What Makes a Film Successful?

Edward Jefferson Halim, Ernest Salim, Jonathan Tejakusuma
UCLA
MATHEMATICS DEPARTMENT

November 2022

Contents

1 Abstract

Movies, commonly known as a story captured by the camera are now currently one of the highest leading industries in the world. Nowadays, film industries care about the quality of a movie and wish to bring entertainment to those who are watching. Thus, tremendous effort and budget have been sacrificed to obtain a solid rating so that the world would view their creation and gain positive income.

In this project, we are curious about how impactful movie aspects are towards the money that is gained by the film industry. Therefore, we conduct research to compare whether or not movie aspects like ratings, budget, etc. really affect the success of a movie by using the methodology of curve fitting.

2 Problem Description

According to IMBD, roughly 2,577 movies are produced each year and there is not a single sign that the production of movies will stop anytime soon. Nowadays, we can clearly see the results of the hard work that is committed by film industries if we compare it with films from the previous decades. Components from more impactful scripts, clear and neat shots from camera equipment, and more appreciated background music. However, everything mentioned above requires money in order to operate smoothly, and there are many other aspects that contributed to the income of the movie alternative to the budget.

As we know there are many factors that formed a movie based on our data. To be precise, we can divide our variables into two types:

1. Controllable Factors
   budget, runtime, crew and casts, languages spoken, country of production, production company, month of the release date, certificate license, does the film has a sequel

2. Uncontrollable Factors
   total world sales, community rating, TMDb popularity, number of awards won, societal trend

Thus, we are curious about how movie industries can keep up with their expenses and question whether the budget affects the income of the movie. From there, develop new interesting questions such as do other factors such as ratings, runtime, list of actors, and even marketing companies also play a role in defining the success of the movie.

Not only this topic is interesting, but it is also important because, in this current era, the entertainment industry is being led by movie companies. We have seen countless movies that did not stop producing even during the pandemic, where people do not have a lot of diversions to fill their days. As a matter of fact, movies are very dear to our hearts and thus why we chose this topic.

3 Simplifications

Based on the background factors in a movie, there are too many aspects that we need to consider. Therefore, we narrow down some of the elements that we think are the most impactful and combined them into one giant data. Also, we can only examine using the data that is available and provided to us from the internet. Thus, the factors that we want to use in our project are:

1. Numerical factors:
   budget, runtime, total world sales, community rating, TMDb popularity, month of the release date

2. Categorical factors:
   production company and certificate license

Therefore, we needed to find right datasets that host all of the data that we need. Here are the datasets that we are going to use:

1. Highest Grossing Movies Dataset
   The dataset is provided from Kaggle. We decided to use this as our main dataset because the dataset was just got updated very recently and it has pre-sorted films based on their total revenue.

2. TMDb Dataset
   TMDb is a popular community forum for all things film where people can look up detailed film's information. Because

TMDb only provides its data through API and HTTP calls, then we have to fetch the .csv dataset from Kaggle for convenience. Despite the Kaggle dataset was last updated 3 years ago, it still provides relevant and reliable information that we can use combined with our main dataset.

From our initial perspective, one of the most defining factors is obviously the film's budget. We firmly believe that the budget outlines most of the dependent variables that we want to experiment on. For example:

- Case 1:
  If a film has more budget, does it mean that they are capable in creating a longer film? Thus, the film has a longer runtime, which can lead to various critical and financial reception.
- Case 2:
  If a film successfully get a funding from one production company, does it mean that the film is going to be more successful financially and critically?
- Case 3:
  If a film invested more budget in advertising and marketing, does it mean that the film will gain more popularity and total sales at the end of the day?

Consequently, all things that are affected by the budget also lead to community rating, popularity, and total sales and revenue. However, all of these aspects are strongly tied to the fact that which production company produces the movie, how the genre affects the success, how it relates to other movies, the month and season of its release date, the certificate rating of the movie, and the marketing distribution company.

Our hypothesis before the test was that the higher the budget of the production, the more popular the production company, the better the reception of the movie, and other trivial categorical variables like the social trend, the month of the release date, and its certificate rating would lead to a higher net income of that movie

4 Mathematical Model

To obtain the model that we desired, we need to break down the steps. To begin with, we start by scraping/researching the data. Once we found the data that we liked, we compiled the data and display them in our preferred programming language, which is Python. Then, we implemented pandas and jupyter notebook to filter the factors that we think are most important. Once everything is done, we picked the top 100 movies and graph them by implementing Seaborn, which is a Python library to graph linear regressions and a tool for curve fitting. Finally, we can count the gradient to obtain our answers.

