

Disney Box Office Predictions

A multivariable analysis

Introduction

This project offered me an opportunity to explore an area that I have wanted to unveil since I was a kid. What makes a movie successful? In my time learning Data Science, I have gathered the tools to understand statistical analysis, R programming, and Python programming. Therefore, why not blend my old passions with the new? I decided to pair one of my most treasured memories of watching classic Disney films with a breakdown of which factors make a successful film. Success can be indicated in a number of ways, such as pop culture relevance, critic ratings, and financial gain. I decided to make financial gain the marker of success so I made box office revenue (in dollars) the dependent variable.

Kaggle was an incredible asset in finding the type of dataset I was looking for. I found datasets with the sorts of factors that I wanted to assess, such as actors, directors, music, budget, distribution, critic scores, runtime, and release date. However, I ran into a problem as I had previously heard that audience ratings (G, PG, and PG-13) were also a critical component that studios take into account as they can limit potential ticket sales when a film is in theaters. Even though I wanted to analyze this variable, it was not available in the dataset I looked for. Thus, I kept searching and ran across another dataset listing Disney plus titles that included the rating. These two consolidated datasets served as the basis of my analysis.

Data Explanation and Cleaning

I began with two datasets, DisneyMoviesDataset.csv and disney_plus_titles.csv that I consolidated into one CSV file. I only wanted to extract the rating column from disney_plus_titles.csv and input them into DisneyMoviesDataset.csv if they matched by title. Since disney_plus_titles.csv contained the entire Disney Plus catalog I was able to acquire the ratings for all of the titles in the DisneyMoviesDataset.csv document. In addition, I took

out the columns that did not appear to be relevant to the Box Office value and which had more than half of the data cells blank, such as Country, Language, and Screenplay. After I did this I then removed observations that contained any N/A and empty cell values. After this, I split the [release date] into two sections, Month and Year, this way I could analyze those two as independent variables.

The next part that needed cleaning was the directed by, starring, music by, and distribution columns. Each of these variables contained an array with strings in each element. For simplicity's sake, I decided to remove any element after the first. This way I could keep the top-billed actor in a film and its first listed director. My reasoning was that it would simplify the analysis process and in film, the top star is usually in the center of the cover and has their name listed at the top of the film poster. This means many films are marketed with the main star and director in mind. Furthermore, the data was filtered to only include observations from 1968 and later. This is because Disney film [demographics](#) show that 92% of Disney viewers are 54 years old and younger. So the year 1968 was picked as the year to begin the analysis as many of the films shown in the late 60s might have been formative to adults now who are Disney consumers with their families. Once this process was complete the new CSV file was exported and titled Clean_Disney_data.csv (**See Figure 17 for Code**).

The CSV file Clean_Disney_data was then loaded into R studio and the columns were renamed to make them easier to reference. The columns were ("title","run_time","budget","box_office","imdb","metascore","rotten_tom","director","actor","music","distr","rating","year","month). The Rotten Tomatoes score (critic rating) was then converted from a percentage to a number (e.g. 90% to .9).

The data was then sorted in descending order by Box Office value. When this was done, there was a noticeable pattern that film Box Office numbers were increasing linearly by year (**See Figure 19 for graph**). While on paper is an indicator of a relationship, it would mean that attending the movie theater is more popular than ever. However, ever since the Covid-19 pandemic and the rise of streaming popularity this is very unlikely. The conclusion of this is that there is likely such a relationship that is caused by inflation. Thus, a film that earned 15,000,000 in 1970 would have earned \$115,094,458.76 in 2022 based on inflation. This information was gathered using an [online calculator](#) which derives its formula from the Consumer Price Index gathered from *The U.S. Labor Department's Bureau of Labor Statistics*. This

online calculator was also used to create a table that had two variables: the Year and the Inflation conversion rate. This table was included in the file 2022_Inflation_Conversion.csv which was opened in R studio. A *for* loop was used to create an array of data that matched the years in the two tables and appended the inflation conversion rate to the Disney table. Then two more columns were created in the Disney Table by multiplying the conversion rate by the nominal dollar value of the box office and budget column. The new columns were titled, " Inflation_budg, Inflation_office" (**See Figure 22**).

The Disney table was then checked for outliers and the only outlier that seemed reasonable to remove was one that was found in the Box Office variable. It had a value of \$36 when the next value higher was \$2,800,000.

After these steps were taken there was still one more step in the cleaning process that needed addressing. The categorical variables in the actor, director, music, rating, and distribution categories were still strings. Furthermore, the categories actor, director, and music had a multitude of people in each category. Thus, it could be extremely impractical to place each of these different individuals in a category. Thus, a different approach was taken, the top 5 most frequent actors, directors, music composers, and distribution studios were pulled from the dataset and were turned into dummy variables. The reasoning here is that Disney must have found success with these individuals which they rehired frequently in their projects (**See Figure 19**).

The dataset began with 23 variables and 430 observations. **After cleaning the dataset, these were whittled down to 13 features and 161 observations.**

Exploratory Data Analysis

When analyzing the components of the Disney data the relationship between budget and box office was very apparent. When the two were measured with a single linear regression there was an R^2 of .528 and a P-value of 2×10^{-16} . This demonstrated a statistically significant indication that on average the budget shares a relationship with Box Office value. (**See figure 5**).

Furthermore, when the relationship between the box office and the budget was explored for both the inflation and nominal values. It was revealed that the nominal variables had a higher R^2 (.528) while the inflationary variables had an R^2 value of (0.4049) (**See Figure 5**). While this might indicate that the relationship is stronger between nominal budget and nominal box office, it is likely that the inflation relationship reveals that box office and budget actually have a weaker relationship than initially anticipated in the real world.

Another observation that stood out was that when each of the critic platforms were compared to Inflation Box Office, Imdb showed the highest R^2 at .3085. However, as will be discussed later, Metascore was selected by the subset predictor before Imdb. This means that IMDb has a stronger correlation with Inflation Box Office in an SLR, but when paired up with other variables in an MLR with other critic platforms, Imdb turns out to be the last out of the three to be selected (**See figure 5 and Figure 1**).

