DAV ASSIGNMENT1

Group Members [Team 9]:

Akhil Patil
Palash Nandecha
Anwer Sharjeel
Francklin Valcourt

DAV Assignment1

Akhil Patil

February 23, 2019

Problem 1

Q 1.1 Import the df1 winter_olymic.csv. Identify the column names and dimension of the data. import the df1

```
df1 = read.csv("D:/Downloads/winter_olympic.csv")
```

check the column names for the df1

```
head(df1)

##   Rank                    NOC Gold Silver Bronze Total  Region
## 1    1       Â RussiaÂ (RUS)*   13     11      9    33 EURASIA
## 2    2        Â NorwayÂ (NOR)   11      5     10    26  EUROPE
## 3    3        Â CanadaÂ (CAN)   10     10      5    25 NORTH_A
## 4    4 Â United StatesÂ (USA)    9      7     12    28 NORTH_A
## 5    5   Â NetherlandsÂ (NED)    8      7      9    24  EUROPE
## 6    6       Â GermanyÂ (GER)    8      6      5    19  EUROPE

colnames(df1)

## [1] "Rank"   "NOC"    "Gold"   "Silver" "Bronze" "Total"  "Region"
```

Check the dimension of df1

```
dim(df1)

## [1] 26  7
```

Q1.2 Data is currently sorted by Rank. Sort data by total medals and country. Assign sorted data to a new data frame.

Creating a new dataframe and assigning the newly sorted df1 to it.

```
newdata <- df1[order(df1$Total, df1$NOC),]
newdata

##    Rank                     NOC Gold Silver Bronze Total    Region
## 25   25        Â CroatiaÂ (CRO)    0      1      0     1    EUROPE
## 26   26     Â KazakhstanÂ (KAZ)    0      0      1     1   EURASIA
## 21   21       Â SlovakiaÂ (SVK)    1      0      0     1    EUROPE
## 20   20        Â UkraineÂ (UKR)    1      0      1     2   EURASIA
## 24   24      Â AustraliaÂ (AUS)    0      2      1     3 AUSTRALIA
## 19   19  Â Great BritainÂ (GBR)    1      1      2     4    EUROPE
## 23   23         Â LatviaÂ (LAT)    0      2      2     4   EURASIA
## 18   18        Â FinlandÂ (FIN)    1      3      1     5    EUROPE
## 8     8        Â BelarusÂ (BLR)    5      0      1     6   EURASIA
## 11   11         Â PolandÂ (POL)    4      1      1     6    EUROPE
## 15   15 Â Czech RepublicÂ (CZE)    2      4      2     8    EUROPE
## 22   22          Â ItalyÂ (ITA)    0      2      6     8    EUROPE
## 17   17          Â JapanÂ (JPN)    1      4      3     8      ASIA
## 16   16       Â SloveniaÂ (SLO)    2      2      4     8    EUROPE
## 13   13    Â South KoreaÂ (KOR)    3      3      2     8      ASIA
## 12   12          Â ChinaÂ (CHN)    3      4      2     9      ASIA
## 7     7    Â SwitzerlandÂ (SUI)    6      3      2    11    EUROPE
## 10   10         Â FranceÂ (FRA)    4      4      7    15    EUROPE
## 14   14         Â SwedenÂ (SWE)    2      7      6    15    EUROPE
## 9     9        Â AustriaÂ (AUT)    4      8      5    17    EUROPE
## 6     6        Â GermanyÂ (GER)    8      6      5    19    EUROPE
## 5     5    Â NetherlandsÂ (NED)    8      7      9    24    EUROPE
## 3     3         Â CanadaÂ (CAN)   10     10      5    25   NORTH_A
## 2     2         Â NorwayÂ (NOR)   11      5     10    26    EUROPE
## 4     4  Â United StatesÂ (USA)    9      7     12    28   NORTH_A
## 1     1        Â RussiaÂ (RUS)*  13     11      9    33   EURASIA
```

Q1.3 Compute the following statistics: a) What is the median number of gold, silver, bronze medals ? Also look at their mean.