Below is our code:

1. Setting Up the Environment

   (a) Importing necessary Python libraries

```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from scipy.stats import linregress

pd.set_option('display.max_colwidth', None)
pd.set_option('display.max_columns', None)
```

   (b) Importing the datasets

```
tmdb = pd.read_csv("tmdb_10000_movies_data.csv")
hgmb = pd.read_csv("highest_grossing_movies.csv")
```

2. Data cleaning and processing

(a) Removing unnecessary columns

```
tmdb.drop(['TMDb_Id', 'Original_Title', 'Cast', 'Crew', 'Genres', 'IMDb_Id',
'Overview', 'Release_Status', 'Languages_Spoken', 'Tagline'], axis = 1, inplace = True)
hgmb.drop(['Unnamed: 0', 'Movie Info', 'Movie Runtime'], axis = 1, inplace = True)
```

(b) Removing outliers and irrelevant entries

```
strip_year = lambda x : x.split("(")[0].strip()

hgmb['Year'] = hgmb['Title'].apply(lambda title : title[-5:-1])
hgmb['Title'] = hgmb['Title'].apply(lambda title : strip_year(title))
hgmb.sort_values(by = ['World Sales (in $)'], ascending = False, inplace = True)
hgmb.reset_index(drop = True, inplace = True)

tmdb = tmdb[tmdb['Release_Date'].notna()]
tmdb['Year'] = tmdb['Release_Date'].apply(lambda date : date[:4])
tmdb = tmdb.loc[tmdb['Rating_average'] != 0]
tmdb = tmdb.loc[tmdb['Budget'] != 0]
tmdb.sort_values(by = ['Revenue'], ascending = False, inplace = True)
tmdb.reset_index(drop = True, inplace = True)
```

3. Data Merging

```
df = pd.DataFrame()

for index, title in enumerate(hgmb['Title']):
    year = hgmb[hgmb['Title'] == title]['Year'].iloc[0]
    entry = tmdb.loc[(tmdb['Title'] == title) & (tmdb['Year'] == year)]
    if (len(entry) == 0):
        hgmb.drop(labels = index, axis = 0, inplace = True)
    else:
        df = pd.concat([df, entry], ignore_index = True)

hgmb.reset_index(drop = True, inplace = True)
hgmb.drop(['Year', 'Title', 'Release Date'], inplace = True, axis = 1)
df.drop(['Revenue'], inplace = True, axis = 1)

df = pd.concat([df, hgmb], axis = 1)
df.rename(columns = {
    'Domestic Sales (in $)': 'Domestic_Sales',
    'International Sales (in $)': 'International_Sales',
    'World Sales (in $)': 'World_Sales',
    'Rating_average': 'Rating_Average'
}, inplace = True)
df['Release_Date'] = pd.to_datetime(df['Release_Date'])
df['Month'] = df['Release_Date'].dt.month
```

4. Plotting Linear Regression

```
sns.regplot(data = df, x, y, fit_reg = True)
```

5. Finding Linear Regression Values

```
linregress(x, y)[0:2]
```

6. Finding Pearson Correlation Coefficient

```
np.corrcoef(x, y)[0][1]
```

5 Solution to The Mathematical Problem

We solved our problem by implementing curve fitting. Then, we found the gradient by using the formula of

$$y = mx + c \tag{1}$$

y: Point in the y-axis
x: Point in the x-axis
m: Gradient of the line
c: Constant

Based on the gradient, we can find how impactful a factor is to the income of the movie. If the slope is negative, that means the more budget that is sacrificed, the less income they get, and vice versa. From here, we can then count all the gradients for other factors and compare whether there is a positive correlation or negative correlation in the linear regression graphs. A positive correlation means that as the x value increases, the y value also increases whereas a negative correlation means that as the x value increases, the y value decreases.

