Final Project

Foundations of Data Science

Avyya_Singh

Tuesday, April 16, 2024

Section 1: Introduction

- **Data:** Disney Movies (Uncleaned), **Source:** Kaggle, **Format:** CSV
- **URL:** https://www.kaggle.com/datasets/adityamaurya123321/disney-picturesuncleaned

The Disney Films dataset offers a comprehensive collection of data on various Disney movies, providing detailed information ranging from directors and cast members to production details, release dates, budgets, and box office earnings. With a focus on data cleaning, this dataset presents an ideal opportunity for enthusiasts to delve into the specifics of Disney's filmography. Whether you're a data aficionado, a cinema enthusiast, or intrigued by the business aspects of filmmaking, this dataset serves as a valuable resource for analysis and learning. By exploring the numbers, patterns, and trends within Disney's extensive catalog, users can enhance their data analysis skills while uncovering fascinating insights into the iconic movies that have shaped generations of audiences worldwide.

What questions/hypotheses will you be investigating?

- Plot the budget and box office revenue against each other to see if there's a correlation between the budget invested in a movie and its revenue.
- Which company has the most average box office revenue?
- Top 10 Directors with most movies.
- Box office and budget comparison

Section 2: Data Wrangling Plan

Iteration 1

Phase 1

1. Read the csv file into R

2. Make column names lowercase
3. Determine if the data is Tidy and if not fix it
4. Identify uids
5. Rename the uid's column and other columns to add _ for more clearity.

Phase 2

```
## 1.
filedata <- read_csv("disney_movies.csv",
                     show_col_types = FALSE
) |>
  ## 2.
  rename_with(tolower)
#> New names:
#> * `` -> `...1`

#check
filedata |> head(5)
#> # A tibble: 5 x 24
#>    ...1 movie_name    `directed by` `story by` `based on` `produced by` starring
#>   <dbl> <chr>         <chr>         <chr>      <chr>      <chr>         <chr>
#> 1     0 Snow White a~ ['David Hand~ "['Ted Se~ ['Snow Wh~ ['Walt Disne~ ['Adria~
#> 2     1 Pinocchio (1~ ['Ben Sharps~ "['Ted Se~ ['The Adv~ ['Walt Disne~ ['Cliff~
#> 3     2 Fantasia (19~ ['James Alga~ "['Joe Gr~ <NA>       ['Walt Disne~ ['Leopo~
#> 4     3 The Reluctan~ ['Alfred Wer~  <NA>      <NA>       ['Walt Disne~ ['Rober~
#> 5     4 Dumbo         ['Ben Sharps~ "['Joe Gr~ ['Helen A~ ['Walt Disne~ ['Edwar~
#> # i 17 more variables: `music by` <chr>, productioncompany <chr>,
#> #   `distributed by` <chr>, `release dates` <chr>, `running time` <chr>,
#> #   country <chr>, language <chr>, budget <chr>, `box office` <chr>,
#> #   `narrated by` <chr>, cinematography <chr>, `release date` <chr>,
#> #   `written by` <chr>, `edited by` <chr>, `screenplay by` <chr>,
#> #   countries <chr>, productioncompanies <chr>
```

```
## 3.
filedata %>% count (...1, movie_name, `directed by`, `story by`,
                    `produced by`, `produced by`, starring, `music by`,
                    `productioncompany`, `distributed by`, `release dates`,
                    `running time`, country, language, budget, `box office`,
                    `narrated by`, cinematography, `release date`, `written by`,
                    `edited by`, `screenplay by`, countries,
                    productioncompanies) %>% filter(n > 1)
#> # A tibble: 0 x 24
#> # i 24 variables: ...1 <dbl>, movie_name <chr>, directed by <chr>,
#> #   story by <chr>, produced by <chr>, starring <chr>, music by <chr>,
#> #   productioncompany <chr>, distributed by <chr>, release dates <chr>,
#> #   running time <chr>, country <chr>, language <chr>, budget <chr>,
#> #   box office <chr>, narrated by <chr>, cinematography <chr>,
#> #   release date <chr>, written by <chr>, edited by <chr>, screenplay by <chr>,
#> #   countries <chr>, productioncompanies <chr>, n <int>
```

The data is Tidy One value per cell, one variable per column and one observation per row.

```
## 4.
filedata %>% count (...1, movie_name
                    ) %>% filter(n > 1)
#> # A tibble: 0 x 3
#> # i 3 variables: ...1 <dbl>, movie_name <chr>, n <int>
```

