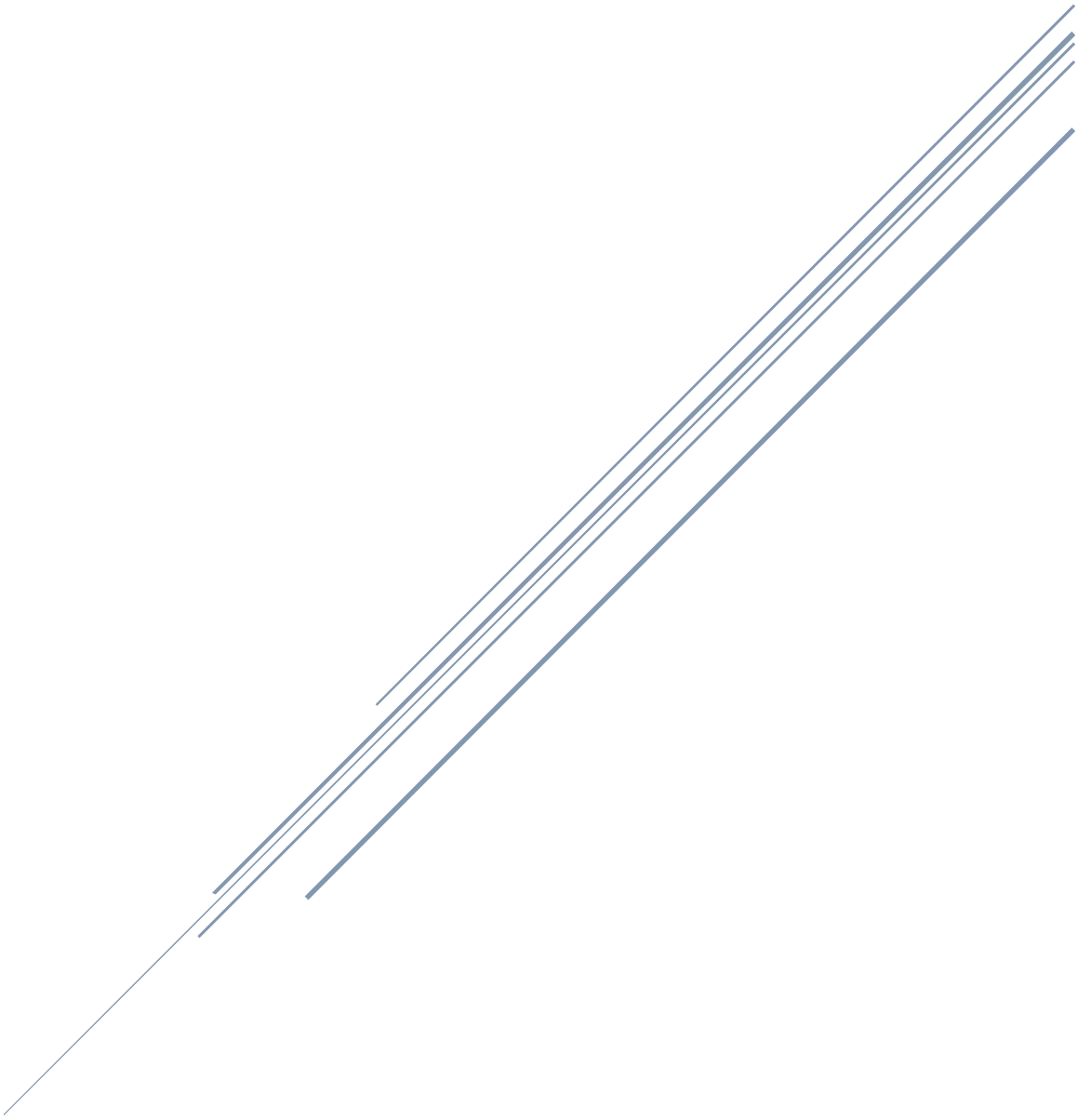
PROJECT REPORT

ANALYSIS IMDB TOP 1000 MOVIES



Kevin Kiding

Table Of Content

Chapter 1

Introduction

1.1. Background

According to Hornby (1995: 434), a movie, or film, is a story or narrative captured through a series of moving pictures intended for viewing on television or in cinemas. Microsoft Encarta (2008) similarly defines a movie as a sequence of projected images that creates the illusion of motion. It is considered one of the most popular forms of entertainment, transporting viewers into imaginary worlds (Microsoft Encarta: 2008). Coulson (1978: 622) adds that movies involve the recording of stories or events on film using moving pictures. Additionally, Lorimer (1995: 506) highlights that films can serve as cultural records, addressing social, political, and various societal issues that may be difficult to convey through other means.

According to the definitions offered, a movie or film is a kind of storytelling conveyed by a succession of moving photographs. It entails projecting pictures onto a screen to create the illusion of motion, allowing viewers to get immersed in various storylines and virtual worlds. Movies are a popular and frequently consumed type of entertainment.

Movies are recordings of tales or happenings that may be viewed on television or in theaters. They employ moving visuals to express narratives and can document cultural characteristics as well as address numerous social, political, and societal concerns. Films offer a unique platform for communication, capturing emotions, relationships, and experiences that may be difficult to communicate in other ways.

The IMDB Movies Dataset is a valuable resource for understanding the movie industry's impact on both businesses and individuals' daily lives. Movies serve as a prominent form of entertainment, inspiration, and cultural expression. IMDB, as a leading platform for movie ratings and information, provides a comprehensive dataset that allows us to analyze the characteristics and trends of the top one thousand movies based on IMDB ratings. This dataset offers valuable insights for movie-makers, producers, and distributors, helping them make informed decisions in areas such as movie production, genre selection, marketing, and distribution.

Additionally, movie enthusiasts and general audiences benefit from this dataset by gaining a deeper understanding of audience preferences, factors influencing movie ratings, and the influential directors shaping the industry. Analyzing the dataset fosters appreciation for the art of movie-making and enables individuals to discover movies aligned with their interests and preferences.

1.2. Project Objective

- Acquire a thorough comprehension of the distribution of movie ratings and audience perceptions as determined by IMDB ratings.
- Analyze the relationship between movie genres and IMDB ratings in order to identify genres with high commercial potential and audience appeal.
- Identify patterns and trends in the dataset pertaining to movie characteristics such as release year, length, and user ratings in order to inform strategic movie production and distribution decisions.
- Determine the most influential directors among the top one thousand movies and assess their impact on movie ratings in order to facilitate future collaborations and partnerships with prominent directors.
- Create visually engaging and impactful data visualizations to effectively communicate analysis findings to stakeholders, such as investors, distributors, and marketing teams, thereby facilitating informed decision-making in the movie industry.

1.3. Methods Applied

- The "IMDB Movies - Top 1000 Movies by IMDB Rating" dataset was obtained from Kaggle and other sources.
- The dataset's structure, attributes, and format were examined in order to obtain a comprehensive understanding of its contents.
- Data Cleaning and Transformation: Data cleaning procedures were executed, including the management of missing values, the removal of duplicates, and the conversion of data types as necessary.
- The distribution of IMDB ratings was examined in order to acquire insight into audience perceptions and preferences.
- The relationship between movie genres and IMDB ratings was investigated to determine commercially successful genres and audience preferences.
- Analysis of Movie Characteristics: Patterns and trends pertaining to movie characteristics, such as release year, length, and user ratings, were analyzed to inform movie production and distribution strategies.
- Directors of Influence: The 1000 most influential directors were identified, and their impact on movie ratings was evaluated to inform future collaborations and partnerships.
- Data Visualization: Visually appealing and persuasive data visualizations, such as bar charts, scatter plots, and heatmaps, were constructed to effectively convey the analysis's findings.

1.4. Benefits and Impacts

- Insights and Determination:
    - To assist the movie industry in making informed and strategic decisions, provide in-depth analysis of audience preferences and IMDB rating trends.
    - Determine factors contributing to a movie's success by analyzing the distribution of IMDB ratings and identifying patterns and preferences.
    - Examine the relationship between movie genres and IMDB ratings to determine genres with commercial success and audience appeal.
    - Examine movie characteristics such as release year, length, and user ratings to identify patterns and trends that can inform production and distribution strategies.
    - Identify the most influential directors of the top one thousand movies and assess their impact on movie ratings in order to inform future collaborations and partnerships.
- Audience Empowerment:
    - Provide information based on IMDB ratings to educate moviegoers about the factors influencing a movie's success.
    - Assist individuals in making educated decisions regarding which movies to view by analyzing the dataset to identify trends and patterns that correspond with their preferences.
    - To increase audience appreciation for the art of movie-making and industry recognition, the most influential directors from the top 1000 movies should be highlighted.
- Research and Resource:
    - Provide a comprehensive dataset for further research and analysis in the movie industry in order to serve as a valuable resource for researchers, movie professionals, and enthusiasts.
    - Permit stakeholders to conduct additional studies and investigate specific aspects of the dataset to obtain a deeper understanding of the dynamics of the movie industry.
    - Facilitate discussions, debates, and collaborations between movie enthusiasts, industry professionals, and researchers, thereby promoting knowledge sharing and advancing the understanding of the movie industry.

1.5. Resources Required

- "IMDB Movies - Top 1000 Movies by IMDB Rating" dataset from publicly available sources like Kaggle.
- Data analysis using the Python programming language and relevant libraries/modules.
- As the computing environment for data processing and analysis, Google Colab was utilized.
- Software for data visualization, such as Matplotlib or Seaborn, is used to create informative visualizations.
- Using word processing and presentation software to write the project report and present the results of the analysis.

