Jennifer Markham
Capstone 3: Movie Recommendation Final Report
Springboard Data Science Bootcamp

I. Background/Motivation

In today’s world, users have a plethora of choices in terms of what content to watch and where - with streaming services, Youtube, standard cable/dish TV- it can be overwhelming. Users need a simple tool that can help guide them in terms of which content they would be most likely to enjoy to help sift through everything that is available. The goal of this study was to leverage 3 datasets from MovieLens that contain a variety of movies with plot overview, genre information, and user ratings to create a recommendation system for users to determine what they would like to watch. Various methods were employed including non-personalized recommendations based on general movie popularity, movie-attribute based recommendations, and user-based recommendations.

II. Exploratory Data Analysis

Before creating a recommendation system, exploratory data analysis was performed to understand the data better. The user ratings data was reviewed first. Histograms were created to visualize the following: number of movies reviewed per user, top-25 movies with the highest number of ratings, and the distribution of user rating values (Fig. 1-3). Most users reviewed between 0-250 movies and gave them a rating between 3-5. The top-3 movies with the greatest number of reviews were Forrest Gump, Pulp Fiction, and Jurassic Park.

Additionally, the user ratings dataframe was reviewed for sparsity to get an understanding of how many observations were missing - i.e. how many movies did not receive user reviews. The concept of sparsity is important for quantifying the amount of data in a matrix as it will be difficult to compare ratings other users had for the same movie if no other users have rated this movie. The formula for sparsity and density (the opposite of sparsity, how many cells are populated in a matrix) are listed below:

$$\text{Sparsity} = (\text{Empty Values})/(\text{Total Cells})$$

$$\text{Density} = 1 - \text{Sparsity}$$

The sparsity and density values were calculated for the user ratings of various movies dataframe and are listed below:

Sparsity: 0.98258, Density: 0.01742

The user ratings data had a high sparsity, indicating a large number of possible observations were missing so it will be important to address this missing data when generating a recommendation system.

Lastly, the movie plot overview and genre data was also reviewed. Histograms were created to visualize the distribution of the lengths of the movie plot overviews and genre ratings in the data (Fig. 4-5). These showed that most movie plot overviews were between 200-400 characters and the most common genres of reviewed movies were comedies and dramas.

III. Modeling

A. Non-personalized recommendation methods

Non-personalized recommendation methods rely upon general popularity trends seen in the movies dataset. These types of recommendations are not relevant to a movie's attributes or a specific user's preferences and thus are typically less successful. The types of non-personalized recommendation methods tried in this project are listed below. A brief summary of each method and its pros/cons are listed below:

1. Movies with Greatest Number of Reviews

This method doesn't take into account user ratings and assumes that just because many users have watched a movie that it must be popular.

2. Movies with Highest Average Rating

This method takes user preferences into account by considering the user ratings; however, it doesn't take into account how many users have reviewed the movie. For example, a movie may only have been reviewed one time by a user that gave it a very high 5 star rating. This movie would falsely appear as being more popular but in reality, it is just this one user that really enjoyed the movie and it doesn't have a broad appeal to a wider audience.

3. Movies with Greater than 50 Reviews with Highest Rating

This method improves upon the previous method by filtering to include only movies with 50 or more reviews and helps to better guarantee that movies with higher average ratings are actually more likely to be enjoyed by more people.

4. Movies Commonly Viewed Together by the Same User

This method improves upon previous methods by incorporating some consideration to specific users by looking for common pairings of movies seen by the same user. However, it still lacks to integrate movie or specific user attribute information- like ratings info. (Fig. 6)

B. Item-based recommendation methods

Item-based recommendations rely upon attributes of an item, such as a movie’s genre or plot overview, to derive a recommendation. Several item-based recommendations were tried and they are listed below with a brief summary. Item-based recommendations are generally more insightful than non-personalized recommendations but they can usually be easily derived and do not generate results that are as unique as those seen in user-based recommendation methods.

1. Jaccard Similarity Scores Based on Movie Genres

   The Jaccard similarity score is a statistic to measure the similarity between two data sets. It is measured as the size of the intersection of two sets divided by the size of their union. This metric was used to determine the similarity between 2 movies based on their genre data.