We can use `...1` and `movie_name` can be used as unique id's.

```
filedata <- filedata %>%
  rename(movie_id = 1)
filedata
#> # A tibble: 525 x 24
#>    movie_id movie_name         `directed by` `story by` `based on` `produced by`
#>       <dbl> <chr>              <chr>         <chr>      <chr>      <chr>
#>  1        0 Snow White and th~ ['David Hand~ "['Ted Se~ ['Snow Wh~ ['Walt Disne~
#>  2        1 Pinocchio (1940 f~ ['Ben Sharps~ "['Ted Se~ ['The Adv~ ['Walt Disne~
#>  3        2 Fantasia (1940 fi~ ['James Alga~ "['Joe Gr~ <NA>       ['Walt Disne~
#>  4        3 The Reluctant Dra~ ['Alfred Wer~  <NA>      <NA>       ['Walt Disne~
#>  5        4 Dumbo              ['Ben Sharps~ "['Joe Gr~ ['Helen A~ ['Walt Disne~
#>  6        5 Bambi              ['David Hand~ "['Perce ~ ['Bambi, ~ ['Walt Disne~
#>  7        6 Saludos Amigos     ['Norman Fer~ "['Homer ~ <NA>       ['Walt Disne~
#>  8        7 Victory Through A~ ['James Alga~  <NA>      ['Victory~ ['Walt Disne~
#>  9        8 The Three Caballe~ ['Norman Fer~ "['Ted Se~ <NA>       ['Walt Disne~
#> 10        9 Make Mine Music    ['Jack Kinne~ "['Homer ~ ['Casey a~ ['Walt Disne~
#> # i 515 more rows
#> # i 18 more variables: starring <chr>, `music by` <chr>,
#> #   productioncompany <chr>, `distributed by` <chr>, `release dates` <chr>,
#> #   `running time` <chr>, country <chr>, language <chr>, budget <chr>,
#> #   `box office` <chr>, `narrated by` <chr>, cinematography <chr>,
#> #   `release date` <chr>, `written by` <chr>, `edited by` <chr>,
#> #   `screenplay by` <chr>, countries <chr>, productioncompanies <chr>
```

```r
## 5.
names(filedata) <- c(
  "movie_id", "movie_name", "directed_by", "story_by", "based_on", "produced_by",
  "starring", "music_by", "production_company", "distributed_by", "release_dates",
  "running_time", "country","language", "budget", "box_office", "narrated_by",
  "cinematography", "release_date", "written_by", "edited_by","screenplay_by",
  "countries","production_companies")
# Check
filedata |> head(5)
#> # A tibble: 5 x 24
#>   movie_id movie_name         directed_by story_by based_on produced_by starring
#>      <dbl> <chr>              <chr>       <chr>    <chr>    <chr>       <chr>
#> 1        0 Snow White and th~ ['David Ha~ "['Ted ~ ['Snow ~ ['Walt Dis~ ['Adria~
#> 2        1 Pinocchio (1940 f~ ['Ben Shar~ "['Ted ~ ['The A~ ['Walt Dis~ ['Cliff~
#> 3        2 Fantasia (1940 fi~ ['James Al~ "['Joe ~ <NA>     ['Walt Dis~ ['Leopo~
#> 4        3 The Reluctant Dra~ ['Alfred W~  <NA>    <NA>     ['Walt Dis~ ['Rober~
#> 5        4 Dumbo              ['Ben Shar~ "['Joe ~ ['Helen~ ['Walt Dis~ ['Edwar~
#> # i 17 more variables: music_by <chr>, production_company <chr>,
#> #   distributed_by <chr>, release_dates <chr>, running_time <chr>,
#> #   country <chr>, language <chr>, budget <chr>, box_office <chr>,
#> #   narrated_by <chr>, cinematography <chr>, release_date <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>, countries <chr>,
#> #   production_companies <chr>
```

Renaming the all the columns for more clarity. So, `movie_id` and `movie_name` are unique id's.

Iteration 2

Phase 1

1. Check the values in all the columns

    a. removed all the unnecessary symbols and letters like `\[|\]|'|"|\\n` in the columns.

    b. cleaning `running_time`, `country` and `budget` columns as they've some unnecessary values and numbers which give misleading information.

    c. cleaning the `box_office` column by removing any characters that are not digits or periods.

