**Disney Movie Success**

By:

Andres Gonzalez
(Section 02)

Christina Lopez
(Section 01)

Katya Mora
(Section 02)

## I. Introduction

The data set for this project was merged from two data files: Disney movies total gross which contains the total gross revenue of Walt Disney Studios Movies from 1937 to 2016 and Disney revenue which contains the revenue of the Walt Disney Company from 1992 to 2016. After merging these data sets together we created a subset that we want to further analyze. The subset included the years 1992 to 2016 where both data sets overlapped each other.

The research interests of this report are: finding what the highest and lowest grossing movies disney has created over the time period, identifying the most common genre produced by Disney, finding the variables that best predict the actual revenue per year, and being able to predict the total revenue in a year where Disney releases 10 movies while 2 of them are comedies. It will be shown that there are only two variables that will predict revenue per year.

The following regression model is obtained:

$$Revenue = 55173.2 - 2511.2comedy - 607.9movie\ count$$

## II. Data Description

**Description of variables of interest for each individual movie**

The variables of interest consisted of Gross Revenue as our response which describes gross inflation-adjusted revenue per million (Millions of US$), Year of the movie released, Genre of the movie, the MPAA rating for the movie and lastly the title of the movie.

**Description of variables of interest for yearly summary**

The variables of interest consisted of Gross Revenue as our response, year of total revenue, count of movies released every year, count of movies released per year with the genre “musical”, count of movies released per year with the genre “comedy”, count of movies released per year with the genre “action”, count of movies released per year with the genre “adventure” and lastly the count of movies released per year with the genre “drama”.

## III. Questions of Interest

1. *What are the highest and lowest grossing movies?*
   To answer this we will use an ordered general table of gross revenue per movie.
2. *What is the most common genre produced by Disney?*

We will use a bar chart of the count of movie genres .

3. *Which variables best predict the actual revenue per year?*
   We will create an AIC variable selection and apply multiple linear regression to the model we find using AIC variable selection.
4. *What is Disney's expected total revenue in a year where they release 10 movies and 2 of them are comedies?*
   To create the 95% prediction interval we will be using the multiple linear regression model.

## IV. Regression Analysis, Results and Interpretation

Question 1

*What are the highest and lowest grossing movies?*

Star Wars Ep. VII: The Force Awakens was the highest grossing movie bringing in 936.7 million dollars. The movie was released in 2015 with a rating of PG-13 in a genre of adventure. To conclude this, we created the chart below that lists the top 8 grossing movies.

Source: Disney Character Success from Kaggle

Summary of the $ **Gross Revenue Per Movie** from 1992 to 2016

Million US $

| Movies | Genre | Rating | Year | $ Gross |
|---|---|---|---|---|
| Star Wars Ep. VII: The Force Awakens | Adventure | PG-13 | 2015 | 936.7 |
| The Lion King | Adventure | G | 1994 | 761.6 |
| The Avengers | Action | PG-13 | 2012 | 660.1 |
| Pirates of the Caribbean: Dead Man'... | Adventure | PG-13 | 2006 | 544.8 |
| Rogue One: A Star Wars Story | Adventure | PG-13 | 2016 | 529.5 |
| Finding Nemo | Adventure | G | 2003 | 518.1 |
| Finding Dory | Adventure | PG | 2016 | 486.3 |
| The Sixth Sense | Thriller/Suspense | PG-13 | 1999 | 485.4 |

This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016

Walt and El Grupo was the lowest grossing movie only making $23,064. The movie was released in 2009, and it is a documentary with a PG rating. To conclude this, we created the same chart below but that lists the top 8 lowest grossing movies from 1992 to 2016.

Source: Disney Character Success from Kaggle

**Summary of the Gross Revenue Per Movie from 1992 to 2016**

Million US $

| Movies | Genre | Rating | Year | $ Gross |
|---|---|---|---|---|
| Walt and El Grupo | Documentary | PG | 2009 | 0.0 |
| Zokkomon | Adventure | PG | 2011 | 0.0 |
| An Alan Smithee Film: Burn Hollywood ... | Comedy | R | 1998 | 0.1 |
| Waking Sleeping Beauty | Documentary | PG | 2010 | 0.1 |
| Gedo Senki (Tales from Earthsea) | Adventure | PG-13 | 2010 | 0.1 |
| Breakfast of Champions | Comedy | R | 1999 | 0.3 |
| Goal! 2: Living the Dream... | Drama | PG-13 | 2008 | 0.3 |
| Morning Light | Documentary | PG | 2008 | 0.3 |

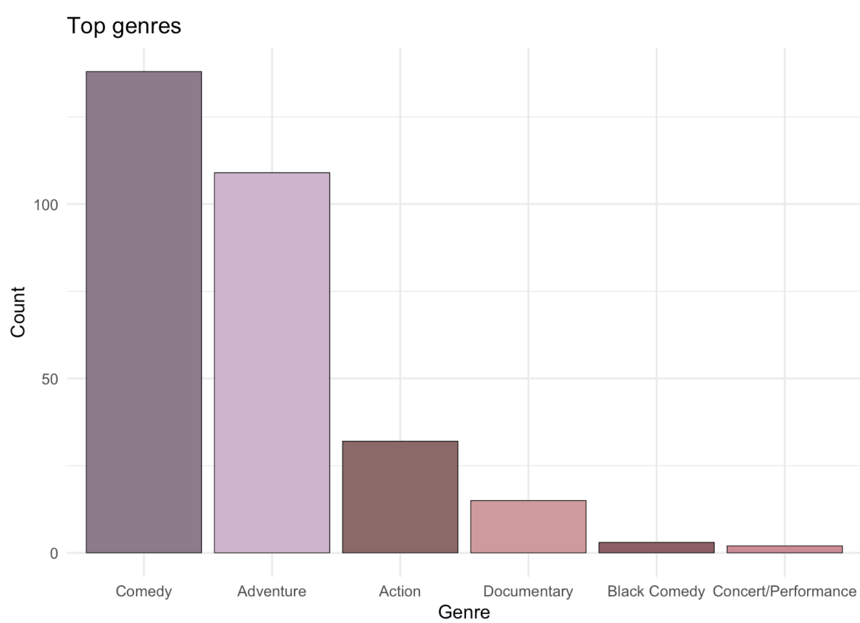
This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016

---

## Question 2

*What is the most common genre produced by Disney?*

The most common genre produced by Disney overall is comedy. Movies include Recess: School's out, Air Bud and Aladdin to name a few. To conclude this we created a bar graph that counted the different genres. As we can see in the bar graph below that comedy had the highest count producing more than 100 movies from 1992 to 2016.



Question 3

*Which variables best predict the actual revenue per year?*

To best determine which variables will predict revenue per year we first identify all variables of interest. In this case our variables of interest consist of 7 different variables. In this data set we also narrowed our data so that both files had overlapping years we can compare, those years are 1992-2016. Therefore we used stepwise regression AIC for variable selection to compare all relevant variables and see which of these are significant for our model.

The result of the stepwise regression AIC (Appendix E), was a 2 predictor model with total_revenue as the response and comedy and movie_count as our predictors. This model produces the lowest AIC value of 431.19. After, to show that all variables in this model were significant , we created a summary of the model which tested the t-tests shown below.

$H_0$: $\beta_{comedy} = 0$ vs $H_1$: $\beta_{comedy} \neq 0$. The t statistic is -5.330 and the p-value is 2.38E-05

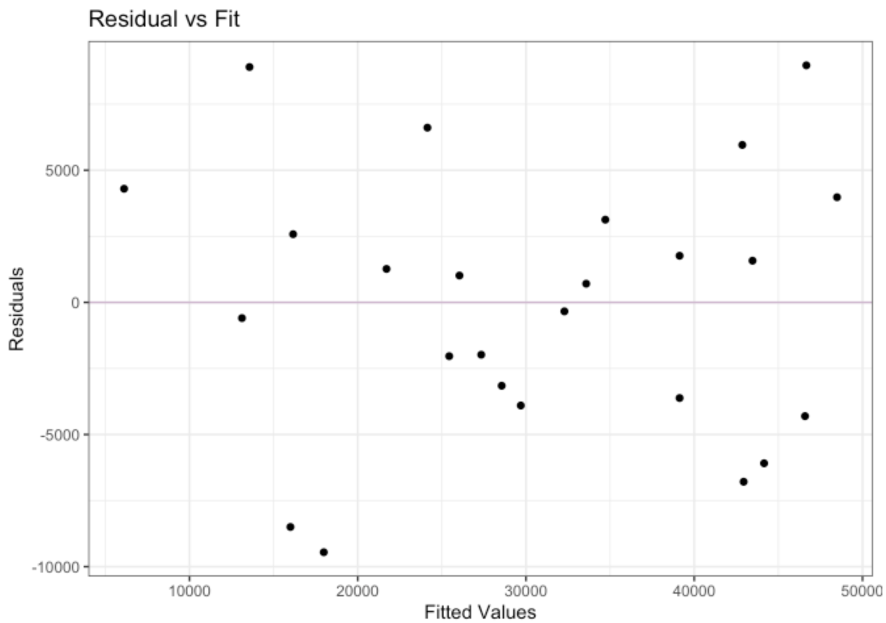
$H_0$: $\beta_{movie\_count} = 0$ vs $H_1$: $\beta_{movie\_count} \neq 0$. The t-statistic is -1.957 and the p-value is 0.063.

