Lab 2: Movie Data

Kaytlyn Daffern

2024-10-13

Question 1:

What is the range of years of production of the movies of this data set (i.e. what is the year of production of the oldest movie and of the most recent movie in this data set)?

```
# Find the range of years of movie production
year_range <- range(movies$year, na.rm = TRUE)
year_range
```

```
## [1] 1893 2005
```

From the movie dataset, the earliest movie was made in 1893, and the most recent was made in 2005.

Question 2:

What proportion of movies have their budget included in this data base, and what proportion doesn't? What are top 5 most expensive movies in this data set?

```
has_budget <- !is.na(movies$budget)
# Find the number of movies with/out budget
budget_count <- table(has_budget)
budget_count
```

```
## has_budget
## FALSE  TRUE
## 53573  5215
```

```
# Find the proportion of movies that did/n't have a budget
budget_proportion <- table(has_budget) / nrow(movies)
budget_proportion
```

```
## has_budget
##      FALSE       TRUE
## 0.91129142 0.08870858
```

```r
# Format and display the top 5 most expensive movies
top5_expensive <- movies %>%
  filter(!is.na(budget)) %>% # remove movies without budgets
  arrange(desc(budget)) %>%
  select(title, budget) %>%
  head(5) %>% # top 5
  mutate(budget = format(budget, big.mark = ",", scientific = FALSE))  # Format without scientific nota

# Display the result as a table using knitr to make it look nice
knitr::kable(top5_expensive, col.names = c("Movie Title", "Budget"))
```

Movie Title	Budget
Spider-Man 2	200,000,000
Titanic	200,000,000
Troy	185,000,000
Terminator 3: Rise of the Machines	175,000,000
Waterworld	175,000,000

Question 3:

What are top 5 longest movies?

```r
top5_longest <- movies %>%
  arrange(desc(length)) %>%
  select(title, length) %>%
  head(5) %>%
  mutate(
    # Convert length from minutes to hours (since they are so long)
    length_hours = round(length / 60, 2),
    # Fix the titles that start with The to the front instead of , The
    title = ifelse(grepl(", The$", title),
                   paste("The", sub(", The$", "", title)),
                   title)
  )

# Display the result in a neat table
knitr::kable(top5_longest %>% select(title, length_hours), col.names = c("Movie Title", "Length (Hours)
```

Movie Title	Length (Hours)
The Cure for Insomnia	87.00
The Longest Most Meaningless Movie in the World	48.00
Four Stars	18.33
Resan	14.55
Out 1	12.88

Question 4:

Of all short movies, which one is the shortest (in minutes)? Which one is the longest? How long are the shortest and the longest short movies?

```r
# Find the shortest and longest short movies
short_movies <- movies %>%
  filter(Short == 1)

# Grab the shortest short movie
shortest_short <- short_movies %>%
  arrange(length) %>%
  select(title, length) %>%
  head(1)

# Grab the longest short movie
longest_short <- short_movies %>%
  arrange(desc(length)) %>%
  select(title, length) %>%
  head(1)

# Combine results into one data frame
results <- rbind(
  data.frame(Type = "Shortest Short Movie", shortest_short),
  data.frame(Type = "Longest Short Movie", longest_short)
)

# Rename the columns
final_output <- results %>%
  select(Type, title, length) %>%
  rename("Movie Title" = title, "Length (Minutes)" = length)

# Display the table and make it look nice using knitr
knitr::kable(final_output, format = "pipe", col.names = c("Type", "Movie Title", "Length (Minutes)"))
```

Type	Movie Title	Length (Minutes)
Shortest Short Movie	17 Seconds to Sophie	1
Longest Short Movie	10 jaar leuven kort	240

