Reliability of Movie Ratings

Lihao Liu, Zikang Chen, Hinn Zhang

Contents

Introduction

Imagine sitting on your couch and thinking about winding down from a day of busy work. You turn on your streaming services to pick your movie for the night. And yet, you are now just realizing that the paradox of choice reveals the real challenge of the day. Despite having thousands of movies, TV shows, and video content available at our fingertips, we often find ourselves scrolling through countless options but still not being able to decide.

Some might argue that the existing recommendation systems that are based on prior searching and watch history help elevate the effort of finding taste-matching content.[1] However, this might not hold true for everyone. Research has shown that history-based algorithms do not account for individual differences, such as personalities, which largely affect the user's willingness to accept the recommendations.[2] Another channel that does not constrain options to within the scope of users' preference to get inspiration for a movie night is by referring to online user ratings, such as those on IMDb. Similar to online shopping, checking out the reviews before committing three hours to a movie gives reassurance and a chance to quantitatively compare choices.

These voluntary ratings are vital for saving time and getting valuable insights into a movie's strengths and weaknesses. Yet, how reliable are the user reviews? Some researchers have found an increasing discrepancy between critic scores and user ratings.[3] And that the general public is giving out less indicative scores of the quality of the film. Besides the uncertainty on credibility, user reviews progressively suffer from the lack of variability in user diversity.[4] These aforementioned phenomena give rise to the question of whether user reviews lose their value over time. Consequently, in this project, we aim to analyze user reviews to offer more context for users to effectively and correctly interpret online information about movies.

To further investigate movie reviews, we identified the popular movie review site IMDb as the source of user comments and ratings of movies available on the most-anticipated streaming service Netflix as the research scope. Our quantitative analysis considers user scores as the component to evaluate the underlying trend among user inputs. This approach is based on the typical user flow. Specifically, users enquire about the quality of a movie through three steps: search for the movie, look at the rating scores, and click to see textual reviews if interested or necessary. User rating scores are the first-available information in the process and should have a deciding factor in the selection process.

[1] https://towardsdatascience.com/the-4-recommendation-engines-that-can-predict-your-movie-tastes-109dc4e10c52
[2] https://link.springer.com/article/10.1007/s10796-017-9800-0
[3] https://stephenfollows.com/are-film-critics-becoming-out-of-sync-with-audiences/
[4] https://news.usc.edu/144379/usc-study-finds-film-critics-like-filmmakers-are-largely-white-and-male/

Problem Statement

Moviegoers who are interested in the evaluation of movies before consumption has limited knowledge about how online user reviews work and their subjectivity to misinformation. By looking at the trend of recent year's reviews, we hope to achieve a more guided overview of online viewer ratings to provide a better decision-making process for viewers.

Alternatively, are the reviews available on IMDb reliable in correctly reflecting the quality of the movie or are they influenced by external factors that are less relevant to movies? Therefore, we identify the following variables related to movies in order to investigate further.

1. Release year: Do people rate movies differently over the years since 2010?
2. Genre: Do movies receive lower/higher ratings due to their genres?
3. Holiday season: Does a movie released during a holiday season receive a higher/lower rating than a movie released during other times of the year?

Data

The two datasets for this project are collected from Kaggle. The first dataset, "Netflix Movies And TV Shows", contains a series of information, including:

- `type`: the type of movie and TV shows
- `title`: title of the show
- `director`: director of the movie/show
- `cast`: actors and actresses involved in the movie/show
- `country`: Country of origin
- `date_added`: Date added to Netflix
- `release_year`: the release year of the show
- `rating`: TV rating of the movie/show (eg. PG-13, TV-MA)
- `duration`: length of the movie/show
- `listed_in`: genre
- `description`: the summary description of the movie/show

The second dataset, "Netflix popular movies dataset", contains an extra column of user ratings and votes to complement the first dataset, including:

- `rating`: users' ratings on IMDb
- `votes`: the number of ratings a show/movie receives

We plan to merge the two datasets as the second dataset contains ratings for movies while the first dataset consists of all movies (as opposed to popular shows in the second dataset).

Data Import

Here we will load all the packages we need for this project.

```
library(tidyverse)
library(dplyr)
library(skimr)
library(viridis)
library(tidymodels)
library(ggridges)
library(rmdformats)
```

Here we will import the csv file into the studio and take a look at the data set:

```
data <- read_csv("netflix_titles.csv")
```

```
## Rows: 8807 Columns: 12-- Column specification --------------------------------------------------------
```

```
## Delimiter: ","
## chr (11): show_id, type, title, director, cast, country, date_added, rating,...
## dbl  (1): release_year
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
movies <- read_csv("n_movies.csv")
```

```
## Rows: 9957 Columns: 9-- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (7): title, year, certificate, duration, genre, description, stars
## dbl (1): rating
## num (1): votes
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
glimpse(data)
```

```
## Rows: 8,807
## Columns: 12
## $ show_id      <chr> "s1", "s2", "s3", "s4", "s5", "s6", "s7", "s8", "s9", "s1~
## $ type         <chr> "Movie", "TV Show", "TV Show", "TV Show", "TV Show", "TV ~
## $ title        <chr> "Dick Johnson Is Dead", "Blood & Water", "Ganglands", "Ja~
## $ director     <chr> "Kirsten Johnson", NA, "Julien Leclercq", NA, NA, "Mike F~
## $ cast         <chr> NA, "Ama Qamata, Khosi Ngema, Gail Mabalane, Thabang Mola~
## $ country      <chr> "United States", "South Africa", NA, NA, "India", NA, NA,~
## $ date_added   <chr> "September 25, 2021", "September 24, 2021", "September 24~
## $ release_year <dbl> 2020, 2021, 2021, 2021, 2021, 2021, 2021, 1993, 2021, 202~
## $ rating       <chr> "PG-13", "TV-MA", "TV-MA", "TV-MA", "TV-MA", "TV-MA", "PG~
## $ duration     <chr> "90 min", "2 Seasons", "1 Season", "1 Season", "2 Seasons~
## $ listed_in    <chr> "Documentaries", "International TV Shows, TV Dramas, TV M~
## $ description  <chr> "As her father nears the end of his life, filmmaker Kirst~
```

```
glimpse(movies)
```

```
## Rows: 9,957
## Columns: 9
## $ title       <chr> "Cobra Kai", "The Crown", "Better Call Saul", "Devil in Oh~
## $ year        <chr> "(2018- )", "(2016- )", "(2015-2022)", "(2022)", "(2022- )~
## $ certificate <chr> "TV-14", "TV-MA", "TV-MA", "TV-MA", "TV-MA", "TV-MA", "TV-~
## $ duration    <chr> "30 min", "58 min", "46 min", "356 min", "24 min", "45 min~
## $ genre       <chr> "Action, Comedy, Drama", "Biography, Drama, History", "Cri~
## $ rating      <dbl> 8.5, 8.7, 8.9, 5.9, 8.6, 7.8, 9.2, 9.5, 6.3, 6.2, 8.7, 4.7~
## $ description <chr> "Decades after their 1984 All Valley Karate Tournament bou~
## $ stars       <chr> "['Ralph Macchio, ', 'William Zabka, ', 'Courtney Henggele~
## $ votes       <dbl> 177031, 199885, 501384, 9773, 15413, 116358, 502160, 18313~
```

We see that the original dataset with movie's features contains 8807 observations with 12 variables, and the dataset with movie ratings contains 9957 observations and 9 variables.

Data Wrangling

Since the data set we eventually wish to utilize is not two separate tibbles, we need to find a way to extract the useful columns and take the combination of two data sets and conduct our analysis from the integrated tibble.

First, we will merge the two data sets so as to get each movie's rating and votes and also leave only movies

with us, since we are not considering TV shows.

```
#filter out only movies
data <- data |> filter(type == "Movie")

#get only the rating and voting columns
movies <- movies[, c("title", "rating", "votes")]

#merge the two data sets
merged <- merge(data, movies, by = "title")
```

Then we want to rename the columns to make their names consistent.

```
# rename columns
merged <- merged |>
  rename("certificate" = "rating.x",
         "rating" = "rating.y")
```

We also want to change the format of the date column from character to date for easier access.

```
# convert dates from character to date type
merged$date_added <- mdy(merged$date_added)
```

Moreover, we extract the month from the date so that we can generate plots in the EDA section.

```
# create a month column
merged$month <- month(merged$date_added)
```

Since we decide to set the unit of durations of movies into minutes, we can clean the column of Duration into a column of integers.

```
merged$duration <- substring(merged$duration, 1, nchar(merged$duration)-4)
merged$duration <- as.integer(merged$duration)
```

Let's take a look at the wrangled data set.

```
skim(merged)
```

Table 1: Data summary

Name	merged
Number of rows	1782
Number of columns	15
Column type frequency:	
character	9
Date	1
numeric	5
Group variables	None

Variable type: character

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
title	0	1.00	2	83	0	1740	0
show_id	0	1.00	2	5	0	1740	0
type	0	1.00	5	5	0	1	0

skim_variable	n_missing	complete_rate	min	max	empty	n_unique	whitespace
director	53	0.97	2	167	0	1461	0
cast	170	0.90	4	575	0	1512	0
country	105	0.94	4	83	0	257	0
certificate	0	1.00	1	6	0	13	0
listed_in	0	1.00	6	64	0	171	0
description	0	1.00	90	222	0	1740	0

Variable type: Date

skim_variable	n_missing	complete_rate	min	max	median	n_unique
date_added	0	1	2009-11-18	2021-09-25	2019-06-18	921

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
release_year	0	1.00	2016.90	6.01	1944	2016.00	2018.0	2020.00	2021	
duration	1	1.00	94.94	26.93	3	83.00	96.0	109.00	312	
rating	20	0.99	6.19	1.10	2	5.50	6.3	7.00	9	
votes	20	0.99	21558.62	89565.78	8	830.25	2741.0	10761.25	1819157	
month	0	1.00	6.62	3.40	1	4.00	7.0	10.00	12	

