

Data Analytics Assignment using R (Group 2)

```
# loading the dataset into R
dataset <- read.csv("movies.csv")
```

```
# Printing the structure of the dataset
str(dataset)
```

```
## 'data.frame':    2000 obs. of  13 variables:
## $ Title          : chr  "Avatar: The Way of Water" "Guillermo del Toro's Pinocchio" "Bullet Train"
## $ Rating         : num  8 7.8 7.3 8 NA 5.9 6.1 6.9 8.2 7.8 ...
## $ Year           : int   2022 2022 2022 2022 2022 2022 2022 2022 2022 2022 ...
## $ Month          : chr   "December" "December" "August" "November" ...
## $ Certificate     : chr   "PG-13" "PG" "R" "R" ...
## $ Runtime        : chr   "192" "117" "127" "114" ...
## $ Directors      : chr   "James Cameron" "Guillermo del Toro, Mark Gustafson" "David Leitch" "Mart
## $ Stars          : chr   "Sam Worthington, Zoe Saldana, Sigourney Weaver, Stephen Lang" "Ewan McGro
## $ Genre          : chr   "Action, Adventure, Fantasy" "Animation, Drama, Family" "Action, Comedy, 
## $ Filming_location : chr   "New Zealand" "USA" "Japan" "Ireland" ...
## $ Budget         : chr   "$350,000,000" "$35,000,000" "$85,900,000" "Unknown" ...
## $ Income         : chr   "$681,081,686" "$71,614" "$239,268,602" "$19,720,823" ...
## $ Country_of_origin: chr   "United States" "United States, Mexico, France" "Japan, United States" "I
```

```
# List the variables in the dataset
names(dataset)
```

```
## [1] "Title"          "Rating"          "Year"
## [4] "Month"          "Certificate"      "Runtime"
## [7] "Directors"      "Stars"           "Genre"
## [10] "Filming_location" "Budget"          "Income"
## [13] "Country_of_origin"
```

```
# Printing the top 15 rows of the dataset
head(dataset, 15)
```

```
##           Title Rating Year   Month Certificate Runtime
## 1  Avatar: The Way of Water    8.0 2022 December      PG-13    192
## 2 Guillermo del Toro's Pinocchio    7.8 2022 December        PG    117
## 3      Bullet Train    7.3 2022   August          R    127
## 4  The Banshees of Inisherin    8.0 2022 November          R    114
## 5             M3gan      NA 2022  January      PG-13    102
## 6      Emancipation    5.9 2022 December          R    132
## 7      Amsterdam    6.1 2022  October          R    134
## 8    Violent Night    6.9 2022 December          R    112
## 9      The Whale    8.2 2022 December          R    117
```