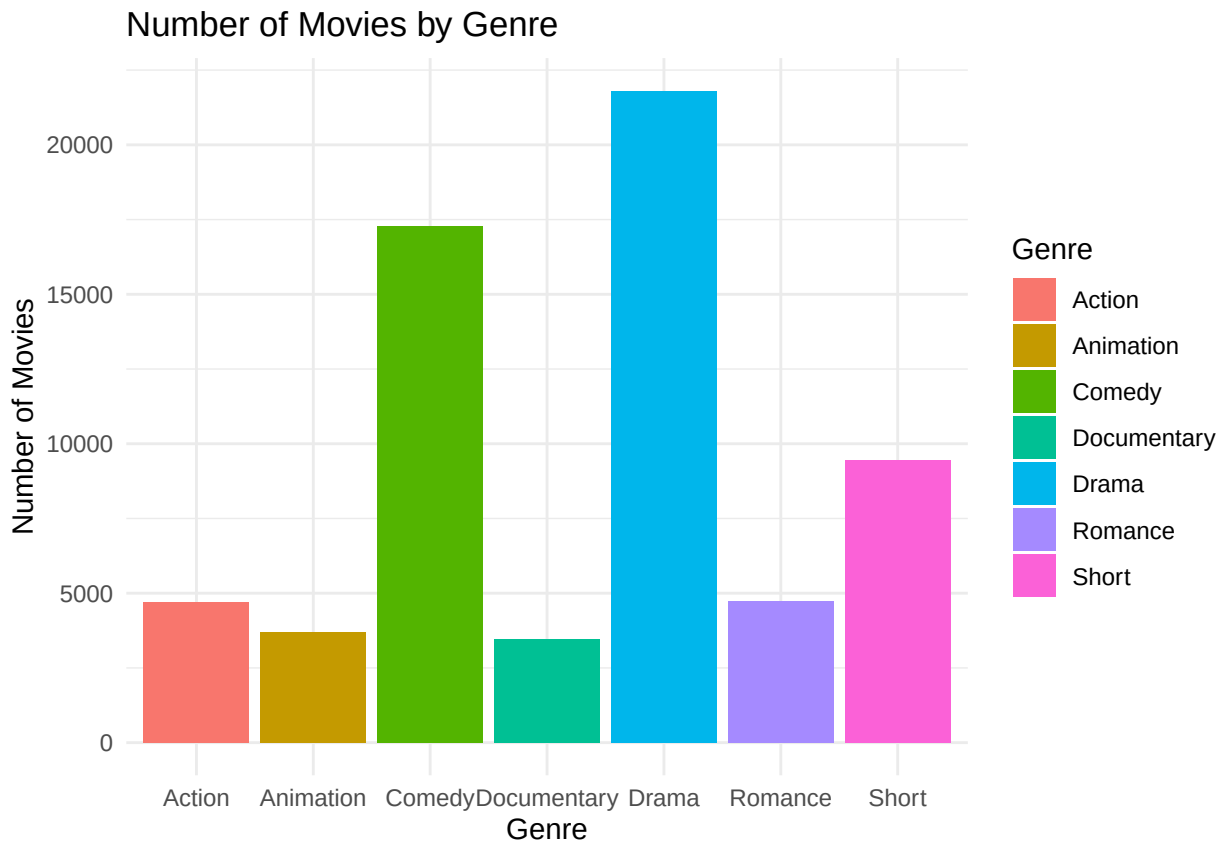
Question 5:

How many movies of each genre (action, animation, comedy, drama, documentary, romance, short) are there in this data base? (use a bar plot)

```r
# Count the number of movies for each genre, removing the na values
genre_counts <- movies %>%
  summarise(
    Action = sum(Action, na.rm = TRUE),
    Animation = sum(Animation, na.rm = TRUE),
    Comedy = sum(Comedy, na.rm = TRUE),
    Drama = sum(Drama, na.rm = TRUE),
    Documentary = sum(Documentary, na.rm = TRUE),
    Romance = sum(Romance, na.rm = TRUE),
    Short = sum(Short, na.rm = TRUE)
  )
```

```r
# Using a df to use later to display
genre_counts_df <- data.frame(
  Genre = c("Action", "Animation", "Comedy", "Drama", "Documentary", "Romance", "Short"),
  Count = as.numeric(genre_counts)
)

# Create a bar plot to show the number of movies in each genre
ggplot(genre_counts_df, aes(x = Genre, y = Count, fill = Genre)) +
  geom_bar(stat = "identity") +
  labs(title = "Number of Movies by Genre", x = "Genre", y = "Number of Movies") +
  theme_minimal()
```



