Introduction:
Group Seven - Cassidy Bell, Jason Johnson, Gorgina Kareem, and Erika Pino.

For project 1, we will explore various aspects of the IMDb dataset to uncover relationships between movie attributes and their ratings. Specifically, we aim to answer several key questions: How does a movie's runtime influence its IMDb rating? Does the number of votes a movie receives correlate with its rating? Which movie genre garners the most votes from IMDb users? And finally, which director excels in the action genre based on Metascore, and how does their performance compare to directors in other genres? By visualizing these questions, we hope to reveal meaningful patterns and insights that can offer a deeper understanding of the factors influencing a movie's reception and success on IMDb.

We selected a dataset that focuses on the top 1,000 movies according to IMDb ratings; IMBd ratings are determined by registered IMDb users who vote on a 1-10 star scale, averaged (IMDb). Additionally, our IMDb dataset includes Metascores, scored 0-100, which are derived from the result of the weighted average of reviews given by a panel of assigned *and respected* industry critics (IMDb). Our selected dataset required some adjustments, resulting in eliminating a number of titles with missing information, refining our working dataset to just over 700 movie titles.

Using this adjusted dataset, we analyzed the relationship between user-generated IMDb ratings, critic-assigned Metascores, and a movie's gross success. We focused particularly on categorical characteristics—*such as genre, certifications lead actor, and runtime*—to identify patterns and correlations. Our goal is to provide insights for the film industry into the factors that influence viewer ratings and how these ratings might align with financial success.

Thesis (based on Intro):
By examining the relationship between IMDb user ratings, Metascores, and categorical attributes such as genre, director, and runtime, we aim to uncover factors that possibly contribute, or correlate, to a movie's success and how these elements align with audience and critical reception.

Data Cleaning:
To perform the data cleaning and analysis, we used **Pandas**. Pandas provided us with the tools to easily load, clean, and explore the IMDb dataset. First, we imported the dataset into a Pandas DataFrame, which allowed us to efficiently handle and analyze the data. After we did that, we used the functions `missing_values = df.isnull()`, `non_missing_values = df.nonnull()`, and `missing_values = df.isna().sum()` to check for missing and non-missing values and then count how many of those values were in the dataset. Once we did that we used `dropna()`, to drop missing values. For example if anything in the gross column is 0 or null it will not help us prove our data. Next, we had to split and extract the data in the Runtime column to make it integers. We used the code `.str.split().str[0].astype(int)`. We had to do a similar code with the gross column; however instead of splitting and extracting the number, we revoked the commas and converted

to integers with `.str.replace(',', '').astype.(int).` The next step in the process was to count the number of genres and limit the data to only show the first genre listed in the column. `.str.split(',').str.len()` and `.str.split(',').str[0]`. Next we focused on the Certification column. We ran a `Certificate.value_counts()` code and then after noticing that there are values that we need to combine, we ran the following code to clean up the certificate values:

```
df2["Certificate"] = df2.Certificate.replace({
    "U": "G",
    "A": "R",
    "UA": "PG-13",
    "U/A": "PG-13",
    "TV-PG": "PG",
    "GP": "PG",
    "Approved": "PG",
    "Passed": "PG",
})
```

Next, we used the code `drop(columns = [,])` to drop a few columns as they were not going to be used with our analysis. We renamed a few columns to make the names cleaner and easier to use with this code: `df2 = df2.rename(columns={`

```
    "Series_Title": "Movie_Title",
    "Certificate": "Rating",
})
```

For visualization, we used Pandas' integration with **Matplotlib** to create bar, pie, and scatter charts including the line of regression. By leveraging these Pandas functions, we were able to preprocess the data, perform the necessary analysis, and present the results in a clear and accessible manner. This combination of data manipulation and visualization tools enabled us to uncover key insights about the IMDb dataset.

Which movie genre has the most number of votes?

