**Movie Recommendation System Using R**

**ALY6040 Data Mining Applications**

Apoorva Anumalla

Harsh Natvarlal Sindhwa

Rajani Kumari Jangiti

Vamshi Reddy Komatireddy



College of Professional Studies, Northeastern University

Under the guidance of Prof. Amin Karimpour

**Abstract**

A Recommendation structure is a sort of data isolating framework that tries to foresee the propensities of a customer and make proposals dependent on these inclinations. There is a wide variety of usage of these structures. These have gotten intelligently standard over the range of the most recent couple of years and are now being utilized in most online stages that we use. Hence, we are utilizing these recommendation systems in order to improve movie predictions based on the interests of user.

**Movie Recommendation System**

Movie Recommendation system is a prediction system which understands the user interests on movies, based on the data collected and anticipates the kind of movies they would like to watch in the next usages. These systems have become very important and standard in many online platforms like Netflix, Amazon Prime, Youtube, etc.

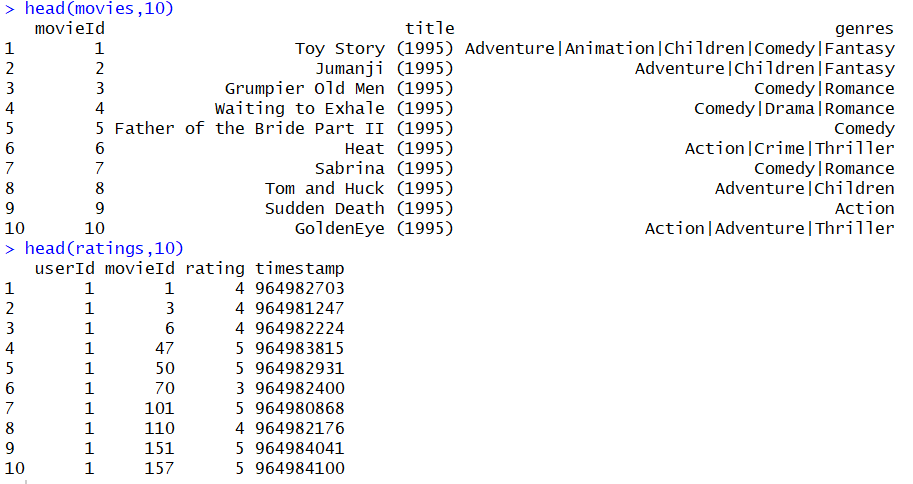
# Recommendation System

Recommendations are basically of two types. The first system is Content based filtering. It utilizes the client’s profile data, their activities, especially recent activities, and then predicts/suggests movies which might grab the user’s attention. The other majorly used recommendation system is Collaborative based filtering. This method gathers data of clients who watches similar kinds of movies, and are comparable, and then suggests or proposes the new items based on the likability’s of each client.

For our “Movie Recommendation System”, we have considered the Content based filtering structure to propose new movies to the users.

## Analysis

For applying this structure, a database from Grouplens.org has been considered. This dataset has information of about 6000 users, covering more than 100000 evaluations from 9000 movies. The whole data is divided into two sets, where the first set carries the data of all the movies along with their names, genres, and ids. The other data set has client appraisals for any movie, their rating from 1-5. The data set is partitioned into test and training data in order to reduce the aggregate of squared mistake between the real evaluations and the anticipated reviews/ratings. The whole framework is meager, as the users have not completely reviewed all kind of movies available.



**Data Cleaning**

In movies dataset there were few empty values in the genres column. So performed data cleaning over the dataset. We replaced those empty sting values with “no genres listed”.



### EDA

To reveal any hidden patterns in the data, we are utilizing EDA (Exploratory Data Analysis), for the analysis. The basic analysis is concentrated on locating the outline of dataset, understanding the most famous or mostly reviewed motion pictures, and identifying the most elevated motion pictures and the patterns of clients watching them.