Question 6:

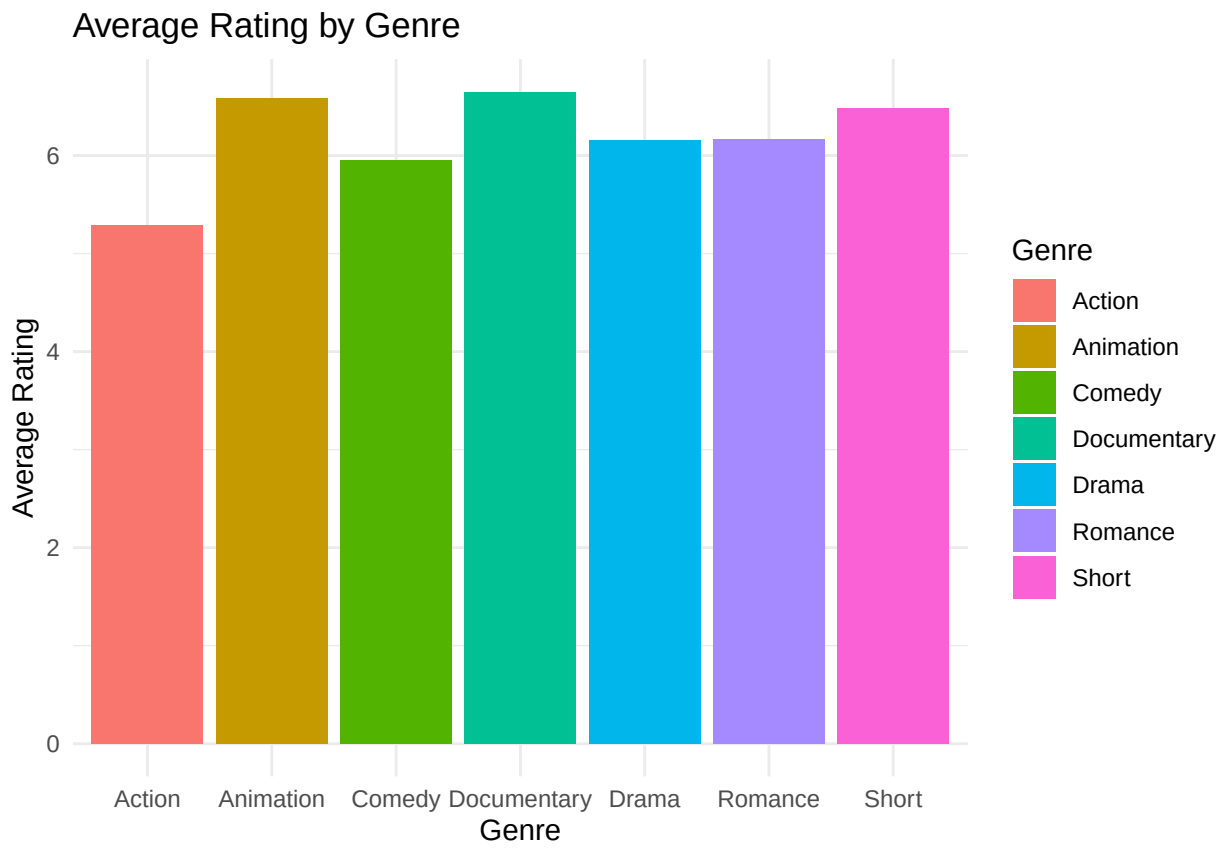
What is the average rating of all movies within each genre? (use a bar plot)

```r
# Calculate the average rating for each genre, removing the na values
average_ratings <- movies %>%
  summarise(
    Action = mean(rating[Action == 1], na.rm = TRUE),
    Animation = mean(rating[Animation == 1], na.rm = TRUE),
    Comedy = mean(rating[Comedy == 1], na.rm = TRUE),
    Drama = mean(rating[Drama == 1], na.rm = TRUE),
    Documentary = mean(rating[Documentary == 1], na.rm = TRUE),
```

```
    Romance = mean(rating[Romance == 1], na.rm = TRUE),
    Short = mean(rating[Short == 1], na.rm = TRUE)
  )

# Create a df to display
average_ratings_df <- data.frame(
  Genre = c("Action", "Animation", "Comedy", "Drama", "Documentary", "Romance", "Short"),
  AverageRating = as.numeric(average_ratings)
)

# Create a bar plot to show the average rating of movies in each genre
ggplot(average_ratings_df, aes(x = Genre, y = AverageRating, fill = Genre)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Rating by Genre", x = "Genre", y = "Average Rating") +
  theme_minimal()
```



