R & Power BI Assignment

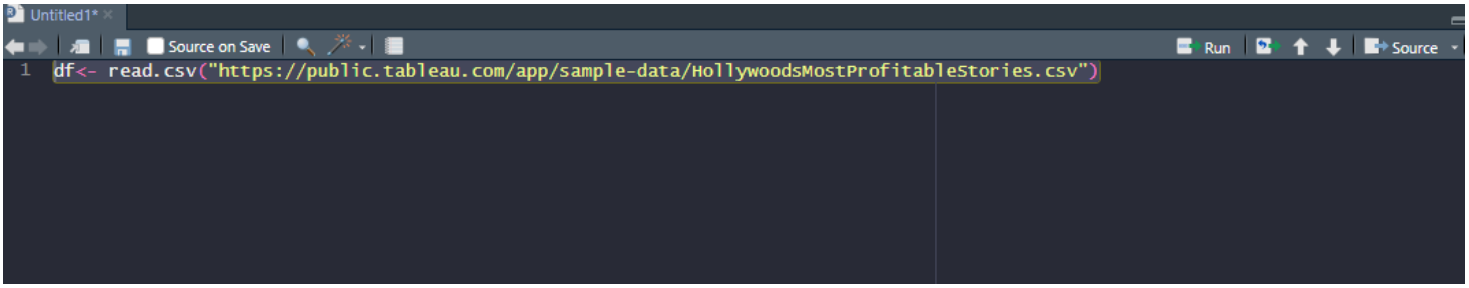
By: Emma Kaas Andersen

Submission Date: 30/05/2024

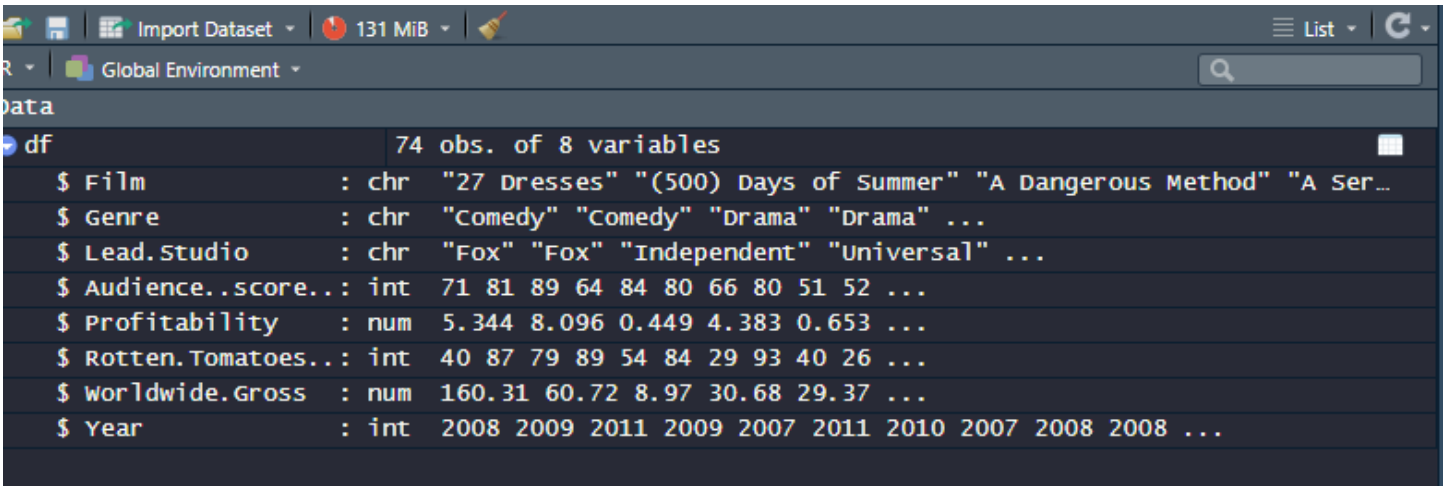
R studio

1.Load data into R Studio

You can load data into R Studio either by importing the dataset from a file on your computer or Microsoft account, or you can use a URL if the data is available online to upload it using a line of code including the URL. In this case we have used a CSV file available from Tableau public and were able to access it in R using the line of programming shown below.



