# Final Group Project: Updated Checkpoint Submission

Group 7

2024-04-10

## Introduction

This document contains the updated checkpoint submission for our Final Group Project. We have thoroughly reviewed the feedback provided and incorporated suggested actions to enhance our project. This document includes all the code and visualizations used in our analysis.

## Data Origin

1. **MovieLens-GroupLens Research**

   Most of our current data is from the small version of the MovieLens dataset, collected and provided by GroupLens Research. The dataset includes several files, such as `movies.csv`, `ratings.csv`, and `links.csv`, which contain information about movies, user ratings, and external links to other movie databases. The MovieLens dataset is regularly updated and comes in different sizes, including the small version with 100,000 ratings applied to 9,000 movies by 600 users. The datasets are available for download from the GroupLens website.

2. **IMDb**

   We wrote a Python crawler to scrape cast data from IMDb, obtaining `cast.csv`. IMDb is a comprehensive movie database that provides information about movies, TV shows, and celebrities. Our crawler extracted cast details for each movie, allowing us to enrich our dataset with additional information about the actors and actresses involved in the films. This data complements the MovieLens dataset, providing a more complete picture of the movies and their cast members.

## Research Questions

1. Does the variety of content, as measured by the number of genres, influence the overall rating of movies?
2. Which genres have been more popular among viewers in the past three decades?
3. Which genres have generated more revenue and interest compared to others?
4. Does the popularity of top cast members necessarily lead to a larger consumer base and business success for movies?
5. When comparing movies where an actor is cast in their major genre versus movies where they are cast in other genres, which category is more likely to be welcomed by audiences and considered a business success?

## Import Libraries and Data

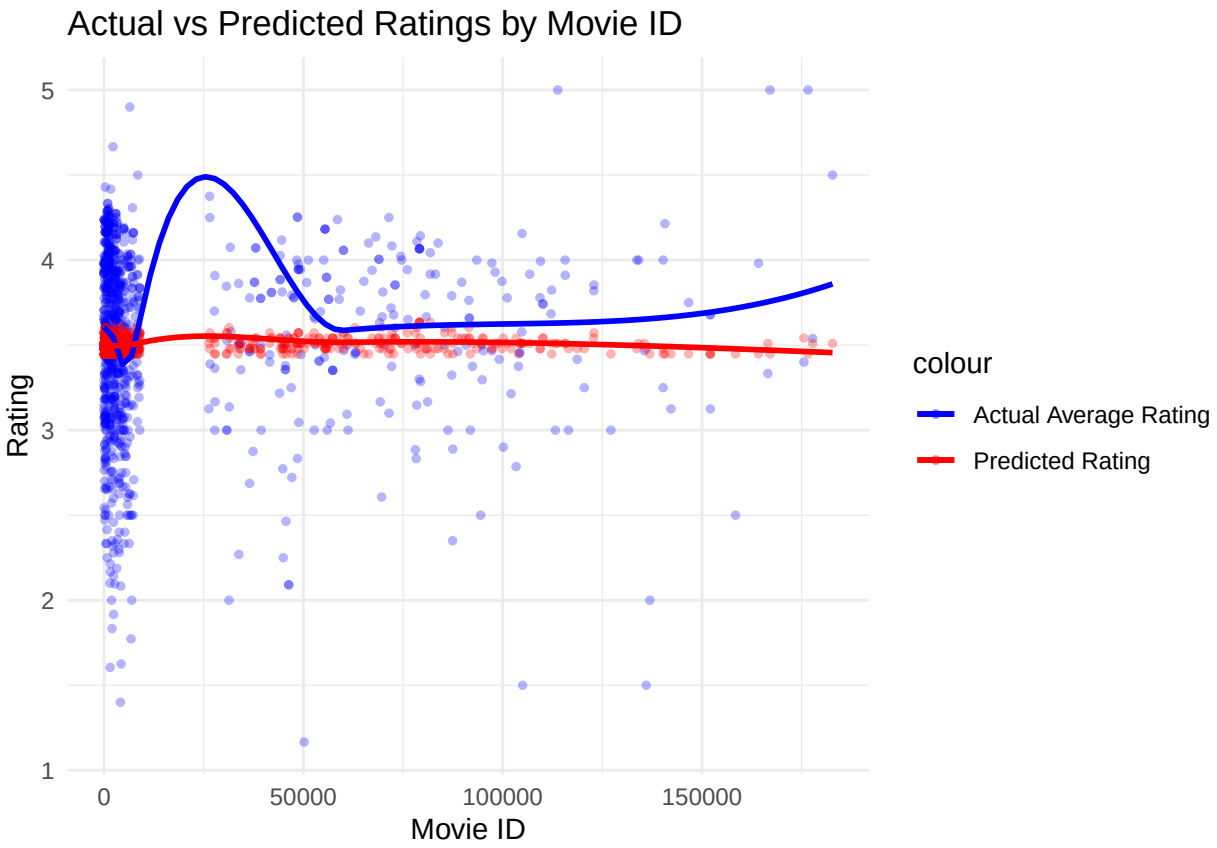
```
library(dplyr)
library(tidyverse)
library(ggplot2)

# Read data
links <- read.csv("data/links.csv")
```

```r
movies <- read.csv("data/movies.csv")
ratings <- read.csv("data/ratings.csv")
cast <- read.csv("data/cast.csv")
```

