Movie Rating Prediction

Process documentation

By: Vaishvik Chaudhari

Objective

Movie Rating

Main Objective: To predict the IMDB rating of a movie

- Get additional data from other sources if required.

- Perform Data Preprocessing and Exploratory Data Analysis which includes data visualization also.

- Create at least 3 different machine learning models to predict IMDB rating of a movie.

- Compare the results and suggest the model which could be useful to deploy into production.

- Optional: You can also use TensorFlow, Keras, or PyTorch to build the models.

Data Set

- movie_metadata.csv : Movies and the metadate about movies extracted from IMDB
- movie_budget.json : Movie and their budget & release date [External]

Data Preprocessing

Step 1: Joining External Dataset

To integrate additional data to enhance movie rating prediction.

- Join the main dataset with an external dataset containing worldwide gross income and release date information.

<u>Rational:</u>

- Worldwide Income: Domestic gross can differ from country to country. A global perspective of movie performance can provide more information in terms of profits.

- Release Date: While we have the release year, adding the month of release can provide insights into the seasonality of movie performance. Movies released during summer and winter months are often more anticipated, potentially leading to higher grossing movies.

Note: The integration of external data can be performed using appropriate data merging techniques such as joining on common identifiers like movie titles or unique identifiers like movie IDs.

Data Preprocessing

Step 2: Data Processing

Handling Missing Values:

- Drop missing values for the following columns:

- 'director_name', 'num_critic_for_reviews', 'duration', 'director_facebook_likes',
- 'actor_3_facebook_likes', 'actor_2_name', 'actor_1_facebook_likes', 'actor_1_name',
- 'actor_3_name', 'facenumber_in_poster', 'num_user_for_reviews', 'language', 'country',
- 'actor_2_facebook_likes', 'plot_keywords'

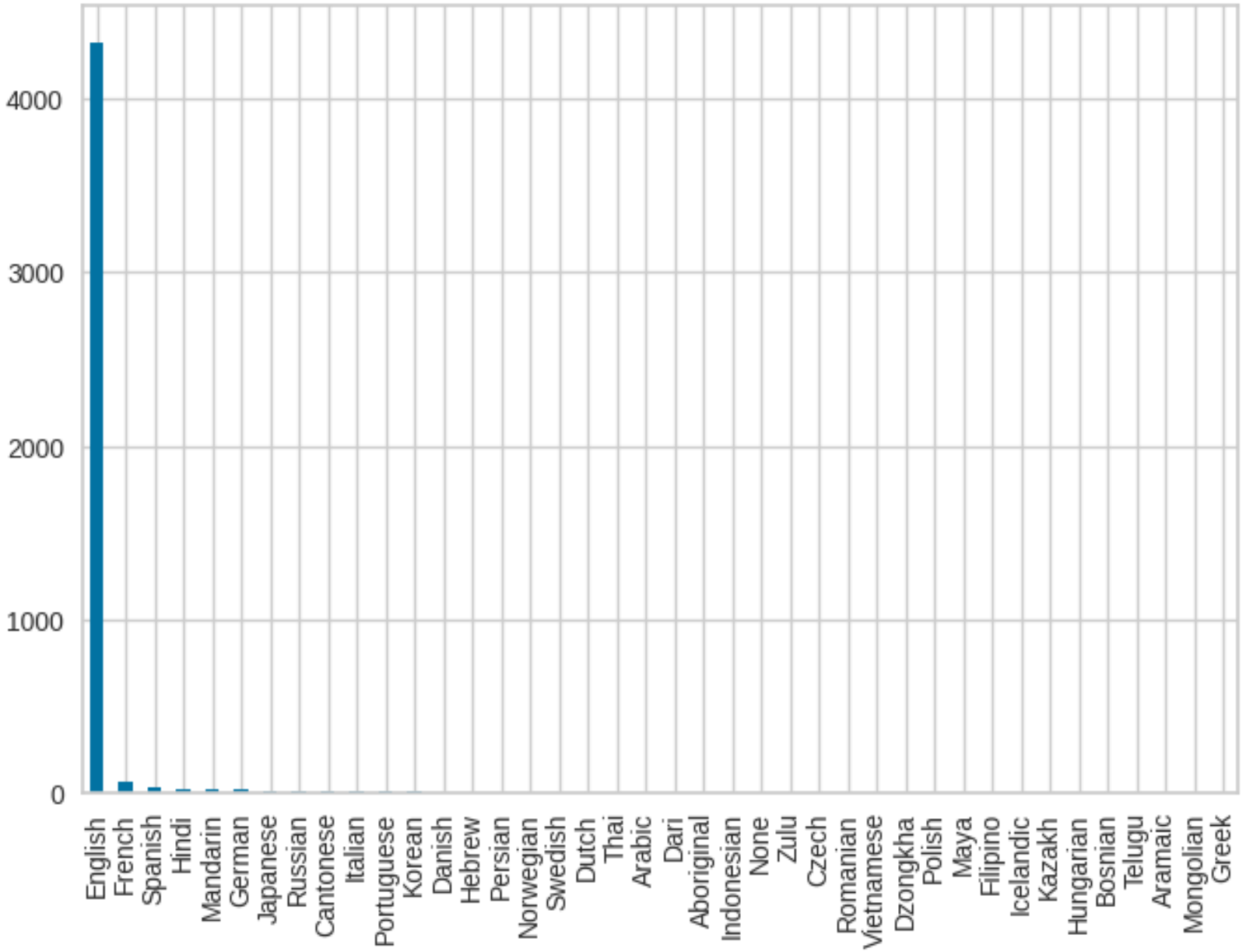
- Fill missing values using appropriate techniques for the following columns:

- 'content_rating', 'aspect_ratio', 'budget', 'gross'

Use the median or most frequent value to fill the missing values in these columns. The choice of technique depends on the specific data characteristics and context.

Note: Handling missing values is essential to ensure the integrity and completeness of the dataset, which is crucial for accurate model training and prediction.