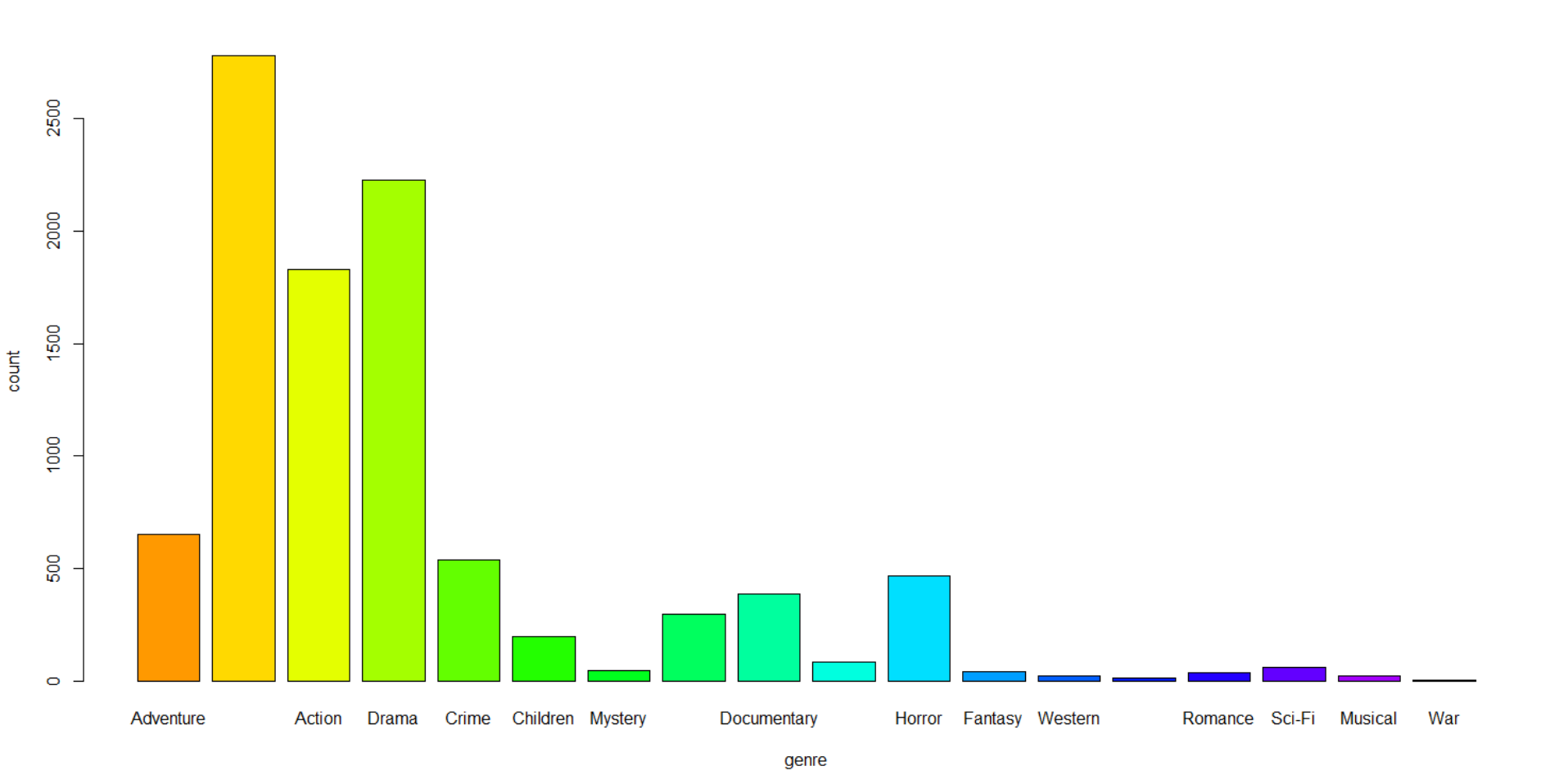
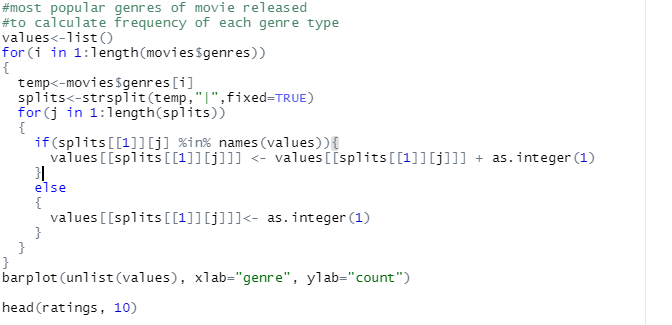


Figure 1: The count of movies based on different genres

Figure 1 is the plot against the genre and the count of movies under that genre.



The frequency of all the available genres are calculated using the above code.

