Name: Hima Spandana
Date: 05-17-2020
Honor Statement:
Group Name: Manipogo

Individual Milestone 7

Topic: Video Games

“An electronic game that involves interaction with a user interface to generate visual feedback on a video device such as a TV screen or computer monitor.”

The video games space is a 70-year old industry that has rapidly reached across the entire globe with its influence. Video games have evolved over time to incorporate new and advancing technology. Now, incorporated with the internet, they connect people in their homes from all over the world. The video games industry has swelled to reach over 2.5 billion gamers around the world and took in $131 billion in 2018, with projections of $300 billion by 2025. Video games have engaged a massive, growing audience so I have decided as a team to try and gain insight into the trends in the sales of video game titles over time. Companies spend millions of dollars and thousands of hours of work on some games but it doesn’t always generate enough revenue to even pay off these expenses. If the company is not big enough, they might even go bankrupt after an unsuccessful game release. One of the biggest indications of the revenue generated by the games is the number of sales.

Using the Video Games Sales data I hope to identify trends and predict the sales in different regions as well as overall sales globally. I have decided to split the data into different rows to show individual predictions of the sales in the different regions such as NA, EU, JP, OTHER SALES, GLOBAL SALES.I believe that some games will sell more than others in different regions because some cultures might prefer some genres over others or some platforms are sold more in one region. I believe people also tend to buy more from local publishers/developers. Because of this reason each member of our group will take one region.I have taken EU_sales that is European region sales

I will use the same explanatory variables to analyze our data within my respected region. The variables consist of critic ratings, user ratings, critic score, user score, year of release, and platform. I hope that these variables will provide sufficient data in my study in order to show a relevant analysis on video game sales.

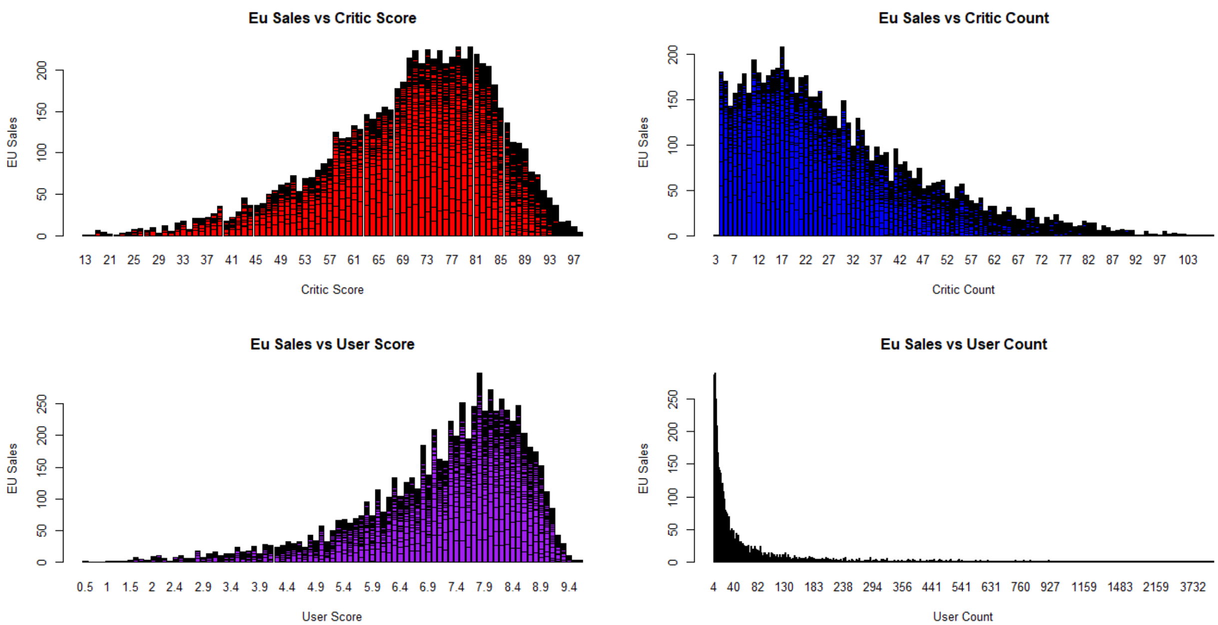
To be able to explore the number of sales a game gets, we will use the data we found on https://www.kaggle.com/kendallgillies/video-game-sales-and-ratings. This dataset contains video games that were released between 1976 and 2017 which sold more than 100,000 copies. The author of this dataset created it mostly using VGChartz and Metacritic. It has 7112 observations and 15 variables. The original data had 17,417 observations,but since not all of the listed video games have information on Metacritic's website we removed the rows with missing values.I also dropped rows when there are missing values in any of the variables. For example, if one game is missing a Critic Score (or Platform information, or genre, etc.), I dropped that row.

The explanatory variables (column section) will consist of the platform, year of release, genre, critic score, critic count, user score, user count, publisher, and the rating. The Eu_sales will be the response variable. The game itself will be rows which are the observations.

Plotted a bar plot for the genre clearly shows that there are a lot more games available in some genres than others, such as there are more than twice the amount of games with Action genre than games with Role-Playing, and the amount of games with Role-Playing tag is more than twice of the ones with Strategy. That might be because some games sell a lot more than the others, thus companies make more of one genre than the other. Also you can see that some games with certain genres get sold less even though there are more games available. E.g. Adventure, Strategy.Similarly, there are a lot more games available to some platform than the others. And this gap might even get bigger if we combine some platforms together such as PS2 and PS3. Also probably the games that are only available to certain platforms affect the sales.

Similarly plotted scatter plot comparing the global sales with the four of the listed regions in the dataset to figure out the linear relationship between them.JP sales is more broadly scattered when compared to others. This tells that in Japan the market share for video games is more when compared to other regions. When compared the year of release with the regional sales the Japan has more number of video games released.In EU region the number of video games usage increased after 2005 and peaked in 2009.

User ratings for Education- with Sports Genre are more widely used, Teenagers prefer Action and Racing genre. Ratings with M and E10+ were also given to few genres, which were almost the same. Plotted individual barplots to compare the EU sales with critic score,critic count, user score and user count. Critic score and user score are negatively skewed, which directly help in spike in EU sales in that region, whereas critic count and user count are positively skewed which did not make any profit in EU sales.



Performed a multiple linear Regression test on the dataset and the response variable wa EU_Sales . Compare with all the explanatory variables do get the desired adjusted R square for the model. For this I have reduced the from 11 variables to 9 variables to get the regression model. The adjusted R square for this model was one. F test looks good. This tells that at least 1 beta is not equal to zero|| 99% of variability global sales is explained by the model. Null Hypothesis is rejected and the alternative is accepted. t-test looks good- The beta associated with global_sales is equal zero, we can reject that and accept the alternative,that it the beta is not equal to zero and use that estimation

Before taking the second order term and the interaction term, we performed the correlation test on the model and found that user score and critic score would be helpful in finding out the second order term and the interaction term in the model. While performing the regression model with the second order term theThe adjusted R square for this model was one. F test looks good. This tells that at least 1 beta is not equal to zero|| 99% of variability global sales is explained by the model. Null Hypothesis is rejected and the alternative is accepted. t-test looks good- The beta associated with global_sales is equal zero, we can reject that and accept the alternative,that the beta is not equal to zero and use that estimation. When using the interaction term for the model, there was no change in the adjusted R square, so removed the interaction term from the model.

Performed Pseudo Random Number Generators for the model, I set the seed and generated a random number, this is to reproduce our experiments.Built a model for test and train dataset, model is built entirely on trained dataset, but is evaluated on the test data. Removed a few variables from the dataset like X,name and platform columns. Approximately 80% of these are 1's and aprroximately 20% of these are 2's.I will use this list to create training and test sets when it is true I will accept it into training data and when it is false I will reject it into my training data. For the test we will get just the opposite of the train set.Training dataset has 5685 observations with 9 variables. the test dataset has 1427 observations with 9 variables.Performed the prediction and actual test on the test and trained the dataset correlation between the actual and predicted value of EU_Sales. Plotted a linear graph between the prediction and actual values after performing the correlation.

Performed K-fold cross validation to do that we have to use the DAAG package, Performed 3 fold cross validations, for which the results are and average mean square error is 1.46

Results for first fold
Sum of squares = 0.09   Mean square = 0   n = 2370

Results for second fold
Sum of squares = 0.08   Mean square = 0   n = 2371

Results for third fold
Sum of squares = 10362   Mean square = 4.37   n = 2371

Lastly performed the backward and forward elimination and compared the initial model with the final model to get the result and can remove the low values if its not giving any change in the values. And created a summary of the empty list which has only y intercepts.

My next goal for the milestone 8 is to perform the multicollinearity for the model and identify the regression pitfalls, variable transformation and do residual analysis.

R Script of the individual milestone

```
> Videogames <- read.csv("C:/Users/spand/Downloads/video_without_na2.csv")
>
```

>

> str(Videogames) # gives the number of observations and the variables along with category type

'data.frame':   7112 obs. of 15 variables:

 $ X   : int 1 3 4 7 8 9 12 14 15 16 ...

 $ Name   : Factor w/ 4524 levels ".hack//Infection Part 1",..: 4345 2125 4347 2607 4343 2610 2124 4337 1885 4338 ...

 $ Platform   : Factor w/ 17 levels "3DS","DC","DS",..: 13 13 13 3 13 13 3 13 16 13 ...

 $ Year_of_Release: int 2006 2008 2009 2006 2006 2009 2005 2007 2010 2009 ...

 $ Genre   : Factor w/ 12 levels "Action","Adventure",..: 11 7 11 5 4 5 7 11 4 11 ...

 $ NA_Sales   : num 41.4 15.7 15.6 11.3 14 ...

 $ EU_Sales   : num 28.96 12.8 10.95 9.15 9.18 ...

 $ JP_Sales   : num 3.77 3.79 3.28 6.5 2.93 4.7 4.13 3.6 0.24 2.53 ...

 $ Other_Sales   : num 8.45 3.29 2.95 2.88 2.84 2.25 1.9 2.15 1.69 1.77 ...

 $ Global_Sales   : num 82.5 35.6 32.8 29.8 28.9 ...

 $ Critic_Score   : int 76 82 80 89 58 87 91 80 61 80 ...

 $ Critic_Count   : int 51 73 73 65 41 80 64 63 45 33 ...

 $ User_Score   : num 8 8.3 8 8.5 6.6 8.4 8.6 7.7 6.3 7.4 ...

 $ User_Count   : int 324 712 193 433 129 595 465 146 106 52 ...

 $ Rating   : Factor w/ 7 levels "AO","E","E10+",..: 2 2 2 2 2 2 2 2 2 2 ...

>

>

> **summary(Videogames)   # gives the min and maximum value of the dataset for each** column

```
       X                                          Name     
 Min.   :    1   Madden NFL 07                          :   9  
 1st Qu.: 2771   LEGO Star Wars II: The Original Trilogy  :   8  
 Median : 6112   Madden NFL 08                          :   8  
 Mean   : 6803   Need for Speed: Most Wanted            :   8  
 3rd Qu.:10308   Harry Potter and the Goblet of Fire      :   7  
 Max.   :17408   Harry Potter and the Order of the Phoenix:   7  
                 (Other)                                :7065  
    Platform    Year_of_Release         Genre     
```

```
 PS2    :1169   Min.   :1985    Action      :1698
 X360   : 888   1st Qu.:2004    Sports      : 981
 PS3    : 790   Median :2007    Shooter     : 900
 PC     : 734   Mean   :2007    Role-Playing: 739
 X      : 586   3rd Qu.:2011    Racing      : 600
 Wii    : 493   Max.   :2016    Platform    : 412
 (Other):2452                   (Other)     :1782
    NA_Sales       EU_Sales        JP_Sales     Other_Sales
 Min.   : 0.0   Min.   : 0.00   Min.   :0.00   Min.   : 0.00
 1st Qu.: 0.1   1st Qu.: 0.02   1st Qu.:0.00   1st Qu.: 0.01
 Median : 0.2   Median : 0.06   Median :0.00   Median : 0.02
 Mean   : 0.4   Mean   : 0.23   Mean   :0.06   Mean   : 0.08
 3rd Qu.: 0.4   3rd Qu.: 0.20   3rd Qu.:0.01   3rd Qu.: 0.07
 Max.   :41.4   Max.   :28.96   Max.   :6.50   Max.   :10.57

  Global_Sales   Critic_Score   Critic_Count     User_Score
 Min.   : 0.0   Min.   :13.0   Min.   :  3.0   Min.   :0.50
 1st Qu.: 0.1   1st Qu.:62.0   1st Qu.: 14.0   1st Qu.:6.50
 Median : 0.3   Median :72.0   Median : 24.0   Median :7.50
 Mean   : 0.8   Mean   :70.2   Mean   : 28.7   Mean   :7.18
 3rd Qu.: 0.7   3rd Qu.:80.0   3rd Qu.: 39.0   3rd Qu.:8.20
 Max.   :82.5   Max.   :98.0   Max.   :113.0   Max.   :9.60

   User_Count     Rating
 Min.   :    4   AO  :   1
 1st Qu.:   11   E   :2162
 Median :   27   E10+: 968
 Mean   :  175   K-A :   1
 3rd Qu.:   88   M   :1489
 Max.   :10766   RP  :   2
                 T   :2489
>
>
```

```
>
> head(Videogames)  # gives the data of the first 6 columns
  X                    Name Platform Year_of_Release    Genre
1 1              Wii Sports      Wii            2006   Sports
2 3           Mario Kart Wii      Wii            2008   Racing
3 4        Wii Sports Resort      Wii            2009   Sports
4 7     New Super Mario Bros.       DS            2006 Platform
5 8                 Wii Play      Wii            2006     Misc
6 9 New Super Mario Bros. Wii      Wii            2009 Platform
  NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales
1     41.4    28.96     3.77        8.45         82.5
2     15.7    12.80     3.79        3.29         35.6
3     15.6    10.95     3.28        2.95         32.8
4     11.3     9.15     6.50        2.88         29.8
5     14.0     9.18     2.93        2.84         28.9
6     14.5     6.95     4.70        2.25         28.4
  Critic_Score Critic_Count User_Score User_Count Rating
1           76           51        8.0        324      E
2           82           73        8.3        712      E
3           80           73        8.0        193      E
4           89           65        8.5        433      E
5           58           41        6.6        129      E
6           87           80        8.4        595      E
> tail(Videogames)   # gives the data of the last 6 columns
         X                          Name Platform
7107 17384 Nancy Drew: The Phantom of Venice       PC
7108 17395         Tom Clancys Splinter Cell       PC
7109 17402               Blacksite: Area 51       PC
7110 17403               Virtua Tennis 2009       PC
7111 17405                    CivCity: Rome       PC
7112 17408                   Super Meat Boy      PS4
     Year_of_Release     Genre NA_Sales EU_Sales JP_Sales
7107            2008 Adventure        0        0        0
```

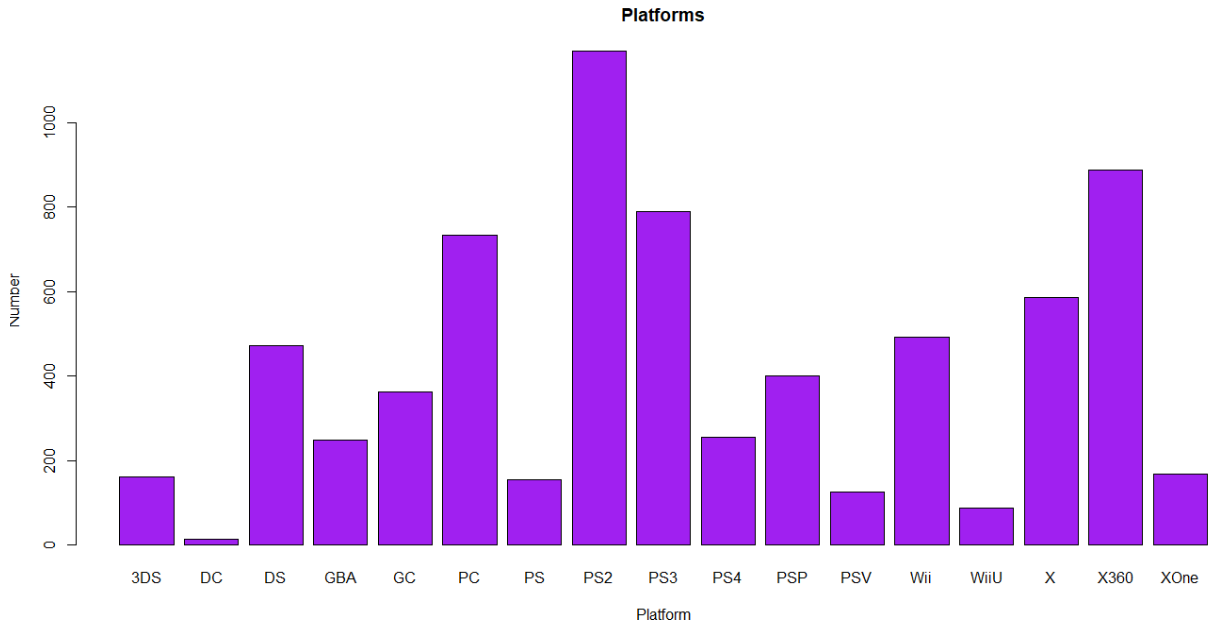
```
7108            2003    Action        0        0        0
7109            2007   Shooter        0        0        0
7110            2009    Sports        0        0        0
7111            2006  Strategy        0        0        0
7112            2016  Platform        0        0        0
    Other_Sales Global_Sales Critic_Score Critic_Count
7107          0         0.01           69            7
7108          0         0.01           91           20
7109          0         0.01           60           20
7110          0         0.01           68            8
7111          0         0.01           67           46
7112          0         0.01           85            7
    User_Score User_Count Rating
7107       6.6          8      E
7108       8.5        291      T
7109       4.9         42      T
7110       6.5         19      E
7111       6.9         32  E10+
7112       7.0        114      T
>
```

> dim(Videogames)   # gives the exact dimensions of the data

```
[1] 7112   15
>
```

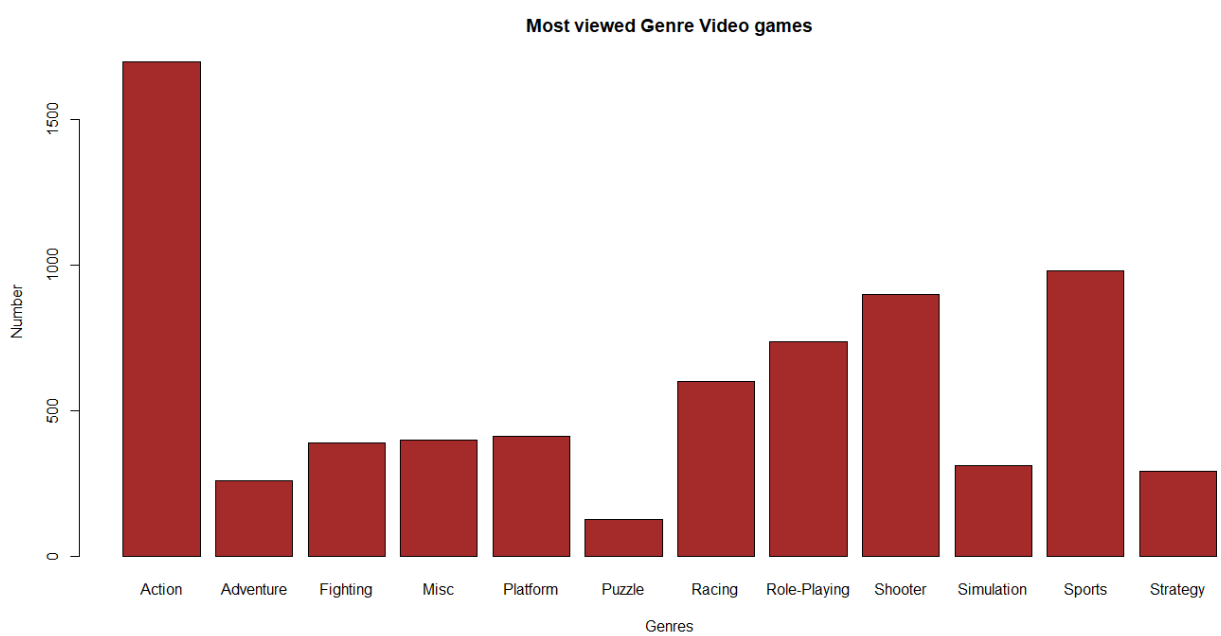
> # Plot a graph between numbers and platform graph

```
> plot(Videogames[, 3], xlab="Platform", ylab= "Number", main = "Platforms", col= "purple")
```



> # Graph of Genres

> plot(Videogames[, 5], xlab="Genres", ylab= "Number", main = "Most viewed Genre Video games", col= "brown")



 # Installing Packages

install.packages('ggplot')

library(ggplot)

install.packages('readr')

library(readr)

install.packages('dplyr')

library(dplyr)

install.packages('ggplot2')

library(ggplot2)

install.packages('scales')

library(scales)