As we can see both comedy and movie_count are both significant since p values are both less than 0.1. Comedy was significant to .001 and movie count was significant to 0.1. The coefficient of determination for the fill model is .856. This shows that 85.6% of variation in total_revenue is explained by variation in the movie type comedy and movie_count.

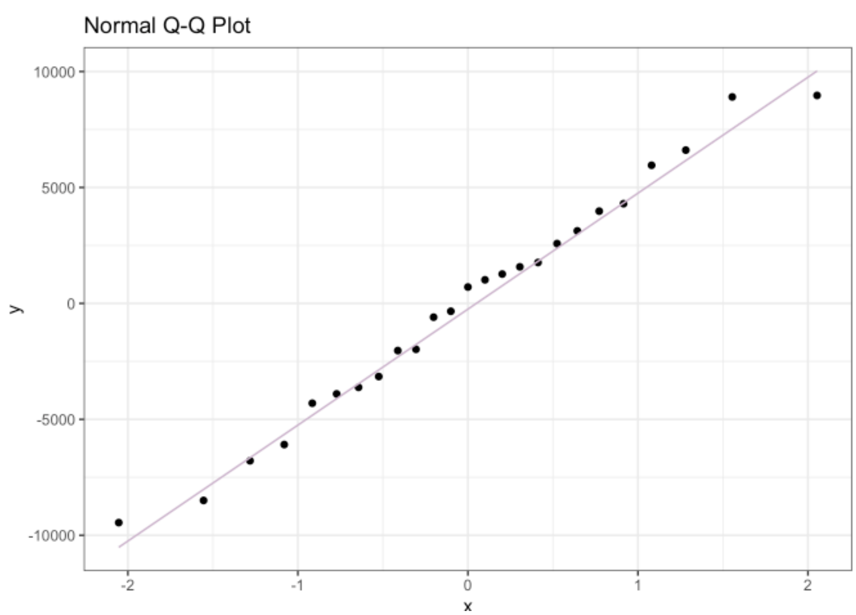
Interaction variables were also checked to verify if movie_count*comedy would be significant to our model. The R function add1 conducts an F test on a full model (with interaction) compared to a reduced model (without interaction). As we see from the results summarized below, the p-value for the interaction term is 0.2044 which is larger than 0.05, therefore we will not include it in the final model.

$H_0$: $\beta_{comedy:movie\_count} = 0$ vs $H_1$: $\beta_{comedy:movie\_count} \neq 0$. The F statistic is 1.7156 and the p-value is 0.204.

Since only the variables movie_count and comedy are significant in our model, line conditions were checked next. In the chart below, we can see that the Plot is “Well-Behaved” which implies that non-linearity is not a problem and the errors have equal variances. From this we can see that a transformation will not be required on x or y.




Next, independence and normality assumptions were checked. As we see from the graph below, it appears normally distributed. We also verified using the Shaprio test, to make sure no additional transformations are required. After conducting we get a p value of 0.91, which is greater than alpha 0.05, therefore we fail to reject our null hypothesis.




In conclusion, there were only 2 variables of interest that will sufficiently predict total revenue as nor interaction terms were needed and transformation required.

Question 4

*What is Disney's expected total revenue in a year where they release 10 movies and 2 of them are comedies?*

We are 95% confident that Disney's total gross revenue in a year where they release 10 movies and 2 of them are comedies will be between 32500.31 and 55643.5 in millions US $.

## V. Conclusion

Paying attention to how high our R - squared were, after initializing the stepwise regression AIC technique to select the best subset predictors to the model for the response. With that being said, we get a 2 predictor model with total_revenue as the response and comedy and movie_count as our main predictors obtaining an AIC of 431.19. We suggested adding an interaction term to the model with having comedy and movie_count multiplied together to see if it reaches a higher R - squared value of 85.6%. Based on the summary of the full model with interaction effect, we decided not to include the interaction term into the model as it is not significant since it has a p - value greater than 0.05. After obtaining a model with significant predictors, we also need to check for residual plot, normal QQ - plot for and any outliers. Moving forward, the linearity and equal variance in the residual plot is "well-behaved", so transformation will not be required for the model. Also, the QQ - plot looks normally distributed, so we also leave the response as it is. In the end, the LINE assumptions are met. Therefore, our final model indicates that 85.6% of the variation is explained by comedy and movie counts for the total revenue.

Finalizing our thoughts from the model is that comedy is the most common genre by Disney between the years 1992 to 2016, and we can successfully predict Disney's annual total revenue by counting the number of total movies released each year and counting the number of comedy movies released each year. We are also 95% that Disney's gross revenue in a year where they release 10 movies and 2 of them are comedies will make between $32,500 M and $55,643 M. We think the one clear thing here is that we should likely avoid time series since the data set contains years in time for movies; otherwise, the residual plot will show a pattern. Lastly, it would be beneficial to include data beyond 2016 to see the most recent changes in movies Disney releases.

## VI. Appendix

Contents:

**Appendix A:** Data Cleaning
**Appendix B:** Exploratory Data Analysis
**Appendix C:** What are the highest and lowest grossing movies?
**Appendix D:** What is the most common genre produced by Disney?
**Appendix E:** Which variables best predict the actual revenue per year?
**Appendix F:** What is Disney's expected total revenue in a year where they release 10 movies and 2 of them are comedies?

# Appendix

2022-12-05

## Appendix A: Data Cleaning

```
library(tidyverse)
library(skimr)
library(gt)
library(scales)
library(broom)
library(RColorBrewer)
```

```
disney <- read_csv(
  paste('https://raw.githubusercontent.com/stinalindaa/',
  'disneymovies/main/disney_movies_total_gross.csv', sep = ""))

skim_without_charts(disney)
```

Table 1: Data summary

| | |
|---|---|
| Name | disney |
| Number of rows | 579 |
| Number of columns | 7 |
| Column type frequency: | |
| character | 6 |
| numeric | 1 |
| Group variables | None |

**Variable type: character**

| skim_variable | n_missing | complete_rate | min | max | empty | n_unique | whitespace |
|---|---|---|---|---|---|---|---|
| movie_title | 0 | 1.00 | 2 | 40 | 0 | 573 | 0 |
| release_date | 0 | 1.00 | 11 | 12 | 0 | 553 | 0 |
| genre | 17 | 0.97 | 5 | 19 | 0 | 12 | 0 |
| MPAA_rating | 56 | 0.90 | 1 | 9 | 0 | 5 | 0 |
| total_gross | 0 | 1.00 | 2 | 12 | 0 | 576 | 0 |
| inflation_adjusted_gross | 0 | 1.00 | 2 | 14 | 0 | 576 | 0 |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
|---|---|---|---|---|---|---|---|---|---|
| index | 0 | 1 | 289 | 167.29 | 0 | 144.5 | 289 | 433.5 | 578 |

Clean up variables and drop NA's.

```
disney2 <- disney |>
  mutate(genre = factor(genre),
         rating = factor(MPAA_rating, levels = c("G", "PG", "PG-13", "R")),
         gross = parse_number(inflation_adjusted_gross),
         release_date = as.Date(release_date, format = "%b %d, %Y"),
         release_year = as.numeric(format(release_date, "%Y")),
         gross_million = round(gross/1000000, digits = 1),
         gross_K = round(gross/1000, digits =3)) |>
  select(-total_gross, -inflation_adjusted_gross, -MPAA_rating ) |>
  drop_na() |>
  filter(release_year >= 1992)
head(disney2)
```

```
## # A tibble: 6 x 9
##   index movie_title       release_~1 genre rating  gross relea~2 gross~3 gross_K
##   <dbl> <chr>             <date>     <fct> <fct>   <dbl>   <dbl>   <dbl>   <dbl>
## 1   116 The Hand That Ro~ 1992-01-10 Thri~ R      1.79e8    1992   179.  178831.
## 2   117 Medicine Man      1992-02-07 Drama PG-13  9.13e7    1992    91.3  91304.
## 3   118 Blame it on the ~ 1992-03-06 Come~ PG-13  5.87e6    1992     5.9   5873.
## 4   119 Noises Off...     1992-03-20 Come~ PG-13  4.63e6    1992     4.6   4632.
## 5   120 Straight Talk     1992-04-03 Come~ PG     4.31e7    1992    43.1  43068.
## 6   123 Encino Man        1992-05-22 Come~ PG     8.14e7    1992    81.4  81369.
## # ... with abbreviated variable names 1: release_date, 2: release_year,
## #   3: gross_million
```

```
revenue <- read_csv(
  paste('https://raw.githubusercontent.com/stinalindaa/',
        'disneymovies/main/disney_revenue_1991-2016.csv', sep = ""))

skim_without_charts(revenue)
```