Chapter 2

Data Acquisition and Preparation

2.1. Dataset Source

In this project, the dataset used is "IMDB Movies - Top 1000 Movies by IMDB Rating" which is taken from a publicly available source, Kaggle. This dataset contains information about the top 1000 movies by IMDB rating. This dataset is a CSV file that can be accessed through the following link: dataset link.

In order to start analyzing the data, the dataset was uploaded to the Google Colab environment using the following code:

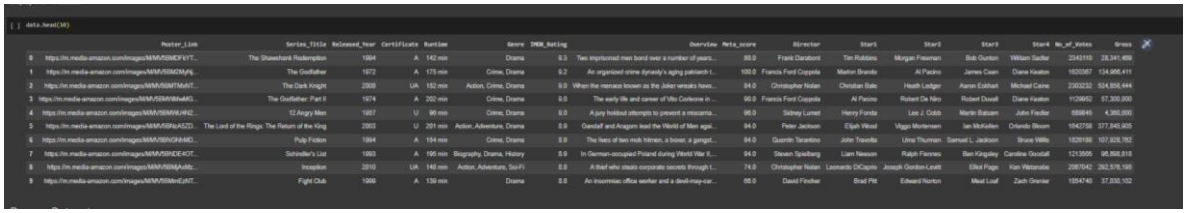
```
link = 'https://raw.githubusercontent.com/kevinnkevinn/data-analyst-portfolio/main/Top%201000%20Movie/imdb_top_1000.csv'
data = pd.read_csv(link)
```

In this segment, we read the dataset in CSV format and store it in data variables using the pandas library. This dataset will serve as a base for additional analysis of preferences and trends in the movie industry.

2.2. Dataset Overview

This project uses the "IMDB Movies - Top 1000 Movies by IMDB Rating" dataset. This information comprises movie title, release year, classification certificate, genre, duration, director, lead actor, and IMDB rating. Pandas on Google Colab retrieved and loaded the dataset. This project will analyze IMDB rating and runtime, genre, and director.

The following is the code to display the top 10 data from the dataset:



For more details, please click on this section.

Let's look for interesting things in the "IMDB Movies - Top 1000 Movies by IMDB Rating" dataset. This set of data is all about the top 1000 movies based on their IMDB grade. Let's look at the information in this file to see what's interesting and useful.

```
data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 16 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Poster_Link    1000 non-null   object 
 1   Series_Title   1000 non-null   object 
 2   Released_Year  1000 non-null   object 
 3   Certificate    899 non-null    object 
 4   Runtime        1000 non-null   object 
 5   Genre          1000 non-null   object 
 6   IMDB_Rating    1000 non-null   float64
 7   Overview       1000 non-null   object 
 8   Meta_score     843 non-null    float64
 9   Director       1000 non-null   object 
 10  Star1          1000 non-null   object 
 11  Star2          1000 non-null   object 
 12  Star3          1000 non-null   object 
 13  Star4          1000 non-null   object 
 14  No_of_Votes    1000 non-null   int64  
 15  Gross          831 non-null    object 
dtypes: float64(2), int64(1), object(13)
memory usage: 125.1+ KB
```

The "IMDB Movies - Top 1000 Movies by IMDB Rating" list shows that there are a few important things that need to be looked at. First of all, there are blanks in some columns, like "Certificate," "Meta_score," and "Gross." When more analysis data is done, these missing value could change how accurate and reliable the information is. Also, the types of data in some fields don't match up with the types of data which they should have. For example, the 'Released_Year' column should have a number or datetime data type instead of an object data type. In the same way, the 'Gross' column, which stores monetary amounts, should be saved as a number data type instead of an object data type. Fixing these problems is necessary to make sure that the information is correct and can be used for further research.

```
data = pd.DataFrame(data)


def convert_runtime(time):
    minutes = time.split(' ')[0]
    if len(minutes) == 2:
        minutes = '0' + minutes
    return minutes + ' min'

# Using a function to change the 'Runtime' column
data['Runtime'] = data['Runtime'].apply(convert_runtime)

data
```

A single line of code is given to answer these worries. In the code, the 'pd.DataFrame()' method is used to turn the 'data' DataFrame into a new DataFrame. Then, a method called 'convert_runtime' is made to change the numbers in the 'Runtime' column. This method splits the time value, pulls out the minutes, and, if required, adds a leading zero, so that the result is always in minutes. Lastly, the 'apply()' method is used to apply the 'convert_runtime' code to each value in the 'Runtime' column. This changes the column to show the new values. By running this code, the 'Runtime' field is standardised, which makes it possible to analyse the lengths of movie in a way that is true and consistent.

2.3. Data Cleaning and Transformation

Several steps were taken in the process of Dataset Cleaning and Transformation based on the code that was given. First, the missing values in the dataset were looked at by adding up all the missing values in each column and figuring out what percentage of the overall number of values were missing. Then, The data was put into a DataFrame called "missing_data."

The next step was to clean the dataset by removing rows with missing values using the dropna() method. This made a new dataset that was clean. The cleaned information was then saved to a new file called "cleaned_data.csv"

```
missing_values = data.isnull().sum()
total_values = len(data)
missing_percentage = (missing_values / total_values) * 100

missing_data = pd.DataFrame({'Jumlah Missing Value': missing_values, 'Persentase Missing Value': missing_percentage})
missing_data['Persentase Missing Value'] = missing_data['Persentase Missing Value'].map("{:.1f}%".format)

print(missing_data)
```

```
             Jumlah Missing Value Persentase Missing Value
Poster_Link                     0                    0.0%
Series_Title                    0                    0.0%
Released_Year                   0                    0.0%
Certificate                   101                   10.1%
Runtime                         0                    0.0%
Genre                           0                    0.0%
IMDB_Rating                     0                    0.0%
Overview                        0                    0.0%
Meta_score                    157                   15.7%
Director                        0                    0.0%
Star1                           0                    0.0%
Star2                           0                    0.0%
Star3                           0                    0.0%
Star4                           0                    0.0%
No_of_Votes                     0                    0.0%
Gross                         169                   16.9%
```

Drop missing value and create new dataset

```
# Drop rows with missing values
data_cleaned = data.dropna()

# Create and save the cleaned dataset to a new file
data_cleaned.to_csv('cleaned_data.csv', index=False)
```

Several column data types were modified to ensure that each column's data types were appropriate. The column 'Released_Year' has been converted to an object (string) data type, whereas the column 'Certificate' has been converted to a category data type. The 'Runtime', 'Gross', 'Meta_score', 'No_of_Votes', and 'IMDB_Rating' columns have been converted to string, float, numeric (integer), and string data types, respectively. The 'Gross' column was processed further by removing commas and other characters, and then converted to a float data type.

```
# Convert 'Released_Year' to object (string) data type
data['Released_Year'] = data['Released_Year'].astype(str)

# Convert 'Certificate' to category data type
data['Certificate'] = data['Certificate'].astype('category')

# Convert 'Runtime' to string data type
data['Runtime'] = data['Runtime'].astype(str)

# Convert 'Gross' to string data type
data['Gross'] = data['Gross'].astype(str)

# Remove commas and other characters from the 'Gross' column values
data['Gross'] = data['Gross'].str.replace(',', '').str.replace('$', '')

# Convert 'Gross' to float data type
data['Gross'] = pd.to_numeric(data['Gross'], errors='coerce')

# Convert 'Meta_score' to float data type
data['Meta_score'] = data['Meta_score'].astype(float)

# Convert 'No_of_Votes' to numeric (integer) data type
data['No_of_Votes'] = pd.to_numeric(data['No_of_Votes'], errors='coerce')

# Convert 'IMDB_Rating' to string data type
data['IMDB_Rating'] = data['IMDB_Rating'].astype(str)
```

Finally, the head of the modified dataset was displayed, providing a preview of the sanitised and transformed data and highlighting the modifications to the dataset's structure and data types.

2.4. Data Exploration

After data cleansing and transformation, the dataset was investigated further. The resultant sanitised dataset contains 714 rows and 16 columns. The columns include 'Poster_Link', 'Series_Title', 'Released_Year', 'Certificate', 'Runtime', 'Genre', 'IMDB_Rating', 'Overview', 'Meta_score', 'Director', 'Star1', 'Star2', 'Star3', 'Star4', 'No_of_Votes', and 'Gross'. The data types of the columns were modified to ensure consistency, with the 'Certificate' column now having the category data type, the 'Meta_score' column having the float64 data type, and the 'No_of_Votes' and 'Gross' columns having the int64 data type.

```
data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 714 entries, 0 to 713
Data columns (total 16 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Poster_Link    714 non-null    object
 1   Series_Title   714 non-null    object
 2   Released_Year  714 non-null    object
 3   Certificate    714 non-null    category
 4   Runtime        714 non-null    object
 5   Genre          714 non-null    object
 6   IMDB_Rating    714 non-null    object
 7   Overview       714 non-null    object
 8   Meta_score     714 non-null    float64
 9   Director       714 non-null    object
 10  Star1          714 non-null    object
 11  Star2          714 non-null    object
 12  Star3          714 non-null    object
 13  Star4          714 non-null    object
 14  No_of_Votes    714 non-null    int64
 15  Gross          714 non-null    int64
dtypes: category(1), float64(1), int64(2), object(12)
memory usage: 84.9+ KB
```

Analysing the 'Certificate' column revealed that it contains a variety of classification values for movies. The frequency distribution of the 'Certificate' column shows different classifications, such as 'U' (Universal), 'A' (Adult), 'UA' (Parental Guidance), 'R' (Restricted), 'PG-13' (Parental Guidance-13), 'G' (General Audience), 'Passed', 'Approved', 'GP' (General Audience - Parental Guidance Suggested), 'TV-PG' (Television - Parental Guidance), and 'U/A' (Universal with Adult). Each classification corresponds to a specific age restriction or audience recommendation.