After data cleaning, the new dataset contains 1782 samples (movies) and some missing values in columns including "director", "cast", and "country". However, we do not plan to use these features in subsequent steps.

```
merged |>
  group_by(title) |>
  filter(n() > 1)
```

```
## # A tibble: 71 x 15
## # Groups:   title [29]
##    title  show_id type  direc~1 cast  country date_added relea~2 certi~3 durat~4
##    <chr>  <chr>   <chr> <chr>   <chr> <chr>   <date>       <dbl> <chr>     <int>
##  1 Aurora s2844   Movie Cristi~ Cris~ Romani~ 2020-03-04    2010 TV-MA       186
##  2 Aurora s2844   Movie Cristi~ Cris~ Romani~ 2020-03-04    2010 TV-MA       186
##  3 Awake  s747    Movie Mark R~ Gina~ United~ 2021-06-09    2021 TV-MA        97
##  4 Awake  s747    Movie Mark R~ Gina~ United~ 2021-06-09    2021 TV-MA        97
##  5 Block~ s6335   Movie July H~ Syru~ France  2018-01-24    2017 TV-MA        85
##  6 Block~ s6335   Movie July H~ Syru~ France  2018-01-24    2017 TV-MA        85
##  7 Bodyg~ s1675   Movie Siddiq~ Salm~ India   2020-11-19    2011 TV-14       130
##  8 Bodyg~ s1675   Movie Siddiq~ Salm~ India   2020-11-19    2011 TV-14       130
##  9 Death~ s5319   Movie Adam W~ Will~ United~ 2017-08-25    2017 TV-MA       100
## 10 Death~ s5319   Movie Adam W~ Will~ United~ 2017-08-25    2017 TV-MA       100
## # ... with 61 more rows, 5 more variables: listed_in <chr>, description <chr>,
## #   rating <dbl>, votes <dbl>, month <dbl>, and abbreviated variable names
## #   1: director, 2: release_year, 3: certificate, 4: duration
```

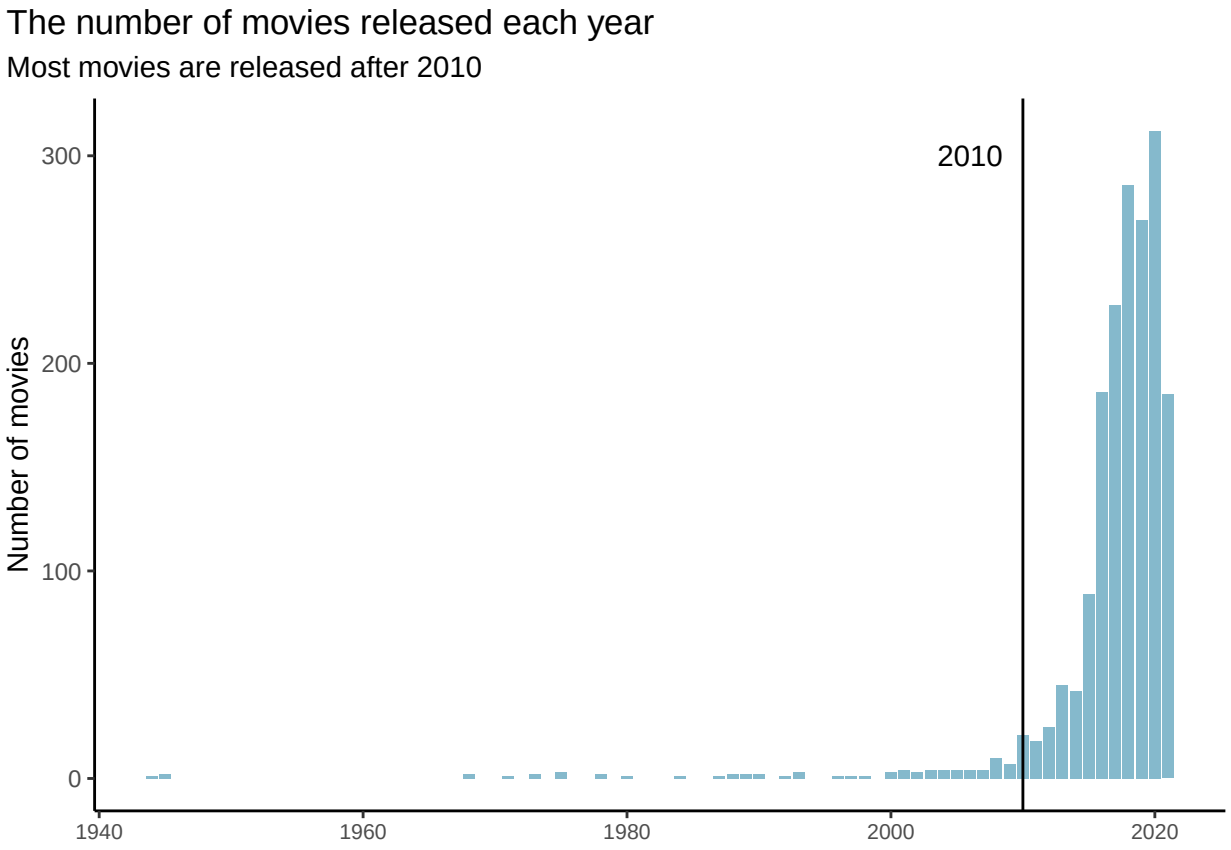
We observed around 40 possible duplicates in the dataset. Yet, most repeated entries are re-entries due to missing values or that they are different movies with the same name (with different ratings). This issue will be addressed in the analysis automatically.

Analysis

Exploratory Data Analysis

How many movies are released in each year upto 2021? We first want to see when the movies are released in this dataset.

```
merged |>
  ggplot(aes(x = release_year)) +
  geom_bar(fill = "#85B9CC") +
  labs(x = "Year released",
       y = "Number of movies",
       title = "The number of movies released each year",
       subtitle = "Most movies are released after 2010") +
  theme(text = element_text(size = 15),
        plot.title.position = "plot") +
  theme_minimal() +
  geom_vline(xintercept = 2010, color = "black") +
  annotate("text", x = 2006, y = 300, label = "2010", color = "black", size = 4) +
  theme_classic() +
  theme(axis.title.x = element_blank(),
        plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5),
        legend.position = "none")
```

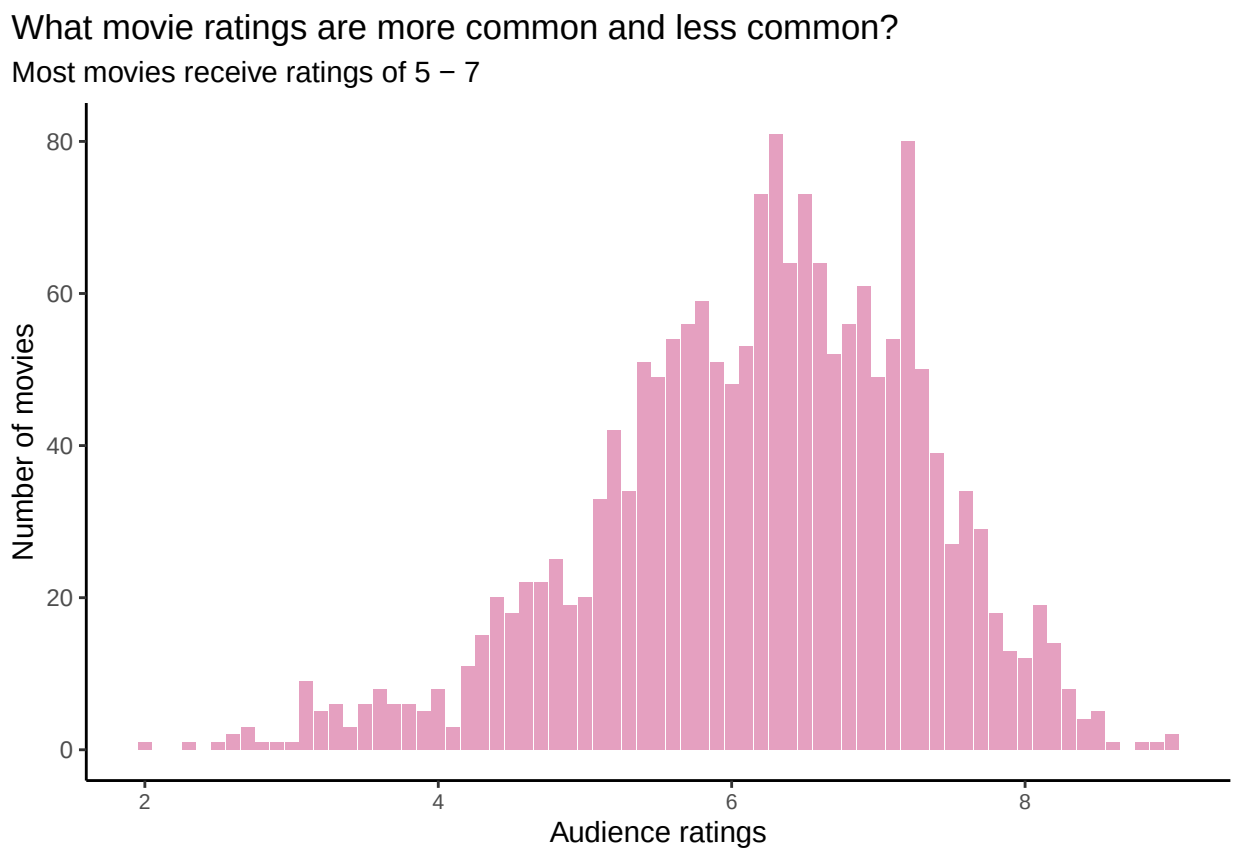


We see that most movies are released after 2010, which suggest most movies are contemporary. We decide to focus on the movies released after 2010 when we answer our questions in the Data Analysis section, so that we can safely draw conclusions using our current knowledge.