After cleaning the dataset, we analyzed the IMDb data to visualize which genre received the most votes. To begin, we generated the list of genres, then applied the `unique()` method to extract all distinct genre entries from the dataset.

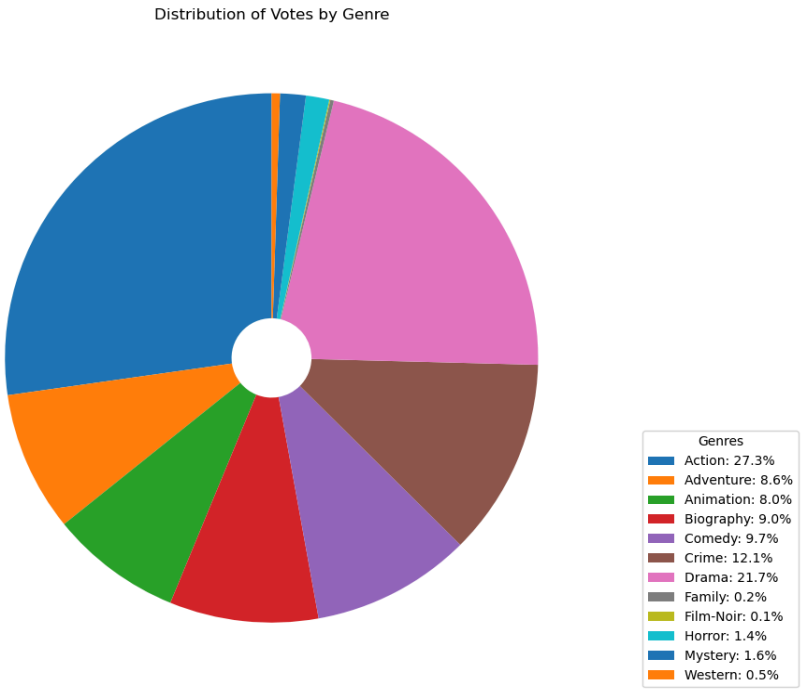
['Drama', 'Crime', 'Action', 'Biography', 'Western', 'Comedy', 'Adventure', 'Animation', 'Horror', 'Mystery', 'Film-Noir', 'Family'].

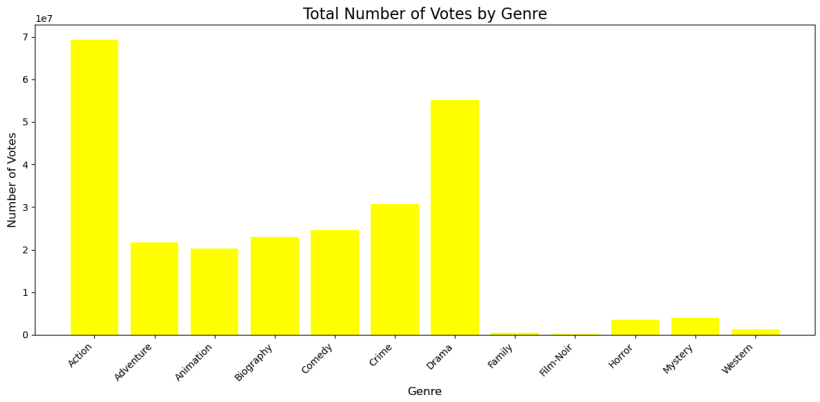
Next, we generated both a bar chart and a pie chart to illustrate the number of votes for each genre. The analysis revealed that **Action** was the most-voted genre, followed by **Drama** and **Crime**. In contrast, **Film-Noir**, **Family**, and **Western** genres received the fewest votes.

Donut chart displayed the different genres per number of votes. The single legend containing

both the genre names and their percentages enhances the chart's clarity. This avoids clutter while still providing detailed information about the percentage share of each genre. Bar chart on the other hand, compared the total number of votes across different genres. Each bar represents a genre, and the height of the bar indicates the number of votes it has received. Bar charts are one of the simplest and most effective ways to compare quantities across categories, which is why this visualization is useful in conveying genre-based data.