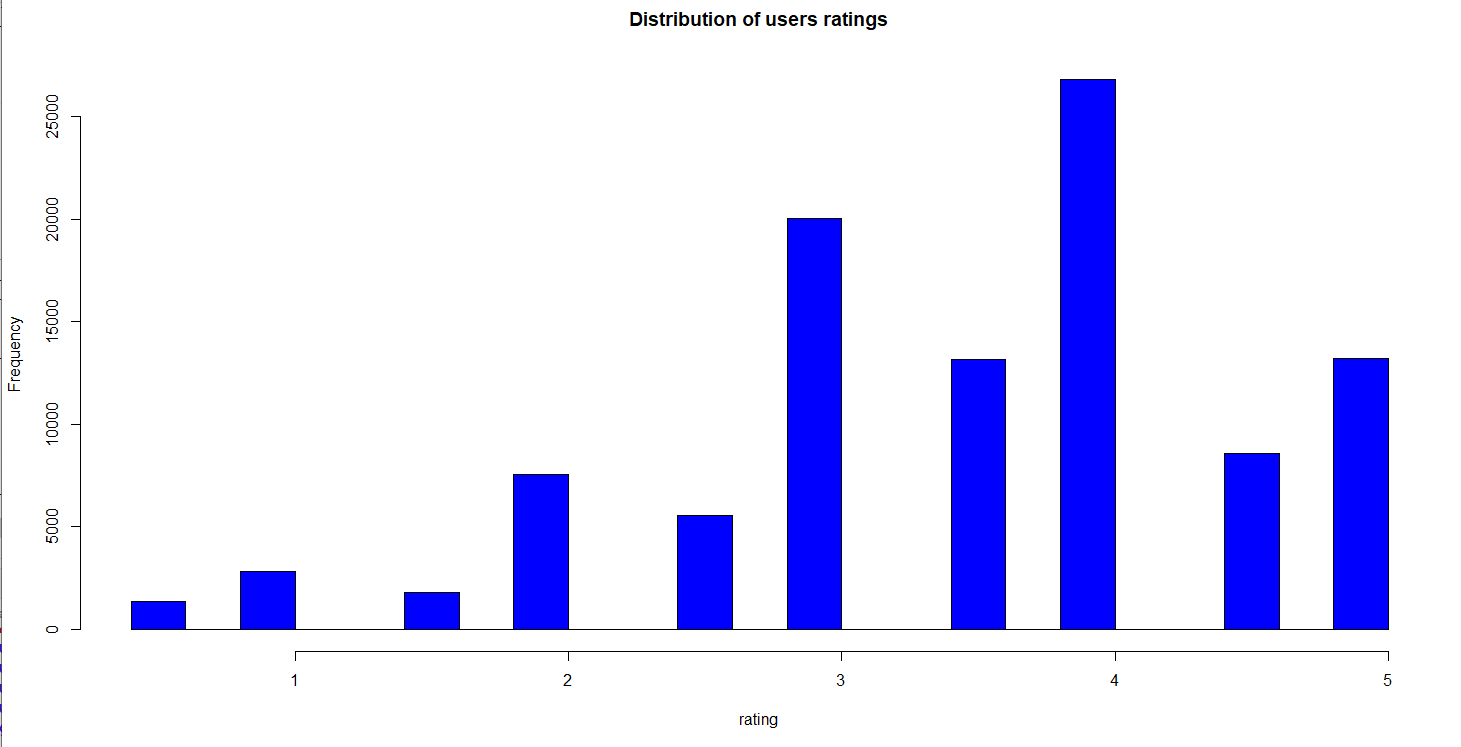
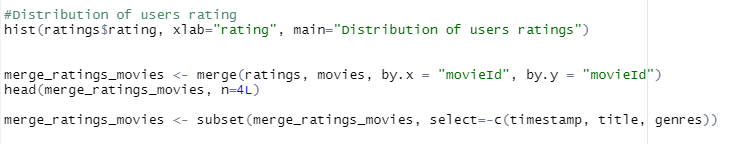


Figure 2: Distribution of user ratings

Figure 2 reveals the frequency of user ratings from 1-5 and it is seen that more than 2500 people have given a rating of 4 and at least 1000 people.