2. Dropping unusual and unnecessary columns from table as they're misleading.
3. Dropping NA values from the table.

Phase 2

```r
## 1a.
cleaned_data <- filedata %>%
  mutate_all(~ str_replace_all(., "\\[|\\]|'|\"|\\\\n", ""))

# Check the cleaned data
cleaned_data |> head(5)
#> # A tibble: 5 x 24
#>   movie_id movie_name         directed_by story_by based_on produced_by starring
#>   <chr>    <chr>              <chr>       <chr>    <chr>    <chr>       <chr>
#> 1 0        Snow White and th~ David Hand~ Ted Sea~ Snow Wh~ Walt Disney Adriana~
#> 2 1        Pinocchio (1940 f~ Ben Sharps~ Ted Sea~ The Adv~ Walt Disney Cliff E~
#> 3 2        Fantasia (1940 fi~ James Alga~ Joe Gra~ <NA>     Walt Disney Leopold~
#> 4 3        The Reluctant Dra~ Alfred Wer~ <NA>     <NA>     Walt Disney Robert ~
#> 5 4        Dumbo              Ben Sharps~ Joe Gra~ Helen A~ Walt Disney Edward ~
#> # i 17 more variables: music_by <chr>, production_company <chr>,
#> #   distributed_by <chr>, release_dates <chr>, running_time <chr>,
#> #   country <chr>, language <chr>, budget <chr>, box_office <chr>,
#> #   narrated_by <chr>, cinematography <chr>, release_date <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>, countries <chr>,
#> #   production_companies <chr>
```

```r
## 1b.
cleaned_data2 <- cleaned_data %>%
  mutate(running_time = str_extract(running_time, "\\d+") %>%
           paste("minutes", sep = ""))

cleaned_data3 <- cleaned_data2 %>%
  mutate(country = str_replace_all(country, "[^a-zA-Z ]", ""),
         language = str_replace_all(language, "[^a-zA-Z ]", ""))

cleaned_data4 <- cleaned_data3 %>%
  mutate(budget = ifelse(!is.na(budget),
                         str_extract(budget, "\\d+") %>%
                           paste("million", sep = ""), NA))

cleaned_data4 |> head(5)
#> # A tibble: 5 x 24
#>   movie_id movie_name         directed_by story_by based_on produced_by starring
#>   <chr>    <chr>              <chr>       <chr>    <chr>    <chr>       <chr>
#> 1 0        Snow White and th~ David Hand~ Ted Sea~ Snow Wh~ Walt Disney Adriana~
#> 2 1        Pinocchio (1940 f~ Ben Sharps~ Ted Sea~ The Adv~ Walt Disney Cliff E~
#> 3 2        Fantasia (1940 fi~ James Alga~ Joe Gra~ <NA>     Walt Disney Leopold~
#> 4 3        The Reluctant Dra~ Alfred Wer~ <NA>     <NA>     Walt Disney Robert ~
#> 5 4        Dumbo              Ben Sharps~ Joe Gra~ Helen A~ Walt Disney Edward ~
#> # i 17 more variables: music_by <chr>, production_company <chr>,
#> #   distributed_by <chr>, release_dates <chr>, running_time <chr>,
#> #   country <chr>, language <chr>, budget <chr>, box_office <chr>,
#> #   narrated_by <chr>, cinematography <chr>, release_date <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>, countries <chr>,
#> #   production_companies <chr>
```

It first converts the **box_office** column to a character type, then uses regular expressions to replace any non-digit or non-period characters with an empty string. Finally, it converts the cleaned **box_office** column back to a numeric type.

```
## 1c.
cleaned_data4$box_office <- as.character(cleaned_data4$box_office)
cleaned_data4$box_office <- str_replace_all(cleaned_data4$box_office, "[^0-9.]", "")

cleaned_data4$box_office <- as.numeric(cleaned_data4$box_office)
#> Warning: NAs introduced by coercion
cleaned_data4 |> head(5)
#> # A tibble: 5 x 24
#>   movie_id movie_name         directed_by story_by based_on produced_by starring
#>   <chr>    <chr>              <chr>       <chr>    <chr>    <chr>       <chr>
#> 1 0        Snow White and th~ David Hand~ Ted Sea~ Snow Wh~ Walt Disney Adriana~
#> 2 1        Pinocchio (1940 f~ Ben Sharps~ Ted Sea~ The Adv~ Walt Disney Cliff E~
#> 3 2        Fantasia (1940 fi~ James Alga~ Joe Gra~ <NA>     Walt Disney Leopold~
#> 4 3        The Reluctant Dra~ Alfred Wer~ <NA>     <NA>     Walt Disney Robert ~
#> 5 4        Dumbo              Ben Sharps~ Joe Gra~ Helen A~ Walt Disney Edward ~
#> # i 17 more variables: music_by <chr>, production_company <chr>,
#> #   distributed_by <chr>, release_dates <chr>, running_time <chr>,
#> #   country <chr>, language <chr>, budget <chr>, box_office <dbl>,
#> #   narrated_by <chr>, cinematography <chr>, release_date <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>, countries <chr>,
#> #   production_companies <chr>
```