The month that also had the highest box office releases was June. This falls in line with summer which is when studios know that a large percentage of their demographic, grade schoolers are free during the day. So they have more time to invest in going to the movies (**See Figure 18**).

When building the model my expectation was that the star power of an actor like Johnny Depp or Tom Hanks would be one of the strongest indicators of a large box office value. As these two actors have starred in franchises such as Toy Story and Pirates of the Caribbean which were some of the highest-grossing Disney films in general. However, during the model selection process the only person that was selected in the subset before the Adjusted R^2 plateaued was Music Producer Hans Zimmer. This means that he has a statistically strong relationship with the Box Office value. (**See Figure 13**)

Model Building

There were several iterations of models that were designed in order to achieve the most optimal result for the Y Inflation Box Office (benchmarked to 2022).

When designing the initial model all of the variables were included including, inflation budget, critic platforms, dummy variables (actors, directors, music producers, and distribution productions), year, and months. When these variables were passed into the subset selector the variables that came back before the Adjusted R^2 plateaued was $X=4$ ($R^2a=0.6215402$) (**See Figure 13 and Figure 4**). These variables were Inflation Budget, Metascore, Rotten Tomatoes, and Hanz Zimmer. These variables were modeled and stored in an MLR called `Inf_model`.

Each of the 4 variables was then checked for quadratic terms using ANOVA and `Inf_model` as a comparison. Using this method it was concluded that Metascore and Rotten Tomatoes had a statistically significant polynomial relationship to Inflation Box Office. Then these four variables were also checked for interactions with each other. Many of these variables had a P value under .05 when compared with the `Inf_Model`. So they were selected to be the next rounds of subset selection. It should be noted that despite not being included in 4 variables for `Inf_model`, IMDb was also tested for interaction with Budget. This is because during the EDA stage of the analysis IMDb had the highest R^2 value (.3085) out of every critic platform when modeled in an SLR with Box Office as the Y. (**See Figure 2**). This model had an R^2a of .621 (**See Figure 13**)

Model 1:

```
modsel_Inf <- lm(Inflation_office ~ Inflation_budg+ metascore + rotten_tom + hz)
```

When the final variables were run through the second model subset selection the number of variables that indicated a significant increase in Box Office was $X=4$ (**See Figure 15**). The model that was selected had an adjusted R^2 of 0.665. This model included an interaction term between (inflation) budget and rotten tomatoes and both Metascore and rotten tomatoes had a polynomial relationship with (inflation) box office (**See Figure 14**). This model was then compared with the first model using ANOVA and had a P-value of = .0000749 (**See Figure 11**)

Model 2:

```
modsel_Inf2 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore, 2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)
```

After the second version of my model was completed I checked for observations with more than twice the mean leverage and identified 16 observations. When checking the outliers from the data by checking for studentized residuals I found 9 observations that were outliers. Then I checked for influence using Cook's distance values and removed 12 observations in order to see if this would increase the model's R^2 and lower SE (**See Figure 23**). Despite removing these observations the adjusted R^2 stayed the same (**See Figure 12**). Thus, we found that model 2 was the best at modeling (Inflation) Box Office with an R^2 of 0.665 and a Residual Standard Error of 25530000.

Final Model: `model_Inf2 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore, 2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)`

Interpretation

It appears that the most statistically significant variables in predicting Inflation Box Office values are the interaction of the inflated budget and Rotten Tomatoes variables, the quadratic of Metascore, cubic of rotten tomatoes, and Hanz Zimmer. This was apparent as there were many tests that narrowed down this conclusion. When I ran the ANOVA function this function showed a P-Value of 7.49e-05 compared to the initial model. This means that this model is the best at utilizing the variables from the Disney dataset in predicting the inflated box office value. However, the residual standard error of 25530000 which indicates that the model might be very flawed at accurately predicting future box office values.

Conclusion

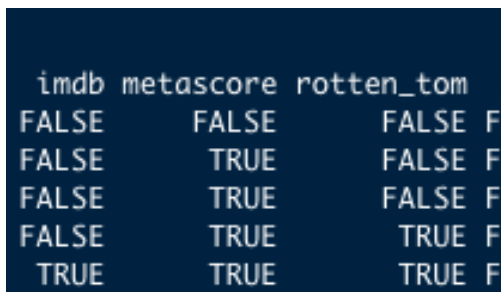
Despite attaining an R^2 of 0.665 from the final model. The score of .665 is not very high at all so it shows a moderately positive relationship between my predictors and outcome. This is in part because when selecting the subsets for model selection I selected the variables up to the "elbow" of the adjusted R^2 graph, maybe if I selected more variables the adjusted R^2 would have increased by a small degree, however, it likely would have been negligible. These results also go to show that predicting the effect and factors that will make a successful film is extremely difficult. My initial theory that ratings would be relevant in

predicting the Box Office value was also very wrong as none of the subsets selectors picked ratings.

This is why studios spend an exorbitant amount of money on focus groups, marketing, franchising, and award lobbying to create financial success. In addition, not every variable that could possibly assist in my predictions was available, some of the other variables that could have had a statistical significance included box office, sequel (boolean), screenplay, and Pixar (boolean). The scope of this dataset was also very limited as there are other studios that might have different results because their demographics differ entirely. I wish to expand upon the model in future iterations of this analysis.