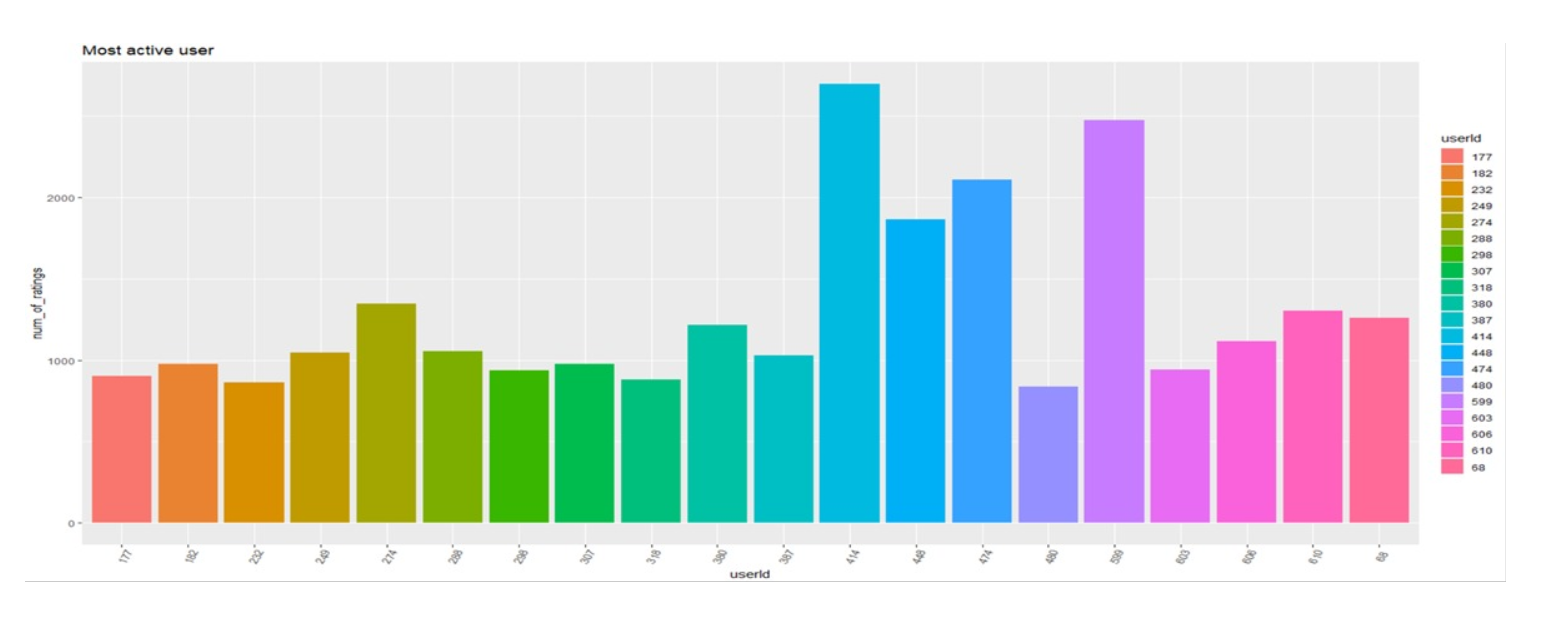
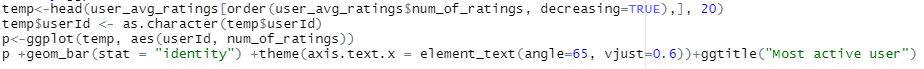


Figure 3: Most Active User

Figure 3 shows which user has been most active and has rated the movies.



The top 20 most active users and their frequency of watching movies is calculated using the above code.

