Popular Movies Analysis Report

Group Member:

Mao Li: ml1911

Ying Liu: yl1206

Chaoying Luo: cl1419

Ruitong Liu: rl1066

Class: ANLY 511

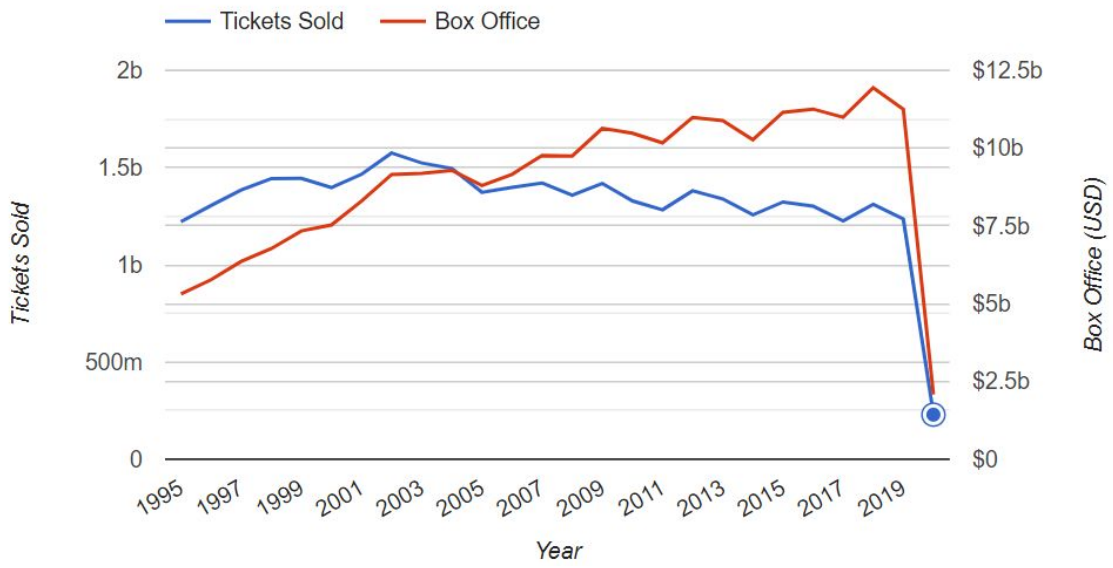
Professor: Purna Gamage

Date: 12/05/2020

Contents

1. Introduction

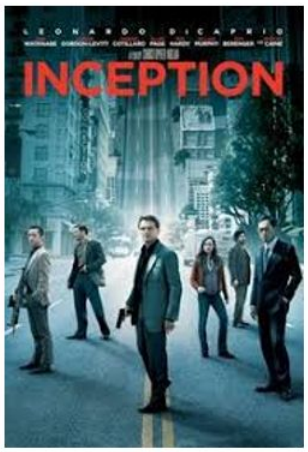
With the continuous improvement of people's living standards, movies play an increasingly important role in people's daily entertainment. Watching movies can be a great date night activity and it could help people release stress even providing encouragement. Nobody doesn't love movies.



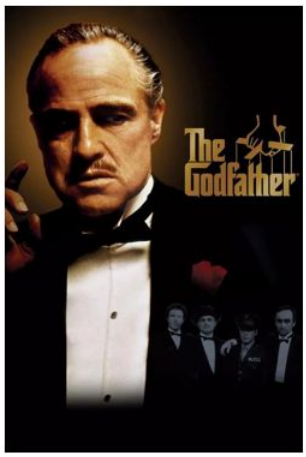
Resource: [The Numbers]

From the plot provided by The Numbers, people spent more than 5 billion dollars which is an enormous figure each year from 1995 to 2019 in cinemas. For this year, influenced by the Covid-19, the box office revenue and tickets sold decreased a lot. Even though the enthusiasm about movies does not abate, more and more people choose to review the former movies.

The Dark Knight (2008)



Inception (2010)



The Godfather (1972)

Since movies become an indispensable part of people's lives, the standard to judge movies has been very high. A good movie could not only bring people emotional resonance but also inspire people to think. There are many excellent movies that have harvested both profit and reputation through movie history like "The dark knight", "Inception" and "The Godfather".

Thus this report is determined to explore the different attributes of popular movies. The report will focus on ten data science questions to analyze. The data science questions are shown below:

- Is the distribution of scores normally distributed?
- What is the difference in means of scores for the movies directed by Woody Allen and Alfred Hitchcock?
- What is the difference in means of revenue for the movies directed by Woody Allen and Alfred Clint Eastwood?
- Is there a relationship between the popularity and runtime of movies?
- Is there a relationship between the popularity and scores of movies?
- Is there a relationship between the score and the revenue of movies?
- Are IMDb users less dispersed and do they have milder assessments than Metascore users?
- What is the correlation between the numeric variables of the dataset?
- Do old movies have better reputations than new movies?
- How to use Naive Bayes to predict The relationship between movie scores and revenue?

2. Analysis

2.1 Datasets

The datasets include basic information like movie titles, year, score, metascore, movie genres, votes, runtime, revenue and description about 10000 of most popular movies. The score and metascore represent the different ratings from users of IMDB and Metascores respectively which could directly reflect the movies' reputation. The revenue could be used to measure the profitability of movies.

2.2 Data Cleaning and Preparation

In the dataset, 9 out of 11 variables are complete - they don't have the NA value. The "Metascore" variables have 3219 missing values, which constitutes over 32% of all observations. Similar is in "Revenue" variables, where 2527 values are empty. It means that every fourth movie does not contain information about revenues. In the cleaning process, all NA values are removed. A "VoteMln" variable was created to increase the readability of charts by showing this unit in millions.

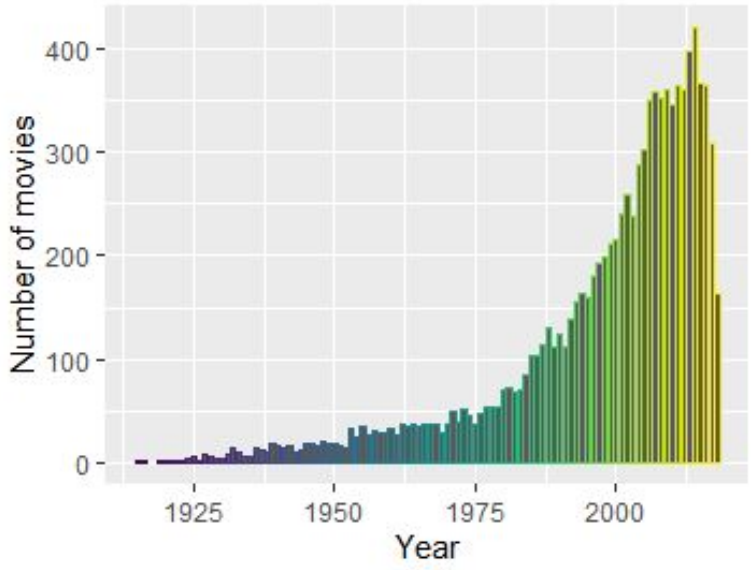
Then, the "Runtime" variables and the "Vote" variables are discretized for Chi-squared test analysis. Specifically, two new columns, "runtime_cat" and "vote_cat", are created. The "runtime_cat" saves the results of categorizing the "Runtime" into three classes, namely "Short", "Medium", and "Long". The "vote_cat" saves the results of categorizing the "Vote" into four levels, which are "Very Popular", "Moderately Popular", "Slightly Popular", and "Not Popular". The data after discretization is partly shown as below.

Rank	Title	Year	Score	Metascore	Genre	Vote	Director	Runtime	Revenue	Description	runtime_cat	vote_cat
1	The Shawsh	1994	9.3	80	Drama	2011509	Frank Darab	142	28.34	Two impriso	Long	Very Popular
2	The Dark Kn	2008	9	84	Action, Crim	1980200	Christopher	152	534.86	When the m	Long	Very Popular
3	Inception	2010	8.8	74	Action, Adve	1760209	Christopher	148	292.58	A thief who	Long	Very Popular
4	Fight Club	1999	8.8	66	Drama	1609459	David Finche	139	37.03	An insomnia	Long	Very Popular
5	Pulp Fiction	1994	8.9	94	Crime, Dram	1570194	Quentin Tar	154	107.93	The lives of	Long	Very Popular
6	Forrest Gum	1994	8.8	82	Drama, Rom	1532024	Robert Zem	142	330.25	The presider	Long	Very Popular
7	The Lord of	2001	8.8	92	Adventure, [	1448561	Peter Jackso	178	315.54	A meek Hob	Long	Very Popular
8	The Matrix	1999	8.7	73	Action, Sci-F	1443130	Lana Wacho	136	171.48	A computer	Long	Very Popular
9	The Lord of	2003	8.9	94	Action, Adve	1431887	Peter Jackso	201	377.85	Gandalf and	Long	Very Popular
11	The Dark Kn	2012	8.4	78	Action, Thril	1338413	Christopher	164	448.14	Eight years a	Long	Very Popular
12	The Lord of	2002	8.7	87	Adventure, [	1294351	Peter Jackso	179	342.55	While Frodo	Long	Very Popular
13	Se7en	1995	8.6	65	Crime, Dram	1229411	David Finche	127	100.13	Two detectiv	Long	Very Popular
14	Interstellar	2014	8.6	74	Adventure, [	1223159	Christopher	169	188.02	A team of ex	Long	Very Popular
15	Gladiator	2000	8.5	67	Action, Adve	1162557	Ridley Scott	155	187.71	A former Ro	Long	Very Popular
16	Django Uncl	2012	8.4	81	Drama, Wes	1159647	Quentin Tar	165	162.81	With the hel	Long	Very Popular
17	Batman Beg	2005	8.3	70	Action, Adve	1150820	Christopher	140	206.85	After trainin	Long	Very Popular
18	Avengers As	2012	8.1	69	Action, Adve	1130818	Joss Whedo	143	623.36	Earth's migh	Long	Very Popular
19	Star Wars: E	1977	8.6	90	Action, Adve	1080484	George Luca	121	322.74	Luke Skywal	Long	Very Popular
20	The Silence	1991	8.6	85	Crime, Dram	1079027	Jonathan De	118	130.74	A young FBI	Medium	Very Popular

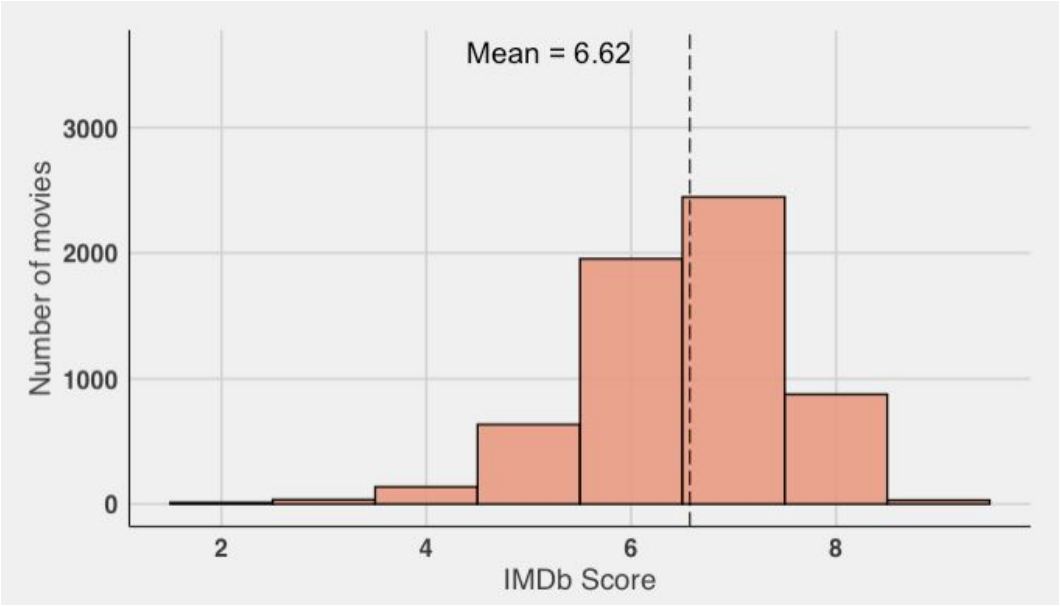
Besides, the movies in the dataset could be divided into two categories based on the year attribute. The movies before 2000 will be classified as old movies and the movies after 2000 will be zoned as new movies. As a result, the new movies have 6060 objects and the old movies have 3940 objects.

In the prediction of the relationship between movie scores and movie box office revenue, Naive Bayes is used. The scores are too scattered, so they were separated into three classes — the score which is higher than 8.5 points (total is 10) is a “high_score”, the score which is higher than 7 points but lower than 8.5 points is a “med_score” and the score which is lower than 7 points is a “low_score”. Then the original data was divided into five parts, four parts for train data and one part for the test data.

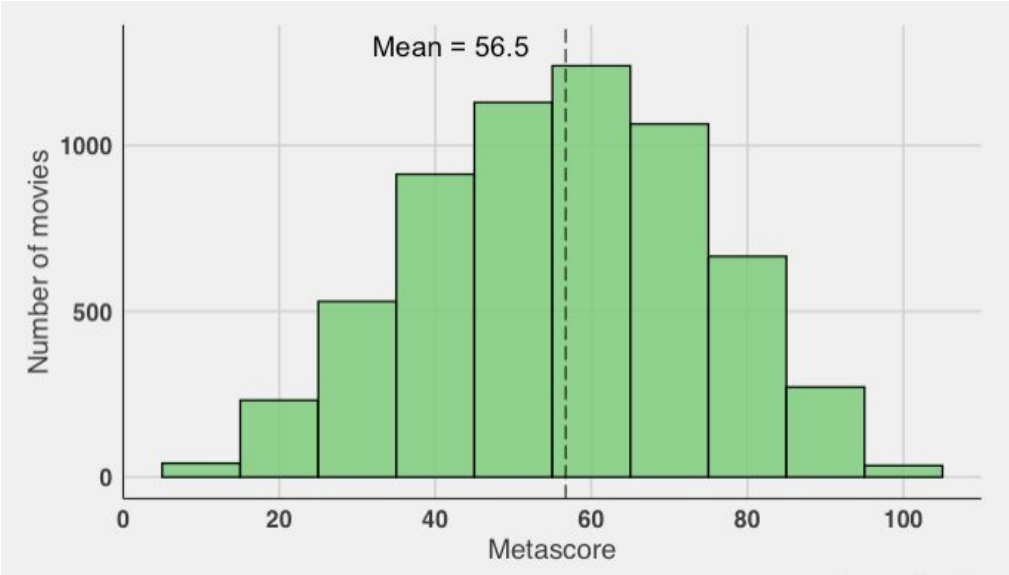
2.3 Data Exploring



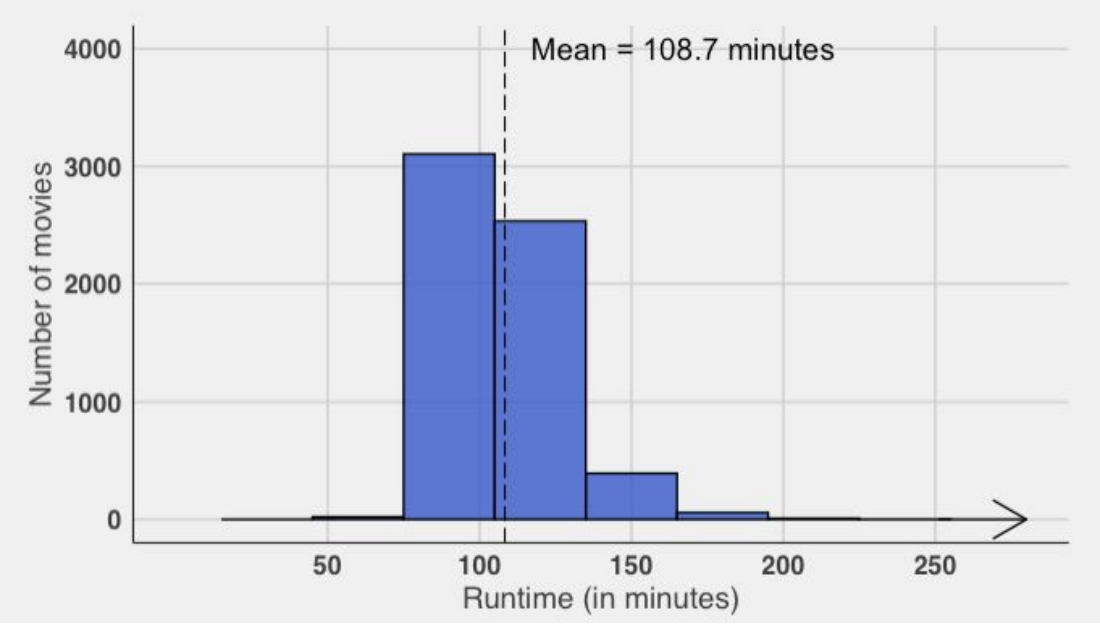
In order to get the distribution of movies by year in our dataset, the bar plot is drawn. From this plot, it is not difficult to find that the number of movies in recent years is prominently higher than in the former years. Since our dataset is about the 10000 of the most voted movies, it is reasonable to deduce that in general, the movies in recent years get more votes and maybe more popular than old movies.



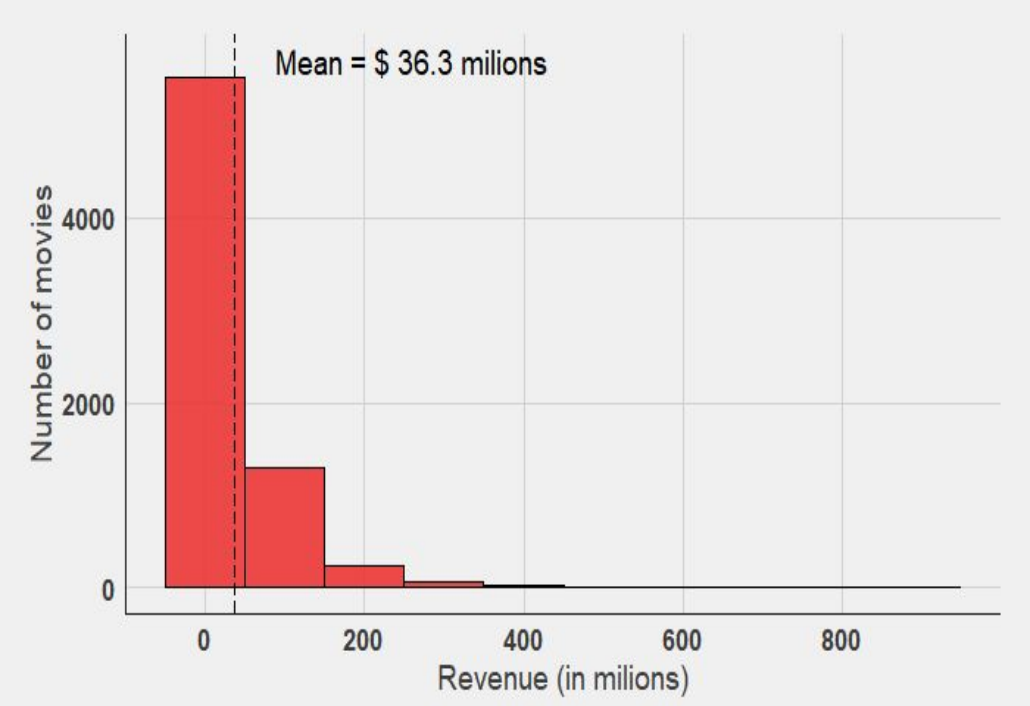
From the plot, 6.62 is the average rating issued by IMDb users. The distribution is roughly symmetrical. Most of the user's ratings are in the range of 6 and 8.



