Personalized Movie Recommendation Machine

Sara Ma(ym2841), Hanrui Yu(hy2716), Qiming Feng (qf2155),
Ting Lei (tl3101), Linzi Guan (lg3183), Xuanyu Li (xl3116)

Introduction

After browsing through some interesting datasets on Kaggle, we decided to make a personalized movie recommendation for both movie lovers and movie makers using five different movie datasets. We would merge the datasets in one database, perform exploratory data analysis, visualize the historical patterns and trends, run machine learning models for predictive analysis and develop two recommendation machines.

Data Cleaning and Preprocessing

We decided to use five different datasets for our project. IMDB, Netflix, Amazon Prime, Hulu and Disney (Appendix: Datasets). IMDb is one of the largest online databases for movies, television and video games. Our project used weighted_average_vote, genre, country, director, actors, reviews_from_users, and reviews_from_critics from this IMDb dataset. Our movie list came from Netflix, Amazon Prime, Hulu and Disney, the four most popular media and video streaming platforms. Our project used title, cast, listed_in (cinematic genre), and description from these four datasets. We first did some data cleaning and removed some abnormal data, then we merged the 4 platform data on movie titles to form a generalized platform dataset and then merged the platform data to the IMDB data on movie titles.

I. Exploratory Data Analysis

Our EDA and visualization could be divided into three main sectors. The first was a general description of the dataset. Our second sector was doing movie analysis from the audience side based on the movie rating rating, which interpreted different types favored by the public and different population groups. Lastly, we had the director sector focusing on analyzing rating by genre, top actors in each genre and showing highest rated directors, and their rating by age group.

Analyzing our dataset generally, the pie chart (Appendix: Figure 1) illustrated percentages of titles that were either movies or TV shows in our dataset. There were mostly 90% data were movies, and the remaining 10% were TV shows. Appendix: Figure 2 demonstrated our movie rating analysis. The largest count of movies was made with the 'TV-MA' rating which was TV Mature Audience Only. The second largest count was the very popular 'R' rating. An R-rated film may include some elements that parents are counseled to take this advisory rating very seriously. The third largest was the 'TV-14' which parents strongly cautioned.

Appendix: Figure 3 of top 10 movie content creating countries showed that the movies from the United States and United Kingdom were taken in most parts. Besides, we found that 2017 and 2018 were the years when most of the movies were released referring to the year wise analysis of Appendix: Figure 4.

WordCloud Analysis and Top Ten Movies Lists: To better understand the audience's formation and preference, we incorporated the WordCloud analysis on gender and four age groups. Then, we identified ten favorite movies for each segment (Appendix: Figure 8&9). We observed that the male population favors movies with contents of women and family, while the female population has a higher tendency to watch film genres related to love and son. Our analysis revealed an interesting pattern of age groups that younger people prefer themes like gender, money, etc. As people grow up, some of their attention gradually shifts to the topics such as life, love, journey, and others. Exploring the audience's age distribution and taste helped us comprehend their potential actions on movie selection and set the basis for the recommendation system.

Genre-Rating Analysis: To view the distribution of movies' average rating, we divided the average rating into nine ranges from 1 to 10, and each range is 1.0 in length (1-2, 2-3, …, 9-10) (Appendix: Figure 10). Most movies are rated in the middle section. The median and mean are in ranges 5-6 and 6-7, with standard deviations of 0.275 and 0.282. In the next step, we identified all 21 genres and counted the number of movies in each genre (Appendix: Figure 11). Films that are rated over seven are considered high quality. After combining average rating ranges with movie genres, we found that the top five genres have relatively stable distribution in terms of movie quantities for high-quality movies (Appendix: Figure 12). However, only drama can reach the 9-10 range, proving that the drama type can be a worthy investment with high potential.

Recommend top rated actors/actresses to movie makers based on identified genre: Actors with 'avg_vote' greater than 9 are 'Great Actors' for movie directors to consider. Following counting the most popular genres, we chose the top 8 ones and listed the corresponding top rated 'Great Actors' sorted them by their 'ave_rating_actor' in descending order and displayed the top 15. For those genres that did not have 15 'great actors', we displayed as many as possible. The 8 graphs were illustrated in the Appendix: Figure 5.

Age Group preferences on directors: We also calculated the top 9 highest rated directors and displayed their ratings by age groups. It was interesting but not surprising to see that different age groups have somewhat different preferences/tastes on directors.The table was shown in the appendix section as Appendix: Figure 6.

II. Movie Recommender

Since our dataset mainly consists of descriptive information, such as genre, director and duration, for movies, and it is difficult to gather data about other users, we have decided to use a content-based filtering model to build our recommendation machine. More specifically, we would recommend movies similar to what the user likes based on certain movie features.

Review and Preprocess the Dataset: There are two criterias in selecting features to be used in computing similarity scores: data relevance and data completeness. Data relevance demonstrates how relevant the feature is in the user's decision making process, or user's preference for a movie. For example, features, such as movies' gross income, budget, year, language and country, may not play a large role in determining whether a person would enjoy a certain movie. Data completeness refers to the wholeness of data - the data should not have missing values or gaps. If one type of feature value is missing for most of our movies in the dataset, then it would be difficult for the recommendation machine to accurately predict how well the input movie matches with the existing movies. For example, there are about 50.1% of missing data in average vote for certain age group (groupsallgenders_0age_avg_vote, males_0age_avg_vote, females_0age_avg_vote) because there are certain types of movies are not suitable for children or teenagers to watch. Finally, we chose genre, director, description, writer, and actors as our features. Moreover, we have performed necessary data processing to convert the raw data format into the desired input for vectorization.

Convert Text Columns into Numerical Values: All five features are categorical data, and to effectively compare each movie, we would need to convert them to vectorized representation of words in these text columns. However, we encountered some difficulties in constructing the vectorizer matrix. Although Term-Frequency-Inverse Document Frequency (TF-IDF) vectorizer works great for the description feature, it does not work with other features such as genre, director, or actors since it penalizes a word for occurring frequently in different movies. On the other hand, CountVectorizer is suitable for the other four features (actors, director, genre, and writer) in its sole frequency representation. Therefore, we have decided to use CountVectorizer for all five

features in our main model, and built out another model using TF-IDF for only description. In the process, we also removed the stopwords and replaced NaN with empty strings.

Measure Similarity between Movies: We have chosen cosine similarity as our similarity metric to compute the similarity between movies. Using the cosine_similarity function in the sklearn package, we were able to construct the similarity score matrix using the vectorized objects.

Define Recommendation Function: Our recommendation function contains two inputs - a movie of user's choice and a default similarity score matrix. Inside the function, it gets the pairwise similarity score of that movie input with all movies in our database, ranks all movies based on the similarity score, records the indices of top 20 most similar movies, re-rank those movies based on their popularity (avg_vote), and return the top 10 most popular, yet similar movies to the user.

III. Rating Prediction System

The rating prediction system aims at helping movie makers identify whether they are able to achieve a higher than average rating or not with the production factors they intend to have including duration, genre, director, actor and production company of the movie.

Variables Identification: To classify whether the movie would have a higher than average rating or not, the dependent variable was set to be a dummy variable "high_vote" (1 with predicted rating higher than the average and 0 otherwise). Indicated by the Exploratory Data Analysis part, genre and duration(Appendix: figure 7) tend to explain some variations in movie ratings, so they were taken as the explanatory variables. Moreover, directors, production companies and actors are most commonly considered as important production factors by movie makers and movies with those famous and experienced production factors are more likely to be successful, so ranks of the reputation of those production factors were also set as the explanatory variables.

