Pre-Release Profitability: Predicting Revenue and Ratings for Hollywood Films

Amy Chitnumsab (ac2295), Tejhan Diallo (td382), & Jennifer Kim (jjk358)

Cornell University

ORIE 4741

05.12.2023

Table of Contents

Introduction

Our project aims to predict the success of movies before their release, both in terms of commercial and public reception success. We use the relevant features of this dataset to 1) predict box office revenue and/or ticket sales as a measure of commercial success and 2) IMDb ratings, a potential measure of public reception and success, which could lean favorable or not. (IMDb ratings are user ratings, weighted to take into account the number of votes and reviews from the general public and not film critics).

Problem Background

Given the profitability and prevalence of the movie and entertainment industries, it is of relevance to predict how well a movie might perform in the box office given past data. More specifically, it is of interest for movies to maximize revenue (and in exchange, profitability) as well as receiving a good reception from the public. Thus, the questions analyzed by this project may be of importance to film distributors and studios, who might have financial interest in how well a film performs at the box office given the necessity of film financing and investing.

Predicting tickets sold for a movie is still a valid metric for profitability and film success. Concerns that the movie theater industry is dying due to factors like the pandemic and rise of disruptors like Netflix exist, but the future of movie theaters is still bright and evolving. Of course, other factors like continued marketability of films from non-box office revenue sources might also factor in, but predicting movies' box-office performance (and if possible, public reception) would be a reliable tool for studio decision makers and movie investors.

The Approach

We use a machine learning approach to tackle the questions posed. The models used in the project are two-fold. First, in order to gauge critical success in the eyes of the public, we decided to classify movies based on whether they would receive a favorable reception (rating) from the public or not. This "positive/negative" classification can be viewed as whether an IMDb rating (provided in the dataset as "vote_average") for a movie was described as greater than or equal to a 7 or not. (Considering that the 75% percentile is a 6.8 rating). Second, for the issue of predicting commercial success, a linear regression model was modeled to estimate projected revenue in 2023 USD.

Both models carefully selected features that would reflect predictions made with information about the movie that would be available to interested parties (e.g. production company) *before* release to theaters. Thus, the techniques used throughout the project are as follows: classification methods (logistic regression), regression (linear regression and least squares), feature engineering (data cleaning and transformation like frequency encoding), and train/test methods for error and model effectiveness analysis.

The Dataset

The dataset used will be The Movies Dataset, which has data compiled from the IMDb (Internet Movie Database) and TMDB (The Movies Database) websites. This dataset has information on 45,000 movies such as budget, revenue, release date, genre, and production company. We combined the initial dataset contained in the csv file "movies_metadata.csv" with additional files called "credits.csv" and "keywords.csv" to get a final file containing the full movies dataset to be used.

We summarize some important columns (features and labels) below, along with information such as the initial type of data the information was formatted in.

Feature Name	Relevant Information
id	Numerical, ensures that

	movies are identifiable across files
Movie Title	String, title of the movie
Overview	String, description of the movie plot
Budget	Numerical, in USD from year of movie release
Runtime	Numerical, in minutes
Genres	Categorical, String. More than one genre type for a movie allowed (e.g. both "Fantasy" and "Action")
Production Companies	String. Many movies have more than one production company.
Cast	String with sections for all cast members, each cast member has curly brackets with entries like ‘character’ and ‘gender.’ Cast for the most important characters are listed first.
Crew	String with multiple sections for all crew members. Each crew member has curly brackets with entries like ‘department,’ ‘name,‘ and ‘gender.’ Department has roles like “Director.”
Release Date	String in the date form “Year-Month-Day.”
Revenue	Numerical, in USD of year of movie release
Vote Count	Numerical, refers to how many total votes of rating a movie got on IMDb
Vote Average	Numerical, the mean IMDb rating on a scale of 1 to 10 w/ 10 being best

Feature Engineering

Data Preparation

After merging to get an initial file, the dataset was further prepared for models by filtering, handling missing data, and adjusting values as needed. The initial number of rows/movies was 45432 with 27 columns. Movies that were not produced from the United States were excluded as the focus and purpose of this project was to analyze Hollywood movies.

Missing and Duplicate Values

All movie entries with missing revenue and budget values were identified and removed. Imputation for missing numeric values like budget and runtime was not considered since imputation was not appropriate in this scenario. The same applied for categorical data such as production company and genres. Data cleaning was also performed with the removal of duplicate rows from the original dataset. Finally, outliers were removed from the dataset for their unusual budget/revenue ratio and/or evidently incorrect budgets (i.e. $1).

Dealing with Strings

Analyzing information for classification and regression necessitated data extraction from strings. Various methods like json , regex, etc. were explored but literal_eval() from the Abstract Syntax Tree module was best at extracting the necessary information for features like cast, crew, genres, and production company. Furthermore, strings for the release date were dealt with by transforming them into datetime objects and creating new columns for the Year and Month.

Trimming

Additionally, features with several dozen entries within itself were shortened and trimmed to only leave the first three entries for better data processing. The cast and crew columns were modified to isolate for the first three people in each feature. Top billing actors and actresses were considered by importance and isolated in a newly generated “cast_short_list” column. For example, the row for *Spider-Man* included “cast_short_list” with names Tobey Maguire, Willem Dafoe, and Kirsten Dunst.

