INTRODUCTION:

Movie streaming platform industry currently becomes more popular because the increasing number of customers who would like to spend time at home to enjoy a show instead of going to the cinema or spending a long time waiting for a show time on channel. It completely changes the hobby of watchers because now people can not just only save more money but being able to enjoy more shows at any time and anywhere. For the team project, we are fortunate to get to know the data source from Kaggle (https://www.kaggle.com/ruchi798/movies-on-netflix-prime-video-hulu-and-disney) with the data provided by four famous platforms in America such as Netflix, Amazon prime, Hulu and Disney. The data set helps my team to explore many interesting aspects of movie streaming platform. We believe these findings are very helpful, (1) it can help the platform companies find out the potential limitations (for example: age limitation or inequality in genres) to improve the platform better, (2) it provided statistic data about the movie industry throughout years, which people may have less knowledge about it before, (3) this will help company to have a good prediction model, which will detect the show efficiently then recommend it for customers as accurately as possible, (4) this will help the business realizes the quality of movies that they are providing for customers to distribute the number of movies on platform appropriately.

This dataset consists of several records of movies and each of the platform they belong to. The four platforms consist of Netflix, Hulu, Prime and Disney. Each of these streaming platforms have their own column respectively and contain 0s and 1s implying if a movie is available to be viewed on them. For example, if a movie record ‘Inception’ has a 1 1 under Netflix column, then it means that ‘Inception’ is available to be watched on Netflix. Likewise, if ‘Inception’ is 0 under Hulu, it implies that the movie is not available to be watched on Hulu.

Other columns in the dataset include Year, Age, IMDb ratings, Directors, Genre, Language and Runtime.

Here is a brief description of each column of the dataset:

1. ID: Unique ID of each movie record
2. Title: The name of the movie record

3. Year: Release year of the movie
4. Age: Target age range for each movie
5. IMDb: rating of a movie from a scale of 1.0 to 10.0
6. Rotten Tomatoes: rating of a movie from 1% to 100%
7. Netflix: states whether the movie is available on Netflix
8. Hulu: states whether the movie is available on Hulu
9. Prime: states whether the movie is available on Prime
10. Disney+: states whether the movie is available on Disney
11. Directors: Name of the director(s) of each of the movie
12. Genre: Type of movie
13. Country: origin of the movie
14. Language: Language of the origin
15. Runtime: duration of the movie

ANALYSIS:

The purpose to this analysis is because we want to answer the questions we have doubts at the beginning.

I. Explanatory Data Analysis:

1. *What are top rating movies?*
2. *Is there difference in the evaluation of movies from different genres on Netflix?*
3. *Ratio of international movies or (movies without English language) within each platform. This is done to understand which platform caters most to international audience.*
4. *Whether the platforms have consideration to teenager customers?*
5. *What are the most popular genres of American movies on different platforms? How the scores of genres are different?*

II. Method:

1. *Is there difference in the number of quality movies of 4 platforms?*
2. *The correlation between the runtime of a show and its independent variables.*

Top Rating Movies (1990-2020)

```
#load package
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(stringr)

#load data
MOV = read.csv("Movies.csv")

#take a look at data
str(MOV)

## 'data.frame':    16744 obs. of  17 variables:
##  $ X              : int  0 1 2 3 4 5 6 7 8 9 ...
##  $ ID             : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Title          : Factor w/ 16744 levels "\"22\" A film about Veterans,
Healthcare and Suicide.",..: 6734 14148 1492 1570 13561 11905 14433 4136 1047
3 6772 ...
##  $ Year           : int  2010 1999 2018 1985 1966 2018 2002 2012 1981 2009
...
##  $ Age            : Factor w/ 6 levels "","13+","16+",..: 2 4 2 5 4 5 4 4
5 4 ...
##  $ IMDb           : num  8.8 8.7 8.5 8.5 8.8 8.4 8.5 8.4 8.4 8.3 ...
##  $ Rotten.Tomatoes: Factor w/ 100 levels "","10%","100%",..: 87 87 84 97 9
8 98 96 87 96 89 ...
##  $ Netflix        : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Hulu           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Prime.Video    : int  0 0 0 0 1 0 1 0 0 0 ...
##  $ Disney.        : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Type           : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Directors      : Factor w/ 11339 levels "","A'Ali de Sousa",..: 1989 62
92 811 9129 9754 1284 9220 8551 10211 8551 ...
##  $ Genres         : Factor w/ 1910 levels "","Action","Action,Adventure",.
.: 178 407 176 509 1906 663 941 1678 3 572 ...
##  $ Country        : Factor w/ 1304 levels "","Afghanistan,France",..: 1255
1061 1061 1061 583 1061 952 1061 1061 442 ...
```

```
##  $ Language       : Factor w/ 1103 levels "","Aboriginal,English",..: 352
103 103 103 785 493 279 261 268 261 ...
##  $ Runtime        : int  148 136 149 116 161 117 150 165 115 153 ...
```

```r
head(MOV)
```

```
##   X ID                             Title Year Age IMDb Rotten.Tomatoes Net
flix
## 1 0  1                         Inception 2010 13+  8.8             87%
1
## 2 1  2                        The Matrix 1999 18+  8.7             87%
1
## 3 2  3            Avengers: Infinity War 2018 13+  8.5             84%
1
## 4 3  4                Back to the Future 1985  7+  8.5             96%
1
## 5 4  5    The Good, the Bad and the Ugly 1966 18+  8.8             97%
1
## 6 5  6 Spider-Man: Into the Spider-Verse 2018  7+  8.4             97%
1
##   Hulu Prime.Video Disney. Type                                   Director
s
## 1    0           0       0    0                           Christopher Nola
n
## 2    0           0       0    0                  Lana Wachowski,Lilly Wachowsk
i
## 3    0           0       0    0                        Anthony Russo,Joe Russ
o
## 4    0           0       0    0                            Robert Zemecki
s
## 5    0           1       0    0                               Sergio Leon
e
## 6    0           0       0    0 Bob Persichetti,Peter Ramsey,Rodney Rothma
n
##                                    Genres                      Country
## 1         Action,Adventure,Sci-Fi,Thriller United States,United Kingdom
## 2                           Action,Sci-Fi                 United States
## 3                 Action,Adventure,Sci-Fi                 United States
## 4                 Adventure,Comedy,Sci-Fi                 United States
## 5                                 Western     Italy,Spain,West Germany
## 6 Animation,Action,Adventure,Family,Sci-Fi                 United States
##                  Language Runtime
## 1 English,Japanese,French     148
## 2                 English     136
## 3                 English     149
## 4                 English     116
## 5                 Italian     161
## 6         English,Spanish     117
```

```r
MOV_1990 = MOV %>% filter(Year > 1990)

Top10_action <- MOV_1990 %>%
  filter(str_detect(Genres, "Action")) %>% group_by(Genres) %>% arrange(desc(
IMDb)) %>% select(Title,Year,IMDb,Rotten.Tomatoes, Type,Directors,Country,Net
flix,Hulu,Language) %>% head(10)

## Adding missing grouping variables: `Genres`

ggplot(Top10_action ,aes(reorder(Title,-IMDb),IMDb)) + geom_bar(stat = "ident
ity",fill="red") + theme(axis.text.x = element_text(angle = 90, vjust = 0.5,
hjust=1)) + labs(x="Genres",y="IMDb",title="Action")
```

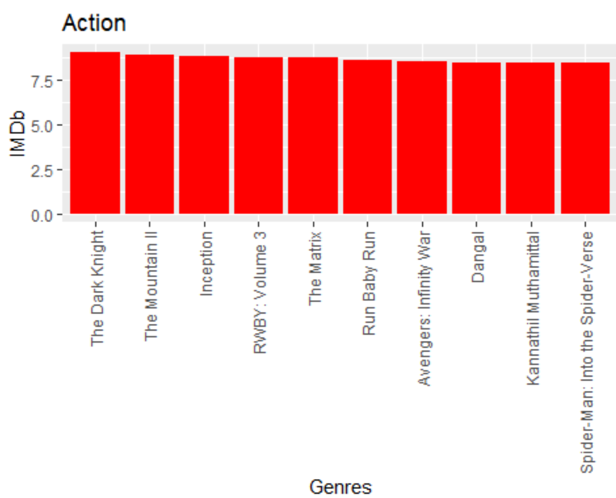
```r
Top10_thriller <- MOV_1990 %>%
  filter(str_detect(Genres, "Thriller")) %>% group_by(Genres) %>% arrange(des
c(IMDb)) %>% select(Title,Year,IMDb,Rotten.Tomatoes, Type,Directors,Country,N
etflix,Hulu,Language) %>% head(10)

## Adding missing grouping variables: `Genres`

ggplot(Top10_thriller ,aes(reorder(Title,-IMDb),IMDb)) + geom_bar(stat = "ide
ntity",fill="cadetblue") + theme(axis.text.x = element_text(angle = 90, vjust
= 0.5, hjust=1)) + labs(x="Genres",y="IMDb",title="Thriller")
```

