# logo_bw.eps *Worcester Polytechnic Institute*

# *Data Science Program*

# Case Study 2

# Recommendation System

# Submitted By

# Sirshendu Ganguly Enbo Tian Dang Tran

## **Date Submitted :** 3/16/2022

## **Date Completed :** 3/16/2022

## **Course Instructor : Prof.** Ngan

# Motivation and Background

# The 1990s is known as the decade that produced the best movies in the world. We come with a question, why did so many of the great works come from the last century? We looked up the history and got the answer. There are two great periods of art, one World War II, one Cold War. The Cold War ended with the collapse of the Soviet Union in 1999. From suffering comes greatness art. However, in the first twenty years of this century, the mainstream of the world is peace and development. Peace requires recreation and amusement, not depth and pain. In addition, Movies at that time also can partly reflect the living condition of people at that time. In order to have a more honest understanding of people’s attitude towards life at the end of last century, we are going to focus on the dataset about movies from 1995 to 2000.

# Data Sources

# In this case study, We collect the MovieLens data set, which contains approximately 3,900 movies made by 6,040 MovieLens users who joined MovieLens in 2000. We get the data frame of Users, Movies, and the users’ Ratings of movies.

# In the data frame of Users, We have the user ID, gender, age, occupation, and zip-code of the users.

# Gender is denoted by ‘M’ale and ‘F’emale.

# Age is chosen from 7 ranges: 1( < 18), 18(18-28), 25(25-34), 35(35-44), 45(45-49), 50(50-55), 56(56+).

# We have 20 occupations and a non specified case: 0: "other" or not specified, 1: "academic / educator", 2: "artist", 3: "clerical / admin", 4: "college / grad student", 5: "customer service", 6: "doctor / health care", 7: "executive / managerial", 8: "farmer", 9: "homemaker", 10: "K-12 student", 11: "lawyer", 12: "programmer", 13: "retired”, 14: "sales / marketing", 15: "scientist", 16: "self-employed", 17: "technician / engineer", 18: "tradesman / craftsman", 19: "unemployed", 20: "writer"

In the data frame of Movies, We have Movie ID, the Title of movies, and Genres. Titles are identical to titles provided by IMDB and the year of release in the bracket. Genres are pipe-separated and are selected from Action, Adventure, Animation, Children's, Comedy, Crime, Documentary, Drama, Fantasy, Film-Noir, Horror, Musical, Mystery, Romance, Sci-Fi, Thriller, War, and Western.

# Methodology

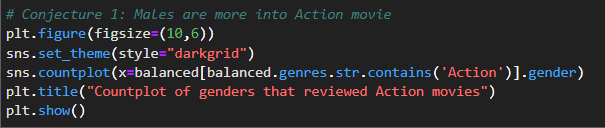
# Problem 1:

# 

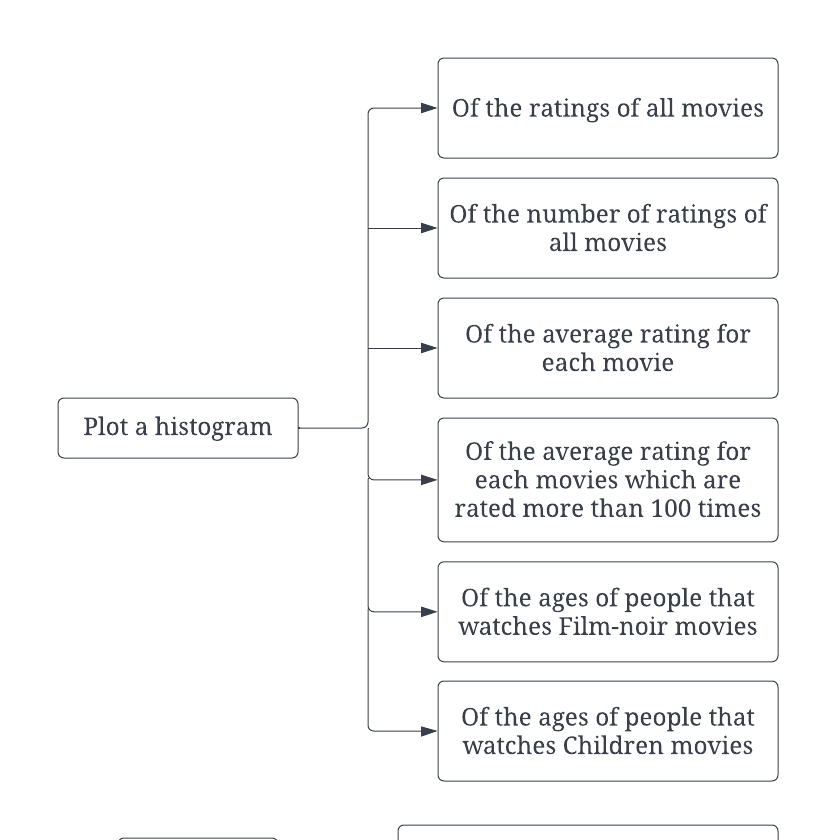
# To begin, we needed to import some necessary libraries. These include the numpy, pandas, matplotlib.pyplot, and seaborn libraries. To import the MovieLens dataset, we first need to read the data from the users, ratings, and movies files into separate dataframes using pandas read\_table() method. Then we merged them into a single Pandas dataframe with the merge() method and stored it in an HDF5 file with the to\_hdf() method.

