



L LOVELY
P ROFESSIONAL
U NIVERSITY

Course: DATA EXPLORATION AND PREPARATION

Course Code: CAP482

CA 2

Dated: - 24/Apr/2024

Submitted by

Name: Anureet Kaur

Roll No:9

Reg:12222064

Section: DE419, Group: 1

Submitted to

Ms. Ranjit Kaur Walia

UID: 28632

Assistant Professor

SCA, LPU


Lovely Faculty of Technology & Sciences

School of Computer Applications

Lovely Professional University

Punjab

Some steps of cleaning the dataset:



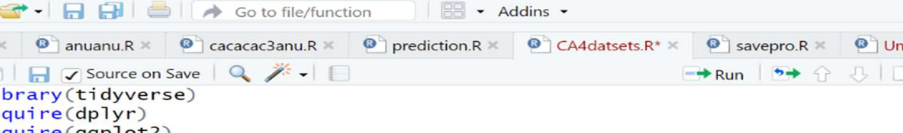
The screenshot shows the RStudio interface with the following elements:

- Top Menu Bar:** File, Edit, Code, View, Plots, Session, Debug, Profile, Tools, Help.
- Top Toolbar:** Includes icons for saving, running, and navigating, along with a "Go to file/function" search bar and an "Addins" dropdown.
- Tab Bar:** Displays several open files: anuanu.R, cacacac3anu.R, prediction.R, CA4datsets.R*, savepro.R, and the active file, Untitled1*. The IMDb dataset is also visible in the background.
- Source Editor:** Contains the following R code:


```
1 #cleaning of data....
2 View(imdb)
3 summary(imdb)
4 library(dplyr)
5 library(tidyverse)
6 imdb <- as.integer(imdb$runtime)
7
8 summary(imdb)
9 rm(imdb)
10
11 imdb <- imdb %>%
12   mutate(certificate = replace(certificate, certificate=="PG", 0))
13 View(imdb)
14
15 summary(imdb)
16 str(imdb)
17 rm(imdb)
18 imdb <- imdb %>%
19   separate(genre, into = c("genre1", "genre2", "genre3"), sep = ",")
20 view(imdb)
21
22 # Convert columns to appropriate data types
23 imdb <- type.convert(imdb, as.is = TRUE)
```
- Status Bar:** Shows the current line as 19:3 and the context as (Top Level).
- Bottom Right Corner:** Labeled "R Script".

Q1) What are the total ratings of all movies?

Ans.) CODE:

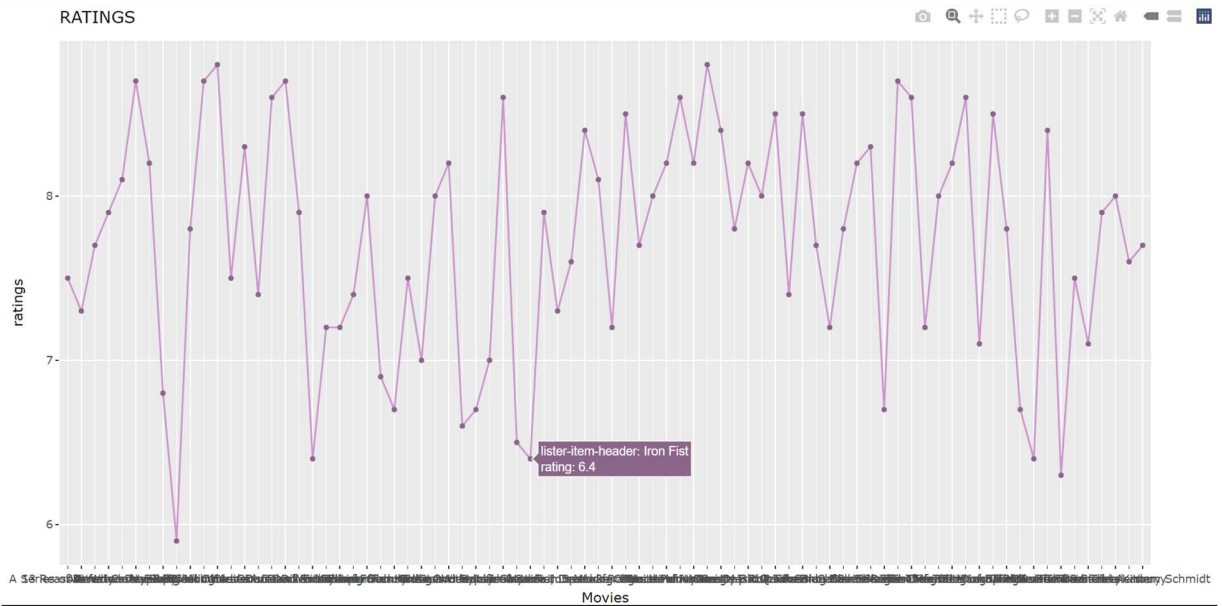


The screenshot shows the RStudio IDE with the following elements:

- Menu Bar:** File, Edit, Code, View, Plots, Session, Build, Debug, Profile, Tools, Help.
- Toolbar:** Includes icons for saving, running, and other standard IDE functions.
- File Explorer:** Shows open files: anuanu.R, anuanu.R, cacaca3anu.R, prediction.R, CA4datsets.R, savepro.R, and Untitled1.
- Source Editor:** Contains the following R code:

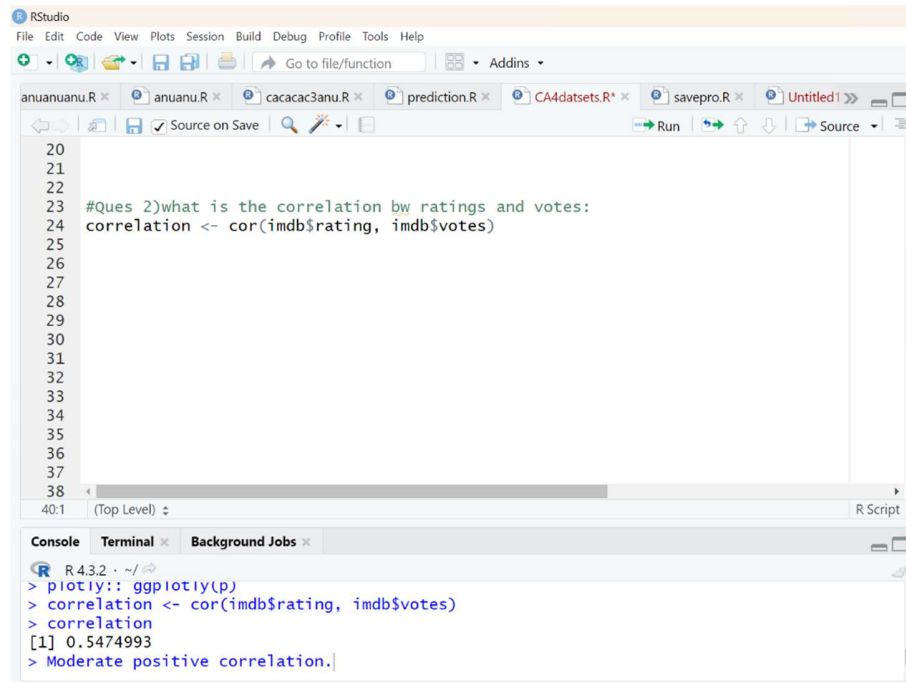

```
1 library(tidyverse)
2 require(dplyr)
3 require(ggplot2)
4 require(ggpubr)
5 require(gridExtra)
6 read.csv(file.choose("imdb"),header = T)
7 View(imdb)
8 #QUES 1)Ratings of movies:
9 library(plotly)
10 anu <- ggplot(data = imdb, aes(x = `lister-item-header`, y = rating, group = 1)) +
11   geom_line(color="plum3")+
12   geom_point(size=1,color="plum4")+
13   labs(title = "RATINGS",
14        x="Movies",
15        y="ratings")
16 plotly::ggplotly(anu)
17
```
- Console:** Shows the command prompt at line 17:1 (Top Level).
- Status Bar:** Indicates the current file is 'R Scrip'.