Data preparation and preprocessing: As the current dataset contains information for both movies and TV shows and attention was only paid to movies in the rating prediction system, the dataset was filtered first to include only movie data. From the EDA on duration and movie ratings(Appendix: figure 7), differences in movie ratings can be found in different duration ranges and boundaries can be approximated as 95 minutes and 120 minutes. Dummy codings were applied on genres and duration ranges. Since reputation matters and one of the most representative features of a movie is worldwide box office, the ranks of worldwide box office of directors, actors, and production companies were determined to approximate the ranks of reputation. To get the data of worldwide box office, a web scraping of the production factors and their corresponding worldwide box office was conducted from THE NUMBERS website(Appendix: Datasets). Rankings of worldwide box office were assigned to each production factor and segmented by quartiles, and missing values with unapproachable worldwide box office were dropped from the dataset. Dummy coding on quartiles of rankings was applied.

Machine learning model specification: With all the features believed to explain the variations in movie rating set as dummies, a baseline was set to avoid perfect collinearity(Appendix: Figure 13). The machine learning model was set up with those features as explanatory variables, whether the movie can achieve a higher than average or not as dependent variable and the train test split was set as 7:3. Regression models including basic linear regression and linear regressions with Lasso and Ridge to avoid overfitting and classification models including logistic Regression, decision tree and random forest were conducted to find a most desirable model.

Regression models:

Basic linear regression: Basic linear regression uses Ordinary Least Squares method without any penalty on weights of features. As linear regression returns a numeric value instead of a classification, a threshold was set to get classification from the result of the regression that the movie will be identified as not reaching the average score if the prediction from regression cannot reach the threshold. To determine the threshold, the most important factor considered is the precision score. In a recommendation system, precision is essential as it would be harmful for a movie maker to get overconfident by the prediction and invest in large sums of money but turn out to be disappointed by the actual rating. Hence, precision score was prioritized in our model and the threshold was chosen to be 0.7. The basic linear model had test scores with Accuracy: 0.59, Precision: 0.78, Recall: 0.23 and an AUC of 0.58 (Appendix: Figure 14).
Linear regressions with Lasso: Lasso regression uses a hyperparameter alpha to penalize the sum of absolute values of weights and would shrink some coefficients to 0 if they have low explanatory power. In our dataset, all coefficients were shrunk to 0 when applying Lasso regression, so it is not appropriate to use Lasso for prediction here and this also suggests that no linear combination of the explanatory variables might be useful for predicting movie ratings with current features selected, so a non-linear model should be considered further.
Linear regressions with Ridge: Ridge regression also uses a hyperparameter alpha, but to penalize the sum of squared values of weights and alpha was selected to be 1 to get the lowest error. The ridge regression model has test scores with Accuracy: 0.61, Precision: 0.76, Recall: 0.25 and an AUC of 0.59 (Appendix: Figure 15).

Classification models:
Logistic regression: Logistic regression returns a binary outcome for classification and the model has test scores with Accuracy: 0.66, Precision: 0.66 and Recall: 0.66 and an AUC of 0.66 (Appendix: Figure 16).
Decision tree: Decision tree does the classification by learning decision rules from data features (Appendix: Figure 17) with test scores: Accuracy: 0.63, Precision: 0.58, Recall: 0.85, AUC: 0.63 (Appendix: Figure 18).
Random forest: Random forest takes multiple decision trees into consideration and our random forest model returned the majority of classifications out of 1000 trees with test scores: Accuracy: 0.66, Precision: 0.63, Recall: 0.72 and an AUC of 0.66 (Appendix: Figure 19). Feature importance was also conducted on the random forest model(Appendix: Figure 20) and duration range appears to be the most important feature.

Cross-validation: We trained our model using the subset of the dataset and then evaluated using the complementary subset of the dataset. For decision tree and random forest, we splitted the datasets into five mutually exclusive subsets. Each time, four of them were selected as trained datasets to fit a model, which then used to predict the target variable by using the data in the fifth subset. The displayed matrics (Accuracy, Precision, Recall) were the averaged results.

Model comparison and selection: By comparing the scores of models and balancing our objective for a high level of precision(Appendix: Figure 21), we determined to use the logistic model for final prediction. And if the director, actor and production company input is not in the current list, an average rank will be given.

IV. Limitations and Future Improvements

We developed two recommendation machines for movie lovers and makers (Appendix: Figure 22): For the recommendation machine to movie lovers, we would design a way to measure accuracy of the machine and incorporate more features, and use more complex vectorization methods in the recommendation. And For the recommendation machine to movie makers, regarding actors in the movie, the model now only considers whether there is a “trending” star in the displayed list, but not whether the actor is the main cast or not nor the “adding” effect of the number of “trending” stars in one movie. For future improvements, we would try to use a score system to determine the actor score of a movie by taking those factors into account.

Appendix

Datasets:

Original datasets used:

IMDb Dataset

1. Stefano Leone, (2019, November). IMDb movies extensive dataset, Version 2. Retrieved October 21, 2021 from https://www.kaggle.com/stefanoleone992/imdb-extensive-dataset

Netflix, Amazon Prime, Hulu and Disney Datasets

2. Shivam Bansal, (2021, October). Amazon Prime Movies and TV Shows, Version 1. Retrieved October 21, 2021 from https://www.kaggle.com/shivamb/amazon-prime-movies-and-tv-shows
3. Shivam Bansal, (2021, October). Disney+ Movies and TV Shows, Version 1. Retrieved October 21, 2021 from https://www.kaggle.com/shivamb/disney-movies-and-tv-shows
4. Shivam Bansal, (2021, October). Hulu Movies and TV Shows, Version 1. Retrieved October 21, 2021 from https://www.kaggle.com/shivamb/hulu-movies-and-tv-shows
5. Shivam Bansal, (2019, December). Netflix Movies and TV Shows, Version 5. Retrieved October 21, 2021 from https://www.kaggle.com/shivamb/netflix-shows

Websites conducted web scraping:

6. *Movie production companies - box office history*. The Numbers. (n.d.). Retrieved December 12, 2021, from https://www.the-numbers.com/movies/production-companies/#production_companies_overview=od1.
7. *Top 100 stars in leading roles at the worldwide box office*. The Numbers. (n.d.). Retrieved December 12, 2021, from https://www.the-numbers.com/box-office-star-records/worldwide/lifetime-acting/top-grossing-leading-stars.
8. *Top grossing director at the worldwide box office*. The Numbers. (n.d.). Retrieved December 12, 2021, from https://www.the-numbers.com/box-office-star-records/worldwide/lifetime-specific-technical-role/director.