## Model Fitting - Genre Count

To fit our model, we first count the number of genres each movie belongs to and create a new column called `genre_count`. Next, we calculate the average rating for each movie and store it in a column named `avg_rating`. Using `genre_count` as our independent variable (x) and `avg_rating` as our dependent variable (y), we then fit a linear model to explore the relationship between the number of genres a movie belongs to and its average rating.

```r
# Generate x and y columns
merged_data <- merge(x = links, y = movies, by = "movieId")
ratings <- ratings %>%
  group_by(movieId) %>%
  mutate(avg_rating = mean(rating))

merged_data <- merge(x = ratings, y = merged_data, by = "movieId")
merged_data$genre_count <- str_count(merged_data$genres, "\\|") + 1

# Fit model on current data
model.train <- lm(avg_rating ~ genre_count, data = merged_data)
predictions <- predict(model.train, newdata = merged_data)
merged_data$predicted_rating <- predictions

# Sample the data
set.seed(123)
sampled_data <- merged_data[sample(nrow(merged_data), min(1000, nrow(merged_data))), ]

# Plot the ground truth vs predicted values
ggplot(sampled_data, aes(x = movieId)) +
  geom_point(aes(y = avg_rating, color = "Actual Average Rating"), size = 1, alpha = 0.3) +
  geom_point(aes(y = predicted_rating, color = "Predicted Rating"), size = 1, alpha = 0.3) +
  geom_smooth(aes(y = avg_rating, color = "Actual Average Rating"), method = "loess", se = FALSE) +
  geom_smooth(aes(y = predicted_rating, color = "Predicted Rating"), method = "loess", se = FALSE) +
  labs(x = "Movie ID", y = "Rating",
       title = "Actual vs Predicted Ratings by Movie ID") +
  scale_color_manual(values = c("Actual Average Rating" = "blue", "Predicted Rating" = "red")) +
  theme_minimal()
```

