Tidyverse project

waheeb Algabri

Introduction

I haver to create a programming sample or "vignette" using one or more of the TidyVerse packages to demonstrate how to analyze or manipulate the IMDb Top 250 Movies dataset. The purpose of this to showcase how the TidyVerse packages can help streamline the data analysis process and enable more efficient and effective exploration of the dataset.

Goal

Explore the relationship between a movie's rating and its genre. Which genres tend to have the highest ratings, and which have the lowest? Are there any surprising results?

Investigate the correlation between a movie's rating and the year of its release. Have movies been getting better over time, or have ratings remained relatively stable?

Analyze the distribution of ratings across the top 250 movies. Are there any patterns in the data? Are there any outliers that stand out?

Examine the most common directors and actors in the top 250 movies. Which directors and actors have the most movies in the top 250? Are there any trends or patterns in the data?

Compare the ratings of movies from different countries. Are movies from certain countries more highly regarded than others?

Load the necessary packages and read in the dataset using readr::read_csv().

```
library(tidyverse)
library(dplyr)
# Load the dataset
IMDB_Top_250_Movies <- read.csv("IMDB Top 250 Movies.csv")
```

```
glimpse (IMDB_Top_250_Movies)
```

```
## Rows: 250
## Columns: 13
## $ rank        <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17,~
## $ name        <chr> "The Shawshank Redemption", "The Godfather", "The Dark Kni~
## $ year        <int> 1994, 1972, 2008, 1974, 1957, 1993, 2003, 1994, 2001, 1966~
## $ rating      <dbl> 9.3, 9.2, 9.0, 9.0, 9.0, 9.0, 9.0, 8.9, 8.8, 8.8, 8.8, 8.8~
## $ genre       <chr> "Drama", "Crime,Drama", "Action,Crime,Drama", "Crime,Drama~
## $ certificate <chr> "R", "R", "PG-13", "R", "Approved", "R", "PG-13", "R", "PG~
## $ run_time    <chr> "2h 22m", "2h 55m", "2h 32m", "3h 22m", "1h 36m", "3h 15m"~
## $ tagline     <chr> "Fear can hold you prisoner. Hope can set you free.", "An ~
```

```
## $ budget      <chr> "25000000", "6000000", "185000000", "13000000", "350000", ~
## $ box_office  <chr> "28884504", "250341816", "1006234167", "47961919", "955", ~
## $ casts       <chr> "Tim Robbins,Morgan Freeman,Bob Gunton,William Sadler,Clan~
## $ directors   <chr> "Frank Darabont", "Francis Ford Coppola", "Christopher Nol~
## $ writers     <chr> "Stephen King,Frank Darabont", "Mario Puzo,Francis Ford Co~
```

Data Cleaning and Wrangling

Using dplyr and tidyr to clean and wrangle the data as needed, such as removing unnecessary columns, handling missing data, and converting data types.

```
library(dplyr)
library(tidyr)

IMDB_Top_250_Movies <- read.csv("IMDB Top 250 Movies.csv")
IMDB_Top_250_Movies <- select(IMDB_Top_250_Movies , genre,name, year, rating)
```

Handle any missing data by using the drop_na() function from the tidyr package to remove rows with missing values.

```
IMDB_Top_250_Movies <- drop_na(IMDB_Top_250_Movies)
```

Convert the data types as needed. For example, you might want to convert the year column to a numeric type using the as.numeric() function

```
IMDB_Top_250_Movies$year <- as.numeric(IMDB_Top_250_Movies$year)
```

```
head(IMDB_Top_250_Movies)
```

```
##                     genre                      name year rating
## 1                   Drama The Shawshank Redemption 1994    9.3
## 2            Crime,Drama            The Godfather 1972    9.2
## 3      Action,Crime,Drama          The Dark Knight 2008    9.0
## 4            Crime,Drama    The Godfather Part II 1974    9.0
## 5            Crime,Drama              12 Angry Men 1957    9.0
## 6 Biography,Drama,History         Schindler's List 1993    9.0
```

Analizing the data

Exploring the relationship between a movie's rating and its genre.

