Springboard - DSC

Capstone Project II

Recommending the Right Movies for the Best User Experience

Final Report

Data Science Career Track

December 2020

Sathwik Kesappragada

Table of Contents

1. Introduction

Company XYZ is a video streaming platform that lets its members watch TV shows and movies without advertisements on any internet-connected device. They want to suggest their users to watch the best quality and most relevant content. They aim to do so by building a recommendation engine that matches on user preferences. Before that they want to know among the successful movies, which features lead to a highly rated film. Company XYZ wants to build a model based on movie attributes recorded such as duration, vote count, vote average, release date, revenue data that was collected from a movie database to recommend the right film. By identifying as many target variables as possible from all the features, the data science team in company XYZ could approach their suggested movies/tv shows more efficiently and effectively.

1.1 Objective

The objectives of this project are to:

- Explore and analyze movies and ratings data for the XYZ video streaming platform
- Develop machine learning models that predict the rating
- Identify the key features that lead to a high rated film
- Identify the final model that captures most of the target variables
- Build a simple recommender system: collaborative filtering and content-based filtering

This report is divided into the following sections:

- Section 2: Dataset
- Section 3: Package Information
- Section 4: Data Wrangling
- Section 5: Exploratory Data Analysis
- Section 6: Machine Learning, Model Selection, & Hyperparameter Tuning
- Section 7: Recommendation Engines
- Section 8: Conclusion
- The programming codes for this report can be found at this GitHub Repository Movie-Recommender-System

1.2 Significance

By thoroughly exploring the dataset, we will identify key features that affect the rating of a film. We will also develop machine learning models that can be used by the software team of company XYZ to suggest accurate films to users yielding to better efficiency and high click rate accuracy/success.

2. Dataset

2.1 Data Description

The datasets were retrieved from the website kaggle, owned by Google, which is an online platform of data science and machine learning professionals who compete, discover and post datasets, research and build models all in one setting. The movie dataset used for this project was last updated November 11, 2017. The data consists of 45,466 rows and 23 columns, 6 of which that are numerical columns, and the rest are categorical columns. A description of each of the columns is provided in *Table 1.1*. The ratings dataset is made up of 2,600,000 rows and 4 columns.

No.	Variable Name	Variable Description	Data Summary
1	adult	Adult or Not	bool, False= Not Adult, True= Adult
2	belongs_to_collection	franchise/series a film belongs to	object, 4494 non-null
3	budget	cost of film	float, 1223 unique values
4	genres	categories associated with each film	object, 45466 non-null
5	homepage	official website of film	object, 7669 unique values
6	id	film ID	int, 45430 unique values
7	imdb_id	id of film associated in the IMDB database	int, 45413 unique values
8	original_language	language the film officially was shot in	object, 89 unique values
9	original_title	name of the film upon release	object, 43367 unique values
10	overview	description of film	object, 44303 unique values
11	popularity	score of how well know the film is	float, 43745 unique values
12	poster_path	url of the film poster image	object, 45021 unique values
13	production_companies	companies that contributed in the making of the film	object, 45463 non-null
14	production_countries	countries where the film was released/shot in	object, 45463 non-null
15	release_date	theatrical release date	datetime, 17333 unique values
16	revenue	total return of the film	float, 6863 unique values
17	runtime	duration of film in minutes	float, 353 unique values
18	spoken_languages	languages spoken in the film	object, 45460 non-null
19	status	state of film (released, rumored, post-production, etc.)	object, 6 unique values

20	tagline	punch line/catch phrase	object, 20283 unique values
21	video	indicates if TMDB has video of film	bool, True: it contains film, False: it does not have film
22	vote_average	average score of film	float, 92 unique values
23	vote_count	number of votes by users	int, 1820 unique values

Table 1.1 Description of dataset

No.	Variable Name	Variable Description	Data Summary
1	userId	identification of reviewer	int, 27047 unique values
2	id	id of film in the database	int, 25560 unique values
3	rating	review of film (out of 5)	float, 10 unique values
4	timestamp	time the review was submitted	datetime, 2112243 unique values

3. Package Information

Here, we used JupyterNotebook (6.03) to run all the code. Pandas (1.0.5), Matplotlib (3.2.2), Numpy (1.18.5), Seaborn (0.11.0) were installed as basic package. Scikit-learn (0.24.0) and sklearn (0.0) were installed as the machine learning library.

4. Data Wrangling

4.1 Dataset Information

General information about the original dataset can be found in *Table 4.1*, sorted by the percentage of unique values by column. It contains columns such as unique value count, missing value percentage, zero value percentage, description, and datatype are provided in the table below.

Table 4.1 General information of dataset

Variable Name	Counts	Unique Value Percentage	Missing Value Percentage	Num of Zeros Percentage	Data Type
imdb_id	45449	99.92	0.04	0.00	int64
id	45466	99.92	0.00	0.00	int64
poster_path	45080	99.87	0.85	0.00	object
overview	44512	99.53	2.10	0.00	object

Variable Name	Counts	Unique Value Percentage	Missing Value Percentage	Num of Zeros Percentage	Data Type
tagline	20412	99.37	55.10	0.00	object
homepage	7782	98.55	82.88	0.00	object
popularity	45461	96.23	0.01	0.15	float64
original_title	45466	95.38	0.00	0.00	object
release_date	45379	38.20	0.19	0.00	datetime64[ns]
revenue	15460	15.10	0.01	83.72	float64
vote_count	45460	4.00	0.01	6.37	float64
budget	45466	2.69	0.01	80.45	float64
runtime	15203	2.32	0.58	3.43	float64
vote_average	45466	0.20	0.01	6.59	float64
original_languages	45455	0.20	0.02	0.00	object
status	45379	0.01	0.19	0.00	object

*variable that has missing value percentage higher than 50% is highlighted in yellow

Variable Name	Counts	Unique Value Percentage	Missing Value Percentage	Num of Zeros Percentage	Data Type
userId	2600000	1.04	0.00	0.00	int64
id	2600000	0.98	0.00	0.00	int32
rating	2600000	0.00	0.00	0.00	float64
timestamp	2600000	81.24	0.00	0.00	datetime64[ns]