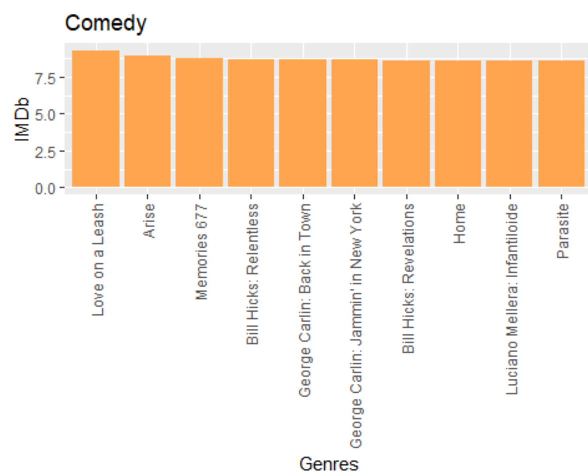
```r
Top10_comedy <- MOV_1990 %>%
  filter(str_detect(Genres, "Comedy")) %>% group_by(Genres) %>% arrange(desc(
IMDb)) %>% select(Title,Year,IMDb,Rotten.Tomatoes, Type,Directors,Country,Net
flix,Hulu,Language) %>% head(10)

## Adding missing grouping variables: `Genres`

ggplot(Top10_comedy,aes(reorder(Title,-IMDb),IMDb)) + geom_bar(stat = "identi
ty",fill="tan1") + theme(axis.text.x = element_text(angle = 90, vjust = 0.5,
hjust=1)) + labs(x="Genres",y="IMDb",title="Comedy")
```

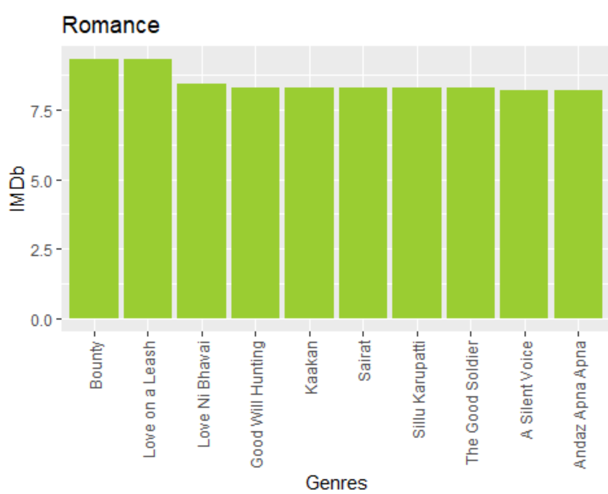
```r
Top10_horror <- MOV_1990 %>%
  filter(str_detect(Genres, "Horror")) %>% group_by(Genres) %>% arrange(desc(
IMDb)) %>% select(Title,Year,IMDb,Rotten.Tomatoes, Type,Directors,Country,Net
flix,Hulu,Language) %>% head(10)

## Adding missing grouping variables: `Genres`

ggplot(Top10_horror,aes(reorder(Title,-IMDb),IMDb)) + geom_bar(stat = "identi
ty") + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1))+ l
abs(x="Genres",y="IMDb",title="Horror")
```

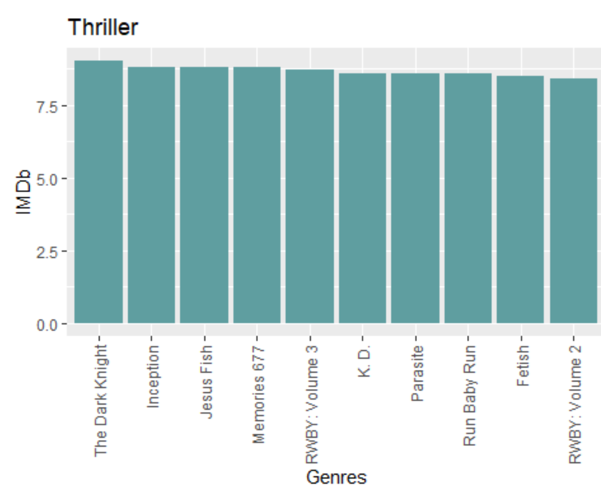
```
Top10_romance <- MOV_1990 %>%
filter(str_detect(Genres, "Romance")) %>% group_by(Genres) %>% arrange(desc(IMDb)) %>% select(Title,Year,IMDb,Rotten.Tomatoes, Type,Directors,Country,Netflix,Hulu,Language) %>% head(10)

## Adding missing grouping variables: `Genres`

ggplot(Top10_romance,aes(reorder(Title,-IMDb),IMDb)) + geom_bar(stat = "identity",fill= "yellowgreen") + theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) + labs(x="Genres",y="IMDb",title="Romance")
```

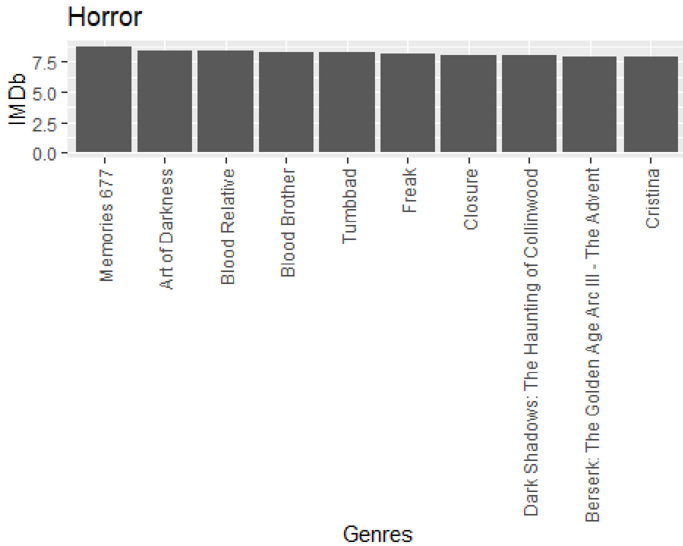
***









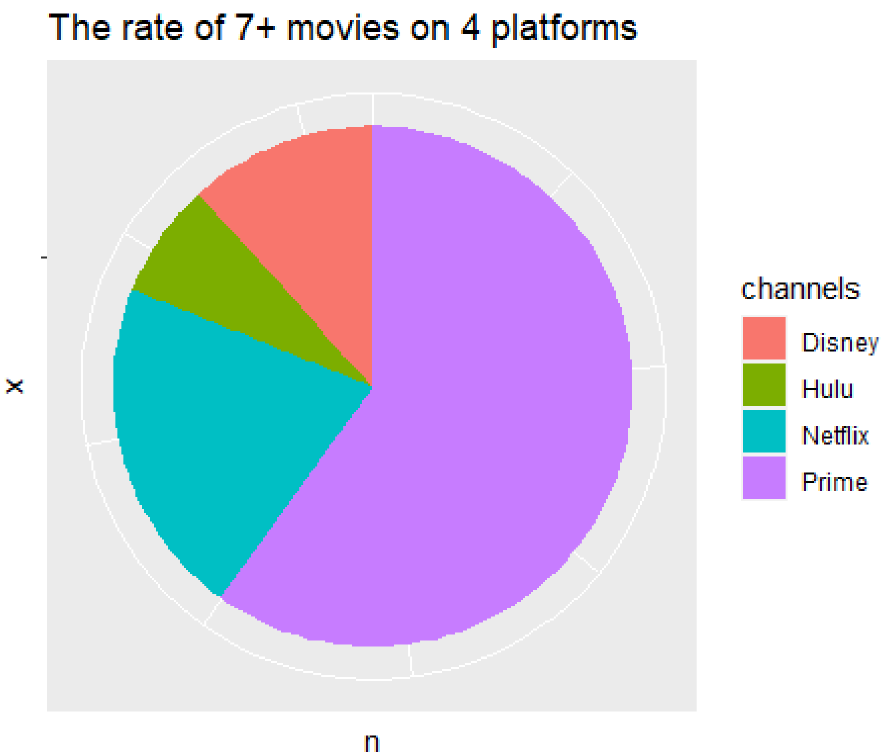


Firstly, we want to start the project by the statistics about the top rating movies of each genres (Horror, Comedy, Action, Thriller, Romance), this result is based on the evaluation of IMDb (an online database of information related to films, television programs, home videos, video games, and streaming content online). It is amazing that there are mostly old movies in the top ratings, this shows that although the movie industry is increasing quickly with more modern facilities and higher budget, the content quality is the most important factor to get the good reviews from watchers.