OUTPUT:



Q2) What is the correlation between ratings and votes.

Ans) CODE & OUTPUT



The screenshot shows the RStudio interface. The script editor contains the following code:

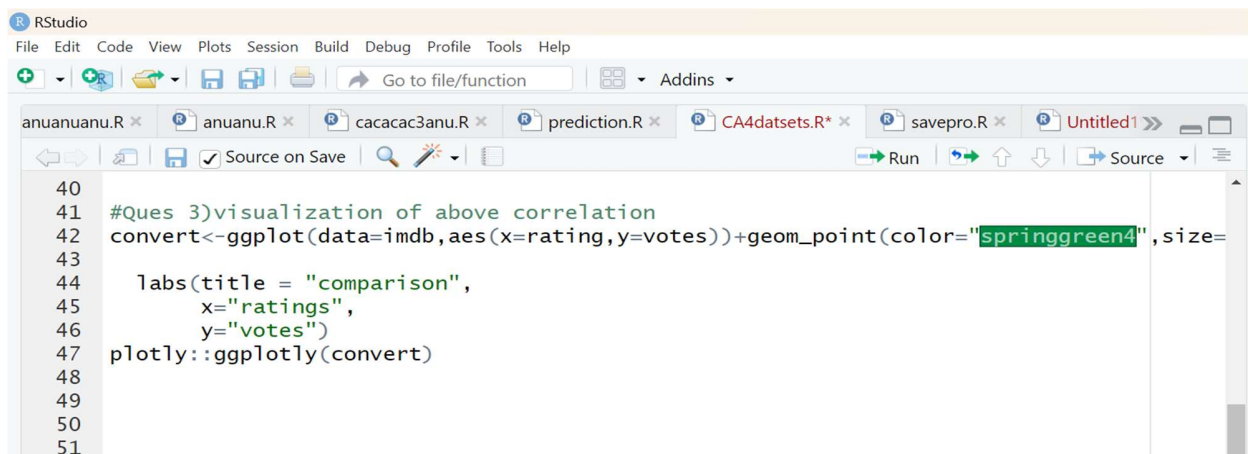
```
20
21
22
23 #Ques 2)what is the correlation bw ratings and votes:
24 correlation <- cor(imdb$rating, imdb$votes)
25
26
27
28
29
30
31
32
33
34
35
36
37
38
```

The console shows the execution of the code:

```
R 4.3.2 ~ /
> plotly::ggplotly(p)
> correlation <- cor(imdb$rating, imdb$votes)
> correlation
[1] 0.5474993
> Moderate positive correlation.
```

Ques 3) show the visualization of correlation.

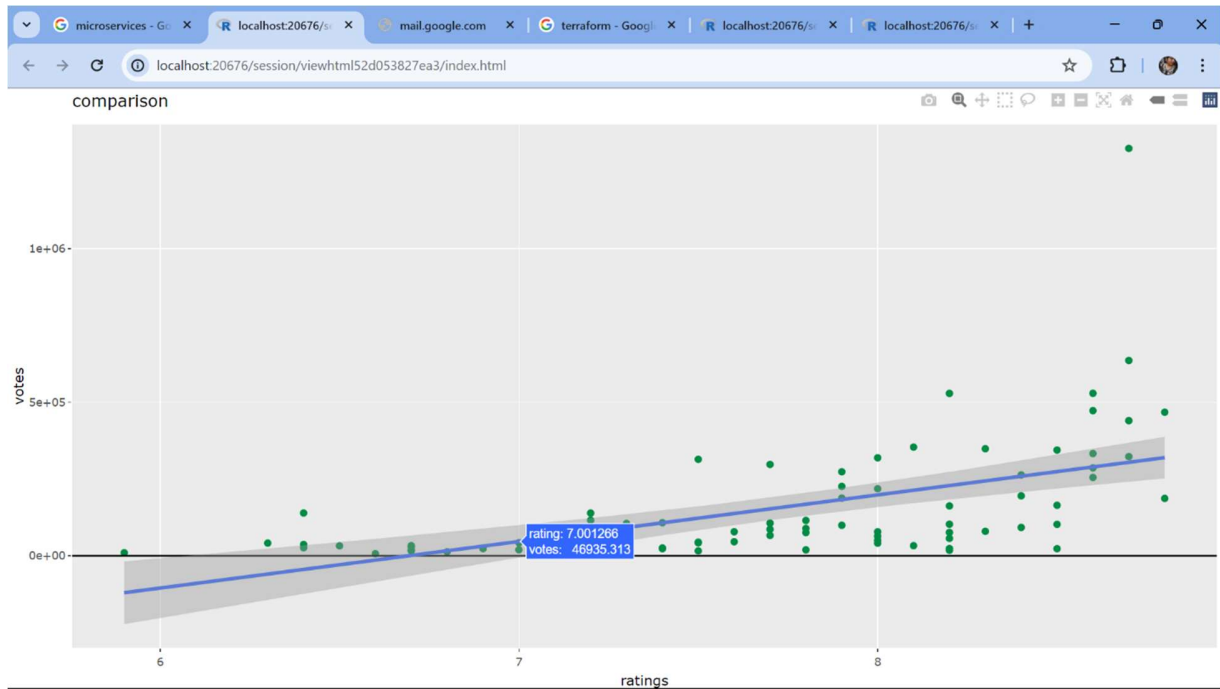
Ans) Code:



The screenshot shows the RStudio interface. The script editor contains the following code:

```
40
41 #Ques 3)visualization of above correlation
42 convert<-ggplot(data=imdb,aes(x=rating,y=votes))+geom_point(color="springgreen4",size=
43
44   labs(title = "comparison",
45     x="ratings",
46     y="votes")
47 plotly::ggplotly(convert)
48
49
50
51
```

OUTPUT:



QUES 4) What is the runtime of movies from MIN-MAX.