Adjusting for Inflation

Because the budget and revenue columns were given in USD of the year a movie was released, they were adjusted for inflation. In order to do so, separate datasets were created using information from CPI (consumer price index) data and average movie ticket prices for each year (references later in Appendix more information in Github code and csv files). Budget and revenue adjustments were treated differently as follows.

First, budget was adjusted based on the CPI data for each year. Missing CPI values for earlier years until the 1970s were imputed using the last available year. Second, revenue was adjusted differently than budget because revenue was assumed to be only dependent on ticket prices, whereas budget encompasses several components for props, costumes, etc. that are not as specifically separable. Thus, average ticket price data from 1910 onwards was collected, with missing data imputed as necessary like for budget. Afterwards, through matching by year, the tickets sold for each movie was estimated and multiplied by the average price of a movie ticket in 2023 in order to generate the estimated revenue the movie would have made this year. Through this way, all revenue and budget was standardized in 2023 USD.

Overall

After these steps, the final dataset generated had 4513 entries/movies, reflecting the vast magnitude of data available for analysis. Even with 5 features per movie entry, the total data points would total over 20,000. Additionally, after removing the top 10% of movies by gross revenue from the model data (in order to remove both outliers and after adjusting for a high skew as explained in the next section), the total unique movies numbered 4023.

Exploratory Data Analysis

Main Stats

The following mentions Figures which are found in the Appendix section. Figure 1 summarizes the distribution of our numerical data such as run time, IMDb rating as vote_average, inflation-adjusted budget, and revenue. Figure 2 displays the number of movies released each year, which shows that the majority of movies in this dataset were released in the past decade. Figure 3 shows the number of movies released within each month, indicating that more movies are released during the second half of the year. Figure 4 shows the distribution of movies across different genres. Figure 8 contains the top 10 production companies that made the most movies in the dataset. Figure 9 shows the top 10 actors who starred in the most movies. Likewise, Figure 10 shows the crew members that worked on the most movies. Figures 8, 9, and 10 were ordered based on the number of movies because having larger sample sizes to study specific variables can more accurately represent generalized data.

Outliers

Most outliers that were removed from the original dataset either had issues with their budget or revenue data. Movies with missing budget or revenue were removed. To limit the dataset to lower- and mid-budget movies, which are the most common movies produced, entries with budget or revenue lower than $1000 or in the top 10% of the range were filtered out. Doing this also gets rid of anomalies such as movies with extremely low budgets but large revenues or movies with record-breaking earnings. To account for entries with ratings based on a small sample of people, movies with a vote count of less than 20 were also removed.

Visualizations

After cleaning the dataset, visualizations were made to explore possible correlations between variables. Doing so helps with making decisions on the feature space for predicting revenue and IMDb ratings. Figure 5 is a correlation heatmap showing the correlation between different numerical values. The strongest positive correlations that revenue has with other variables are budget, ratings (vote_average), and runtime. The correlation values were about 0.25-0.60. Ratings have a positive 0.34 correlation with runtime and a weak positive 0.11 correlation with month. A surprising result in this figure is that ratings have a slightly negative -0.05 to -0.10 correlation with budget and year released.

Figure 6 is a multi-bar chart showing the proportion of movies with good and bad ratings grouped by genre. This plot indicates that the genres with the largest proportion of highly rated movies are the ones that have a lower number of movies under those genres. 40-50% of Documentary and Western movies are highly rated, while roughly 10% of Horror and Family movies are highly rated. But based on figure 4, there are less than 100 Documentary and Western movies and more than 400 Horror and Family movies in the dataset. So results may be skewed by sample size.

Figure 7 is a bar chart showing the average revenue of movies by genre. Similar to results from figure 6, this chart illustrates that genres with less movies have higher average revenue than genres with more movies. The genres with highest revenues such as Animation have less than 100 movies, while genres with lower revenues such as Drama have almost 2000 movies.

Looking at the top 10 production companies in Figure 8, only approximately 20% of their movies are highly rated. Average revenue of each company listed is roughly $150 million. These findings suggest that the most well-known production companies such as Warner Bros. and Universal Pictures may create the most movies, but a large portion of them do not necessarily have good ratings or earn the most revenue.

As for the top 10 cast members in Figure 9, the average revenue of movies that these actors starred in range from about $100-$200 million. The only exception is Julianne Moore whose movies have an average revenue of about $90 million. However, there is no clear trend in the proportion of highly rated movies that these actors starred in. So having a famous or experienced cast may not always have a strong correlation with higher ratings.

Figure 10 shows that the average revenue of movies that the top 10 crew members worked on is between $90-$180 million, which is slightly lower than the average revenues corresponding to the top 10 cast members. So the crew may have a weaker correlation with revenue. But the proportion of highly rated movies also fluctuates across these people, meaning that there may not be a clear correlation between crew and ratings.

Revenue Prediction

Linear Regression

The linear regression model was chosen to attempt to predict the revenue of an unreleased movie. The statsmodels package, specifically the OLS linear model functions, was used to generate the models and results.

Real Value Data

Our first predictor that served as a baseline for future iterations was the inflation adjusted budget. Naturally, movies with higher budgets would see higher revenue, and this correlation was clear from the start. Another predictor we used was runtime (The length of the movie). We didn’t prioritize other numerical values such as release year or date, as they were shown to be extremely weak predictors relative to other features.