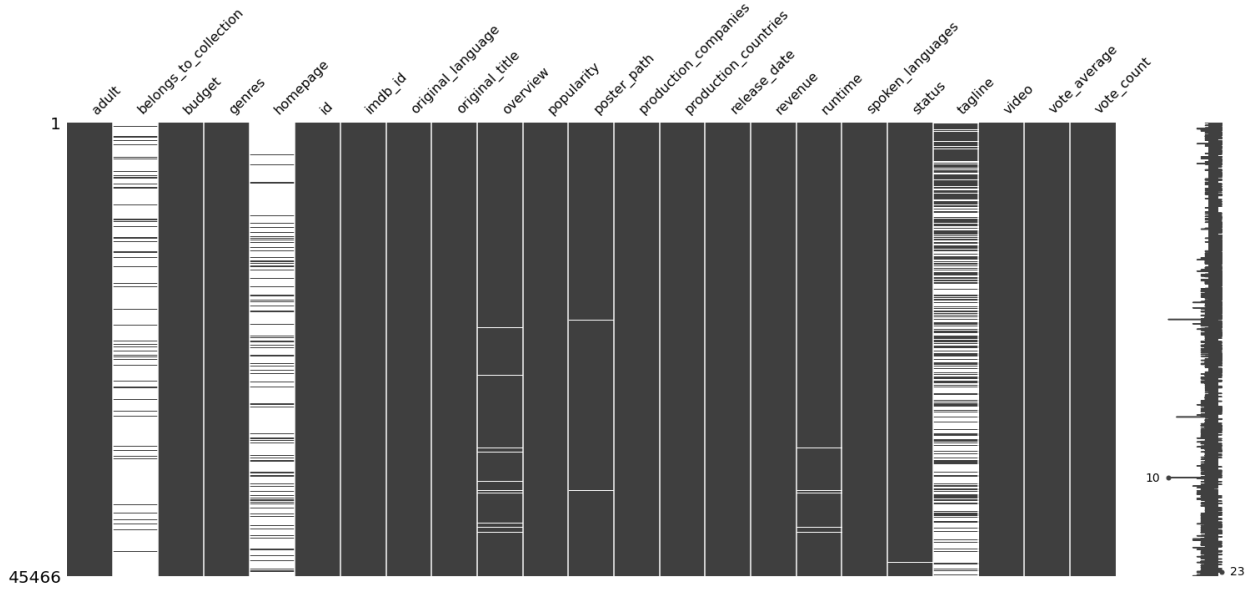


Figure 4.1 Missing data in movies dataset

4.2 Data Cleaning

The shape of the dataset is 45466 rows by 23 columns. Some columns were in the form of a stringified JSON object that needed to be converted into lists. The data frame struggled with any analysis pertaining to these columns (genres, production_countries, belongs_to_collection) because they were unhashable lists. Additionally, all columns with misclassified data types were converted into proper types (integer, string, float) after removing unvaluable data. Release_date was broken up into day, month, year to extract more findings. Duplicates were identified by using a combination of release_date and original_title as a key and dropped.

Table 4.2 Preprocessing variables

Variable Names	Missing Value Percentage	Process Method
genres	0.00	data type correction
production_companies	0.01	data type correction, (one-hot-encoding)
production_countries	0.00	data type correction, (one-hot-encoding)
belongs_to_collections	0.00	data type correction
spoken_languages	0.00	data type correction, imputed with empty string
release_date	0.19	data type correction
budget	0.01	data type correction
popularity	0.01	data type correction
rating (ratings)	0.00	multiplied by 2
timestamp (ratings)	0.00	data type correction

id (ratings)	0.00	data type correction
id	0.00	data type correction

* some features listed are from ratings dataset and are labeled

Blackout	Blackout	Blackout	Blackout
FALSE	FALSE	FALSE	FALSE
[]	[]	[]	[]
0.000000	0.000000	0.000000	0.000000
[Thriller, Mystery]	[Thriller, Mystery]	[Thriller, Mystery]	[Action, Thriller]
NaN	NaN	NaN	NaN
141971	141971	141971	100063
tt1180333	tt1180333	tt1180333	tt0077241
fi	fi	fi	en
Blackout	Blackout	Blackout	Blackout
Recovering from a nail gun shot to the head an...	Recovering from a nail gun shot to the head an...	Recovering from a nail gun shot to the head an...	A black comedy of violent criminals who terror...
0.411949	0.411949	0.411949	0.314595
VSZ9coCzxOCW2wE2Qene1H1fKO.jpg	/8VSZ9coCzxOCW2wE2Qene1H1fKO.jpg	/8VSZ9coCzxOCW2wE2Qene1H1fKO.jpg	/ddyDGQBLbG1LjK01dz9Nb1NQstf.jpg
{'name': 'Filmiteollisuus Fine', 'id': 5166}]	[{'name': 'Filmiteollisuus Fine', 'id': 5166}]	[{'name': 'Filmiteollisuus Fine', 'id': 5166}]	[]
[Finland]	[Finland]	[Finland]	[United States of America]
12/26/2008	12/26/2008	12/26/2008	8/25/1978
0.000000	0.000000	0.000000	0.000000
108.000000	108.000000	108.000000	92.000000
[suomi]	[suomi]	[suomi]	[English]
Released	Released	Released	Released
Which one is the first to return - memory or t...	Which one is the first to return - memory or t...	Which one is the first to return - memory or t...	The night the power failed.... and the shock b...
False	False	False	False
6.700000	6.700000	6.700000	5.000000
3.000000	3.000000	3.000000	1.000000

Figure 4.2

5. Exploratory Data Analysis

5.1 Summary Statistic

Statistics such as count, mean, standard deviation, percentiles of each numerical variable were compiled in *Table 5.1*.

Table 5.1 Summary statistics

variable name	count	mean	std	min	25%	50%	75%	max
budget	45430.0	4224828	17428530	0.00	0.00	0.00	0.00	380000000
popularity	45430.0	2.92	6.01	0.00	0.39	1.13	3.68	547.49
revenue	45430.0	11212880	64352130	0.00	0.00	0.00	0.00	2787965000
runtime	45173.0	94.12	38.42	0.00	85.00	95.00	107.00	1256.00
vote_average	45430.0	5.62	1.92	0.00	5.00	6.00	6.80	10.00
vote_count	45430.0	109.94	491.47	0.00	3.00	10.00	34.00	14075.00
year	45346.0	1992.00	24.05	1874.00	1978.00	2001.00	2010.00	2020.00
day	45346.0	14.21	9.28	1.00	6.00	14.00	22.00	31.00
month	45346.0	6.46	3.63	1.00	3.00	7.00	10.00	12.00
num_genres	45430.0	2.00	1.13	0.00	1.000	2.00	3.00	8.00

5.2 Overall Distribution

The overall distribution of numerical variables is visualized in *Figure 5.2*.