## 10	The Fabelmans	7.8	2022	November	PG-13	151
## 11	The Menu	7.5	2022	November	R	107
## 12	Babylon	7.7	2022	December	R	188
## 13	X	6.6	2022	March	R	105
## 14	Bones and All	7.0	2022	November	R	131
## 15	Black Adam	6.5	2022	October	PG-13	125
##	Directors					
## 1	James Cameron					
## 2	Guillermo del Toro, Mark Gustafson					
## 3	David Leitch					
## 4	Martin McDonagh					
## 5	Gerard Johnstone					
## 6	Antoine Fuqua					
## 7	David O Russell					
## 8	Tommy Wirkola					
## 9	Darren Aronofsky					
## 10	Steven Spielberg					
## 11	Mark Mylod					
## 12	Damien Chazelle					
## 13	Ti West					
## 14	Luca Guadagnino					
## 15	Jaume Collet Serra					
##	Stars					
## 1	Sam Worthington, Zoe Saldana, Sigourney Weaver, Stephen Lang					
## 2	Ewan McGregor, David Bradley, Gregory Mann, Burn Gorman					
## 3	Brad Pitt, Joey King, Aaron Taylor Johnson, Brian Tyree Henry					
## 4	Colin Farrell, Brendan Gleeson, Kerry Condon, Pat Shortt					
## 5	Jenna Davis, Amie Donald, Allison Williams, Violet McGraw					
## 6	Will Smith, Ben Foster, Charmaine Bingwa, Gilbert Owuor					
## 7	Christian Bale, Margot Robbie, John David Washington, Alessandro Nivola					
## 8	David Harbour, John Leguizamo, Beverly D Angelo, Alex Hassell					
## 9	Brendan Fraser, Sadie Sink, Ty Simpkins, Hong Chau					
## 10	Michelle Williams, Gabriel LaBelle, Paul Dano, Judd Hirsch					
## 11	Ralph Fiennes, Anya Taylor Joy, Nicholas Hoult, Hong Chau					
## 12	Brad Pitt, Margot Robbie, Jean Smart, Olivia Wilde					
## 13	Mia Goth, Jenna Ortega, Brittany Snow, Kid Cudi					
## 14	Timoth e Chalamet, Taylor Russell, Mark Rylance, Kendle Coffey					
## 15	Dwayne Johnson, Aldis Hodge, Pierce Brosnan, Noah Centineo					
##	Genre	Filming_location	Budget	Income		
## 1	Action, Adventure, Fantasy	