Assignment 01

Cameron Atkins

Due: 2023-09-08

1.Data used for the assignment

The data used for this project is IMDb's list of Top 1000 movies of all time. The original data featured nine colums of data, but for the purposes of this assignment, columns of data was reduced to four.

Link to Kaggle DataSet (https://www.kaggle.com/datasets/inductiveanks/top-1000-imdb-movies-dataset)

Importing the data from the working directory

```
data <- read.csv("Top_1000_IMDb_movies_New_version.csv", header = TRUE)
```

Taking a look at the data

```
head(data)
```

```
##   X                                     Movie.Name Year.of.Release Watch.Time
## 1 0                       The Shawshank Redemption            1994        142
## 2 1                                  The Godfather            1972        175
## 3 2                                The Dark Knight            2008        152
## 4 3                               Schindler's List            1993        195
## 5 4                                   12 Angry Men            1957         96
## 6 5 The Lord of the Rings: The Return of the King            2003        201
##   Movie.Rating Metascore.of.movie  Gross     Votes
## 1          9.3                 82  28.34 27,77,378
## 2          9.2                100 134.97 19,33,588
## 3          9.0                 84 534.86 27,54,087
## 4          9.0                 95   96.9 13,97,886
## 5          9.0                 97   4.36  8,24,211
## 6          9.0                 94 377.85 19,04,166
##
Description
## 1                                                  Over the course of several y
ears, two convicts form a friendship, seeking consolation and, eventually, redemption
through basic compassion.
## 2 Don Vito Corleone, head of a mafia family, decides to hand over his empire to hi
s youngest son Michael. However, his decision unintentionally puts the lives of his l
oved ones in grave danger.
## 3   When the menace known as the Joker wreaks havoc and chaos on the people of Got
ham, Batman must accept one of the greatest psychological and physical tests of his a
bility to fight injustice.
## 4              In German-occupied Poland during World War II, industrialist Oskar
Schindler gradually becomes concerned for his Jewish workforce after witnessing their
persecution by the Nazis.
## 5        The jury in a New York City murder trial is frustrated by a single member
whose skeptical caution forces them to more carefully consider the evidence before ju
mping to a hasty verdict.
## 6                                             Gandalf and Aragorn lead the World
of Men against Sauron's army to draw his gaze from Frodo and Sam as they approach Mou
nt Doom with the One Ring.
```

Select 4 Variables: Year of Release, Watch Time, MetaScore, and Gross Ticket Sales

```
keep_columns <- c(3, 4, 6, 7)
```

Reduce the number of variables used

```
data <- data[, keep_columns]
```

View new data after reduction

```
head(data)
```

```
##   Year.of.Release Watch.Time Metascore.of.movie  Gross
## 1            1994        142                 82  28.34
## 2            1972        175                100 134.97
## 3            2008        152                 84 534.86
## 4            1993        195                 95   96.9
## 5            1957         96                 97   4.36
## 6            2003        201                 94 377.85
```

2. Summary of the data

Structure of the data

```
str(data)
```

```
## 'data.frame':    1000 obs. of  4 variables:
##  $ Year.of.Release   : chr  "1994" "1972" "2008" "1993" ...
##  $ Watch.Time        : int  142 175 152 195 96 201 202 140 154 148 ...
##  $ Metascore.of.movie: int  82 100 84 95 97 94 90 86 95 74 ...
##  $ Gross             : chr  "28.34" "134.97" "534.86" "96.9" ...
```

Summary of the data

```
summary(data)
```

```
##  Year.of.Release      Watch.Time    Metascore.of.movie    Gross
##  Length:1000        Min.   : 45.0   Min.   : 28.00     Length:1000
##  Class :character   1st Qu.:103.0   1st Qu.: 71.00     Class :character
##  Mode  :character   Median :120.0   Median : 80.00     Mode  :character
##                     Mean   :124.3   Mean   : 79.01
##                     3rd Qu.:139.0   3rd Qu.: 88.00
##                     Max.   :321.0   Max.   :100.00
##                                     NA's   :155
```