Transforming Categorical Data

Many-hot encoding

Many-hot encoding was used to transform the genres column into a usable feature for the models. This method was implemented using a simple nested for loop that ran through all entries in the dataset and created a list of 1s and 0s for each entry.

Frequency Encoding

Due to the format of the production companies, cast, and crew columns, movies can fall into multiple categories for each of these variables. But using many-hot encoding would create lists that extend runtime by a lot. Instead, Frequency encoding was used to transform them into usable features. First, the fraction of movies that fall into each category of each variable was calculated. Then for every movie, the total frequency score was calculated by summing up the fractions corresponding to the categories that the movie falls into. Thus, the resulting features are singular numerical values for each entry.

Embedding the Movie Title and Overview

To handle description-based columns such as movie title and overview, the Universal Sentence Encoder introduced in Homework 3 was used to transform the text. The final features are numerical representations of the text in the form of arrays with 512 numbers.

We used the sentence encoder to transform and analyze data from the “description”, “overview”, and “title” columns.

Fit and Regularization

For each iteration of the model, we recorded the Mean Standard Error (MSE) of the model’s performance on the train dataset and the test dataset. This allowed us to easily record instances of overfitting and weak predictors in our models. Some features, primarily categorical & text based ones, made our model quite prone to overfitting. To mitigate this, we utilized L1 & L2 regularization when applicable.

Rating Classification

Logistic Regression

The logistic regression model was chosen to attempt to predict whether an unreleased movie will have a good or bad IMDb rating. The scikit-learn package containing built-in functions for logistic regression was utilized to generate the results. Both models were trained and tested on a 80:20 train-test dataset, which was created using train-test-split from statsmodels. Feature transformation done for the linear regression models were also used in the logistic regression models.

Regularization

The scikit-learn functions incorporate regularization by default and allow for changing the type of regularization. Thus, 2 model versions were made by varying the regularization type once features have been selected. One model uses L1 regularization and another uses L2 regularization.

Feature and Model Selection

Model Statistics

To determine the feature space of the models, some basic model statistics were generated. These include feature coefficients and their corresponding p-values. Features were chosen based on how statistically significant they were in the model, meaning that their p-values must be smaller than 0.05 for a 95% significance level. Due to the regularization type affecting the features’ p-values, an optimal model was also selected partly based on how small the p-values were.

Accuracy Score

In addition to using the p-values to select an optimal model, a scikit-learn built-in function that calculates a model’s accuracy score was utilized as well. This accuracy score is calculated by finding the fraction of predictions that were correct from all of the predictions made. A better model would have a higher accuracy score.

Confusion Matrix

A third method of checking the accuracy of the models is generating a confusion matrix. This matrix shows the number of correct predictions made for good and bad ratings, the number of false positives, and the number of false negatives. In this scenario, a false positive is a movie that was predicted to have good ratings but truly has bad ratings. A false negative is a movie that was predicted to have bad ratings, but actually has good ratings. A better model would have a higher number of correct predictions and a minimal number of false positives and false negatives.

Results and Discussion

Revenue Prediction

Base Model

The base model predicted revenue with runtime and budget alone, and it was improved with new features each iteration.

Budget unsurprisingly proved to be a reasonable predictor of revenue by itself, though this trend does not necessarily hold for critics' approval. When predicting revenue with budget & runtime alone in our 80:20 train-test data split, we consistently recorded a train MSE of around 7-8k from multiple samples, with the test MSE usually being around 3-400 more than the train MSE. (Fig 11)

Genres were transformed into one-hot encoded vectors, and when added as a predictor into our model, it reduced both the test and the train MSE by around 5% on average. Another predictor that uniformly strengthened the model was including the 3 most prominent cast members as predictors in the form of a frequency encoded 1x3 vector. The addition of these parameters saw our train and test MSE both decrease by an average of 4%. (Fig 12)

When using text embedding for features such as “overview” and “title”, we encountered heavy overfitting of the data, with the test MSE usually being twice as large as the train MSE. (Fig 13) To combat this, we used L1 regularization and tuned the regularization parameters to improve train and test MSE simultaneously. L2 regularization did not work well with our model, presumably due to the large amount of individually weak parameters introduced with the embeddings (512), and many 0-value parameters introduced with many-hot encoding.

Regularized Models

When using L1 Regularization, we concluded that an alpha value around 0.1 served as the best parameter. When using textual embedding on “title”, we were able to decrease train MSE by approx. 17% and test MSE by approx. 7%. This massive improvement wasn’t seen with the “overview” embedding however, as it only decreased train and test MSE by less than 1% at best when regularized.

Final Model

With all features considered and properly regularized, we were able to come up with a final model that predicted the revenue with a MSE of 5796 and a mean absolute error of 6655, which corresponds to mean absolute errors of $76 and $86 million, respectively (Fig 14). This is a 24% and 16% improvement from the base model in train and test MSE, respectively. When you take into consideration that most movies in our dataset make in the hundred million range, this does offer significant predictive value in a real world scenario.

Rating Classification

Both logistic regression models used to predict IMDb ratings had a feature space containing genres, budget, runtime, and year. When input as individual features, these variables had the smallest p-values compared to other potential features. However, some overfitting may have occurred once the variables were combined into one feature space. When all of the features were combined, the resulting p-values for genre, runtime, and year spiked up to 0.99 or 1. The only feature that remained statistically significant with a p-value less than 0.05 was budget. As a result, budget is the stronger indicator of ratings in these models.