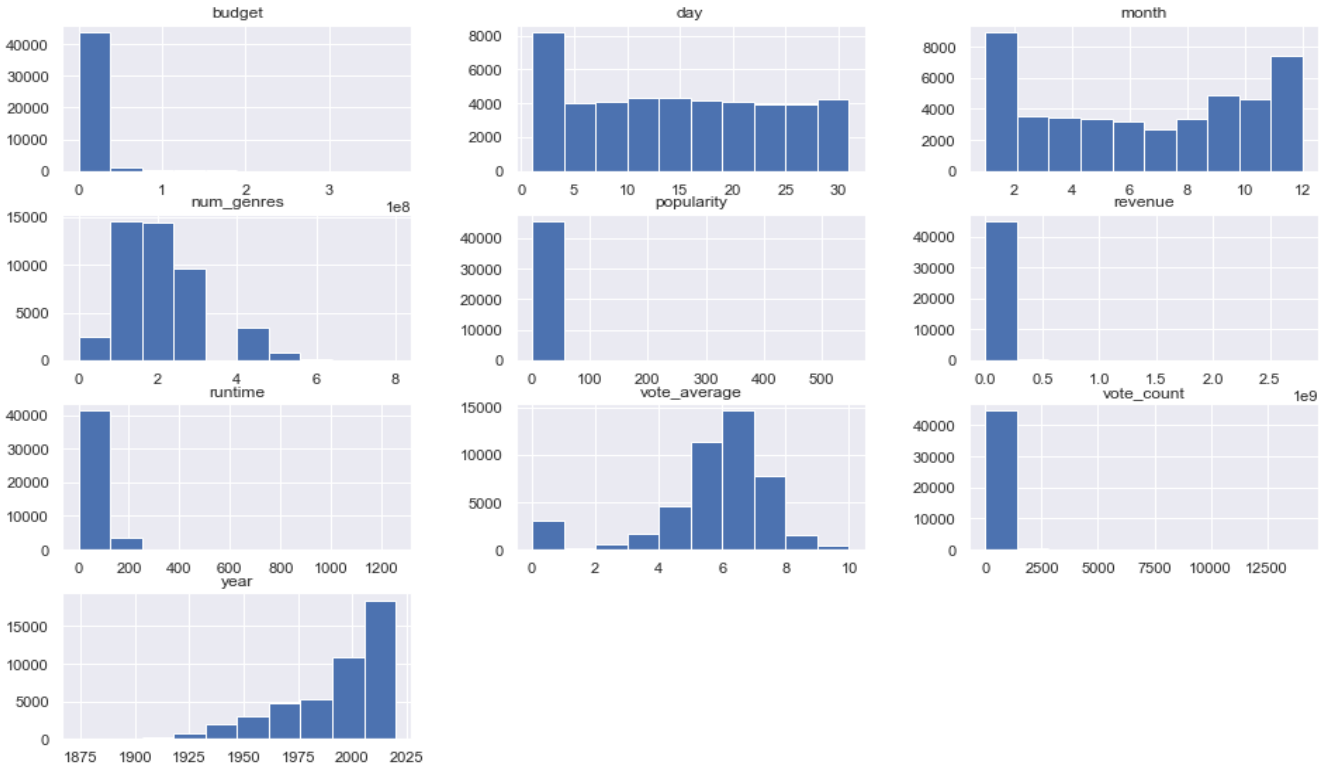


Figure 5.2 Distributions of numerical features

Please note that this was visualized after data type correction

Here are basic plots, each displaying some meaning that will be discussed later:

- Most films have budgets below 100 million
- Most films get released in the first five days of every month
- January has the most films
- Most films have one to three genres
- Significant amount of the votes fall between 5.0 to 8.0
- There has been an increase in films being produced every year

5.3 Release Dates

First question that pops up in mind when thinking about an upcoming film is the release date. Movie makers are always concerned and meticulous about choosing an appropriate date for release, making sure the time periods do not overlap with the competition and anticipating significant amounts of attention surrounding the promotion.

- Oldest film in the dataset: Passage of Venus (1874-12-09)
- Latest film in the dataset: Avatar (2020-12-16) (yet to release)

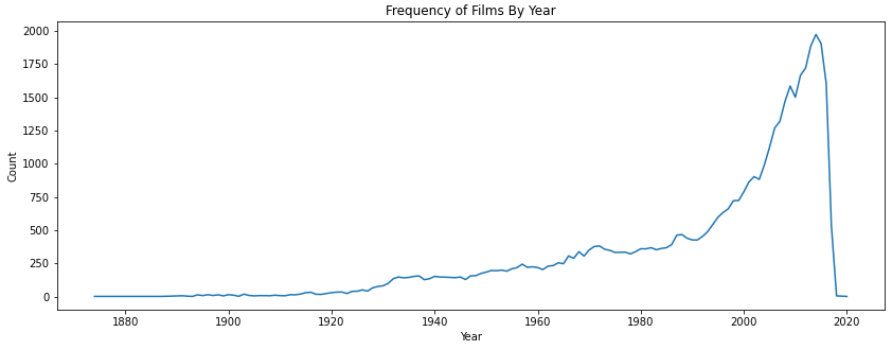


Figure 5.3, With technological advancements every year, the filmmaking process is becoming much easier that more movies are being produced at quicker speeds and this can be displayed by the graph on the left. In the age of video streaming platforms, content creators want to get their works out.

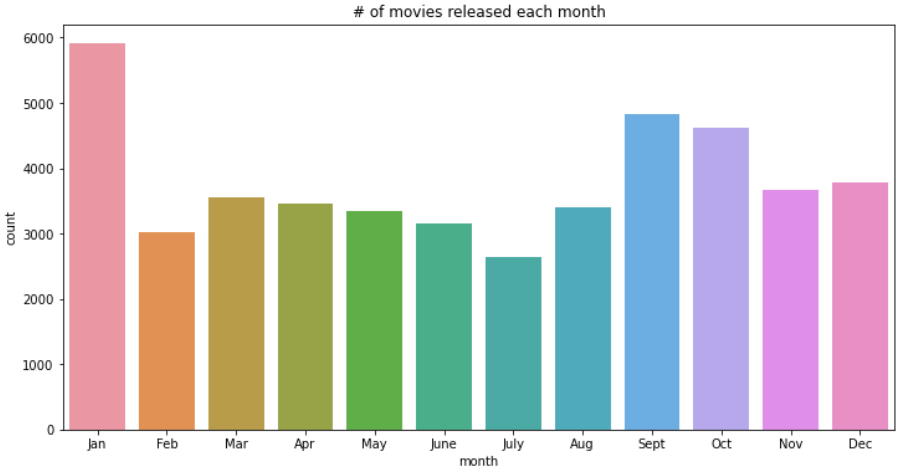


Figure 5.3, After splitting the release date feature into multiple subparts, the months that have the greatest number of movies released can be visualized. One may expect summer for having the most films announced, but the graph on the left does not agree. Most theatrical release dates of films fall in the months of January, September, and October. Subpar movies that were not released in the previous year are released at the beginning of a new year.