Irrelevant columns were eliminated from the dataset utilising the provided code. Columns 'Poster_Link', 'Overview', 'Star1', 'Star2', 'Star3', and 'Star4' were eliminated from the dataset. These columns were regarded superfluous for the dataset's intended analysis and insights. Eliminating these columns simplifies the dataset so that it focuses on the most important attributes and factors influencing movie ratings and audience preferences, making it more concise and efficient for further analysis.

```
# Remove irrelevant columns
columns_to_drop = ['Poster_Link', 'Overview', 'Star1', 'Star2', 'Star3', 'Star4']
data.drop(columns_to_drop, axis=1, inplace=True)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 714 entries, 0 to 713
Data columns (total 10 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Series_Title   714 non-null    object
 1   Released_Year  714 non-null    object
 2   Certificate    714 non-null    category
 3   Runtime        714 non-null    object
 4   Genre          714 non-null    object
 5   IMDB_Rating    714 non-null    object
 6   Meta_score     714 non-null    float64
 7   Director       714 non-null    object
 8   No_of_Votes    714 non-null    int64
 9   Gross          714 non-null    int64
dtypes: category(1), float64(1), int64(2), object(6)
memory usage: 51.4+ KB
```

Chapter 3

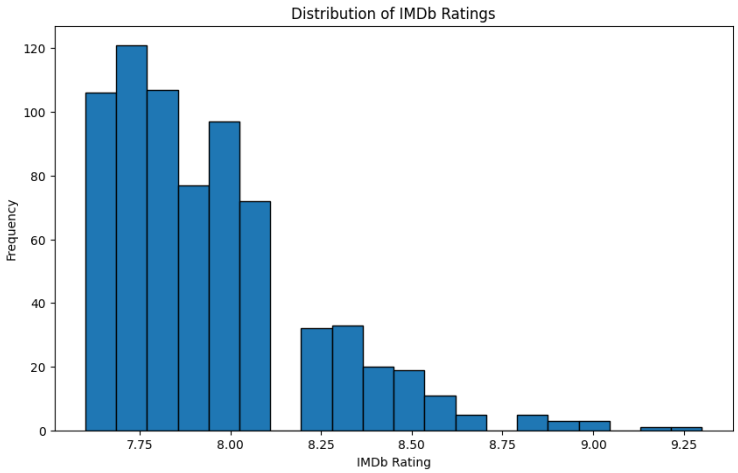
Analysis and Findings

3.1. Distribution of IMDB Ratings

The given code converts the dataset 'IMDB_Rating' column to float data type and then calculates the lowest and maximum IMDB ratings. To guarantee that the highest rating is within the modified range, an offset of 0.01 is added. The code then uses a histogram with 20 squares to illustrate the distribution of IMDb ratings. IMDb ratings are represented on the x-axis, the frequency of movies in each rating bin is represented on the y-axis, and the title of the plot is "IMDb Rating Distribution."

Looking at the distribution chart data, we can see that most of the movies in the IMDB Top 1000 data set have an IMDb rating of 7.6 to 8.5. The most common ratings were at 7.7, 7.8, and 8.0, with 121, 107, and 97 occurrences respectively. The frequency drops as the rating increases, indicating that the higher ratings are less prevalent. There are few films with ratings higher than 8.5, and the ratings 9.0, 9.2, and 9.3 have a very low frequency, with only one film reflecting each of those ratings.

The distribution of IMDb ratings on the IMDB Top 1000 Movie data set appears to represent a generally good rating of the films included. The fact that most of the ratings fell between 7.6 and 8.5 indicates that most of the films in the data set were well-received by audiences. The distribution also shows a progressive decrease in frequency as ratings increase, implying that getting higher ranks is both more difficult and rare. Overall, the distribution highlights the quality and acceptability of films in the IMDB Top 1000 data set.

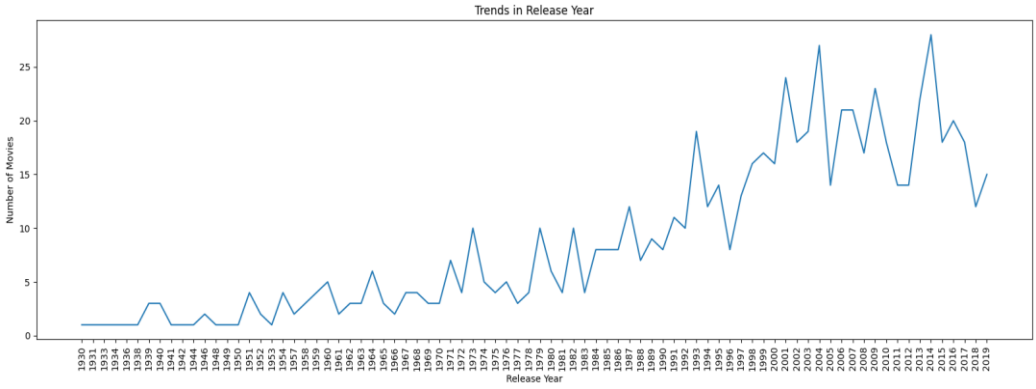


For a more clear view of the distribution chart, please click here.

3.2. Trend Analysis of Movie Release Year

The trend analysis of movie release years offers valuable insight into the distribution and development of film production over time. The dataset contains the number of films released in each year between 1930 and 2019. Throughout the years, the data exposes fascinating patterns and fluctuations in the film industry.

Looking at the data, we can see that early film production was comparatively low, with only a handful of films released each year. However, as the 20th century progresses towards its midpoint, the number of films produced increases gradually. This growth continues, with sporadic fluctuations, until the early 2000s, when the dataset reveals the greatest number of films released in a single year (27 in 2004). After that, there is a gradual decline in filmmaking, but the industry remains relatively stable.



For a more clear view of the distribution chart, please click here.

The dynamics of the movie industry can be better understood by examining the trends of the year of release. This provides findings on the ebb and flow of movie production over time.

The observed patterns can be influenced by factors such as technological advancements, shifting audience preferences, and economic factors. This dataset shows that the movie industry has experienced periods of growth and periods of stability, reflecting the dynamic nature of the movie industry.

The trend analysis of movie release years highlights the dynamic character of the entertainment industry from a broader perspective. It highlights the ongoing efforts of filmmakers and production companies to meet audience demand and adapt to changing market conditions. The analysis also illustrates the extensive history and cultural significance of movies as an artistic medium and popular entertainment medium.

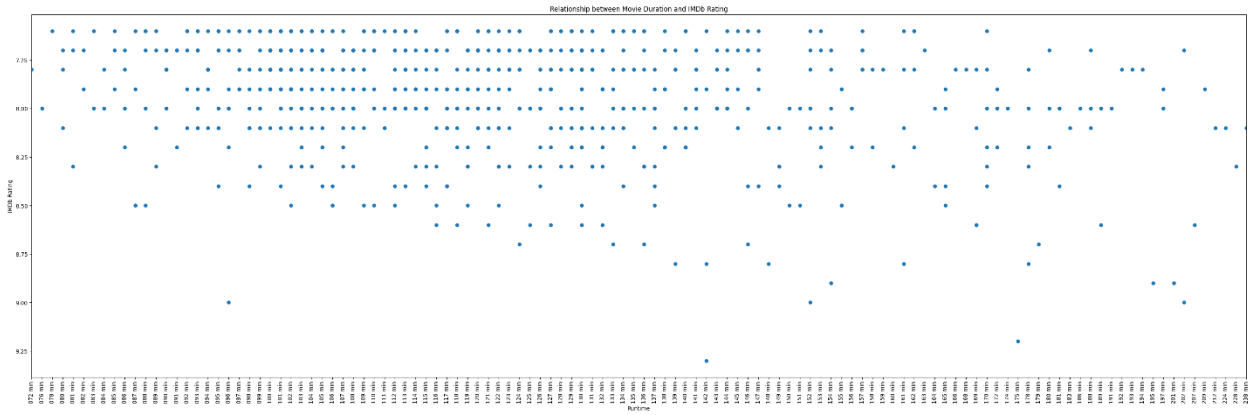
Overall, the trend analysis of film release years yields in-depth information about the evolution of the film industry. It explains the patterns and fluctuations of film production over time, demonstrating the industry's responsiveness to various influences. It demonstrates the prevalence and enduring relevance of film as a medium of storytelling and entertainment.

3.3. Relationship between IMDB Rating and Movie Duration

Using a scatterplot, the link between an IMDB rating and a movie's duration was examined. The dataset had a broad variety of movie lengths, from 72 minutes to 238 minutes, and IMDB ratings, which ranged from 7.6 to 9.3 (out of 10). The scatterplot, however, did not show a clear linear relationship between running time and IMDB rating. The graph's data point distribution showed no discernable trend, indicating that a movie's rating is not solely based on its running length. The ratings are probably influenced by a number of other elements, including the plot, the acting, and the production's overall quality.