 # A scatter plot to compare the global sales with the remaining sales variables present in the dataset

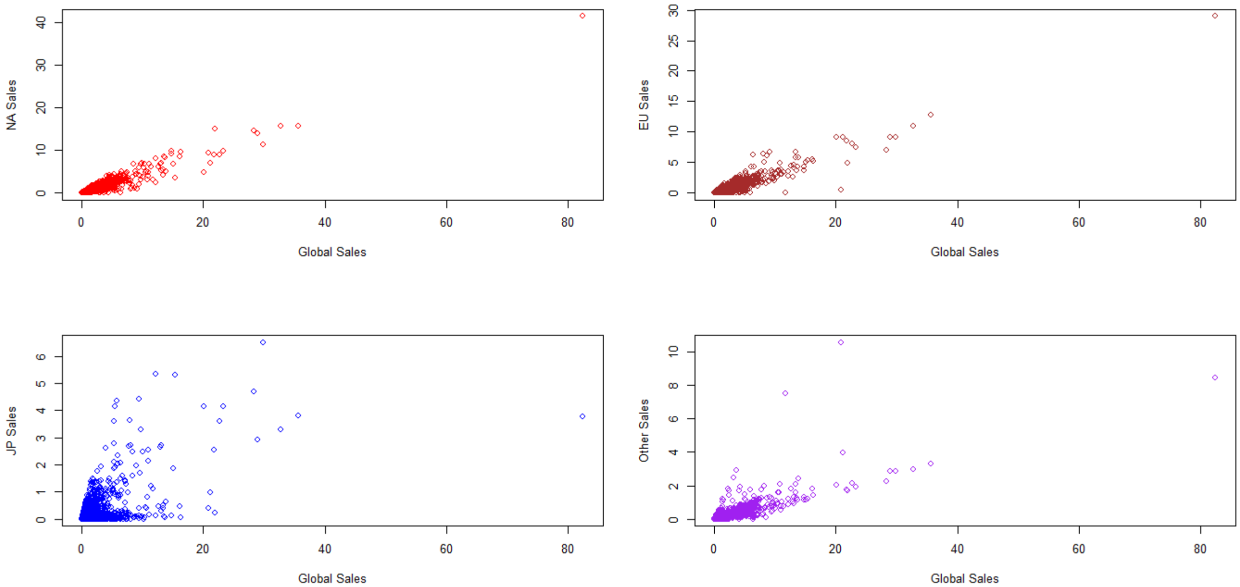
par(mfrow = c(2,2))

plot(Videogames$Global_Sales,Videogames$NA_Sales, xlab ="Global Sales", ylab = "NA Sales", col = 'red')

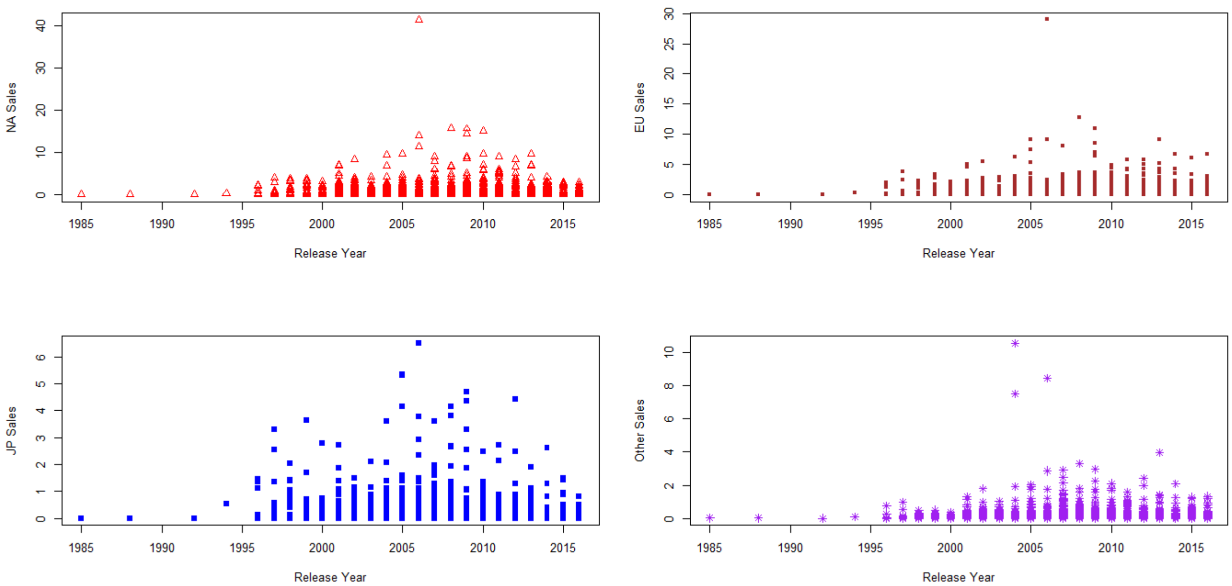
plot(Videogames$Global_Sales,Videogames$EU_Sales, xlab ="Global Sales", ylab = "EU Sales", , col = 'brown')

plot(Videogames$Global_Sales,Videogames$JP_Sales, xlab ="Global Sales", ylab = "JP Sales", col = 'blue')

plot(Videogames$Global_Sales,Videogames$Other_Sales, xlab ="Global Sales", ylab = "Other Sales", col = 'purple')




```
# A scatter plot with year of release with respective sales of the regions
par(mfrow = c(2,2))
plot(Videogames$Year_of_Release,Videogames$NA_Sales, xlab ="Release Year", ylab = "NA Sales", col = 'red', pch= 2)
plot(Videogames$Year_of_Release,Videogames$EU_Sales, xlab ="Release Year", ylab = "EU Sales", , col = 'brown', pch= 20, lwd= 2)
plot(Videogames$Year_of_Release,Videogames$JP_Sales, xlab ="Release Year", ylab = "JP Sales", col = 'blue', pch= 15)
plot(Videogames$Year_of_Release,Videogames$Other_Sales, xlab ="Release Year", ylab = "Other Sales", col = 'purple', pch= 8)
```



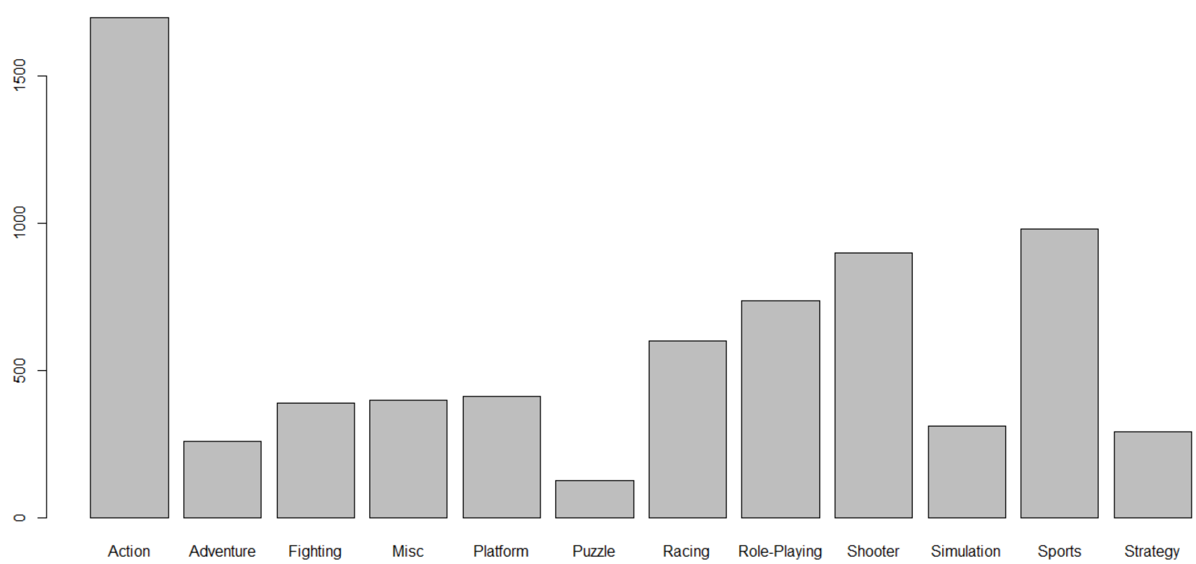
A barplot to show the number of Genres of games played from 1985 to 2015

par(mfrow = c(1,1))

plot(Videogames$NA_Sales~ Videogames$Genre)

table(Videogames$Genre)

barplot(table(Videogames$Genre))



#Videogames$Rating

```
  [1] E    E    E    E    E    E    E    E    E    E    M    M   
 [13] E    M    M    E    E    M    M    M    M    M    M    M   
 [25] M    E    T    M    E    M    E    E    M    E    E    E   
 [37] M    M    M    E10+ E    M    M    T    T    E10+ M    M   
 [49] E    E    E    M    E    M    E    E    E    T    T    T   
 [61] E    M    M    E    M    M    M    T    E    E    E10+ M   
 [73] T    T    E    T    E    T    E10+ M    T    M    T    E   
 [85] M    E    E    M    M    M    E    E    M    M    E    T   
 [97] T    M    E10+ M    E    M    M    M    T    M    E    M   
[109] M    E10+ T    M    T    E10+ E    M    M    E    E    T   
[121] T    E    E    M    E    T    T    M    E    M    T    M   
[133] M    E    T    E    E    M    T    T    M    M    E    E   
[145] M    M    T    T    T    M    T    E    T    M    M    T   
[157] E    E    T    T    T    T    T    E    E    T    E    T   
[169] E    E10+ E    T    E    E    E    M    E10+ T    E    M   
[181] M    T    E10+ M    E10+ M    T    E    M    E    M    E10+
[193] E    E    M    E    T    E    M    M    M    T    M    M   
[205] M    M    E    E    T    M    M    M    M    E10+ E10+ M   
[217] T    E    E10+ T    T    T    E    E10+ M    E    E    E10+
[229] T    E    M    E    E    E    T    E    E10+ M    E    E   
[241] M    E    T    M    T    T    E    T    E    M    M    M   
[253] M    T    E    M    M    M    M    E    M    T    M    M   
[265] E10+ E    M    M    T    M    M    T    T    E    E10+ E   
[277] E    T    T    M    T    T    E10+ M    M    M    T    E   
[289] E    M    E10+ E    E    E    E    T    M    M    T    T   
[301] T    E    E    E    E    E    E    E    E    M    E    E   
[313] M    M    M    T    E    T    E10+ E10+ M    E10+ M    M   
[325] M    T    E    E    M    M    E    T    T    E    M    M   
[337] T    M    E    M    M    M    T    E    T    M    E    M   
[349] E    E    E10+ E    E    E10+ T    T    M    E    T    M   
[361] M    E    M    T    M    M    T    E    T    M    E10+ E10+
[373] M    T    T    M    E    T    E    M    E    M    T    E   
```

[385] E10+ M E T E10+ E T T T M T T

[397] E T T T T E E E M E M E10+

[409] E E E10+ E T M E T T E E M

[421] E10+ E E E10+ E E T E T M E E

[433] E T E M M E T T E10+ T E M

[445] E10+ T E E E10+ M E E T E10+ M M

[457] M T E E T E10+ E T E10+ T T E10+

[469] M E E E T E M T E10+ E E10+ E

[481] M T E E E10+ M E E E M E M

[493] E E E10+ M T E E T E T E M

[505] T T T T T E10+ E E M T E M

[517] E E E T E10+ T E10+ E M M T E

[529] E E E E E M E T M M E E

[541] M M M T E E10+ E T E10+ E E10+ E

[553] E10+ M T T E E10+ M T E10+ E T T

[565] E T T M E10+ E T M E T E E10+

[577] E M E10+ M T T M E10+ T M E E10+

[589] E10+ E M E10+ M T M E AO E T E

[601] E10+ T T T E T E M M M T E

[613] M T E K-A T M E T M E E E

[625] T E M E T E T E10+ E E T T

[637] E T T E M E E E E E10+ T T

[649] E T M M E T E M E E E T

[661] T M E M T E10+ E E10+ T E E M

[673] M M T T T M T E M E10+ T M

[685] M T T M T E10+ E E E M E T

[697] M M T T E E M E10+ M E T M

[709] E E E T M E10+ M T T E E T

[721] E E T E M T E T E E10+ E E

[733] E E10+ E M E M T E10+ E T E10+ T

[745] E M E M E M E E M E M E

[757] E E10+ E E T E10+ E E E M E M

[769] M E E M M T E E E10+ E10+ T E

```
[781] E    E    M    M    M    E    E    E10+ M    T    E    E10+
[793] E10+ T    T    T    E    E    E10+ T    M    M    E    M
[805] T    M    T    T    E    T    M    M    E    T    E10+ M
[817] E    M    E    M    T    T    E    E    E    M    T    T
[829] E    E10+ T    T    E    M    E    T    T    T    T    E
[841] M    M    E    T    M    E    T    E    E10+ T    T    M
[853] T    E10+ T    E10+ E    T    E    T    M    M    E    M
[865] E    M    M    M    E10+ E    E    E    T    E    E    T
[877] E    E    M    E    M    T    E    E    M    M    M    E10+
[889] T    E10+ E    E    M    M    E    E    E    T    M    E10+
[901] M    E    T    E10+ E    M    E10+ E    E    E    E    T
[913] M    T    M    M    E    M    E    T    E10+ E10+ M    M
[925] T    T    M    M    M    E    T    T    E    M    M    M
[937] T    E    T    M    E    E    E10+ T    T    E    E    E
[949] M    T    M    M    E10+ T    M    E    T    E10+ E10+ T
[961] M    T    E    T    T    E10+ T    E    M    M    M    T
[973] M    T    T    E    E10+ E    E    M    E    M    E10+ M
[985] M    E10+ T    T    E    T    M    T    E    E    E10+ E
[997] M    E    E    E
 [ reached getOption("max.print") -- omitted 6112 entries ]
Levels: AO E E10+ K-A M RP T
>
> Genre_Rating_Edu <- table(Videogames$Genre[Videogames$Rating == 'E'])
> Genre_Rating_Edu


      Action    Adventure     Fighting         Misc     Platform
         199           51            6          167          246
      Puzzle       Racing Role-Playing      Shooter   Simulation
          95          359           74           25          109
      Sports     Strategy
         785           46
> #-----
> Genre_Rating_Teen <- table(Videogames$Genre[Videogames$Rating == 'T'])
```

```
> Genre_Rating_Teen

      Action    Adventure     Fighting         Misc     Platform 
         604           85          326          138           58 
      Puzzle       Racing Role-Playing      Shooter   Simulation 
           7          141          403          298          168 
      Sports     Strategy 
         110          151 
> #----
> Genre_Rating_Middle <- table(Videogames$Genre[Videogames$Rating == 'M'])
> Genre_Rating_Middle

      Action    Adventure     Fighting         Misc     Platform 
         579           90           46           11            3 
      Puzzle       Racing Role-Playing      Shooter   Simulation 
           0           18          159          543            5 
      Sports     Strategy 
          11           24 
> #---
> Genre_Rating_E10 <- table(Videogames$Genre[Videogames$Rating == 'E10+'])
> Genre_Rating_E10

      Action    Adventure     Fighting         Misc     Platform 
         315           34           14           82          105 
      Puzzle       Racing Role-Playing      Shooter   Simulation 
          25           82          103           34           30 
      Sports     Strategy 
          75           69 
```

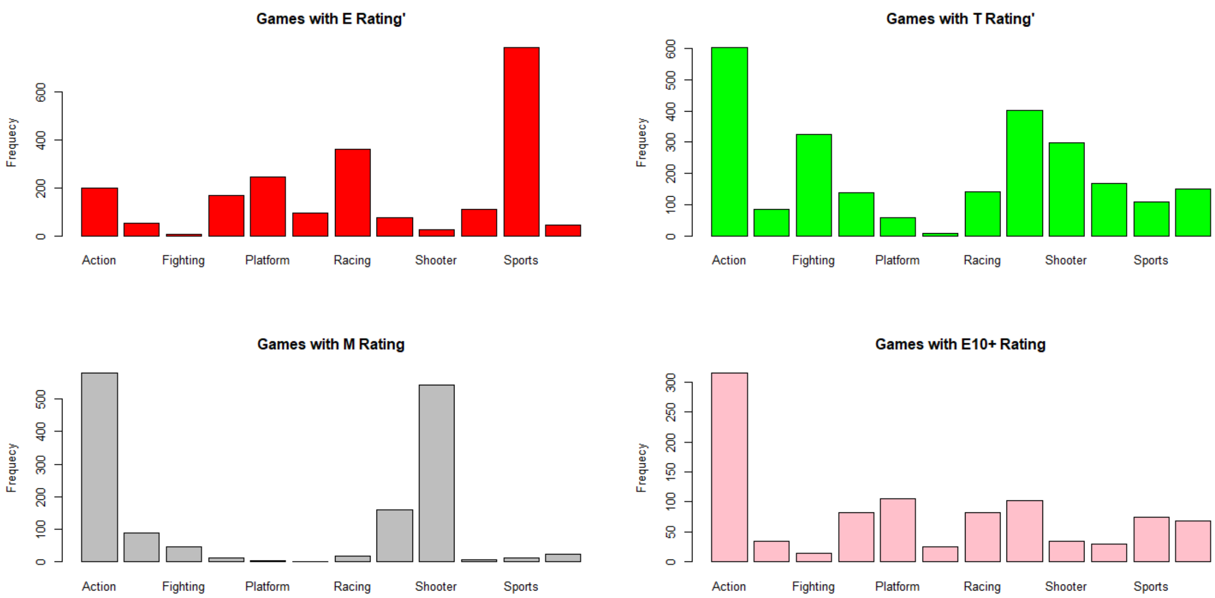
#Individual games with specific rating graphs

```
par(mfrow = c(2,2))
barplot(Genre_Rating_Edu, ylab= "Frequecy", main = "Games with E Rating"", col= 'red')
```

barplot(Genre_Rating_Teen, ylab= "Frequecy", main = "Games with T Rating'", col= 'Green')

barplot(Genre_Rating_Middle, ylab= "Frequecy", main = "Games with M Rating", col= 'Grey')

barplot(Genre_Rating_E10, ylab= "Frequecy", main = "Games with E10+ Rating", col= 'Pink')



> table(Videogames$EU_Sales, Videogames$Critic_Score)

	13	17	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35
0	0	0	3	1	1	0	2	1	1	3	2	5	1	4	3	5	7	5	6
0.01	1	0	1	1	0	1	0	0	2	2	2	1	0	2	0	2	1	0	4
0.02	0	0	0	0	0	0	0	1	1	1	0	1	1	0	1	0	0	0	2
0.03	0	1	0	0	1	0	0	1	0	0	1	1	0	1	0	0	3	1	3
0.04	0	0	0	0	0	0	0	1	0	1	0	0	0	1	0	1	2	0	1
0.05	0	0	1	1	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0
0.06	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	2
0.07	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	1	0	0	0
0.08	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	1	1	0	0

0.09	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	1
0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
0.11	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0	0	0

	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54
0	5	11	7	12	3	7	8	11	7	13	13	10	13	16	14	15	8	9	9
0.01	4	2	5	5	4	3	5	1	6	5	8	6	11	8	6	10	8	12	9
0.02	4	5	3	3	1	0	2	4	4	6	4	7	2	6	5	7	7	11	10
0.03	1	2	0	1	1	1	3	3	4	1	1	4	5	5	11	7	3	8	4
0.04	0	0	1	2	1	1	4	4	2	0	0	3	2	5	5	5	2	2	5
0.05	1	1	0	5	1	0	2	4	1	0	0	1	2	2	2	2	3	6	5
0.06	1	0	1	0	0	0	1	1	1	2	1	0	2	3	1	1	2	4	4
0.07	1	0	1	1	1	0	0	1	2	0	0	1	2	1	2	3	0	3	3
0.08	1	0	1	1	0	0	0	0	0	2	0	2	1	1	0	1	2	1	2
0.09	1	0	0	3	1	1	0	4	1	1	2	3	1	1	0	3	1	2	0
0.1	0	0	1	0	2	2	0	0	1	0	2	2	0	1	2	2	1	3	1
0.11	0	0	0	0	1	0	0	0	2	1	2	0	1	0	0	0	0	0	0

	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73
0	14	21	16	22	20	18	26	23	37	27	22	23	32	27	28	34	38	22	29
0.01	9	18	14	14	11	18	17	18	16	17	18	29	17	23	22	21	29	24	17
0.02	15	8	9	18	13	10	8	18	16	18	14	17	17	15	22	22	17	24	25
0.03	5	5	7	9	10	13	8	11	13	11	14	11	14	12	14	15	14	16	15
0.04	1	3	6	9	8	9	11	7	6	11	7	13	13	8	11	13	14	8	10
0.05	3	1	5	9	4	2	11	3	6	3	10	6	6	7	6	7	9	9	7
0.06	1	3	0	4	2	3	4	3	0	6	6	5	3	8	4	8	7	10	5
0.07	0	2	5	4	4	4	5	3	4	3	5	5	2	6	6	1	5	8	9
0.08	1	3	5	3	4	3	4	1	3	1	4	2	3	4	9	6	4	6	3
0.09	3	2	1	6	2	3	6	4	4	2	2	3	0	6	7	3	2	3	7
0.1	1	4	1	5	3	1	3	1	0	2	3	3	6	3	3	5	6	2	5
0.11	1	1	2	3	1	3	2	1	3	3	1	6	4	3	4	6	6	4	3