According to the plot, the average rating for movies from the database by Metascore users is 56.5 points. The distribution is also symmetrical, but extreme values are relatively often here.



Movies in the database take an average of 108.7 minutes. The distribution is characterized by a small right-side asymmetry. Most of the films last between 75 and 135 minutes.



The revenue from movies is characterized by a strong right-side asymmetry. Most movies have little revenue, only a few films made a big profit. The average revenue from the movie was over $ 36 million.

2.4 Statistical Methods

Anderson Darling Test will be used to check whether the distribution of score is normally distributed. The Anderson Darling Test is a measure of how well the data fits a specified distribution. It is commonly used as a test for normality. In this instance, the hypotheses for the AD-test are: H_0: The data comes from a normal distribution. H_1: The data does not come from a normal distribution. Then, calculate the AD Statistic and find the statistic's p-value. If p-value is less than the chosen alpha level, the null hypothesis can be rejected. Otherwise, the alternative hypothesis will be rejected. Moreover, the density plot or Q-Q plot will be used to explore the data distribution.

Two Sample Bootstrap will be used to find the difference of the means of scores for the movies directed by Woody Allen and Alfred Hitchcock. Two Sample Bootstrap solves the problem of comparing independent samples from two populations by mimicking how the data were obtained. In this instance, it will first draw a resample from scores of movies directed by Woody Allen with replacement and a separate resample from scores of movies directed by Alfred Hitchcock. Then, compute the difference between these two sample means. Finally, repeat this resampling process 10000 times and construct the bootstrap distribution of the difference between sample means. Similarly, two sample bootstrap will also be used to find the difference of the means of revenue for the movies directed by Woody Allen and Clint Eastwood.

A Chi squared test for two-way tables will be used to figure out the relationship between the popularity and runtime of movies. Chi-squared Test for Independence is a test for the relationship between two categorical variables. Therefore, in this instance, data discretization is necessary. Specifically, categorize the column "Runtime" into three levels, namely "Short", "Medium", and "Long", and the column "Vote" into four levels, which are "Very Popular", "Moderately Popular", "Slightly Popular", and "Not Popular". Then, make a two-way table of these two categorical variables to find their connection. Next, calculate the expected count for each cell and X squared. Finally, find the statistic's p-value to make a conclusion.

Multiple Linear Regression will be performed to find whether there is a relationship between the popularity and scores of movies. Linear regression as an approach for supervised learning, is a useful tool for predicting a quantitative response. In this example, the Multiple Linear Regression model will take the vote for movies as the response and other numeric variables as the predictors. Which predictors appear to have a statistically significant relationship to the response will be explored. Furthermore, the plot of the response on the predictor Score will be made. Finally, p-value and R-square will be utilized to find the fitness of the regression model. Similarly, multiple linear regression will also be applied to find the relationship between the movie scores (IMDb score and Metascore) and the movie revenues.

Hypothesis Testing will also be used to see whether IMDb users are less dispersed and do they have milder assessments than Metascore users. User feedback is one of the most important things on the IMDb site and Metacritic site. The scores from which the two samples are drawn are normally distributed, and the two populations are independent of each other. In order to compare two sample differences, a 95% confidence interval is calculated.

Student's t-test will be used to compare the sample mean of the score of old movies and new movies. Since it is impossible to get the population mean of these two categories, the student's t-test will be more appropriate than the z-test method. Besides, the significance level is set to 0.05.

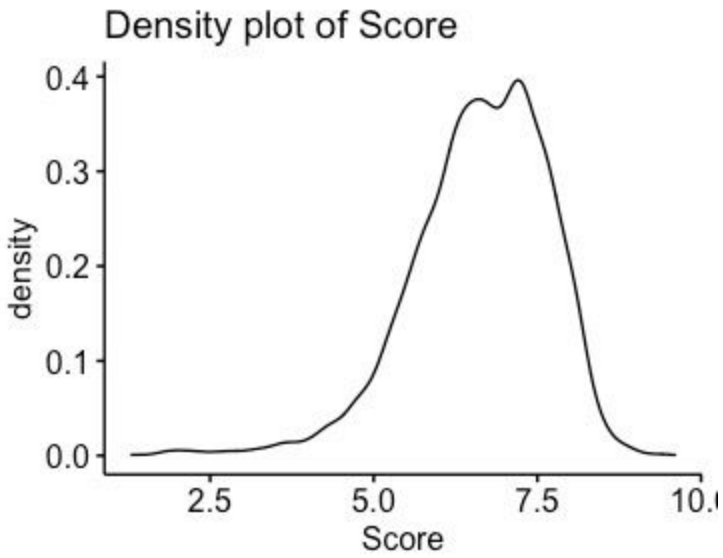
Naive Bayes classifiers are built on Bayesian classification methods based on applying Bayes' theorem with strong (naïve) independence assumptions between the features. They are the models that assign class labels to problem instances, represented as vectors of feature values, where the class labels are drawn from some finite set. These rely on Bayes's theorem, which is an equation describing the relationship of conditional probabilities of statistical quantities. In Bayesian classification, we are interested in finding the probability of a label given some observed features. What it needs to do with Naive Bayes Classifier is to find out which class has a bigger probability for the new given conditions.

3. Results

3.1 Anderson Darling Test

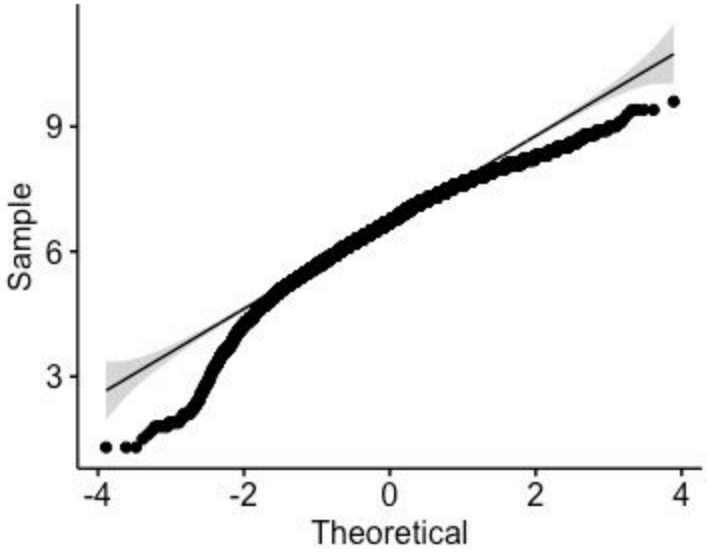
Question: Is the distribution of scores normally distributed?

First, make a density plot to explore the distribution of scores.



The density plot shows that the distribution of scores is not bell shaped, so it is not a normal distribution.

Then, make a Q-Q plot to explore the distribution of scores.



The Q-Q plot draws the correlation between a sample of scores and the normal distribution. A 45-degree reference line is also plotted. According to the picture, both the bottom end and the upper end of the Q-Q plot deviate from the straight line so the distribution of scores is not normally distributed.

Finally, perform the AD-test.

```
        Anderson-Darling normality test

data:  my_data$Score
A = 58.572, p-value < 2.2e-16
```

The null hypothesis is that the distribution of scores is the same as the normal distribution. The alternative hypothesis is that the distribution of scores is significantly different from normal distribution. From the output, the p-value is less than 0.05, so the null hypothesis is rejected, which means that the distribution of scores is non-normal.

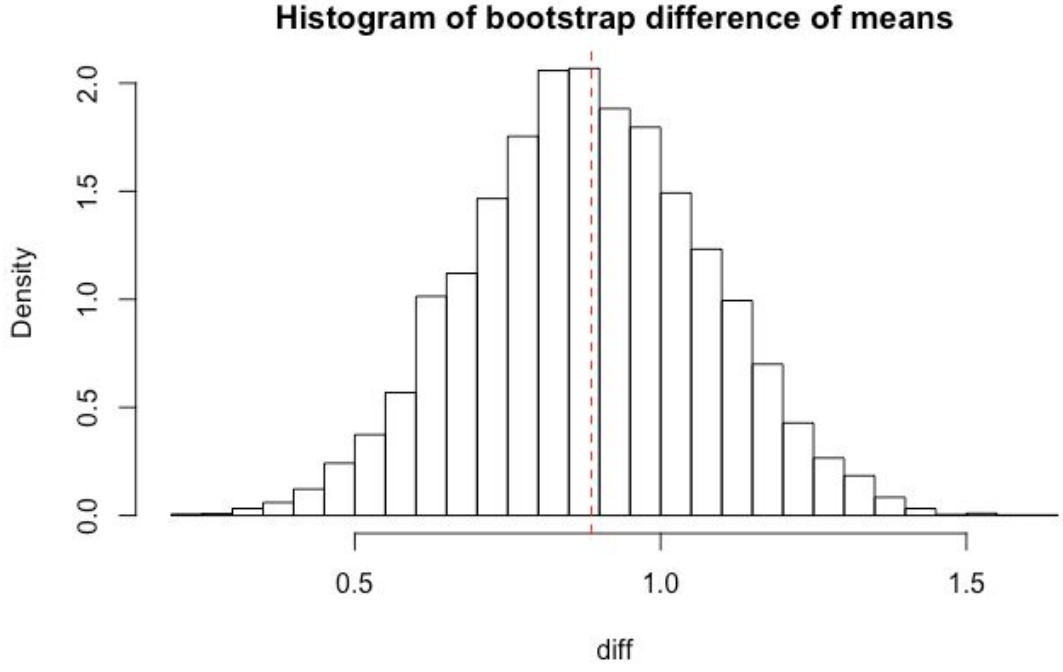
3.2 Two Sample Bootstrap

Question 1: What is the difference of means of scores for the movies directed by Woody Allen and Alfred Hitchcock?

First, make side by side boxplots of scores of movies directed by Woody Allen and Alfred Hitchcock respectively.

The boxplots of scores of movies directed by Woody Allen and Alfred Hitchcock show that the scores of movies directed by Alfred Hitchcock is generally higher than scores of movies directed by Woody Allen.

Then, bootstrap the difference of means, repeat the resampling process 10000 times, and plot the result as a histogram. Add a red line representing the true difference of means.



The bootstrap distribution of difference of means looks quite normal, with some skewness.

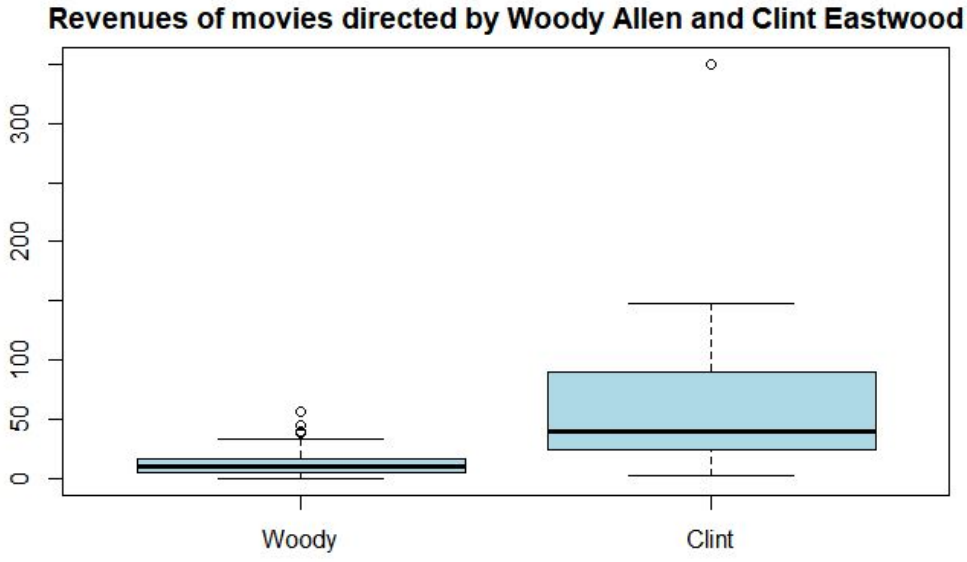
Finally, find the 95% bootstrap percentile interval for the difference of means.

2.50%	97.50%
0.5066667	1.2666667

The 2.5% and 97.5% points of the bootstrap distribution give the interval (0.51, 1.27), so the conclusion is with 95% confidence, the difference of means between scores of movies directed by Alfred Hitchcock and Woody Allen is between 0.51 and 1.27.

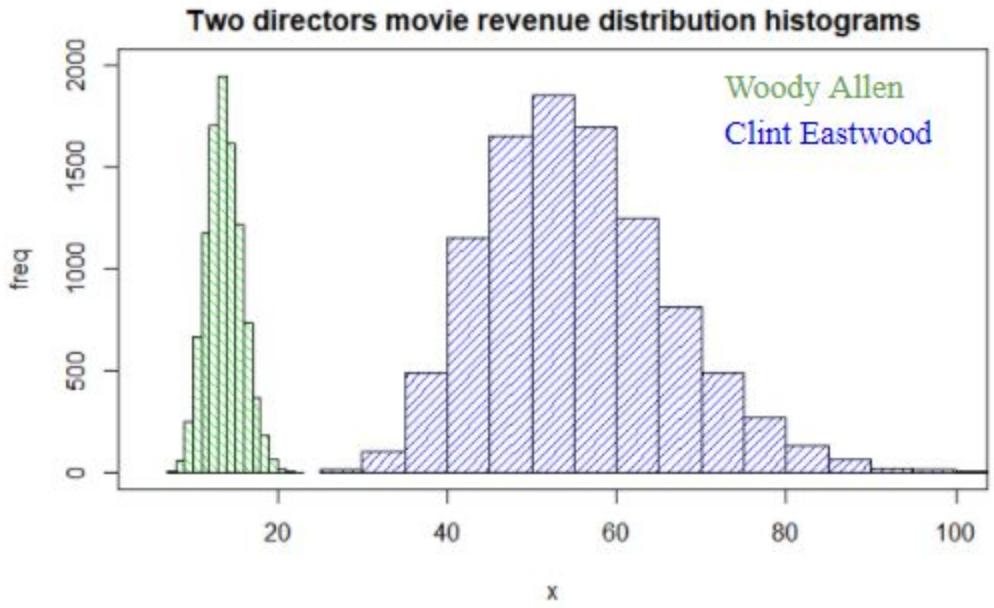
Question 2: What is the difference of means of revenue for the movies directed by Woody Allen and Alfred Clint Eastwood?

First, make side by side boxplots of revenue of movies directed by Woody Allen and Clint Eastwood respectively.



The boxplots of scores of movies directed by Woody Allen and Clint Eastwood show that the revenue of movies directed by Clint Eastwood is generally higher than revenue of movies directed by Woody Allen.

Then, simulate the bootstrap 10,000 times and display the bootstrap distribution of the sample revenue means.



From the histograms, the mean of Woody Allen and Clint Eastwood movie revenue has a normal distribution. Clint Eastwood has a higher box office.

3.3 Chi-squared test

Question: Is there a relationship between the popularity and runtime of movies?

First, make a two-way table of the two categorical variables “runtime_cat” and “vote_cat” to find their connection.

	Long	Medium	Short
Moderately Popular	166	144	8
Not Popular	872	3559	779
Slightly Popular	207	317	47
Very Popular	25	1	0

The table above shows that the length of most very popular movies and moderately popular movies are long. 317 slightly popular movies have a medium runtime. Most of the not welcomed movies also have a medium runtime. Therefore, a relationship between the popularity and runtime of movies can be assumed.

Then, perform the Chi squared test.

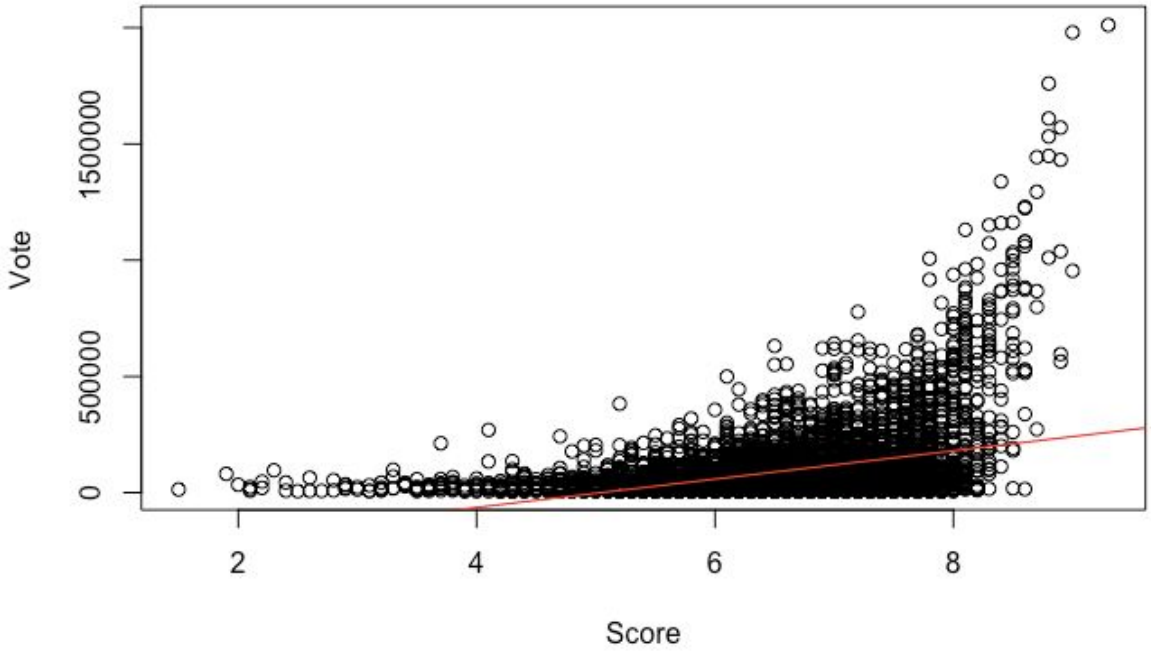
```
        Pearson's Chi-squared test

data:  a
X-squared = 430.72, df = 6, p-value < 2.2e-16
```

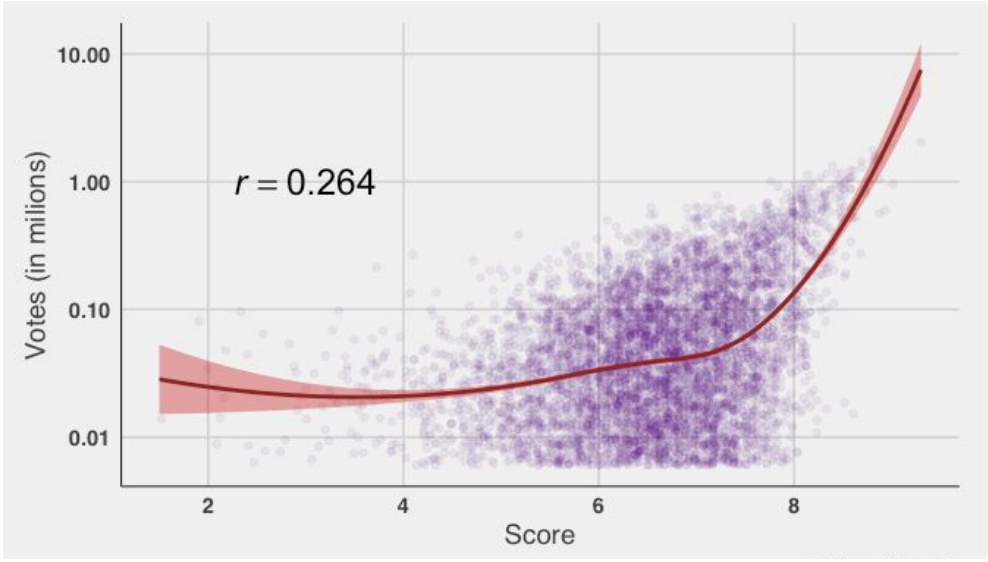
The null hypothesis is that the popularity and runtime of movies are independent. The popularity does not vary by runtime of movies. The alternative hypothesis is that the popularity and runtime of movies are dependent. The popularity does vary by runtime of movies. Since p-value is less than 0.05, the null hypothesis is rejected. Thus, there is enough evidence to conclude that there is a significant relationship between the popularity and runtime of movies. They are dependent.