We wrote some functions to collect basic details about the MovieLens dataset and print it out:

* Wrote a function that gets the counts of movies with an average rating over 4.5.
  + Takes in a dataframe.
  + Grouped the dataframe by the movie ids and their average with the groupby() and mean() methods. That’ll get us the average rating that each movie has.
  + Return the counts of the movies with an average rating of more than 4.5 by using conditional indexing and getting the number of rows from the shape() tuple.
* To get the count of movies with an average rating over 4.5 among men and women we simply passed in our dataframe with only the ratings made by either men or women.
  + For men, we passed in the dataframe with only the rows that have ‘M’ for their gender.
  + For men, we passed in the dataframe with only the rows that have ‘F’ for their gender.
* Wrote a function that gets the counts of movies with a median rating over 4.5.
  + Identical to our mean method but replace mean() by median().
* To get the count of movies with a median rating over 4.5 among men and women we once again passed in our dataframe with conditional indexing.
  + For men over 30, we passed in the dataframe with only the rows that have ‘M’ for their gender and an age larger than 30.
  + For women over 30, we passed in the dataframe with only the rows that have ‘F’ for their gender and an age larger than 30.
* We decided to use the number of reviews to determine the most popular movies. This works as the more reviews a movie has the more people have viewed that movie. However, the popularity of a movie determined by this method does not equate to the quality of the movie. Since some bad movies can create a lot of negative publicity around them which will arouse interest and cause more people to view them.
  + To actually get the count of reviews each movie has, first we group the dataframe by the title with their counts. Now each column has the number of counts/ratings that a movie has.
  + We kept only 1 column, ‘age’, and renamed it to ‘count’.
  + Then we simply sort the new dataframe by the count number to get the top 10 most popular movies.
* We made 2 conjectures that men are more into action movies and that women are more into romance movies.
  + First, we created a balanced dataset that contained the same number of reviews from men and women.
    - Get the total number of reviews by women with shape.
    - Get the reviews by men with the same number as women using the sample() method.
    - Concatenate all the reviews by women and the same number of reviews by men.
  + Visualized the counts of each gender that reviewed action movies and romance movies using the matplotlib and seaborn libraries.
    - Used the countplot() method from the seaborn library to get the counts of all the reviews made by each gender to a specific genre of movie.

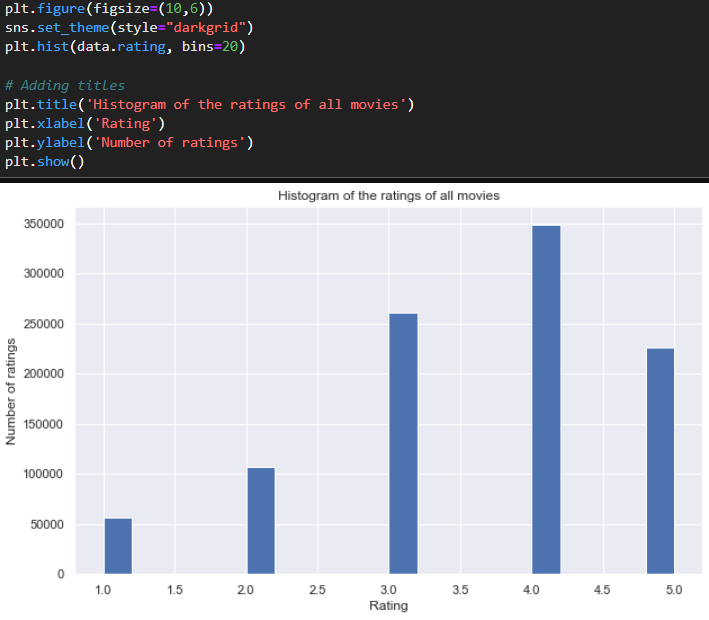


**Problem 2:**

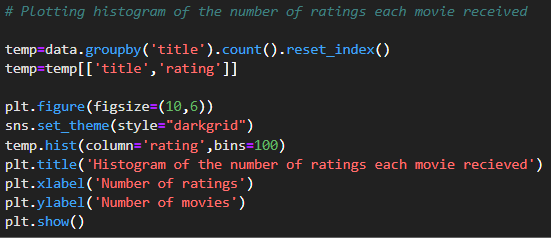


Used the matplotlib and seaborn libraries for histogram creations.

* Plotted a histogram of the ratings of all movies by passing in data.rating to the hist() method from matplotlib.