The distribution of ratings We also want to see what ratings do movies generally receive.

```
merged |>
  ggplot(aes(x = rating)) +
  geom_bar(fill = "#E5A0C0") +
  labs(x = "Audience ratings",
       y = "Number of movies",
       title = "What movie ratings are more common and less common?",
       subtitle = "Most movies receive ratings of 5 - 7") +
  theme_classic() +
  theme(
        plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5),
        legend.position = "none")
```

```
## Warning: Removed 20 rows containing non-finite values (`stat_count()`).
```

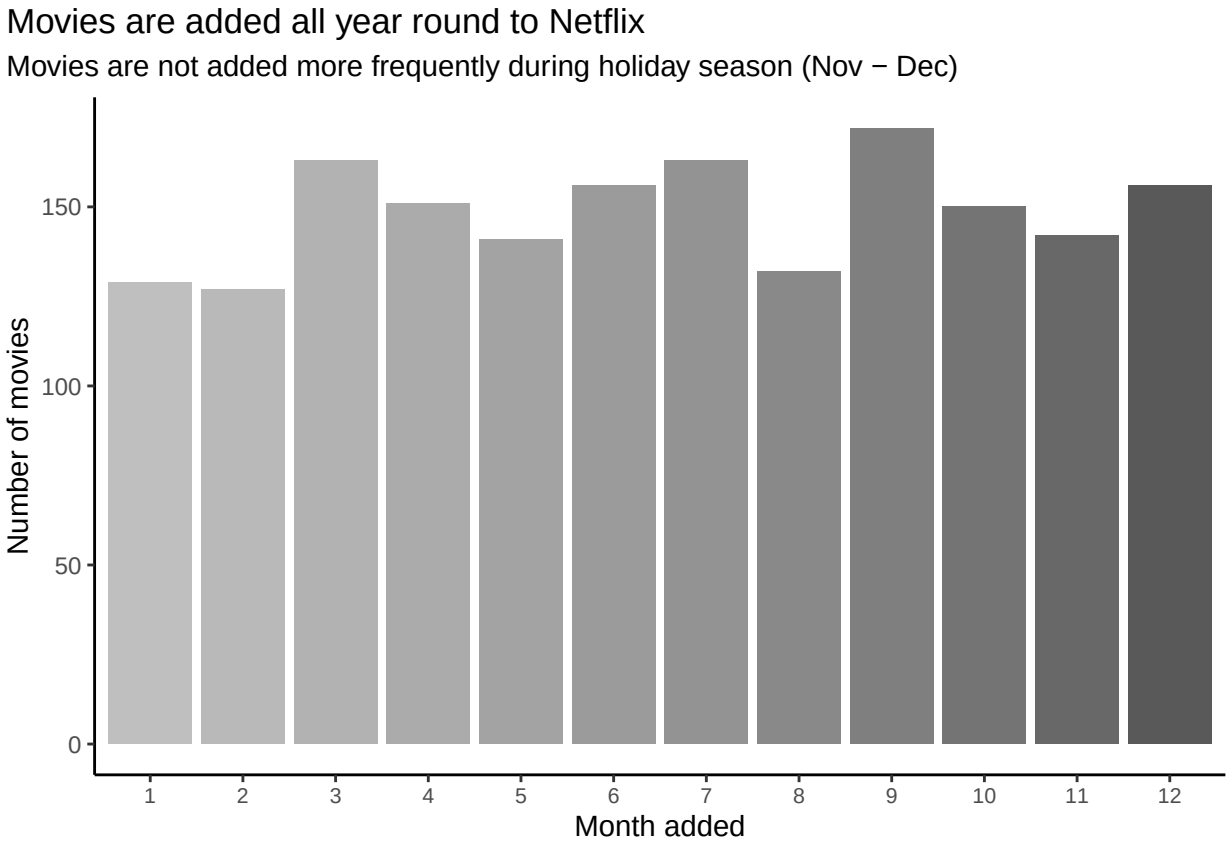


Release date of movies The plot shows most movies receive a rating in the range of 5 to 7, so we can infer that a rating above 7 tends to be a high rating, while a rating below 5 tends to be a low rating.

We also want to see when movies are added to Netflix in a year.

```
merged |>
  ggplot(aes(x = as.factor(month), fill = as.factor(month))) +
  geom_bar() +
  labs(x = "Month added",
       y = "Number of movies",
       title = "Movies are added all year round to Netflix",
```

```
      subtitle = "Movies are not added more frequently during holiday season (Nov - Dec)") +
  scale_fill_grey(start = 0.75, end = 0.35) +
  theme_classic() +
  theme(legend.position = "none",
        plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5))
```

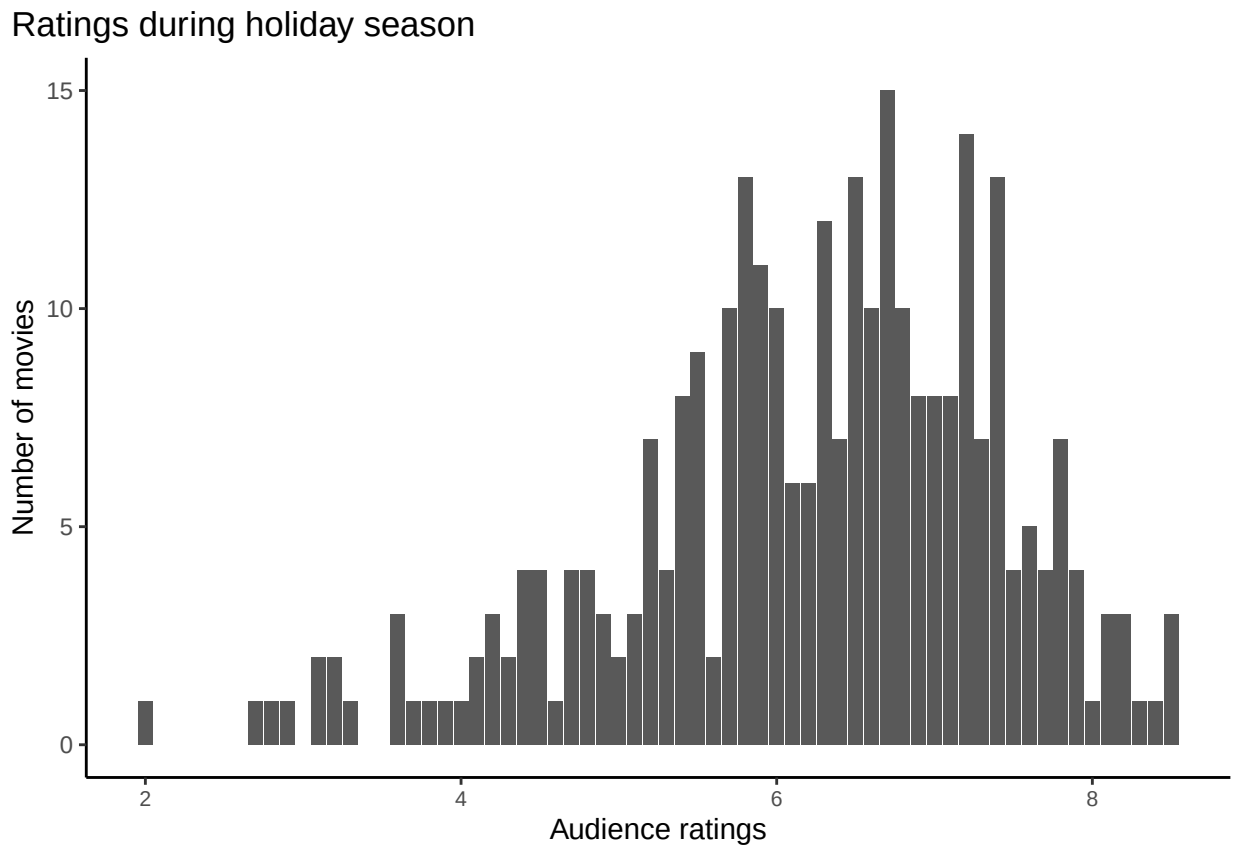


The sample sizes of movies on Netflix are roughly even across all months, so we can safely compare holiday season movies to non-holiday season movies.

```
merged |>
  filter(month == 11 | month == 12) |>
  ggplot(aes(x = rating)) +
  geom_bar() +
  labs(x = "Audience ratings",
       y = "Number of movies",
       title = "Ratings during holiday season") +
  theme(text = element_text(size = 15),
        plot.title.position = "plot") +
  theme_classic() +
  theme(
        plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5),
        legend.position = "none")
```

How do user ratings differ in different time of the year?

```
## Warning: Removed 3 rows containing non-finite values (`stat_count()`).
```