Table 4: Data summary

| | |
|---|---|
| Name | revenue |
| Number of rows | 26 |
| Number of columns | 8 |
| Column type frequency: | |
| numeric | 8 |
| Group variables | None |

**Variable type: numeric**

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
|---|---|---|---|---|---|---|---|---|---|
| index | 0 | 1.00 | 12.50 | 7.65 | 0 | 6.25 | 12.5 | 18.75 | 25 |
| Year | 0 | 1.00 | 2003.50 | 7.65 | 1991 | 1997.25 | 2003.5 | 2009.75 | 2016 |

| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
|---|---|---|---|---|---|---|---|---|---|
| Studio Entertainment[NI 1] | 1 | 0.96 | 6445.04 | 1570.28 | 2593 | 5994.00 | 6701.0 | 7364.00 | 9441 |
| Disney Consumer Products[NI 2] | 2 | 0.92 | 2591.05 | 877.11 | 724 | 2182.25 | 2475.5 | 3085.00 | 4499 |
| Disney Interactive[NI 3][Rev 1] | 14 | 0.46 | 713.67 | 386.48 | 174 | 341.00 | 740.0 | 1002.50 | 1299 |
| Walt Disney Parks and Resorts | 0 | 1.00 | 8512.62 | 4253.95 | 2794 | 5143.50 | 7276.5 | 11318.25 | 16974 |
| Disney Media Networks | 3 | 0.88 | 12877.70 | 6736.88 | 359 | 8540.50 | 13207.0 | 17938.00 | 23689 |
| Total | 0 | 1.00 | 29459.69 | 13846.67 | 6111 | 22598.75 | 28906.5 | 38008.00 | 55632 |

```
movies.revenue <- disney2 |>
  mutate(Year = release_year) |>
  inner_join(revenue, by = c("Year" = "Year"))
head(movies.revenue)
```

```
## # A tibble: 6 x 17
##   index.x movie_t~1 release_~2 genre rating   gross relea~3 gross~4 gross_K  Year
##     <dbl> <chr>     <date>     <fct> <fct>    <dbl>   <dbl>   <dbl>   <dbl> <dbl>
## 1     116 The Hand~ 1992-01-10 Thri~ R       1.79e8    1992   179.  178831.  1992
## 2     117 Medicine~ 1992-02-07 Drama PG-13   9.13e7    1992    91.3  91304.  1992
## 3     118 Blame it~ 1992-03-06 Come~ PG-13   5.87e6    1992     5.9   5873.  1992
## 4     119 Noises O~ 1992-03-20 Come~ PG-13   4.63e6    1992     4.6   4632.  1992
## 5     120 Straight~ 1992-04-03 Come~ PG      4.31e7    1992    43.1  43068.  1992
## 6     123 Encino M~ 1992-05-22 Come~ PG      8.14e7    1992    81.4  81369.  1992
## # ... with 7 more variables: index.y <dbl>, `Studio Entertainment[NI 1]` <dbl>,
## #   `Disney Consumer Products[NI 2]` <dbl>,
## #   `Disney Interactive[NI 3][Rev 1]` <dbl>,
## #   `Walt Disney Parks and Resorts` <dbl>, `Disney Media Networks` <dbl>,
## #   Total <dbl>, and abbreviated variable names 1: movie_title,
## #   2: release_date, 3: release_year, 4: gross_million
```

```
yearly.summary <- movies.revenue |>
  group_by(Year, Total) |>
  summarize(movie_count = n()) |>
  rename(total_revenue = Total) |>
  ungroup()
```

```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
```

```
head(yearly.summary)
```

```
## # A tibble: 6 x 3
##    Year total_revenue movie_count
##   <dbl>         <dbl>       <int>
## 1  1992          7502          19
## 2  1993          8529          24
## 3  1994         10414          27
## 4  1995         12525          32
## 5  1996         18739          27
## 6  1997         22473          23
```

```r
movies.revenue2 <- movies.revenue |>
  mutate(action = ifelse(genre == "Action", 1, 0),
         adventure = ifelse(genre == "Adventure",1,0),
         musical = ifelse(genre == "Musical", 1, 0),
         drama = ifelse(genre == "Drama", 1, 0),
         comedy = ifelse(genre == "Comedy", 1, 0)) |>
  filter(Year >= 1992)
head(movies.revenue2)
```

```
## # A tibble: 6 x 22
##   index.x movie_t~1 release_~2 genre rating   gross relea~3 gross~4 gross_K  Year
##     <dbl> <chr>     <date>     <fct> <fct>    <dbl>   <dbl>   <dbl>   <dbl> <dbl>
## 1     116 The Hand~ 1992-01-10 Thri~ R       1.79e8    1992   179.  178831.  1992
## 2     117 Medicine~ 1992-02-07 Drama PG-13   9.13e7    1992    91.3  91304.  1992
## 3     118 Blame it~ 1992-03-06 Come~ PG-13   5.87e6    1992     5.9   5873.  1992
## 4     119 Noises O~ 1992-03-20 Come~ PG-13   4.63e6    1992     4.6   4632.  1992
## 5     120 Straight~ 1992-04-03 Come~ PG      4.31e7    1992    43.1  43068.  1992
## 6     123 Encino M~ 1992-05-22 Come~ PG      8.14e7    1992    81.4  81369.  1992
## # ... with 12 more variables: index.y <dbl>,
## #   `Studio Entertainment[NI 1]` <dbl>, `Disney Consumer Products[NI 2]` <dbl>,
## #   `Disney Interactive[NI 3][Rev 1]` <dbl>,
## #   `Walt Disney Parks and Resorts` <dbl>, `Disney Media Networks` <dbl>,
## #   Total <dbl>, action <dbl>, adventure <dbl>, musical <dbl>, drama <dbl>,
## #   comedy <dbl>, and abbreviated variable names 1: movie_title,
## #   2: release_date, 3: release_year, 4: gross_million
```

```r
yearly.summary2 <- movies.revenue2 |>
  group_by(Year, Total) |>
  summarize(movie_count = n(),
            action_count = sum(action),
            adventure = sum(adventure),
            musical = sum(musical),
            drama = sum(drama),
            comedy = sum(comedy)) |>
  rename(total_revenue = Total) |>
  ungroup() |>
  select(-Year)
```

```
## `summarise()` has grouped output by 'Year'. You can override using the
## `.groups` argument.
```

```r
head(yearly.summary2)
```

```
## # A tibble: 6 x 7
##   total_revenue movie_count action_count adventure musical drama comedy
##           <dbl>       <int>        <dbl>     <dbl>   <dbl> <dbl>  <dbl>
## 1          7502          19            1         1       0     4     11
## 2          8529          24            2         5       2     2      9
## 3         10414          27            3         3       0     5     13
## 4         12525          32            2         5       0    10      9
## 5         18739          27            2         5       1     7      9
## 6         22473          23            2         1       0     5     11
```

# Appendix B: Exporatory Data Analysis

Disney movies total gross

```
skim_without_charts(disney2)
```

Table 6: Data summary

| | |
|---|---|
| Name | disney2 |
| Number of rows | 449 |
| Number of columns | 9 |
| Column type frequency: | |
| character | 1 |
| Date | 1 |
| factor | 2 |
| numeric | 5 |
| Group variables | None |

**Variable type: character**

| skim_variable | n_missing | complete_rate | min | max | empty | n_unique | whitespace |
|---|---|---|---|---|---|---|---|
| movie_title | 0 | 1 | 2 | 40 | 0 | 447 | 0 |

**Variable type: Date**

| skim_variable | n_missing | complete_rate | min | max | median | n_unique |
|---|---|---|---|---|---|---|
| release_date | 0 | 1 | 1992-01-10 | 2016-12-16 | 2001-11-21 | 425 |