2. Term frequency inverse document frequency & cosine similarity on Movie Plot Overviews

   The TF-IDF technique is very useful in text applications. The formula reduces the weight of common words that occur in many documents and increases the weight of words that do not occur in many documents, which is ideal as we do not want to increase the weight of common words like “the”, “and”, and “or” that may be common in language but are not important to distinguish its content..This was used to measure how important a term is within a movie text overview versus the entire set of movie text overviews. Cosine similarity measures the similarity between two vectors of an inner product space. It is measured by the cosine of the angle between two vectors and determines whether two vectors are pointing in roughly the same direction. It is often used to measure document similarity in text analysis as it is a helpful similarity metric to use when items have more variation in their data, which is what was done in this case with our TF-IDF data.

C. User-based recommendation methods

1. Term frequency inverse document frequency to Generate User Profiles Based on Movie Plot Overviews

   Determine words associated with plot overviews of movies reviewed by each user in an effort to garner insights about what topics are common among movies viewed by that user.

2. Cosine Similarities of User Review Data

   Review the cosine similarity of sets of movie review data per user. This assumes that users with similar review data, would have similar tastes.

3. K-Nearest Neighbors Regression to Predict User’s Movie Ratings

   This method finds the k users that are closest measured by a specific metric, in this case cosine similarity was used, to the user in question. Then it averages the rating those users gave the movie we are trying to get a rating for. This method is helpful because it allows us to predict a user's rating for a movie even if they have not yet seen that movie.

D. Singular Vector Decomposition/Matrix Factorization to Predict User's Movie Ratings

This method addressed the sparsity issue seen in the exploratory analysis of the user ratings data. This method permits all values for user ratings to be filled in per movie so it will be more useful to derive user preferences.

IV. Conclusion

Numerous methods can be used to generate recommendation systems and each methodology has its merits and disadvantages. Non-personalized recommendations are easy to implement however, each method employed in this study has drawbacks which are important to consider if this is deployed in a real-world setting. Item-based recommendations are again fairly facile to implement and do not require as frequent of updating (i.e. item inventory does not change as frequently as a user-base may), however the issues garnered from these types of recommendations are typically less useful than those that come from user-based recommendations. Lastly, user-based recommendations are typically more complex/costly to collect as the data isn't always readily available and requires more frequent updating as user bases change regularly. User-based recommendations can provide more unexpected results than item-attribute recommendations so they can be an incredibly valuable marketing tool.

In this study examples of the above methods were employed, to evaluate which recommendation system is superior, predicted ratings for a user/movie were generated using the K-Nearest Neighbors Regressor and SVD/Matrix Factorization user-based recommendation methods. The K-Nearest Neighbors Regressor used cosine similarity as the metric and 3 for the number of neighbors to compare. The rating values predicted were then compared to the actual rating the user gave this specific movie using the root-mean square error. The results are summarized below.

K-Nearest Neighbors Regressor vs Actual Ratings : RMSE = 2.0

SVD/Matrix Factorization vs Actual Ratings : RMSE = 1.898

Based on these findings, the SVD/Matrix Factorization recommendation system is superior and was able to generate a user rating with less error associated. This testing was completed on a very small user sample and should be expanded for future work. Additionally, further work could include optimizing each of these models- K-Nearest Neighbors Regressor and SVD/Matrix Factorization. For example, different numbers of neighbors could be tried to determine if the error between predicted and actual user ratings could be reduced. Similarly, when constructing the SVD on the original user ratings matrix, a number of latent features needed to be specified. In this study, the default of 6 latent features was used however testing different numbers of latent features could improve results and improve predicted ratings.