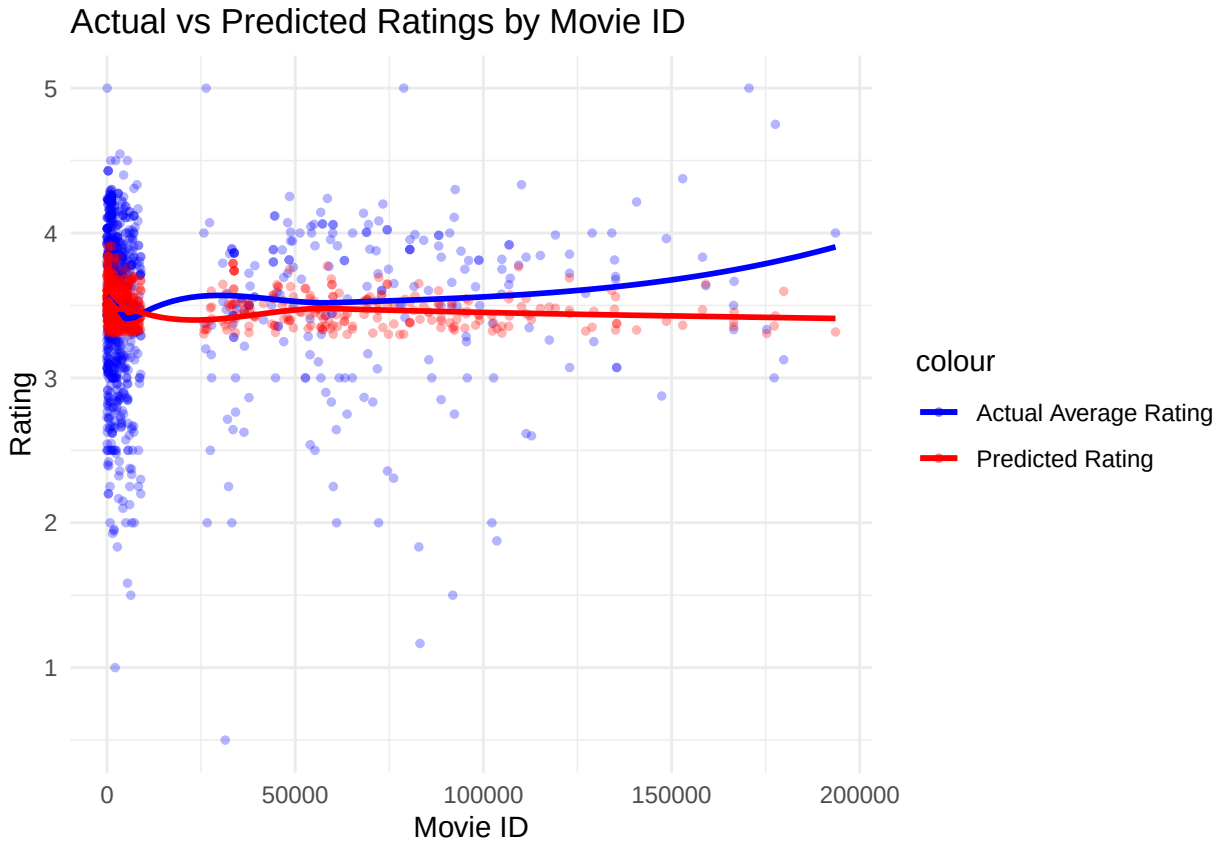



## Model Fitting - Genre Count + Actor Occurrences

In our second attempt, we introduced an actor score to our model, in addition to genre counts. For each actor, we calculated their actor score based on the number of times they appeared in all the movies. For each movie, the actor scores of all its actors were summed to obtain the movie's `actor_score`. We used `genre_count` and `actor_score` as our independent variables (x) and `avg_rating` as our dependent variable (y). A linear model was then fitted to explore the relationship between these variables and the average rating of the movies.

```
# Merge cast with previous data by imdbId
merged_cast_data <- merge(cast, merged_data, by = "imdbId")

# Create a function to count occurrences of each actor
get_actor_frequency <- function(df) {
  # Combine all actor columns and create a vector
  actor_vector <- unlist(df[, grepl("Actor", names(df))])
  # Count occurrences of each actor
  actor_frequency <- table(actor_vector)
  return(actor_frequency)
}

# Apply the function to your merged cast data
actor_frequency <- get_actor_frequency(merged_cast_data)

# Calculate the actor score for each movie
merged_cast_data$actor_score <- apply(merged_cast_data[, grepl("Actor", names(merged_cast_data))], +
                                      1, function(x) sum(actor_frequency[x]))
```

```r
# Fit model on data with both genre count and actor score
model.train <- lm(avg_rating ~ genre_count + actor_score, data = merged_cast_data)
predictions <- predict(model.train, newdata = merged_cast_data)
merged_cast_data$predicted_cast_rating <- predictions

# Sample the data
set.seed(123)
sampled_cast_data <- merged_cast_data[sample(nrow(merged_cast_data), min(1000, nrow(merged_cast_data)))

# Plot the ground truth vs predicted values
ggplot(sampled_cast_data, aes(x = movieId)) +
  geom_point(aes(y = avg_rating, color = "Actual Average Rating"), size = 1, alpha = 0.3) +
  geom_point(aes(y = predicted_cast_rating, color = "Predicted Rating"), size = 1, alpha = 0.3) +
  geom_smooth(aes(y = avg_rating, color = "Actual Average Rating"), method = "loess", se = FALSE) +
  geom_smooth(aes(y = predicted_cast_rating, color = "Predicted Rating"), method = "loess", se = FALSE)
  labs(x = "Movie ID", y = "Rating",
       title = "Actual vs Predicted Ratings by Movie ID") +
  scale_color_manual(values = c("Actual Average Rating" = "blue", "Predicted Rating" = "red")) +
  theme_minimal()
```