3.4 Multiple Linear Regression

Question 1: Is there a relationship between the popularity and scores of movies?

First, make scatter plots of the response Vote on the predictor "Score" with a linear regression line to explore their relationship.





The plots show that there is a small positive correlation between the popularity of movies and movie scores. Most movies have less than 500 thousand votes so most of the observations are at the bottom of the plots. The red line shows that very good rated movies are much more popular than others.

Then, perform the Multiple Linear Regression with the vote for movies as the response and other numeric variables as the predictors.

```
Call:
lm(formula = Vote ~ Year + Score + Metascore + Runtime + Revenue,
    data = my_data)

Residuals:
    Min      1Q  Median      3Q     Max
-777413  -49647  -15262   26663 1785591

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.782e+06  2.407e+05  -7.400 1.54e-13 ***
Year         7.136e+02  1.192e+02   5.985 2.29e-09 ***
Score        5.396e+04  2.211e+03  24.399  < 2e-16 ***
Metascore   -5.144e+02  1.135e+02  -4.533 5.92e-06 ***
Runtime      6.173e+02  8.043e+01   7.675 1.92e-14 ***
Revenue      1.278e+03  2.153e+01  59.354  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 107600 on 6119 degrees of freedom
Multiple R-squared:  0.4877,    Adjusted R-squared:  0.4873
F-statistic:  1165 on 5 and 6119 DF,  p-value: < 2.2e-16
```

The null hypothesis is that the coefficient of each variable equals to 0. In other words, there is no relationship between the vote and each variable. The alternative hypothesis is that at least one coefficient of each variable is non-zero. In other words, there is at least one relationship between the vote and a variable. Since p-value is less than 0.05, the null hypothesis is rejected. Thus, there is enough evidence to conclude that vote has at least one relationship with the variables above.

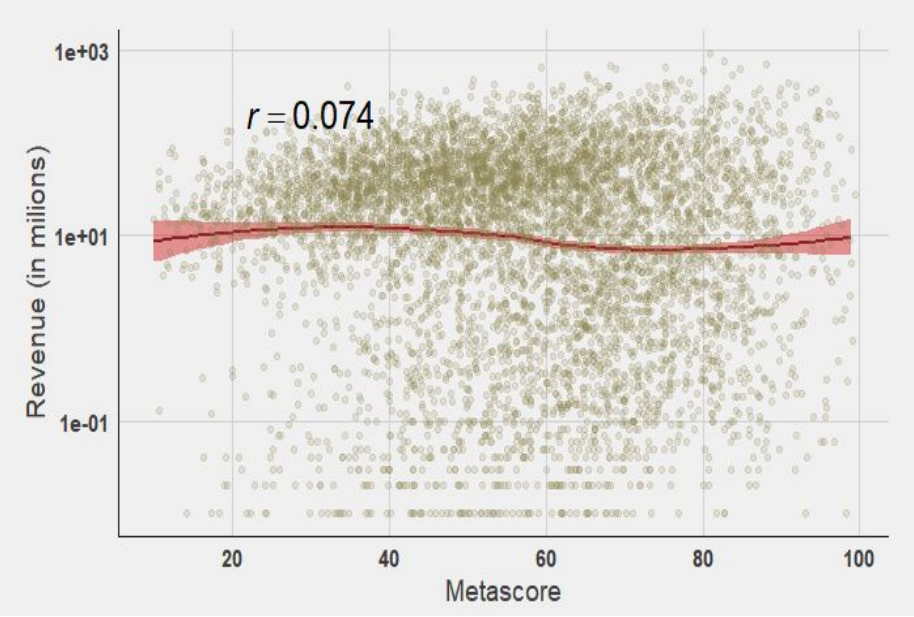
From the output, all the predictors have a statistically significant relationship to votes for movies. Predictors, like Year, Score, Runtime, and Revenue, have a positive relationship with vote, while Metascore has a negative relationship with vote. The coefficient on Score in terms of vote is 53960, which means that an additional 1 point increase in scores of movies can lead to an increase in votes by approximately 53960.

Question 2: Is there a relationship between the score and the revenue of movies?

First, perform the multiple linear regression with the revenue for movies as the response and IMDb score and Metascore as the predictors.

```
Call:
lm(formula = Revenue ~ IMDbscore + Metascore, data = movie_score_revenue)

Residuals:
   Min     1Q Median     3Q    Max
-62.01 -38.68 -19.44  12.35 883.41

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) -20.05876    6.24385  -3.213  0.00132 **
IMDbscore    10.78573    1.28758   8.377  < 2e-16 ***
Metascore    -0.16015    0.06908  -2.318  0.02047 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 65.31 on 6069 degrees of freedom
Multiple R-squared:  0.01655,   Adjusted R-squared:  0.01623
F-statistic: 51.08 on 2 and 6069 DF,  p-value: < 2.2e-16
```

The coefficient of Metascore is negative. It’s unnormal. Take a look at this Scatter plot.



From the plot, there is no correlation between Metascore and the revenue from the movie. The scores are definitely more dispersed than revenue. It is impossible to relate these two variables. Then we delete the Metascore variable and adjust the linear regression model.

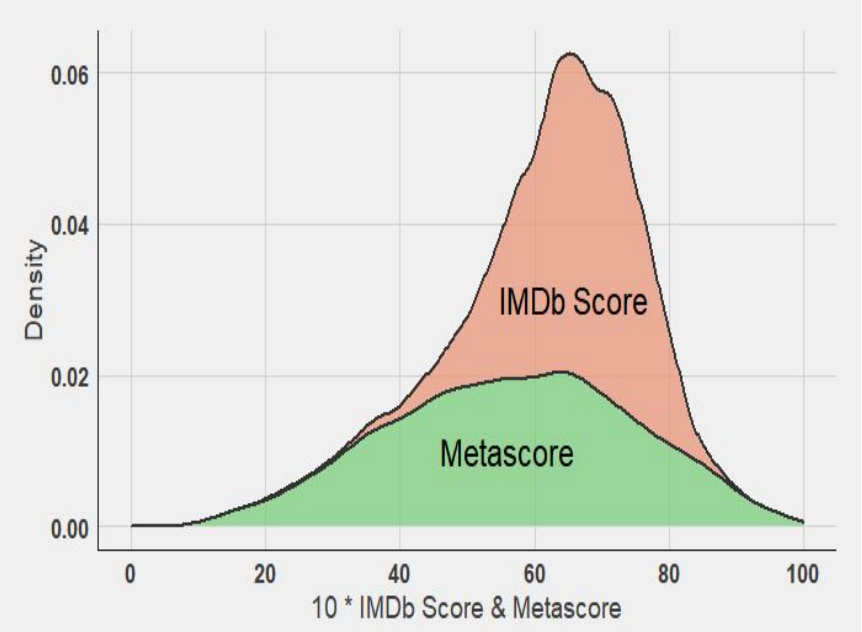
```
Call:
lm(formula = Revenue ~ IMDbscore, data = movie_score_revenue)

Residuals:
   Min     1Q Median     3Q    Max
-59.76 -39.23 -18.83  12.61 882.65

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -14.7287     5.8072  -2.536   0.0112 *
IMDbscore     8.5926     0.8737   9.834   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 65.33 on 6070 degrees of freedom
Multiple R-squared:  0.01568,   Adjusted R-squared:  0.01552
F-statistic: 96.71 on 1 and 6070 DF,  p-value: < 2.2e-16
```

There is a positive correlation between the popularity of movies and the opinion of IMDB Users. Very good rated movies by IMDB users are much more popular than others. The coefficient on IMDB Score in terms of vote is 8.5926, which means that a unit increase in IMDb score results in an 8.6 million increase in revenue, all other variables held constant.

3.5 Hypothesis Testing

Question: Are IMDb users less dispersed and do they have milder assessments than Metascore users?

First, use a density plot to show the scores from which the two score samples are drawn normally distributed.



This clearly shows critics' votes from Metascore are more scattered, and IMDb users are close to average.

Hypothesis Testing will also be used to see whether IMDb users are less dispersed. For the difference of mean, a 95% Confidence Interval was calculated.

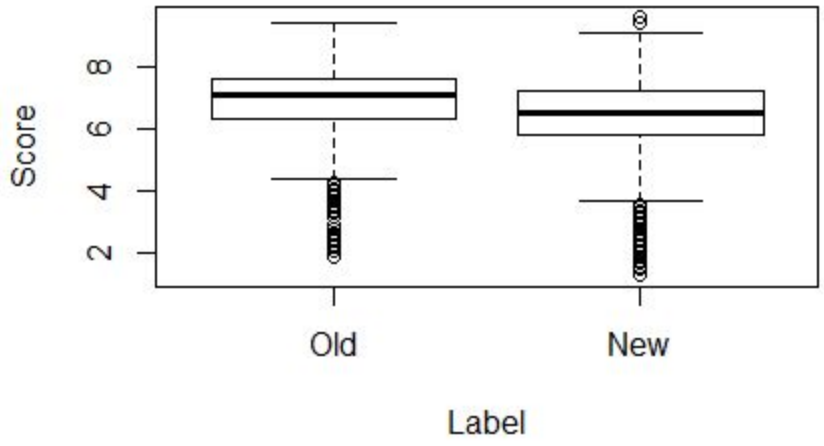
2.50%	97.50%
-50.40254	-49.55779

CI does not contain 0. True difference is likely to be negative. That means we reject the null hypothesis. Thus, there are some differences between the ratings of IMDB and Metascore users.

3.6 Student's t-test

Question: Do old movies have better reputations than new movies?

Firstly, after dividing the dataset into two categories based on year, a box plot is made to visually show the distribution of IMDb scores and skewness of movies in different categories through displaying the data quartiles and averages.



From the boxplot, it is clear that there are many outliers in both two categories, but in general, old movies have better scores than new movies. The presence of outliers under the lower whisker is not hard to interpret, because many movies that discord with the taste of the masses tend to receive more negative comments and lower ratings. The outliers over the upper whisker signifies that there are some extraordinary movies made after 2000.

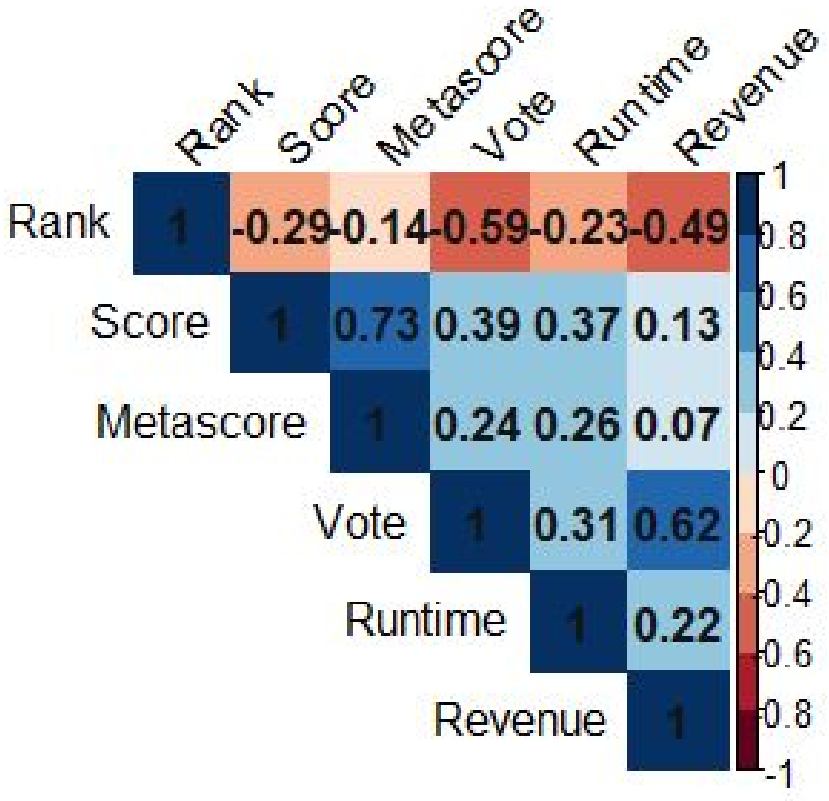
```
        Welch Two Sample t-test

data:  oldmovies and newmovies
t = 21.688, df = 8601.4, p-value < 2.2e-16
alternative hypothesis: true difference in means is
greater than 0
95 percent confidence interval:
 0.4147423       Inf
sample estimates:
mean of x mean of y
 6.899492  6.450710
```

The student's t-test is utilized to compare the difference in means of these two categories. The null hypothesis is that the score of old movies is equal to the score of new movies. Besides, the alternative hypothesis is that the score of old movies is greater than new movies. Since the p-value is less than 5%, at the 5% significance level, we have enough evidence to reject the null hypothesis and accept the alternative hypothesis.

3.7 Correlation Analysis

Question: What is the correlation between the variables of the dataset?

In order to visualize the correlation between different variables in the dataset, a coefficient heatmap is drawn as below:

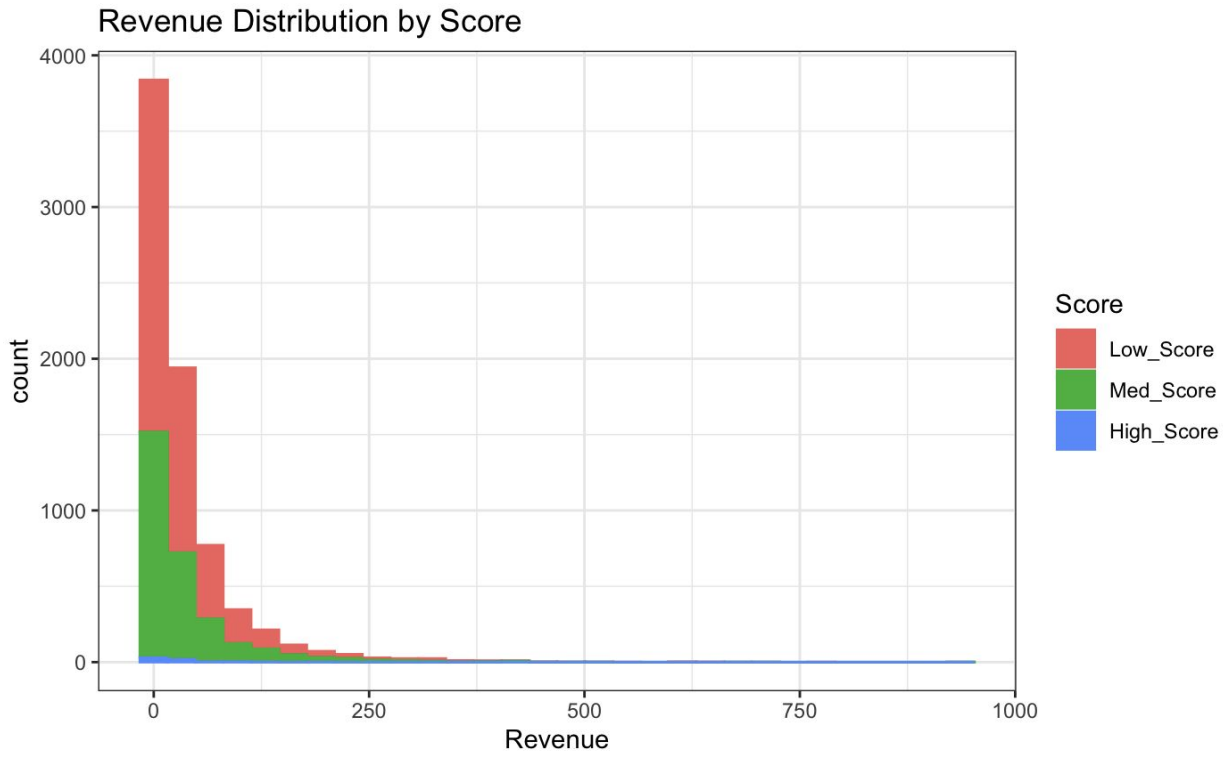


From the heatmap, it is not difficult to find that the score and the metascore have the strongest positive correlation, besides, the rank and the vote, the rank and the revenue have the top 2 of the strongest negative correlation.

3.8 Naive Bayes

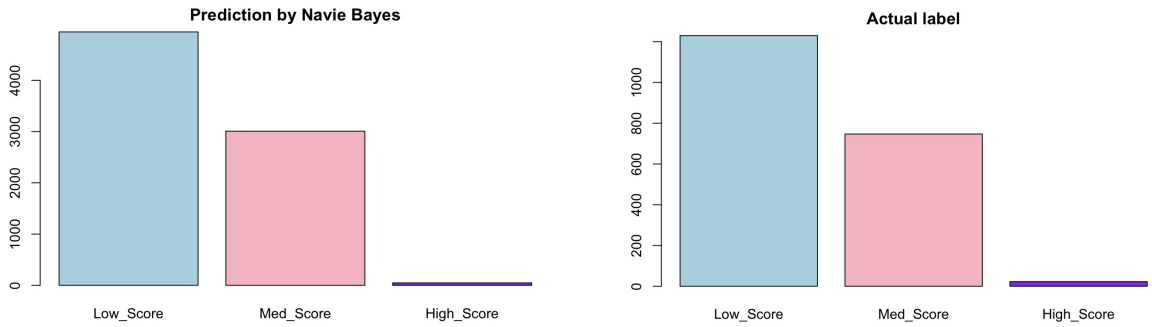
Question: How to classify movies from the different scores and different box office revenue by using Naive Bayes?