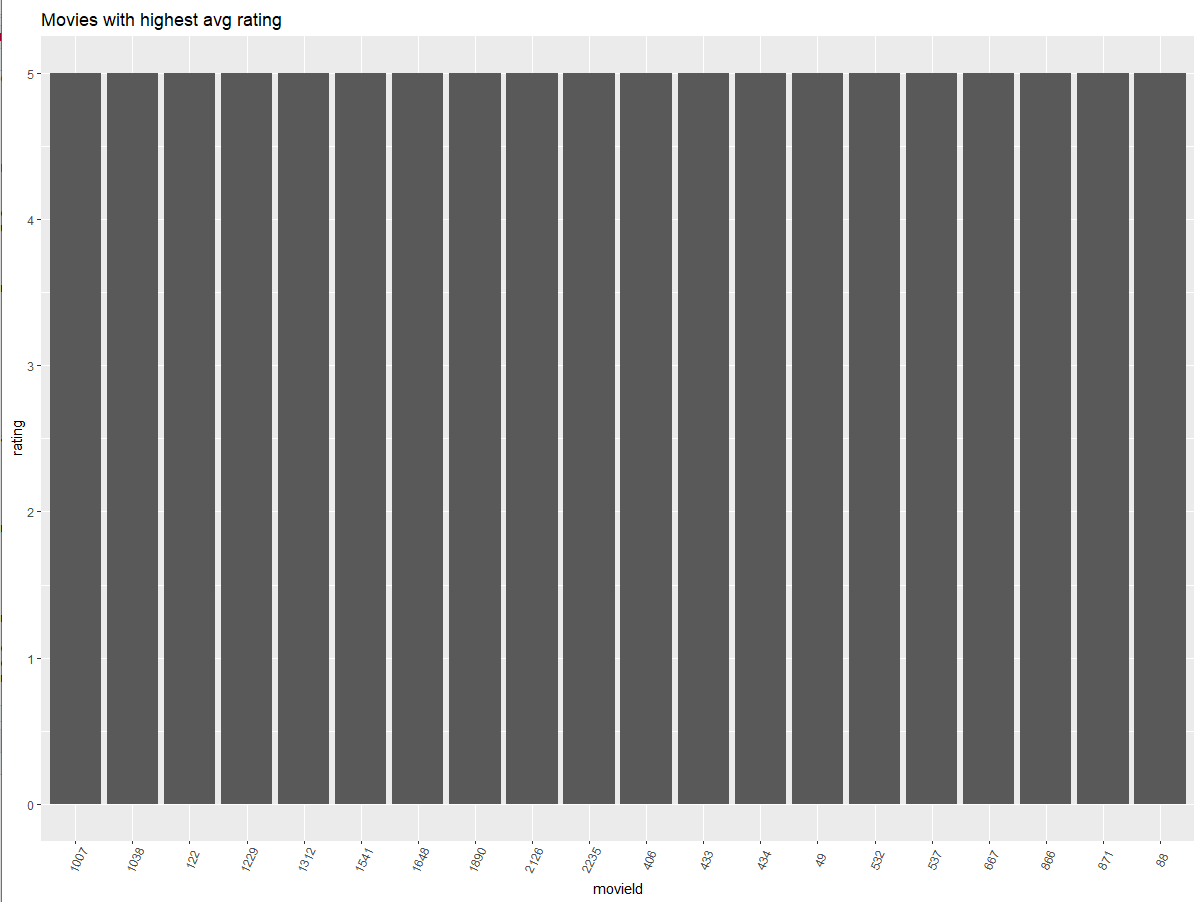
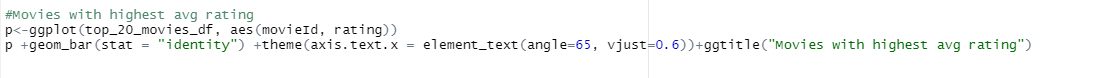


Figure 4: Movies with highest Average rating

Figure 4 shows the movies with their ids, which have the highest possible rating. Below code is used to identify above pattern.



