Virginia Polytechnic and State University

Author: Abdullah Al Muhit

Instructor: Reza Jafari

Date: 12/13/2024

Course Name: Information Visualization

Course Code:CS5764

Table of Contents

Table of Figures and Table:

Figures:

Tables

Deployed Link for Dashboard:

https://dashapp-1000016244847.northamerica-northeast1.run.app/

Abstract

This project will explain approaches for data visualization using a real-world dataset for Information Visualization Final Term Project. The dataset contains features for different movies for different genres with both categorical and numerical features. This dataset can be used for analytical insights as well as running different machine learning models.

The first phase of the project was developed using different kinds of static graphs which were helpful to gain patterns and relationship from the dataset. Several kinds of visualization methods were used to analyze essential features like votes, year, rating, runtime, director and stars.

In phase II, using Dash an interactive web-based dashboard was developed. This dashboard enables users to explore dataset using different dynamic tools like cleaning tools, outliers' detection, PCA, normality test etc.

In the final phase, Dockerized the application developed in phase II and launched it on GCP; thus, making it live on the web by a link shared. Deploying the application made it scalable and usable in the production environment.

This project put emphasis on the integration of static as well as interactive visualization techniques which provides high level of insights using Python-based data analysis and visualization. Moreover, this project implements the teachings from CS5764 and shows how it can be implemented in real-life environments.

Introduction

For modern data analysis, data visualization is an important part because transformations of raw data into visualization can inform new patterns and hidden insights from a dataset which can help in decision-making and enhance understanding. This project explores different visualization techniques for a real-world diverse dataset. This project has 3 progressive phases, each focusing on unique aspects of data visualization, interactive web-based dashboard and deployment.

Phase I emphasize static visualization with different kinds of plotting, providing foundational insights into the dataset. Techniques such as bar plots, heatmaps, pair plots etc. are used to analyze both numerical and categorical features.

In phase II, using Dash framework a web-based dashboard was developed. This part of the project enables users to dynamically explore the dataset through multiple functionalities which includes outlier detection and removal, PCA, normality test, statistical analysis etc. Users can upload dataset and analyze numerical and categorical features of the data which makes the insight accessible to a broader audience.

In the third and final phase, the dashboard was containerized using Docker and deployed on Google Cloud Platform (GCP). This enables real-time access and interaction with the dashboard.

This project demonstrates Python-based data visualization techniques with modern web applications which can be accessed through web servers.

Description of the Dataset

The dataset that was used for this project is scrapped from Kaggle (https://www.kaggle.com/datasets/rajugc/imdb-movies-dataset-based-on-genre/data).

There were 16 different datasets based on genre which were merged into one. The merged dataset has 368,300 observations and 14 features. The numerical features are 'votes', 'rating', 'gross (in $)' and categorical features include 'movie_name', 'director', 'star', 'description', 'runtime', 'year' etc. 'runtime' and 'year' were later converted to numerical columns (Figure 1.1).

The dataset contains a lot of null values (Figure 1.1) which were handled carefully. There were some unwanted values and duplicate values as well.

This dataset can be used for multiple purposes. First, a movie recommendation system can be developed based on rating, votes etc. Moreover, for the entertainment industry, people can also analyze the audience sentiment based on genre, director and stars.

```
In [152]: df.shape
Out[152]: (368300, 14)
In [153]: df.dtypes
Out[153]:
movie_id         object
movie_name       object
year             object
certificate      object
runtime          object
genre            object
rating          float64
description      object
director         object
director_id      object
star             object
star_id          object
votes           float64
gross(in $)     float64
dtype: object
```

Figure 1.1: Shape and Null values

Preprocessing the Data

For this project, the dataset underwent various steps of preprocessing to ensure the quality of the data analysis. The steps undertaken are detailed below:

1. Handling Missing Values

- Null values were identified across all features (Figure 1.1).
- Features like 'certificate' and 'gross (in $)' were excluded from the null value removal process to preserve potentially useful information as these two features have more than 50% null values.
- For all other features, missing values were dropped.

2. Removing Duplicate Values

- There were only 3 duplicate values which were dropped.

3. Addressing Unwanted Values

- Inconsistent or unwanted values, such as non-standard entries in categorical features or outliers in numerical features, were identified and rectified.
- 'year' had values in roman which were handled and removed from the dataset.
- 'director' and 'star' had values with '\' which were removed from the text.

4. Feature Engineering

- **Year**:
    - Originally a categorical feature, **year** was converted to a numerical feature representing the release year of the movie.
- **Runtime**:

- Originally represented as a categorical feature (e.g., "120 min"), **runtime** was extracted and converted to a numerical feature indicating the movie's duration in minutes.

5. Dropping Statistically Insignificant Columns:

- Dropped unimportant columns: ‘movie_id’, ‘description’, ‘stard_id’, ‘direcotr_id’

The Final Cleaned Data

The final cleaned data have no missing information other than ‘certificate’ and ‘gross (in $)’. Any unwanted values and duplicate values were handled effectively. The final dataset has 213,636 observations and 10 features (Figure 2.1). Finally, there is 5 numerical and 5 categorical features (Figure 2.1).

```
In [169]: df.dtypes
Out[169]:
movie_name      object
year             int64
certificate     object
runtime        float64
genre           object
rating         float64
director        object
star            object
votes          float64
gross(in $)    float64
dtype: object
```

Figure 2.1: Final Cleaned Dataset

Outlier Detection and Removal

Outliers were detected and removed from the numerical features.

1. Outlier Detection

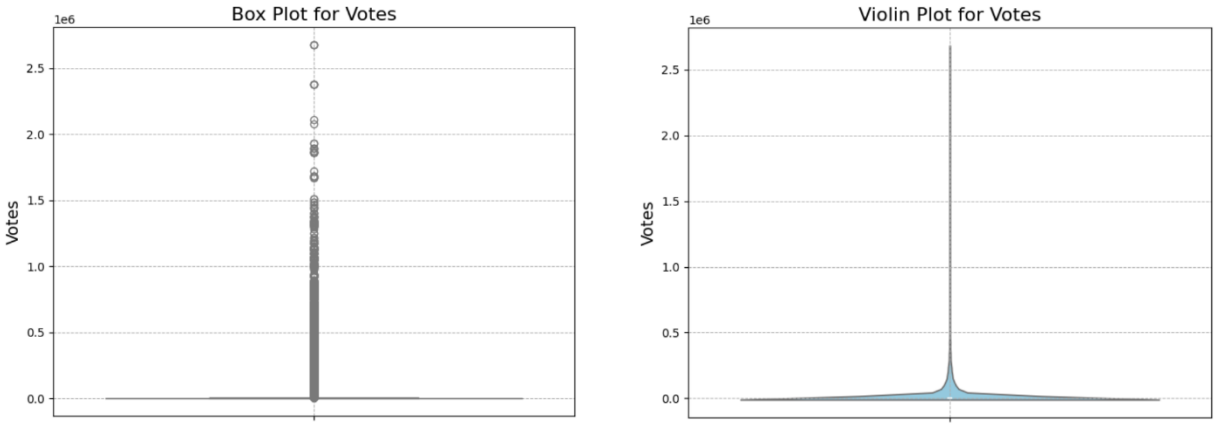


Figure 3.1: Box and Violin Plot for Votes

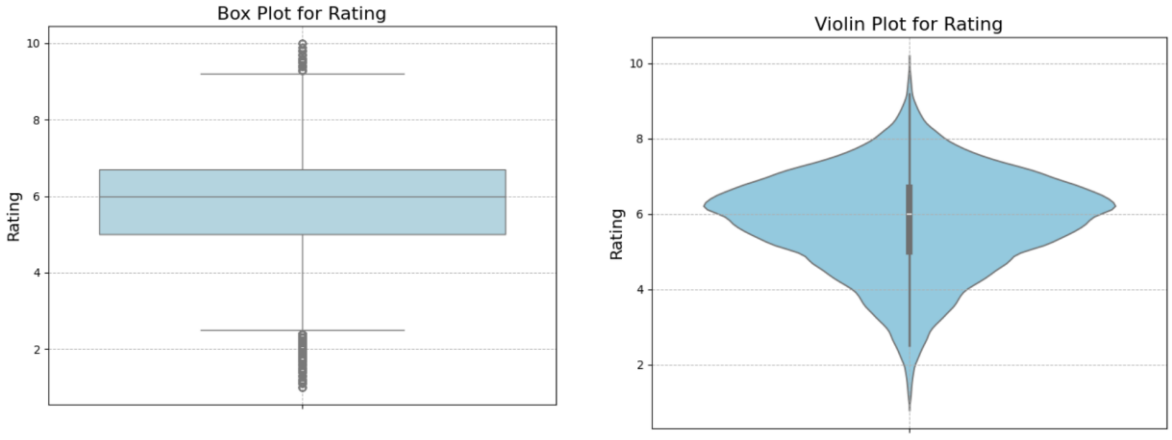


Figure 3.2: Box and Violin Plot for Rating

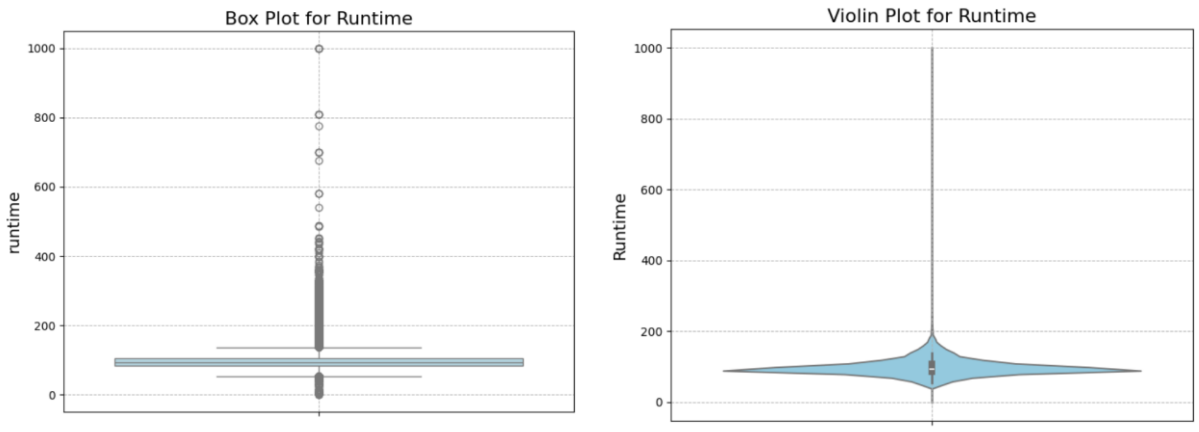


Figure 3.3: Box and Violin plot for Runtime

Outliers in votes and runtime were predominantly at the higher end, reflecting extreme values for blockbuster movies or unusually long runtimes (Figure 3.1 and 3.3).

The rating feature had fewer outliers due to its constrained range (typically 1 to 10) (Figure 3.2).

The removal of outliers ensured a more normalized distribution for the features, making the dataset more suitable for statistical analysis and machine learning techniques.

2. Outlier Removal Using the IQR Method

The **Interquartile Range (IQR) Method** was applied to remove detected outliers using the following theory:

IQR =Q3 -Q1

Outliers = (Q1-1.5*IQR) or (Q3+1.5*IQR)

After removing the outliers again, the box and violin plot were used for visualizing outliers (Figure 3.4).

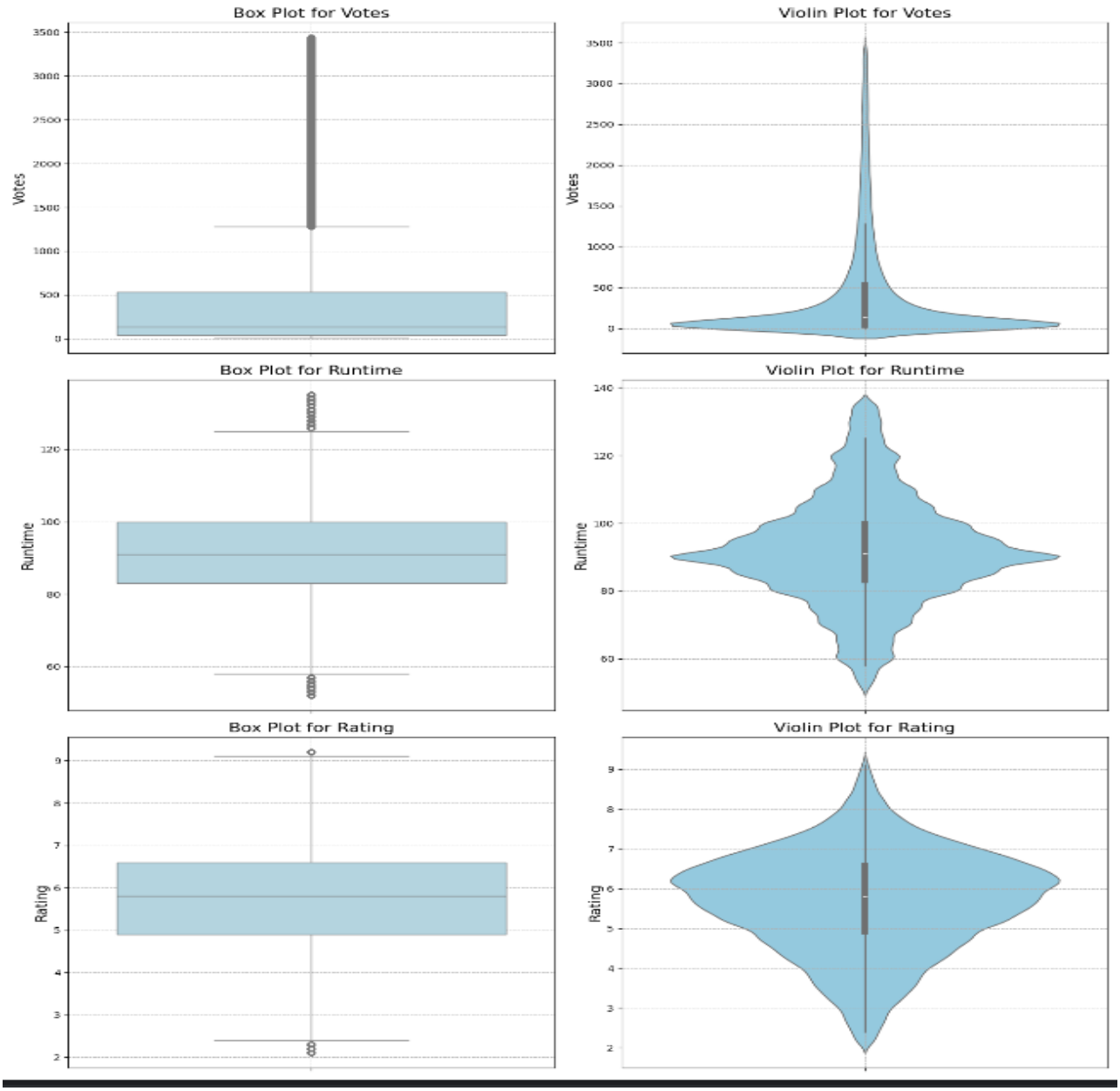


Figure 3.4: After removing outliers using IQR

As visualized, even though there are still some outliers, most of the outliers were removed and a reduced number of outliers are visualized.

Principal Component Analysis (PCA)

For reducing the dimensionality of the dataset PCA was implemented. The main objective for applying PCA was to find the principal components which explain more than 90%

variance. PCA was implemented on numerical features: ‘year’, ‘rating’, ‘runtime’ and ‘votes. ‘gross (in $)’ was excluded because it has more than 50% null values.

Here are the steps of PCA analysis:

- **Standardization**: numerical features were standardized using Scikit learn package.
- **PCA**: PCA analysis was implemented on the standardized data
- **Determining the number of components**: Number of components was determined to be n_components = 4 (Figure 4.1)

To visualize the effect of number of components, the graph was developed using number of components (Figure 4.1).

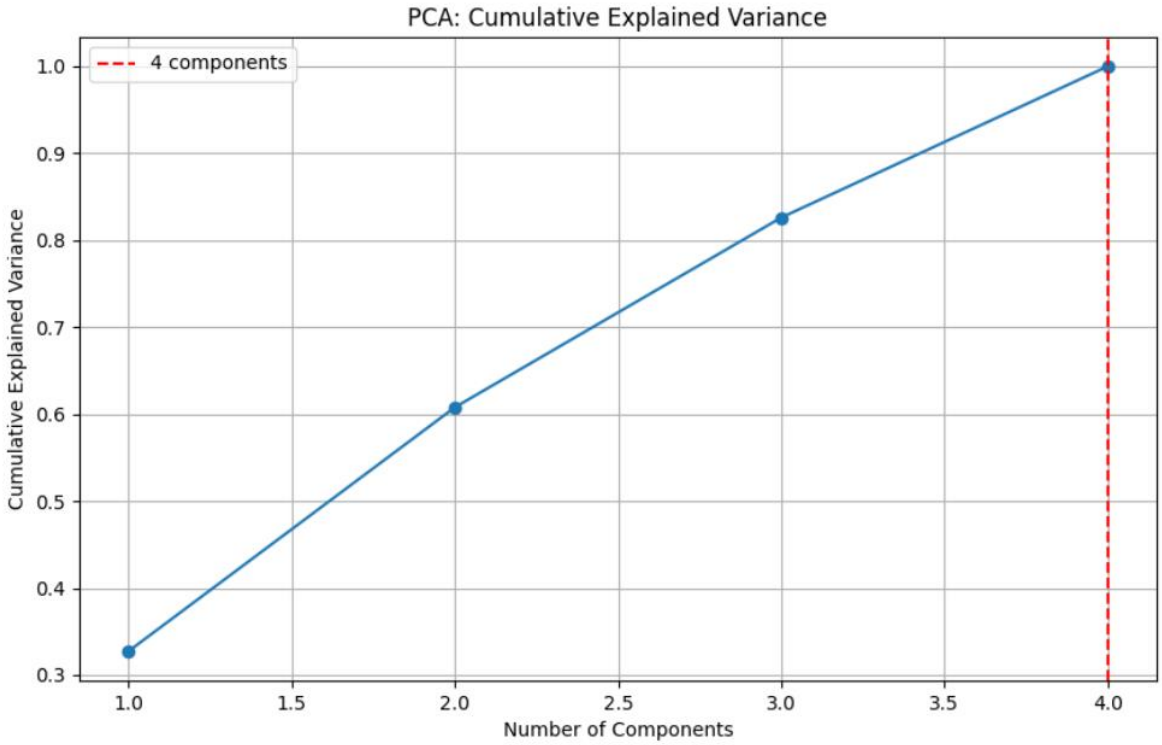


Figure 4.1: Cumulative Explained Variance

The graph shows that we need all four numerical features to explain 90% of the variance.