To ensure clarity and aesthetic appeal in the data representation, I'm removing the **release_date**, **countries**, **production_companies**, and **release_dates** columns due to extensive data inconsistencies and high levels of missing values, which could potentially mislead the analysis.

```
## 2.
final_data <- cleaned_data4 %>%
  select(-release_date, -countries,
         -production_companies, -release_dates)
# Check
final_data |> head(5)
#> # A tibble: 5 x 20
#>   movie_id movie_name         directed_by story_by based_on produced_by starring
#>   <chr>    <chr>              <chr>       <chr>    <chr>    <chr>       <chr>
#> 1 0        Snow White and th~ David Hand~ Ted Sea~ Snow Wh~ Walt Disney Adriana~
#> 2 1        Pinocchio (1940 f~ Ben Sharps~ Ted Sea~ The Adv~ Walt Disney Cliff E~
#> 3 2        Fantasia (1940 fi~ James Alga~ Joe Gra~ <NA>     Walt Disney Leopold~
#> 4 3        The Reluctant Dra~ Alfred Wer~ <NA>     <NA>     Walt Disney Robert ~
#> 5 4        Dumbo              Ben Sharps~ Joe Gra~ Helen A~ Walt Disney Edward ~
#> # i 13 more variables: music_by <chr>, production_company <chr>,
#> #   distributed_by <chr>, running_time <chr>, country <chr>, language <chr>,
#> #   budget <chr>, box_office <dbl>, narrated_by <chr>, cinematography <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>
```

I'm dropping NA values here as an illustrative example. However, for my data analysis plots, I won't be considering NA values since nearly all of my data contains NA values. Consequently, removing NA values results in an empty dataset with 0 rows.

```
## 3.
clean_nona <- final_data |> drop_na()
clean_nona
#> # A tibble: 0 x 20
#> # i 20 variables: movie_id <chr>, movie_name <chr>, directed_by <chr>,
#> #   story_by <chr>, based_on <chr>, produced_by <chr>, starring <chr>,
#> #   music_by <chr>, production_company <chr>, distributed_by <chr>,
#> #   running_time <chr>, country <chr>, language <chr>, budget <chr>,
#> #   box_office <dbl>, narrated_by <chr>, cinematography <chr>,
#> #   written_by <chr>, edited_by <chr>, screenplay_by <chr>
```

Section 3 & 4: Results & Visualizations with Answers

Ques 1. Plot the budget and box office revenue against each other to see if there's a correlation between the budget invested in a movie and its revenue.

The plot illustrates the relationship between movie budgets and box office revenues. Most data points cluster around lower budgets, while a few outliers indicate movies with exceptionally high revenues. The horizontal red line suggests a weak correlation between budget and revenue.

However, other factors like marketing and star power also influence box office success. Further analysis, such as correlation coefficients, can provide deeper insights. Overall, while higher budgets may correlate with higher revenues, success at the box office is influenced by various factors beyond just budget size.

In conclusion, while there seems to be a general trend of higher box office revenues for movies with higher budgets, the relationship is not straightforward, and various factors contribute to a movie's financial success at the box office.