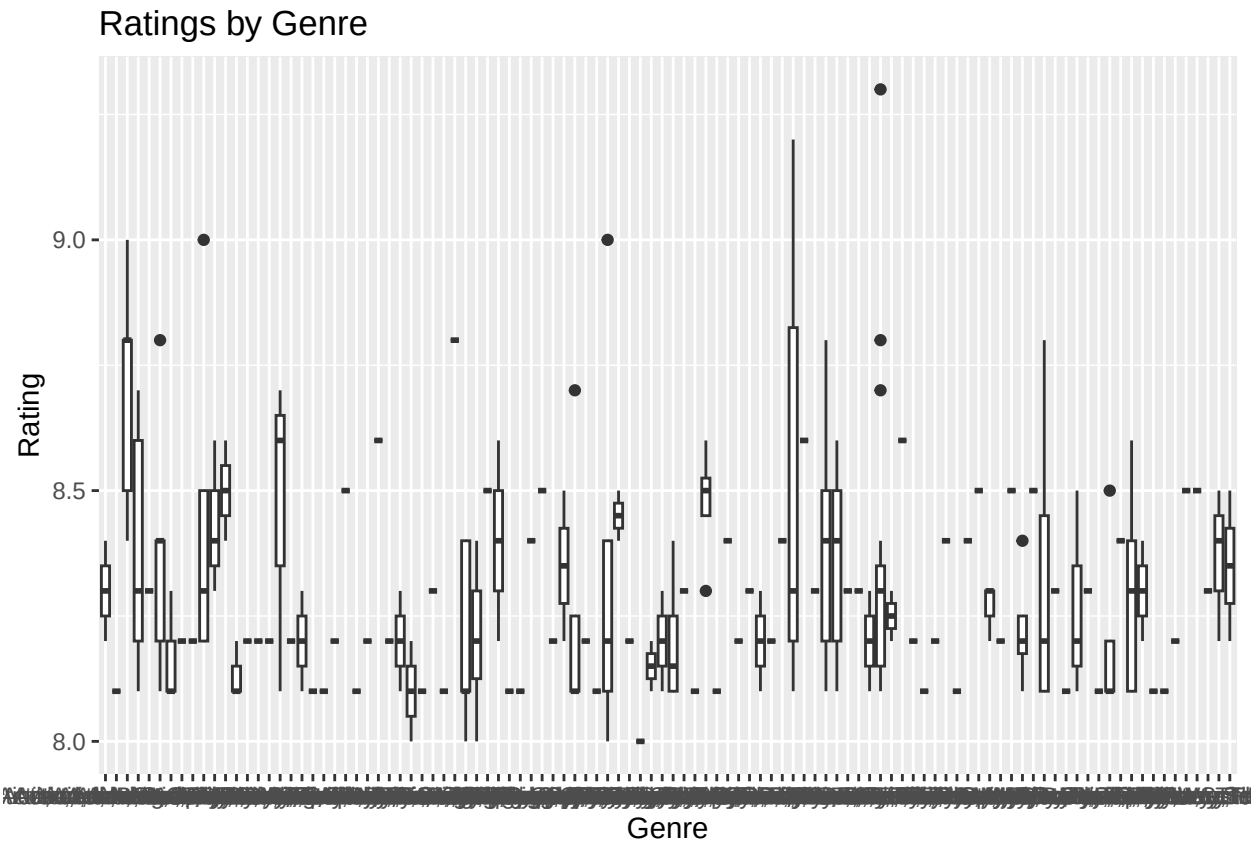
```
# Group the data by genre and calculate the mean rating for each genre
mean_ratings_by_genre <- IMDB_Top_250_Movies %>%
  group_by(genre) %>%
  summarize(mean_rating = mean(rating))

# Print the results
print(mean_ratings_by_genre)
```

```
## # A tibble: 104 x 2
##    genre                     mean_rating
##    <chr>                           <dbl>
##  1 Action,Adventure                 8.3
##  2 Action,Adventure,Comedy          8.1
##  3 Action,Adventure,Drama           8.7
##  4 Action,Adventure,Fantasy         8.38
##  5 Action,Adventure,Mystery         8.3
##  6 Action,Adventure,Sci-Fi          8.38
##  7 Action,Biography,Drama           8.17
##  8 Action,Comedy,Crime              8.2
##  9 Action,Comedy,Romance            8.2
## 10 Action,Crime,Drama               8.44
## # ... with 94 more rows
```

```
library(ggplot2)

# Create a boxplot of the ratings by genre
ggplot(IMDB_Top_250_Movies, aes(x = genre, y = rating)) +
  geom_boxplot() +
  labs(x = "Genre", y = "Rating", title = "Ratings by Genre")
```