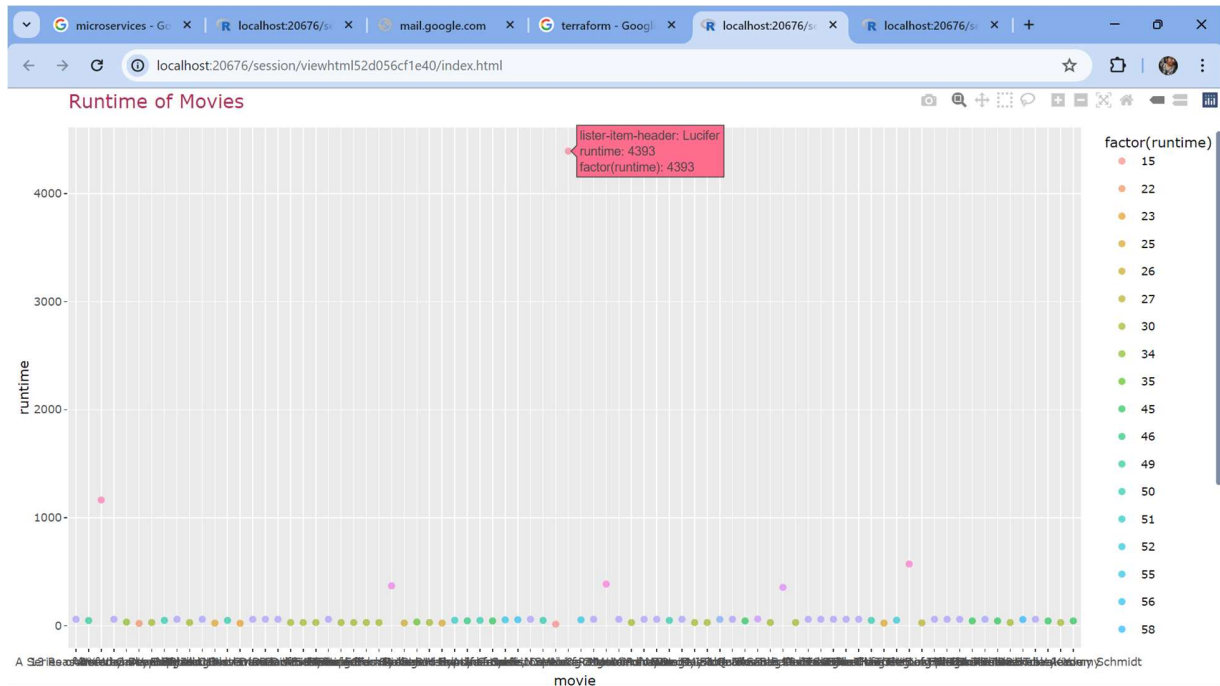
ANS) code:

```

RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
anuanuanu.R x anuanu.R x cacacac3anu.R x prediction.R x CA4datsets.R* x savepro.R x Untitled1 >>
Source on Save Run
54
55
56
57
58 #ques 4)time duration of movies:(MIN-MAX)
59 p <- ggplot(data = imdb, aes(x = `lister-item-header`, y = runtime, col = factor(runtime)))
60   geom_point(alpha = 3/5) +
61   labs(title = "Runtime of Movies",
62        x="movie",
63        y="runtime")+
64   theme(plot.title = element_text(size = 15, color = "maroon"))
65 plotly:: ggplotly(p)
66
67
68

```

OUTPUT:



Ques 5) Create a boxplot of Relationship between ratings and certificates.