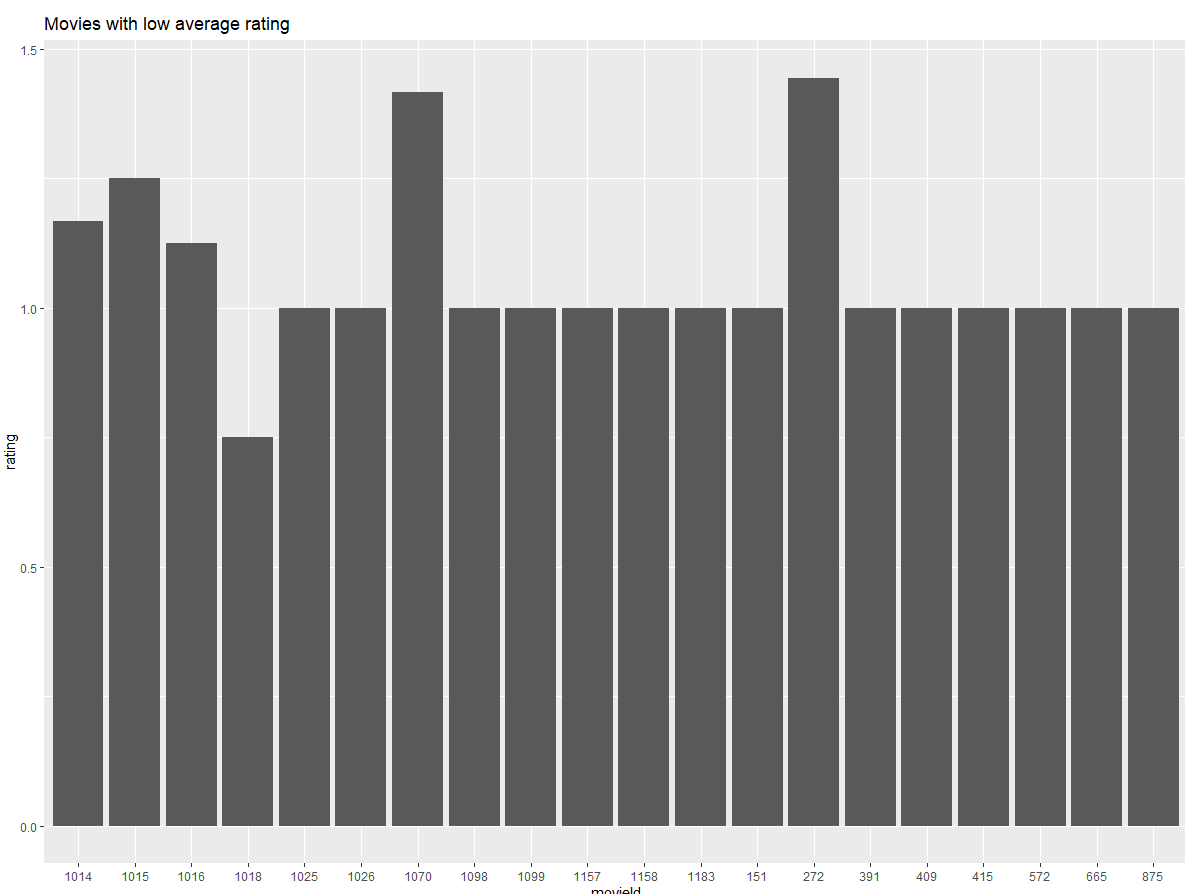
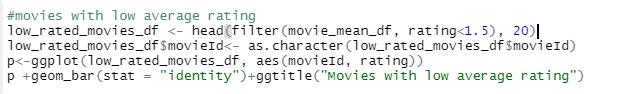


Figure 5: Movies with lowest average rating

Figure 5 clearly reveals the ids of movies whose rating stand on or below 1.5 on a total possible rating of 5. And the lowest ever average seen here is between 0.5 to 1.0.

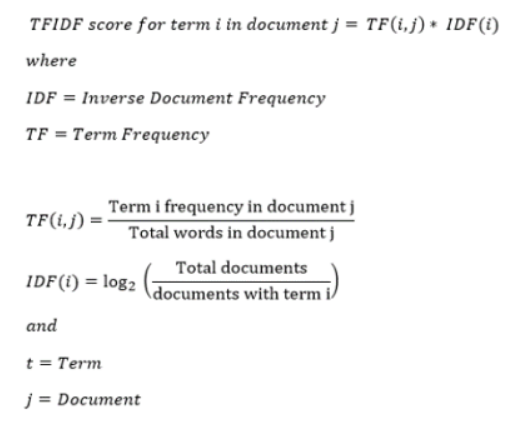


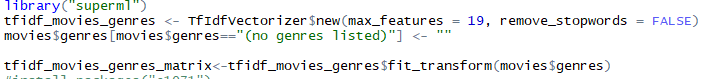
Top 20 movies which have ratings less than 1.5 were requested with this code.

#### Content Based Filtering

Term Frequency (TF) and Inverse Document Frequency(IDF) are the two terms that are used in information retrieval systems and content based filtering mechanisms. They help in finding out the relative importance of a document/article/movie or a news item.

The Term Frequency (TF) is the calculated frequency of a word in a document whereas Inverse Document Frequency (IDF) is the inverse document frequency among the whole corpus of documents. For instance, if we search for “What Is the best cartoon show?” on any search engine. It is sure to pull out that the most frequently occurred word would either be ‘the’ or ‘best’ rather than ‘cartoon’. But ‘cartoon’ has higher importance of our search. Hence, TF-IDF weighting negates the effect of high frequency words in determining the importance of an item.





In content-based isolating, proposals are made considering relationships between movie profile and client profile. A client profile is content that is viewed as appropriate to the client in kind of catchphrases or features. A client profile might be viewed as a ton of consigned watchwords (terms, features) assembled by figuring from things discovered captivating by the customer. Item profile includes the features of items.