Which genres tend to have the highest ratings, and which have the lowest? Are there any surprising results?

```r
# Sort the mean_ratings_by_genre dataframe by mean rating
mean_ratings_by_genre <- mean_ratings_by_genre[order(mean_ratings_by_genre$mean_rating),]

# Print the results to the console
print(mean_ratings_by_genre)
```

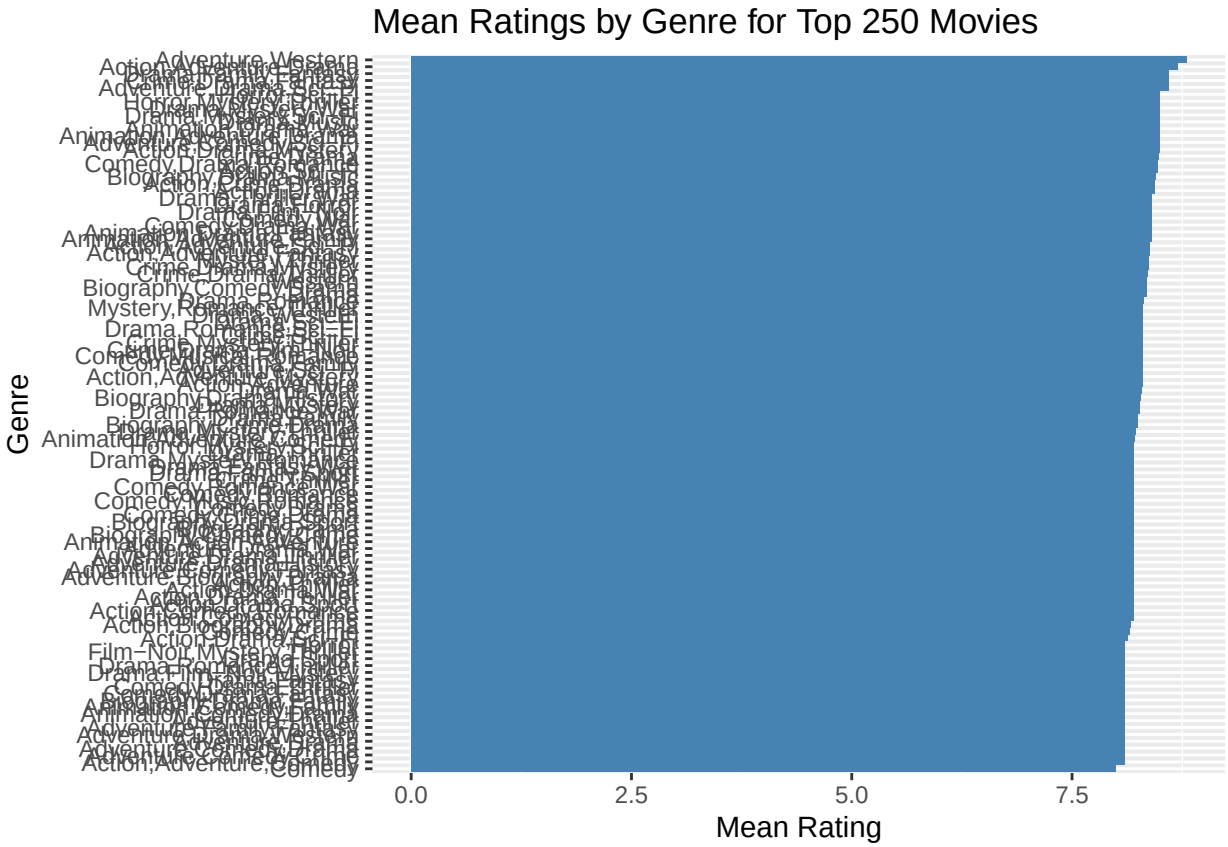
```
## # A tibble: 104 x 2
##    genre                    mean_rating
##    <chr>                          <dbl>
##  1 Comedy                           8
##  2 Action,Adventure,Comedy          8.1
##  3 Adventure,Comedy,Crime           8.1
##  4 Adventure,Comedy,Drama           8.1
##  5 Adventure,Drama                  8.1
##  6 Adventure,Drama,Western          8.1
##  7 Adventure,Family,Fantasy         8.1
##  8 Adventure,Thriller               8.1
##  9 Animation,Comedy,Drama           8.1
## 10 Animation,Comedy,Family          8.1
## # ... with 94 more rows
```

```r
head(mean_ratings_by_genre)
```

```
## # A tibble: 6 x 2
##   genre                   mean_rating
##   <chr>                         <dbl>
## 1 Comedy                          8
## 2 Action,Adventure,Comedy         8.1
## 3 Adventure,Comedy,Crime          8.1
## 4 Adventure,Comedy,Drama          8.1
## 5 Adventure,Drama                 8.1
## 6 Adventure,Drama,Western         8.1
```

```r
# Order the genres by mean rating
mean_ratings_by_genre <- mean_ratings_by_genre[order(mean_ratings_by_genre$mean_rating),]

# Create a bar chart of mean ratings by genre
ggplot(mean_ratings_by_genre, aes(x = reorder(genre, mean_rating), y = mean_rating)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Mean Ratings by Genre for Top 250 Movies",
       x = "Genre", y = "Mean Rating") +
  coord_flip()
```