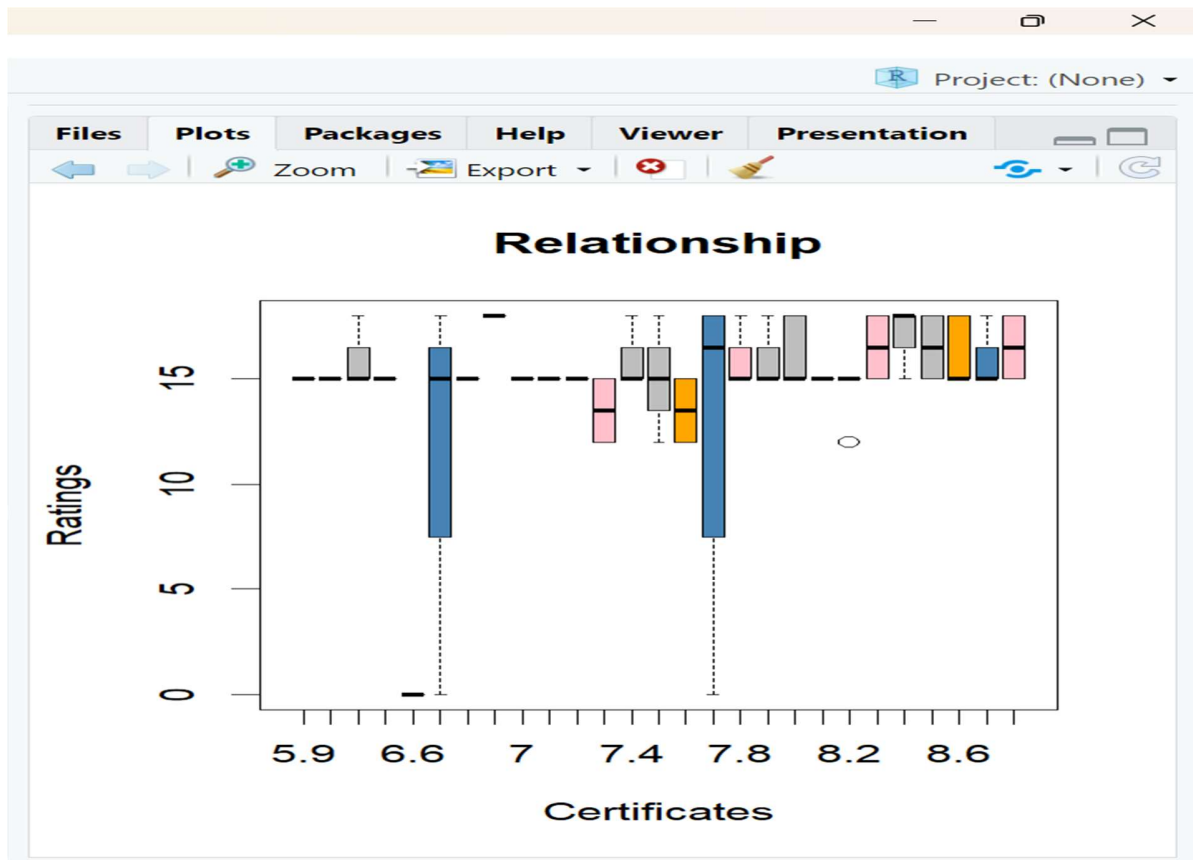
Ans) CODE:

```

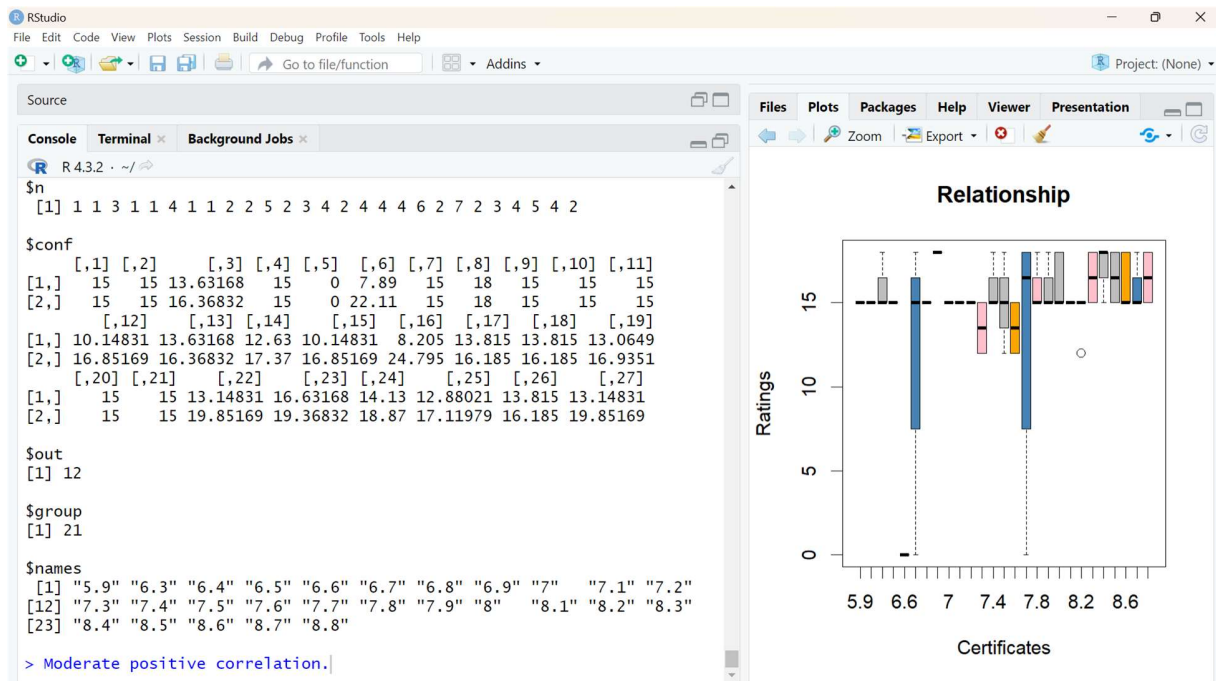
86 |
87 # Ques 6) Boxplot of the relation between certificate and rating:
88 boxplot(certificate ~ rating, data = imdb, main = "Relationship",
89         xlab = "Certificates", ylab = "Ratings",
90         col = c("steelblue", "pink", "grey", "grey", "orange"),
91         border = "black")
92
86:1 (Top Level) R Scrip

```

OUTPUT:

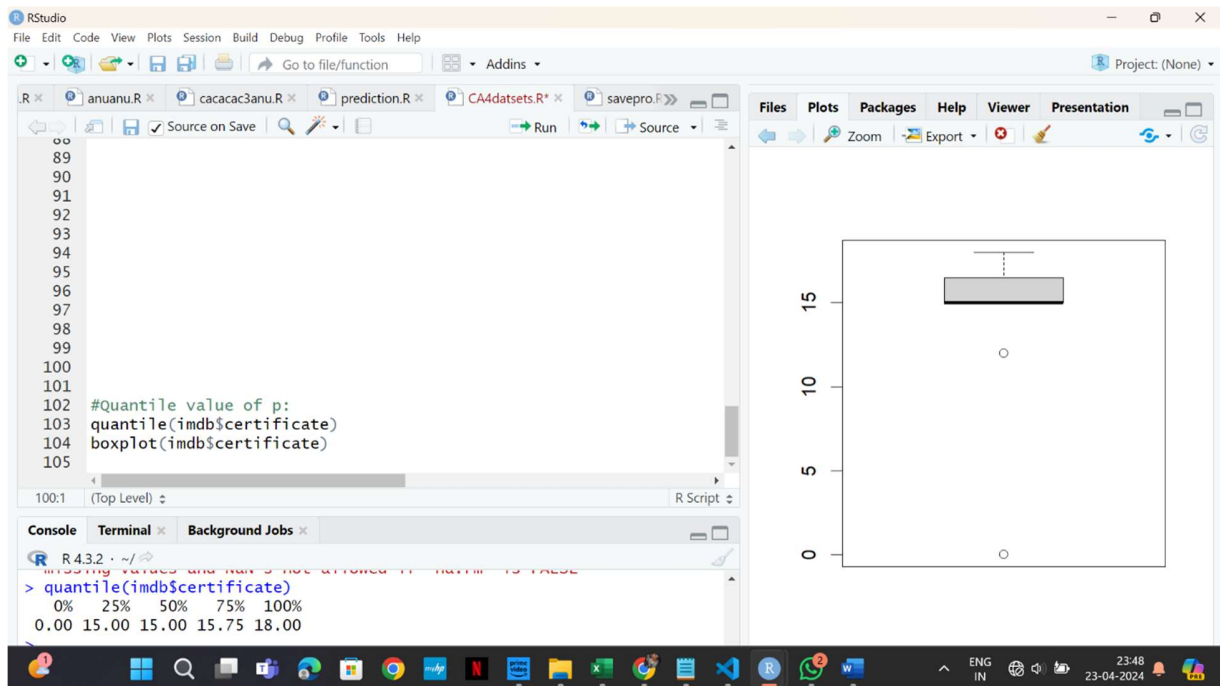


Another output:



QUES 6) What will be the quantile value of column certificate? Visualize

Ans)



Ques 7) What will be the prediction?

Ans) code & output:

The screenshot shows the RStudio environment. The script editor contains the following R code:

```
103  
104  
105 #Ques7) what will be the prediction ?  
106 mrm2=lm(votes~rating+certificate+runtime,data=imdb)  
107 sales1= -1.034+1.453*200+ 4.283*100+2.979*150  
108  
109  
110 |  
111  
112  
110:1 (Top Level) ↕ R Script
```

The console shows the execution of the code:

```
> mrm2  
  
Call:  
lm(formula = votes ~ rating + certificate + runtime, data = imdb)  
  
Coefficients:  
(Intercept)      rating  certificate      runtime  
-1.034e+06    1.453e+05    4.283e+03    2.979e+01  
  
> sales= -1.034+1.453*200+ 4.283*100+2.979*150  
> sales1= -1.034+1.453*200+ 4.283*100+2.979*150  
> sales1  
[1] 1164.716  
> |
```

Ques 8) What is the maximum runtime of all the movies?