While no clear linear pattern was found, it is still possible to find groups of data points that fall within certain movie length intervals and have comparable IMDB ratings. This implies that certain rating ranges can be associated with certain runtimes, although this relationship does not hold true for all movies. This scatterplot illustrates the need to consider various elements when assessing movie performance and ratings, as runtime alone is not a benchmark indicator for assessing movie quality.

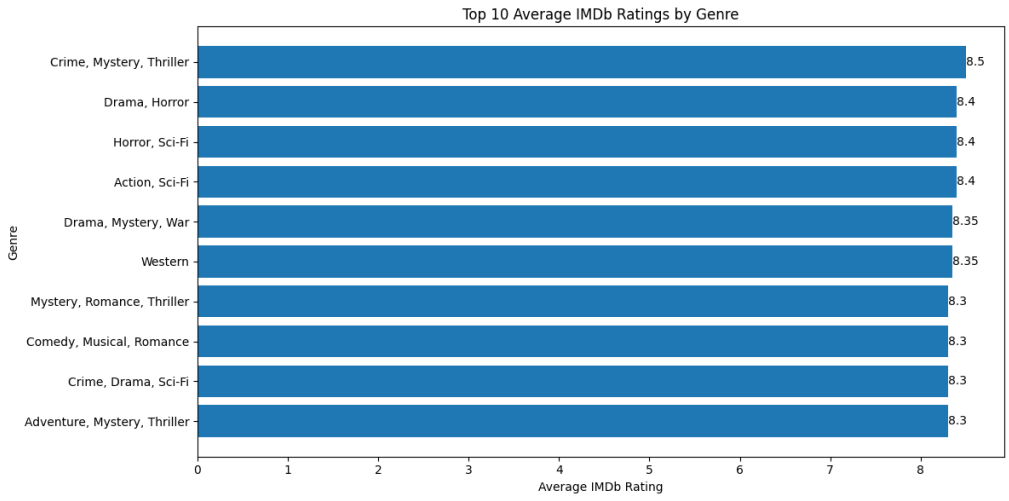
The scatterplot showing the correlation between IMDB rating and movie length, in conclusion, emphasises the variety of elements affecting a movie's rating. Although there isn't a clear linear relationship visible, the representation enables the identification of patterns and clusters in the data. This exemplifies both the subjective nature of audience choices and the complex nature of filmmaking. It emphasises the fact that factors other than just a movie's duration have an impact on IMDB ratings.



To see the chart more clearly, click here

3.4. Relationship between Movie Genre and IMDB Ratings

The information between various categories of film and the average rating received by each category on IMDB is emphasized by the presented information. The information includes average ratings for each category, which, according to IMDB, range from 7.60 to 8.40 (out of 10). The information is presented in the form of a vertical bar graph, where each category is denoted by a bar containing an average rating. The graph enables the comparison of ratings across different types of content.



In this graphic image, I have only shown the top 10 values. Please click here to see the chart more clearly.

The bar graph illustrates a distinct disparity in the average ratings across different movie genres. On IMDB, genres such as mystery, adventure, drama, and musical frequently receive

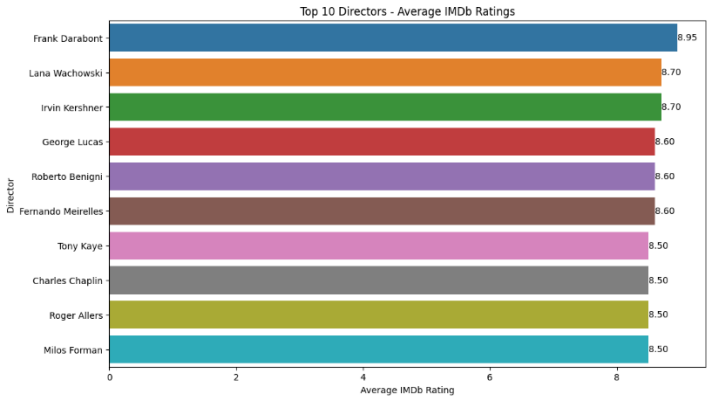
high ratings exceeding 8.0. However, genres like horror, western, science fiction, and action often have average ratings ranging from 7.6 to 8.4 out of 10. This data suggests that certain genres consistently achieve higher ratings compared to others, indicating diverse preferences and responses from both viewers and critics. Animation, drama, and biography genres appear to be more popular and tend to receive favorable reviews, while horror and science fiction genres often elicit mixed reactions.

It is important to note that the average ratings presented in the graph are subjective and influenced by individual viewers' preferences and perspectives. The IMDB rating takes into account various factors, including the strength of the storyline, the quality of acting performances, the directing, and the production value, which collectively contribute to the overall movie experience. Therefore, ratings are not solely determined by the movie's genre alone but rather a combination of different elements.

The bar graph effectively visualizes the relationship between movie genres and their average IMDB ratings, providing valuable insights into general audience perceptions and trends across various genres. However, it is crucial to recognize that individual interests and viewpoints may vary, and even within the same genre, movies can receive significantly different ratings. This data analysis and visualization can offer filmmakers, producers, and viewers a better understanding of potential audience reactions to movies of different genres. Nonetheless, it is essential to consider the subjective nature of ratings and the multitude of factors that contribute to a movie's overall quality and commercial success..

3.5. IMDB Rating Analysis of Movie Directors

A vertical bar graph illustrates the results of the study of the IMDB ratings for film directors. The graph illustrates the average IMDb rating for a variety of filmmakers, illustrating the wide range of ratings received by these directors. The data covers filmmakers with high average ratings on IMDB, such as Frank Darabont, Lana Wachowski, Irvin Kershner, and George Lucas. These directors' ratings range from 8.60 to 8.95 out of a possible 10. On the other side, filmmakers such as Ruben Fleischer, Joseph Kosinski, Jonathan Lynn, and Jonathan Levine have received average scores of 7.60 out of 10.



Due to the abundance of data, I have only displayed the top 10 charts. To view the entire graph, please click here.

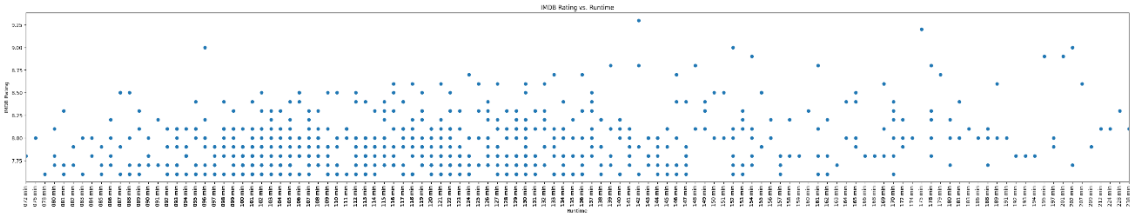
The bar graph illustrates changes in average scores among various directors, revealing disparities in viewer and reviewer reactions. High average ratings are routinely attained by directors like Frank Darabont and Lana Wachowski, indicating their capacity to make powerful and highly acclaimed films. Filmmakers like Joseph Kosinski and Ruben Fleischer, on the other hand, have lower average scores, reflecting a range of responses from viewers to their work. It is crucial to keep in mind that the IMDB average ratings shown in the graph are arbitrary and affected by different viewers' choices. Insights into the general opinion of various directors and how their average IMDB ratings contrast with one another may be gained by analysing the data and looking at the bar graph. Viewers, directors, and producers may find this information useful in determining how audiences may react to films made by various filmmakers.

It is important to recognise the arbitrary nature of ratings and the nuanced relationship between movie success and quality. The data given gives a thorough summary of the average IMDB ratings for film directors while showing the range of those ratings. The graph's visual portrayal, which highlights the contrasts between filmmakers who regularly obtain good ratings and those who don't, highlights the distinctions. This research offers insights into how consumers perceive films based on the filmmakers involved, which is useful for both professionals in the film industry and movie aficionados.

3.6. Relationship between Average IMDB Rating and Film Duration

The scatterplot graph illustrates the link that exists between the average rating on IMDB and the running time of the film. The length of the movie, measured in minutes, is shown along the x-axis, and the average rating, as given by IMDB, is displayed along the y-axis. Every single point on the graph represents a different movie.

The scatterplot reveals that there is no obvious linear connection between the length of the film and its average rating on IMDB. This can be seen by comparing the two variables. The points are dispersed around the graph, and there is no obvious pattern or trend to be seen. The fact that this is the case hints that the length of a movie is not the only factor that determines its average rating on IMDB.



To see a more detailed chart, click here

On the other hand, it is possible to see that films with relatively high average IMDB ratings are prevalent throughout a wide range of running times. For example, movies that get high scores ranging from 8.0 to 8.5 tend to be shorter in length (about 80-90 minutes). In a similar vein, movies that run for longer periods of time (about 140 to 160 minutes) tend to get higher ratings.

