R and Power BI Project

Hollywood's Most Profitable Stories dataset

Alena Pavlioglo

Just IT Data Analyst Bootcamp

23/03/2023

The goal of this lab is to learn the statistical concepts and to analyse "Hollywood's Most Profitable Stories". dataset on Power BI.

CONTENT

Data Preparation

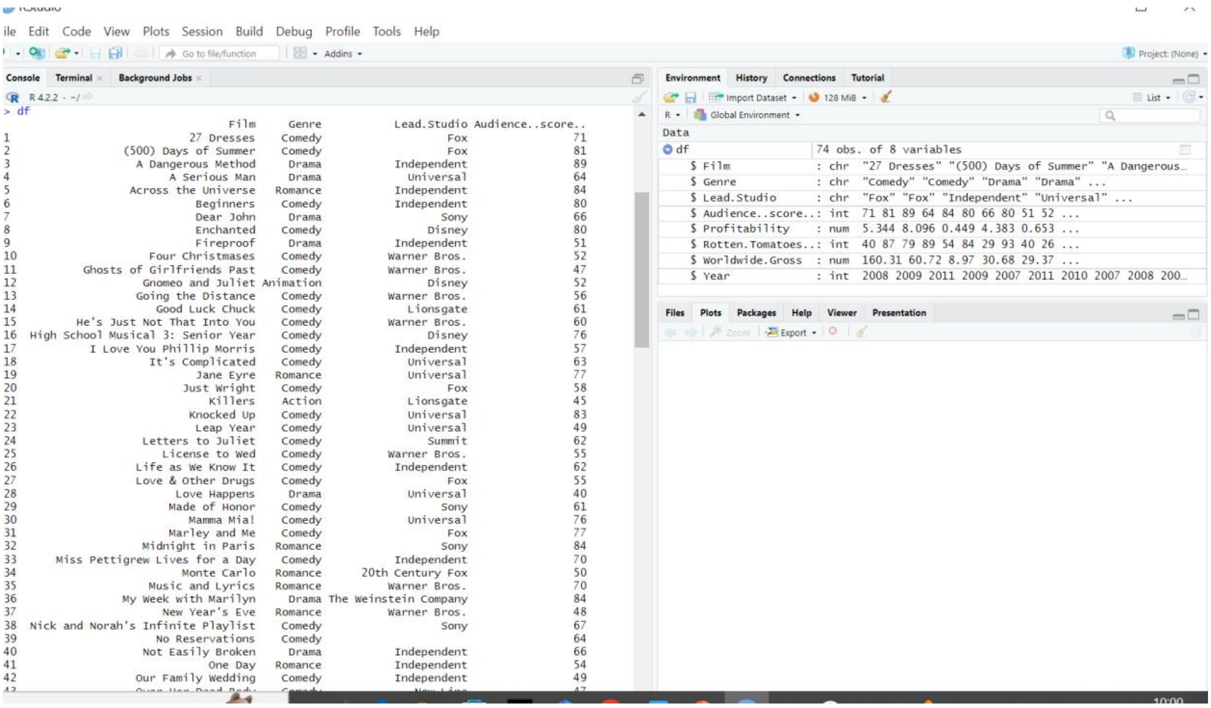
Data Cleaning

Exploratory Data Analysis

Creating Power BI Dashboard

Data Preparation

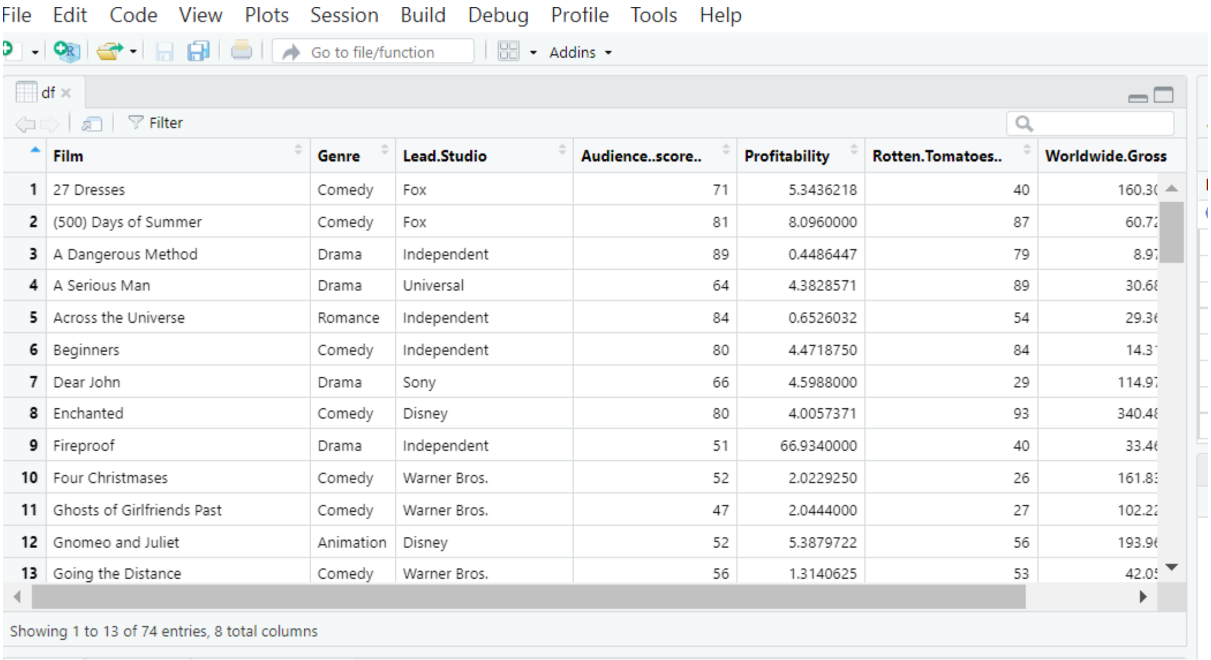
Load data and view the data.



```
> df
                                Film     Genre           Lead.Studio Audience..score..
1                        27 Dresses    Comedy                   Fox                71
2                (500) Days of Summer    Comedy                   Fox                81
3                 A Dangerous Method     Drama           Independent                89
4                       A Serious Man     Drama             Universal                64
5                 Across the Universe   Romance           Independent                84
6                           Beginners    Comedy           Independent                80
7                           Dear John     Drama                  Sony                66
8                           Enchanted    Comedy                Disney                80
9                           Fireproof     Drama           Independent                51
10                   Four Christmases    Comedy          Warner Bros.                52
11           Ghosts of Girlfriends Past    Comedy          Warner Bros.                47
12                 Gnomeo and Juliet Animation                Disney                52
13                 Going the Distance    Comedy          Warner Bros.                56
14                    Good Luck Chuck    Comedy             Lionsgate                61
15          He's Just Not That Into You    Comedy          Warner Bros.                60
16   High School Musical 3: Senior Year    Comedy                Disney                76
17           I Love You Phillip Morris    Comedy           Independent                57
18                   It's Complicated    Comedy             Universal                63
19                          Jane Eyre   Romance             Universal                77
20                        Just Wright    Comedy                   Fox                58
21                            Killers    Action             Lionsgate                45
22                         Knocked Up    Comedy             Universal                83
23                          Leap Year    Comedy             Universal                49
24                  Letters to Juliet    Comedy                Summit                62
25                    License to Wed    Comedy          Warner Bros.                55
26                  Life as We Know It    Comedy           Independent                62
27                  Love & Other Drugs    Comedy                   Fox                55
28                       Love Happens     Drama             Universal                40
29                      Made of Honor    Comedy                  Sony                61
30                         Mamma Mia!    Comedy             Universal                76
31                     Marley and Me    Comedy                   Fox                77
32                  Midnight in Paris   Romance                  Sony                84
33         Miss Pettigrew Lives for a Day    Comedy           Independent                70
34                        Monte Carlo   Romance      20th Century Fox                50
35                    Music and Lyrics   Romance          Warner Bros.                70
36                 My Week with Marilyn     Drama The Weinstein Company                84
37                       New Year's Eve   Romance          Warner Bros.                48
38  Nick and Norah's Infinite Playlist    Comedy                  Sony                67
39                    No Reservations    Comedy                                      64
40                  Not Easily Broken     Drama           Independent                66
41                            One Day   Romance           Independent                54
42                  Our Family Wedding    Comedy           Independent                49
```

```
Data
df                     74 obs. of 8 variables
   $ Film             : chr  "27 Dresses" "(500) Days of Summer" "A Dangerous…
   $ Genre            : chr  "Comedy" "Comedy" "Drama" "Drama" ...
   $ Lead.Studio      : chr  "Fox" "Fox" "Independent" "Universal" ...
   $ Audience..score..: int  71 81 89 64 84 80 66 80 51 52 ...
   $ Profitability    : num  5.344 8.096 0.449 4.383 0.653 ...
   $ Rotten.Tomatoes..: int  40 87 79 89 54 84 29 93 40 26 ...
   $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
   $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 200…
```