Ans) Code & Output:

RStudio interface showing the following R code in the script editor:

```
90  
91 #Ques8)What is the maximum runtime of movies:  
92 reet1<- imdb %>%  
93   group_by(runtime) %>%  
94   summarise(Movies = n())  
95 max_genre1 <- reet1$runtime[which.max(reet1$Movies)]  
96  
97  
98  
99
```

The console output shows a tibble with 27 rows and 2 columns: runtime and Movies. The first 10 rows are displayed:

	runtime	Movies
1	15	1
2	22	1
3	23	1
4	25	3
5	26	1
6	27	1
7	30	17
8	34	1
9	35	1
10	45	6

i 17 more rows
Use `print(n = ...)` to see more rows

Ques 9) What is the average and median value of all the votings?

Ans) Code & Output:

RStudio interface showing the following R code in the script editor:

```
117  
118  
119  
120 #Ques 9) What is the average value and median of all the votings:  
121 mean(imdb$votes)  
122 median(imdb$votes)  
123  
124  
125  
126  
127  
128  
129
```

The console output shows the results of the calculations:

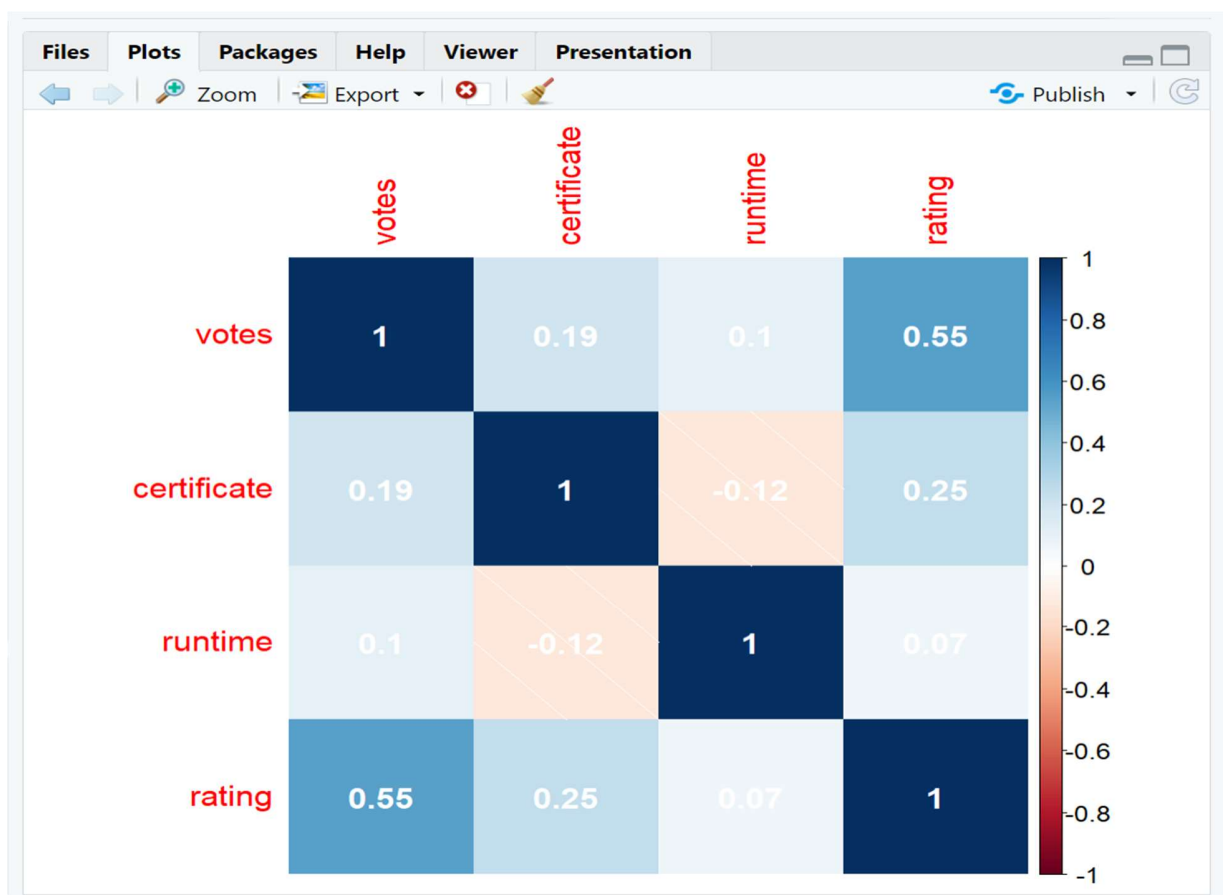
```
> mean(imdb$votes)  
[1] 155994.6  
> median(imdb$votes)  
[1] 83033.5  
>
```

Ques 10) Correlogram of all the numeric value columns:

CODE:

```
123  
124  
125  
126  
127 #Ques 10) Corrplot of all the numeric columns  
128 library(corrplot)  
129 d2<- imdb[, c("votes","certificate","runtime","rating")]  
130 correlation_matrix <- cor(d2)  
131 corrplot(correlation_matrix, method = "shade", addCoef.col = "white", interactive = TRUE)  
132  
133  
134 |  
135  
136  
137  
138
```

OUTPUT:

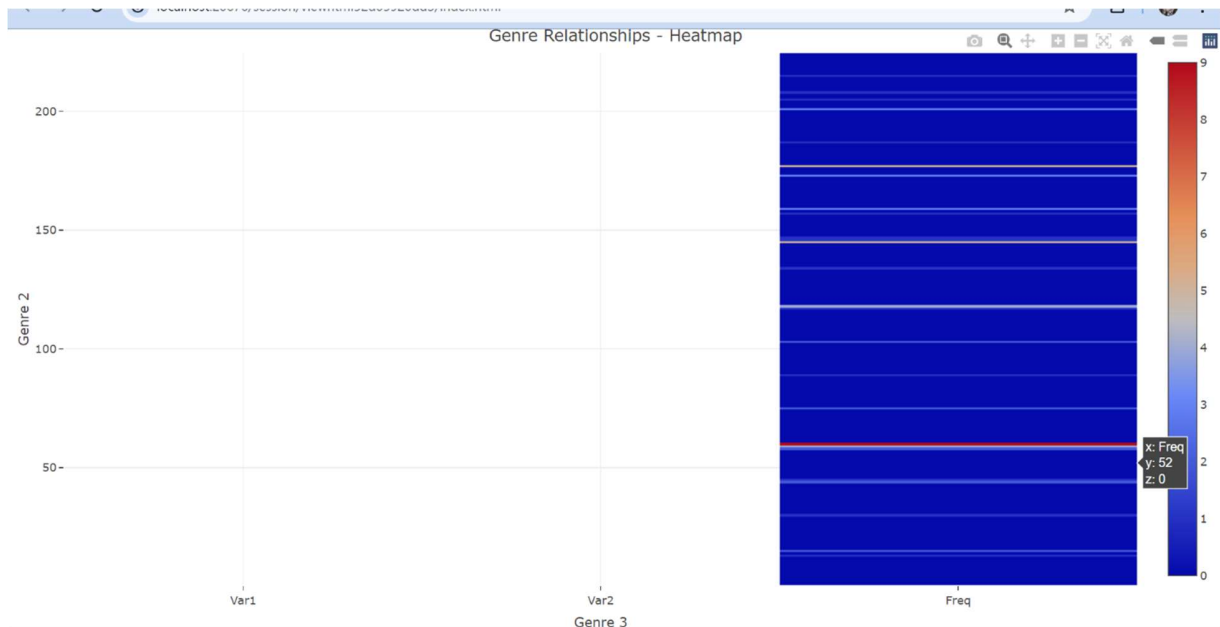


Ques 11) What is the relationship and distribution of genre 1 and genre 2.

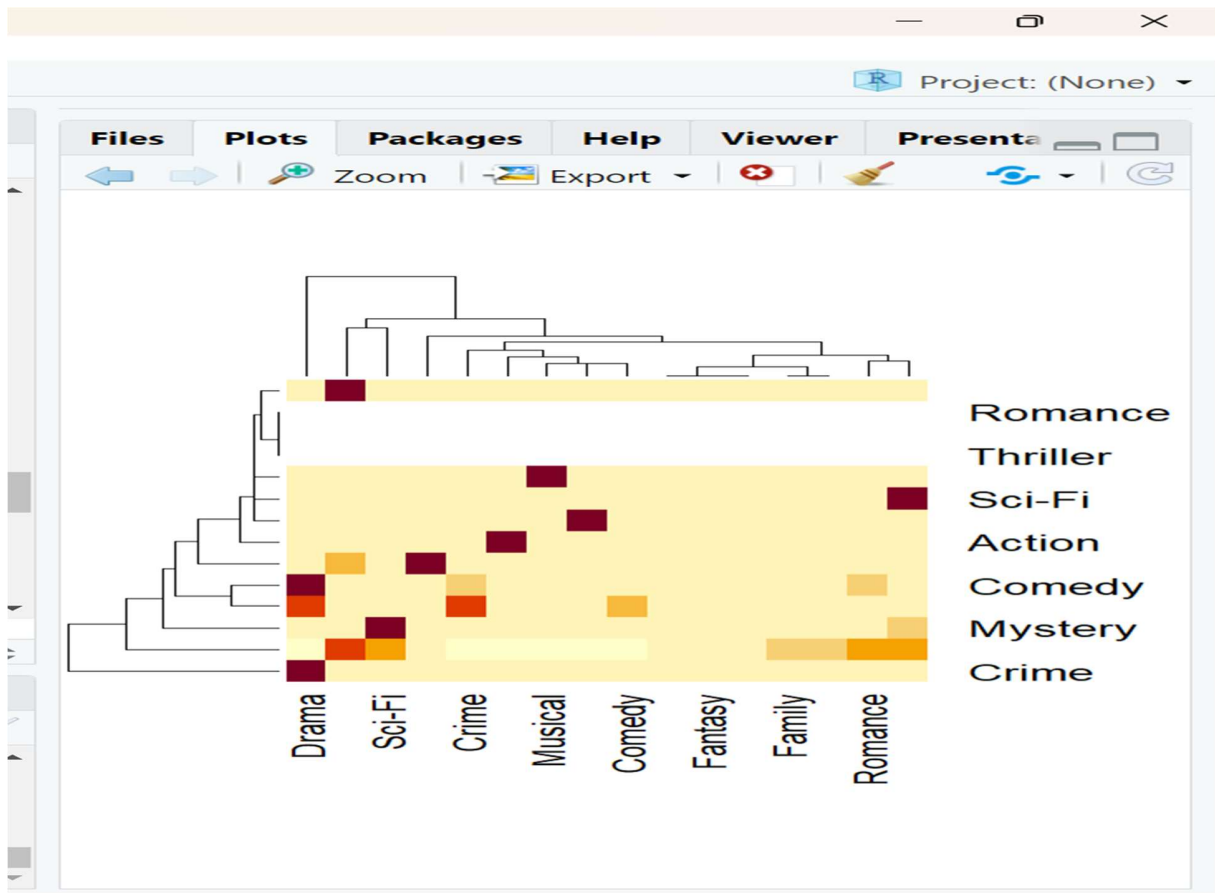
Ans)CODE:

```
RStudio
File Edit Code View Plots Session Build Debug Profile Tools Help
Go to file/function
Addins
test.R anuanuanu.R anuanu.R cacacac3anu.R prediction.R CA4datasets.R savepro.R Untitled1* imdb
Source on Save Run
133 genre2_distribution <- imdb %>%
134   group_by(genre2) %>%
135   summarise(count = n()) %>%
136   arrange(desc(count))
137 ggplot(genre2_distribution, aes(x = reorder(genre2, -count), y = count)) +
138   geom_bar(stat = "identity") +
139   labs(title = "Genre Distribution - Genre 2")
140 genre3_distribution <- imdb %>%
141   group_by(genre3) %>%
142   summarise(count = n()) %>%
143   arrange(desc(count))
144 ggplot(genre3_distribution, aes(x = reorder(genre3, -count), y = count)) + geom_bar(stat = "identity") +
145   labs(title = "Genre Distribution - Genre 3")
146 genre_relationships <- table(imdb$genre2, imdb$genre3)
147 heatmap(genre_relationships)
148 genre_relationships_df <- as.data.frame(genre_relationships)
149 heatmap_plot <- plot_ly(z = ~as.matrix(genre_relationships_df),
150   x = colnames(genre_relationships_df),
151   y = rownames(genre_relationships_df),
152   type = "heatmap")
153 heatmap_plot <- heatmap_plot %>%
154   layout(title = "Genre Relationships - Heatmap",
155     xaxis = list(title = "Genre 3"),
156     yaxis = list(title = "Genre 2"))
157 heatmap_plot
144:75 (Top Level) R Script
Console
```