For instance, if the customer watched a movie of fiction genre and rated it with a 4 or 5, then the model proposes some other fiction-based movies to the user, as per his interest. We will use the cosine similitude to calculate a numeric sum that demonstrates the resemblance between two films. Deductively, it is described as follows:

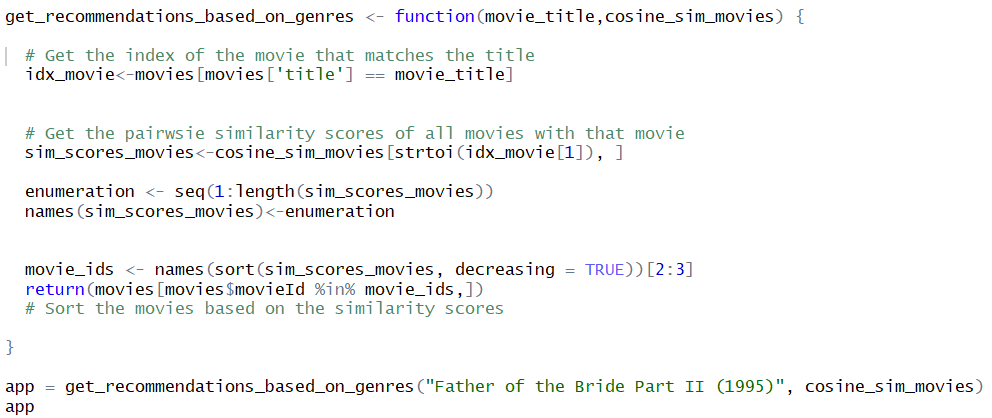
Similarity=

where,

a.b -> Dot product of two vectors.

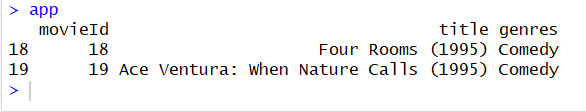
-> Product of magnitude of each vector

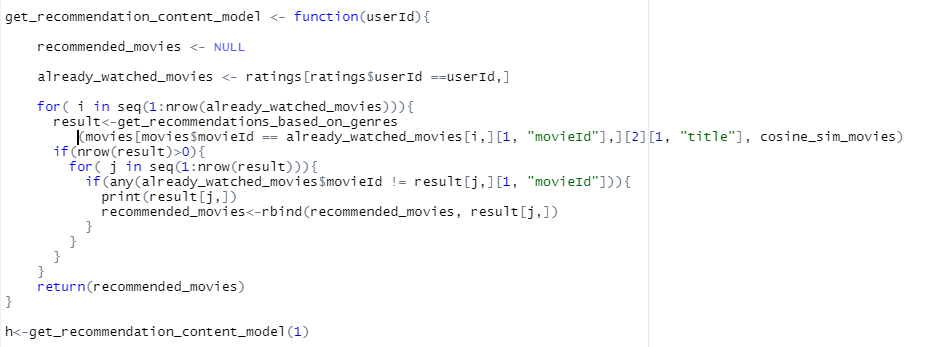
We use the cosine likeness score since it is autonomous of greatness and is generally simple and quick to figure.



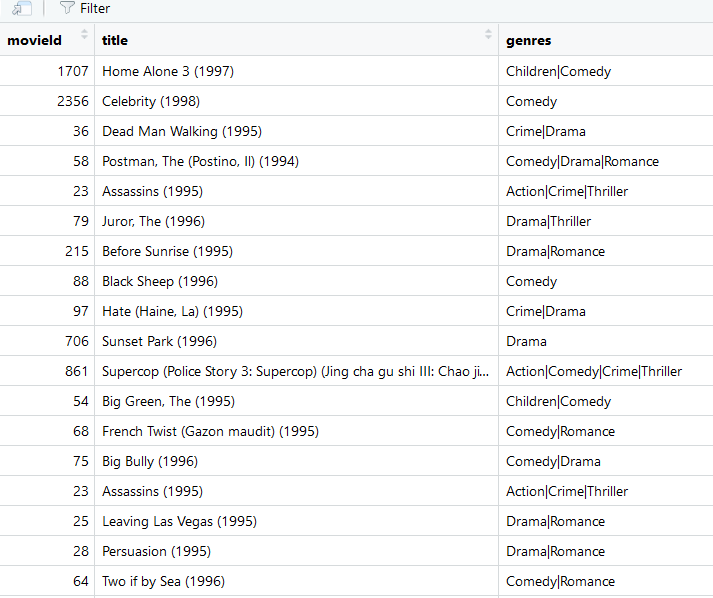
Here, the similarity scores are calculated for the movies, based on the title and then the movies are sorted based on the scores. Below is the obtained result when movies are calculated for cosine value based on the title of “Father of Bride Part II (1995)”.