	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92

```
  0      18 30 20 28 30 26 24 22 28 20  9 10  5  4  2  5  4  4  2
  0.01   16 27 11 16 13 14 19 15 19 14 11  8  7  3  4  2  1  3  2
  0.02   25 25 17 14 26 16 17 14 14 13 12 11  6  6  3  7  9  1  5
  0.03   18 16 18 13 15 16 12 10 12 14  9  5  8  5  3  7  4  3  1
  0.04    6 13 14 14 16 16  9  9 11 10  5 10  5  5  6  5  3  1  1
  0.05    6 12  8 10 10 10  6 14  5  9  4  4  4  3  2  3  3  2  0
  0.06   12  5  7  9  5  6  6  4  8  9  3  4  6  3  3  3  2  1  1
  0.07    7  7 11  4  8  3  6  8  5  5  5  2  3  5  3  0  1  3  0
  0.08    4  2  6  4  3  4  6  8  4  4  6  5  4  2  1  3  0  4  3
  0.09    6  5  5  7  2  4  4  3  1  3  1  3  4  0  8  1  2  1  4
  0.1     6  5  2  7  4  2  4  5  4  4  4  3  1  5  2  0  0  1  0
  0.11    6  3  8  8  6  1  4  4  7  2  2  1  3  1  1  2  3  0  0

        93 94 95 96 97 98
  0      1  1  0  1  0  1
  0.01    3  1  0  0  0  0
  0.02    2  1  0  1  0  0
  0.03    1  1  0  0  0  0
  0.04    2  0  0  0  0  0
  0.05    0  0  0  0  1  0
  0.06    0  0  0  0  0  0
  0.07    0  0  0  0  0  0
  0.08    1  0  0  0  0  0
  0.09    1  1  0  0  0  0
  0.1     2  1  0  0  0  0
  0.11    0  0  0  1  0  0
 [ reached getOption("max.print") -- omitted 260 rows ]
> EUSales_Criticscore <-  table(Videogames$EU_Sales, Videogames$Critic_Score)
>
> par(mfrow = c(1,1))
> barplot(EUSales_Criticscore, xlab = 'Critic Score', ylab= 'EU Sales')
>
>
```

```
>
> table(Videogames$EU_Sales, Videogames$Critic_Count)

        3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21
  0     0 55 43 36 45 39 51 25 48 36 35 28 37 29 39 26 35 28 18
  0.01  0 26 19 24 20 26 21 19 19 23 25 16 24 26 20 21 20 17 22
  0.02  0 21 21 11 11 13 18 14 27 19 17 21 19 22 26 19 18 16 18
  0.03  0  9 13  9 15 13 14 16 15 15  8 17 10 15 20  9 13 13 10
  0.04  0 11 12  6  6  8 12 11 17 16 10 10 11 10 19 10  7 11 12
  0.05  0 12 10  7  3  9  8  4  5  7  5  9 11  8  5 10  8  8  7
  0.06  0  4  6  4  4  5  1  8  5  6  4  5  6  2  8  9  2  7  4
  0.07  0  1  5  3  3  4  2  4  9  3  4  4  6  6  7  7  7  2  6
  0.08  0  2  4  1  4  3  3  5  2  4  6  6  4  5  6  5  4  7  5

       22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40
  0    24 24 27 31 22 19 13 24 24 10  9 18 12 11 10 10  8  1  2
  0.01 24 22 21 17 14 15  9 10 15 14 10 12 16 11  9 10  6  7  5
  0.02 16 14 15 15 15 11 18 13 14 10 11  8  9 10  7  7 13  9  8
  0.03 13 10  9 19 12 13  7  9  8 16  5 10  8  7  7  8  5  8  4
  0.04  9  7  8  9  6 12  9  1  5 10  6  3  6  2  6  5  3  6  7
  0.05 11  6  2  4 11  5  4  1  6  6  3  3  5  3  0  4  1  4  4
  0.06 10  5  3  5  1  3  3  6  4  5  4  9  2  5  1  3  3  3  4
  0.07  5  3  5  4  3  6  5  5  4  6  4  7  3  4  3  4  0  3  1
  0.08  3  2  4  4  4  5  3  3  2  4  3  3  3  2  2  4  2  1  2

       41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59
  0    11  2  4  6  6  3  0  5  4  0  3  1  2  1  4  0  3  0  0
  0.01  7  7  7  5  3  5  1  5  3  2  4  2  3  1  0  3  0  1  2
  0.02  6  8  3  7  1 11  4  2  7  2  4  8  1  5  4  2  3  5  2
  0.03  7  3  2  6  2  7  3  2  1  3  6  4  0  3  3  2  4  1  2
  0.04  2  0  6  6  3  4  2  3  2  2  5  1  1  3  3  2  1  0  2
  0.05  5  2  4  4  2  1  2  5  1  4  3  1  2  2  3  1  1  1  2
  0.06  2  5  6  3  3  4  3  1  1  0  0  1  1  2  2  0  1  0  1
```

0.07	2	3	2	1	0	0	1	1	4	5	0	2	2	3	2	2	0	0	0
0.08	2	2	3	3	1	2	0	2	1	0	2	1	0	2	1	1	0	1	0

	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78
0	3	0	1	0	2	0	0	0	1	0	0	0	0	1	0	0	0	0	0
0.01	0	0	0	2	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
0.02	0	0	2	0	0	2	0	0	1	0	0	1	1	0	0	2	0	0	1
0.03	0	1	2	0	0	0	1	2	0	2	1	0	0	0	0	0	2	1	0
0.04	0	3	1	1	0	1	0	1	0	1	1	0	1	0	0	1	1	0	0
0.05	0	0	1	0	1	0	0	0	0	1	2	1	0	0	0	1	0	0	0
0.06	0	0	0	0	0	3	0	0	0	0	0	0	0	0	0	0	1	0	0
0.07	2	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.08	0	0	0	0	1	0	1	0	1	1	1	1	0	0	0	1	0	0	0

	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.01	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.02	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.03	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.04	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.05	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.06	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.07	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0.08	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	98	99	100	101	102	103	104	105	106	107	113
0	0	0	0	0	0	0	0	0	0	0	0
0.01	0	0	0	0	0	0	0	0	0	0	0
0.02	0	0	0	0	0	0	0	0	0	0	0
0.03	0	0	0	0	0	0	0	0	0	0	0
0.04	0	0	0	0	0	0	0	0	0	0	0
0.05	0	0	0	0	0	0	0	0	0	0	0
0.06	0	0	0	0	0	0	0	0	0	0	0

```
  0.07   0  0   0   0   0   0   0   0   0   0   0
  0.08   0  0   0   0   0   0   0   0   0   0   0
 [ reached getOption("max.print") -- omitted 263 rows ]
> EUSales_Criticcount <- table(Videogames$EU_Sales, Videogames$Critic_Count)
> barplot(EUSales_Criticcount, xlab = 'Critic Count', ylab= 'EU Sales')
>
>
>
> table(Videogames$EU_Sales, Videogames$User_Score)

       0.5 0.6 0.7 0.9  1 1.2 1.3 1.4 1.5 1.7 1.8 1.9  2 2.1 2.2
  0      1   1   1   1  0   0   1   1   0   3   2   1  3   2   3
  0.01   0   0   0   0  2   0   1   0   2   0   2   0  1   1   0
  0.02   1   0   0   0  0   1   0   1   0   1   0   0  0   1   2
  0.03   0   0   0   0  0   0   0   0   0   1   0   0  1   0   0
  0.04   0   0   0   0  0   0   0   0   0   0   0   0  1   1   0
  0.05   0   0   0   0  0   0   0   0   0   0   0   1  1   0   0
  0.06   0   0   0   0  0   1   0   0   0   0   0   0  0   0   0
  0.07   0   0   0   0  0   0   0   0   0   1   0   0  0   1   0
  0.08   0   0   0   0  0   0   0   0   1   0   0   0  1   0   0
  0.09   0   0   0   0  0   0   0   0   0   0   0   0  0   0   0
  0.1    0   0   0   0  0   0   0   0   0   0   0   0  0   0   0

       2.3 2.4 2.5 2.6 2.7 2.8 2.9  3 3.1 3.2 3.3 3.4 3.5 3.6
  0      2   2   3   2   2   6   2  3   7   2   3   1   7   5
  0.01   0   1   3   0   1   5   0  3   1   0   0   2   3   1
  0.02   0   1   1   1   1   3   2  0   1   1   0   1   1   1
  0.03   0   0   2   0   0   2   0  1   2   1   1   0   3   0
  0.04   0   0   0   0   0   1   0  1   0   1   0   1   0   0
  0.05   0   0   0   0   1   0   0  0   1   0   1   0   4   1
  0.06   0   0   0   0   0   0   1  0   0   1   0   2   2   0
  0.07   0   1   0   0   0   0   1  1   0   0   0   0   0   0
  0.08   0   2   0   0   0   0   0  1   0   0   2   0   0   0
```

0.09	0	0	0	0	0	0	1	0	0	0	0	2	0	1
0.1	0	0	0	0	0	0	0	2	1	0	0	0	0	1

	3.7	3.8	3.9	4	4.1	4.2	4.3	4.4	4.5	4.6	4.7	4.8	4.9	5	5.1
0	3	4	2	7	3	7	2	6	11	6	4	11	7	4	7
0.01	2	1	0	5	2	3	3	2	2	6	3	6	3	7	2
0.02	2	3	1	0	5	1	2	2	4	4	3	2	2	8	2
0.03	1	1	3	2	3	1	0	1	2	0	1	2	2	9	2
0.04	1	4	0	3	1	2	3	1	0	1	1	2	2	5	1
0.05	1	5	0	2	1	1	0	3	1	0	1	0	1	2	3
0.06	0	0	1	1	1	0	0	3	1	0	0	2	1	3	1
0.07	1	0	0	1	0	1	1	1	1	1	1	2	0	1	0
0.08	0	1	0	1	2	0	0	2	0	0	2	2	0	2	0
0.09	1	0	1	0	0	0	2	1	0	0	1	2	1	2	0
0.1	0	0	0	0	0	0	1	2	1	1	0	2	1	0	0

	5.2	5.3	5.4	5.5	5.6	5.7	5.8	5.9	6	6.1	6.2	6.3	6.4	6.5
0	6	17	14	10	15	7	20	2	21	16	15	26	14	25
0.01	7	10	6	7	12	12	8	10	10	9	16	10	9	11
0.02	5	5	11	2	3	6	10	9	10	4	10	14	10	10
0.03	3	2	3	5	6	1	9	6	8	2	8	9	6	10
0.04	1	4	4	2	4	2	2	3	7	4	3	9	9	8
0.05	2	3	1	4	2	2	6	1	4	4	8	2	2	4
0.06	1	2	1	1	3	3	1	1	5	4	2	3	3	4
0.07	1	2	0	1	3	1	1	2	5	2	1	3	3	4
0.08	4	1	0	0	0	3	0	4	2	1	2	1	2	0
0.09	0	1	0	1	0	2	2	5	5	1	5	2	6	2
0.1	2	1	2	2	1	1	3	2	3	1	0	2	2	3

	6.6	6.7	6.8	6.9	7	7.1	7.2	7.3	7.4	7.5	7.6	7.7	7.8	7.9	8
0	18	18	27	16	43	24	19	48	34	34	24	42	35	36	41
0.01	10	11	21	15	22	19	16	21	18	29	18	21	41	15	29
0.02	10	17	19	9	21	11	12	21	18	24	24	19	26	27	26

0.03	10	6	16	2	14	8	21	17	9	22	13	14	21	11	12
0.04	7	3	10	2	8	11	7	13	9	11	7	12	18	10	15
0.05	4	4	6	5	10	7	4	8	7	9	4	8	13	7	12
0.06	7	2	4	8	4	5	2	4	7	6	6	5	9	8	8
0.07	5	2	7	5	3	4	5	9	6	4	7	7	4	4	10
0.08	2	1	5	0	7	5	4	5	3	7	3	4	7	5	4
0.09	6	1	4	3	9	3	3	2	8	6	4	5	6	1	9
0.1	3	2	4	6	5	3	1	4	6	2	2	2	3	1	3

	8.1	8.2	8.3	8.4	8.5	8.6	8.7	8.8	8.9	9	9.1	9.2	9.3	9.4
0	26	27	28	22	22	16	12	15	7	9	3	2	3	3
0.01	22	22	24	18	18	16	15	18	7	8	3	3	6	0
0.02	19	18	24	23	25	18	20	9	7	11	9	7	4	0
0.03	20	21	17	18	14	15	10	19	12	9	8	2	2	1
0.04	14	22	15	7	14	13	12	11	12	9	6	5	1	0
0.05	9	11	4	11	10	8	11	10	2	3	4	4	1	0
0.06	8	11	6	8	8	6	5	1	5	7	2	2	0	0
0.07	4	4	10	4	8	7	5	7	7	3	3	0	2	0
0.08	5	9	5	4	8	5	4	3	12	2	3	0	2	0
0.09	3	4	4	5	5	3	1	4	4	1	7	1	0	0
0.1	5	5	10	7	8	5	3	4	4	3	0	1	1	0

	9.5	9.6
0	1	1
0.01	0	0
0.02	1	0
0.03	0	0
0.04	0	1
0.05	0	0
0.06	0	0
0.07	0	0
0.08	0	0
0.09	0	0

```
  0.1     0   0
 [ reached getOption("max.print") -- omitted 261 rows ]
> EUSales_Userscore <- table(Videogames$EU_Sales, Videogames$User_Score)
> barplot(EUSales_Userscore, xlab = 'User Score', ylab= 'EU Sales')
>
>
>
> table(Videogames$EU_Sales, Videogames$User_Count)

       4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22
  0   89 76 70 50 55 41 30 32 25 24 26 33 29 25 26 18 15 23 11

      23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41
  0   13 10 17 12 10  7  7  8 11 13  4 10  7  4 10  7  5  2  3

      42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
  0    4  5  9  6  1  5  4  3  2  2  4  1  0  2  0  3  0  1  2

      61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79
  0    5  1  4  1  2  1  2  1  0  2  1  5  2  2  2  3  1  0  2

      80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98
  0    0  1  1  0  0  1  0  4  0  1  1  0  0  0  1  1  2  0  0

      99 100 101 102 103 104 105 106 107 108 109 110 111 112
  0    1   1   0   1   0   0   0   0   1   2   2   0   0   0

      113 114 115 116 117 118 119 120 121 122 123 124 125 126
  0     0   1   0   1   0   1   1   0   1   0   0   1   0   1

      127 128 129 130 131 132 133 134 135 136 137 138 139 140
  0     0   0   1   0   0   1   1   0   0   0   0   1   0   0
```

	141	142	143	144	145	146	147	148	149	150	151	152	153	154
0	0	1	0	0	0	0	0	1	0	0	0	0	0	0

	155	156	157	158	159	160	161	162	163	164	165	166	167	168
0	0	0	0	1	0	0	1	0	0	1	0	2	1	0

	169	170	171	172	173	174	175	176	177	178	179	180	181	182
0	0	0	0	0	0	0	0	1	0	0	0	0	0	1

	183	184	185	186	187	188	189	190	191	192	193	195	196	198
0	0	0	0	0	0	1	0	0	0	0	1	0	0	0

	199	200	201	202	203	204	205	206	207	208	209	210	211	212
0	0	0	0	1	1	0	1	0	0	0	0	0	0	0

	213	214	215	216	217	218	219	220	221	222	223	224	225	226
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

	227	228	229	230	231	232	233	234	235	236	237	238	239	240
0	0	0	0	0	1	0	0	0	0	0	0	2	0	0

	241	243	244	245	246	247	248	249	250	251	253	255	256	257
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

	258	259	260	261	262	263	264	265	266	267	268	269	270	271
0	0	0	0	0	0	0	0	1	0	0	0	0	0	0

	272	273	274	275	276	277	278	279	280	281	282	283	284	285
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	286	287	288	289	290	291	292	293	294	295	296	297	298	299
0	0	1	0	0	0	1	0	0	0	0	0	0	0	0

	300	301	303	304	305	307	308	309	310	311	312	313	314	315
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	316	317	318	319	321	324	325	326	327	328	329	330	331	332
0	0	0	0	0	1	0	0	0	0	0	0	0	0	0

	333	335	336	337	339	340	341	342	343	345	347	348	349	350
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	351	352	353	354	355	356	357	358	359	361	362	365	366	367
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	368	369	371	374	375	376	377	378	381	384	385	386	387	388
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	390	391	393	395	397	399	400	403	404	405	407	409	410	412
0	0	0	0	0	0	0	1	0	0	0	0	0	0	0

	413	415	418	421	423	424	425	429	432	433	434	435	436	438
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	439	440	441	442	443	444	445	447	450	452	453	455	456	457
0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

	458	461	462	463	465	467	468	469	470	472	473	479	482	483
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	487	488	490	491	492	493	494	496	498	503	507	511	512	516
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	517	518	519	521	524	525	526	530	531	533	534	536	539	541
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	542	543	544	545	546	547	548	549	552	553	555	556	557	558
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	559	561	562	565	566	567	569	570	571	572	573	575	576	580
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	583	586	587	589	590	595	598	600	603	605	606	607	608	609
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	610	611	614	619	620	621	623	624	628	629	631	632	633	635
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	642	645	646	648	654	657	658	660	661	664	666	667	669	670
0	0	0	0	0	0	0	0	0	1	0	0	0	0	0

	671	675	681	682	684	685	686	688	695	696	697	698	700	702
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	706	708	710	712	713	715	716	717	718	719	723	726	730	734
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	735	736	738	747	752	754	755	760	762	763	769	771	774	775
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	776	777	778	782	784	787	789	792	798	800	803	808	813	815
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	818	819	820	823	825	835	842	845	846	848	850	859	865	870
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	874	877	888	894	895	896	897	900	902	905	907	910	912	914
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