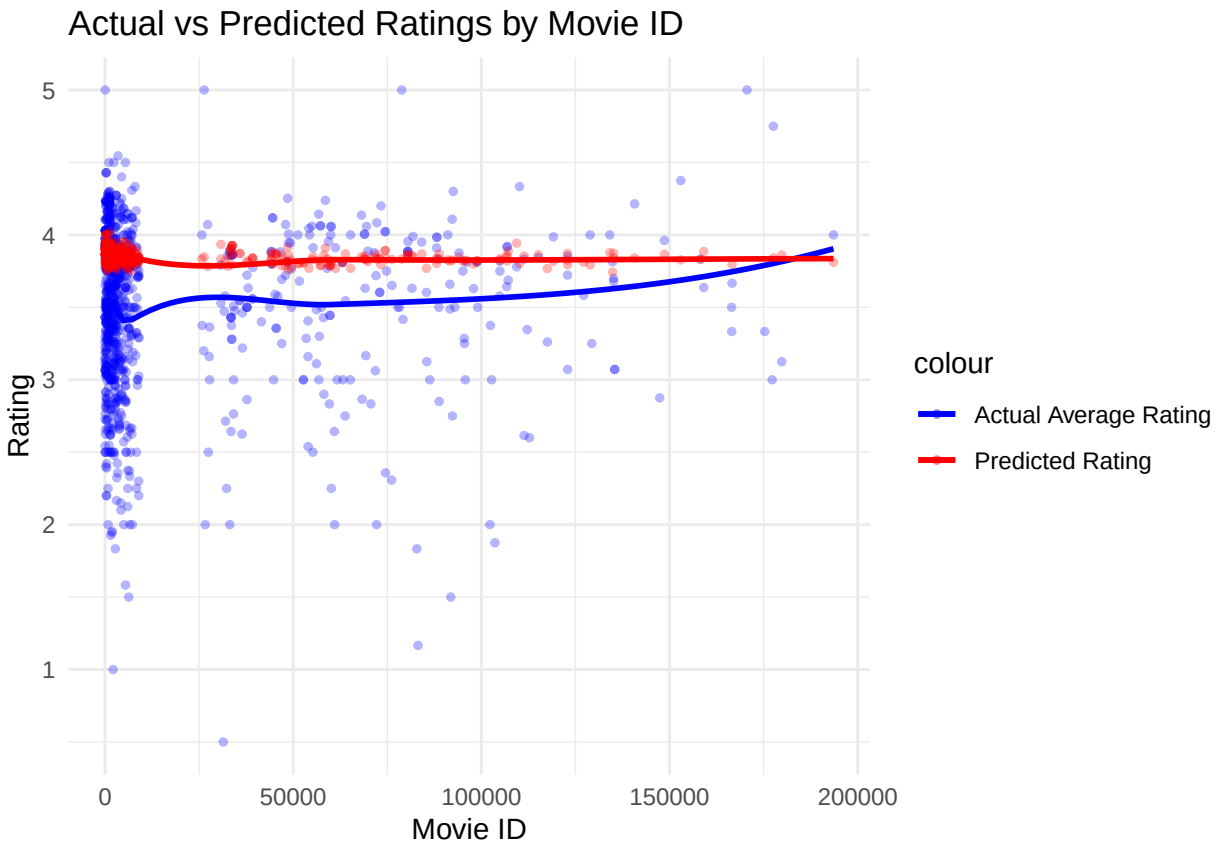



## Model Fitting - Genre Count + Good Actor Occurrences

In our third attempt, we refined the actor score from our second attempt by considering only the number of appearances in movies with a rating higher than 3.5. We defined these as good movies. For each actor, we calculated their actor score based on their appearances in good movies. For each movie, we summed the actor scores of all its actors to obtain the movie's `good_actor_score`. We used `genre_count` and

`good_actor_score` as our independent variables (x) and `avg_rating` as our dependent variable (y). A linear model was then fitted to explore the relationship between these variables and the average rating of the movies.

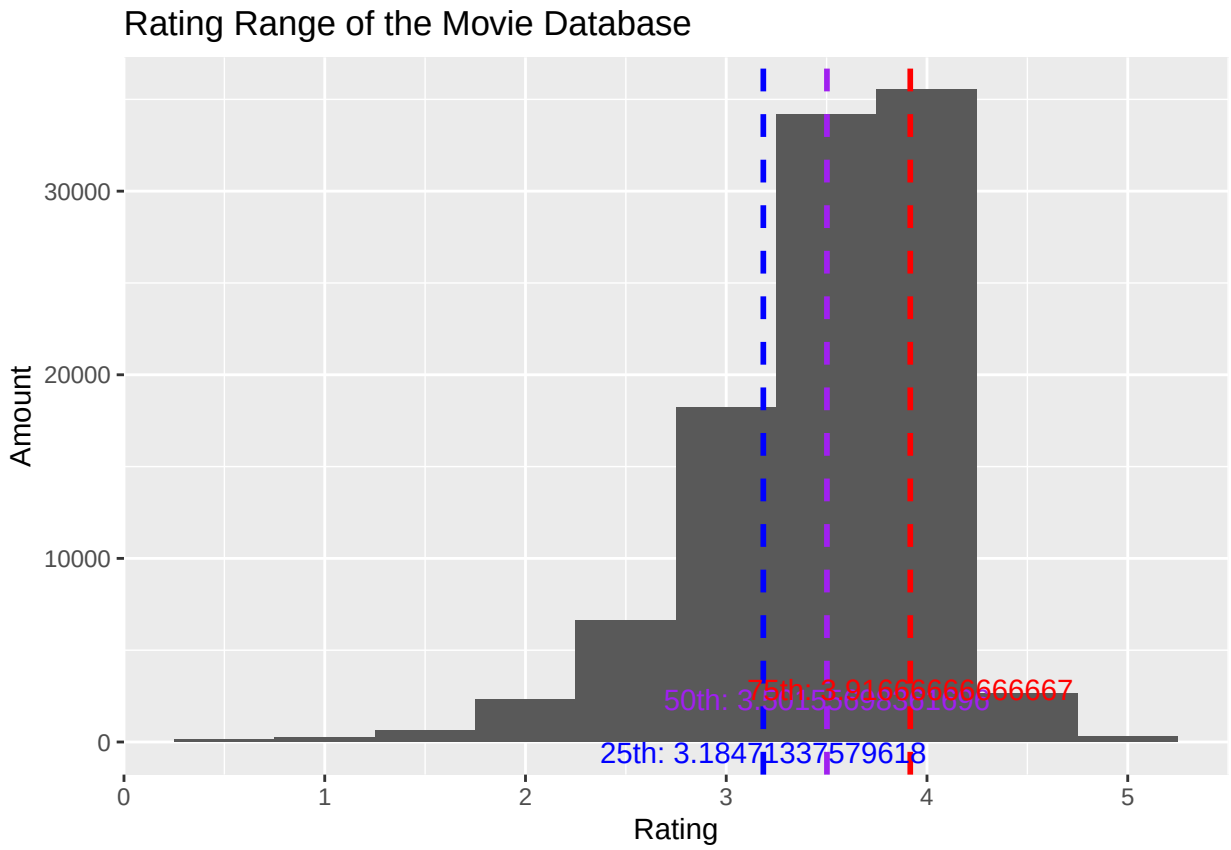
```
# Filter the merged data to include only movies with a rating > 3.5
movies_cast <- merge(cast, merged_data, by = "imdbId")
good_movies <- movies_cast[movies_cast$avg_rating > 3.5, ]

# Count actor occurrences only in good movies
get_good_actor_frequency <- function(df) {
  actor_vector <- unlist(df[, grepl("Actor", names(df))])
  actor_frequency <- table(actor_vector)
  return(actor_frequency)
}

# Apply the function to the filtered good movies data
good_actor_frequency <- get_good_actor_frequency(good_movies)