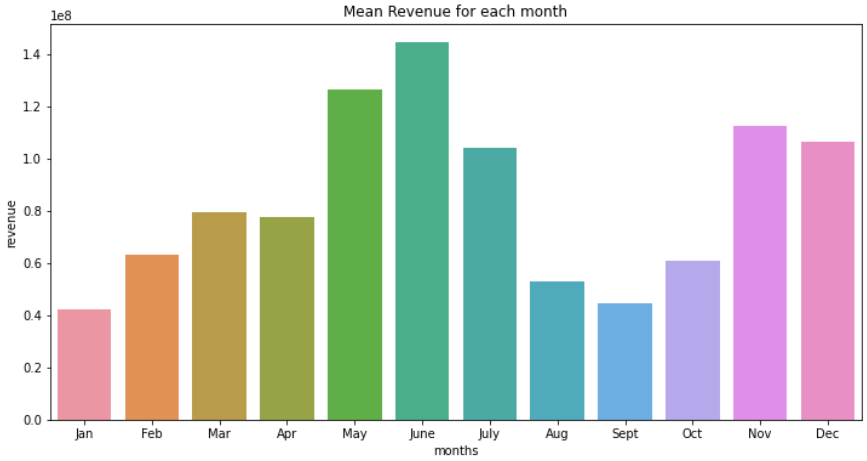


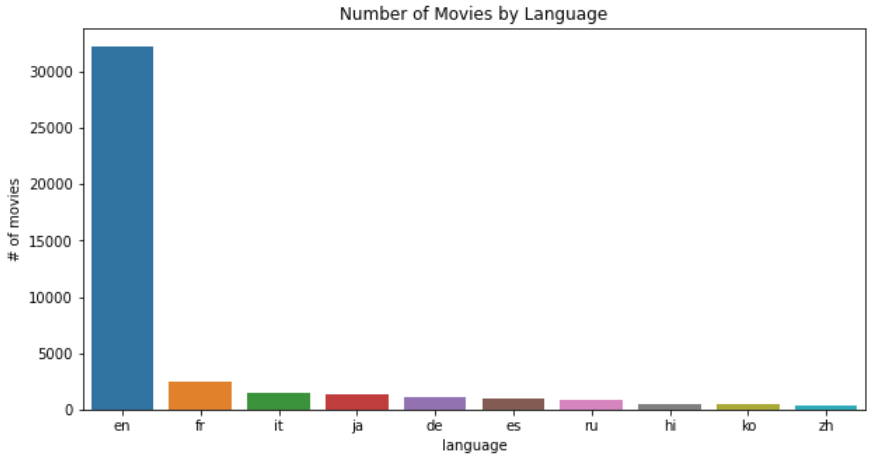
Figure 5.3, Continuing with examining the months feature, summer is usually the time when films make the most money in return. There is always hype surrounding a film that gets released around the time most people want to spend their time outside being active. May, June, and November are the top three months with the highest turnout. Movie theaters are expected to get busy during holiday season.

5.4 Films Released by Country and Language

Most of the films (~40%) in the dataset being inspected are from Hollywood. There exists a small subset (~14%) of films without a country label and British Cinema appears third on the list (~5%). Together with, English, French, and Italian dominating in the spoken languages section, much of the data set focuses on English flicks. Language and country of origin are imperative characteristics of any movie. Film has become a widely known form of communication and highly influential art.

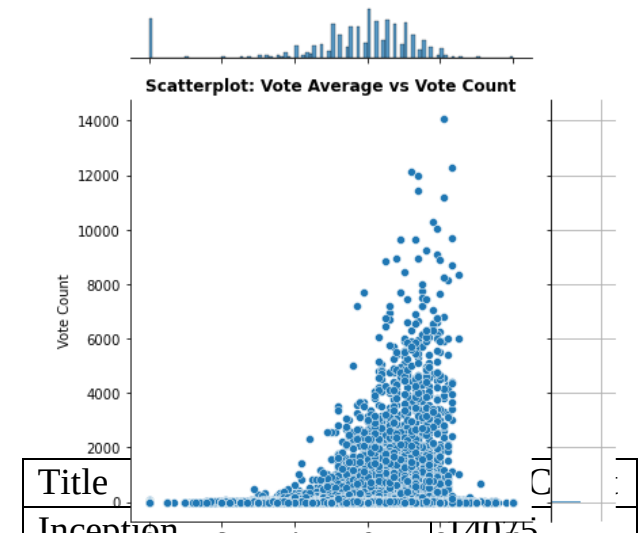
country	# of movies
United States of America	17841
-	6279
United Kingdom	2238
France	1653
Japan	1354
Italy	1030
Canada	840
Germany	748
Russia	735
India	735

Figure 5.4 Film Count by Language



5.5 Vote Count & Vote Average

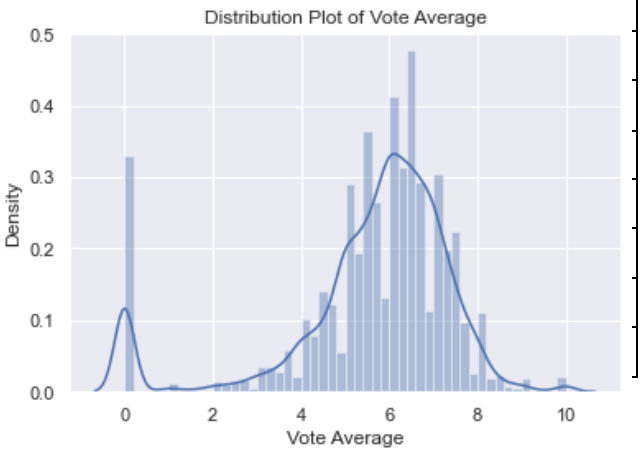
The chart on the left shows the films with the greatest number of votes from the movies dataset. Three out of the top then, Avengers, Deadpool, Guardians of the Galaxy, are from the Marvel franchise.



Title	C
Inception	14075
The Dark Knight	12269
Avatar	12114

Relationship between vote count and vote average is visualized on the left. A dual sided histogram and scatterplot suggests that more votes are likely to yield to a higher vote average and resembles a true rating of a film.

