Disney Movie Success

Andres Gonzalez, Christina Lopez and Katya Mora

2022-12-17

# Appendix A: Data Cleaning

library(tidyverse)  
library(skimr)  
library(dplyr)  
library(gt)  
library(scales)  
library(broom)

disney <- read\_csv(  
 paste('https://raw.githubusercontent.com/stinalindaa/',  
 'disneymovies/main/disney\_movies\_total\_gross.csv', sep = ""))  
   
skim\_without\_charts(disney)

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 × 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)

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 |
| 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 × 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 × 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

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 × 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

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.

head(yearly.summary2)

## # A tibble: 6 × 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)

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**

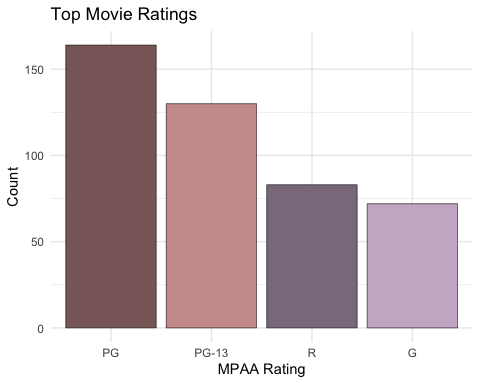
| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| genre | 0 | 1 | FALSE | 12 | Com: 138, Adv: 109, Dra: 89, Act: 32 |
| rating | 0 | 1 | FALSE | 4 | PG: 164, PG-: 130, R: 83, G: 72 |

**Variable type: numeric**

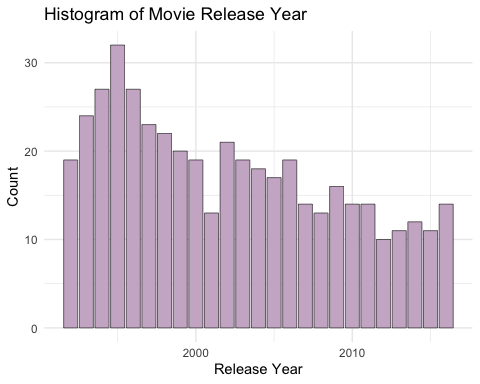
| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| index | 0 | 1 | 351.10 | 1.32770e+02 | 116.00 | 237.00 | 352.00 | 466.0 | 578.0 |
| gross | 0 | 1 | 98410523.66 | 1.19627e+08 | 2984.00 | 24267154.00 | 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 |

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.

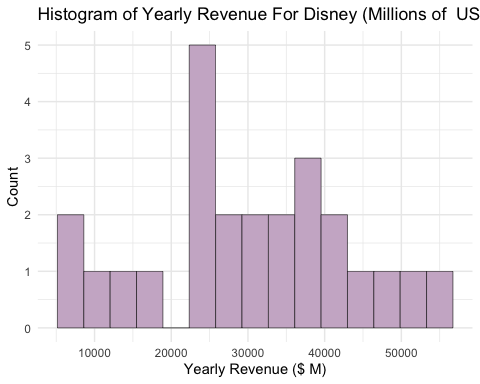


disney2 |>  
 select(release\_year) |>  
 filter(release\_year >= 1991) |>  
 group\_by(release\_year) |>  
 summarise(n = n()) |>  
 ggplot(aes(x = release\_year, y = n)) +   
 geom\_col(fill = "thistle3", color = "black", size = 0.2)+   
 ggtitle("Histogram of Movie Release Year") +   
 theme\_minimal() + 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 $")) %>%  
 tab\_source\_note(  
 source\_note = md("This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016")) %>%  
 tab\_caption(  
 caption = md("Source: Disney Character Success from Kaggle")) %>%  
 tab\_stubhead(label = md("Movies")) %>%  
 opt\_table\_font(font = google\_font("Mouse Memoirs"), weight = 100) %>%  
 cols\_label(genre = "Genre",  
 rating = "Rating",  
 gross\_million = "$ Gross") %>%  
 data\_color(  
 columns = gross\_million,  
 fn = scales::col\_numeric(  
 palette = "RdPu",  
 domain = c(1000, 480)))

Table 1: Summary of the <w:rPr><w:b w:val="true"></w:b></w:rPr>$ Gross Revenue Per Movie<w:rPr><w:b w:val="false"></w:b></w:rPr> 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 $")) %>%  
 tab\_source\_note(  
 source\_note = md("This file contains data on the Revenue and Gross of the Walt Disney Company from 1992 to 2016")) %>%  
 tab\_caption(  
 caption = md("Source: Disney Character Success from Kaggle")) %>%  
 tab\_stubhead(label = md("Movies")) %>%  
 opt\_table\_font(font = google\_font("Mouse Memoirs"), weight = 100) %>%  
 cols\_label(genre = "Genre",  
 rating = "Rating",  
 gross\_million = "$ Gross") %>%  
 data\_color(  
 columns = gross\_million,  
 fn = scales::col\_numeric(  
 palette = "magma",  
 domain = c(0, 0.8)))