EDA

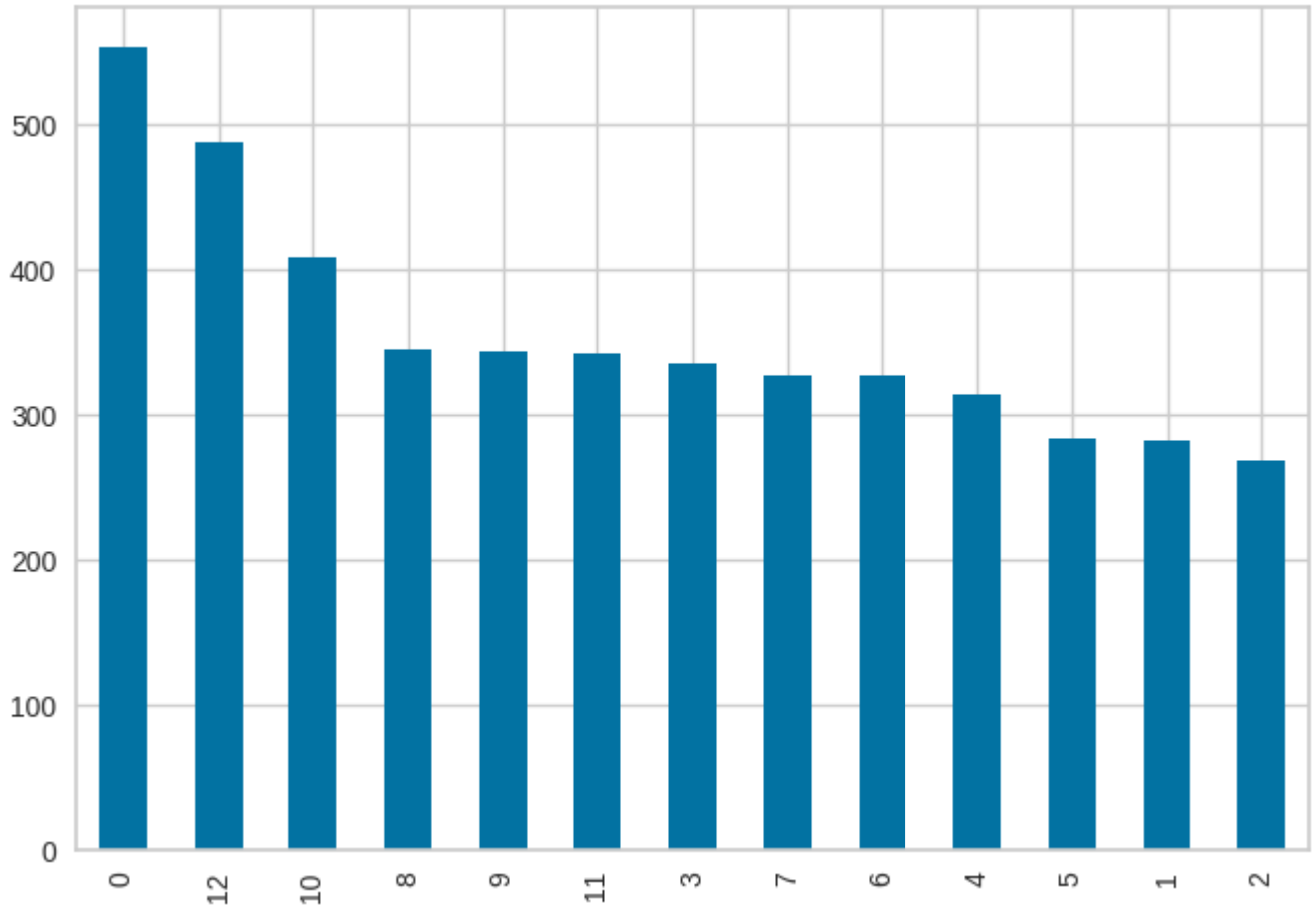


Movies language: English is most frequent with 94%. Removing language column as it don’t add value, give high skew.

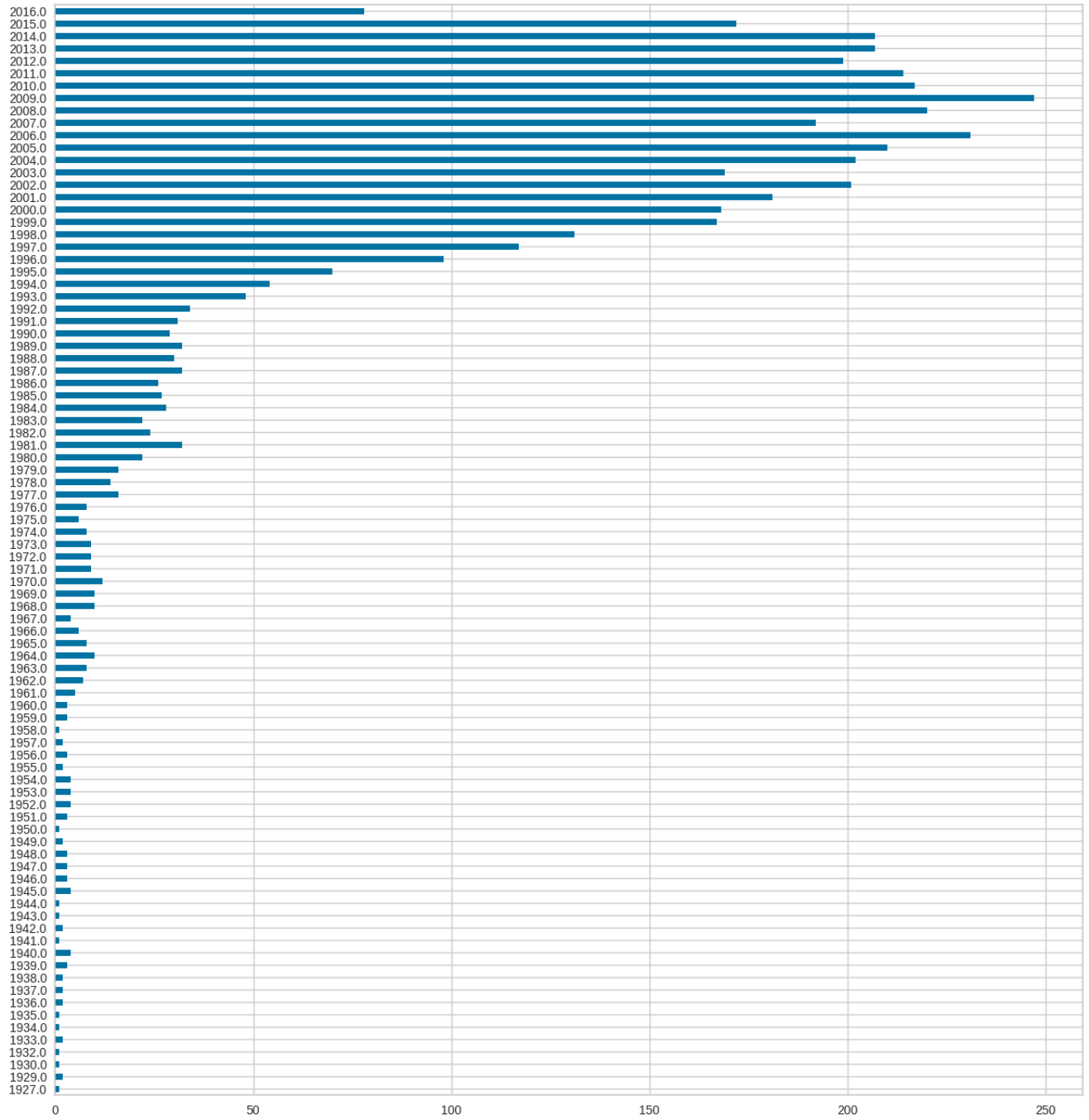
Movie Release Months

- Out of the available data, 87% of the movies have a recorded release month.
- For the 87% of movies with recorded release months, it is observed that December is the most favored month for movie releases, likely due to the holiday season.
- However, it is important to note that approximately 13% of the movies in the dataset do not have a recorded release month due to missing data.