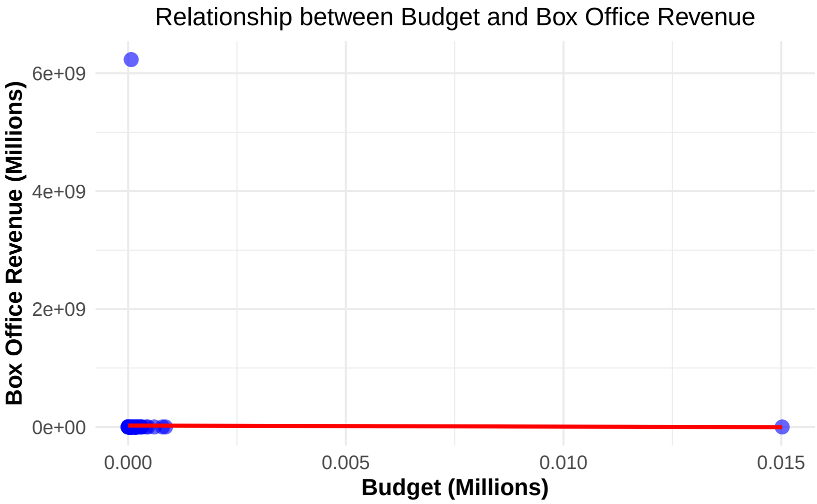
```r
final_data <- final_data %>%
  filter(!is.na(budget) & grepl("^[0-9.]+million$", budget, ignore.case = TRUE))

final_data <- final_data %>%
  mutate(budget_millions = as.numeric(gsub("million", "", budget, ignore.case = TRUE))/1e6
         box_office_millions = box_office / 1e6)

final_data <- final_data %>%
  filter(!is.na(box_office))

plot <- ggplot(final_data, aes(x = budget_millions, y = box_office_millions)) +
  geom_point(color = "blue", alpha = 0.6, size = 3) +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(x = "Budget (Millions)", y = "Box Office Revenue (Millions)",
       title = "Relationship between Budget and Box Office Revenue") +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5),
        axis.text = element_text(size = 10),
        axis.title = element_text(size = 12, face = "bold"))

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



Ques 2. Which company has the most average box office revenue?

In examining the dataset detailing the average box office revenues of various distribution companies, it's evident that Disney+ stands out with a staggering average box office revenue of $6,231,905,262,319,052. This astronomical figure underscores Disney+'s unparalleled dominance in the entertainment industry.

However, for the purpose of our comparative analysis, we opt to focus on the next most significant players in the field. Among these are Hoyts Distribution, Walt Disney Studios-Motion Pictures, and Walt Disney StudiosHome Entertainment, Walt Disney StudiosMotion Pictures.

Upon closer examination, it becomes apparent that Walt Disney StudiosHome Entertainment, Walt Disney StudiosMotion Pictures emerges as the top contender among these three, boasting an impressive average box office revenue of $54,363,795. This substantial figure underscores the formidable presence and enduring success of Disney in the realm of film distribution.

In summary, while Disney+ reigns supreme in terms of sheer box office revenue, our comparative analysis highlights Walt Disney StudiosHome Entertainment, Walt Disney StudiosMotion Pictures as the standout performer among the select distribution companies under consideration.

```
distribution_companies <- c("RKO Radio Pictures", "United Artists", "Buena Vista Distribut
                            "Hoyts Distribution", "Walt Disney", "Gaumont, Lionsgate, Univ
                            "Pathé Distribution", "Disney+")

filtered_data <- final_data %>%
  filter(str_detect(distributed_by, paste(distribution_companies, collapse = "|")))
```

```
distribution_comparison <- filtered_data %>%
  group_by(distributed_by) %>%
  summarise(avg_box_office = mean(box_office, na.rm = TRUE)) %>%
  arrange(desc(avg_box_office))

distribution_top_10 <- head(distribution_comparison, 8)
table_gt <- distribution_top_10 %>%
  gt() %>%
  tab_header(
    title = "Top 8 Distribution Companies by Average Box Office Revenue"
  ) %>%
  fmt_number(
    columns = c(avg_box_office),
    decimals = 8
  )
table_gt
```

Top 8 Distribution Companies by Average Box Office Revenue

distributed_by	avg_box_
Disney+	6,231,905,262,319,052.000
Walt Disney StudiosHome Entertainment, Walt Disney StudiosMotion Pictures	54,363,795.265
Walt Disney StudiosMotion Pictures	51,157,740.675
Hoyts Distribution	13,687,027.000
Buena Vista Distribution	1,876,732.294
RKO Radio Pictures	1,067,628.964
United Artists	799,000.000
Walt Disney StudiosMotion Pictures, Walt Disney StudiosMotion Picturesa	3,840.000

So, we compare the three companies Walt Disney StudiosHome Entertainment, Walt Disney StudiosMotion Pictures has most.

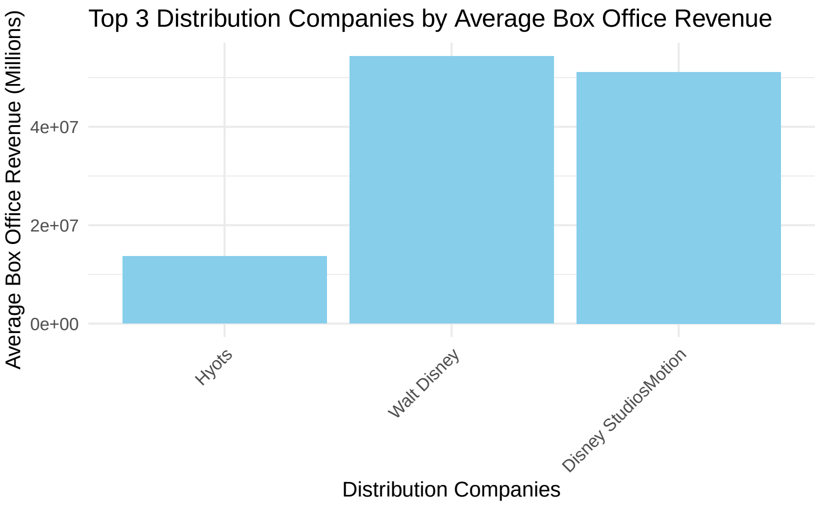
```
distribution_companies2 <- c("Hoyts Distribution", "Walt Disney")