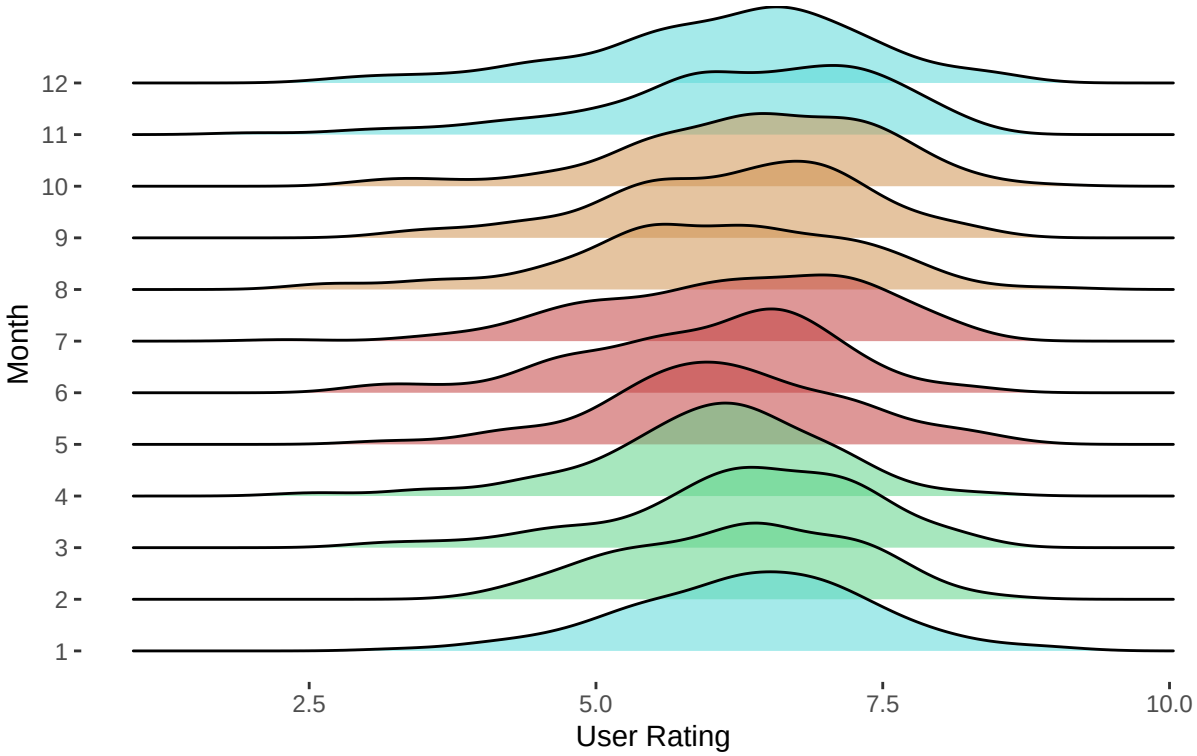


The shape of the distribution of ratings for movies released in the holiday season is roughly the same as that of all movie ratings. Yet, we look to investigate the distributions more closely.

```
merged |>
  ggplot(aes(x = rating, y = as.factor(month), fill = as.factor(month),alpha = 0.9)) +
  geom_density_ridges() +
  labs(title = "User ratings on IMDb in different months",
       subtitle = "The distributions of ratings in different months seem consistent throughout the year
       x = "User Rating",
       y = "Month") +
   scale_fill_manual(values = c("#5ADAD9","#5ED389","#5ED389","#5ED389",
                                "#C44242","#C44242","#C44242","#D09452",
                                "#D09452","#D09452","#5ADAD9","#5ADAD9")) +
  theme(legend.position = "none",
        plot.title.position = "plot",
        panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank(),
        panel.background = element_blank())
```

```
## Picking joint bandwidth of 0.345
```

```
## Warning: Removed 20 rows containing non-finite values (`stat_density_ridges()`).
```



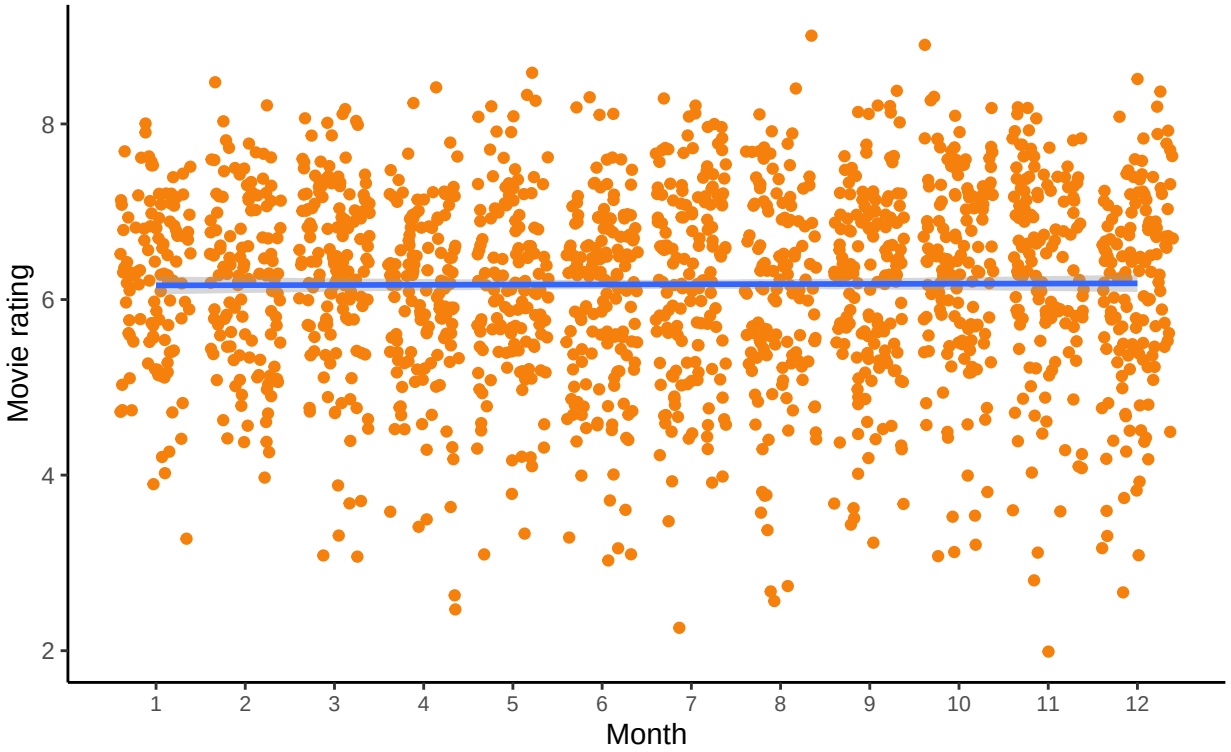
The above visualization contains distribution of user ratings during all 12 months during a year. No significant seasonal bias is observed in terms of scores of movies.

```
merged |>
  filter(release_year > 2010) |>
  ggplot(aes(x = month, y = rating)) +
  geom_jitter(color = "#F5800B") +
  geom_smooth(method = "lm") +
  labs(title = "Trend of ratings by month",
       subtitle = "The variabiilty of ratings remain roughly the same across months",
       x = "Month",
       y = "Movie rating") +
  theme_classic() +
  scale_x_continuous(breaks = seq(1, 12)) +
  theme(plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5),
        legend.position = "none")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 16 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 16 rows containing missing values (`geom_point()`).
```



The visualization shows movies are added to Netflix all year round, and there is a roughly even distribution of movies across all months. Therefore, we can safely carry out our analysis when we compare holiday seasons to non-holiday seasons.

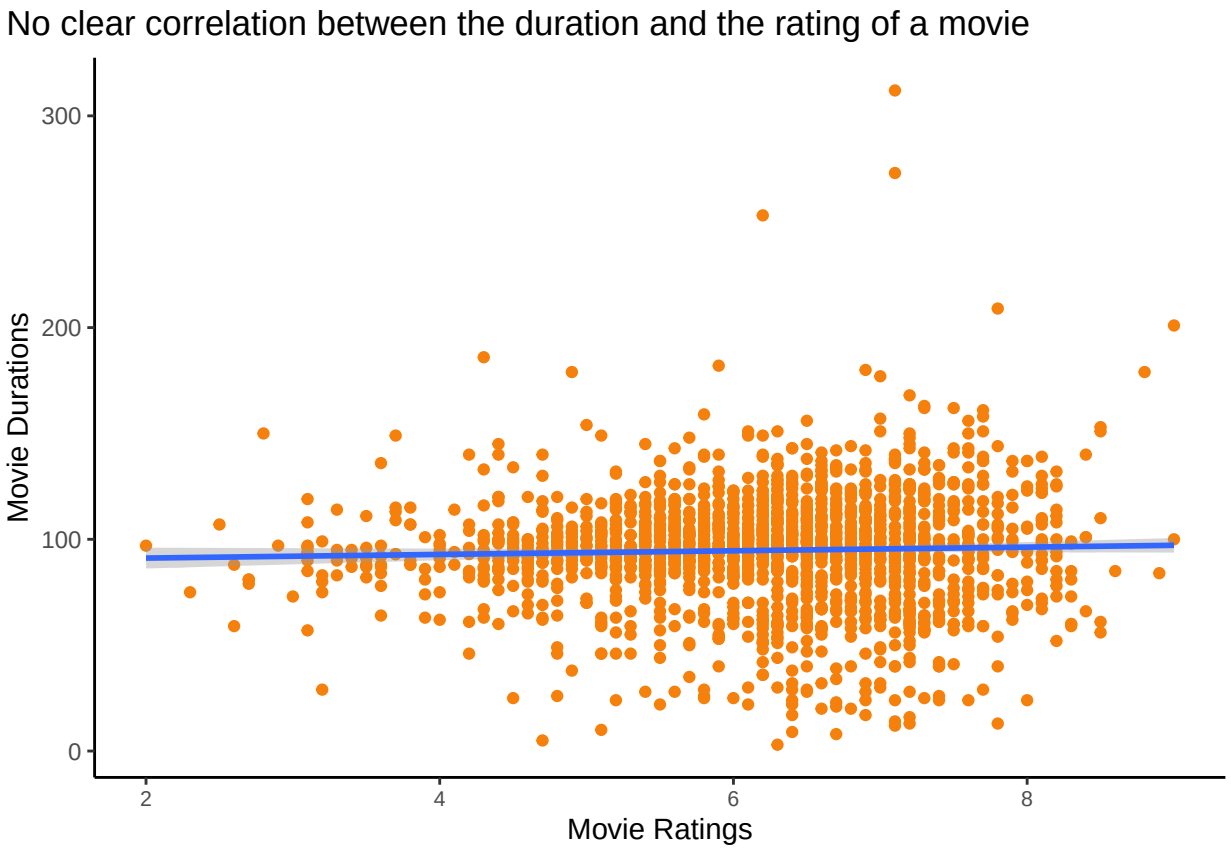
Now we are done with checking the basis of our analysis, we want to gain more information from the data set to have a better understanding of it. One thing really triggered our interests is the relationship between ratings and the lengths of the movies.

```
ggplot(merged, aes(x = rating, y = duration)) +
  geom_point(color = "#F5800B") +
  geom_smooth(method = "lm")+
  labs(x = "Movie Ratings",
       y = "Movie Durations",
       title = "No clear correlation between the duration and the rating of a movie") +
  theme_classic() +
  theme(plot.title.position = "plot",
        axis.text.x = element_text(size = 7.5),
        legend.position = "none")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 21 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 21 rows containing missing values (`geom_point()`).
```



However, although we anticipated that the ratings of movies that are either too long or too short may receive low ratings, we cannot actually get this conclusion from the plot. There seems to be no correlation between the duration and the rating of a movie just through observing the plot.