Appendix:

Figure 1 (Adjusted R² of Critics Platforms):



imdb	metascore	rotten_tom	
FALSE	FALSE	FALSE	F
FALSE	TRUE	FALSE	F
FALSE	TRUE	FALSE	F
FALSE	TRUE	TRUE	F
TRUE	TRUE	TRUE	F

Figure 2 (Quadratic Terms):

```
#-----Check for Quadratic terms-----|  
  
#Inflation buget  
lin <- lm(Inflation_office ~ Inflation_budg + metascore + rotten_tom + hz)  
x1 <- Inflation_budg  
x2 <- Inflation_budg^2  
quad <- lm(Inflation_office ~ metascore + rotten_tom + hz + x1 + x2)  
anova(lin,quad)  
#Linear  
  
#meta score  
quad <- lm(Inflation_office ~ Inflation_budg + poly(metascore,2,row=TRUE) + rotten_tom + hz)  
cube <- lm(Inflation_office ~ Inflation_budg + poly(metascore,3,row=TRUE) + rotten_tom + hz)  
anova(Inf_model, quad)  
anova(quad,cube)  
#add meta score^2  
  
#Rotten tom  
quad <- lm(Inflation_office ~ Inflation_budg + poly(rotten_tom,2,row=TRUE) + metascore + hz)  
cube <- lm(Inflation_office ~ Inflation_budg + poly(rotten_tom,3,row=TRUE) + metascore + hz)  
anova(Inf_model,quad)  
anova(Inf_model,cube)  
anova(quad,cube)  
#add rotten tom^3
```


Figure 3 (Plot SLR Relationships):

```
#Budget vs Rotten Tomatoes
plot(disney$Inflation_budg,disney$rotten_tom,xlab="Inflation Budget",ylab="Metascore", main= "Inflation Budget vs MetaScore")
slr <- lm(Inflation_budg ~ rotten_tom)
summary(slr)
#R^2 0.04174

#Budget vs Metascore
plot(disney$Inflation_budg,disney$metascore,xlab="Inflation Budget",ylab="Metascore", main= "Inflation Budget vs MetaScore")
slr <- lm(Inflation_budg ~ metascore)
summary(slr)
#r2 = .05144

#Budget vs IMDb
plot(disney$Inflation_budg,disney$imdb,xlab="Inflation Budget",ylab="Imdb", main= "Inflation Budget vs Imdb")
slr <- lm(Inflation_budg ~ imdb)
summary(slr)
#r2 = 0.1165
```

Figure 4 (Model 1 Subset Selection):

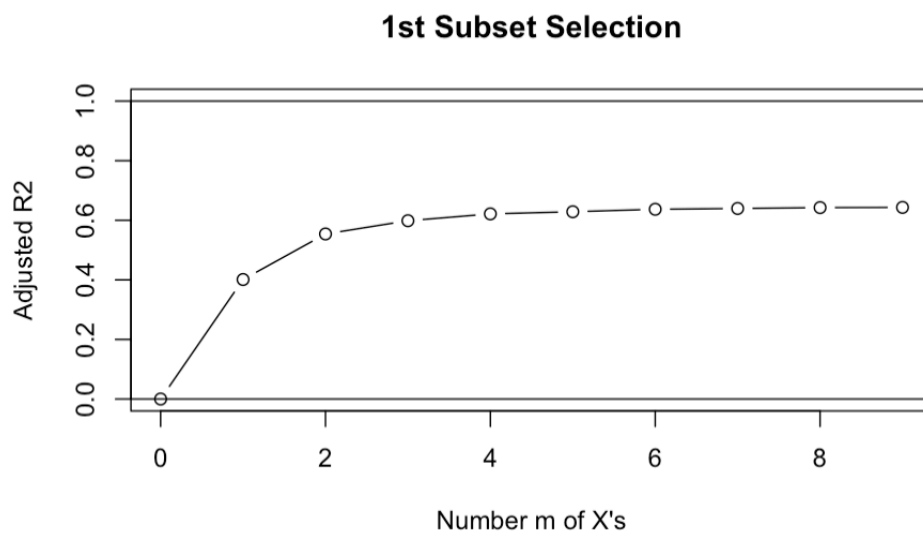


Figure 5 (Exploratory Data Analysis Code):

```
#-----EDA -----

#Nominal data- Budget vs Box Office
plot(disney$budget,disney$box_office,xlab="Budget",ylab="Box Office", main= "Box Office vs Year")
slr <- lm(box_office~ budget)
summary(slr)
#r2 = .528

#Inflation data - Budget vs Box Office
plot(disney$Inflation_budg,disney$Inflation_office,xlab="Inflation Budget",ylab="Box Office Inflation", main= "Box Office Infaltion vs Budget")
slr <- lm(Inflation_office~ Inflation_budg)
summary(slr)
#r2 = 0.4049

#Months vs Box Office
plot(disney$month,disney$Inflation_office,xlab="Months",ylab="Box Office", main= "Box Office vs Months")

#Run Time vs Inflation box Office
plot(disney$run_time,disney$Inflation_office,xlab="Run Time",ylab="Box Office", main= "Inf Box Office vs Run Time")
slr <- lm(Inflation_office~ run_time)
summary(slr)
#r2 = 0.05942

#Year vs Box Office
plot(disney$year,disney$box_office,xlab="Year",ylab="Box Office", main= "Box Office vs Year")
slr <- lm(box_office~ year)
abline(slr,col="red")
summary(slr)
#r2 = 0.1509

#Metascore vs Box Office |
plot(disney$metascore,disney$Inflation_office,xlab="Metascore",ylab="Box Office", main= "Box Office vs Metascore")
slr <- lm(Inflation_office~ metascore)
summary(slr)
#r2 = 0.2783

# IMDb vs Box Office
plot(disney$IMDB,disney$Inflation_office,xlab="Imdb",ylab="Box Office", main= "Box Office vs ImDb")
slr <- lm(Inflation_office~ imdb)
summary(slr)
#r2 = .3085

#Rotten Tomatoes
slr <- lm(Inflation_office~ rotten_tom)
summary(slr)
#r2 = 0.1792

#Inf Budget vs Inf Box
plot(disney$Inflation_budg,disney$Inflation_office,xlab="Inflation Budget",ylab="Inflation Box Office", main= "Inf Box Office vs Inf Budget")
slr <- lm(Inflation_office~ Inflation_budg)
summary(slr)
#r2 =0.4049

##Budget vs Year
plot(disney$year,disney$Inflation_budg,xlab="Year",ylab="Inflation Budget", main= "Year vs Inflation Budget")
slr <- lm(Inflation_budg~ year)
summary(slr)
#r2 = 0.1771

#Budget vs Rotten Tomatoes
plot(disney$Inflation_budg,disney$rotten_tom,xlab="Inflation Budget",ylab="Metascore", main= "Inflation Budget vs MetaScore")
slr <- lm(Inflation_budg ~ rotten_tom)
summary(slr)
#R^2 0.04174

#Budget vs Metascore
plot(disney$Inflation_budg,disney$metascore,xlab="Inflation Budget",ylab="Metascore", main= "Inflation Budget vs MetaScore")
slr <- lm(Inflation_budg ~ metascore)
summary(slr)
#r2 = .05144

#Budget vs IMDb
plot(disney$Inflation_budg,disney$imdb,xlab="Inflation Budget",ylab="Imdb", main= "Inflation Budget vs Imdb")
slr <- lm(Inflation_budg ~ imdb)
summary(slr)
#r2 = 0.1165
```