New Zealand	\$350,000,000	\$681,081,686		
## 2	Animation, Drama, Family	USA	\$35,000,000	\$71,614		
## 3	Action, Comedy, Thriller	Japan	\$85,900,000	\$239,268,602		
## 4	Comedy, Drama	Ireland	Unknown	\$19,720,823		
## 5	Horror, Sci-Fi, Thriller	New Zealand	Unknown	Unknown		
## 6	Action, Thriller	Unknown	\$120,000,000	Unknown		
## 7	Comedy, Drama, History	USA	\$80,000,000	\$31,245,810		
## 8	Action, Comedy, Crime	Canada	\$20,000,000	\$59,595,460		
## 9	Drama	USA	Unknown	\$1,858,238		
## 10	Drama	USA	\$40,000,000	\$9,500,361		
## 11	Comedy, Horror, Thriller	USA	\$35,000,000	\$65,878,071		
## 12	Comedy, Drama, History	USA	\$78,000,000	\$1,470		
## 13	Horror, Mystery, Thriller	New Zealand	\$1,000,000	\$14,779,858		
## 14	Drama, Horror, Romance	USA	\$16,000,000	\$14,134,907		
## 15	Action, Adventure, Fantasy	USA	\$195,000,000	\$391,273,355		

```
##                               Country_of_origin
## 1                               United States
## 2                United States, Mexico, France
## 3                        Japan, United States
## 4        Ireland, United Kingdom, United States
## 5                               United States
## 6                               United States
## 7                United States, Japan
## 8                United States, Canada
## 9                               United States
## 10                              United States
## 11                              United States
## 12                              United States
## 13                United States, Canada
## 14                Italy, United States
## 15 United States, Canada, New Zealand, Hungary
```

```
# User defined function to calculate profit based on the budget and the income
```

```
calculate_profit <- function(dataset) {
  dataset$profit <- dataset$income - dataset$budget
  return(dataset)
}
```

```
# Filtering rows based on ratings
```

```
new_df <- dataset[dataset$Rating > 8.5, ]
```

```
# Creating a new data frame by joining dependant and independant variables
```

```
dependent_var <- c("Rating", "Income")
```

```