# Calculate the actor score for each movie based on good occurrences
movies_cast$good_actor_score <- apply(movies_cast[, grepl("Actor", names(movies_cast))], 1, function(x)

# Update the model to use the new good_actor_score variable
model.train <- lm(avg_rating ~ genre_count + good_actor_score, data = movies_cast)
predictions <- predict(model.train, newdata = movies_cast)
movies_cast$predicted_good_rating <- predictions

# Sample the data
set.seed(123)
sampled_cast_data <- movies_cast[sample(nrow(movies_cast), min(1000, nrow(movies_cast))), ]

# Plot the ground truth vs predicted values
ggplot(sampled_cast_data, aes(x = movieId)) +
  geom_point(aes(y = avg_rating, color = "Actual Average Rating"), size = 1, alpha = 0.3) +
  geom_point(aes(y = predicted_good_rating, color = "Predicted Rating"), size = 1, alpha = 0.3) +
  geom_smooth(aes(y = avg_rating, color = "Actual Average Rating"), method = "loess", se = FALSE) +
  geom_smooth(aes(y = predicted_good_rating, color = "Predicted Rating"), method = "loess", se = FALSE)
  labs(x = "Movie ID", y = "Rating",
       title = "Actual vs Predicted Ratings by Movie ID") +
  scale_color_manual(values = c("Actual Average Rating" = "blue", "Predicted Rating" = "red")) +
  theme_minimal()
```