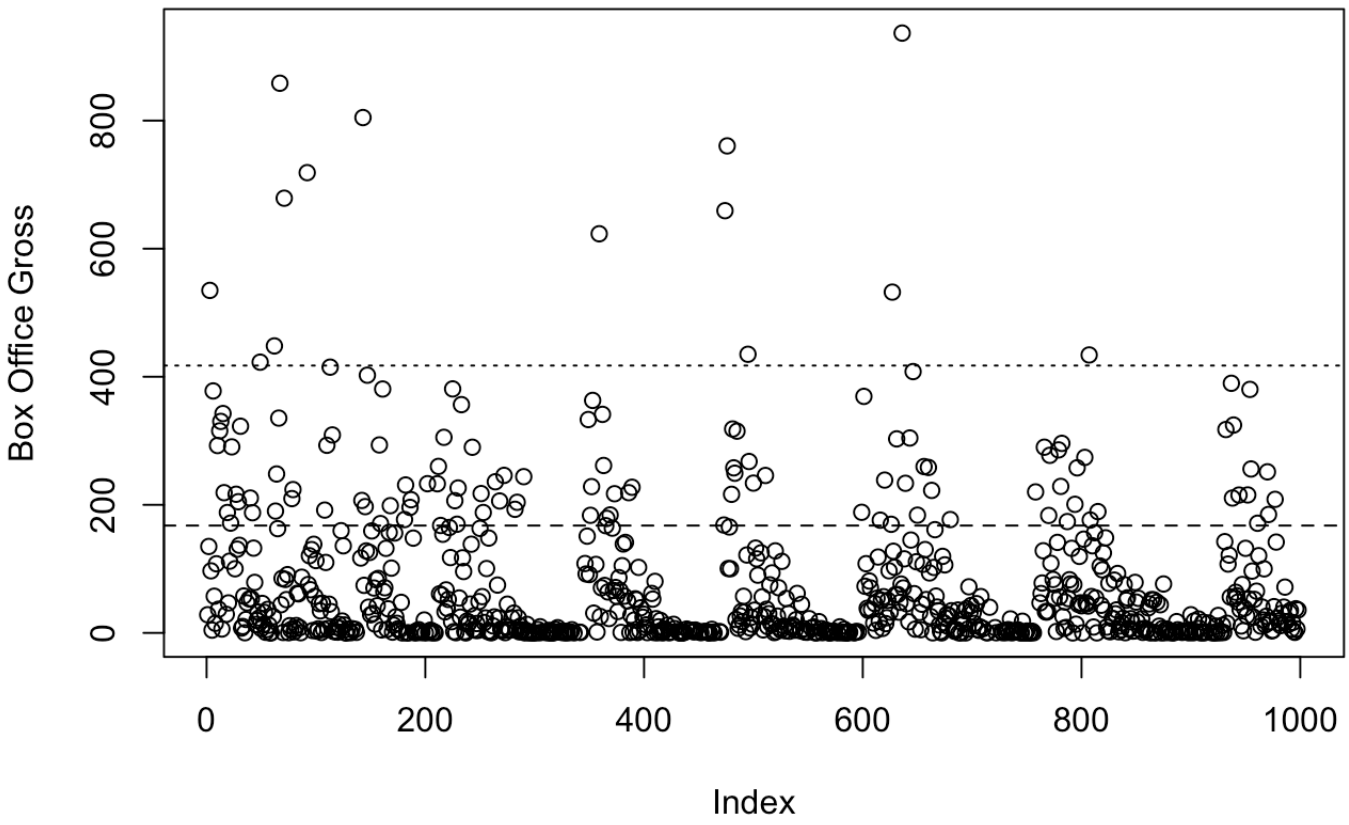
3. Search for any outliers in the box office gross

```
# Convert gross figures from character to numeric
numeric_gross <- as.numeric(data$Gross)
```

```
## Warning: NAs introduced by coercion
```

```
xu= quantile(numeric_gross, probs = 0.85, na.rm = TRUE)
xl = quantile(numeric_gross, probs = 0.15, na.rm = TRUE)
IQR = xu - xl

plot(numeric_gross, ylab='Box Office Gross')
abline(h=xu, lty=2)
abline(h=xu+1.5*IQR, lty=3)
```



Out of the 1000 movies, only 13 earned more than US $400 million. However, dataset doesn't specify if amounts are in current or real dollars.