OUTPUT:HEATMAP

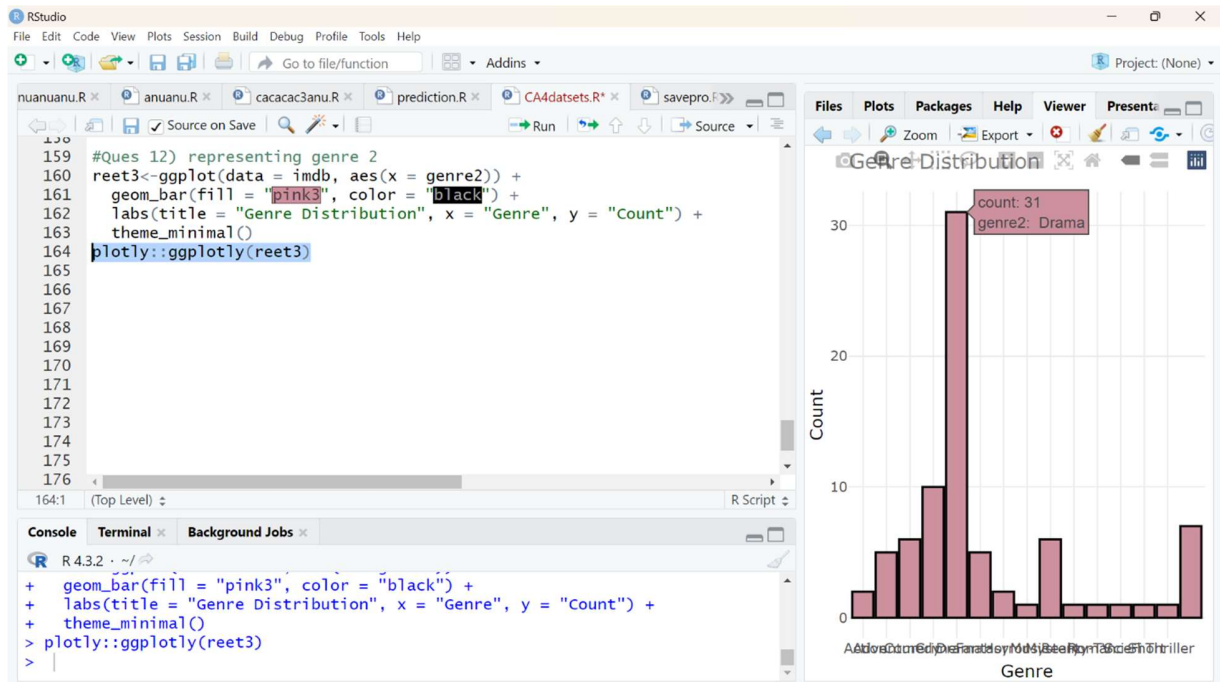


Without using plotly:



Ques 12) representing barplot of genre 2.

Ans- Code & output:



Ques-13) representing data analysis of all columns:

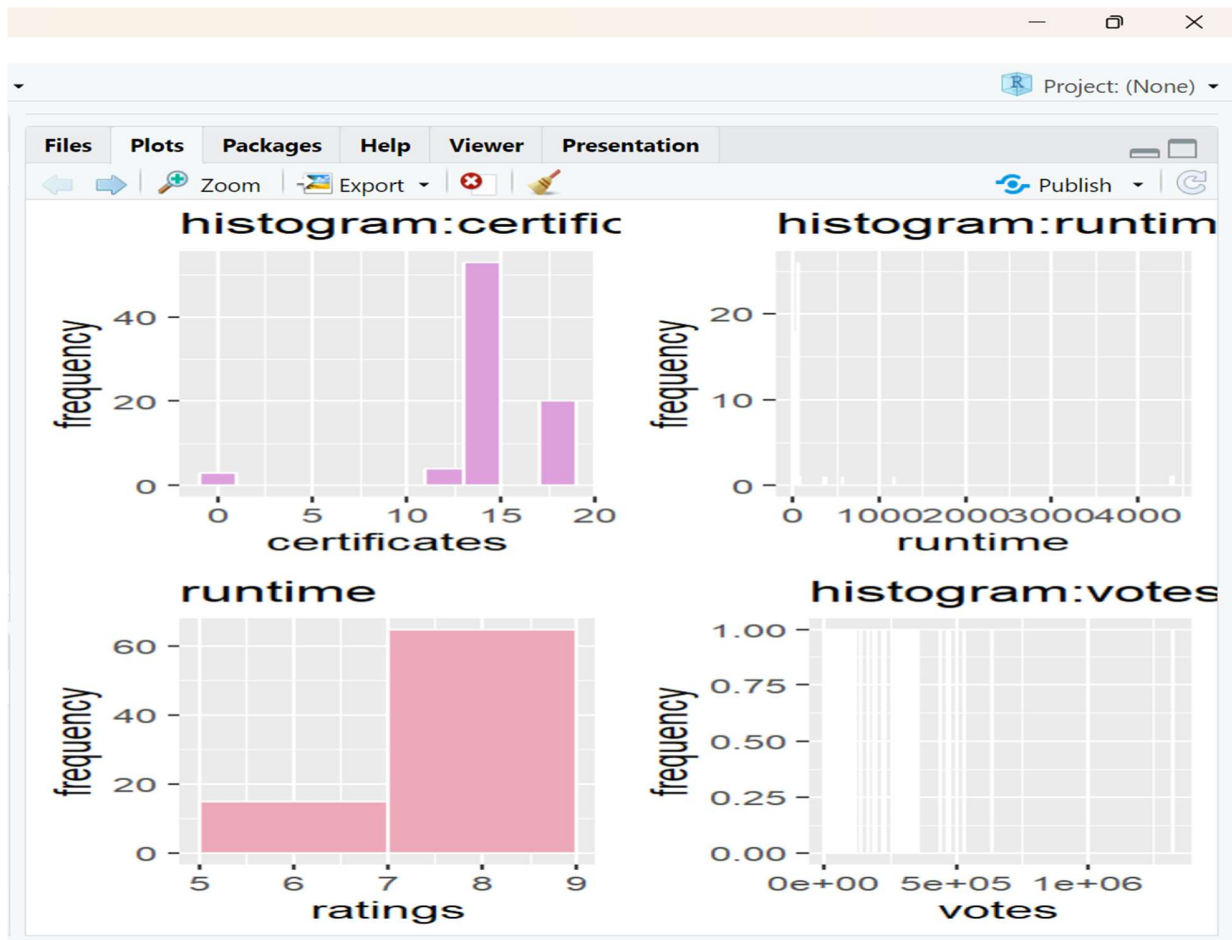
Ans) CODE:

```

165 #Ques 13) analysis of each column:
166 hist_plot_cer<-ggplot(imdb,aes(x=certificate))+
167   geom_histogram(binwidth=2,fill="plum",color="white")+labs(title = "histogram:certificates",x="certificates",y='
168 hist_plot_run<-ggplot(imdb,aes(x=runtime))+
169   geom_histogram(binwidth=4,fill="steelblue",color="white")+labs(title = "histogram:runtime",x="runtime",y="frequ
170 hist_plot_rat<-ggplot(imdb,aes(x=rating))+
171   geom_histogram(binwidth=2,fill="pink2",color="white")+labs(title = "runtime",x="ratings",y="frequency")
172 hist_plot_vote<-ggplot(imdb,aes(x=votes))+
173   geom_histogram(binwidth=5,fill="black",color="white")+labs(title = "histogram:votes",x="votes",y="frequency")
174 grid.arrange(hist_plot_cer,hist_plot_run,hist_plot_rat,hist_plot_vote,ncol=2)
175
176 |
177
178
179

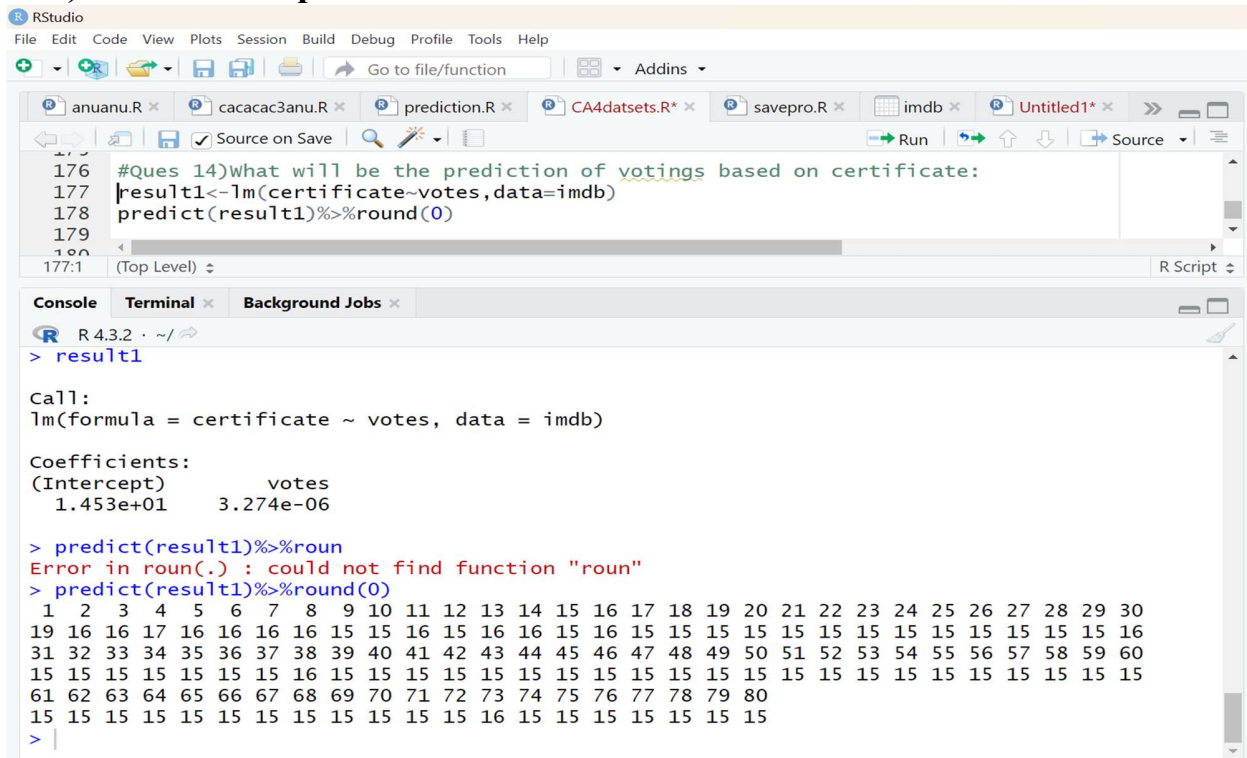
```