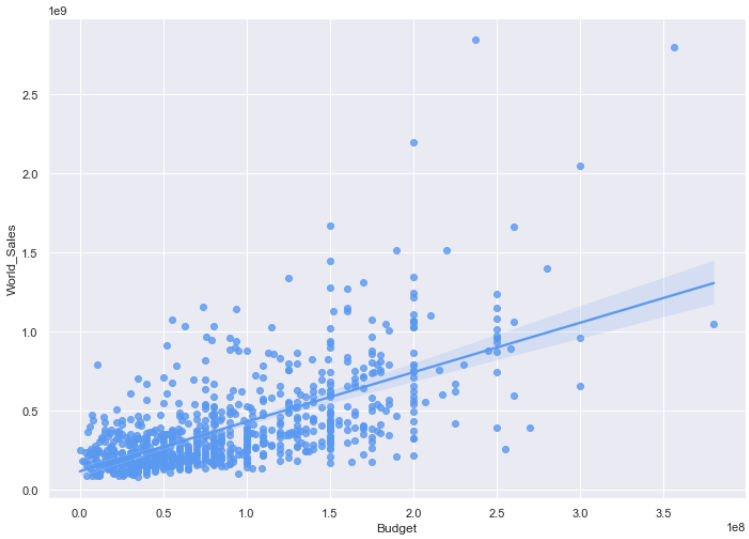
Then, we can find the Pearson correlation coefficient by this formula

$$r = \frac{\sum(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum(x_i - \bar{x})^2(y_i - \bar{y})^2}}$$

r: Pearson correlation coefficient
x_i: x variable samples
$\bar{x}$: mean of x values
y_i: y variable samples
$\bar{y}$: mean of y values

6 Results

1. **Linear Correlation between Budget and Sales**



The code to produce this graph

```
sns.regplot(data = df, x = 'Budget', y = 'World_Sales', fit_reg = True)
```

Using this linregress function from SciPy, we can find the slope and the intercept of our regression line.

```
linregress(df['Budget'], df['World_Sales'])[0:2]
```

Thus, the linear equation of the regression line is

$$y = 3.1354x + 116005746.0059$$

For more insight, we can find the the linear correlation coefficient between the two data by

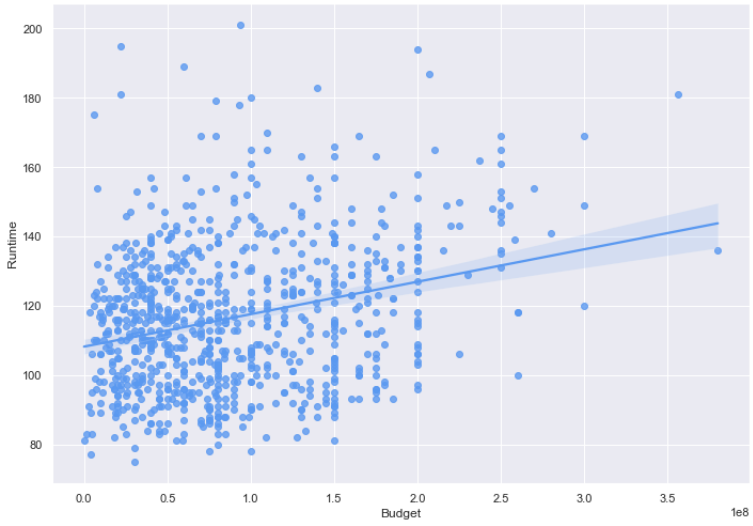
```
np.corrcoef(df['Budget'], df['World_Sales'])[0][1]
```

$$r = 0.6419$$

Description

As we can see from the graph above, there exists a positive correlation between the data. Moreover, this does satisfy our assumption and hypothesis that with bigger budget, the bigger income and revenue a film can reap. However, budget is not the only variable that contributes to the financial success of a film.

2. **Linear Correlation between Budget and Runtime**
While sales is dependent outcome that we can't control, a film runtime is a variable that plays an equally important role in the success of a film, and we think that budget is the main factor of deciding how long the film would be.