Question 7:

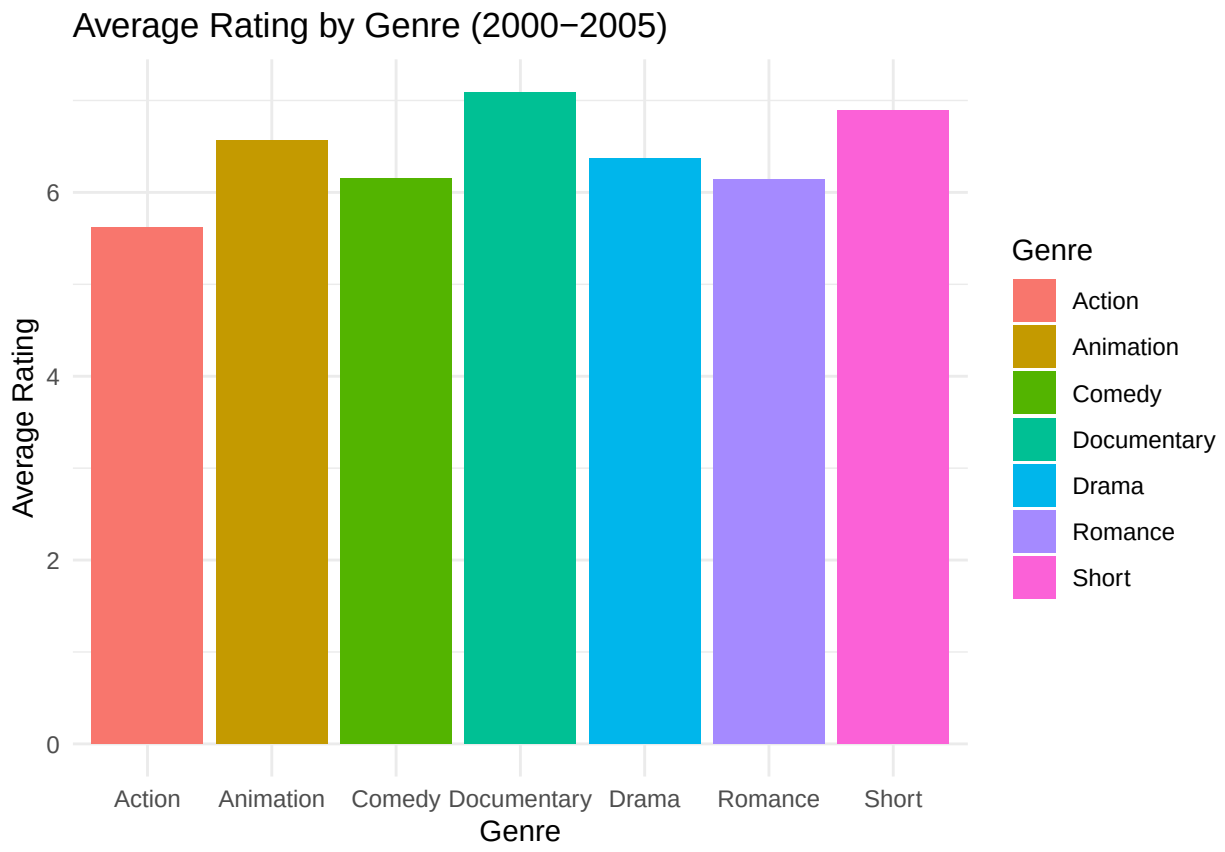
What is the average rating of all movies within each genre that were produced in the years 2000-2005?

```
# Calculate the average rating for each genre (for movies from 2000-2005)
average_ratings_2000_2005 <- movies %>%
  filter(year >= 2000 & year <= 2005) %>%
  summarise(
```

```
    Action = mean(rating[Action == 1], na.rm = TRUE),
    Animation = mean(rating[Animation == 1], na.rm = TRUE),
    Comedy = mean(rating[Comedy == 1], na.rm = TRUE),
    Drama = mean(rating[Drama == 1], na.rm = TRUE),
    Documentary = mean(rating[Documentary == 1], na.rm = TRUE),
    Romance = mean(rating[Romance == 1], na.rm = TRUE),
    Short = mean(rating[Short == 1], na.rm = TRUE)
  )

# Create a df to display
average_ratings_2000_2005_df <- data.frame(
  Genre = c("Action", "Animation", "Comedy", "Drama", "Documentary", "Romance", "Short"),
  AverageRating = as.numeric(average_ratings_2000_2005)
)

# Create a bar plot to show the average ratings (2000-2005)
ggplot(average_ratings_2000_2005_df, aes(x = Genre, y = AverageRating, fill = Genre)) +
  geom_bar(stat = "identity") +
  labs(title = "Average Rating by Genre (2000-2005)", x = "Genre", y = "Average Rating") + theme_minimal
```