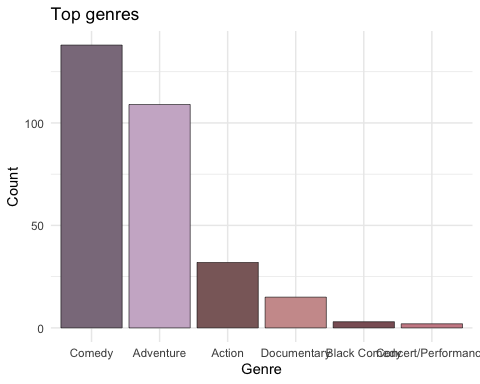
Table 1: Summary of the <w:rPr><w:b w:val="true"></w:b></w:rPr>Gross Revenue Per Movie<w:rPr><w:b w:val="false"></w:b></w:rPr> from 1992 to 2016

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()) |>  
 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:

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:

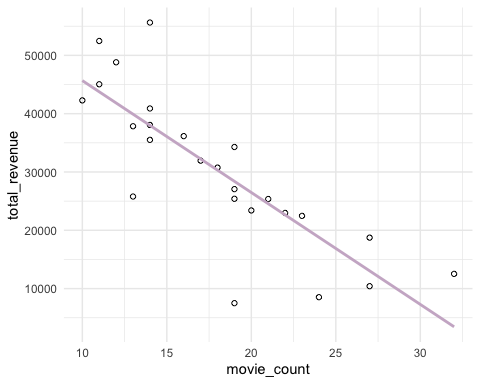
add1(mod.full, ~.+comedy\*movie\_count, test = 'F')

## 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:

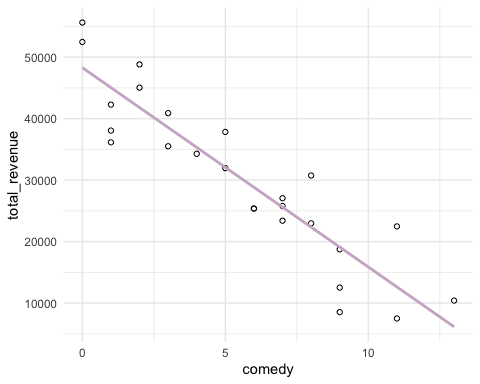
yearly.summary2 |>  
 ggplot(aes(x = movie\_count, y = total\_revenue)) +  
 geom\_point(shape = 21, color = "black") +   
 geom\_smooth(color = "thistle3", method = "lm", se = FALSE) +   
 theme\_minimal()

## `geom\_smooth()` using formula = 'y ~ x'

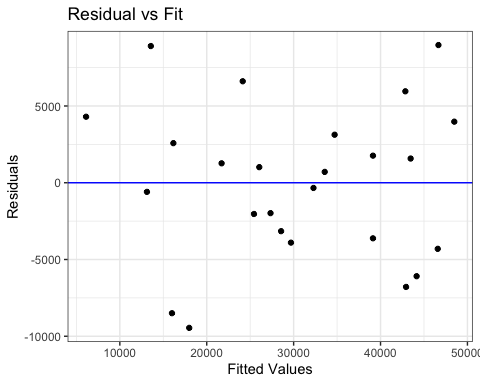


yearly.summary2 |>  
 ggplot(aes(x = comedy, y = total\_revenue)) +  
 geom\_point(shape = 21, color = "black") +   
 geom\_smooth(color = "thistle3", method = "lm", se = FALSE) +   
 theme\_minimal()

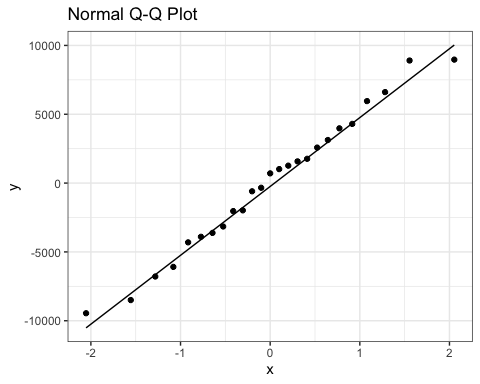
## `geom\_smooth()` using formula = 'y ~ x'



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()

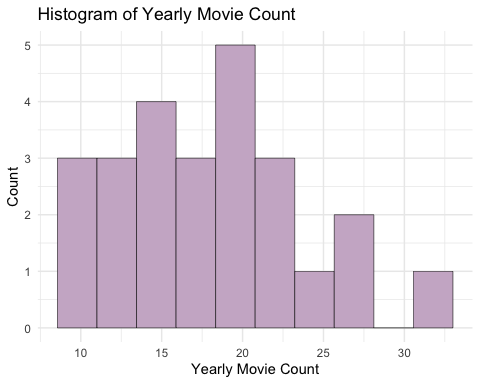


shapiro.test(resid(mod.full))

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

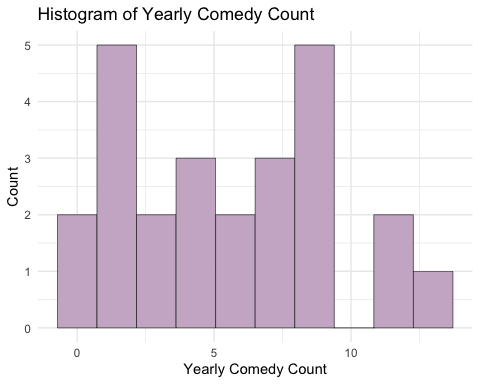
Histogram of movie count

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