**Variable type: factor**

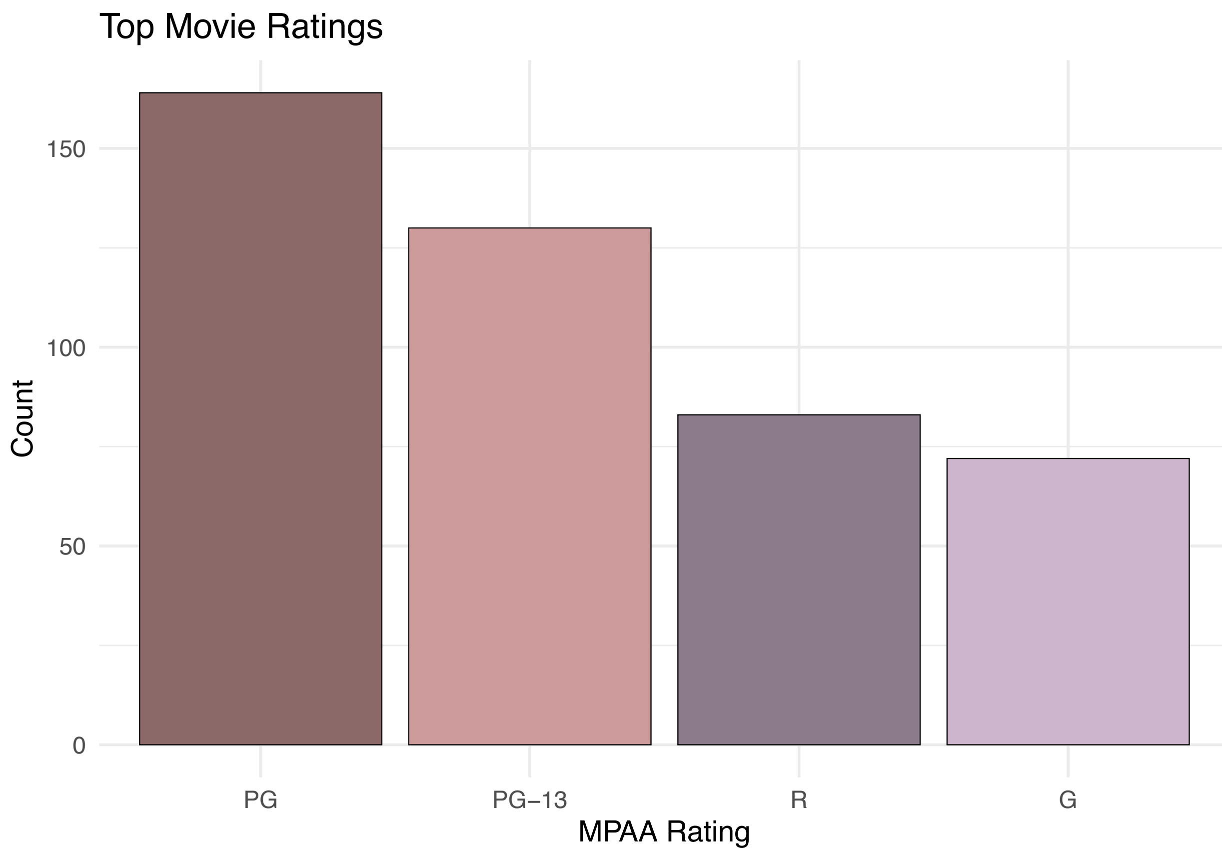
| skim_variable | n_missing | complete_rate | ordered | n_unique | top_counts |
|---|---|---|---|---|---|
| genre | 0 | 1 | FALSE | 12 | Com: 138, Adv: 109, Dra: 89, Act: 32 |
| rating | 0 | 1 | FALSE | 4 | PG: 164, PG-: 130, R: 83, G: 72 |

**Variable type: numeric**

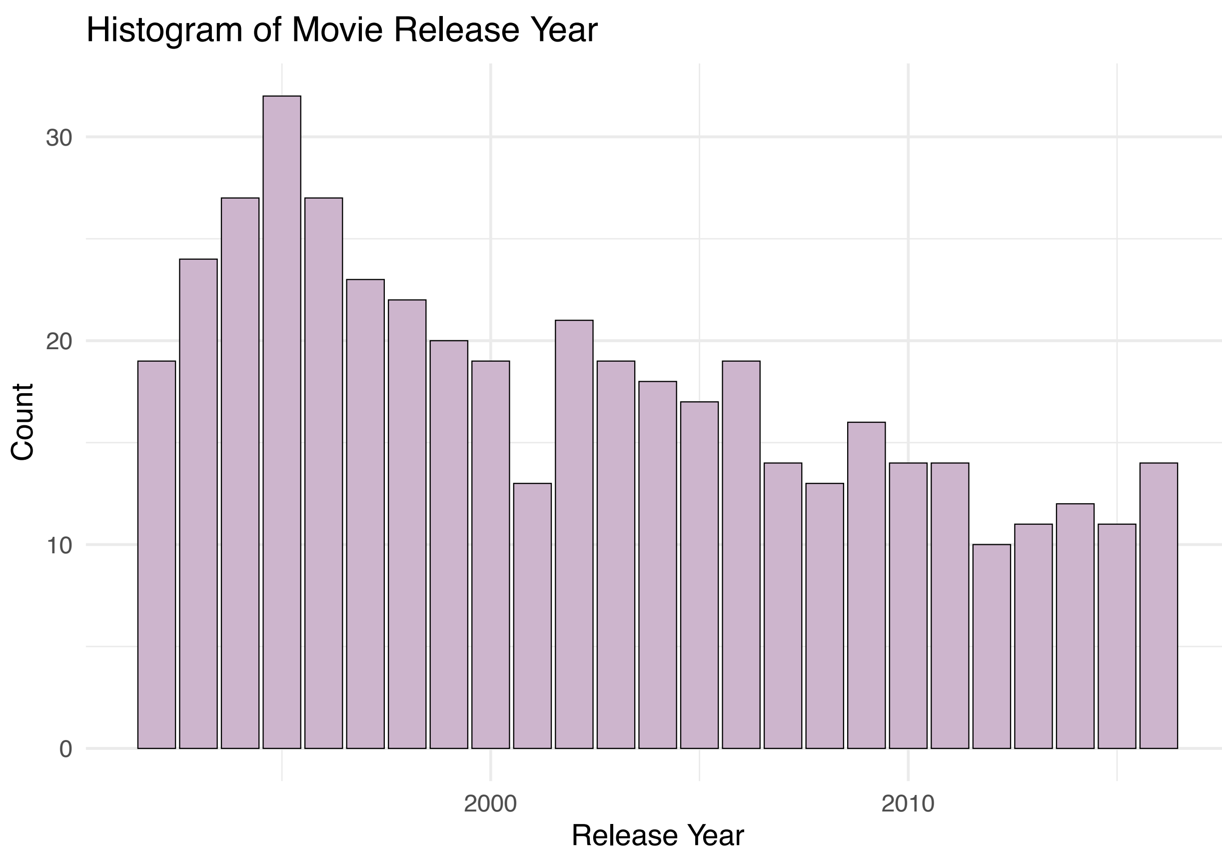
| skim_variable | n_missing | complete_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
|---|---|---|---|---|---|---|---|---|---|
| index | 0 | 1 | 351.10 | 1.32770e+02 | 116.00 | 237.00 | 352.00 | 466.0 | 578.0 |
| gross | 0 | 1 | 98410523.66 | 1.19627e+08 | 2984.00 | 24267154.00 | 55961409.00 | 116965668.0 | 936662225.0 |
| release_year | 0 | 1 | 2002.14 | 7.03000e+00 | 1992.00 | 1996.00 | 2001.00 | 2008.0 | 2016.0 |
| gross_million | 0 | 1 | 98.41 | 1.19630e+02 | 0.00 | 24.30 | 56.00 | 117.0 | 936.7 |
| gross_K | 0 | 1 | 98410.52 | 1.19627e+05 | 2.98 | 24267.15 | 55961.41 | 116965.7 | 936662.2 |

```r
colors = c("thistle3", "thistle4", "rosybrown3", "rosybrown4")
disney2 |>
  group_by(rating) |>
  summarize(n = n()) |>
  ggplot(aes(x = reorder(rating, -n), y = n)) +
  geom_col(aes(fill = reorder(rating,n)), color = "black", size = 0.2) +
  scale_fill_manual(values = colors) +
  ggtitle("Top Movie Ratings")+ labs(x = "MPAA Rating", y = "Count") +
  theme_minimal() + theme(legend.position = "none")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
```