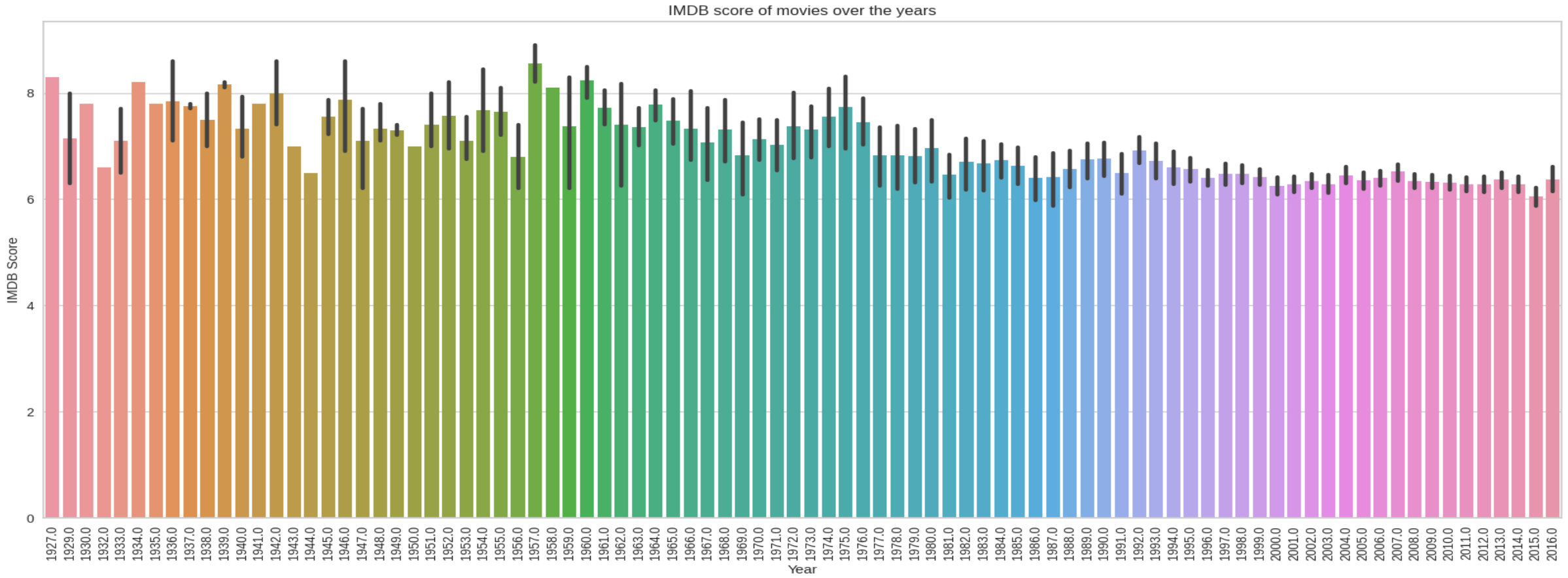
Note: Analyzing the distribution of movie release months can provide insights into the seasonality of movie releases and potential correlations between release month and movie performance.



