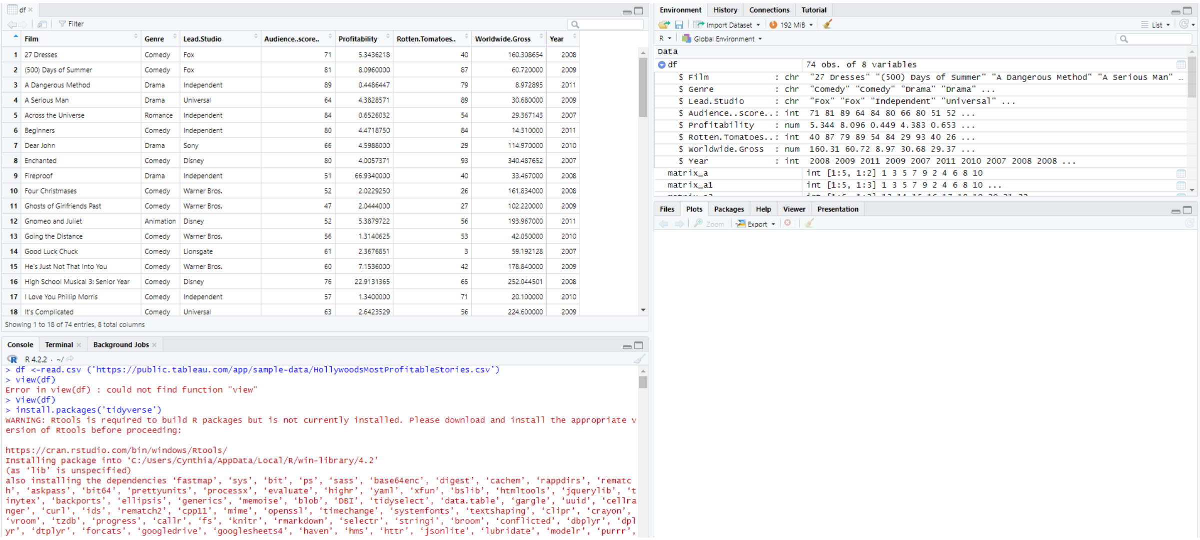
R Studio

Step 1: Initial Exploratory Analysis



- Import the data into R by using the data frame function
- Install the package ‘**tidyverse**’ (it helps to transform and better present data)

```
    C:\Users\Cynthia\AppData\Local\Temp\RtmpysdBoP\downloaded_packages
> library(tidyverse)
── Attaching core tidyverse packages ──────────────────────────── tidyverse 2.0.0 ──
✓ dplyr     1.1.1     ✓ readr     2.1.4
✓ forcats   1.0.0     ✓ stringr   1.5.0
✓ ggplot2   3.4.1     ✓ tibble    3.2.0
✓ lubridate 1.9.2     ✓ tidyr     1.3.0
✓ purrr     1.0.1
── 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
Warning messages:
1: package ‘tidyverse’ was built under R version 4.2.3
2: package ‘tidyr’ was built under R version 4.2.3
3: package ‘readr’ was built under R version 4.2.3
4: package ‘purrr’ was built under R version 4.2.3
5: package ‘dplyr’ was built under R version 4.2.3
6: package ‘stringr’ was built under R version 4.2.3
7: package ‘forcats’ was built under R version 4.2.3
8: package ‘lubridate’ was built under R version 4.2.3
> 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 ...
> |
```