To gain a deeper understanding of the distribution of votes across genres, we also considered the broader trends within the dataset. The dominance of **Action**, **Drama**, and **Crime** genres in terms of vote counts likely reflects their widespread popularity and mainstream appeal. These genres tend to attract a larger, more diverse audience, which could explain their higher vote totals. In contrast, genres like **Film-Noir**, **Family**, and **Western** typically cater to more specialized audiences, leading to fewer votes. This variation may also be influenced by factors such as the historical context of the films, the changing popularity of certain genres over time, or the limited appeal of more niche genres. Visualizing these trends helps us comprehend the distribution of audience preferences across genres.





Which movie rating certificate has the highest average Metascore - in comparison to IMDb review scores?

Using the cleaned dataset, which included redefining outdated title certifications, we selected the relevant columns to create a double-bar graph for a side-by-side comparison. Specifically, we extracted the certification ratings, IMDb ratings, and Metascores to calculate their respective averages for each certificate category. Since Metascores are given on a 0-100 scale, we converted them to a comparable 0-10 scale to align with IMDb ratings.

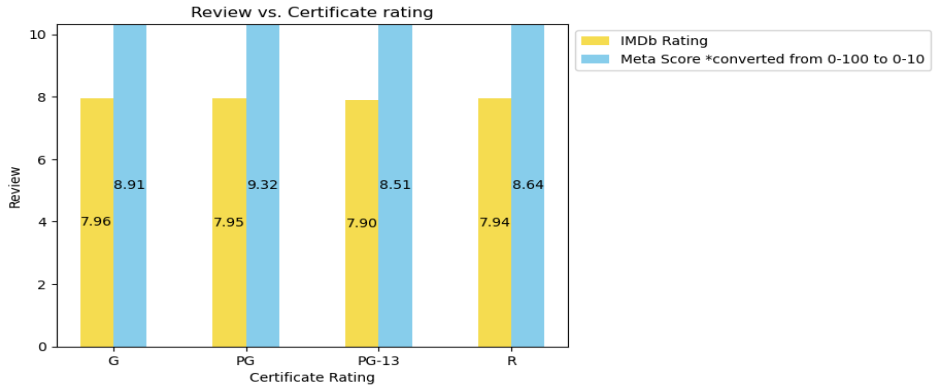
Reflected in the code below

```
# Group by the 'Rating' column
grouped = movie_ratings_df.groupby('Rating')

# Calculate the mean of 'IMBD average' for each group
average_rating = grouped['IMDB_Rating'].mean()

# Calculate the mean of 'Meta avergae' for each group
average_score = grouped['Meta_score'].mean()

# Convert Meta Average from 0-100 scale to 1-10 scale
average_score_1_to_10 = (average_score / 10) + 1

# Convert Meta Average from 0-100 scale to 1-10 scale
average_score_1_to_10 = [min((score / 10) +1, 10) for score in
average_score]

# Print Meta Conversion
print("Original Meta Scores:\n", average_score)
print("Converted Meta Scores (1-10 scale):\n", average_score_1_to_10)
```

```
#Print Mean of IMDb ratings
print(average_rating)
```

The analysis revealed that G-rated movies had the highest average IMDb rating of 7.96, although IMDb averages across all certificates were relatively close, hovering around 7.9. In contrast, PG-rated movies had the highest average Metascore at 9.32, showing slightly more variation between certificates (Converted Meta Scores, 0-10 scale: G: 8.9, PG: 9.32, PG-13: 8.51, R: 8.64). A legend was included to describe the Metascore conversion process and reflect the respective averages.