We are also curious about the what movie genres are included in the dataset.

```
merged |>
  filter(release_year >= 2010) |>
  separate_rows(listed_in, sep = ", ") |>
  group_by(listed_in) |>
  summarise(count = n()) |>
  arrange(desc(count))
```

```
## # A tibble: 20 x 2
##    listed_in                count
##    <chr>                    <int>
##  1 International Movies       738
##  2 Dramas                     636
##  3 Comedies                   383
##  4 Documentaries              280
##  5 Independent Movies         234
##  6 Thrillers                  209
##  7 Stand-Up Comedy            204
##  8 Action & Adventure         182
##  9 Children & Family Movies   156
## 10 Romantic Movies            143
## 11 Horror Movies              106
```

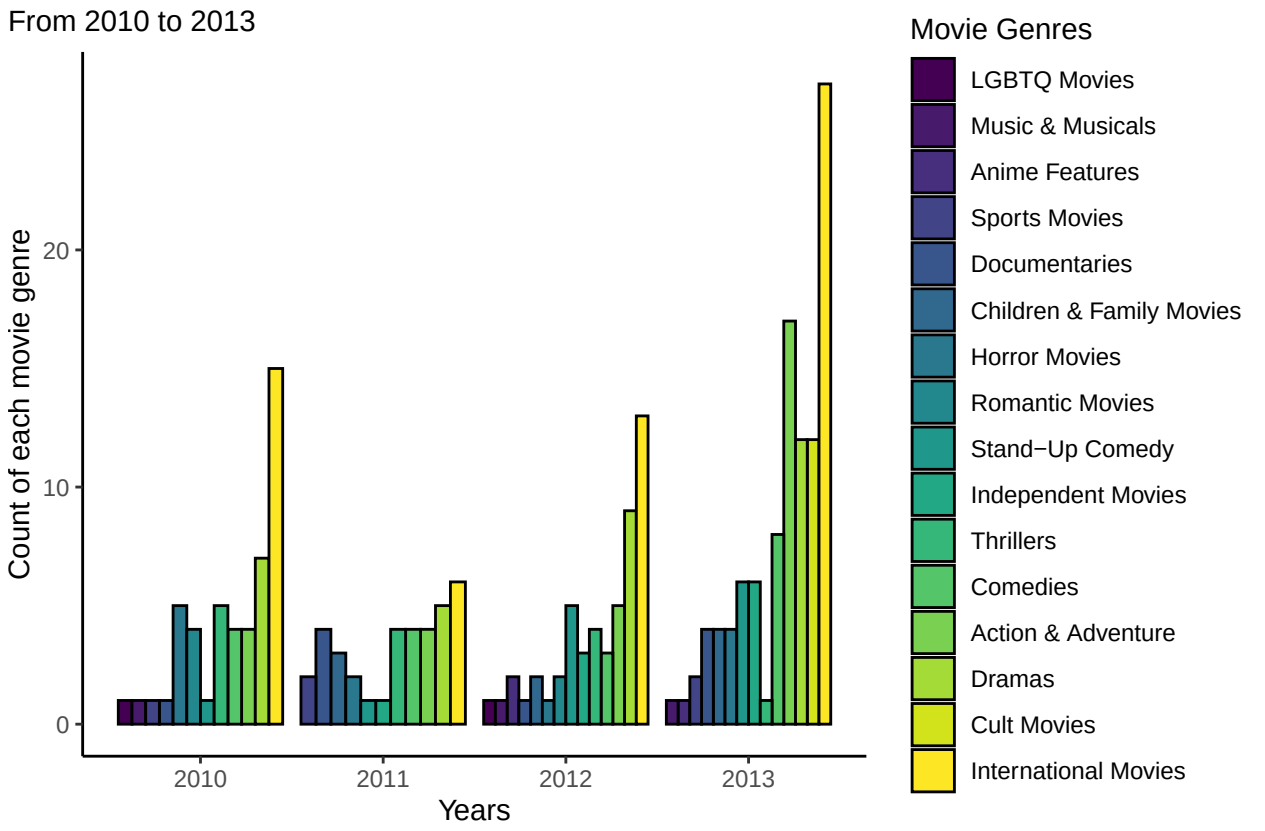
```
## 12 Music & Musicals            81
## 13 Sports Movies               63
## 14 LGBTQ Movies                41
## 15 Sci-Fi & Fantasy            37
## 16 Anime Features              15
## 17 Cult Movies                 14
## 18 Faith & Spirituality        13
## 19 Movies                       4
## 20 Classic Movies               1
```

```
#create a tibble in longer format with rows for each combination of movie name and movie type from 2010
merged_long_10_13 <- merged |>
  filter(release_year >= 2010) |>
  filter(release_year < 2014) |>
  separate_rows(listed_in, sep = ", ") |>
  group_by(release_year,listed_in) |>
  summarise(count = n())

#create a tibble in longer format with rows for each combination of movie name and movie type from 2014
merged_long_14_17 <- merged |>
  filter(release_year >= 2014) |>
  filter(release_year < 2018) |>
  separate_rows(listed_in, sep = ", ") |>
  group_by(release_year,listed_in) |>
  summarise(count = n())

#create a tibble in longer format with rows for each combination of movie name and movie type from 2018
merged_long_18_21 <- merged |>
  filter(release_year >= 2018) |>
  filter(release_year < 2022) |>
  separate_rows(listed_in, sep = ", ") |>
  group_by(release_year,listed_in) |>
  summarise(count = n())
```

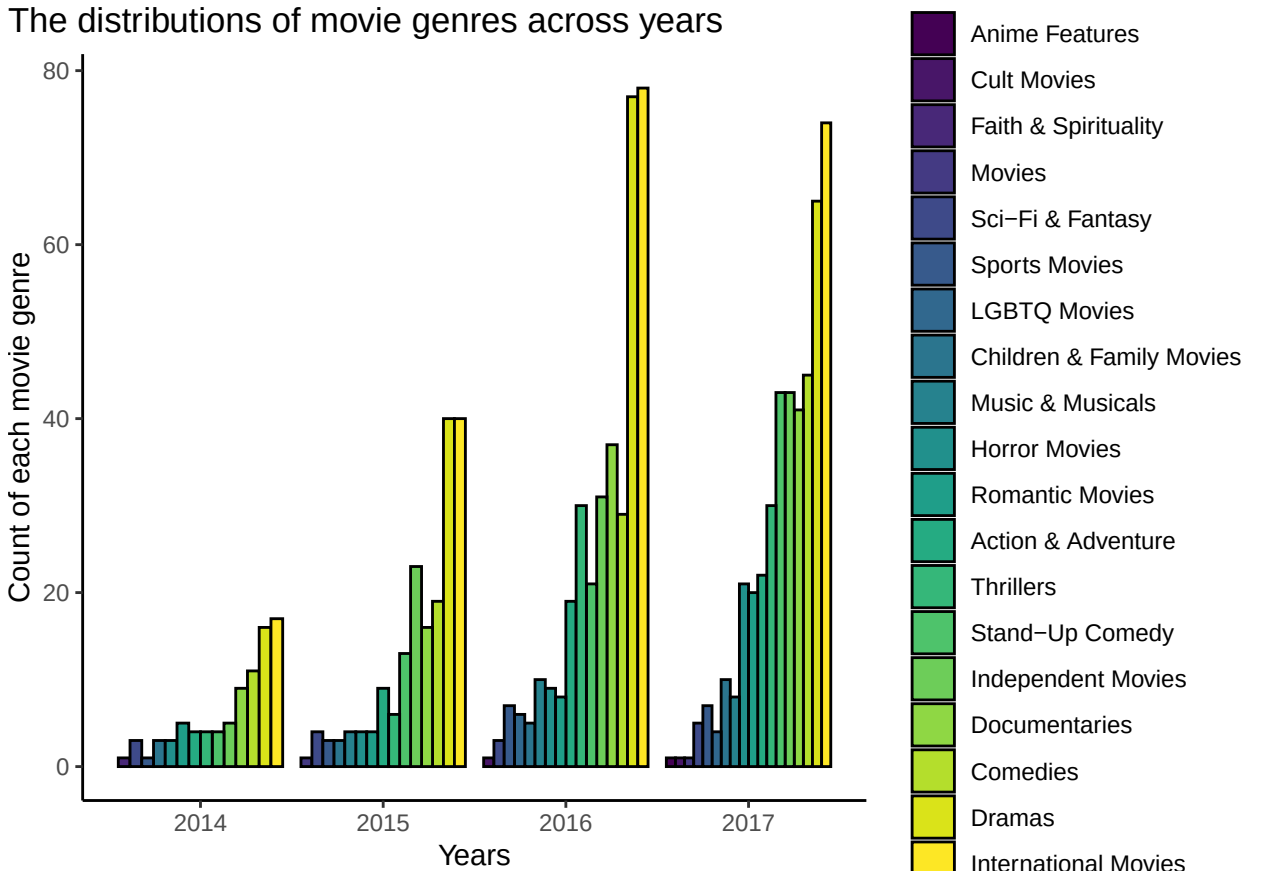
Movies count by genre Overall, there are an increasing number of movies produced each year from 2010 to 2021, following the previous observation. Further, we see that "International Movies", "Dramas", and "Comedies" appear the most frequently. We wanted to see if this still holds true in each year since 2010, so we will check the distribution of movie genres across different years. Since there are many years in the data, we are going to take a look at the distributions of the movie genres in three plots, from year 2010 to 2013, 2014 to 2017, and 2018 to 2021 respectively.

```
# Create distribution plot for year 2010-2013
ggplot(merged_long_10_13, aes(x = release_year, y = count, fill = reorder(listed_in, count))) +
  geom_bar(stat = "identity",
           position = "dodge",
           color = "black") +
  scale_fill_viridis(discrete = TRUE) +
  labs(x = "Years",
       y = "Count of each movie genre",
       title = "The distributions of movie genres across years",
       subtitle = "From 2010 to 2013") +
  theme_classic() +
  theme(plot.margin = unit(c(0, 0, 0, 0), "cm"),
        plot.title.position = "plot") +
```