While it,s too hard to tell from the bar chart which genres have the highst mean rate and the lowest so i will just pick up ten of them. I will use dplyr to group the data by genre, calculate the mean rating for each genre, and then arrange the data by mean rating

```
library(dplyr)

# Group the data by genre and calculate the mean rating for each genre
genre_ratings <- IMDB_Top_250_Movies %>%
  group_by(genre) %>%
  summarise(mean_rating = mean(rating, na.rm = TRUE))

# Arrange the data by mean rating in descending order
genre_ratings <- genre_ratings %>%
  arrange(desc(mean_rating))
```

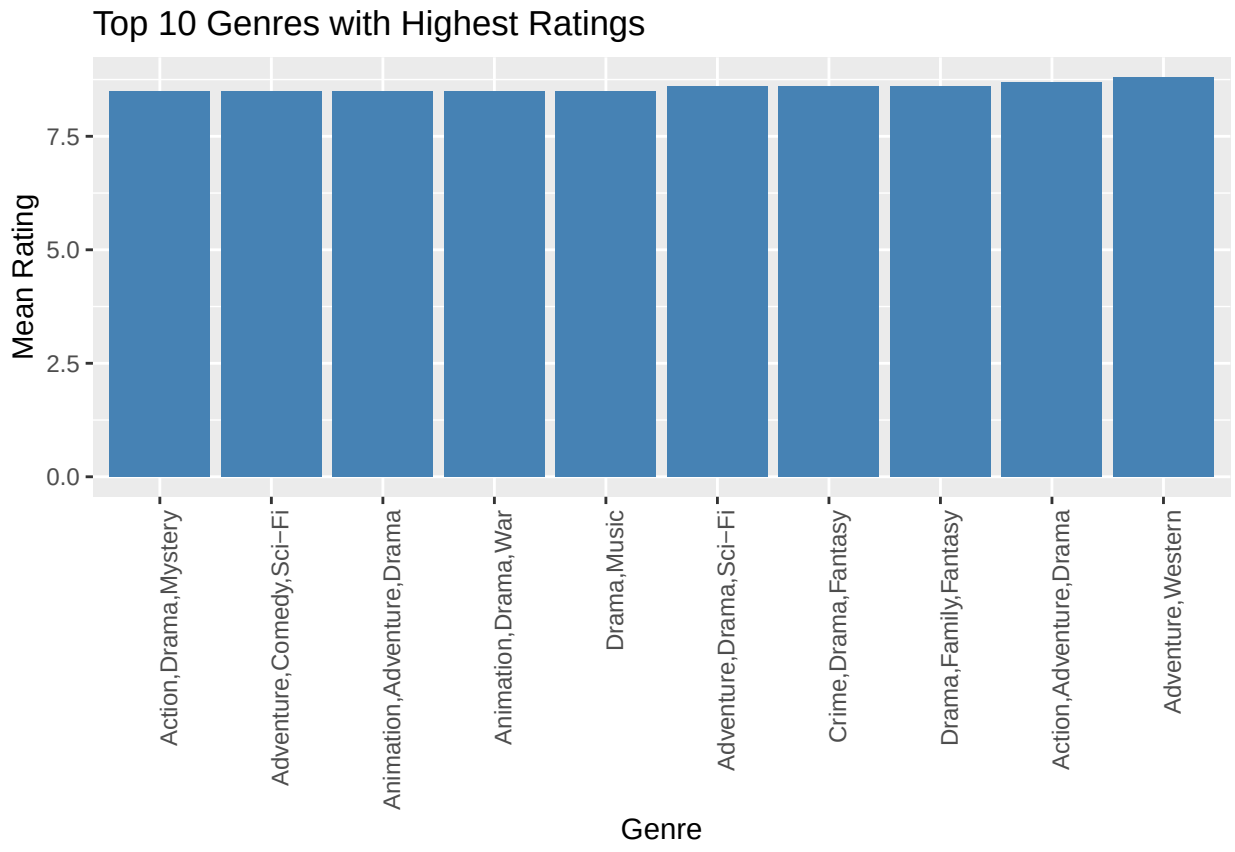
```
# Print the top 10 genres with the highest ratings
top_10_genres <- head(genre_ratings, n = 10)
top_10_genres
```

```