First, use the good data that is prepared in data cleaning and preparation (three classes for movie scores) to make a plot for Revenue distribution by Score.



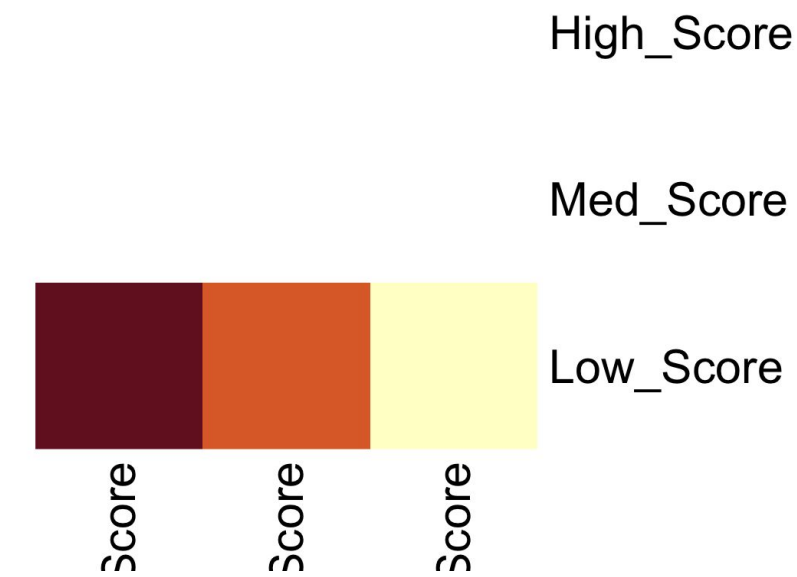
The plot shows that the high scoring movies have higher revenue than low scoring movies.

Then, make a prediction by naive bayes model and plot the result.



Predicted labels are exactly the same as the actual labels.

Finally, make a confusion matrix plot showing the precision of the naive bayes model.



The plot shows high accuracy of the model. The predicted labels are just the actual labels so there are just three squares in the graph.

4. Conclusions

Under the background of film industrialization reform, the influence of film rating is prominent. People gradually develop the habit of scoring after watching a movie, to provide reference for others.

The movies directed by Alfred Hitchcock always have high ratings. One of the reasons may be that he was always finding ways to add new elements to his films. His innovative storytelling successfully attracts most of his audience. Thus, if a director wants to get higher ratings for his movie, innovation style is highly suggested. Clint Eastwood's movies can always get a high revenue. One of his strengths is that he can always make movies within a reasonable budget. So if a director wants to make a profitable movie, he had better master the ability to control the movies' costs first.

From the analysis, we can see that there is a pretty close relationship between movie box office revenue and IMDb score. In other words, in most cases, movies with higher box office revenue have higher scores. One of the reasons for this result is that people prefer to use movie scores as a reference for whether to watch a movie. The higher the scores, the more willing people are to watch the movie. When more people go to the movie, the rating of this good movie may be higher. Thus, this is a positive cycle.

Moreover, old movies are higher rated than new movies. This situation could be interpreted from two potential aspects. Firstly, with the improvement of people's connoisseurship, they are becoming more and more nitpicky in movies, so the scores of movies are becoming lower and lower. However, this aspect can not explain why there are still many high-rated new movies. Another possible reason is that given the striking profitability of the movie industry in recent years, many small-cost movies that are made in order to make money instead of focusing on movies themselves have appeared. The success of this kind of movie encourages more directors to make a profitable movie instead of a good movie. As a result, the rating of recent poorly-made movies are lower than that of old movies made with heart.

The strong positive correlation between the IMDb score and the metascore means that these two ratings from the different platforms tend to have a similar trend. This could be the evidence that the ratings from these two platforms could represent the taste of the masses. The strong negative correlation between the rank and the revenue signifies that the movies that have a higher rank (i.e. the movies that have more votes) tend to have higher revenues. The good movies always have more votes as they will be viewed by more people, in that case, movies can be more profitable in the market.

5. Appendix

Group 8: Popular Movies Analysis

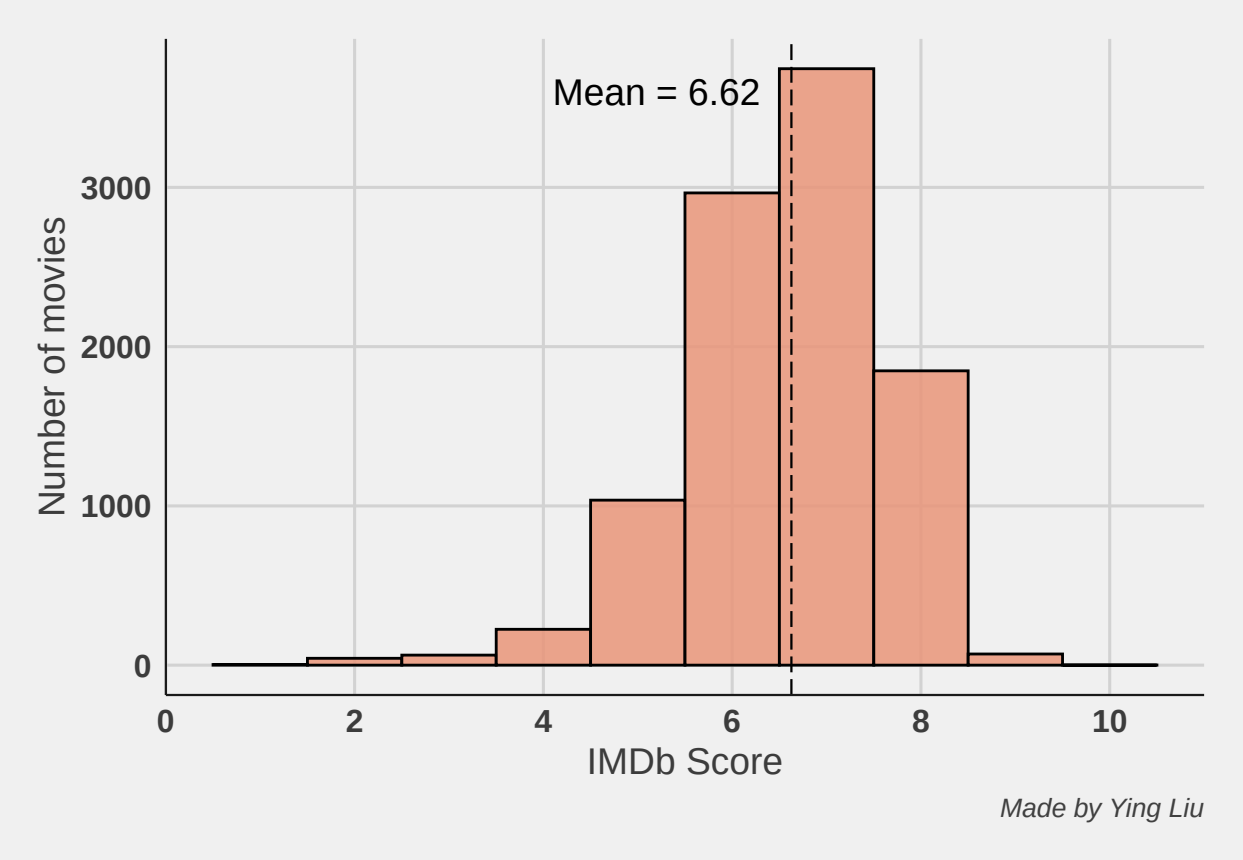
Ying Liu, Chaoying Luo, Mao Li, Ruitong Liu

12/09/2020

EDA Part

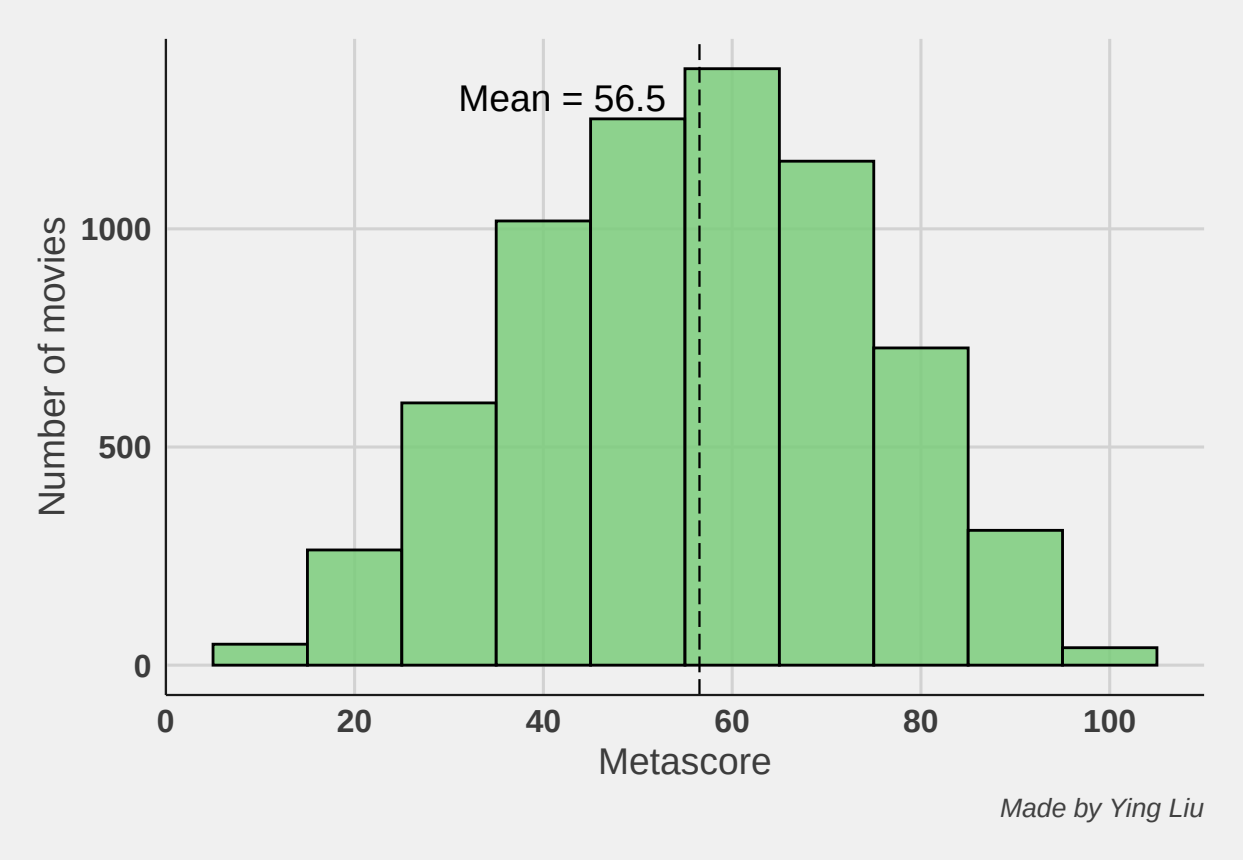
IMDb Score

```
library(ggplot2)
library(ggthemes)

# load data
my_data <- read.csv("movies.csv")

ggplot(my_data, aes(Score))+
  geom_histogram(binwidth = 1, fill = "darksalmon", col = "black", alpha = 0.85)+
  geom_vline(xintercept = mean(my_data$Score), linetype = "longdash", size = 0.4)+
  labs(x = "IMDb Score", y = "Number of movies", caption = "Made by Ying Liu")+
  scale_x_continuous(breaks = c(0,2,4,6,8,10))+
  annotate("text", x = 5.2, y = 3600, label = "Mean = 6.62", size = 5)+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 12, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"), plot.caption = element_text(color = "g
```



Metascore

```
my_data_no_na <- my_data[complete.cases(my_data$Metascore), ]
ggplot(my_data_no_na, aes(Metascore))+
  geom_histogram(binwidth = 10, fill = "palegreen3", col = "black", alpha = 0.85)+
  geom_vline(xintercept = mean(my_data_no_na$Metascore), linetype = "longdash", size = 0.4)+
  labs(x = "Metascore", y = "Number of movies", caption = "Made by Ying Liu")+
  scale_x_continuous(breaks = c(0,20,40,60,80,100))+
  annotate("text", x = 42, y = 1300, label = "Mean = 56.5", size = 5)+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 12, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"), plot.caption = element_text(color = "g
```

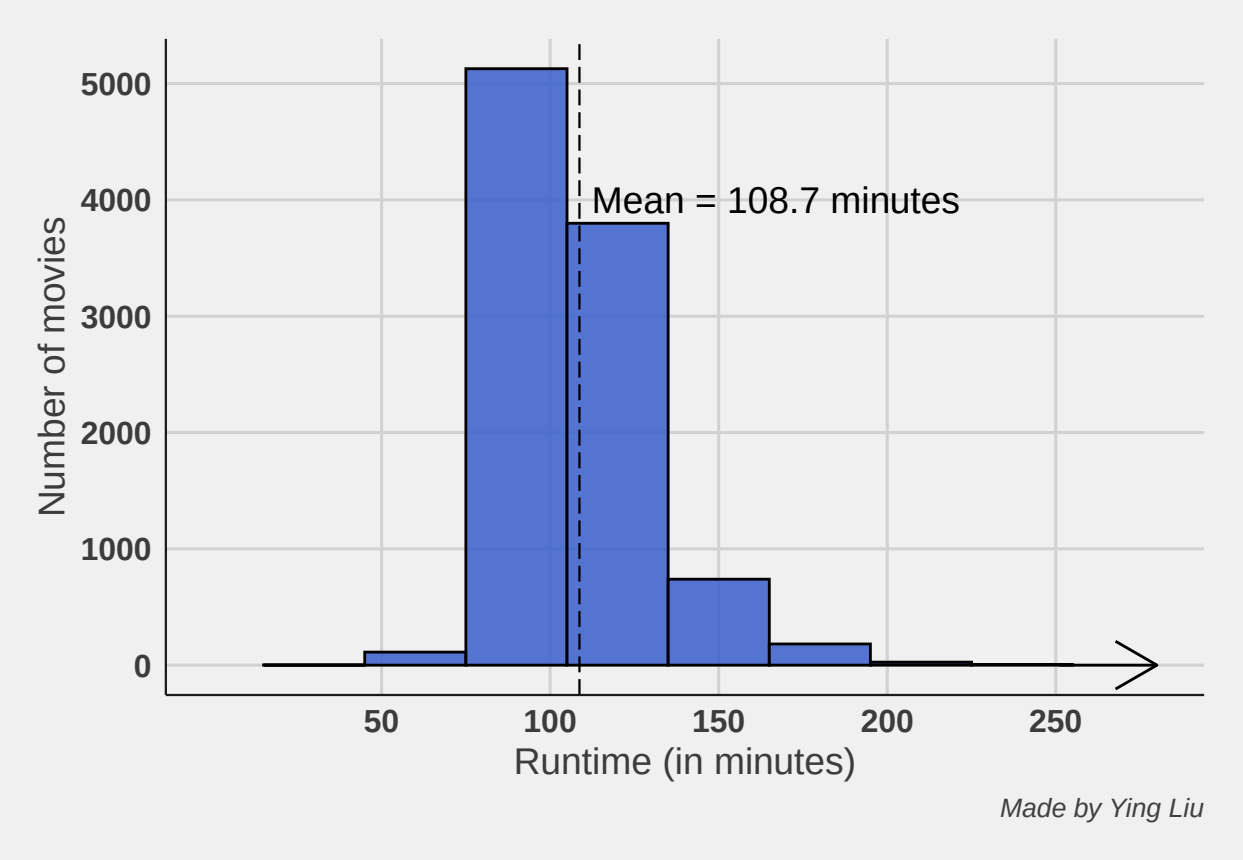


Runtime

```
ggplot(my_data, aes(Runtime))+
  geom_histogram(binwidth = 30, fill = "royalblue3", col = "black", alpha = 0.85)+
  geom_vline(xintercept = mean(my_data$Runtime), linetype = "longdash", size = 0.4)+
  labs(x = "Runtime (in minutes)", y = "Number of movies", caption = "Made by Ying Liu")+
  scale_x_continuous(limits=c(0,280), breaks = c(50,100,150,200,250))+
  annotate("text", x = 167, y = 4000, label = "Mean = 108.7 minutes", size = 5)+
  annotate("segment", x = 251, xend = 280, y = 0, yend = 0, size = 0.5, arrow = arrow())+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 12, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"), plot.caption = element_text(color = "g
```

```
## Warning: Removed 5 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 2 rows containing missing values (geom_bar).
```


5000
4000
3000
2000
1000
0
Number of movies
Mean = 108.7 minutes
50
100
150
200
250
Runtime (in minutes)
Made by Ying Liu

```r
library("dplyr")
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library("ggplot2")
library("viridis")
```

```
## Loading required package: viridisLite
```

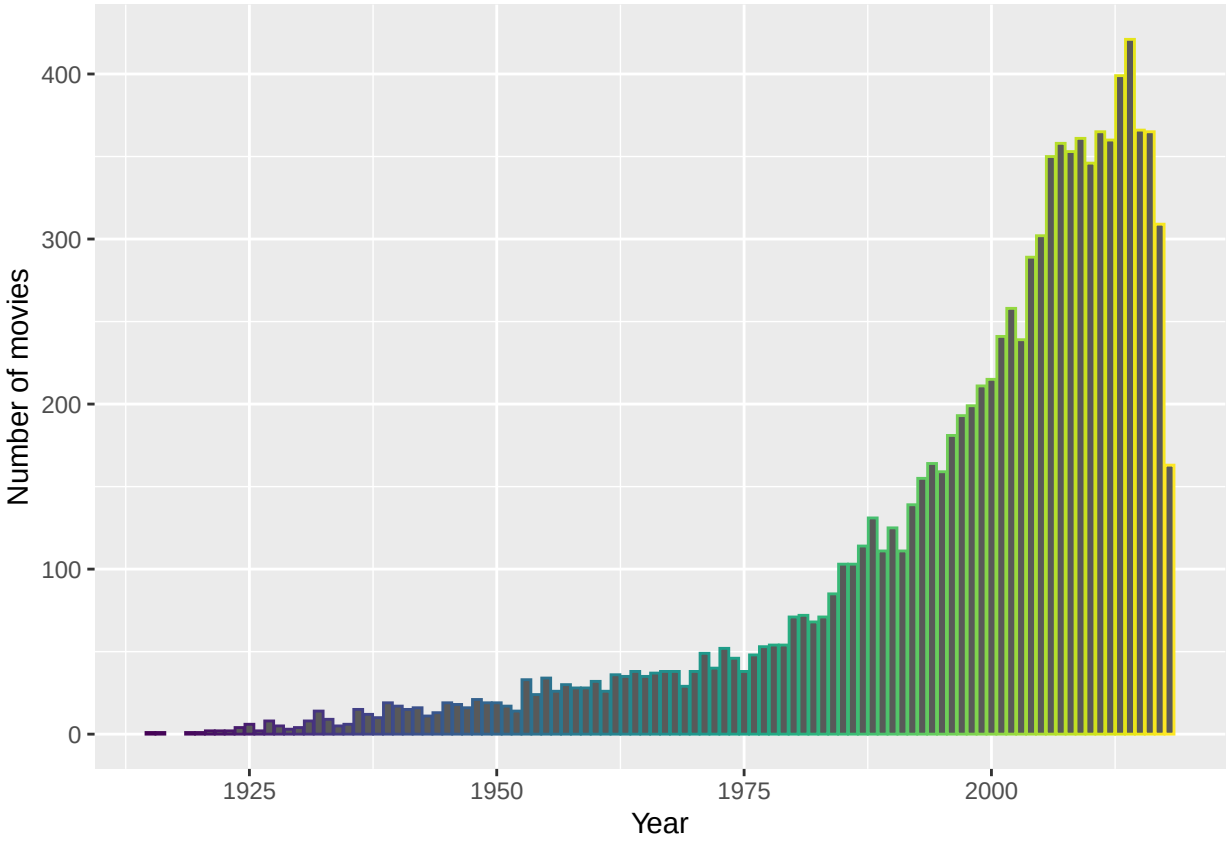
```r
df <- read.csv('C:\\Users\\Lenovo\\Desktop\\511 project\\movies.csv')
df.groupby <- group_by(df,Year)
df.sum <- summarise(df.groupby,
                    counts=n())
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```r
sum(df.sum$counts)
```

```
## [1] 10000
```

```r
ggplot(df.sum,aes(Year,counts),fill=diqu) +
  geom_bar(stat = 'identity',color=viridis(102)) +
  labs(y='Number of movies')
```



Since our dataset is about the 10000 of the most voted movies, it is reasonable to deduce that in general, the movies in recent years get more votes and may be more popular than old movies.