Question 8:

For each of the first 6 genres (not including short movies) consider only movies from 1990 until the last year recorded and plot a function of the number of movies in this data base of corresponding genre produced by

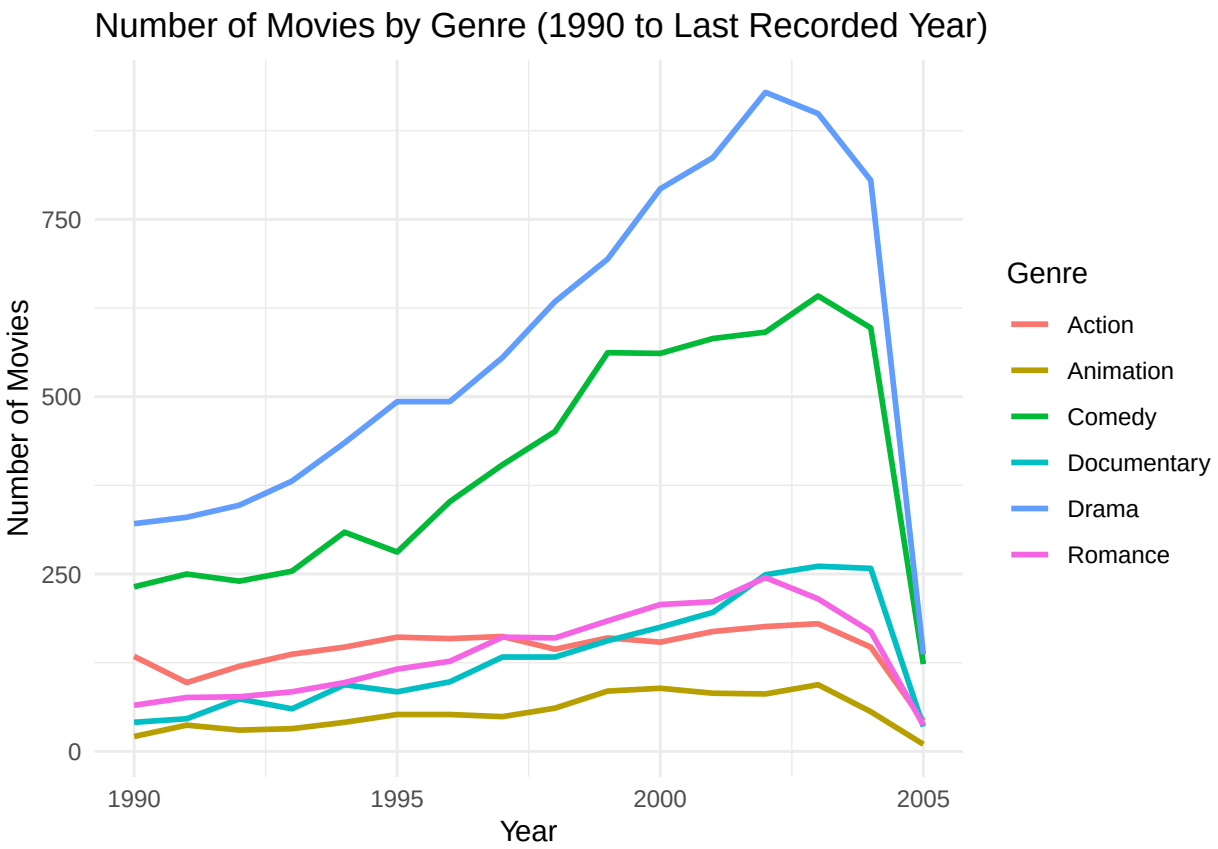
year, for years from 1990 until the last year recorded. For each of the 6 genres you should have one curve, and plot all the curves in the same figure. Naturally, use different colors, and appropriate legend.

```r
# Filter the data for movies from 1990 onward
movies_1990_onward <- movies %>%
  filter(year >= 1990)

# Count the number of movies in each genre by year removing na
genre_by_year <- movies_1990_onward %>%
  group_by(year) %>%
  summarise(
    Action = sum(Action, na.rm = TRUE),
    Animation = sum(Animation, na.rm = TRUE),
    Comedy = sum(Comedy, na.rm = TRUE),
    Drama = sum(Drama, na.rm = TRUE),
    Documentary = sum(Documentary, na.rm = TRUE),
    Romance = sum(Romance, na.rm = TRUE)
  )

# Create a long-format df to display
genre_by_year_long <- data.frame(
  year = rep(genre_by_year$year, 6),
  Genre = factor(rep(c("Action", "Animation", "Comedy", "Drama", "Documentary", "Romance"),
                     each = nrow(genre_by_year))),
  Count = c(genre_by_year$Action, genre_by_year$Animation, genre_by_year$Comedy,
            genre_by_year$Drama, genre_by_year$Documentary, genre_by_year$Romance)
)

# Create the plot with different colors for each genre
ggplot(genre_by_year_long, aes(x = year, y = Count, color = Genre)) +
  geom_line(linewidth = 1) +
  labs(title = "Number of Movies by Genre (1990 to Last Recorded Year)",
       x = "Year", y = "Number of Movies") + theme_minimal()
```

