Project overview:

The process of extracting data from a webpage is known as web scraping. This data is gathered and then exported in a way that the user will find more valuable. A spreadsheet or an API, for example. There are many techniques for web scraping. we have used r studio tool to scraping our data table.

Data pre-processing techniques are used when the data is inconsistent, which indicates that the data is not recorded in accordance with the restrictions on the column, noisy, which may contain a variety of mistakes or outliers, and incomplete, which indicates that some attribute value is missing. In our given dataset initially, I have seen some missing value. I tried to fix it. After changing format, I have added a new column using another column value what had been given in our question condition. After adding column then I tried to handle categorical value to numerical value what is our discretization part.

Data Visualization is the approach used to offer patterns in the data using visual cues such as graphs, charts, maps, and many more. we have used scatter plot to represent our data visualization.

Project solution design:

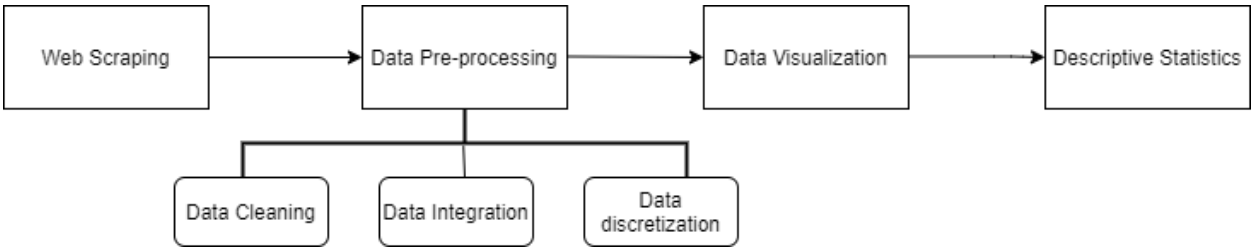


Fig: project processing Block diagram

Data Collection via Web Scraping:

we use R studio as a tool to scrap. we have selected IMDb movie website to collect our data sets. AT first we import our basic library function rvest install and then we start to scrap the data from IMDb movie website. In this process, we use a selector gadget to simply select data on a website and it will determine its HTML/CSS tags, ids and classes.

Code:

```
library(rvest)
library(dplyr)
link="https://www.imdb.com/search/title/?title_type=feature&num_votes=25000,&genres=adventure&sor
t=user_rating,desc"
page = read_html(link)
name = page %>% html_nodes(".lister-item-header a") %>% html_text()
year = page %>% html_nodes(".text-muted.unbold") %>% html_text()
rating = page %>% html_nodes(".ratings-imdb-rating strong") %>% html_text()
RunTime = page %>% html_nodes(".runtime") %>% html_text()
US_Box_Office =page %>% html_nodes(".ghost~ .text-muted+ span") %>% html_text()
PgRating =page %>% html_nodes(".certificate") %>% html_text()
length(US_Box_Office)<-length(name)
length(PgRating)<-length(name)
movies = data.frame(name, year, rating, RunTime, US_Box_Office,PgRating, stringsAsFactors =
FALSE)
write.csv(movies, "movies.csv")
```

Output

	name	year	rating	US_Box_Office	RunTime	PgRating
1	The Lord of the Rings: The Return of the King	(2003)	9.0	$377.85M	201 min	PG-13
2	Inception	(2010)	8.8	$292.58M	148 min	PG-13
3	The Lord of the Rings: The Fellowship of the Ring	(2001)	8.8	$315.54M	178 min	PG-13
4	The Lord of the Rings: The Two Towers	(2002)	8.8	$342.55M	179 min	PG-13
5	Il buono, il brutto, il cattivo	(1966)	8.8	$6.10M	161 min	PG
6	777 Charlie	(2022)	8.8	$290.48M	164 min	PG-13
7	The Empire Strikes Back	(1980)	8.7	$188.02M	124 min	PG
8	Interstellar	(2014)	8.6	$322.74M	169 min	PG
9	Star Wars	(1977)	8.6	$10.06M	121 min	R
10	Sen to Chihiro no kamikakushi	(2001)	8.6	$187.71M	125 min	G
11	Gladiator	(2000)	8.5	$210.61M	155 min	PG-13
12	Back to the Future	(1985)	8.5	$422.78M	116 min	PG
13	The Lion King	(1994)	8.5	$858.37M	88 min	PG
14	Kaithi	(2019)	8.5	$190.24M	145 min	R
15	Avengers: Endgame	(2019)	8.4	$248.16M	181 min	PG-13
16	Spider-Man: Into the Spider-Verse	(2018)	8.4	$85.16M	117 min	PG
17	Raiders of the Lost Ark	(1981)	8.4	$678.82M	115 min	R
18	Aliens	(1986)	8.4	$209.73M	137 min	PG
19	Avengers: Infinity War	(2018)	8.4	$223.81M	149 min	PG
20	Coco	(I) (2017)	8.4	$120.54M	105 min	PG-13
21	WALL·E	(2008)	8.4	$56.95M	98 min	PG-13
22	Inglourious Basterds	(2009)	8.3	$309.13M	153 min	PG-13
23	2001: A Space Odyssey	(1968)	8.3	$191.80M	149 min	PG
24	Star Wars: Episode VI - Return of the Jedi	(1983)	8.3	$293.00M	131 min	PG-13
25	Toy Story	(1995)	8.3	$44.82M	81 min	R
26	Up	(2009)	8.3	$2.38M	96 min	PG

27	Lawrence of Arabia	(1962)	8.3	$415.00M	218 min	R
28	Mononoke-hime	(1997)	8.3	$13.28M	134 min	R
29	Toy Story 3	(2010)	8.3	$804.75M	103 min	PG-13
30	North by Northwest	(1959)	8.3	$402.45M	136 min	PG-13
31	Kantara	(2022)	8.3	$4.71M	148 min	G
32	Spider-Man: No Way Home	(2021)	8.2	$380.84M	148 min	PG
33	Jurassic Park	(1993)	8.2	$197.17M	127 min	*NA*
34	Hauru no ugoku shiro	(2004)	8.2	$47.70M	119 min	*NA*
35	Finding Nemo	(2003)	8.2	$1.23M	100 min	*NA*
36	Indiana Jones and the Last Crusade	(1989)	8.2	$12.10M	127 min	*NA*
37	Kimetsu no Yaiba: Mugen Ressha-Hen	(2020)	8.2	$5.01M	117 min	*NA*
38	Monty Python and the Holy Grail	(1975)	8.2	$154.06M	91 min	*NA*
39	The Great Escape	(1963)	8.2	$59.10M	172 min	*NA*
40	The Treasure of the Sierra Madre	(1948)	8.2	$305.41M	126 min	*NA*
41	Klaus	(2019)	8.2	$52.29M	96 min	*NA*
42	Le salaire de la peur	(1953)	8.2	$381.01M	131 min	*NA*
43	Dersu Uzala	(1975)	8.2	$206.45M	142 min	*NA*
44	Mad Max: Fury Road	(2015)	8.1	$217.58M	120 min	*NA*
45	The Grand Budapest Hotel	(2014)	8.1	*NA*	99 min	*NA*
46	Pirates of the Caribbean: The Curse of the Black Pearl	(2003)	8.1	*NA*	143 min	*NA*
47	Stand by Me	(1986)	8.1	*NA*	89 min	*NA*
48	Harry Potter and the Deathly Hallows: Part 2	(2011)	8.1	*NA*	130 min	*NA*
49	Ratatouille	(2007)	8.1	*NA*	111 min	*NA*
50	How to Train Your Dragon	(2010)	8.1	*NA*	98 min	*NA*

Fig: Scrapping datasets table

Data Pre-processing:

Now the most important phase of the data analysis starts which is data pre- processing. We are going to use pre-processing techniques on these two datasets to prepare a complete dataset for analysis and visualization.

Data Cleaning

At first, we clean the data in year, runtime and Us_Box_Office.

Code:

```
newDatasets$US_Box_Office <- gsub("\\$", "", newDatasets$US_Box_Office
)
newDatasets$US_Box_Office <- gsub("M", "", newDatasets$US_Box_Office)
newDatasets$RunTime <- gsub("min", "", newDatasets$RunTime)
newDatasets$year <- gsub("[()]", "", newDatasets$year)


newDatasets$US_Box_Office <- as.numeric(newDatasets$US_Box_Office)
```

Output:

	name	year	rating	US_Box_Office	RunTime	PgRating
1	The Lord of the Rings: The Return of the King	2003	9.0	377.85	201	PG-13
2	Inception	2010	8.8	292.58	148	PG-13
3	The Lord of the Rings: The Fellowship of the Ring	2001	8.8	315.54	178	PG-13
4	The Lord of the Rings: The Two Towers	2002	8.8	342.55	179	PG-13
5	Il buono, il brutto, il cattivo	1966	8.8	6.10	161	PG
6	777 Charlie	2022	8.8	290.48	164	PG-13
7	The Empire Strikes Back	1980	8.7	188.02	124	PG
8	Interstellar	2014	8.6	322.74	169	PG
9	Star Wars	1977	8.6	10.06	121	R
10	Sen to Chihiro no kamikakushi	2001	8.6	187.71	125	G
11	Gladiator	2000	8.5	210.61	155	PG-13
12	Back to the Future	1985	8.5	422.78	116	PG
13	The Lion King	1994	8.5	858.37	88	PG
14	Kaithi	2019	8.5	190.24	145	R

Showing 1 to 14 of 50 entries, 6 total columns

Fig: After Clean dataset

Handling the missing data: The PgRating, Box office columns value in the dataset has some missing data. This issue must be resolved before to incorporating a data set into a model; otherwise, it will seriously impact that model. So we should handle this dataset. we can handle 2 ways either Replace the data or Discard. As it is categorical values so we cannot replace. That's why decided to remove or discarded from our dataset.

Code:

```
newDatasets<- na.omit(movies)
```

Result:

17	Raiders of the Lost Ark	1981	8.4	678.82	115	R
18	Aliens	1986	8.4	209.73	137	PG
19	Avengers: Infinity War	2018	8.4	223.81	149	PG
20	Coco	I 2017	8.4	120.54	105	PG-13
21	WALL-E	2008	8.4	56.95	98	PG-13
22	Inglourious Basterds	2009	8.3	309.13	153	PG-13
23	2001: A Space Odyssey	1968	8.3	191.80	149	PG
24	Star Wars: Episode VI - Return of the Jedi	1983	8.3	293.00	131	PG-13
25	Toy Story	1995	8.3	44.82	81	R
26	Up	2009	8.3	2.38	96	PG
27	Lawrence of Arabia	1962	8.3	415.00	218	R
28	Mononoke-hime	1997	8.3	13.28	134	R
29	Toy Story 3	2010	8.3	804.75	103	PG-13
30	North by Northwest	1959	8.3	402.45	136	PG-13
31	Kantara	2022	8.3	4.71	148	G
32	Spider-Man: No Way Home	2021	8.2	380.84	148	PG

Showing 7 to 32 of 32 entries, 6 total columns

Figure: After handling missing values

Data Integration

Data integration is a process where we need to integrate new data from different source or tabl
e

Data discretization

Data discretization is a one kind of reducing process.in can be categorical to numerical. Let's no
w imagine that we must add a new column of data to our data table based on estimates of the
US_Box_Office. Here is my code to add new column name “Catagory”.In our dataset ‘categories
’ column had four categorical value. which I have replace 4 numerical value: “flop” replaced by
4. “Average” replaced by 3. “Super Hit” replaced by 2 and “Block Blaster” replaced by 1.