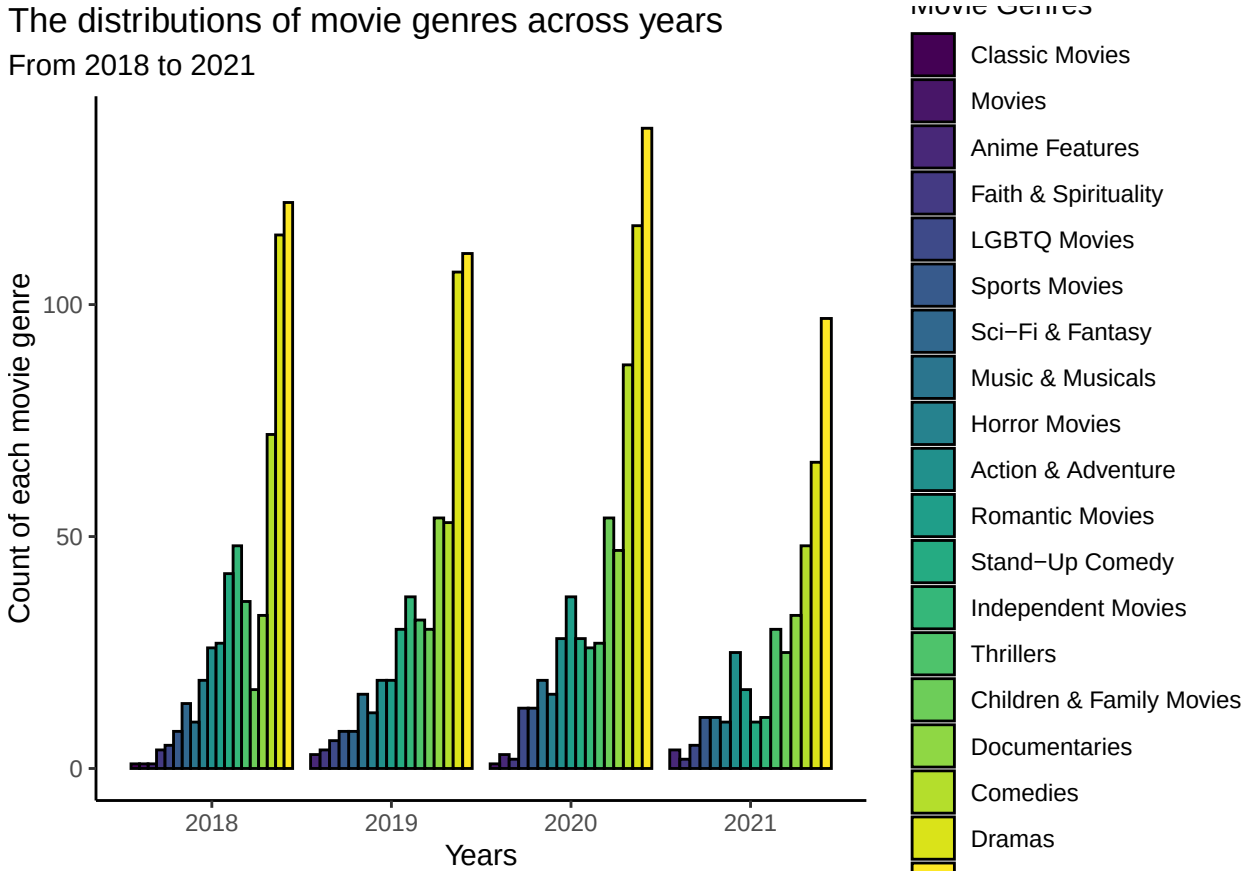
```
  guides(fill = guide_legend("Movie Genres"))
```




```
#create distribution plot for year 2014-2017
ggplot(merged_long_14_17, aes(x = release_year, y = count, fill = reorder(listed_in, count))) +
  geom_bar(stat = "identity",
           position = "dodge",
           color = "black") +
  scale_fill_viridis(discrete = TRUE) +
  labs(x = "Years",
       y = "Count of each movie genre",
       title = "The distributions of movie genres across years",
       subtile = "From 2014 to 2017") +
  theme_classic() +
  theme(plot.margin = unit(c(0, 0, 0, 0), "cm"),
        plot.title.position = "plot") +
  guides(fill = guide_legend("Movie Genres"))
```




```
#create distribution plot for year 2018-2021
ggplot(merged_long_18_21, aes(x = release_year, y = count, fill = reorder(listed_in, count))) +
  geom_bar(stat = "identity",
           position = "dodge",
           color = "black") +
  scale_fill_viridis(discrete = TRUE) +
  labs(x = "Years",
       y = "Count of each movie genre",
       title = "The distributions of movie genres across years",
       subtitle = "From 2018 to 2021") +
  theme_classic() +
  theme(plot.margin = unit(c(0, 0, 0, 0), "cm"),
        plot.title.position = "plot") +
  guides(fill = guide_legend("Movie Genres"))
```



From the plots we can clearly observe that "International Movies", "Dramas", and "Comedies" appeared the most often in most of the years from 2010 to 2021. Can this possibly indicate any trends about the public's movie tastes? Further researches are needed to gain more insights.

We found that the majority of the movies has several genres attached to it, so we wanted to simplify the genres to include only one category.

```
merged <- merged |>
  mutate(genre = case_when(
    grepl("Dramas", listed_in) ~ "Dramas",
    grepl("Comedies", listed_in) ~ "Comedies",
    grepl("Comedy", listed_in) ~ "Comedies",
    grepl("Documentaries", listed_in) ~ "Documentaries",
    grepl("Thrillers", listed_in) ~ "Thrillers",
    grepl("Action", listed_in) ~ "Action",
    grepl("Children", listed_in) ~ "Children",
    grepl("Romantic", listed_in) ~ "Romantic",
    grepl("Horror Movies", listed_in) ~ "Thrillers",
    grepl("Music", listed_in) ~ "Music",
    TRUE ~ NA))

merged |>
  filter(release_year >= 2010) |>
  group_by(genre) |>
  summarise(count = n()) |>
  arrange(desc(count))
```

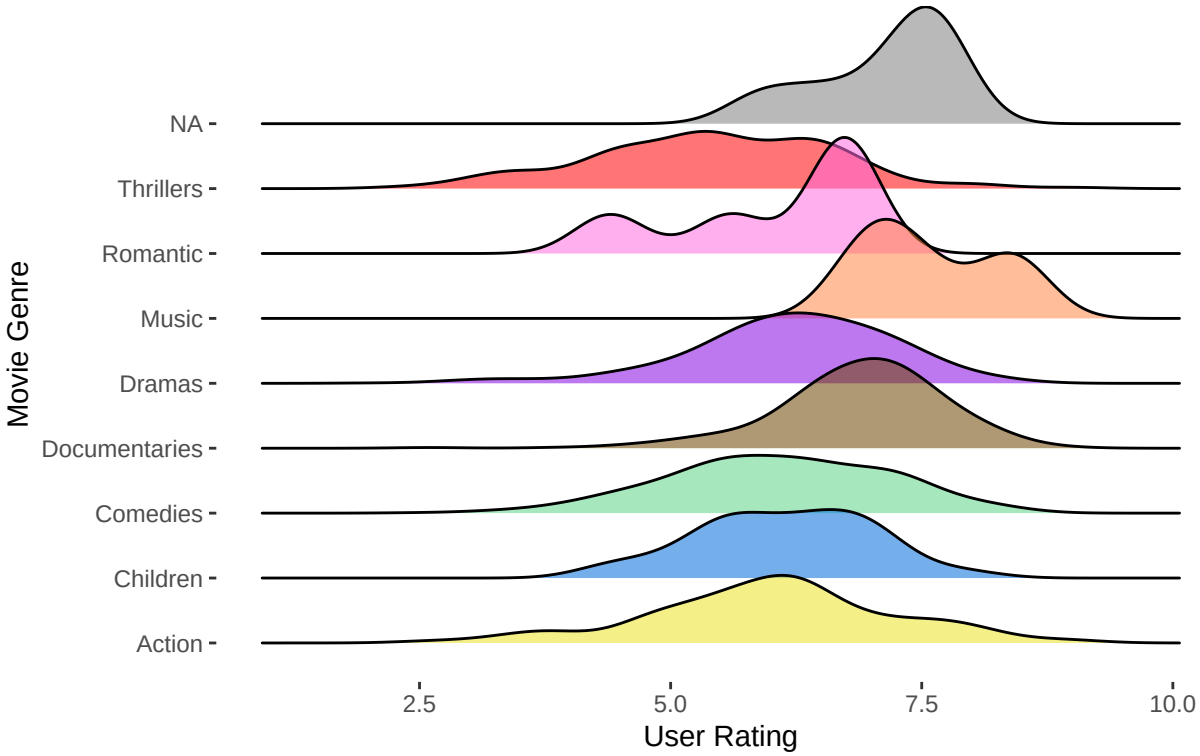
```
## # A tibble: 9 x 2
##   genre         count
##   <chr>         <int>
## 1 Dramas          636
## 2 Comedies        466
## 3 Documentaries   277
## 4 Thrillers       155
## 5 Action          108
## 6 Children         46
## 7 <NA>              7
## 8 Music             6
## 9 Romantic          5
```

After simplifying, we see that "Dramas", "Comedies", and "Documentaries" are now the most-frequently appeared genres since 2010.

```
merged |>
  ggplot(aes(x = rating, y = as.factor(genre), fill = as.factor(genre), alpha = 1)) +
  geom_density_ridges() +
  labs(title = "User ratings on IMDb in different genres",
       subtitle = "The distributions of ratings are different in different genres",
       x = "User Rating",
       y = "Movie Genre") +
   scale_fill_manual(values = c("#EBE224","#006EDD","#5ED389","#6E4400",
                                "#8908E2","#FF8747","#FF6CE0","#FF0000",
                                "#D09452")) +
  theme(legend.position = "none",
        plot.title.position = "plot",
        panel.grid.major.x = element_blank(),
        panel.grid.minor.x = element_blank(),
        panel.grid.major.y = element_blank(),
        panel.grid.minor.y = element_blank(),
        panel.background = element_blank())
```

```
## Picking joint bandwidth of 0.355
```

```
## Warning: Removed 20 rows containing non-finite values (`stat_density_ridges()`).
```



The plot shows different rating distributions across different movie genres. For example, musical movies seem to receive high ratings, while thrillers tend to receive low ratings. We will further explore this trend in the following Question 2.