The Consideration For Age-Limited Movies

```
#Whether the different platforms consider the kid viewers
summary(MOV$Age)

##       13+  16+  18+   7+  all
## 9390 1255  320 3474 1462  843

netflix_age = MOV %>% filter(Age == "7+") %>% select(ID,Title,Netflix,Hulu,Pr
ime.Video,Disney.,Genres) %>% group_by(Netflix) %>% count

hulu_age = MOV %>% filter(Age == "7+") %>% select(ID,Title,Netflix,Hulu,Prime
.Video,Disney.,Genres) %>% group_by(Hulu) %>% count

prime.video_age = MOV %>% filter(Age == "7+") %>% select(ID,Title,Netflix,Hul
u,Prime.Video,Disney.,Genres) %>% group_by(Prime.Video) %>% count

disney_age = MOV %>% filter(Age == "7+") %>% select(ID,Title,Netflix,Hulu,Pri
```

```
me.Video,Disney.,Genres) %>% group_by(Disney.) %>% count

channels = c("Netflix","Hulu","Prime","Disney")
n = rbind(netflix_age[2,2] / 1462,hulu_age[2,2] / 1462,prime.video_age[2,2]/
1462,disney_age[2,2]/ 1462)
n_channels = cbind(channels,n)
ggplot(n_channels,aes(x="",y=n,fill=channels)) + geom_bar(width = 1, stat = "
identity") + coord_polar("y", start=0) + labs(title = "The rate of 7+ movies
on 4 platforms")  + theme(axis.text.x=element_blank())
```



***

We assume that if a platform has less movies which are suitable with the kids or teenagers, this will be disadvantage. Because most of parents manage the contents that their children watch, they do not want their children get access with platform with less suitable products. Additionally, the movies being suitable for children will always be preferred when a family decide to host the family cinema at home. From the summary, we have the comparison of the number of movies for each age in all platforms as below:

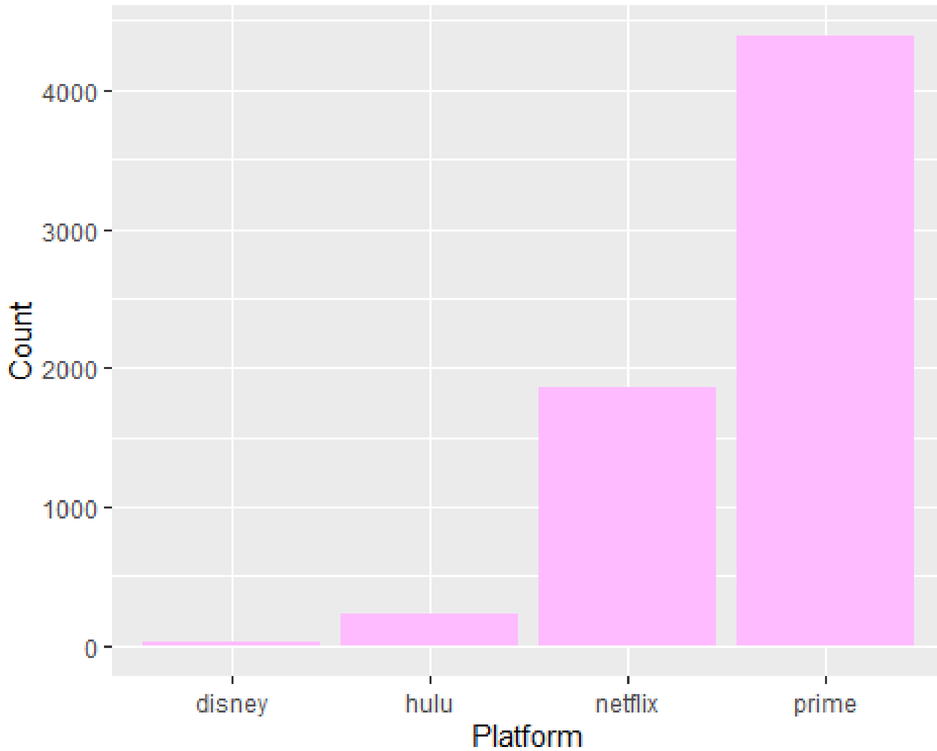
```
##       13+  16+  18+   7+  all
## 9390 1255  320 3474 1462  843
```

The summary indicates there are mostly the movies for people who are over 13. From the pie chart, the Prime Video have the largest number of these shows, while Disney even has fewer number of these movies than Netflix. This result may be not accurate because of the inequality of the number of movies on each platform when Disney just started to jump into streaming industry. However, this subject is still a necessary consideration that we would like to explore more in the future.

Top Platforms For Non-Speaking English Customers

```
#Top platforms contains foreign language movies
not_USA_netflix = filter(MOV, !grepl("United States",Country))
not_USA_netflix = not_USA_netflix %>% subset(Netflix == "1") %>% select(ID,Ti
tle,Genres,Country) %>% count()

not_USA_hulu = filter(MOV, !grepl("United States",Country))
not_USA_hulu = not_USA_hulu %>% subset(Hulu == "1") %>% select(ID,Title,Genre
s,Country) %>% count()

not_USA_prime = filter(MOV, !grepl("United States",Country))
not_USA_prime = not_USA_prime %>% subset(Prime.Video == "1") %>% select(ID,Ti
tle,Genres,Country) %>% count()

not_USA_disney = filter(MOV, !grepl("United States",Country))
not_USA_disney = not_USA_disney %>% subset(Disney. == "1") %>% select(ID,Titl
e,Genres,Country) %>% count()

not_USA = cbind(not_USA_netflix, not_USA_hulu, not_USA_prime, not_USA_disney)
colnames(not_USA) = c("netflix","hulu","prime","disney")

Count = t(not_USA)
not_USA = data.frame(Platform= row.names(Count), Count, row.names=NULL)
ggplot(not_USA,aes(Platform,Count)) + geom_bar(stat="identity",fill="plum1")
```

***



The reason why we want to explore this is because we want to know which platforms support better for people who cannot understand English. As the plot show, Prime and Netflix are the most popular movies having different languages for subtitle. Therefore, they will even have more customers from out of America. This is really important if the companies want to investigate into the global market.