independent_var <- c("Budget", "Year", "Runtime")
```

```
new_df2 <- cbind(dataset[, dependent_var], dataset[, independent_var])
```

```
# Removing missing values
```

```
dataset <- dataset[complete.cases(dataset), ]
```

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

```
# Removing duplicated data
```

```
dataset <- unique(dataset)
```

```
# Re-ordering rows in descending order
```

```
dataset <- dataset[order(dataset$Year, decreasing = TRUE), ]
```

```
dataset <- dataset[order(dataset$Rating, decreasing = TRUE), ]
```

```
# Renaming some of the column names
```

```
colnames(dataset) <- c("Movie Title", "Rating", "Year", "Month", "Certificate", "Runtime",
"Directors", "Cast", "Genre", "Location", "Budget", "Income", "Country")
```

```
# Adding new variables by using a mathematical function
```

```
dataset$Year_2x <- dataset$Year * 2
```

```
# Creating a training set using a random number generator engine
```

```
set.seed(123)
```

```
train_index <- sample(1:nrow(dataset), 0.8 * nrow(dataset))
```

```
train_set <- dataset[train_index, ]
```

```
test_set <- dataset[-train_index, ]
```

```
# Printing the summary statistics of the dataset
```

```
summary(dataset)
```

```
## Movie Title           Rating           Year           Month
## Length:1998          Min.      :1.900      Min.      :2003      Length:1998
## Class :character      1st Qu.:6.125      1st Qu.:2007      Class :character
## Mode  :character      Median :6.700      Median :2012      Mode  :character
##                               Mean   :6.668      Mean   :2012
##                               3rd Qu.:7.300      3rd Qu.:2017
##                               Max.    :9.600      Max.    :2022
## Certificate           Runtime           Directors          Cast
## Length:1998          Length:1998          Length:1998          Length:1998
## Class :character      Class :character      Class :character      Class :character
## Mode  :character      Mode  :character      Mode  :character      Mode  :character
##
##
## Genre                