To view the data

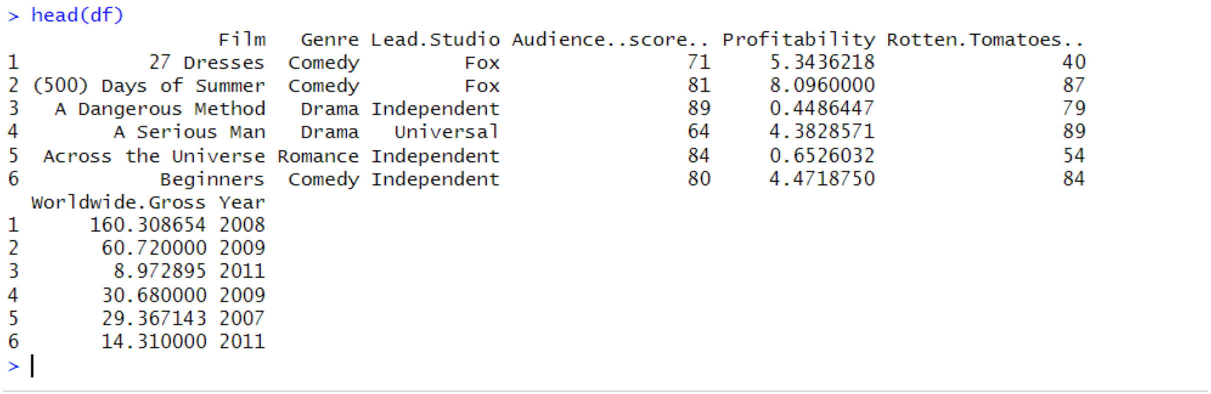

File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
df
Filter

	Film	Genre	Lead.Studio	Audience..score..	Profitability	Rotten.Tomatoes..	Worldwide.Gross
1	27 Dresses	Comedy	Fox	71	5.3436218	40	160.3
2	(500) Days of Summer	Comedy	Fox	81	8.0960000	87	60.7
3	A Dangerous Method	Drama	Independent	89	0.4486447	79	8.9
4	A Serious Man	Drama	Universal	64	4.3828571	89	30.6
5	Across the Universe	Romance	Independent	84	0.6526032	54	29.3
6	Beginners	Comedy	Independent	80	4.4718750	84	14.3
7	Dear John	Drama	Sony	66	4.5988000	29	114.9
8	Enchanted	Comedy	Disney	80	4.0057371	93	340.4
9	Fireproof	Drama	Independent	51	66.9340000	40	33.4
10	Four Christmases	Comedy	Warner Bros.	52	2.0229250	26	161.8
11	Ghosts of Girlfriends Past	Comedy	Warner Bros.	47	2.0444000	27	102.2
12	Gnomeo and Juliet	Animation	Disney	52	5.3879722	56	193.9
13	Going the Distance	Comedy	Warner Bros.	56	1.3140625	53	42.0

Showing 1 to 13 of 74 entries, 8 total columns

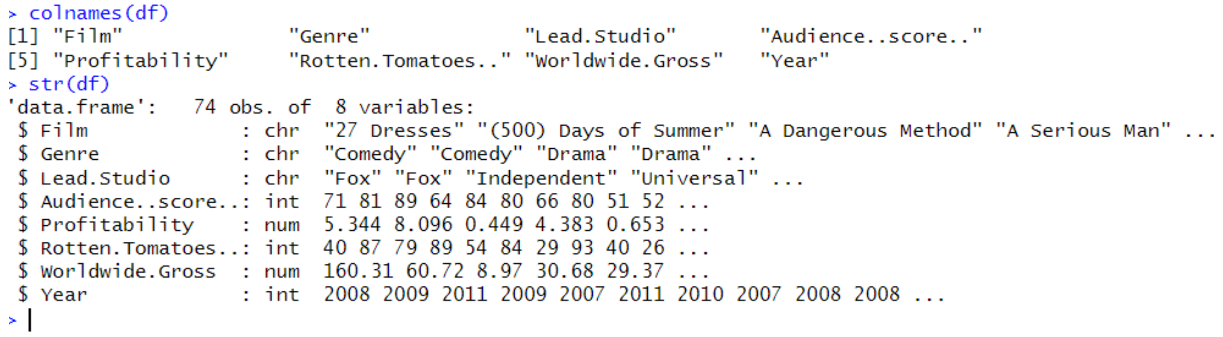
The head() function in R is used to display the first *n* rows present in the input data frame.

Head(df)



```
> head(df)
                    Film   Genre Lead.Studio Audience..score.. Profitability Rotten.Tomatoes..
1             27 Dresses  Comedy         Fox                71     5.3436218                40
2 (500) Days of Summer  Comedy         Fox                81     8.0960000                87
3   A Dangerous Method   Drama Independent                89     0.4486447                79
4        A Serious Man   Drama   Universal                64     4.3828571                89
5  Across the Universe Romance Independent                84     0.6526032                54
6            Beginners  Comedy Independent                80     4.4718750                84
  Worldwide.Gross Year
1      160.308654 2008
2       60.720000 2009
3        8.972895 2011
4       30.680000 2009
5       29.367143 2007
6       14.310000 2011
> |
```

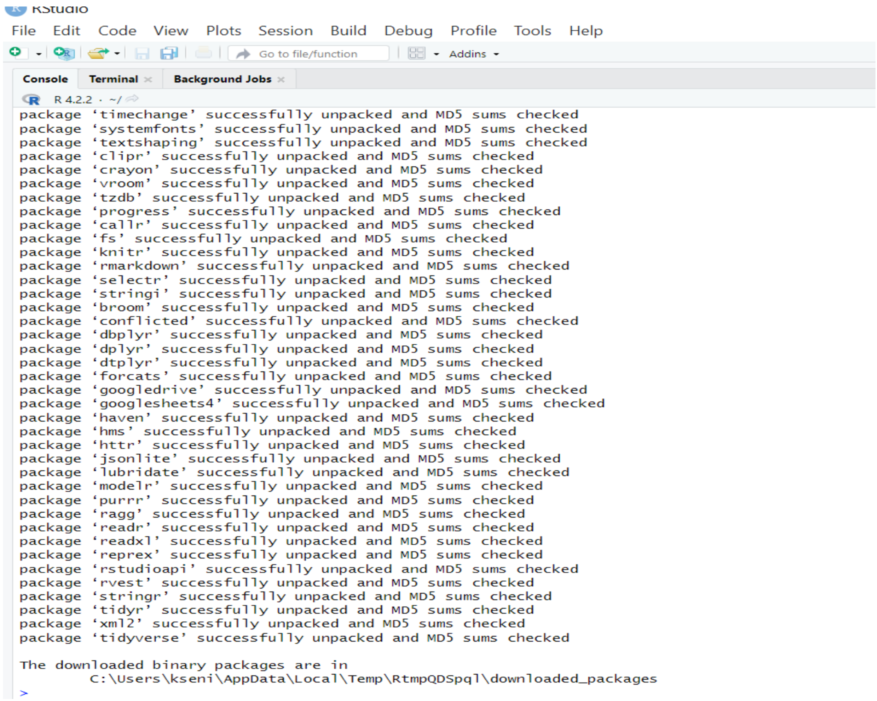
By checking the column names and structure I found out that there are 8 variables and 74 objects.



```
> colnames(df)
[1] "Film"              "Genre"             "Lead.Studio"       "Audience..score.."
[5] "Profitability"     "Rotten.Tomatoes.." "Worldwide.Gross"   "Year"
> str(df)
'data.frame':	74 obs. of  8 variables:
 $ Film             : chr  "27 Dresses" "(500) Days of Summer" "A Dangerous Method" "A Serious Man" ...
 $ Genre            : chr  "Comedy" "Comedy" "Drama" "Drama" ...
 $ Lead.Studio      : chr  "Fox" "Fox" "Independent" "Universal" ...
 $ Audience..score..: int  71 81 89 64 84 80 66 80 51 52 ...
 $ Profitability    : num  5.344 8.096 0.449 4.383 0.653 ...
 $ Rotten.Tomatoes..: int  40 87 79 89 54 84 29 93 40 26 ...
 $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
 $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 2008 ...
> |
```