Normality Test

1. Methods Used

1. **Kolmogorov-Smirnov (KS) Test** :

- Tests whether a sample comes from a specified distribution (e.g., normal distribution).
- Null Hypothesis (H0H_0H0): The data follows a normal distribution (Figure 5.1, 5.2,5.3 ,5.4).

2. **Shapiro-Wilk Test**:
   - Specifically designed to assess normality.
   - Null Hypothesis (H0H_0H0): The data follows a normal distribution (Figure 5.1, 5.2, 5.3, 5.4).
3. **D'Agostino and Pearson's DaD_aDa K-Squared Test**:
   - Combines skewness and kurtosis to evaluate the normality of a dataset.
   - Null Hypothesis (H0H_0H0): The data follows a normal distribution (Figure 5.1, 5.2, 5.3, 5.4).
4. **Q-Q Plots**:
   - A visual method to assess normality by comparing the quantiles of the dataset against a theoretical normal distribution (Figure 5.5).

Results of Normality Test:

None of the numerical features are normal according to the 4 tests. For further clarification, QQ plot was plotted to visualize the distribution of the features (Figure 5.4).

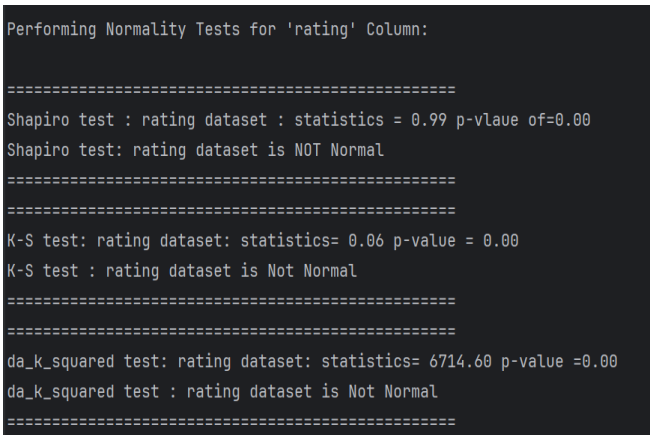


Figure 5.1: Normality test for rating

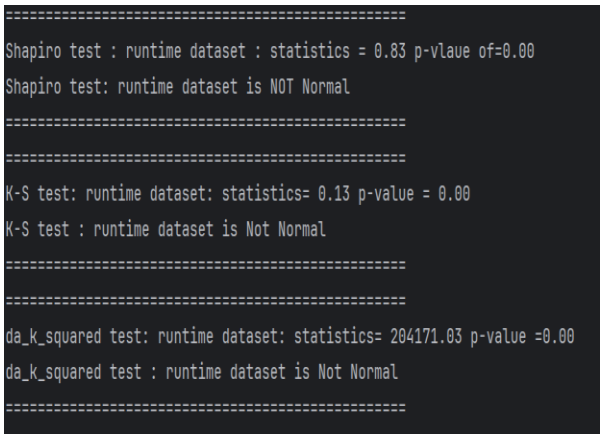


Figure 5.2: Normality test for runtime

```
Performing Normality Tests for 'gross(in $)' Column:

==================================================
Shapiro test : gross(in $) dataset : statistics = 0.50 p-vlaue of=0.00
Shapiro test: gross(in $) dataset is NOT Normal
==================================================
==================================================
K-S test: gross(in $) dataset: statistics= 0.32 p-value = 0.00
K-S test : gross(in $) dataset is Not Normal
==================================================
==================================================
da_k_squared test: gross(in $) dataset: statistics= 24298.48 p-value =0.00
da_k_squared test : gross(in $) dataset is Not Normal
==================================================
```

Figure 5.3: Normality test for gross (in $)

```
==================================================
Shapiro test : votes dataset : statistics = 0.15 p-vlaue of=0.00
Shapiro test: votes dataset is NOT Normal
==================================================
==================================================
K-S test: votes dataset: statistics= 0.43 p-value = 0.00
K-S test : votes dataset is Not Normal
==================================================
==================================================
da_k_squared test: votes dataset: statistics= 398298.86 p-value =0.00
da_k_squared test : votes dataset is Not Normal
==================================================
```

Figure 5.4: Normality test for votes

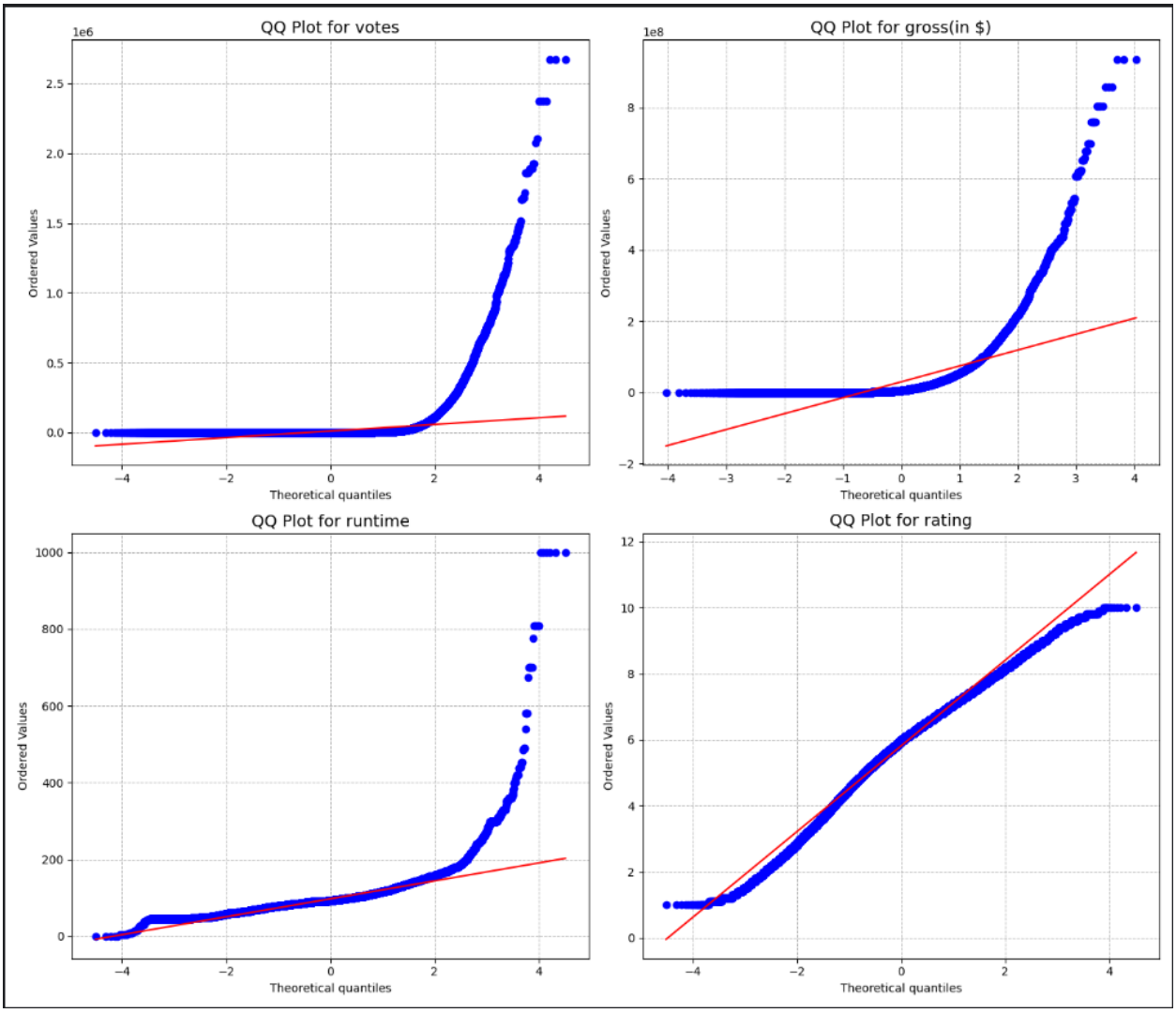


Figure 5.5: QQ plot for numerical features

As mentioned, the QQ plots (Figure 5.5) show that, not any of the numerical features are normal. ‘rating’ has almost a normal distribution, however there are some fluctuations at the head and tail of the distribution.

Data Transformation

To address the non-normality observed in some of the features three types of transformation were applied: log transformation, square root transformation and Yeo-Johnson Transformation.

Observations After Transformation:

- Log Transformation has proved to be the most effective transformation for making the dataset normal.
- Square Root transformation does not really affect the data.
- Yeo-Johson Transformation normalizes the rating feature effectively (Figure 6.1).

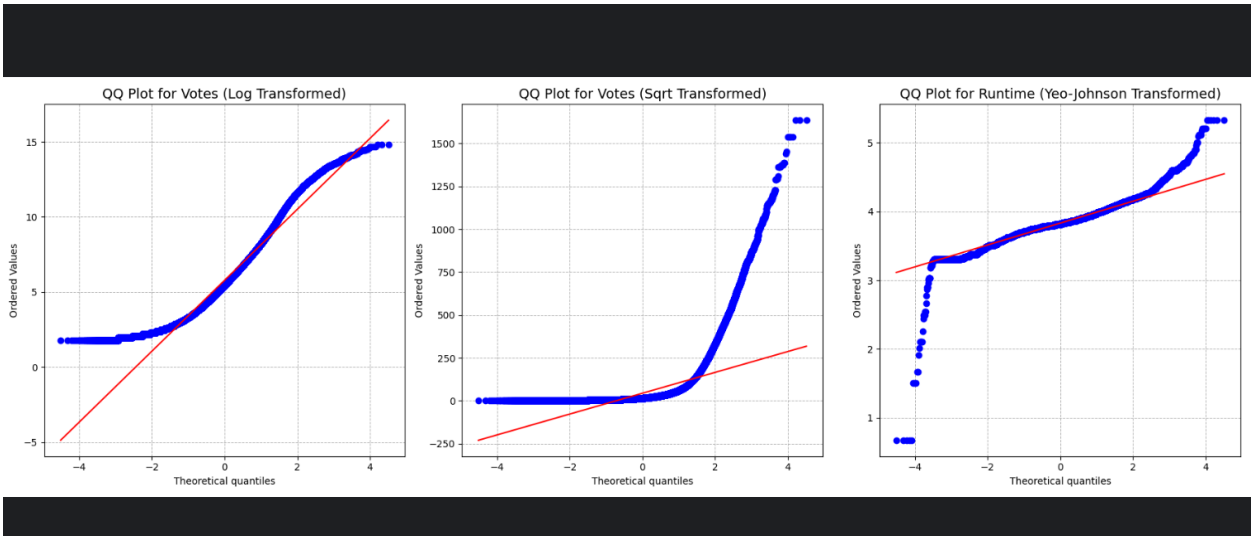


Figure 6.1: QQplot after transformation

By transforming the data, the dataset is now much better fit for Gaussian-Distributed model and ensures reliable interpretability.

Heatmap and Pearson Correlation Coefficient Matrix

The Pearson correlation coefficient matrix was used to examine the linear relationship between the features. A heatmap were visualized to see the correlation with a cbar.

*Pearson Correlation Coefficient (*r) shows a linear relationship between two features.

- r = +1 shows positive correlation
- r = 0 shows no correlation
- r = -1 shows negative correlation

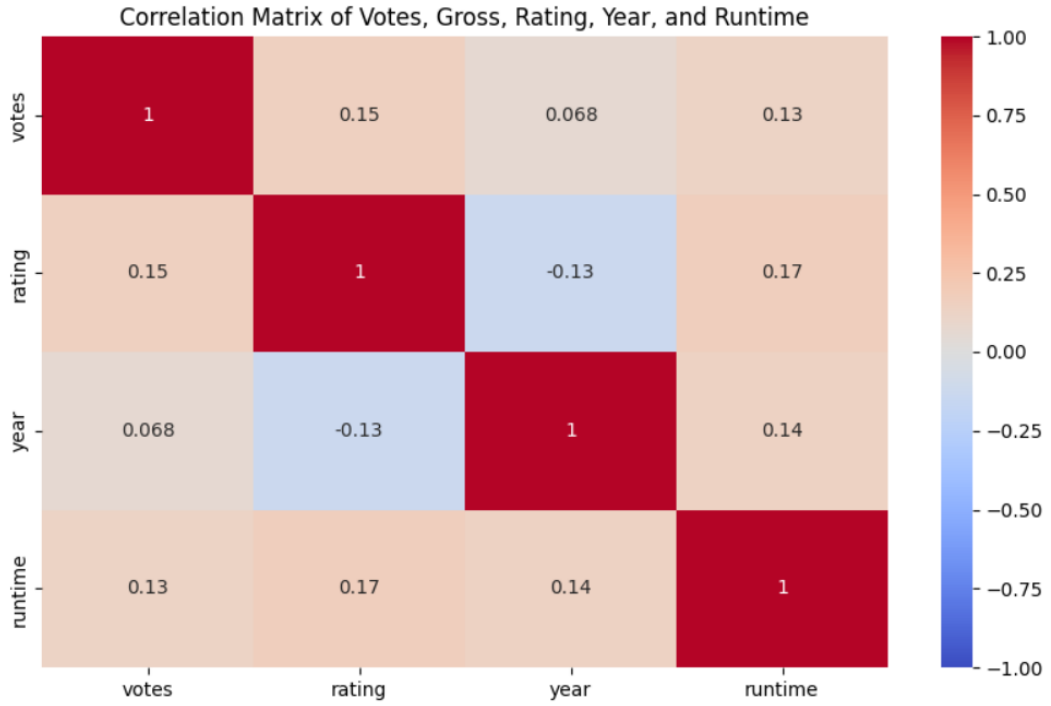


Figure 7.1: Heatmap of Correlation between numerical features

Observations (Figure 7.1):

- Positive Correlation: rating and votes have positive correlation suggesting higher votes comes with higher rating. Also, rating and runtime has also positive correlation between them.
- No Correlation: Most of the values are very close to zero which shows the correlation between them is very weak.
- Negative Correlation: year and rating have negative correlation suggesting modern movies received less ratings.

Statistics:

	year	runtime	...	votes_sqrt	runtime_yeojohnson
count	213636.000000	213636.000000	...	213636.000000	213636.000000
mean	1993.504943	97.647162	...	44.006175	3.831884
std	26.154679	25.695658	...	91.969326	0.162350
min	1894.000000	1.000000	...	2.236068	0.674565
25%	1976.000000	85.000000	...	6.928203	3.756840
50%	2003.000000	93.000000	...	15.198684	3.819248
75%	2015.000000	106.000000	...	37.469988	3.909354
max	2023.000000	999.000000	...	1635.705047	5.327599

Figure 8.1: statistics of numerical features

Descriptive statistics were calculated for the primary numerical features (Figure 8.1), including:

- **Votes (Log-Transformed)**: votes_log
- **Runtime (Square Root-Transformed)**: runtime_sqrt
- **Rating (Yeo-Johnson-Transformed)**: rating_yeojohnson
- **Year**: year

Metrics calculated:

- **Mean**: Average value of the feature.
- **Median**: Middle value, indicating central tendency.
- **Standard Deviation (Std Dev)**: Spread of the data.
- **Minimum and Maximum**: Range of the data
- **Quantiles**: first quantile, second quantile and third quantile (25%,50% and 75%)

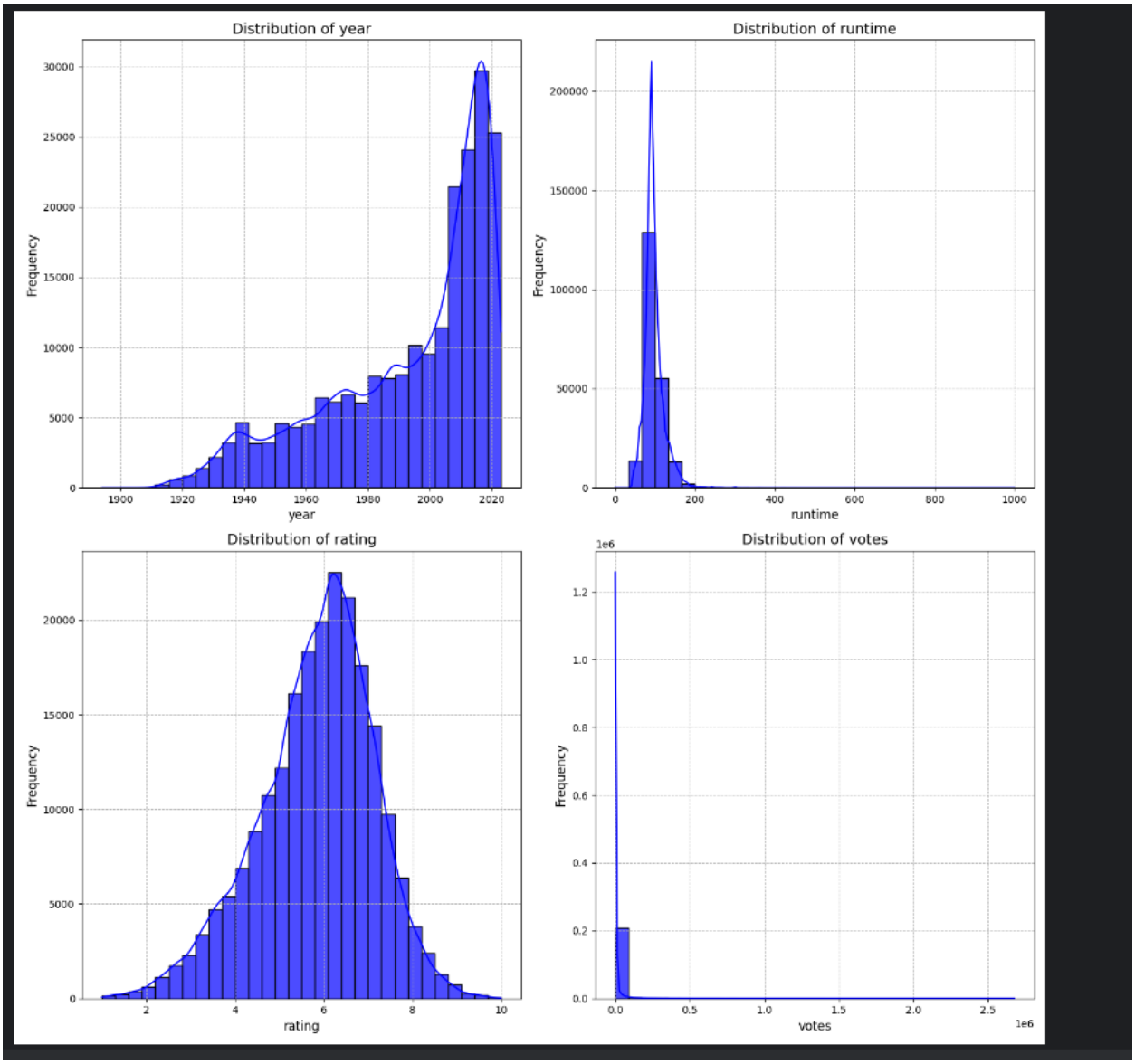


Figure 8.2: KDE for numerical features

Observations (Figure 8.2):

- Most frequency of movies are in the year of 2015 which is shown by the KDE plot
- Most of the ratings are from 5 to 7. The peak of the graph shows the most frequent rating, which is 6.5.
- Most of the runtime is typically from 90-200 minutes with some runtime to 1000 which shows there are potential outliers in the data.
- For votes, it is highly skewed with some data with high votes.