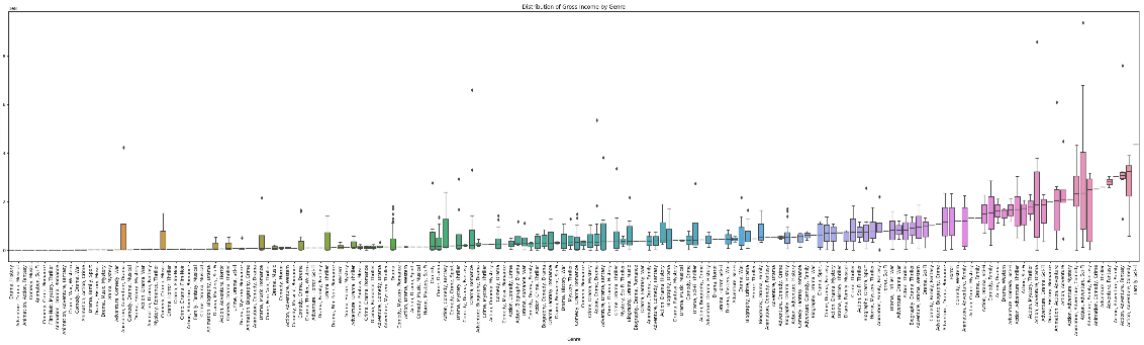
On the other side, films of varying lengths have been shown to have lower average reviews on IMDB. This suggests that the running time of the movie is not the only criterion that determines the rating. The average rating on IMDB is determined in large part by a number of other elements, including the quality of the plot, acting, and directing, as well as the overall production value.

3.7. Analysis of Movie Revenue Distribution by Genre

The information presented in the study concentrates on the distribution of movie earnings across various genres, with the median and average earnings for each category provided. The dataset contains a variety of genres, including drama, history, adventure, musical, animation, action, fantasy, comedy, music, sci-fi, crime, romance, film noir, and many more. The median gross income for each genre is given, ranging from $55,000 for "Drama, History" to $151,086 for "Animation, Action, Fantasy," among other categories.

The information offers perceptions on the usual gross earnings for various movie genres from an unbiased standpoint. A comprehension of the financial success of numerous categories is provided by the coupling of each genre with its associated median income. Outlier statistics, which show infrequent occurrences in the dataset that drastically depart from the main trend, must nonetheless be acknowledged. The report makes no mention of how to deal with these outliers explicitly.

The study provides a framework for analysing the financial success of various movie genres by highlighting the variances in median gross revenues. Directors, financiers, and other industry experts may find this information useful while deciding on the production, release, and marketing plans for a film.



In order to see the boxplot graph more clearly, please click here.

3.8. Correlation Analysis between Features

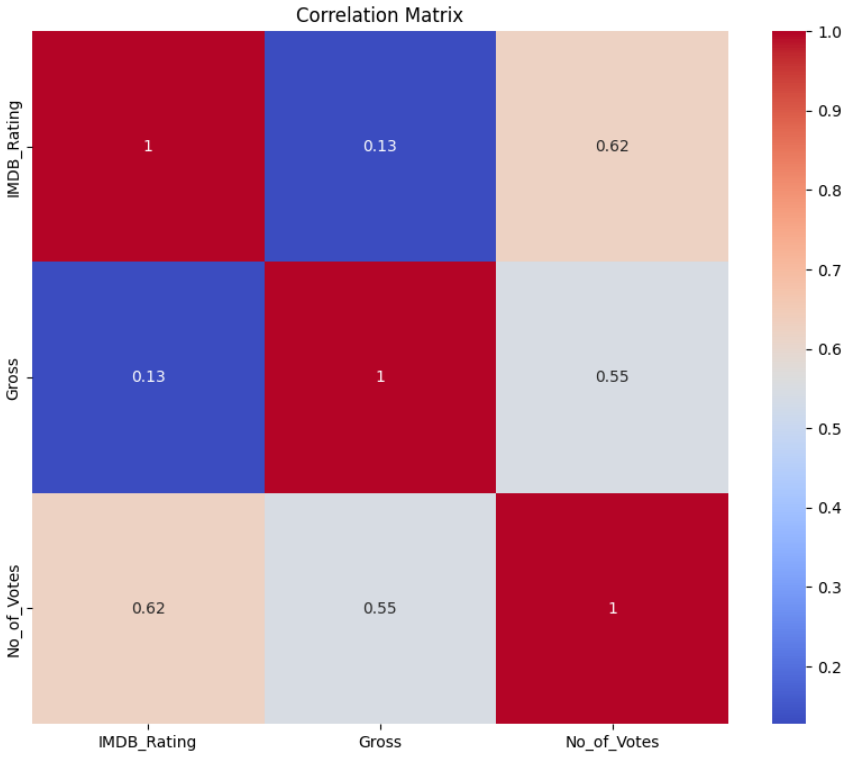
Correlation analysis investigates the strength and direction of correlations between various variables. Three parameters were examined in this study: IMDB rating, total revenue, and number of votes. The correlation values shed light on these connections.

The correlation coefficient of 0.13 reveals a modest relationship between IMDB rating and gross income. Although there is a minor positive link, implying that higher-rated films earn somewhat more money, the correlation is insufficient to draw clear conclusions regarding the relationship between the two criteria.

The correlation coefficient of 0.62 between IMDB rating and number of votes indicates a modest relationship. This suggests that films with higher ratings tend to earn more votes, indicating a greater correlation between a film's popularity as assessed by the number of votes and its IMDB rating.

Similarly, the correlation coefficient of 0.55 between gross income and the number of votes indicates a modest relationship. It implies that bigger-grossing films receive more votes, reflecting a higher level of audience interest and participation.

The study finds a modest link between IMDB rating and gross revenue, with a minor positive correlation. However, there are larger correlations between IMDB rating and number of votes, as well as gross income and number of votes. These studies shed light on how movie ratings, commercial performance, and spectator participation may all interact.


Correlation Matrix
IMDB_Rating
Gross
No_of_Votes
1
0.13
0.62
0.13
1
0.55
0.62
0.55
1
IMDB_Rating
Gross
No_of_Votes
1.0
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2

Chapter 4

Conclusion

4.1. Summary of Findings

1. IMDB Rating and Gross: The link between IMDB Rating and Gross is weak (0.13), which shows that better reviews and higher gross earnings have a small positive relationship. But the association isn't strong enough to say what the relationship between the two factors is for sure.
2. There is a moderately positive link (0.62) between the IMDB rating and the number of votes, which suggests that movies with better scores tend to get more votes. This shows that there is a strong link between how involved the crowd is and how the movie is rated.
3. Gross and Number of Votes: There is also a reasonable relationship between Gross and Number of Votes (0.55), which suggests that movies that make more money tend to get more votes. This shows that there is a link between how much money a movie makes and how interested the crowd is in it.
4. Movie Genre and Median Gross Income: The median gross income for each movie genre is different, with Adventure, Drama, Musical, Animation, Action, and Fantasy, in general, making more money than other genres. This information gives directors, managers, and marketers an idea of how different types of movies do financially, which can help them make decisions.

Overall, the research shows that the number of votes and IMDB scores are good indicators of how successful a movie will be. Even though there is less of a link between IMDB Rating and Gross, there are stronger links between IMDB Rating and Number of Votes and Gross and Number of Votes. These results show how important it is for the film business to think about what the audience wants and how involved they are.

4.2. Key Insights

1. Audience preferences and ratings: By analyzing IMDB ratings, it is possible to comprehend audience preferences and patterns. This data can assist the film industry in making strategic decisions and guiding production and distribution strategies.
2. Popular movie genres By analyzing IMDB ratings and movie genres, it is possible to identify popular genres and comprehend audience preferences. Typically, Animation, Adventure, Drama, Musical, and Mystery receive the highest average ratings.
3. Movie attributes and smart choices: Analyzing film characteristics such as release year, runtime, and user ratings can provide insights for making intelligent production and

distribution decisions. Understanding, for instance, the relationship between IMDB ratings and film length can aid in determining the optimal length for a film.

4. Influential filmmakers: The study identifies the most influential filmmakers from the top one thousand films and assesses their impact on the ratings. This information can be utilized to promote future partnerships and collaborations with renowned directors.
5. Data visualization for stakeholders: The project employs data visualization techniques such as bar charts, scatter graphs, and heatmaps to effectively communicate the conclusions of the analysis. This contributes to the education of film industry stakeholders, such as financiers, distributors, and marketing teams.
6. Film industry study and analysis: The dataset offers a comprehensive resource for researching and analyzing the film industry, to the benefit of academics, industry professionals, and moviegoers. It facilitates dialogues, debates, and collaborations among film enthusiasts, industry professionals, and academics to promote information exchange and comprehension of the dynamics of the film industry.
7. Audience Empowerment: By analyzing the dataset, individuals can discover films they enjoy and develop an appreciation for filmmaking. It assists movie enthusiasts and general audiences in discovering viewer preferences, ratings, and industry leaders, enabling them to make informed decisions when selecting movies to watch.

4.3. Recommendations

Identify genres with high commercial potential by analyzing the relationship between film genres and gross revenue to determine which genres typically perform well financially. Calculate the average revenue for each genre and emphasize those that consistently generate a high income. This data can assist in identifying genres with high commercial potential.

Explore audience appeal: Analyzing the relationship between IMDB ratings and movie genres to determine which genres are more likely to receive positive ratings from viewers. Determine the average IMDB rating for each genre and identify those that receive consistently high ratings. These disciplines are apt to appeal to a large audience.

Consider market trends: In addition to analyzing the relationship between genres and ratings/revenue, you may wish to incorporate market trends and audience preferences into your recommendations. Recent trends in the industry, box office successes, and popular genres should be investigated. This analysis will provide insight into the changing preferences of moviegoers and can assist in identifying genres that correspond to current market demands.

Discuss correlation analysis: Detail the correlation analysis performed between IMDB ratings, gross revenue, and the number of ballots. Explain the strengths and weaknesses of these correlations, emphasizing that even though the correlations are not particularly strong, they still offer valuable insights into the relationships between these factors. Discuss the correlation between higher ratings and increased earnings and viewer engagement.