filtered_data <- final_data %>%
  filter(str_detect(distributed_by, paste(distribution_companies2, collapse = "|")))

distribution_comparison2 <- filtered_data %>%
  group_by(distributed_by) %>%
  summarise(avg_box_office = mean(box_office, na.rm = TRUE)) %>%
  arrange(desc(avg_box_office))

distribution_top_3 <- head(distribution_comparison2, 3)
distribution_top_3$avg_box_office_million <- distribution_top_3$avg_box_office / 1e6
distribution_top_3
#> # A tibble: 3 x 3
#>   distributed_by                           avg_box_office avg_box_office_million
#>   <chr>                                             <dbl>                  <dbl>
#> 1 Walt Disney StudiosHome Entertainment, ~      54363795.                   54.4
#> 2 Walt Disney StudiosMotion Pictures            51157741.                   51.2
#> 3 Hoyts Distribution                            13687027                    13.7

ggplot(distribution_top_3, aes(x = distributed_by, y = avg_box_office)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(
    title = "Top 3 Distribution Companies by Average Box Office Revenue",
    x = "Distribution Companies",
    y = "Average Box Office Revenue (Millions)"
  ) +
  scale_x_discrete(labels = c("Hoyts Distribution" = "Hyots", "Walt Disney StudiosMotion P
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Ques 3. Top 10 Directors with most movies.

The `gt()` function from the `gt` package is utilized to create a nicely formatted table (`table_gt`) displaying the top 10 directors along with the number of movies they've directed.

Additionally, a bar plot is generated using **`ggplot2`** (**`plot`**) to visually represent the top 10 directors and the number of movies they've directed. The x-axis represents the directors (ordered by the number of movies they've directed), while the y-axis represents the count of movies.

This code helps us find out which directors have made the most movies in our dataset. It counts how many movies each director has directed and then lists the top 10 directors with the highest numbers. For example, from our data, we see that directors like Clyde Geronimi and Hamilton Luske have each directed eight movies, making them the most prolific directors in our dataset.

```r
director_counts <- final_data %>%
  separate_rows(directed_by, sep = ",") %>%
  mutate(directed_by = trimws(directed_by)) %>%
  group_by(directed_by) %>%
  summarise(num_movies = n()) %>%
  arrange(desc(num_movies))
top_directors <- head(director_counts, 10)

table_gt <- top_directors %>%
  gt() %>%
  tab_header(
    title = "Top 10 Directors by Number of Movies"
  ) %>%
  cols_label(
    directed_by = "Director",
    num_movies = "Number of Movies"
  )

table_gt
```