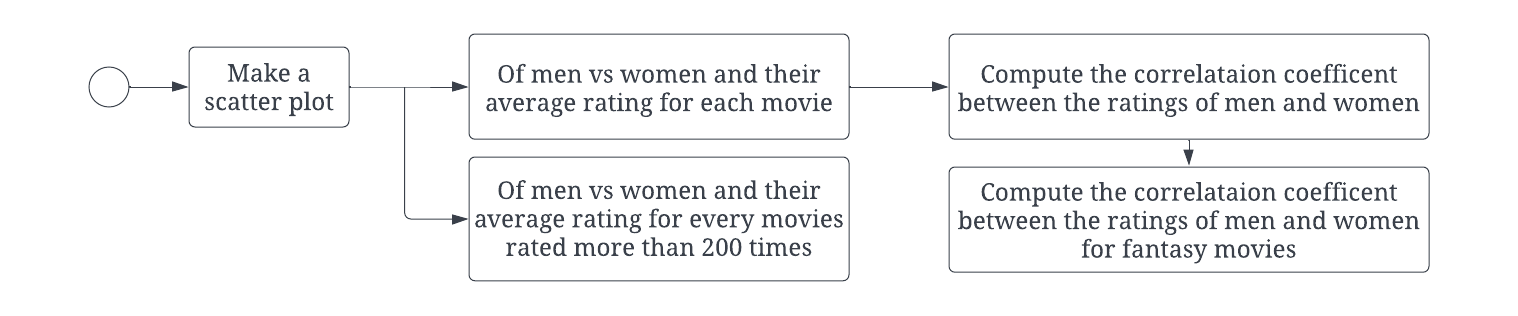


* Plotted a histogram of the number of ratings each movie received.
  + Get the rating count by grouping the original dataset by the titles then use pandas count() method.
  + Used seaborn and matplotlib to create a histogram.



* Plotted a histogram of the average rating for each movie.
  + Get the average rating each movie received by grouping the data frame by their movie ids with the mean of their ratings.
  + Pass in the resulting dataframe to the hist() method from matplotlib.
* Plotted a histogram of the average rating for movies that are rated more than 100 times.
  + Created a series of movie ids and the number of ratings they received using pandas values\_counts().
  + Get the index/movie id of all the movies that have rating counts larger than 100.
  + Create a new dataframe containing only the movies that have more than 100 ratings by using isin() with the list of ids from the previous step.
  + The rest is the same as the steps for plotting a histogram of the average rating for each movie.
* Conjecture 1: People older than 56 years old are more into Film-Noir movies
  + Create a balanced dataset where we sample the same number of ratings from each age range.
  + Create a subset of the data frame containing only ratings for Film-Noir movies then plot it out in a bar graph.
* Conjecture 2: Children younger than 18 years old are more into Children’s movies
  + Create a subset of the data frame containing only ratings for Children’s movies then plot it out in a bar graph.

**Problem 3:**

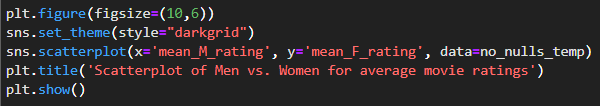


To make a scatter plot we would require a dataframe like the following:

| movie\_id | mean\_F\_rating | mean\_M\_rating |
| --- | --- | --- |
| X | aa | bb |
| Y | cc | dd |

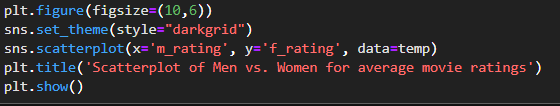
Make a scatter plot of men vs women and their mean rating for every movie:

* Create 2 dictionaries for the men and women ratings using the to\_dict() method after grouping all the ratings by the movie ids and the mean of their ratings.
* Create a new dataframe with only the unique movie\_ids from the original data set.
  + Add the attributes mean\_F\_rating and mean\_M\_rating to the dataframe with the values from our dictionaries.
    - If the movie id is not found in the dictionary, set the attribute value to NaN.
  + Find all the rows that don’t have nulls/NaN and put them in a new dataframe.
    - Utilized the isna() method to check for NaN.
* Used the matplotlib and seaborn libraries to create a scatter plot with the final data frame no\_nulls\_temp



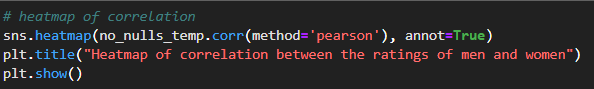
Make a scatter plot of men vs women and their mean rating for movies rated more than 200 times

* Created a series of movie ids and the number of ratings they received using pandas values\_counts().
* Get the index/movie id of all the movies that have rating counts larger than 200.
* Create a new dataframe containing only the movies that have more than 200 ratings by using isin() with the list of ids from the previous step.
* Create a new dataframe by concatenating 2 separate dataframes of the average ratings from men and women.
* Reformatted the dataframe with proper column titles and indices.
* Used the matplotlib and seaborn libraries to create a scatter plot with the final data frame.



Compute the correlation coefficient between the ratings of men and women:

* Used the corr() function with the Pearson method from pandas on the previous final data frame no\_nulls\_temp.
* Used the matplotlib and seaborn libraries to visualize the correlation coefficients in a heatmap.



Conjecture: Fantasy movies are usually received similarly by both genders so the rating to a fantasy movie given by one gender can be used to predict the other gender rating.

* Creates a dataframe for movies with the fantasy genre.
* Use seaborn and matplotlib to plot a scatterplot and heatmap of the data frame to get the correlation coefficients.