The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Budget', y = 'Runtime', fit_reg = True)
linregress(df['Budget'], df['Runtime'])[0:2]
np.corrcoef(df['Budget'], df['Runtime'])[0][1]
```

Thus, the linear equation of the regression line is

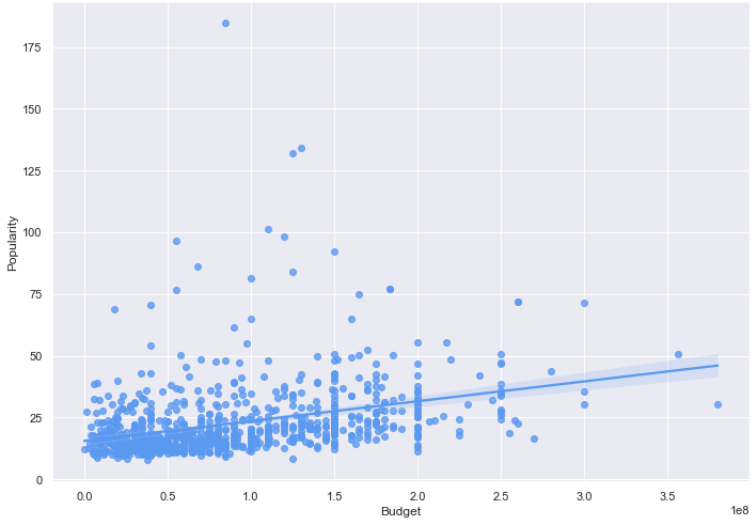
$$y = 9.3645x + 108.1855$$

$$r = 0.2791$$

Description

From the above findings, we can see that, despite the weak linear correlation connection, there's still a positive connection between budget and a film's runtime. Thus, we can conclude that a film budget plays a role in defining its runtime.

3. **Linear Correlation between Budget and Popularity**



The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Budget', y = 'Popularity', fit_reg = True)
linregress(df['Budget'], df['Popularity'])[0:2]
np.corrcoef(df['Budget'], df['Runtime'])[0][1]
```

Thus, the linear equation of the regression line is

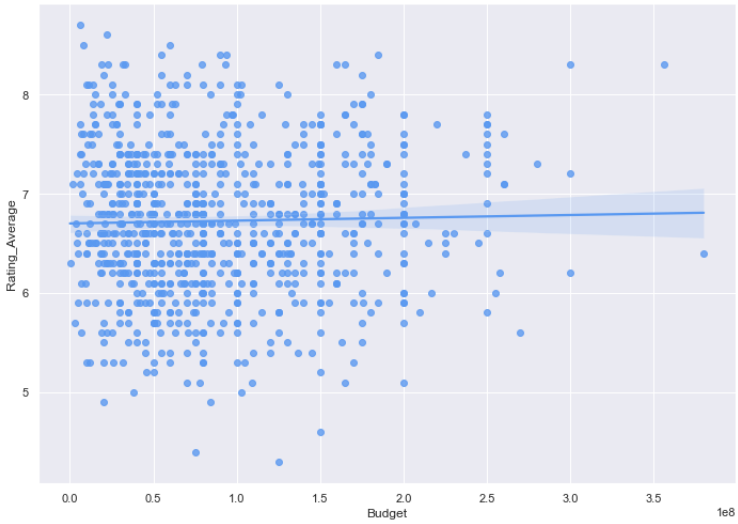
$$y = 8.0478x + 15.4577$$

$$r = 0.3370$$

Description

From the above experimentation, we can see that there is a decent connection between budget and popularity metric. This tells us that more budget lead to more resources that can help a film reach more audiences and market, which eventually increases the overall popularity of a film.

4. **Linear Correlation between Budget and Rating**



The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Budget', y = 'Rating_Average', fit_reg = True)
linregress(df['Budget'], df['Rating_Average'])[0:2]
np.corrcoef(df['Budget'], df['Rating_Average'])[0][1]
```

Thus, the linear equation of the regression line is

$$y = 2.8338x + 6.7002$$

$$r = 0.0239$$

Description

Despite a film's budget can usher the film to reach its financial success, it may not be able to garner the film good critical receptions. This is proven from the above experimentation that there is a weak correlation between budget and the film's rating, hence the slope of the regression is almost linear. What makes film critically and artistically good is the expertise in creating the film, though it's clear that budget can provide better production resources like better technologies, actors, and other many aspects, still, the experiment specified that the higher the budget, does not translate to a higher rating.

5. **Linear Correlation between Runtime and Popularity**

The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Runtime', y = 'Popularity', fit_reg = True)
linregress(df['Runtime'], df['Popularity'])[0:2]
np.corrcoef(df['Runtime'], df['Popularity'])[0][1]
```

Thus, the linear equation of the regression line is

$$y = 0.1750x + 2.2323$$

$$r = 0.2458$$

Description

Our results show that there is a positive correlation between runtime and popularity. The longer a movie is, the more people would acknowledge the movie. We believe that this can occur since the audience can walk out of the movie without the feeling of regret spending their money for a shorter movie. Thus, people in general prefer movies with longer runtime to shorter ones.