Interestingly, we found that more child-friendly ratings (G/PG) consistently received higher average scores in both categories. This was unexpected, given our assumption that most IMDb reviewers and critics would favor more mature-adult content. However, we did not account for the potential influence of adults with children among IMDb reviewers. Additionally, while IMDb ratings include the number of votes per movie, the Metascore panel size, which could range from 4+ critics, and its influence remains unclear.

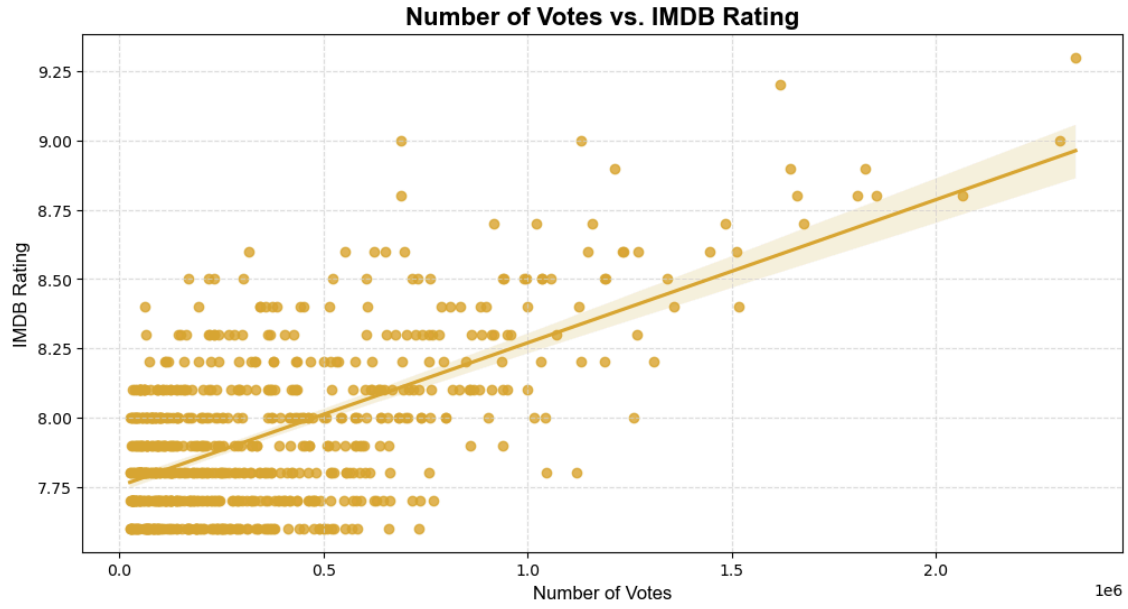


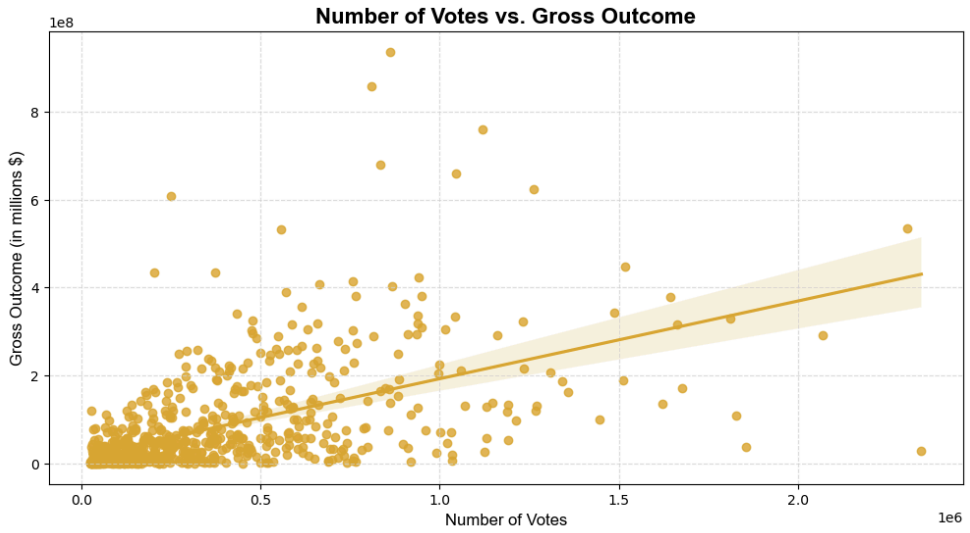
Does the number of votes affect the IMDb rating or gross outcome more?
We wanted to make sure that we ran a graph that showed the outcome to this question the best. A scatter plot was made for this comparison. A similar kind of code to the one below was used for both charts. The code used was:

```
# make a scatter plot
# Change Canvas Size
plt.figure(figsize=(12, 6))