MEDIAN

```
medianGoldMedals <- median(df1$Gold, na.rm = FALSE)
medianGoldMedals

## [1] 2.5

medianSilverMedals <- median(df1$Silver, na.rm = FALSE)
medianSilverMedals

## [1] 3

medianBronzeMedals <- median(df1$Bronze, na.rm = FALSE)
medianBronzeMedals

## [1] 2
```

MEAN

```
result.meanGold <- mean(df1$Gold, trim = 0, na.rm = FALSE)
result.meanGold

## [1] 3.807692

result.meanSilver <- mean(df1$Silver, trim = 0, na.rm = FALSE)
result.meanSilver

## [1] 3.730769

result.meanBronze <- mean(df1$Bronze, trim = 0, na.rm = FALSE)
result.meanBronze
```

```
## [1] 3.807692
```

b) For gold, look at summary stats, including:IQR, min, max, mean, var, sd, skew we get IQR, min, max, mean, meadian using Summary function. IQR is the difference between 75th and 25th percentile

```
summary(df1$Gold)

##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   0.000   1.000   2.500   3.808   5.750  13.000

IQR(df1$Gold)

## [1] 4.75
```

Thus IQR for gold medals is 5.750-1.000 = 4.750

we get variance of gold medals column from:

```
var(df1$Gold)

## [1] 14.64154
```

We get standard deviation of gold column from:

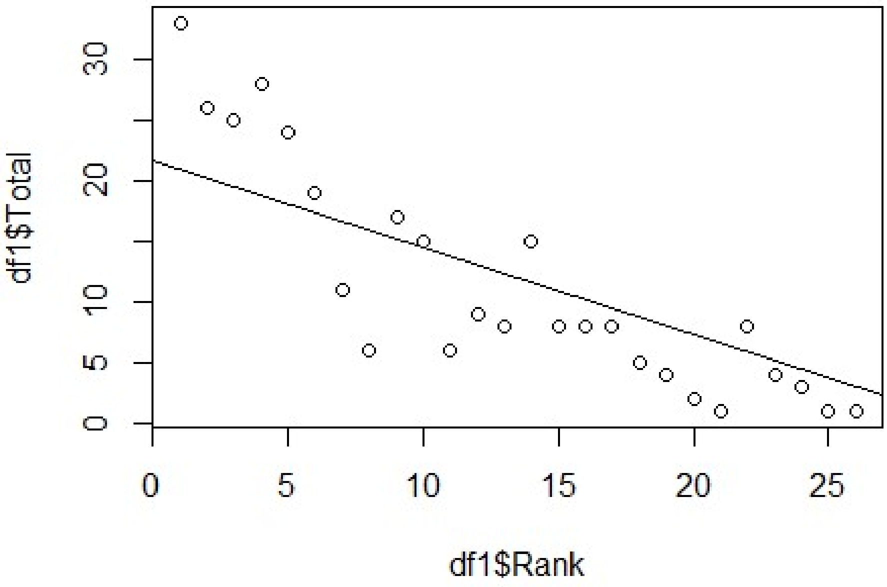
```
sd(df1$Gold)

## [1] 3.826426
```

we can get skewness of gold column by:

```
#install.packages("moments")
library(moments)

## Warning: package 'moments' was built under R version 3.5.2

skewness(df1$Gold)

## [1] 0.9322427
```

Q1.4 What is the correlation between Rank and Total medals? Is this expected or surprising?

```
cor( df1$Total,df1$Rank)

## [1] -0.874864

plot(df1$Rank, df1$Total)
linearRegression = lm(df1$Rank~df1$Total)
abline(linearRegression)
```



```
summary(linearRegression)

##
## Call:
## lm(formula = df1$Rank ~ df1$Total)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -9.3398 -0.9571  0.0055  2.9049  6.0967
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 21.64927    1.18240  18.310 1.32e-15 ***
## df1$Total   -0.71824    0.08117  -8.849 5.07e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.781 on 24 degrees of freedom
## Multiple R-squared:  0.7654, Adjusted R-squared:  0.7556
## F-statistic:  78.3 on 1 and 24 DF,  p-value: 5.065e-09
```