6. **Linear Correlation between Runtime and Rating**

The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Runtime', y = 'Rating_Average', fit_reg = True)
linregress(df['Runtime'], df['Rating_Average'])[0:2]
np.corrcoef(df['Runtime'], df['Rating_Average'])[0][1]
```

Thus, the linear equation of the regression line is

$$y = 0.0157x + 4.8943$$

$$r = 0.4456$$

Description

Notably, the runtime clearly influenced the rating of a movie. The longer a movie is, the higher the rating, which is shown by the positive slope. Personally, we think that the longer the movie, the better a director can explore to share their creativity and thus, create a movie that is enjoyable despite the time.

7. **Linear Correlation between Runtime and Sales**

The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Runtime', y = 'World_Sales', fit_reg = True)
linregress(df['Runtime'], df['World_Sales'])[0:2]
np.corrcoef(df['Runtime'], df['World_Sales'])[0][1]
```

Thus, the linear equation of the regression line is

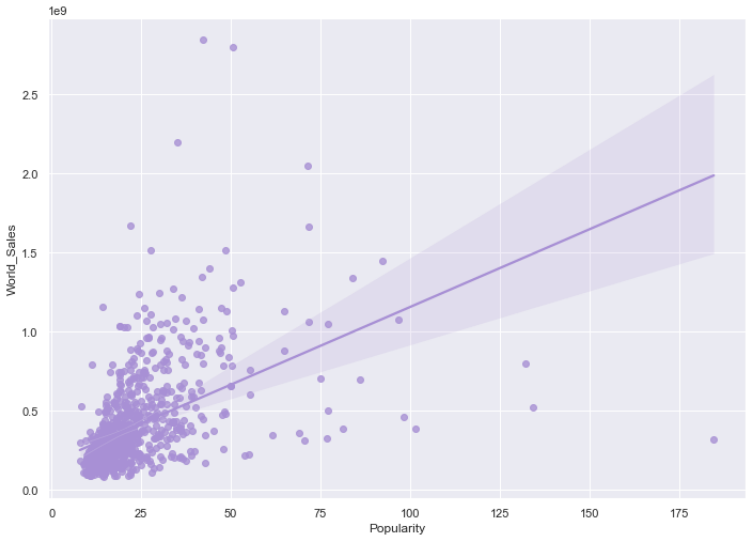
$$y = 4675801.7514x - 149479345.3608$$

$$r = 0.3211$$

Description

From the above graph, there is a positive correlation between runtime and sales. Just like mentioned above in graph 5, where the audience wanted movies with longer runtime since they think that it is worth spending their money to watch a longer movie. Thus, increasing the ticket sales for the movie.

8. **Linear Correlation between Popularity and Sales**



The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Popularity', y = 'World_Sales', fit_reg = True)
linregress(df['Popularity'], df['World_Sales'])[0:2]
np.corrcoef(df['Popularity'], df['World_Sales'])[0][1]
```

Thus, the linear equation of the regression line is

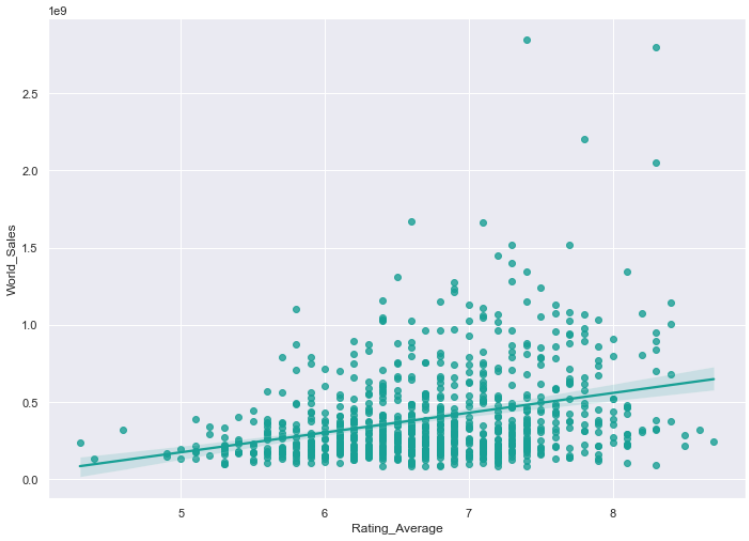
$$y = 9830269.1764x - 172944258.3445$$

$$r = 0.4806$$

Description

It is clear that a more popular movie would have a higher net income since popular movies possess solid advertising. Furthermore, the higher reputation of a movie leads to a great number of audience that consequently increases the number of ticket sales.

9. **Linear Correlation between Rating and Sales**



The code to produce this graph and equation values