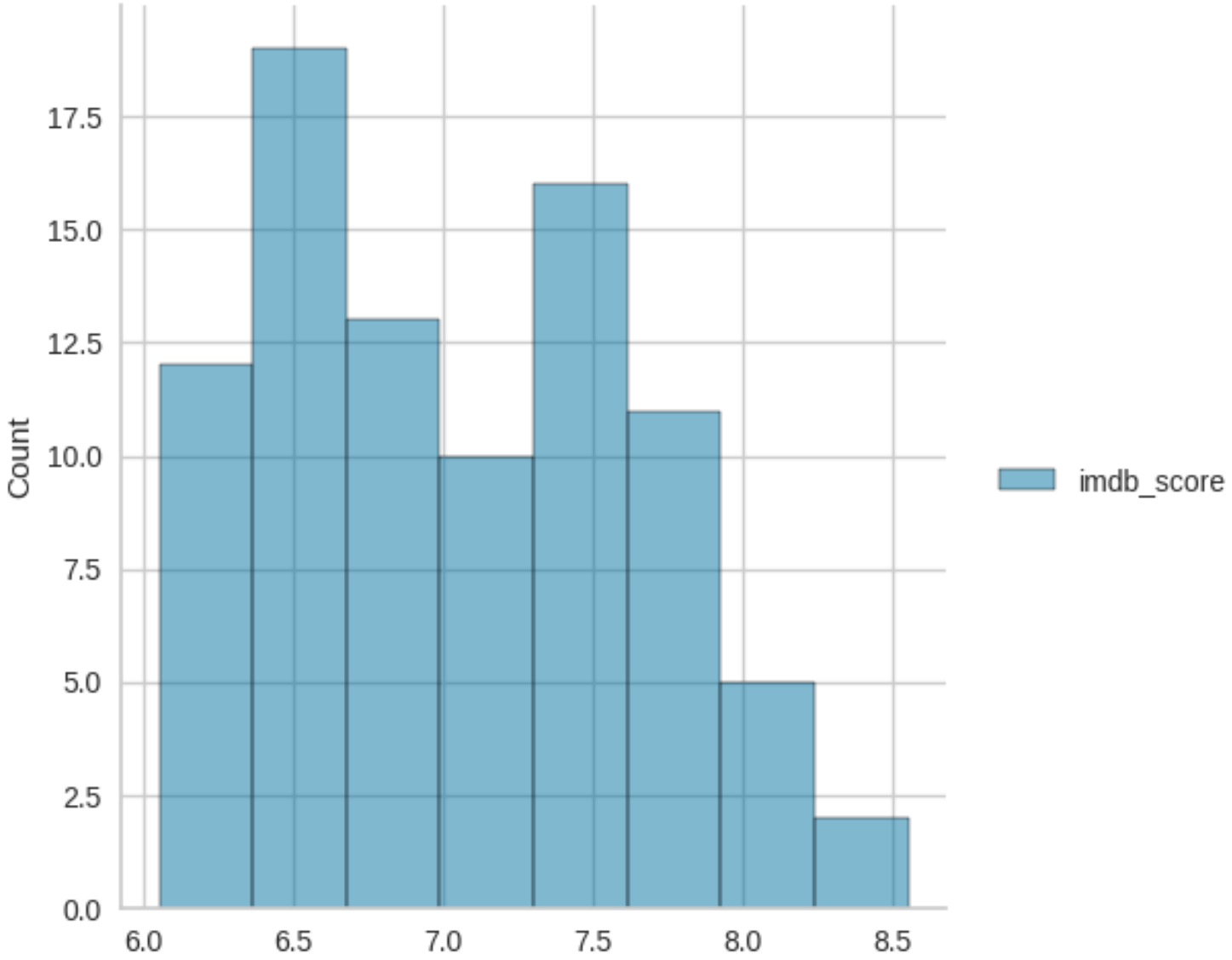
Number of movies releases by Year



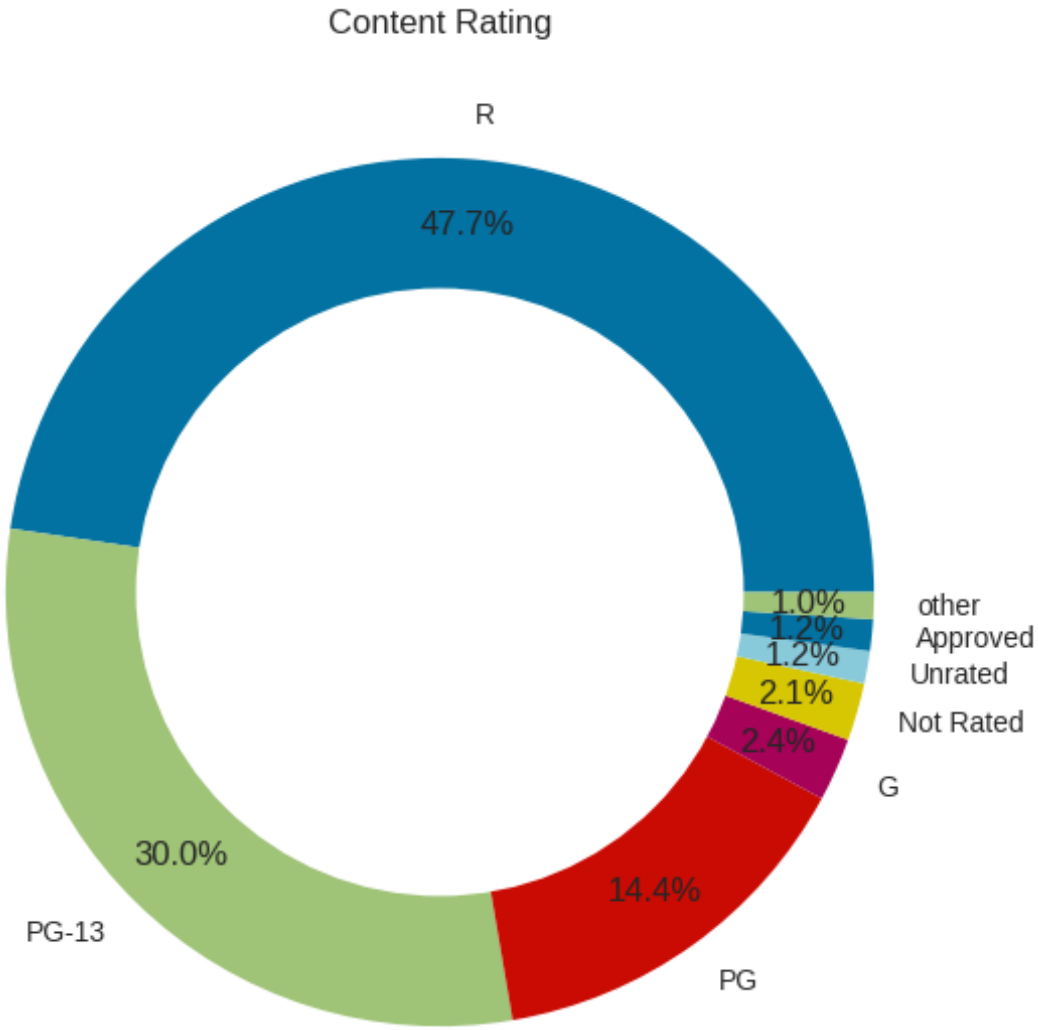
The range of IMDB scores of movies for every year
‘Old is gold trend’



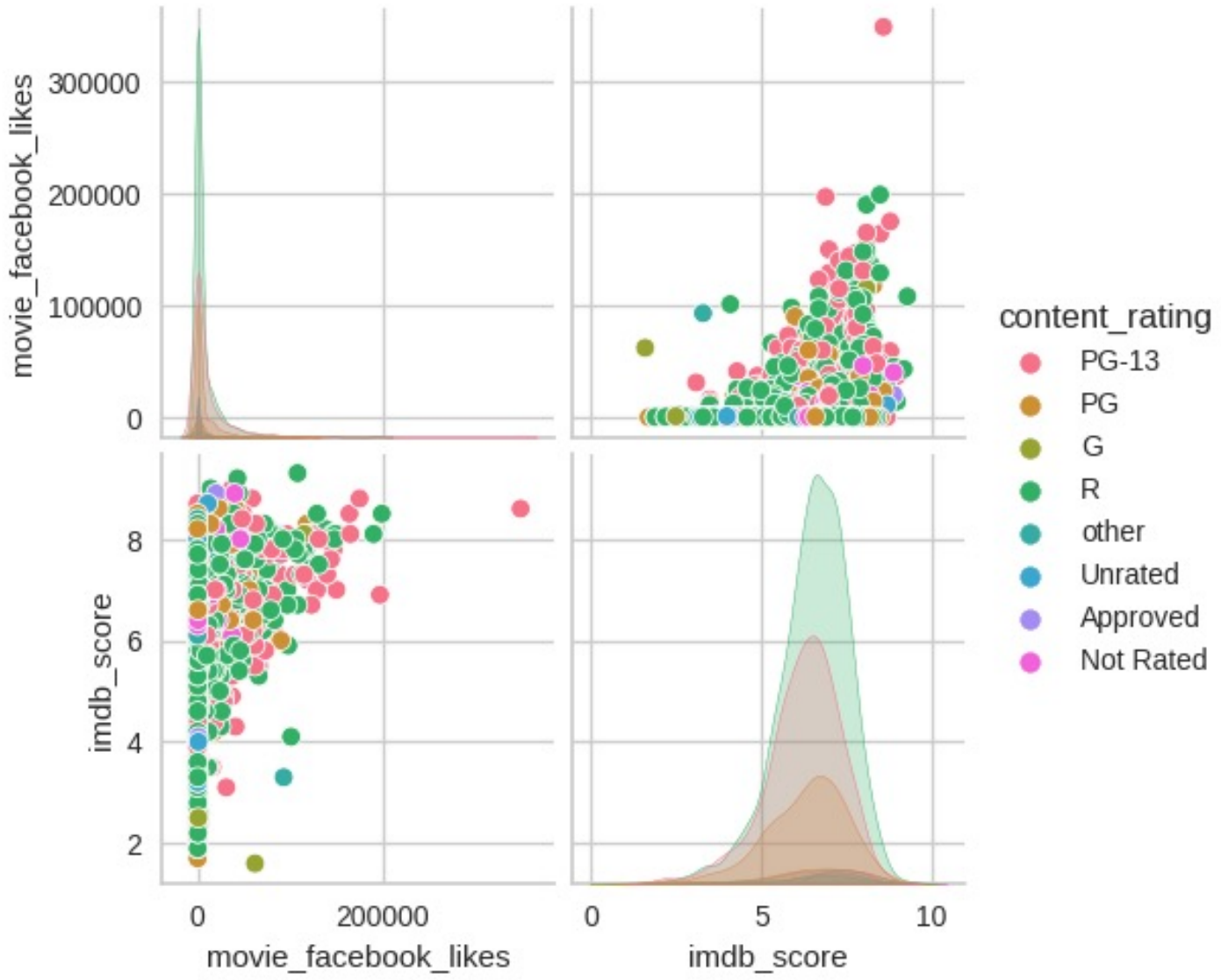
Calculate population mean from all the movies up to 2016 on imdb_score, Taking the mean of IMDB score or all the movies released in every year