- Import library ‘**tidyverse**’
- Check the data type of each variable

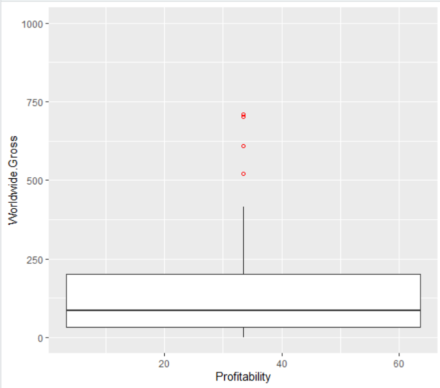
Step 2: Clean Data

```
> colSums(is.na(df))
             Film             Genre      Lead.Studio Audience..score..     Profitability Rotten.Tomatoes..
                0                 0                0                 1                 3                 1
  Worldwide.Gross              Year
                0                 0
> df <- na.omit(df)
> colSums(is.na(df))
             Film             Genre      Lead.Studio Audience..score..     Profitability Rotten.Tomatoes..
                0                 0                0                 0                 0                 0
  Worldwide.Gross              Year
                0                 0
> dim(df[duplicated(df$Film,)])[1]
[1] 70
> df$Profitability <- round(df$Profitability, digit=2)
> dim (df)
[1] 70  8
> df$Worldwide.Gross <- round(df$Worldwide.Gross, digit=2)
> dim (df)
[1] 70  8
>
```

- Using **'colSums(is.na)'** to count the NA value in each variable inside the data frame
- **'Na.omit'** is used to drop the missing value (NA)
- Using **'duplicated'** to check for duplicate
- '**Round**' is used to round off values
- **'Dim'** is used to get the dimensions of the data frame

Step 2.1 Outlier Removal

```
R 4.2.2 · ~/
"ggplot(df,aes(x=Profitability, y=Worldwide.Gross)
geom_boxplot"
> ggplot(df,aes(x=Profitability, y=Worldwide.Gross))+
+ geom_boxplot(outlier.colour = 'red', outlier.shape = 1)+
+ scale_x_continuous(labels=scales::comma)+ coord_cartesian(ylim= c(0,1000))
Warning message:
Continuous x aesthetic
i did you forget `aes(group = ...)`?
> ggplot(df,aes(x=Profitability, y=Worldwide.Gross))+geom_boxplot(outlier.colour = 'red', outlier.shape = 1)+scale_x
_continuous(labels=scales::comma)+ coord_cartesian(ylim= c(0,1000))
Warning message:
Continuous x aesthetic
i did you forget `aes(group = ...)`?
> 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$Profitability< (Q3+1.5*IQR))
Error: unexpected ')' in "no_outliers <- subset(df, df$Profitability> (Q1-1.5*IQR))& df$Profitability< (Q3+1.5*IQ
R))"
> no_outliers <- subset(df, df$Profitability> (Q1-1.5*IQR))& df$Profitability< (Q3+1.5*IQR)
Error in FUN(left, right) :
  operations are possible only for numeric, logical or complex types
> no_outliers <- subset(df,df$Profitability> (Q1-1.5*IQR) & df$Profitability< (Q3+1.5*IQR))
>
> no_outliers <- subset(df,df$Profitability> (Q1-1.5*IQR) & df$Profitability< (Q3+1.5*IQR))
>
> dim(no_outliers)
[1] 65  8
> |
```



- **Other outliers** are problematic and should be removed because they represent measurement errors, data entry or processing errors, or poor sampling
- The boxplot is shown on the right-hand side
- To remove outliers in 'Profitability', we first need to calculate the value of Q1(25%), Q3(75%) and IQR (Q3-Q1). Then, find
    1. Upper boundary (Anything above Q3 + 1.5 x IQR is an outlier)
    2. Lower boundary (Anything below Q1 - 1.5 x IQR is an outlier)

  The value that is out of this range will be removed to increase the accuracy

  The syntax of 'no_outliers' is getting the data in the range of the upper boundary and lower boundary