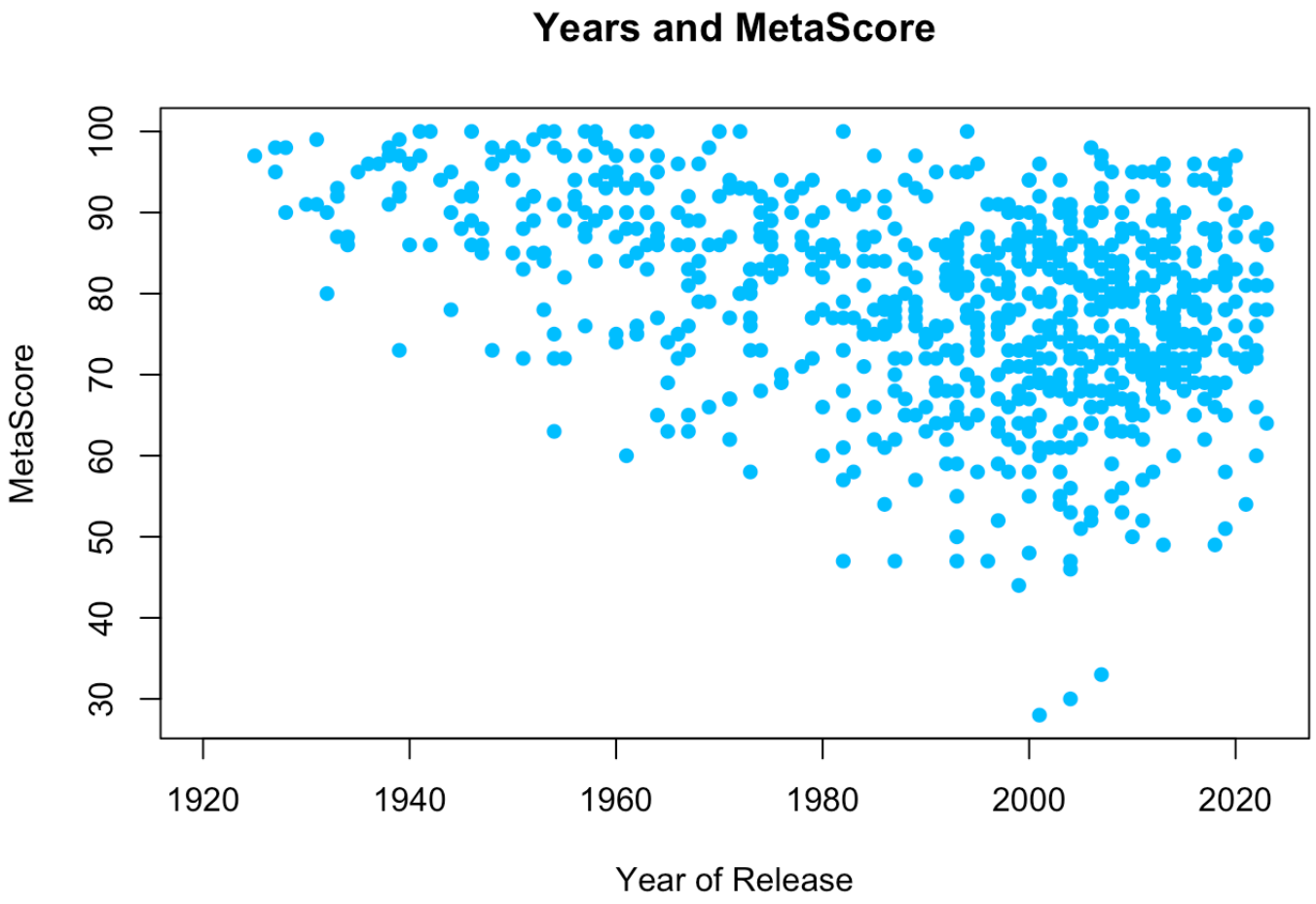
4. Visualizations for some variables

Scatterplot between 2 variables

A visualization that can help determine if films produced recently have higher metascores than older films.

```
plot(data$Year.of.Release, data$Metascore.of.movie,
     main = "Years and MetaScore",
     xlab = "Year of Release",
     ylab = "MetaScore",
     pch = 16,
     col = "deepskyblue"
     )
```

```
## Warning in xy.coords(x, y, xlabel, ylabel, log): NAs introduced by coercion
```