Highlight data limitations: Recognize the limitations of the dataset utilized for the analysis. Mention the absence of data in certain fields and the need for accurate and dependable information to ensure the validity of the correlation analysis and the conclusions that follow. Address the data type inconsistencies and any other potential data quality issues identified during the process of data purification.

Provide visualizations: Include visualizations such as graphs and charts to support your findings and improve the legibility of your report. Consider constructing a scatter diagram to illustrate the correlation between runtime and IMDB rating. In addition, visualize the relationship between IMDB ratings and gross revenue, as well as the relationship between IMDB ratings and the number of ballots.

Conclusion and recommendations Provide a summary of your findings and recommendations based on your analysis. Based on correlation analysis, market trends, and current industry demands, identify the genres with high commercial potential and strong audience appeal. Suggest concentrating future film productions and investments on these genres.

Chapter 5

Project Design and Implementation

5.1. Software and Tools Used

The following applications and tools were used for analysis:

- Python: The provided code is written in Python, a widely-used programming language for data analysis and manipulation.
- Jupyter Notebook and Google Colab provide an interactive environment for executing Python code and displaying the output, respectively. Jupyter Notebook facilitates the execution and documentation of code, making it appropriate for data analysis initiatives.
- Pandas is a Python library used for manipulating and analyzing data. It offers data structures and functions for working efficiently with structured data, including importing datasets, performing data cleaning, managing missing values, and conducting exploratory data analysis.
- NumPy is a Python library essential for scientific computation. It provides support for enormous multidimensional arrays and matrices, as well as a set of mathematical functions for efficiently operating on these arrays.
- Seaborn and Matplotlib: Seaborn and Matplotlib are Python libraries for data visualization. Seaborn is built atop Matplotlib and provides a high-level interface for creating visually appealing and informative statistical graphics. Matplotlib is a robust plotting library that permits extensive visualization customization.

Python, Jupyter Notebook or Google Colab, Pandas, NumPy, Seaborn, and Matplotlib were used to import the dataset, investigate data information, sanitize and transform the data, analyze the relationships between variables, and visualize the results. The provided code demonstrates how these software and tools were used to perform data analysis tasks, including importing the dataset, exploring data information, managing missing values, converting data types, creating visualizations, and performing correlation analysis.

5.2. Data Analysis and Visualization Techniques

Data analysis and visualization are critical components of the data science and decision-making process, enabling businesses and organizations to gain important insights, spot trends, and effectively communicate results. Various statistical and analytical approaches are used in data analysis to transform raw data into relevant information. Data cleaning, organization, and processing are part of the process to eliminate errors and inconsistencies. Descriptive statistics summarize and explain data, but exploratory data analysis approaches, such as data

visualization, help in finding insights and patterns that may not be apparent from numerical summaries alone.

Data visualization is essential to displaying and sharing data successfully. Complex information can be portrayed in an easier and more accessible way by using charts, graphs, and maps. Visualization draws attention to patterns, trends, and relationships, which improves interpretation and understanding. These visualizations are especially useful for communicating large data sets or complex relationships that would be difficult to understand with tables or text alone. Bar charts, line charts, scatter plots, histograms, and heat maps are common visualization techniques tailored to specific types of data and research objectives. Visualizations should be clear, simple, and visually appealing, with appropriate colors, labels, and annotations to aid understanding.

Data analysis and visualization techniques are used in a variety of disciplines, including business, finance, healthcare, marketing, and social sciences. These techniques enable organizations to gain meaningful insights, make data-driven choices, and effectively communicate results to stakeholders. Analysts and decision-makers can use these approaches to examine complex data sets, find patterns, discover anomalies, and gain a better understanding. Overall, data analysis and visualization is a useful method for unearthing relevant information in data and encouraging effective communication, thereby aiding the evidence-based decision-making process.

5.3. Code Snippets and Explanations

1. Importing Libraries and Loading the Dataset.

This code excerpt imports the required data analysis and visualization libraries, including pandas, numpy, seaborn, and matplotlib. These libraries offer numerous functions and instruments for data manipulation, statistical analysis, and plotting. Using the pandas library's read_csv function, we then import the "IMDB Movies - Top 1000 Movies by IMDB Rating" dataset from the specified URL.

2. Exploring the Dataset and Cleaning Data

After the dataset has been loaded, we begin to investigate its contents. The data.head(10) command displays the initial 10 entries of a dataset to provide an overview of the data. The data.info() command provides a summary of the structure of the dataset, including the number of rows, columns, and data types for each column.

Next, we conduct data cleansing operations. Each value in the 'Runtime' column of the 'data' DataFrame is modified by the 'convert_runtime' function. This function converts the values at runtime to a uniform format. The dataset's missing values are then tallied, and the percentage of missing values for each column is calculated and displayed using the data.commands

isnull().sum() and missing_data. To assure the integrity of the data, rows with incomplete values are deleted and the cleaned dataset is saved to a new file named 'cleaned_data.csv'.

3. Analyzing and Visualizing the Data

The following code fragments concentrate on analyzing and displaying the dataset. We begin by analyzing the IMDb rating distribution using a histogram. Using the plt.hist function, the minimum and maximum IMDb ratings are computed, and a histogram with an adjusted range and specified segments is generated.

Counting the number of films released each year, we then analyze the trends in movie release years. To assure precise analysis, rows with 'PG' values in the 'Released_Year' column are removed. The resulting data is then plotted as a line chart to illustrate the trends in film releases over time.

Using a scatter diagram, we also examine the relationship between IMDb rating and film length. The dataset is queried for IMDb ratings and movie durations, which are then sorted based on durations. Using the plt.scatter function, we can visualize any prospective relationship between movie length and IMDb rating.

In addition, we investigate the correlation between film genres and IMDb evaluations. Using the data, the average IMDb rating for each genre is calculated.groupby('Genre')['IMDB_Rating'].mean() command, and the resulting horizontal bar chart is displayed.

Chapter 6
Limitations and Future Work

6.1. Limitations of the Analysis

Certain restrictions must be observed while analysing the IMDB Top 1000 Movie dataset. To begin, because the information is confined to the top 1000 films based on IMDB ratings, it may not reflect the whole movie business. This may result in a bias towards highly rated films and may not adequately reflect the range of films across genres and historical periods. Second, the existence of missing or incomplete data might have an impact on the analysis's correctness and dependability. While attempts were attempted to rectify missing information, the findings should be interpreted with caution.

Another thing to keep in mind is that while correlation analysis exposes correlations between variables, it does not suggest causality. The correlation coefficients show the intensity and direction of links, but further elements and study are needed to completely understand the underlying causes of these interactions. Furthermore, the temporal element of the data should be considered. The dataset contains films from several years, and trends and patterns may have altered over time. The report may not accurately reflect the current situation of the film business or subsequent advancements.

Finally, while the IMDB Top 1000 Movie dataset study gives useful insights, it is critical to be mindful of its limits. These include concerns with the dataset's representativeness, missing data, the nature of correlation analysis, and the data's temporal dimension. Considering these constraints can help you analyse the data more accurately and avoid jumping to conclusions. Exploring new data sources and undertaking more research can offer a more complete picture of the film business.

6.2. Areas for Future Exploration and Improvement

In the examination of the IMDB Top 1000 Movie dataset, there are various areas for further investigation and enhancement. To begin, including other variables such as budget, production company, and genre-specific characteristics might offer a more complete knowledge of the aspects driving movie ratings and financial success. It would be feasible to discern distinct patterns and trends within different genres or production environments by taking these extra variables into account.

Furthermore, a more in-depth examination of audience demographics and preferences might give useful information. Investigating the link between movie ratings and demographics such as viewer age, gender, or geographical region might give a better understanding of the

influence of a varied audience on movie popularity. This might assist filmmakers and producers in tailoring their material to certain target audiences while also improving their marketing efforts.

In terms of improvement, more comprehensive and up-to-date datasets should be collected. Extending the dataset beyond the top 1000 films and encompassing a broader range of time periods and genres would offer a more accurate representation of the film industry. Furthermore, for reliable analysis, data accuracy and missing data concerns must be addressed. Future analyses can provide more robust and informative conclusions by increasing data quality and combining more diverse and broad information, allowing for a better knowledge of the dynamics of the film business.

Chapter 7

References

Anggraeni, P., Mujiyanto, J., & Sofwan, A. (Year). The implementation of transposition translation procedures in English-Indonesian translation of epic movie subtitle. English Department, Faculty of Languages and Arts, Universitas Negeri Semarang, Indonesia.

Coulson. (1978). Title of Coulson's work. (p. 622).

Harshit Shankhdhar. (n.d.). IMDb Dataset of Top 1000 Movies and TV Shows. Retrieved from Kaggle: https://www.kaggle.com/harshitshankhdhar/imdb-dataset-of-top-1000-movies-and-tv-shows

Hornby, A.S. (2010). Equivalence. In Oxford Advanced Learner's Dictionary (8th ed., p. 495). Oxford: Oxford University Press.

Lorimer. (1995). Title of Lorimer's work. (p. 506).

Microsoft Corporation. (2008). Microsoft Encarta.