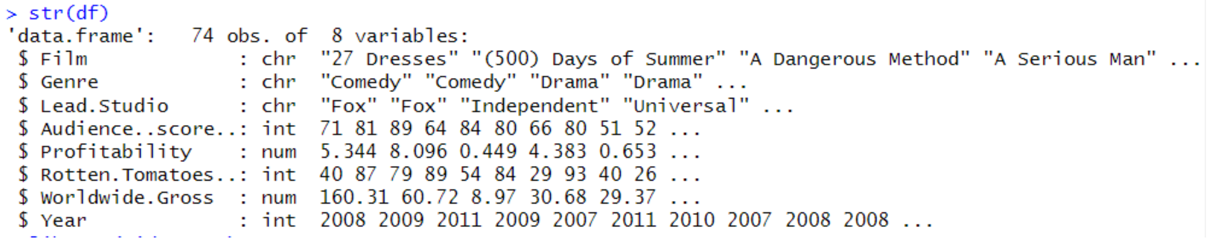
Load packages

To be able to work and use different functions to make statistical and graphical analysis on datasets we have to load some libraries.



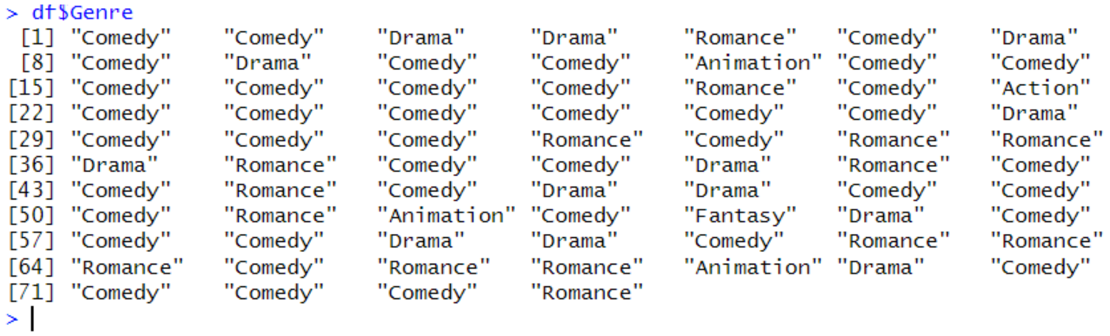
```
package ‘timechange’ successfully unpacked and MD5 sums checked
package ‘systemfonts’ successfully unpacked and MD5 sums checked
package ‘textshaping’ successfully unpacked and MD5 sums checked
package ‘clipr’ successfully unpacked and MD5 sums checked
package ‘crayon’ successfully unpacked and MD5 sums checked
package ‘vroom’ successfully unpacked and MD5 sums checked
package ‘tzdb’ successfully unpacked and MD5 sums checked
package ‘progress’ successfully unpacked and MD5 sums checked
package ‘callr’ successfully unpacked and MD5 sums checked
package ‘fs’ successfully unpacked and MD5 sums checked
package ‘knitr’ successfully unpacked and MD5 sums checked
package ‘rmarkdown’ successfully unpacked and MD5 sums checked
package ‘selectr’ successfully unpacked and MD5 sums checked
package ‘stringi’ successfully unpacked and MD5 sums checked
package ‘broom’ successfully unpacked and MD5 sums checked
package ‘conflicted’ successfully unpacked and MD5 sums checked
package ‘dbplyr’ successfully unpacked and MD5 sums checked
package ‘dplyr’ successfully unpacked and MD5 sums checked
package ‘dtplyr’ successfully unpacked and MD5 sums checked
package ‘forcats’ successfully unpacked and MD5 sums checked
package ‘googledrive’ successfully unpacked and MD5 sums checked
package ‘googlesheets4’ successfully unpacked and MD5 sums checked
package ‘haven’ successfully unpacked and MD5 sums checked
package ‘hms’ successfully unpacked and MD5 sums checked
package ‘httr’ successfully unpacked and MD5 sums checked
package ‘jsonlite’ successfully unpacked and MD5 sums checked
package ‘lubridate’ successfully unpacked and MD5 sums checked
package ‘modelr’ successfully unpacked and MD5 sums checked
package ‘purrr’ successfully unpacked and MD5 sums checked
package ‘ragg’ successfully unpacked and MD5 sums checked
package ‘readr’ successfully unpacked and MD5 sums checked
package ‘readxl’ successfully unpacked and MD5 sums checked
package ‘reprex’ successfully unpacked and MD5 sums checked
package ‘rstudioapi’ successfully unpacked and MD5 sums checked
package ‘rvest’ successfully unpacked and MD5 sums checked
package ‘stringr’ successfully unpacked and MD5 sums checked
package ‘tidyr’ successfully unpacked and MD5 sums checked
package ‘xml2’ successfully unpacked and MD5 sums checked
package ‘tidyverse’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in
	C:\Users\kseni\AppData\Local\Temp\RtmpQDSpql\downloaded_packages
>
```

Check data types:



```
> str(df)
'data.frame':	74 obs. of  8 variables:
 $ Film             : chr  "27 Dresses" "(500) Days of Summer" "A Dangerous Method" "A Serious Man" ...
 $ Genre            : chr  "Comedy" "Comedy" "Drama" "Drama" ...
 $ Lead.Studio      : chr  "Fox" "Fox" "Independent" "Universal" ...
 $ Audience..score..: int  71 81 89 64 84 80 66 80 51 52 ...
 $ Profitability    : num  5.344 8.096 0.449 4.383 0.653 ...
 $ Rotten.Tomatoes..: int  40 87 79 89 54 84 29 93 40 26 ...
 $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
 $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 2008 ...
```

To access the data in a single column to explore ethe data (for example column Genre)

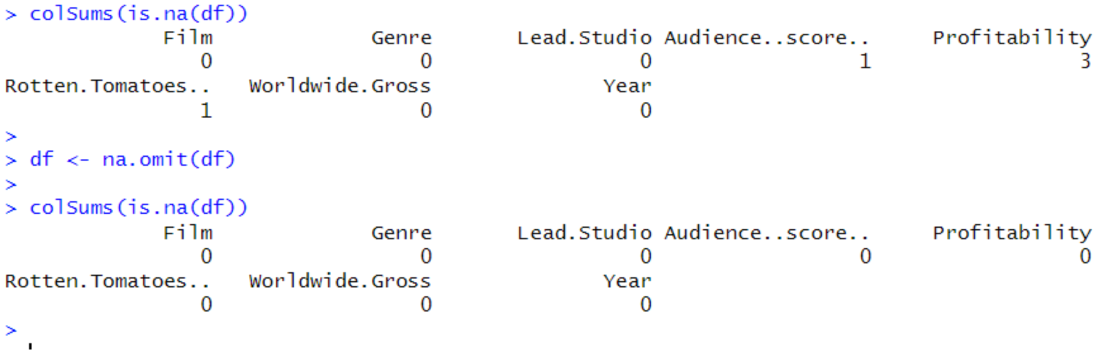


```
> df$Genre
 [1] "Comedy"    "Comedy"    "Drama"     "Drama"     "Romance"   "Comedy"    "Drama"    
 [8] "Comedy"    "Drama"     "Comedy"    "Comedy"    "Animation" "Comedy"    "Comedy"   
[15] "Comedy"    "Comedy"    "Comedy"    "Comedy"    "Romance"   "Comedy"    "Action"   
[22] "Comedy"    "Comedy"    "Comedy"    "Comedy"    "Comedy"    "Comedy"    "Drama"    
[29] "Comedy"    "Comedy"    "Comedy"    "Romance"   "Comedy"    "Romance"   "Romance"  
[36] "Drama"     "Romance"   "Comedy"    "Comedy"    "Drama"     "Romance"   "Comedy"   
[43] "Comedy"    "Romance"   "Comedy"    "Drama"     "Drama"     "Comedy"    "Comedy"   
[50] "Comedy"    "Romance"   "Animation" "Comedy"    "Fantasy"   "Drama"     "Comedy"   
[57] "Comedy"    "Comedy"    "Drama"     "Drama"     "Comedy"    "Romance"   "Romance"  
[64] "Romance"   "Comedy"    "Romance"   "Romance"   "Animation" "Drama"     "Comedy"   
[71] "Comedy"    "Comedy"    "Comedy"    "Romance"  
> |
```