Top Movies From Different Genres On Platforms

```
#4.What is the most popular genres of American movies in three platforms ? Wh
ether these movies from that each genres get different scores ?

#Netflix
Action = MOV %>% filter(Netflix == 1,str_detect(Genres, "Action"),Country ==
"United States")  %>% count()
Romance = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Romance"),Country
== "United States")  %>% count()
Horror = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Horror"),Country ==
"United States")  %>% count()
Thriller = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Thriller"),Countr
y == "United States")  %>% count()
Comedy = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Comedy"),Country ==
"United States")  %>% count()

Netflix = data.frame(Action = Action,Romance = Romance,Horror = Horror, Thril
```

```
ler = Thriller, Comedy = Comedy)

colnames(Netflix) = c("Action","Romance","Horror","Thriller","Comedy")
Count = t(Netflix)
Netflix = data.frame(Genres= row.names(Count), Count, row.names=NULL)

#Hulu
Action = MOV %>% filter(Hulu == 1,str_detect(Genres, "Action"),Country == "Un
ited States")  %>% count()
Romance = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Romance"),Country ==
"United States")  %>% count()
Horror = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Horror"),Country == "U
nited States")  %>% count()
Thriller = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Thriller"),Country =
= "United States")  %>% count()
Comedy = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Comedy"),Country == "U
nited States")  %>% count()

Hulu = data.frame(Action = Action,Romance = Romance,Horror = Horror, Thriller
= Thriller, Comedy = Comedy)

colnames(Hulu) = c("Action","Romance","Horror","Thriller","Comedy")
Count = t(Hulu)
Hulu = data.frame(Genres= row.names(Count), Count, row.names=NULL)


#Prime
Action = MOV %>% filter(Prime.Video == 1,str_detect(Genres, "Action"),Country
== "United States")  %>% count()
Romance = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Romance"),Coun
try == "United States")  %>% count()
Horror = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Horror"),Countr
y == "United States")  %>% count()
Thriller = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Thriller"),Co
untry == "United States")  %>% count()
Comedy = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Comedy"),Countr
y == "United States")  %>% count()

Prime.Video = data.frame(Action = Action,Romance = Romance,Horror = Horror, T
hriller = Thriller, Comedy = Comedy)

colnames(Prime.Video) = c("Action","Romance","Horror","Thriller","Comedy")

Count = t(Prime.Video)
Prime.Video = data.frame(Genres= row.names(Count), Count, row.names=NULL)


#Disney
Action = MOV %>% filter(Disney. == 1,str_detect(Genres, "Action"),Country ==
"United States")  %>% count()
```

```r
Romance = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Romance"),Country
== "United States")  %>% count()
Horror = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Horror"),Country ==
"United States")  %>% count()
Thriller = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Thriller"),Countr
y == "United States")  %>% count()
Comedy = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Comedy"),Country ==
"United States")  %>% count()

Disney = data.frame(Action = Action,Romance = Romance,Horror = Horror, Thrill
er = Thriller, Comedy = Comedy)

colnames(Disney) = c("Action","Romance","Horror","Thriller","Comedy")
Count = t(Disney)
Disney = data.frame(Genres= row.names(Count), Count, row.names=NULL)

#Netflix
Action = MOV %>% filter(Netflix == 1,str_detect(Genres, "Action"),Country ==
"United States")  %>% count()
Romance = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Romance"),Country
== "United States")  %>% count()
Horror = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Horror"),Country ==
"United States")  %>% count()
Thriller = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Thriller"),Countr
y == "United States")  %>% count()
Comedy = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Comedy"),Country ==
"United States")  %>% count()

Netflix = data.frame(Action = Action,Romance = Romance,Horror = Horror, Thril
ler = Thriller, Comedy = Comedy)

colnames(Netflix) = c("Action","Romance","Horror","Thriller","Comedy")
Count = t(Netflix)
Netflix = data.frame(Genres= row.names(Count), Count, row.names=NULL)

#Netflix
Action = MOV %>% filter(Netflix == 1,str_detect(Genres, "Action"),Country ==
"United States")
avg_Ac = mean(Action[,'IMDb'])

Romance = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Romance"),Country
== "United States")
avg_Ro = mean(Romance[,'IMDb'])

Horror = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Horror"),Country ==
"United States")
avg_Ho = mean(Horror[,'IMDb'])
Thriller = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Thriller"),Countr
y == "United States")
avg_Th = mean(Thriller[,'IMDb'],na.rm =TRUE)
```

```
Comedy = MOV %>%  filter(Netflix == 1,str_detect(Genres, "Comedy"),Country ==
"United States")
avg_Co = mean(Comedy[,'IMDb'],na.rm =TRUE)

Netflix['IMDb'] =  c(avg_Ac,avg_Ro,avg_Ho,avg_Th,avg_Co)

#Plot
ggplot(Netflix) +
  geom_col(aes(x = Genres, y = Count), size = 1, colour="sienna3", fill = "ta
n1") +
  geom_line(aes(x = Genres, y = IMDb*50), size = 1.5, color="black", group =
1,stat="identity") + scale_y_continuous(sec.axis = sec_axis(~./50,name = "IMD
b Mean Rate"),name="Number of Movies") + labs(title="Netflix")
```

```
#Hulu
Action = MOV %>% filter(Hulu == 1,str_detect(Genres, "Action"),Country == "Un
ited States")
avg_Ac = mean(Action[,'IMDb'])

Romance = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Romance"),Country ==
"United States")
avg_Ro = mean(Romance[,'IMDb'])

Horror = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Horror"),Country == "U
nited States")
avg_Ho = mean(Horror[,'IMDb'],na.rm =TRUE)
Thriller = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Thriller"),Country =
= "United States")
avg_Th = mean(Thriller[,'IMDb'],na.rm =TRUE)
Comedy = MOV %>%  filter(Hulu == 1,str_detect(Genres, "Comedy"),Country == "U
nited States")
avg_Co = mean(Comedy[,'IMDb'],na.rm =TRUE)

Hulu['IMDb'] =  c(avg_Ac,avg_Ro,avg_Ho,avg_Th,avg_Co)

#Plot
ggplot(Hulu) +
  geom_col(aes(x = Genres, y = Count), size = 1, colour="coral2", fill = "cor
al2") +
  geom_line(aes(x = Genres, y = IMDb*15), size = 1.5, color="black", group =
1,stat="identity") + scale_y_continuous(sec.axis = sec_axis(~./15,name = "IMD
b Mean Rate"),name="Number of Movies") + labs(title="Hulu")
```

```
#Prime.Video
Action = MOV %>% filter(Prime.Video == 1,str_detect(Genres, "Action"),Country
== "United States")
avg_Ac = mean(Action[,'IMDb'],na.rm =TRUE)
```

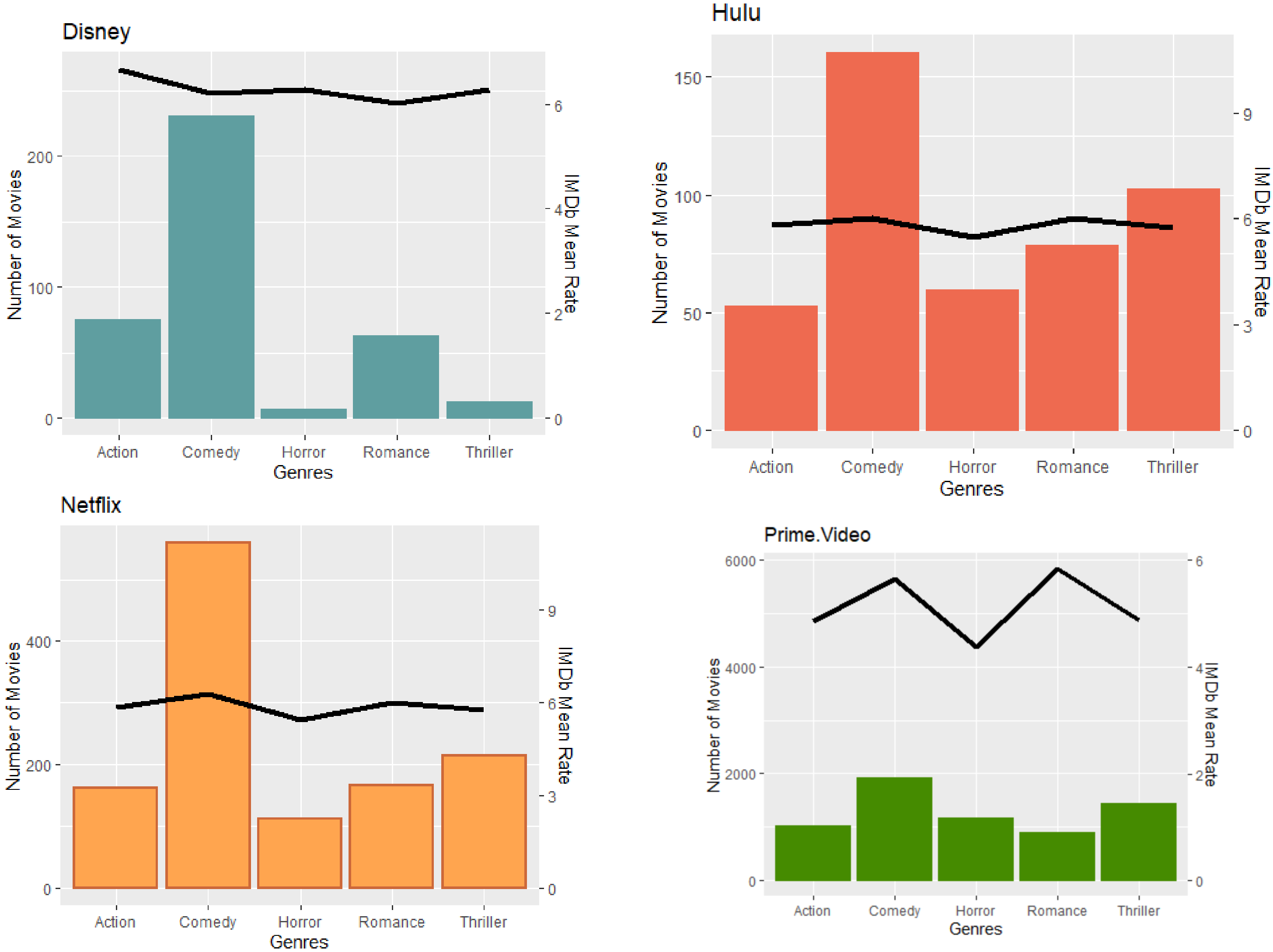
```
Romance = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Romance"),Coun
try == "United States")
avg_Ro = mean(Romance[,'IMDb'],na.rm =TRUE)

Horror = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Horror"),Countr
y == "United States")
avg_Ho = mean(Horror[,'IMDb'],na.rm =TRUE)
Thriller = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Thriller"),Co
untry == "United States")
avg_Th = mean(Thriller[,'IMDb'],na.rm =TRUE)
Comedy = MOV %>%  filter(Prime.Video == 1,str_detect(Genres, "Comedy"),Countr
y == "United States")
avg_Co = mean(Comedy[,'IMDb'],na.rm =TRUE)

Prime.Video['IMDb'] =  c(avg_Ac,avg_Ro,avg_Ho,avg_Th,avg_Co)

#Plot
ggplot(Prime.Video) +
  geom_col(aes(x = Genres, y = Count), size = 1, colour="chartreuse4", fill =
"chartreuse4") +
  geom_line(aes(x = Genres, y = IMDb*1000), size = 1.5, color="black", group
= 1,stat="identity") + scale_y_continuous(sec.axis = sec_axis(~./1000,name =
"IMDb Mean Rate"),name="Number of Movies") + labs(title="Prime.Video")
```

```
#Disney
Action = MOV %>% filter(Disney. == 1,str_detect(Genres, "Action"),Country ==
"United States")
avg_Ac = mean(Action[,'IMDb'],na.rm =TRUE)

Romance = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Romance"),Country
== "United States")
avg_Ro = mean(Romance[,'IMDb'],na.rm =TRUE)

Horror = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Horror"),Country ==
"United States")
avg_Ho = mean(Horror[,'IMDb'],na.rm =TRUE)
Thriller = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Thriller"),Countr
y == "United States")
avg_Th = mean(Thriller[,'IMDb'],na.rm =TRUE)
Comedy = MOV %>%  filter(Disney. == 1,str_detect(Genres, "Comedy"),Country ==
"United States")
avg_Co = mean(Comedy[,'IMDb'],na.rm =TRUE)

Disney['IMDb'] =  c(avg_Ac,avg_Ro,avg_Ho,avg_Th,avg_Co)


#Plot
```

```
ggplot(Disney) +
  geom_col(aes(x = Genres, y = Count), size = 1, colour="cadetblue", fill = "
cadetblue") +
  geom_line(aes(x = Genres, y = IMDb*40), size = 1.5, color="black", group =
1,stat="identity") + scale_y_continuous(sec.axis = sec_axis(~./40,name = "IMD
b Mean Rate"),name="Number of Movies") + labs(title="Disney")
```

```
Netflix

##      Genres Count     IMDb
## 1    Action   163 5.846012
## 2   Romance   167 6.001198
## 3    Horror   112 5.450893
## 4  Thriller   214 5.759434
## 5    Comedy   559 6.277738

Hulu

##      Genres Count     IMDb
## 1    Action    52 5.803846
## 2   Romance    78 6.014103
## 3    Horror    59 5.468966
## 4  Thriller   102 5.732673
## 5    Comedy   160 5.991824

Prime.Video

##      Genres Count     IMDb
## 1    Action  1007 4.848085
## 2   Romance   871 5.838979
## 3    Horror  1153 4.377170
## 4  Thriller  1416 4.882988
## 5    Comedy  1902 5.653478

Disney

##      Genres Count     IMDb
## 1    Action    74 6.655405
## 2   Romance    62 6.016129
## 3    Horror     6 6.266667
## 4  Thriller    11 6.272727
## 5    Comedy   230 6.196087
```