Summarizing the linear model of Rank vs Total considering Total as the independent variable and Rank as the dependent variable we get a p-value that is between 0 and 0.0001 which shows that total of number of medals has high impact on determing the Rank. Thus there exists a high correlation between Total number of Medal and Rank. It is expected that the value of correlation is negative towards -1 since as the value of total Medals increases the value of rank decreases following an inverse relation i.e if total number of highest medals is 100 that rank will be 1. This means the variable is changing in negative direction.

Problem 2 Q2.1 Import the df1 movies.csv. Look at the column names and dimension of the data

```
df2 = read.csv("D:/Downloads/movies.csv")
```

viewing column names of df1

```
colnames(df2)

## [1] "Rank"          "Movie"         "Release_Date" "Distributor"
## [5] "Genre"         "MPAA"          "Gross_Sales"  "Tickets_Sold"
```

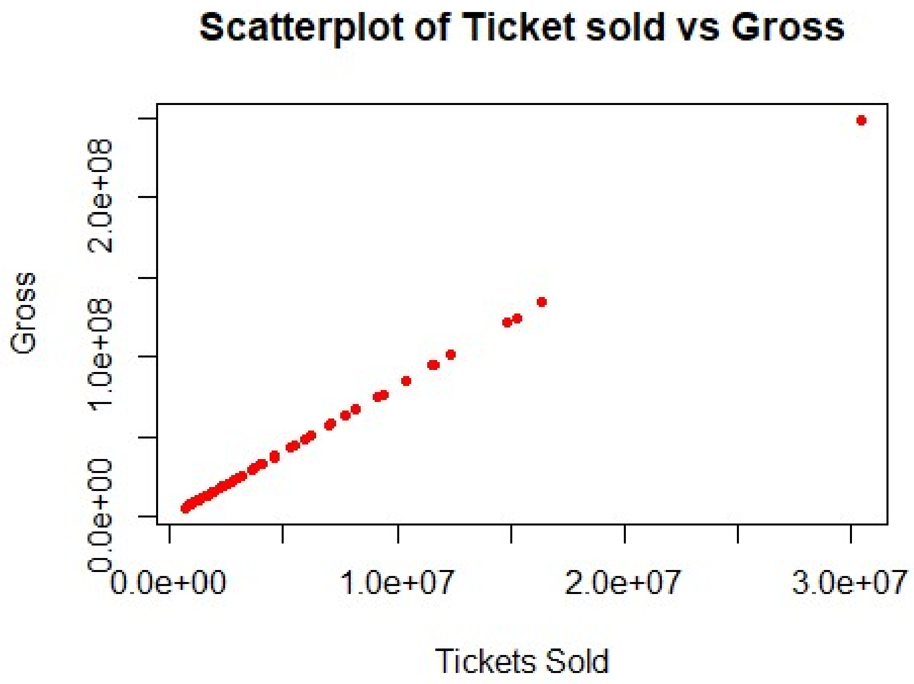
Analyzing dimension of data

```
dim(df2)

## [1] 50  8
```