1. Data Cleaning

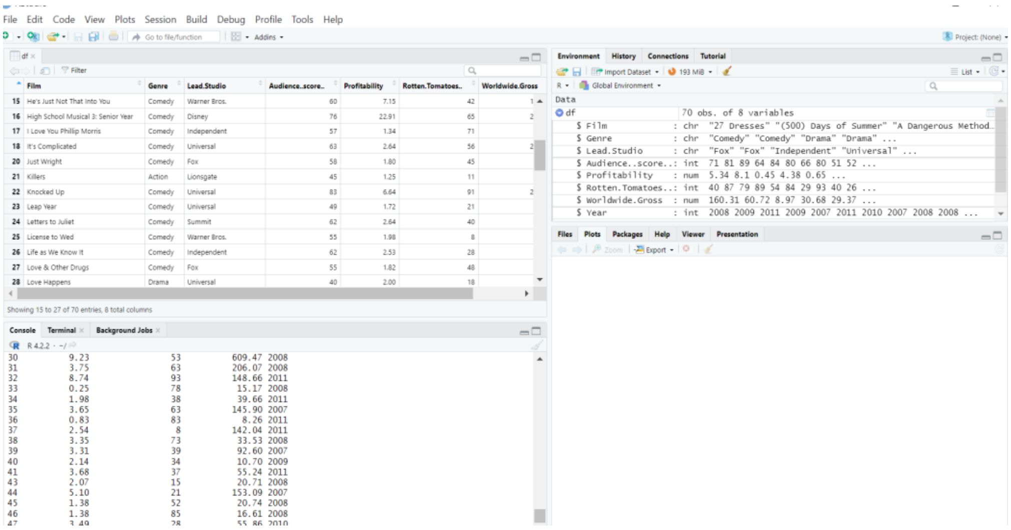
The purpose of data cleaning is to identify, correct, or remove inaccurate raw data for downstream purposes.

Checking missing values. To find the length of columns for missing values we use colSum(is.na(df)) and to remove all rows that contains at least on NA we use command df <- na.omit(df) and check if the rows have been removed. It is very important to handle missing values since it can bias the results and reduce the accuracy.



```
> colSums(is.na(df))
             Film             Genre       Lead.Studio Audience..score..     Profitability 
                0                 0                 0                 1                 3 
Rotten.Tomatoes..   Worldwide.Gross              Year 
                1                 0                 0 
> 
> df <- na.omit(df)
> 
> colSums(is.na(df))
             Film             Genre       Lead.Studio Audience..score..     Profitability 
                0                 0                 0                 0                 0 
Rotten.Tomatoes..   Worldwide.Gross              Year 
                0                 0                 0 
> 
```

Removing duplicates and rounding the values to 2 places. Checking the dimension of the new data frame. Removing duplicates from data set is also very important in order to maintain accuracy and avoid misleading statistics. Then we specify the number of decimal places (2) to which we need to round the "Profitability" and "Worldwide.Gross" Columns.


File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
Project: (None)
df
Filter
Environment History Connections Tutorial
Import Dataset
193 MiB
List
R
Global Environment
Files Plots Packages Help Viewer Presentation
Zoom
Export
Console Terminal Background Jobs
R 4.2.2 · ~/

	Film	Genre	Lead.Studio	Audience..score..	Profitability	Rotten.Tomatoes..	Worldwide.Gross
15	He's Just Not That Into You	Comedy	Warner Bros.	60	7.15	42	1
16	High School Musical 3: Senior Year	Comedy	Disney	76	22.91	65	2
17	I Love You Phillip Morris	Comedy	Independent	57	1.34	71	
18	It's Complicated	Comedy	Universal	63	2.64	56	2
20	Just Wright	Comedy	Fox	58	1.80	45	
21	Killers	Action	Lionsgate	45	1.25	11	
22	Knocked Up	Comedy	Universal	83	6.64	91	2
23	Leap Year	Comedy	Universal	49	1.72	21	
24	Letters to Juliet	Comedy	Summit	62	2.64	40	
25	License to Wed	Comedy	Warner Bros.	55	1.98	8	
26	Life as We Know It	Comedy	Independent	62	2.53	28	
27	Love & Other Drugs	Comedy	Fox	55	1.82	48	
28	Love Happens	Drama	Universal	40	2.00	18	

Showing 15 to 27 of 70 entries, 8 total columns

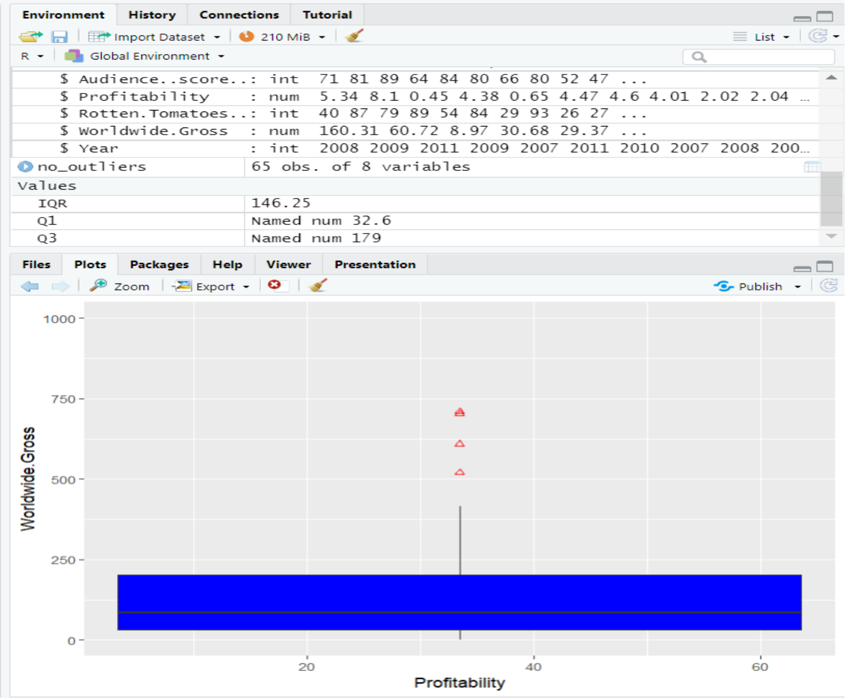
```
30          9.23          53        609.47  2008
31          3.75          63        206.07  2008
32          8.74          93        148.66  2011
33          0.25          78         15.17  2008
34          1.98          38         39.66  2011
35          3.65          63        145.90  2007
36          0.83          83          8.26  2011
37          2.54           8        142.04  2011
38          3.35          73         33.53  2008
39          3.31          39         92.60  2007
40          2.14          34         10.70  2009
41          3.68          37         55.24  2011
43          2.07          15         20.71  2008
44          5.10          21        153.09  2007
45          1.38          52         20.74  2008
46          1.38          85         16.61  2008
47          3.49          28         55.86  2010
```

```
Data
df                      70 obs. of 8 variables
  $ Film             : chr  "27 Dresses" "(500) Days of Summer" "A Dangerous Method…
  $ Genre            : chr  "Comedy" "Comedy" "Drama" "Drama" ...
  $ Lead.Studio      : chr  "Fox" "Fox" "Independent" "Universal" ...
  $ Audience..score..: int  71 81 89 64 84 80 66 80 51 52 ...
  $ Profitability    : num  5.34 8.1 0.45 4.38 0.65 ...
  $ Rotten.Tomatoes..: int  40 87 79 89 54 84 29 93 40 26 ...
  $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
  $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 2008 ...
```

Step 2.1

Check for outliers using a boxplot.

Boxplot is the useful tool to detect potential outliers and helps to visualize a quantitative variable by displaying minimum, median, first and third quartiles and maximum and any observation that was classified as a suspected outlier using the IQR criterion. We create a boxplot and to adjust the y-axis we use coord_cartesian and set up y-axis range from 0 to 1000, just as we specified using ylim() argument. To label the scale of the x-axis we have applied scale_x_continuous to change x-axis when x-variable is continuous.