Data Visualization

- Average Rating Trend Over Time: The line plot (Fig. 9.1) shows that at the

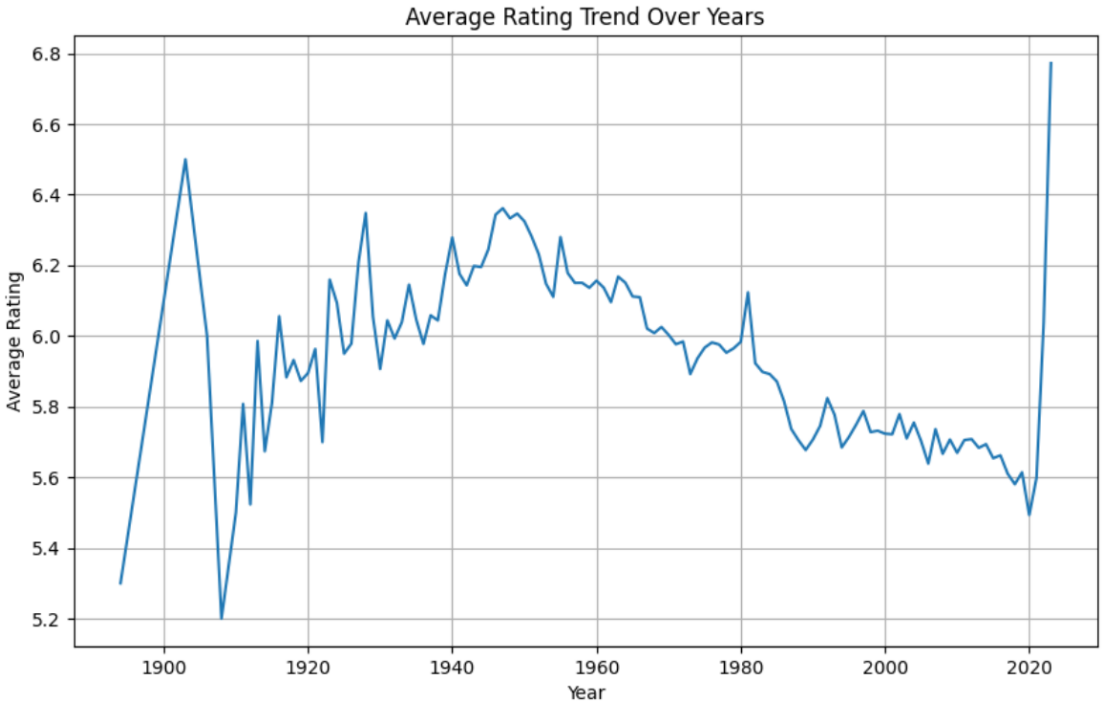


Figure 9.1: Line plot for rating over time

beginning of 1900 the ratings are not stable. From1940 the rating started to be more stable as there was modern equipment for movies. There is a sharp rise of rating from 2020 which may be because of COVID-19 when people started to watch a lot of movies.

- Votes and Gross Over Time: From the beginning to the end the vote is always steady. But there is an increasing tendency for gross over the years. Again, in the year 2020, the movies generated more revenue which is again maybe because people started to watch a lot of movies during the COVID-19 season (Figure 9.2).

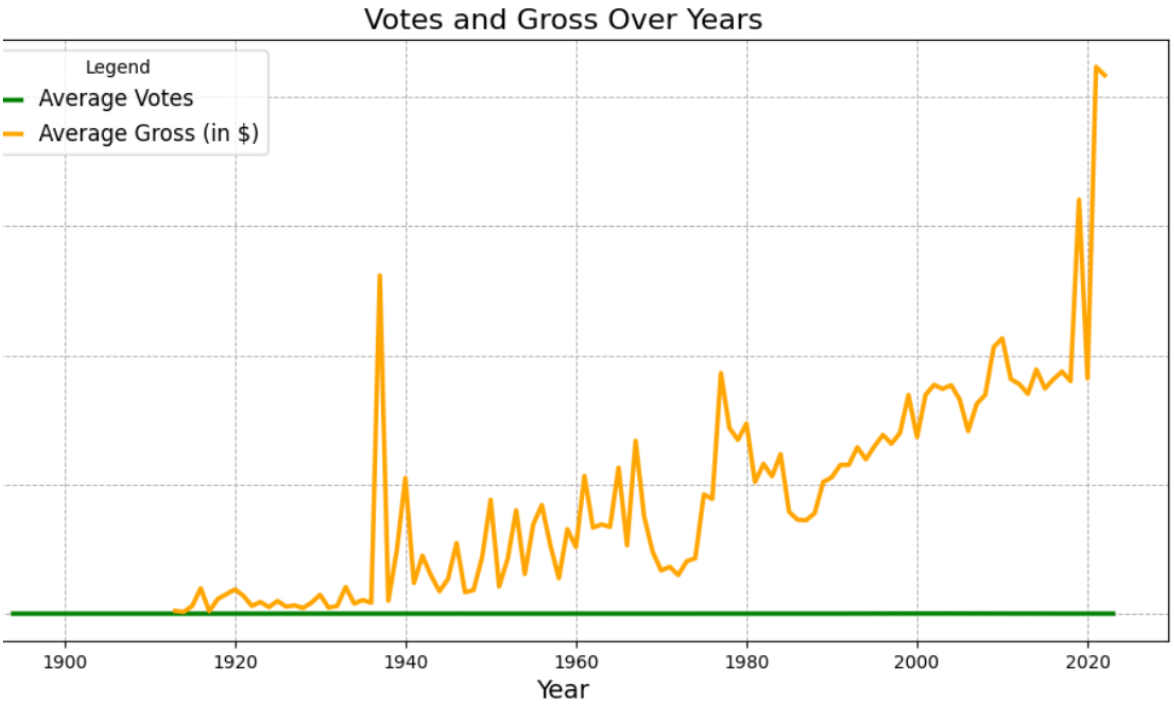


Figure 9.2: Line plot of votes and gross over year

- Average Gross Revenue by Genre: In figure 9.3, average gross revenue by genre is shown. As visualized, animation has generated the highest revenue among the genres and fil-noir genre has generated least amount of revenue maybe because this genre is new in the entertainment industry.

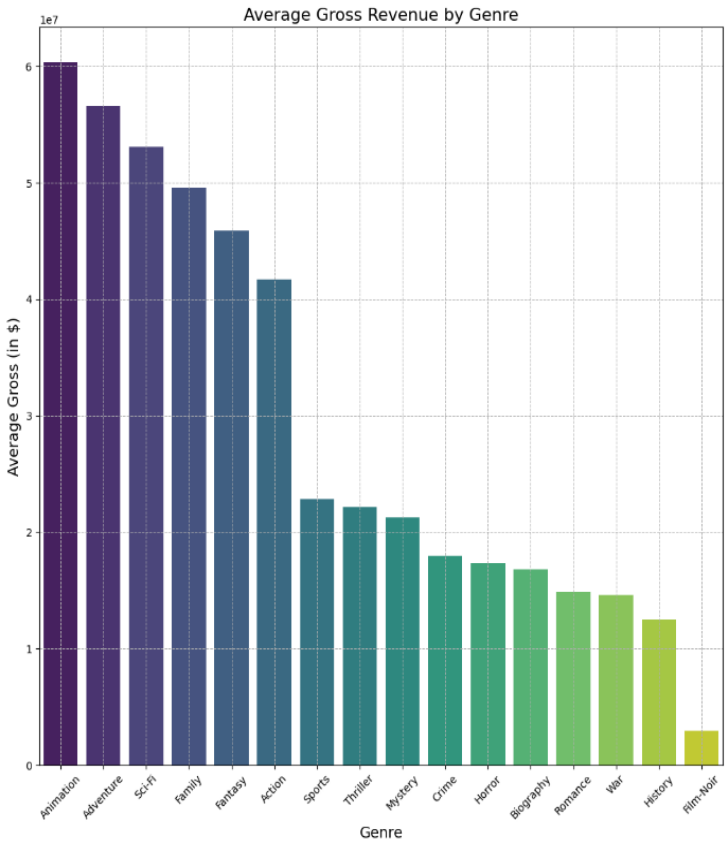


Figure 9.3: Bar plot of average gross revenue by Genre

- Average rating by Genre and Top 5 certificates: Top 5 certificates were chosen because certificate feature is around 70% null values. In the figure (Fig 9.4) we can see the top 5 certificates which are approved, not rated, PG, PG 13 and R. For different genres, different kinds of certificates are more popular. For example, for the genre fil-noir PG-13 rated movies are more popular whereas for action genre approved movies are more popular. This gives an idea on frequency of different certification on different genre.

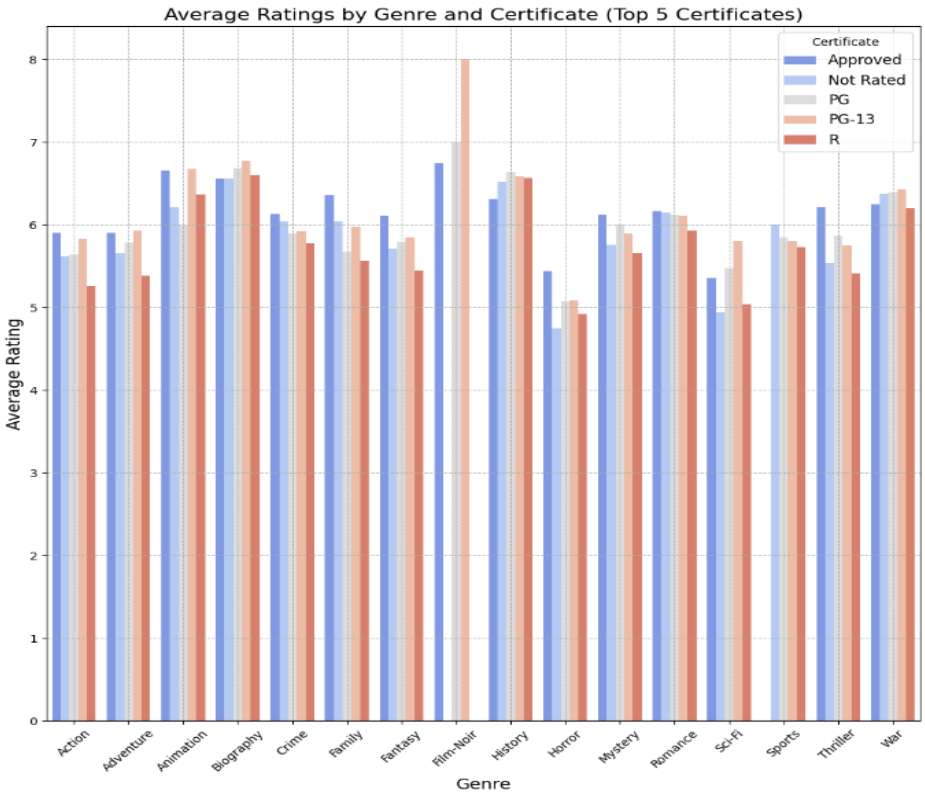


Figure 9.4: Average rating by Genre and top 5 certificates (Group Bar Plot)

- Frequency of Movies by Genre and Top 5 certificates: The following stacked bar plot (Fig 9.5) shows the frequency of movies by certification and genres. Thriller type movies have the most frequency with the same ratio of R rated and not rated types of movies. Also, there are a noticeable number of movies which are not rated. The least frequent movie is of the film-noir genre.

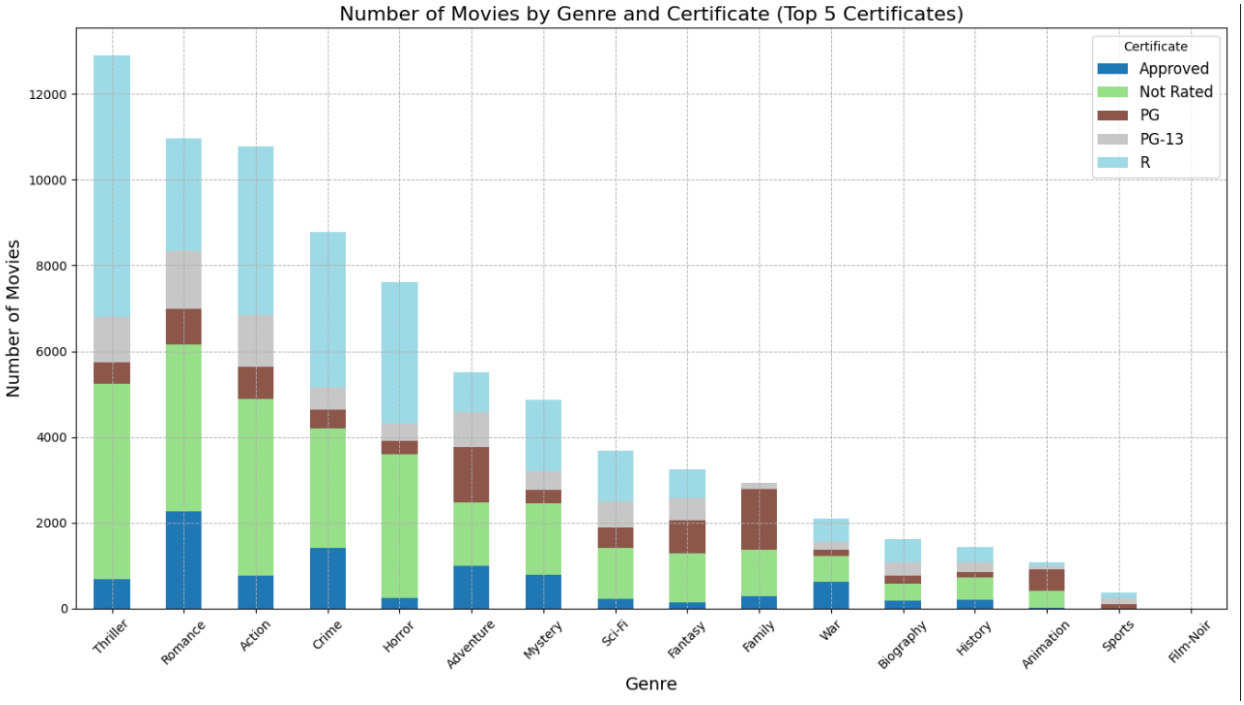


Figure 9.5: Frequency of movies by genre and certification (Stacked bar)

- Movies Count by Genre:

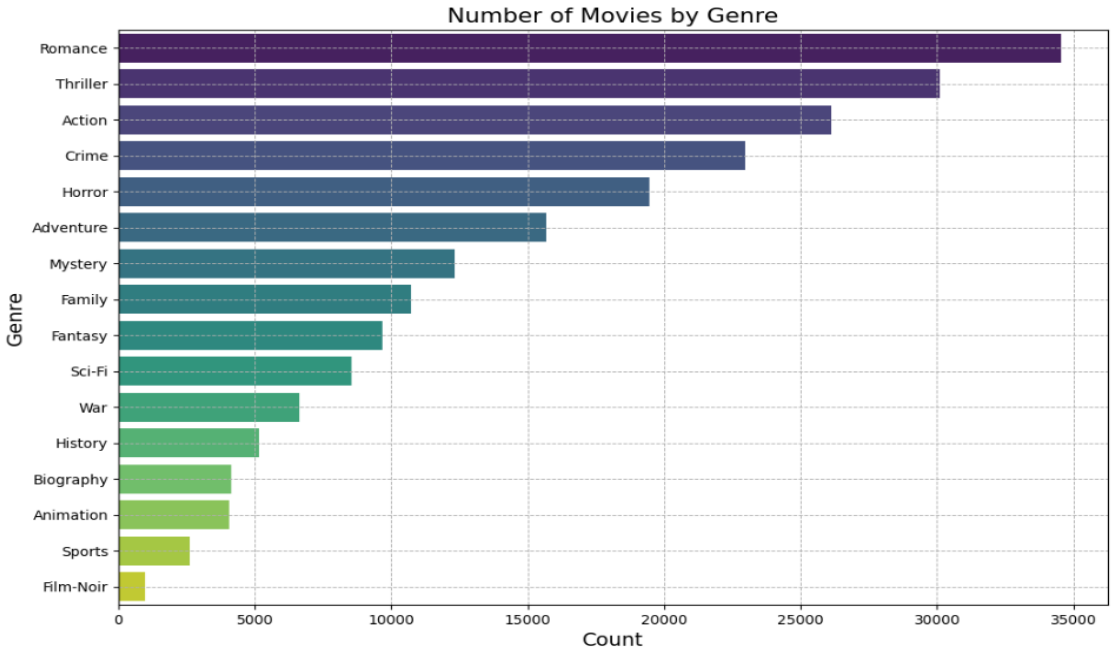


Figure 9.6: Frequency of movies by genre

Most frequency with around 34000 movies is the thriller type movies whereas film-noir have only 900 movies which shows there is imbalance of genre (Fig 9.6).

- Most common genre over time:

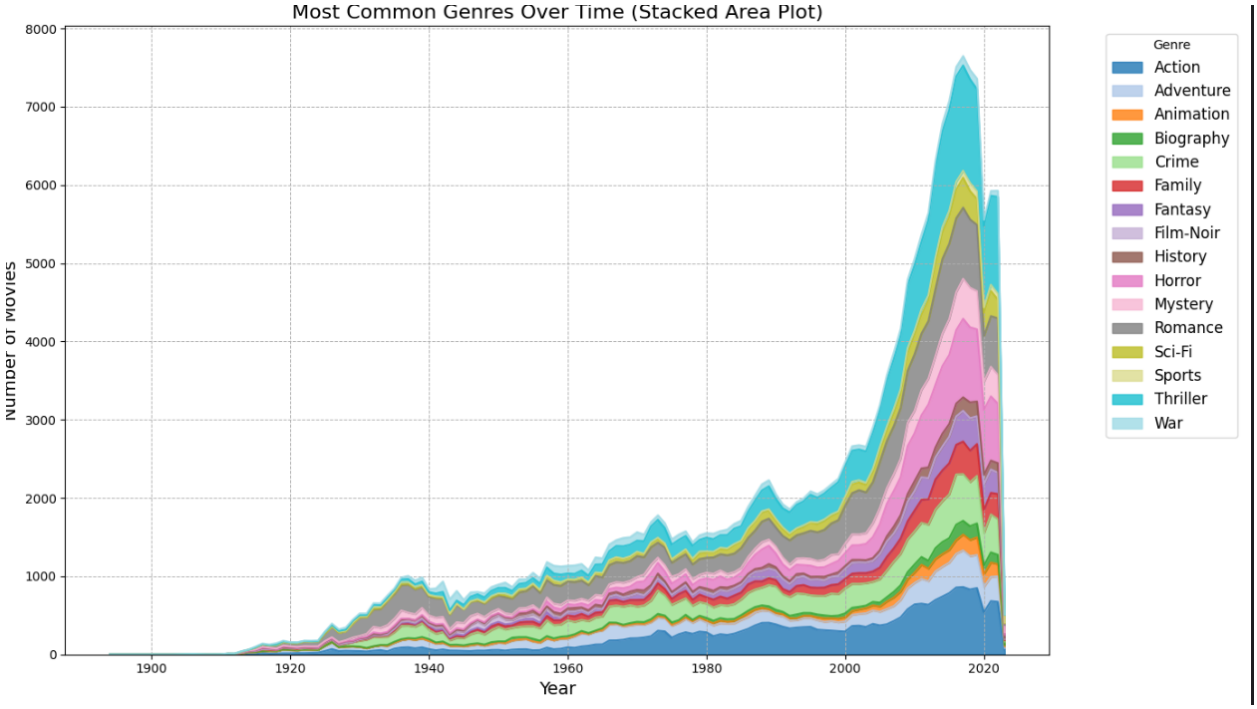


Figure 9.7: Stacked area plot for most common genre over time

Most common genre from the year of 1900 to 2020 is the thriller type movies (Fig 9.7). However, people were less attracted to action type movies which have the lowest popularity.