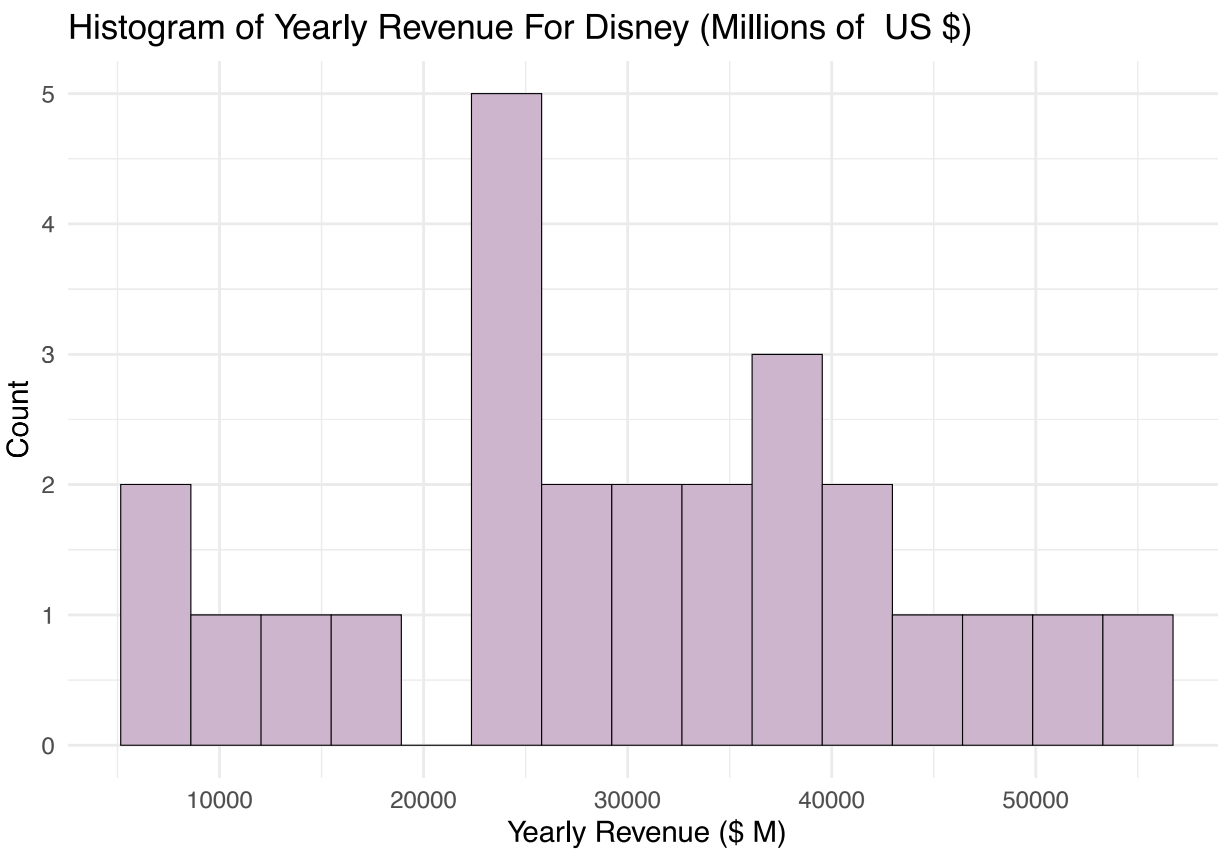



```r
disney2 |>
  select(release_year) |>
  filter(release_year >= 1991) |>
  group_by(release_year) |>
  summarise(n = n()) |>
  ggplot(aes(x = release_year, y = n)) +
  geom_col(fill = "thistle3", color = "black", size = 0.2)+
  ggtitle("Histogram of Movie Release Year") +
  theme_minimal() + labs(x = "Release Year", y = "Count")
```




Yearly summary

```
yearly.summary |>
  ggplot(aes(x = total_revenue)) +
  geom_histogram(bins = 15, fill = "thistle3", color = "black", size = 0.2)+
  ggtitle("Histogram of Yearly Revenue For Disney (Millions of  US $)") +
  theme_minimal() + labs(x = "Yearly Revenue ($ M)", y = "Count")
```




## Appendix C: What are the highest and lowest grossing movies?

Highest

```
movies.revenue2 |>
  select(movie_title,genre,rating,Year,gross_million) %>%
  arrange(desc(gross_million)) %>%
  head(8) %>%
  gt(rowname_col = "movie_title") %>%
  tab_header(
    title = md("Summary of the **$ Gross Revenue Per Movie** from 1992 to 2016"),
    subtitle = md("Million US $")) %>%
  tab_source_note(
    source_note = md("This file contains data on the Revenue and Gross of the Walt Disney Company from 
  tab_caption(
    caption = md("Source: Disney Character Success from Kaggle")) %>%
  tab_stubhead(label = md("Movies")) %>%
  opt_table_font(font = google_font("Mouse Memoirs"), weight = 100) %>%
  cols_label(genre = "Genre",
             rating = "Rating",
             gross_million = "$ Gross")
```

Summary of the **$ Gross Revenue Per Movie** from 1992 to 2016

Million US $

| Movies | Genre | Rating | Year | $ Gross |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Star Wars Ep. VII: The Force Awakens | Adventure | PG-13 | 2015 | 936.7 |
| The Lion King | Adventure | G | 1994 | 761.6 |
| The Avengers | Action | PG-13 | 2012 | 660.1 |
| Pirates of the Caribbean: Dead Man'... | Adventure | PG-13 | 2006 | 544.8 |
| Rogue One: A Star Wars Story | Adventure | PG-13 | 2016 | 529.5 |
| Finding Nemo | Adventure | G | 2003 | 518.1 |
| Finding Dory | Adventure | PG | 2016 | 486.3 |
| The Sixth Sense | Thriller/Suspense | PG-13 | 1999 | 485.4 |

This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016

Lowest

```
movies.revenue2 |>
  select(movie_title,genre,rating,Year,gross_million) %>%
  arrange(gross_million) %>%
  head(8) %>%
  gt(rowname_col = "movie_title") %>%
  tab_header(
    title = md("Summary of the **Gross Revenue Per Movie** from 1992 to 2016"),
    subtitle = md("Million US $")) %>%
  tab_source_note(
    source_note = md("This file contains data on the Revenue and Gross of the Walt Disney Company from 1
  tab_caption(
    caption = md("Source: Disney Character Success from Kaggle")) %>%
  tab_stubhead(label = md("Movies")) %>%
  opt_table_font(font = google_font("Mouse Memoirs"), weight = 100) %>%
  cols_label(genre = "Genre",
             rating = "Rating",
             gross_million = "$ Gross")
```

Summary of the **Gross Revenue Per Movie** from 1992 to 2016

Million US $

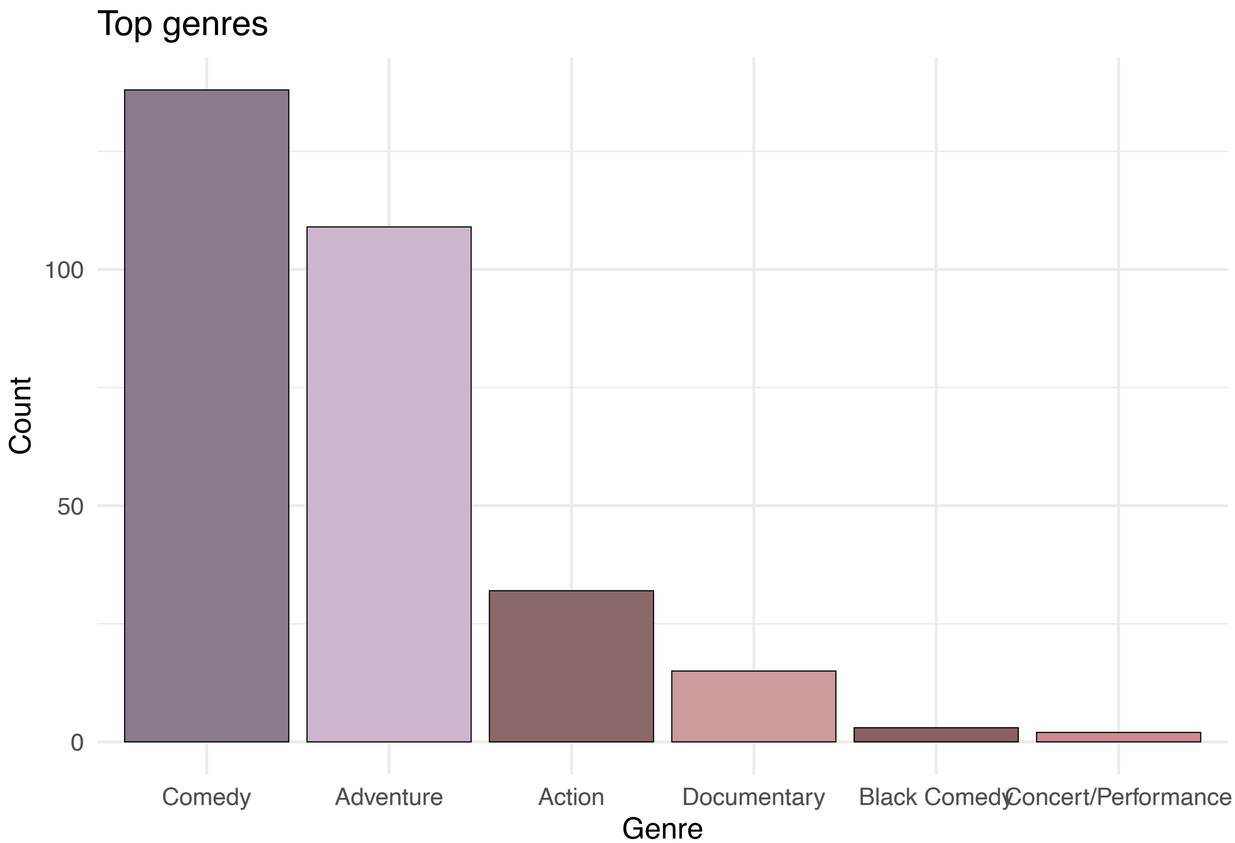
| Movies | Genre | Rating | Year | $ Gross |
|---|---|---|---|---|
| Walt and El Grupo | Documentary | PG | 2009 | 0.0 |
| Zokkomon | Adventure | PG | 2011 | 0.0 |
| An Alan Smithee Film: Burn Hollywood ... | Comedy | R | 1998 | 0.1 |
| Waking Sleeping Beauty | Documentary | PG | 2010 | 0.1 |
| Gedo Senki (Tales from Earthsea) | Adventure | PG-13 | 2010 | 0.1 |
| Breakfast of Champions | Comedy | R | 1999 | 0.3 |
| Goal! 2: Living the Dream... | Drama | PG-13 | 2008 | 0.3 |
| Morning Light | Documentary | PG | 2008 | 0.3 |