***



In general, comedy is the most popular show on the platforms and its mean rating is very high, this indicates that there are many good comedy movies on platforms. Horror movies are so low quality, thus it is produced or provided less compared to other genres.

For Netflix, Hulu and Prime, the second popular one is thriller movie, however, its mean rating is not high (mostly under 6). Meanwhile, this is opposite to Disney platform, the thriller movies are not popular on this platform, instead action and romance movies are the second popular in Disney, and action movies get very high mean rating.

Methods:

1. Is there difference in the number of quality movies of 4 platforms?

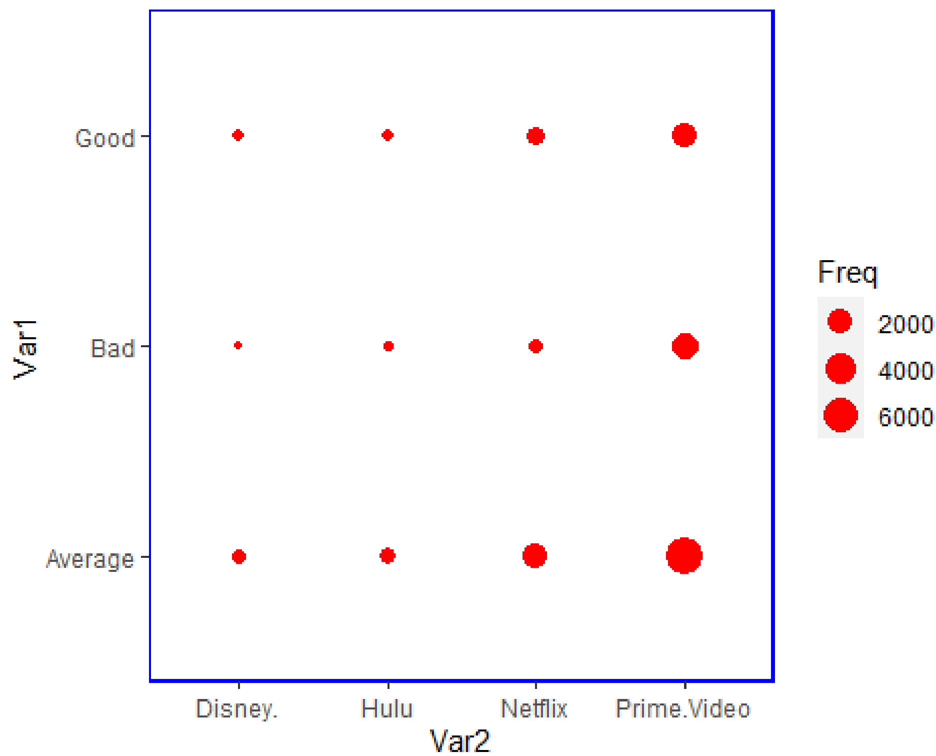
The reason: Since there are the competition between famous platforms, it is helpful to investigate if there is the inequality in the number of quality movies in 4 platforms. If a platform produces so much bad-quality movies, this will cost them a lot and even not bring back the profit, leading to the decrease of customers when they have the bad impression after using that platform for the first time. Therefore, we decide to use chi-square to know the difference and explore the association between number of bad or good movies and its platforms.

```
#load package
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union

library(ggplot2)
library(stringr)
library("gplots")

##
## Attaching package: 'gplots'

## The following object is masked from 'package:stats':
##
##     lowess

library(tidyr)
library(corrplot)

## corrplot 0.84 loaded

#load data
MOV = read.csv("Movies.csv")
```

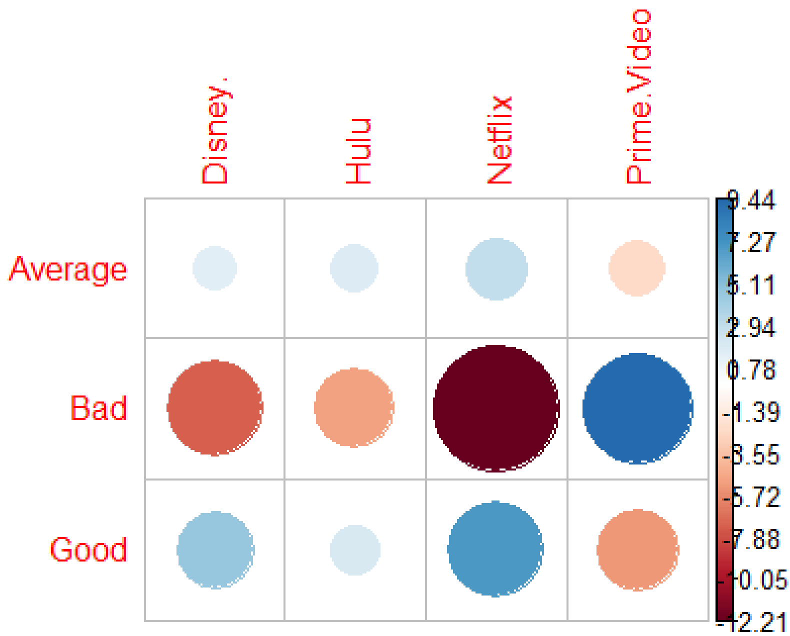
```
#Assign the string value into scores
MOV$IMDb = MOV$IMDb %>% replace_na(mean(MOV$IMDb,na.rm=TRUE))
MOV$IMDb = with(MOV, ifelse(IMDb >=7 , "Good",
                  ifelse(IMDb >=5, "Average", "Bad")))

#Total movies
MOV %>% group_by(IMDb) %>% count()

## # A tibble: 3 x 2
## # Groups:   IMDb [3]
##   IMDb        n
##   <chr>   <int>
## 1 Average  9329
## 2 Bad      3620
## 3 Good     3795

#shape the data.frame
x = MOV %>% gather(Platform,Yes,Netflix:Disney.)
x = x[!(x$Yes %in% c(0)), ]
head(x,10)

##    X ID                             Title Year Age IMDb Rotten.Tomatoes Ty
pe
## 1  0  1                         Inception 2010 13+ Good             87%
0
## 2  1  2                        The Matrix 1999 18+ Good             87%
0
## 3  2  3            Avengers: Infinity War 2018 13+ Good             84%
0
## 4  3  4                Back to the Future 1985  7+ Good             96%
0
## 5  4  5    The Good, the Bad and the Ugly 1966 18+ Good             97%
0
## 6  5  6 Spider-Man: Into the Spider-Verse 2018  7+ Good             97%
0
## 7  6  7                       The Pianist 2002 18+ Good             95%
0
## 8  7  8                  Django Unchained 2012 18+ Good             87%
0
## 9  8  9           Raiders of the Lost Ark 1981  7+ Good             95%
0
## 10 9 10              Inglourious Basterds 2009 18+ Good             89%
0
##                                    Directors
## 1                          Christopher Nolan
## 2             Lana Wachowski,Lilly Wachowski
## 3                    Anthony Russo,Joe Russo
## 4                           Robert Zemeckis
## 5                              Sergio Leone
## 6  Bob Persichetti,Peter Ramsey,Rodney Rothman
## 7                            Roman Polanski
```

```
## 8                                Quentin Tarantino
## 9                                 Steven Spielberg
## 10                               Quentin Tarantino
##                                             Genres
## 1             Action,Adventure,Sci-Fi,Thriller
## 2                                Action,Sci-Fi
## 3                      Action,Adventure,Sci-Fi
## 4                      Adventure,Comedy,Sci-Fi
## 5                                      Western
## 6   Animation,Action,Adventure,Family,Sci-Fi
## 7                    Biography,Drama,Music,War
## 8                                Drama,Western
## 9                             Action,Adventure
## 10                         Adventure,Drama,War
##                                        Country
## 1             United States,United Kingdom
## 2                            United States
## 3                            United States
## 4                            United States
## 5                 Italy,Spain,West Germany
## 6                            United States
## 7   United Kingdom,France,Poland,Germany
## 8                            United States
## 9                            United States
## 10                   Germany,United States
##                                        Language Runtime Platform Yes
## 1                      English,Japanese,French     148  Netflix   1
## 2                                      English     136  Netflix   1
## 3                                      English     149  Netflix   1
## 4                                      English     116  Netflix   1
## 5                                      Italian     161  Netflix   1
## 6                              English,Spanish     117  Netflix   1
## 7                       English,German,Russian     150  Netflix   1
## 8                English,German,French,Italian     165  Netflix   1
## 9   English,German,Hebrew,Spanish,Arabic,Nepali     115  Netflix   1
## 10               English,German,French,Italian     153  Netflix   1
```