- Numerical Features by Genre: In figure 9.8, there is a positive correlation between votes and gross revenue which implies that higher voted movies generated higher revenue. This figure also proves that genre affects the features like votes, rating and gross. The diagonal plots shows the distribution of each features.

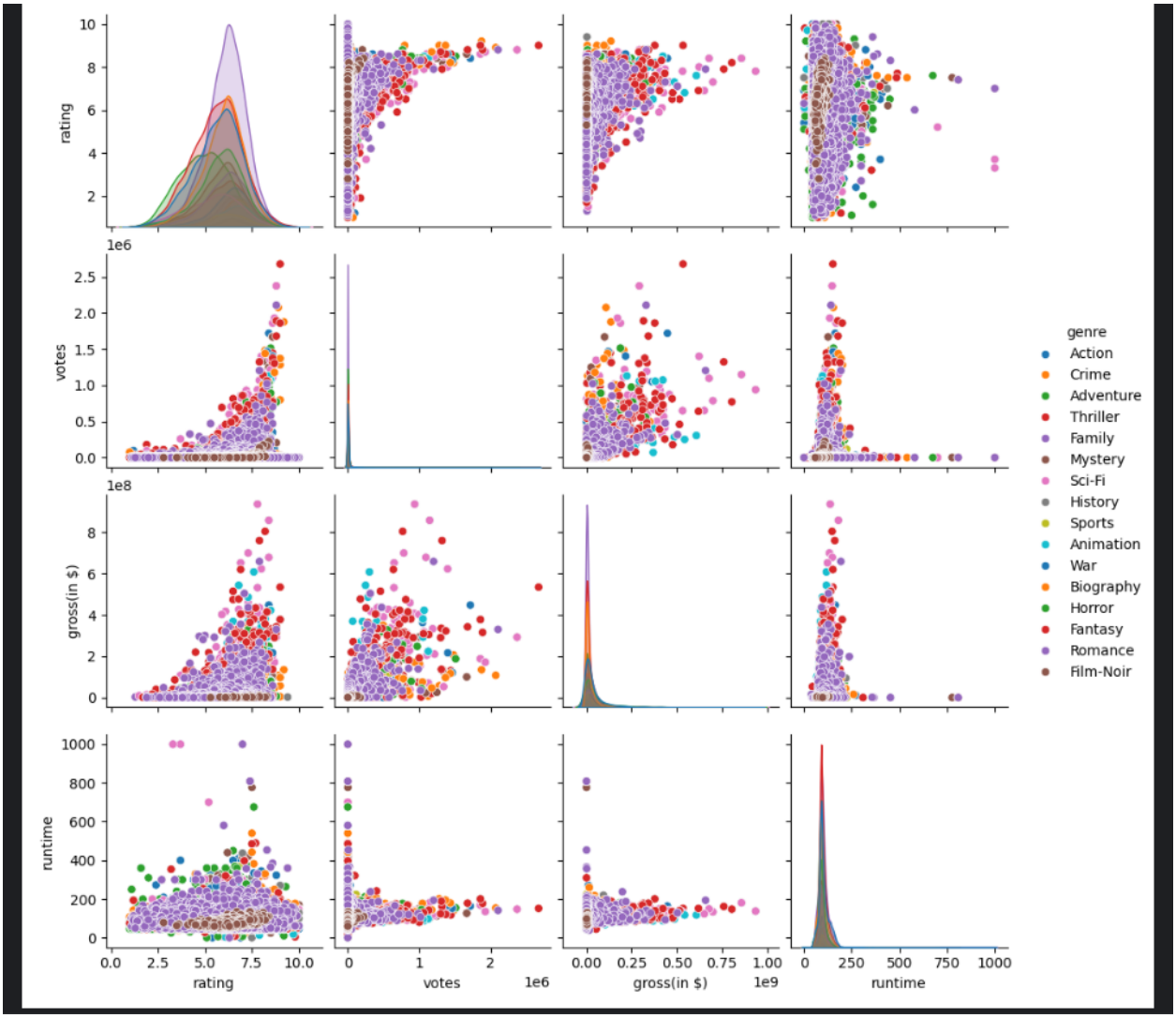


Figure 9.8: Pair plot for numerical columns by genre

- Rating Distribution over Decade: The rating distribution for each decade (Fig 9.9) shows, in the decade of 2020 the rating distribution was the highest as mentioned before which may be because of the covid. As always, in the beginning of 1900 rating density is the lowest,

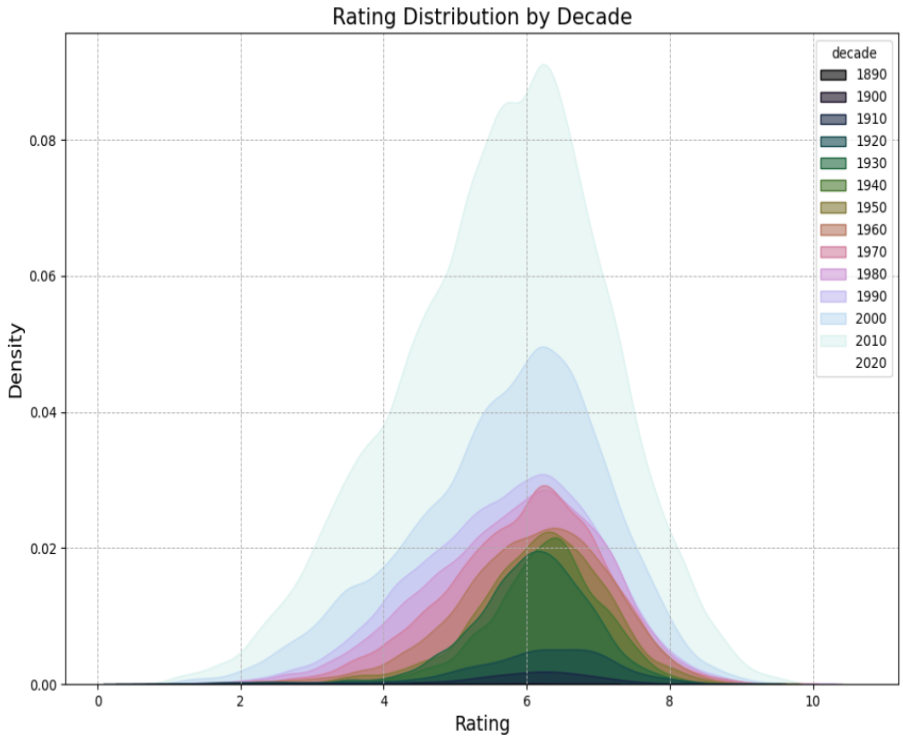


Figure 9.9: KDE plot of Rating Distribution by decade

- Votes vs Gross Revenue by Genre:

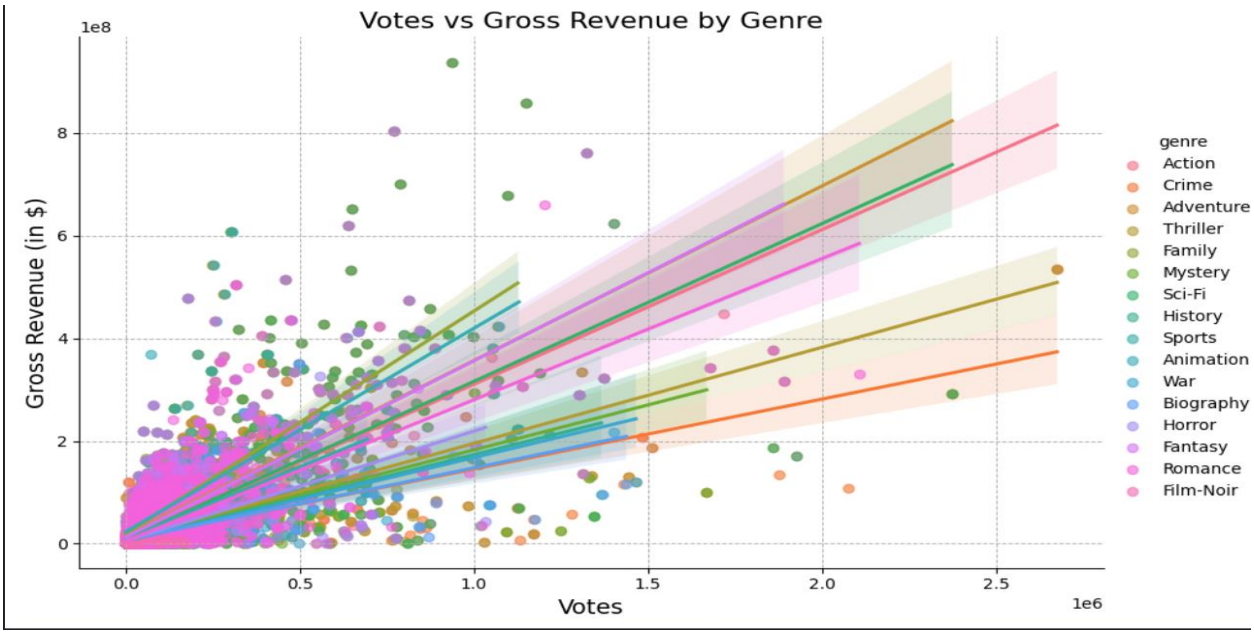


Figure 9.10: reg plot for votes vs genre

The plot (Fig 9.10) shows the positive correlation between votes and gross revenue. The steeper the slope of the line the higher the revenue. Animation and Adventure type movies have generated the highest revenue among the genres. The shade on the line shows the confidence interval.

- Rating vs Runtime:

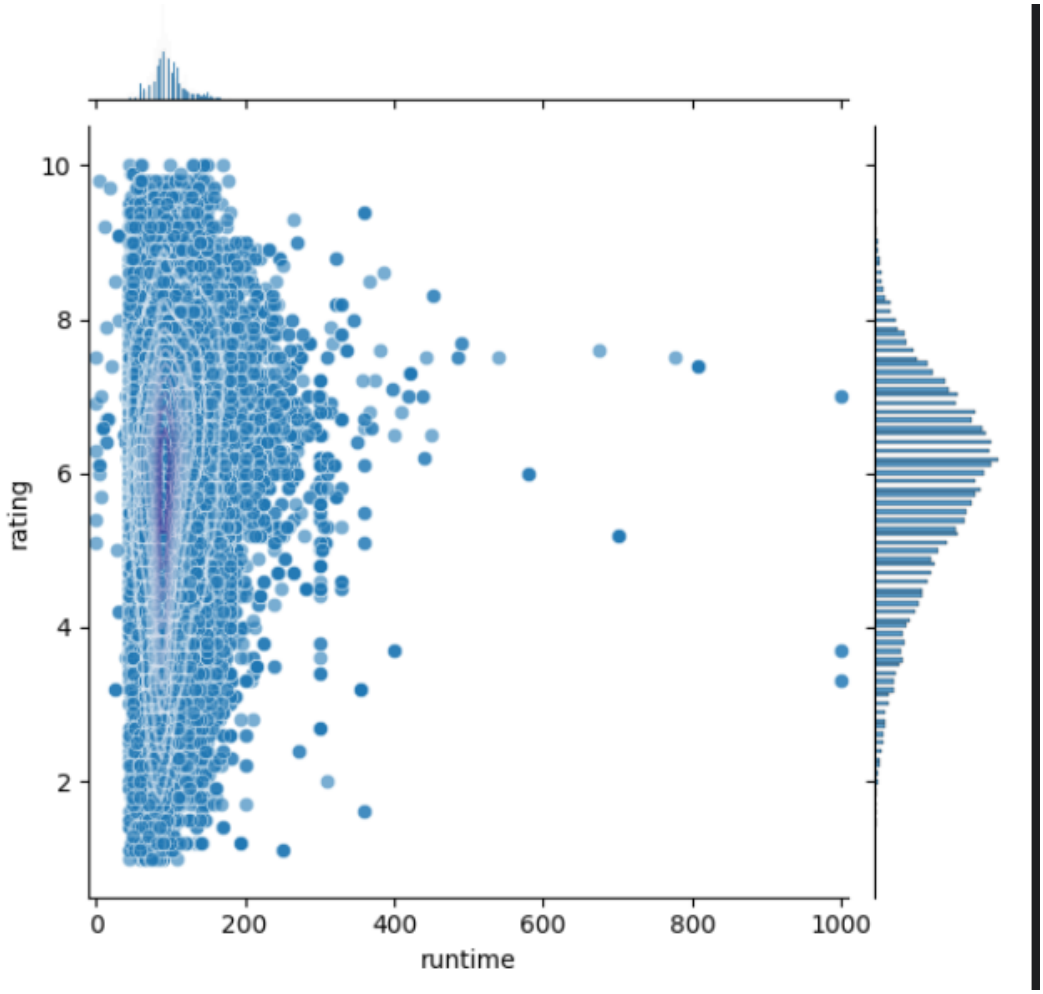


Figure 9.11: rating vs runtime joint plot

The joint plot (Fig 9.11) shows the rating distribution is mostly between 5 to 8 and runtime is between 80-200 with some runtime with the value of 1000 min which may be potential outliers. Also, we can see central clustering for the movies.

- Top 5 directors by Rating: The highest rated director is Richard Trophe with a mean rating of 6.2 and the second highest rated director is William Beaudine. However, for a single movie, the highest rating is received by Willim Beaudine with near 6.9 rating (Fig 9.12).

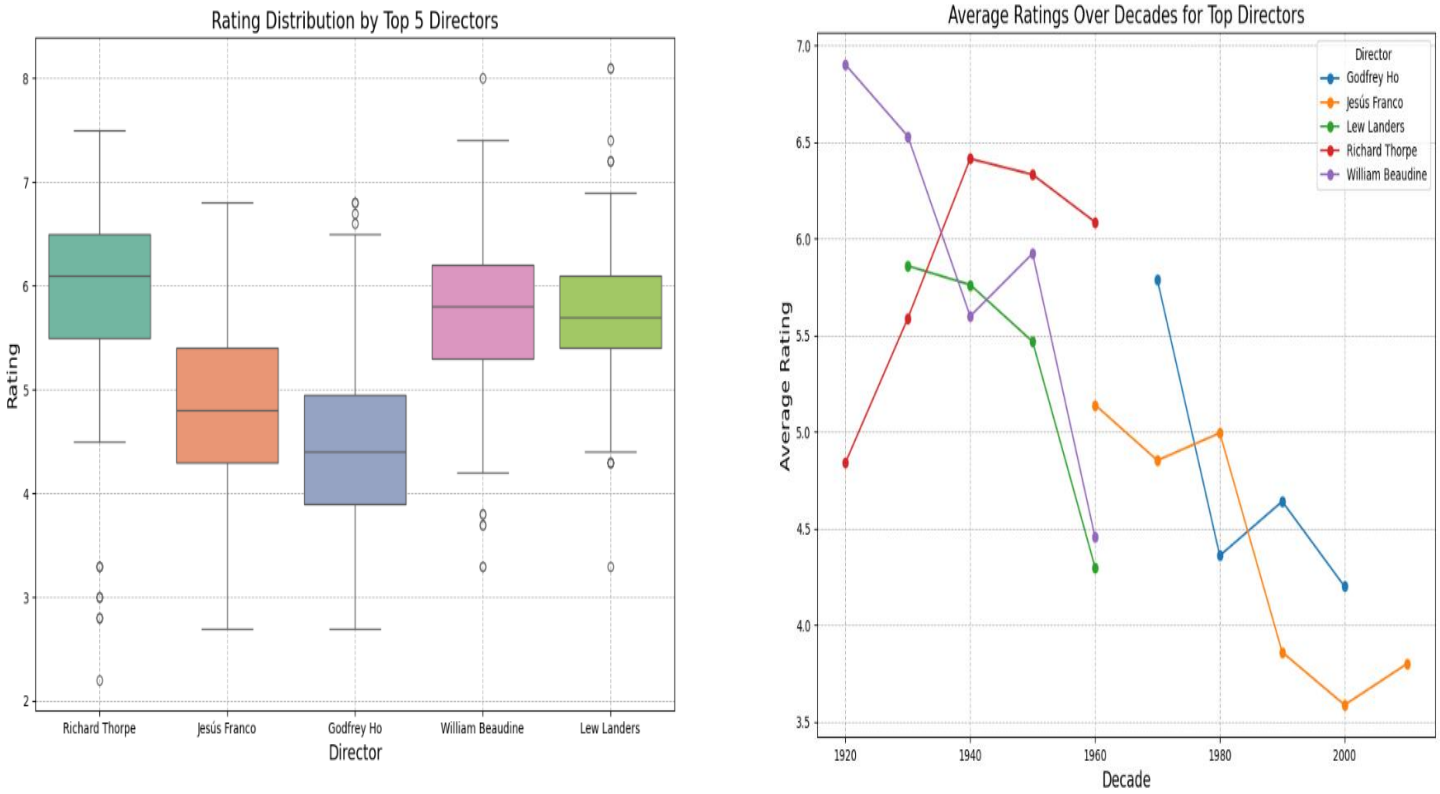


Figure 9.12: Top 5 ratings by directors

- Rug plot for Runtime:

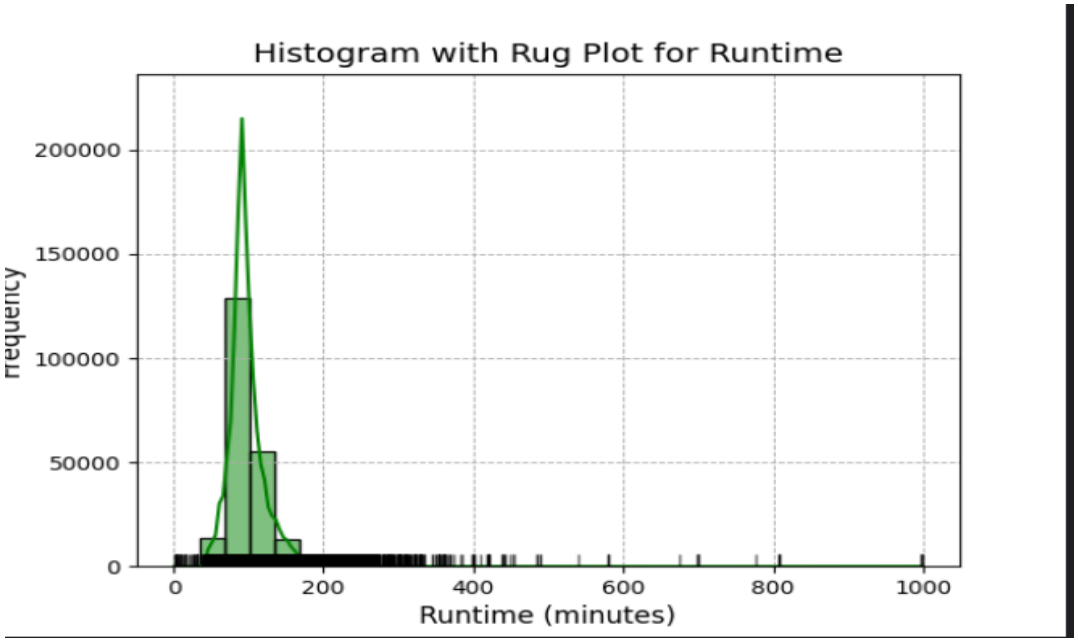


Figure 9.13: Rug plot for Runtime

The rug plot implies that most of the runtime is from 80-150 minutes. However, there are some runtimes with more than four hundred minutes which is because of outliers (Fig 9.13).