L1 vs. L2 Regularization

To determine whether using L1 or L2 regularization yielded better results, the p-values, accuracy score, and confusion matrix of both models were investigated. Based on Figures 15-18, both models had little to no difference in their accuracy score of 80.46% on the training set and 79.23% on the test set. Their confusion matrices were the same as well. On the training set, 2417 movies had correct predictions of bad ratings and 587 movies were false negatives. On the test set, 595 movies had correct predictions of bad ratings and 156 were false negatives.

The most noticeable difference between the 2 model results were their p-values in Figures 17-18. Compared to L2 regularization, L1 regularization had larger effects on penalizing the features that had weaker correlations with ratings. The results from L2 regularization had slightly more moderate p-values. The majority of the values were not completely penalized to 1.0, but all values were above the 0.05 threshold. On the other hand, using L1 regularization led to penalizing all of the genre to 1.0 and budget being the only statistically significant feature left. Therefore, using L1 regularization was slightly better than using L2 regularization because the variables having stronger correlations with ratings and more statistical significance can be clearly seen.

Conclusion

We have found that when it comes to monetary success of movies, the crew members serve to be one of the biggest predictors of high revenue. The model tells us that a combination of successful & reputable writers, directors, and producers has a stronger effect on potential success than the Production company and crew members, whose effects are still significant, but probably not as consistent.

Contrary to our original predictions, release date was not necessarily a strong predictor for monetary success, and neither was runtime.

As for the logistic regression models, it can only be concluded that budget may be a strong indicator of IMDb ratings of an unreleased movie. All other variables considered such as runtime, year, and genre were not great indicators of ratings perhaps due to the lack of data on movies with good ratings. So patterns in ratings based on these variables may not have been as clear for the models to detect.

Future Improvements

Because our error in the linear regression was more substantial than we would have preferred, it is of note that our models require further refinement and tuning. One of the biggest areas of inconsistency was the overfitting that occurred when using manyhot encoding for production companies, cast, and crew, and also when using textual encoding for descriptions. These parameters turned out to be great for improving the model in the train set, but almost always became weaker when generalized to train data. Shrinking the size of the encoding, and using less features for the textual embedding also didn’t mitigate this overfitting issue. We hypothesize that this issue might be caused by the fact that unique company names, crew members, and descriptive language in the train dataset isn’t universal, which means that many of the predictors that we utilized in the training of the model won’t be compatible for a smaller, possibly more niche set of generalized data, hence the overfitted results.

There were also multiple limitations with the logistic regression models that require improvements. Because the p-values of many features seem to be statistically insignificant, more data should be collected to investigate if these results are accurate. Especially as the dataset contains mostly movies with bad ratings, having a more even distribution of highly and low rated movies would help to improve results. Furthermore, the ways in which our categorical data was transformed may not be the best methods to highlight the relationships between them and ratings. More types of feature transformations on the categorical variables should be explored.

Some concepts we could consider for the future would be non-linear models such as decision trees, and random forests. Another potentially useful tactic would be to do sentiment analysis on the keywords of the movie. These improvements require more time and study, but will be worthwhile.

Fairness & Weapons of Math Destruction

Our models aim to predict revenue and public ratings before a movie is released. Variations of our model may affect issues of fairness since they may affect the decisions that studios and other investors make before finalizing and deciding movies to be released to the public. Even if movies are released, those with lesser predicted revenues or ratings may be chosen for release in less theaters and gain less exposure to audiences.

Thus, fairness must also be considered since errors may change data distribution in the future—if predictions show that certain types of movie will have worse ratings or revenue, then only those movies might be made based on these predictions. If features are not selected correctly and if fairness is not properly considered, predictions create negative feedback cycles. This issue would cause potential *weapons of math destruction* if not addressed correctly due to a feedback loop. (Other criteria for WMD's like measurable outcomes and potentially harmful predictions are less pronounced here. Revenue and rating are both fairly measurable, especially rating).

For supervised learning models like ours, it is also important to assess if the models we use meet the legal requirements of fairness. Considering that Hollywood has and had contentious inclusion issues and discriminatory practices in areas like casting, we must consider how to deal with *protected attributes* like race and gender of cast members for classification of public perception. In future iterations of the project, it might be wise to consider unawareness of classifiers and measures of fairness like demographic parity, equalized odds, and equality of opportunity. Only then will disparate treatment and disparate impact be prevented when deciding which films do get made in Hollywood based on views of their potential profitability and public reception.

References

CPI Data Information from:
https://www.bls.gov/cpi/data.htm

Ticket Price Information from:
https://help.imdb.com/article/imdbpro/industry-research/box-office-mojo-by-imdbpro-faq/GCWTV4MQKGWRAUAP?ref_=mojo_cso_md#inflation
https://www.davemanuel.com/whatitcost.php
https://www.the-numbers.com/market/

Frequency encoding inspiration from:
https://www.kdnuggets.com/2021/05/deal-with-categorical-data-machine-learning.html

Embedding the movie title from:
https://static.googleusercontent.com/media/research.google.com/en//pubs/archive/46808.pdf
https://tfhub.dev/google/universal-sentence-encoder/4

Appendix

The following contains information about the contributions of each group member, the figures referred to in this report, and additional information on all of our files and code.