V. Appendix - Figures

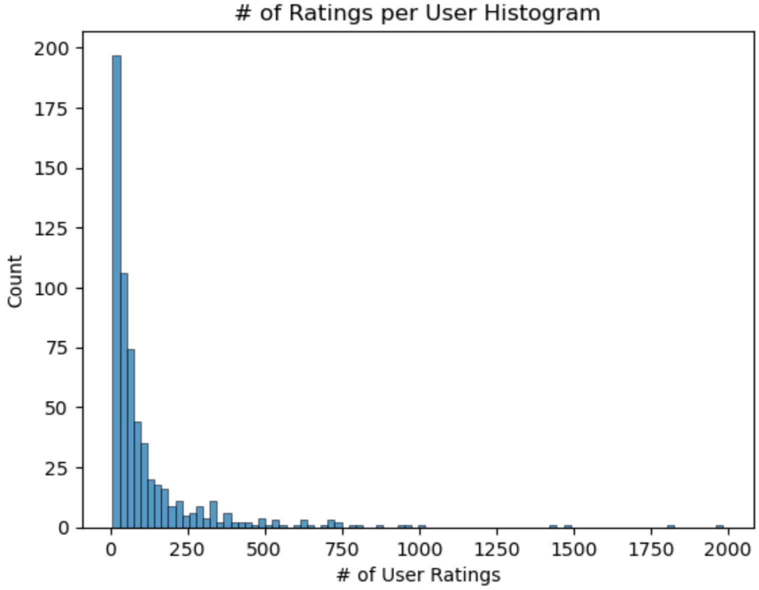

Figure 1. Histogram of number of ratings per user.

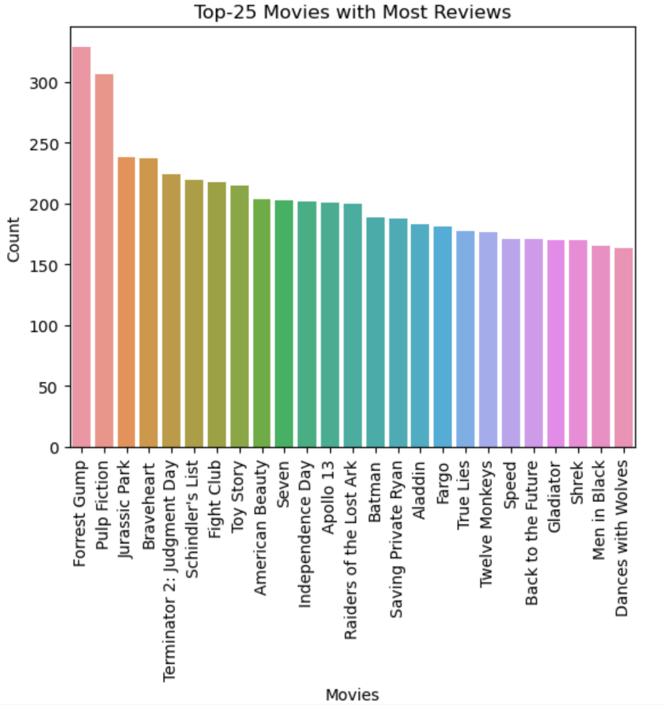

Figure 2. Histogram showing top-25 movies with greatest number of reviews.

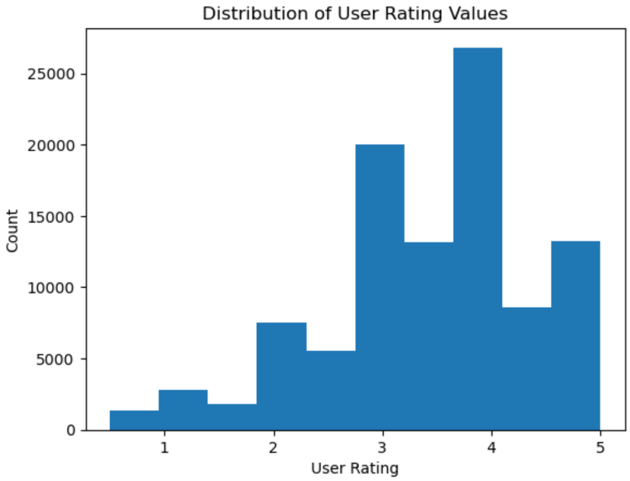

Figure 3. Distribution of user ratings for movies, typically movies were well-liked receiving ratings of 3-5.