## Graph Plotting

To explore our research questions, we have created several plots below.

```
# Analyzing the rating range and distributions of the movies
q25 <- quantile(merged_data$avg_rating, probs = 0.25)
q50 <- mean(merged_data$avg_rating)
q75 <- quantile(merged_data$avg_rating, probs = 0.75)

ggplot(merged_data, aes(x = avg_rating)) +
  geom_histogram(binwidth=0.5) +
  geom_vline(xintercept = q25, linetype = "dashed", color = "blue", size=1) +
  geom_vline(xintercept = q75, linetype = "dashed", color = "red", size=1) +
  geom_vline(xintercept = q50, linetype = "dashed", color = 'purple', size=1) +
  labs(title = "Rating Range of the Movie Database", x = "Rating", y = "Amount") +
  annotate("text", x = q25, y = 0, label = paste("25th:", q25), vjust = 1, color = "blue") +
  annotate("text", x = q50, y = 0, label = paste("50th:", q50), vjust = -1.5, color = "purple") +
  annotate("text", x = q75, y = 0, label = paste("75th:", q75), vjust = -2, color = "red")
```




**Discussion**

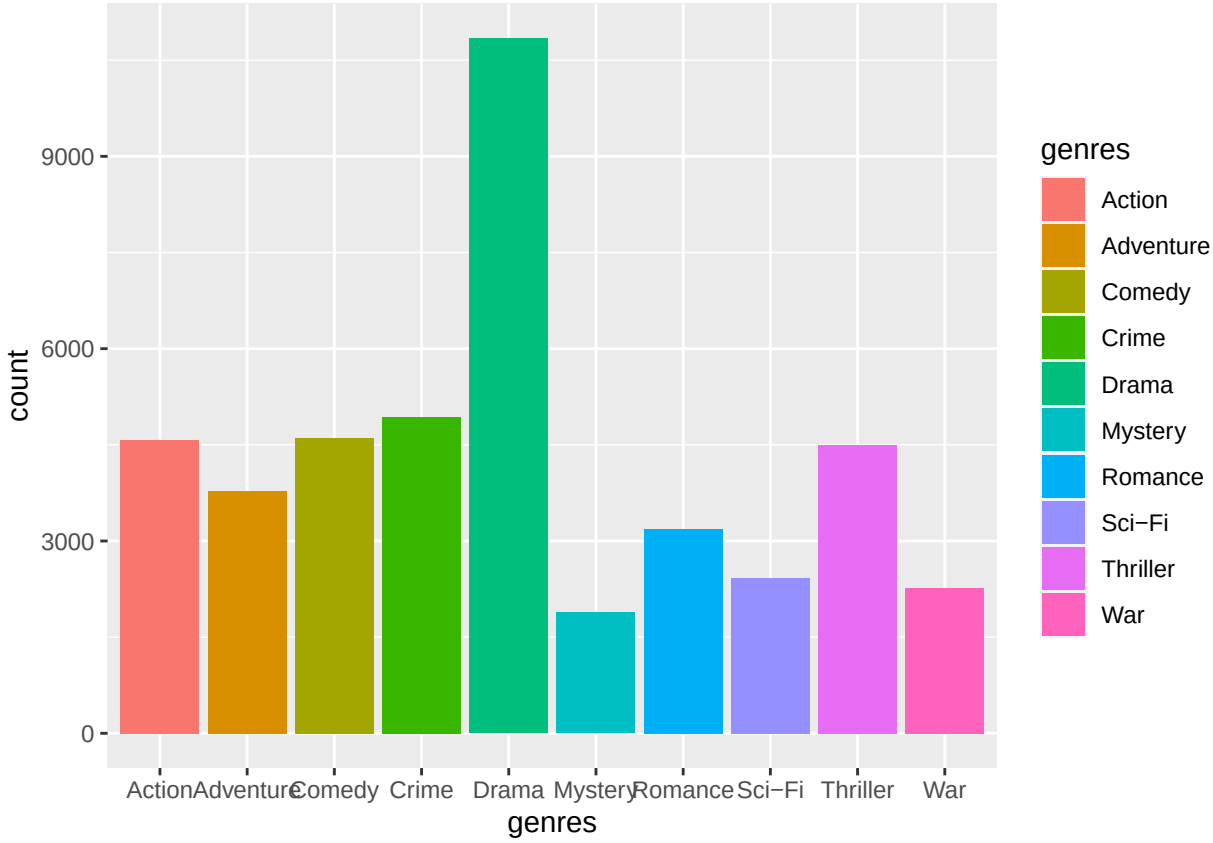
The plot provides some clear insights:

1. **Central Tendency:** The noticeable peak in the bar height around the median rating of approximately 3.5 suggests that a significant number of movies receive average scores. This concentration around the median indicates that while some films receive exceptionally high or low ratings, most are rated moderately. This pattern may reflect either a consistent quality in the movie selection or a general tendency of the audience to rate movies as average.