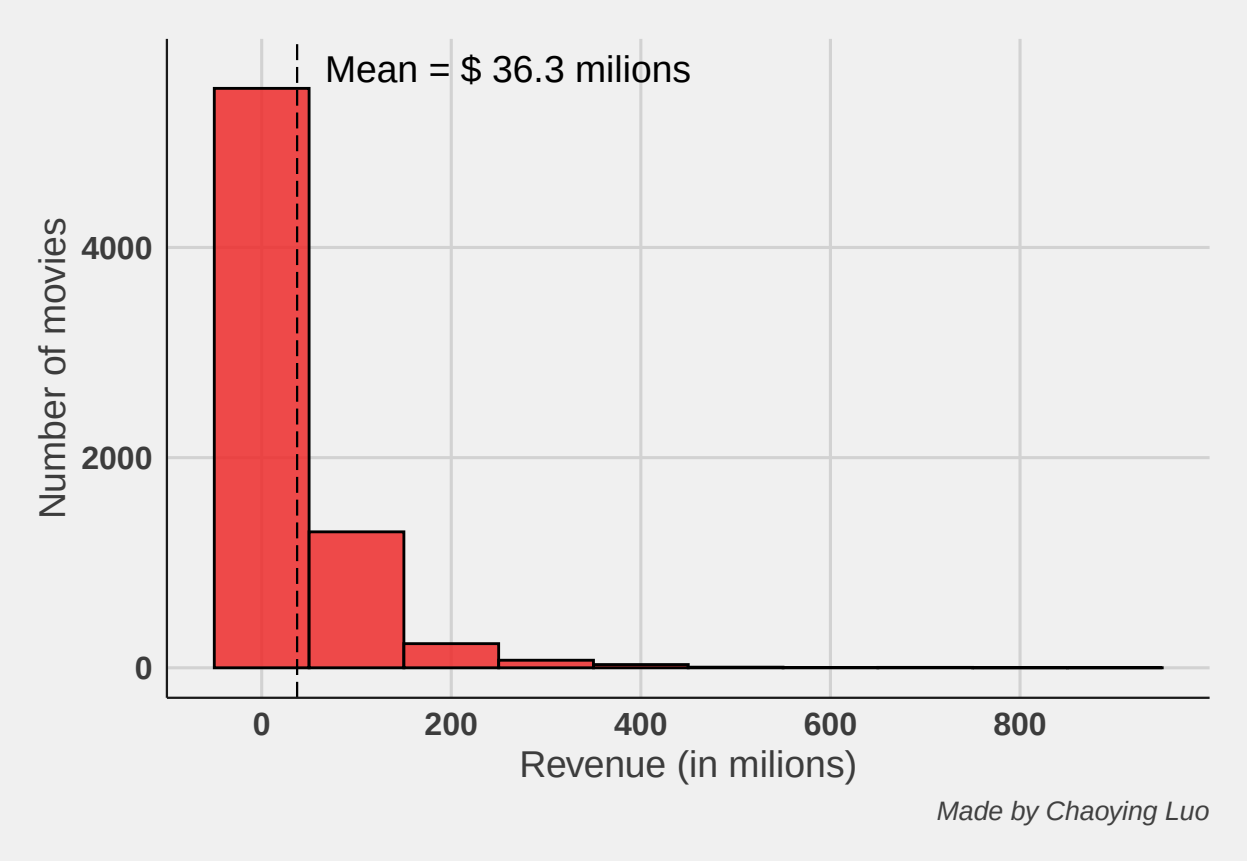
Revenue

```
movie1 <-read.csv('movies1.csv',header = T,sep=',',encoding = 'utf-8')
```

```
library(ggplot2)
library(ggthemes)
```

```
## Warning: package 'ggthemes' was built under R version 4.0.3
```

```
ggplot(movie1, aes(Revenue))+
  geom_histogram(binwidth = 100, fill = "firebrick2", col = "black", alpha = 0.85)+
  geom_vline(xintercept = mean(movie1$Revenue), linetype = "longdash", size = 0.4)+
  labs(x = "Revenue (in milions)", y = "Number of movies", caption = "Made by Chaoying Luo")+
  scale_x_continuous(breaks = c(0,200,400,600,800))+
  annotate("text", x = 260, y = 5700, label = "Mean = $ 36.3 milions", size = 5)+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 12, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"),
        plot.caption = element_text(color = "gray25", face = "italic", size = 10))
```



Analysis Part

Is the distribution of scores normally distributed? (Anderson Darling Test)

```
#install.packages('tinytex')
library(ggplot2)
library(dplyr, warn.conflicts = F)
library(gridExtra, warn.conflicts = F)
library(ranger, warn.conflicts = F)
library(Metrics)
library(reshape2)
library(RColorBrewer)
library(corrplot, warn.conflicts = F)
```

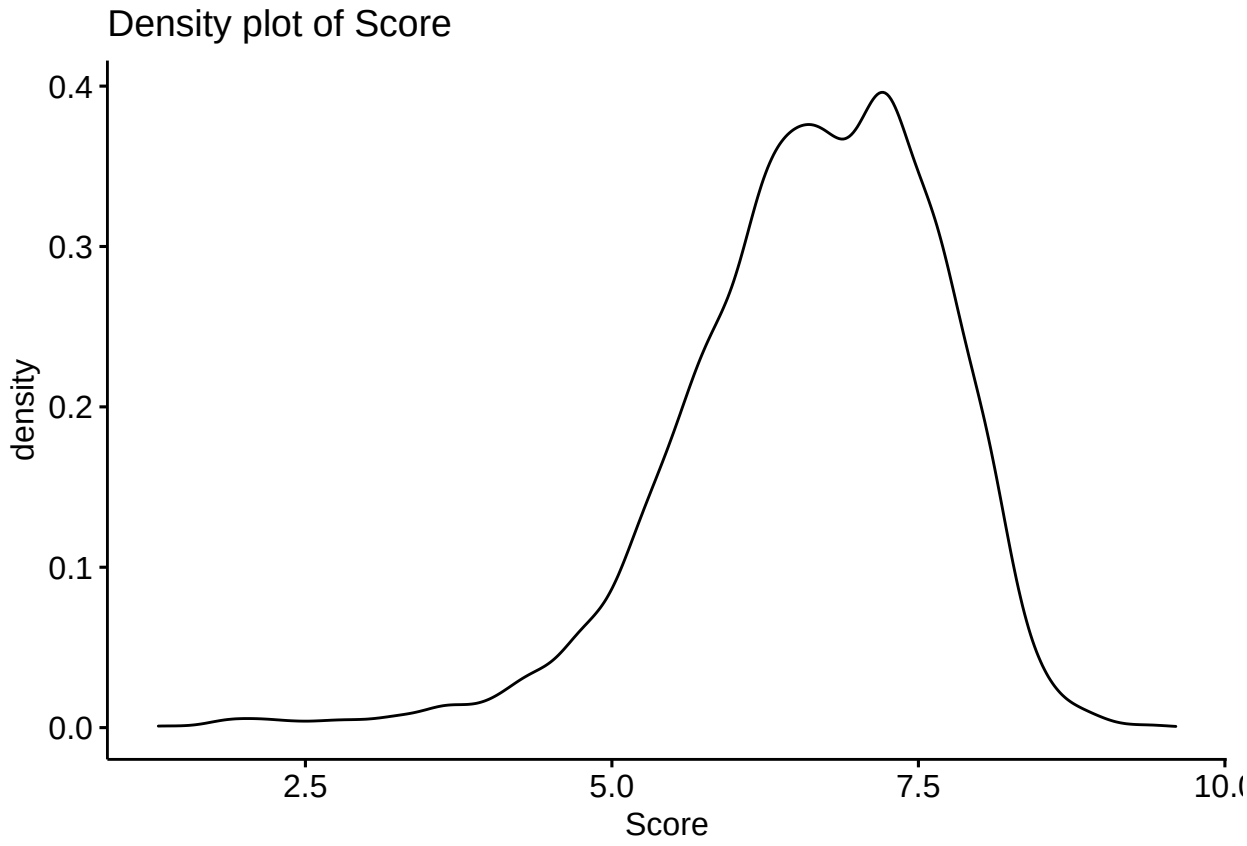
```
## corrplot 0.84 loaded
```

```
library(ggthemes)

# load data
my_data <- read.csv("movies.csv")
head(my_data)
```

```
##   Rank                     Title Year Score Metascore                       Genre
## 1    1 The Shawshank Redemption 1994   9.3        80                       Drama
## 2    2          The Dark Knight 2008   9.0        84      Action, Crime, Drama
## 3    3                Inception 2010   8.8        74 Action, Adventure, Sci-Fi
## 4    4               Fight Club 1999   8.8        66                       Drama
## 5    5             Pulp Fiction 1994   8.9        94               Crime, Drama
## 6    6             Forrest Gump 1994   8.8        82             Drama, Romance
##      Vote          Director Runtime Revenue
## 1 2011509    Frank Darabont     142   28.34
## 2 1980200 Christopher Nolan     152  534.86
## 3 1760209 Christopher Nolan     148  292.58
## 4 1609459     David Fincher     139   37.03
## 5 1570194 Quentin Tarantino     154  107.93
## 6 1532024   Robert Zemeckis     142  330.25
##
## 1
## 2 When the menace known as the Joker emerges from his mysterious past, he wreaks havoc and chaos on
## 3                                                                                A thief who s
## 4
## 5                                                                                          '
## 6                                                                                The pres
```

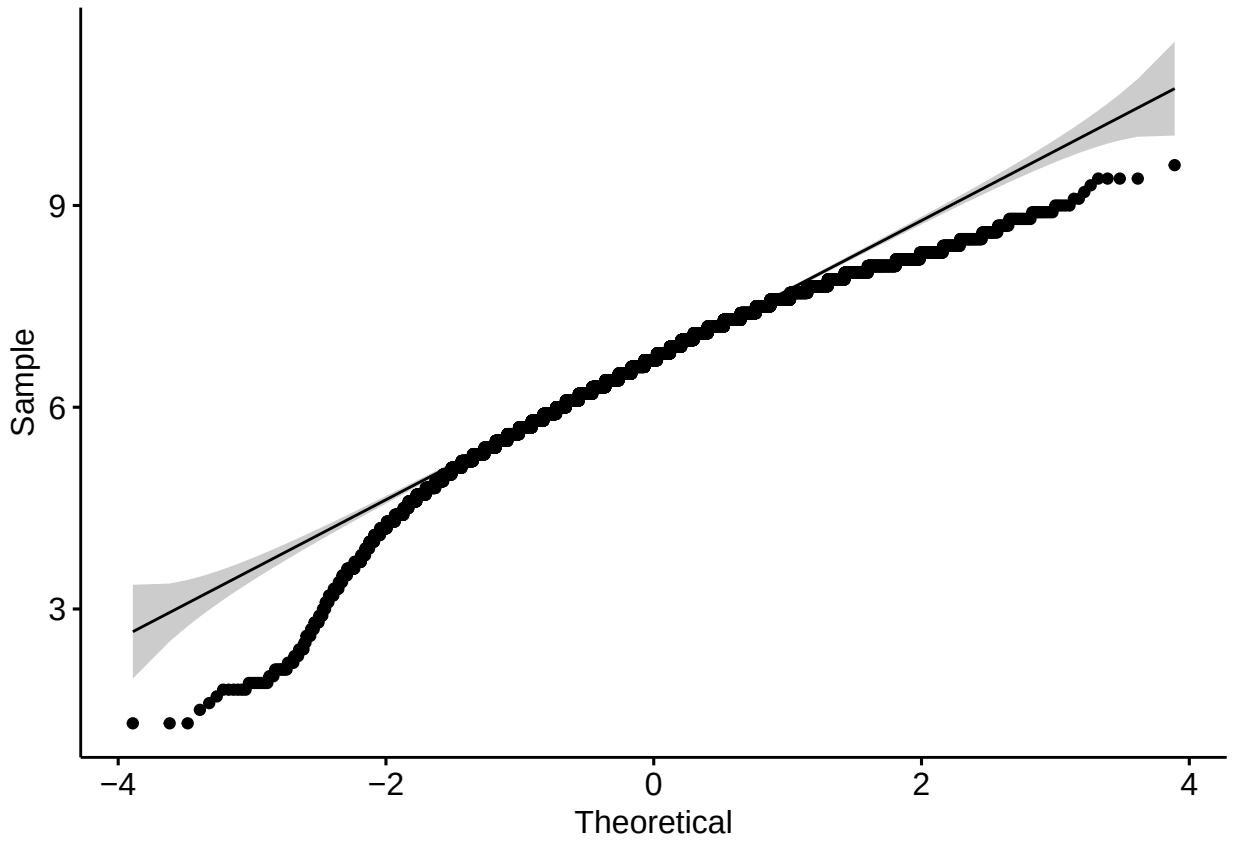
Density plot:

```
library("ggpubr")
ggdensity(my_data$Score,
          main = "Density plot of Score",
          xlab = "Score")
```



Q-Q plot:

```
library(ggpubr)
ggqqplot(my_data$Score)
```



AD-test

```
library(nortest)
ad.test(my_data$Score)
```

```
##
##  Anderson-Darling normality test
##
## data:  my_data$Score
## A = 58.572, p-value < 2.2e-16
```

Null hypothesis: the distribution of the data is the same as the normal distribution.

Alternative hypothesis: the distribution of the data is significantly different from normal distribution.

From the output, the p-value < 0.05, the null hypothesis is rejected, so the distribution is non-normal.

What is the difference of means of scores for the movies directed by Woody Allen and Alfred Hitchcock? (Two sample Bootstrap)

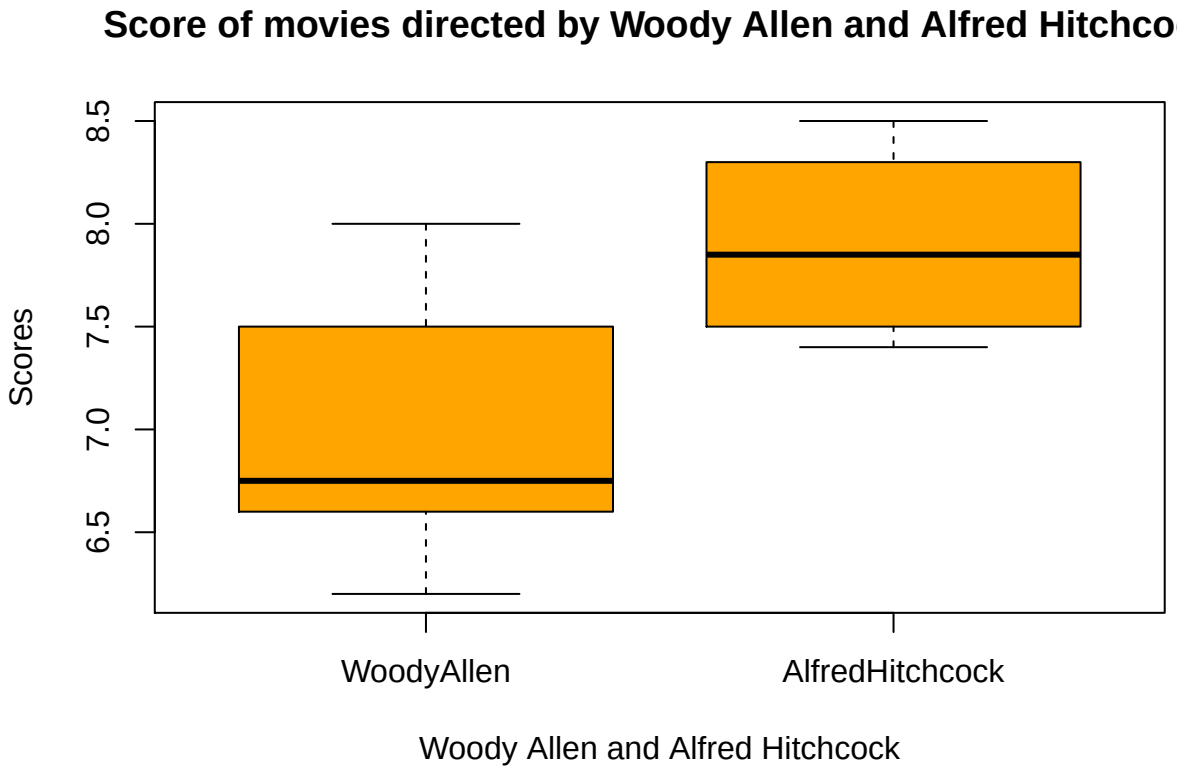
```
# check NAs for the two attributes
sum(is.na(my_data$Score)) # 0 NA
```

```
## [1] 0
```

```
sum(is.na(my_data$Director)) # 1 NA
```