2. Take a look at the data



You can double click the title of the database and it will open an additional window where you can view the data. You can also use the code 'View(df)'.

Untitled1* df

Filter

	Film	Genre	Lead.Studio	Audience..score..	Profitability	Rotten.Tomatoes..	Worldwide.Gross	Year
1	27 Dresses	Comedy	Fox	71	5.3436218	40	160.308654	2008
2	(500) Days of Summer	Comedy	Fox	81	8.0960000	87	60.720000	2009
3	A Dangerous Method	Drama	Independent	89	0.4486447	79	8.972895	2011
4	A Serious Man	Drama	Universal	64	4.3828571	89	30.680000	2009
5	Across the Universe	Romance	Independent	84	0.6526032	54	29.367143	2007
6	Beginners	Comedy	Independent	80	4.4718750	84	14.310000	2011
7	Dear John	Drama	Sony	66	4.5988000	29	114.970000	2010
8	Enchanted	Comedy	Disney	80	4.0057371	93	340.487652	2007
9	Fireproof	Drama	Independent	51	66.9340000	40	33.467000	2008
10	Four Christmases	Comedy	Warner Bros.	52	2.0229250	26	161.834000	2008
11	Ghosts of Girlfriends Past	Comedy	Warner Bros.	47	2.0444000	27	102.220000	2009
12	Gnomeo and Juliet	Animation	Disney	52	5.3879722	56	193.967000	2011
13	Going the Distance	Comedy	Warner Bros.	56	1.3140625	53	42.050000	2010
14	Good Luck Chuck	Comedy	Lionsgate	61	2.3676851	3	59.192128	2007
15	He's Just Not That Into You	Comedy	Warner Bros.	60	7.1536000	42	178.840000	2009
16	High School Musical 3: Senior Year	Comedy	Disney	76	22.9131365	65	252.044501	2008
17	I Love You Phillip Morris	Comedy	Independent	57	1.3400000	71	20.100000	2010
18	It's Complicated	Comedy	Universal	63	2.6423529	56	224.600000	2009
19	Jane Eyre	Romance	Universal	77	NA	85	30.147000	2011

You can also use “summary(df)” to give you a summary of the ‘df’ database, results are shown below.

```
> summary(df)
    Film              Genre            Lead.Studio        Audience..score..  Profitability     Rotten.Tomatoes..  Worldwide.Gross
 Length:74          Length:74          Length:74          Min.   :35.00     Min.   : 0.005   Min.   : 3.00     Min.   :  0.025
 Class :character   Class :character   Class :character   1st Qu.:52.00     1st Qu.: 1.791   1st Qu.:27.00     1st Qu.: 32.447
 Mode  :character   Mode  :character   Mode  :character   Median :64.00     Median : 2.642   Median :45.00     Median : 73.199
                                                          Mean   :64.14     Mean   : 4.742   Mean   :47.36     Mean   :136.352
                                                          3rd Qu.:76.00     3rd Qu.: 4.851   3rd Qu.:65.00     3rd Qu.:190.185
                                                          Max.   :89.00     Max.   :66.934   Max.   :96.00     Max.   :709.820
                                                          NA's   :1         NA's   :3        NA's   :1
     Year
 Min.   :2007
 1st Qu.:2008
 Median :2009
 Mean   :2009
 3rd Qu.:2010
 Max.   :2011
```

3. Install tidyverse package

You can install packages using a line of programming. The ‘tidyverse’ package is useful for data visualization, manipulation and exploration.

```
5
6  install.packages("tidyverse")
```