Code:

```
newDatasets$Category <- as.factor(
ifelse(newDatasets$US_Box_Office>=0.0  &  newDatasets$US_Box_Office<=50.0, 'f
lop',
ifelse(newDatasets$US_Box_Office>50.0 & newDatasets$US_Box_Office<=100.0, 'Av
erage',
ifelse(newDatasets$US_Box_Office>100.0 & newDatasets$US_Box_Office<=200.0, 'S
uper Hit',
ifelse(newDatasets$US_Box_Office>200, 'Block Blaster','non')))))
```

Output:

	name	year	rating	US_Box_Office	RunTime	PgRating	Category
1	The Lord of the Rings: The Return of the King	2003	9.0	377.85	201	PG-13	Block Blaster
2	Inception	2010	8.8	292.58	148	PG-13	Block Blaster
3	The Lord of the Rings: The Fellowship of the Ring	2001	8.8	315.54	178	PG-13	Block Blaster
4	The Lord of the Rings: The Two Towers	2002	8.8	342.55	179	PG-13	Block Blaster
5	Il buono, il brutto, il cattivo	1966	8.8	6.10	161	PG	flop
6	777 Charlie	2022	8.8	290.48	164	PG-13	Block Blaster
7	The Empire Strikes Back	1980	8.7	188.02	124	PG	Super Hit
8	Interstellar	2014	8.6	322.74	169	PG	Block Blaster
9	Star Wars	1977	8.6	10.06	121	R	flop
10	Sen to Chihiro no kamikakushi	2001	8.6	187.71	125	G	Super Hit
11	Gladiator	2000	8.5	210.61	155	PG-13	Block Blaster
12	Back to the Future	1985	8.5	422.78	116	PG	Block Blaster
13	The Lion King	1994	8.5	858.37	88	PG	Block Blaster
14	Kaithi	2019	8.5	190.24	145	R	Super Hit

Showing 1 to 14 of 32 entries, 8 total columns

Figure: After integrating a new column based on US_Box_Office

Code:

```
newDatasets$label <- as.factor(
  ifelse(newDatasets$Category=='flop', 4,
       ifelse(newDatasets$Category=='Average', 3,
            ifelse(newDatasets$Category=='Super Hit', 2,
                 ifelse(newDatasets$Category=='Block Blaster', 1,'non')))))
```

Output:

	name	year	rating	US_Box_Office	RunTime	PgRating	Category	label
1	The Lord of the Rings: The Return of the King	2003	9.0	377.85	201	PG-13	Block Blaster	1
2	Inception	2010	8.8	292.58	148	PG-13	Block Blaster	1
3	The Lord of the Rings: The Fellowship of the Ring	2001	8.8	315.54	178	PG-13	Block Blaster	1
4	The Lord of the Rings: The Two Towers	2002	8.8	342.55	179	PG-13	Block Blaster	1
5	Il buono, il brutto, il cattivo	1966	8.8	6.10	161	PG	flop	4
6	777 Charlie	2022	8.8	290.48	164	PG-13	Block Blaster	1
7	The Empire Strikes Back	1980	8.7	188.02	124	PG	Super Hit	2
8	Interstellar	2014	8.6	322.74	169	PG	Block Blaster	1
9	Star Wars	1977	8.6	10.06	121	R	flop	4
10	Sen to Chihiro no kamikakushi	2001	8.6	187.71	125	G	Super Hit	2
11	Gladiator	2000	8.5	210.61	155	PG-13	Block Blaster	1
12	Back to the Future	1985	8.5	422.78	116	PG	Block Blaster	1
13	The Lion King	1994	8.5	858.37	88	PG	Block Blaster	1
14	Kaithi	2019	8.5	190.24	145	R	Super Hit	2

Showing 1 to 14 of 32 entries, 8 total columns

Figure: After converting categorical values to numerical values

Descripting statistics

Data can be described or summarized using descriptive statistics in relevant and practical ways.

Code:

```
#max

max(newDatasets$US_Box_Office) #min
min(newDatasets$US_Box_Office) #min
mean(newDatasets$US_Box_Office) #median
median(newDatasets$US_Box_Office) #mode
mode<- function(x){ value<- unique(x)table<- tabulate(match(x,value)) value[table ==max(table)]
}
mode(newDatasets$US_Box_Office) #Range
max(newDatasets$US_Box_Office) - min(newDatasets$US_Box_Office) #variance
var(newDatasets$US_Box_Office) #Standard Deviation
sd(newDatasets$US_Box_Office) #Quartile
quantile(newDatasets$US_Box_Office) #Interquartile
IQR(newDatasets$US_Box_Office)
```

Output:

```
> max(newDatasets$US_Box_Office)
[1] 858.37
> min(newDatasets$US_Box_Office)
[1] 2.38
> mean(newDatasets$US_Box_Office)
[1] 265.655
> median(newDatasets$US_Box_Office)
[1] 235.985
> mode<- function(x)
+     { value<- unique(x)
+     table<- tabulate(match(x,value))
+     value[table ==max(table)]
+ }
> mode(newDatasets$US_Box_Office)
 [1] 377.85 292.58 315.54 342.55   6.10 290.48 188.02 322.74  10.06 187.71 210.61 422.78 858.37
[14] 190.24 248.16  85.16 678.82 209.73 223.81 120.54  56.95 309.13 191.80 293.00  44.82   2.38
[27] 415.00  13.28 804.75 402.45   4.71 380.84
> max(newDatasets$US_Box_Office) - min(newDatasets$US_Box_Office)
[1] 855.99
> var(newDatasets$US_Box_Office)
[1] 46163.16
> sd(newDatasets$US_Box_Office)
[1] 214.8561
> quantile(newDatasets$US_Box_Office)
     0%      25%      50%      75%     100%
  2.380 111.695 235.985 351.375 858.370
> IQR(newDatasets$US_Box_Office)
[1] 239.68
```

Data Visualization:

Data Visualization is the approach used to offer patterns in the data using visual cues such as graphs, charts, maps, and many more. This is helpful because it facilitates intuitive and simple understanding of the vast amounts of data, allowing for better decision-making.

1)Grouping

Code:

```
ggplot(data=newDatasets,mapping=aes(
 x=US_Box_Office,y=RunTime,color=PgRating,shape=PgRating))+
 geom_point(colour="blue",alpha=1,size=2)+
 labs(titles="Run Time vs Box Office")
```

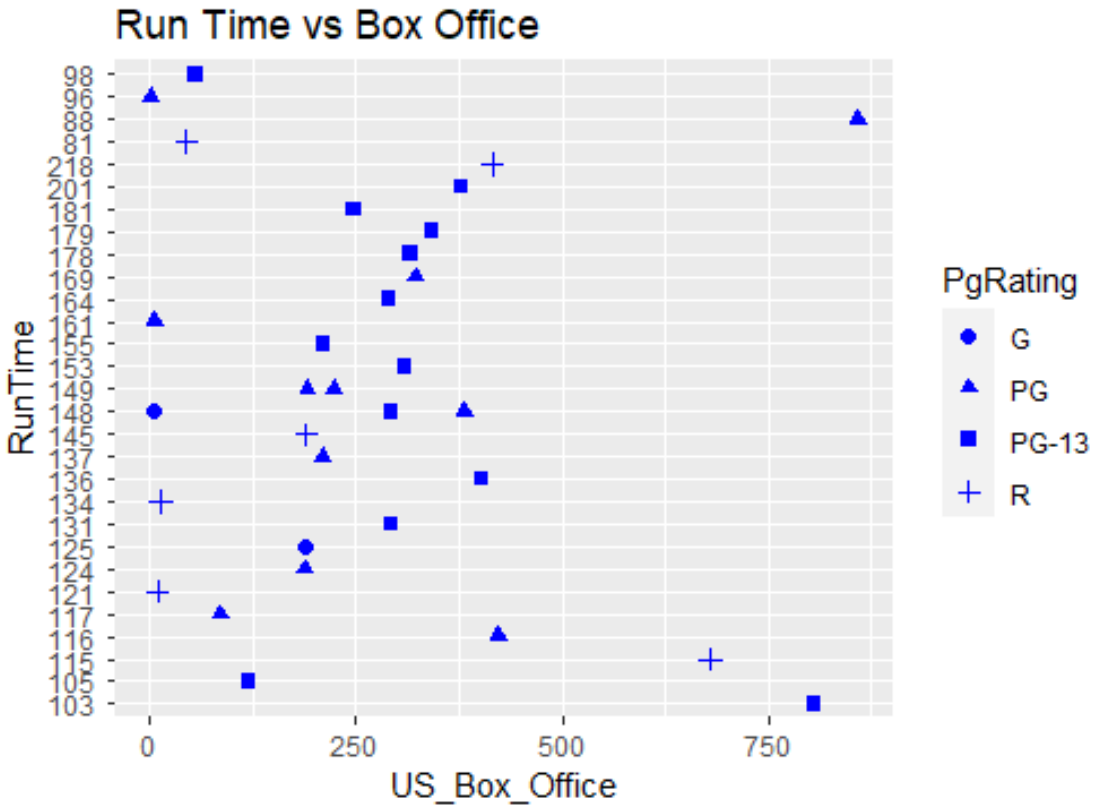


Fig: After plotting scatter plot X=us_Box_office and y=RunTime & Grouping

2) plot of Rating vs Box Office

Code:

```
ggplot(data=newDatasets,mapping=aes(
 x=US_Box_Office,y=rating,color=Category,shape=Category))+
 geom_point(alpha=1)+
```

```
geom_smooth(method = "lm", se = FALSE,size=1)+
theme_gray()+
scale_x_continuous(breaks = seq(0,900,200),
labels = scales::dollar,labs(x = "US_Box_Office (Million)"))+
scale_y_discrete(breaks = seq(0,10,1),labs(y ="Rating(Out of 10)"))+
scale_color_manual(values = c("red","green","blue","black")) +
labs(titles="Rating vs Box Office")
```