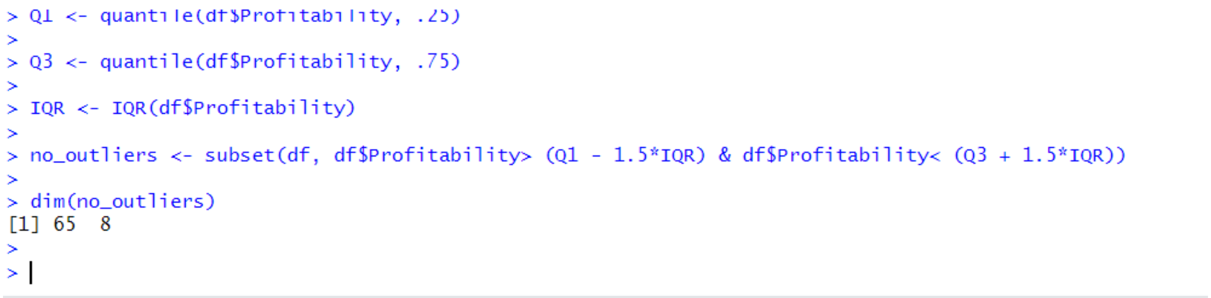


```
  $ Audience..score..: int  71 81 89 64 84 80 66 80 52 47 ...
  $ Profitability    : num  5.34 8.1 0.45 4.38 0.65 4.47 4.6 4.01 2.02 2.04 …
  $ Rotten.Tomatoes..: int  40 87 79 89 54 84 29 93 26 27 ...
  $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
  $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 200…
no_outliers            65 obs. of 8 variables
Values
  IQR                  146.25
  Q1                   Named num 32.6
  Q3                   Named num 179
```

Quartiles are three values that split dataset into quarters.
Q1 First quartile: 25% of all the values fall below that value.
Q2: Second quartile / Median: This value splits the data in half.
Q3 Third quartile: 25% of the data are above this value.

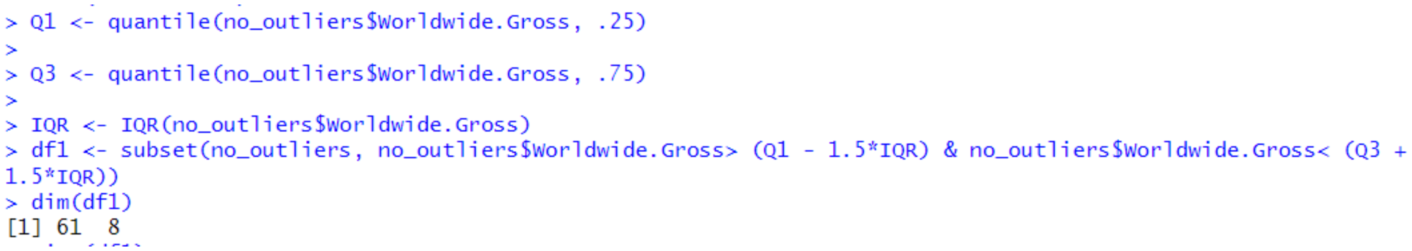
The interquartile range (IQR) is the range between the first and third quartiles. IQR = Q3 -Q1

Observations considered as potential outliers by the IQR criterion are displayed as points in the boxplot.
By removing outliers in” Profitability” and “Worldwide.Gross” and put a condition in which new subset will meet criteria that all observation above
Q3 +1.5*IQR and below Q1 -1.5*IQR are considered as potential outliers and will be removed.



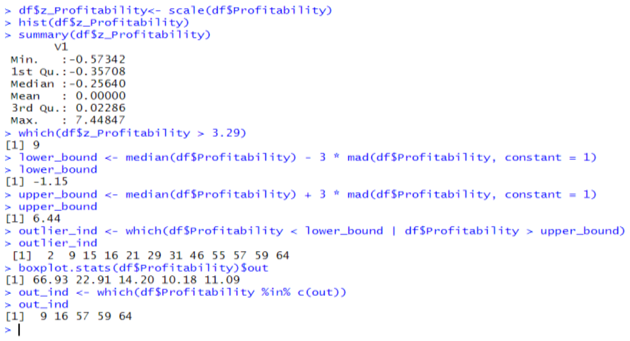
```
> Q1 <- quantile(df$Profitability, .25)
>
> Q3 <- quantile(df$Profitability, .75)
>
> IQR <- IQR(df$Profitability)
>
> no_outliers <- subset(df, df$Profitability> (Q1 - 1.5*IQR) & df$Profitability< (Q3 + 1.5*IQR))
>
> dim(no_outliers)
[1] 65  8
>
> |
```

The dimensions for new “no_outliers” are 65 objects and 8 attributes, but after removing the outliers for “Worldwide.Gross” we notice that we have a new df1 with 61 objects and 8 attributes.

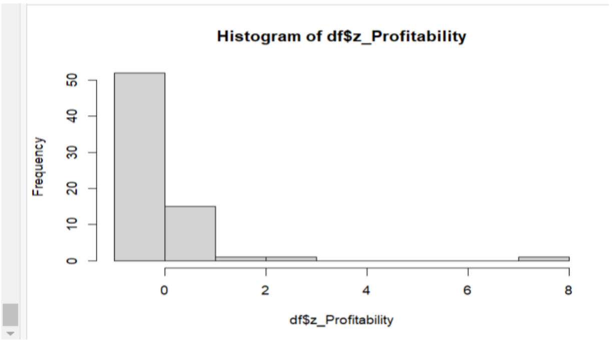


```
> Q1 <- quantile(no_outliers$Worldwide.Gross, .25)
>
> Q3 <- quantile(no_outliers$Worldwide.Gross, .75)
>
> IQR <- IQR(no_outliers$Worldwide.Gross)
> df1 <- subset(no_outliers, no_outliers$Worldwide.Gross> (Q1 - 1.5*IQR) & no_outliers$Worldwide.Gross< (Q3 +
1.5*IQR))
> dim(df1)
[1] 61  8
```

There are various methods for extracting the values of the potential outliers and I was wondering which of them are more accurate, how they work for R and whether I will come to the same results.
One possible way also based on the IQR criterion by using boxplot.stats()$out and function which()
to extract the row number corresponding to these outliers. We can also try z-score and Hampel filter by using median() and mad() functions.

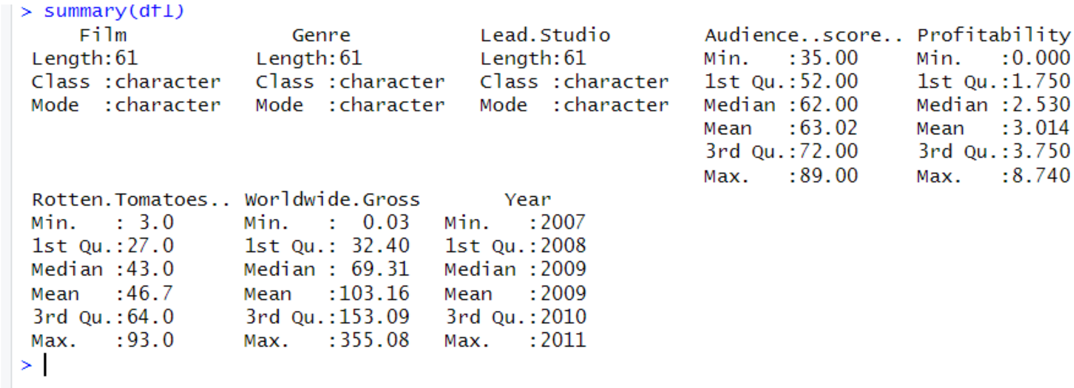


```
> df$z_Profitability<- scale(df$Profitability)
> hist(df$z_Profitability)
> summary(df$z_Profitability)
       V1
 Min.   :-0.57342
 1st Qu.:-0.35708
 Median :-0.25640
 Mean   : 0.00000
 3rd Qu.: 0.02286
 Max.   : 7.44847
> which(df$z_Profitability > 3.29)
[1] 9
> lower_bound <- median(df$Profitability) - 3 * mad(df$Profitability, constant = 1)
> lower_bound
[1] -1.15
> upper_bound <- median(df$Profitability) + 3 * mad(df$Profitability, constant = 1)
> upper_bound
[1] 6.44
> outlier_ind <- which(df$Profitability < lower_bound | df$Profitability > upper_bound)
> outlier_ind
 [1]  2  9 15 16 21 29 31 46 55 57 59 64
> boxplot.stats(df$Profitability)$out
[1] 66.93 22.91 14.20 10.18 11.09
> out_ind <- which(df$Profitability %in% c(out))
> out_ind
[1]  9 16 57 59 64
> |
```



Step 3: Exploratory Data Analysis

The summary is an exploratory data analysis tool that provides insight into the distribution of values for one Variable. This set of statistics describes where data values occur, their central tendency, variability, and the general shape of their distribution.