2. **Diversity in Ratings:** The considerable gap between the 25th percentile (blue dashed line) and the 75th percentile (red dashed line) points to a broad distribution of ratings. This is further evidenced by the bar chart, which shows most ratings falling between 3 and 4. Consequently, our analysis can be guided in this direction to ascertain whether a movie is likely to be mediocre (with a rating of around 3) or phenomenal (with a rating of more than 3).

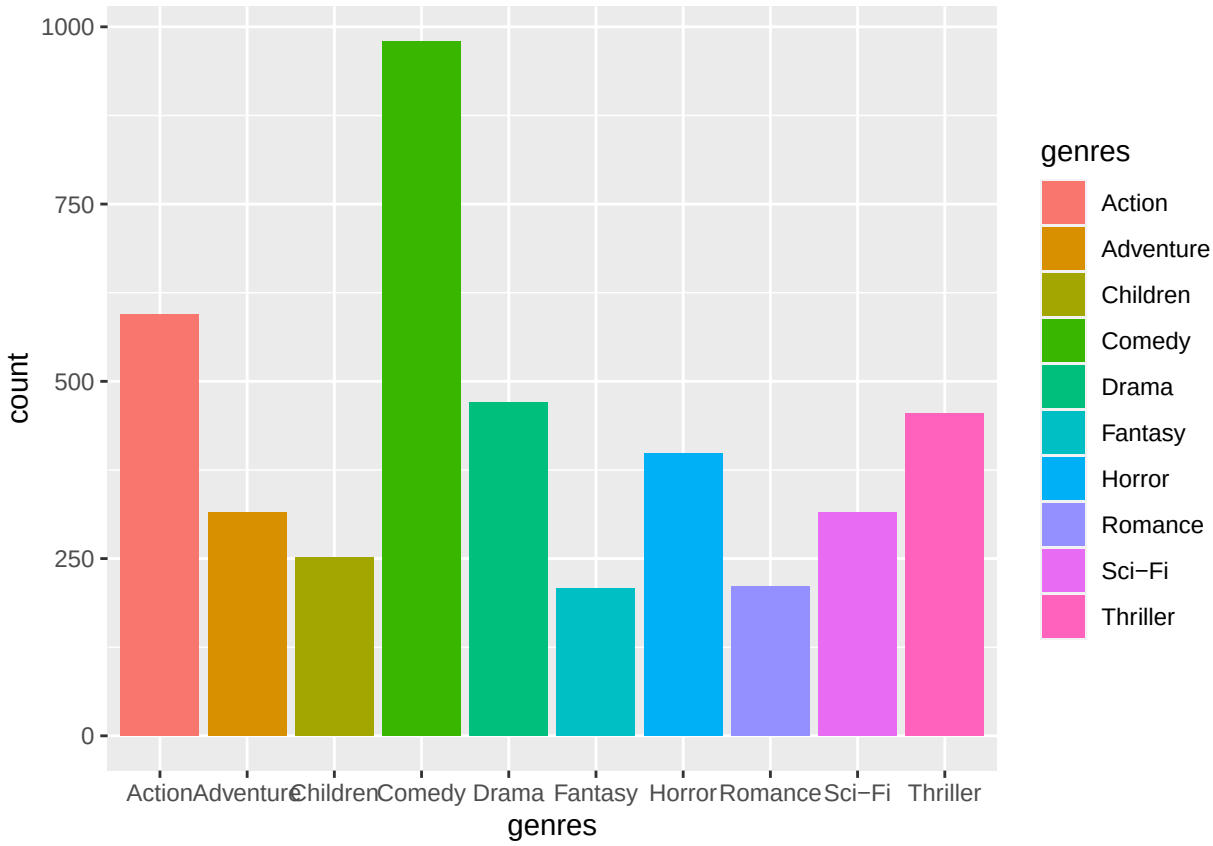
These observations lead our analysis to further investigate which features may influence the final ratings.

```
genres_ratings_10 <- merged_data %>%
  group_by(genres) %>%
  summarise(average_rating = mean(avg_rating, na.rm = TRUE)) %>%
  arrange(desc(average_rating)) %>%
  slice_head(n=10)

movies_over_4_rating <- merged_data %>%
  filter(avg_rating >= 4)
movies_under_2_rating <- merged_data %>%
  filter(avg_rating <=2)
```

```r
genres_separated <- movies_over_4_rating %>%
  separate_rows(genres, sep = "\\|")
genres_separated_low <- movies_under_2_rating %>%
  separate_rows(genres, sep="\\|")

genre_counts <- genres_separated %>%
  count(genres, name = "count") %>%
  arrange(desc(count)) %>%
  slice_head(n=10)
genre_counts_low <- genres_separated_low %>%
  count(genres, name="count") %>%
  arrange(desc(count)) %>%
  slice_head(n=10)

ggplot(genre_counts, aes(x = genres, y = count, fill=genres )) +
     geom_bar(stat="identity")
```




```r
ggplot(genre_counts_low, aes(x = genres, y = count,fill=genres )) +
     geom_bar(stat="identity")
```