Q 2.2

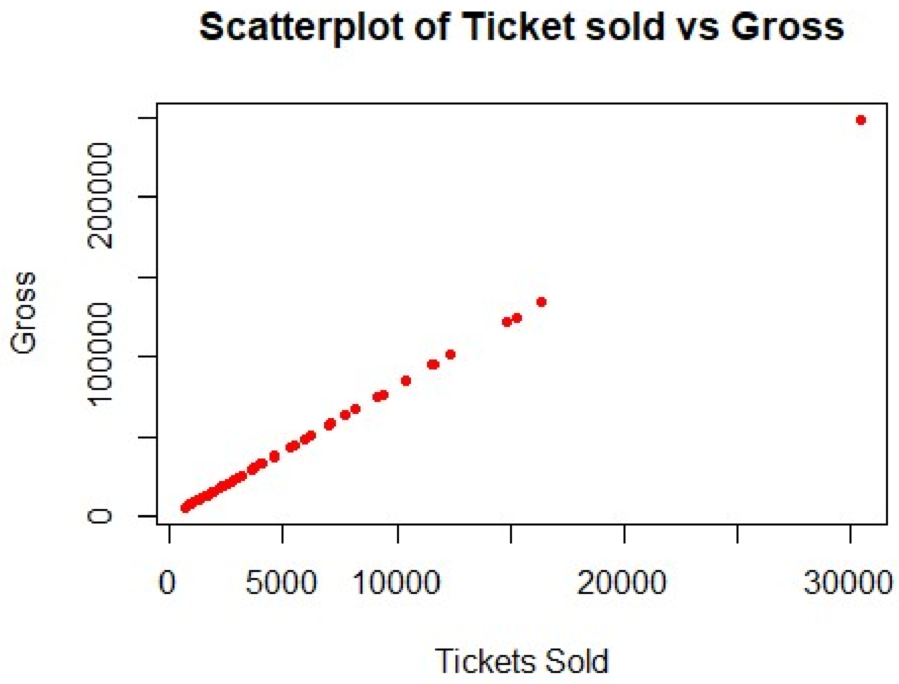
Obtain the following scatterplots a) Tickets Sold and Gross (Is the trend expected?) According to me this trend was expected, since more the tickets sold more will be the gross_sales

```
plot(df2$Tickets_Sold, df2$Gross_Sales, main="Scatterplot of Ticket sold vs G
ross", xlab= "Tickets Sold",ylab = "Gross", pch=20, col="red")
```



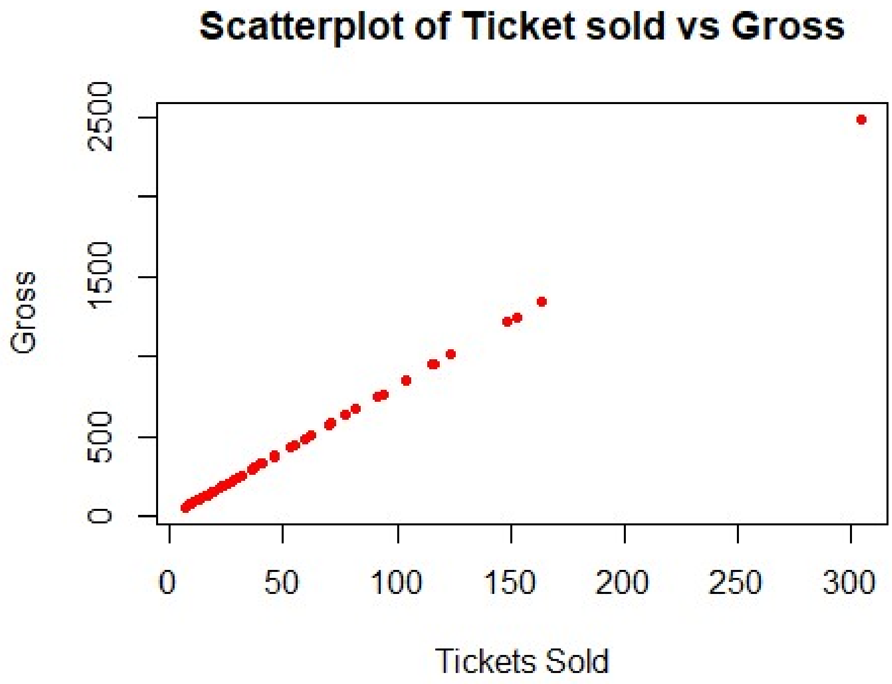
b) redo scatter plot, adjusting scales, divide by 1000

```
plot(df2$Tickets_Sold/1000, df2$Gross_Sales/1000, main="Scatterplot of Ticket
sold vs Gross", xlab= "Tickets Sold",ylab = "Gross", pch=20, col="red")
```