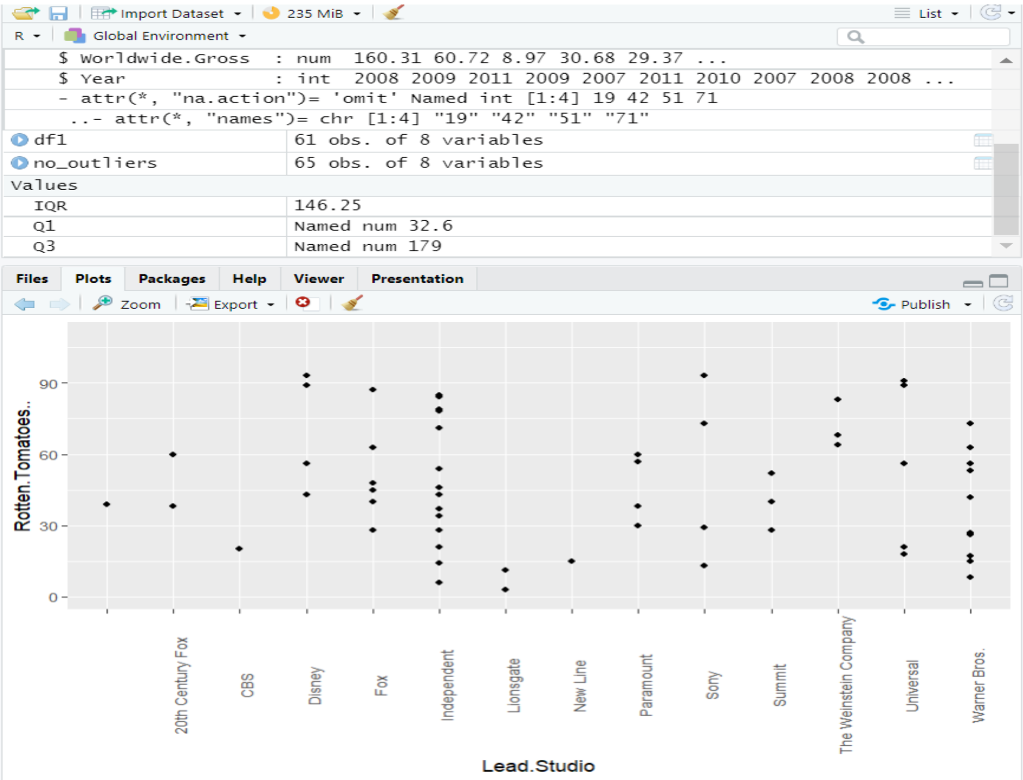


```
> summary(df1)
     Film              Genre             Lead.Studio        Audience..score.. Profitability  
 Length:61          Length:61          Length:61          Min.   :35.00     Min.   :0.000  
 Class :character   Class :character   Class :character   1st Qu.:52.00     1st Qu.:1.750  
 Mode  :character   Mode  :character   Mode  :character   Median :62.00     Median :2.530  
                                                          Mean   :63.02     Mean   :3.014  
                                                          3rd Qu.:72.00     3rd Qu.:3.750  
                                                          Max.   :89.00     Max.   :8.740  
 Rotten.Tomatoes.. Worldwide.Gross        Year     
 Min.   : 3.0      Min.   :  0.03   Min.   :2007  
 1st Qu.:27.0      1st Qu.: 32.40   1st Qu.:2008  
 Median :43.0      Median : 69.31   Median :2009  
 Mean   :46.7      Mean   :103.16   Mean   :2009  
 3rd Qu.:64.0      3rd Qu.:153.09   3rd Qu.:2010  
 Max.   :93.0      Max.   :355.08   Max.   :2011  
> |
```

From the summary we can see that the Median for “Worldwide.Gross”(69.31) is close to Q1(32.40) thanQ3 (153.09). Therefore, the distribution of values is right- skewed.

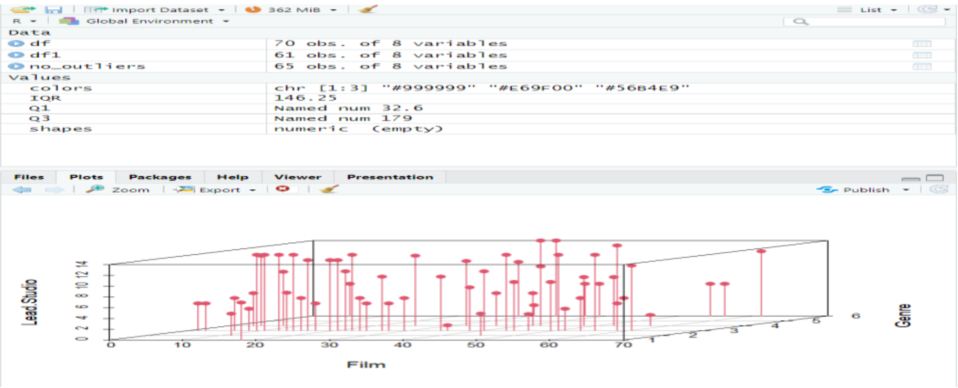
Bivariate analysis refers to the analysis of two variables to determine relationships between them.

Scatterplot



```
 $ Worldwide.Gross  : num  160.31 60.72 8.97 30.68 29.37 ...
 $ Year             : int  2008 2009 2011 2009 2007 2011 2010 2007 2008 2008 ...
 - attr(*, "na.action")= 'omit' Named int [1:4] 19 42 51 71
  ..- attr(*, "names")= chr [1:4] "19" "42" "51" "71"
df1             61 obs. of 8 variables
no_outliers     65 obs. of 8 variables
Values
  IQR           146.25
  Q1            Named num 32.6
  Q3            Named num 179
```

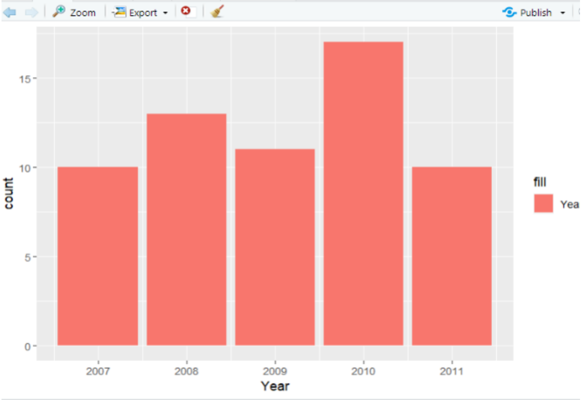
To explore more interesting options for displaying the scatterplots I uploaded a “skatterplot3d” package and build 3d scatterplot in Films, Genres and Lead.Studious.

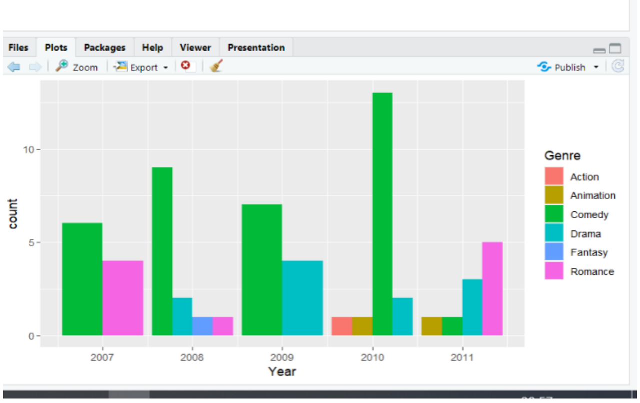


```
Data
df              70 obs. of 8 variables
df1             61 obs. of 8 variables
no_outliers     65 obs. of 8 variables
Values
  colors        chr [1:3] "#999999" "#E69F00" "#56B4E9"
  IQR           146.25
  Q1            Named num 32.6
  Q3            Named num 179
  shapes        numeric (empty)
```

Bar chart

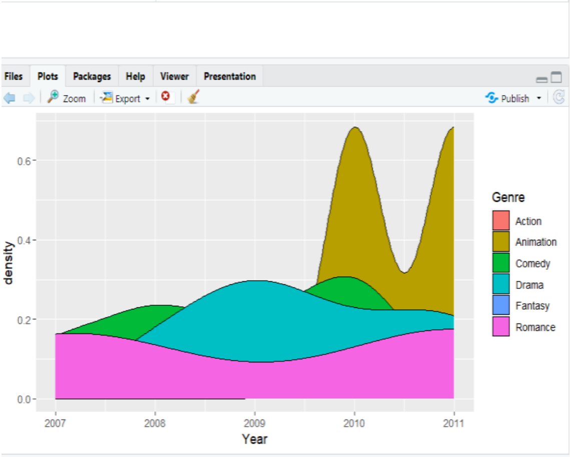
I have created a bar chat by Year and grouped it by Genre.