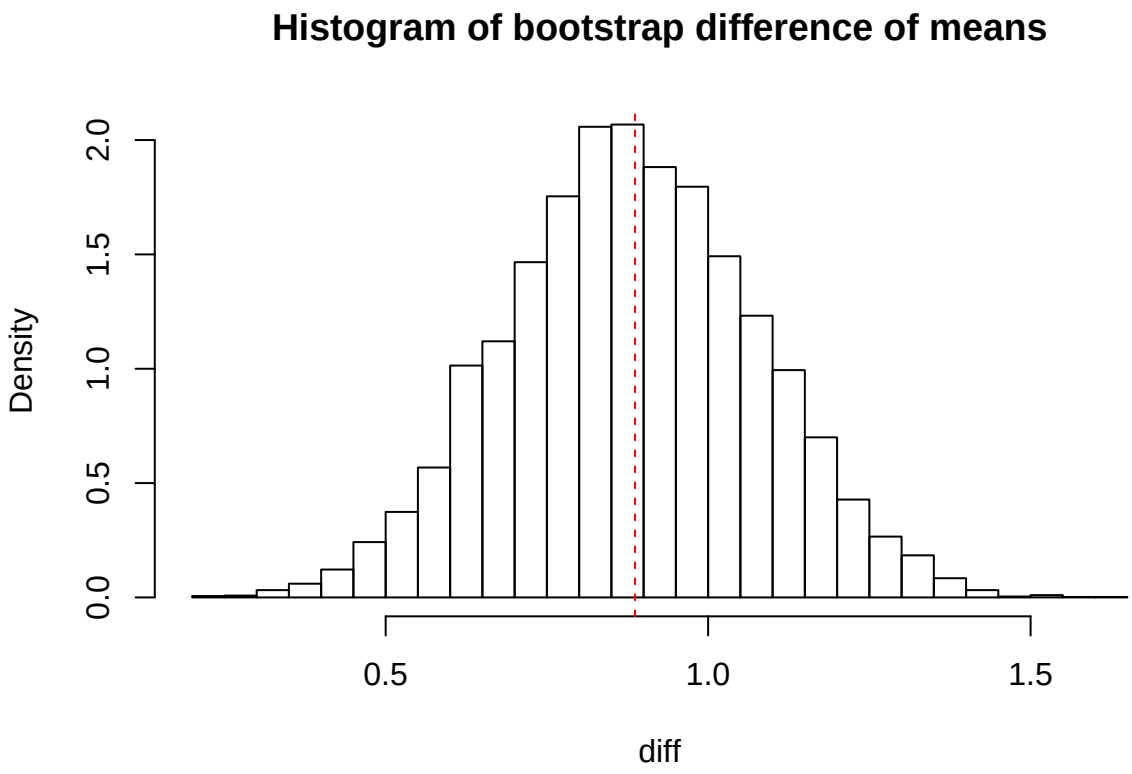
```
## [1] 1
```

```r
# clean the dataset
my_data <- na.omit(my_data)

# make side by side boxplots
WoodyAllen = my_data$Score[my_data$Director=="Woody Allen"]
AlfredHitchcock = my_data$Score[my_data$Director=="Alfred Hitchcock"]
labels = c("WoodyAllen", "AlfredHitchcock")
boxplot(WoodyAllen, AlfredHitchcock, names=labels,
        col="orange",
        main = "Score of movies directed by Woody Allen and Alfred Hitchcock",
        xlab = "Woody Allen and Alfred Hitchcock",
        ylab = "Scores")
```



The boxplots of scores of movies directed by Woody Allen and Alfred Hitchcock show that the scores of movies directed by Alfred Hitchcock is generally higher than scores of movies directed by Woody Allen.

```r
set.seed(123)
# bootstrap the difference of means
diff <- rep(NA,10000)

for (j in 1:10000){
  boot.WoodyA <- mean(sample(WoodyAllen, length(WoodyAllen), replace = T))
  boot.AlfredH <- mean(sample(AlfredHitchcock, length(AlfredHitchcock), replace = T))
  diff[j] <- boot.AlfredH-boot.WoodyA # the difference
}

# plot the bootstrap the difference of means
hist(diff, breaks = 40, prob = T, main = "Histogram of bootstrap difference of means")
truedifference = mean(AlfredHitchcock)-mean(WoodyAllen)
abline(v = truedifference, col = "red", lty = 2)
```



Comment:

The bootstrap distribution of difference of means looks quite normal, with some skewness.

```
# Find the 95% bootstrap percentile interval for the difference of means.
quantile(diff, c(0.025,0.975))
```

```
##      2.5%     97.5%
## 0.5066667 1.2666667
```

Conclusion:

The 2.5% and 97.5% points of the bootstrap distribution give us the interval (0.51, 1.27), so we are 95% confident that the difference of means between scores of movies directed by Alfred Hitchcock and Woody Allen is between 0.51 and 1.27.

Is there a relationship between the popularity and runtime of movies? (Chi-squared test for two-way tables)

Null hypothesis is that the popularity and runtime of movies are independent. The popularity do not vary by runtime of movies.

Alternative hypothesis is that the popularity and runtime of movies are dependent. The popularity do vary by runtime of movies.

```
# Discretization
# categorize runtime
my_data$runtime_cat = numeric(length((my_data$Runtime)))
my_data$runtime_cat[my_data$Runtime<=90]="Short"
my_data$runtime_cat[my_data$Runtime<=120 & my_data$Runtime>=91]="Medium"
```

```
my_data$runtime_cat[my_data$Runtime>=121]="Long"
my_data$runtime_cat = as.factor(my_data$runtime_cat)
str(my_data$runtime_cat)
```

```
##  Factor w/ 3 levels "Long","Medium",..: 1 1 1 1 1 1 1 1 1 1 ...
```

```
# categorize vote
my_data$vote_cat = numeric(length((my_data$Vote)))
my_data$vote_cat[my_data$Vote<=167625]="Not Popular"
my_data$vote_cat[my_data$Vote<=335251 & my_data$Vote>=167626]="Slightly Popular"
my_data$vote_cat[my_data$Vote<=1005754 & my_data$Vote>=335252]="Moderately Popular"
my_data$vote_cat[my_data$Vote>=1005755]="Very Popular"
my_data$vote_cat = as.factor(my_data$vote_cat)
str(my_data$vote_cat)
```

```
##  Factor w/ 4 levels "Moderately Popular",..: 4 4 4 4 4 4 4 4 4 4 ...
```

```
# test
a = table(my_data$vote_cat, my_data$runtime_cat)
chisq.test(a)
```

```
## Warning in chisq.test(a): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  a
## X-squared = 430.72, df = 6, p-value < 2.2e-16
```

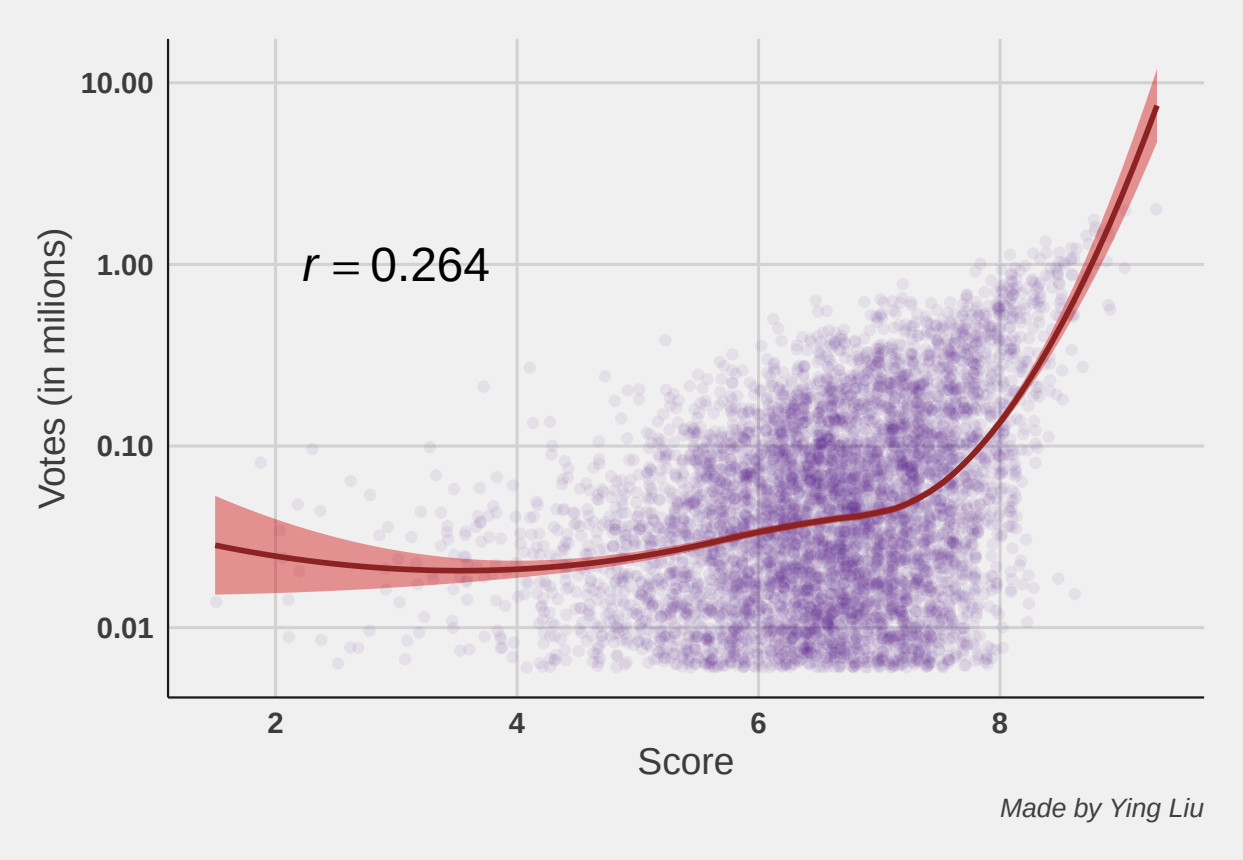
Conclusion:

Since p-value is less than 0.05, we can reject the null hypothesis. Thus, there is enough evidence to conclude that there is a significant relationship between the popularity and runtime of movies. They are dependent.

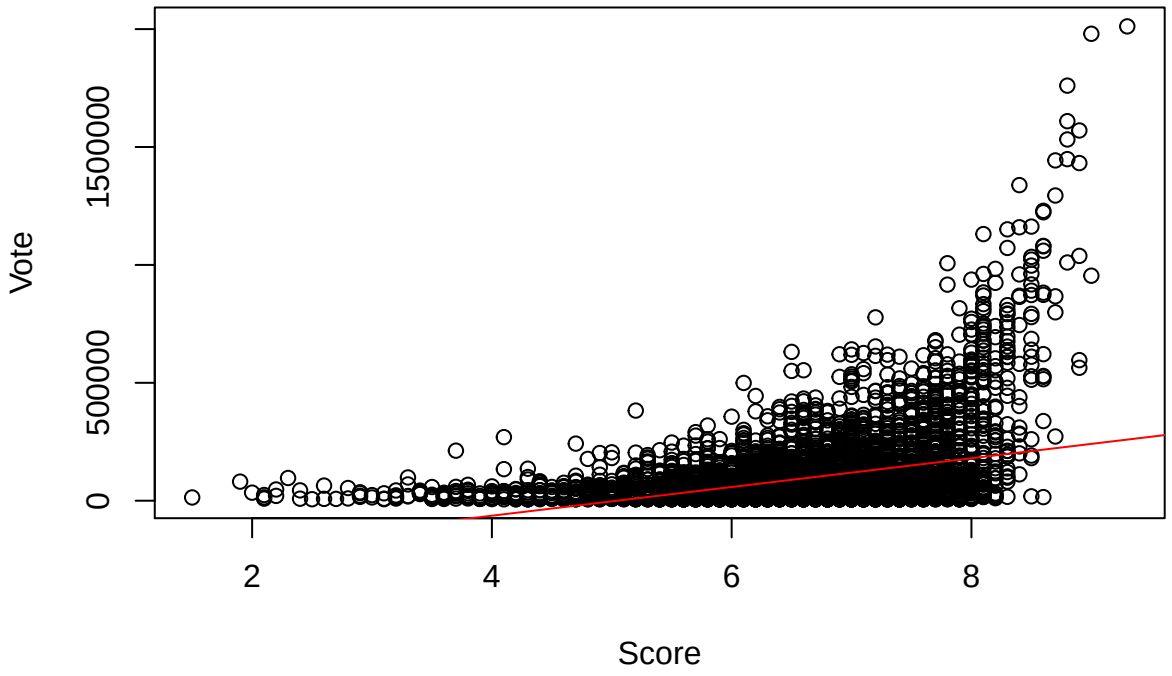
Is there a relationship between the popularity and scores of movies? (Multiple Linear Regression)

Correlation between the score and the vote. (Linear regression)

```
my_data %<>%
  as_tibble() %>%
  mutate(VoteMln = Vote/1000000)

ggplot(my_data, aes(x=Score, y=VoteMln)) +
  geom_jitter(alpha = 0.07, col = "purple4") +
  geom_smooth(method = "loess", fill = "red3", color = "brown4", formula = y ~ x)+
  scale_y_continuous(trans="log10", name = "Votes (in milions)")+
  scale_x_continuous(name = "Score", breaks = c(0,2,4,6,8,10))+
  annotate("text", x = 3, y = 1, label = "italic(r) == 0.264", parse = T, size = 6.5)+
  labs(caption = "Made by Ying Liu")+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 11, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"), plot.caption = element_text(color = "g
```




```
# make a plot
lm.fit1=lm(Vote~Score,data=my_data)
plot(Vote~Score,data=my_data)
abline(lm.fit1, col="red")
```



```r
lm.fit=lm(Vote~Year+Score+Metascore+Runtime+Revenue,data=my_data)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Vote ~ Year + Score + Metascore + Runtime + Revenue,
##     data = my_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -777413  -49647  -15262   26663 1785591
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.782e+06  2.407e+05  -7.400 1.54e-13 ***
## Year         7.136e+02  1.192e+02   5.985 2.29e-09 ***
## Score        5.396e+04  2.211e+03  24.399  < 2e-16 ***
## Metascore   -5.144e+02  1.135e+02  -4.533 5.92e-06 ***
## Runtime      6.173e+02  8.043e+01   7.675 1.92e-14 ***
## Revenue      1.278e+03  2.153e+01  59.354  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 107600 on 6119 degrees of freedom
## Multiple R-squared:  0.4877, Adjusted R-squared:  0.4873
## F-statistic:  1165 on 5 and 6119 DF,  p-value: < 2.2e-16
```

Observation:

The F statistic is very large, so there is a relationship between the predictor and the response. Since $R^2 \approx 0.4877$, the relationship is fairly strong.

The predictor Score, has a positive and statistically significant relationship with votes. The coefficient on Score in terms of vote is 53960, which means that an additional 1 increase in score of movies can lead to an increase in votes by approximately 53960.

Are IMDb users less dispersed and do they have milder assessments than Metascore users? (Hypothesis testing)

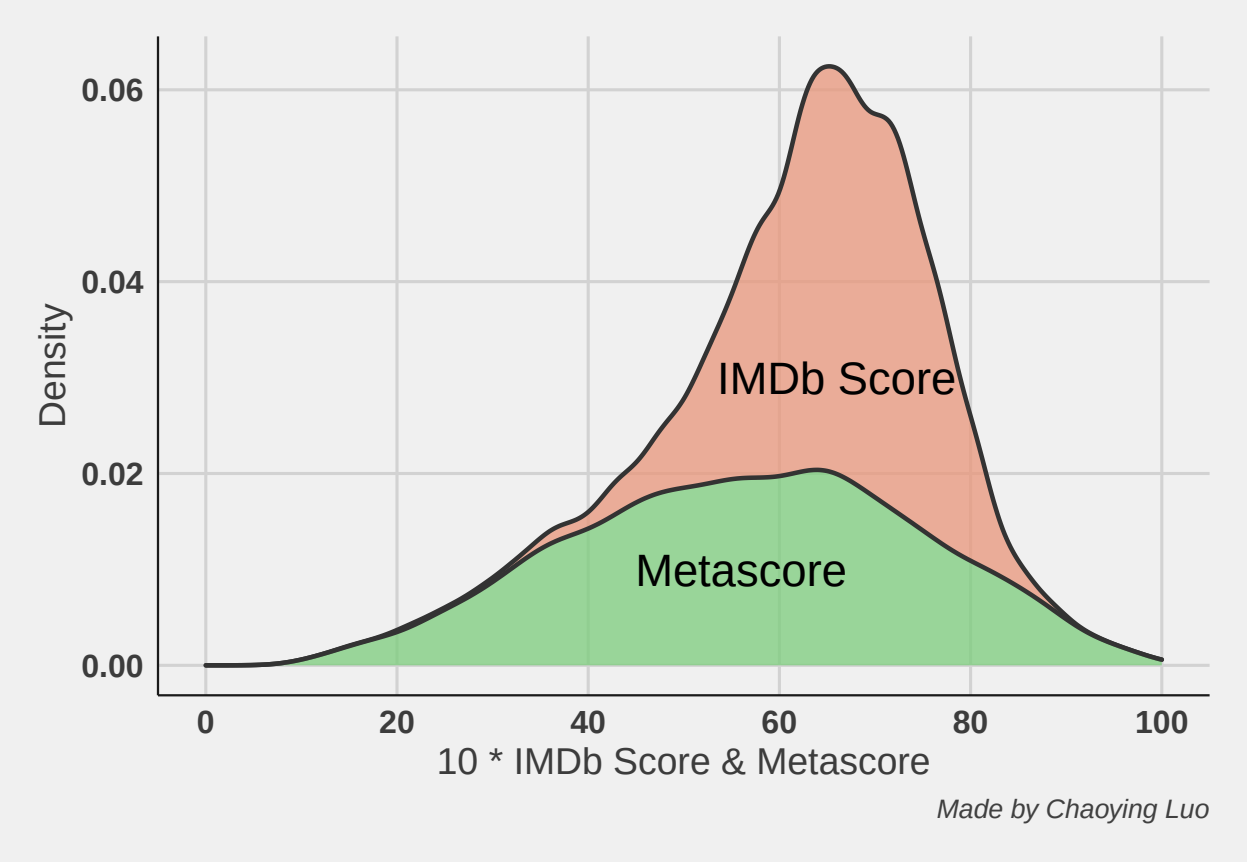
```
movie_score <-read.csv('movie_score.csv',header = T,sep=',',encoding = 'utf-8')
head(movie_score)
```

```
##   IMDbscore Metascore
## 1       9.3        80
## 2       9.0        84
## 3       8.8        74
## 4       8.8        66
## 5       8.9        94
## 6       8.8        82
```

```
library(ggplot2)
library(ggthemes)
```

```
## Warning: package 'ggthemes' was built under R version 4.0.3
```

```
IMDbscore <- data.frame(movie_score$IMDbscore * 10)
MetaScore <- data.frame(movie_score$Metascore)

names(IMDbscore)[1] <- paste("Score")
names(MetaScore)[1]<- paste("Score")

IMDbscore$group <- "IMDb Score"
MetaScore$group <- "Metascore"
Together <- rbind(IMDbscore, MetaScore)

ggplot(Together, aes(Score, fill = group)) +
  geom_density(alpha = 0.75, position = "stack", size = 0.8, show.legend = F, col = "gray20")+
  scale_x_continuous(limits = c(0,100), breaks = c(0,20,40,60,80,100))+
  annotate("text", x = 56, y = 0.01, label = "Metascore", size = 6)+
  annotate("text", x = 66, y = 0.03, label = "IMDb Score", size = 6)+
  scale_fill_manual(values=c("darksalmon", "palegreen3"))+
  labs(y = "Density", x = "10 * IMDb Score & Metascore", caption = "Made by Chaoying Luo")+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 12, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10")
        , plot.caption = element_text(color = "gray25",
                                      face = "italic", size = 10))
```




```
chisq.test(table(movie_score$IMDbscore,movie_score$Metascore))
```

```
## Warning in chisq.test(table(movie_score$IMDbscore, movie_score$Metascore)): Chi-
## squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(movie_score$IMDbscore, movie_score$Metascore)
## X-squared = 20257, df = 6497, p-value < 2.2e-16
```

```
z.1 <- rep(NA,1000)
for (j in 1:1000){
movie_score.IMDbscore <- mean(sample(movie_score$IMDbscore, length(movie_score$IMDbscore), replace = T))
movie_score.Metascore <- mean(sample(movie_score$Metascore, length(movie_score$Metascore), replace = T))
z.1[j] <- movie_score.IMDbscore - movie_score.Metascore #the difference
}
```

```
ci.1 <- quantile(z.1, c(.025, .975))
ci.1
```

```
##      2.5%     97.5%
## -50.40254 -49.55779
```

CI does not contain 0. True difference is likely to be negative. That means we reject the null hypothesis at 5% significance level. That means there are some differences between the two population means.