Figures:

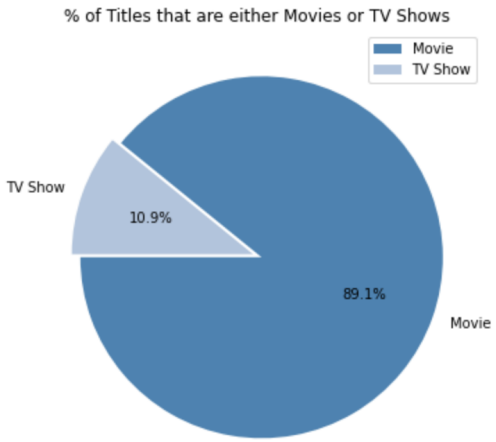


Figure 1. Percentage of Titles That Are Either Movie or TV Shows

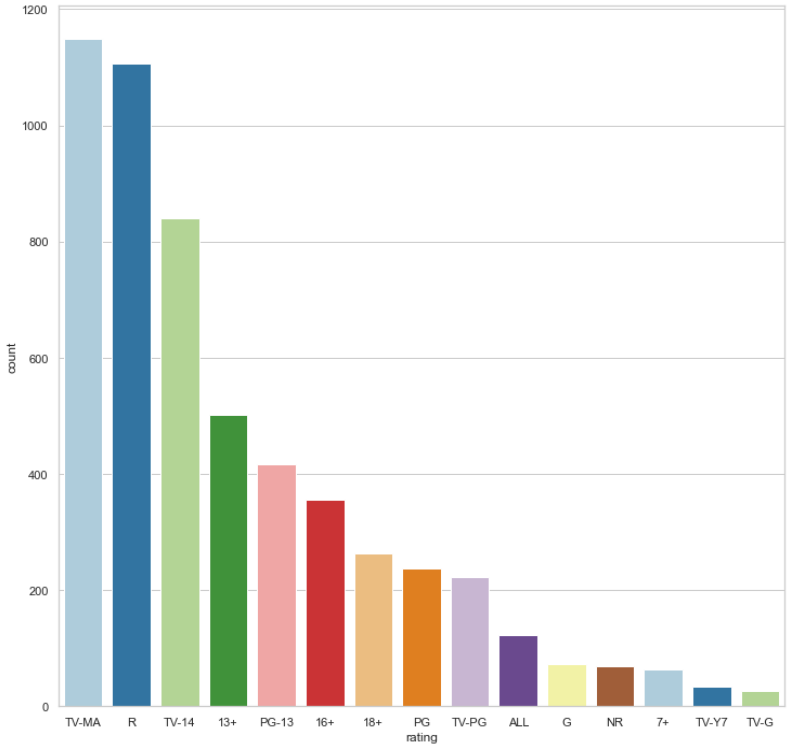


Figure 2. Movie Ratings Analysis - Which Movie Rating Counts The Most in Our Dataset

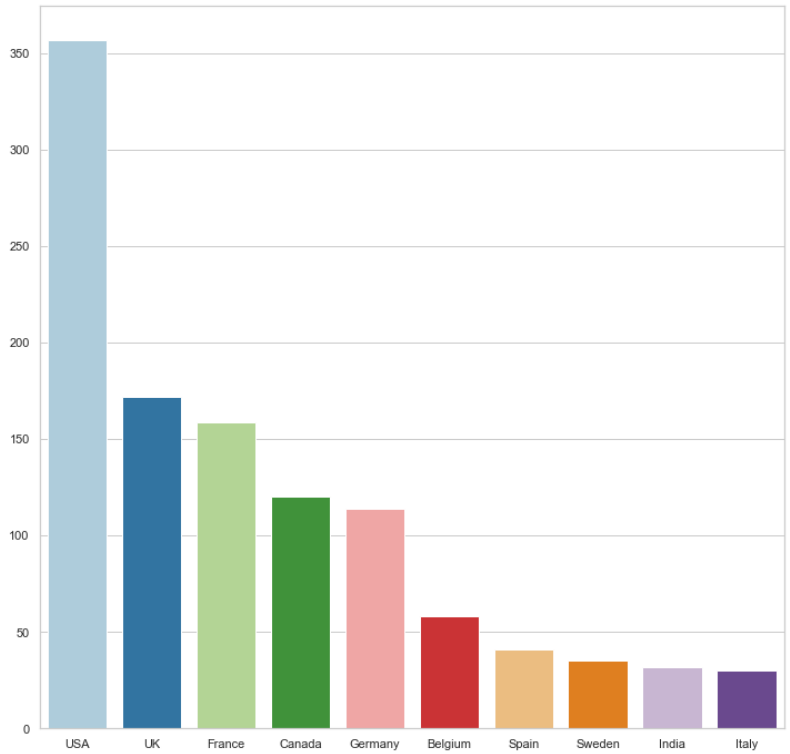


Figure 3. Top 10 Movie Content Creating Countries

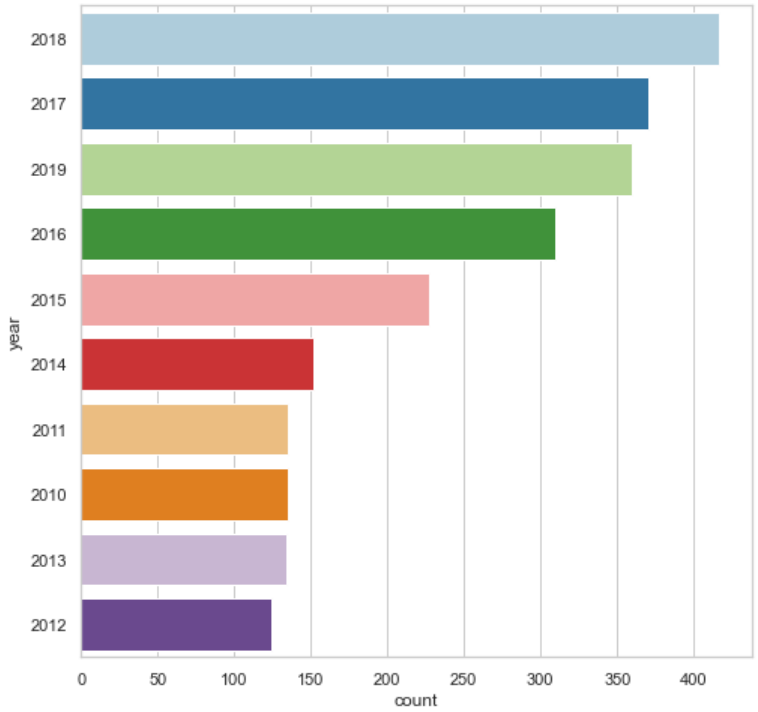
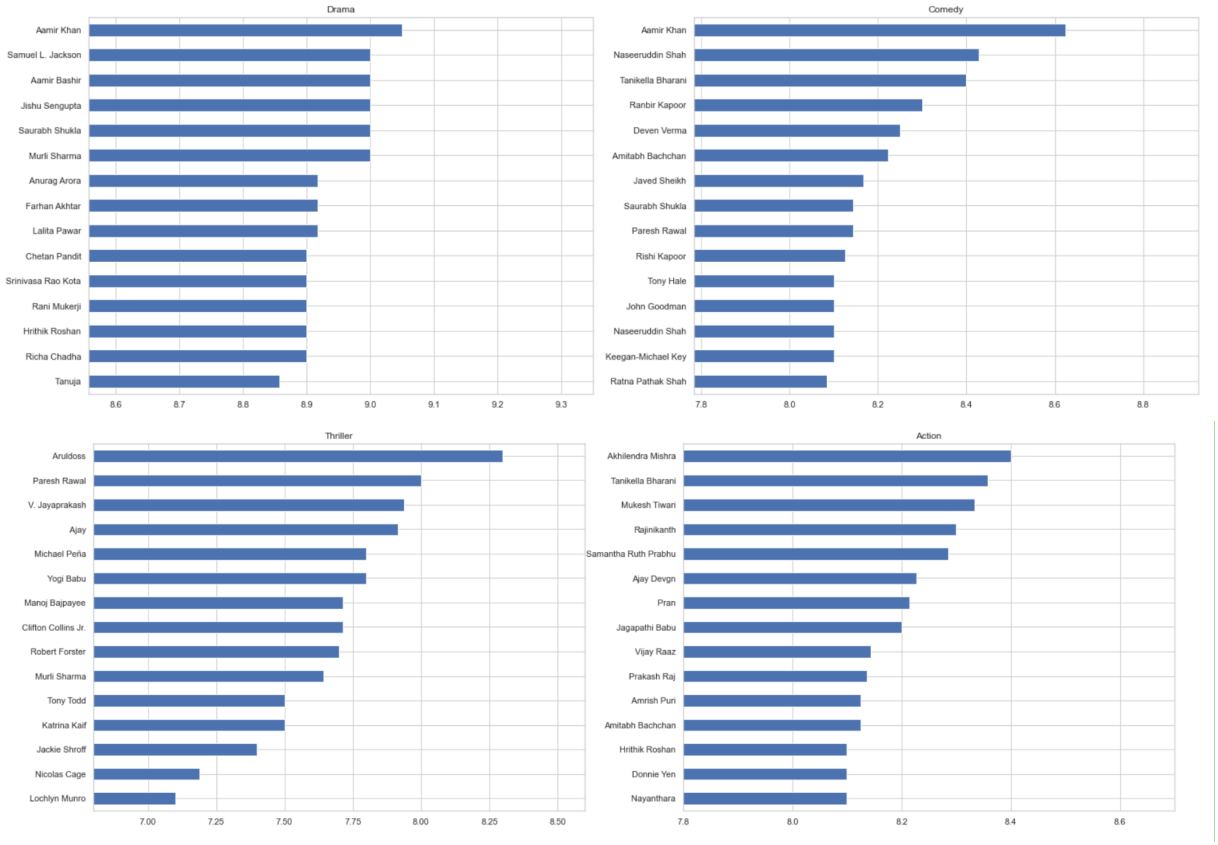