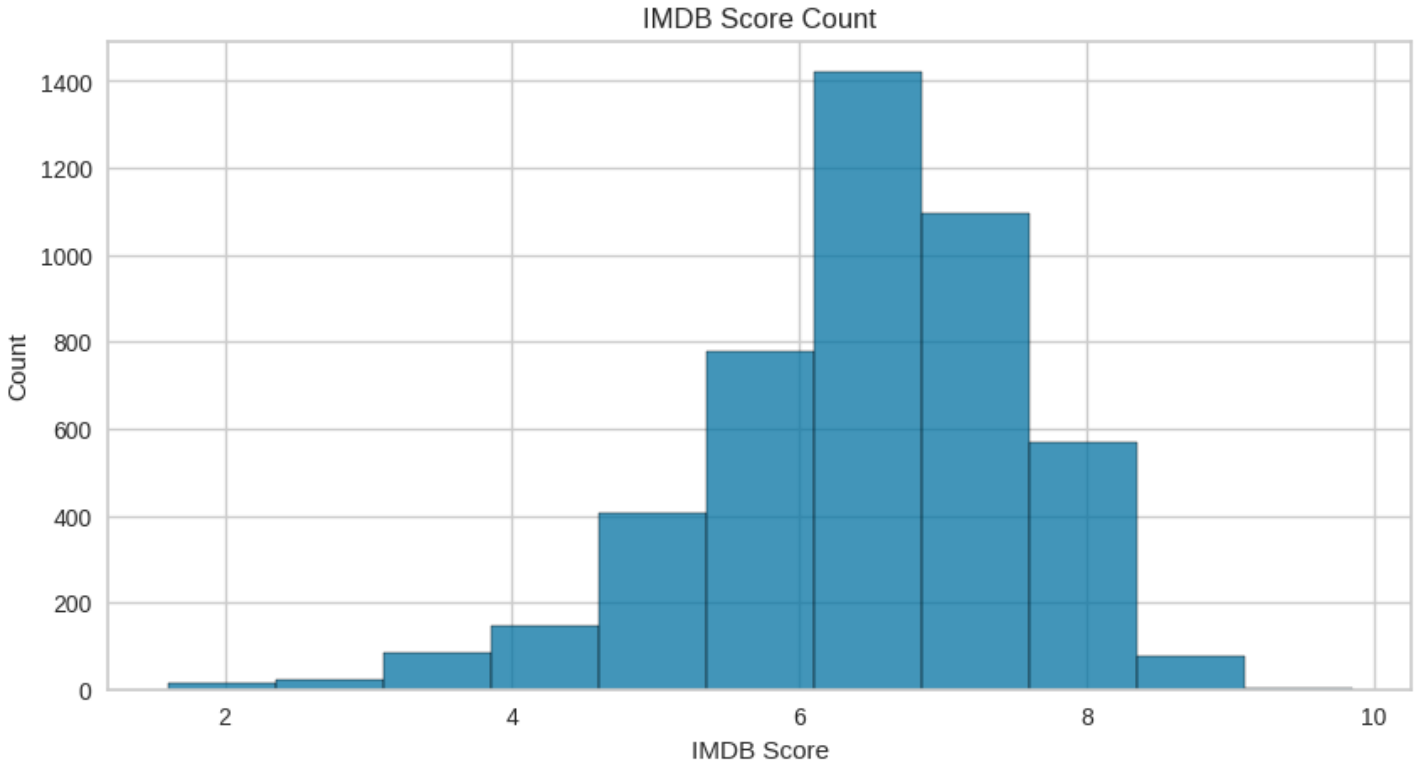
Movie content rating



Pairplot of 'movie facebook likes', 'gross', 'imdb score’ & 'content rating'



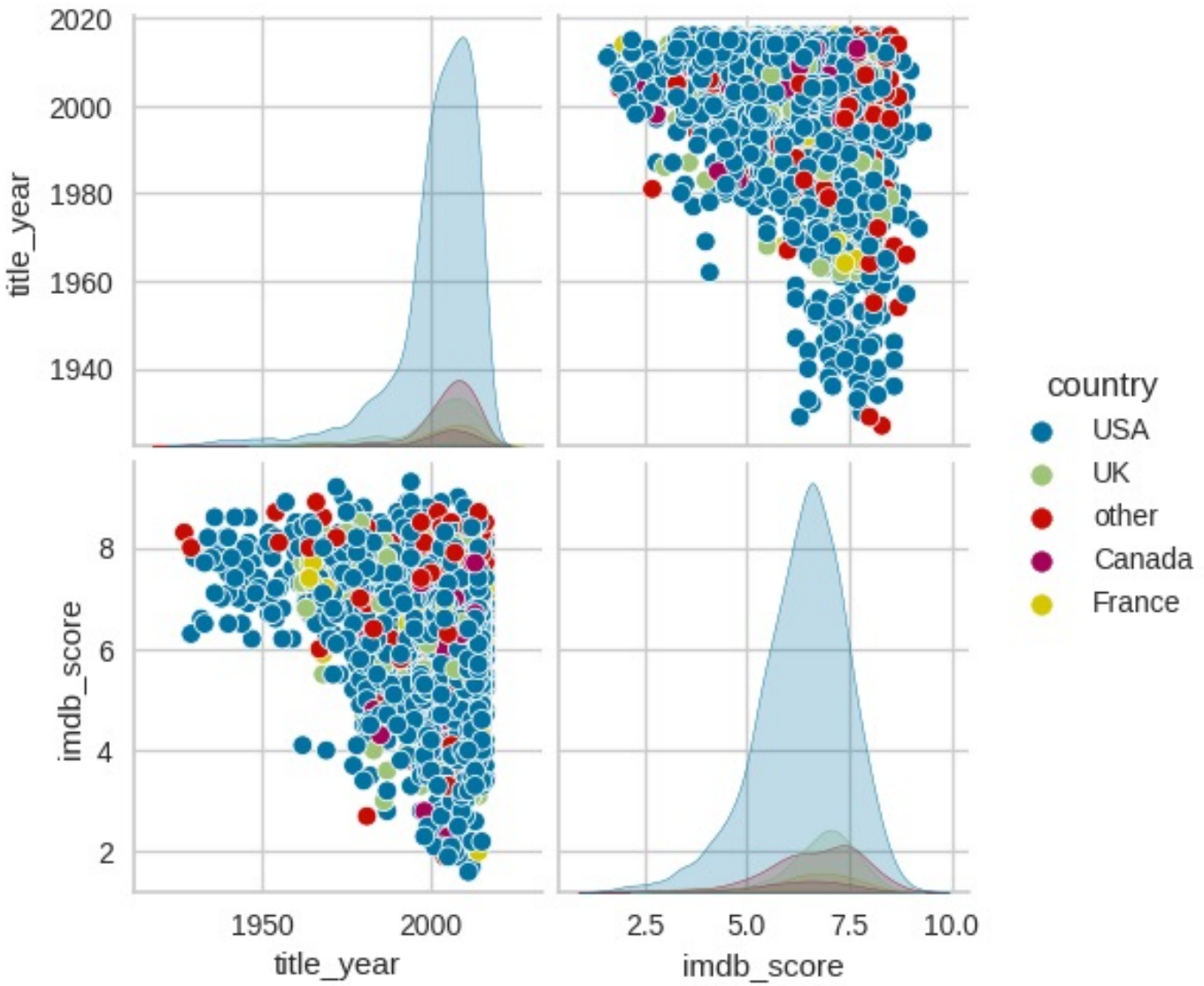
Count of movies by IMDB scores [Left skewed]:
We have mostly good movies ☺