916 918 921 925 927 934 935 937 938 943 945 948 956 957

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

965 968 972 977 980 981 982 986 993 994 1000 1007 1008

0 0 1 0 0 0 0 1 0 0 0 0 0 0

1010 1017 1025 1031 1034 1035 1036 1046 1049 1059 1062

0 0 0 0 0 0 0 0 0 0 0 1

1067 1070 1075 1077 1079 1083 1084 1087 1095 1096 1100

0 0 0 0 0 0 0 0 0 0 0 0

1101 1104 1111 1112 1113 1115 1123 1124 1130 1136 1137

0 0 0 0 0 0 0 0 0 0 0 0

1139 1142 1158 1159 1162 1166 1173 1176 1177 1185 1187

0 0 0 0 0 0 0 0 0 0 0 0

1189 1194 1199 1200 1204 1208 1212 1219 1224 1226 1228

0 0 0 0 0 0 0 0 0 0 0 0

1232 1233 1240 1242 1243 1253 1258 1259 1263 1270 1272

0 0 0 0 0 0 0 0 0 0 0 0

1289 1292 1297 1300 1302 1304 1309 1314 1318 1320 1322

0 0 0 0 0 0 0 0 0 0 0 0

1327 1333 1341 1350 1353 1369 1386 1408 1409 1410 1412

0 0 0 0 0 0 0 0 0 0 0 0

1417 1434 1440 1446 1451 1456 1472 1474 1475 1476 1477

0 0 0 0 0 0 0 0 0 0 0 0

	1478	1482	1483	1500	1502	1505	1509	1510	1512	1525	1538
0	0	0	0	0	0	0	0	0	0	0	0

	1552	1553	1562	1570	1579	1586	1603	1604	1606	1607	1615
0	0	0	0	0	0	0	0	0	0	0	0

	1621	1625	1639	1652	1657	1658	1659	1662	1663	1664	1667
0	0	0	0	0	0	0	0	0	0	0	0

	1675	1679	1682	1702	1715	1733	1736	1747	1749	1759	1762
0	0	0	0	0	0	0	0	0	0	0	0

	1855	1859	1861	1862	1864	1868	1874	1890	1907	1972	1979
0	0	0	0	1	0	0	0	0	0	0	0

	1982	1983	1988	2005	2053	2069	2075	2077	2094	2113	2118
0	0	0	0	0	0	0	0	0	0	0	0

	2144	2159	2174	2197	2199	2200	2210	2234	2241	2256	2261
0	0	0	0	0	0	0	0	0	0	0	0

	2298	2301	2312	2327	2328	2332	2376	2382	2411	2427	2429
0	0	0	0	0	0	0	0	0	0	0	0

	2455	2468	2475	2496	2506	2530	2546	2647	2659	2661	2664
0	0	0	0	0	0	0	0	0	0	0	0

	2702	2715	2746	2755	2775	2779	2849	2954	2955	2958	2989
0	0	0	0	0	0	0	0	0	0	0	0

	3018	3057	3083	3153	3179	3191	3212	3273	3301	3416	3441
0	0	0	0	0	1	0	0	0	0	0	0

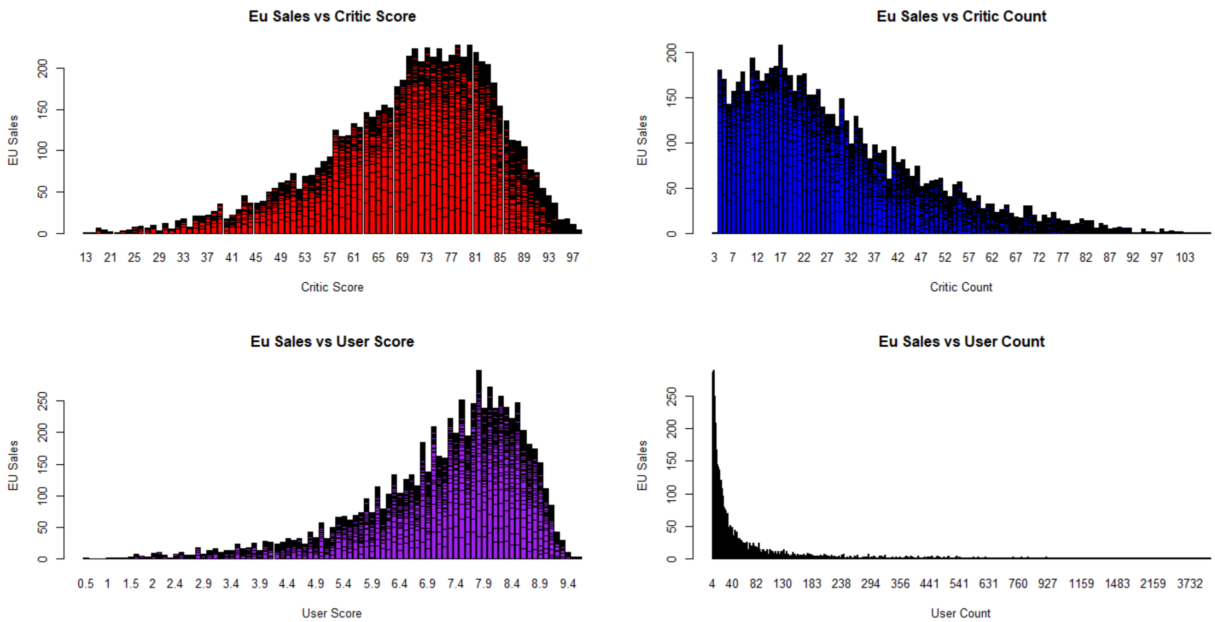
```
      3479 3535 3564 3571 3572 3575 3585 3602 3629 3722 3725
  0      0    0    0    1    0    0    0    0    0    0    0

      3732 3733 3750 3826 3951 3974 3987 4009 4109 4131 4160
  0      0    0    0    0    0    0    0    0    0    1    0

      4258 4408 4546 4578 4942 5146 5226 5237 5319 5389 5670
  0      0    0    0    0    0    0    0    0    0    0    0

      5946 6035 6165 6432 6441 7238 7350 7565 8039 8702 8715
  0      0    0    0    0    0    0    0    0    0    0    0

      9142 9643 9857 10270 10766
  0      0    0    0     0     0
 [ reached getOption("max.print") -- omitted 271 rows ]
> EUSales_Usercount <- table(Videogames$EU_Sales, Videogames$User_Count)
> barplot(EUSales_Usercount, xlab = 'User Count', ylab= 'EU Sales')
> par(mfrow = c(2,2))
> barplot(EUSales_Criticscore, xlab = 'Critic Score', ylab= 'EU Sales', main = "Eu Sales vs Critic Score", col= 'red')
> barplot(EUSales_Criticcount, xlab = 'Critic Count', ylab= 'EU Sales',main = "Eu Sales vs Critic Count", col= 'blue')
> barplot(EUSales_Userscore, xlab = 'User Score', ylab= 'EU Sales', main = "Eu Sales vs User Score", col= 'purple')
> barplot(EUSales_Usercount, xlab = 'User Count', ylab= 'EU Sales', main = "Eu Sales vs User Count", col= 'pink')
```




```
> # Building the multiple regression model
>
> #Model1
> lm(EU_Sales ~ NA_Sales+ Global_Sales+JP_Sales+
Other_Sales+Critic_Score+Critic_Count+ User_Score + User_Count, data= Videogames)

Call:
lm(formula = EU_Sales ~ NA_Sales + Global_Sales + JP_Sales +
    Other_Sales + Critic_Score + Critic_Count + User_Score +
    User_Count, data = Videogames)

Coefficients:
 (Intercept)      NA_Sales  Global_Sales      JP_Sales
   -3.75e-04     -1.00e+00      1.00e+00     -1.00e+00
 Other_Sales  Critic_Score  Critic_Count    User_Score
   -1.00e+00     -7.55e-07      2.56e-06      1.41e-05
  User_Count
    1.31e-07
```

```
> model1<- lm(EU_Sales ~ NA_Sales+ Global_Sales+JP_Sales+
Other_Sales+Critic_Score+Critic_Count+ User_Score + User_Count, data= Videogames)
> summary(model1)

Call:
lm(formula = EU_Sales ~ NA_Sales + Global_Sales + JP_Sales +
    Other_Sales + Critic_Score + Critic_Count + User_Score +
    User_Count, data = Videogames)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020042  0.000090  0.000237  0.000296  0.020302

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  -3.75e-04   4.09e-04    -0.92     0.36
NA_Sales     -1.00e+00   3.18e-04 -3146.04   <2e-16 ***
Global_Sales  1.00e+00   2.08e-04  4817.10   <2e-16 ***
JP_Sales     -1.00e+00   4.03e-04 -2481.31   <2e-16 ***
Other_Sales  -1.00e+00   5.01e-04 -1996.40   <2e-16 ***
Critic_Score -7.55e-07   6.87e-06    -0.11     0.91
Critic_Count  2.56e-06   4.23e-06     0.60     0.55
User_Score    1.41e-05   6.13e-05     0.23     0.82
User_Count    1.31e-07   1.35e-07     0.97     0.33
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7103 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 1.17e+07 on 8 and 7103 DF,  p-value: <2e-16

>
```

```
> #Model2
> lm(EU_Sales ~ NA_Sales+ Global_Sales+JP_Sales+ Other_Sales, data= Videogames)

Call:
lm(formula = EU_Sales ~ NA_Sales + Global_Sales + JP_Sales +
    Other_Sales, data = Videogames)

Coefficients:
 (Intercept)      NA_Sales  Global_Sales      JP_Sales
   -0.000243     -0.999759      0.999901     -0.999849
 Other_Sales
   -0.999849

> model2<- lm(EU_Sales ~ NA_Sales+ Global_Sales+JP_Sales+ Other_Sales, data=
Videogames)
> summary(model2)

Call:
lm(formula = EU_Sales ~ NA_Sales + Global_Sales + JP_Sales +
    Other_Sales, data = Videogames)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020098  0.000147  0.000228  0.000244  0.020221

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.43e-04   7.61e-05    -3.2   0.0014 **
NA_Sales     -1.00e+00   3.15e-04 -3171.2   <2e-16 ***
Global_Sales  1.00e+00   2.05e-04  4879.1   <2e-16 ***
JP_Sales     -1.00e+00   3.96e-04 -2525.6   <2e-16 ***
Other_Sales  -1.00e+00   5.00e-04 -1997.8   <2e-16 ***
---
```

```
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7107 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 2.33e+07 on 4 and 7107 DF,  p-value: <2e-16

>
>
> vg <- Videogames[,-c(1,2,3,4,5,15)]  # Reduced dataframe(got down from 15 varibles to 9 variables)
>
> lm(EU_Sales~., data=vg)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Coefficients:
 (Intercept)      NA_Sales      JP_Sales   Other_Sales
   -3.75e-04     -1.00e+00     -1.00e+00     -1.00e+00
Global_Sales  Critic_Score  Critic_Count    User_Score
    1.00e+00     -7.55e-07      2.56e-06      1.41e-05
  User_Count
    1.31e-07

> model3<-lm(EU_Sales~., data=vg)
> summary(model3)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020042  0.000090  0.000237  0.000296  0.020302
```

```
Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  -3.75e-04   4.09e-04    -0.92     0.36
NA_Sales     -1.00e+00   3.18e-04 -3146.04   <2e-16 ***
JP_Sales     -1.00e+00   4.03e-04 -2481.31   <2e-16 ***
Other_Sales  -1.00e+00   5.01e-04 -1996.40   <2e-16 ***
Global_Sales  1.00e+00   2.08e-04  4817.10   <2e-16 ***
Critic_Score -7.55e-07   6.87e-06    -0.11     0.91
Critic_Count  2.56e-06   4.23e-06     0.60     0.55
User_Score    1.41e-05   6.13e-05     0.23     0.82
User_Count    1.31e-07   1.35e-07     0.97     0.33
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7103 degrees of freedom
Multiple R-squared:      1,	Adjusted R-squared:      1
F-statistic: 1.17e+07 on 8 and 7103 DF,  p-value: <2e-16

>
> #Model4 same as model 1
> lm(EU_Sales~., data=vg)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Coefficients:
 (Intercept)      NA_Sales      JP_Sales   Other_Sales
   -3.75e-04     -1.00e+00     -1.00e+00     -1.00e+00
Global_Sales  Critic_Score  Critic_Count    User_Score
    1.00e+00     -7.55e-07      2.56e-06      1.41e-05
  User_Count
    1.31e-07
```

```
> model4<-lm(EU_Sales~., data=vg)
> summary(model4)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020042  0.000090  0.000237  0.000296  0.020302

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  -3.75e-04   4.09e-04    -0.92     0.36
NA_Sales     -1.00e+00   3.18e-04 -3146.04   <2e-16 ***
JP_Sales     -1.00e+00   4.03e-04 -2481.31   <2e-16 ***
Other_Sales  -1.00e+00   5.01e-04 -1996.40   <2e-16 ***
Global_Sales  1.00e+00   2.08e-04  4817.10   <2e-16 ***
Critic_Score -7.55e-07   6.87e-06    -0.11     0.91
Critic_Count  2.56e-06   4.23e-06     0.60     0.55
User_Score    1.41e-05   6.13e-05     0.23     0.82
User_Count    1.31e-07   1.35e-07     0.97     0.33
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7103 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 1.17e+07 on 8 and 7103 DF,  p-value: <2e-16

>
```

> # corellation of the new dataframe

```
> cor(vg)
             NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales
```

```
NA_Sales       1.0000    0.838   0.4659      0.7276       0.9549
EU_Sales       0.8382    1.000   0.5185      0.7182       0.9385
JP_Sales       0.4659    0.519   1.0000      0.3935       0.6119
Other_Sales    0.7276    0.718   0.3935      1.0000       0.8054
Global_Sales   0.9549    0.938   0.6119      0.8054       1.0000
Critic_Score   0.2341    0.212   0.1458      0.1915       0.2373
Critic_Count   0.2863    0.268   0.1711      0.2420       0.2932
User_Score     0.0857    0.055   0.1271      0.0567       0.0879
User_Count     0.2458    0.284   0.0737      0.2420       0.2649
             Critic_Score Critic_Count User_Score User_Count
NA_Sales           0.234        0.286     0.0857     0.2458
EU_Sales           0.212        0.268     0.0550     0.2844
JP_Sales           0.146        0.171     0.1271     0.0737
Other_Sales        0.191        0.242     0.0567     0.2420
Global_Sales       0.237        0.293     0.0879     0.2649
Critic_Score       1.000        0.392     0.5837     0.2643
Critic_Count       0.392        1.000     0.1937     0.3611
User_Score         0.584        0.194     1.0000     0.0189
User_Count         0.264        0.361     0.0189     1.0000
```

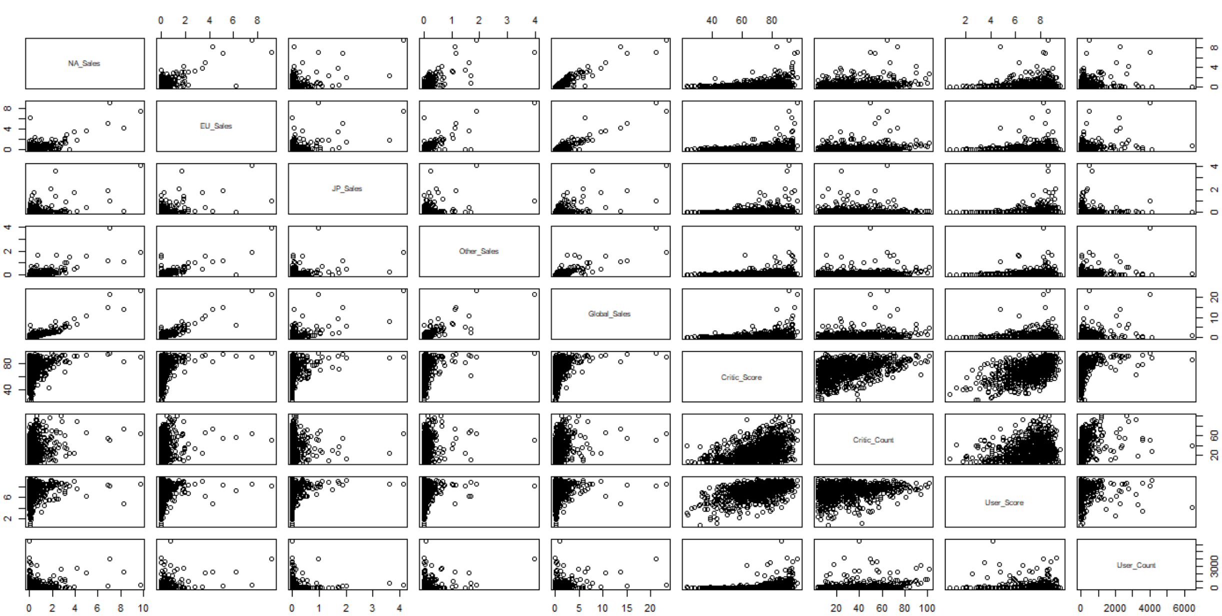
```
> # performing pairwise scatterplor
> # plotting a scatterplot for 1000 random obeservation from vg dataframe, without
replacement
>
> nrow(vg) # gives the total number of rows in a dataframe
[1] 7112
> sample(1:nrow(vg), 1000, replace = FALSE)
   [1]  592 5769  996 5124 6264 5029 5227 1301 1061 5445 2715 2410
  [13]  180 4375 4437 6325 4158 2662 4169 4021 3187 2057 2812 3656
  [25] 2332 5027 3276 2683  560 5546 4937 1642 5019 5278 3275  107
  [37] 2292 3664 5693 2349 6091 3974 6624 6046 6515  747 2657 5422
```

```
 [49] 4544 1743 5087 6742 5194  599 6927  733 4646 5825 2461 4863
 [61] 3781 3690 3600 6781  184 4276 3424 3030 4252 5128 6631  624
 [73] 7105 1103 5582 4629 5815  153 1104  752 3012 3431  811 4507
 [85] 5491 6245 7098 3283 6030 6306  922 3354 2787 2889 1926  335
 [97] 6444 6259  921 2849 3027 2626 4776  897 2389 1862 6192 5401
[109] 1690  299 2387  212  176 4023 3840 2120 5530 3976 3263 3184
[121] 2670 2960 6951 6307 5877 1431  372 5165 4899 4951 5349 4414
[133] 2107 5631 2768 4470 4219 3649 5177  152 1526 5956 5342  344
[145] 3916 1025 6678 2725 2834 2351 1854 4586  979 3375 5949   61
[157] 2160 5461 3196 2793 6560  360 2320 1549  627 1287 1929 5149
[169]   91 3722  200 4144  433 6498 5736 1122 3229 3736 4303 3881
[181] 6368 1170 5042 5645 2695 5148 6902 1617 1319 1093  813 3771
[193] 1206 3220 3820 3818 2093 5058 3883  647  100 3528 1272 6924
[205] 6829 3019 6692 4648  142 5709 2216 5580 3189  131 2383 5735
[217] 6549  817 4293 3804 1877 4538 1773 2172 1607 2906 2887 6785
[229] 4982 5884 6467 4552 6717 6671 3432  799 6676 5012 5912 1990
[241] 1324 1673 3954  129 2777 4948 5988 1629 1903 5730 6969 2077
[253] 4183 6080 4849  835 6945 5640 6438 4341 4755  779  981 4930
[265] 3035 6622  322 5384 3454 3067   38 6680 2758 4285 7048 4720
[277] 5122  827 4225  406 5911 2315  763 1556  489 1135 5496 6875
[289] 4357  610 2434 7076 1344 1066 3718 6166 1006  584 2707 3802
[301] 4657  341 3864 2851 2643 5282 1310 5929 4306 1519 2935 4581
[313] 6246 6301 6800 2390 2990  214 2275 6887  615 6657 6414 1213
[325]  357 4532 2096 3659 6168  211 5199 1111 5830 4622 1432 2697
[337] 1412  122 4871 4917 3654 6148 4238  874 2479 7063 3230 3789
[349] 4765 7039 5762 4157 2106 5486 3559 5668 4614 6050  879 1110
[361]  207 3091 2553 2983 6979 6977 2781 5385 6023 2205 4907 1052
[373] 2125 1215   31 1041 4617  596 5678 4596 2118  534 2798 1167
[385] 1187 4531  988 1980 3555 2458  210 1442 1420 3182 3841  266
[397] 4897 3463 6587  764  883  283 4572 2259 6684 4571 3345  790
[409] 3927 2374 4550 6823 6543 1460 1927 1106  376 6131 3565 2816
[421] 5025 1417 6929 3991 3977 1659 4028 4594 5931  274 4853 5548
[433] 6231 2191 2197 2185 6466 1594 3866 3835 6833 1497 7047 5315
```

[445] 3491 990 6492 6314 900 6145 3624 795 1237 3619 2828 6338
[457]   8 5187 5604 626 4398 5987 3309 5216 2163 145 2582 2319
[469] 5691 3994 5739 6725 2529 3677 1615   97 5655 6527 4519 5996
[481] 554 3154 845 3549 6230 6554 1724 3131 5484 4430 4728 315
[493] 4621 3894 3566 3790 4195 6136 4719 1051 5982 953 6103 1231
[505] 3605 3844 1685 5259 771 6378 1869 5924 1314 2879 1297 264
[517] 2448 6147 5020 4175 4781 4421 4439 6787 4425 7091 1897 3352
[529] 654 2759 5554 4502 3127 6422 1601 2881 842 4051 5529 5303
[541] 3364 563 1924   24 6501 5627 1267 4749 4373 718 5555 7014
[553] 3142 2523 5685 5688 1905 5231 4257 3570 6141 5910 1441 908
[565] 5189 257 1639 4613 4387 5764 2215   81 5443 672 2186 4545
[577] 3571 2740 702 6947 2905 706 1987 6331 5204 4025 6545 6449
[589] 657 289 6204 6173 6790 1825 825 6405 5126 1338 5009 6620
[601] 4239 6760 6714 6028 5941 1014 785 4394 694 4007 5388 4768
[613] 1120 3168 1635 4962 4420 3527 4369 4317 2724 2505 2998 723
[625] 5601 5302 3271 428 936 3842 5108 2134 3058 4869 4626 202
[637] 820 3143 6480 7074 4649 1459 6630 277 6815 386 4200 1989
[649] 5415 2462 3482 2959 2423 4485 5222 1622 2018 2666 1070 3785
[661] 3956 1823 5403 1696 5861 6308 3715 5136 3897 2407 6164 881
[673] 5102   71   37 3621 660 6334 3086 3133 6688 141 511 2062
[685] 2925 6513 804 192 193 5994 1446 5252 1587 3698 2659 5847
[697] 2555 213 3518 4642 6718 1178 2869 2693 6450 1922 1804 6858
[709] 3387 3887 2671 5593 4151 6253 5081 5292 512 261 5897 5200
[721] 139 1695 2773 5524 894 3050 611 1434 3777 1466 6118 1747
[733] 6104 2534 6345 1849 242 4558 295 5370 1000 4980 4364 2861
[745] 116 5198 635 1339 6792 6445   98 591 5088 6528 6090 2717
[757] 609 6220 4111 6709 143 2352 6454 5646 4528 232 1422 4787
[769] 3738 4735 5513 4310 3123 2369 150 1047 1896 2645 5217 860
[781] 1395 2342 1139 1286 1652 5761 2142 3830 4958 2988 5107 6505
[793] 4658 4385 1566 2734 7011 5649 3903 2029 1269 3159 5837 6660
[805] 2078 719 2711 3348 2090 5632 1881 4840 2870 419 1501 924
[817] 2729 2603 4462 5224 1166 6309 3825 5151 5396 2114 439 2095
[829] 3488 4886 3495 1307 2979 562 5389 3130 2043 1143 3731 6988

```
 [841] 3191 3223 4351 3541 3113 2363 6647 3413 5188 3183 2416 3544
 [853] 2655 5212 3396 1275 4332  529 2150 2282 3023 1384 1973 1255
 [865] 6652 5812 1762 6365 2002 4736 5878  182 3162 1394 1886 2827
 [877] 1693 1720 3069 1611 3292 3712 1879 1700 6288 2702 3855 2244
 [889] 4638   76  318 3560 6868 5770 3631  821 4356 1847  935 2919
 [901]  218 5374 1016 2947 5865 3013 2636 5721 4083 2913 6172 1546
 [913] 3010 2048 1017 5062 3486 1614 2413 4140 6548 2098 3350 1574
 [925] 5543 5916 6663 3629 6912  994 3717 4300 7037 3914 3907 6188
 [937] 1095 3129 6984 6852 1683  438 1197 1181 3297 2918 2682 2792
 [949] 4969 2980  855 1577 6113 4418  623 3545 7045 4509 5166 5250
 [961] 5857 3856 5433 2807 4331 3307 4766 1822 4295 4237 3869 3980
 [973] 5504 3399 5674 4905 1755  417 5294 6311 1118 2068 6637 6942
 [985] 3610 3732 3109 5141   60 6940 1392 3233 3112  658 4724 6997
 [997] 6715  154 1898 7085
>
> vgsample<- vg[sample(1:nrow(vg), 1000, replace = FALSE),]  # 1000 random observations from vg
> plot(vgsample)
```