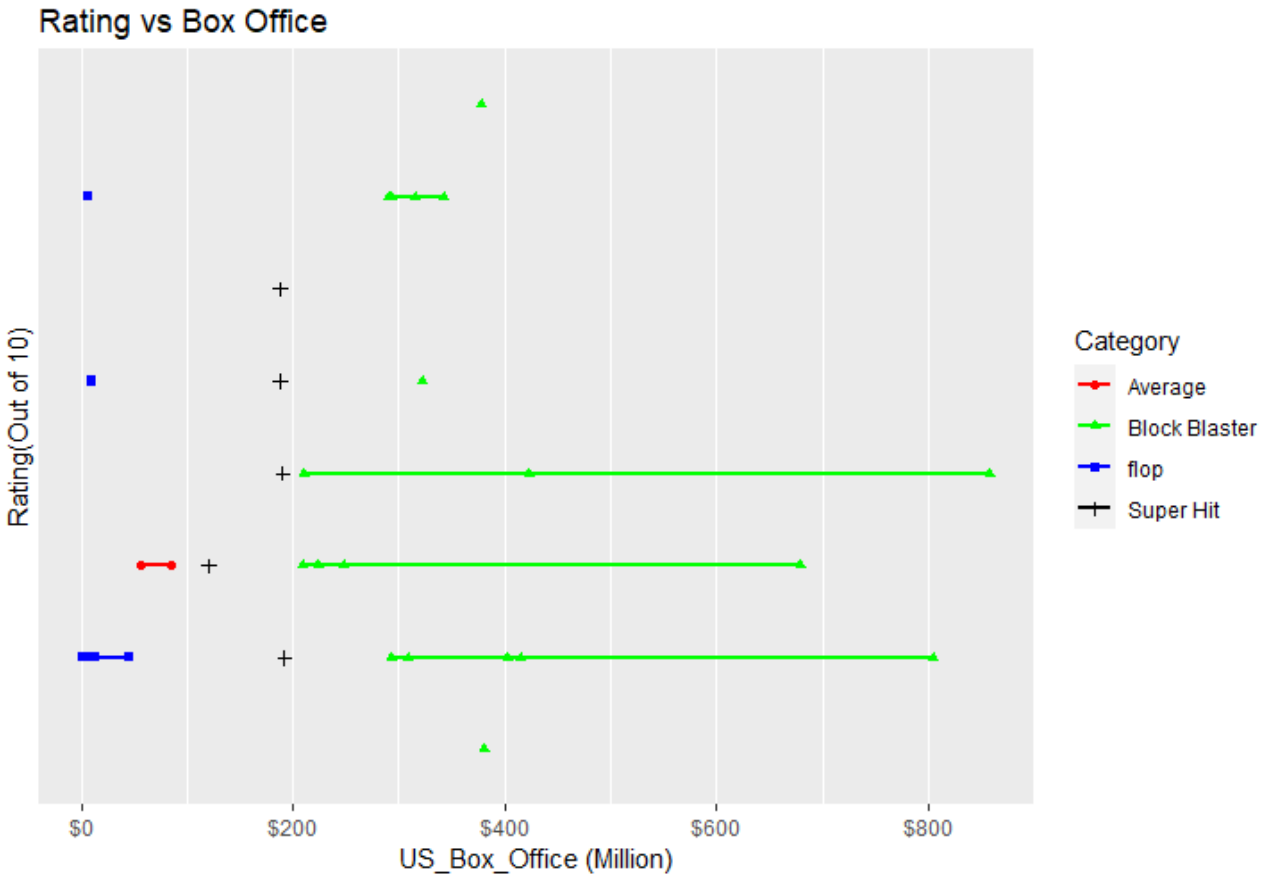


Fig: After plotting scatter plot X=us_Box_office and y=Rating & Grouping

3) After applying facet's function

Code:

```
ggplot(data=newDatasets,mapping=aes(
 x=US_Box_Office,y=rating,color=Category,shape=Category))+
 geom_point(alpha=1)+
 geom_smooth(method = "lm", se = FALSE,size=1)+
 theme_gray()+
 scale_x_continuous(breaks = seq(0,900,200),
```

```
labels = scales::dollar,labs(x = "US_Box_Office (Million)"))+
scale_y_discrete(breaks = seq(0,10,1),labs(y ="Rating(Out of 10)"))+
scale_color_manual(values = c("red","green","blue","black")) +
facet_wrap(~Category)+
labs(titles="Rating vs Box Office")
```

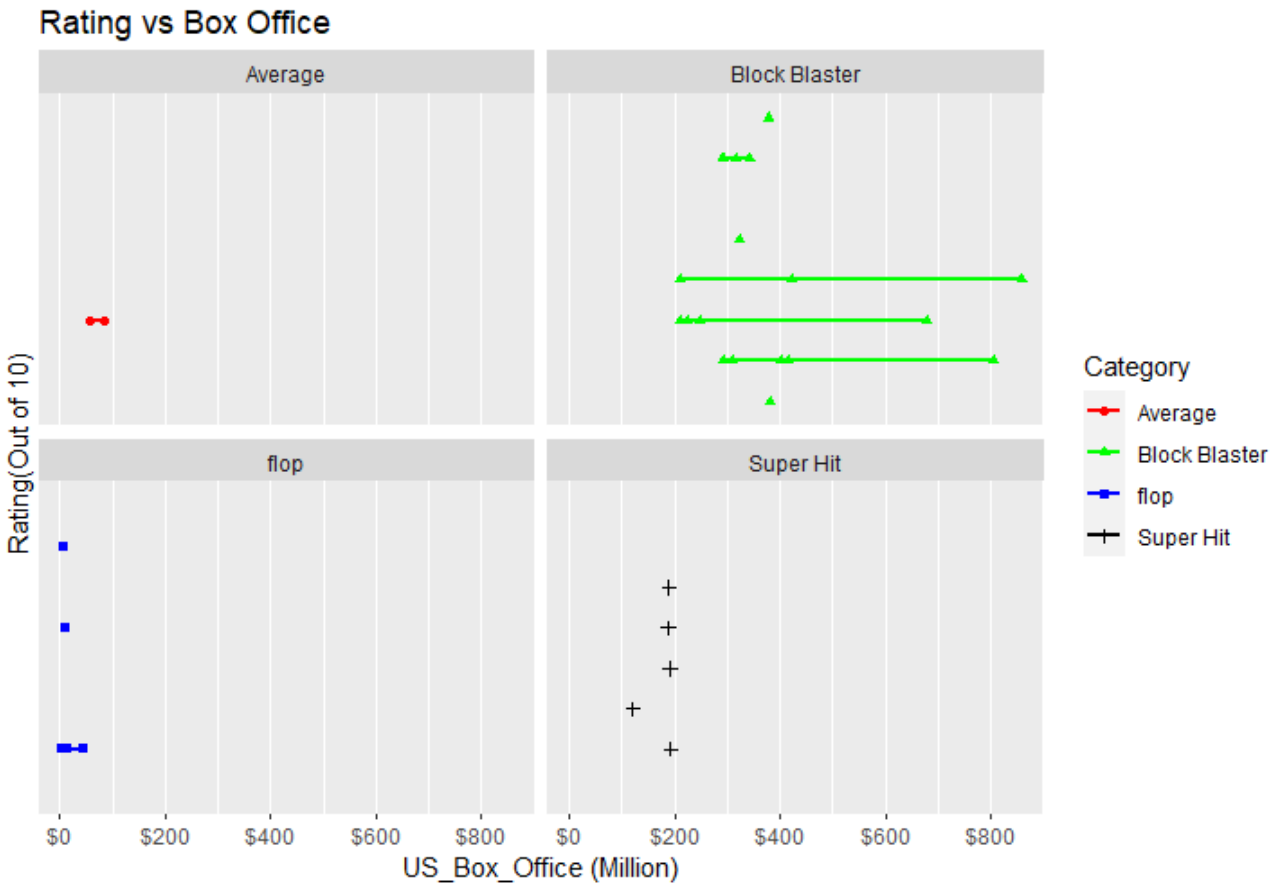


Fig: After applying linear method and change theme

4)After applying facets function & labeled

Code:

```
ggplot(data=newDatasets,mapping=aes(
 x=US_Box_Office,y=rating,color=PgRating,shape=PgRating))+
 geom_point(alpha=1)+
 geom_smooth(method = "lm", se = FALSE,size=1)+
 theme_gray()+
```

```
scale_x_continuous(breaks = seq(0,900,200),
labels = scales::dollar,labs(x = "US_Box_Office (Million)"))+
scale_y_discrete(breaks = seq(0,10,1),labs(y ="Rating(Out of 10)"))+
scale_color_manual(values = c("red","green","blue","black")) +
facet_wrap(~PgRating)+
labs(titles="Rating vs Box Office")
```