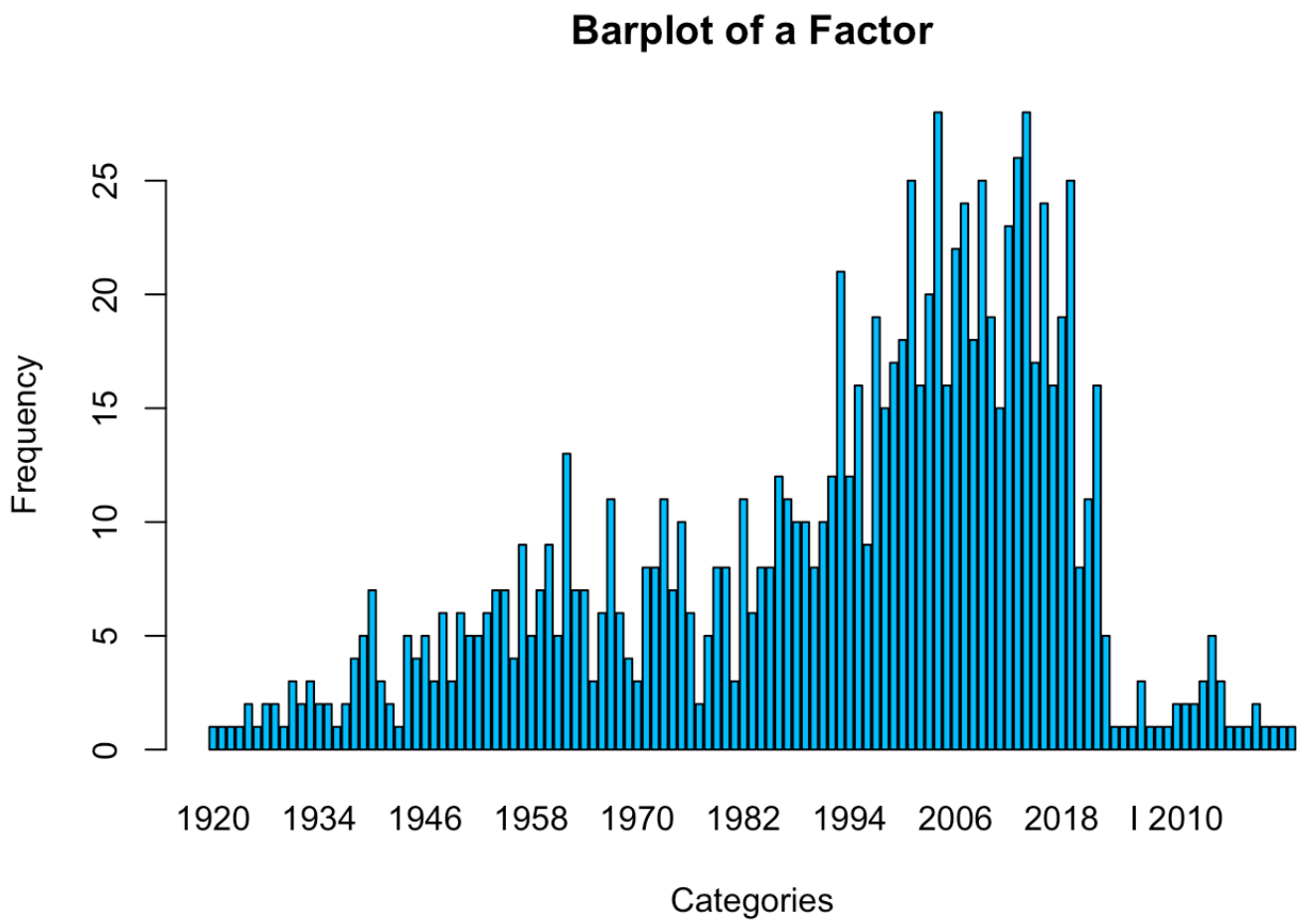


One barplot providing the count of the factor variable

```
year_factor <- factor(c(data$Year.of.Release))
year_table <- table(year_factor)
year_table
```

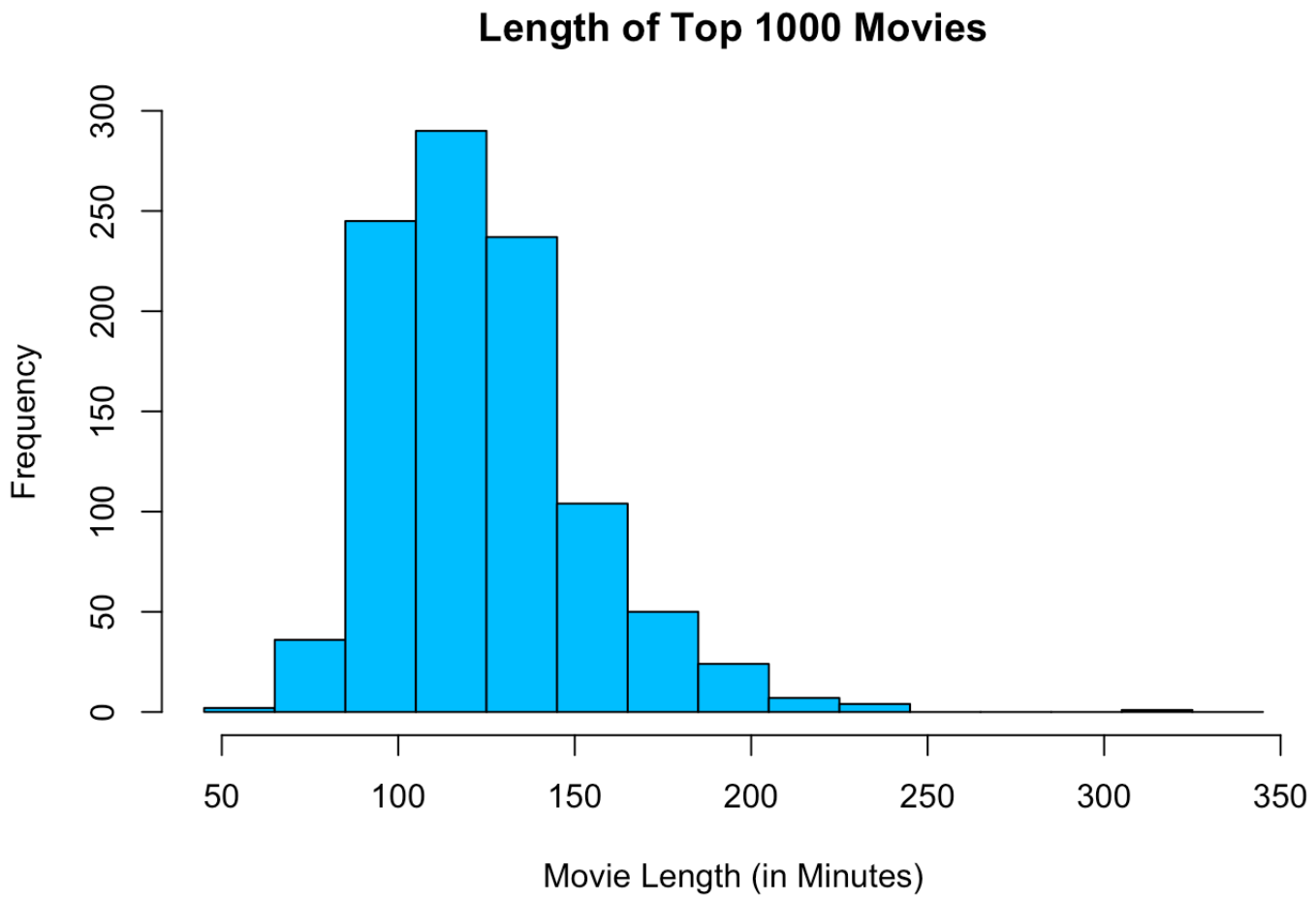
```
## year_factor
##     1920     1921     1922     1924     1925     1926     1927     1928
##        1        1        1        1        2        1        2        2
##     1930     1931     1932     1933     1934     1935     1936     1937
##        1        3        2        3        2        2        1        2
##     1938     1939     1940     1941     1942     1943     1944     1945
##        4        5        7        3        2        1        5        4
##     1946     1947     1948     1949     1950     1951     1952     1953
##        5        3        6        3        6        5        5        6
##     1954     1955     1956     1957     1958     1959     1960     1961
##        