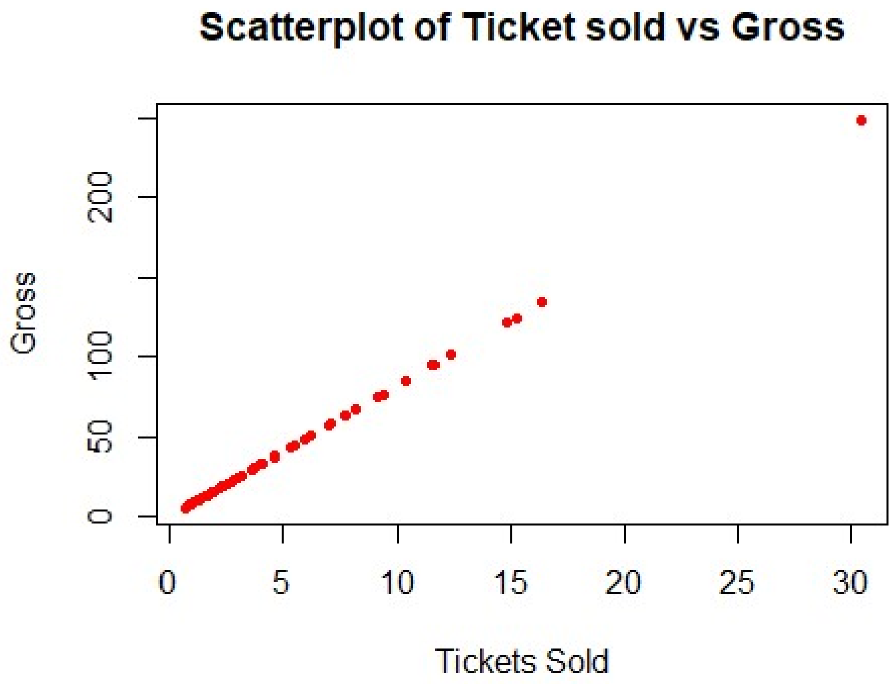
c) redo scatter plot, adjusting scales, divide by 100,000

```
plot(df2$Tickets_Sold/100000, df2$Gross_Sales/100000, main="Scatterplot of Ticket sold vs Gross", xlab= "Tickets Sold",ylab = "Gross", pch=20, col="red")
```



d) redo scatter plot, adjusting scales, divide by 1,000,000

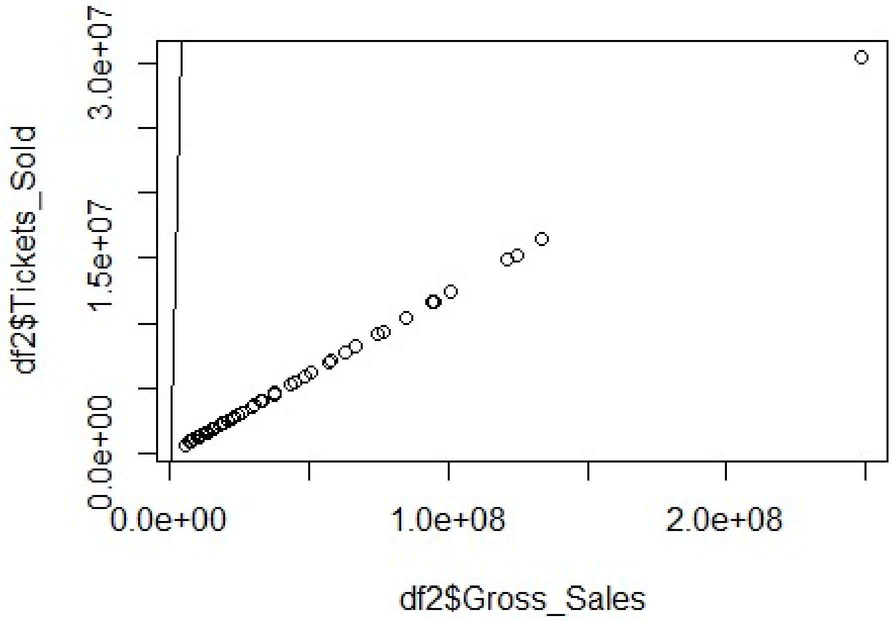
```
plot(df2$Tickets_Sold/1000000, df2$Gross_Sales/1000000, main="Scatterplot of Ticket sold vs Gross", xlab= "Tickets Sold",ylab = "Gross", pch=20, col="red"
)
```



Q2.3

What is the correlation between tickets sold and sales? Is this expected? This is expected since more the number of tickets sold for a particular movie, more will be the total gross sale for that movie.