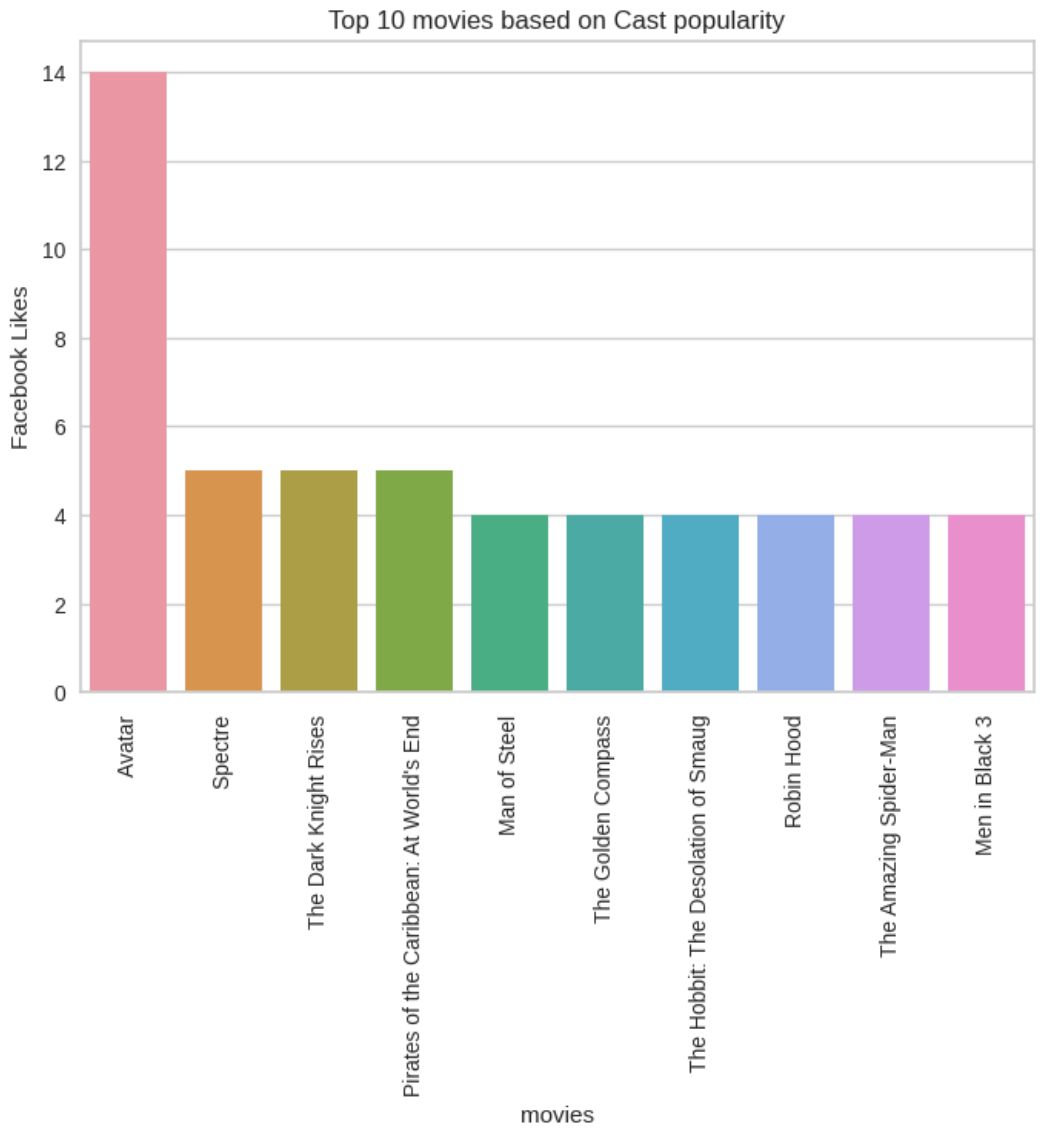
Features year of release and IMDB score with respect to countries in a pairplot to show how the movies fared based on their country of origin

From the above plot we can infer that the significant rise in the number of total movies made that was observed around year 2009 was infact due to the growth of American film industry.

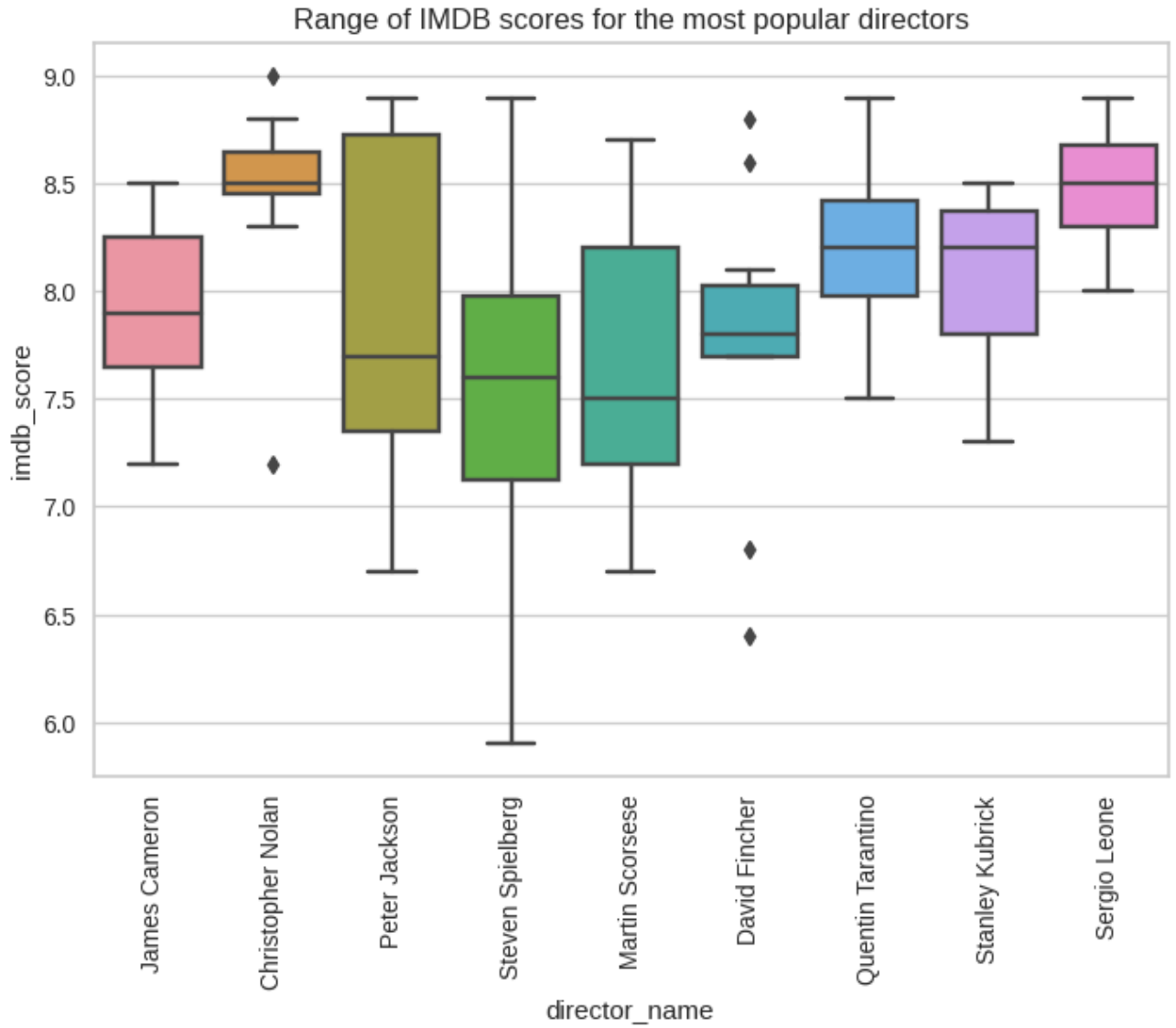
This can also be observed in IMDB score, the distribution for USA is narrow and also spread over much greater area as compared to all the other countries