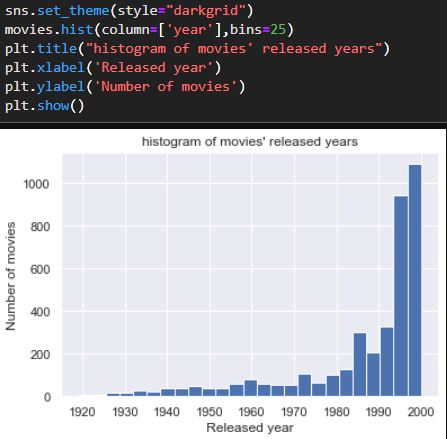
**Problem 4:**

Business question: what would a user want to watch based on other users with similar interests?

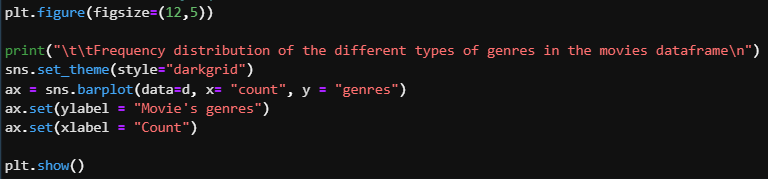
Solution: Movie Recommendation system with collaborative filtering

Data Exploration:

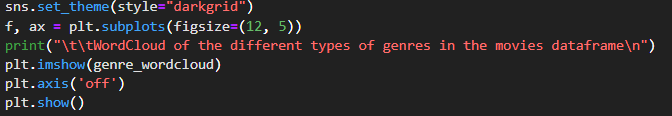
* Movie Dataframe:
  + Used the info() method on the data frame to see the total number of movies from the number of non-null values.
  + Added a column to the dataframe for the released year.
    - Get the year from the title of the movie, from the 5th to the 2nd last characters.
  + Made a histogram of the movies’ released year with the matplotlib and seaborn libraries.



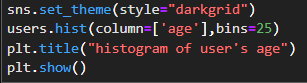
* + Remove the year from the movies’ title.
    - Set the new title to be the original title from the 1st character to the 6th last character.
  + Data visualization for the movie genres:
    - Horizontal bar chart:
      * Create a set of all the genre keywords with a for loop through the genres column and splitting the values.
      * Wrote a function that counts the number of times each of the genre keywords appear.
        + Create a dictionary for all the unique genres.
        + Go through the genres column in the MovieLens dataset and increment the count of a genre if it appears.
        + Convert the dictionary to a list to sort it by the frequency of the genres and return that list
      * Used the function above to get access to a list of genre keywords sorted by decreasing frequency.
      * Convert the list back into a dictionary.
      * Create a dataframe from the dictionary with genres and count columns
      * Used the matplotlib and seaborn libraries to create a horizontal bar graph of the frequency distribution of the different types of movie genres



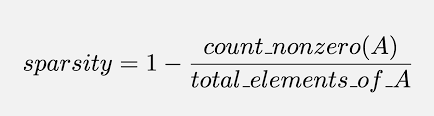
* + - Wordcloud:
      * Installed the word cloud module with “pip install wordcloud” or “conda install -c conda-forge wordcloud”.
      * Create a word cloud of the genres with the WordCloud() and generate\_from\_frequencies() methods with our dictionary of genres and frequencies.
      * Plotted out the word cloud with the seaborn and matplotlib libraries.



* Rating Dataframe:
  + Dropped the timestamp column as it is unnecessary to our analysis.
  + Used the info() method on the data frame to see the total number of ratings from the number of non-null values.
* User Dataframe:
  + Used the info() method on the data frame to see the total number of users from the number of non-null values.
  + Plotted a histogram of the users’ age with the seaborn and matplotlib libraries.



Predictive Modeling:

* Changed the format of the Rating dataframe by pivoting it to have one row per user and one column per movie with the rating that a user gave to a movie the intersection of the row and column.
  + Defaulted the rating to movies that a user hasn't rated to 0.
* De-normalize the data and convert it from a dataframe to a numpy array.
  + Normalize by getting the mean of each user ratings.
  + Minus each of the users’ ratings by their normalization.
* Check the sparsity of the ratings dataset.
  + Get the sparsity level with the formula
  + Here, the non-zero count is the length of the ratings dataframe and the total element value is the length of the users dataframe multiplied by the length of the movies dataframe.
* Set up SVD
  + Used the Scipy function svds because it allows us to choose how many latent factors we want to use to approximate the original ratings matrix (instead of having to truncate it after).
  + Passed in the de-normalized rating dataset from earlier with a k value of 50.
* As we are going to leverage matrix multiplication to get predictions, we convert sigma to the diagonal matrix form with the numpy diag() method.
* Add the users mean rating back to get the actual star ratings prediction.
  + Create a prediction matrix for every user
* Making predictions from the decomposed matrices. Write a function to recommend movies for any user.
  + Takes in the prediction matrix, the user ID, the original movie and rating dataframes, and the number of recommendations to give.
  + Sort the user’s predictions in descending order.
    - Used pandas iloc[] on the prediction matrix with the userID - 1 since the user ID in the prediction matrix starts at 0 instead of 1.
  + Get the user’s data and merge it with the movie dataframe.
  + Print out the number of movies the user has already rated and how many movies the user would like to be recommended.
  + Create a dataframe of the recommended movies:
    - Get the movies the user hasn’t rated by using isin() on the movie ids from the movie dataframe with the merged dataframe from the user’s data earlier.
    - Merge that with the user’s sorted predictions.
    - Renamed the columns appropriately.
    - Sort the data by the prediction values in descending order.
    - Restrict the data frame to the top n movies with n being the number of recommendations passed into the function.
  + Return the movies the user has already rated and the movies with the highest predicted ratings that the specified user hasn't already rated.
* Test the recommendation system with user 6038.