> **# scatter plot is made to identify the second order terms**



> # Second order Term

```
> vg <- Videogames[,-c(1,2,3,4,5,15)]  # Reduced dataframe(got down from 15 varibles to 9 variables)
>
>
> lm(EU_Sales~., data=vg)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Coefficients:
 (Intercept)      NA_Sales      JP_Sales   Other_Sales
   -3.75e-04     -1.00e+00     -1.00e+00     -1.00e+00
Global_Sales  Critic_Score  Critic_Count    User_Score
    1.00e+00     -7.55e-07      2.56e-06      1.41e-05
  User_Count
    1.31e-07

> model4<-lm(EU_Sales~., data=vg)
> summary(model4)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020042  0.000090  0.000237  0.000296  0.020302

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  -3.75e-04   4.09e-04    -0.92     0.36
NA_Sales     -1.00e+00   3.18e-04 -3146.04   <2e-16 ***
```

```
JP_Sales     -1.00e+00   4.03e-04 -2481.31   <2e-16 ***
Other_Sales  -1.00e+00   5.01e-04 -1996.40   <2e-16 ***
Global_Sales  1.00e+00   2.08e-04  4817.10   <2e-16 ***
Critic_Score -7.55e-07   6.87e-06    -0.11     0.91
Critic_Count  2.56e-06   4.23e-06     0.60     0.55
User_Score    1.41e-05   6.13e-05     0.23     0.82
User_Count    1.31e-07   1.35e-07     0.97     0.33
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7103 degrees of freedom
Multiple R-squared:     1,      Adjusted R-squared:     1
F-statistic: 1.17e+07 on 8 and 7103 DF,  p-value: <2e-16

> vg$User_ScoreSQ<- (vg$User_Score)^2
> vg$Critic_CountSQ<- (vg$Critic_Count)^2
> # F test looks good
> # This shows that atleat 1 beta is not equal to zero|| 99% of variablity global sales is explined by the model
> # Null Hypothesis is rejected and the alternative is accepted
> # t-test looks good- The beta associated with global_sales is equal zero, we can reject that and accept the alternative,that it the beta is not equal to zero and use that estimation
>
>
>
> # Second order Term and Interaction term
> vg <- Videogames[,-c(1,2,3,4,5,15)]  # Reduced dataframe(got down from 15 varibles to 9 variables)
>
>
> lm(EU_Sales~., data=vg)

Call:
```

```
lm(formula = EU_Sales ~ ., data = vg)

Coefficients:
 (Intercept)      NA_Sales      JP_Sales   Other_Sales
   -3.75e-04     -1.00e+00     -1.00e+00     -1.00e+00
Global_Sales  Critic_Score  Critic_Count    User_Score
    1.00e+00     -7.55e-07      2.56e-06      1.41e-05
  User_Count
    1.31e-07

> model5<-lm(EU_Sales~., data=vg)
> summary(model5)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020042  0.000090  0.000237  0.000296  0.020302

Coefficients:
              Estimate Std. Error  t value Pr(>|t|)
(Intercept)  -3.75e-04   4.09e-04    -0.92     0.36
NA_Sales     -1.00e+00   3.18e-04 -3146.04   <2e-16 ***
JP_Sales     -1.00e+00   4.03e-04 -2481.31   <2e-16 ***
Other_Sales  -1.00e+00   5.01e-04 -1996.40   <2e-16 ***
Global_Sales  1.00e+00   2.08e-04  4817.10   <2e-16 ***
Critic_Score -7.55e-07   6.87e-06    -0.11     0.91
Critic_Count  2.56e-06   4.23e-06     0.60     0.55
User_Score    1.41e-05   6.13e-05     0.23     0.82
User_Count    1.31e-07   1.35e-07     0.97     0.33
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.00594 on 7103 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 1.17e+07 on 8 and 7103 DF,  p-value: <2e-16

> vg$User_ScoreSQ<- (vg$User_Score)^2
> vg$Critic_CountSQ<- (vg$Critic_Count)^2
> vg$oth_glo <- (vg$Other_Sales*vg$Global_Sales)
> # F test looks good
> # Null Hypothesis is rejected and the alternative is accepted
> # This shows that at least 1 beta is not equal to zero|| 99% of variability global sales is explained by the model
> # t-test looks good- The beta associated with global_sales is equal zero, we can reject that and accept the alternative,that it the beta is not equal to zero and use that estimation
>
>
> # There is no change in the adjusted r square of the model, since there is no change we can probably take the interaction term out and we will be back to model4
> # Checked with other interaction terms and binary variables, but there is no change in the adjusted r square



> #Checking regression after adding second order term user_score square
> lm(EU_Sales~., data=vg)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Coefficients:
  (Intercept)      NA_Sales      JP_Sales   Other_Sales
    -1.72e-03     -1.00e+00     -1.00e+00     -1.00e+00
 Global_Sales  Critic_Score  Critic_Count    User_Score
```

```
     1.00e+00      -2.65e-06       1.24e-05       4.58e-04
    User_Count    User_ScoreSQ  Critic_CountSQ        oth_glo
     1.53e-07      -3.50e-05      -1.45e-07      -1.94e-05

> model4<-lm(EU_Sales~., data=vg)
> summary(model4)

Call:
lm(formula = EU_Sales ~ ., data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020335  0.000040  0.000213  0.000366  0.020252

Coefficients:
               Estimate Std. Error  t value Pr(>|t|)
(Intercept)    -1.72e-03   9.53e-04    -1.80    0.071 .
NA_Sales       -1.00e+00   3.19e-04 -3130.72   <2e-16 ***
JP_Sales       -1.00e+00   4.11e-04 -2433.45   <2e-16 ***
Other_Sales    -1.00e+00   5.14e-04 -1945.38   <2e-16 ***
Global_Sales    1.00e+00   2.12e-04  4712.68   <2e-16 ***
Critic_Score   -2.65e-06   6.96e-06    -0.38    0.703
Critic_Count    1.24e-05   1.24e-05     1.00    0.320
User_Score      4.58e-04   2.99e-04     1.53    0.126
User_Count      1.53e-07   1.41e-07     1.08    0.279
User_ScoreSQ   -3.50e-05   2.34e-05    -1.50    0.134
Critic_CountSQ -1.45e-07   1.55e-07    -0.94    0.349
oth_glo        -1.94e-05   1.22e-05    -1.59    0.113
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7100 degrees of freedom
Multiple R-squared:     1,     Adjusted R-squared:     1
```

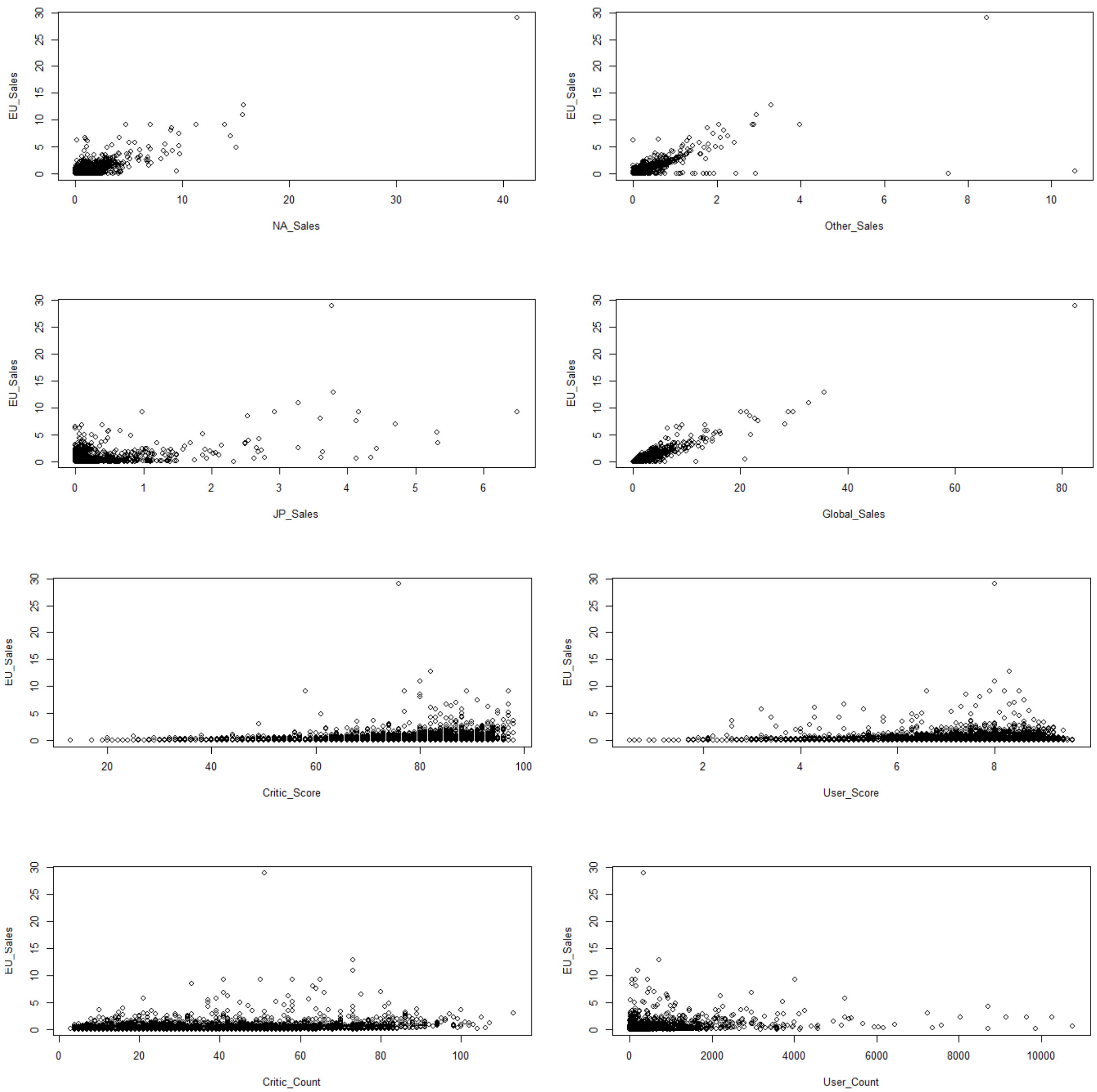
F-statistic: 8.48e+06 on 11 and 7100 DF, p-value: <2e-16

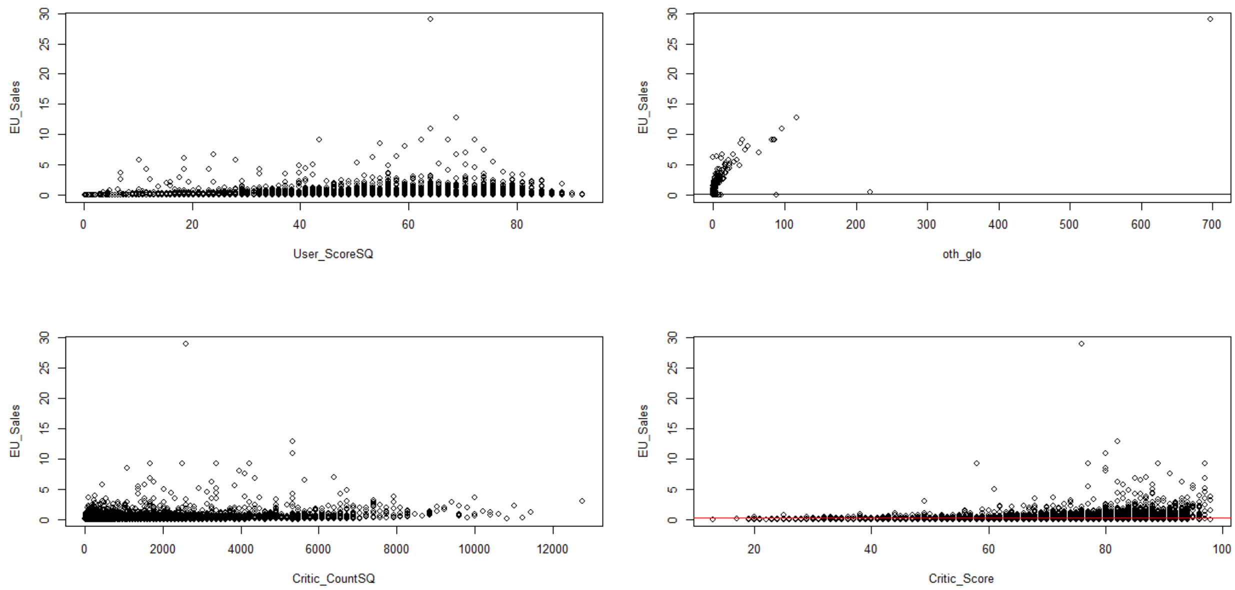
> plot(EU_Sales~., data=vg)

Hit <Return> to see next plot:

Hit <Return> to see next plot:

Hit <Return> to see next plot:





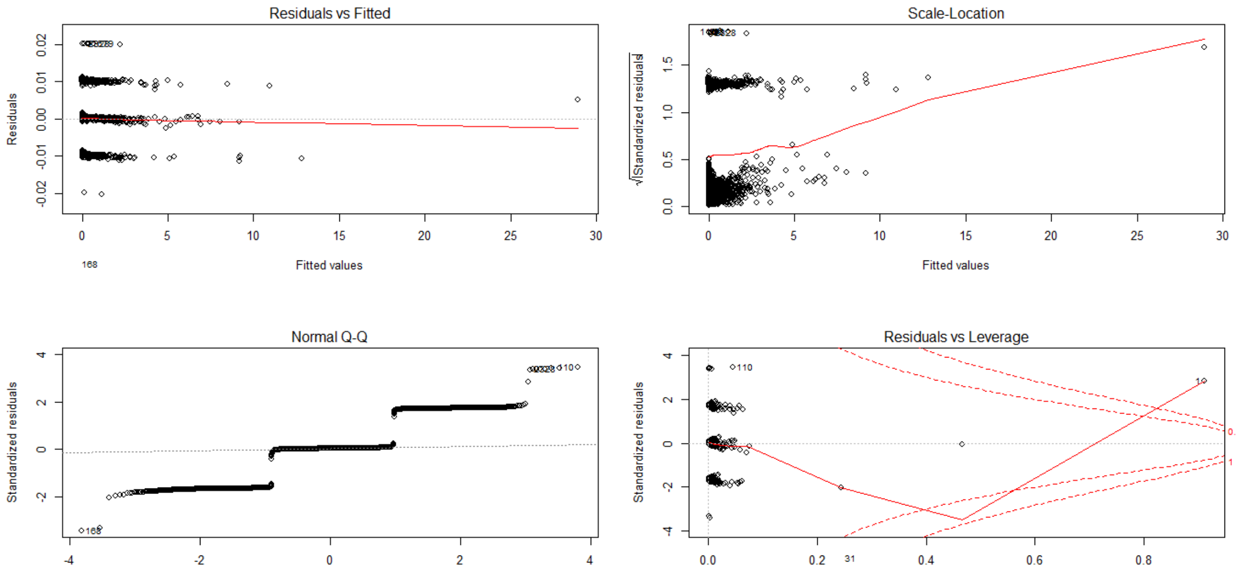

```
> vg$EU_Sales
  [1] 28.96 12.80 10.95  9.15  9.18  6.95  7.48  8.03  4.91  8.49
 [11]  9.14  0.40  9.20  5.17  5.49  5.35  5.09  4.26  3.70  5.73
 [21]  4.26  3.60  6.77  5.73  4.51  4.19  2.56  4.37  3.45  2.81
 [31]  0.01  3.36  3.07  3.87  3.05  4.82  3.64  3.70  2.58  3.11
 [41]  3.27  1.97  2.30  2.47  3.42  2.85  3.63  2.37  2.43  6.75
 [51]  1.85  2.80  6.13  1.53  3.48  2.25  5.01  2.07  6.42  1.72
 [61]  1.72  2.81  3.48  2.36  1.90  3.42  2.11  2.98  2.25  2.83
 [71]  2.99  2.89  2.22  2.14  1.75  1.04  3.02  2.75  2.16  1.90
 [81]  2.75  2.26  2.20  2.75  2.53  4.24  1.80  2.52  1.30  2.61
 [91]  1.58  1.20  1.97  1.34  1.26  6.22  2.80  1.59  1.73  1.88
[101]  4.33  2.01  1.83  1.71  0.00  2.19  1.98  1.47  2.57  0.67
[111]  1.89  2.32  2.07  1.56  1.91  1.93  1.64  1.90  1.24  1.85
[121]  0.55  1.64  2.29  1.83  2.23  1.21  0.77  2.81  1.40  1.69
[131]  1.08  1.55  1.55  0.26  0.75  2.43  3.47  2.17  2.23  0.63
[141]  1.08  1.41  1.80  3.28  1.16  1.99  1.38  0.01  1.17  0.99
[151]  1.72  0.26  1.69  2.00  1.34  1.79  1.57  1.48  2.10  1.27
[161]  1.41  1.95  1.97  1.12  1.39  1.29  0.99  1.11  1.97  1.24
[171]  0.26  0.91  0.24  1.51  1.23  1.29  1.11  1.29  2.39  1.03
```

[181]	1.37	1.79	1.33	1.72	0.58	1.24	1.31	1.11	0.00	2.89
[191]	2.01	1.01	2.55	2.30	1.47	1.82	1.04	0.21	2.22	0.00
[201]	1.93	1.52	1.73	0.04	2.18	1.03	1.47	1.27	1.51	1.85
[211]	2.09	0.87	1.13	1.17	1.76	2.16	0.48	1.35	0.66	1.17
[221]	1.94	1.60	1.89	1.12	1.43	1.56	1.91	1.01	1.56	2.00
[231]	1.69	0.86	1.58	2.48	1.18	1.59	1.21	0.81	1.09	0.00
[241]	0.83	2.45	0.96	1.55	1.22	1.17	0.67	1.02	1.49	0.98
[251]	0.90	1.40	1.77	1.13	2.39	0.49	0.93	0.38	1.42	2.36
[261]	1.02	0.95	1.01	0.34	1.01	1.79	0.71	2.01	1.27	0.97
[271]	1.36	2.07	1.20	1.05	1.03	2.28	1.47	0.92	1.12	1.70
[281]	1.29	1.39	0.91	1.05	1.00	1.11	1.28	0.92	1.68	1.05
[291]	1.57	2.12	0.30	1.11	2.30	1.93	1.00	1.29	1.12	0.59
[301]	0.76	0.56	1.07	0.16	0.00	1.10	1.25	1.03	1.56	1.03
[311]	1.00	0.50	0.85	0.94	0.81	0.28	0.16	1.21	0.98	1.01
[321]	0.88	0.84	0.84	0.79	1.21	1.04	1.85	0.20	1.06	1.32
[331]	0.63	0.00	1.18	1.26	0.86	1.43	1.01	0.80	1.37	1.15
[341]	1.10	0.51	0.70	0.77	0.68	0.77	0.04	0.79	1.14	1.24
[351]	0.86	0.32	1.12	1.04	1.12	0.72	0.82	0.15	0.63	1.31
[361]	0.53	0.09	0.85	0.67	0.55	1.04	0.98	0.22	0.74	0.86
[371]	1.23	1.13	1.43	1.12	0.77	1.14	0.07	0.97	1.19	0.92
[381]	0.52	0.48	0.92	0.11	0.64	1.34	0.12	1.11	1.39	0.51
[391]	0.92	1.09	0.95	0.58	0.63	1.11	1.40	1.11	0.73	0.47
[401]	0.80	0.14	0.12	1.79	0.80	1.22	0.44	0.82	0.21	1.63
[411]	1.05	0.82	0.88	0.84	0.08	1.03	0.27	0.22	0.59	0.55
[421]	1.22	0.57	0.12	0.83	0.19	1.65	1.12	0.85	0.02	0.52
[431]	0.77	0.36	0.84	0.33	0.43	1.04	0.84	0.54	0.53	0.12
[441]	0.93	1.19	0.08	0.60	0.97	0.22	1.17	0.26	0.00	0.70
[451]	0.15	0.37	0.61	0.57	1.32	0.69	0.02	1.08	0.45	0.01
[461]	0.86	0.15	0.13	0.45	0.43	1.09	0.00	0.61	0.70	0.72
[471]	0.43	0.25	1.06	0.64	0.72	0.80	0.91	0.80	0.99	1.27
[481]	1.21	0.81	0.07	0.54	0.60	0.73	0.10	0.73	0.06	0.59
[491]	0.79	0.44	0.15	0.25	0.81	0.63	0.53	0.71	0.91	0.84
[501]	0.62	0.50	0.83	0.72	1.03	0.40	0.85	0.02	0.24	0.96