Location           Budget           Income
## Length:1998          Length:1998          Length:1998          Length:1998
## Class :character      Class :character      Class :character      Class :character
## Mode  :character      Mode  :character      Mode  :character      Mode  :character
##
##
## Country              Year_2x
## Length:1998          Min.      :4006
## Class :character      1st Qu.:4014
## Mode  :character      Median :4024
##                               Mean   :4025
##                               3rd Qu.:4034
##                               Max.    :4044
```

```
# Using the income variables and performing these statistical functions
```

```
mean(dataset$Rating)
```

```
## [1] 6.667618
```

```
median(dataset$Rating)
```

```
## [1] 6.7
```

```
mode(dataset$Rating)
```

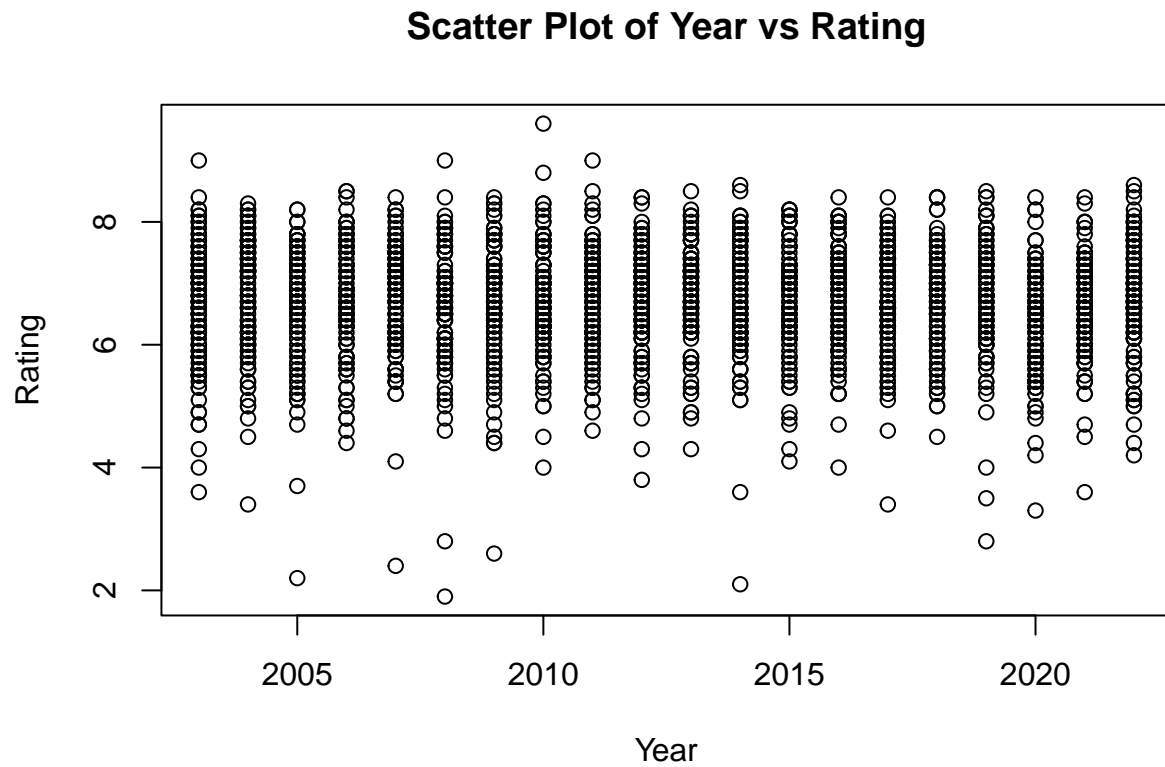
```
## [1] "numeric"
```

```
range(dataset$Rating)
```

```
## [1] 1.9 9.6
```

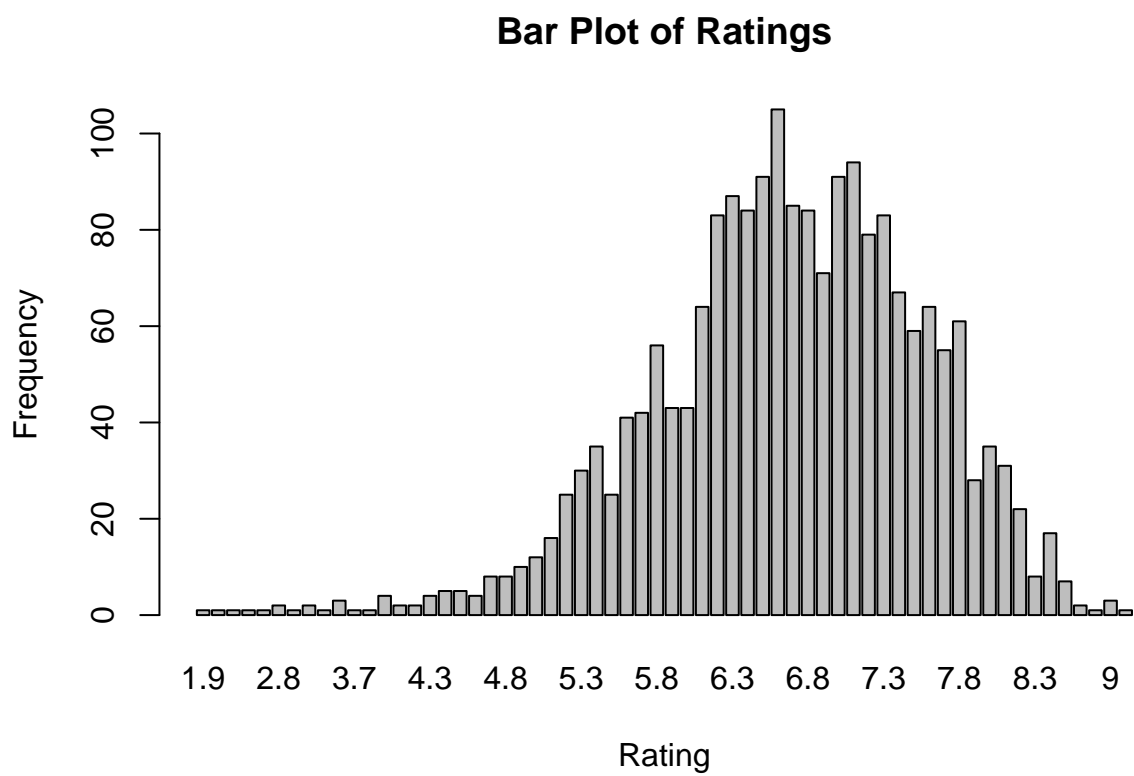
```
# Scatter plot for year vs rating
```

```
plot(dataset$Year, dataset$Rating, xlab = "Year", ylab = "Rating", main = "Scatter Plot of Year vs Rating")
```

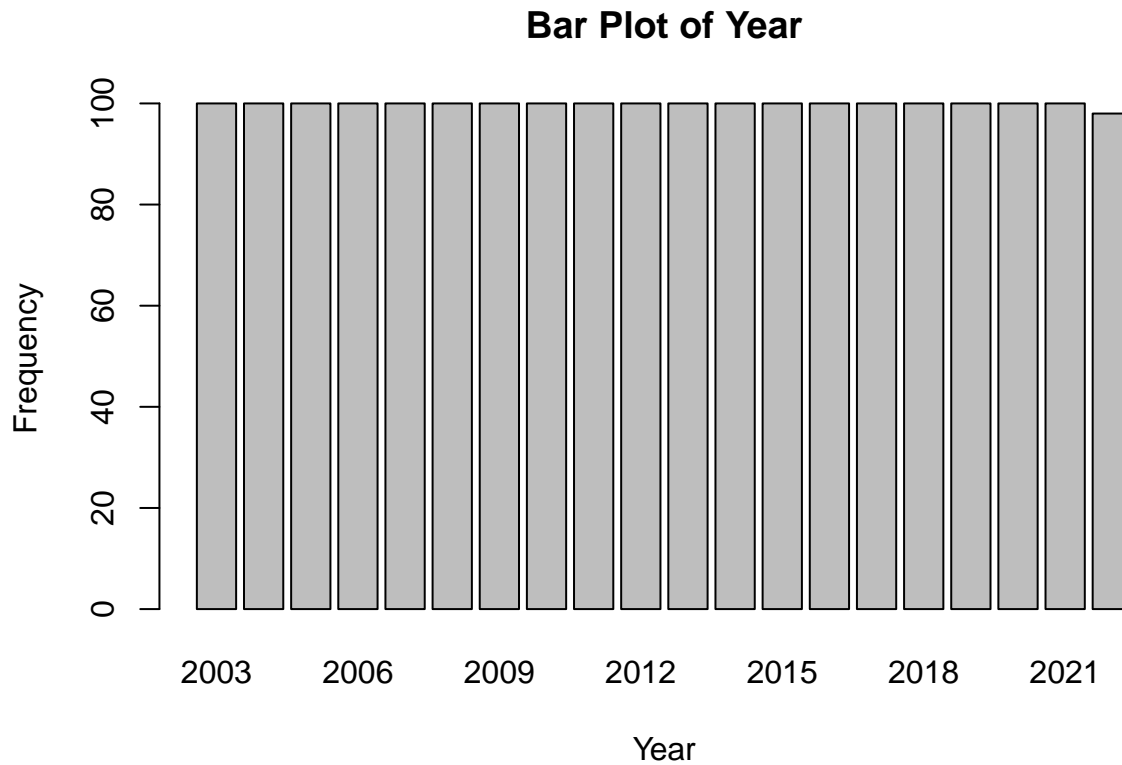


```
# Bar plot for the budget and income variables
```

```
barplot(table(dataset$Rating), xlab = "Rating", ylab = "Frequency", main = "Bar Plot of Ratings")
```



```
barplot(table(dataset$Year), xlab = "Year", ylab = "Frequency", main = "Bar Plot of Year")
```



```
# Finding the correlation between year released and ratings
model <- lm(Rating ~ Year, data = dataset)
summary(model)
```

```
##
## Call:
## lm(formula = Rating ~ Year, data = dataset)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4.7784 -0.5312  0.0336  0.6360  2.9264
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  11.482389   7.136100   1.609   0.108
## Year         -0.002392   0.003546  -0.675   0.500
##
## Residual standard error: 0.9132 on 1996 degrees of freedom
## Multiple R-squared:  0.000228,    Adjusted R-squared:  -0.0002729
## F-statistic: 0.4552 on 1 and 1996 DF,  p-value: 0.4999
```