- 3D plot between Votes, Gross and Rating:

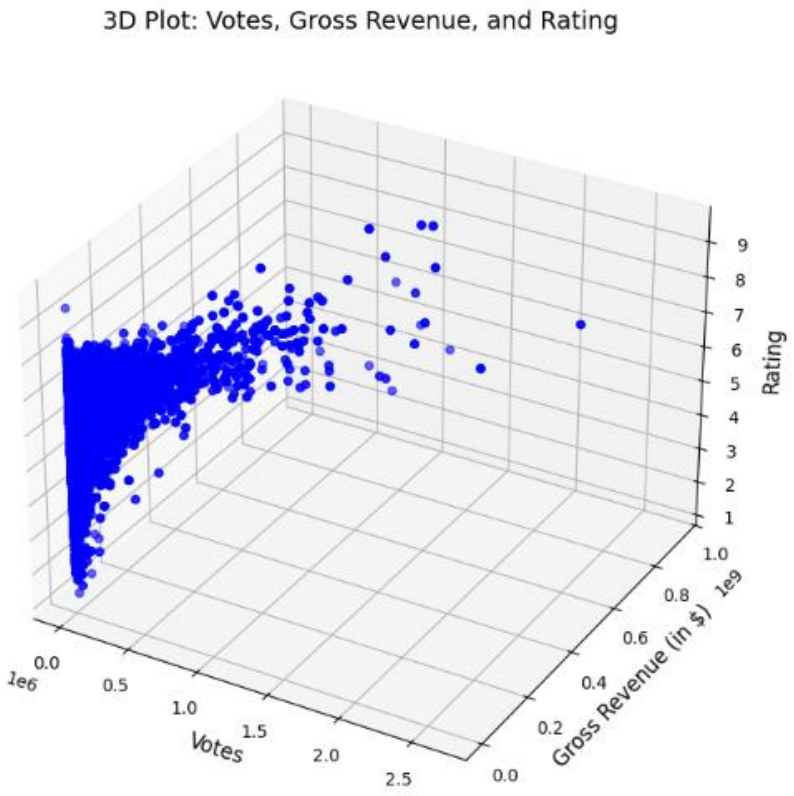


Figure 9.14: 3D plot between votes gross and rating

In Figure 9.14, the 3d plot shows cluster where movies perform extremely well. The runtime has a broad range with potential outliers.

- Contour plot votes vs gross revenue (density is rating): Figure 9.15 shows the rating density based on votes and gross revenue. Red part shows higher rating where blue part shows lower density of rating. Also, dense cluster is noticable in the plot.

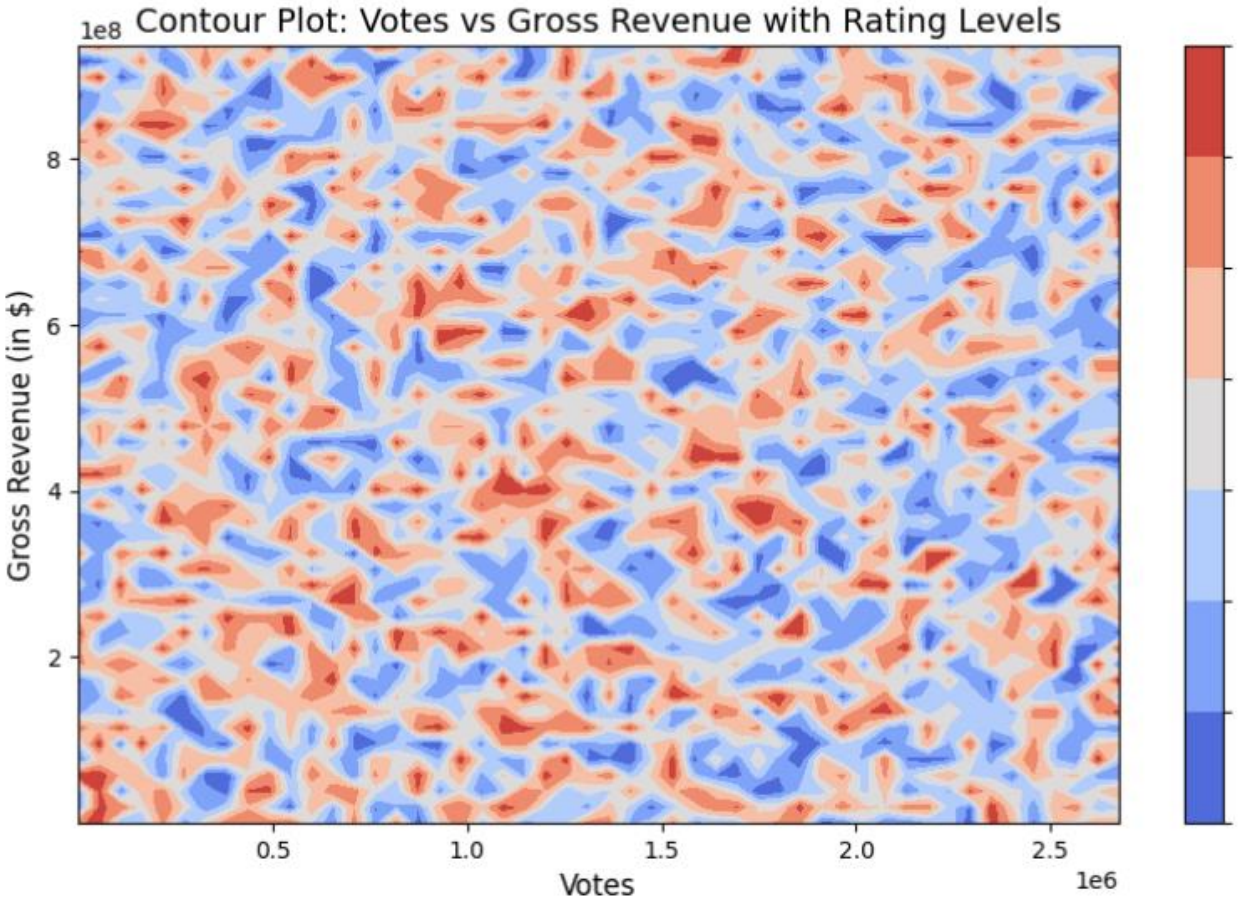


Figure 9.15: Contour plot

- Runtime vs Rating Hexbin Plot: The hexbin plot (Fig 9.16) shows the count of movies by runtime and rating where yellow color shows the maximum count of movies. There is a noticeable count of movies where the rating is from 5 to 8 and runtime is between 80 to 150 minutes.

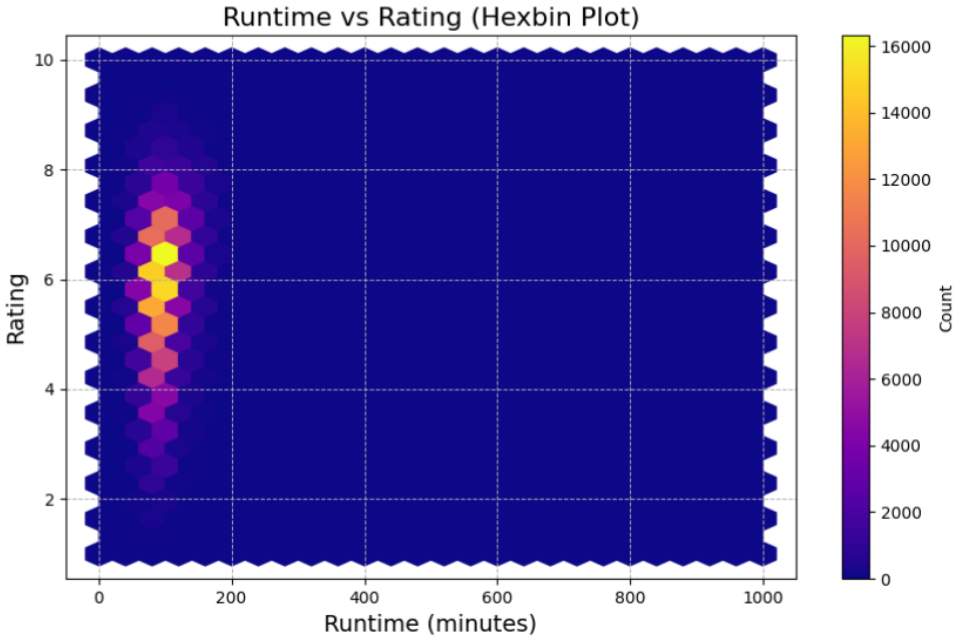


Figure 9.16: Hexbin Plot

- Strip Plot (Rating vs Genre and Runtime vs Genre):

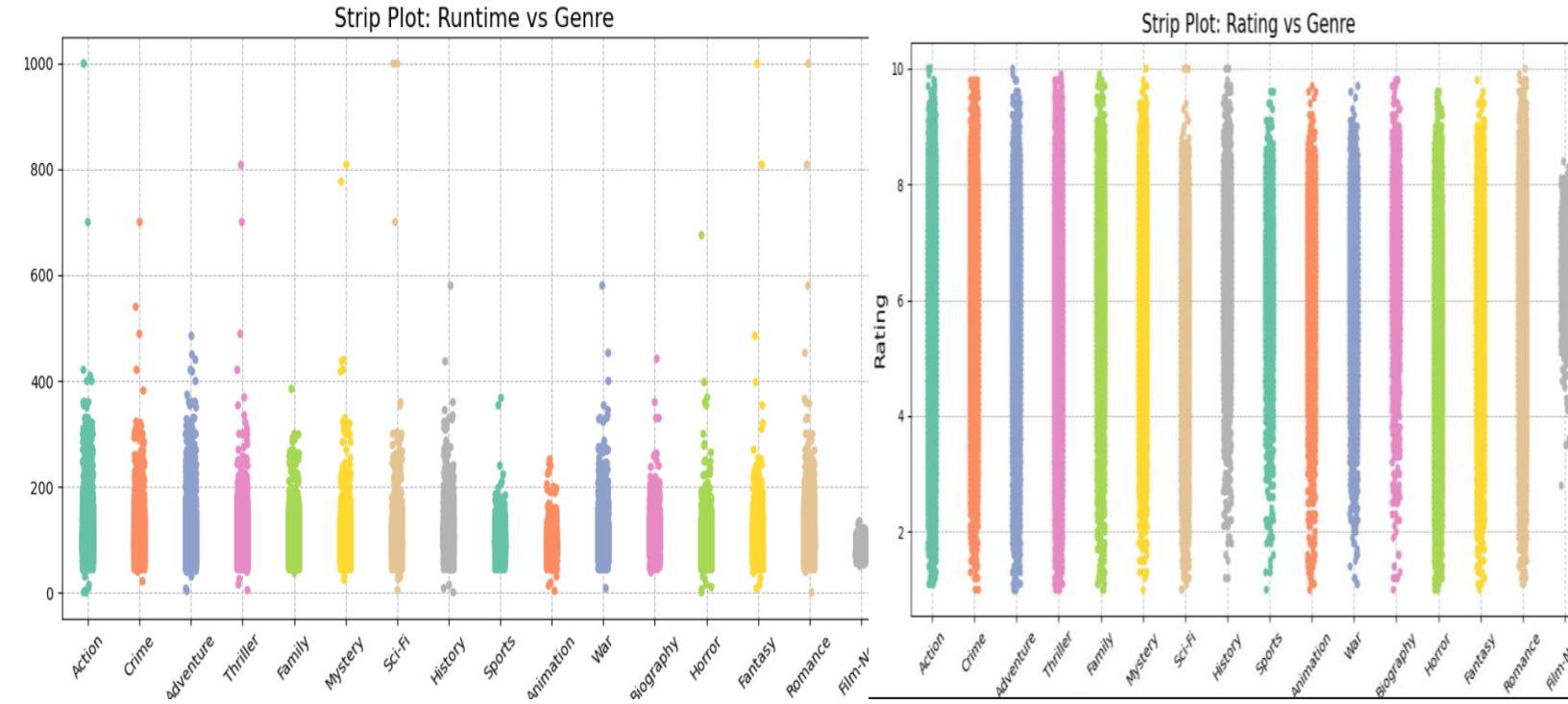


Figure 9.17: Strip plot for rating and runtime vs genre

The noticeable thing in the plot (Fig 9.17) is that the spread of runtime for each genre shows the outliers with more than two hundred minutes of runtime. However, for rating, the values are always between 0-10 which shows there are no outliers as well as rating is almost constant for all the genres.

- Swarm Plot: The plot shows us the spread of datapoints for each genre (Fig 9.18).

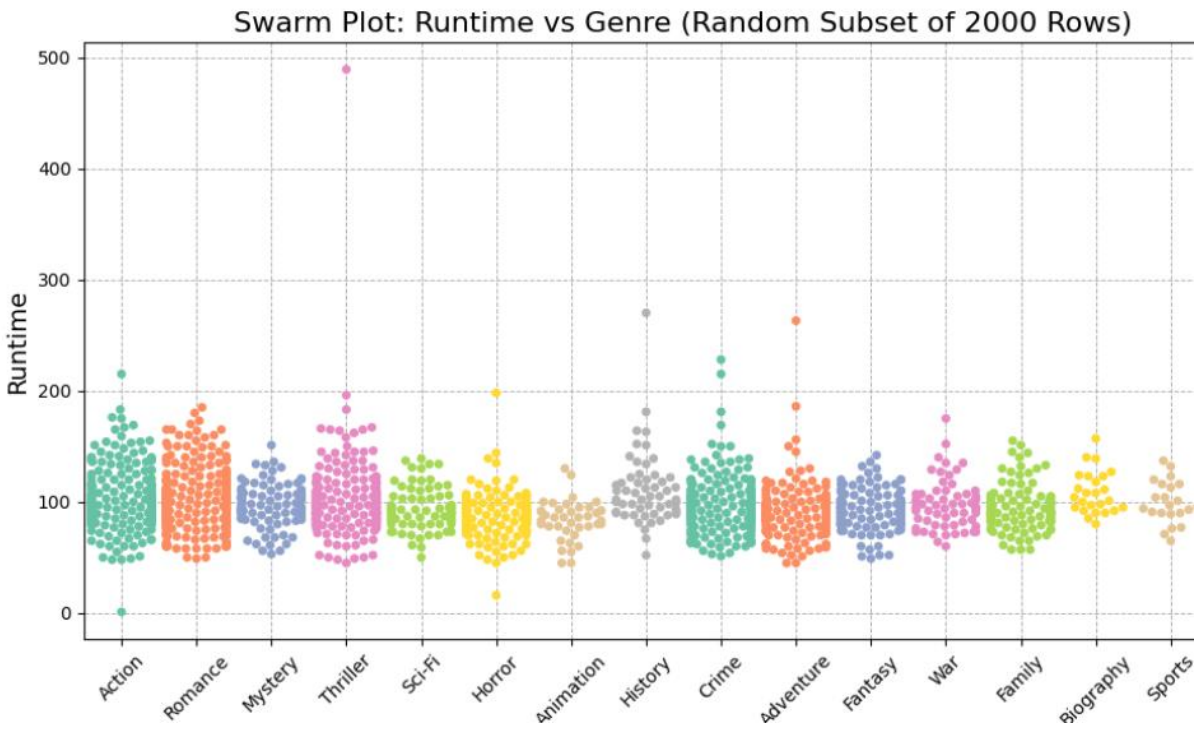


Figure 9.18: Swarm Plot

Subplots

- Violin Plot vs Box Plot

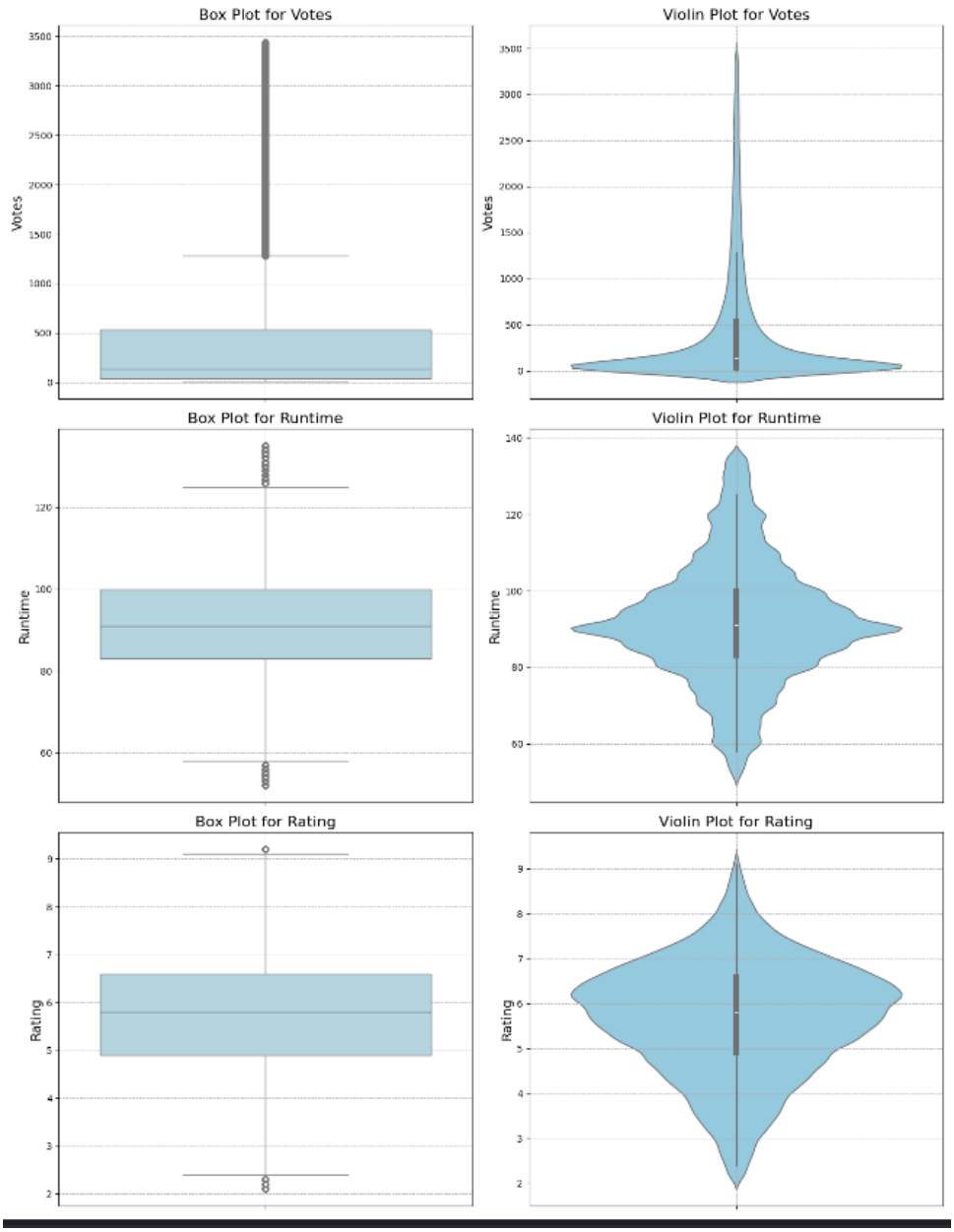


Figure 10.1: Subplots for removed outliers.