[511]	0.59	0.25	0.54	0.44	0.96	0.77	1.42	0.20	1.47	0.67
[521]	0.30	0.83	0.67	0.03	1.05	0.68	1.02	0.46	0.81	0.37
[531]	0.61	0.26	0.31	0.94	0.31	0.21	0.31	0.07	0.05	0.12
[541]	0.60	0.63	0.64	0.32	0.71	0.04	0.33	0.56	0.49	0.44
[551]	0.12	0.93	0.52	1.24	0.39	0.41	0.09	0.41	0.62	0.62
[561]	0.79	0.48	1.00	0.04	0.73	0.02	1.02	1.10	0.91	0.48
[571]	0.41	0.58	0.63	0.45	0.18	1.08	0.04	0.51	0.39	0.88
[581]	1.04	0.28	0.66	1.27	0.89	0.64	0.64	0.26	0.92	0.95
[591]	0.58	0.50	0.60	0.93	0.65	1.01	0.61	0.20	0.74	0.02
[601]	0.62	0.23	0.89	0.79	1.26	0.15	0.93	0.58	0.40	0.54
[611]	0.72	0.91	0.96	0.80	0.73	0.16	0.55	0.75	0.96	0.75
[621]	1.03	0.00	0.83	0.14	0.56	0.43	0.72	0.66	0.53	0.03
[631]	0.16	0.79	0.03	0.69	0.44	0.56	0.71	0.42	0.49	0.18
[641]	0.89	0.37	0.05	0.34	0.73	0.82	0.47	0.76	0.07	0.00
[651]	0.33	0.53	0.59	0.44	0.69	0.86	0.20	0.59	0.00	0.74
[661]	0.29	1.01	0.02	0.76	0.29	0.61	0.58	0.02	0.44	0.54
[671]	0.38	0.67	0.46	0.67	0.95	0.87	0.45	0.92	1.08	1.06
[681]	0.53	0.66	0.77	0.78	0.47	0.49	0.52	0.44	0.15	0.68
[691]	0.66	0.18	0.70	0.88	0.03	0.21	0.98	0.50	0.39	0.34
[701]	0.55	0.40	0.91	0.60	0.65	0.05	0.87	0.34	0.60	0.66
[711]	0.12	0.70	0.42	0.52	0.54	0.66	0.00	0.27	0.40	0.46
[721]	0.52	0.05	0.39	0.49	0.42	0.32	0.55	0.58	0.09	0.62
[731]	0.45	1.02	0.61	0.56	0.63	0.50	0.15	0.69	0.12	0.21
[741]	0.11	0.30	0.69	0.72	0.54	0.37	0.63	0.88	0.02	0.80
[751]	0.25	0.34	0.81	0.48	0.62	0.09	0.95	0.76	0.46	0.11
[761]	0.41	0.02	0.62	0.85	0.38	0.60	0.04	0.83	0.48	0.44
[771]	0.12	1.58	0.69	0.36	0.37	1.10	0.26	0.48	0.43	0.70
[781]	0.65	0.58	0.45	0.53	0.35	0.09	0.60	0.57	0.50	0.53
[791]	0.11	0.02	0.48	0.60	0.78	0.40	0.54	0.43	0.67	0.59
[801]	0.80	0.57	0.87	0.55	0.48	0.46	0.44	0.87	0.43	0.02
[811]	0.47	0.47	0.07	0.66	0.65	0.70	0.63	0.52	0.15	0.43
[821]	0.90	0.40	0.31	0.05	0.64	0.44	0.03	0.77	1.16	0.47
[831]	0.62	0.32	0.52	0.78	0.07	0.49	0.20	0.38	0.43	0.00

```
 [841]  0.52  0.69  0.68  0.37  0.57  0.35  0.51  0.56  0.51  0.20
 [851]  0.59  0.40  0.03  0.34  0.19  0.29  0.37  0.40  0.24  0.46
 [861]  0.48  0.47  0.32  0.55  0.31  0.35  0.45  0.75  1.08  0.89
 [871]  0.32  0.08  0.37  0.54  0.56  0.38  0.39  0.75  0.21  0.00
 [881]  0.57  1.13  0.33  0.42  0.37  0.45  0.46  0.43  0.54  0.77
 [891]  0.22  0.08  0.48  0.62  0.47  0.06  0.07  0.40  0.39  0.78
 [901]  0.17  0.54  0.33  0.51  0.28  0.39  0.38  0.44  0.52  1.05
 [911]  0.53  0.18  0.43  0.30  0.69  0.37  0.53  0.47  0.02  0.53
 [921]  0.01  0.41  0.45  0.59  0.33  0.55  0.34  0.57  0.51  0.70
 [931]  0.45  0.57  0.66  0.87  0.52  0.50  0.60  0.47  0.28  0.72
 [941]  0.42  0.52  0.57  0.65  0.05  0.76  0.73  0.89  0.49  0.21
 [951]  0.34  0.44  0.16  0.28  0.43  0.75  0.03  0.22  0.05  0.04
 [961]  0.51  0.35  0.06  0.40  0.50  0.34  0.35  0.51  0.30  0.49
 [971]  0.57  0.49  0.56  0.31  0.13  0.08  0.56  0.41  0.37  0.38
 [981]  0.39  0.58  0.45  0.50  0.35  0.44  0.43  0.23  0.04  0.02
 [991]  0.56  0.41  0.06  0.01  0.39  0.16  0.57  0.47  0.34  0.00
 [ reached getOption("max.print") -- omitted 6112 entries ]
> mean(vg$EU_Sales)
[1] 0.233
>
> mean.vg= mean(vg$EU_Sales)   # mean of the EU sales
> abline(h= mean.vg)  # Ploting a horizontal line at the mean value of the EUsales
>
>
>
> plot(EU_Sales~Critic_Score, data = vg)
> mean(vg$EU_Sales)
[1] 0.233
>
> mean.EU_Sales = mean(vg$EU_Sales)
> abline(h= mean.EU_Sales, col= 'red')
>
> model4<-lm(EU_Sales~., data=vg)
```

```
> abline(h= mean.EU_Sales, col= 'red')
> plot(model4)
```




```
> # ====Pseudo Random Number Generators======
>
> # we will set and seed and generate a random number, this is reproduce our experiments
>
> set.seed(123)
> runif(1)        # generates a psudo random number
[1] 0.288
>
>
>
> # -------test and trained data set -------
# model is buit entirely on trained dataset, but is evaluated on the test data
> vg <- Videogames[,-c(1,2,3,4,5,15)]      # removed X,name and platform columns
>
> # partition of the dataset
>
> partition<- sample(2, nrow(vg), replace = TRUE, prob = c(0.80,0.20))
> # we can give the value over and over again and the distribution is 80 and 20
```

```
>
>
> partition # approximately 80% of these are 1's and aprroximately 20% of these are 2's
  [1] 1 1 2 2 1 1 2 1 1 2 1 1 1 1 2 1 1 1 2 2 1 1 2 1 1 1 1 1 1 2
 [31] 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 2 1 1
 [61] 1 1 1 2 1 2 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 1 1
 [91] 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 1 1 1 2 1 1 2 1 1 1 2 1 1 1
[121] 1 1 1 1 2 1 1 1 1 1 2 1 1 1 1 2 1 2 1 1 1 1 1 2 1 1 1 1 1 2
[151] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 2 1 2
[181] 1 1 1 1 1 1 1 2 2 1 1 2 1 2 1 1 1 1 1 1 2 1 1 1 2 1 1 1 1 1
[211] 1 1 1 1 1 1 1 2 2 1 2 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 2 1 2 1
[241] 1 1 1 1 1 1 2 2 1 1 1 1 1 1 1 1 1 1 2 2 2 1 2 1 1 1 1 1 1 2
[271] 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 2 2 1 1 1 1
[301] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 1 1 2 2 1 1 1 1 1 2 1 1 2 1
[331] 1 1 2 1 1 1 1 1 2 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 2 1 1 1 2 1
[361] 1 2 1 1 1 1 1 1 1 1 1 2 1 1 2 1 1 1 2 1 1 1 1 1 2 1 1 1 1 2
[391] 1 1 1 1 1 1 1 1 2 2 1 2 1 1 1 1 1 1 1 1 2 1 1 1 1 2 1 1 1 1
[421] 1 1 1 2 1 1 1 1 2 2 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1
[451] 1 1 1 1 2 2 1 1 1 2 1 1 1 1 1 1 1 1 2 1 2 1 1 1 1 1 1 1 1 1
[481] 2 1 1 2 1 1 1 1 2 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1
[511] 1 2 1 1 1 1 1 1 2 1 1 1 1 1 1 2 1 2 1 2 1 1 2 1 1 1 2 1 1 2
[541] 1 2 1 1 2 1 2 2 1 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 1 1 2 1 2 1
[571] 2 1 1 2 1 1 1 1 1 2 2 2 1 2 1 1 1 2 1 1 1 2 1 1 2 1 1 2 1 1
[601] 1 1 1 1 2 1 2 1 1 1 1 1 2 1 1 2 1 2 1 2 2 1 1 1 1 1 2 1 1 1
[631] 2 1 1 1 1 2 1 2 1 1 1 2 1 1 1 1 1 1 1 1 2 1 1 1 2 2 1 1 1 2
[661] 2 1 1 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 2 2 2 1 1 1 1 1 1
[691] 1 1 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 1 1
[721] 1 1 2 1 1 2 1 1 1 1 2 1 2 1 1 2 2 1 1 2 1 1 1 1 1 1 1 1 2 1
[751] 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 2 2 1 2 1 1 1 1 1 2 1
[781] 2 1 1 2 1 1 1 1 2 2 1 1 1 2 1 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1
[811] 1 1 2 1 1 1 1 1 1 1 1 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2
[841] 1 1 1 1 1 2 1 1 1 1 1 2 1 1 1 2 1 1 2 1 1 1 1 1 1 1 1 1 1 1
[871] 1 1 1 2 1 1 1 2 1 1 2 1 1 1 2 1 1 1 1 1 1 1 1 2 1 1 1 1 1 2
```

```
 [901] 1 2 1 1 1 1 2 2 1 1 1 2 1 1 2 1 1 1 1 2 1 1 2 1 1 1 1 1 1 2
 [931] 1 1 1 2 1 1 2 2 1 2 1 1 1 1 1 1 1 1 1 1 1 2 1 1 2 1 1 1 1 1
 [961] 1 1 1 2 1 1 1 1 1 1 1 2 1 1 1 2 1 1 1 1 1 2 1 2 1 1 1 1 1 1
 [991] 1 1 1 1 2 1 1 1 1 1
 [ reached getOption("max.print") -- omitted 6112 entries ]
> # we will use this list to create training and test set
>
>
>
> # ----traing dataset---
> train<- vg[partition==1, ]
> partition==1     # when it is true I will accept it into training data and when it is false i will reject it into my training data
   [1]  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
  [11]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE FALSE
  [21]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
  [31] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
  [41]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
  [51]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
  [61]  TRUE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE
  [71]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
  [81]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE FALSE  TRUE  TRUE
  [91]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [101]  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE
 [111]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
 [121]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
 [131] FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE
 [141]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
 [151]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [161]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [171]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE
 [181]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE
 [191]  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

```
[201] FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
[211]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE
[221] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[231]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
[241]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
[251]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
[261] FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
[271]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[281]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[291]  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE  TRUE  TRUE
[301]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[311]  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE
[321]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE
[331]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[341]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[351] FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
[361]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[371]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE
[381]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
[391]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
[401]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[411] FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[421]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
[431]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[441]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[451]  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE
[461]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[471] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[481] FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
[491]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
[501]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
[511]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[521]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE
```

```
[531]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE
[541]  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE
[551]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[561] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE
[571] FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
[581] FALSE FALSE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
[591]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE
[601]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE
[611]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE FALSE  TRUE FALSE
[621] FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
[631] FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE FALSE  TRUE  TRUE
[641]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[651] FALSE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE  TRUE FALSE
[661] FALSE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[671]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[681]  TRUE FALSE FALSE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[691]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
[701]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[711]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
[721]  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[731] FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE FALSE
[741]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[751]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE
[761]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
[771] FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE
[781] FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE
[791]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[801]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
[811]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[821]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
[831]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
[841]  TRUE  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
[851]  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE  TRUE
```

```
 [861]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [871]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE
 [881] FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
 [891]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
 [901]  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
 [911]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE FALSE
 [921]  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE FALSE
 [931]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE
 [941]  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [951]  TRUE FALSE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
 [961]  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [971]  TRUE FALSE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE
 [981]  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
 [991]  TRUE  TRUE  TRUE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE
 [ reached getOption("max.print") -- omitted 6112 entries ]
>
> # ----test dataset---
> test<- vg[partition==2, ]
> partition==2    # here we will get just the opposite of the train set
   [1] FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE
  [11] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE  TRUE
  [21] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
  [31]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
  [41] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
  [51] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
  [61] FALSE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE
  [71] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
  [81] FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE  TRUE FALSE FALSE
  [91] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [101] FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE
 [111] FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
 [121] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [131]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE
```

```
[141] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
[151] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[161] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[171] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE
[181] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
[191] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[201]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[211] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE
[221]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[231] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE
[241] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE
[251] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
[261]  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
[271] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[281] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[291] FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE FALSE FALSE
[301] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[311] FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE  TRUE
[321] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE
[331] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[341] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[351]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE
[361] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[371] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE
[381] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
[391] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE  TRUE
[401] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[411]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[421] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE
[431] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[441] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[451] FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE
[461] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
```

```
[471]  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[481]  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE
[491] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
[501] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[511] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[521] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE
[531] FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE
[541] FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE  TRUE FALSE FALSE
[551] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[561]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE
[571]  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
[581]  TRUE  TRUE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE
[591] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE
[601] FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE
[611] FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE  TRUE FALSE  TRUE
[621]  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
[631]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE  TRUE FALSE FALSE
[641] FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[651]  TRUE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE FALSE  TRUE
[661]  TRUE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[671] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[681] FALSE  TRUE  TRUE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
[691] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
[701] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
[711] FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE
[721] FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
[731]  TRUE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE FALSE  TRUE
[741] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[751] FALSE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE
[761] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
[771]  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE FALSE
[781]  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE
[791] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
```