And the results are as follows:





The recommendations that are shown are as follows:



#### Evaluating the Model

Created a function where we can classify the movie when a movie id is passed. We have used a simple KNN classifier to predict the recommended movie.

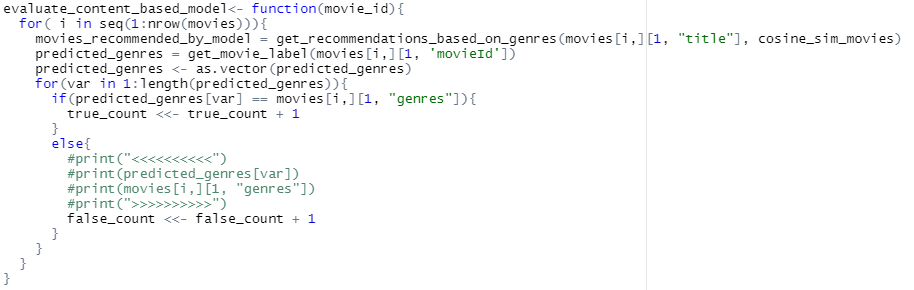
**KNN**

KNN is a simple supervised machine learning algorithm which can solve any classification problem. In KNN the value K is found out using the elbow method. The elbow method gives out the most significant value of K which helps in deciding the nearest neighbors to classify the data. We have chosen the value of K as 5 which gave us better results in the model.

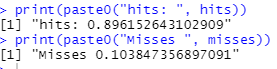
The get\_movie\_label function returns the predicted values by taking the valid movie id argument which is eventually performed by the KNN classifier.



Then we tried to evaluate the classifier by using evaluate\_content\_model to see how accurate results we obtained using the knn classifier by calculating the hit and miss results.



We then printed the hit count and miss count values to find how accurate is our model. We achieved an accuracy of 89%.



#### Conclusion

The following conclusion are noted from this project:

* The highest number of movies has ‘Comedy’ and ‘Drama’ genres.
* The analysis was conducted, and the results were visualized.
* The lowest number of movies were of ‘War’ genre.
* The advantage of content-based recommendation system is that the movies that are not rated can be recommended and it is not dependent the ratings of users to make predictions.
* The disadvantage of content- based recommendation system is that it doesn’t work properly for new users that have never rated any movie.

#### Future Scope

The recommendation system can be improved by using other two filtering methods which are Demographic/simple filtering, and Collaborative filtering techniques.

**Demographic/simple filtering:** It offers summarized recommendations to every client, in light of film pervasiveness just as type. The model prescribes similar motion pictures to customers with comparative socioeconomics. Since each customer is unique and has their own preferences, this system is seen as exorbitantly essential. The crucial idea behind this structure is that films that are progressively standard and generally adulated will have a higher probability of being delighted in by the ordinary group. With this filter, there are chances for high recommendations with consideration of many features.

**Collaborative Filtering**: Content-based sifting techniques are only skillful to recommend films subject to their likenesses to various movies. The significant assumption behind the network situated isolating technique is that similar customer tendencies over the items could be abused to endorse those things to a customer who has not seen or used it already. In simpler terms, we acknowledge that customers who agreed in the past will agree later. Regardless, a couple of issues remain for this strategy. Regardless, the rule issue is adaptability. The count creates with both the customer and the thing. Similarly, sparsity is another concern. Thus, Collaborative filtering can have better impact on recommendations provided to the clients/users.

**References**

Tanner, G. (2019, February 12). Building a book Recommendation System using Keras. Retrieved May 15, 2020, from <https://towardsdatascience.com/building-a-book-recommendation-system-using-keras-1fba34180699>

Deep Learning based Recommender System: A Survey and New ... (n.d.). Retrieved May 15, 2020, from <https://arxiv.org/pdf/1707.07435.pdf>

Kordík, P. (2019, December 15). Machine Learning for Recommender systems - Part 1 (algorithms, evaluation and cold start). Retrieved May 15, 2020, from <https://medium.com/recombee-blog/machine-learning-for-recommender-systems-part-1-algorithms-evaluation-and-cold-start-6f696683d0ed>

Isinkaye, F., Folajimi, Y., & Ojokoh, B. (2015, August 20). Recommendation systems: Principles, methods and evaluation. Retrieved May 15, 2020, from <https://www.sciencedirect.com/science/article/pii/S1110866515000341>