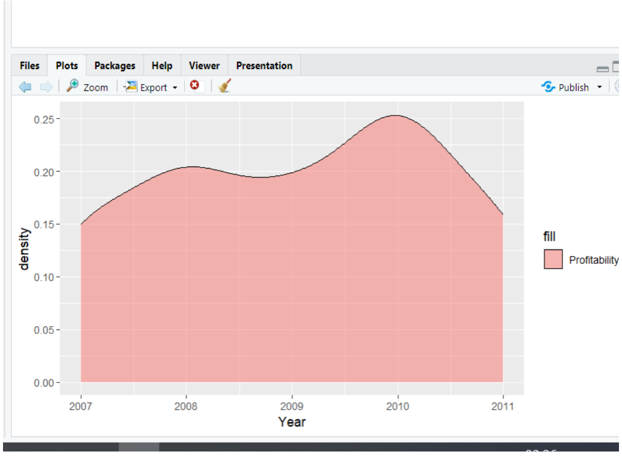




Density plot

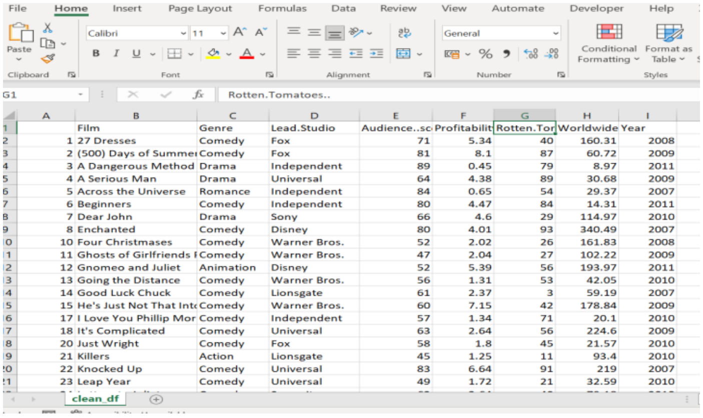
I was really interested to build a Density plot since it visualizes the distribution of data over a continuous interval or time period. The peaks of a Density Plot help display where values are concentrated over the interval. I have chosen 2 different parameters and visualized the density by Genre and Profitability aver the Year.





Step 4: Export data

Finally, our clean data is ready do be exported to Power BI to analyze the data and visualize results by creating different charts. We upload our already cleaned dataset as a CSV file to Power BI.

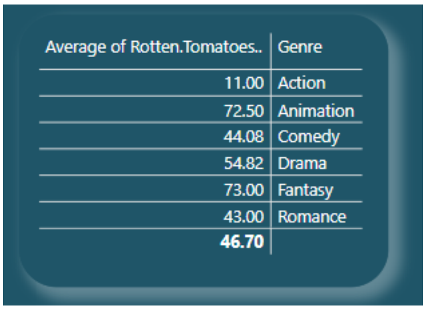


G1 Rotten.Tomatoes..

	Film	Genre	Lead.Studio	Audience..sc	Profitabilit	Rotten.Tor	Worldwide	Year
1	27 Dresses	Comedy	Fox	71	5.34	40	160.31	2008
2	(500) Days of Summer	Comedy	Fox	81	8.1	87	60.72	2009
3	A Dangerous Method	Drama	Independent	89	0.45	79	8.97	2011
4	A Serious Man	Drama	Universal	64	4.38	89	30.68	2009
5	Across the Universe	Romance	Independent	84	0.65	54	29.37	2007
6	Beginners	Comedy	Independent	80	4.47	84	14.31	2011
7	Dear John	Drama	Sony	66	4.6	29	114.97	2010
8	Enchanted	Comedy	Disney	80	4.01	93	340.49	2007
10	Four Christmases	Comedy	Warner Bros.	52	2.02	26	161.83	2008
11	Ghosts of Girlfriends I	Comedy	Warner Bros.	47	2.04	27	102.22	2009
12	Gnomeo and Juliet	Animation	Disney	52	5.39	56	193.97	2011
13	Going the Distance	Comedy	Warner Bros.	56	1.31	53	42.05	2010
14	Good Luck Chuck	Comedy	Lionsgate	61	2.37	3	59.19	2007
15	He's Just Not That Into	Comedy	Warner Bros.	60	7.15	42	178.84	2009
17	I Love You Phillip Mor	Comedy	Independent	57	1.34	71	20.1	2010
18	It's Complicated	Comedy	Universal	63	2.64	56	224.6	2009
20	Just Wright	Comedy	Fox	58	1.8	45	21.57	2010
21	Killers	Action	Lionsgate	45	1.25	11	93.4	2010
22	Knocked Up	Comedy	Universal	83	6.64	91	219	2007
23	Leap Year	Comedy	Universal	49	1.72	21	32.59	2010

Step 5: Create Power BI Dashboard

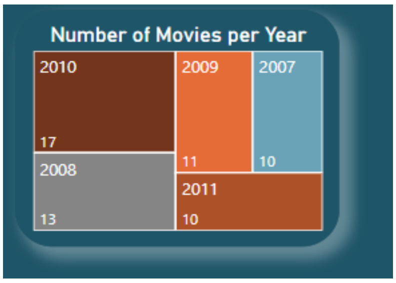
2. The average Rotten Tomatoes ratings of each genre



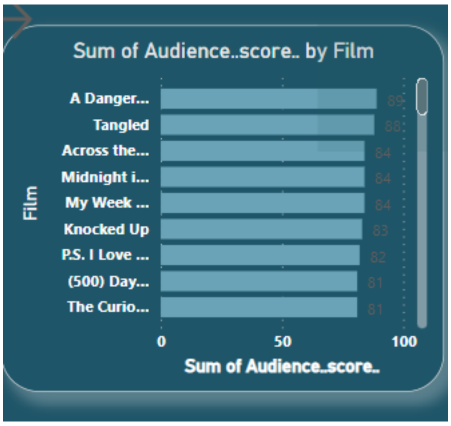
Average of Rotten.Tomatoes..	Genre
11.00	Action
72.50	Animation
44.08	Comedy
54.82	Drama
73.00	Fantasy
43.00	Romance
46.70	

3. The number of movies produced per year.

I created a treemap to visualize the number of movies produced per Year.



4. The audience scores for each film.

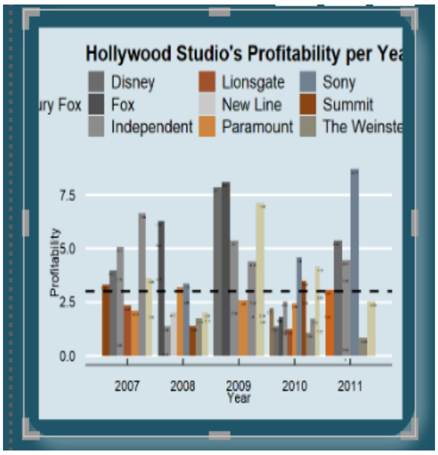


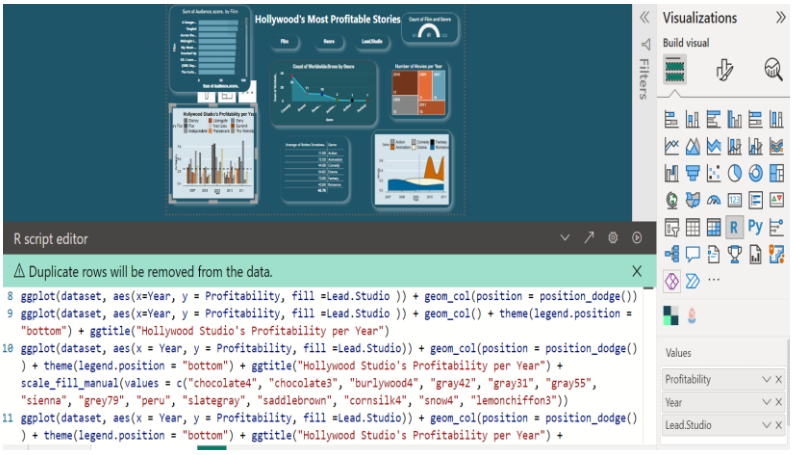
Clustered bar Chart seems to be suitable for analysing this relationship since we have a large amount of data that can be grouped.