Figure 5.5



The distribution of vote average is shown on the left. The drawn line is not only attempting to symbolize a Gaussian distribution, but also indicates the outliers in the dataset. A huge subset of vote average falls in the range of 4.5 to 6.5.

Figure 5.5

5.6 Popularity, Budget & Revenue

After a film gets released, success metrics are usually monitored to determine the results.

Title	Popularity
Minions	547.49
Wonder Woman	294.34
Beauty and the Beast	287.25
Baby Driver	228.03
Big Hero 6	213.85
Deadpool	187.86
Guardians of the Galaxy Vol. 2	185.33
Avatar	185.07
John Wick	183.87
Gone Girl	154.80

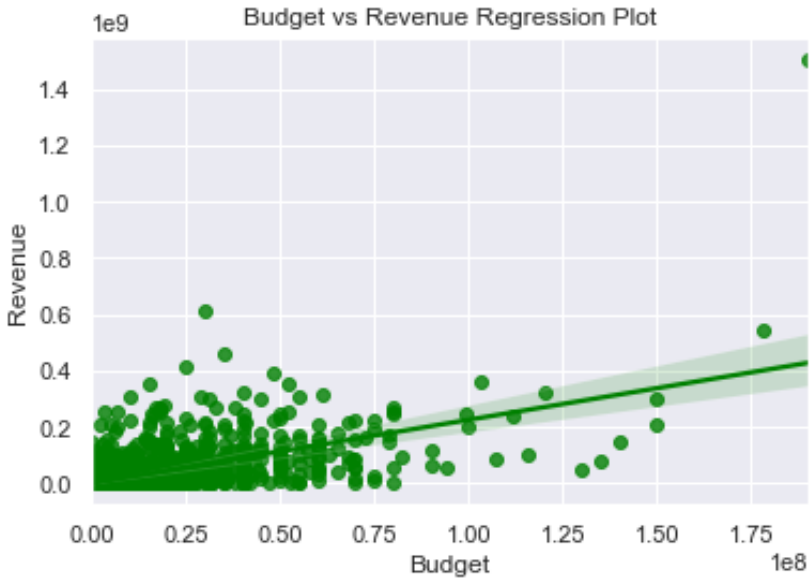
Minions, Wonder Woman, and Beauty and the Beast are the most popular films according to TMDB's Popularity Score.

Over a long period of time, Avatar, Star Wars: The Force Awakens, and Titanic made the most in return accumulating billions of dollars.

Title	Revenue
Avatar	2787965000
Star Wars: The Force Awakens	2068224000
Titanic	1845034000
The Avengers	1519558000
Jurassic World	1513529000
Furious 7	1506249000
Avengers: Age of Ultron	1405404000
Harry Potter and the Deathly Hallows: Part 2	1342000000
Frozen	1274219000
Beauty and the Beast	1.262886000

Title	Budget
Pirates of the Caribbean: On Stranger Tides	380000000
Pirates of the Caribbean: At World's End	300000000
Avengers: Age of Ultron	280000000
Superman Returns	270000000
Transformers: The Last Knight	260000000
Tangled	260000000
John Carter	260000000
Spider-Man 3	258000000
The Lone Ranger	255000000
The Dark Knight Rises	250000000

Two out six of the Pirates of the Caribbean movies are the two highest budgeted films in the dataset. Avengers: Age of Ultron and Superman Returns follows.



This regression plot emphasizes on the relationship between budget and revenue and reveals that not every highly budgeted film result in a profit. Most film budgets are below 50 million.

Figure 5.6

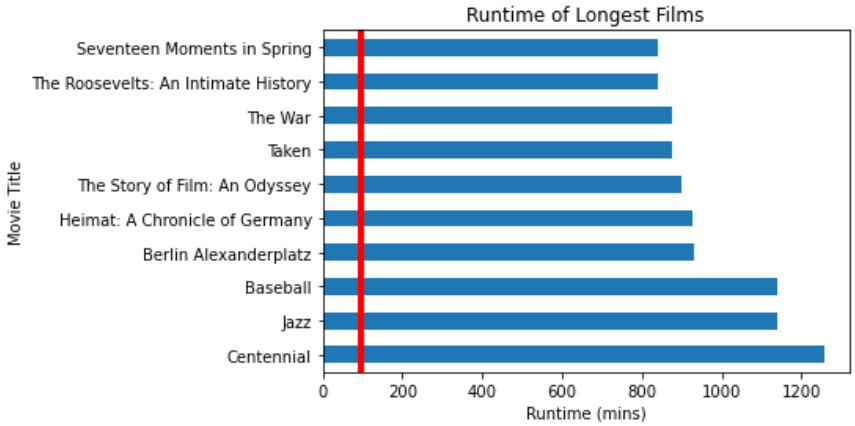
*This graph was made with movies that have only one genre

5.7 Runtime

The average length of a film is about 1 hour and 34 minutes. The top ten longest and shortest films of the dataset are listed below. The shortest films made were outcomes of the initial spike in filmmaking during the late 1800s and early 1900s. The first ever films made lacked present-day technology. Back then, one minute of pictures displayed sequentially was astonishing. Several of these long films are between 15 hours to 20 hours and are all one-hour episode TV shows.

Title	Runtime
Mr. Edison at Work in His Chemical Laboratory	1.00
Grandma's Reading Glass	1.00
What Happened on Twenty-Third Street, New York City	1.00
The Magician	1.00
Panorama pris d'un train en marche	1.00
Divers at Work on the Wreck of the "Maine"	1.00
After the Ball	1.00
Between Calals and Dover	1.00
The Surrender of Tournavos	1.00
Blacksmith Scene	1.00