This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016

## Appendix D: What is the most common genre produced by Disney?

```
colors4 = c("thistle4", "thistle3", "rosybrown4", "rosybrown3", "lightpink4", "lightpink3")
disney2 |>
  select(genre) |>
  group_by(genre) |>
  summarise(n = n()) |>
```

```r
  head(6) |>
  ggplot(aes(x = reorder(genre, -n), y = n)) +
  geom_col(aes(fill = reorder(genre, -n)), color = "black", size = .2) +
  scale_fill_manual(values = colors4) +
  ggtitle("Top genres") + labs(x = "Genre", y = "Count") +
  theme_minimal() + theme(legend.position = "none")
```




# Appendix E: Which variables best predict the actual revenue per year?

Variable selection:

```r
n = nrow(yearly.summary2)
mod0 = lm(total_revenue ~ 1, data = yearly.summary2)
mod.all = lm(total_revenue ~., data = yearly.summary2)
step(mod0, scope = list(lower = mod0, upper = mod.all))
```

```
## Start:  AIC=475.64
## total_revenue ~ 1
##
##                 Df  Sum of Sq        RSS    AIC
## + comedy         1 3511904893  714385033 433.20
## + movie_count    1 2832168601 1394121325 449.92
## + drama          1 1193650237 3032639688 469.35
## + adventure      1  858021438 3368268488 471.97
## + action_count   1  498819240 3727470685 474.50
```

```
## <none>                         4226289926 475.64
## + musical       1  169228624 4057061301 476.62
##
## Step:  AIC=433.2
## total_revenue ~ comedy
##
##                Df  Sum of Sq        RSS    AIC
## + movie_count   1  105911512  608473521 431.19
## <none>                        714385033 433.20
## + drama         1   50188455  664196578 433.38
## + musical       1   42135833  672249199 433.68
## + adventure     1   24703606  689681427 434.32
## + action_count  1    2234042  712150991 435.12
## - comedy        1 3511904893 4226289926 475.64
##
## Step:  AIC=431.19
## total_revenue ~ comedy + movie_count
##
##                Df Sum of Sq        RSS    AIC
## <none>                       608473521 431.19
## + musical       1  23449554  585023967 432.21
## + action_count  1  19633920  588839601 432.37
## + adventure     1   5720004  602753516 432.95
## + drama         1    137329  608336192 433.18
## - movie_count   1 105911512  714385033 433.20
## - comedy        1 785647804 1394121325 449.92
```

```
##
## Call:
## lm(formula = total_revenue ~ comedy + movie_count, data = yearly.summary2)
##
## Coefficients:
## (Intercept)       comedy  movie_count
##     55173.2      -2511.2       -607.9
```

AIC Model:

```
mod.full <- yearly.summary2 |>
  lm(formula = total_revenue ~ comedy + movie_count)
summary(mod.full)
```

```
##
## Call:
## lm(formula = total_revenue ~ comedy + movie_count, data = yearly.summary2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9454.0 -3619.1   706.5  3128.4  8969.2
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  55173.2     4001.6  13.788 2.64e-12 ***
## comedy       -2511.2      471.2  -5.330 2.38e-05 ***
## movie_count   -607.9      310.6  -1.957   0.0632 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 5259 on 22 degrees of freedom
## Multiple R-squared:  0.856,  Adjusted R-squared:  0.8429
## F-statistic:  65.4 on 2 and 22 DF,  p-value: 5.509e-10
```

Check interaction terms:

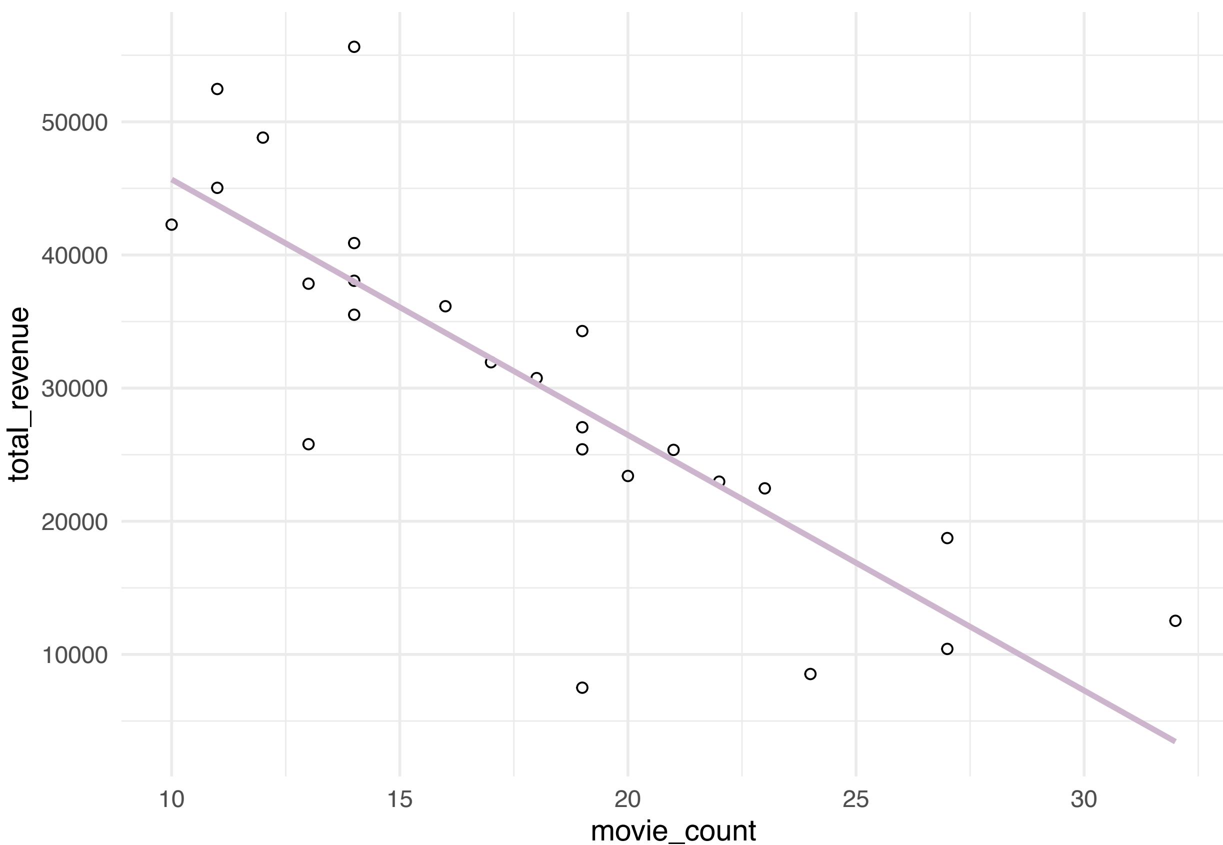
```
add1(mod.full, ~.+comedy*movie_count, test = 'F')
```

```
## Single term additions
##
## Model:
## total_revenue ~ comedy + movie_count
##                    Df Sum of Sq       RSS    AIC F value Pr(>F)
## <none>                          608473521 431.19
## comedy:movie_count  1  45954919 562518602 431.23  1.7156 0.2044
```

Check Model assumptions:

```
yearly.summary2 |>
  ggplot(aes(x = movie_count, y = total_revenue)) +
  geom_point(shape = 21, color = "black") +
  geom_smooth(color = "thistle3", method = "lm", se = FALSE) +
  theme_minimal()
```