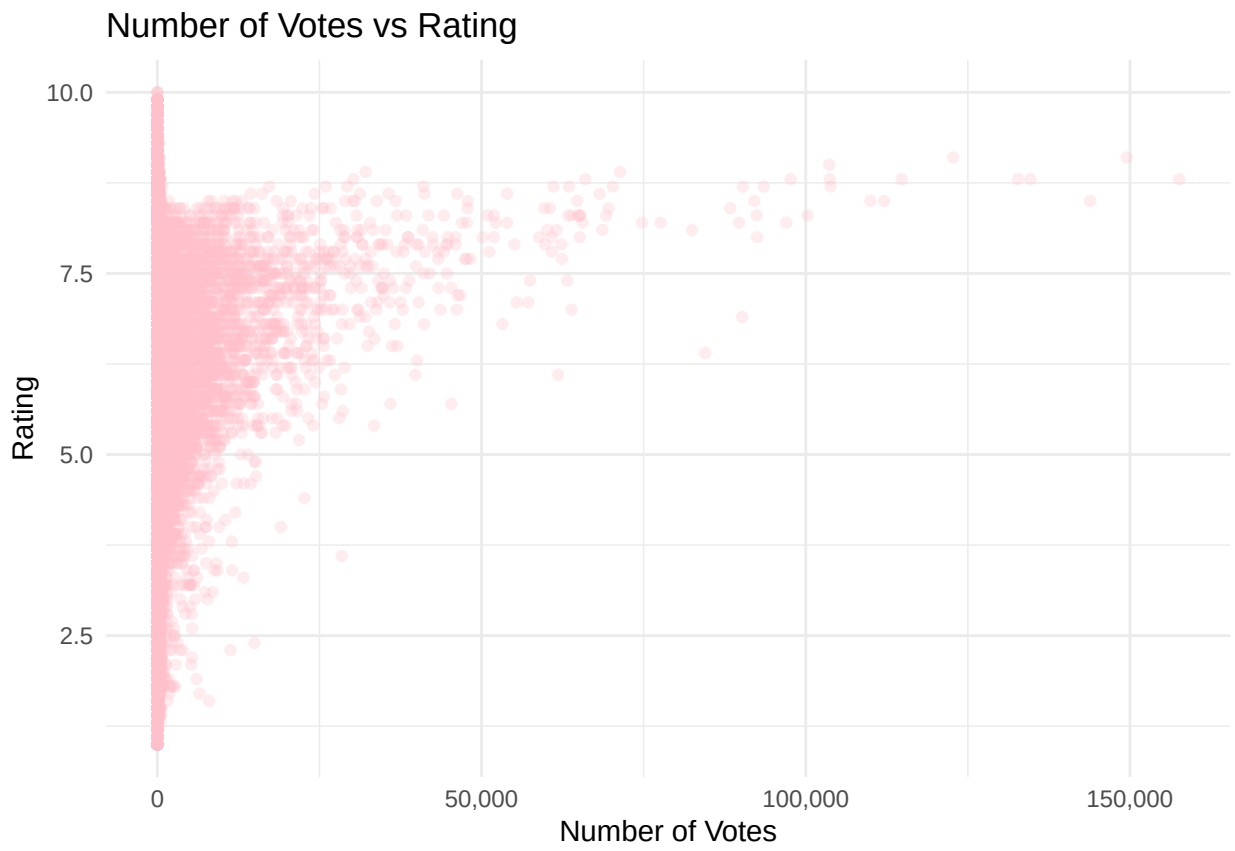


Finally, formulate 3 questions of your choice related to this dataset and answer them. At least one of the answers should include some plot. Use any kind of plot system (base, plotly, or ggplot2). Impress me and the grader with your answers!

Question 9 pt.1:

How are the number of votes correlated to the rating of a movie?

```
# Scatter plot of number of votes vs rating
ggplot(movies, aes(x = votes, y = rating)) +
  geom_point(alpha = 0.3, color = "pink") +
  labs(title = "Number of Votes vs Rating",
       x = "Number of Votes", y = "Rating") +
  theme_minimal() +
  scale_x_continuous(labels = scales::comma)  # Show votes in comma format
```



Question 9 pt.2:

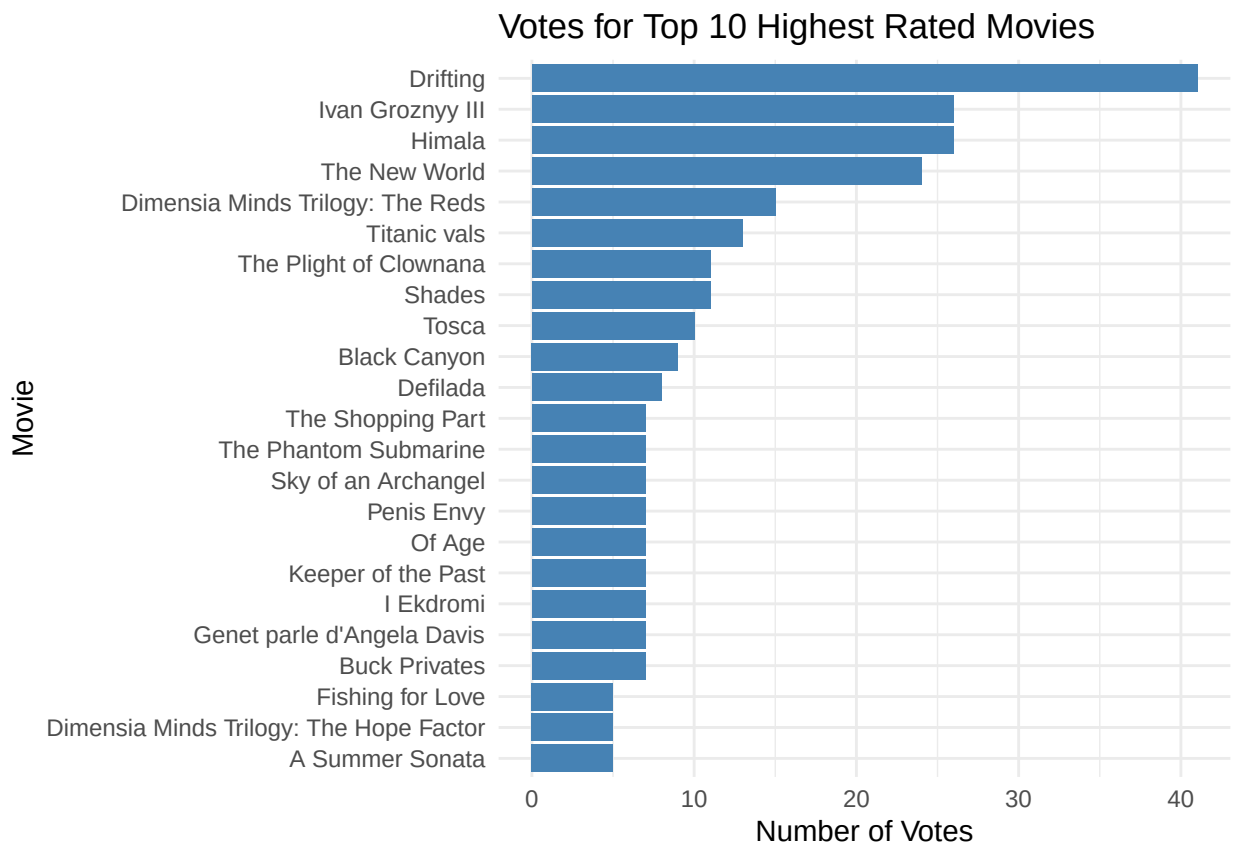
What is the distribution of votes for the 10 highest rating movies?

```
# Function to move 'The', 'A', and 'I' to the front of the movie title
correct_title_format <- function(title) {
  if (grepl(", The$", title)) {
    title <- gsub(", The$", "", title)
    title <- paste("The", title)
  } else if (grepl(", A$", title)) {
    title <- gsub(", A$", "", title)
    title <- paste("A", title)
  } else if (grepl(", I$", title)) {
    title <- gsub(", I$", "", title)
    title <- paste("I", title)
  }
  return(title)
}

# Grab the top 10 highest-rated movies
top_rated_movies_votes <- movies %>%
  filter(!is.na(rating)) %>%
  arrange(desc(rating)) %>%
  top_n(10, rating)
```

```r
# Apply the function to correct the movie titles
top_rated_movies_votes$title <- sapply(top_rated_movies_votes$title, correct_title_format)

# Plot the number of votes for the top 10 movies
ggplot(top_rated_movies_votes, aes(x = reorder(title, votes), y = votes)) +
  geom_bar(stat = "identity", fill = "steelblue") +
  labs(title = "Votes for Top 10 Highest Rated Movies",
       x = "Movie", y = "Number of Votes") +