Model Evaluation

* Used the Surprise library that provides various ready-to-use powerful prediction algorithms including SVD to evaluate its RMSE on the MovieLens dataset
  + Load the Reader Library
  + Load the ratings dataframe with the Dataset library
  + Load the SVD library
  + Split the dataset for 5-fold evaluation of the RMSE with cross\_validate from surprise.model\_selection.
* Build a train set with build\_full\_trainset() then trains it on SVD with fit().
* Used SVD to predict the ratings that user 6038 will give to the movies on our recommended list.

# Results

**Problem 1:**

21 movies have an average rating over 4.5 overall.

23 movies have an average rating over 4.5 overall among men.

51 movies have an average rating over 4.5 overall among women.

86 movies have a median rating over 4.5 among men over age 30.

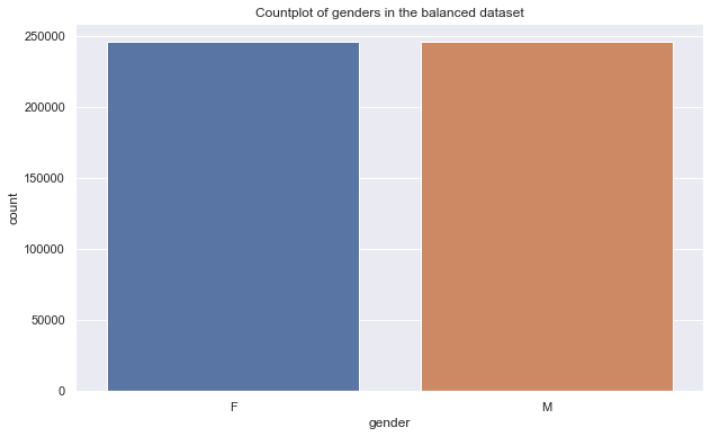
149 movies have a median rating over 4.5 among women over age 30.

Top 10 most popular movies:



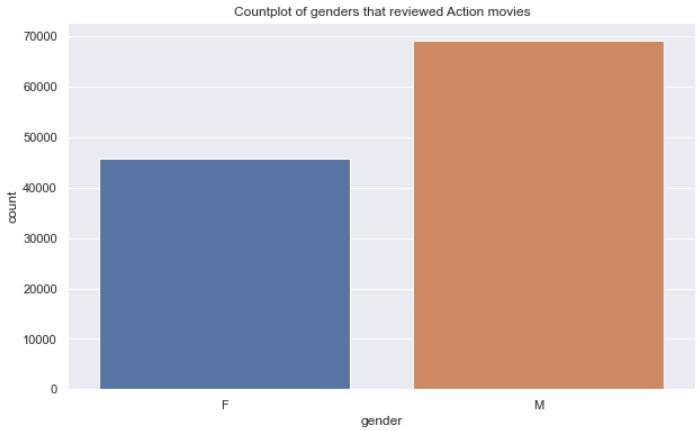
Seeing that all the most popular movies in the dataset are old classics from the 90s, we can infer that the dataset is relatively old. The data was most likely collected around the early 2000s. We can also see how popular the Action Sci-fi genres were with more than half of the top movies being of that genre (Star Wars, Jurassic Park, Terminator, The Matrix, Back to the Future).

Balanced bar graph of ratings made by men and women:



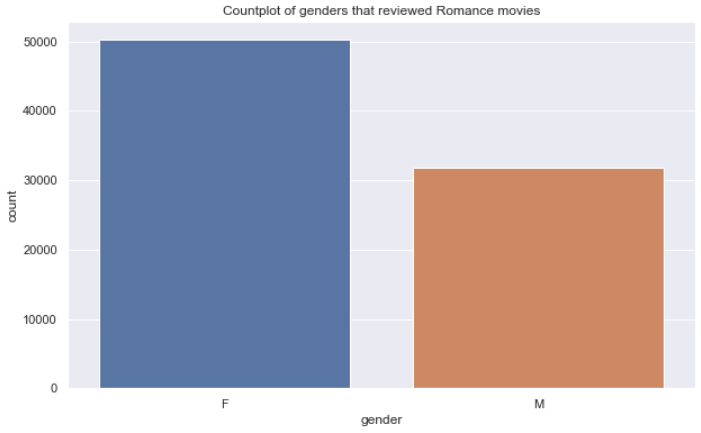
By using this balanced dataset to see which gender watches a certain genre more, it’ll give a more fair and trustworthy result.