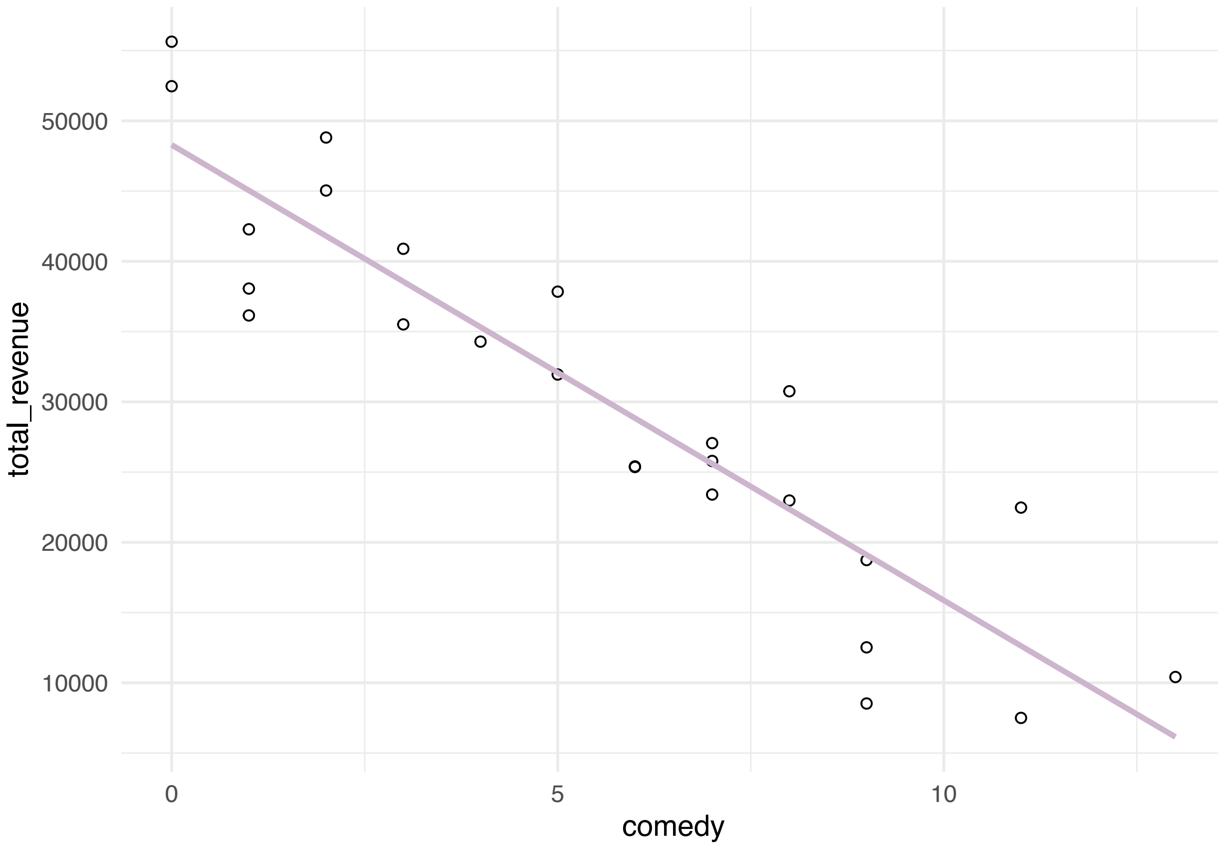
```
## `geom_smooth()` using formula = 'y ~ x'
```



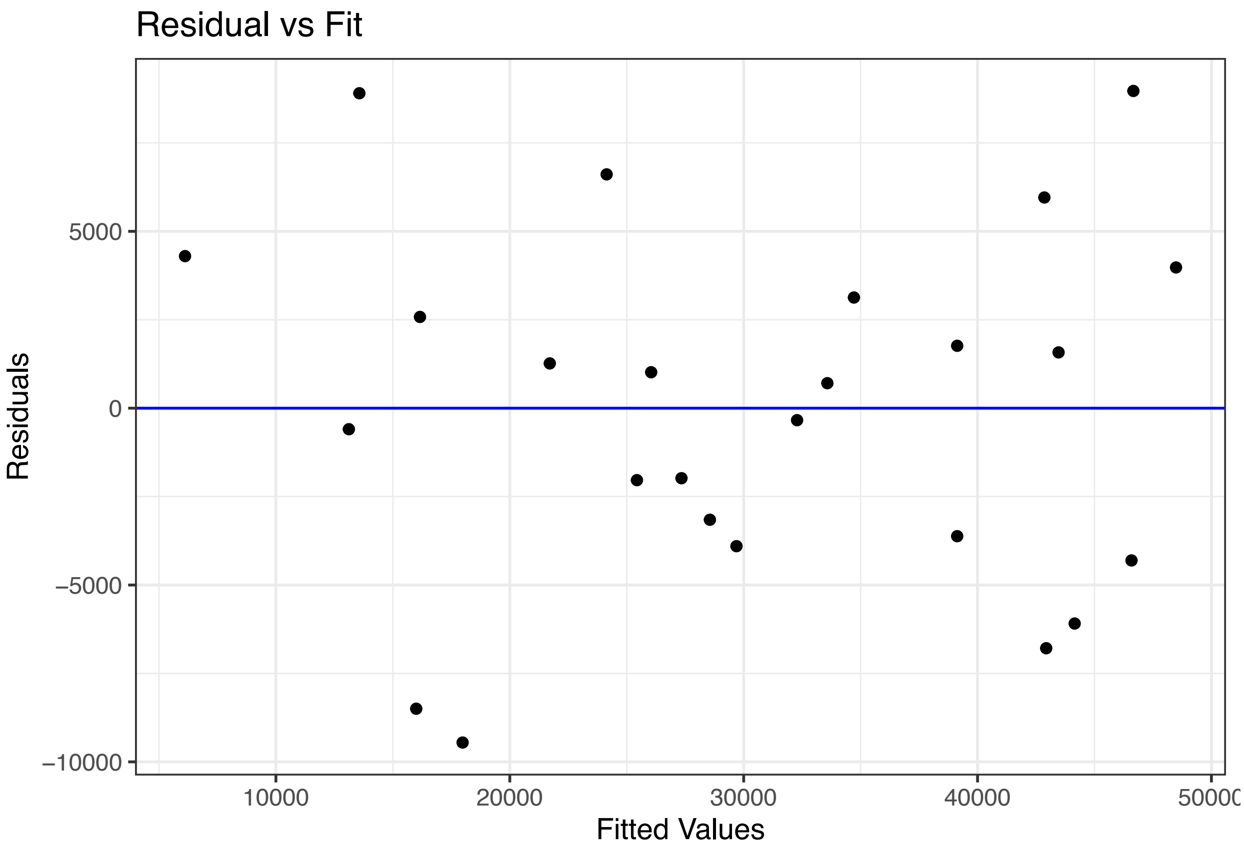

```
yearly.summary2 |>
  ggplot(aes(x = comedy, y = total_revenue)) +
  geom_point(shape = 21, color = "black") +
```

```
  geom_smooth(color = "thistle3", method = "lm", se = FALSE) +
  theme_minimal()
```

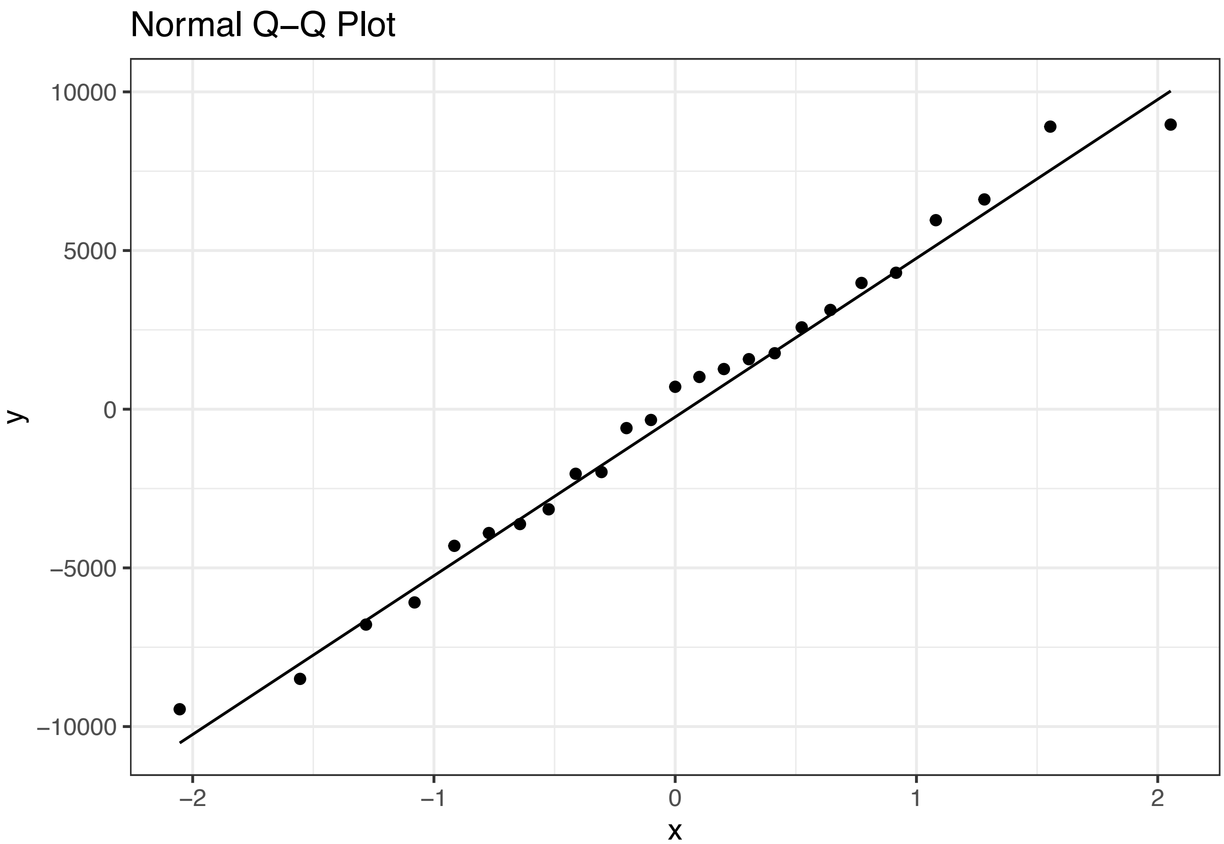
```
## `geom_smooth()` using formula = 'y ~ x'
```




```
mod.full |>
  augment() |>
  ggplot(aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, colour = 'blue') +
  labs(x = 'Fitted Values', y = 'Residuals') +
  ggtitle('Residual vs Fit') +
  theme_bw()
```




```
mod.full |>
  augment() |>
ggplot(aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line() +
  ggtitle('Normal Q-Q Plot') +
  theme_bw()
```