You can see that the package has installed successfully below:

```
install.packages( 'tidyverse' )
ARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of
tools before proceeding:

tps://cran.rstudio.com/bin/windows/Rtools/
nstalling package into 'C:/Users/emmak/AppData/Local/R/win-library/4.4'
s 'lib' is unspecified)
rying URL 'https://cran.rstudio.com/bin/windows/contrib/4.4/tidyverse_2.0.0.zip'
ontent type 'application/zip' length 431340 bytes (421 KB)
ownloaded 421 KB

ckage 'tidyverse' successfully unpacked and MD5 sums checked

e downloaded binary packages are in
        C:\Users\emmak\AppData\Local\Temp\Rtmp6vofd7\downloaded_packages
```

4. Import tidyverse library

Now that the tidyverse package has been downloaded, using the below line of code, you can import the tidyverse library which was downloaded so that it can be used in the workbook to query data.

```
library(tidyverse)
```

```
> library(tidyverse)
── Attaching core tidyverse packages ──────────────────────────────────────── tidyverse 2.0.0 ──
✓ dplyr     1.1.4     ✓ readr     2.1.5
✓ forcats   1.0.0     ✓ stringr   1.5.1
✓ ggplot2   3.5.1     ✓ tibble    3.2.1
✓ lubridate 1.9.3     ✓ tidyr     1.3.1
✓ purrr     1.0.2
── Conflicts ────────────────────────────────────────────────────────── tidyverse_conflicts() ──
✗ dplyr::filter() masks stats::filter()
✗ dplyr::lag()    masks stats::lag()
ℹ Use the conflicted package to force all conflicts to become errors
>
```

5. Check the data types

```
str(df)
```

This shows which datatype has been assigned to each column. For example, from here you can see that 'Film' is lister as having chr (character) datatype while the 'audience score' is listed as int (integer) datatype.

```
> 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 ...
```

6. Check for missing values

The code below tells us how much data is missing from each column.

```
colSums(is.na(df))
```

Below we can see that there are a few columns with some missing values. Profitability is missing 3 values while audience scores and rotten tomato's scores are each missing 1 value.

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

7. Drop missing values

Here we can omit any rows from the dataset which have missing values.

```
df <- na.omit(df)
```

8. Check to make sure that the rows have been removed

We can see below that there are no longer any missing data in any rows when running the code from question 6.

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

Earlier we had 74 observations and now when looking at the data we have 70 observations, meaning that 4 rows have been removed from the table when omitting values in questions 7.

```
Data
df                          70 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 ...
    - attr(*, "na.action")= 'omit' Named int [1:4] 19 42 51 71
     ..- attr(*, "names")= chr [1:4] "19" "42" "51" "71"
```