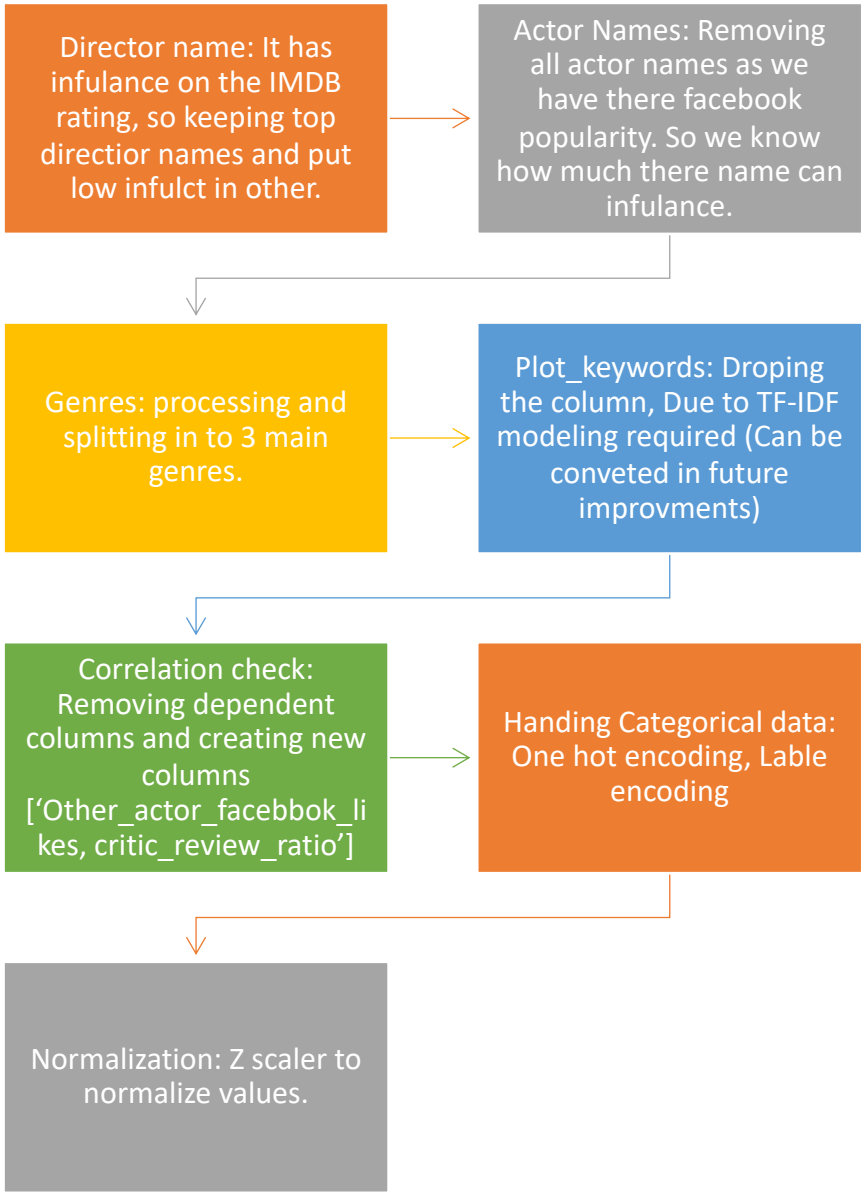
Top 10 movies based on Cast popularity



Range of IMDB scores for the most popular directors



Feature Engineering and data modeling

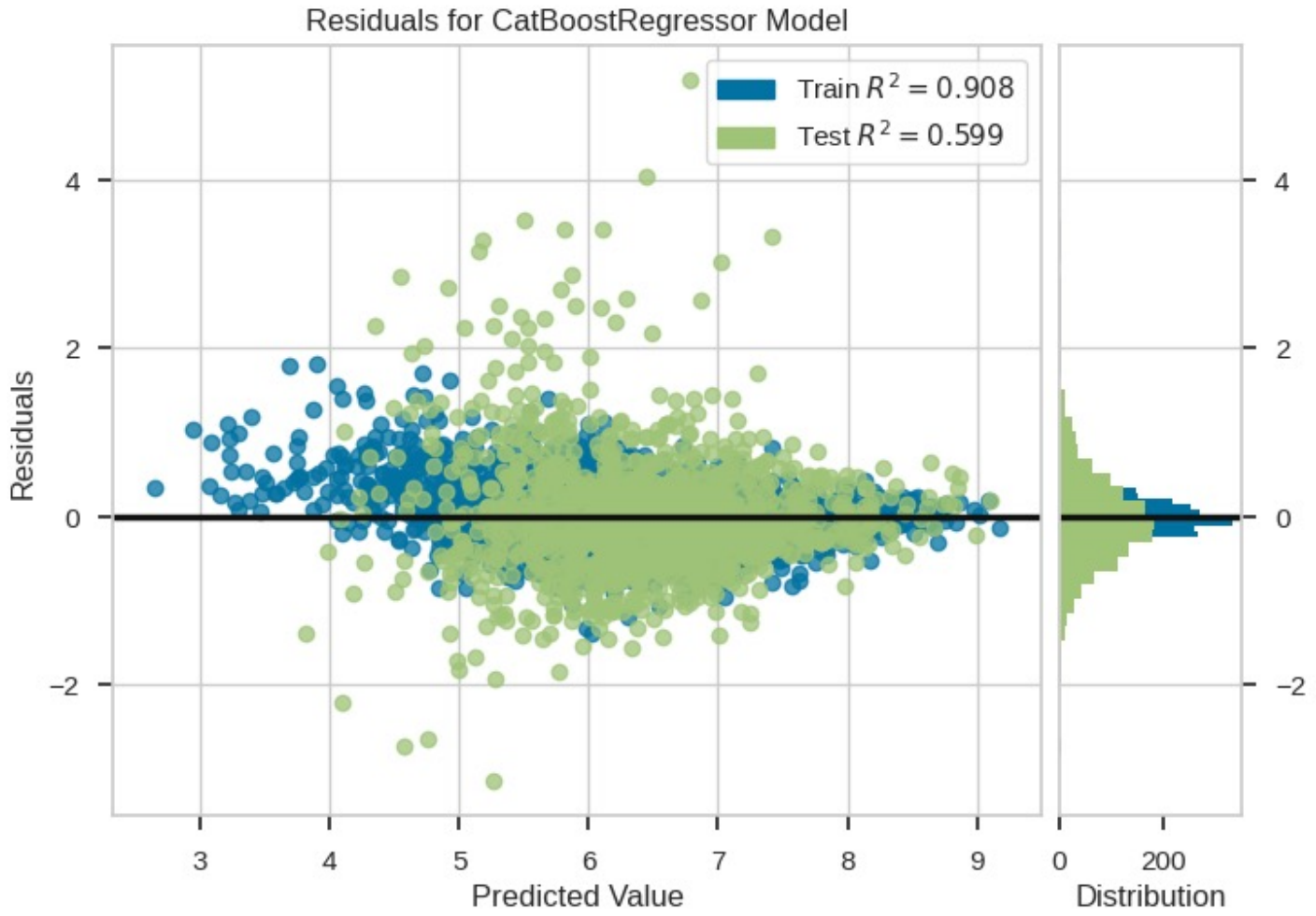


	Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)
catboost	CatBoost Regressor	0.4863	0.4672	0.6817	0.6057	0.1076	0.0878	0.2910
lightgbm	Light Gradient Boosting Machine	0.4959	0.4771	0.6891	0.5973	0.1085	0.0895	0.3680
gbr	Gradient Boosting Regressor	0.5186	0.5208	0.7198	0.5611	0.1133	0.0942	0.2450
et	Extra Trees Regressor	0.5344	0.5414	0.7341	0.5435	0.1150	0.0964	0.4620
xgboost	Extreme Gradient Boosting	0.5378	0.5508	0.7408	0.5331	0.1154	0.0955	0.1820
rf	Random Forest Regressor	0.5441	0.5548	0.7437	0.5310	0.1162	0.0980	0.2010
br	Bayesian Ridge	0.6629	0.7804	0.8818	0.3412	0.1362	0.1189	0.2710
lr	Linear Regression	0.6632	0.7809	0.8821	0.3407	0.1361	0.1189	0.8780
ridge	Ridge Regression	0.6632	0.7809	0.8821	0.3407	0.1361	0.1189	0.0890
ada	AdaBoost Regressor	0.7312	0.8224	0.9062	0.3006	0.1336	0.1215	0.1320
knn	K Neighbors Regressor	0.7106	0.8774	0.9352	0.2590	0.1413	0.1249	0.1970
omp	Orthogonal Matching Pursuit	0.7291	0.8987	0.9460	0.2436	0.1431	0.1300	0.1500
lar	Least Angle Regression	0.6953	0.9045	0.9432	0.2361	0.1421	0.1235	0.0990
huber	Huber Regressor	0.6614	0.9281	0.9444	0.2067	0.1360	0.1206	0.1510
dt	Decision Tree Regressor	0.7869	1.1832	1.0869	-0.0063	0.1719	0.1388	0.2590
lasso	Lasso Regression	0.8493	1.1955	1.0920	-0.0082	0.1621	0.1510	0.0890