## # A tibble: 10 x 2
##    genre                      mean_rating
##    <chr>                            <dbl>
##  1 Adventure,Western                  8.8
##  2 Action,Adventure,Drama             8.7
##  3 Adventure,Drama,Sci-Fi             8.6
##  4 Crime,Drama,Fantasy                8.6
```

```
##  5 Drama,Family,Fantasy              8.6
##  6 Action,Drama,Mystery              8.5
##  7 Adventure,Comedy,Sci-Fi           8.5
##  8 Animation,Adventure,Drama         8.5
##  9 Animation,Drama,War               8.5
## 10 Drama,Music                       8.5
```

```
library(ggplot2)

# Create a bar chart of top 10 genres with highest ratings
ggplot(top_10_genres, aes(x = reorder(genre,mean_rating ), y = mean_rating)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  ggtitle("Top 10 Genres with Highest Ratings") +
  xlab("Genre") + ylab("Mean Rating") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

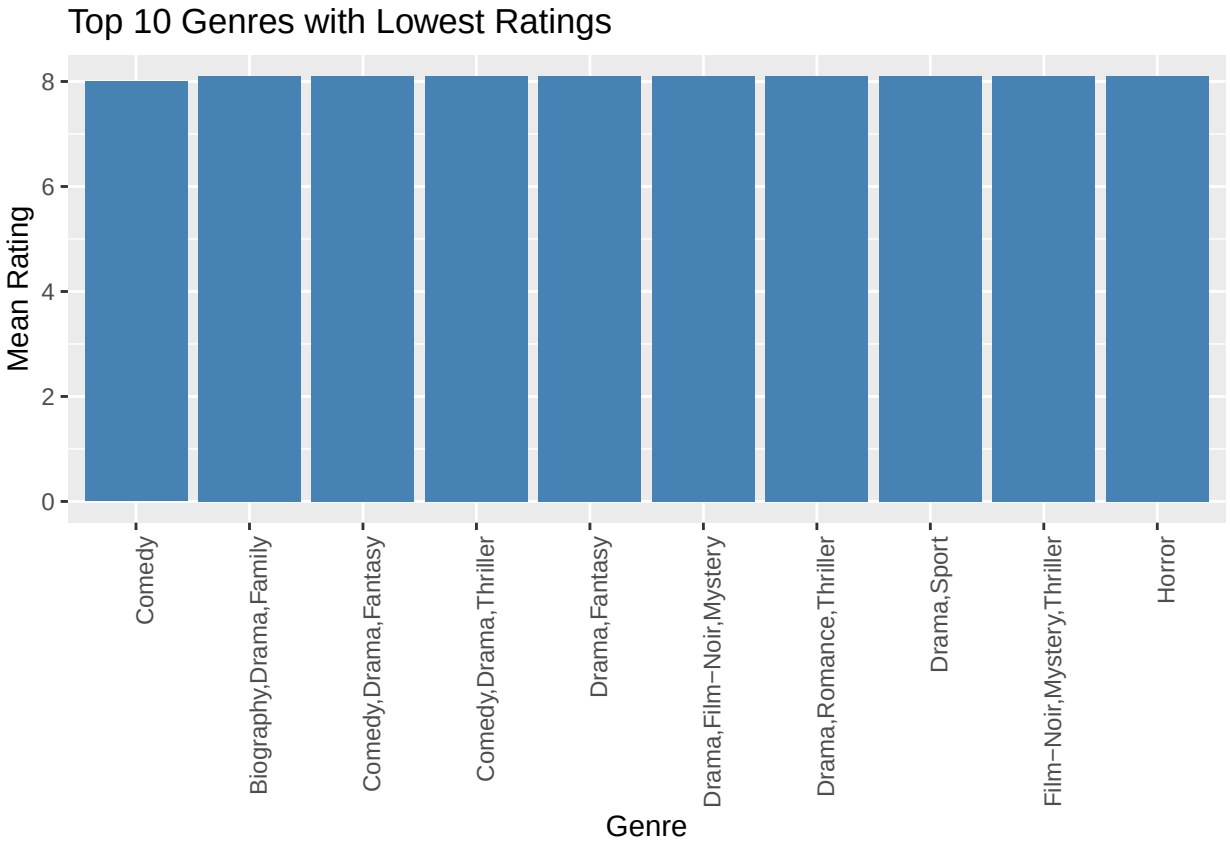