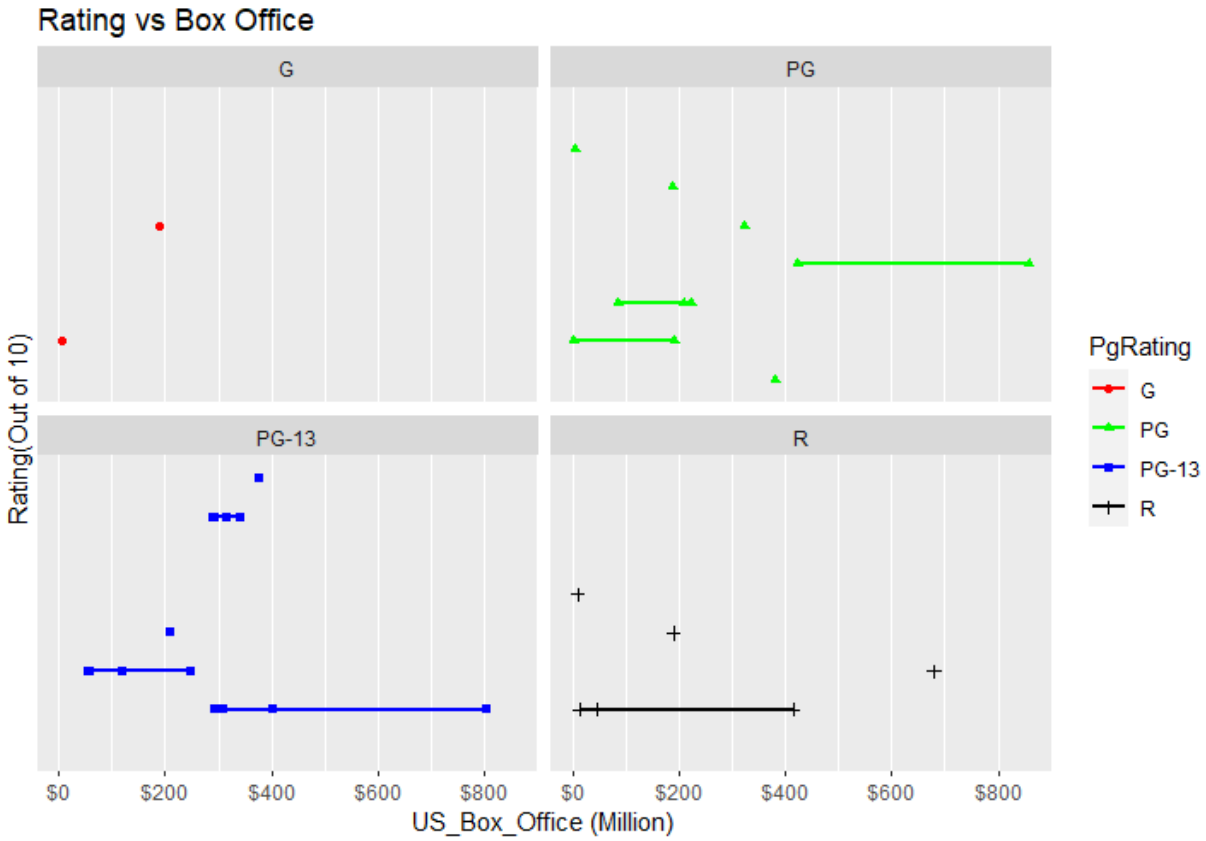


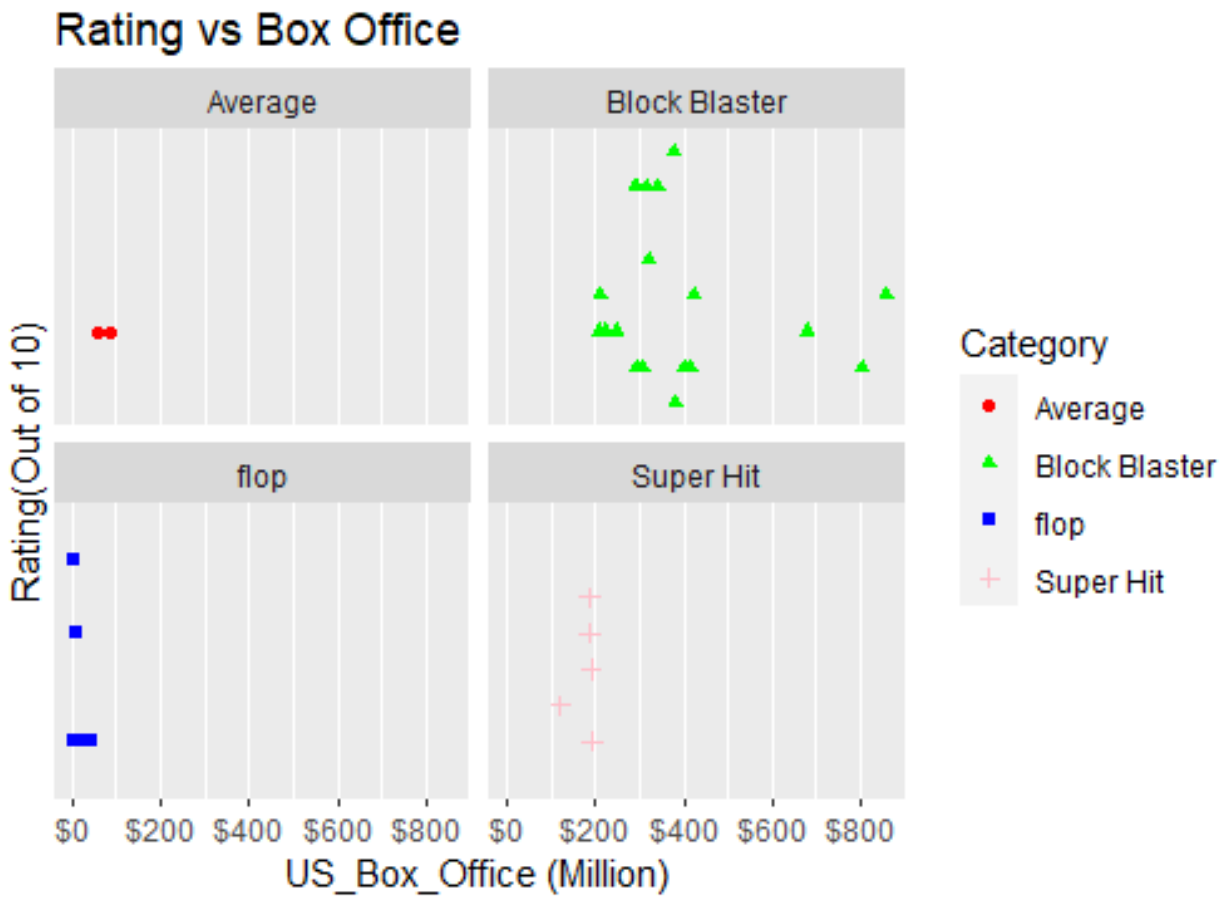
Fig: After labelled

5) let's draw a scatter plot of Rating vs Box Office

Code:

```
ggplot(data=newDatasets,mapping=aes(
  x=US_Box_Office,y=rating,color=Category,shape=Category))+
 geom_point(alpha=1)+
 #geom_smooth(method = "lm", se = FALSE,size=3)+
 theme_gray()+
 scale_x_continuous(breaks = seq(0,900,200),labels = scales::dollar,labs(x = "US_Box_Office
(Million)"))+
 scale_y_discrete(breaks = seq(0,10,1),labs(y ="Rating(Out of 10)"))+
```

```
 scale_color_manual(values = c("red","green","blue","pink")) +
 facet_wrap(~Category)+
labs(titles="Rating vs Box Office")
```



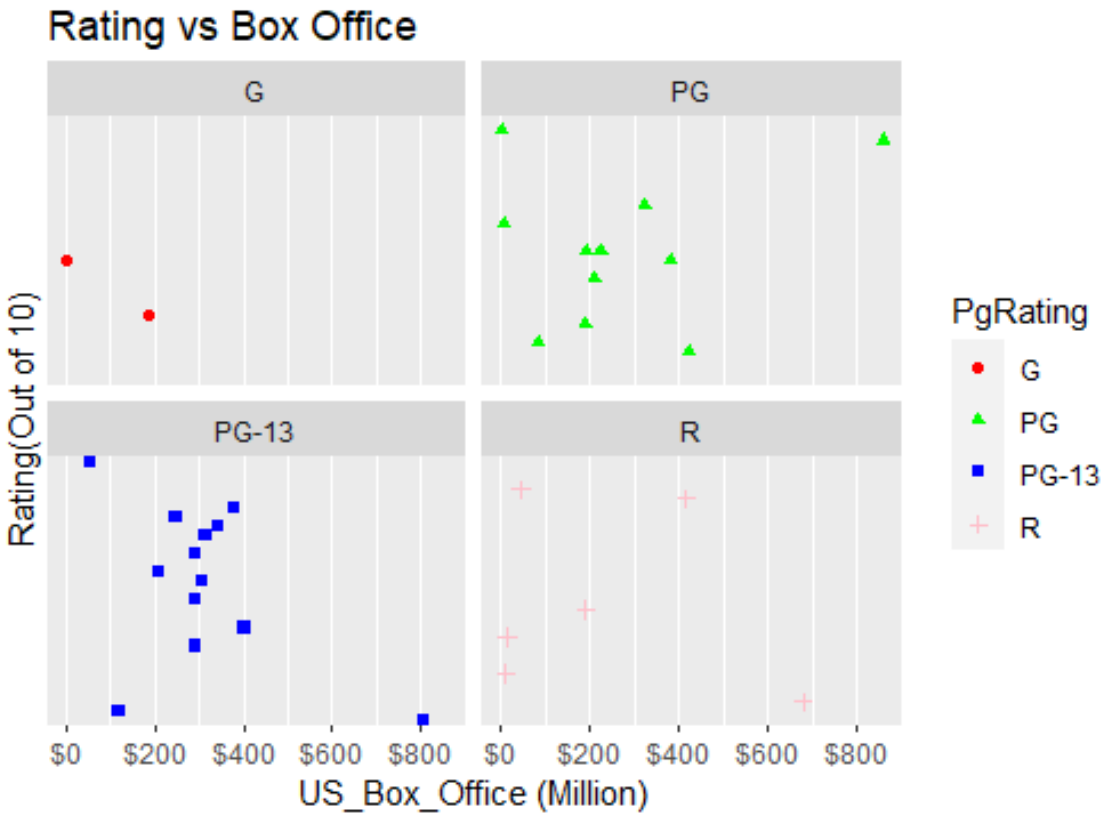
From this scatter plot, we can understand that Category are depends on Rating. Most of the Movie are blockbuster and only 2 movies are average category.

6) let's draw a scatter plot of PgRating vs Box Office

Code:

```
ggplot(data=newDatasets,mapping=aes(
 x=US_Box_Office,y=RunTime,color=PgRating,shape=PgRating))+
 geom_point(alpha=1)+
 theme_gray()+
 scale_x_continuous(breaks = seq(0,900,200),labels = scales::dollar,labs(x = "US_Box_Office
(Million)"))+
```

```
scale_y_discrete(breaks = seq(0,10,1),labs(y ="Rating(Out of 10)"))+
scale_color_manual(values = c("red","green","blue","pink")) +
facet_wrap(~PgRating)+
labs(titles="Rating vs Box Office")
```

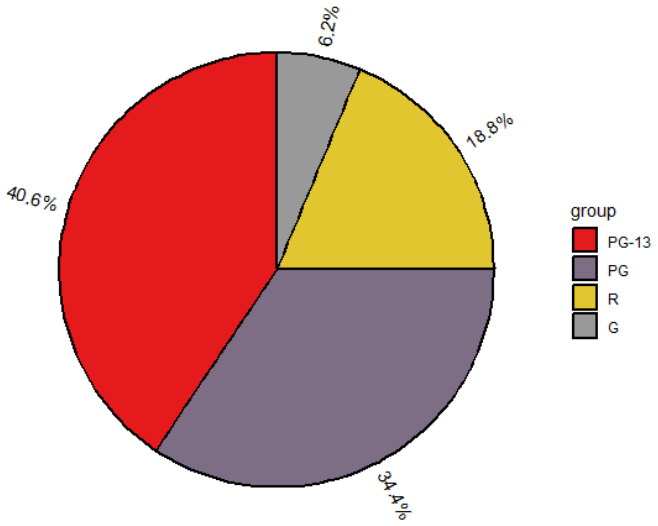


From this scatter plot, we can understand that most of the Movie are PG rated and only 2 movies are G rated. There are also some R rated movies in the list.