This plot shows how the box plot and violin plot are related to each other. Box plot shows only the summary whereas violin plot shows the skewness of the whole data. Boxplot is more appropriate for clearly showing the outliers, but violin plot puts emphasis on shape distribution (Fig 10.1).

- Different Kind of Transformation: The figure 10.2 shows how log transformation, square root transformation and Yeo-Jhonson Transformation affect the dataset's distribution. Log transformation makes the distribution more close to normal distribution but square root transformation makes it more skewed. The yeo-

Johnson transformation also makes the data more towards normal distribution. We can compare the two figures before and after (Fig 10.2 vs Fig 10.3) and see how each transformation works.

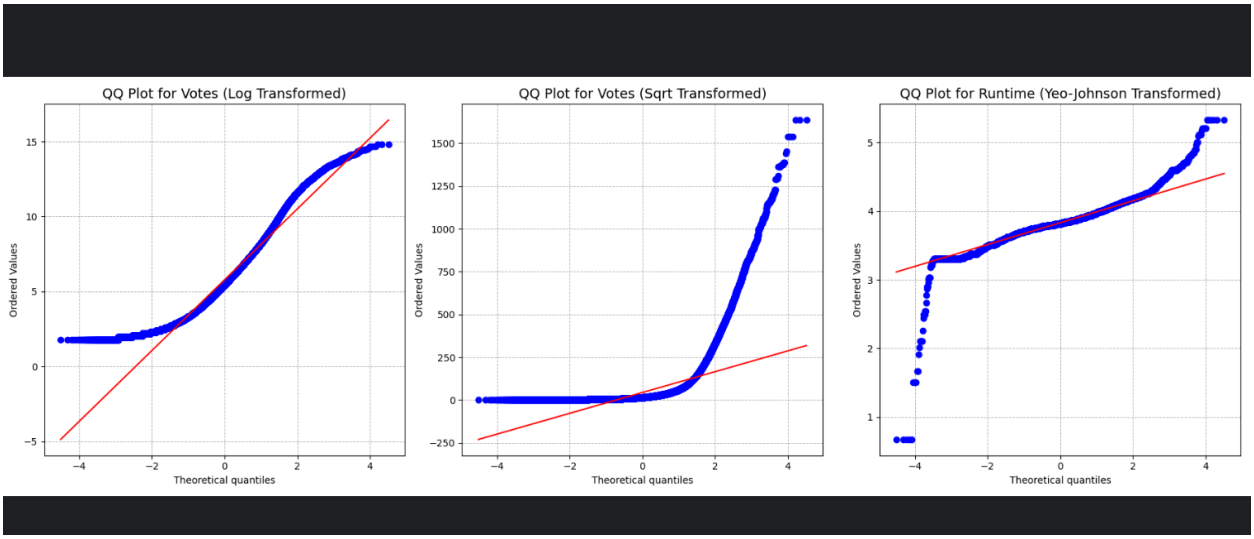


Figure 10.2: QQ plot for different transformation

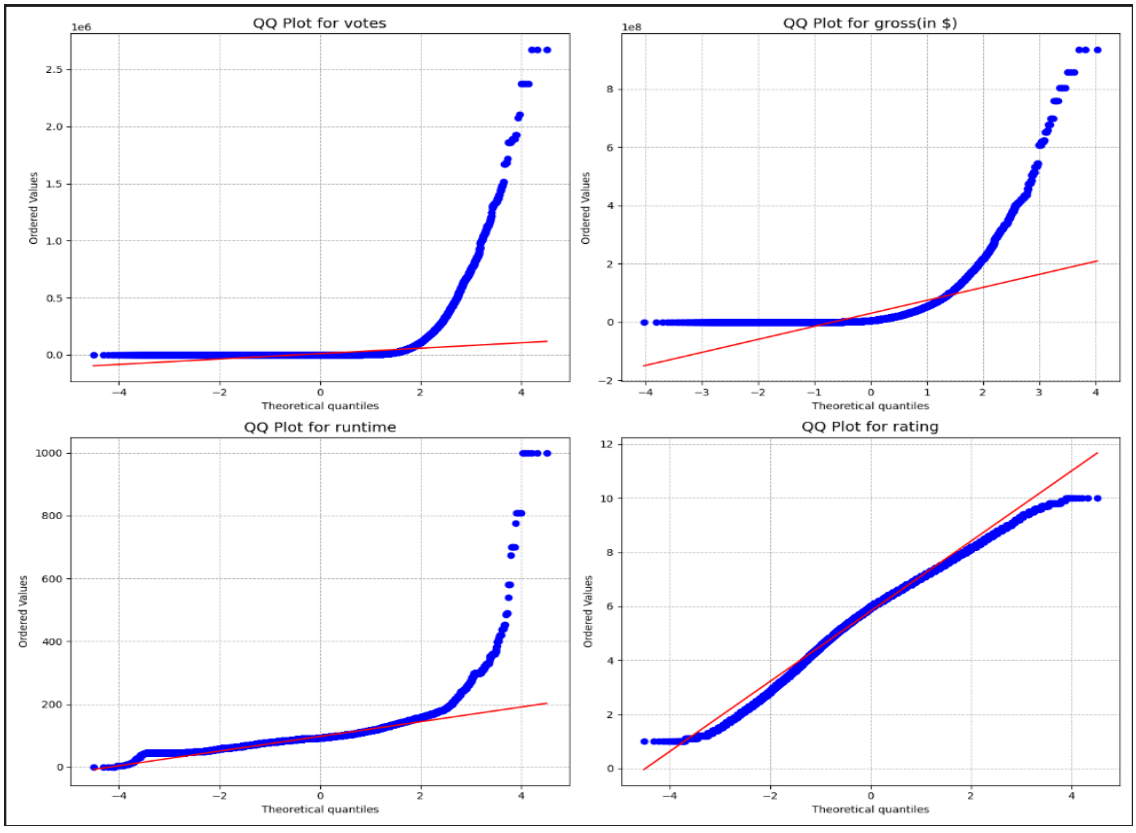


Figure 10.3: Before transformation

- Numerical Feature Distribution:

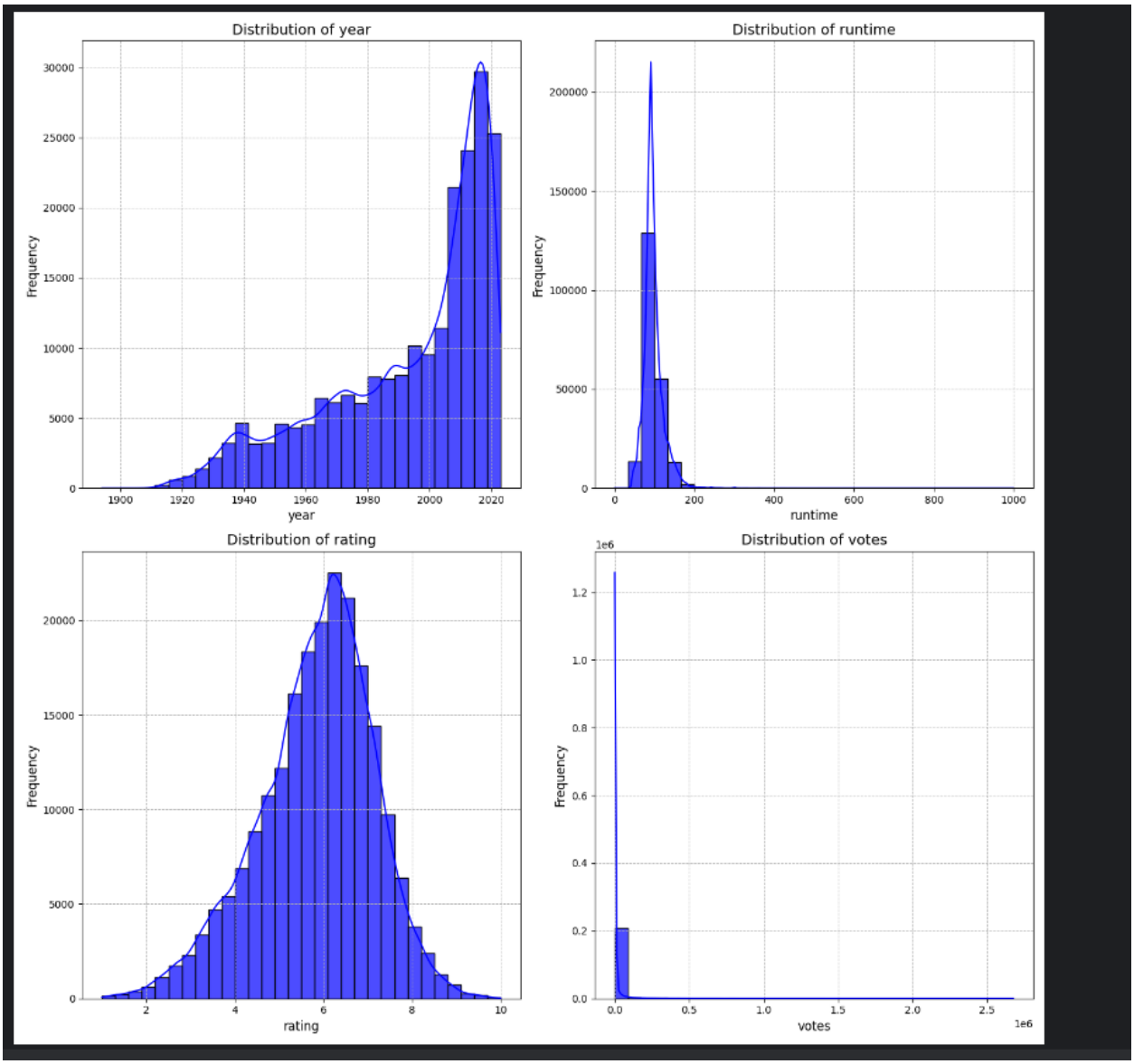


Figure 10.4: Numerical Feature Distribution

The comparison shows how the numerical features are distributed in the data and if there are any potential outliers.

Tables and Observations

- Table for rating trend over time:

Observation	Explanation
Initial Fluctuations	The earlier years show high variability in ratings, possibly due to fewer movies or inconsistent audience standards.
Mid-Century Stability	Ratings stabilize between 1940 and 1970, suggesting consistent audience reception and production quality.
Recent Decline	A gradual decline in ratings from the 1990s to 2010s reflects shifts in audience expectations or market saturation.
Recent Peak	The sharp rise in the most recent years may reflect outliers or highly rated modern movies.

- Table for votes and gross over time:

Feature	Observation
Votes	Shows a steady, minimal increase in audience engagement over time. However, it remains relatively flat, likely due to
Gross	Highlights significant peaks corresponding to blockbuster years. Notably, gross increases substantially in recent decades, refl
Comparison	The correlation between votes and gross is minimal in earlier years but more pronounced in later years. Peaks in gross

- Table for Average Gross revenue by genre:

Average Gross Revenue by Genre:

Observation	Explanation
Top Genre	Animation has the highest average gross revenue, suggesting that animated movies are highly profitable.
Other High Performers	Adventure, Sci-Fi, and Family genres also perform well, reflecting their broad audience appeal.
Low Performers	Genres like Film-Noir and History generate the lowest average gross revenue, likely due to niche audiences.

- Average Rating by Genre and Top 5 certificate:

Average Ratings by Genre and Certificate (Top 5 Certificates):

Observation	Explanation
Certificate Differences	Certificates like PG-13 and R have more consistent ratings across genres.
High Variability	Some genres, such as Animation and Sci-Fi, show higher ratings for PG-certified movies.
Genre Trends	Genres like Romance and Biography have lower average ratings compared to Animation and Fantasy.

- Number of movies by genre and top 5 certificate:

Number of Movies by Genre and Certificate (Top 5 Certificates):

Observation	Explanation
Top Genres	Thriller, Romance, and Action genres dominate in terms of the total number of movies.
Certificate Distribution	Most movies are either 'Not Rated' or rated R, suggesting broader coverage across genres.
Underrepresented Genres	Genres like Sports, Animation, and Film-Noir have fewer movies, indicating niche production.

- Number of Movies by genre:

Number of Movies by Genre:

Genre	Observation
Romance	The most common genre, indicating a strong audience preference for romantic films.
Thriller	Thriller is the second most frequent genre, suggesting high demand for suspenseful movies.
Action	Action ranks high, reflecting its broad appeal and box-office popularity.
Film-Noir	The least frequent genre, showing its niche nature and limited production.
Sports	Another less frequent genre, likely targeting a smaller, dedicated audience.

- Most Common Genres over Decade:

Most Common Genres Over Time (Stacked Area Plot):

Observation	Explanation
Dominance of Modern Era	The number of movies has increased dramatically after the 1980s, with the highest contributions from Action, Thr
Steady Growth	Genres like Adventure, Fantasy, and Sci-Fi show consistent growth over time, reflecting audience demand for in
Niche Genres	Genres like Film-Noir and Sports have remained niche with minimal contributions across yea
Post-2000 Spike	A significant increase in the diversity of genres is visible after 2000, likely due to advancements in global cine

- Distribution of rating:

Distribution of Ratings:

Observation	Explanation
Central Tendency	The majority of ratings are concentrated between 5.5 and 7.0, indicating most movies are rated average to slightly above
Skewness	The distribution is slightly left-skewed, with fewer movies receiving extremely low or high ratings.
Peak Frequency	The peak of the distribution occurs around 6.5, showing this as the most common rating.
Outliers	A small number of movies have ratings below 3.0 or above 8.5, representing poorly received or highly acclaimed film

- Subplot for Numerical Features:

Subplot Explanations:

Plot	Observation
Distribution of Year	Steady increase in the number of movies over time.
Distribution of Runtime	Most movies have runtimes between 90 and 120 minutes.
Distribution of Rating	Ratings are concentrated between 5.5 and 7.0.
Distribution of Votes	Highly skewed with most movies receiving fewer votes.

Explanation
The number of movies produced has grown exponentially since the 1980s, indicating a significant expansion of the film industry.
The runtime distribution is right-skewed, with a few outliers exceeding 300 minutes, representing exceptionally long movies.
This shows that most movies receive average to slightly above-average audience reception, with very few highly rated or poorly rated films.
A small number of blockbuster movies dominate the vote count, while the majority have minimal audience engagement.

- Parplot Exlpanation:

Pair Plot Explanation:

Feature Pair	Observation
Rating vs Votes	Strong positive correlation; movies with higher votes tend to have higher ratings.
Gross vs Votes	Moderate positive correlation; movies with higher gross revenue tend to have more votes.
Runtime vs Rating	No clear trend; movies of varying runtimes can achieve high ratings.
Gross vs Rating	Weak positive correlation; movies with higher ratings tend to have higher gross revenue.
Genre Clustering	Distinct clustering patterns observed for some genres (e.g., Animation, Sci-Fi).

- Correlation Matrix:

Feature Pair	Correlation	Explanation
Votes vs Rating	0.15	Weak positive correlation; movies with higher ratings tend to have more votes, but the relationship is
Votes vs Year	0.068	Minimal correlation; the number of votes is not significantly influenced by the year of relea
Votes vs Runtime	0.13	Weak positive correlation; longer movies tend to receive slightly more votes, but the effect is
Rating vs Year	-0.13	Weak negative correlation; recent movies have slightly lower ratings, potentially due to changes in audier
Rating vs Runtime	0.17	Weak positive correlation; longer movies tend to have slightly higher ratings.
Year vs Runtime	0.14	Weak positive correlation; movies in recent years tend to have slightly longer runtimes.