# Create a basic plot
sns.regplot(data=df, x="No_of_Votes", y="IMDB_Rating", scatter=True,
fit_reg=True, color="goldenrod")
# Customizations
# Change colors
```

```
# Add in reference lines
# Add Labels/Title
plt.xlabel("Number of Votes (in millions)", fontfamily="Arial", fontsize=12)
plt.ylabel("IMDB Rating", fontfamily="Arial", fontsize=12)
plt.title("Number of Votes vs. IMDB Rating", fontweight="bold", fontsize=16,
fontfamily="Arial")
# Set X/Y Limits
#plt.ylim(10, 16)
#plt.xlim(0, 14)
# Add in a background grid
plt.grid(linestyle="--", color="lightgrey", alpha=0.75)
# Show/Save the Graph
plt.show()
```

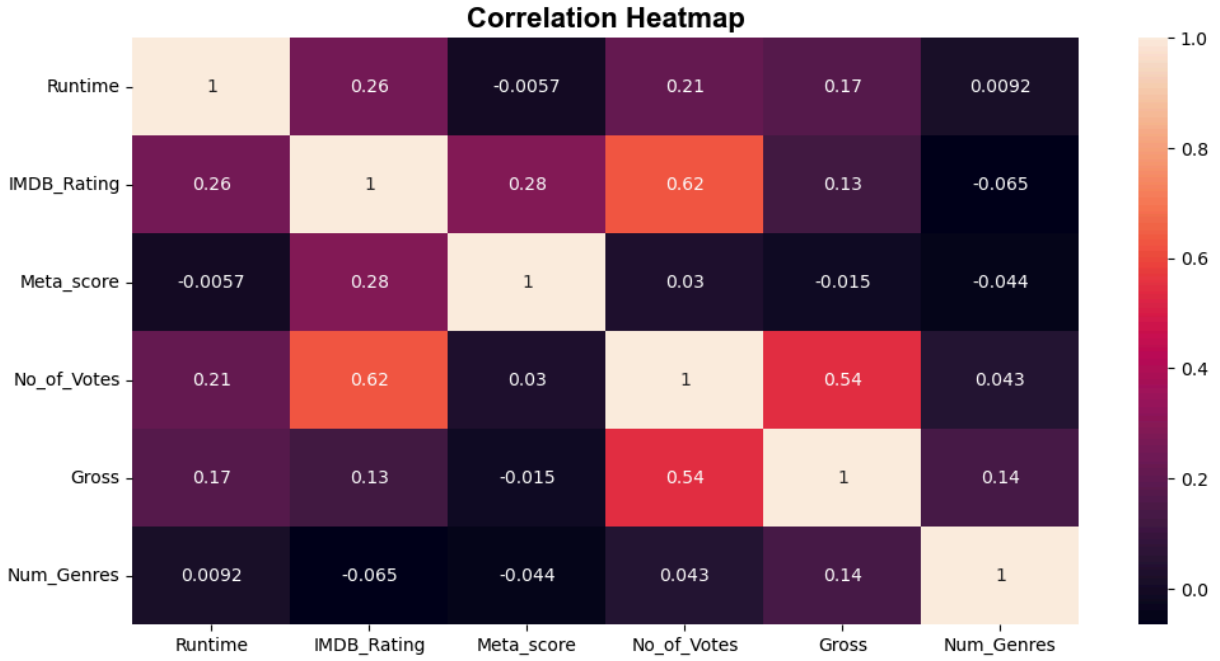
Using the scatter plot allowed us to see where each of the movies would fall in the comparison between number of votes and either IMDB rating or gross outcome. It appears that there is a stronger correlation between the number of votes and the IMDb ratings.That can be noticed because the regression line is positive and the plot points are grouped together at the beginning on the graph.