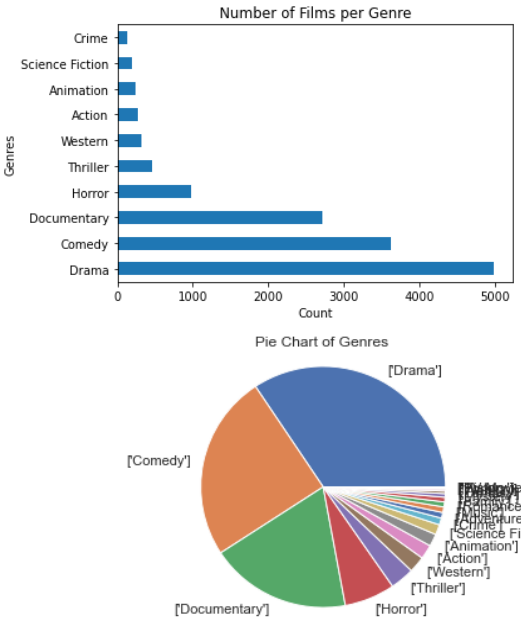
Figure 5.7

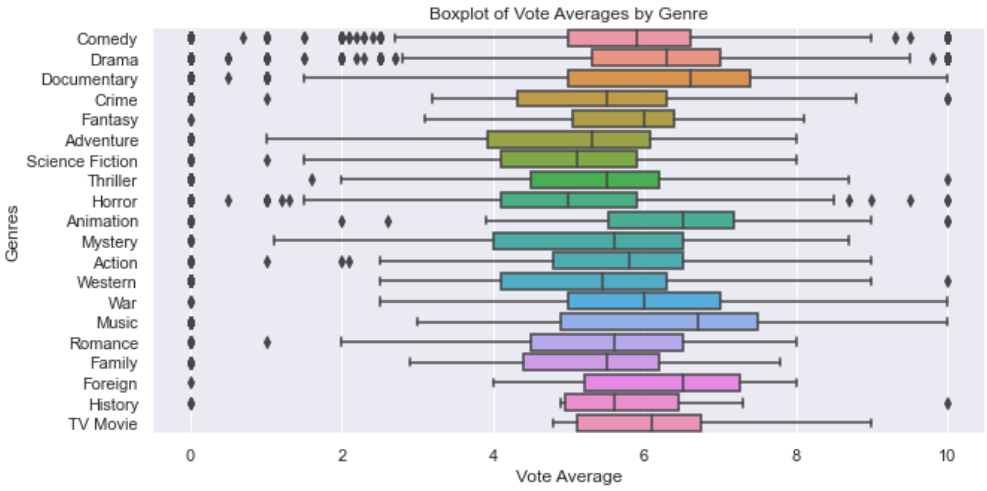


5.8 Genres

Not only were ‘genres’ a tough feature to access because it was a unhashable list, but also there were 4065 different combinations of ‘genres’ entered, some ranging from one to eight genres per a movie. Observing films with one genre was much simpler to accomplish and draw analysis from. Drama had the highest vote average count and vote average mean. Comedy and Documentary follows. TV movie, History, and Foreign had the least number of films. Drama and Romance, Comedy and Drama, Comedy and Romance are the most popular pairs of genres.

Figure 5.8





ıgs Dataset

ɡs datasets, the shape of the data is 1183206 rows by 30 columns. ıg the data was to identify valuable features that affect the rating of a film using machine learning models so it could help improve the recommendation system’s accuracy and success rate. Any null ratings and null user IDs in records represents users not having seen that specific film and are deemed unimportant.

5.10 Variable Correlation Coefficient

After plotting heatmap *Figure 5.10*, not much correlation coefficient was found among the variables.

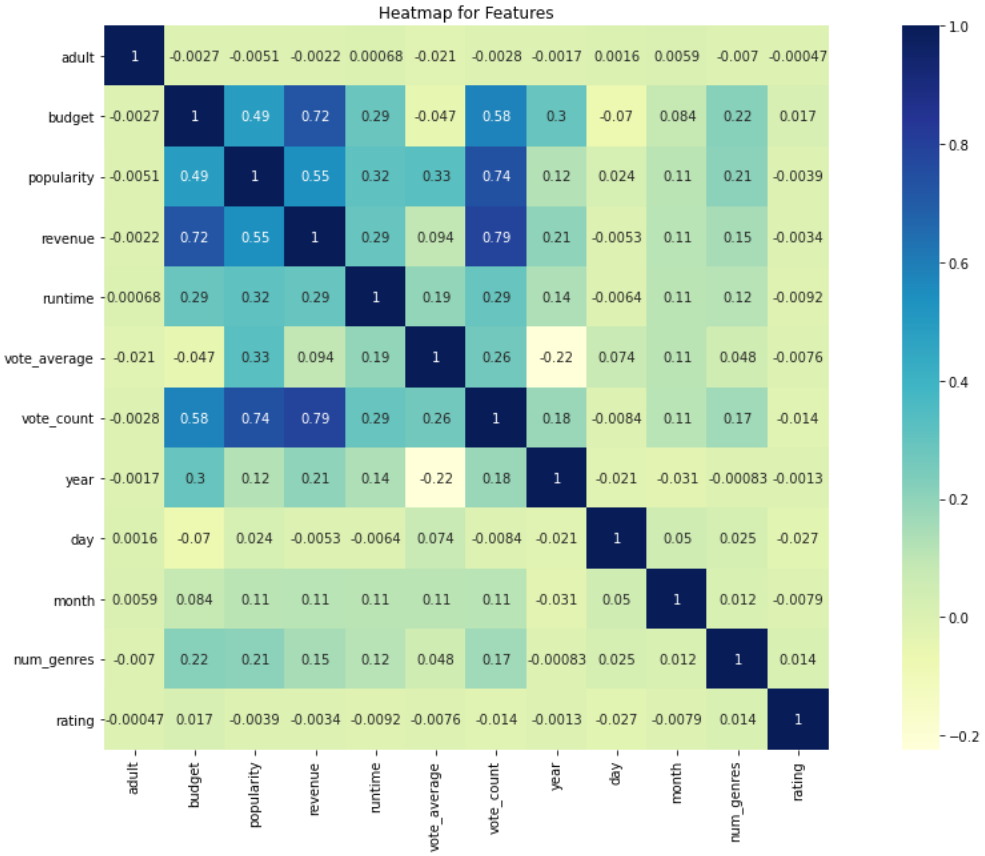


Figure 5.10 Heatmap of Correlation coefficients between variables

6. Machine Learning: Regression – Predicting Movie Ratings

Models ran on a small fraction (2%) of the merged data set due to computational power. Considering the objective of determining key features, variables such as ‘revenue’, ‘budget’, ‘vote_count’, and ‘runtime’ needed to contain values greater than 0. The data was split 75%/25% training/testing sets to run against ‘ratings’ variable. The process of feature engineering in total added 24 variables: 10 for each of the one hot encoded categorical variables, production countries and production companies, top 3 genres, and ‘is_Summer’.