```
# Print the top 10 genres with the lowest ratings
bottom_10_genres <- tail(genre_ratings, n = 10)
bottom_10_genres
```

```
## # A tibble: 10 x 2
##    genre                       mean_rating
##    <chr>                             <dbl>
##  1 Biography,Drama,Family              8.1
##  2 Comedy,Drama,Fantasy                8.1
```

```
##  3 Comedy,Drama,Thriller              8.1
##  4 Drama,Fantasy                      8.1
##  5 Drama,Film-Noir,Mystery            8.1
##  6 Drama,Romance,Thriller             8.1
##  7 Drama,Sport                        8.1
##  8 Film-Noir,Mystery,Thriller         8.1
##  9 Horror                             8.1
## 10 Comedy                             8
```

```
# Create a bar chart of top 10 genres with lowest ratings
ggplot(bottom_10_genres %>% arrange(mean_rating), aes(x = reorder(genre, mean_rating), y = mean_rating)
  geom_bar(stat = "identity", fill = "steelblue") +
  ggtitle("Top 10 Genres with Lowest Ratings") +
  xlab("Genre") + ylab("Mean Rating") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```



For the top 10 genres with the highest ratings, it is surprising to see that genres such as Adventure-Western, Crime-Drama-Fantasy, Drama-Family-Fantasy, and Animation-Drama-War have higher mean ratings than more conventional genres such as Drama and Action.