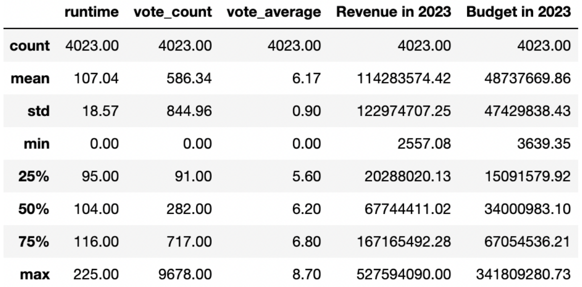
Contributions

<u>Amy Chitnumsab:</u>
Code - Data Cleaning, Exploratory Data Analysis, Feature Transformation, Logistic Regression Models
Report - Exploratory Data Analysis, Transforming Categorical Data in Revenue Prediction, Rating Classification, Results and Discussion

<u>Tejhan Diallo:</u>
Code - Data Cleaning, Linear Regression Models
Report - Revenue Prediction, Results and Discussion, Future Improvements

<u>Jennifer Kim:</u>
Code - Data Cleaning, Adjustments for Inflation on Budget and Revenue
Report - Introduction, Data Preparation, Future Improvements, Fairness and Weapons of Math Destruction

Figures



	runtime	vote_count	vote_average	Revenue in 2023	Budget in 2023
count	4023.00	4023.00	4023.00	4023.00	4023.00
mean	107.04	586.34	6.17	114283574.42	48737669.86
std	18.57	844.96	0.90	122974707.25	47429838.43
min	0.00	0.00	0.00	2557.08	3639.35
25%	95.00	91.00	5.60	20288020.13	15091579.92
50%	104.00	282.00	6.20	67744411.02	34000983.10
75%	116.00	717.00	6.80	167165492.28	67054536.21
max	225.00	9678.00	8.70	527594090.00	341809280.73

Figure 1: Basic statistics showing the distribution of numerical data, including the number of movies in our dataset, mean, standard deviation, minimum value, maximum value, 25% percentile value, 50% percentile value or median, and the 75% percentile value

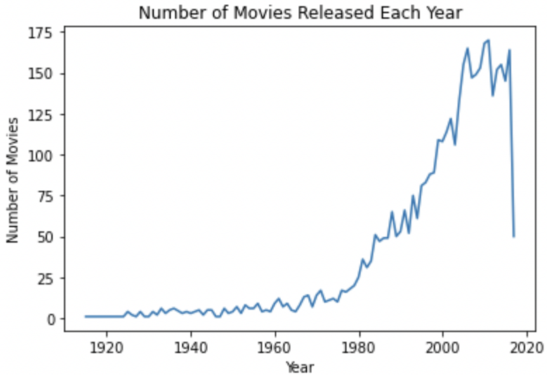


Figure 2: Line graph showing the distribution of movies across years

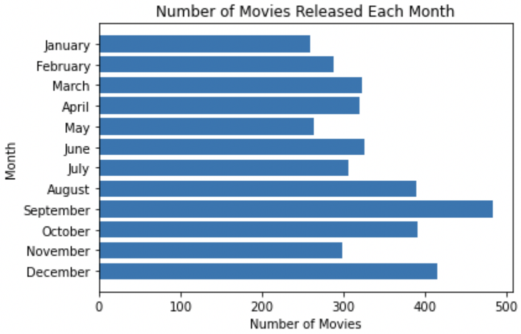


Figure 3: Bar chart showing distribution of movies across various months

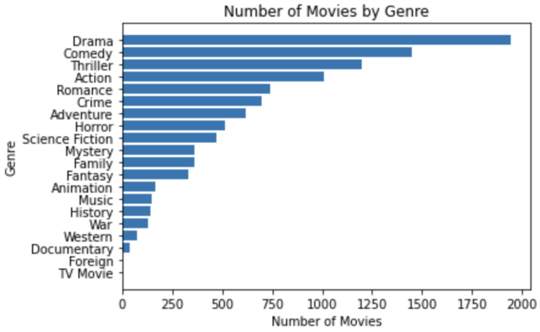


Figure 4: Bar chart showing the distribution of movies across each genre

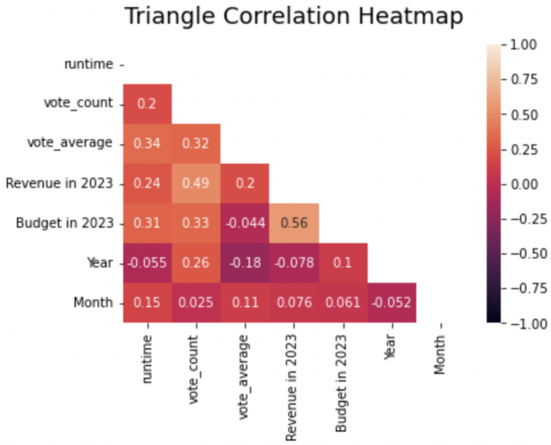


Figure 5: Triangle correlation heatmap showing the correlation values between different numerical values, shade of regions indicate the strength of correlation