7) Next, we try to measure and analyse the PgRating categories that the Movies belong to

Code:

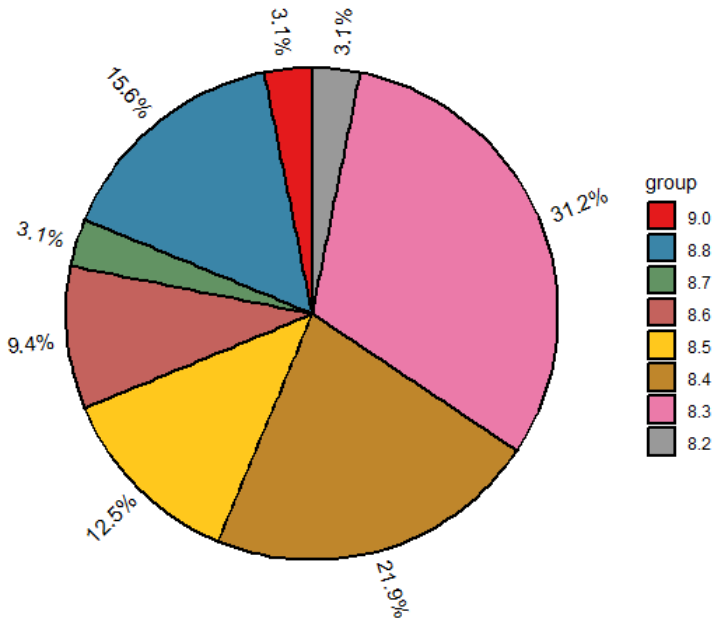
```
library(ggpie)
library(dplyr)
newDatasets %>% ggpie(group_key = "PgRating",count_type = "full", label_type =
                        "circle",
                      label_info = "ratio", label_pos = "out", nudge_x = 10)
```



8) Next, we try to measure and analyse the rating that the Movies belong to

Code:

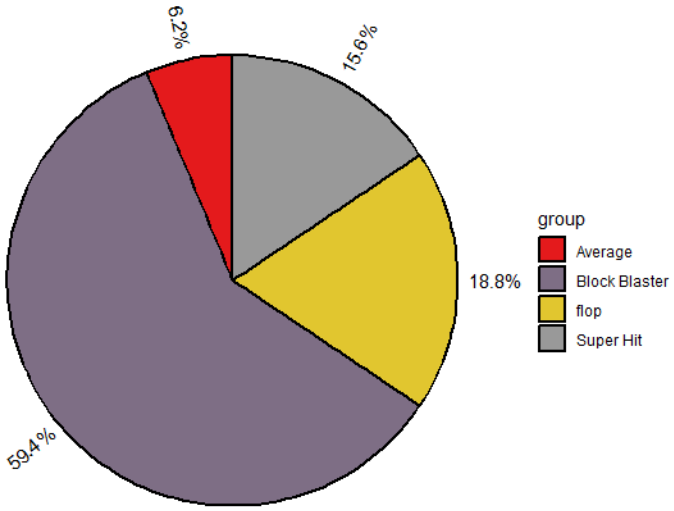
```
library(ggpie)

library(dplyr)

newDatasets %>% ggpie(group_key = "rating",count_type = "full", label_type =

                        "circle",

                      label_info = "ratio", label_pos = "out", nudge_x = 10)
```



9) Next, we try to measure and analyse the Category that the Movies belong to

Code:

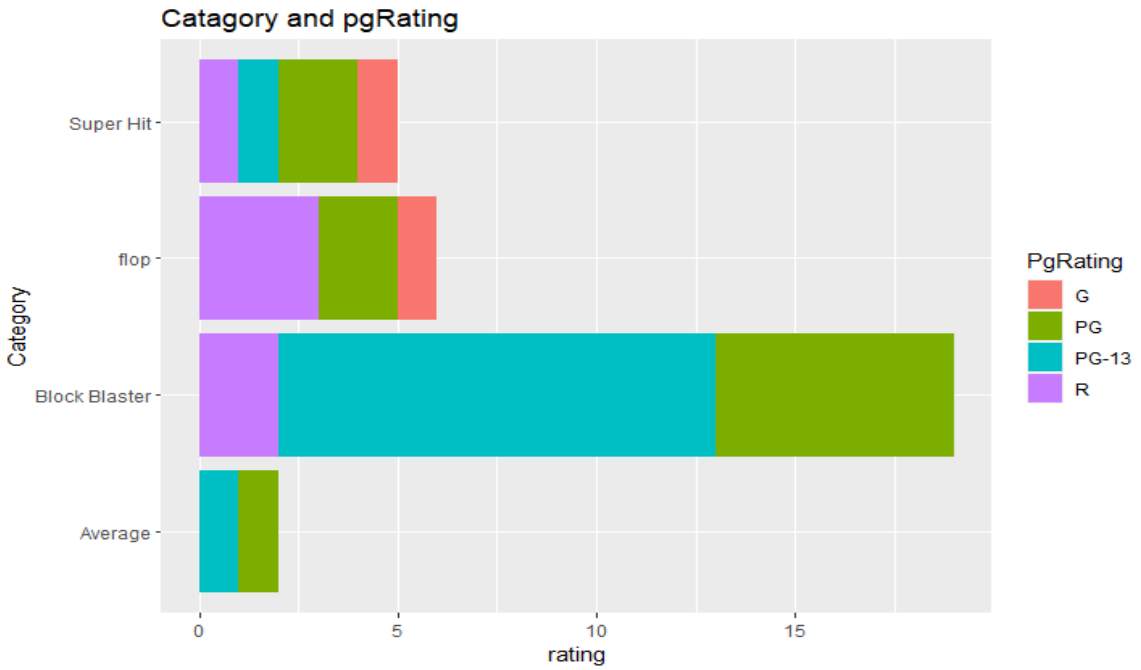
```
library(ggpie)
library(dplyr)
newDatasets %>% ggpie(group_key = "Category",count_type = "full", label_type =
                        "circle",
                      label_info = "ratio", label_pos = "out", nudge_x = 10)
```



10) Now the most important part of the visualization. We need to see the Pg Rating of the of the movies According to Category.

Code:

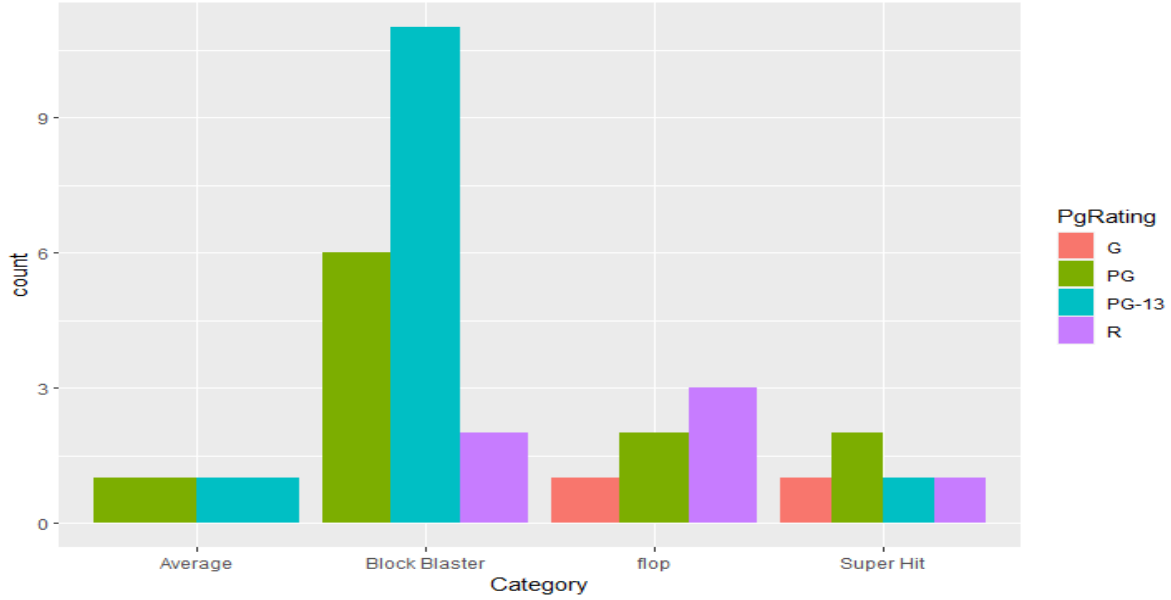
```
ggplot(newDatasets,aes(x=Category, fill=PgRating ))+
 geom_bar()+
 labs(title = "Catagory and pgRating", x ="Category",
     y="rating")+
 coord_flip()
```



11) Now we run a comparison between Category and Pg Rating

Code:

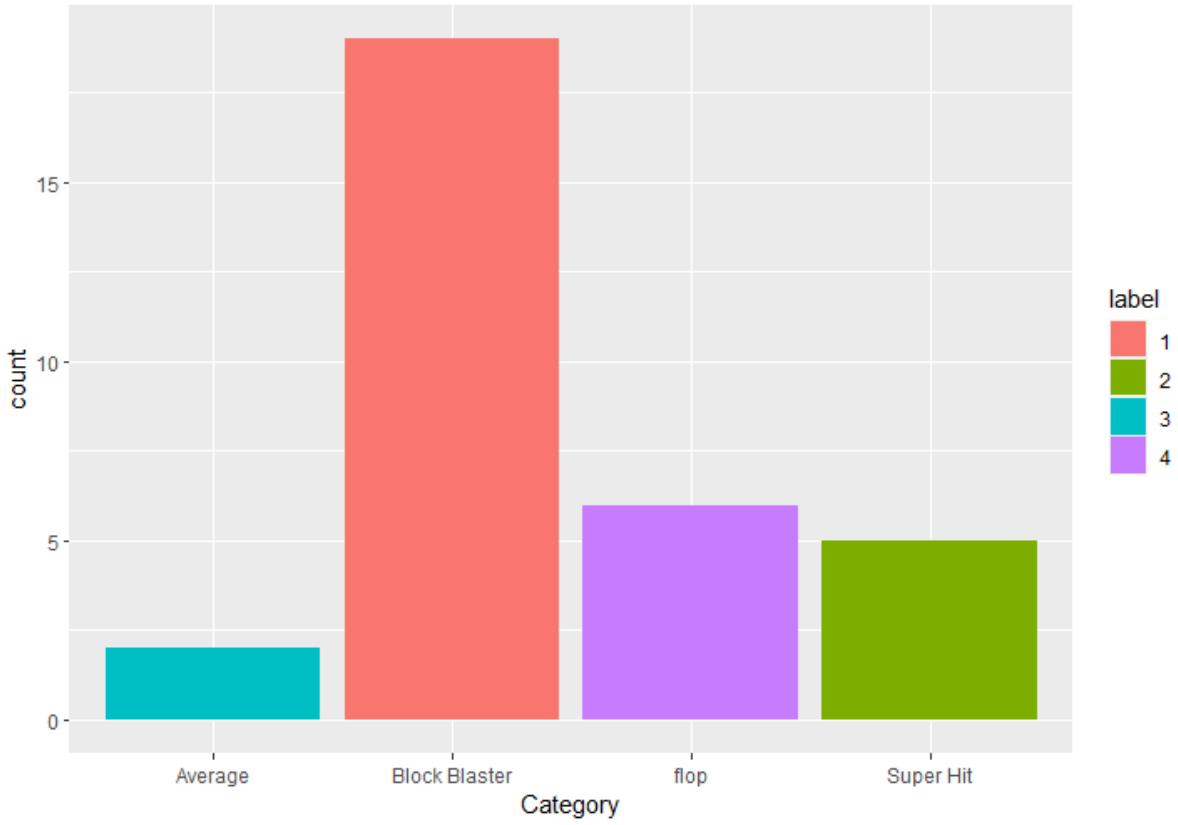
```
ggplot(newDatasets, aes(x= Category, fill= PgRating))+
  geom_bar(position = "dodge")
```



12) Now we run and identify Category with proper label

Code:

```
ggplot(newDatasets, aes(x= Category, fill= label))+
  geom_bar(position = "dodge")
```



Shiny Dashboard Implementation:

For the shiny dashboard implementation, we tried to create a reactive app based on our topic.