Figure 4. Year Wise Analysis



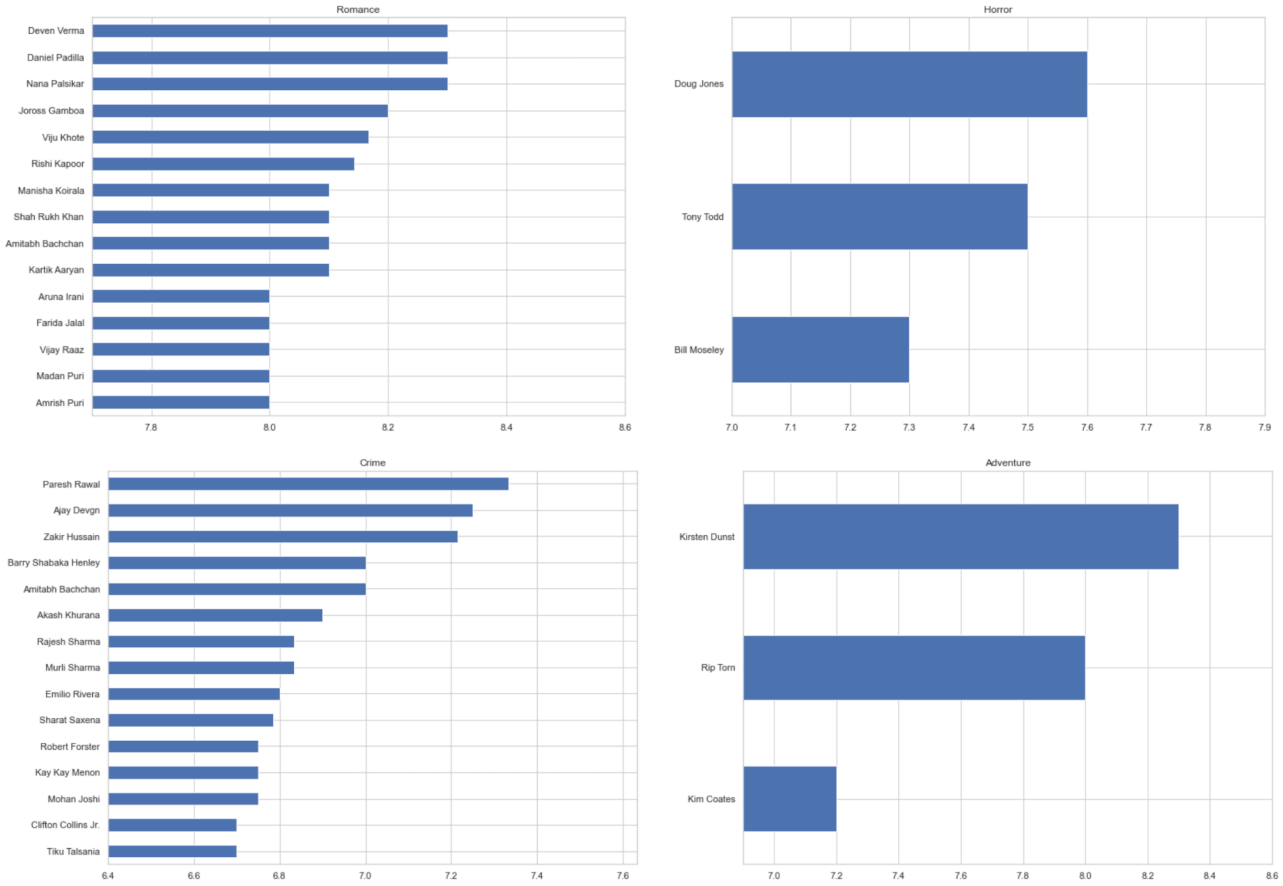


Figure 5. Genre Wise Analysis - 8 graphs above

	director	avg_vote	<18_avg_vote	18_30_avg_vote	30_45_avg_vote	>45_avg_vote
0	Balavalli Darshith Bhat	9.2	NaN	6.3	4.0	1.0
1	Raghav Peri	9.1	NaN	6.3	5.5	6.7
2	Antoneta Kastrati	9.1	7.4	9.8	7.0	6.9
3	Christopher Nolan	8.8	9.0	9.0	8.7	8.1
4	Jonathan Butterell	8.7	NaN	8.5	8.9	8.2
5	K. Viswanath	8.7	9.5	8.7	8.8	5.7
6	Harjit Singh	8.6	NaN	8.7	8.3	8.6
7	Gulzar	8.6	NaN	8.6	8.7	7.1
8	Abhijeet Shirish Deshpande	8.6	NaN	8.6	8.5	5.6
9	Parthiban	8.6	9.6	8.7	8.2	6.9