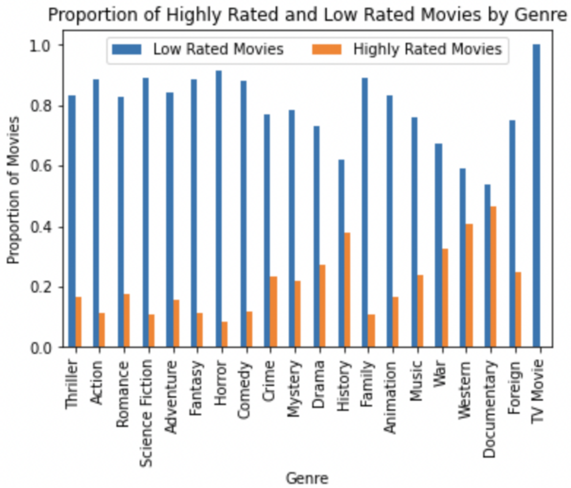


Figure 6: Multi-bar chart showing the proportion of highly and low IMDb rated movies by genre

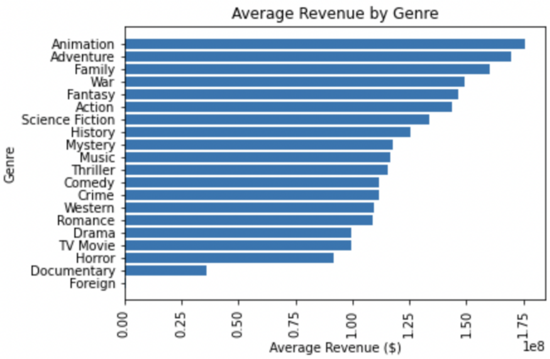
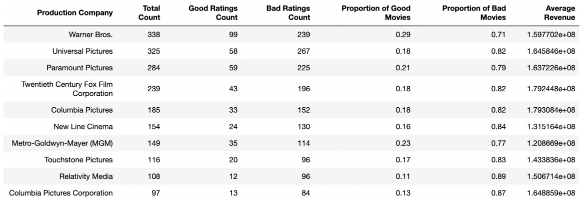
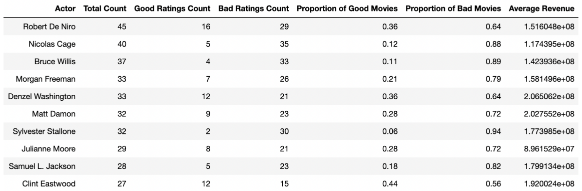


Figure 7: bar chart showing the average revenue of movies under each genre



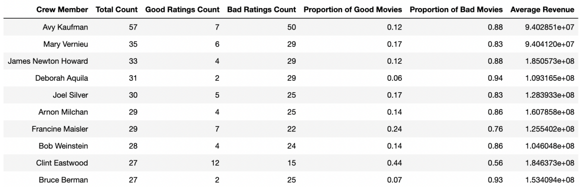
Production Company	Total Count	Good Ratings Count	Bad Ratings Count	Proportion of Good Movies	Proportion of Bad Movies	Average Revenue
Warner Bros.	338	99	239	0.29	0.71	1.597702e+08
Universal Pictures	325	58	267	0.18	0.82	1.645846e+08
Paramount Pictures	284	59	225	0.21	0.79	1.637226e+08
Twentieth Century Fox Film Corporation	239	43	196	0.18	0.82	1.792448e+08
Columbia Pictures	185	33	152	0.18	0.82	1.793084e+08
New Line Cinema	154	24	130	0.16	0.84	1.315164e+08
Metro-Goldwyn-Mayer (MGM)	149	35	114	0.23	0.77	1.208669e+08
Touchstone Pictures	116	20	96	0.17	0.83	1.433836e+08
Relativity Media	108	12	96	0.11	0.89	1.506714e+08
Columbia Pictures Corporation	97	13	84	0.13	0.87	1.648859e+08

Figure 8: statistics on the top 10 production companies based on the number of movies released, statistics include the total number of movies released by each company, the number and proportion of highly rated and low rated movies produced, and the average revenue of movies made by each company



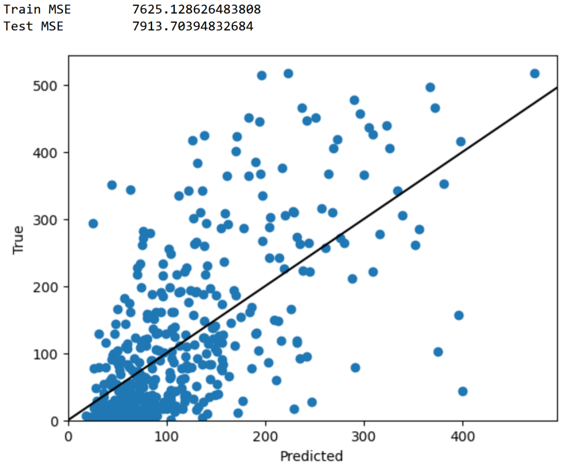
Actor	Total Count	Good Ratings Count	Bad Ratings Count	Proportion of Good Movies	Proportion of Bad Movies	Average Revenue
Robert De Niro	45	16	29	0.36	0.64	1.516048e+08
Nicolas Cage	40	5	35	0.12	0.88	1.174395e+08
Bruce Willis	37	4	33	0.11	0.89	1.423936e+08
Morgan Freeman	33	7	26	0.21	0.79	1.581496e+08
Denzel Washington	33	12	21	0.36	0.64	2.065062e+08
Matt Damon	32	9	23	0.28	0.72	2.027552e+08
Sylvester Stallone	32	2	30	0.06	0.94	1.773985e+08
Julianne Moore	29	8	21	0.28	0.72	8.961529e+07
Samuel L. Jackson	28	5	23	0.18	0.82	1.799134e+08
Clint Eastwood	27	12	15	0.44	0.56	1.920024e+08