Code:

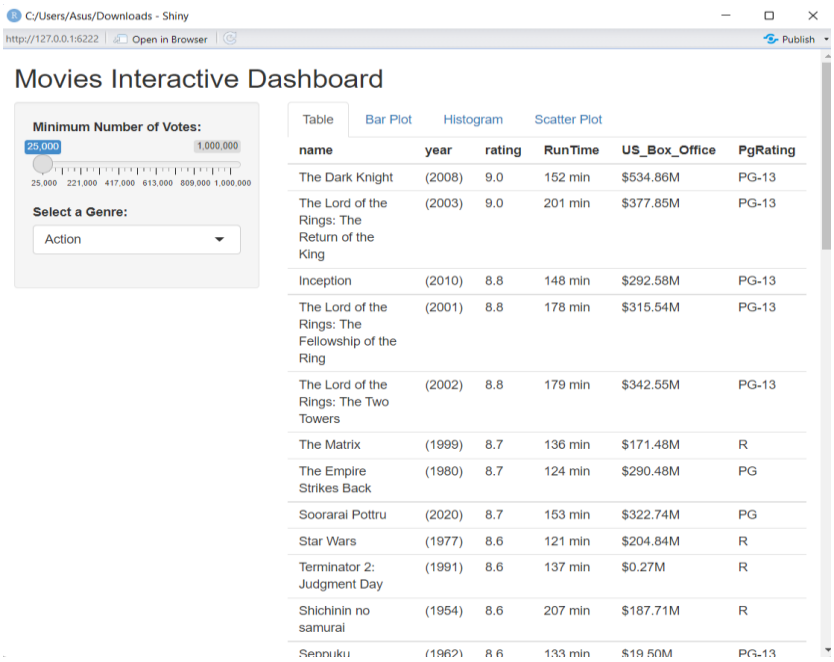
 # Define UI

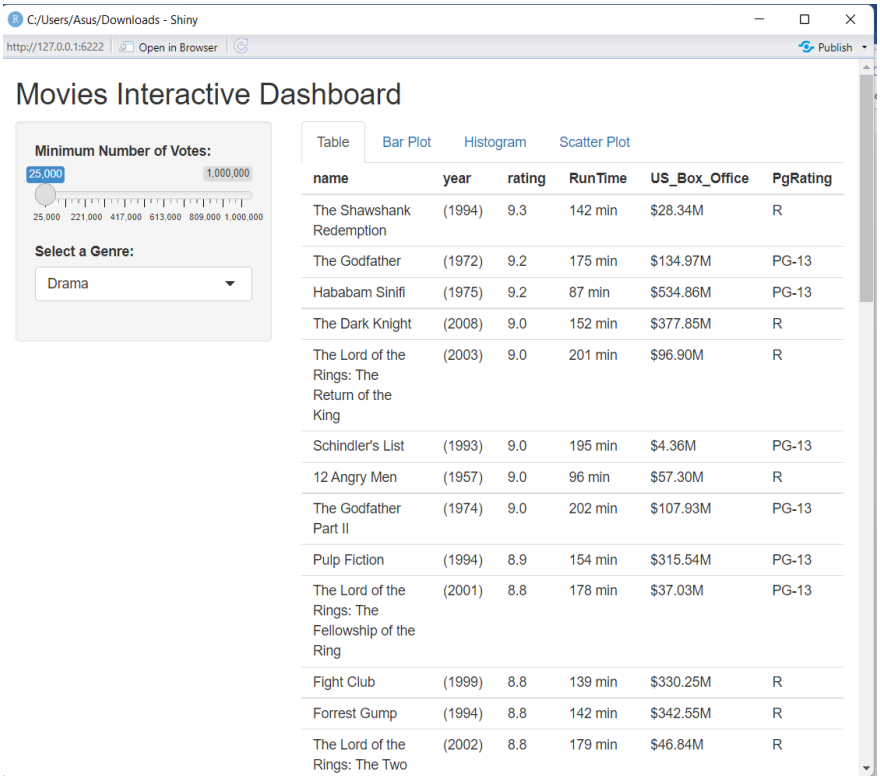
```
ui <- fluidPage(
  titlePanel("Movies Interactive Dashboard"),
  sidebarLayout(
    sidebarPanel(
      sliderInput("votes", "Minimum Number of Votes:", min = 25000, max = 1000000, value = 25000, step
= 1000),
      selectInput("genre", "Select a Genre:", choices = c("Action", "Adventure", "Comedy", "Drama",
"Horror", "Mystery", "Romance", "Sci-Fi", "Thriller"))
    ),
    mainPanel(
      tabsetPanel(
        tabPanel("Table", tableOutput("movies")),
        tabPanel("Bar Plot", plotOutput("boxoffice")),
        tabPanel("Histogram", plotOutput("ratinghist")),
        tabPanel("Scatter Plot", plotOutput("earnings "))
      )
    )
  )
)
```

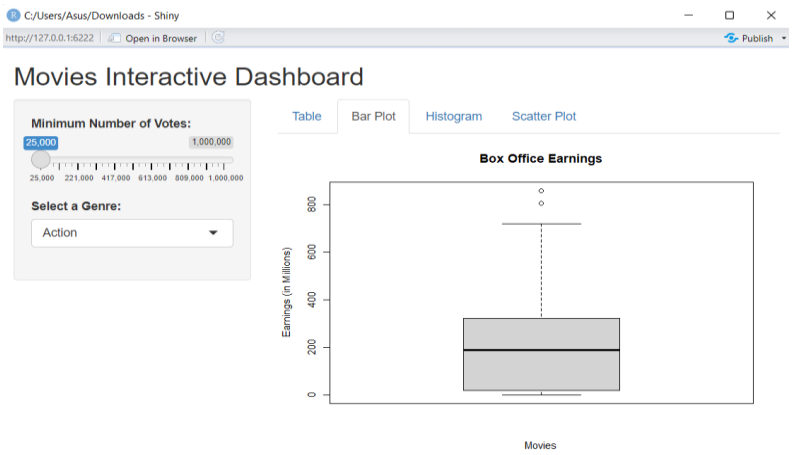
 # Define server

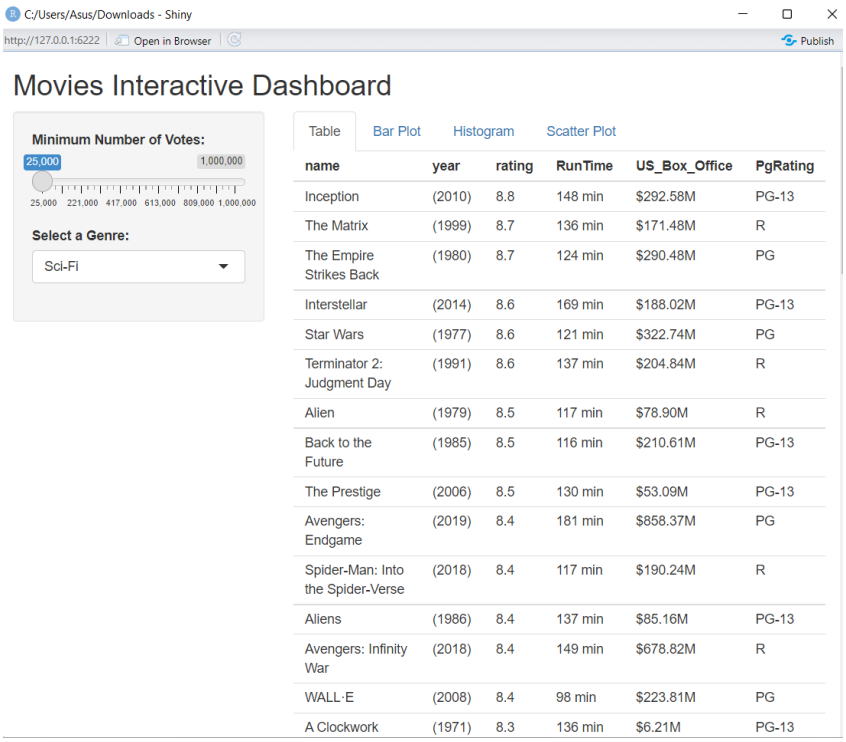
```
server <- function(input, output) {
  link <- reactive(paste0("https://www.imdb.com/search/title/?title_type=feature&num_votes=", input$votes,
",&genres=", input$genre, "&sort=user_rating,desc"))

  # Scrape data from IMDb website
  movies_data <- reactive({
    page <- read_html(link())
    name <- page %>% html_nodes(".lister-item-header a") %>% html_text()
    year <- page %>% html_nodes(".text-muted.unbold") %>% html_text()
    rating <- page %>% html_nodes(".ratings-imdb-rating strong") %>% html_text()
    RunTime <- page %>% html_nodes(".runtime") %>% html_text()
    US_Box_Office <-page %>% html_nodes(".ghost~ .text-muted+ span") %>% html_text()
    PgRating <-page %>% html_nodes(".certificate") %>% html_text()
    length(US_Box_Office) <- length(name)
    length(PgRating) <- length(name)
    movies_df <- data.frame(name, year, rating, RunTime, US_Box_Office, PgRating, stringsAsFactors = FALSE)
    return(movies_df)
  })
```

```
 # Render table of movies

 output$movies <- renderTable({
  movies_data()
 })

 # Render box plot of US box office earnings
 output$boxoffice <- renderPlot({
  boxplot(as.numeric(gsub("[^0-9.]", "", movies_data()$US_Box_Office)),
          main = "Box Office Earnings",
          xlab = "Movies",
          ylab = "Earnings (in Millions)")
 })


 # Render histogram of ratings

 output$ratinghist <- renderPlot({
  hist(as.numeric(movies_data()$rating),
       main = "Movie Ratings Histogram",
       xlab = "Rating",
       ylab = "Frequency",
       col = "blue",
       border = "white",
       breaks = seq(0, 10, 0.5))
 })


 # Render scatter plot of ratings vs. box office earnings
 output$earnings <- renderPlot({
  plot(as.numeric(gsub("[^0-9.]", "", movies_data()$US_Box_Office)),
       as.numeric(movies_data()$rating),
       xlab = "Box Office Earnings (in Millions)",
       ylab = "Rating",
       main = "Ratings vs. Box Office Earnings",
       col = "red")
 })
}

# Run the app

shinyApp(ui = ui, server = server)
```