Bar graph of the number of men and women that reviewed Action Movies:



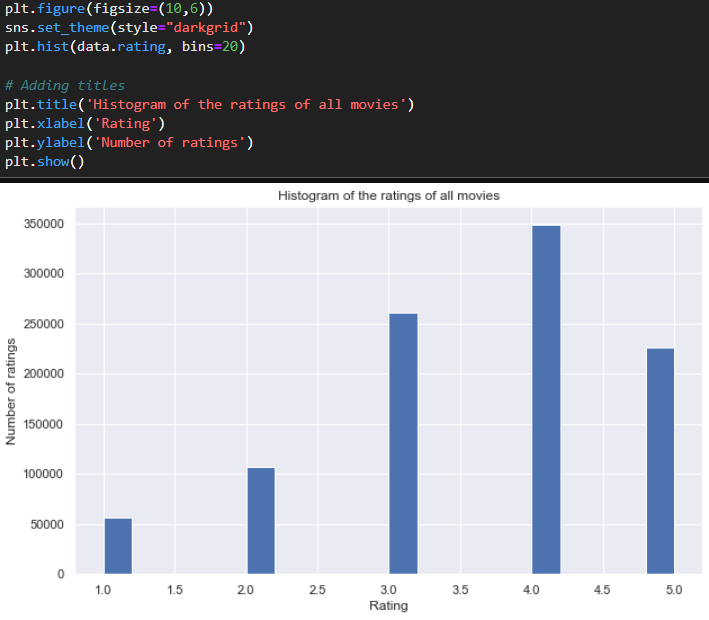
This confirms our conjecture that men watch action movies more than women.

Bar graph of the number of men and women that reviewed Romance Movies:



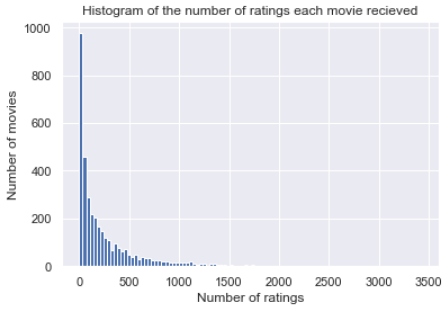
This confirms our conjecture that women watch romance movies more than men.

Histogram of the ratings of all movies:



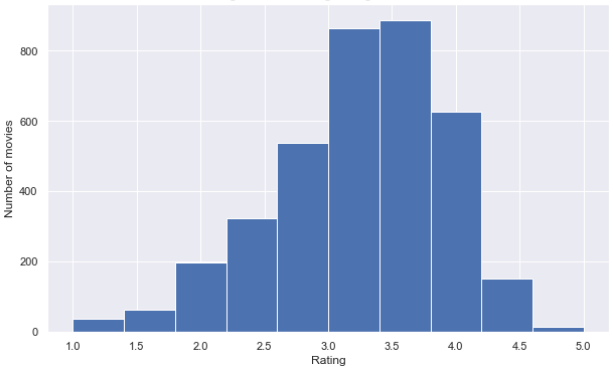
From this histogram, we can see that most people tend to give positive reviews with the majority of them being 4 stars. This makes sense as some people can be more reserved in giving 5 stars and saving it for the “perfect” movies. The majority of positive reviews also make sense as usually, people would only go to watch a movie if they’re interested in the movie and would be more likely to enjoy it.

Histogram of the number of ratings each movie received:



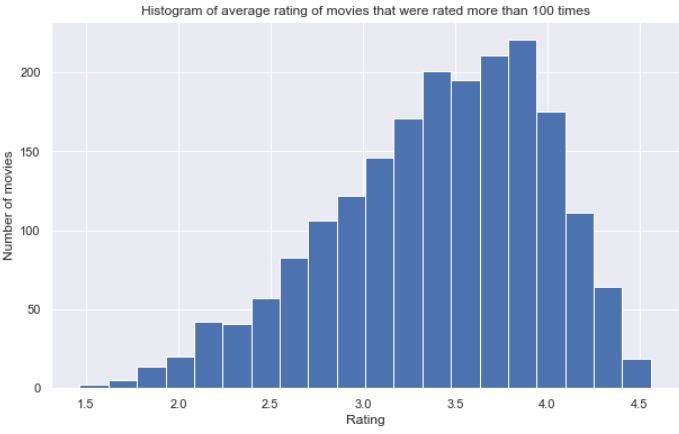
We can see that most movies only received a small amount of ratings so we can infer that there’s more lesser known movies than popular movies. This makes sense as the number of indie movies that come out can easily outnumber the annual summer blockbuster movies.