Figure 6:

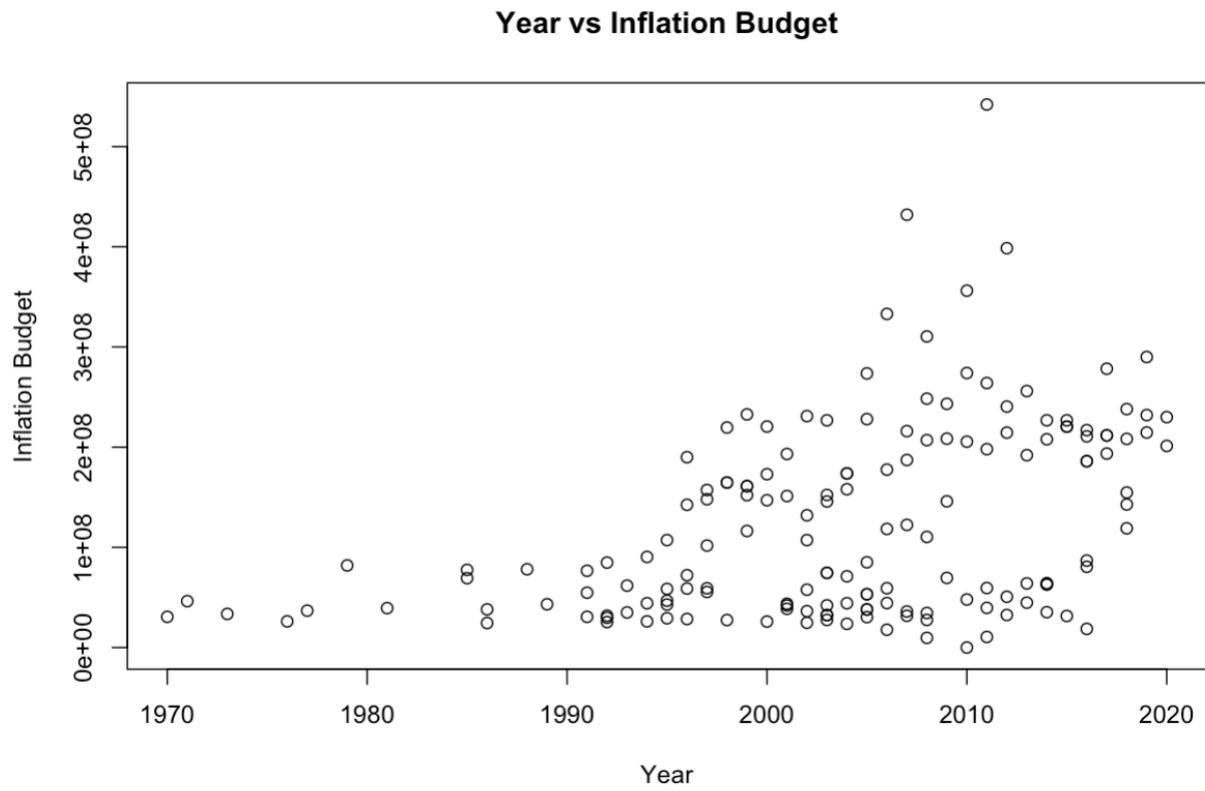


Figure 7:

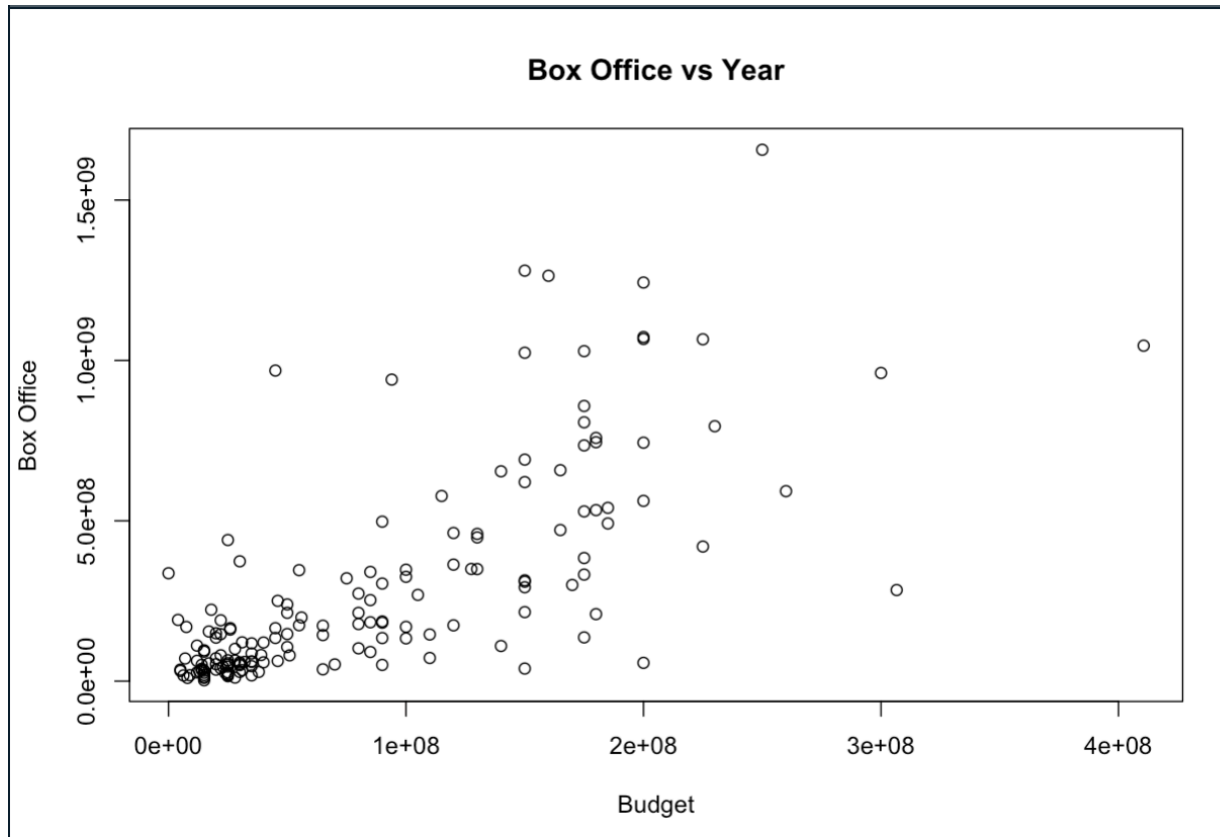


Figure 8:

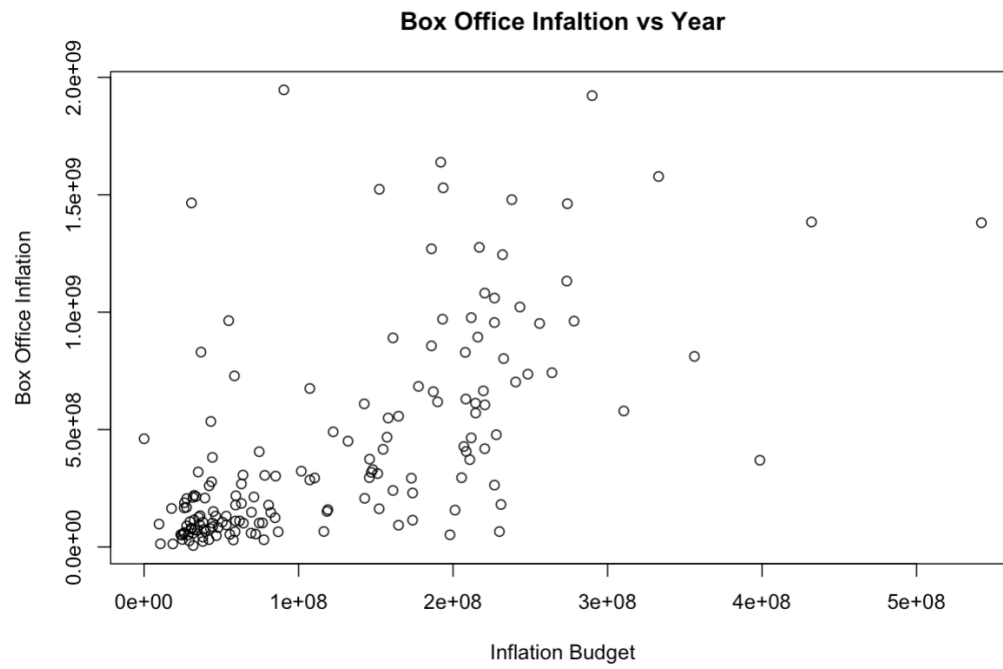


Figure 9 (Interactions Terms):

```
#-----Interactions -----  
  
#Interactions subgroups  
  
#Metascore and Inf Budg  
meta_rot <- rotten_tom*metascore  
mlr_int <- lm(Inflation_office ~ Inflation_budg*metascore + rotten_tom + hz)  
anova(Inf_model, mlr_int)  
#probability that interaction term is better EX:#We conclude that the relationship between mpg and weight depends on transmission  
# type.  
#Yes  
#P = 0.01471  
  
#Budget and Rotten Tom  
meta_bug <- Inflation_budg*metascore  
mlr_int <- lm(Inflation_office ~ Inflation_budg*rotten_tom + hz + metascore)  
anova(Inf_model, mlr_int)  
#Yes  
# P= 0.01198  
  
#Budget Imdb  
mlr_int <- lm(Inflation_office ~ Inflation_budg*imdb + metascore + rotten_tom + hz)  
anova(Inf_model, mlr_int)  
#Yes  
#P = 0.0009448  
  
#MetaScore and Rotten Tom  
mlr_int <- lm(Inflation_office ~ Inflation_budg + metascore*rotten_tom + hz)  
anova(Inf_model, mlr_int)  
#Yes  
#p = 0.0001144  
  
#Combination of the lowest p= scores  
mlr_int <- lm(Inflation_office ~ Inflation_budg*imdb + metascore*rotten_tom + hz)  
anova(Inf_model, mlr_int)  
#P = 3.117e-05
```

Figure 10 (Creating Dummy Variables):

```
#Directors
#Ron Clements
#John Lasseter
#Jon Turteltaub
#Andrew Stanton
#Wolfgang Reitherman

rc <- as.numeric(disney$director=="Ron Clements")
jl <- as.numeric(disney$director=="John Lasseter")
jt <- as.numeric(disney$director=="Jon Turteltaub")
as <- as.numeric(disney$director=="Andrew Stanton")
wr <- as.numeric(disney$director=="Wolfgang Reitherman")

#Music Producers
#Randy Newman
#Michael Giacchino
#Alan Menken
#John Debney
#Hans Zimmer

rw <- as.numeric(disney$music=="Randy Newman")
mg <- as.numeric(disney$music=="Michael Giacchino")
am <- as.numeric(disney$music=="Alan Menken")
jd <- as.numeric(disney$music=="John Debney")
hz <- as.numeric(disney$music=="Hans Zimmer")

#Actors
#Johnny Depp
#Tom Hanks
#Tim Allen
#Owen Wilson
#Nicolas Cage

jdd <- as.numeric(disney$actor=="Johnny Depp")
th <- as.numeric(disney$actor=="Tom Hanks")
ta <- as.numeric(disney$actor=="Tim Allen")
ow <- as.numeric(disney$actor=="Owen Wilson")
nc <- as.numeric(disney$actor=="Nicolas Cage")
```