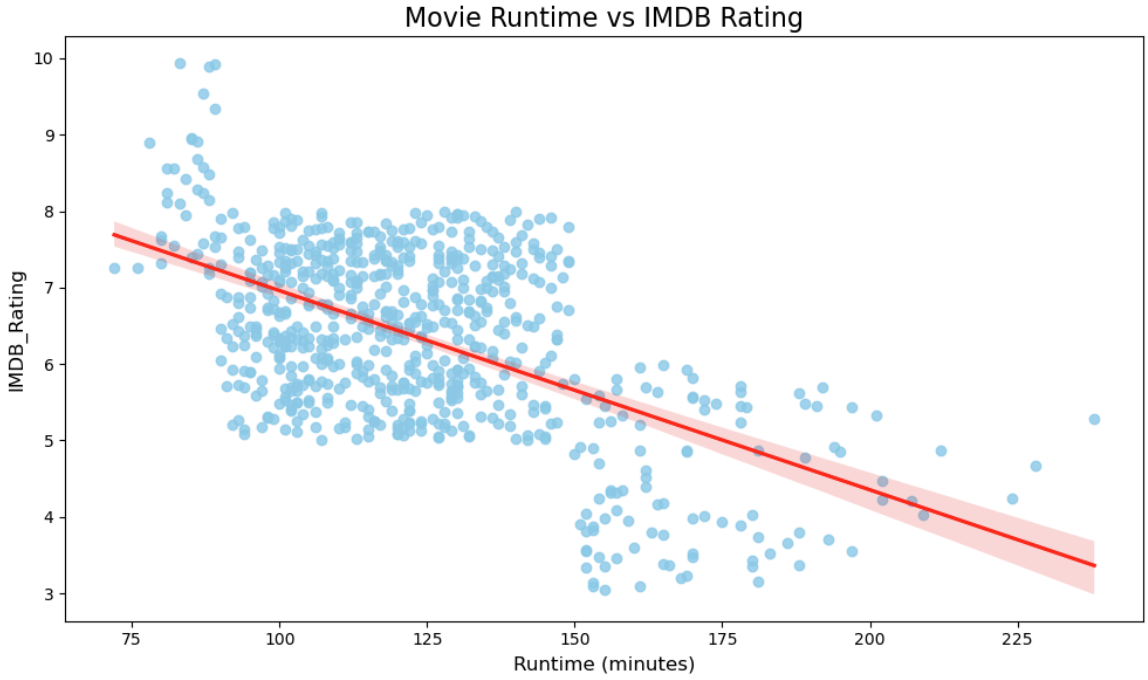


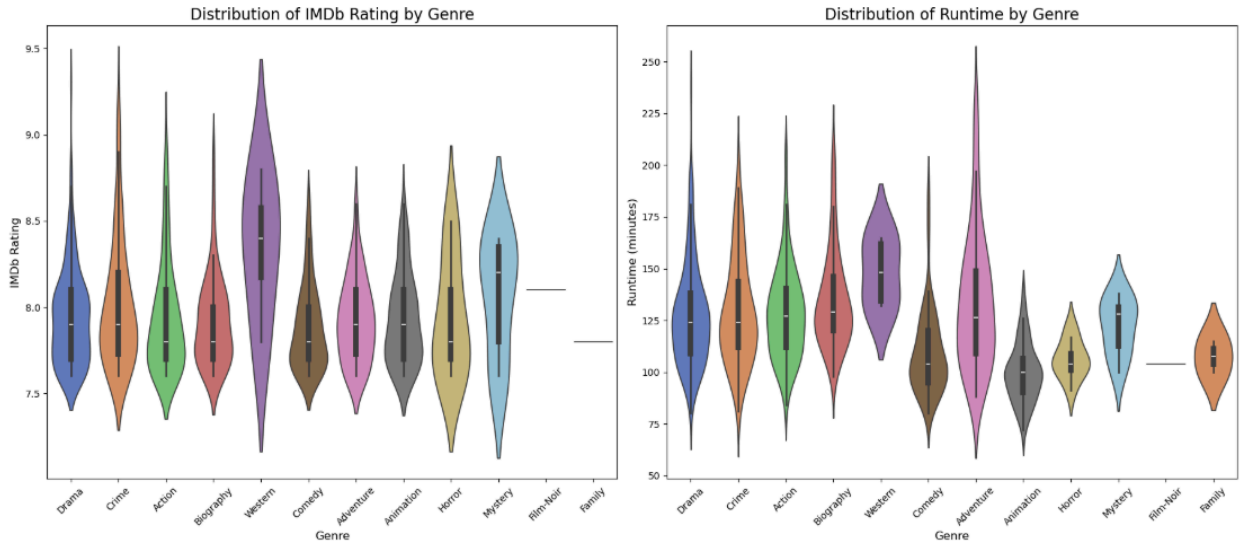
Once we created both of these graphs, we created a heatmap to confirm that our conclusions were accurate. When looking at this heatmap we can notice that the strongest correlation of this data is between the number of votes and the IMDb rating. The second highest would be between the number of votes and the gross outcome. This reinforces that the graphs we made up above are true. After further analysis of the heatmap, we also see that the largest negative correlation is between IMDb rating and the number of genres. That shows that the number of genres a movie has, has a negative correlation with the IMDb rating those movies have. The code below is what was used to create the heatmap:

```
# Change Canvas Size
plt.figure(figsize=(12, 6))

# Create a basic plot
sns.heatmap(corrs, annot=True)

# Customization
plt.title("Correlation Heatmap", fontweight="bold", fontsize=16,
fontfamily="Arial")

# Show/Save the Graph
plt.show()
```



In regards to IMDb ratings and runtime, there are some factors that need to be considered. In this Analysis, the movies that have the shorter runtime of under 150 minutes have a IMDb movie rating. Whereas, the movies that have the longer runtime of 150 minutes and beyond have resulted in lower IMDb movie ratings. Even though this Analysis is based off of over 700 movies, there will be a select few that will stand out as having a high runtime and a high IMDB movie rating and there are a select few with a low runtime and a low IMBb movie rating. For example, The Shawshank Redemption has the highest IMDb rating of all movies with a 9.3 with a runtime of 142 minutes. Also Searching has the lowest IMDb rating of 7.6 with a runtime of 102 minutes. However, there are other variables to recognize as well regarding runtime and IMDB rating. Gone with the Wind has the highest runtime of all movies in this dataset of 238 minutes, which means your day would have to be pretty free to watch this movie, but has a IMDB rating of 8.1. La planete sauvage has the lowest runtime of all movies of 72 minutes with a IMBb score of 7.8. Not to get off topic, but The Shawshank Redemption was directed by Frank Darabont who has been nominated for 30 awards and won 15 with notable movies such as The Green Mile, King Kong and A Nightmare on Elm Street 3 just to name a few. Whereas La Planete sauvage was directed by Rene Laloux who is a French animator and film director who was nominated for three awards and won three awards for older films. The demographic with these two directors is night and day which could've played a part in their individual recognitions. Overall, the difference in ratings between movies with short and long movies are statistically significant with the exception of a handful based on different scenarios and variables.