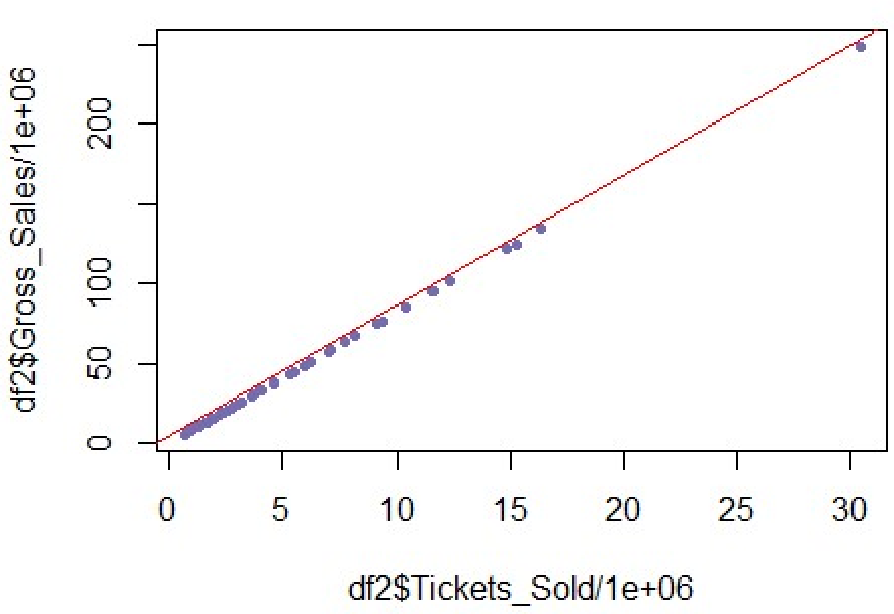
```r
plot(df2$Gross_Sales,df2$Tickets_Sold)
regression_model = lm(df2$Gross_Sales~df2$Tickets_Sold)
abline(regression_model)
```



```
summary(regression_model)

##
## Call:
## lm(formula = df2$Gross_Sales ~ df2$Tickets_Sold)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8109 -1.7869  0.2974  1.7460  3.9306
##
## Coefficients:
##                   Estimate Std. Error   t value Pr(>|t|)
## (Intercept)      3.980e+00  4.699e-01 8.471e+00 4.32e-11 ***
## df2$Tickets_Sold 8.160e+00  6.123e-08 1.333e+08  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.332 on 48 degrees of freedom
## Multiple R-squared:      1,  Adjusted R-squared:      1
## F-statistic: 1.776e+16 on 1 and 48 DF,  p-value: < 2.2e-16

cor(df2$Gross_Sales,df2$Tickets_Sold)

## [1] 1
```