Figure 11 (Model 1 vs Model 2 Output):

```
> anova(modsel_Inf,modsel_Inf2)
Analysis of Variance Table

Model 1: Inflation_office ~ Inflation_budg + metascore + rotten_tom +
  hz
Model 2: Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore,
  2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz
  Res.Df    RSS Df Sum of Sq    F    Pr(>F)
1     156 1.1484e+19
2     153 9.9696e+18   3 1.5148e+18 7.749 7.49e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Figure 12 (Model 2 vs Model 3 Output):

```
> anova(modsel_Inf2,modsel_Inf3)
Analysis of Variance Table

Model 1: Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore,
  2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz
Model 2: Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore,
  2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz
  Res.Df    RSS Df Sum of Sq    F    Pr(>F)
1     142 7.5831e+18
2     142 7.5831e+18   0      0
>
```


Figure 13 (Model 1 Code):

```
(Top 5 influential features)
#------(Inflation) Model 1 -----
modsel_Inf <- regsubsets(Inflation_office ~ run_time + Inflation_budg + budget +
                        imdb + metacore + rotten_tom + year + month +
                        g + pg + pg_13 + bvp + wds + wdsmp + bvdp + bvd +
                        jl + jas + jt + bb + gt + rw + mg + am + jd + hz + jdd
                        + th + ta + jc + nc,disney,nbest=1)

vars<-summary(modsel_Inf)$which
vars
# R2a spot plot:
r2a <- summary(modsel_Inf)$adjr2
r2a
plot(0:9,c(0,r2a),type="b",ylim=c(0,1),xlab="Number m of X's",ylab="Adjusted R2")
abline(h=c(0,1))

modsel_Inf <- lm(Inflation_office ~Inflation_budg+ metacore + rotten_tom + hz)
summary(disney)
#Elbow shows at 4 variables ( Inflation_budg, metacore, rotten_tom, hanz zimmer )
#R2a = 0.6215402
```

Figure 14 (Model 2 Code):

```
#------(Inflation) Model 2 -----
modsel_Inf2 <- regsubsets(Inflation_office ~ run_time + Inflation_budg + budget +
                        imdb + poly(metacore,2,row=TRUE) + poly(rotten_tom,3,row=TRUE) +
                        Inflation_budg*imdb + metacore*rotten_tom + Inflation_budg*rotten_tom+
                        year + month + g + pg + pg_13 + bvp + wds + wdsmp + bvdp + bvd +
                        jl + jas + jt + bb + gt + rw + mg + am + jd + hz + jdd
                        + th + ta + jc + nc,disney,nbest=1)

vars<-summary(modsel_Inf2)$which
vars
# R2a spot plot:
r2a <- summary(modsel_Inf2)$adjr2
r2a
plot(0:9,c(0,r2a),type="b",ylim=c(0,1),xlab="Number m of X's",ylab="Adjusted R2")
abline(h=c(0,1))

#Updated Model
modsel_Inf2 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metacore, 2, row = TRUE) + poly(rotten_tom, 3, row = TRUE) + hz)
summary(modsel_Inf2)
#New R2a =0.665

anova(modsel_Inf,modsel_Inf2)
```

Figure 15 (2nd Subset Selection):

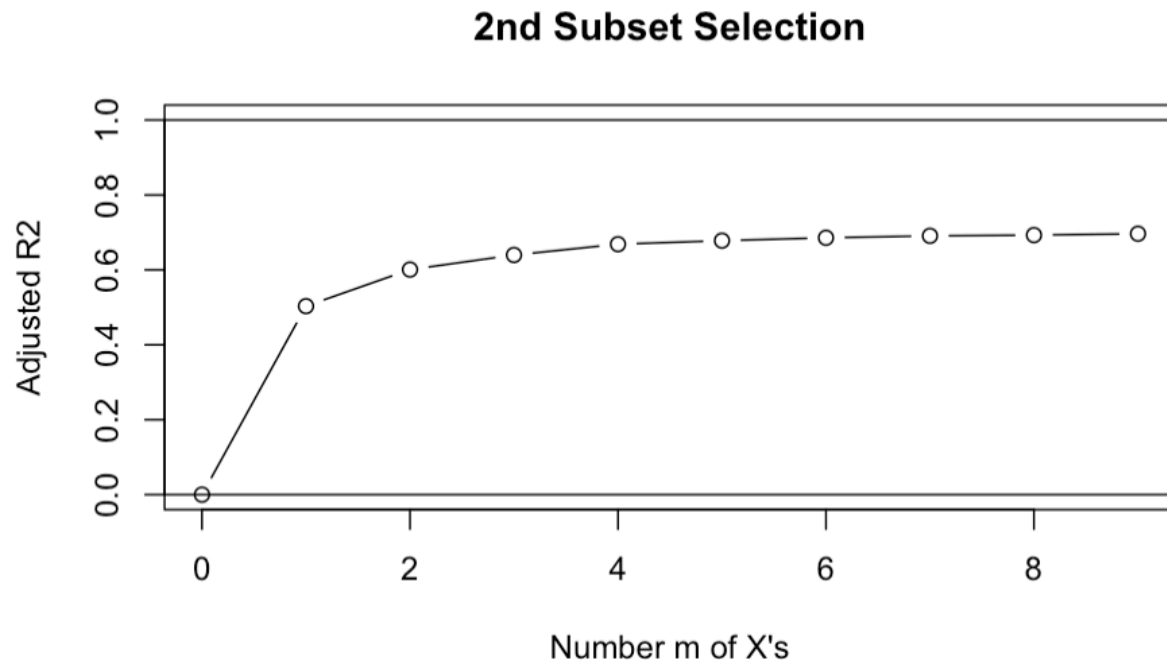


Figure 16 (Summary of Model 2):