- Rating Distribution by Decade KDE:

Rating Distribution by Decade (KDE Plot):

Observation	Explanation
Consistency Over Decades	The peak rating density for most decades lies around 6, showing consistent audience evaluati
Shift in Density	Movies from the 2000s and later decades show slightly wider distributions, indicating more
Low Ratings in Early Decades	Early decades like the 1900s and 1910s have a narrower distribution and lower density, reflecting limite
Broader Range in Recent Decades	Recent decades (e.g., 1990s, 2000s) exhibit broader and higher density peaks, indicating higher movie p
Audience Preference Stability	The central tendency of ratings hasn't shifted significantly, suggesting audience preference for average t

- Regplot:

Observation	Explanation
Positive Correlation	There is a general positive correlation between votes and gross revenue across genres. Movies with higher
Genre-Based Trends	The slopes of regression lines differ by genre, indicating that the relationship between votes and
Outliers	Certain movies (outliers) with high gross revenue do not have correspondingly high votes, possibly due to bloc
Uncertainty Bands	Shaded areas around regression lines represent confidence intervals, showing the uncertainty of
Dominance of Certain Genres	Some genres, like Action and Adventure, appear to have stronger correlations between votes and gross rever

- Jointplot :

Observation	Explanation
Runtime Concentration	Most movies have a runtime between 50 to 200 minutes, indicating a standard length for f
Rating Concentration	The majority of movies are rated between 5.5 and 7.0, reflecting audience preferences for moderate
Density Patterns	The highest density occurs for runtimes of 80 to 120 minutes with ratings around 6.0 to
Outliers	Outliers exist for movies with very long runtimes (over 400 minutes) and varying ratings, suggesting
Marginal Histograms	The runtime distribution is skewed towards shorter movies, and ratings are slightly right-skewed, centerir
Runtime and Rating Relationship	No strong linear relationship between runtime and rating; ratings are consistent across differe

- Director Top 5:

Observation	Explanatio
Trends Over Time	The line plot shows how average ratings of the top 5 directors evolved over decades. Some directors like Richard Tho
Performance Stability	Directors like Godfrey Ho and Jesús Franco have consistent patterns, ir
Rating Variability	The box plot reveals variability in ratings for each director. Richard Thorpe has a wider range
Outliers	Directors like Richard Thorpe and William Beaudine have significant outlie
Director Comparison	Jesús Franco has the lowest median ratings among the top 5 directors,

- QQ Plots (Before Transformation):

Feature	Observation
Votes	The QQ plot for votes shows a strong deviation from the theoretical normal line, particularly in the upper quantiles. This
Gross (in $)	The gross revenue plot also deviates significantly from the normal distribution, showing heavy upper tails. This suggests the
Runtime	The QQ plot for runtime shows deviations from normality, particularly in the upper quantiles. Movies with exceptic
Rating	The QQ plot for ratings follows the normal line more closely compared to the other features. However, minc

- QQ Plots (After Transformation):

Feature	
Votes (Log Transformed)	The log transformation of votes reduces the skewness, as seen in the QQ plot. However, deviations from t
Votes (Sqrt Transformed)	The square root transformation also reduces skewness but is less effective compared to the l
Runtime (Yeo-Johnson Transformed)	The Yeo-Johnson transformation effectively reduces skewness in runtime, as the QQ plot aligns mor

- Hexbin Plot (Rating and Runtime by year):

Aspect	Observation
Runtime Concentration	Most movies have runtimes between 80 to 150 minutes, as indicated by the densest yellow hexagons in t
Rating Concentration	Ratings are densely concentrated between 5 and 7 for most movies, aligning with standard audience
Outliers	The sparse distribution of hexagons beyond 200 minutes suggests there are very few long movies, and these tend to
Color Intensity	The color bar indicates the count of movies in each hexagon. The bright yellow areas represent the highest c
Insights	The majority of movies have standard runtimes (~90-150 minutes) and average ratings (~6), with very few extremes in

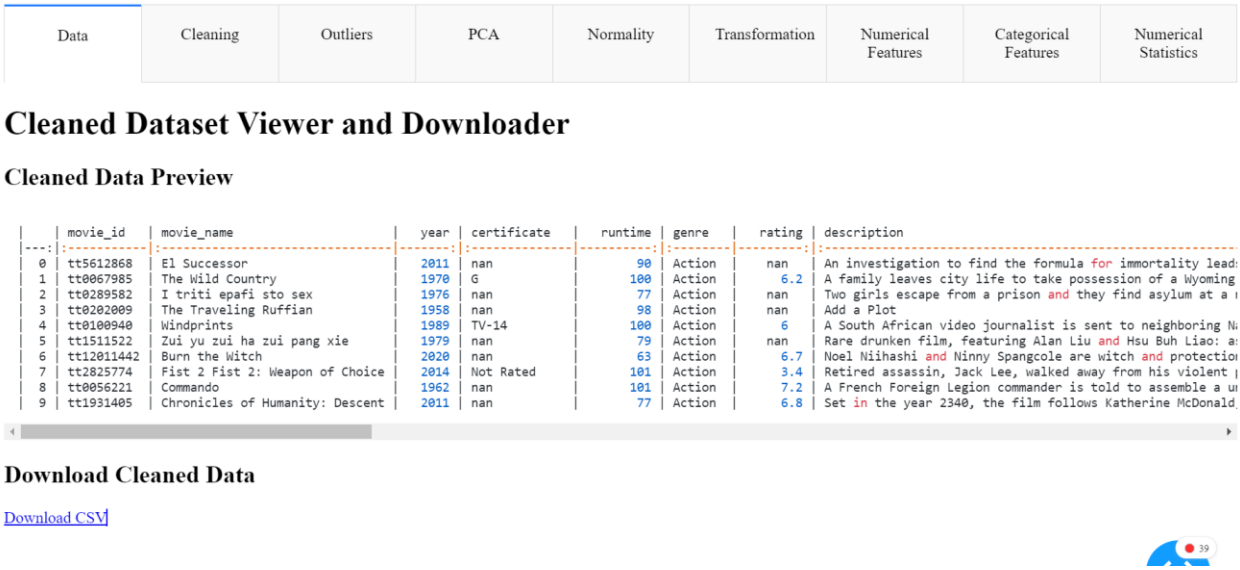
- Comparison Between 3D and Contour Plot:

3D Plot	Contour Plot
Visualizes the relationship between Votes, Gross Revenue, and Rating in 3D space.	Represents density or level variations in Votes and Gros
Scatter points in 3D space showing clustering and spread.	Color-coded contours to show density and
Captures complex relationships between three variables clearly in 3D.	Easier to interpret density and level-based pat
May be harder to interpret due to overlapping points and 3D perspective.	Does not show individual data points explicit
Analyzing clusters or trends when working with three numerical variables.	Visualizing density distributions or level varia

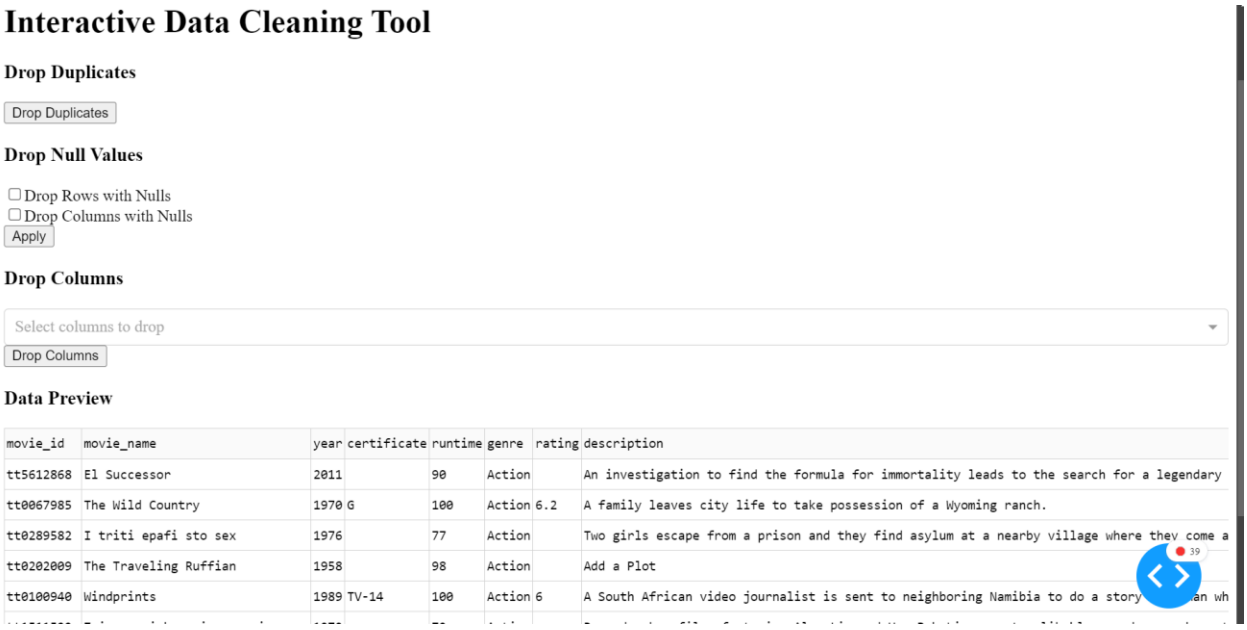
Dashboard

In Phase II, a dashboard using Dash was developed. A different dataset with 16000 observations were made where each genre had one thousand observations. The dashboard has 9 tabs:

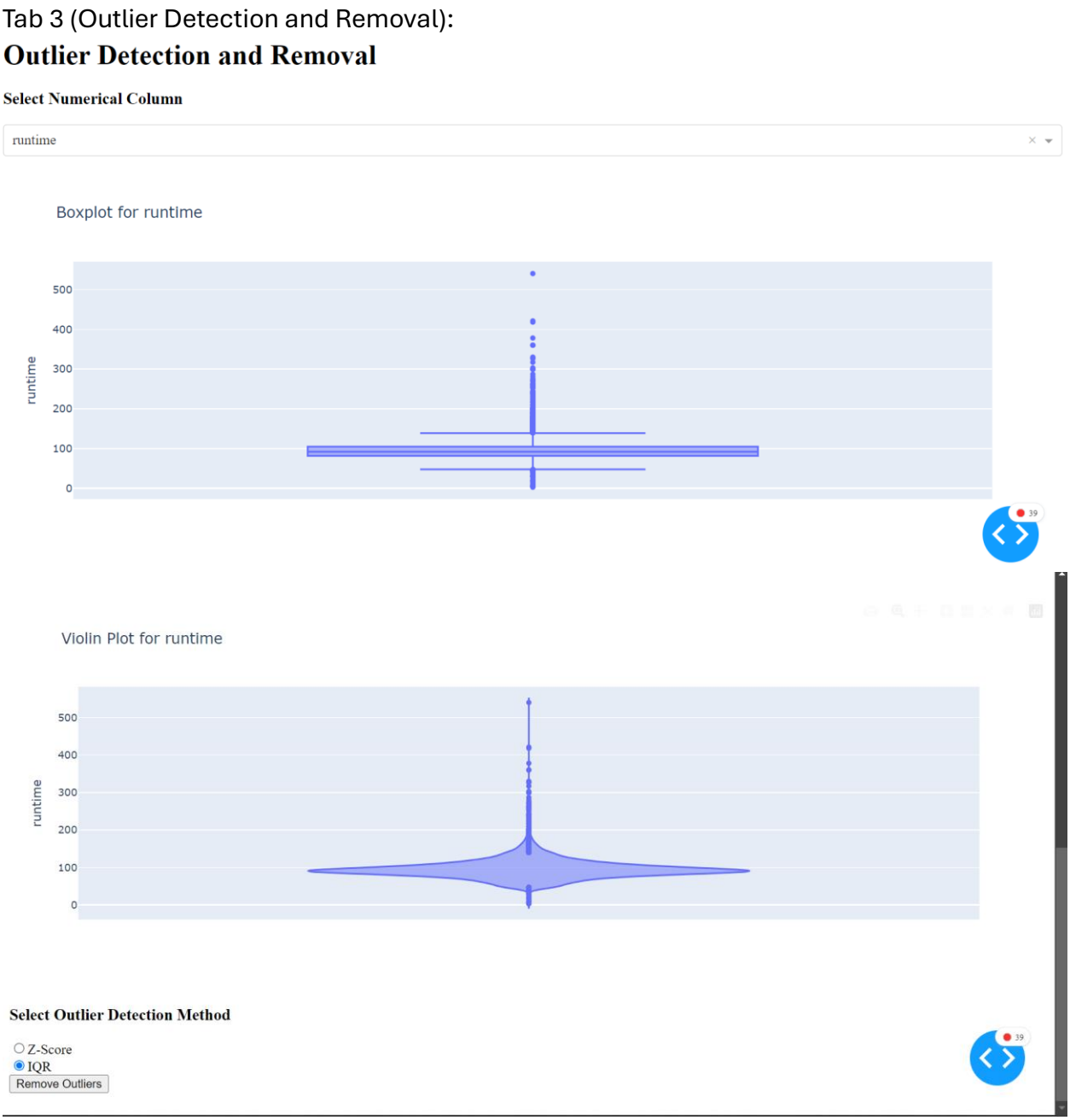
1. TAB 1 (Dataset and Download Dataset):



2. TAB 2 (Data Cleaning Tools):

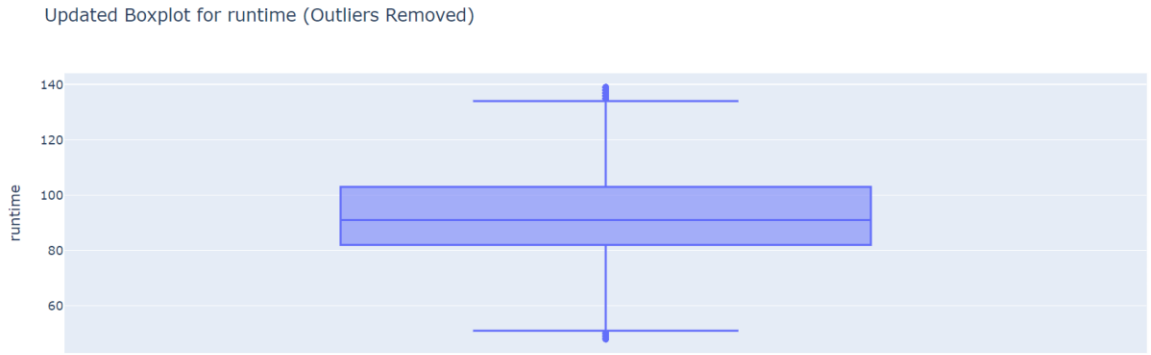


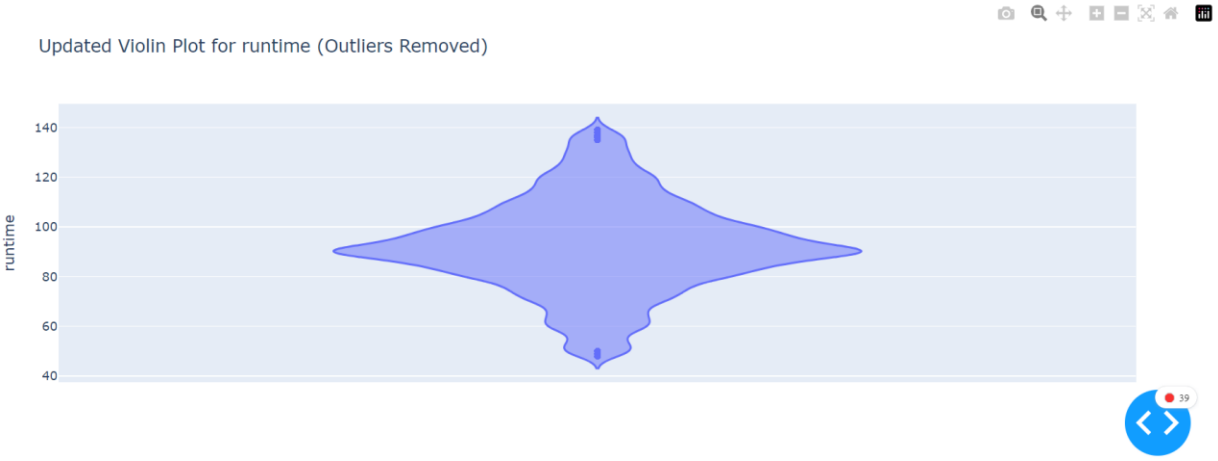
3. Tab 3 (Outlier Detection and Removal):