Histogram of the average rating for each movie:



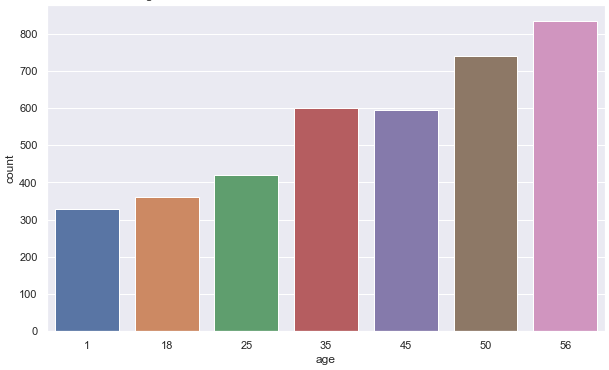
The histogram here can be seen to more left skewed like the histogram of the raw round ratings. With the average ratings, we have a better sense of how the ratings are actually distributed. Here we can see that most movies actually have an average reviews of around 3.5 instead of the 4 that we’d presume from the raw rating histogram. We can also see the incredibly small amount of movies that have an average rating of 1 or 5 as people are less likely to rate closer to the extremities.

Histogram of the average rating of movies that were rated more than 100 times:



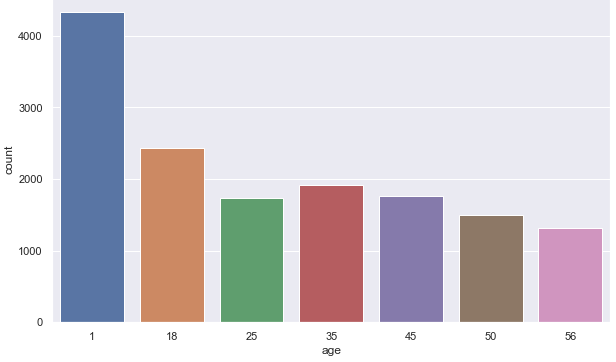
Here, we get an even better sense of how the ratings are distributed with more bars. Notice also how the tail ends of the histogram don’t include 1 and 5. This is because it’s near impossible for a movie to have an average rating of 1 or 5 unless there’s only one or a few ratings of 1 or 5, which is impossible when the histogram only includes the movies that were rated more than 100 times. More than 100 people can’t all share the exact same views so they can’t give the exact same ratings. It’d therefore make more sense to trust movies that were reviewed more than 100 times as we get a more accurate sense of the quality of the movie from the majority opinions.

Age distribution for reviews of Film-Noir movies from balanced dataset:



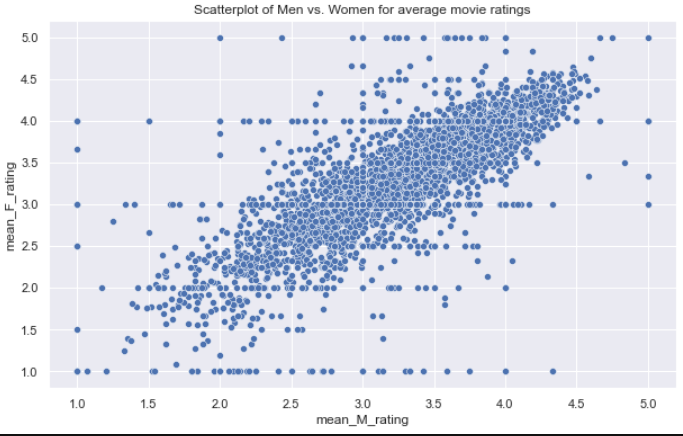
Here we can see that most of the ratings were from the age bracket 56 which confirms our hypothesis that people older than 56 years old are more into film-noir movies.

Age distribution for reviews of Children's movies from balanced dataset:

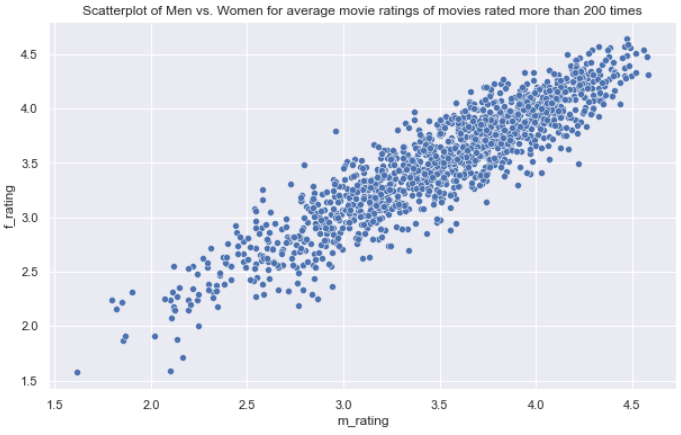


Similarly, here we can see that most of the ratings were from the age bracket 1 which confirms our hypothesis that people younger than 18 years old are more into Children movies.