7        7        4        9        5        7        9        5
##     1962     1963     1964     1965     1966     1967     1968     1969
##       13        7        7        3        6       11        6        4
##     1970     1971     1972     1973     1974     1975     1976     1977
##        3        8        8       11        7       10        6        2
##     1978     1979     1980     1981     1982     1983     1984     1985
##        5        8        8        3       11        6        8        8
##     1986     1987     1988     1989     1990     1991     1992     1993
##       12       11       10       10        8       10       12       21
##     1994     1995     1996     1997     1998     1999     2000     2001
##       12       16        9       19       15       17       18       25
##     2002     2003     2004     2005     2006     2007     2008     2009
##       16       20       28       16       22       24       18       25
##     2010     2011     2012     2013     2014     2015     2016     2017
##       19       15       23       26       28       17       24       16
##     2018     2019     2020     2021     2022     2023   I 1985   I 1995
##       19       25        8       11       16        5        1        1
##   I 2001   I 2004   I 2006   I 2007   I 2008   I 2010   I 2011   I 2013
##        1        3        1        1        1        2        2        2
##   I 2014   I 2015   I 2017   I 2019   I 2020   I 2022  II 2016  II 2018
##        3        5        3        1        1        1        2        1
##  II 2022 III 2016 III 2018
##        1        1        1
```

```
barplot(year_table,
        main = "Barplot of a Factor",
        xlab = "Categories",
        ylab = "Frequency",
        col = "deepskyblue",  # Set the bar color (you can customize)
        border = "black"  # Set the bar border color (you can customize)
)
```