Figure 6. Rating By Age Group

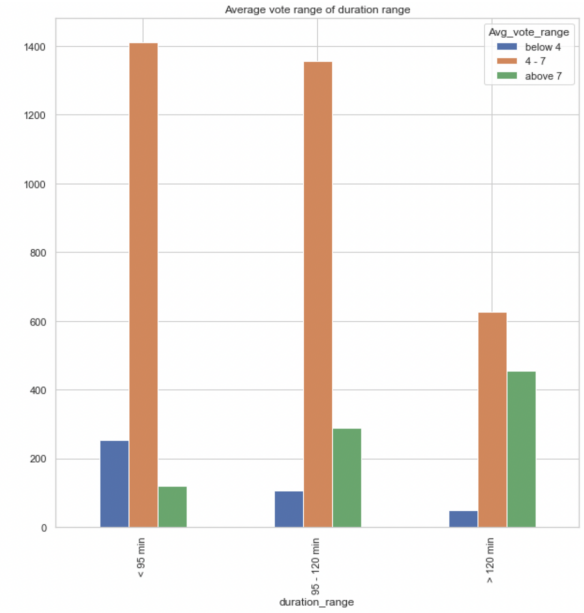


Figure 7. Rating Analysis by duration range

Female:

	title	females_allages_avg_vote
0	2 States	10.0
1	Boyz 2	10.0
2	Takatak	10.0
3	Take Care Good Night	9.9
4	3rd Class	9.7
5	Manusangada	9.7
6	Judge Singh LLB	9.5
7	Tikli and Laxmi Bomb	9.5
8	The Far Frontier	9.3
9	Maha Maha	9.1

Male:

	title	males_allages_avg_vote
0	Zana	9.3
1	Pulp Fiction	8.9
2	Schindler's List	8.9
3	Everybody's Talking About Jamie	8.8
4	Inception	8.8
5	Sankarabharanam	8.7
6	Seven	8.6
7	Ani... Dr. Kashinath Ghanekar	8.6
8	City of God	8.6
9	Vikram Vedha	8.6

Figure 8. Word Cloud Analysis by gender

Below 18:

	title	allgenders_0age_avg_vote
0	Karnan	10.0
1	Robert	10.0
2	Görümce	10.0
3	Elephant Song	10.0
4	Withdrawn	10.0
5	Boyz 2	10.0
6	Anjaan	10.0
7	Magi	10.0
8	Teenkahon	10.0
9	The Gospel of John	10.0

18 to 30:

	title	allgenders_18age_avg_vote
0	Bloodline	10.0
1	Skin Deep	10.0
2	Free Ride	10.0
3	Serena	10.0
4	One More Saturday Night	10.0
5	Desperado	10.0
6	The Boys	10.0
7	Modern Love	10.0
8	Home and Away	10.0
9	The Stand-In	10.0

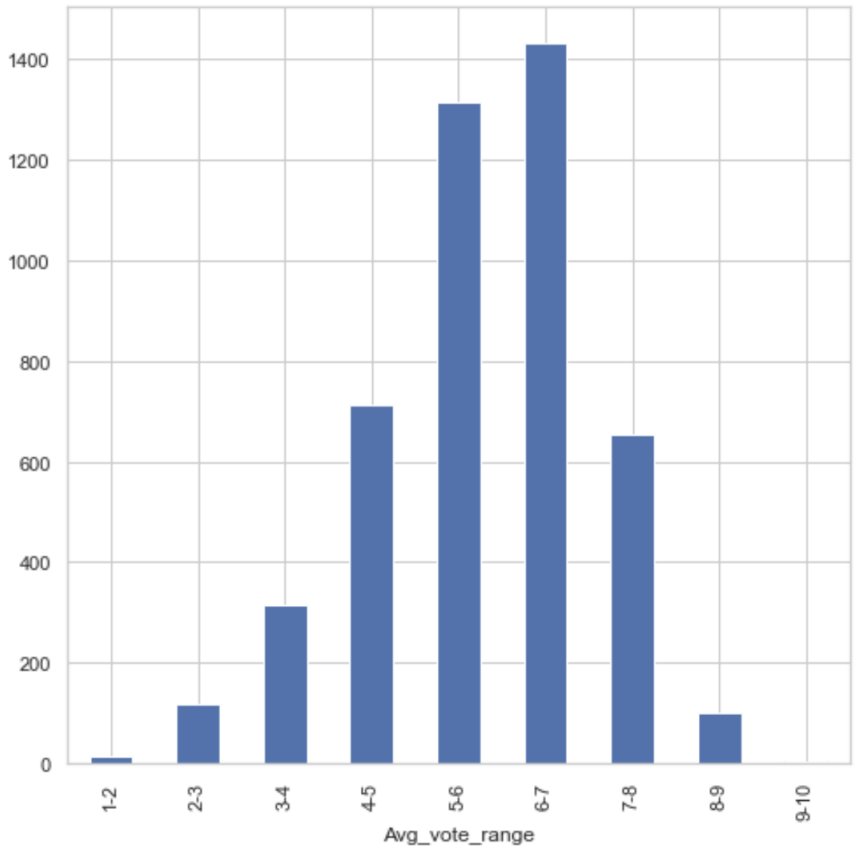
30 to 45:

	title	allgenders_30age_avg_vote
0	Hello Memsaheb	9.2
1	3rd Class	9.2
2	Copper Bill	9.0
3	Pulp Fiction	8.9
4	Everybody's Talking About Jamie	8.9
5	Schindler's List	8.9
6	Almost Human	8.9
7	Sankarabharanam	8.8
8	I Love Lucy	8.7
9	Gol Maal	8.7

Above 45:

	title	allgenders_45age_avg_vote
0	Iddari Lokam Okate	10.0
1	2 States	10.0
2	Bhaskar Oru Rascal	10.0
3	Master	9.9
4	Teen Aur Aadha	9.0
5	Pulp Fiction	8.7
6	Schindler's List	8.7
7	Eh Janam Tumhare Lekhe	8.6
8	Jaanu	8.5
9	The Gospel of Matthew	8.5

Figure 9. Word Cloud Analysis by age group



Avg_vote_range	count	mean	std	min	25%	50%	75%	max
1-2	27.0	1.855556	0.169464	1.5	1.75	1.9	2.000	2.0
2-3	238.0	2.660924	0.259394	2.1	2.50	2.7	2.900	3.0
3-4	633.0	3.621643	0.264017	3.1	3.40	3.6	3.800	4.0
4-5	1537.0	4.571893	0.279934	4.1	4.30	4.6	4.800	5.0
5-6	2957.0	5.581163	0.275365	5.1	5.40	5.6	5.800	6.0
6-7	3190.0	6.509436	0.282291	6.1	6.30	6.5	6.800	7.0
7-8	1557.0	7.432498	0.267781	7.1	7.20	7.4	7.600	8.0
8-9	216.0	8.298148	0.211959	8.1	8.10	8.2	8.400	8.9
9-10	4.0	9.125000	0.050000	9.1	9.10	9.1	9.125	9.2