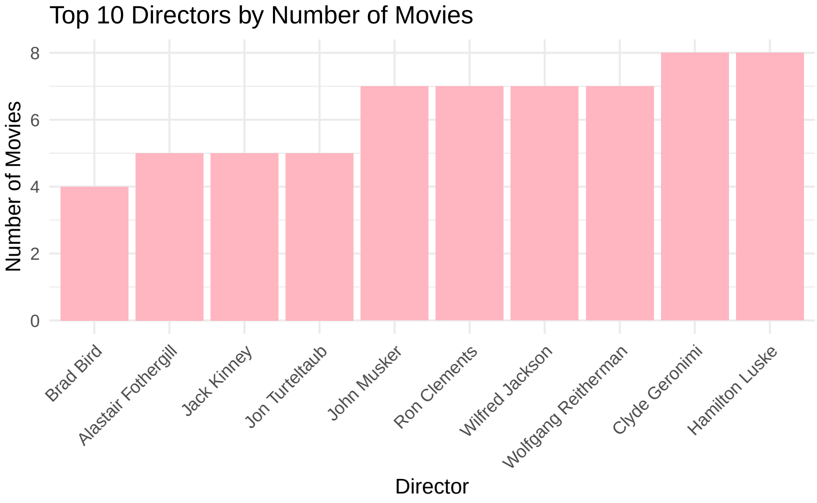
Top 10 Directors by Number of Movies

Director	Number of Movies
Clyde Geronimi	8
Hamilton Luske	8
John Musker	7
Ron Clements	7
Wilfred Jackson	7
Wolfgang Reitherman	7
Alastair Fothergill	5
Jack Kinney	5

Jon Turteltaub	5
Brad Bird	4

```
plot <- ggplot(top_directors, aes(x = reorder(directed_by, num_movies), y = num_movies)) +
  geom_bar(stat = "identity", fill = "lightpink") +
  labs(
    x = "Director",
    y = "Number of Movies",
    title = "Top 10 Directors by Number of Movies"
  ) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

plot
```



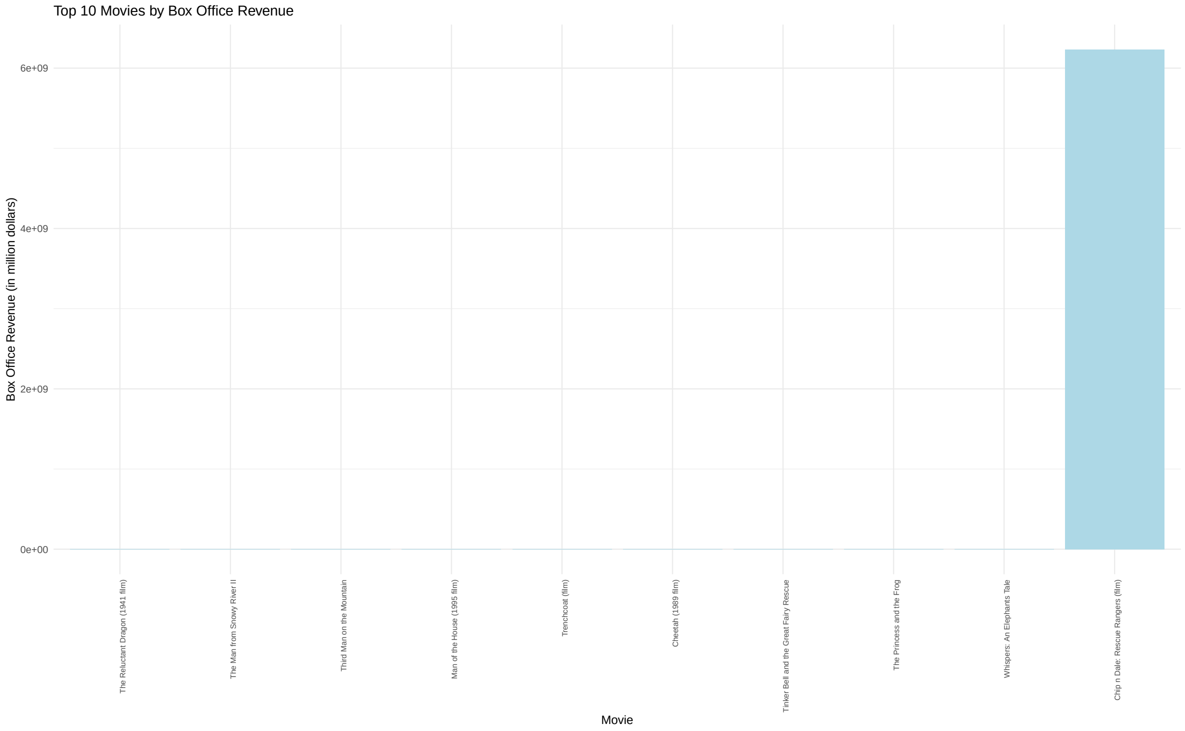
Ques 4. Box office and budget comparison

In the movie industry, budgets and box office revenues play a crucial role in determining success. Among the top 10 movies analyzed, “Alice in Wonderland” (2010) stands out with the highest budget, estimated at around 0.015 million dollars. On the other hand, “Chip ’n Dale: Rescue Rangers” emerges as the clear winner in terms of box office revenue, generating over 6 billion dollars in revenue, making it the top earner among the selected movies. These figures highlight the financial aspects of movie production and how certain films can significantly outperform others at the box office.