Histogram of the length of the Top 1000 movies

```
hist(data$Watch.Time,
     breaks = seq(45,350, by = 20),
     main = "Length of Top 1000 Movies",
     xlab = "Movie Length (in Minutes)",
     ylab = "Frequency",
     col = "deepskyblue",  # Set the bar color (you can customize)
     border = "black"    # Set the bar border color (you can customize)
)
```



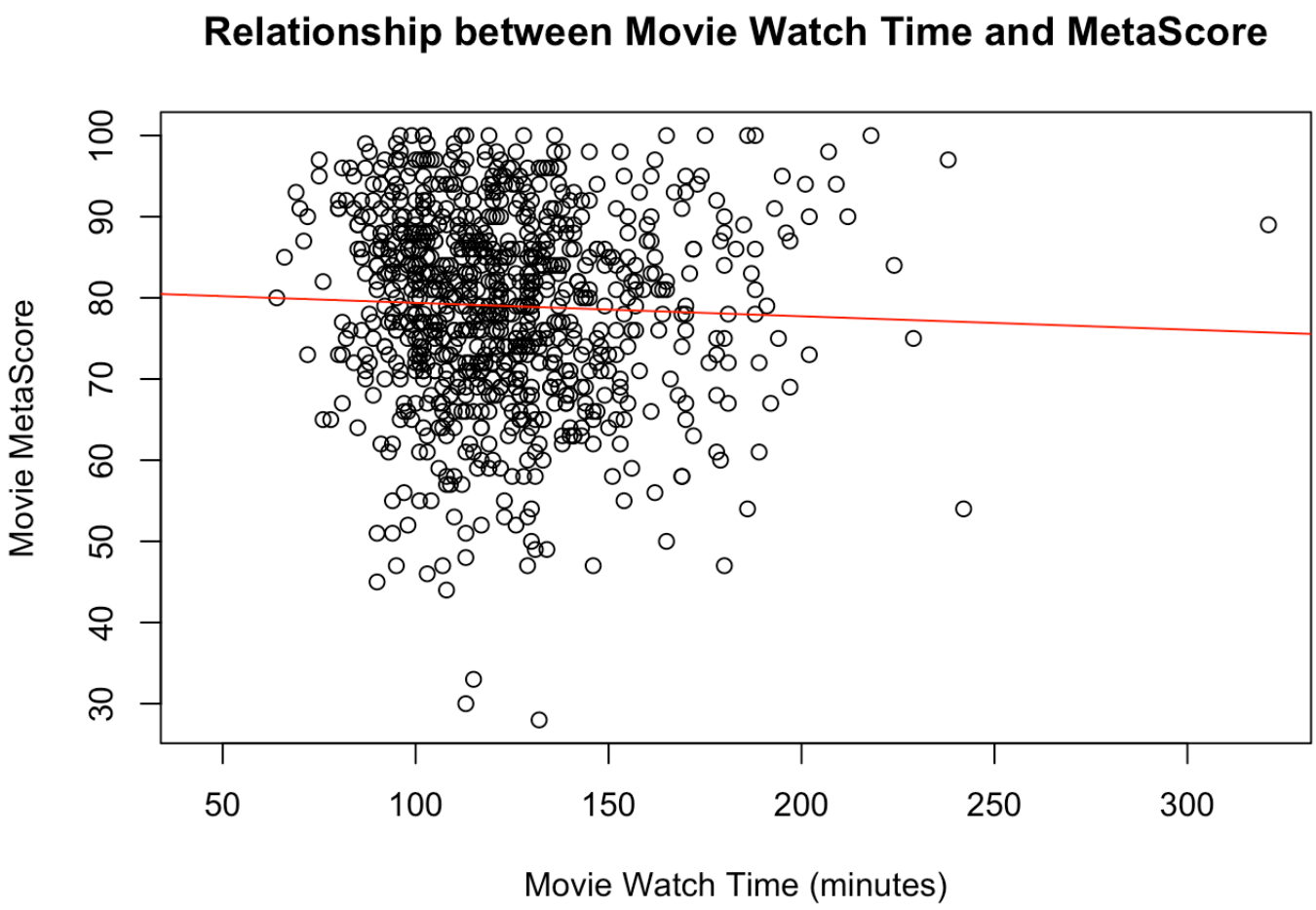
5. Create a linear model for two of the numerical variables

Plotting watch time as x and metascore as y to determine if there is any correlation between the length of the movie and its Metascore.

```
df <- data.frame(x = c(data$Watch.Time), y=c(data$Metascore.of.movie))
model <- lm(y ~ x, data = df)
summary(model)
```

```
##
## Call:
## lm(formula = y ~ x, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -50.868  -7.597   1.109   8.601  22.557
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 81.05585    1.86239  43.523   <2e-16 ***
## x           -0.01657    0.01473  -1.125    0.261
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 11.97 on 843 degrees of freedom
##   (155 observations deleted due to missingness)
## Multiple R-squared:  0.0015, Adjusted R-squared:  0.0003157
## F-statistic: 1.266 on 1 and 843 DF,  p-value: 0.2607
```

```
plot(main = "Relationship between Movie Watch Time and MetaScore",
     df$x,
     df$y,
     xlab = "Movie Watch Time (minutes)",
     ylab = "Movie MetaScore")
abline(model, col = "red")
```


Relationship between Movie Watch Time and MetaScore
Movie MetaScore
30
40
50
60
70
80
90
100
50
100
150
200
250
300
Movie Watch Time (minutes)