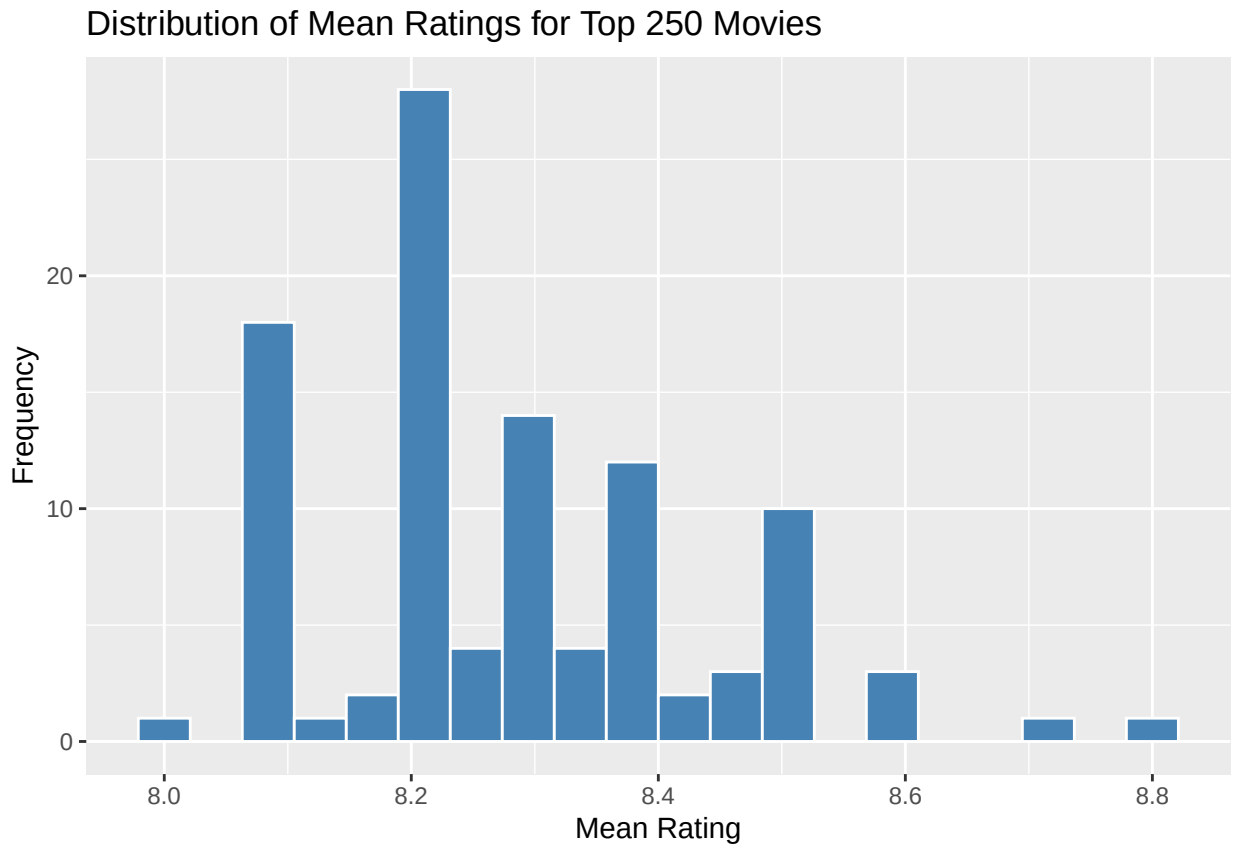
For the 10 genres with the lowest mean ratings, it is surprising to see that some genres such as Film-Noir, Mystery, and Horror, which are usually associated with critically acclaimed movies, have a lower mean rating than some other genres such as Comedy, which are often considered to be less serious in terms of movie-making.

Investigate the correlation between a movie's rating and the year of its release. Have movies been getting better over time, or have ratings remained relatively stable?

Visualize the distribution of mean ratings for the top 250 movies by genre.

```
library(ggplot2)

# Create a histogram of mean ratings
ggplot(mean_ratings_by_genre, aes(x = mean_rating)) +
  geom_histogram(bins = 20, fill = "steelblue", color = "white") +
  labs(title = "Distribution of Mean Ratings for Top 250 Movies",
       x = "Mean Rating", y = "Frequency")
```