```
sns.regplot(data = df, x = 'Rating', y = 'World_Sales', fit_reg = True)
linregress(df['Rating'], df['World_Sales'])[0:2]
np.corrcoef(df['Rating'], df['World_Sales'])[0][1]
```

Thus, the linear equation of the regression line is

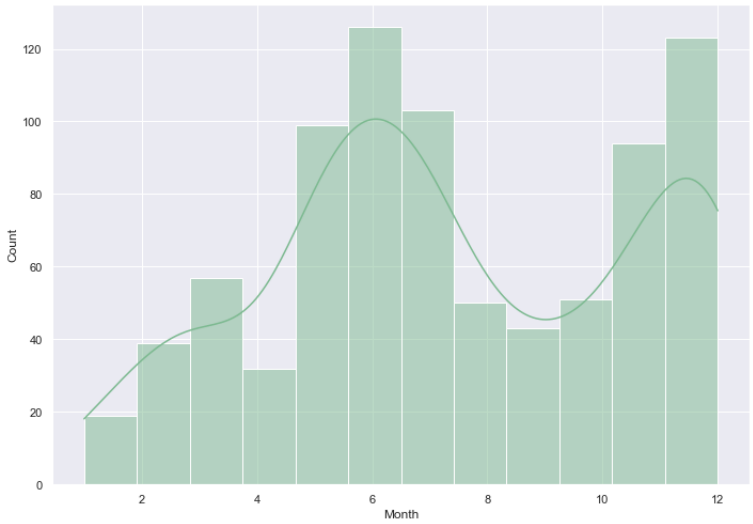
$$y = 128177536.6642x - 466663877.7494$$

$$r = 0.3104$$

Description

It is obvious that the higher a rating of a movie is, the higher the net income the movie will gain. Audiences are biased towards movies that have better ratings, thus, higher ratings would make them curious to watch the movie. Thus, it increases the number of ticket sales and escalade the revenue that is generated by the production company.

10. **Month of the Release Date**



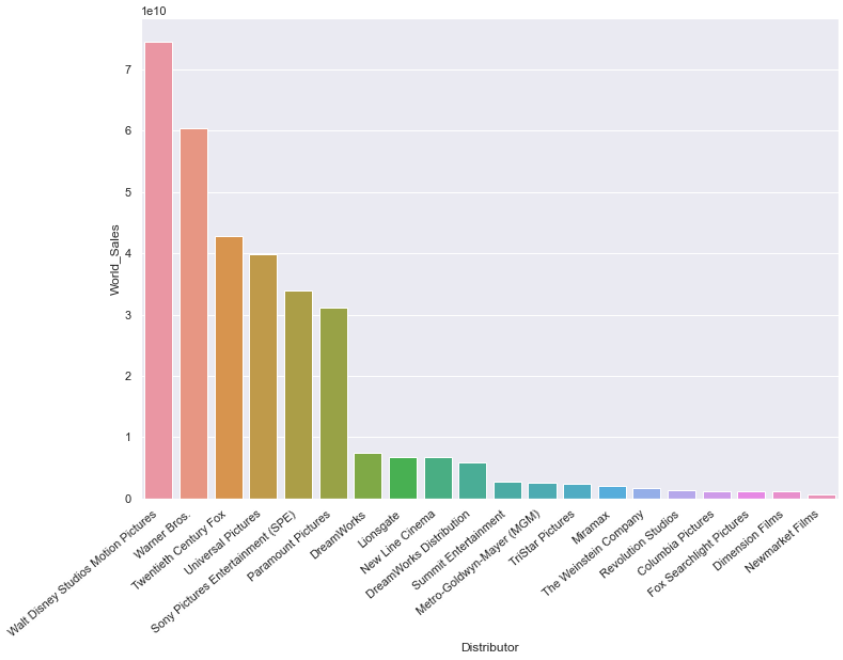
The code to produce this graph