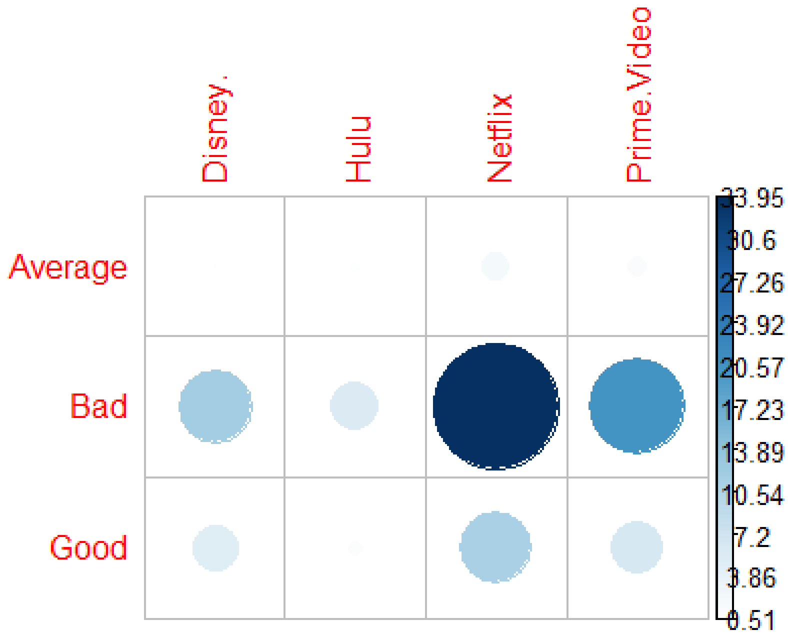
```
#count & apply chisquared
table = table(x$IMDb,x$Platform)
chi = chisq.test(table)
table = data.frame(table)

#plot the table
p <- ggplot(table, aes(x =Var2, y = Var1))
p+geom_point( aes(size=Freq),colour="red")+theme(panel.background=element_bla
nk(), panel.border = element_rect(colour = "blue", fill=NA, size=1))
```



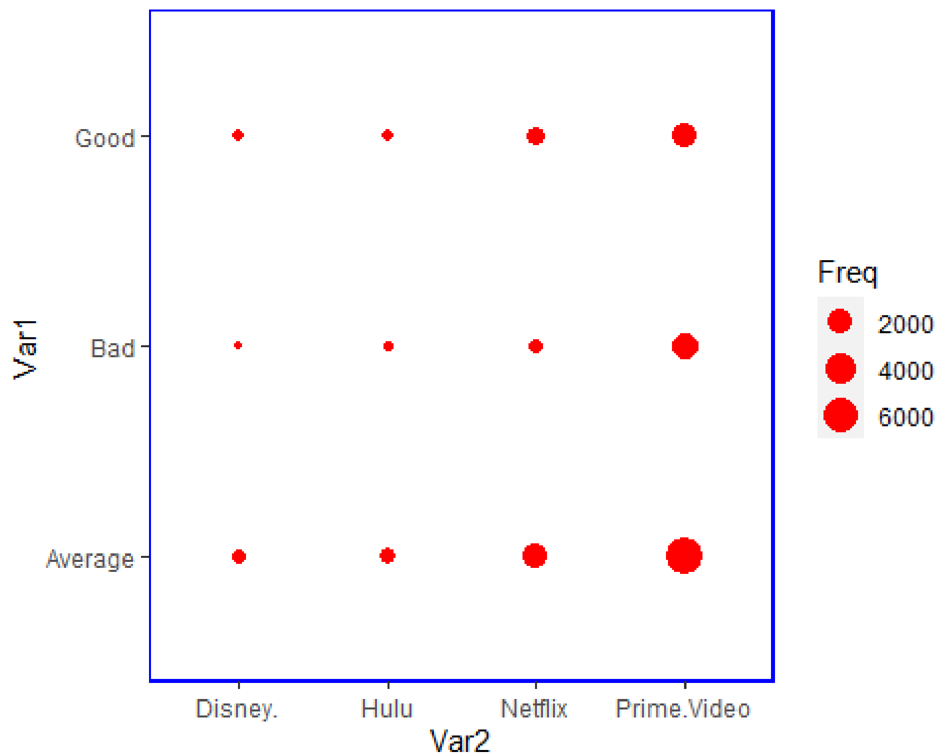

```
#summary
chi

##
##  Pearson's Chi-squared test
##
## data:  table
## X-squared = 439.19, df = 6, p-value < 2.2e-16

#plot the correlation between the residuals
corrplot(round(chi$residuals,3), is.cor = FALSE)
```




```
# Contibution in percentage (%)
contrib <- 100*round(chi$residuals,3)^2/chi$statistic
round(contrib, 3)

##
##           Disney.   Hulu  Netflix Prime.Video
##   Average   0.513  0.725   2.101       1.349
##   Bad      11.905  5.570  33.945      20.273
##   Good      4.921  0.902  11.349       6.447

# Visualize the contribution
corrplot(contrib, is.cor = FALSE)
```




```
# printing the p-value
chi$p.value

## [1] 1.040328e-91

# printing the mean
chi$statistic

## X-squared
##  439.1905
```



***

The plot indicates that most average movies come from Netflix and Prime Video. Meanwhile, the number of good movies in Prime Video are even lower than bad movies. Before we take a look at chi-squared result, it is necessary to provide the hypothesis for problem.

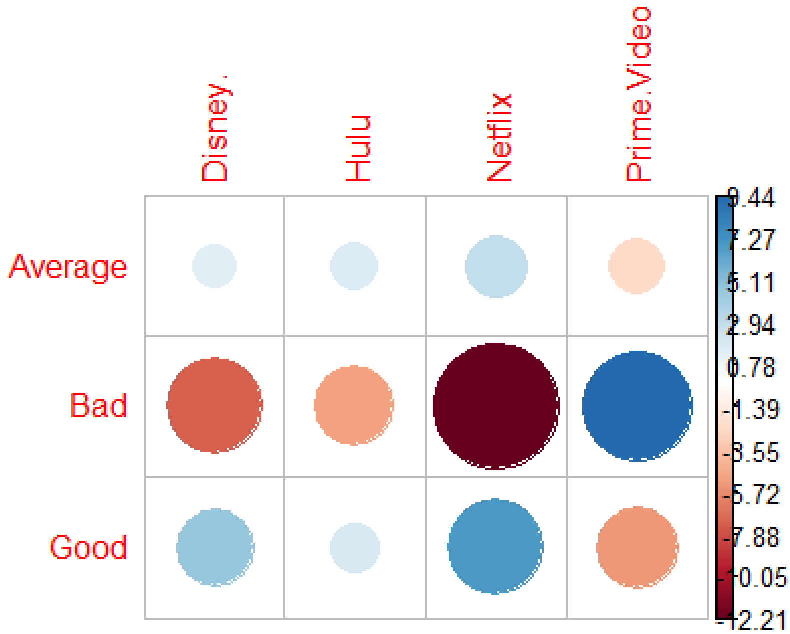
__Hypothesis:__

H0: there is no difference in the number of different quality movies in 4 platforms (claim)

H1: there is difference in the number of different quality movies in 4 platforms

```
chi

##
##  Pearson's Chi-squared test
##
## data:  table
## X-squared = 439.19, df = 6, p-value < 2.2e-16
```