Bias & Limitations:

After reviewing all of our data and our data cleaning process we noticed that we had at least one bias and two limitations. The bias we noticed was an age bias on the IMDb ratings. IMDb ratings are published online by any registered user; however those are not a true representation of all movie watchers as younger children and older adults are less likely to post a review on the movies they watch. This is because the ability and access to the internet are needed to post a review and those groups.This could be a reason if G-rated, family, animated movies have low

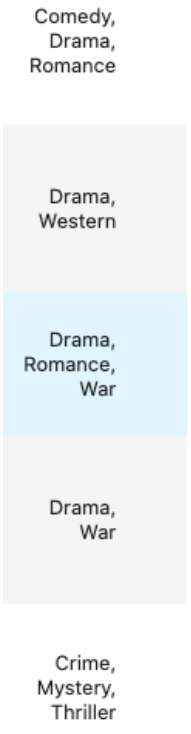
ratings, as the typical audience for those (young kids), probably do not rate movies as often as the typical audience for other types of movies.

The first limitation we noticed was when we were cleaning the certification ratings. The list of certification ratings were: U, A, UA, R, PG-13, PG, G, Passed, Approved, TV-PG, U/A, GP. There was no clear definition as to what the outdated movie rating would be equivalent to in today's movie ratings. We used our best judgement and the following code to combine the certification ratings:

```
df2["Certificate"] = df2.Certificate.replace({
    "U": "G",
    "A": "R",
    "UA": "PG-13",
    "U/A": "PG-13",
    "TV-PG": "PG",
    "GP": "PG",
    "Approved": "PG",
    "Passed": "PG",
})
```

We were able to narrow the certification ratings from 12 down to four common ones that are used today.

The second limitation we noticed was while we were cleaning the genre column. Originally the genre column listed out each genre that the movie was classified under (shown on the side). Since the data was a list with spaces and commas, we had to change it to make it easier to manipulate. We cleaned the column to only show the first genre listed by using the code below:




```
# Count the number of genres
df2['Num_Genres'] = df2['Genre'].str.split(',').str.len()

# Extract the first genre using split
df2['Genre'] = df2['Genre'].str.split(',').str[0]
df2.info()
```

Since we cleaned the data to only show the first genre, it limits our findings because when we say a movie is classified by a genre it would not give the full picture of the data as it does have the possibility to be classified as many genres.

Conclusion:
In summary, our exploration of the IMDb dataset revealed intriguing correlations between movie attributes and their reception by both audiences and critics. By investigating factors such as runtime, genre, vote counts, and certifications, we identified patterns that highlight how specific attributes correlate with higher IMDb ratings and Metascores. Action, Drama, and Crime emerged as dominant genres in popularity, while child-friendly certifications surprisingly garnered the highest average ratings across both metrics.

The data cleaning process, including the consolidation of outdated certifications and refinement of genres, enabled more streamlined analyses but also introduced limitations, such as the loss of multi-genre classifications. Furthermore, biases, particularly related to the age demographics of IMDb voters, were identified, emphasizing the need for caution in interpreting these results.

Despite these constraints, our findings provide valuable insights into audience and critic preferences, offering a foundation for further exploration into the dynamics of movie success. Our visualizations and insights shed light on patterns that connect audience preferences with critical reception. These results could guide filmmakers, marketers, and researchers in understanding the elements that resonate most with diverse audiences and drive positive reception in the competitive entertainment industry.

Citation:

IMDb.com. (n.d.). IMDb Help Center Ratings FAQ. IMDb. *https://help.imdb.com/article/imdb/track-movies-tv/ratings-faq/G67Y87TFYYP6TWAV?showReportContentLink=false#*

Shankhdhar, Harshit. “IMDB Movies Dataset.” *Kaggle*. www.kaggle.com/datasets/harshitshankhdhar/imdb-dataset-of-top-1000-movies-and-tv-shows.

https://chatgpt.com/