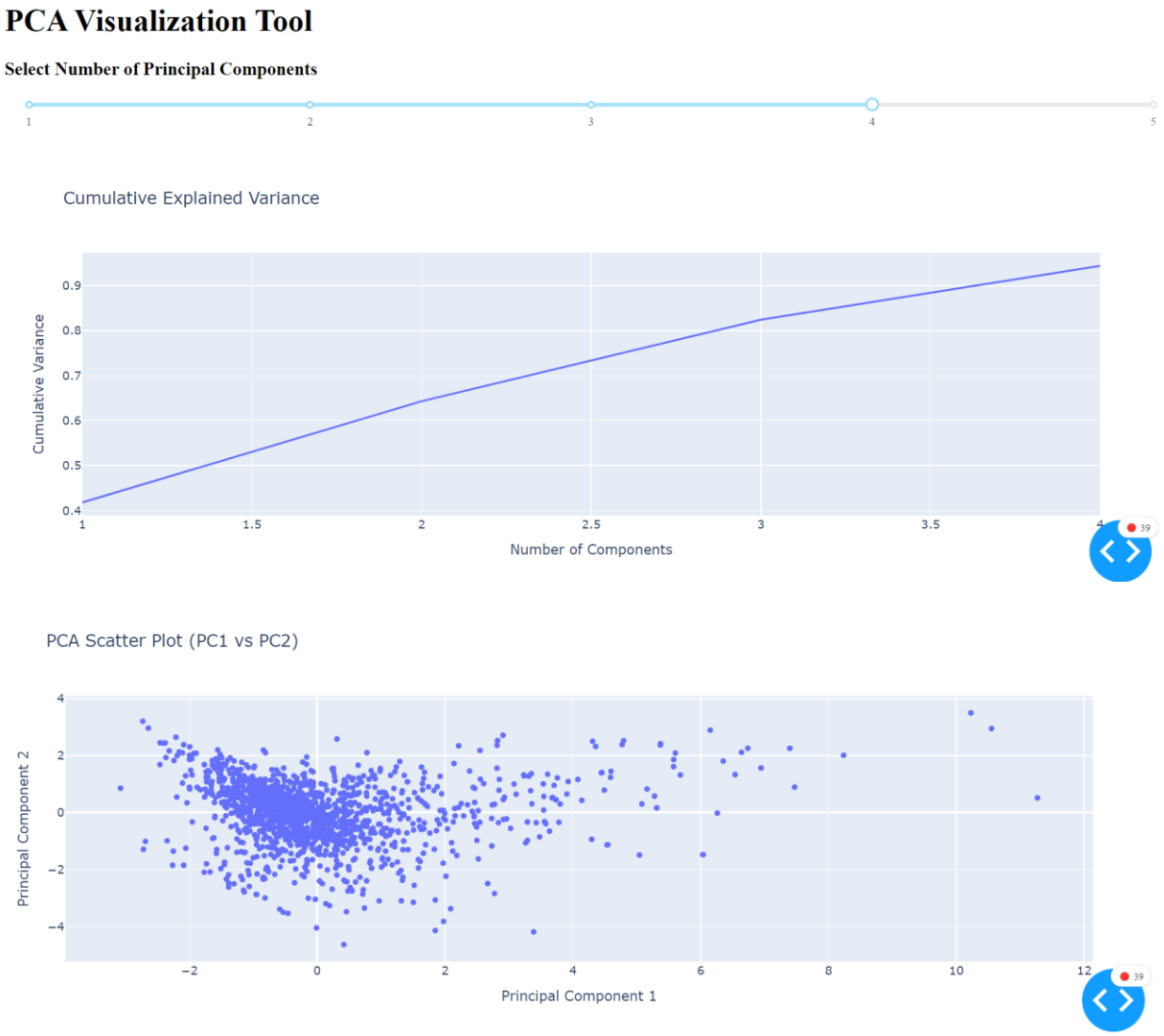
Select Outlier Detection Method

○ Z-Score
◉ IQR
Remove Outliers



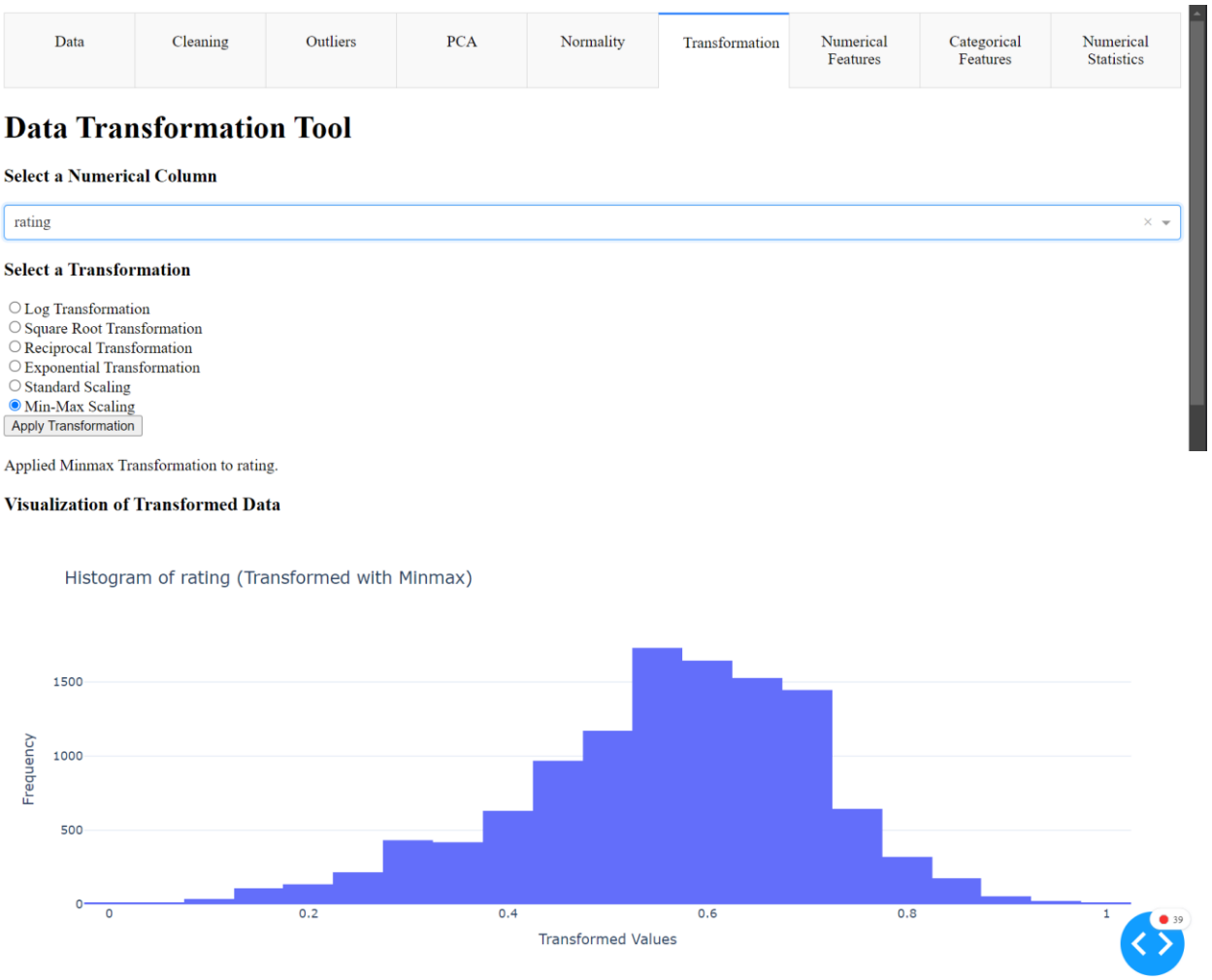


4. Tab 4 (PCA):

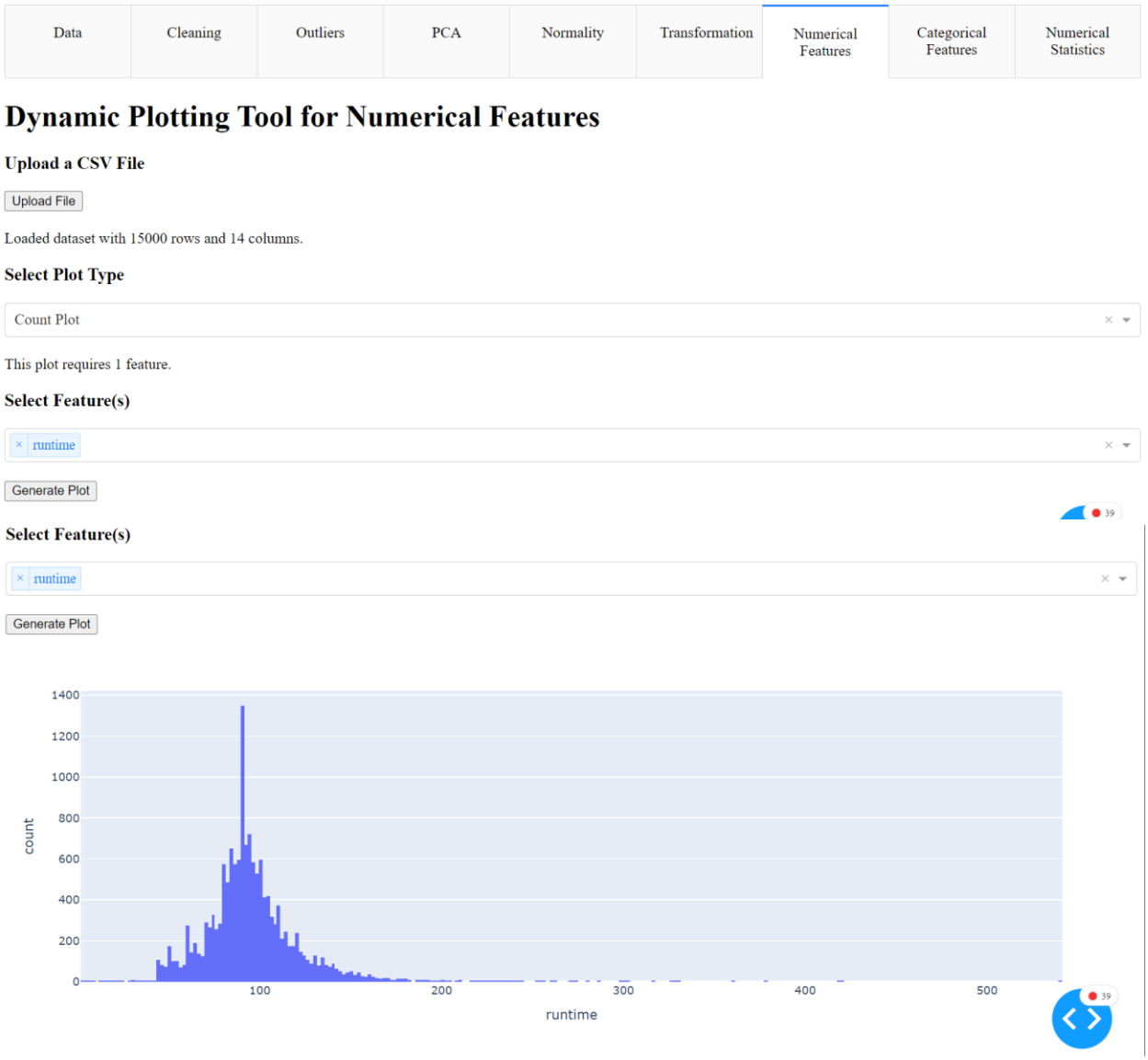

PCA Visualization Tool
Select Number of Principal Components
1
2
3
4
5
Cumulative Explained Variance
Cumulative Variance
0.9
0.8
0.7
0.6
0.5
0.4
1
1.5
2
2.5
3
3.5
Number of Components
PCA Scatter Plot (PC1 vs PC2)
Principal Component 2
4
2
0
−2
−4
−2
0
2
4
6
8
10
12
Principal Component 1

5. Tab 5 (Normality Test with QQ plot):

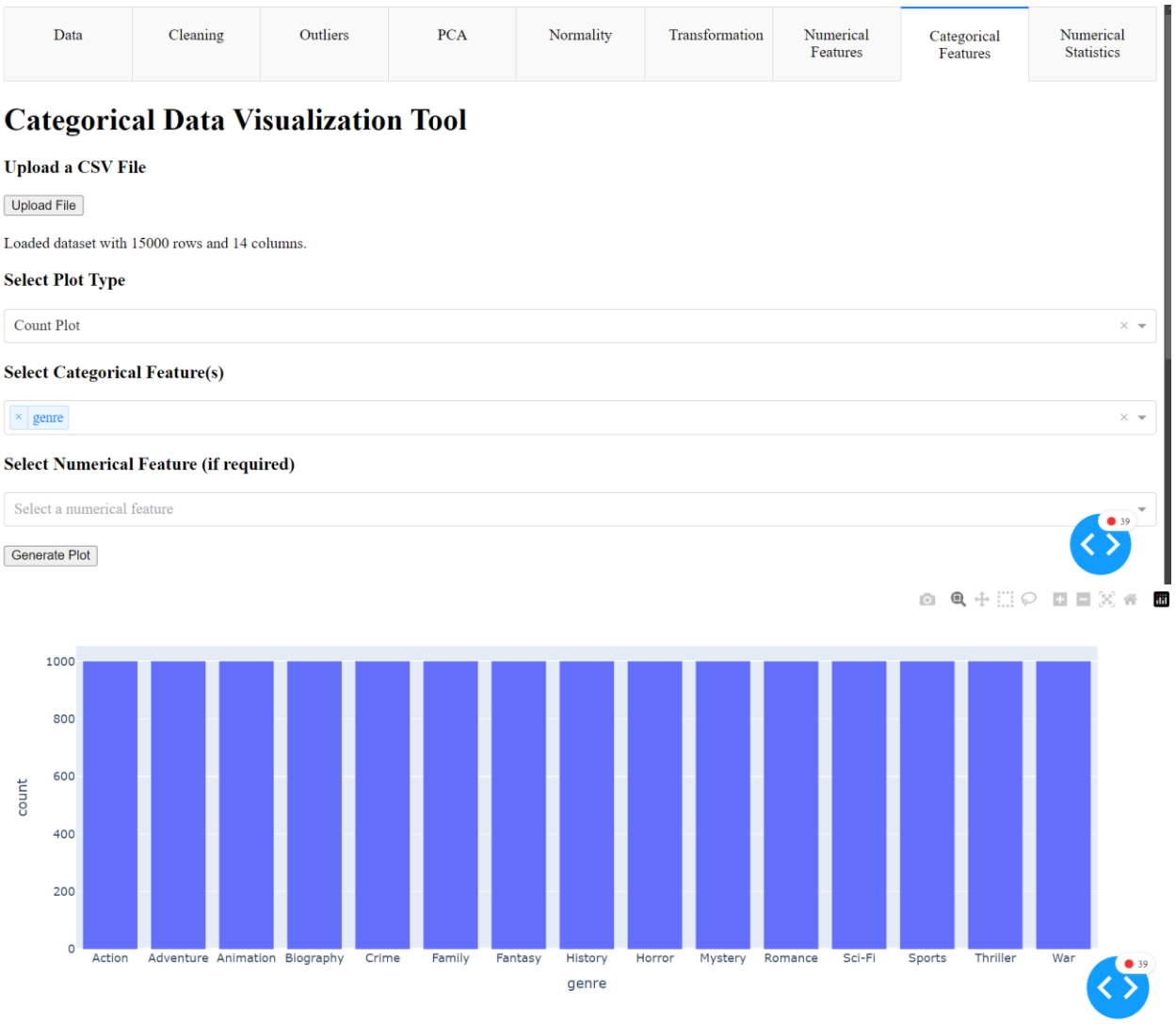
6. Tab 6 (Data Transformation Tool):



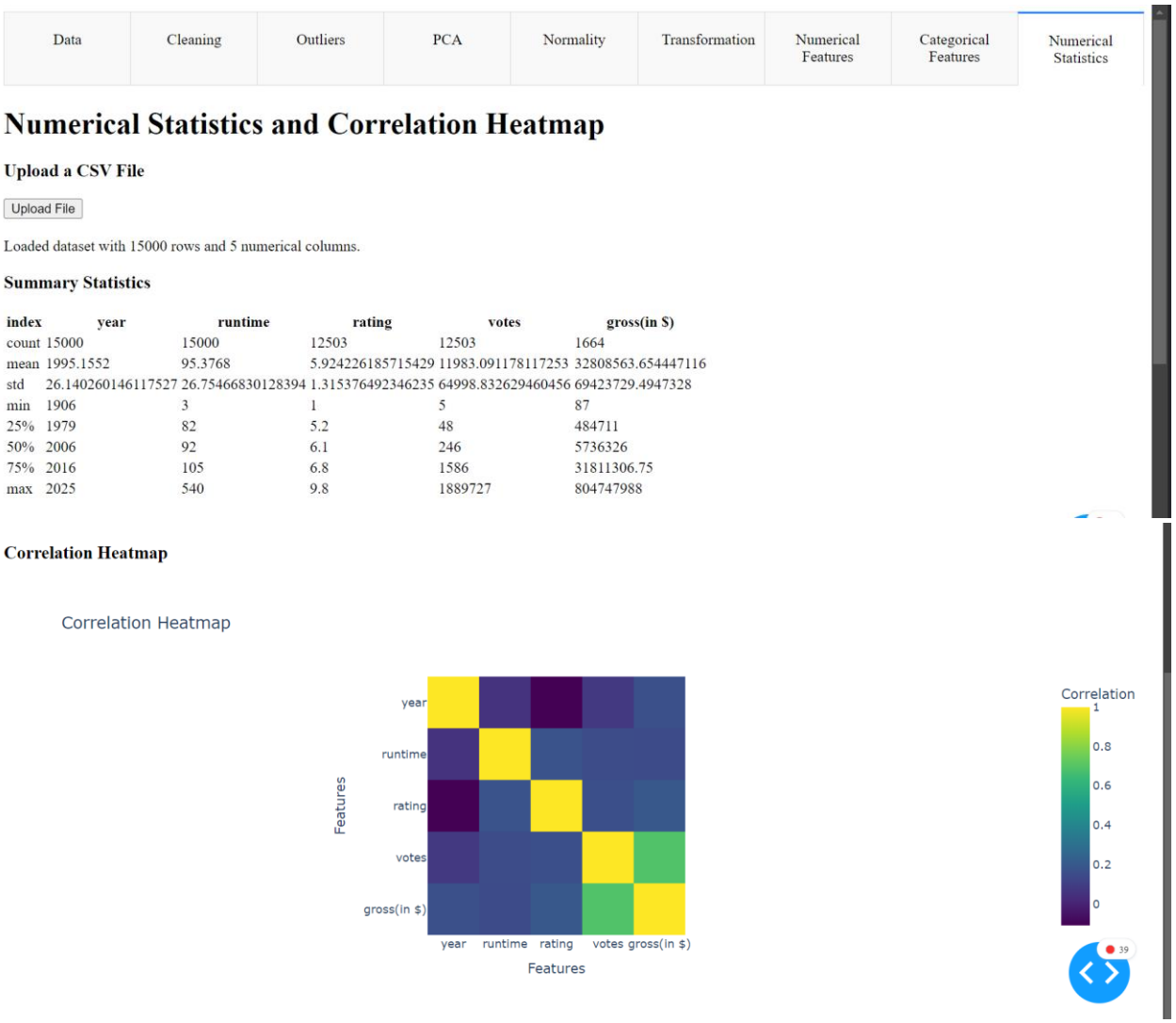
7. Tab 7 (Dynamc Plot for Numerical Features):



8. Tab 8(Plotting for categorical features):



9. Tab 9 (Statistics):



Conclusion

This project explored a comprehensive approach to data visualization, dimensionality reduction, and statistical analysis using a dataset of movies spanning various genres and attributes. The implementation of advanced Python-based techniques across the three project phases has demonstrated the power of data-driven insights and interactive tools in understanding complex datasets.

Key Learnings

1. **Data Insights**:

- The analysis revealed significant trends, such as the increasing number of movies produced over time, the skewed distribution of audience votes, and the average runtime and rating patterns.
- The correlation analysis highlighted relationships between key features, such as the inverse correlation between rating and release year, suggesting a potential bias toward older movies or shifting audience expectations.

2. **Dimensionality Reduction and Data Preparation**:
    - Principal Component Analysis (PCA) effectively reduced the dataset's dimensionality while retaining most of the variance, ensuring a more computationally efficient and interpretable representation of the data.
    - Transformations, such as log, square root, and Yeo-Johnson transformations, normalized skewed features and improved their suitability for modeling.
3. **Deployment and Usability**:
    - The interactive dashboard was successfully containerized and deployed on Google Cloud Platform (GCP), demonstrating the project's scalability and real-world applicability.
    - The dashboard is user-friendly, offering essential functionalities for exploring and analyzing the dataset.

This project highlights the importance of combining static and interactive visualization with the future scope like running a machine learning model in the which can be also implemented in the dashboard.

References:

- Rajugopal, R. (n.d.). *IMDb movies dataset based on genre* [Data set]. Kaggle. Retrieved December 13, 2024, from https://www.kaggle.com/datasets/rajugc/imdb-movies-dataset-based-on-genre/data
- Maryadi, R. (n.d.). *IMDB movie data* [Code and analysis]. Kaggle. Retrieved from https://www.kaggle.com/code/rizkymaryadi/imdb-movie-data
- Plotly Technologies Inc. (2015). Dash [Computer software]. Retrieved from https://dash.plotly.com

- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Duchesnay, É. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12*, 2825–2830. Retrieved from https://scikit-learn.org
- Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. *Nature, 585*(7825), 357–362. https://doi.org/10.1038/s41586-020-2649-2
- Waskom, M. L. (2021). seaborn: Statistical data visualization. *Journal of Open Source Software, 6*(60), 3021. https://doi.org/10.21105/joss.03021
- Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing in Science & Engineering, 9*(3), 90–95. https://doi.org/10.1109/MCSE.2007.55
- Virtanen, P., Gommers, R., Oliphant, T. E., Haberland, M., Reddy, T., Cournapeau, D., ... & van der Walt, S. J. (2020). SciPy 1.0: Fundamental algorithms for scientific computing in Python. *Nature Methods, 17*, 261–272. https://doi.org/10.1038/s41592-019-0686-2