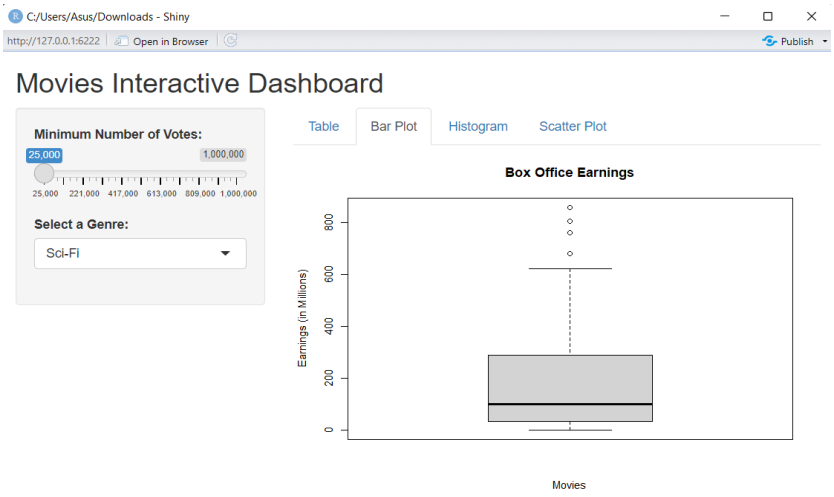

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Action
Table
Bar Plot
Histogram
Scatter Plot
name year rating RunTime US_Box_Office PgRating
The Dark Knight (2008) 9.0 152 min $534.86M PG-13
The Lord of the Rings: The Return of the King (2003) 9.0 201 min $377.85M PG-13
Inception (2010) 8.8 148 min $292.58M PG-13
The Lord of the Rings: The Fellowship of the Ring (2001) 8.8 178 min $315.54M PG-13
The Lord of the Rings: The Two Towers (2002) 8.8 179 min $342.55M PG-13
The Matrix (1999) 8.7 136 min $171.48M R
The Empire Strikes Back (1980) 8.7 124 min $290.48M PG
Soorarai Pottru (2020) 8.7 153 min $322.74M PG
Star Wars (1977) 8.6 121 min $204.84M R
Terminator 2: Judgment Day (1991) 8.6 137 min $0.27M R
Shichinin no samurai (1954) 8.6 207 min $187.71M R
Seppuku (1962) 8.6 133 min $19.50M PG-13

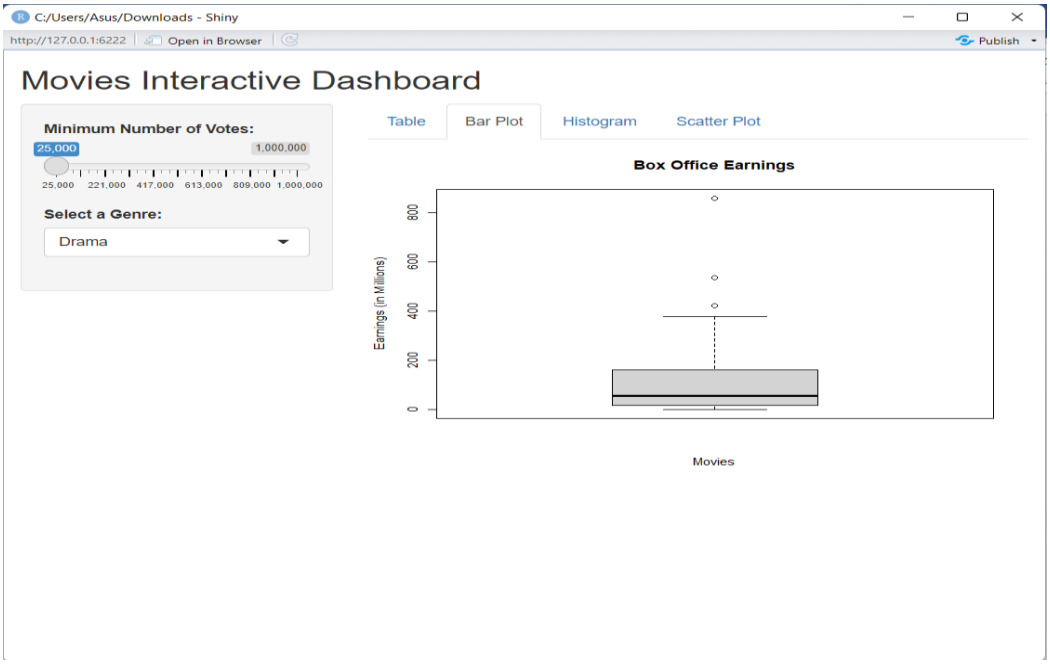

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Drama
Table
Bar Plot
Histogram
Scatter Plot
name year rating RunTime US_Box_Office PgRating
The Shawshank Redemption (1994) 9.3 142 min $28.34M R
The Godfather (1972) 9.2 175 min $134.97M PG-13
Hababam Sinifi (1975) 9.2 87 min $534.86M PG-13
The Dark Knight (2008) 9.0 152 min $377.85M R
The Lord of the Rings: The Return of the King (2003) 9.0 201 min $96.90M R
Schindler's List (1993) 9.0 195 min $4.36M PG-13
12 Angry Men (1957) 9.0 96 min $57.30M R
The Godfather Part II (1974) 9.0 202 min $107.93M PG-13
Pulp Fiction (1994) 8.9 154 min $315.54M PG-13
The Lord of the Rings: The Fellowship of the Ring (2001) 8.8 178 min $37.03M PG-13
Fight Club (1999) 8.8 139 min $330.25M R
Forrest Gump (1994) 8.8 142 min $342.55M R
The Lord of the Rings: The Two (2002) 8.8 179 min $46.84M R

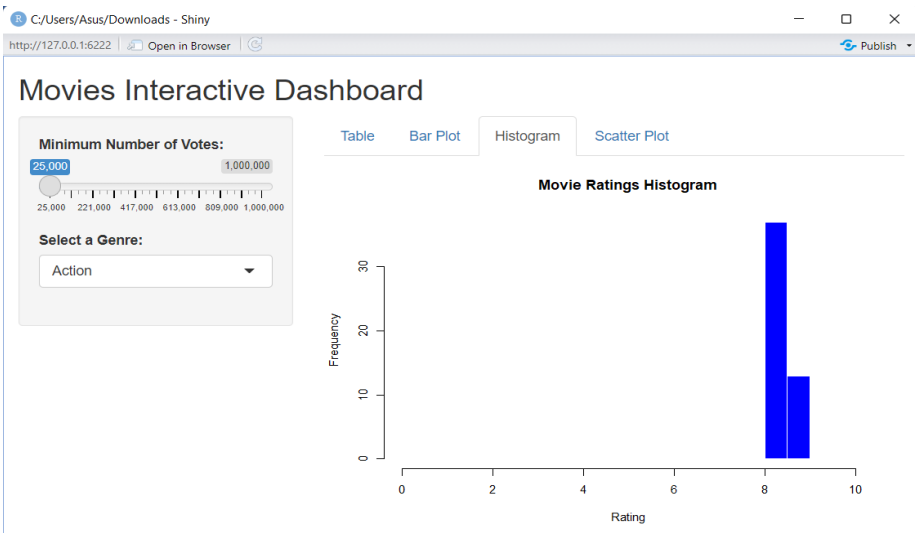


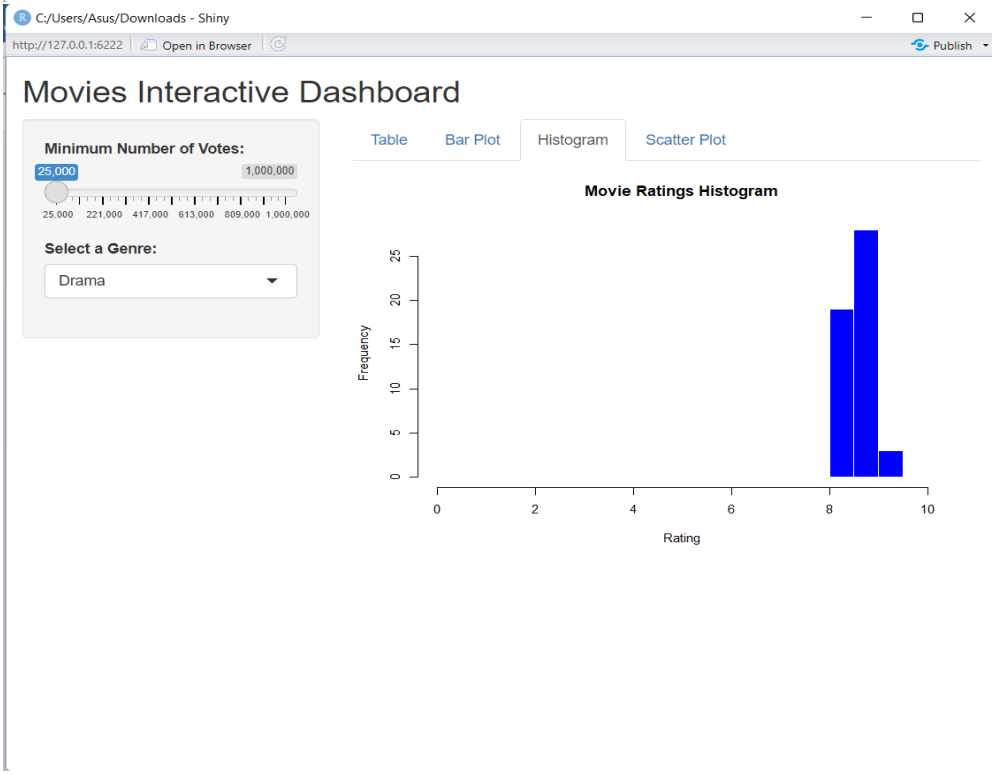


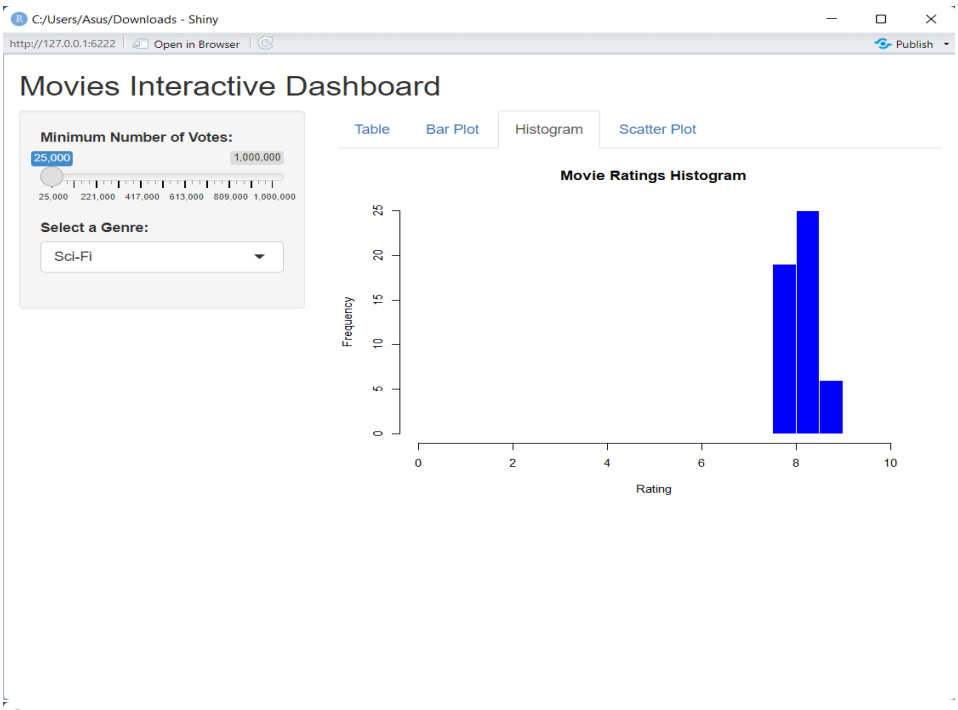
name	year	rating	RunTime	US_Box_Office	PgRating
Inception	(2010)	8.8	148 min	$292.58M	PG-13
The Matrix	(1999)	8.7	136 min	$171.48M	R
The Empire Strikes Back	(1980)	8.7	124 min	$290.48M	PG
Interstellar	(2014)	8.6	169 min	$188.02M	PG-13
Star Wars	(1977)	8.6	121 min	$322.74M	PG
Terminator 2: Judgment Day	(1991)	8.6	137 min	$204.84M	R
Alien	(1979)	8.5	117 min	$78.90M	R
Back to the Future	(1985)	8.5	116 min	$210.61M	PG-13
The Prestige	(2006)	8.5	130 min	$53.09M	PG-13
Avengers: Endgame	(2019)	8.4	181 min	$858.37M	PG
Spider-Man: Into the Spider-Verse	(2018)	8.4	117 min	$190.24M	R
Aliens	(1986)	8.4	137 min	$85.16M	PG-13
Avengers: Infinity War	(2018)	8.4	149 min	$678.82M	R
WALL·E	(2008)	8.4	98 min	$223.81M	PG
A Clockwork	(1971)	8.3	136 min	$6.21M	PG-13