```
 [801] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [811] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [821] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [831] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
 [841] FALSE FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
 [851] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE FALSE
 [861] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [871] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE
 [881]  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [891] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE  TRUE
 [901] FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE  TRUE FALSE FALSE
 [911] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE  TRUE
 [921] FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE  TRUE
 [931] FALSE FALSE FALSE  TRUE FALSE FALSE  TRUE  TRUE FALSE  TRUE
 [941] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
 [951] FALSE  TRUE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [961] FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
 [971] FALSE  TRUE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE
 [981] FALSE  TRUE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE FALSE
 [991] FALSE FALSE FALSE FALSE  TRUE FALSE FALSE FALSE FALSE FALSE
 [ reached getOption("max.print") -- omitted 6112 entries ]
>
>
> # training dataset has 5685 observations with 9 variables
> # test dataset has 1427 observations with 9 variables
>
>
> model7 <- lm(EU_Sales~ .,data= train)
>
> prediction <- predict(model7,test)
>
> prediction    # these all the predictions of the observations in the test data
        3         4         7        10        15        19
```

1.09e+01	9.15e+00	7.48e+00	8.48e+00	5.49e+00	3.70e+00
20	23	30	31	49	58
5.73e+00	6.77e+00	2.81e+00	2.37e-02	2.43e+00	2.08e+00
64	66	67	86	87	88
2.36e+00	3.42e+00	2.11e+00	4.24e+00	1.80e+00	2.53e+00
103	105	106	110	113	117
1.83e+00	1.19e-03	2.20e+00	6.50e-01	2.07e+00	1.63e+00
125	131	136	138	144	150
2.24e+00	1.09e+00	2.43e+00	2.18e+00	3.28e+00	1.00e+00
172	178	180	188	189	192
9.10e-01	1.28e+00	1.03e+00	1.12e+00	1.12e-02	1.01e+00
194	201	205	218	219	221
2.29e+00	1.93e+00	2.18e+00	1.36e+00	6.60e-01	1.93e+00
229	237	239	247	248	259
1.56e+00	1.21e+00	1.08e+00	6.60e-01	1.01e+00	1.42e+00
260	261	263	270	276	293
2.35e+00	1.01e+00	1.02e+00	9.70e-01	2.28e+00	3.10e-01
295	296	315	316	319	320
2.30e+00	1.94e+00	8.11e-01	2.90e-01	9.80e-01	1.01e+00
326	329	333	339	346	351
1.04e+00	1.06e+00	1.19e+00	1.37e+00	7.80e-01	8.60e-01
355	359	362	372	375	379
1.13e+00	6.30e-01	9.03e-02	1.13e+00	7.70e-01	1.19e+00
385	390	399	400	402	411
6.40e-01	5.10e-01	7.30e-01	4.70e-01	1.40e-01	1.05e+00
416	424	429	430	433	444
1.02e+00	8.30e-01	2.03e-02	5.30e-01	8.40e-01	6.10e-01
455	456	460	469	471	481
1.31e+00	6.90e-01	1.02e-02	7.10e-01	4.30e-01	1.21e+00
484	489	490	495	508	512
5.40e-01	6.02e-02	6.00e-01	8.10e-01	2.02e-02	2.50e-01
519	526	528	530	533	537
1.46e+00	6.70e-01	4.60e-01	3.70e-01	3.10e-01	3.10e-01

540	542	545	547	548	553
1.10e-01	6.20e-01	7.00e-01	3.20e-01	5.50e-01	5.30e-01
561	567	569	571	574	580
7.90e-01	1.02e+00	9.10e-01	4.20e-01	4.60e-01	8.80e-01
581	582	584	588	592	595
1.04e+00	2.90e-01	1.28e+00	2.60e-01	5.00e-01	6.40e-01
598	605	607	613	616	618
2.00e-01	1.26e+00	9.40e-01	9.60e-01	1.70e-01	7.40e-01
620	621	627	631	636	638
7.50e-01	1.03e+00	7.20e-01	1.70e-01	5.70e-01	4.30e-01
642	651	655	656	660	661
3.60e-01	3.30e-01	7.00e-01	8.60e-01	7.40e-01	2.90e-01
666	676	682	683	684	700
6.10e-01	8.70e-01	6.70e-01	7.81e-01	7.91e-01	3.30e-01
703	718	723	726	731	733
9.20e-01	2.70e-01	3.90e-01	3.20e-01	4.50e-01	6.20e-01
736	737	740	749	757	770
5.10e-01	1.60e-01	2.10e-01	2.00e-02	9.50e-01	4.50e-01
771	773	779	781	784	789
1.20e-01	6.70e-01	4.40e-01	6.50e-01	5.30e-01	5.00e-01
790	794	805	813	823	840
5.30e-01	5.90e-01	4.80e-01	7.00e-02	3.10e-01	-5.08e-05
846	852	856	859	874	878
3.60e-01	3.90e-01	2.90e-01	2.40e-01	5.40e-01	7.50e-01
881	885	894	900	902	907
5.70e-01	3.60e-01	6.20e-01	7.80e-01	5.40e-01	3.80e-01
908	912	915	920	923	930
4.50e-01	1.80e-01	6.80e-01	5.30e-01	4.50e-01	7.00e-01
934	937	938	940	952	955
8.61e-01	6.00e-01	4.60e-01	7.12e-01	4.30e-01	4.30e-01
964	972	976	982	984	995
4.00e-01	5.00e-01	7.99e-02	5.80e-01	5.00e-01	3.90e-01
1003	1004	1024	1040	1048	1049

4.80e-01	4.90e-01	3.10e-01	1.60e-01	3.00e-02	6.70e-01
1051	1052	1068	1071	1072	1073
4.30e-01	8.90e-01	5.10e-01	6.40e-01	1.70e-01	4.00e-02
1088	1090	1096	1100	1104	1107
4.60e-01	6.80e-01	4.00e-02	2.80e-01	3.10e-01	4.00e-02
1108	1111	1117	1126	1131	1140
3.80e-01	2.80e-01	2.30e-01	4.80e-01	3.30e-01	3.60e-01
1143	1150	1153	1157	1169	1173
5.00e-01	6.70e-01	4.40e-01	3.03e-02	3.20e-01	3.00e-01
1183	1196	1198	1200	1201	1205
7.30e-01	4.00e-02	4.99e-02	3.70e-01	5.00e-01	4.50e-01
1211	1215	1217	1219	1223	1224
-6.72e-05	2.50e-01	8.99e-02	3.20e-01	4.60e-01	-5.95e-05
1228	1238	1244	1250	1255	1258
1.70e-01	6.90e-01	3.90e-01	3.40e-01	5.70e-01	4.00e-01
1267	1269	1274	1281	1290	1293
3.00e-01	5.80e-01	9.90e-03	5.10e-01	1.60e-01	3.90e-01
1298	1303	1312	1320	1322	1324
3.30e-01	5.80e-01	2.70e-01	3.10e-01	1.99e-02	3.90e-01
1338	1345	1357	1362	1375	1377
1.70e-01	4.99e-02	4.10e-01	6.40e-01	2.60e-01	2.99e-02
1382	1384	1387	1390	1395	1407
4.99e-02	3.40e-01	6.20e-01	4.00e-01	1.50e-01	4.10e-01
1409	1411	1420	1423	1427	1428
3.30e-01	5.20e-01	5.98e-02	3.60e-01	3.00e-01	-8.75e-05
1430	1433	1437	1458	1460	1462
1.70e-01	3.01e-02	3.10e-01	5.30e-01	3.50e-01	3.50e-01
1465	1466	1477	1478	1479	1482
1.60e-01	4.10e-01	6.20e-01	3.20e-01	3.40e-01	2.60e-01
1483	1484	1489	1492	1495	1496
2.50e-01	7.99e-01	2.60e-01	3.40e-01	1.40e-01	3.00e-02
1500	1501	1511	1512	1530	1532
2.60e-01	1.30e-01	2.20e-01	1.10e-01	3.50e-01	4.60e-01

1538	1549	1550	1559	1562	1567
2.30e-01	2.30e-01	4.00e-01	2.90e-01	2.70e-01	4.20e-01
1569	1584	1591	1603	1604	1606
2.70e-01	2.40e-01	9.92e-03	2.50e-01	3.20e-01	2.00e-01
1613	1614	1625	1629	1634	1635
2.30e-01	2.98e-02	3.10e-01	2.20e-01	1.60e-01	1.99e-02
1641	1643	1645	1647	1657	1660
1.10e-01	4.10e-01	4.00e-02	1.80e-01	5.80e-01	-1.44e-04
1666	1682	1689	1693	1694	1695
2.60e-01	2.90e-01	2.40e-01	2.10e-01	2.98e-02	5.00e-01
1696	1703	1704	1708	1713	1715
4.60e-01	2.70e-01	3.10e-01	2.40e-01	1.60e-01	3.20e-01
1718	1720	1730	1733	1734	1737
3.30e-01	2.99e-02	1.70e-01	1.40e-01	1.90e-01	3.70e-01
1738	1739	1741	1746	1748	1750
-3.30e-04	3.10e-01	2.00e-02	1.20e-01	1.90e-01	3.10e-01
1751	1754	1756	1759	1765	1766
2.99e-02	4.00e-01	4.70e-01	3.90e-01	2.80e-01	2.40e-01
1769	1786	1789	1793	1797	1801
3.41e-01	9.79e-03	2.90e-01	1.20e-01	2.90e-01	1.50e-01
1806	1808	1820	1823	1825	1827
1.50e-01	1.80e-01	-1.06e-04	-2.27e-04	4.10e-01	2.60e-01
1830	1832	1838	1850	1851	1856
1.98e-02	2.90e-01	9.88e-03	9.79e-03	3.00e-02	1.40e-01
1859	1863	1867	1871	1875	1878
1.40e-01	4.21e-01	2.60e-01	1.40e-01	1.30e-01	2.70e-01
1880	1891	1894	1897	1903	1918
1.80e-01	3.80e-01	-1.89e-04	1.30e-01	3.40e-01	1.90e-01
1922	1929	1937	1945	1948	1960
1.30e-01	2.70e-01	4.50e-01	2.30e-01	1.80e-01	2.00e-02
1968	1971	1978	1992	1995	2010
2.00e-01	1.80e-01	9.98e-02	2.30e-01	1.60e-01	9.89e-03
2014	2020	2022	2037	2038	2044

1.60e-01	1.10e-01	3.20e-01	3.90e-01	2.00e-01	2.30e-01
2045	2046	2052	2054	2061	2063
3.40e-01	1.20e-01	1.90e-01	2.40e-01	2.10e-01	1.30e-01
2068	2069	2071	2078	2087	2096
2.70e-01	2.60e-01	2.40e-01	2.40e-01	3.50e-01	5.00e-02
2103	2108	2109	2119	2120	2124
2.10e-01	4.50e-01	-1.18e-04	2.99e-02	3.80e-01	1.90e-01
2126	2129	2137	2138	2151	2157
2.90e-01	5.98e-02	2.30e-01	2.30e-01	3.40e-01	1.60e-01
2160	2165	2167	2174	2177	2180
-8.71e-05	9.96e-03	2.20e-01	2.10e-01	-1.40e-04	1.30e-01
2186	2193	2194	2200	2222	2224
9.86e-03	2.40e-01	1.90e-01	2.20e-01	9.97e-02	1.50e-01
2228	2231	2236	2239	2243	2244
8.99e-02	2.50e-01	3.00e-01	1.10e-01	9.97e-02	2.30e-01
2245	2250	2251	2259	2261	2267
3.99e-02	1.60e-01	2.20e-01	1.50e-01	1.10e-01	2.10e-01
2274	2275	2276	2281	2291	2294
3.10e-01	1.10e-01	4.97e-02	3.40e-01	1.80e-01	9.98e-02
2297	2299	2301	2308	2315	2337
2.20e-01	1.70e-01	9.98e-02	1.30e-01	1.40e-01	9.97e-02
2338	2339	2342	2343	2351	2363
2.00e-01	1.10e-01	1.90e-01	7.97e-02	2.99e-02	2.10e-01
2370	2376	2379	2382	2384	2394
2.00e-01	3.20e-01	5.98e-02	2.00e-01	1.70e-01	1.10e-01
2395	2399	2403	2404	2410	2414
2.30e-01	2.00e-01	3.99e-02	1.10e-01	2.00e-01	1.90e-01
2421	2427	2445	2450	2454	2455
1.90e-01	1.30e-01	1.40e-01	9.97e-02	1.10e-01	1.90e-01
2462	2478	2484	2493	2495	2496
1.90e-01	2.00e-01	1.20e-01	1.99e-02	2.98e-02	9.97e-02
2500	2508	2525	2529	2530	2534
9.77e-03	7.99e-02	2.30e-01	2.00e-01	1.50e-01	2.98e-02

2538	2539	2540	2545	2547	2554
3.40e-01	9.95e-03	1.40e-01	1.99e-02	1.30e-01	1.20e-01
2561	2565	2569	2582	2586	2587
9.97e-02	4.98e-02	1.20e-01	1.80e-01	2.10e-01	1.90e-01
2595	2598	2603	2619	2620	2636
1.50e-01	4.80e-01	9.88e-03	3.97e-02	1.80e-01	1.40e-01
2641	2644	2646	2650	2651	2652
9.97e-02	1.70e-01	1.50e-01	1.70e-01	3.98e-02	1.20e-01
2677	2682	2683	2687	2689	2695
-1.00e-02	1.70e-01	9.81e-03	1.70e-01	-2.21e-04	3.70e-01
2697	2699	2705	2706	2707	2708
1.50e-01	1.20e-01	8.98e-02	1.30e-01	1.40e-01	8.97e-02
2710	2715	2720	2723	2724	2725
1.50e-01	1.70e-01	-2.59e-04	9.83e-03	1.20e-01	3.30e-01
2732	2734	2739	2744	2757	2775
1.98e-02	1.60e-01	3.98e-02	1.20e-01	1.97e-02	1.10e-01
2783	2784	2786	2789	2794	2799
9.81e-03	9.98e-02	1.60e-01	1.60e-01	3.97e-02	1.70e-01
2807	2809	2810	2813	2814	2817
2.50e-01	9.76e-03	1.70e-01	7.97e-02	5.98e-02	2.10e-01
2823	2833	2836	2839	2842	2850
1.60e-01	1.60e-01	1.60e-01	1.50e-01	4.97e-02	1.50e-01
2854	2857	2867	2875	2879	2882
2.40e-01	1.70e-01	9.98e-02	2.00e-02	1.50e-01	2.70e-01
2885	2889	2891	2900	2904	2906
6.98e-02	9.77e-03	1.60e-01	1.50e-01	1.50e-01	7.97e-02
2912	2914	2927	2941	2945	2952
1.40e-01	1.50e-01	6.98e-02	9.81e-03	1.70e-01	1.20e-01
2955	2960	2966	2976	2978	2985
1.20e-01	1.97e-02	1.40e-01	3.99e-02	8.98e-02	9.97e-02
2986	2992	2994	2996	3001	3008
1.20e-01	1.50e-01	1.50e-01	5.99e-02	4.96e-02	1.40e-01
3009	3010	3016	3018	3025	3026

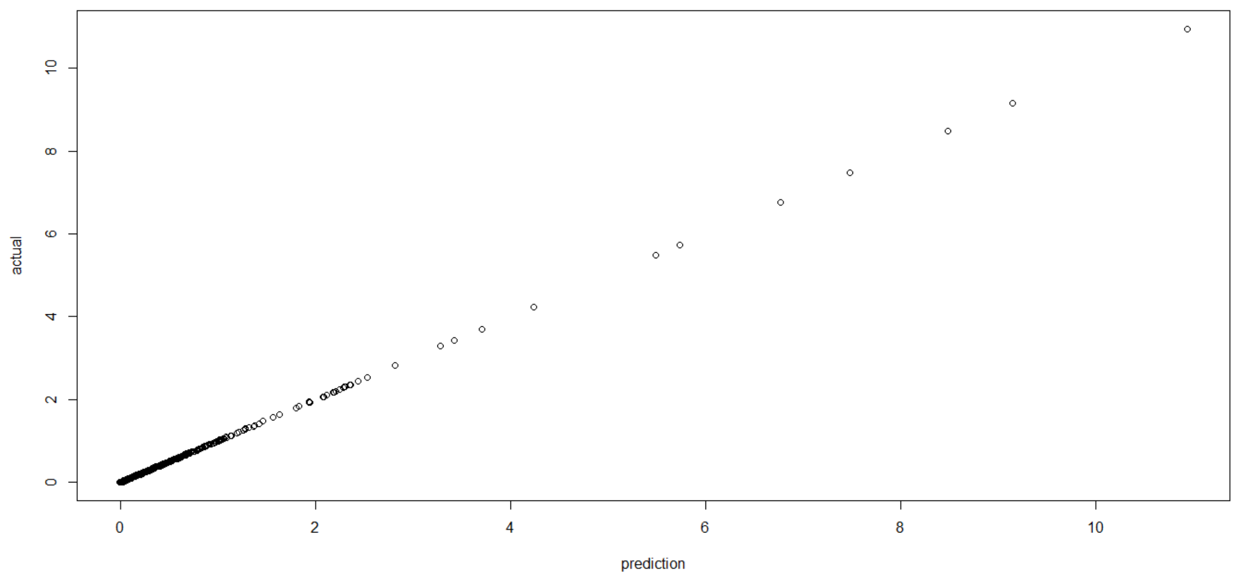
5.98e-02	3.98e-02	1.98e-02	9.01e-02	6.97e-02	-2.35e-04
3027	3029	3039	3049	3052	3055
1.40e-01	1.10e-01	8.97e-02	-2.01e-04	1.40e-01	1.50e-01
3059	3060	3067	3070	3071	3075
7.97e-02	2.20e-01	-1.48e-04	1.30e-01	6.97e-02	1.40e-01
3080	3086	3090	3101	3106	3108
2.00e-01	6.97e-02	7.97e-02	1.70e-01	8.98e-02	2.97e-02
3115	3117	3122	3123	3125	3129
1.10e-01	2.99e-02	3.20e-01	1.10e-01	7.99e-02	1.97e-02
3132	3140	3147	3151	3162	3173
1.30e-01	7.99e-02	1.30e-01	7.97e-02	1.30e-01	-1.02e-02
3176	3179	3181	3193	3194	3212
9.98e-02	7.97e-02	2.00e-01	6.97e-02	1.40e-01	1.30e-01
3214	3216	3218	3220	3222	3234
1.98e-02	6.98e-02	1.70e-01	6.99e-02	1.80e-01	8.98e-02
3236	3242	3243	3245	3246	3254
3.00e-02	9.81e-03	3.97e-02	1.40e-01	7.98e-02	-1.03e-02
3257	3258	3265	3266	3272	3274
2.20e-01	-2.66e-04	6.98e-02	1.10e-01	-2.68e-04	1.30e-01
3275	3284	3286	3288	3296	3299
1.30e-01	3.97e-02	7.97e-02	6.97e-02	9.72e-03	1.50e-01
3304	3313	3316	3320	3327	3332
6.97e-02	7.98e-02	1.40e-01	1.50e-01	7.98e-02	2.70e-01
3334	3338	3341	3342	3347	3348
6.98e-02	1.60e-01	1.70e-01	9.85e-03	9.98e-02	1.80e-01
3350	3352	3358	3363	3366	3373
1.50e-01	1.20e-01	1.20e-01	1.30e-01	1.20e-01	9.97e-02
3375	3377	3380	3386	3395	3406
1.98e-02	1.20e-01	-2.91e-04	5.97e-02	6.98e-02	7.98e-02
3408	3413	3414	3416	3418	3422
6.98e-02	-2.74e-04	1.60e-01	7.97e-02	9.98e-02	5.98e-02
3436	3437	3460	3465	3466	3479
3.97e-02	6.97e-02	1.10e-01	3.97e-02	6.97e-02	-3.06e-04

3486	3487	3488	3493	3496	3502
9.98e-02	-2.35e-04	2.98e-02	2.30e-01	5.97e-02	9.74e-03
3509	3510	3528	3529	3531	3536
5.98e-02	1.10e-01	5.97e-02	3.98e-02	5.00e-02	9.66e-03
3541	3544	3545	3550	3553	3555
4.97e-02	2.50e-01	8.00e-02	-3.21e-04	9.82e-03	1.40e-01
3565	3567	3570	3573	3574	3583
1.98e-02	6.97e-02	9.98e-02	8.96e-02	4.97e-02	9.91e-03
3584	3595	3597	3605	3606	3608
9.97e-02	9.97e-02	9.98e-02	5.98e-02	5.97e-02	5.97e-02
3625	3634	3642	3645	3649	3650
8.97e-02	-2.60e-04	4.97e-02	1.20e-01	9.96e-02	7.98e-02
3654	3658	3663	3673	3674	3676
1.98e-02	-1.88e-04	-5.45e-05	1.10e-01	-1.92e-04	1.10e-01
3681	3683	3686	3687	3688	3702
5.98e-02	8.99e-02	-2.58e-04	4.97e-02	1.10e-01	9.69e-03
3704	3712	3717	3720	3723	3730
9.98e-02	9.97e-02	-2.39e-04	-2.47e-04	1.98e-02	9.80e-03
3735	3736	3738	3740	3742	3750
5.98e-02	-2.76e-04	2.99e-02	1.96e-02	9.97e-02	9.98e-02
3752	3761	3763	3765	3767	3770
9.97e-02	2.98e-02	2.97e-02	9.85e-03	9.97e-02	5.97e-02
3771	3772	3775	3776	3782	3784
2.97e-02	9.97e-02	-2.46e-04	9.75e-03	2.97e-02	9.97e-02
3793	3795	3797	3800	3805	3806
5.98e-02	9.77e-03	8.96e-02	4.98e-02	4.98e-02	9.83e-03
3818	3820	3824	3830	3835	3842
1.90e-01	4.97e-02	9.97e-02	4.97e-02	9.97e-02	3.97e-02
3849	3850	3862	3867	3869	3871
4.97e-02	5.97e-02	6.97e-02	8.98e-02	5.98e-02	3.97e-02
3874	3879	3880	3884	3898	3907
6.98e-02	-2.52e-04	4.98e-02	-2.30e-04	8.97e-02	9.98e-02
3908	3918	3919	3922	3926	3929

9.76e-03	8.98e-02	1.40e-01	3.98e-02	4.99e-02	3.99e-02
3930	3931	3932	3935	3940	3946
1.40e-01	5.99e-02	1.98e-02	1.10e-01	1.10e-01	9.75e-03
3947	3950	3951	3962	3968	3971
-2.95e-04	1.96e-02	7.99e-02	8.97e-02	3.97e-02	7.97e-02
3973	3974	3977	3978	3982	3985
1.40e-01	1.98e-02	-2.25e-04	4.98e-02	3.98e-02	6.97e-02
3993	4001	4003	4006	4009	4011
9.80e-03	8.97e-02	1.97e-02	-3.15e-04	8.97e-02	9.72e-03
4015	4016	4030	4035	4042	4044
2.98e-02	1.30e-01	4.97e-02	-2.23e-04	5.97e-02	9.96e-02
4049	4056	4074	4075	4076	4077
1.98e-02	4.98e-02	2.97e-02	3.97e-02	9.69e-03	7.97e-02
4083	4085	4097	4103	4106	4109
7.96e-02	7.97e-02	3.97e-02	6.97e-02	6.97e-02	2.98e-02
4119	4120	4121	4124	4125	4133
4.97e-02	3.98e-02	7.97e-02	-2.28e-04	-1.03e-02	3.97e-02
4139	4152	4153	4159	4169	4171
6.98e-02	3.97e-02	9.74e-03	9.72e-03	3.97e-02	4.97e-02
4180	4183	4197	4207	4211	4228
9.68e-03	-2.10e-04	7.99e-02	-1.02e-02	1.90e-01	-1.03e-02
4233	4241	4246	4248	4249	4250
5.97e-02	6.97e-02	3.97e-02	-3.05e-04	7.98e-02	8.98e-02
4253	4255	4265	4266	4269	4271
7.98e-02	9.79e-03	3.97e-02	6.97e-02	9.98e-02	6.97e-02
4280	4301	4302	4303	4306	4315
1.97e-02	2.97e-02	2.97e-02	4.96e-02	1.50e-01	5.98e-02
4331	4341	4353	4356	4359	4360
3.97e-02	7.97e-02	3.99e-02	2.97e-02	6.97e-02	9.71e-03
4365	4366	4369	4373	4374	4375
4.97e-02	7.96e-02	7.97e-02	4.98e-02	7.97e-02	9.98e-02
4390	4391	4397	4400	4401	4406
2.98e-02	7.97e-02	6.02e-02	3.97e-02	9.77e-03	7.98e-02

4408	4415	4425	4426	4438	4442
4.97e-02	-1.02e-02	2.98e-02	8.96e-02	5.99e-02	1.10e-01
4452	4464	4465	4466	4470	4483
1.97e-02	3.98e-02	9.04e-02	5.99e-02	3.97e-02	8.99e-02
4510	4521	4527	4533	4544	4548
3.98e-02	6.98e-02	6.97e-02	6.99e-02	-2.43e-04	-1.20e-04
4565	4567	4569	4571	4575	4583
2.99e-02	-3.05e-04	4.96e-02	4.97e-02	1.30e-01	2.97e-02
4584	4589	4593	4601	4610	4612
1.20e-01	3.97e-02	2.98e-02	6.98e-02	5.97e-02	4.97e-02
4614	4619	4622	4627	4629	4634
5.97e-02	6.97e-02	7.97e-02	5.96e-02	9.86e-03	6.98e-02
4635	4649	4652	4653	4656	4660
9.68e-03	1.98e-02	4.98e-02	2.98e-02	4.98e-02	-2.72e-04
4667	4685	4686	4687	4701	4702
9.75e-03	1.40e-01	-2.15e-04	2.97e-02	9.79e-03	5.97e-02
4708	4709	4719	4724	4729	4731
2.98e-02	3.97e-02	3.97e-02	-2.56e-04	2.97e-02	7.99e-02
4732	4736	4737	4746	4751	4753
4.98e-02	2.97e-02	2.97e-02	-1.80e-04	5.98e-02	1.97e-02
4754	4766	4767	4772	4794	4796
6.98e-02	-1.02e-02	-2.41e-04	-3.09e-04	3.97e-02	2.97e-02
4798	4815	4817	4823	4826	4848
2.98e-02	5.97e-02	3.98e-02	9.73e-03	2.97e-02	-2.55e-04
4861	4862	4870	4882	4889	4890
2.98e-02	4.98e-02	-2.30e-04	1.98e-02	4.97e-02	-3.03e-04
4901	4902	4903	4907	4925	4933
7.96e-02	-2.76e-04	2.97e-02	7.97e-02	-2.95e-04	3.98e-02
4934	4944	4958	4960	4961	4970
1.20e-01	9.79e-03	2.97e-02	9.68e-03	4.03e-02	1.97e-02
4972	4973	4980	4986	4992	4996
9.81e-03	-2.11e-04	1.97e-02	3.97e-02	1.97e-02	3.97e-02
5001	5011	5013	5014	5015	5018

```
 3.97e-02  1.20e-01 -1.03e-02  3.96e-02  3.97e-02  1.98e-02
     5020      5021      5027      5035
 4.96e-02  4.97e-02 -2.79e-04  6.97e-02
 [ reached getOption("max.print") -- omitted 405 entries ]
>
> # Actual EU_Sales can be identified
> actual = test$EU_Sales
>
> # corellation between the actual and predicted value of EU_Sales
> cor(prediction,actual)
[1] 1
>
>
> par(mfrow = c(1,1))
> plot(prediction,actual)
> # the plot looks very good
```



----- Cross Validation-----

K-fold cross validation to do that we have to use the DAAG package

```
install.packages('DAAG')
library(DAAG)


# cross validation linear model
#Performed 3 fold cross validations
out<- cv.lm(data=vg, form.lm = formula(EU_Sales~.), plotit = "Observed", m= 3)


#results for first fold
# Sum of squares = 0.09    Mean square = 0    n = 2370

#results for second fold
#Sum of squares = 0.08    Mean square = 0    n = 2371


#results for third fold
# Sum of squares = 10362    Mean square = 4.37    n = 2371