Data Analysis

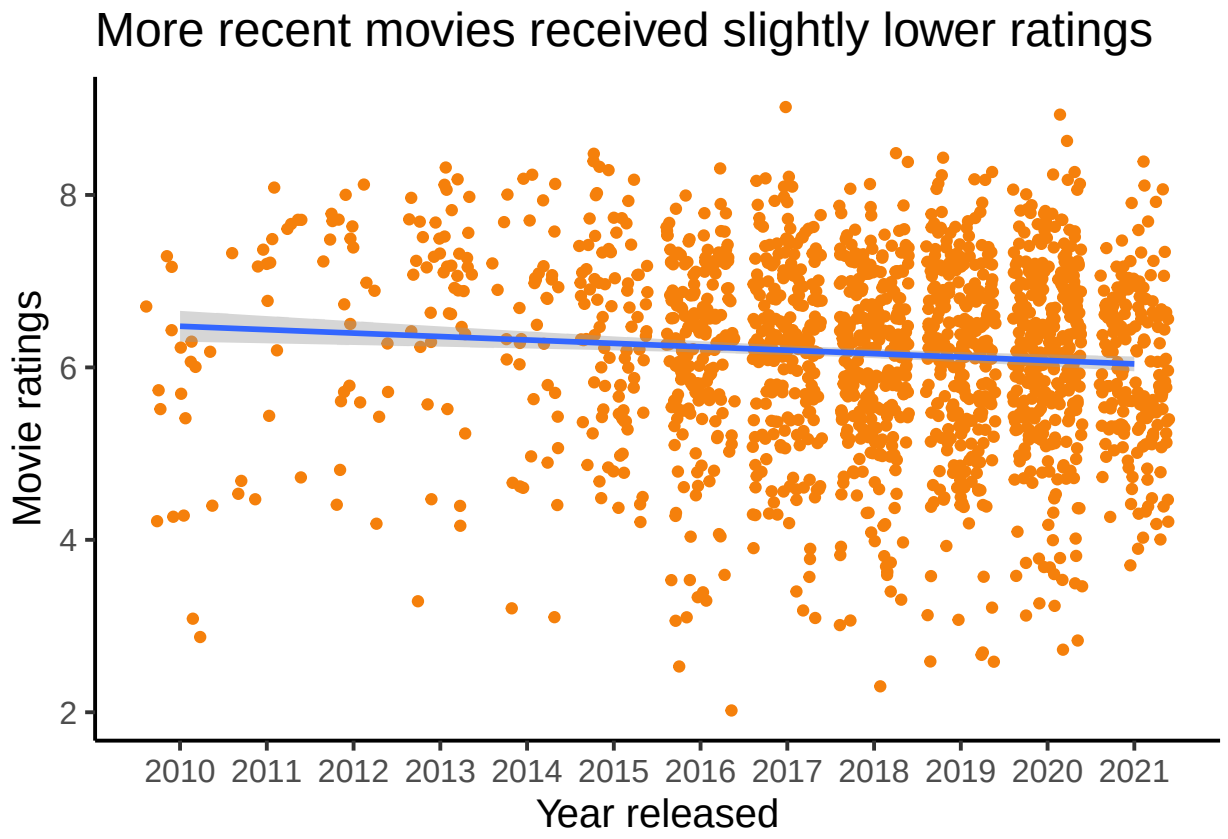
Question 1: Do people rate movies differently over the years since 2010? To find out if people give higher ratings over the years since 2010, let's take a look from the plot first to see if there is any clear trend.

```
merged |>
  filter(release_year >= 2010) |>
  ggplot(aes(x = release_year, y = rating)) +
  geom_jitter(color = "#F5800B") +
  geom_smooth(method = "lm") +
  labs(x = "Year released",
       y = "Movie ratings",
       title = "More recent movies received slightly lower ratings") +
  scale_x_continuous(breaks = seq(2010, 2021)) +
  theme(plot.title.position = "plot") +
  theme_classic(base_size = 15)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 18 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 18 rows containing missing values (`geom_point()`).
```



We see that movies released more recently tend to receive slightly lower ratings than movies released earlier. This might be caused by a change in the public taste of movies, but we are not sure about the exact reasons behind this. In addition, as the plot shows, the amount of data before 2015 is significantly less than the amount of data after 2015, so the decreasing trend we found might not be accurate. To obtain more accurate findings, we will build a model to see if such a decreasing trend exists.

```
# create a data frame that contains data since 2010
recent <- merged |>
  filter(release_year >= 2010)

# convert release year to factors
recent$release_year <- factor(recent$release_year)

# fit a linear model with release year
mod_yr <- linear_reg() |>
  set_engine("lm") |>
  fit(rating ~ release_year, data = recent)

mod_yr |>
  tidy()
```

```
## # A tibble: 12 x 5
##    term             estimate std.error statistic  p.value
##    <chr>               <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)         5.47      0.248     22.1  5.60e-95
##  2 release_year2011    1.16      0.355      3.28 1.06e- 3
##  3 release_year2012    1.08      0.332      3.25 1.18e- 3
```

```
##  4 release_year2013    1.38      0.295      4.68 3.10e- 6
##  5 release_year2014    0.873     0.300      2.91 3.63e- 3
##  6 release_year2015    0.937     0.273      3.43 6.17e- 4
##  7 release_year2016    0.696     0.260      2.67 7.61e- 3
##  8 release_year2017    0.736     0.258      2.85 4.39e- 3
##  9 release_year2018    0.602     0.256      2.35 1.88e- 2
## 10 release_year2019    0.672     0.256      2.62 8.81e- 3
## 11 release_year2020    0.680     0.255      2.66 7.77e- 3
## 12 release_year2021    0.511     0.260      1.96 4.96e- 2
```

```
# check R^2
glance(mod_yr)
```

```
## # A tibble: 1 x 12
##   r.squ~1 adj.r~2 sigma stati~3 p.value    df logLik   AIC   BIC devia~4 df.re~5
##     <dbl>   <dbl> <dbl>   <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>   <int>
## 1  0.0267  0.0203  1.08    4.18 3.98e-6    11 -2519. 5064. 5135.   1955.    1676
## # ... with 1 more variable: nobs <int>, and abbreviated variable names
## #   1: r.squared, 2: adj.r.squared, 3: statistic, 4: deviance, 5: df.residual
```

We can see from the model that, after 2010, where movies received an average rating of 5.47, people are indeed giving lower ratings, as we see the difference between each consecutive year and 2010 decreases, although not significant. People tend to give around 6.63 in 2011 and 6.54 in 2012, but only tend to give 6.15 in 2020 and 5.98 in 2021.

To evaluate the strength of the fit of this model, we also checked the R^2 value, which turns out to be 0.027. Therefore, roughly only 2.7% of the variability in movie ratings can be explained by their release year. This suggests the year a movie is released might not be a strong and appropriate predictor for its rating.

Question 2: Do movies receive lower/higher ratings due to their genres? To answer this question, we first create a linear model of rating and genre.

```
# fit a linear model
mod_gr <- linear_reg() |>
  set_engine("lm") |>
  fit(rating ~ genre, data = recent)

mod_gr |>
  tidy()
```

```
## # A tibble: 8 x 5
##   term               estimate std.error statistic  p.value
##   <chr>                 <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)           5.91     0.0988    59.8   0
## 2 genreChildren         0.215    0.180      1.20  2.32e- 1
## 3 genreComedies         0.104    0.110      0.946 3.44e- 1
## 4 genreDocumentaries    0.979    0.116      8.42  8.16e-17
## 5 genreDramas           0.260    0.107      2.44  1.48e- 2
## 6 genreMusic            1.67     0.427      3.92  9.30e- 5
## 7 genreRomantic         0.129    0.465      0.276 7.82e- 1
## 8 genreThrillers       -0.538    0.129     -4.18  3.09e- 5
```

```
# check R^2
glance(mod_gr)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squa~1 sigma stati~2  p.value    df logLik   AIC   BIC devia~3
```

```
##       <dbl>        <dbl> <dbl>   <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>
## 1     0.134        0.131  1.02    37.1 1.68e-48     7 -2410. 4837. 4886.   1730.
## # ... with 2 more variables: df.residual <int>, nobs <int>, and abbreviated
## #   variable names 1: adj.r.squared, 2: statistic, 3: deviance
```

We see that after 2010 action movies, the baseline genre, receive an average rating of 5.91. Musical movies receive an average rating of 7.58, which is the highest rating among all genres after 2010, while thrillers only receive an average rating of 5.37, which is the lowest rating among all genres after 2010. As for the most numerous movies in the dataset, documentaries receive an average rating of 6.89, dramas movies receive an average rating of 6.17, and comedies receive an average rating of 6.01. It is somewhat expected that thrillers receive low ratings, since most thrillers are of low quality.

The R^2 of this model is 0.13. Therefore, roughly 13% of the variability in movie ratings can be explained by their genre. This R^2 value is greater than that of the year model, so the genre might a better predictor for its rating than the year a movie is released in.