- **Shape of Test set: (8130,35)**
- **Shape of Train set: (2711, 35)**

6.1 Data Preprocessing and Feature Selection/Engineering

- Categorical variables 'production_companies' and 'production_countries' were converted into dummy variables and only top 10 appearing values were kept while rest were dropped because there were many differing sets
- 'original_language' transformed into a binary feature, indicating 1 if the film was originally shot in English and 0 otherwise
- is_Summer' is a binary feature added indicating if the film's release date month falls during the summertime (May – September)
- Any nulls in 'runtime' were imputed using the mean value
- Nulls of 'year', 'month', 'day' were dropped
- Categorical variables such as 'original_title', 'adult', 'genres', 'homepage', 'belongs_to_collection', 'production_companies', 'production_countries', 'spoken_languages' are all dropped
- "Is_(Drama, Comedy, Thriller)" is a binary feature added to check if the film listed as one of these top 3 genres

6.2 Model Metrics

Model Metrics using to measure model performance and their meaning:

- Explained Variance Score: measures the difference between the model and actual data
- Mean Absolute Error (MAE): mean of all absolute errors
- Mean Squared Error (MSE): distance between the regression line and predicted values (set of errors)
- Mean Squared Log Error (MSLE): percentual difference between the model and actual data
- R^2 score: (coefficient of determination) how well the data is to a fitted regression line, indicates the variance
- Median Absolute Error: median differences between predicted and actual observations
- Root Mean Squared Error (RMSE): square root of MSE, direct relationship to R^2, measures the differences between residuals

6.3 Model Performance, Hyperparameter Tuning, & Evaluation

Baseline Model Metrics:

At first, the models were trained using baseline implementation, in other words hyperparameters were left as default. The evaluations were all conducted over the same train/test split 75%/25% with 5folds Cross Validation. (best model is highlighted in yellow)

Hyperparameter Tuned Metrics:

On the contrary, hyperparameters were tweaked and given a range of values for its arguments. Model performance improved in all aspects, but the results remain similar.

	Model	Explained Variance Score	Mean Absolute Error (MAE)	Mean Squared Error (MSE)	Mean Squared Log Error (MSLE)	R^2 score	Median Absolute Error	RMSE
1	RF	0.13	1.57	4.03	0.19	0.13	1.28	2.01
2	GB	0.09	1.64	4.23	0.11	0.09	1.24	2.04
3	XGB	0.12	1.58	4.07	0.10	0.12	1.28	2.11

	Model	Explained Variance Score	Mean Absolute Error (MAE)	Mean Squared Error (MSE)	Mean Squared Log Error (MSLE)	R^2 score	Median Absolute Error	RMSE	Final RMSE
1	RF	0.13	1.57	4.03	0.10	0.13	1.26	2.01	2.00
2	GB	0.13	1.58	4.04	0.10	0.13	1.26	2.01	2.00
3	XGB	0.13	1.58	4.05	0.10	0.13	1.29	2.01	2.06

6.4 Feature Importances

As displayed in the chart above, Random Forest outperforms the other models in all categories. Even though the R^2 score is low here, conclusions can still be drawn on statistically significant predictors. Regardless of the value, predictors that affect the model's performance are meaningful and can be displayed for each model by the figures below.

Random Forest

Figure 6.4 For Random Forest, Revenue, Runtime, Popularity are the top 3 most important features

Gradient Boosting

Figure 6.4 For Gradient Boosting, Revenue, Popularity, and Budget are the top 3 most important features

XGBoost

Figure 6.4, For XGBoost, these are the top 3 most important features:

production companies**:** 'Columbia Pictures','Intermedia Films', 'Warner Bros.', 'C-2 Pictures', 'IMF Internationale Medien und Film GmbH & Co. 3 Produktions KG', 'Mostow/Lieberman Productions',

production companies: 'Columbia Pictures', 'Amblin Entertainment', 'Columbia Pictures Corporation, 'Parkes_MacDonald Image Nation',

production companies: 'Village Roadshow Pictures', 'Robert Simonds Productions', 'Warner Bros.', 'Phoneix Pictures', 'Underground', 'Proposal Productions'

The idea behind this machine learning process was to identify which metrics have the most value when it comes to predicting the rating of a film. From the feature importances, revenue and popularity appeared to be in the top 3 for two of the higher performing models. Furthermore, those two variables affect movie reviews.

7. Recommender Systems

Movies data file is merged with credits and keywords to make a combined dataset.

Credits: credits for a particular film (Director, Cast, Characters, etc.)

Keywords: keywords pertaining to the plot of a film

There are two main types of recommender systems: (1) collaborative filtering and (2) content-based filtering. In context, focusing on user behavior, collaborative filtering can be used to discover patterns based off user preferences, and suggestions can be determined using correlation or similarity. Here, movies that have not been watched will appear as nulls and prior to executing a model, ratings will be imputed with zeros. Another implementation can involve imputing the ratings using the average as a technique used to predict user behavior however, following the intuition that users will rate similar items, similarly, the nearest neighborhood will consist of most similar items to recommend. Content based filtering performs based off the characteristics of a film. Using feature importances from the models built and focusing on those specific aspects yield to another approach for recommendations.

7.1 Collaborative Filtering

Correlation

Another recommendation technique is creating a matrix with user IDs as rows and film titles as columns, making it easier to see what rating each user has given to every movie. Null records indicate a user has yet to watch that movie. On the side, create another data frame with movie titles, rating, and number of ratings, sorted by the count. Then taking a specific movie (column of all ratings) from the first matrix create a variable and correlate it with the full movie matrix. With that result using the second data frame created, identify films that are highly correlated to each other, but also have a count of more than 100 ratings. Below are the results for Jumanji.

original_title	Correlation	num_ratings
EVA	1.0	1062
Shiloh	1.0	578
Saving Grace	1.0	263
Du rififi chez les hommes	1.0	483
Juste une question d'amour	1.0	972

Figure 7.1

KNearest Neighbor

Similarly, to content-based filtering, another movie matrix can be created with movie titles as rows and user IDs as columns. Next, converting the pivot table into an array matrix using scipy library will help with calculating neighbors of observed films by cosine similarity and Euclidean distance. Whichever films are the closest to the film entered, based on their distance five films will be recommended. Below is an example with the film 300.

```
Recommendations for 300:

1: Rocky Balboa, with distance of 0.6608025529091446:
2: The Prestige, with distance of 0.6872639724192523:
3: Madagascar, with distance of 0.6942923891642457:
4: Whale Rider, with distance of 0.6973719879556649:
5: Blood: The Last Vampire, with distance of 0.7076322952914149:
```

Figure 7.1

Predictive Machine Learning

	uid	iid	actualRating	Estimation	Details	#_items_rated_user	#_users_rated_item	err
0	1602.0	tt0762107	4.0	4.852716	{'was_impossible': False}	155	34	0.852716
1	9341.0	tt0014646	10.0	9.295414	{'was_impossible': False}	308	263	0.704586
2	21467.0	tt0461804	4.0	4.314972	{'was_impossible': False}	190	239	0.314972
3	22811.0	tt0061590	8.0	7.551860	{'was_impossible': False}	201	1081	0.448140
4	9279.0	tt0057426	6.0	5.045622	{'was_impossible': False}	400	690	0.954378

Figure 7.2

Using matrix factorization and singular value decomposition, ratings that have been imputed by zeros can be replaced with predicted ratings which could yield much more accurate movie suggestions to the user. This process is best incorporated when doing collaborative filtering since it involves ratings and other users. Keeping in mind that the rating scale is 1-10, RMSE of this model was 1.70 making it 0.85 for a 1-5 scale. There exist predictions that have extremely high errors between the actual and predicted ratings and served as outliers. This is mostly due to personal preference. Some people may have a biased opinion towards their favorite movie director, actor, production company, and automatically give a positive review. Moreover, this method could also be used for a cold start problem. When a new user joins, he/she will be prompted with a handful of choices so the system can understand the person's preferences. On the other hand, for a movie that gets newly added to the database, it will be compared and suggested based on similarity across features.