Figure 10. Average vote range analysis

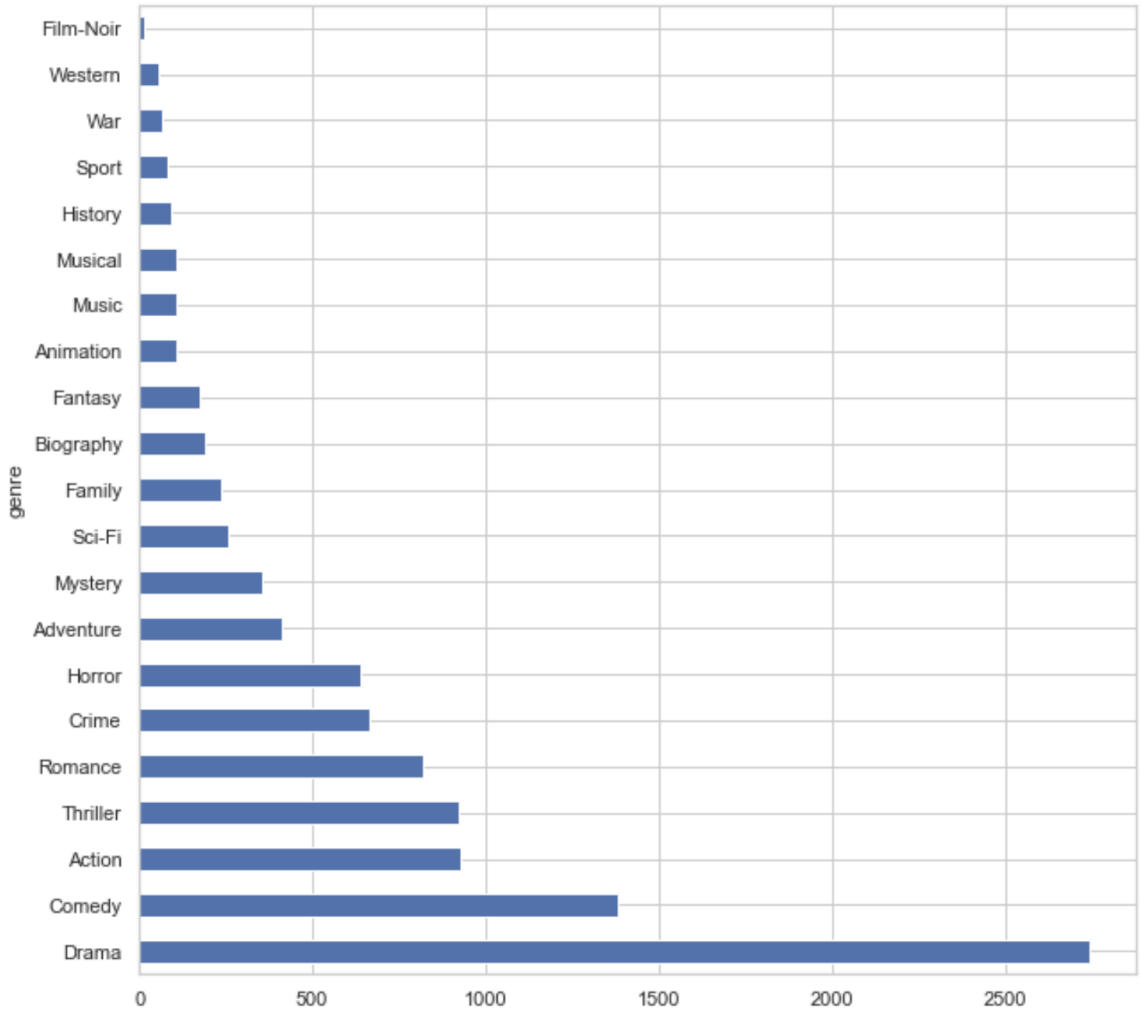


Figure 11. Movie analysis by genre

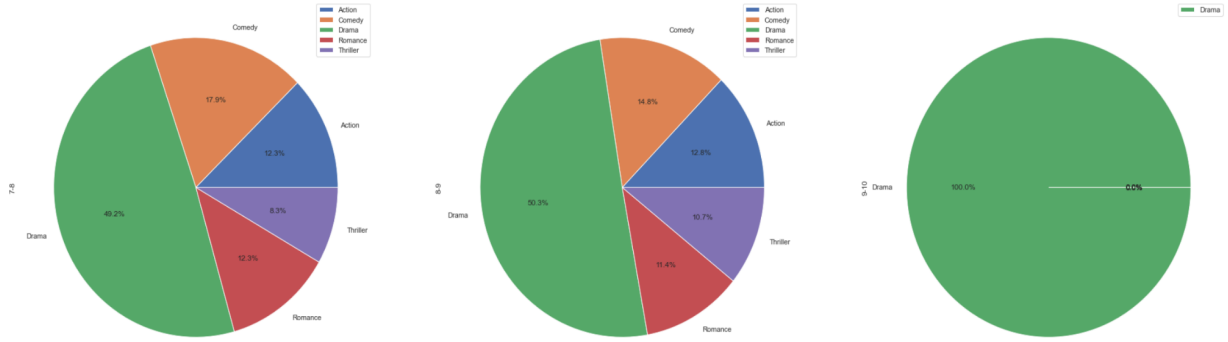


Figure 12. Top rating movie analysis by movie genre

Explanatory Variables(All Dummies)	Baseline to avoid perfect collinearity
DURATION RANGE: '< 95 min', '> 120 min'	'95 - 120 min'
GENRE: 'Adventure', 'Animation', 'Biography', 'Comedy', 'Crime', 'Drama', 'Family', 'Fantasy', 'History', 'Horror', 'Music', 'Musical', 'Mystery', 'Romance', 'Sci-Fi', 'Sport', 'Thriller', 'War', 'Western'	'Action'
DIRECTOR RANK RANGE: '25th_to_50th_percentile_director','50th_to_75th_percentile_director','75th_to_100th_percentile_director'	'0th_to_25th_percentile_director'
ACTOR RANK RANGE: '25th_to_50th_percentile_actor','50th_to_75th_percentile_actor','75th_to_100th_percentile_actor'	'0th_to_25th_percentile_actor'
PRODUCTION COMPANY RANK RANGE: '25th_to_50th_percentile_production','50th_to_75th_percentile_production','75th_to_100th_percentile_production'	'0th_to_25th_percentile_production'