```
sns.histplot(df['Month'], kde = True, bins = 12)
```

Description

As we can see from the observation that we have made, June and December are the most popular month for releasing a movie. Our assumption is because those are the time when people are having their holidays. Hence there are terms: "summer blockbuster" and "christmas movies".

11. **Distribution Companies to Total Sales Barplot**



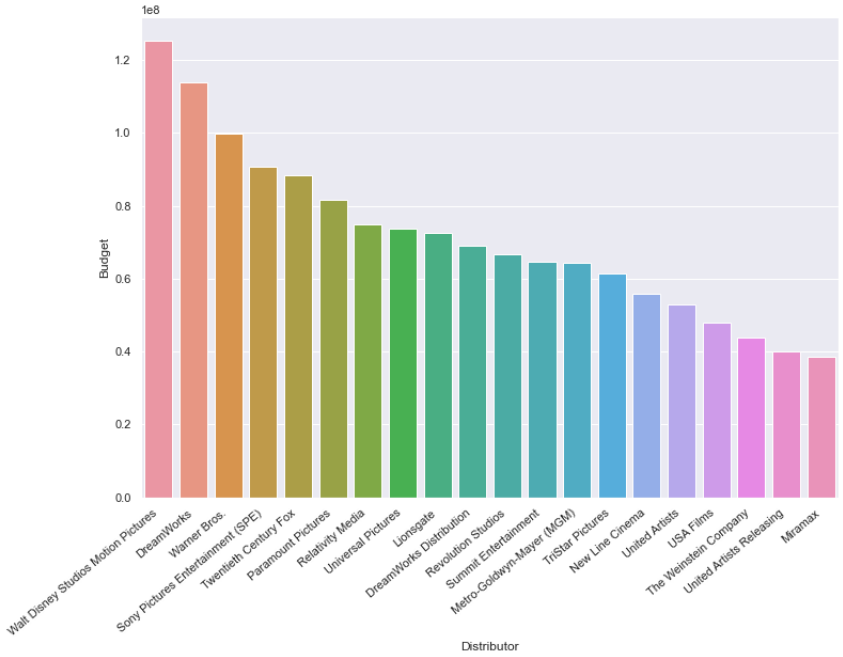
The code to produce this graph

```
df2 = df.groupby(['Distributor']).mean().sort_values(by = 'World_Sales', ascending
= False).head(20)
```

```
ax = sns.barplot(data = df2, x = df2.index, y = 'World_Sales')
ax.set_xticklabels(ax.get_xticklabels(), rotation = 40, ha = "right")
plt.show()
```

We only consider the first 20 companies which gained the most sales and revenue.
From the above experiment, we can see that distribution companies like Walt Disney, Warner Bros, and Twentieth Century Fox are the leading frontrunners in making their movies financially successful. Reaching almost $7.5 \cdot 10^10$ dollars in total revenue. This also tells that their significance plays a role in publishing and marketing their films to wider audiences.

12. **Distribution Companies to Average Budget Barplot**



The code to produce this graph

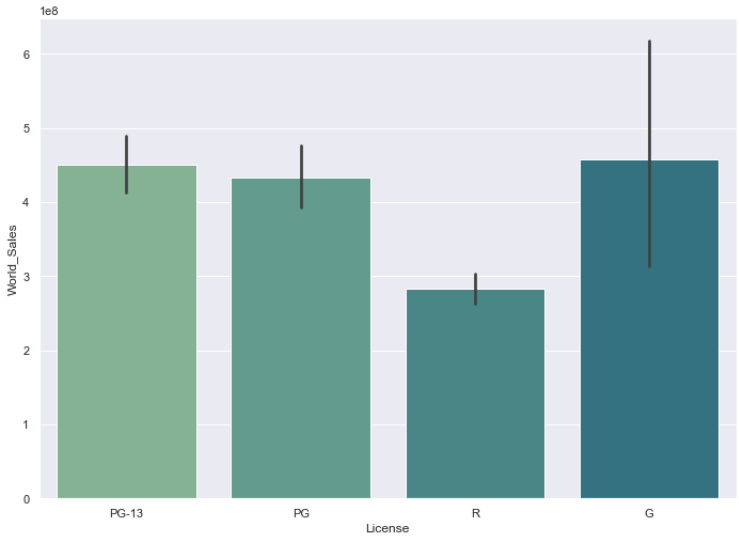
```
df3 = df.groupby(['Distributor']).mean().sort_values(by = 'Budget', ascending =
False).head(20)