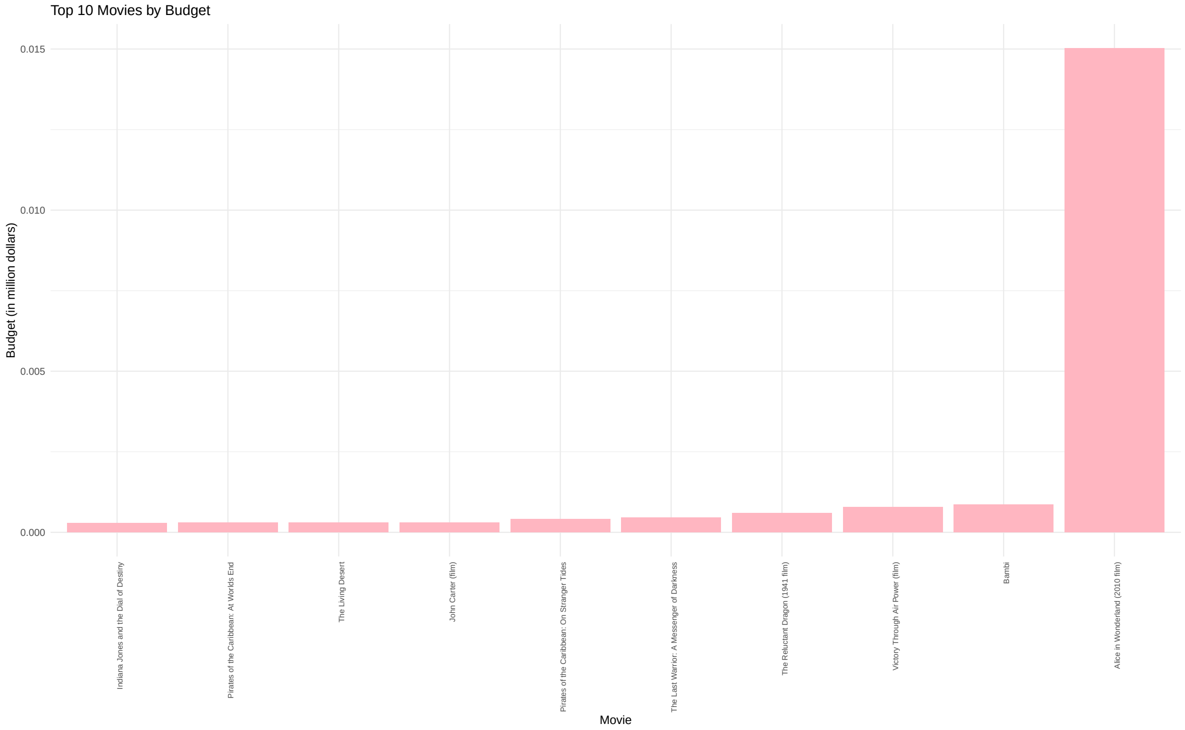
```r
final_data <- final_data %>%
  mutate(budget_millions = as.numeric(gsub("million", "", budget, ignore.case = TRUE)) / 1
         box_office_millions = box_office / 1e6)

top_10_box_office <- final_data %>%
  arrange(desc(box_office_millions)) %>%
  slice(1:10)

top_10_budget <- final_data %>%
  arrange(desc(budget_millions)) %>%
  slice(1:10)
ggplot(top_10_box_office, aes(x = fct_reorder(movie_name, box_office_millions),
                              y = box_office_millions)) +
  geom_bar(stat = "identity", fill = "lightblue") +
  labs(title = "Top 10 Movies by Box Office Revenue",
       x = "Movie",
       y = "Box Office Revenue (in million dollars)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 7))
```



```
ggplot(top_10_budget, aes(x = fct_reorder(movie_name, budget_millions),
                          y = budget_millions)) +
  geom_bar(stat = "identity", fill = "lightpink") +
  labs(title = "Top 10 Movies by Budget",
       x = "Movie",
       y = "Budget (in million dollars)") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, size = 7))
```



References

1. For information on regular expressions, see [https://r4ds.hadley.nz/regexps.html#regular-expressions-in-other-places].
2. For details about logicals, refer to [https://r4ds.hadley.nz/logicals.html#footnotes].
3. To learn about pattern control in regular expressions, visit [https://r4ds.hadley.nz/regexps.html#pattern-control].
4. Irizarry, R. A. (2019).Introduction to data science: Data analysis and prediction algorithmswith R. CRC Press.