Figure 9: statistics on the top 10 cast members or actors based on the total number of movies each person starred in, statistics include the total number of movies, the number and proportion of highly and low rated movies, and the average revenue of movies that each person starred in



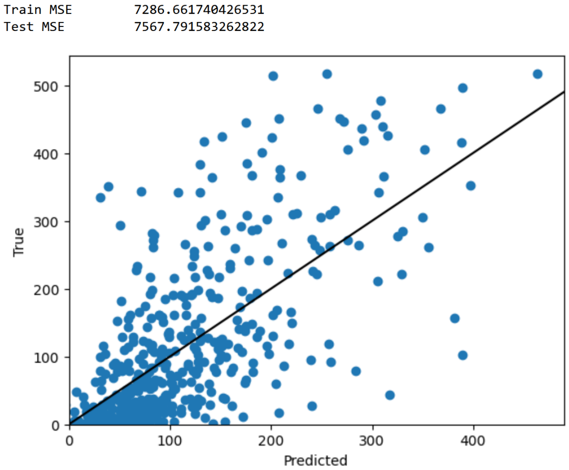
Crew Member	Total Count	Good Ratings Count	Bad Ratings Count	Proportion of Good Movies	Proportion of Bad Movies	Average Revenue
Avy Kaufman	57	7	50	0.12	0.88	9.402851e+07
Mary Vernieu	35	6	29	0.17	0.83	9.404120e+07
James Newton Howard	33	4	29	0.12	0.88	1.850573e+08
Deborah Aquila	31	2	29	0.06	0.94	1.093165e+08
Joel Silver	30	5	25	0.17	0.83	1.283933e+08
Arnon Milchan	29	4	25	0.14	0.86	1.607858e+08
Francine Maisler	29	7	22	0.24	0.76	1.255402e+08
Bob Weinstein	28	4	24	0.14	0.86	1.046048e+08
Clint Eastwood	27	12	15	0.44	0.56	1.846373e+08
Bruce Berman	27	2	25	0.07	0.93	1.534094e+08

Figure 10: statistics on the top 10 crew members based on the total number of movies that they worked on, statistics include the total number of movies, the number and proportion of highly and low rated movies, and the average revenue of movies that each person worked on



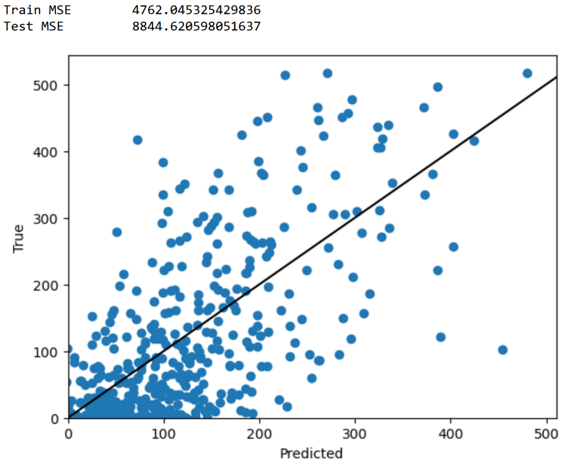
```
Train MSE        7625.128626483808
Test MSE         7913.70394832684
```

Figure 11: Base Model: Linear Regression model predicting revenue with budget and runtime alone.



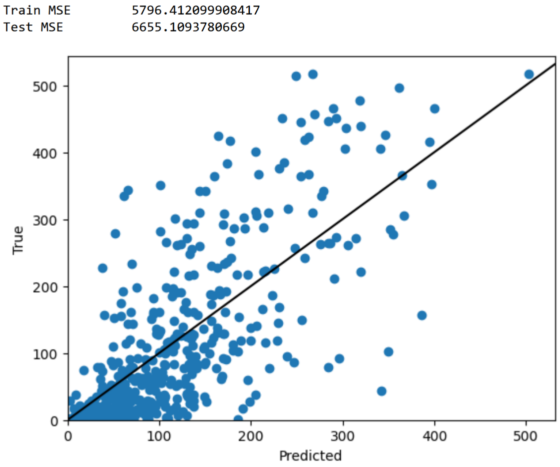
```
Train MSE        7286.661740426531
Test MSE         7567.791583262822
```

Figure 12: Linear regression model predicting revenue with budget, runtime, genre, and frequency encodings.



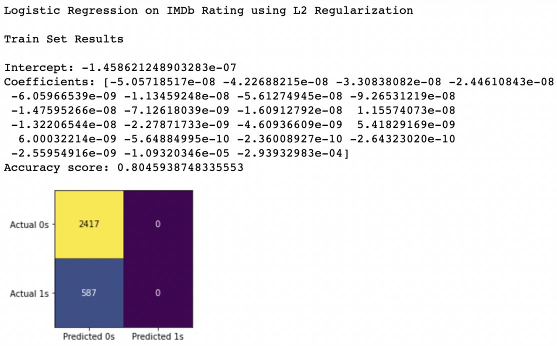
```
Train MSE        4762.045325429836
Test MSE         8844.620598051637
```

Figure 13: Linear regression with budget, runtime, genre many-hot encoding, and unregularized text embedding of overview.



```
Train MSE        5796.412099908417
Test MSE         6655.1093780669
```

Figure 14: Final Model: Linear regression with budget, runtime, frequency encoding, and regularized text embedding.