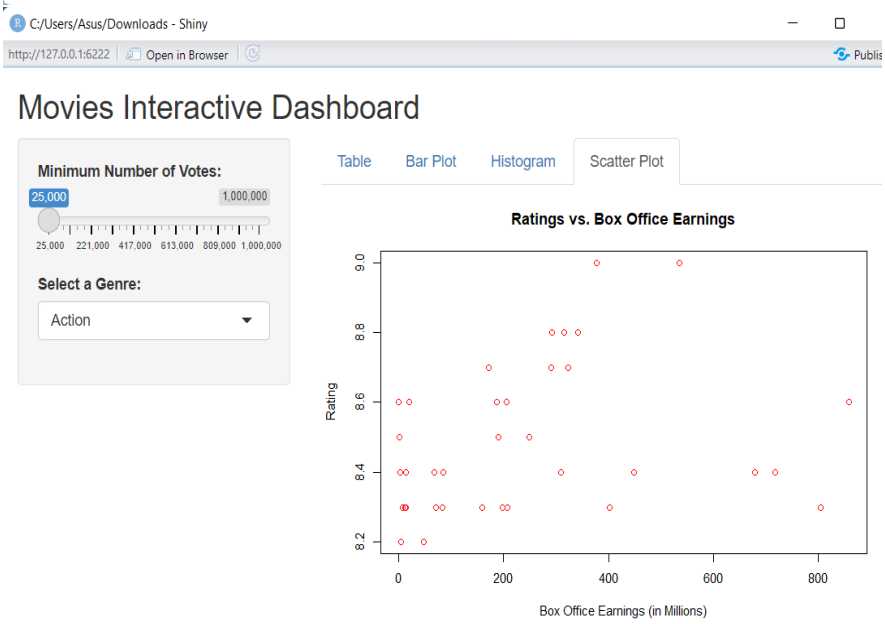

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Sci-Fi
Table
Bar Plot
Histogram
Scatter Plot
Box Office Earnings
Earnings (in Millions)
0
200
400
600
800
Movies

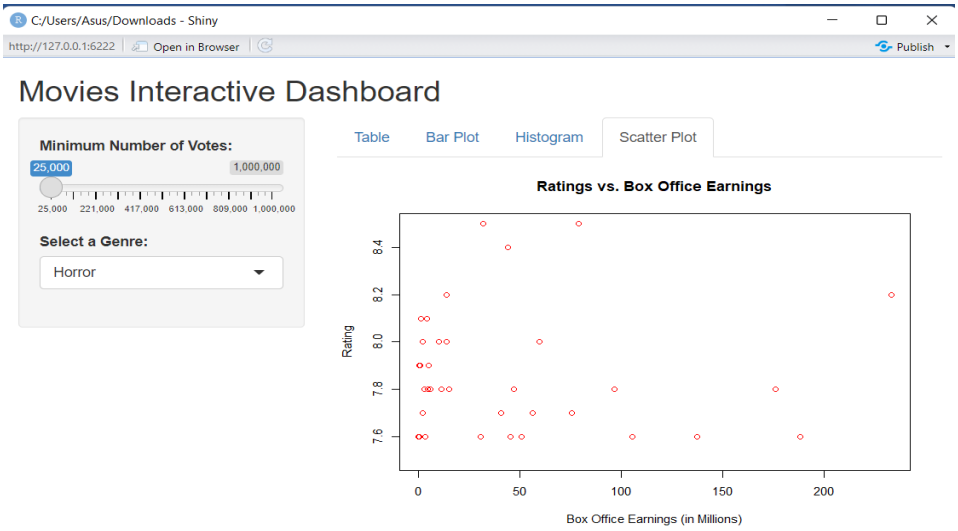

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Drama
Table
Bar Plot
Histogram
Scatter Plot
Box Office Earnings
Earnings (in Millions)
0
200
400
600
800
Movies

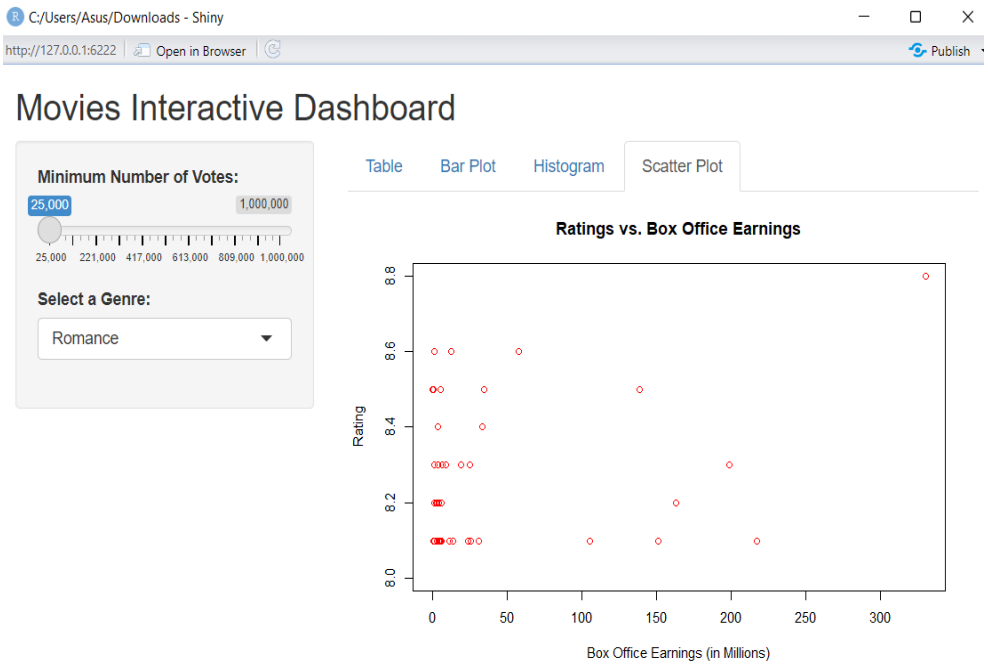

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Action
Table
Bar Plot
Histogram
Scatter Plot
Movie Ratings Histogram
Frequency
0
10
20
30
0
2
4
6
8
10
Rating

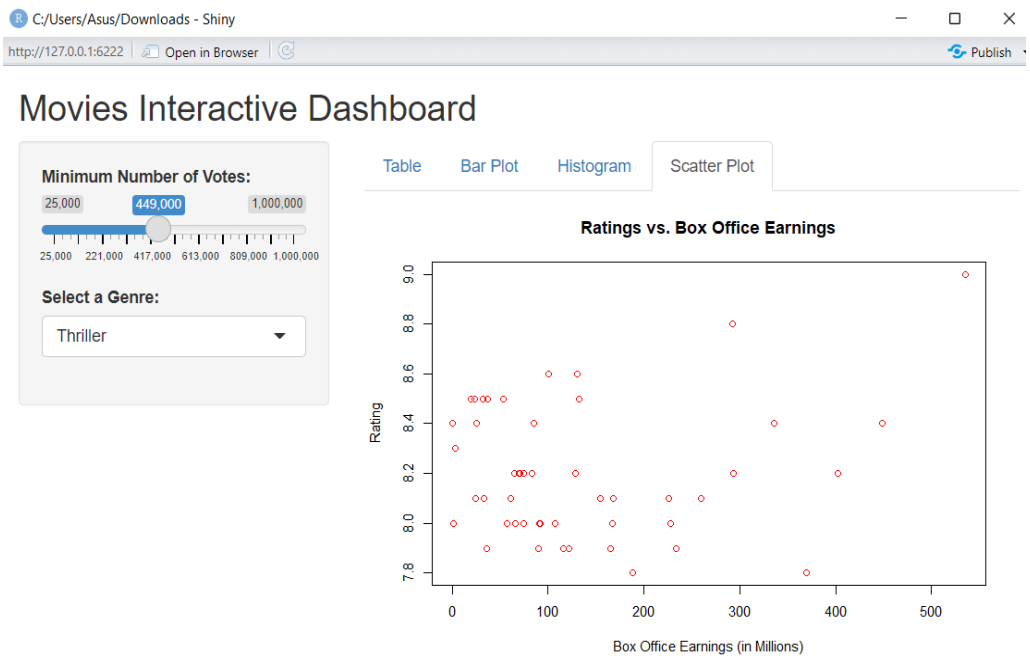

C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Drama
Table
Bar Plot
Histogram
Scatter Plot
Movie Ratings Histogram
Frequency
0
5
10
15
20
25
0
2
4
6
8
10
Rating


C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Sci-Fi
Table
Bar Plot
Histogram
Scatter Plot
Movie Ratings Histogram
Frequency
0 5 10 15 20 25
0 2 4 6 8 10
Rating


C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Action
Table
Bar Plot
Histogram
Scatter Plot
Ratings vs. Box Office Earnings
Rating
8.2 8.4 8.6 8.8 9.0
0 200 400 600 800
Box Office Earnings (in Millions)


C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Horror
Table
Bar Plot
Histogram
Scatter Plot
Ratings vs. Box Office Earnings
Rating
Box Office Earnings (in Millions)


C:/Users/Asus/Downloads - Shiny
http://127.0.0.1:6222
Open in Browser
Publish
Movies Interactive Dashboard
Minimum Number of Votes:
25,000
1,000,000
25,000 221,000 417,000 613,000 809,000 1,000,000
Select a Genre:
Romance
Table
Bar Plot
Histogram
Scatter Plot
Ratings vs. Box Office Earnings
Rating
Box Office Earnings (in Millions)



Discussion:

This project discusses the process of web scraping, which involves extracting data from web pages and exporting it in a more valuable format. The project also covers data pre-processing techniques, which are used to handle inconsistent, noisy, and incomplete data. In the given dataset, the PgRating and Box office columns have missing data, which must be resolved before incorporating the dataset into a model. Since the missing data is categorical, it cannot be replaced, so it must be discarded. The project also includes data visualization techniques, such as scatter plots, to offer patterns in the data using visual cues. Overall, the project provides insights into the process of web scraping and the importance of data pre-processing and visualization in making sense of the extracted data.