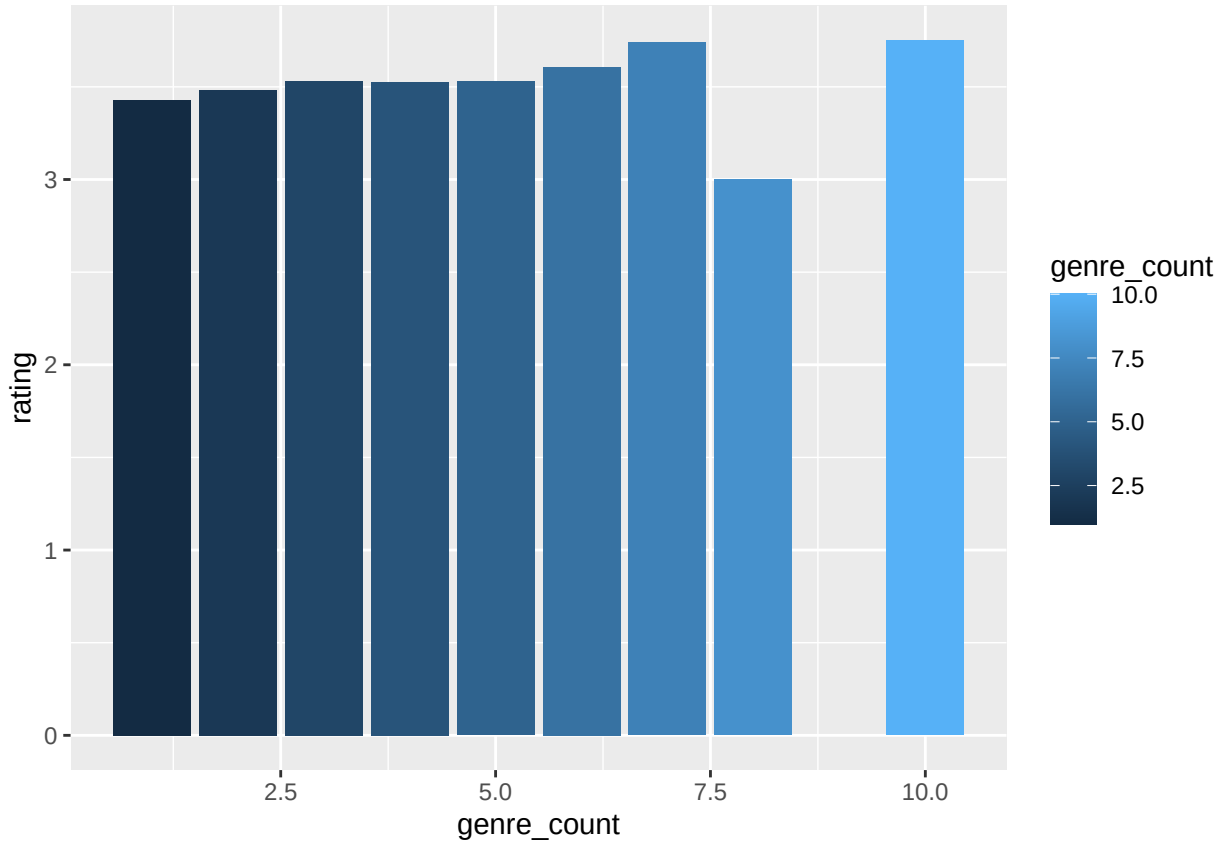

**Discussion**

The bar chart illustrates the distribution of movies with ratings above 4 across different genres. Drama stands out prominently, indicating a strong preference for this genre among highly-rated movies. The Documentary and Comedy genres also have significant representation, suggesting their appeal in high-quality films. In contrast, genres like Action, Animation, Crime, Horror, Romance, and Thriller have fewer movies with high ratings, with Horror being the least represented. This suggests that audiences and critics may favor the storytelling and content often associated with Drama and Documentaries over the characteristics typically found in the other mentioned genres for highly-rated films.

The bar graph showing genre counts for movies with ratings of 2 or less reveals a different trend. Drama, which was dominant in higher ratings, is less prevalent here, suggesting it is generally well-received. Comedy, despite being popular in higher ratings, is the most common genre among lower-rated movies, indicating a divided reception. Action and Thriller genres also appear more frequently in this lower rating spectrum, possibly reflecting critical disdain for certain films within these genres. Sci-Fi and Romance have moderate counts, while Children's movies are the least represented, suggesting these genres may have moderate reception. Comparing this to the summary for movies rated 4 and above, there is a noticeable contrast: genres that excel in higher ratings, such as Drama and Documentary, are less common among lower-rated movies. This could imply that well-executed dramas and documentaries resonate well with audiences, whereas comedies can be hit-or-miss. Action and Thriller genres, while not as common among top-rated movies, appear more frequently at the lower end of the ratings spectrum, which may indicate a consistent lower critical reception or a split in audience preference.

```
genrecount_rating <- merged_data %>%
  group_by(genre_count) %>%
  summarise(rating = mean(avg_rating))

ggplot(genrecount_rating, aes(x = genre_count, y=rating, fill=genre_count )) +
```

```
    geom_bar(stat="identity")
```




**Discussion**

The bar chart seems to illustrate the relationship between the number of genres a movie spans and its average rating. The trend indicates that movies covering a broader range of genres do not necessarily garner higher average ratings. Notably, films with a single genre focus tend to receive moderate ratings. There is a distinct peak for movies that encompass around ten genres, which achieve the highest average ratings. This suggests a potential sweet spot where a wide diversity of genres correlates with greater appreciation from viewers or critics. However, the ratings slightly decline for movies with fewer genres, around seven to nine, before peaking. This pattern could suggest that beyond a certain threshold, adding more genres does not enhance a movie's reception and may even detract from its overall quality or coherence.

## Future Work

### Dataset

For future analyses, we might consider incorporating additional datasets or conducting further investigations by Python crawler to address our research questions more comprehensively:

1. **Cast Data:** We have already utilized a Python crawler to obtain cast data and will continue to employ this technique to gather more comprehensive information about the actors and actresses involved in the films. This data is crucial for exploring the impact of star power on movie ratings and success.
2. **Temporal Trends:** Analyzing changes in genre popularity and movie ratings over time could provide insights into evolving audience preferences and industry trends.
3. **Revenue and Budget Data:** Integrating data on movie revenues and budgets could help us explore the financial success of different genres and the impact of star power on a movie's commercial performance.

4. **Social Media and Marketing Data:** Investigating the role of social media buzz and marketing campaigns in influencing movie popularity and ratings could offer a more holistic view of what drives audience interest and engagement.
5. **Audience Demographics:** Examining how different demographic groups (e.g., age, gender, region) respond to various genres and cast members could reveal targeted strategies for maximizing a movie's appeal.

By expanding our dataset and employing a variety of analytical approaches, including continued Python crawling for additional cast data and other relevant information, we can deepen our understanding of the factors that influence movie ratings, genre popularity, and overall success in the film industry.

### Model

Currently, our primary focus has been on feature extraction and analysis. We have employed a linear model to examine the relationships between various variables. Moving forward, we intend to explore a range of modeling approaches to gain a deeper understanding of the factors influencing movie ratings and success. By constructing and comparing different models, we aim to identify more nuanced relationships and potentially uncover new insights into the dynamics of the film industry.

## Conclusion

In this proposal, we have undertaken an extensive analysis of movie ratings and their influencing factors, using datasets from MovieLens and IMDb. We have explored the impact of genre diversity, cast popularity, and other factors on movie ratings. Our findings suggest that while a broader range of genres and star power can positively influence ratings, the relationship is complex and warrants further investigation.

Moving forward, we aim to expand our dataset and employ more sophisticated modeling techniques to uncover deeper insights. By continuously refining our approach and incorporating additional data sources, we hope to provide a more comprehensive understanding of the factors that contribute to a movie's success and appeal. Our goal is to not only enhance our analytical skills but also to contribute meaningful knowledge to the field of film analytics.