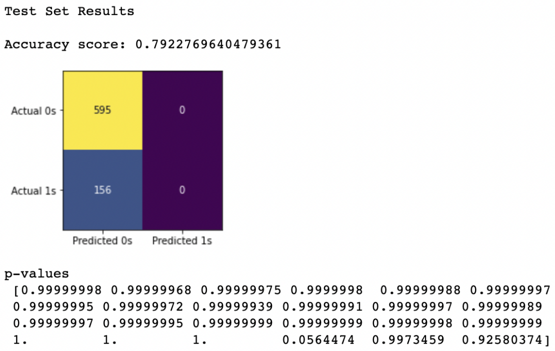


```
Logistic Regression on IMDb Rating using L2 Regularization

Train Set Results

Intercept: -1.458621248903283e-07
Coefficients: [-5.05718517e-08 -4.22688215e-08 -3.30838082e-08 -2.44610843e-08
 -6.05966539e-09 -1.13459248e-08 -5.61274945e-08 -9.26531219e-08
 -1.47595266e-08 -7.12618039e-09 -1.60912792e-08  1.15574073e-08
 -1.32206544e-08 -2.27871733e-09 -4.60936609e-09  5.41829169e-09
  6.00032214e-09 -5.64884995e-10 -2.36008927e-10 -2.64323020e-10
 -2.55954916e-09 -1.09320346e-05 -2.93932983e-04]
Accuracy score: 0.8045938748335553
```

Figure 15: logistic regression model with L2 regularization results for training dataset, results include model coefficients, accuracy score, and confusion matrix heatmap

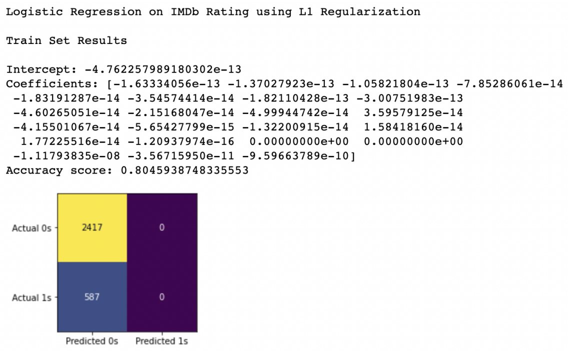


```
Test Set Results

Accuracy score: 0.7922769640479361
```

```
p-values
 [0.99999998 0.99999968 0.99999975 0.9999998  0.99999988 0.99999997
 0.99999995 0.99999972 0.99999939 0.99999991 0.99999997 0.99999989
 0.99999997 0.99999995 0.99999999 0.99999999 0.99999998 0.99999999
 1.         1.         1.         0.0564474  0.9973459  0.92580374]
```

Figure 16: logistic regression model with L2 regularization results for testing dataset, results include model coefficients, accuracy score, confusion matrix heatmap, and final p-values of each feature

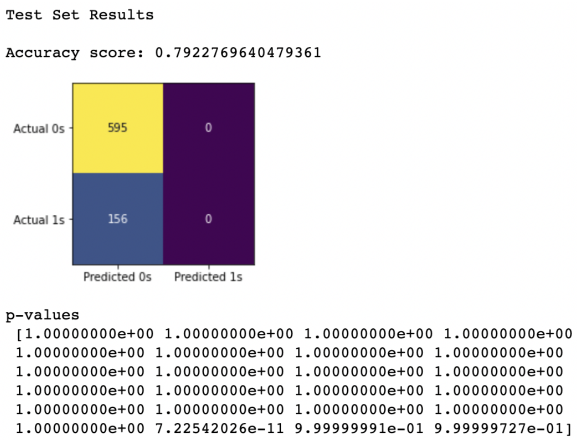


```
Logistic Regression on IMDb Rating using L1 Regularization

Train Set Results

Intercept: -4.762257989180302e-13
Coefficients: [-1.63334056e-13 -1.37027923e-13 -1.05821804e-13 -7.85286061e-14
 -1.83191287e-14 -3.54574414e-14 -1.82110428e-13 -3.00751983e-13
 -4.60265051e-14 -2.15168047e-14 -4.99944742e-14  3.59579125e-14
 -4.15501067e-14 -5.65427799e-15 -1.32200915e-14  1.58418160e-14
  1.77225516e-14 -1.20937974e-16  0.00000000e+00  0.00000000e+00
 -1.11793835e-08 -3.56715950e-11 -9.59663789e-10]
Accuracy score: 0.8045938748335553
```

Figure 17: logistic regression model with L1 regularization results for training dataset, results include model coefficients, accuracy score, and confusion matrix heatmap



```
Test Set Results

Accuracy score: 0.7922769640479361
```

```
p-values
 [1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00
 1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00
 1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00
 1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00
 1.00000000e+00 1.00000000e+00 1.00000000e+00 1.00000000e+00
 1.00000000e+00 7.22542026e-11 9.99999991e-01 9.99999727e-01]
```

Figure 18: logistic regression model with L1 regularization results for testing dataset, results include model coefficients, accuracy score, confusion matrix heatmap, and final p-values of each feature

External Links for Additional Information

Links to Google Drive Folder

https://drive.google.com/drive/folders/169U9q14EYYLiR-z9LyTrnJ3z03tyRAT_?usp=sharing

Link to GitHub repository

https://github.com/tejhan/Movie-Success-Prediction