Correlation between the score and the revenue. (Linear regression)

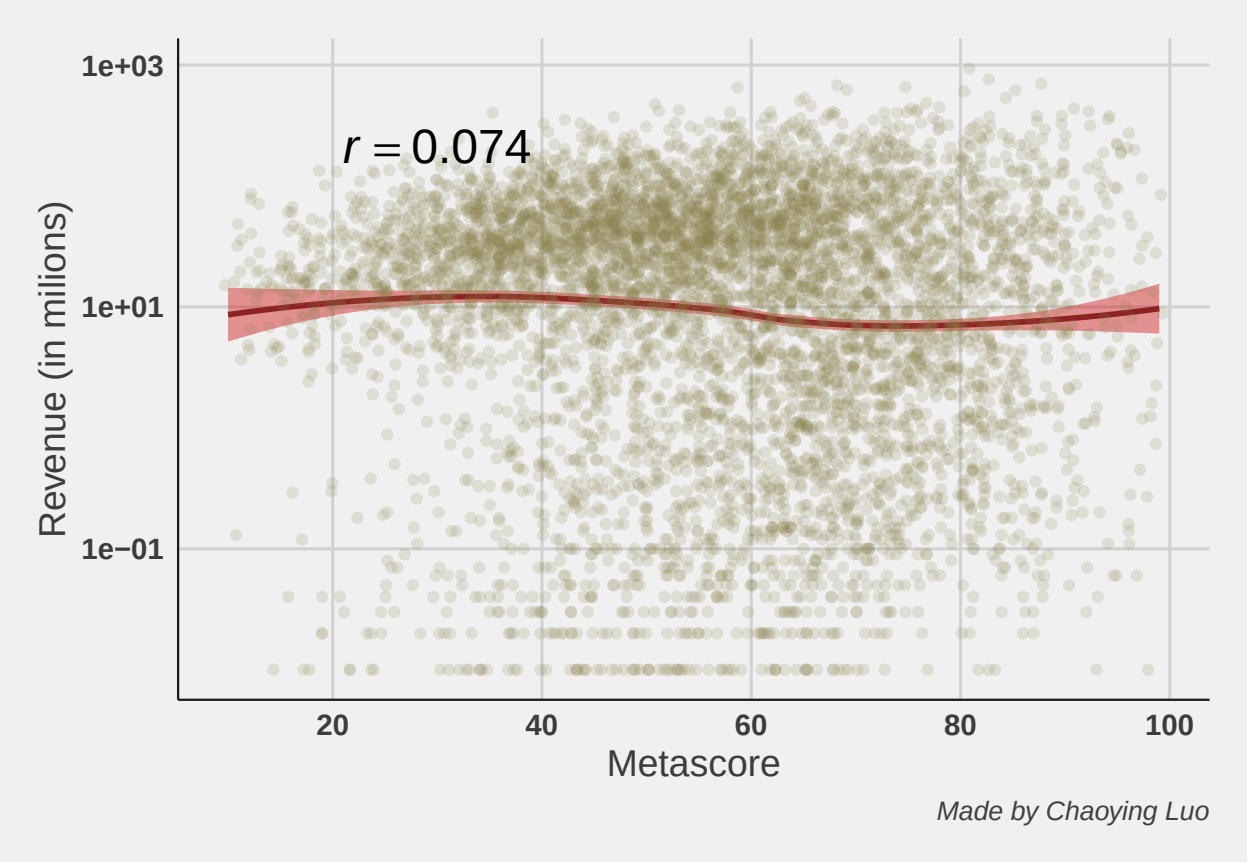
```
movie_score_revenue <-read.csv('movie_score_revenue.csv',header = T,sep=',',encoding = 'utf-8')
head(movie_score_revenue)
```

```
##   IMDbscore Metascore    Vote          Director Runtime Revenue
## 1       9.3        80 2011509    Frank Darabont     142   28.34
## 2       9.0        84 1980200 Christopher Nolan     152  534.86
## 3       8.8        74 1760209 Christopher Nolan     148  292.58
## 4       8.8        66 1609459     David Fincher     139   37.03
## 5       8.9        94 1570194 Quentin Tarantino     154  107.93
## 6       8.8        82 1532024   Robert Zemeckis     142  330.25
```

```
lm.fit=lm(Revenue~IMDbscore+Metascore,data=movie_score_revenue)
summary(lm.fit)
```

```
##
## Call:
## lm(formula = Revenue ~ IMDbscore + Metascore, data = movie_score_revenue)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -62.01 -38.68 -19.44  12.35 883.41
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -20.05876    6.24385  -3.213  0.00132 **
## IMDbscore    10.78573    1.28758   8.377  < 2e-16 ***
## Metascore    -0.16015    0.06908  -2.318  0.02047 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.31 on 6069 degrees of freedom
## Multiple R-squared:  0.01655,    Adjusted R-squared:  0.01623
## F-statistic: 51.08 on 2 and 6069 DF,  p-value: < 2.2e-16
```

```
ggplot(movie_score_revenue, aes(x=Metascore, y=Revenue)) +
  geom_smooth(method = "loess", color = "brown4", fill = "red3", formula = y ~ x)+
  geom_jitter(alpha=0.17, col = "lightgoldenrod4")+
  scale_y_continuous(trans="log10", name="Revenue (in milions)")+
  scale_x_continuous(name="Metascore", breaks = seq(0,100,20))+
  annotate("text", x = 30, y = 210, label = "italic(r) == 0.074", parse = T, size = 6.5)+
  labs(caption = "Made by Chaoying Luo")+
  theme_fivethirtyeight()+
  theme(axis.title = element_text(size = 14), axis.text = element_text(size = 11, face = "bold"),
        axis.line = element_line(size = 0.4, colour = "grey10"),
        plot.caption = element_text(color = "gray25", face = "italic", size = 10))
```



There is no correlation between Metascore and the revenue from the movie. The scores are definitely more dispersed than revnue. It isn't possible to relate these two variables.

```
lm.fit2=lm(Revenue~IMDbscore,data=movie_score_revenue)
summary(lm.fit2)
```

```
##
## Call:
## lm(formula = Revenue ~ IMDbscore, data = movie_score_revenue)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -59.76 -39.23 -18.83  12.61 882.65
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -14.7287     5.8072  -2.536   0.0112 *
## IMDbscore     8.5926     0.8737   9.834   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 65.33 on 6070 degrees of freedom
## Multiple R-squared:  0.01568,    Adjusted R-squared:  0.01552
## F-statistic: 96.71 on 1 and 6070 DF,  p-value: < 2.2e-16
```

There is a positive correlation between the popularity of movie and opinion of IMDB Users. Very good rated

movies by IMDB users are much more popular than others. A unit increase in IMDbscore results in 8.6 million increase in revenue, all other variables held constant.

The difference of the means of revenue for the movies of these two directors Woody Allen and Clint Eastwood. (Two sample Bootstrap)

```
director_revenue <-read.csv('director_revenue.csv',header = T,sep=',',encoding = 'utf-8')
director_revenue
```

```
##    Woody.Allen Clint.Eastwood
## 1        56.82         148.10
## 2        39.20         100.49
## 3        23.22         350.13
## 4        23.09          90.14
## 5        33.41         101.16
## 6        45.70          35.74
## 7        16.69         125.07
## 8        10.53          13.76
## 9         5.31          37.49
## 10       40.08          37.31
## 11       11.10          33.60
## 12       10.51          32.75
## 13       18.25          90.46
## 14        4.03          31.16
## 15        0.97          71.52
## 16       10.63          31.80
## 17        3.25          41.41
## 18       10.57          50.01
## 19       11.80          15.70
## 20        7.50          67.64
## 21        6.70          42.72
## 22       17.07          25.08
## 23        9.71          47.03
## 24       11.29          16.64
## 25       13.38          21.63
## 26        4.20          46.71
## 27        3.83          10.60
## 28       14.79          35.40
## 29        3.20          36.25
## 30       10.56           2.18
## 31        4.84           2.32
## 32        5.03          24.27
## 33       10.39           4.48
## 34        1.40             NA
## 35       10.76             NA
## 36        2.74             NA
## 37        7.33             NA
## 38        0.49             NA
```

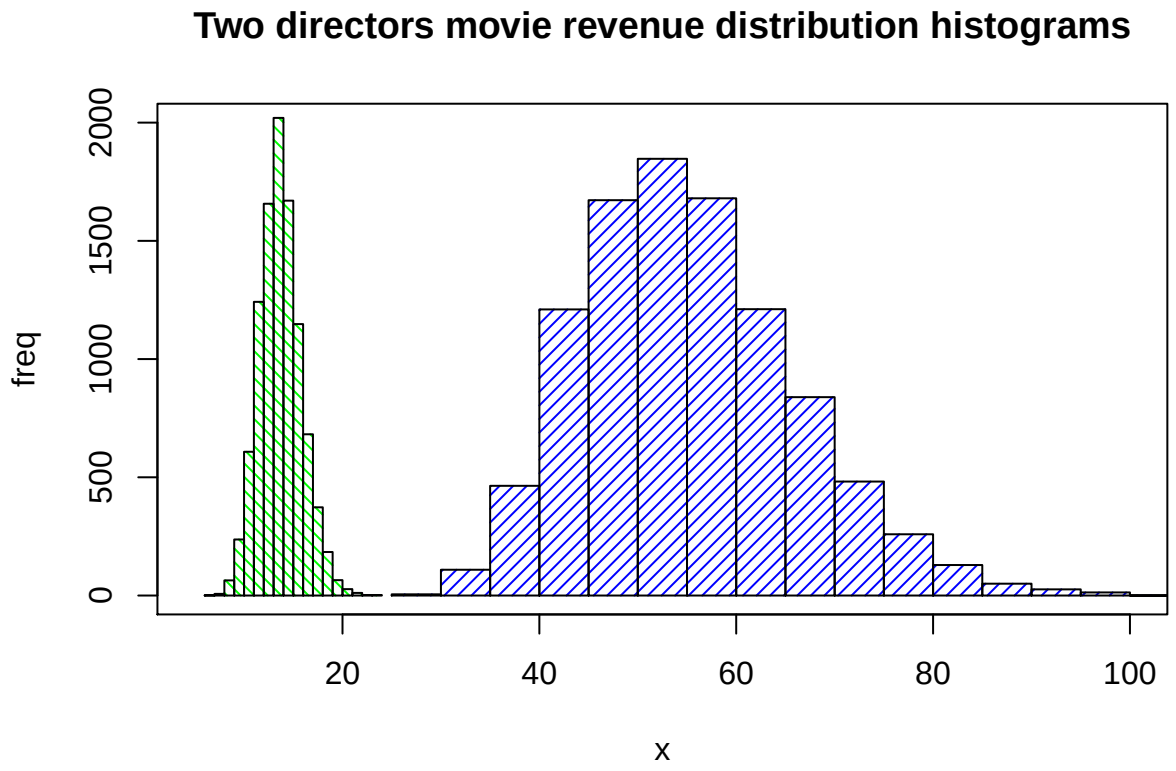
```
n1 <- length(director_revenue$Woody.Allen)
Clint<-director_revenue$Clint.Eastwood[-which(is.na(director_revenue$Clint.Eastwood))]
```

```r
n2 <- length(Clint)
N <- 10000
Woody.mean <- numeric(N)
for (i in 1:N)
{
  x <- sample(director_revenue$Woody.Allen, n1, replace = TRUE)
  Woody.mean[i] <- mean(x) #bootstrap sample mean
}
Clint.mean <- numeric(N)
for (i in 1:N)
{
  x <- sample(Clint, n2, replace = TRUE)
  Clint.mean[i] <- mean(x) #bootstrap sample mean
}

Woody<-director_revenue$Woody.Allen

combined <- cbind(Woody,Clint)
```

```
## Warning in cbind(Woody, Clint): number of rows of result is not a multiple of
## vector length (arg 2)
```

```r
boxplot(combined,col="lightblue",main="Revenues of movies directed by Woody Allen and Clint Eastwood")
```

```
n <- 10000
p3<-0.5
plot1 <- hist(Woody.mean,plot=FALSE)
plot2 <- hist(Clint.mean,plot=FALSE)

plot(0,0,type="n",xlim=c(5,100),ylim=c(0,2000),xlab="x"
     ,ylab="freq",main="Two directors movie revenue distribution histograms")
plot(plot1,col="green",density=20,angle=135,add=TRUE)
plot(plot2,col="blue",density=20,angle=45,add=TRUE)
```




```
#hist(Woody.mean, main = "Bootstrap distribution of Woody Allen movie revenue")
#hist(Clint.mean, main = "Bootstrap distribution of Clint Eastwood movie revenue")
```

From the histograms, the mean of Woody Allen and Clint Eastwood movie revenue has a normal distribution. Clint Eastwood has a higher box office.

Do old movies have better reputations than new movies?(Student's t-test)

```
library("dplyr")
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("tidyr")
library("stringr")
library("corrplot")
```

```
## corrplot 0.84 loaded
```

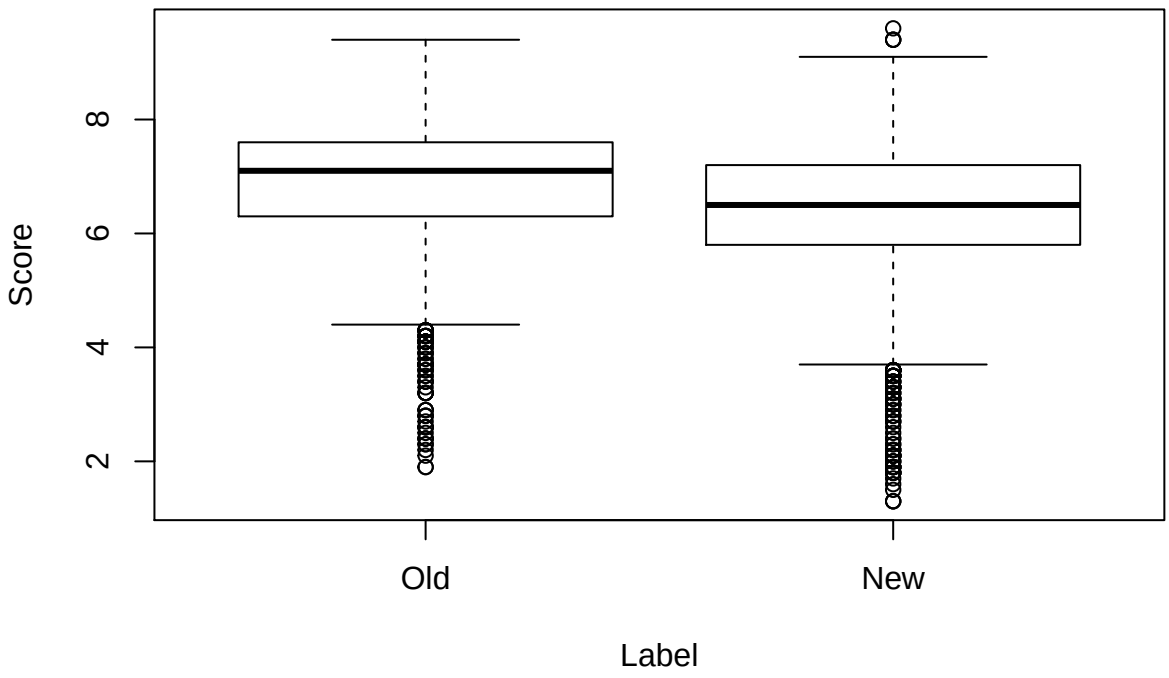
```
library("RColorBrewer")

df <- read.csv('C:\\Users\\Lenovo\\Desktop\\511 project\\movies.csv')
# Hypothesis Testing
sum(is.na(df$Score))
```

```
## [1] 0
```

```
sum(is.na(df$Metascore))
```

```
## [1] 3219
```

```
oldmovies <- df[df$Year<2000,'Score']
newmovies <- df[df$Year>=2000,'Score']
old <- data.frame(oldmovies,rep('Old',length(oldmovies)))
colnames(old) <- c('Score','Label')
new <- data.frame(newmovies,rep('New',length(newmovies)))
colnames(new) <- c('Score','Label')
df.new <- rbind(old,new)
boxplot(Score~Label,data=df.new)
```



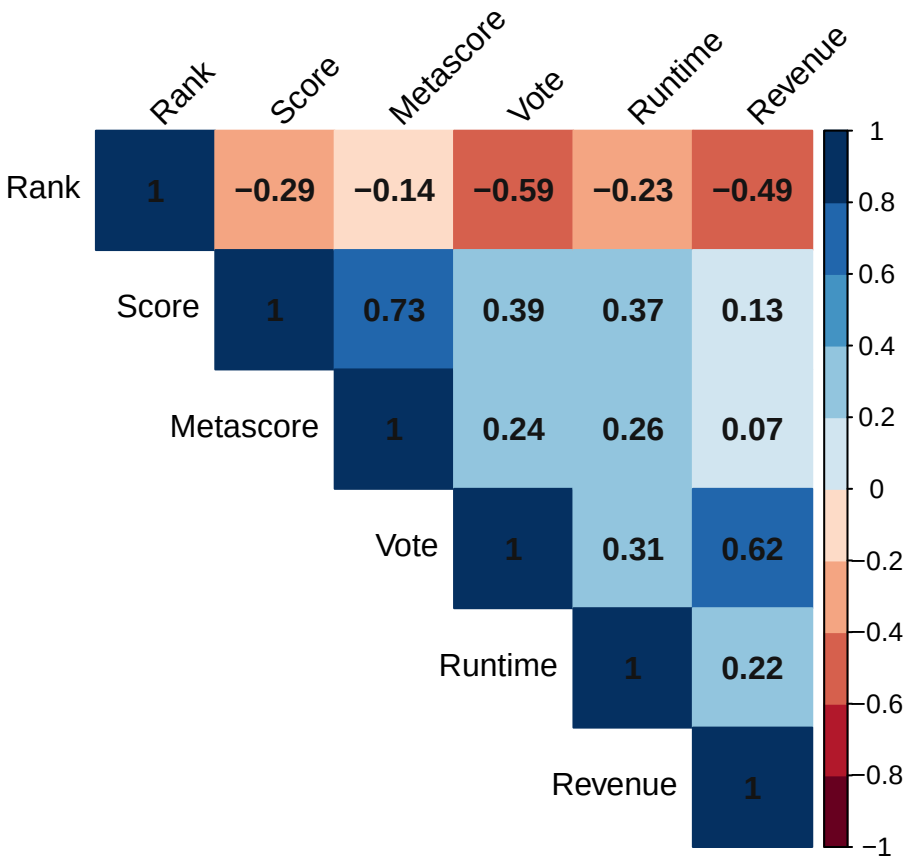

```
t.test(oldmovies,newmovies,alternative = 'greater',conf.level = 0.95)
```

```
##
##  Welch Two Sample t-test
##
## data:  oldmovies and newmovies
## t = 21.688, df = 8601.4, p-value < 2.2e-16
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  0.4147423       Inf
## sample estimates:
## mean of x mean of y
##  6.899492  6.450710
```

The null hypothesis is that the score of old movies is equal to the score of new movies. The alternative hypothesis is that the score of old movies is greater than new movies. Since the p-value is less than 5%, at the 5% significance level, we have enough evidence to reject the null hypothesis.

What is the correlation between the variables of the dataset?

```
df.complete <- df[complete.cases(df),]
cols.numeric <- unlist(lapply(df, is.numeric))
cor.matrix <- cor(df.complete[,cols.numeric])
corrplot(cor.matrix[c(1, 3:7),c(1,3:7)], method="color", type = "upper", col=brewer.pal(n=10, name="RdBu
```



From the heatmap, it is not difficult to find that the score and the metascore have the strongest positive correlation, besides, the rank and the vote, the rank and the revenue have the top 2 of the strongest negative correlation.

Using Naive Bayes to Predict The Relationship Between Movie Scores and Revenue.

```
library(datasets)
library(class) ## for knn
library(mlr) ## for vis
```

```
## Loading required package: ParamHelpers
```

```
## 'mlr' is in maintenance mode since July 2019. Future development
## efforts will go into its successor 'mlr3' (<https://mlr3.mlr-org.com>).
```

```
library(ggplot2)
library(plyr) ## load this BEFORE dplyr
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 4.0.2
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(lattice)
library(caret)
```

```
##
## Attaching package: 'caret'
```

```
## The following object is masked from 'package:mlr':
##
##     train
```

```
library(e1071)
```

```
##
## Attaching package: 'e1071'
```

```
## The following object is masked from 'package:mlr':
##
##     impute
```

```
#library(ElemStatLearn)
library(gmodels)
```

```
## Warning: package 'gmodels' was built under R version 4.0.2
```

```
library(GGally)
```

```
## Warning: package 'GGally' was built under R version 4.0.2
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(randomForest)
```

```
## Warning: package 'randomForest' was built under R version 4.0.2
```

```
## randomForest 4.6-14
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
library(stringr)
library(naivebayes)
```

```
## naivebayes 0.9.7 loaded
```

```
library(mclust)
```

```
## Package 'mclust' version 5.4.6
## Type 'citation("mclust")' for citing this R package in publications.
```

```
library(cluster)
library(tm)
```

```
## Loading required package: NLP
```

```
##
## Attaching package: 'NLP'
```

```
## The following object is masked from 'package:ggplot2':
##
##     annotate
```

```
library(rpart)
library(rattle)
```

```
## Loading required package: tibble
```

```
## Loading required package: bitops
```

```
## Rattle: A free graphical interface for data science with R.
## Version 5.4.0 Copyright (c) 2006-2020 Togaware Pty Ltd.
## Type 'rattle()' to shake, rattle, and roll your data.
```

```
##
## Attaching package: 'rattle'
```

```
## The following object is masked from 'package:randomForest':
##
##     importance
```

```
library(rpart.plot)
library(RColorBrewer)
#library(Cairo)
library(philentropy)
library(forcats)
library(lsa) #for cosine similarity
```

```
## Loading required package: SnowballC
```

```
library(corrplot)
```

```
## corrplot 0.84 loaded
```

```
library(igraph)
```

```
##
## Attaching package: 'igraph'
```

```
## The following object is masked from 'package:tibble':
##
##     as_data_frame
```

```
## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union
```

```
## The following object is masked from 'package:class':
##
##     knn
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##     union
```

```
library(pastecs)
```

```
##
## Attaching package: 'pastecs'
```

```
## The following objects are masked from 'package:dplyr':
##
##     first, last
```

```
library(ggpubr)
```

```
##
## Attaching package: 'ggpubr'
```

```
## The following object is masked from 'package:plyr':
##
##     mutate
```

```
library(psych)
```

```
## Warning: package 'psych' was built under R version 4.0.2
```

```
##
```

```
## Attaching package: 'psych'

## The following objects are masked from 'package:philentropy':
##
##     manhattan, minkowski

## The following object is masked from 'package:mclust':
##
##     sim

## The following object is masked from 'package:randomForest':
##
##     outlier

## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

```
library(readxl)
library(tinytex)
```

```
df <- read.csv("/Users/ruitongliu/Desktop/movies.csv")
str(df)
```

```
## 'data.frame':    10000 obs. of  11 variables:
##  $ Rank       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Title      : chr  "The Shawshank Redemption" "The Dark Knight" "Inception" "Fight Club" ...
##  $ Year       : int  1994 2008 2010 1999 1994 1994 2001 1999 2003 1972 ...
##  $ Score      : num  9.3 9 8.8 8.8 8.9 8.8 8.8 8.7 8.9 9.2 ...
##  $ Metascore  : int  80 84 74 66 94 82 92 73 94 NA ...
##  $ Genre      : chr  "Drama" "Action, Crime, Drama" "Action, Adventure, Sci-Fi" "Drama" ...
##  $ Vote       : int  2011509 1980200 1760209 1609459 1570194 1532024 1448561 1443130 1431887 1378253
##  $ Director   : chr  "Frank Darabont" "Christopher Nolan" "Christopher Nolan" "David Fincher" ...
##  $ Runtime    : int  142 152 148 139 154 142 178 136 201 175 ...
##  $ Revenue    : num  28.3 534.9 292.6 37 107.9 ...
##  $ Description: chr  "Two imprisoned men bond over a number of years, finding solace and eventual re
```

```
set.seed(9850)
u_num <- runif(nrow(df))
```

```
Newdf <- df[order(u_num),]
(summary(Newdf))
```

```
##       Rank          Title                Year          Score
##  Min.   :    1   Length:10000       Min.   :1915   Min.   :1.300
##  1st Qu.: 2501   Class :character   1st Qu.:1991   1st Qu.:6.000
##  Median : 5000   Mode  :character   Median :2004   Median :6.700
##  Mean   : 5000                      Mean   :1998   Mean   :6.628
##  3rd Qu.: 7500                      3rd Qu.:2011   3rd Qu.:7.400
##  Max.   :10000                      Max.   :2018   Max.   :9.600
##
##    Metascore        Genre                Vote           Director
##  Min.   :10.00   Length:10000       Min.   :   6015   Length:10000
##  1st Qu.:44.00   Class :character   1st Qu.:  10147   Class :character
##  Median :57.00   Mode  :character   Median :  21172   Mode  :character
##  Mean   :56.53                      Mean   :  64488
##  3rd Qu.:70.00                      3rd Qu.:  62052
##  Max.   :99.00                      Max.   :2011509
```

```
##  NA's   :3219
##     Runtime         Revenue       Description
##  Min.   : 45.0   Min.   :  0.00   Length:10000
##  1st Qu.: 94.0   1st Qu.:  1.89   Class :character
##  Median :105.0   Median : 15.09   Mode  :character
##  Mean   :108.7   Mean   : 36.26
##  3rd Qu.:118.0   3rd Qu.: 43.86
##  Max.   :450.0   Max.   :936.66
##                  NA's   :2527
```

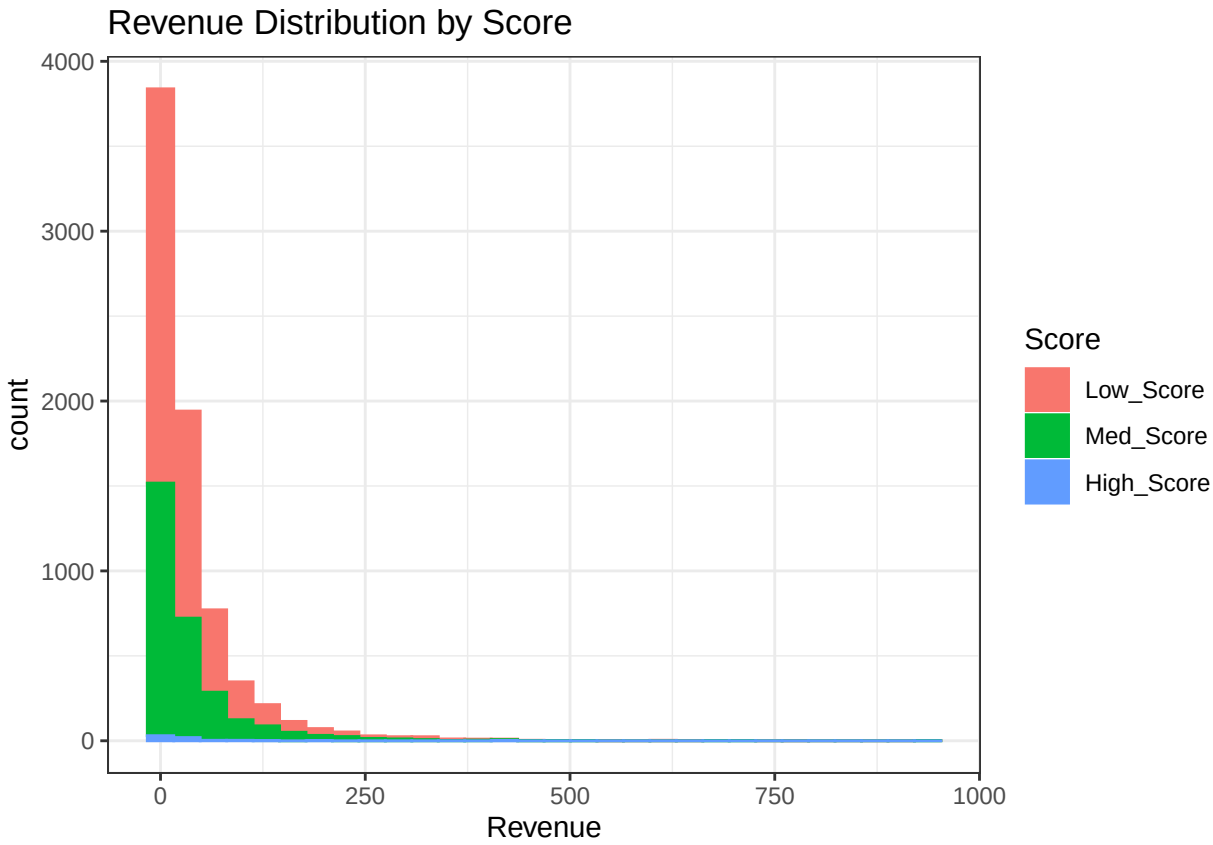
```
Newdf$Score <-
  cut(df$Score, breaks=c(-Inf, 7, 8.5, Inf),
      labels=c("Low_Score","Med_Score","High_Score"))
```

```
a <- table(Newdf$Revenue, Newdf$Score)
```

```
c <- ggplot(Newdf, aes(x=Revenue, fill=Score, color=Score)) +
  geom_histogram() + labs(title="Revenue Distribution by Score")
c + theme_bw()
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 2527 rows containing non-finite values (stat_bin).
```

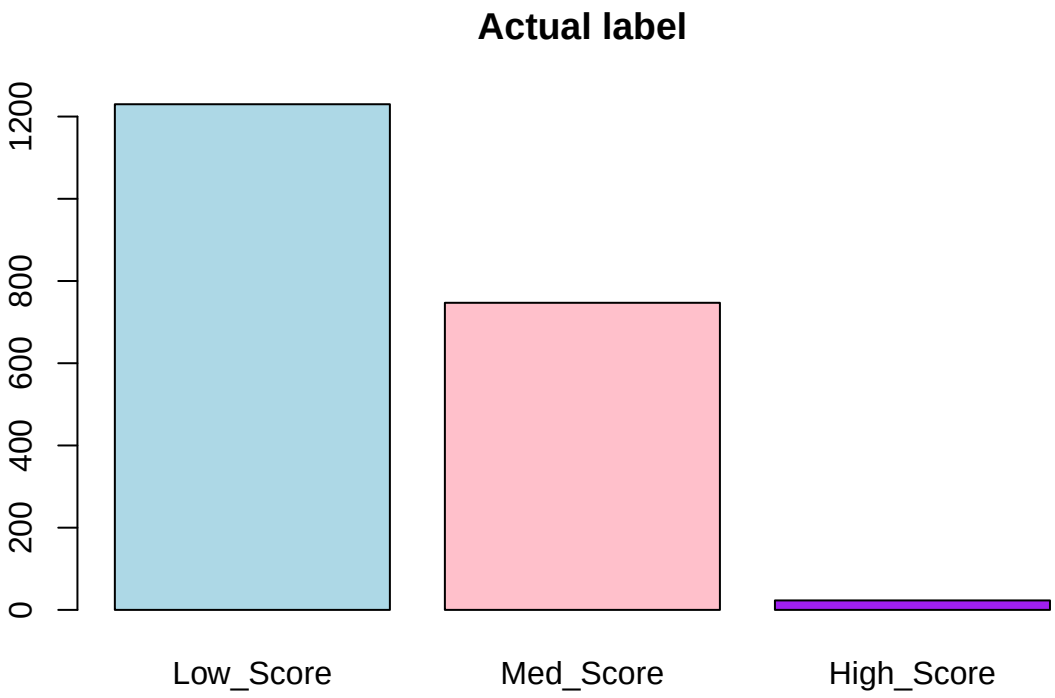



```
## ---------------------Now we can split the data--------------------:
n <- round(nrow(Newdf)/5)
s <- sample(1:nrow(Newdf), n)
```

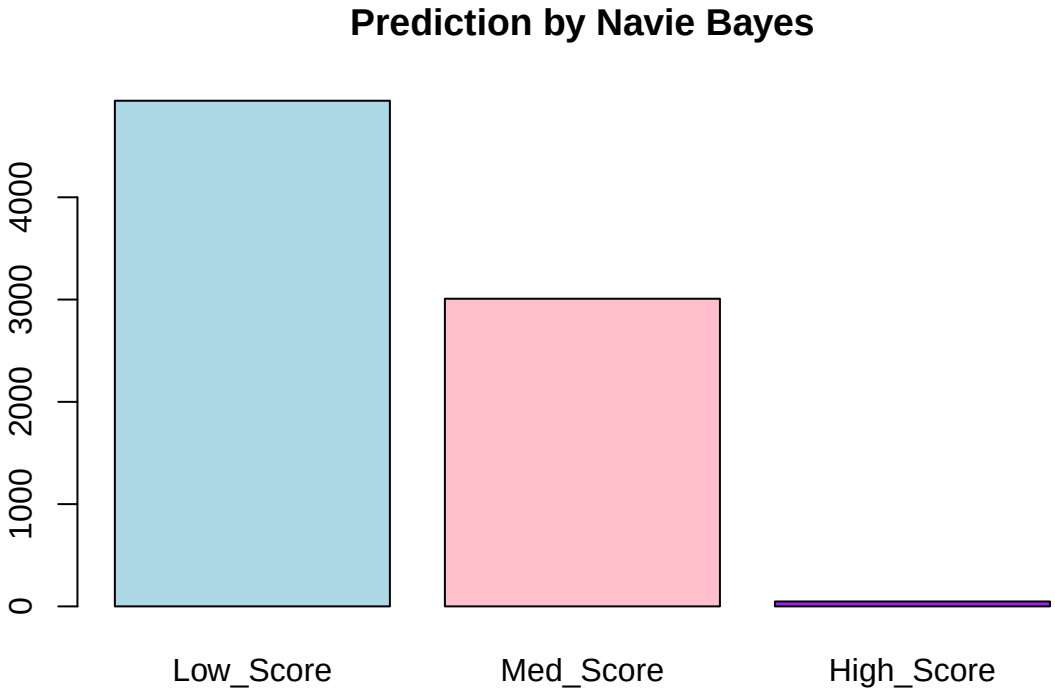
```
TestSet <- Newdf[s,]
#The trainng set is the not sample
```

```
TrainSet <- Newdf[-s,]
```

```
Test_label <- TestSet$Score
Test_num <- TestSet[,-c(1,2,3,4,5,6,7,8,9,11)]
Train_label <- TrainSet$Score
Train_num <- TrainSet[,-c(1,2,3,5,6,7,8,9,11)]
plot(Test_label, col = c('lightblue','pink','purple'), main = 'Actual label')
```




```
plot(Train_label, col = c('lightblue','pink','purple'), main = 'Prediction by Navie Bayes')
```



```r
## Use NB classifier:
NBclassfier <- naiveBayes(Score ~.,data=Train_num)
NBClassifier_Prediction <- predict(NBclassfier, Test_num)
```

```
## Warning in predict.naiveBayes(NBclassfier, Test_num): Type mismatch between
## training and new data for variable 'Revenue'. Did you use factors with numeric
## labels for training, and numeric values for new data?
```

```r
NBclassfier
```

```
##
## Naive Bayes Classifier for Discrete Predictors
##
## Call:
## naiveBayes.default(x = X, y = Y, laplace = laplace)
##
## A-priori probabilities:
## Y
##  Low_Score  Med_Score High_Score
##      0.618      0.376      0.006
##
## Conditional probabilities:
##             Revenue
## Y                [,1]     [,2]
##   Low_Score  35.86207 60.77688
##   Med_Score  37.63833 64.94681
##   High_Score 35.40514 75.54795
```

```r
heatmap(table(NBClassifier_Prediction, Test_label), Rowv = NA, Colv = NA, main = "Prediction by using Na
```

Prediction by using Naive Bayes