ax = sns.barplot(data = df3, x = df3.index, y = 'Budget')
ax.set_xticklabels(ax.get_xticklabels(), rotation = 40, ha = "right")
plt.show()
```

Description

Despite Walt Disney, Warner Bros, and Twentieth Century Fox leading the charge as three of the most successful companies in creating films, the three may not be the ones which provide the biggest average budget for their films. This means that these companies are the leading institution when it comes to financial resources.

13. **Certificate License to Average Sales Barplot**



The code to produce this graph

```
sns.barplot(data = df, x = 'License', y = 'World_Sales')
```

Description

Finally, certificate rating of G (general) makes the film more accessible to people of all ages, thus making the film more financially successful.

7 Improvement

Even though we think our answer is already accurate, there will always be room for improvement. To begin with, we can always compare more data from movies that do not produce such a high income and with a lower budget. By doing so, we can compare movies that succeeded and whatnot. However, comparing a larger amount of values also implies more scattered outcomes. In addition, we did not use the most updated data since our data is based from early 2022, where some of the highest-grossing movies in the current chart are not available in our dataset. Also, by having a more robust dataset where it can provide us with more complete variables such as number of awards won, casts net worth, or how significant the crew is can be very helpful in getting a more accurate result.

8 Conclusion

Based on our results, we can safely assume that the higher the budget, the higher the revenue. Furthermore, we have proved our first case in simplification by our 2nd graph that the higher the budget, the longer the runtime of the movie will be. And on our 7th graph, we proved that the higher the runtime, the higher the world sales or revenue, and the higher the popularity. The most noticeable growth is the comparison between popularity and revenue. Thus, budget is the root of all of these quantitative factors and as the budget increases, all of the factors will continue to grow too.

However, even though our results are clear and reasonable, we are questioning whether our 4th graph, where we compare budget and rating since the result does not complement our hypothesis. Based on the results, we have a positive slope, but it is not really noticeable like the others. However, if we link them to the other results (Budget with runtime, runtime with popularity, then popularity with world sales), we should have a guaranteed positive slope that we thought would be more recognizable.

That being said, we are assured that a higher budget would lead to a more successful movie based on revenue. If we wanted to create a movie that is most profitable, we would have to sacrifice a lot of initial budgets, increase the runtime of the movie, release our movie in the month of June or December, and focus on the general audience. Lastly, if we can choose our own production company, we should pick Walt Disney or Warner Bros since they can help us generate the most revenue compared to other companies.

Overall from this project, we learned how to utilize Python libraries and the programming environment in general. Firstly, the exploration of pandas, which is the beating heart of our current project. Secondly, the implementation of seaborn granted us the curve fitting and the display of our graph. Thirdly, we utilized SciPy and Numpy to provide us with the mathematical values for our observations. By applying all of these frameworks, it grants us critical problem-solving about data cleaning, data processing, and data merging.

9 References

Seaborn: Regression Plots. GeeksforGeeks. Retrieved December 5, 2022, from https://www.geeksforgeeks.org/seaborn-regression-plots/: :text=Regression

Dhara, R. (2020, April 9). TMDb datasets. Kaggle. Retrieved December 5, 2022, from https://www.kaggle.com/datasets/4e39326174055025d147198ace987af8a503baaa4b28203afa06c6e02b5755de

Jupyter. Jupyter notebook tutorial in Python. (n.d.). Retrieved December 5, 2022, from https://plotly.com/python/ipython-notebook-tutorial/: :text=Jupyter

Python pandas tutorial. Tutorials Point. (n.d.). Retrieved December 5, 2022, from https://www.tutorialspoint.com/python$_pandas/index.html$

$Scientist, G.M.I.D., Author : BrendanMartinFounderofLearnDataSci, amp; Author : LaurenWashington LeadDataScientistamp; MLDeveloper.(n.d.).Pythonpandastutorial : Acompleteintroductionforbeginners.LearnDataScience–Tutorials, Books, Courses, andMore.RetrievedDecember5, 2022, fromhttps : //www.learndatasci.com/tutorials/python–pandas – tutorial – complete – introduction – for – beginners/$