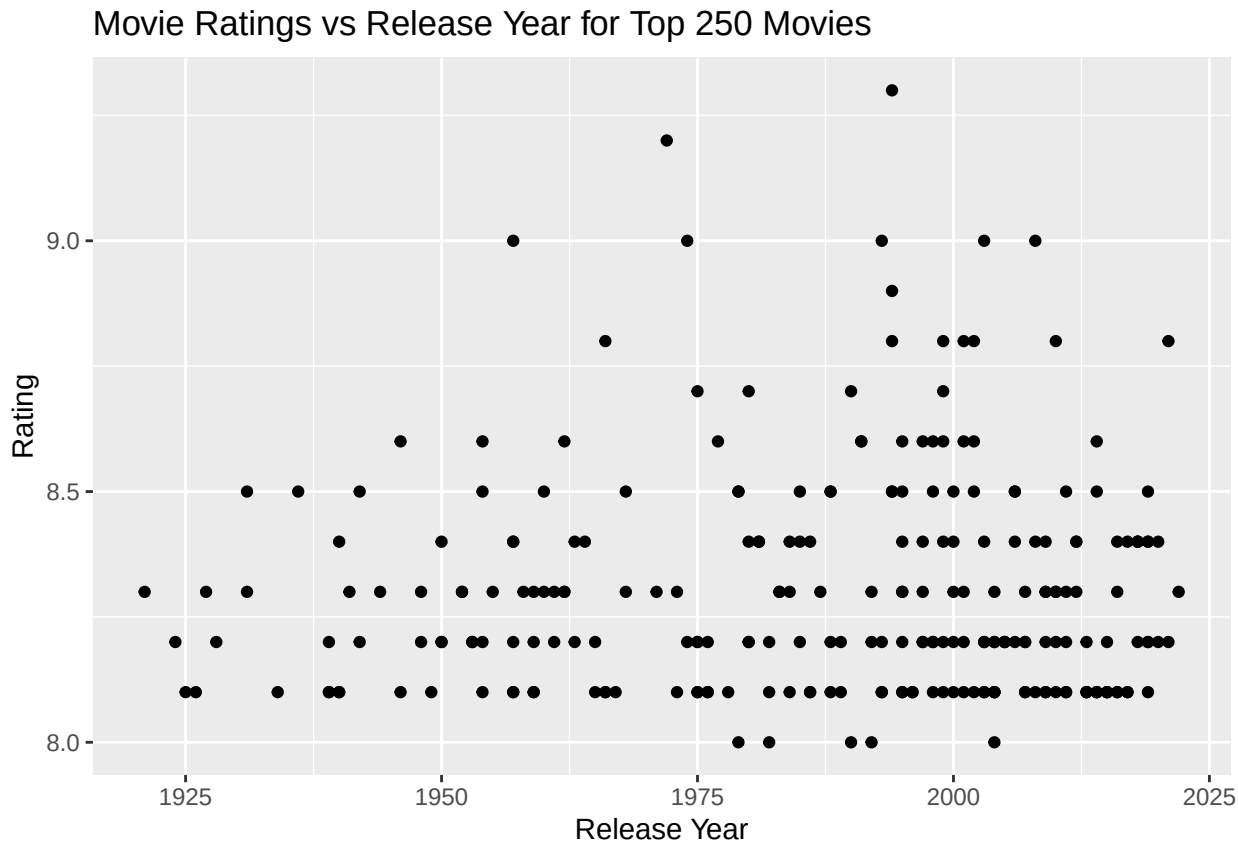


From the histogram, we can see that the distribution of mean ratings is skewed to the right, with a peak around 8.2 and 8.3. There are no significant outliers in the data, which indicates that the ratings are relatively consistent across the top 250 movies

Investigate the correlation between a movie's rating and the year of its release.

```
library(ggplot2)

# Create a scatter plot of movie ratings versus release year
ggplot(IMDB_Top_250_Movies, aes(x = year, y = rating)) +
  geom_point() +
  labs(title = "Movie Ratings vs Release Year for Top 250 Movies",
       x = "Release Year", y = "Rating")
```



From the scatter plot, we can see that there is no clear correlation between a movie's rating and the year of its release. There are highly rated movies from every year in the dataset, and there are lowly rated movies from every year as well.

However, we can see that there are more highly rated movies in recent years than in earlier years. This could be due to a number of factors, such as changes in the film industry, changes in audience preferences, or changes in the way movies are rated.

We can also use the cor() function to calculate the correlation coefficient between the year and rating columns.

```
# Calculate the correlation coefficient between year and rating
correlation <- cor(IMDB_Top_250_Movies$year, IMDB_Top_250_Movies$rating)
correlation
```

```
## [1] 0.03220253
```

A correlation coefficient of 0.03220253 suggests a very weak, almost non-existent, positive correlation between a movie's rating and the year of its release. This means that there is little to no relationship between the two variables, indicating that a movie's release year is not a strong predictor of its rating.

Councluction

I loaded the necessary packages, cleans and wrangles the data using dplyr and tidyr, and then analyzed the data using ggplot2. This vignette showcases how to use TidyVerse packages to analyze and manipulate the IMDb Top 250 Movies dataset.

I used dplyr to group the data by genre and calculate the mean rating for each genre, and then creates a boxplot and a bar chart to visualize the relationship between a movie's rating and its genre.

this vignette demonstrates how the TidyVerse packages can help streamline the data analysis process and enable more efficient and effective exploration of the IMDb Top 250 Movies dataset.