We also wanted to add genre to the model created in Question 1 to see how it affects the rating.

```
mod_full <- linear_reg() |>
  set_engine("lm") |>
  fit(rating ~ release_year + genre, data = recent)

mod_full |>
  tidy()
```

```
## # A tibble: 19 x 5
##    term               estimate std.error statistic  p.value
##    <chr>                 <dbl>     <dbl>     <dbl>    <dbl>
##  1 (Intercept)           5.41      0.248    21.9   2.43e-93
##  2 release_year2011      0.926     0.332     2.79  5.32e- 3
##  3 release_year2012      0.945     0.309     3.05  2.30e- 3
##  4 release_year2013      1.21      0.277     4.38  1.29e- 5
##  5 release_year2014      0.545     0.281     1.94  5.23e- 2
##  6 release_year2015      0.593     0.256     2.31  2.08e- 2
##  7 release_year2016      0.366     0.244     1.50  1.34e- 1
##  8 release_year2017      0.456     0.242     1.89  5.95e- 2
##  9 release_year2018      0.345     0.240     1.44  1.50e- 1
## 10 release_year2019      0.356     0.241     1.48  1.39e- 1
## 11 release_year2020      0.374     0.240     1.56  1.19e- 1
## 12 release_year2021      0.227     0.244     0.928 3.54e- 1
## 13 genreChildren         0.349     0.180     1.94  5.23e- 2
## 14 genreComedies         0.188     0.110     1.71  8.74e- 2
## 15 genreDocumentaries    1.07      0.117     9.19  1.14e-19
## 16 genreDramas           0.356     0.107     3.31  9.53e- 4
## 17 genreMusic            1.79      0.423     4.24  2.35e- 5
## 18 genreRomantic         0.317     0.462     0.686 4.93e- 1
## 19 genreThrillers       -0.437     0.129    -3.39  7.13e- 4
```

```
# check R^2
glance(mod_full)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squa~1 sigma stati~2  p.value    df logLik   AIC   BIC devia~3
##       <dbl>        <dbl> <dbl>   <dbl>    <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>
## 1     0.162        0.153  1.00    17.9 4.14e-52    18 -2382. 4804. 4913.   1675.
## # ... with 2 more variables: df.residual <int>, nobs <int>, and abbreviated
## #   variable names 1: adj.r.squared, 2: statistic, 3: deviance
```

The model shows that, after 2010, when action movies received an average rating of 5.42, people give lower ratings across the years. People tend to give a rating of around 6.35 in 2011 and 6.36 in 2012, but only tend to give a rating of 5.79 in 2020 and 5.65 in 2021. Again, we see the same results that musicals receive an average rating of 7.21, which makes it the highest-rated genre, while thrillers only receive an average rating of 4.98, which makes it the lowest-rated genre. As for the most numerous genres: dramas receive a rating of 5.77 on average, comedies receive a rating of 5.61, and documentaries receive a rating of 6.49.

The adjusted R^2 of this model is 0.15. Therefore, roughly 15% of the variability in movie ratings can be explained by their genre and release year.

Question 3: Does a movie released during a holiday season receive a higher/lower rating than a movie released during other times of the year? We start by calculating the average ratings of movies released during the holiday season, non-holiday season, and all months.

We define the holiday season to be from Thanksgiving to Christmas, which is from November to December.

```
recent |>
  filter(month == 11 | month == 12) |>
  summarise(`mean rating` = mean(rating, na.rm = TRUE))
```

```
##   mean rating
## 1     6.18172
```

The mean ratings of movies released in holiday seasons is 6.182.

```
recent |>
  filter(month != 11 & month != 12) |>
  summarise(`mean rating` = mean(rating, na.rm = TRUE))
```

```
##   mean rating
## 1    6.161604
```

The mean ratings of movies released in non-holiday seasons is 6.162, which is slightly lower than that of movies released during the holiday season.

```
mean(recent$rating, na.rm = TRUE)
```

```
## [1] 6.164929
```

The mean ratings of all movies is 6.165, which is also slightly lower than that of movies released during the holiday season.

To summarize, the average rating is slightly higher for the movies released during the holiday season, compared to movies released in non-holiday seasons and all movies.

We also attempt to fit a linear model to further support our findings.

```
# convert months to factors
recent$month <- factor(recent$month)

# create linear model
mod_mth <- linear_reg() |>
  set_engine("lm") |>
  fit(rating ~ month, data = recent)

tidy(mod_mth)
```

```
## # A tibble: 12 x 5
##    term        estimate std.error statistic p.value
##    <chr>          <dbl>     <dbl>     <dbl>   <dbl>
```

```
##  1 (Intercept)   6.20       0.101    61.3    0
##  2 month2        0.0370     0.141     0.263  0.792
##  3 month3        0.0852     0.134     0.634  0.526
##  4 month4       -0.204      0.137    -1.49   0.136
##  5 month5       -0.0755     0.137    -0.551  0.582
##  6 month6       -0.224      0.134    -1.68   0.0938
##  7 month7        0.0362     0.134     0.270  0.787
##  8 month8       -0.154      0.139    -1.11   0.268
##  9 month9       -0.0319     0.133    -0.239  0.811
## 10 month10       0.142      0.136     1.05   0.295
## 11 month11       0.0755     0.139     0.544  0.587
## 12 month12      -0.103      0.135    -0.762  0.446
```

```
# check R^2
glance(mod_mth)
```

```
## # A tibble: 1 x 12
##   r.squ~1 adj.r~2 sigma stati~3 p.value    df logLik   AIC   BIC devia~4 df.re~5
##     <dbl>   <dbl> <dbl>   <dbl>   <dbl> <dbl>  <dbl> <dbl> <dbl>   <dbl>   <int>
## 1  0.0110 0.00453  1.09    1.70  0.0683    11 -2532. 5091. 5162.   1986.    1676
## # ... with 1 more variable: nobs <int>, and abbreviated variable names
## #   1: r.squared, 2: adj.r.squared, 3: statistic, 4: deviance, 5: df.residual
```

The table shows, for example, the average movie ratings in January is 6.2 and the average rating is 6.34 during October, while the average movie rating in November is 6.28, and the average rating in December is 6.1. The adjusted R^2 of this model is 0.011, which means roughly only 1.1% of the variability in movie ratings can be explained by their release month.

This indicates movie ratings in November and December might not be significantly different from other months in a year, but we cannot be sure to draw this conclusion due to the weak model. Therefore, we will conduct an ANOVA test below to check the difference. Our null hypothesis is that there is no significant difference between the ratings for movies released during holiday seasons and non-holiday seasons. We will use the typical threshold of 0.05 for the alpha value.

```
# create a column that discriminates between holiday and non-holiday seasons and convert it to factors
recent <- recent |>
  mutate(is_nov_dec = ifelse(month %in% c(11, 12), "Nov_Dec", "Other_Months"),
         is_nov_dec = factor(is_nov_dec))

# create an ANOVA model
anova_model <- aov(rating ~ is_nov_dec, data = recent)

tidy(anova_model)
```

```
## # A tibble: 2 x 6
##   term          df     sumsq meansq statistic p.value
##   <chr>      <dbl>     <dbl>  <dbl>     <dbl>   <dbl>
## 1 is_nov_dec     1    0.0942 0.0942    0.0791   0.779
## 2 Residuals   1686 2008.     1.19     NA       NA
```

Since the p-value of 0.78 is larger than 0.05, we fail to reject the null hypothesis. Consequently, there is not a difference between the ratings for movies released during the holiday season and other months.

Results

- Q1

We found that people give slightly lower movies ratings over the years since 2010. However, the trend might not be significant since the linear model is weak.

Therefore, we propose that the lack of significance is hardly due to shifts in user preferences or user behavior. In fact, the weak result might be due to an increased number of movies produced over the years, which caused more variation in ratings that is difficult to generalize.

- Q2

We found that movies of different genres indeed receive different ratings. For example, musical movies receive an average rating of 7.58, which is the highest-rated genre, while thrillers only receive an average rating of 5.37, which is the lowest-rated genre.

The model combining release year and genre produced a stronger support for our findings in Question 1 and Question 2.

This finding is particularly helpful as a user implication. Specifically, when making decisions among cross-category movies, direct comparison of numeric scores is less reliable or meaningful. Users should consider other movies in the same genre in order to correctly infer the reputation of said movies.

- Q3

As an extension to Q1, we looked at ratings while condtioning the presence of major holidays. We found no significant difference between ratings for movies released during the holiday season compared to other months of the year from our linear model and ANOVA test.

Limitation

Several limitations should be considered in this study. The data used in this project is highly skewed after 2015, which limits our ability to gain insights into the trend across the entire timeline. Consequently, the findings of this study may not be representative of the entire population of movies. Our future steps could explore additional data sources or use different sampling methods to mitigate this limitation. Further, while this study explores several factors that contribute to user ratings of movies, there could be unexplored factors that might potentially be more predictive of these ratings. For example, sales data and maturity rating could influence user ratings but were not included in this analysis. Finally, due to the limitation of available data, we were not able to confirm the submission time of the ratings. Instead, we based our analysis on the release date of the movies. This may have introduced some inaccuracies in our findings as user ratings may have been submitted at different times after the movie was released. Exploring alternative methods for estimating the timing of user ratings, such as using data from social media or other online platforms, will benefit the accuracy of the results.

Conclusion

In conclusion, our investigation into IMDb reviews and ratings has revealed several key insights into the reliability and factors affecting movie ratings. We found that there is a slight decline in movie ratings over the years since 2010, suggesting that user preferences and expectations might have shifted over time. Moreover, we discovered that different genres of movies receive varying ratings, with musicals enjoying the highest average rating and thrillers receiving the lowest. However, we did not find a significant difference in ratings for movies released during the holiday season compared to other months of the year.

These findings highlight the importance of taking a nuanced approach when interpreting online movie reviews and ratings. Users should be aware of the potential subjectivity and external factors that could influence ratings and not solely rely on them for making decisions on which movies to watch. By understanding the trends and variations in user ratings, moviegoers can make more informed choices and find content that best matches their preferences and interests. Ultimately, a deeper understanding of online movie reviews and ratings can lead to better decision-making processes for viewers and a more enjoyable movie-watching experience.