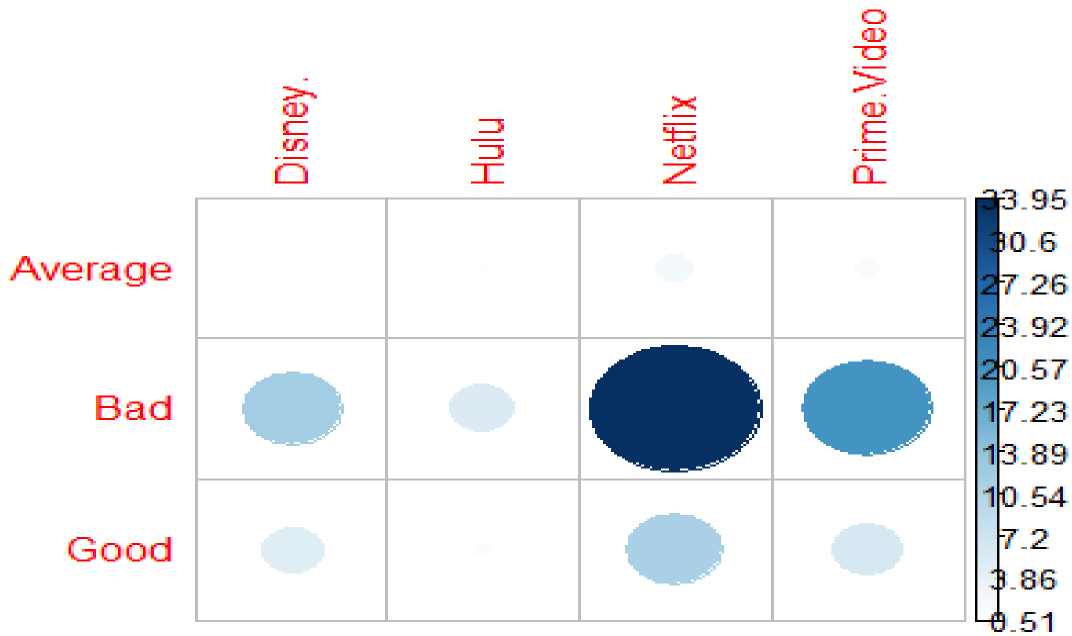
With the X-squared greater than critical value (12.592), the p-value less than 0.05 means this evidence is creditable. Therefore, we conclude that there is enough evidence to reject the null hypothesis. In other words, it can be concluded that there is the difference in the number of different quality movies in 4 platforms. Next, because the sign of standardized residuals is very important to understand the relation between the variables, we want to visualize the association between the variables by the plot based on the residuals from chi-squared:



As the blue color indicates the positive residuals, there is a positive relationship between the column Prime Video and the number of bad movies. Meanwhile, there is the positive relationship between the number of good movies and Netflix. For negative residuals, we can see that Disney and Netflix are highly negative related with the number of bad movies.

```
          Disney.    Hulu Netflix Prime.Video
Average     1.501   1.785   3.038      -2.434
Bad        -7.231  -4.946 -12.210       9.436
Good        4.649   1.990   7.060      -5.321
```

The relative contribution of each cell to the total Chi-square score give some indication of the nature of the dependency between rows and columns of the contingency table.



From the plot, the most contributing cells to the Chi-square are Disney/Bad movies (~12%), Netflix/Bad movies (~34%), Prime Video/Bad movies (~20%). The contribution is equal to over 60% for the total Chi-square score; thus, it accounts for most of the difference between expected and observed values.

Whether platforms should provide more past horror movies?

The reason: When customers use platform, they can watch any movies despite its release date. This benefit enables them to get access to horror movies from the past, which they have not experienced in the cinema. However, the challenge is that if these old movies do not adapt the demand of customers, the companies will cost a lot of money, causing bad situation for businesses. Therefore, it should be a try to determine the linear model between Year and IMDb by lm function in R.

```
#Whether the current movies is better than old movies ?
#Whether the current movies is better than old movies ?
```

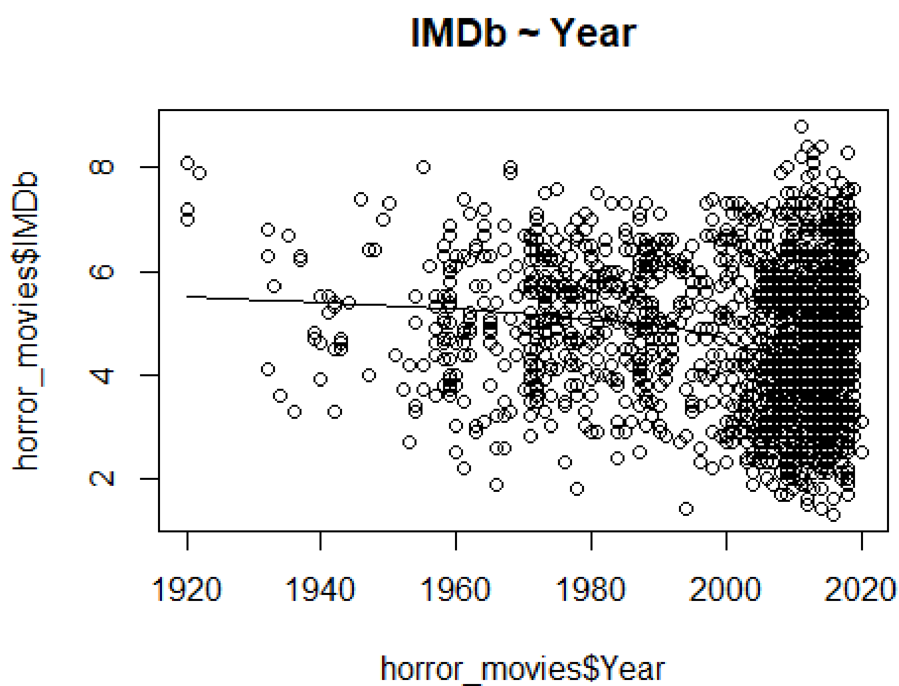
```r
horror_movies = MOV %>% filter(str_detect(Genres, "Horror")) %>% select(Year,
IMDb,Age,Runtime)
action_movies = MOV %>% filter(str_detect(Genres, "Action")) %>% select(Year,
IMDb,Age,Runtime)
action_movies = MOV %>% filter(str_detect(Genres, "Drama")) %>% select(Year,I
MDb,Age,Runtime)

#linear relationship
scatter.smooth(x=horror_movies$Year, y=horror_movies$IMDb, main="IMDb ~ Year"
)

#check outlier
par(mfrow=c(1, 2))  # divide graph area in 2 columns
boxplot(horror_movies$Year, main="Year", sub=paste("Outlier rows: ", boxplot.
stats(horror_movies$Year)$out))
boxplot(horror_movies$IMDb, main="IMDb", sub=paste("Outlier rows: ", boxplot.
stats(horror_movies$IMDb)$out))
```

```r
#check correlation
cor(horror_movies$Year,horror_movies$IMDb)

## [1] -0.1481772

model = lm(IMDb~Year,horror_movies)
sum = summary(model)
sum

##
## Call:
## lm(formula = IMDb ~ Year, data = horror_movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4320 -1.0454  0.0170  0.9871  4.1520
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 26.417698   3.086755   8.558  < 2e-16 ***
## Year        -0.010825   0.001541  -7.026 2.82e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.324 on 2199 degrees of freedom
## Multiple R-squared:  0.02196,    Adjusted R-squared:  0.02151
## F-statistic: 49.37 on 1 and 2199 DF,  p-value: 2.821e-12

AIC(model)

## [1] 7486.876
```

```r
par(mfrow=c(2,2))
plot(model)
```

***

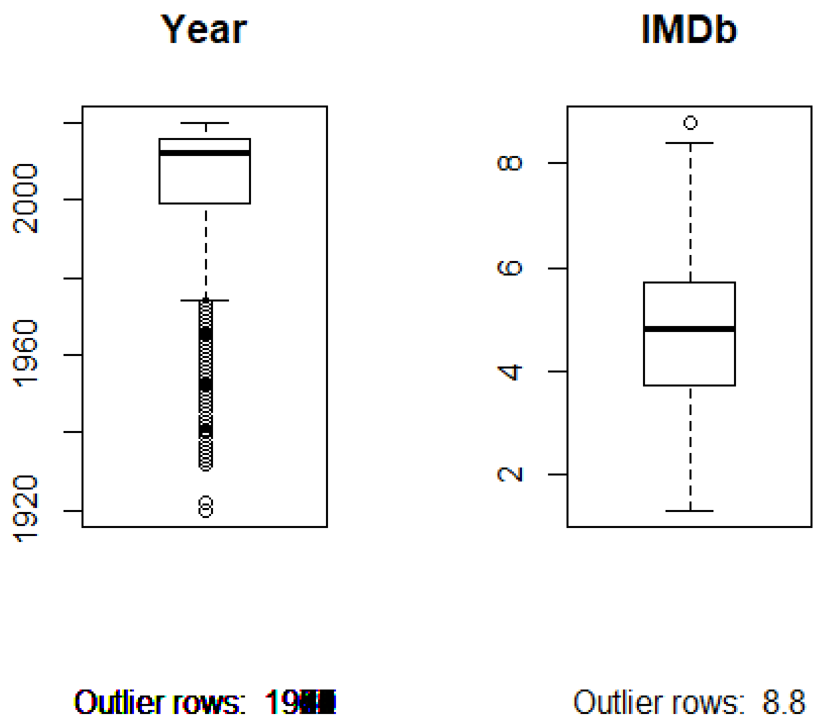
Firstly, we will check the correlation between the year and the IMDb of Horror movies.




```r
#check correlation
cor(horror_movies$Year,horror_movies$IMDb)

## [1] -0.1481772
```

The plot shows that the number of Horror movies decrease since 1960, a bunch of horror movies are released between 2000-2020. This proves that the movie production has more consideration to horror type. However, the plot also show there is a high range of scorings in these two decades. The assumption we have here is that the number of bad movies is very large because since the demand of people to horror movies is increasing, the movie production released a lot of horror movies to adapt, but they did not care about the quality of it. Next, it is necessary to check the outliers to know the issues better.