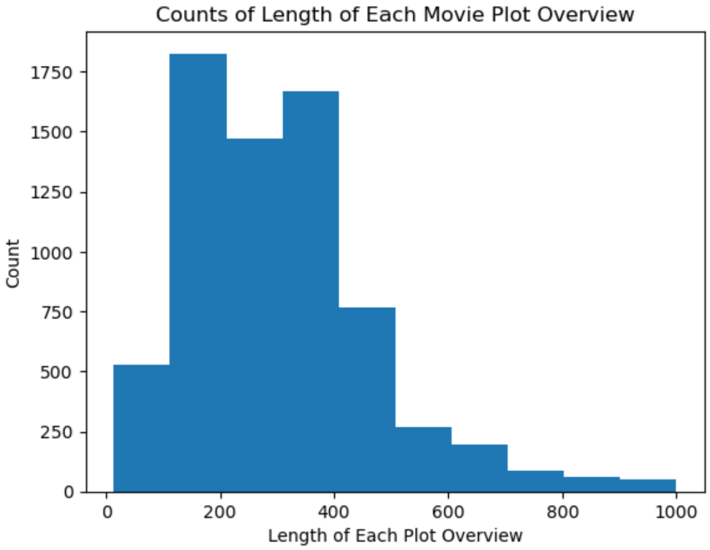

Figure 4. Distribution of lengths of movie plot overviews, typically plot overviews for each movie were between 200-400 characters.

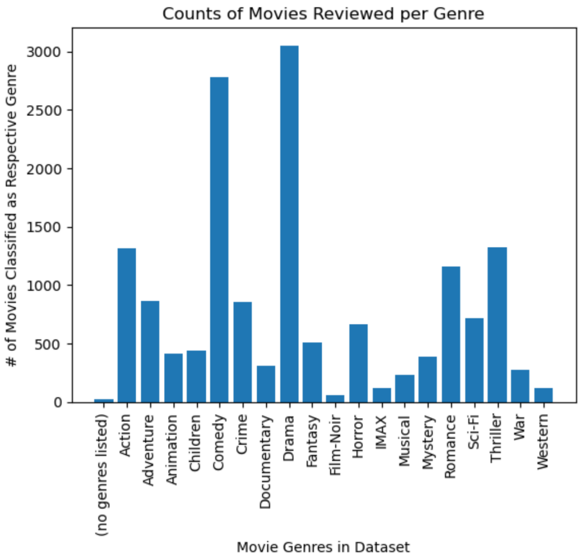

Figure 5. Histogram of genres of movies included in the dataset, the most common genre classifications were comedy and drama. Many movies were classified as more than 1 genre.

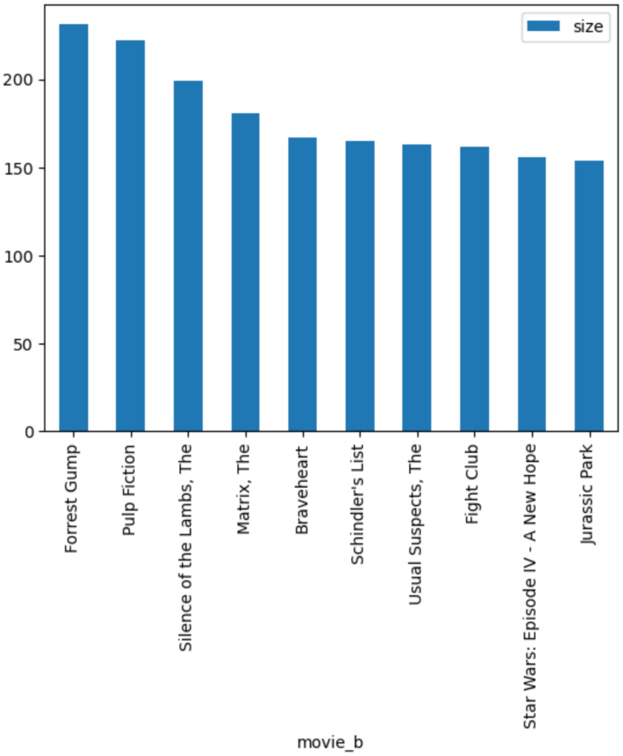

Figure 6. Histogram showing the top-10 movies most commonly viewed by users who have also seen The Shawshank Redemption.