9. Summary Statistics

```
summary(df)
```

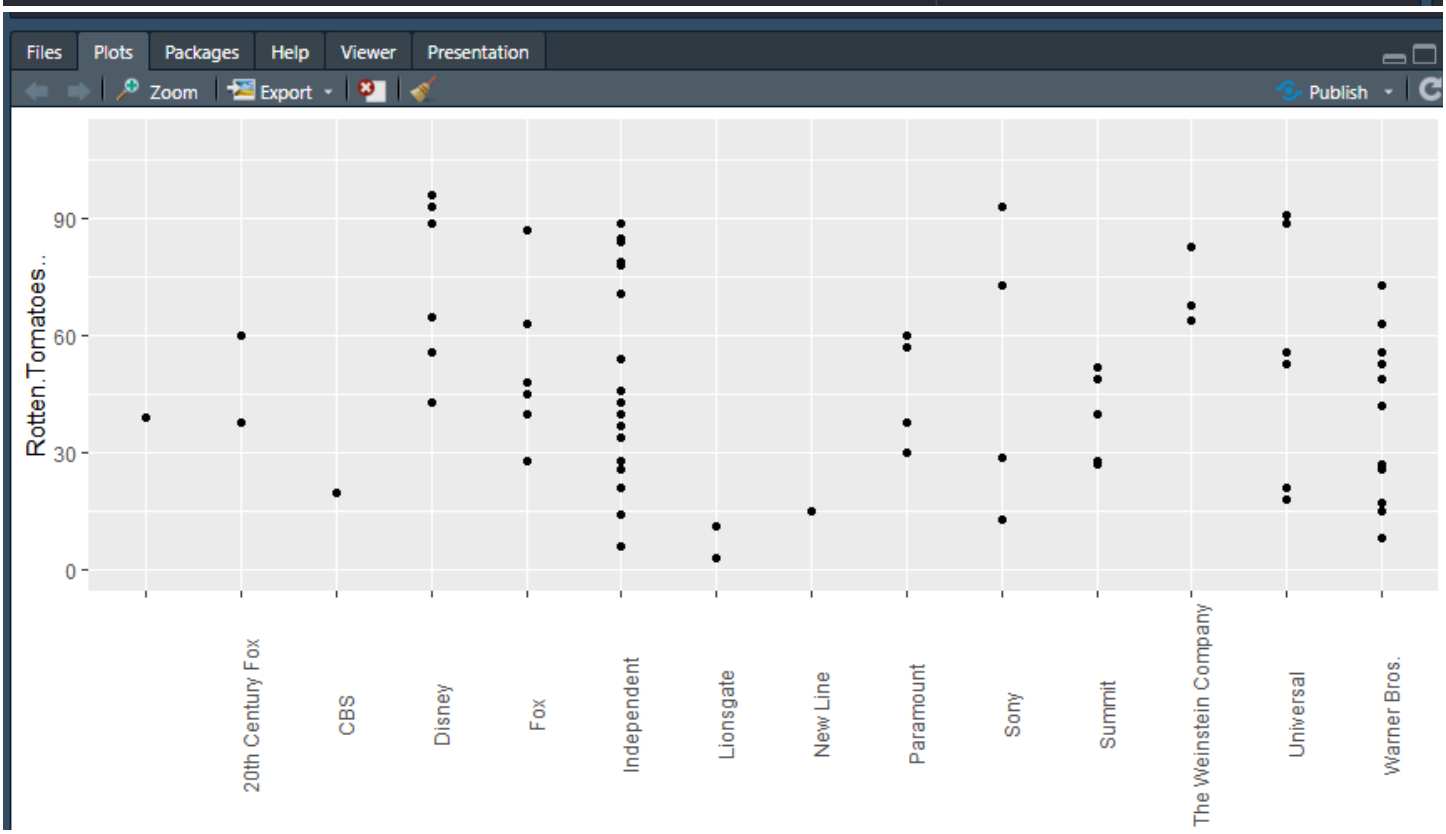
We can also see in the summary of data that there are no longer and 'N/A' values.

```
> summary(df)
     Film              Genre           Lead.Studio        Audience..score..  Profitability     Rotten.Tomatoes..
 Length:70          Length:70          Length:70          Min.   :35.00      Min.   : 0.005   Min.   : 3.00
 Class :character   Class :character   Class :character   1st Qu.:53.25      1st Qu.: 1.802   1st Qu.:27.25
 Mode  :character   Mode  :character   Mode  :character   Median :64.50      Median : 2.646   Median :45.50
                                                           Mean   :64.46      Mean   : 4.785   Mean   :47.76
                                                           3rd Qu.:75.50      3rd Qu.: 4.977   3rd Qu.:64.75
                                                           Max.   :89.00      Max.   :66.934   Max.   :96.00
 Worldwide.Gross        Year
 Min.   :  0.025   Min.   :2007
 1st Qu.: 32.809   1st Qu.:2008
 Median : 85.891   Median :2009
 Mean   :141.933   Mean   :2009
 3rd Qu.:202.467   3rd Qu.:2010
 Max.   :709.820   Max.   :2011
>
```

10. Scatterplot

R Studio is also able to show you charts based on the data. Below is a scatterplot chart which shows the distribution of rotten tomatoes scores for each studio:

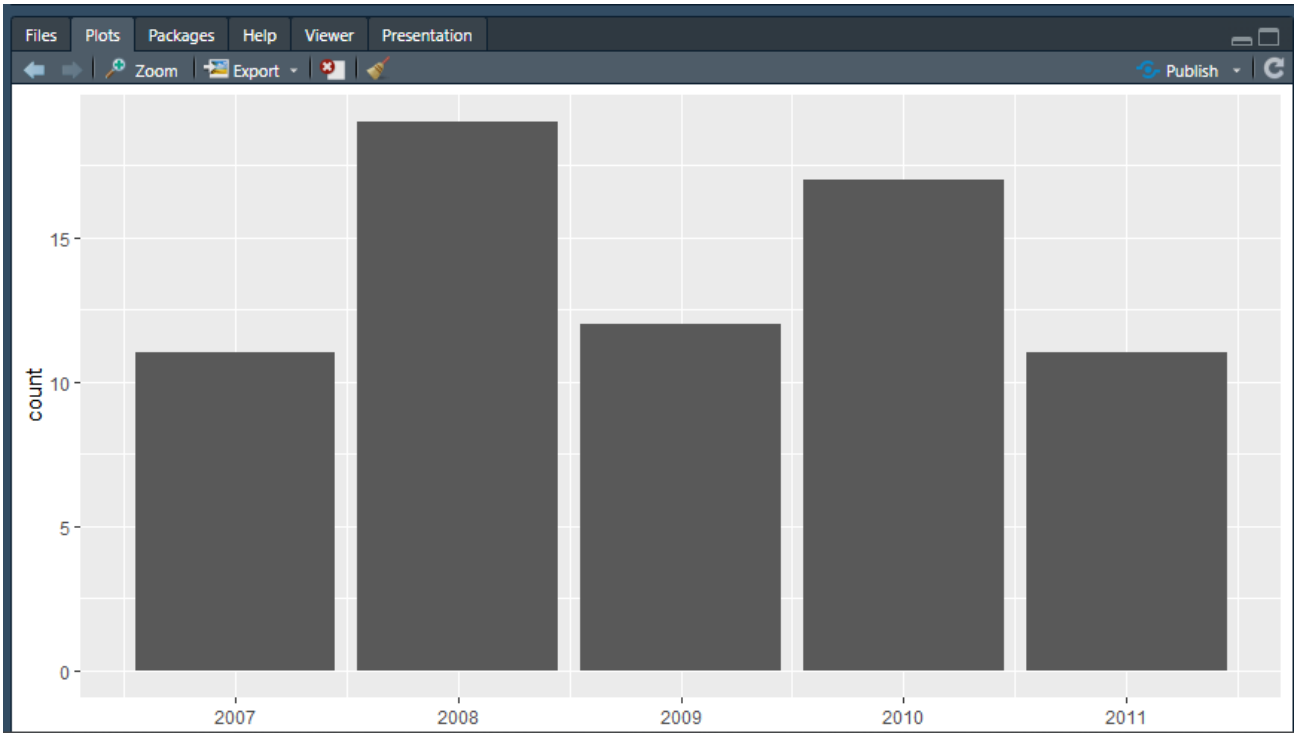
```
ggplot(df, aes(x=Lead.Studio, y=Rotten.Tomatoes..)) + geom_point()+ scale_y_continuous(labels = scales::comma)+coord_carte
```



11. Bar Chart

Below is a bar chart showing how many movies were made each year. We can see that out of the years included in this data that the most movies were made in 2008.

```
ggplot(df, aes(x=Year)) + geom_bar()
```



12. Export Clean data

Here you can export the cleaned data (not including the rows which were removed earlier) into a csv file which can be used in other software, such as Power BI which I will be using next to do some more detailed visualisations using this data

```
write.csv(df, "clean_df.csv")
```

The code below shows you where the file has been saved on your computer.

```
getwd()
```

```
> getwd()
[1] "C:/Users/emmak/OneDrive/Documents/Day5practice"
>
```

Power BI

For this dashboard there were a few requirements as part of the brief. The requirements were:

- Show the average Rotten Tomatoes rating of each genre
- The number of movies produced each year
- The audience score for each film
- The profitability per studio
- The worldwide gross per genre
- Use the brand colours which are blue, green and brown as part of the visualization

Considering all these requirements, I made the visualizations below and tried to make it visually pleasing and user-friendly. I used the company colours and tried to use the best charts for each data comparison which the client wanted to see from this project.