Chapter 8

Appendics

8.1. Detailed Data Cleaning Steps

- Import library package:

```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

- Load dataset:

```
link = 'https://raw.githubusercontent.com/kevinnkevinn/data-analyst-portfolio/main/Top%201000%20Movie/imdb_top_1000.csv'
data = pd.read_csv(link)
```

- Display the first 10 rows of dataset:

```
data.head(10)
```

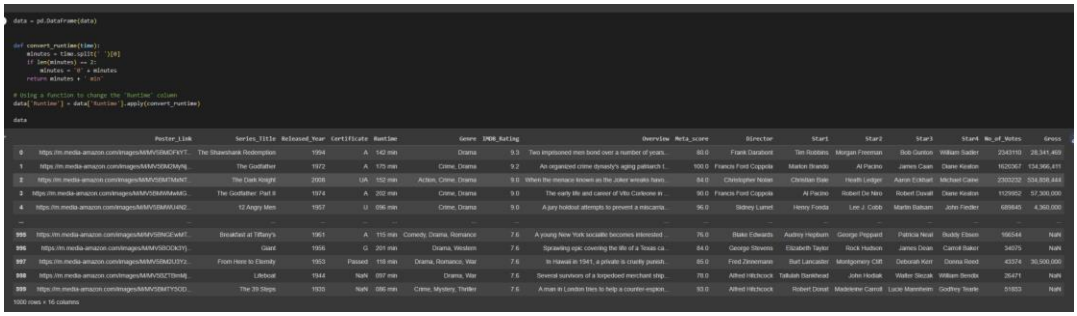
	Poster_Link	Series_Title	Released_Year	Certificate	Runtime	Genre	IMDB_Rating	Overview	Meta_score	Director	Star1	Star2	Star3	Star4	No_of_Votes	
0	https://m.media-amazon.com/images/M/MV5BMDFkYT...	The Shawshank Redemption	1994	A	142 min	Drama	9.3	Two imprisoned men bond over a number of years...	80.0	Frank Darabont	Tim Robbins	Morgan Freeman	Bob Gunton	William Sadler	2343110	28,3
1	https://m.media-amazon.com/images/M/MV5BM2MyNj...	The Godfather	1972	A	175 min	Crime, Drama	9.2	An organized crime dynasty's aging patriarch t...	100.0	Francis Ford Coppola	Marlon Brando	Al Pacino	James Caan	Diane Keaton	1620367	134,9
2	https://m.media-amazon.com/images/M/MV5BMTMxNT...	The Dark Knight	2008	UA	152 min	Action, Crime, Drama	9.0	When the menace known as the Joker wreaks havo...	84.0	Christopher Nolan	Christian Bale	Heath Ledger	Aaron Eckhart	Michael Caine	2303232	534,8
3	https://m.media-amazon.com/images/M/MV5BMWMwMG...	The Godfather: Part II	1974	A	202 min	Crime, Drama	9.0	The early life and career of Vito Corleone in ...	90.0	Francis Ford Coppola	Al Pacino	Robert De Niro	Robert Duvall	Diane Keaton	1129952	57,3
4	https://m.media-amazon.com/images/M/MV5BMWU4N2...	12 Angry Men	1957	U	96 min	Crime, Drama	9.0	A jury holdout attempts to prevent a miscarria...	96.0	Sidney Lumet	Henry Fonda	Lee J. Cobb	Martin Balsam	John Fiedler	689845	4,3
5	https://m.media-amazon.com/images/M/MV5BNzA5ZD...	The Lord of the Rings: The Return of the King	2003	U	201 min	Action, Adventure, Drama	8.9	Gandalf and Aragorn lead the World of Men agai...	94.0	Peter Jackson	Elijah Wood	Viggo Mortensen	Ian McKellen	Orlando Bloom	1642758	377,8
6	https://m.media-amazon.com/images/M/MV5BNGNkMD...	Pulp Fiction	1994	A	154 min	Crime, Drama	8.9	The lives of two mob hitmen, a boxer, a gangst...	94.0	Quentin Tarantino	John Travolta	Uma Thurman	Samuel L. Jackson	Bruce Willis	1826188	107,9
7	https://m.media-amazon.com/images/M/MV5BNDE4OT...	Schindler's List	1993	A	195 min	Biography, Drama, History	8.9	In German-occupied Poland during World War II,...	94.0	Steven Spielberg	Liam Neeson	Ralph Fiennes	Ben Kingsley	Caroline Goodall	1213505	96,8
8	https://m.media-amazon.com/images/M/MV5BMjAxMz...	Inception	2010	UA	148 min	Action, Adventure, Sci-Fi	8.8	A thief who steals corporate secrets through t...	74.0	Christopher Nolan	Leonardo DiCaprio	Joseph Gordon-Levitt	Elliot Page	Ken Watanabe	2067042	292,5
9	https://m.media-amazon.com/images/M/MV5BMmEzNT...	Fight Club	1999	A	139 min	Drama	8.8	An insomniac office worker and a devil-may-car...	66.0	David Fincher	Brad Pitt	Edward Norton	Meat Loaf	Zach Grenier	1854740	37,0

- Explore dataset information:

```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 16 columns):
 #   Column         Non-Null Count  Dtype  
---  ------         --------------  -----  
 0   Poster_Link    1000 non-null   object 
 1   Series_Title   1000 non-null   object 
 2   Released_Year  1000 non-null   object 
 3   Certificate    899 non-null    object 
 4   Runtime        1000 non-null   object 
 5   Genre          1000 non-null   object 
 6   IMDB_Rating    1000 non-null   float64
 7   Overview       1000 non-null   object 
 8   Meta_score     843 non-null    float64
 9   Director       1000 non-null   object 
 10  Star1          1000 non-null   object 
 11  Star2          1000 non-null   object 
 12  Star3          1000 non-null   object 
 13  Star4          1000 non-null   object 
 14  No_of_Votes    1000 non-null   int64  
 15  Gross          831 non-null    object 
dtypes: float64(2), int64(1), object(13)
memory usage: 125.1+ KB
```

- Convert the 'Runtime' column in the 'data' DataFrame by applying the 'convert_runtime' function to each value and assign the modified column back to 'data':



- Count missing values:

```
missing_values = data.isnull().sum()
total_values = len(data)
missing_percentage = (missing_values / total_values) * 100

missing_data = pd.DataFrame({'Jumlah Missing Value': missing_values, 'Persentase Missing Value': missing_percentage})
missing_data['Persentase Missing Value'] = missing_data['Persentase Missing Value'].map("{:.1f}%".format)

print(missing_data)
```

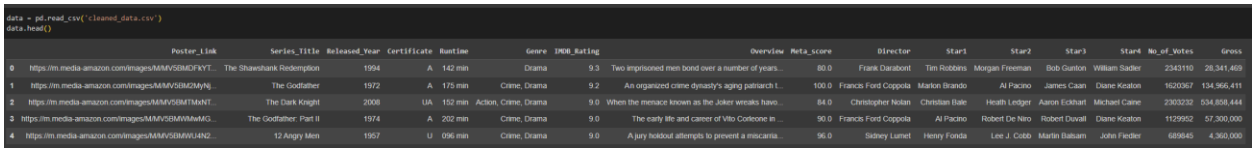
```
              Jumlah Missing Value Persentase Missing Value
Poster_Link                      0                     0.0%
Series_Title                     0                     0.0%
Released_Year                    0                     0.0%
Certificate                    101                    10.1%
Runtime                          0                     0.0%
Genre                            0                     0.0%
IMDB_Rating                      0                     0.0%
Overview                         0                     0.0%
Meta_score                     157                    15.7%
Director                         0                     0.0%
Star1                            0                     0.0%
Star2                            0                     0.0%
Star3                            0                     0.0%
Star4                            0                     0.0%
No_of_Votes                      0                     0.0%
Gross                          169                    16.9%
```

- Drop missing values and create a new dataset:

```
# Drop rows with missing values
data_cleaned = data.dropna()

# Create and save the cleaned dataset to a new file
data_cleaned.to_csv('cleaned_data.csv', index=False)
```

- Check the new dataset:



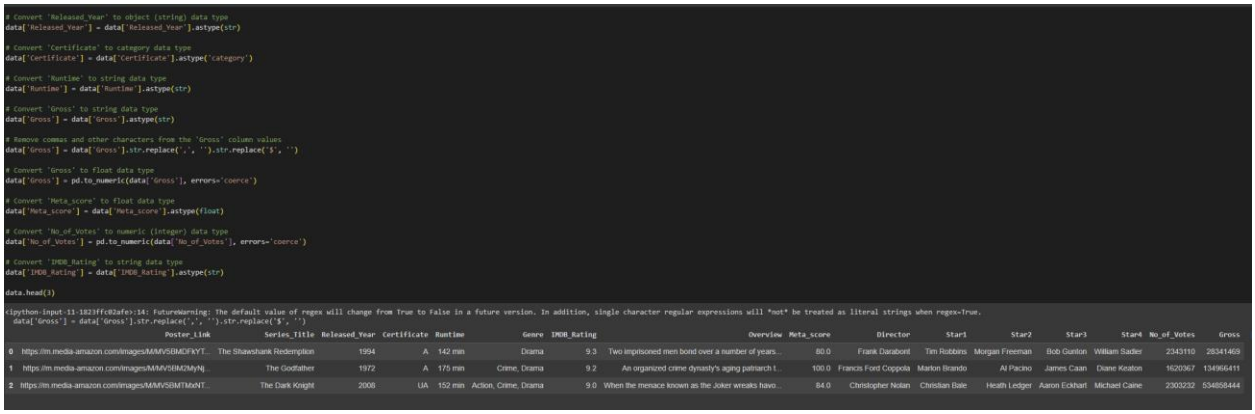
- Check for duplicate entries:

```
# Check for duplicate rows
duplicate_rows = data[data.duplicated()]