When comparing the brute force method (imputation with zeros) and predicted ratings (imputed predictions), the biggest difference was the Euclidean distance. Here the movies are listed as IDs.

```
Recommendations for 1964:

1: 1310, with distance of 0.0011612150816283 64
2: 434, with distance of 0.001191177392222964
3: 4767, with distance of 0.0012076189661280878
4: 1023, with distance of 0.001208123707562514
5: 2007, with distance of 0.001213115905096096
```

Figure 7.1

7.2 Content-Based Filtering

Soup: All Categorical Features

Taking a fraction of the combined dataset and filling in the null values for categorical columns such as overview and tagline with empty strings will help the system moving forward. In effect, create a column by combining overview and tagline to get a soup of details. Then, another column is created with a combination of keywords, cast, genres, and details after all the variables agree. Using TfidfVectorizer fit transform the combination of all aspects of a movie and cosine similarity to match films that are similar. Below are the recommendations for the movie Toy Story.

```
print(get_recs('Toy Story',cosine_sim, indices))
```

executed in 36ms, finished 03:32:18 2020-12-29

```
1                            Jumanji
2                  Grumpier Old Men
3                 Waiting to Exhale
4        Father of the Bride Part II
5                               Heat
6                            Sabrina
7                       Tom and Huck
8                       Sudden Death
9                          GoldenEye
10           The American President
11       Dracula: Dead and Loving It
12                             Balto
13                             Nixon
14                 Cutthroat Island
15                           Casino
16             Sense and Sensibility
17                        Four Rooms
18    Ace Ventura: When Nature Calls
19                        Money Train
```

Figure 7.1

Hybrid Technique: Scaled Weighted Average and Scaled Popularity Score

Like TMDB, IMDB has their own scaling method. They rank their films using a set formula and it is shown below. Using this ranking method and setting the cut off to be at the 90th percentile for vote counts, the weighted average calculated can help identify the top films. Now with the help of MinMaxScaler, 50% importance can be given to valuable features such as popularity and the weighted average just calculated to retrieve a new list of to be recommended based off ratings and popularity score. Below is the list of top 10 movies.

Weighted Rating (WR) =

$$(\frac{v}{v+m} * R) + (\frac{m}{v+m} * C)$$

where,

- v: number of votes for each film
- m: minimum number of votes
- R: the average rating of a fimn
- C: the mean vote throughout the dataset

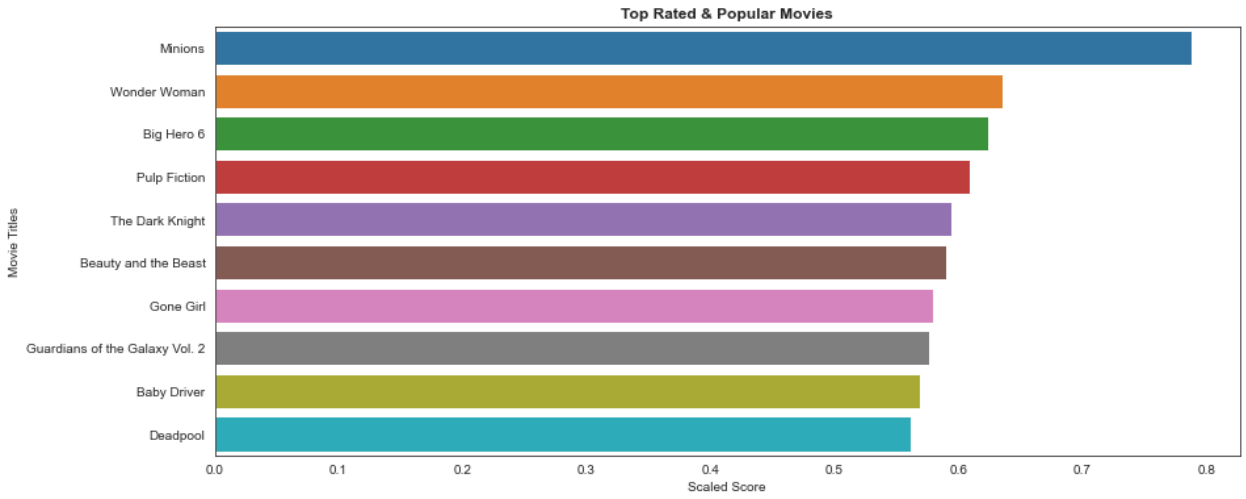


Figure 7.2, Minions, Wonder Woman, Big Hero 6 are the best according to the scaled variables.

8. Conclusion

This report highlights and visualizes lot of metrics that are monitored surrounding film. After performing some data wrangling, processing, and feature engineering, the results give insights on which elements play a key part in a successful film.

Drama, Comedy, and Thriller are the most popular genres in the dataset. Minions and Wonder Woman are the top films respective to popularity and ratings. The Shawshank Redemption and Dilwale Dulhania Le Jayenga are the top films according to IMDB's ranking method. There exists a small correlation between vote count and vote average. A high vote count does not entail a good film. Inception and The Dark Knight have the most votes. Most movies get released in January, September, October while the most returns are made during the summer. Avatar and Star Wars: The Force Awakens both broke 2-billion-dollar mark for revenue.

Model performance can be improved with the addition of more features/variables such as figuring out the weekday based on the day of release. Classification can be executed to determine whether a film was a hit or not depending on relevant features. The hyperparameter tuned Random Forest was the best performing model with an R^2 of 13.3%. Revenue, runtime, popularity are influential predictors. Additionally, there were four recommender systems built with different ideas and algorithms. All these methods combined may be how production level engines are run. Future work involves deployment and a better approach for predicting on ratings instead of imputing with zeros.