  coord_flip() +
  theme_minimal() +
  scale_y_continuous(labels = scales::comma)  # Show votes in comma format
```



Question 9 pt.3:

What is the average length of movies in each genre?

```r
# Calculate the average length for each genre and create a data frame
length_by_genre <- data.frame(
  Genre = c("Action", "Animation", "Comedy", "Drama", "Documentary", "Romance"),
  Avg_Length_Minutes = c(
    mean(movies$length[movies$Action == 1], na.rm = TRUE),
    mean(movies$length[movies$Animation == 1], na.rm = TRUE),
    mean(movies$length[movies$Comedy == 1], na.rm = TRUE),
    mean(movies$length[movies$Drama == 1], na.rm = TRUE),
```

```
    mean(movies$length[movies$Documentary == 1], na.rm = TRUE),
    mean(movies$length[movies$Romance == 1], na.rm = TRUE)
  )
)

# Display the result as a neat table
knitr::kable(length_by_genre, col.names = c("Genre", "Average Length (Minutes)"),
      caption = "Average Movie Length by Genre")
```

Table 4: Average Movie Length by Genre

Genre	Average Length (Minutes)
Action	98.99851
Animation	19.69431
Comedy	75.36402
Drama	95.99376
Documentary	70.42512
Romance	99.02108