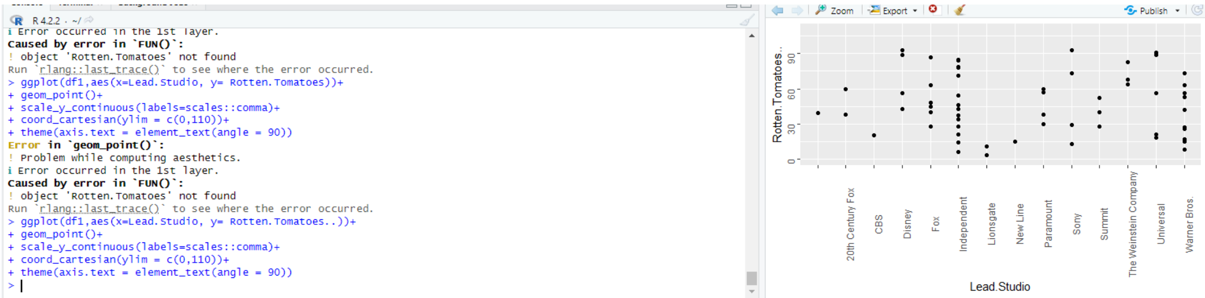
  Dimension of 'no_outliers' data **65, 8**

```
> Q1 <- quantile(no_outliers$Worldwide.Gross.25)
> Q3 <- quantile(no_outliers$Worldwide.Gross, .75)
> Q1 <- quantile(no_outliers$Worldwide.Gross, .25)
> IQR <- IQR(no_outliers$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
```

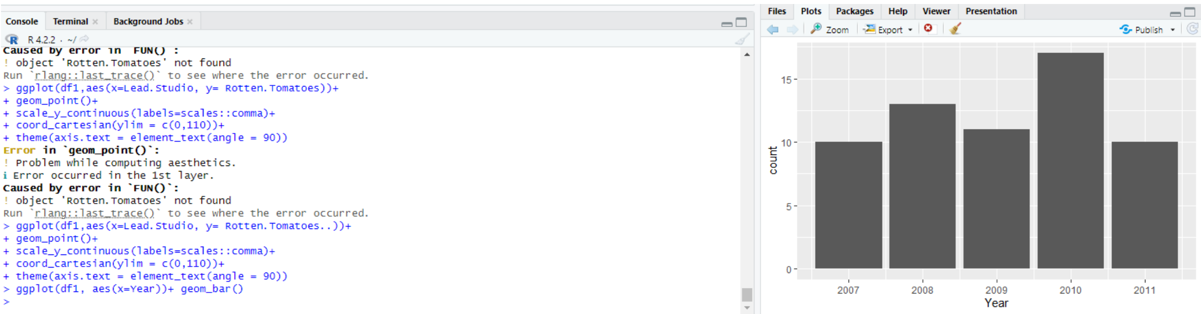
- Use the 'no_outliers' data to continue to remove outer outliers in 'Worldwide. Gross'

- The data frame dimension has now been reduced to **61, 8**

Step 3: Exploratory Data Analysis

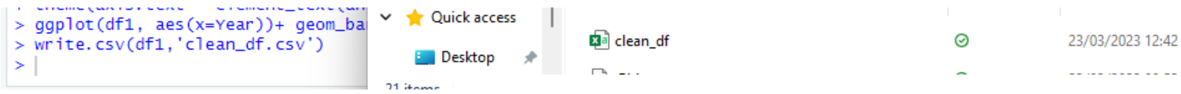


- Scatter plot of the **df1**, showing the rotten tomatoes rating for every movie per studio
- According to rotten tomatoes, 'Independent' produced the highest number of movies and it also has a few movies rate above 60%. Whereas, overall 'Lionsgate' produce movies with the lowest rating



- Bar chart of the **df1**, count the year
- From this graph, we can tell that 2010 produce the most movies and a sharp decline in 2011

Step 4: Export Data



- **Write.csv** is used to export data

Power BI

In this dashboard, I used a range of visualisation and embedded R script into the Power BI to show the performance of Hollywood Movies. Also, I mainly use the colour blue, brown and green to meet the client's criteria

Type of visualisations used:

- Use **'card'** to display the number of films in the dataset
- **'Slicer**' is used to show the 6 types of genres and years (2007-2011)
- **'Stacked bar chart'** shows the profitability per studio. Different types of films show in a different colour on each bar
- Using the **'ggplot'** in R script to create a bar chart for finding the average rotten tomatoes rating of each genre. Gradient blue is used to show the type of genre
- **'Line chart'** is used to see how can movies ratings and scores affect the worldwide gross
- **'Area chart'** suggest the worldwide gross per genre
- **'Donut chart'** shows the number of movies produced per year