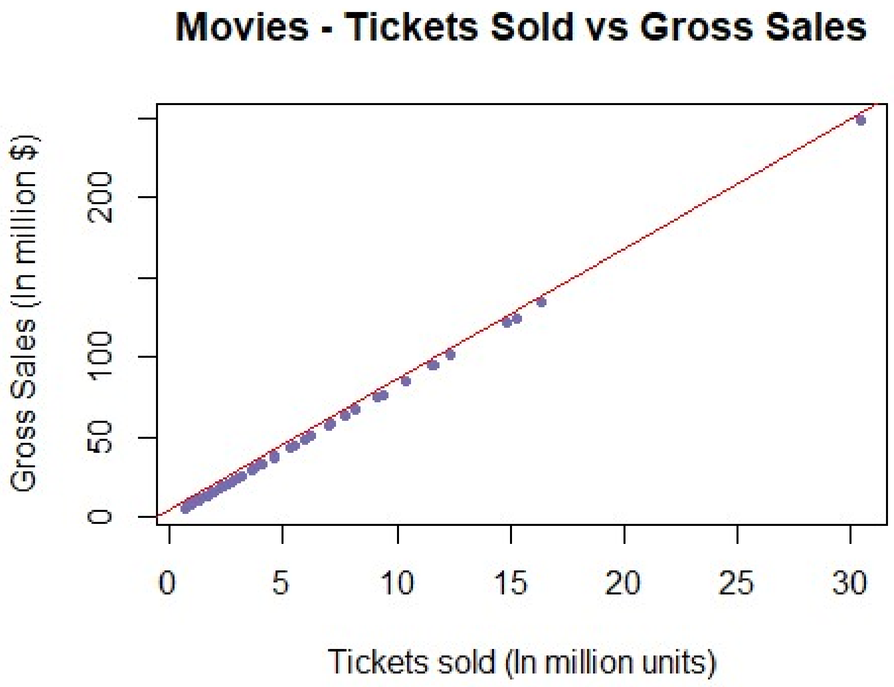
Q2.4 Scatter plots with lines 4a)Do scatter plots with millions scale add a regression line.

```
plot(df2$Tickets_Sold/1000000, df2$Gross_Sales/1000000,
     pch = 20, col = "#756bb1")
abline(lm(df2$Gross_Sales ~ df2$Tickets_Sold), col="red")
```



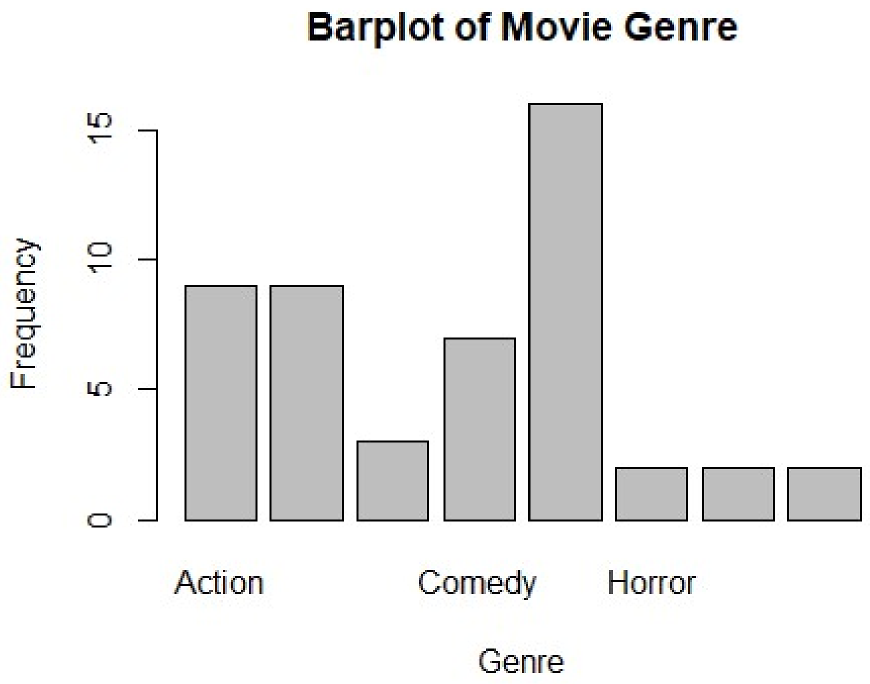
4b. Add xlabel, ylabel and plot title

```
plot(df2$Tickets_Sold/1000000, df2$Gross_Sales/1000000,
     pch = 20, col = "#756bb1",
     xlab = "Tickets sold (In million units)",
     ylab = "Gross Sales (In million $)",
     main = "Movies - Tickets Sold vs Gross Sales")
abline(lm(df2$Gross_Sales ~ df2$Tickets_Sold), col="red")#Q5
```