OUTPUT:



Ques 14) What will be the prediction of voting based on certificate?

Ans) Code and output:



RStudio interface showing the code for Ques 14 and its output in the console.

```
#Ques 14)What will be the prediction of votings based on certificate:
result1<-lm(certificate~votes,data=imdb)
predict(result1)%>%round(0)
```

Console output:

```
R 4.3.2 ~ /
> result1

Call:
lm(formula = certificate ~ votes, data = imdb)

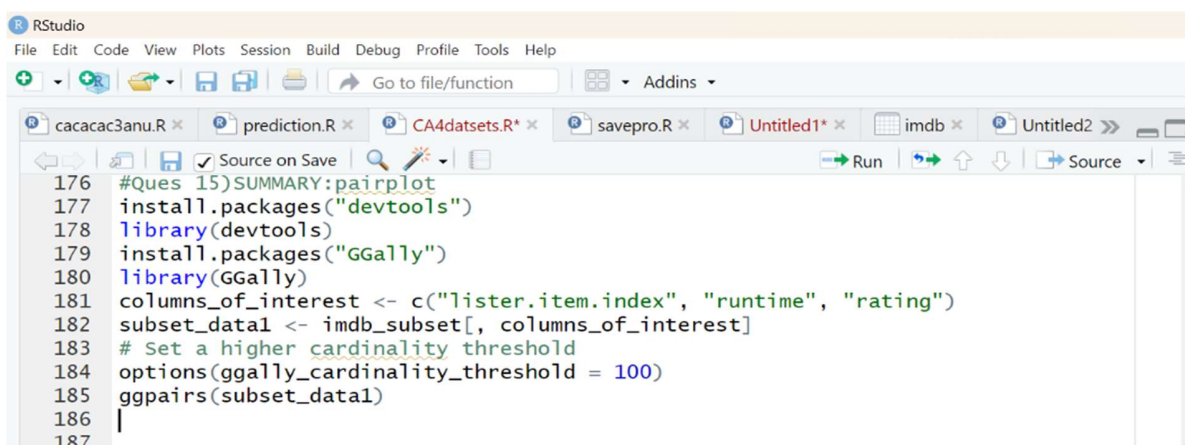
Coefficients:
(Intercept)      votes 
 1.453e+01     3.274e-06 

> predict(result1)%>%roun
Error in roun(.) : could not find function "roun"
> predict(result1)%>%round(0)
 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
19 16 16 17 16 16 16 16 15 15 16 15 16 16 15 16 15 15 15 15 15 15 15 15 15 15 15 15 16
31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
15 15 15 15 15 15 15 16 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15
61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80
15 15 15 15 15 15 15 15 15 15 15 15 16 15 15 15 15 15 15 15
```

Ques 15) How the columns are related with each other.

Ans)

CODE:



RStudio interface showing the code for Ques 15.

```
#Ques 15)SUMMARY:pairplot
install.packages("devtools")
library(devtools)
install.packages("GGally")
library(GGally)
columns_of_interest <- c("listner.item.index", "runtime", "rating")
subset_data1 <- imdb_subset[, columns_of_interest]
# Set a higher cardinality threshold
options(ggally_cardinality_threshold = 100)
ggpairs(subset_data1)
```

OUTPUT:

