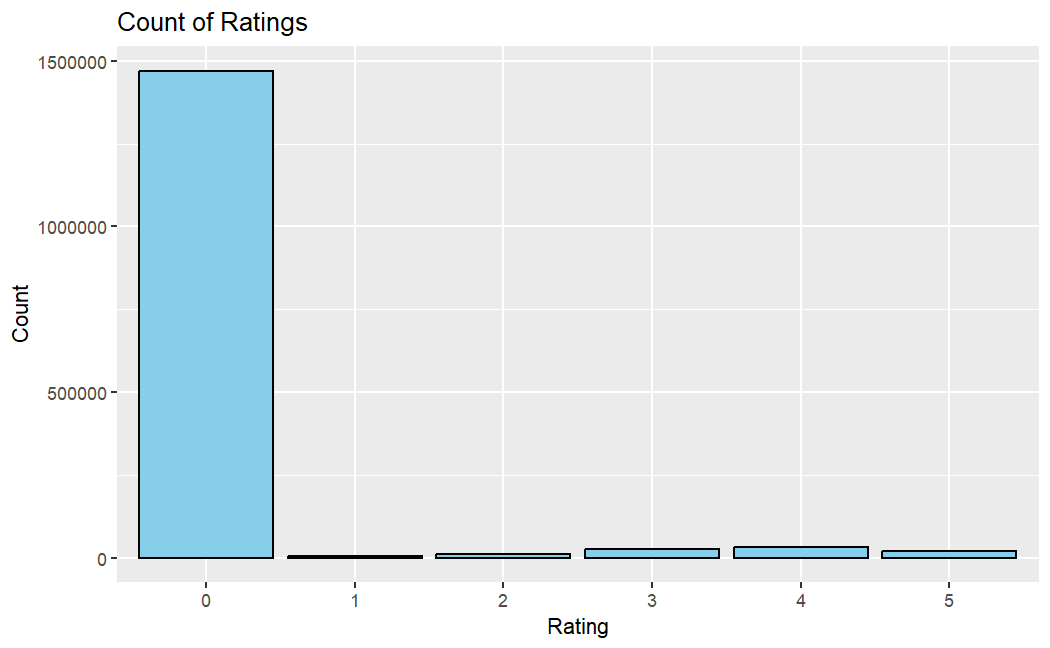


Figure 7 Ratings after removing unrated values

Figure 6 Ratings with unrated values

1. **Methodology**

This project uses the collaborative filtering system to build a movie recommendation system based on the MovieLense dataset. Collaborative filtering is a type of recommendation system that predicts user preferences for items (such as movies) by leveraging the preferences and behavior of a group of users. This method is used based on the assumption that users with similar historical tastes will exhibit similar preferences in the future. There are two main approaches to collaborative filtering: User-Based Collaborative Filtering (UBCF) and Item-Based Collaborative Filtering (IBCF). UBCF makes recommendations by comparing the similarities between two users, in this case, the algorithm will compare the similarities of the users based on the ratings the give movies. Whereas item-based collaborative filtering makes recommendations to a user by comparing how similar the items (movies) they review are.

This project employs the UBCF and IBCF algorithms found in the recommender lab package provided by R studio. The default algorithm works by computing the cosine similarities between the users and items, a higher cosine similarity score indicates a stronger similarity between the users and movies. However, there are other forms of correlation comparisons this algorithm uses are Pearson correlation coefficient, or Jaccard similarity.

1. **Results**

After creating a test model using the 70/30 test-train split, the algorithm successfully predicted movies for a user based on the UBSF and IBCF. The evaluation results indicate the performance of the two collaborative filtering algorithms. The user-based model exhibits better accuracy with an RMSE (Root Mean Squared Error) of approximately 1.20, indicating that its predictions are on average about 1.20 rating units away from the actual ratings. Additionally, its MSE (Mean Squared Error) of approximately 1.43 and MAE (Mean Absolute Error) of about 0.94 further suggests its reliability in providing accurate predictions. On the other hand, the item-based model demonstrates slightly lower predictive accuracy, as seen by its higher RMSE of approximately 1.51, MSE of approximately 2.27, and MAE of around 1.09. Overa both user-based and item-based collaborative filtering models provide predictions of user ratings, but the user-based model appears to have slightly better predictive accuracy. Listed below are the results of the recommendations provided to the same user using both collaborative filtering systems.

Top 10 Item Based Recommendations for a user.

$`0`

[1] "Braindead (1992)" "Pather Panchali (1955)"

[3] "World of Apu, The (Apur Sansar) (1959)" "Last Supper, The (1995)"

[5] "In the Bleak Midwinter (1995)" "Ruling Class, The (1972)"

[7] "MatchMaker, The (1997)" "Laura (1944)"

[9] "In the Company of Men (1997)" "Good Will Hunting (1997)"

Top 10 User Based Recommendations for a user

[1] "Hugo Pool (1997)" "Gilligan's Island: The Movie (1998)"

[3] "Ill Gotten Gains (1997)" "Legal Deceit (1997)"

[5] "Mighty, The (1998)" "Men of Means (1998)"

[7] "Condition Red (1995)" "Price Above Rubies, A (1998)"

[9] "Men With Guns (1997)" "Mirage (1995)"

1. **Conclusion**

Recommendation systems are powerful tools used to assists users and companies in making decisions when faced with a variety of choices. They are useful for helping companies understand users and more accurately cater to their every changing needs. There are varying methods of building recommendations systems, the project employs the collaborative filtering system. Modern companies use a hybrid approach which enables them to utilize the best features of different methods to provide a more accurate outcome. The results indicate that the user-based model tends to outperform the item-based model in terms of predictive accuracy, as evidenced by lower RMSE, MSE, and MAE values. This suggests that leveraging similarities between users to make predictions yields more accurate results compared to similarities between items. However, other research show stronger evidence that item-based models have better prediction accuracy. It is important to note that the effectiveness of collaborative filtering depends on the characteristics of the dataset and the specific recommendation task. Further experimentation and analysis are necessary to gain a deeper understanding of the nuances between these models and to make informed decisions regarding their effectiveness. Additional research is also necessary to compare the different types of recommendation system to determine the system with the most accurate predictions.