```
> model_Inf2 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore, 2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)
> summary(model_Inf2)
```

Call:
lm(formula = Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore,
2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)

Residuals:

	Min	1Q	Median	3Q	Max
	-716760839	-135864968	-15962549	71552718	1262447241

Coefficients:

	Estimate	Std. Error	t value	Pr(> t)
(Intercept)	2.203e+08	3.335e+08	0.661	0.5097
poly(metascore, 2, raw = TRUE)1	-1.884e+07	1.280e+07	-1.471	0.1433
poly(metascore, 2, raw = TRUE)2	2.551e+05	9.808e+04	2.601	0.0102 *
poly(rotten_tom, 3, raw = TRUE)1	2.283e+09	1.149e+09	1.988	0.0486 *
poly(rotten_tom, 3, raw = TRUE)2	-5.432e+09	2.251e+09	-2.413	0.0170 *
poly(rotten_tom, 3, raw = TRUE)3	3.114e+09	1.427e+09	2.181	0.0307 *
hz	6.043e+08	1.010e+08	5.984	1.48e-08 ***
Inflation_budg:rotten_tom	3.124e+00	3.542e-01	8.822	2.38e-15 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 255300000 on 153 degrees of freedom
Multiple R-squared: 0.6797, Adjusted R-squared: 0.665
F-statistic: 46.38 on 7 and 153 DF, p-value: < 2.2e-16

Figure 17 (Initial Data Consolidation and cleaning):

```
import warnings
warnings.simplefilter(action='ignore', category=FutureWarning)
import pandas as pd
import matplotlib.pyplot as plt

#-----Data Consolidation-----

df_plus = pd.read_csv(r'disney_plus_titles.csv')
df_box = pd.read_csv(r'DisneyMoviesDataset.csv')
disney = pd.merge(df_box,
                  df_plus[['title', 'rating']],
                  on='title')
disney.to_csv(r'Disney_data.csv', index=False)
#-----Data Manipulation Methods -----

#Read the name of columns of dataframe
def read_col(df):
    for col in df.columns:
        print(col)

#-----Loading Data-----
disney = pd.read_csv(r'Disney_data.csv')
disney = disney.set_index('title')
#-----Cleaning Data-----

# Drop unnecessary columns
disney = disney[["Running time (int)", "Budget (float)", "Box office (float)", "Release date (datetime)",
               "imdb", "metascore", "rotten_tomatoes", "Directed by", "Starring", "Music by", "Distributed by", "rating"]]

#Remove row with empty cells
disney = disney.dropna()
print(disney)

#Formating data correctly (Removing brackets and dashes)

#Directed by
disney['Directed by'] = disney['Directed by'].str.replace('[', '').str.split(",").str[0]
disney['Directed by'] = disney ['Directed by'].str.replace(']', '')

#Starring
disney['Starring'] = disney['Starring'].str.replace('[', '').str.split(",").str[0]
disney['Starring'] = disney ['Starring'].str.replace(']', '')

#Music by
disney['Music by'] = disney['Music by'].str.replace('[', '').str.split(",").str[0]
disney['Music by'] = disney ['Music by'].str.replace(']', '')

#Distributed by
disney['Distributed by'] = disney['Distributed by'].str.replace('[', '').str.split(",").str[0]
disney['Distributed by'] = disney ['Distributed by'].str.replace(']', '')

#Seperating Months and year as two variables
disney_append = disney[['Release Year', 'Release Month', 'Release Day']] = disney['Release date (datetime)'].str.split('-', expand=True)
disney.join(disney_append)
disney = disney.drop(['Release date (datetime)', 'Release Day'], axis=1)
disney.to_csv(r'New_Disney_data.csv', index=True)

#Drop films made before 1968
disney.drop(disney[disney['Release Year'].astype(int) < 1968].index, inplace = True)

print("-----Dataset information-----")
print(disney)
read_col(disney)

#Save CSV file
disney.to_csv(r'Clean_Disney_data.csv', index=True)

#-----Data Analysis-----
```


Figure 19:

```
# ----- DATA LOADING -----
disney <- read.csv("Clean_Disney_data.csv")
attach(disney)

#-----Data Cleaning -----

#Renaming Variables and sorting by Box Office in Decending Order
names(disney) <- c("title", "run_time", "budget", "box_office", "imdb", "metascore", "rotten_tom", "director", "actor", "music", "distr", "rating", "year", "month")

#-----Converting Percentages-----
disney$rotten_tom <- as.numeric(sub("%","",disney$rotten_tom))/100
str(disney$rotten_tom)

#-----Appending inflation columns (Box Office and Budget)

conversion <- read.csv("2022_Inflation_Conversion.csv")#dataframe created using online resources
disney <- disney[order(disney$year),] #Order data by year

#Append Inflation
conv <- c()
for (x in disney$year) {
  print(x)
  index <- which(conversion$Year == x)
  change <- conversion[index,"Conversion"]
  conv <- append(conv, change)
}
disney$Inflation <- conv

#Budget Adjusted for Inflation
disney$Inflation_budg <- disney$Inflation * disney$budget
#Box Office Adjusted for Inflation
disney$Inflation_office <- disney$Inflation * disney$box_office

str(disney)
plot(disney$year,disney$Inflation_office,xlab="Year",ylab="Box Office", main= "Box Office vs Year")

#Ordering data by box office
disney <- disney[order(-disney$box_office),]
disney$box_office

#-----Checking for outliers-----
#str(disney)
#checking if there are any outliers that might need to be removed
boxplot(disney$Inflation_budg)$out
boxplot(disney$run_time)$out
boxplot(disney$budget)$out
boxplot(disney$imdb)$out
boxplot(disney$Inflation_office)$out
boxplot(disney$metascore)$out