Q2.5. Obtain a bar plot of genre we get a barplot of various movie genres using the below mentioned function

```
plot(df2$Genre, type = "bar", xlab ='Genre', ylab = 'Frequency', main = "Barp
lot of Movie Genre")

## Warning in plot.window(xlim, ylim, log = log, ...): graphical parameter
## "type" is obsolete

## Warning in axis(if (horiz) 2 else 1, at = at.l, labels = names.arg, lty =
## axis.lty, : graphical parameter "type" is obsolete

## Warning in title(main = main, sub = sub, xlab = xlab, ylab = ylab, ...):
## graphical parameter "type" is obsolete

## Warning in axis(if (horiz) 1 else 2, cex.axis = cex.axis, ...): graphical
## parameter "type" is obsolete
```

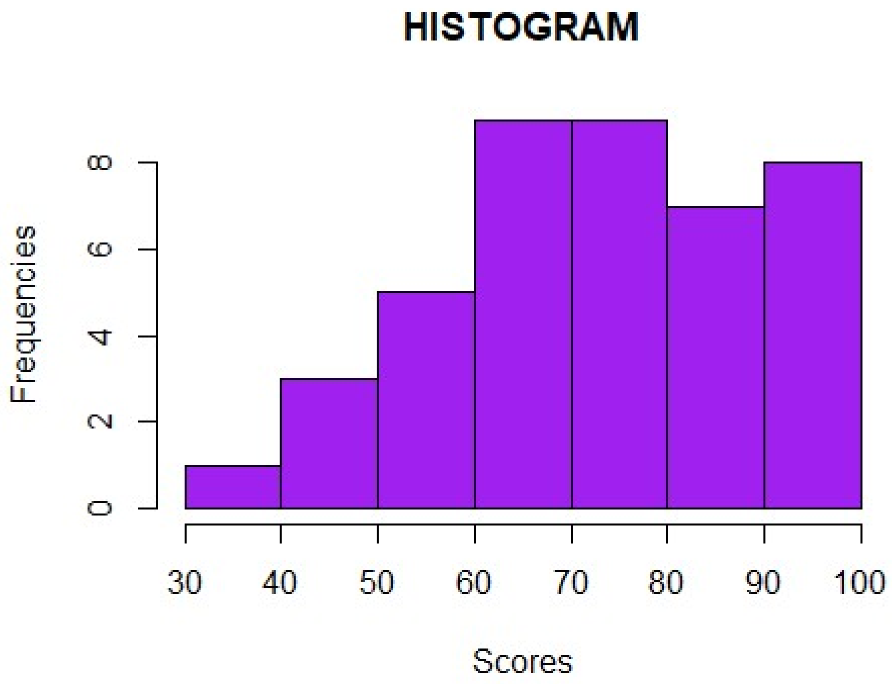


Problem 3 Q3.1. FIND FREQUENCY, RELATIVE, CUMULATIVE frequency

SCORES FREQUENCY	CUMULATIVE FREQUENCY	RELATIVE FREQUENCY
30-39 - 37 = 1	1	0.02
40-49 - 44,49,48 = 3	4	0.07
50-59 - 51,55,54,58,54 = 5	9	0.12
60-69 - 69,64,67,67,67,62,69,64,69 = 9	18	0.21
70-79 - 76,78,78,72,72,76= 6	24	0.14
80-89 - 84,88,80,83,84,83,86,80,82,80 = 10	34	0.23
90-99 - 93,93,92,96,97,97,93,95 = 8	42	0.19

Histogram plot

```
freq <- c(84,88,76,44,80,83,51,93,69,78,49,55,78,93,64,84,54,92,96,72,97,37,9
7,67,83,93,95,67,72,67,86,76,80,58,62,69,64,82,48,54,80,69)
hist(freq,breaks=5, main="HISTOGRAM", xlab="Scores", ylab="Frequencies", col
= "purple")
```


HISTOGRAM
Frequencies
0
2
4
6
8
30
40
50
60
70
80
90
100
Scores