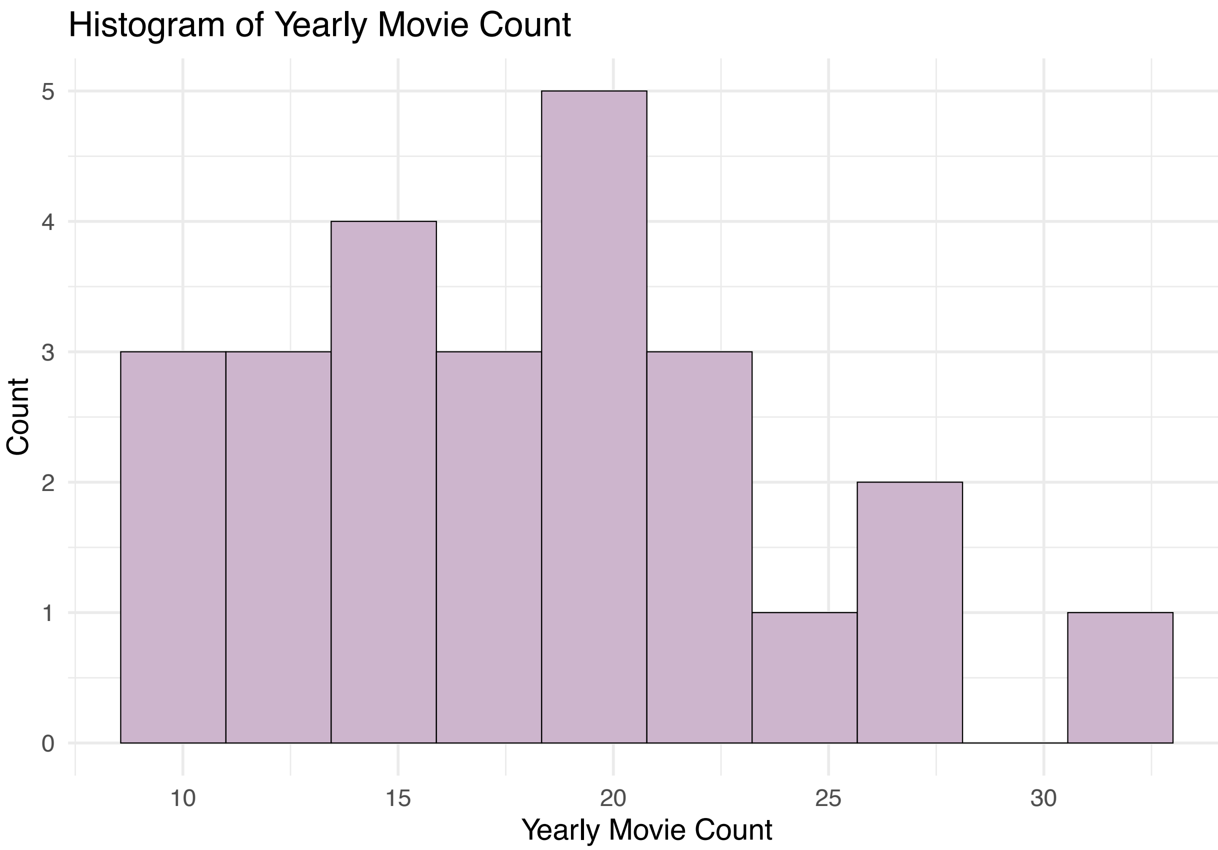



```
shapiro.test(resid(mod.full))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  resid(mod.full)
## W = 0.98134, p-value = 0.9102
```

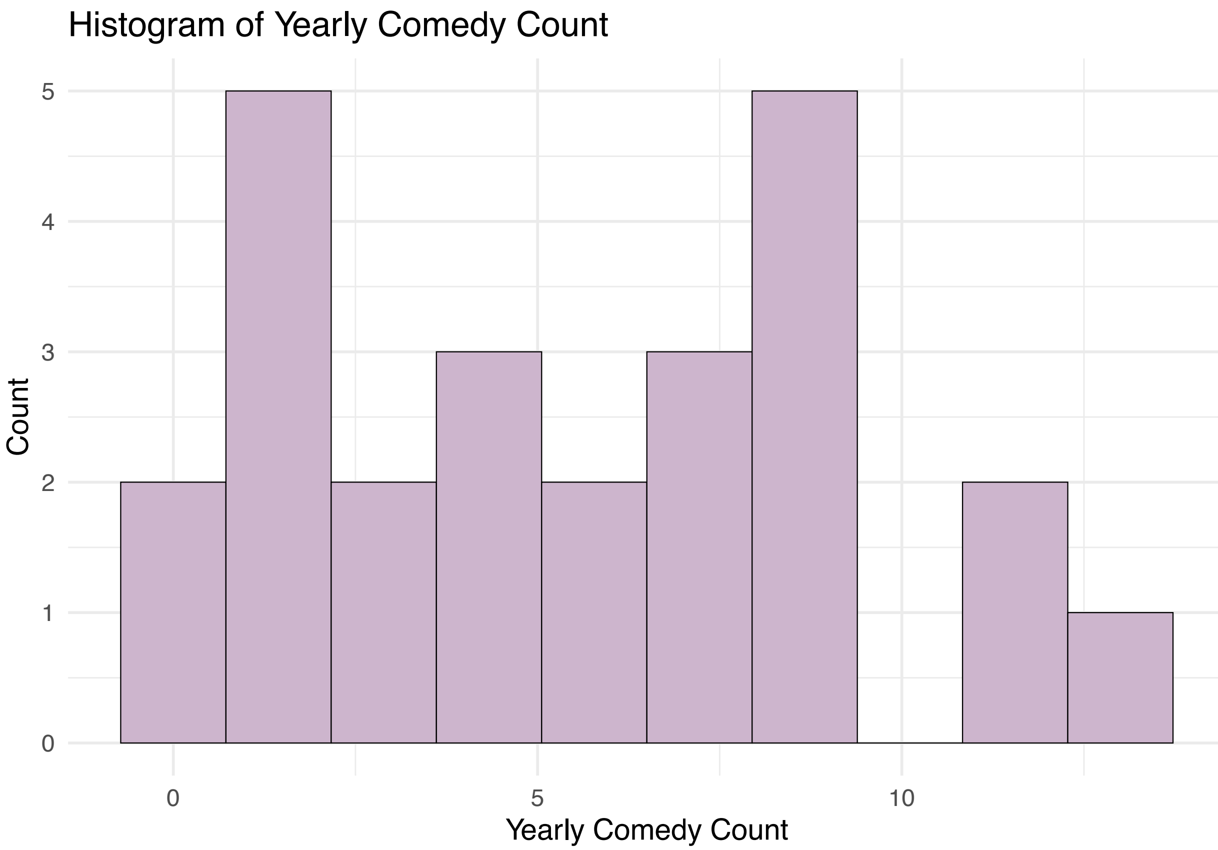
Histogram of movie count

```
yearly.summary2 |>
  ggplot(aes(x = movie_count)) +
  geom_histogram(bins = 10, fill = "thistle3", color = "black", size = 0.2)+
  ggtitle("Histogram of Yearly Movie Count") +
  theme_minimal() + labs(x = "Yearly Movie Count", y = "Count")
```




Histogram of comedy count

```
yearly.summary2 |>
  ggplot(aes(x = comedy)) +
  geom_histogram(bins = 10, fill = "thistle3", color = "black", size = 0.2)+
  ggtitle("Histogram of Yearly Comedy Count") +
  theme_minimal() + labs(x = "Yearly Comedy Count", y = "Count")
```




## Appendix F: What is Disney's expected total revenue in a year where they release 10 movies and 2 of them are comedies?

```
new = data.frame(comedy = 2, movie_count = 10)
prediction = predict(mod.full,new,interval = "prediction", level = 0.95)
prediction
```

```
##        fit      lwr     upr
## 1 44071.91 32500.31 55643.5
```