# Remove outlier from Box office ($36: Index 162)
index <- which.min(table(disney$box_office))

#output -> lowest value : 36
remove <- which(disney$box_office == 36)
disney <- disney[-remove,]
disney$box_office
```

Figure 21:

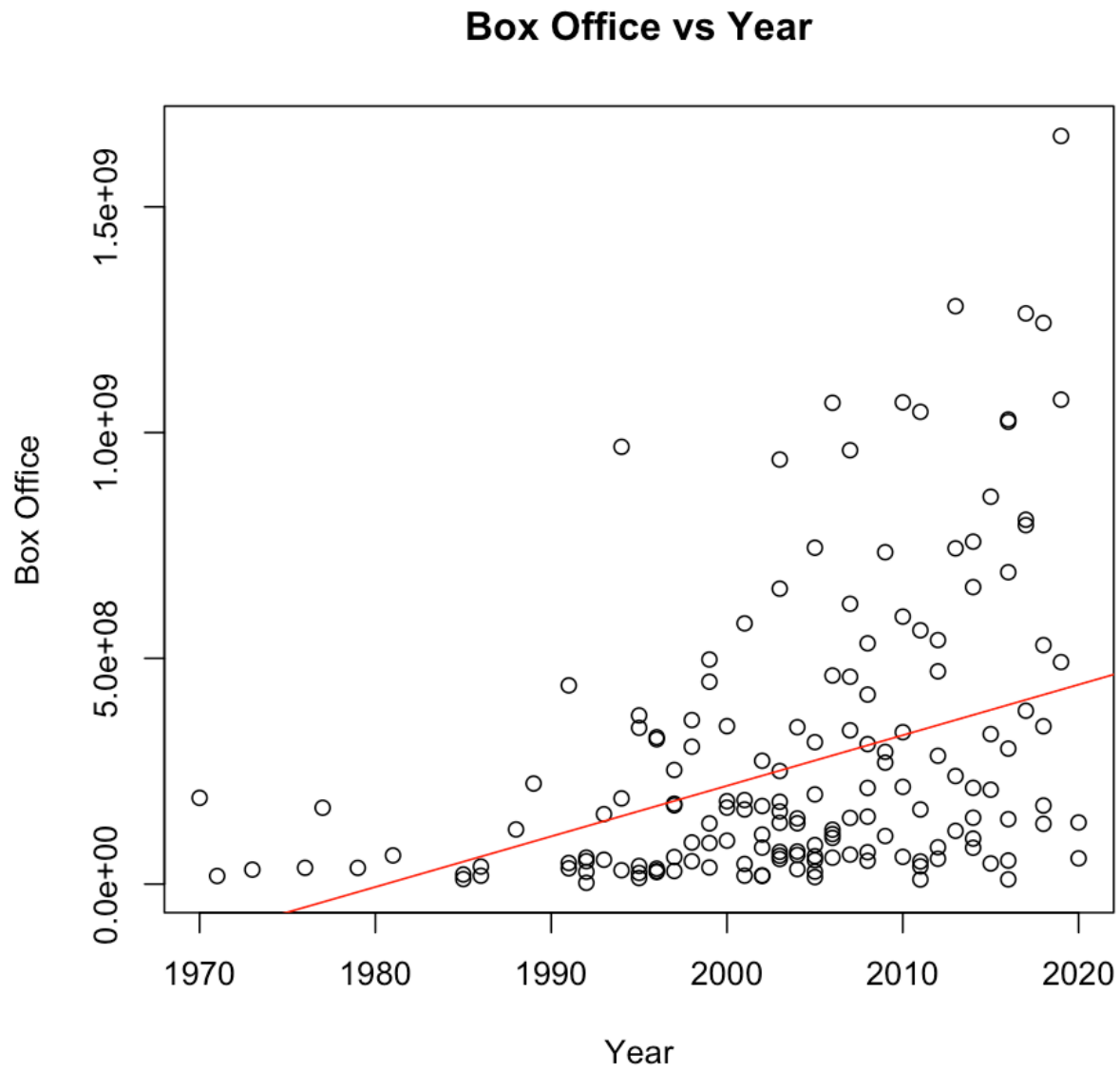


Figure 22 (Appending Inflation column):

```
#-----Appending inflation columns (Box Office and Budget)

conversion <- read.csv("2022_Inflation_Convertion.csv")#dataframe created using online resources
disney <- disney[order(disney$year),] #Order data by year

#Append Inflation
conv <- c()
* for (x in disney$year) {
  print(x)
  index <- which(conversion$Year == x)
  change <- conversion[index,"Conversion"]
  conv <- append(conv, change)
* }
disney$Inflation <- conv

#Budget Adjusted for Inflation
disney$Inflation_budg <- disney$Inflation * disney$budget
#Box Office Adjusted for Inflation
disney$Inflation_office <- disney$Inflation * disney$box_office
```

Figure 23 (Removing variables using Cook Distance variables):

```
#-----Influence-----
D <- cooks.distance(modsel_Inf2)

# Flag the high Cook's distance observations:
k=7
summary(disney)
n=161
high_cook <- which(D > 4/(n-k-1))
high_cook
disney[high_cook,]

now <- disney[-high_cook,]
hz <- as.numeric(now$music=="Hans Zimmer")
attach(disney)
attach(now)

modsel_Inf2 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore, 2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)
modsel_Inf3 <- lm(Inflation_office ~ Inflation_budg:rotten_tom + poly(metascore, 2, raw = TRUE) + poly(rotten_tom, 3, raw = TRUE) + hz)
summary(modsel_Inf3)

anova(modsel_Inf2,modsel_Inf3)
```

References:

<https://www.usinflationcalculator.com/>

<https://www.kaggle.com/datasets/unanimad/disney-plus-shows>

<https://www.kaggle.com/datasets/therealsampat/disney-movies-dataset>

<https://www.demandsage.com/disney-users/>