5. The profitability per studio.

I have created stacked column chart displays the contribution of each Studio's profitability per Year by using geom_bar() and position = "dodge" to make it "side by side". I have use scale_fill_manual() to specify the colours and labelled the titles.

I wanted to add an extra touch to my bar charts, so I added a line representing an average of all the bars. In my example, this would give us an insight into which Studio over which Year performed better than average.





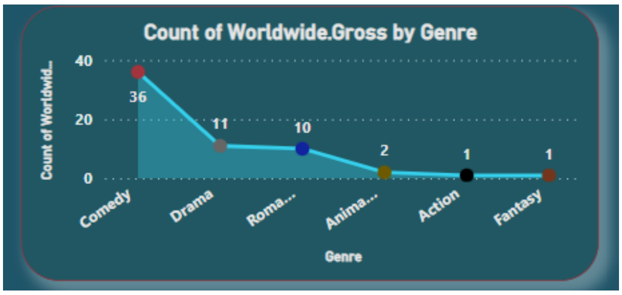
R script editor

⚠ Duplicate rows will be removed from the data.

```
ggplot(dataset, aes(x=Year, y = Profitability, fill =Lead.Studio )) + geom_col(position = position_dodge())
ggplot(dataset, aes(x=Year, y = Profitability, fill =Lead.Studio )) + geom_col() + theme(legend.position =
"bottom") + ggtitle("Hollywood Studio's Profitability per Year")
ggplot(dataset, aes(x = Year, y = Profitability, fill =Lead.Studio)) + geom_col(position = position_dodge()
) + theme(legend.position = "bottom") + ggtitle("Hollywood Studio's Profitability per Year") +
scale_fill_manual(values = c("chocolate4", "chocolate3", "burlywood4", "gray42", "gray31", "gray55",
"sienna", "grey79", "peru", "slategray", "saddlebrown", "cornsilk4", "snow4", "lemonchiffon3"))
ggplot(dataset, aes(x = Year, y = Profitability, fill =Lead.Studio)) + geom_col(position = position_dodge()
) + theme(legend.position = "bottom") + ggtitle("Hollywood Studio's Profitability per Year") +
```

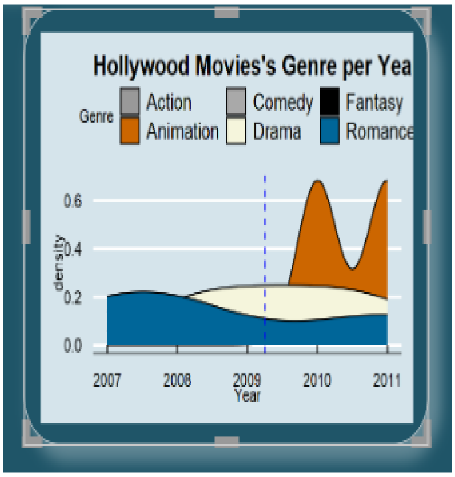
6. The worldwide gross per genre.

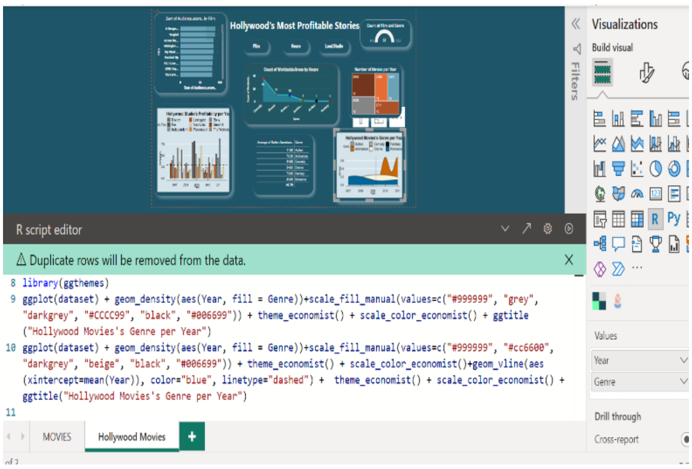
I used area chart to visualize and we can see that Comedy Genre has the highest Worldwide Gross and it is the leader, since the rest of the Genres have significant difference.



I decided to create a density plot for Genre per Year in R by using geom_density() function and applied theme_economist() + scale_color_economist() to change the background colour.

I also use the geom_vline() layer to add a vertical line for the mean of Year

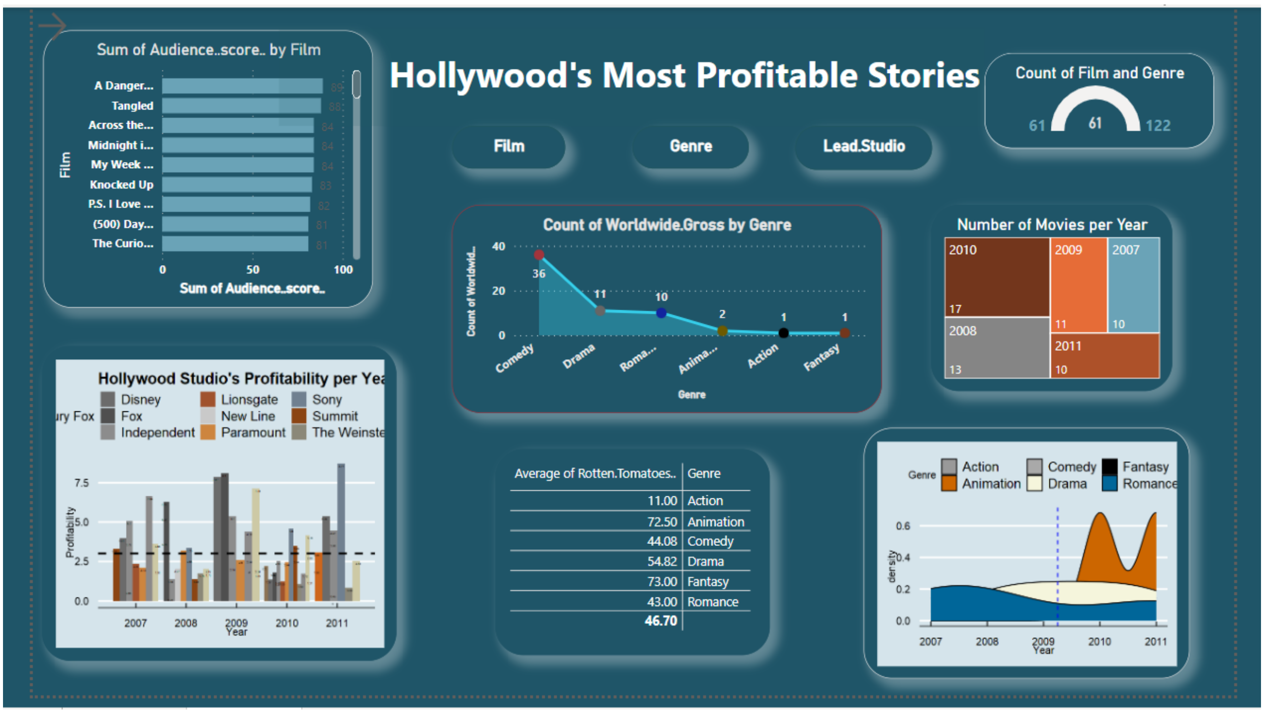




R script editor

⚠ Duplicate rows will be removed from the data.

```
library(ggthemes)
ggplot(dataset) + geom_density(aes(Year, fill = Genre))+scale_fill_manual(values=c("#999999", "grey",
"darkgrey", "#CCCC99", "black", "#006699")) + theme_economist() + scale_color_economist() + ggtitle
("Hollywood Movies's Genre per Year")
ggplot(dataset) + geom_density(aes(Year, fill = Genre))+scale_fill_manual(values=c("#999999", "#cc6600",
"darkgrey", "beige", "black", "#006699")) + theme_economist() + scale_color_economist()+geom_vline(aes
(xintercept=mean(Year)), color="blue", linetype="dashed") +  theme_economist() + scale_color_economist() +
ggtitle("Hollywood Movies's Genre per Year")

```



Hollywood's Most Profitable Stories

Average of Rotten.Tomatoes..	Genre
11.00	Action
72.50	Animation
44.08	Comedy
54.82	Drama
73.00	Fantasy
43.00	Romance
46.70	

https://app.powerbi.com/links/PtZ6SXLsJ3?ctid=6efd0f20-57c8-4447-b53f-00d4992ca50b&pbi_source=linkShare