en	Elastic Net	0.8493	1.1955	1.0920	-0.0082	0.1621	0.1510	0.0900
llar	Lasso Least Angle Regression	0.8493	1.1955	1.0920	-0.0082	0.1621	0.1510	0.0980
dummy	Dummy Regressor	0.8493	1.1955	1.0920	-0.0082	0.1621	0.1510	0.1360
par	Passive Aggressive Regressor	1.0065	5.3272	1.7845	-3.7996	0.1915	0.1710	0.1550

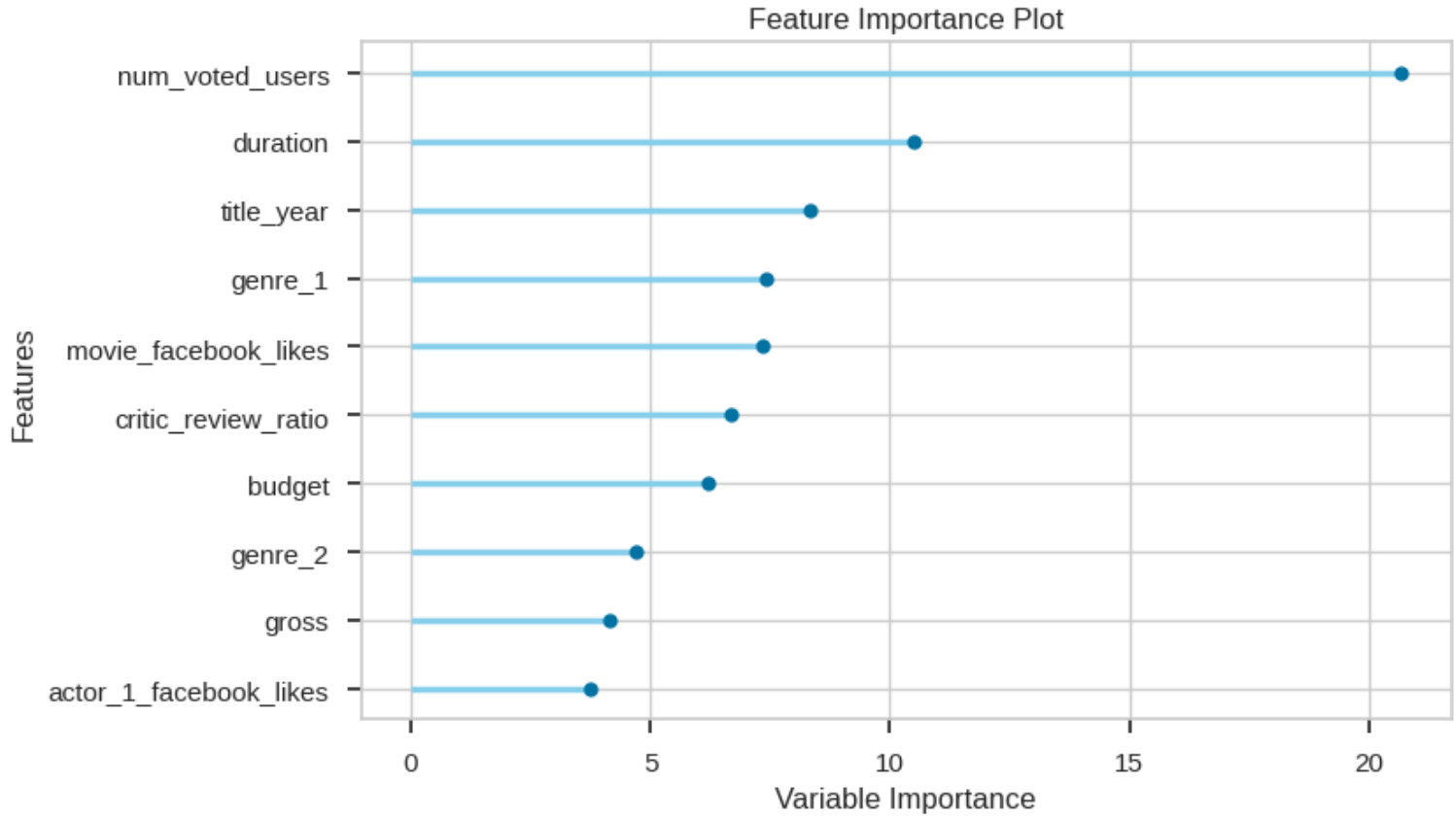
20 Base Model Comparison

- As we can see CatBoost has the best R2 and MAPE score.
- Overall boosting models are in lead like Xboost and LightGBM.

Best model residual check



Feature Importance for CatBoost Model



Future Work

Use TF-IDF to improve model, by modeling plot_keywords Column.

Hyperparameter tuning for more models

Ensemble modeling (Bagging/Boosting)

Conclusion:

Model Evaluation and Selection

- The best-performing model for the given dataset and feature engineering is Catboost.
- Due to the size of the dataset, the scope of using deep learning models is limited, as they typically require a large amount of data to achieve optimal performance.
- Catboost, a gradient boosting algorithm, has demonstrated superior performance on this dataset, considering the feature engineering performed.
- Catboost provides excellent results even with smaller datasets, making it a suitable choice for movie rating prediction in this scenario.

Note: The selection of Catboost as the best model is based on its performance and compatibility with the dataset characteristics.

Model	MAE	MSE	RMSE	R2	RMSLE	MAPE	TT (Sec)	
catboost	CatBoost Regressor	0.4863	0.4672	0.6817	0.6057	0.1076	0.0878	0.2910