Figure 13. Explanatory Variables and baselines of the prediction model

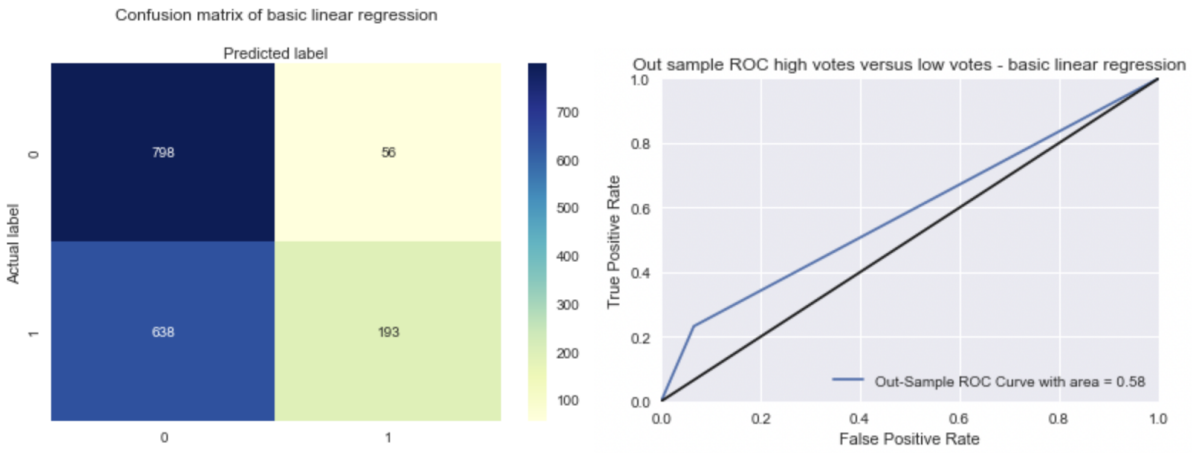


Figure 14. Model results of Basic linear regression

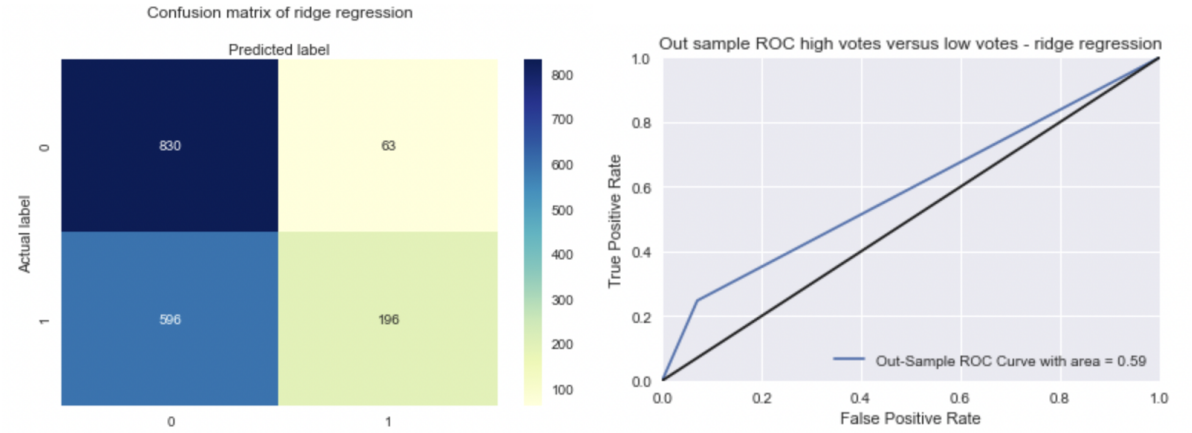


Figure 15. Model results of Ridge regression

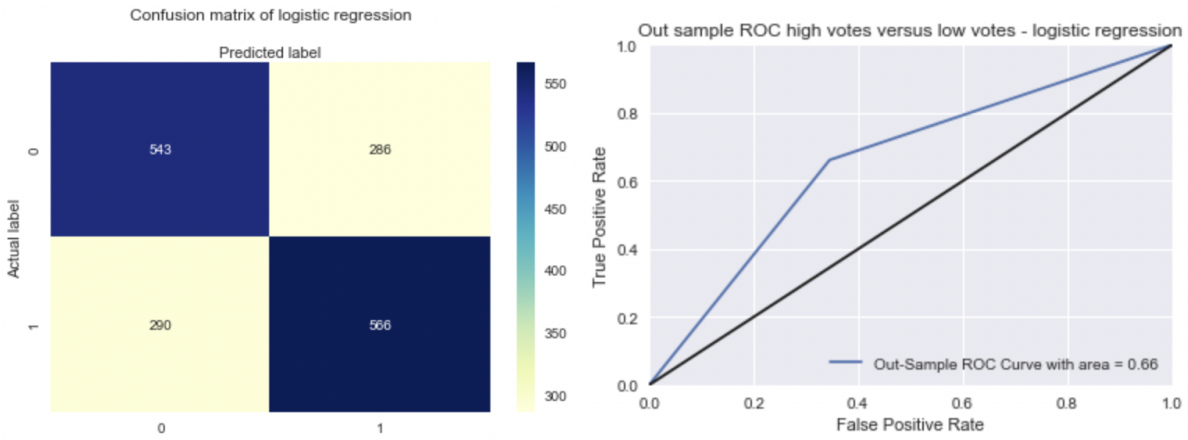


Figure 16. Model results of Logistic regression

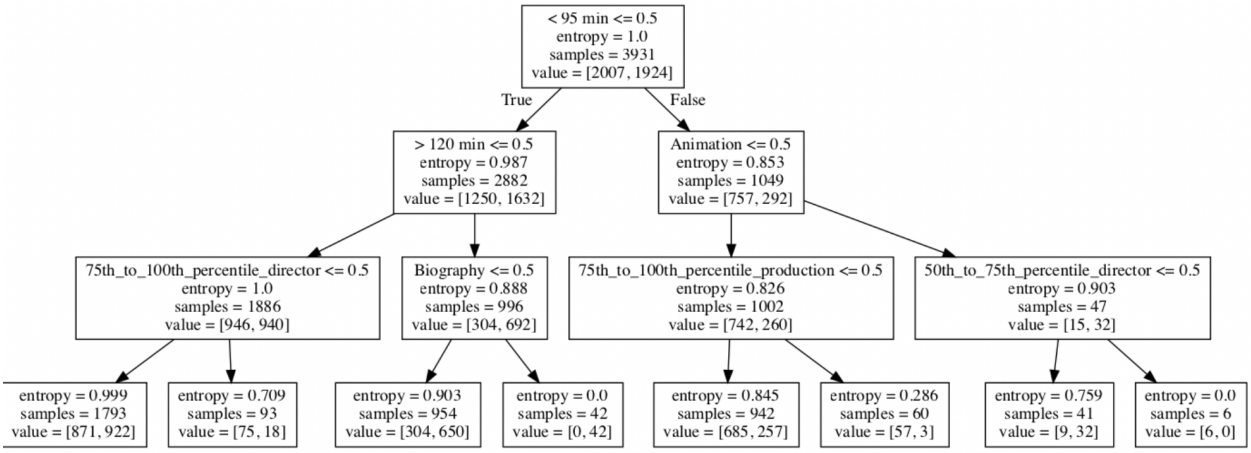


Figure 17. Decision Tree Diagram

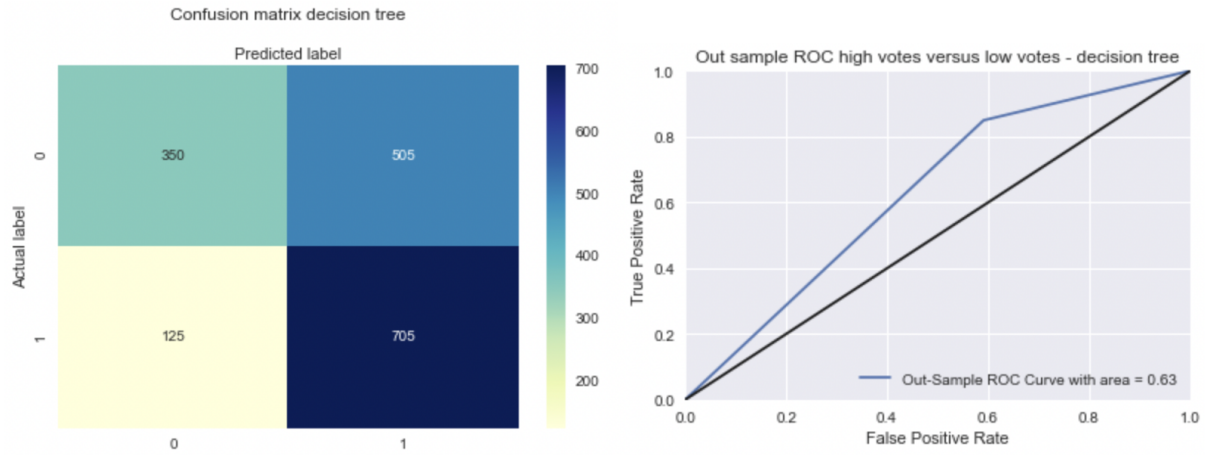


Figure 18. Model results of Decision Tree

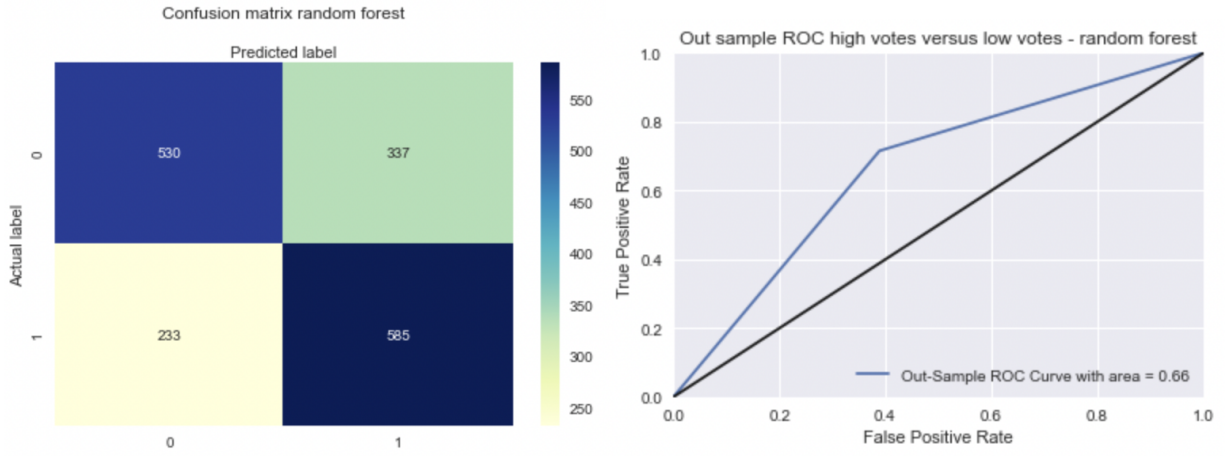


Figure 19. Model results of Random Forest

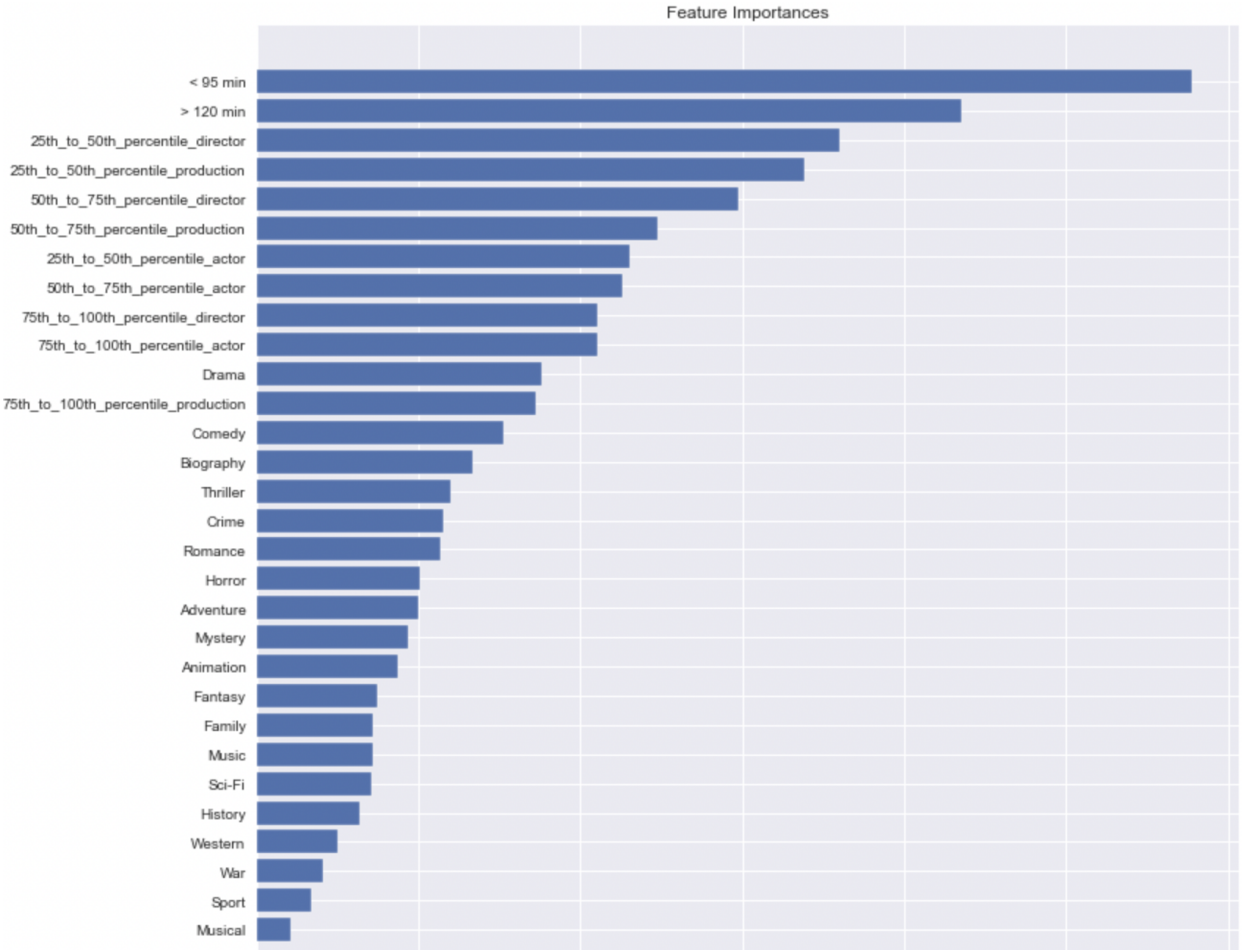


Figure 20. Feature Importance

	Basic Linear Regression	Ridge Regression	Logistic Regression	Decision Tree	Random Forest
Accuracy	0.59	0.61	0.66	0.63	0.66
Precision	0.78	0.76	0.66	0.58	0.63
Recall	0.23	0.25	0.66	0.85	0.72
AUC(ROC)	0.58	0.59	0.66	0.63	0.66

Figure 21. Model results comparison

Recommendation machine for movie lovers:

```
Please enter the name of your favorite movie: For example, [Jessie] for the movie Jessie Jessie
Do you want recommendations based on [1] for description only or [2] for a combination of genre, director, description?  2
                        title  rating
9                    Teenkahon     7.4
17                Chandramukhi     7.1
13  It Takes a Man and a Woman     6.7
11            Operation Finale     6.6
18                    Vox Lux     5.9
7                 The Neighbor     5.8
8                     Euphoria     5.8
4    Hamara Dil Aapke Paas Hai     5.6
16             Suburban Gothic     5.5
```

Recommendation machine for movie makers:

```
What is the duration of your movie in minutes? 128
What is the genre of your movie? Action
Who is the director? Guy Ritchie
Who is the main actor? Jude Law
What is the production_company? Warner Bros.
------------------------------------------------------------
For a movie of
 GENRE: Action
 DURATION: 128 minutes
 DIRECTOR: Guy Ritchie
 MAIN ACTOR: Jude Law
 PRODUCTION COMPANY: Warner Bros.
Congratulations, your movie will beat the average rating in the market!
------------------------------------------------------------
Do you want to try another combination: Yes or No? No
------------------------------------------------------------
Thanks for using our recommendation system. Have a wonderful day!
------------------------------------------------------------
```

Figure 22. Recommendation machines overview