# Print the duplicate rows, if any
if len(duplicate_rows) > 0:
    print("Duplicate rows:")
    print(duplicate_rows)
else:
    print("No duplicate rows found.")
```

```
No duplicate rows found.
```

- Customize the columns data type:



```
# Convert 'Released_Year' to object (string) data type
data['Released_Year'] = data['Released_Year'].astype(str)

# Convert 'Certificate' to category data type
data['Certificate'] = data['Certificate'].astype('category')

# Convert 'Runtime' to string data type
data['Runtime'] = data['Runtime'].astype(str)

# Convert 'Gross' to string data type
data['Gross'] = data['Gross'].astype(str)

# Remove commas and other characters from the 'Gross' column values
data['Gross'] = data['Gross'].str.replace(',', '').str.replace('$', '')

# Convert 'Gross' to float data type
data['Gross'] = pd.to_numeric(data['Gross'], errors='coerce')

# Convert 'Meta_score' to float data type
data['Meta_score'] = data['Meta_score'].astype(float)

# Convert 'No_of_Votes' to numeric (integer) data type
data['No_of_Votes'] = pd.to_numeric(data['No_of_Votes'], errors='coerce')

# Convert 'IMDB_Rating' to string data type
data['IMDB_Rating'] = data['IMDB_Rating'].astype(str)

data.head(3)
```

```
<ipython-input-11-1823ffc02afe>:14: FutureWarning: The default value of regex will change from True to False in a future version. In addition, single character regular expressions will *not* be treated as literal strings when regex=True.
  data['Gross'] = data['Gross'].str.replace(',', '').str.replace('$', '')
```

	Poster_Link	Series_Title	Released_Year	Certificate	Runtime	Genre	IMDB_Rating	Overview	Meta_score	Director	Star1	Star2	Star3	Star4	No_of_Votes	Gross
0	https://m.media-amazon.com/images/M/MV5BMDFkYT...	The Shawshank Redemption	1994	A	142 min	Drama	9.3	Two imprisoned men bond over a number of years...	80.0	Frank Darabont	Tim Robbins	Morgan Freeman	Bob Gunton	William Sadler	2343110	28341469
1	https://m.media-amazon.com/images/M/MV5BM2MyNj...	The Godfather	1972	A	175 min	Crime, Drama	9.2	An organized crime dynasty's aging patriarch t...	100.0	Francis Ford Coppola	Marlon Brando	Al Pacino	James Caan	Diane Keaton	1620367	134966411
2	https://m.media-amazon.com/images/M/MV5BMTMxNT...	The Dark Knight	2008	UA	152 min	Action, Crime, Drama	9.0	When the menace known as the Joker wreaks havo...	84.0	Christopher Nolan	Christian Bale	Heath Ledger	Aaron Eckhart	Michael Caine	2303232	534858444

- Delete any unnecessary columns that are irrelevant to the analysis

```
# Remove irrelevant columns
columns_to_drop = ['Poster_Link', 'Overview', 'Star1', 'Star2', 'Star3', 'Star4']
data.drop(columns_to_drop, axis=1, inplace=True)
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 714 entries, 0 to 713
Data columns (total 10 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Series_Title   714 non-null    object
 1   Released_Year  714 non-null    object
 2   Certificate    714 non-null    category
 3   Runtime        714 non-null    object
 4   Genre          714 non-null    object
 5   IMDB_Rating    714 non-null    object
 6   Meta_score     714 non-null    float64
 7   Director       714 non-null    object
 8   No_of_Votes    714 non-null    int64
 9   Gross          714 non-null    int64
dtypes: category(1), float64(1), int64(2), object(6)
memory usage: 51.4+ KB
```

- Analysis data
  - Distribution of IMDB Ratings:

```
# Convert 'IMDB_Rating' to float data type
data['IMDB_Rating'] = data['IMDB_Rating'].astype(float)

# Calculate the minimum and maximum IMDb ratings
min_rating = data['IMDB_Rating'].min()
max_rating = data['IMDB_Rating'].max()

# Add a small offset to the maximum rating
max_rating += 0.01

# Plot the distribution of IMDb ratings with adjusted range
plt.figure(figsize=(10, 6))
plt.hist(data['IMDB_Rating'], bins=20, edgecolor='black', range=(min_rating, max_rating))
plt.xlabel('IMDb Rating')
plt.ylabel('Frequency')
plt.title('Distribution of IMDb Ratings')

# Show the plot
plt.show()
```

  - Trend analysis of movie release year:

```
# Filter out rows with 'PG' value in the 'Released_Year' column
data = data[data['Released_Year'] != 'PG']

# Analyze trends in release year after removing 'PG' rows
release_year_counts = data['Released_Year'].value_counts().sort_index()

plt.figure(figsize=(20, 6))
plt.plot(release_year_counts.index, release_year_counts.values)
plt.xlabel('Release Year')
plt.ylabel('Number of Movies')
plt.title('Trends in Release Year')
plt.xticks(rotation=90)
plt.show()
```

- Analysis of the Relationship between IMDb Rating and Movie Duration:

```
# Retrieve IMDb Rating and Movie Duration columns
imdb_ratings = data['IMDB_Rating']
durations = data['Runtime']

# Sort durations and imdb_ratings based on durations
sorted_indices = durations.argsort()
sorted_durations = durations.iloc[sorted_indices]
sorted_imdb_ratings = imdb_ratings.iloc[sorted_indices]

plt.figure(figsize=(40, 12))
plt.scatter(sorted_durations, sorted_imdb_ratings)
plt.xlabel('Runtime')
plt.ylabel('IMDb Rating')
plt.title('Relationship between Movie Duration and IMDb Rating')

# Memperbalik urutan nilai pada sumbu y
plt.gca().invert_yaxis()
plt.xticks(rotation=90)

# Sort the x-axis values in ascending order
plt.gca().set_xlim(sorted_durations.min(), sorted_durations.max())

plt.show()
```

- Analysis of the Relationship between Average IMDb Rating and Film Duration (Minutes):

```
data['IMDB_Rating'] = pd.to_numeric(data['IMDB_Rating'], errors='coerce')

# Calculate the average IMDb rating for each genre
genre_ratings = data.groupby('Genre')['IMDB_Rating'].mean().sort_values()

plt.figure(figsize=(12, len(genre_ratings) * 0.5))  # Resize images based on the number of genres
plt.barh(genre_ratings.index, genre_ratings.values)  # Display all genres
plt.xlabel('Average IMDb Rating')  # Change the x-axis label
plt.ylabel('Genre')  # Change the y-axis label
plt.title('Average IMDb Ratings by Genre')

# Add value labels to each bar
for i, rating in enumerate(genre_ratings.values):
    plt.text(rating, i, str(round(rating, 2)), ha='left', va='center')

plt.tight_layout()  # Adjust the layout to remove extra spacing
plt.show()
```

- Analysis of Movie Revenue Distribution by Genre:

```
data['IMDB_Rating'] = pd.to_numeric(data['IMDB_Rating'], errors='coerce')

# Calculate the average IMDb rating for each genre
genre_ratings = data.groupby('Genre')['IMDB_Rating'].mean().sort_values()

plt.figure(figsize=(12, len(genre_ratings) * 0.5))  # Resize images based on the number of genres
plt.barh(genre_ratings.index, genre_ratings.values)  # Display all genres
plt.xlabel('Average IMDb Rating')  # Change the x-axis label
plt.ylabel('Genre')  # Change the y-axis label
plt.title('Average IMDb Ratings by Genre')

# Add value labels to each bar
for i, rating in enumerate(genre_ratings.values):
    plt.text(rating, i, str(round(rating, 2)), ha='left', va='center')

plt.tight_layout()  # Adjust the layout to remove extra spacing
plt.show()
```

- Analysis of the relationship between average IMDb rating and film duration (minutes):

```
# Sort the data by runtime
sorted_data = data.sort_values('Runtime')

# Create a larger scatter plot
fig, ax = plt.subplots(figsize=(40, 6))

# Create the scatter plot
scatter = ax.scatter(sorted_data['Runtime'], sorted_data['IMDB_Rating'])

# Set the labels and title
ax.set_xlabel('Runtime')
ax.set_ylabel('IMDB Rating')
ax.set_title('IMDB Rating vs. Runtime')

# Set the x-axis ticks to be sorted and rotated
ax.set_xticks(sorted_data['Runtime'])
ax.set_xticklabels(sorted_data['Runtime'], rotation=90)

# Remove gap between scatter points and the edge of the plot
ax.margins(x=0)

# Show the plot
plt.show()
```

- Analysis of movie revenue distribution by genre:

```
# Group data by genre and calculate the median gross income for each genre
genre_gross_income = data.groupby('Genre')['Gross'].median().sort_values()

# Plot the distribution of gross income by genre
plt.figure(figsize=(45, 10))
sns.boxplot(x=data['Genre'], y=data['Gross'], order=genre_gross_income.index)
plt.xlabel('Genre')
plt.ylabel('Gross Income (in millions)')
plt.title('Distribution of Gross Income by Genre')

# Rotate x-axis labels for better readability
plt.xticks(rotation=90)

plt.show()
```

- Correlation analysis between features

```
# Select the relevant columns for correlation analysis
selected_columns = ['IMDB_Rating', 'Gross', 'Runtime', 'No_of_Votes']

# Create a correlation matrix
correlation_matrix = data[selected_columns].corr()

# Plot the correlation matrix using a heatmap
plt.figure(figsize=(10, 8))
sns.heatmap(correlation_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Matrix')
plt.show()
```