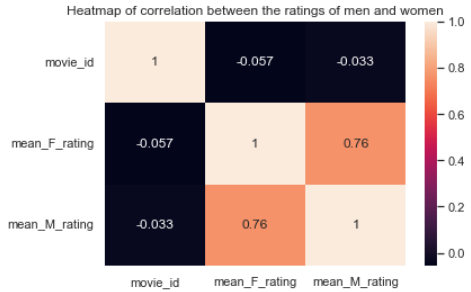
Scatterplot of Men vs Women for average movie ratings:



Scatterplot of Men vs Women for average movie ratings of movies rated more than 200 times:

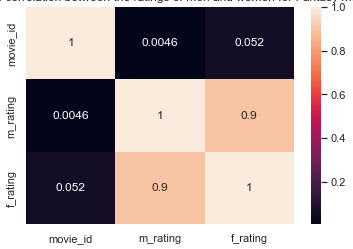


Heatmap of Correlation between the ratings of men and women:



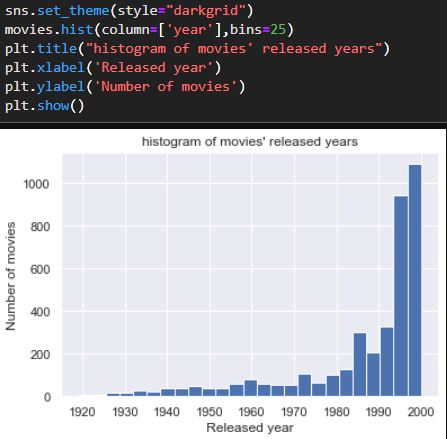
The correlation coefficients here can be seen to be 0.76 which shows a good positive correlation. This shows that there’s some good correlation between the ratings made by men and women

Heatmap of correlation between the ratings of men and women for Fantasy movies:



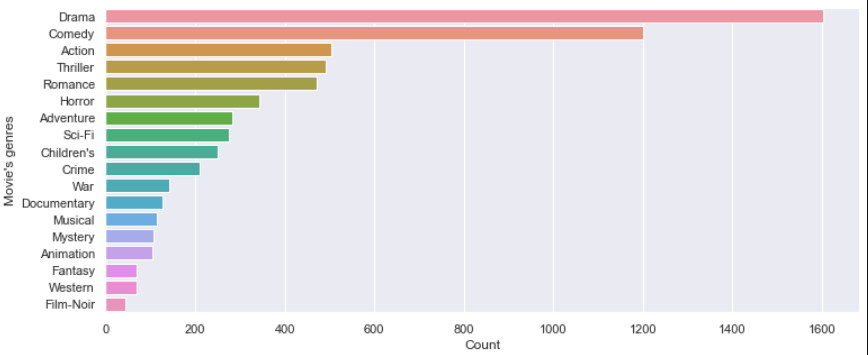
The correlation coefficient here is higher, 0.9, which shows some pretty good correlation so we can reliably use the ratings of a Fantasy movie made by one gender to predict the rating of the other gender.

Histogram of the movies’ released years:



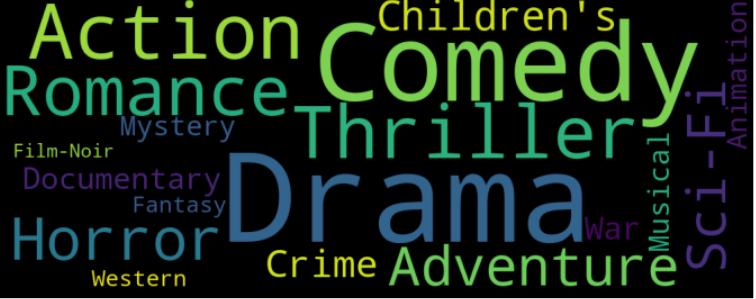
This confirms our further analysis of the most popular movies that most of the movies in the dataset were released in the 90s.

Frequency distribution of the different movie genres:



From here, we can see that most movies have the genres drama and comedy as people are more entertained by them. The least popular genre is film-noir as it’s a more niche and serious type of movie that only appeals to a small number of people.

WordCloud of the different movie genres:



Top 20 movies that user 6083 has already rated:



Top 20 movies that user 6083 will most likely enjoy:



The list of recommendations looks reasonable. User 6083 rated a Star Wars movie well (4) so the system recommends some other Star Wars movies. Although we didn’t take in genre specifically as a factor in the collaborative filtering system, the system still successfully recommends mostly comedy movies which appear in most of the movies that use 6083 rated 4 and 5.

Estimated rating of the recommended movies:



After running the data evaluation with surprise SVD, we can see that most of the movies in the recommended list have an estimated rating (est) of 3.2 or higher with the average to be 3.9. We can presume that user 6083 will enjoy these movies with these reasonably good ratings.

# 

# Conclusion

The most popular movie is American Beauty, which has 3428 counts for this movie. The Star Wars Series were also popular in the period of 1995 to 2000. We got three Star Wars movies and a total of about 6000 counts. We can see that men like Action movies more and Women like Romance movies more. all of the rating distributions fit a left skewed distribution which means the rating times do not have a significant effect on the rating distribution. By doing the histogram and conjecture of the age ranges, we can conclude that people in their age range like the genre movies made for them more. For example, Children were interested in children’s movies, and people older than 56 were interested in Film-Noir movies more. Since the correlation is about 0.763190 between male and female, which is larger than 0.75, we can say that it is a strong positive correlation between the rating of man and woman. The two most popular movie genres are Drama and Comedy. This may be because the special effects were not mature enough. Although people love Sci-fi movies better as we can see in the top 10 popular movies, The most movies made were still Drama and Comedy which do not need too much special effects. For our business question, based on our predicted model, the most possible predicted model of rating is 3.9. Thus, we can conclude that people would be interested in the movies based on the other users with similar interest.