The correlation for this model means as the year increases, the IMDb scores decrease.

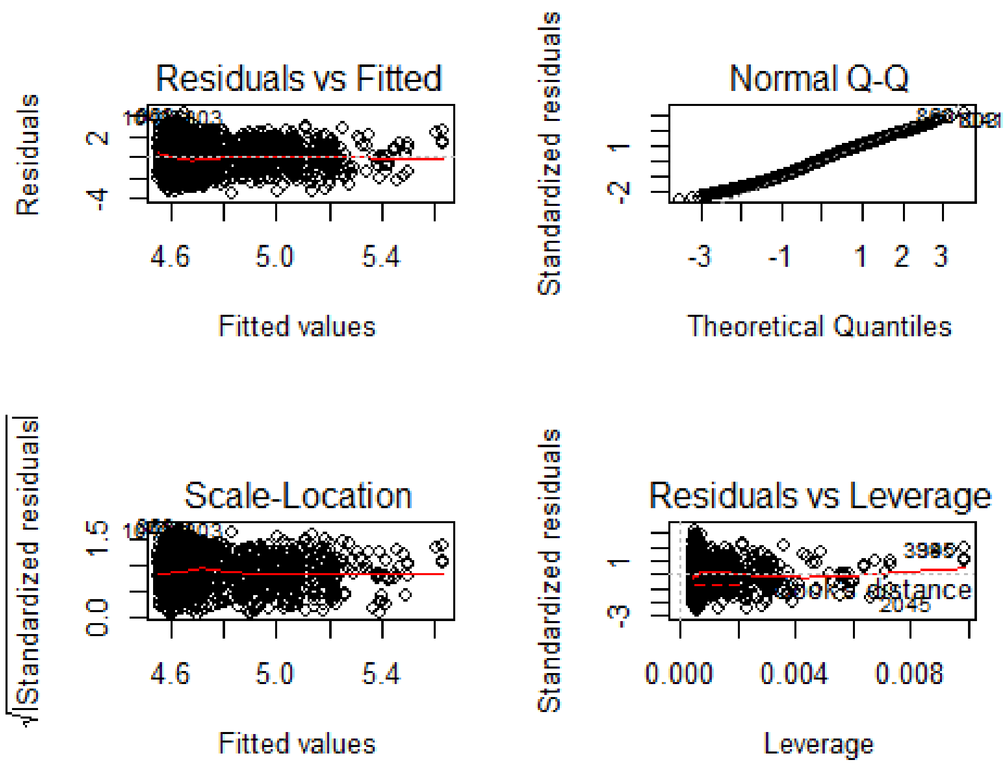


As the plot shows, meanwhile the year column has many outliers which are the past movies, the IMDb just has very few outliers, an outlier it has is 8.8 score. The mean in Year boxplot is around 2010 and in the IMDb is around 5, which quite bad. Because we do not have many appropriate independent variables in this data set, we just use the year for the prediction.

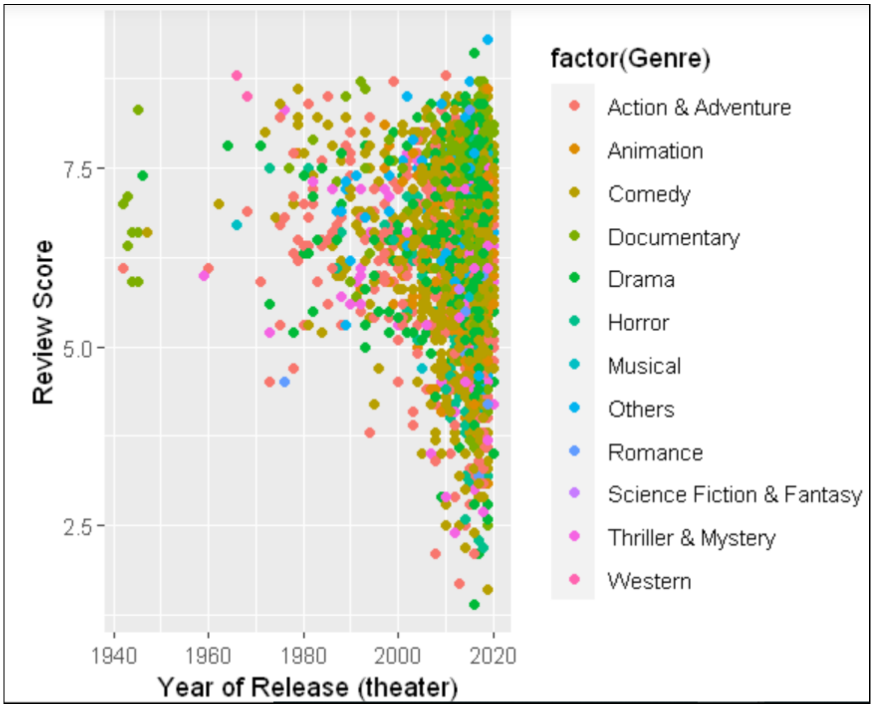
```
## Call:
## lm(formula = IMDb ~ Year, data = horror_movies)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.4320 -1.0454  0.0170  0.9871  4.1520
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 26.417698   3.086755   8.558  < 2e-16 ***
## Year        -0.010825   0.001541  -7.026 2.82e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.324 on 2199 degrees of freedom
## Multiple R-squared:  0.02196,    Adjusted R-squared:  0.02151
## F-statistic: 49.37 on 1 and 2199 DF,  p-value: 2.821e-12
```

```r
AIC(model)

## [1] 7486.876
```

The residuals indicate the close of the predictor points to actual data. The median residuals mean the actual value is more than the predicted value; thus, the prediction is not high. For the coefficient, every unit of year increase, the scores will decrease 0.010. The p value less than 0.05 shows this evidence is reliable. However, when we conduct the AIC to know the accuracy of this model, the value is too high so this model is not actually for an accurate prediction for future data, this is because we do not have many predictor variables.



For plot 1 (Residuals vs Fitted), the points show the residuals of model have non-linear patterns. For plot 2 (Normal Q-Q), the residuals deviate quite severely, thus, it is not normally distributed. For plot 3 (Scale-Location), as the line is not too curl, we can believe there is no homoscedasticity in this model. For plot 4(Residuals vs Leverage), this plot helps us to find influential cases. The plot shows although there are some outliers but it does not actually affect the model.



The correlation between the runtime of a show and its independent variables. We will use the correlation concept to apply into this method. In this question, we can test the linear relationship between the runtime and other independent variables to check to see if there is any correlation between them. For example, we can create a linear scatter plot on r to understand whether and how the runtime of a movie is affected by other independent variables like the Age column or the Review column. Then using the cor() function in r, we can find the correlation coefficient. If the correlation coefficient is closer to -1 or +1 then there is a strong negative or positive linear relationship depending on whether it is closer to -1 or + 1 respectively. If the correlation coefficient is closer to 0, then it is a weak linear relationship.

This test would show whether certain variables like the Year or age affect the runtime of the movies. For instance, it is useful to study whether movies having lower age rating can affect the runtime. In other words, whether children or family movies would tend to have a lower screening time to cater to those audiences. Or whether older movies can be longer than newer movies. The runtime between a movie released in 1980s and a movie released in 2010s.

CONCLUSION:

It is interesting to explore data from movie streaming platforms. After applying Chi-squared test and the correlation method, we have seen many potential factors from the data. This is important for the model because we can base on it to predict the factor in the future accurately or even it is used for machine learning, which is very important for any platforms.

References:

Bluman, A. G. (2009). Elementary statistics: A step by step approach. New York, NY: McGraw-Hill Higher Education.

Chi-Square Test of Independence in R. Retrieved from http://www.sthda.com/english/wiki/chi-square-test-of-independence-in-r