# Avg mean squre error
# 1.46

        4166    4169    4172    4173    4174    4178
        4182    4185    4186    4188    4199    4200
        4203    4204    4207    4212    4215    4217
        4222    4223    4224    4225    4226    4229
        4232    4234    4235    4236    4241    4242
        4243    4246    4249    4251    4269    4270
        4273    4274    4283    4284    4287    4288
        4291    4292    4294    4295    4298    4299
        4304    4305    4307    4308    4309    4310
        4313    4315    4325    4328    4330    4336
```

4337	4338	4340	4345	4346	
4352	4353	4357	4358	4360	
4361	4362	4363	4365	4367	
4372	4374	4377	4379	4380	
4382	4383	4384	4388	4389	4392
4398	4402	4406	4407	4409	
4411	4414	4423	4424	4426	4431
4437	4440	4442	4443	4444	4451
4453	4457	4459	4462	4468	4471
4474	4479	4483	4484	4486	
4493	4497	4507	4509	4510	
4512	4518	4519	4531	4532	4533
4541	4543	4544	4546	4549	4550
4551	4554	4556	4557	4561	4565
4570	4572	4575	4582	4590	4593
4594	4596	4600	4601	4602	4604
4606	4608	4616	4617	4620	4626
4627	4630	4639	4641	4646	4650
4653	4654	4656	4660	4664	4671
4673	4676	4679	4683	4684	4686
4688	4690	4692	4693	4696	
4698	4700	4702	4703	4704	4708
4713	4714	4717	4718	4721	
4724	4725	4728	4736	4737	
4738	4740	4744	4749	4751	4756
4758	4759	4761	4764	4771	
4772	4775	4778	4780	4783	4787
4788	4791	4792	4793	4802	4805
4806	4810	4813	4825	4828	
4833	4834	4840	4842	4843	
4844	4846	4848	4849	4850	
4852	4859	4861	4864	4865	
4867	4869	4870	4872	4877	4881

4883 4884 4885 4889 4890 4892
4901 4903 4904 4906 4907
4908 4911 4912 4918 4919 4921
4924 4926 4927 4928 4929 4933
4934 4936 4937 4940 4944 4945
4946 4948 4949 4953 4954 4955
4958 4959 4962 4963 4965 4970
4972 4975 4976 4981 4983 4987
4989 4992 4996 5000 5005 5006
5008 5010 5011 5014 5015 5017
5018 5022 5024 5026 5027 5031
5035 5040 5042 5043 5048 5052
5053 5058 5065 5067 5069 5070
5072 5074 5077 5080 5083
5086 5089 5093 5094 5097 5101
5108 5109 5112 5114 5115 5117
5118 5119 5120 5125 5126 5132
5134 5138 5139 5142 5146 5148
5158 5163 5171 5174 5181 5192
5197 5204 5208 5214 5215
5216 5219 5224 5225 5228 5237
5242 5245 5247 5251 5253
5256 5264 5269 5275 5276 5285
5286 5288 5292 5299 5300 5302
5303 5307 5310 5312 5315 5316
5318 5320 5321 5328 5334 5335
5336 5337 5338 5348 5349
5350 5354 5357 5358 5364
5369 5371 5373 5374 5378 5379
5380 5386 5391 5396 5397 5398
5401 5404 5409 5411 5414 5417
5418 5420 5422 5424 5432
5449 5450 5454 5456 5459 5467

5468 5469 5470 5474 5475
5477 5480 5481 5482 5484
5486 5491 5495 5499 5504
5506 5510 5514 5518 5519 5520
5522 5524 5528 5534 5539
5542 5543 5544 5545 5547 5549
5550 5552 5553 5554 5559 5561
5564 5567 5569 5570 5580 5581
5582 5583 5587 5591 5596 5598
5599 5601 5602 5607 5609
5611 5617 5621 5629 5631 5638
5639 5641 5642 5643 5645
5648 5649 5667 5668 5672
5676 5680 5682 5684 5685 5686
5687 5691 5695 5696 5700 5703
5704 5711 5716 5717 5718
5719 5723 5727 5728 5731 5735
5737 5742 5748 5752 5753
5755 5759 5760 5761 5766
5767 5769 5772 5775 5776
5778 5780 5784 5787 5793
5797 5804 5806 5809 5814
5815 5816 5819 5822 5823
5828 5832 5834 5835 5837
5838 5843 5851 5855 5857
5863 5864 5865 5867 5869 5872
5874 5878 5884 5885 5895 5896
5897 5900 5904 5908 5912 5916
5917 5919 5920 5921 5925 5932
5933 5934 5935 5937 5938 5939
5940 5942 5950 5958 5959 5961
5966 5967 5970 5973 5974
5978 5981 5982 5984 5987 5989

5990	5993	6001	6004	6011	6014
6015	6023	6024	6026	6028	
6033	6034	6036	6037	6055	
6059	6060	6061	6065	6066	6068
6069	6070	6072	6074	6075	6079
6084	6085	6089	6091	6092	6096
6097	6098	6101	6102	6105	
6110	6115	6116	6119	6122	6127
6128	6129	6132	6136	6141	6145
6155	6158	6176	6177	6178	6179
6183	6185	6186	6188	6190	6196
6200	6201	6204	6208	6209	
6210	6212	6216	6218	6222	6223
6224	6225	6226	6227	6228	
6229	6233	6234	6239	6240	
6242	6244	6246	6252	6256	
6258	6261	6262	6266	6267	
6273	6277	6286	6289	6291	6294
6295	6297	6298	6299	6301	6309
6310	6314	6316	6317	6318	6321
6324	6329	6333	6336	6338	6351
6354	6358	6359	6361	6363	
6370	6371	6373	6374	6379	
6386	6389	6398	6401	6407	6408
6415	6418	6420	6427	6433	6434
6435	6436	6444	6446	6450	6451
6452	6455	6461	6464	6465	
6469	6470	6475	6485	6488	
6489	6491	6499	6500	6501	
6506	6508	6509	6515	6516	6520
6526	6530	6533	6539	6541	6542
6545	6547	6549	6550	6556	6559
6570	6571	6572	6573	6580	

6584	6586	6587	6588	6593	6599
6604	6605	6606	6610	6619	6620
6621	6623	6628	6629	6631	
6634	6635	6639	6642	6646	
6648	6652	6653	6659	6663	6665
6667	6669	6670	6674	6675	6676
6681	6684	6686	6688	6696	
6699	6700	6701	6702	6703	
6704	6705	6707	6712	6714	
6716	6717	6721	6724	6725	6727
6730	6731	6733	6737	6739	
6754	6756	6757	6760	6762	
6763	6764	6767	6772	6778	6781
6788	6792	6793	6795	6797	
6799	6805	6810	6812	6816	6817
6830	6831	6837	6840	6842	6847
6852	6857	6859	6864	6865	
6870	6871	6872	6873	6874	6876
6877	6884	6885	6890	6892	6894
6896	6901	6905	6907	6909	6910
6913	6919	6921	6924	6925	
6929	6930	6931	6950	6953	
6954	6958	6967	6970	6973	
6974	6975	6981	6982	6985	
6989	6990	6993	6994	6995	
6996	6997	6999	7003	7009	
7010	7013	7015	7021	7022	7023
7027	7028	7030	7031	7037	
7038	7041	7044	7048	7049	
7051	7054	7058	7059	7065	
7070	7071	7074	7076	7081	
7083	7088	7090	7094	7097	
7100	7102	7103	7105	7107	7110

[reached getOption("max.print") -- omitted 4 rows]

Sum of squares = 0.09   Mean square = 0   n = 2370

fold 2

Observations in test set: 2371

3	9	11	13	15	16
17	18	21	25	27	29
37	38	44	45	46	50
52	54	55	57	59	60
62	63	69	71	76	77
78	85	86	87	88	92
94	99	100	101	102	106
108	111	113	121	125	136
145	146	147	151	152	156
157	158	159	161	167	172
179	180	181	182	184	
185	187	190	191	192	195
207	208	209	210	214	216
219	220	224	225	227	231
233	236	237	242	246	247
249	250	252	253	258	
259	269	271	276	281	
282	284	287	288	290	293
298	300	301	303	305	306
308	309	311	312	317	318
334	335	336	337	339	342
348	355	356	358	373	374
377	378	381	384	389	391
393	397	398	400	403	407
409	412	415	422	424	426
429	430	432	434	435	437
438	439	441	443	444	445

446 447 449 454 461 464
466 468 470 472 477
479 480 481 496 501 502
507 509 513 514 516 517
520 523 527 529 537 539
540 545 547 549 552 558
559 567 570 574 576 578
580 585 591 592 601 607
608 610 611 617 618 620
624 632 634 636 638 640
648 651 654 666 670 671
672 676 677 683 687 688
689 690 691 692 696 698
699 703 705 707 708 709
716 718 721 725 727 732
739 742 744 752 753 754
759 766 767 768 773 775
778 779 780 783 785 793
794 795 796 805 806 807
812 819 821 826 830 832
833 837 838 839 842 844
847 848 851 855 862 865
867 869 870 872 873 878
881 884 888 890 892
900 902 903 905 906 913
915 917 918 921 925 926
929 937 942 945 946
948 950 957 959 960 965
968 970 971 974 980
982 987 989 990 992 1003
1008 1009 1013 1014 1016 1017
1021 1022 1024 1026 1039
1042 1043 1048 1053 1054 1058

1059 1067 1068 1069 1070
1071 1072 1074 1079 1081 1083
1086 1095 1106 1110 1112 1113
1121 1125 1126 1129 1132 1133
1136 1137 1138 1139 1140 1143
1151 1153 1159 1160 1164 1172
1173 1175 1177 1179 1183 1187
1188 1189 1192 1210 1211 1214
1218 1219 1220 1223 1225 1227
1229 1230 1231 1232 1233 1234
1235 1240 1241 1242 1245 1246
1255 1256 1258 1261 1262 1263 1268
1273 1274 1279 1281 1284 1285
1286 1288 1289 1293 1294 1295
1296 1300 1301 1308 1315 1324
1326 1329 1330 1332 1333 1334
1340 1341 1346 1347 1354 1357
1361 1363 1364 1369 1370 1371
1373 1375 1379 1380 1381
1383 1385 1386 1388 1390 1395
1397 1400 1401 1402 1403 1416
1419 1421 1423 1425 1427 1428
1429 1431 1435 1440 1443 1448
1449 1453 1457 1461 1466
1467 1468 1469 1472 1474
1475 1476 1477 1480 1482 1486
1487 1488 1490 1493 1494 1496
1508 1509 1512 1513 1514 1515
1524 1526 1528 1530 1531 1534
1538 1540 1543 1551 1559 1567
1569 1571 1573 1575 1576 1583
1584 1585 1597 1600 1605 1607
1608 1610 1614 1620 1623

1624 1626 1630 1631 1632 1633
1638 1648 1649 1652 1658
1660 1662 1667 1671 1676 1679
1682 1684 1692 1697 1699 1703
1710 1711 1714 1727 1728 1729
1731 1733 1736 1738 1739 1740
1747 1748 1750 1751 1754 1759
1760 1764 1765 1768 1769 1770
1771 1772 1774 1775 1776 1779
1781 1783 1788 1791 1792
1794 1800 1803 1813 1815 1817
1818 1820 1821 1822 1825 1828
1830 1834 1836 1837 1838 1839
1840 1844 1846 1848 1849
1851 1853 1854 1855 1858 1859
1860 1866 1873 1874 1878
1879 1882 1883 1893 1897
1909 1912 1915 1928 1929 1933
1934 1935 1937 1938 1939 1943
1948 1949 1951 1953 1954 1955
1957 1958 1960 1961 1964 1966
1970 1972 1985 1989 1991 1996
2000 2001 2005 2006 2008 2012
2015 2017 2021 2023 2025 2028
2032 2036 2039 2041 2046 2049
2056 2061 2063 2064 2065 2067
2073 2082 2084 2085 2087 2089
2091 2095 2101 2102 2104
2106 2108 2109 2110 2111
2112 2114 2117 2119 2121 2126
2130 2131 2134 2140 2141 2143
2144 2147 2152 2159 2161 2167
2168 2170 2179 2184 2185 2187

2191 2194 2197 2201 2206 2209
2212 2214 2216 2220 2229
2234 2238 2239 2242 2249 2251
2252 2253 2257 2264 2266 2267
2271 2272 2274 2279 2280 2281
2283 2300 2305 2306 2307 2308
2309 2312 2316 2319 2320 2321
2324 2326 2333 2334 2335 2336
2337 2338 2341 2344 2346 2348
2349 2351 2352 2357 2363 2369
2371 2374 2375 2376 2386 2391
2392 2394 2395 2396 2397 2401
2404 2411 2417 2421 2423 2424
2429 2430 2431 2432 2434 2435
2437 2439 2441 2449 2450 2455
2456 2461 2463 2464 2465 2467
2470 2471 2472 2479 2480 2481
2482 2483 2484 2493 2494 2495
2501 2505 2510 2511 2512 2514
2515 2517 2519 2520 2526
2532 2544 2548 2549 2550 2553
2558 2559 2560 2562 2581 2585
2589 2592 2595 2596 2610 2611
2617 2618 2622 2627 2630 2631
2633 2637 2642 2644 2648 2650
2654 2655 2656 2660 2662 2663
2670 2671 2673 2675 2676 2679
2682 2683 2685 2690 2695 2696
2699 2701 2702 2708 2712 2714
2721 2724 2726 2736 2746 2747
2749 2753 2754 2756 2757 2759
2760 2762 2769 2770 2771 2772
2774 2779 2780 2784 2788 2790

2796 2797 2798 2801 2802 2803
2806 2809 2810 2811 2814
2820 2826 2827 2828 2832 2849
2851 2852 2855 2856 2858 2860
2862 2864 2870 2876 2878 2882
2883 2889 2890 2891 2893 2895
2897 2901 2909 2915 2925 2926
2928 2934 2941 2947 2952 2953
2954 2959 2960 2963 2967 2968
2969 2970 2972 2977 2979 2981
2983 2984 2985 2990 2991 3001
3002 3003 3019 3023 3026 3030
3031 3033 3039 3045 3050 3055
3062 3070 3071 3076 3080 3081
3082 3084 3085 3088 3090 3092
3097 3098 3102 3106 3115 3116
3120 3123 3127 3135 3139 3143
3144 3145 3146 3147 3148 3155
3157 3158 3161 3167 3170 3171
3177 3184 3186 3187 3189 3199
3200 3202 3208 3210 3211 3213
3215 3218 3224 3226 3229 3230
3235 3236 3237 3242 3247
3249 3250 3254 3261 3262 3264
3265 3267 3269 3272 3275
3281 3284 3286 3290 3294 3298
3302 3303 3305 3306 3311 3313
3314 3320 3323 3327 3332 3334
3338 3340 3341 3343 3345 3349
3350 3354 3358 3360 3361 3364
3365 3366 3371 3373 3374 3376
3378 3380 3383 3384 3385 3386
3388 3389 3391 3395 3402 3403

3404 3406 3407 3408 3410 3412
3415 3416 3434 3436 3437 3441
3443 3445 3447 3452 3457 3460
3463 3465 3466 3468 3469 3476
3479 3481 3483 3484 3485 3493
3494 3496 3497 3499 3501 3505
3506 3509 3510 3511 3512 3513
3514 3517 3519 3520 3529 3532
3535 3546 3547 3551 3556
3557 3561 3563 3565 3566 3569
3571 3572 3576 3579 3580 3585
3586 3587 3588 3589 3594 3597
3603 3604 3605 3609 3613 3614
3616 3626 3627 3629 3631 3633
3637 3638 3641 3643 3645 3646
3649 3651 3653 3654 3656 3658
3659 3665 3667 3670 3680 3683
3684 3693 3694 3695 3697 3700
3701 3703 3705 3706 3707
3709 3711 3715 3718 3728
3729 3730 3733 3734 3737 3738
3739 3740 3741 3744 3747 3748
3753 3754 3758 3759 3762 3763
3764 3766 3770 3772 3773 3774
3775 3779 3788 3790 3793 3796
3799 3800 3802 3807 3808
3815 3821 3823 3827 3828
3829 3833 3835 3839 3841 3847
3851 3852 3857 3858 3859 3860
3866 3877 3879 3884 3890
3891 3892 3897 3898 3900 3903
3904 3905 3909 3912 3913
3916 3917 3918 3921 3922 3928

3932 3934 3935 3938 3943 3945
3952 3953 3955 3958 3965 3971
3974 3975 3978 3983 3986 3990
3994 4001 4003 4010 4011
4012 4015 4016 4019 4020 4021
4026 4032 4035 4036 4038
4043 4044 4049 4050 4055 4056
4058 4063 4066 4069 4071
4073 4075 4081 4096 4098
4109 4110 4112 4116 4117 4120
4122 4123 4124 4125 4127 4129
4134 4135 4138 4140 4143 4147
4149 4150 4151 4152 4154 4159
4164 4165 4171 4177 4181
4183 4189 4190 4192 4193
4194 4195 4196 4198 4202 4205
4206 4209 4218 4219 4221 4227
4237 4238 4240 4245 4255 4256
4257 4258 4260 4261 4262 4267
4268 4271 4278 4280 4281 4282
4285 4289 4297 4301 4302 4311
4314 4316 4317 4319 4320 4322
4323 4326 4327 4333 4343 4348
4349 4354 4359 4369 4370 4371
4373 4376 4378 4381 4385 4390
4393 4396 4399 4401 4404 4416
4419 4421 4422 4425 4428 4433
4441 4447 4448 4450 4452 4454
4455 4458 4460 4461 4465 4469
4475 4476 4478 4480 4482 4485
4487 4488 4489 4490 4495 4499
4500 4501 4504 4514 4515 4517
4522 4523 4524 4526 4527 4534

4535 4536 4539 4540 4547
4552 4553 4559 4560 4562 4563
4564 4566 4567 4569 4571 4573
4574 4578 4581 4584 4588 4589
4591 4595 4610 4612 4613 4615
4618 4621 4622 4625 4629 4635
4637 4638 4642 4649 4659 4661
4662 4663 4666 4668 4669 4674
4682 4685 4691 4695 4697 4699
4711 4715 4716 4720 4722 4726
4727 4730 4731 4732 4733 4741
4742 4745 4748 4754 4757 4760
4765 4767 4768 4769 4776
4777 4781 4782 4784 4785
4790 4798 4800 4801 4803 4804
4807 4816 4817 4822 4823 4827
4830 4831 4836 4839 4841 4845
4851 4854 4855 4856 4857 4868
4871 4873 4874 4875 4878 4879
4880 4882 4887 4893 4894 4895
4896 4899 4900 4914 4916 4923
4935 4938 4941 4950 4951
4952 4956 4961 4967 4969 4982
4984 4990 4995 4998 5003 5004
5007 5012 5016 5025 5028
5030 5033 5041 5044 5047 5050
5051 5054 5055 5057 5059 5060
5062 5064 5068 5071 5076 5079
5081 5082 5085 5087 5088 5092
5095 5096 5098 5099 5100
5102 5104 5105 5106 5107
5110 5111 5122 5124 5127 5128
5129 5135 5136 5141 5147 5149

5151	5153	5154	5156	5159	5160
5161	5162	5165	5166	5167	5168
5169	5173	5176	5177	5178	
5180	5182	5184	5186	5187	5189
5191	5195	5198	5199	5200	
5201	5206	5211	5217	5220	5222
5223	5226	5230	5231	5232	5235
5236	5238	5240	5241	5248	5250
5261	5263	5265	5266	5268	5272
5277	5279	5280	5281	5287	
5290	5294	5295	5297	5301	
5304	5306	5309	5311	5313	5319
5323	5324	5330	5331	5333	
5341	5342	5343	5344	5346	
5351	5352	5353	5359	5363	
5365	5366	5382	5383	5384	
5387	5390	5399	5400	5402	5405
5408	5410	5412	5415	5419	5421
5426	5429	5430	5433	5435	
5438	5440	5441	5442	5443	
5444	5445	5446	5448	5455	
5457	5460	5461	5465	5471	
5485	5489	5490	5492	5494	5498
5501	5508	5509	5512	5513	
5516	5525	5526	5530	5531	5532
5535	5536	5546	5551	5556	5558
5560	5562	5563	5572	5574	5575
5576	5588	5589	5590	5592	5594
5597	5600	5604	5606	5608	5612
5614	5615	5616	5618	5619	
5620	5622	5624	5625	5627	5628
5630	5632	5633	5634	5636	
5637	5644	5646	5651	5654	5655

5658	5661	5663	5666	5671	5674
5677	5678	5679	5689	5694	5697
5698	5701	5702	5705	5706	5710
5712	5713	5721	5725	5729	5732
5733	5734	5736	5743	5744	
5746	5747	5751	5757	5762	
5765	5770	5771	5779	5781	5782
5785	5790	5792	5795	5800	
5801	5807	5808	5810	5811	5818
5820	5821	5824	5825	5826	5830
5833	5839	5840	5841	5847	5848
5849	5852	5854	5856	5858	5859
5861	5866	5876	5877	5879	5880
5882	5883	5886	5888	5889	5891
5892	5893	5894	5898	5899	5905
5910	5913	5915	5922	5924	
5936	5943	5944	5947	5948	
5949	5951	5952	5953	5962	5964
5971	5976	5979	5980	5983	5988
5994	5995	5996	5997	5999	
6002	6005	6006	6012	6017	
6018	6019	6020	6021	6022	
6027	6030	6031	6032	6035	6039
6040	6043	6044	6046	6047	
6049	6054	6057	6058	6062	
6067	6073	6077	6080	6081	
6083	6087	6088	6093	6094	
6099	6103	6107	6113	6117	6118
6120	6121	6125	6130	6133	
6135	6137	6138	6140	6143	
6144	6159	6161	6162	6163	6164
6166	6167	6168	6172	6173	6181
6182	6184	6192	6193	6195	6197

6198 6199 6205 6206 6207 6214
6215 6220 6230 6232 6237
6241 6243 6247 6250 6257 6259
6260 6263 6272 6274 6275 6276
6280 6281 6282 6283 6284 6287
6292 6293 6296 6300 6305 6307
6308 6311 6312 6313 6319
6322 6323 6327 6328 6330
6332 6334 6342 6343 6344 6347
6348 6349 6353 6356 6357 6362
6366 6372 6377 6378 6384 6388
6390 6393 6397 6399 6402 6403
6404 6406 6410 6412 6419 6422
6428 6432 6437 6438 6440 6442
6445 6449 6454 6456 6457 6459
6460 6462 6463 6472 6473 6477
6478 6479 6480 6481 6483 6486
6487 6494 6495 6496 6497 6502
6507 6511 6513 6521 6522
6523 6525 6529 6532 6535
6543 6544 6548 6551 6552
6553 6554 6557 6560 6561
6564 6565 6566 6568 6574 6577
6578 6579 6581 6582 6583
6585 6589 6591 6592 6594
6600 6601 6608 6611 6612 6613
6614 6615 6617 6618 6624
6625 6627 6630 6637 6638
6640 6647 6650 6657 6658 6661
6664 6668 6672 6673 6677 6679
6683 6685 6687 6689 6690
6691 6693 6695 6697 6706
6708 6709 6710 6711 6723

6732 6742 6743 6747 6751
6758 6761 6768 6769 6771
6773 6774 6775 6776 6777
6779 6783 6786 6787 6789
6790 6791 6800 6802 6803
6807 6808 6809 6813 6815
6818 6823 6824 6825 6828 6832
6833 6834 6839 6846 6849
6853 6856 6861 6862 6868 6869
6875 6878 6879 6880 6882 6883
6887 6891 6893 6897 6900
6908 6911 6912 6914 6917
6918 6920 6923 6927 6928
6932 6935 6937 6938 6942
6943 6944 6945 6955 6957
6963 6964 6965 6966 6983
6984 6986 6987 6988 6992
7000 7006 7011 7018 7032
7034 7035 7036 7042 7046
7047 7053 7055 7060 7061
7062 7063 7067 7075 7079
7082 7085 7086 7087 7091
7093 7096 7099 7109 7112
[reached getOption("max.print") -- omitted 4 rows]

Sum of squares = 0.08   Mean square = 0   n = 2371

fold 3
Observations in test set: 2371
1 4 5 6 7 8 20 22
24 26 31 32 34 35 39 42
48 49 65 66 67 70 72 74 79
80 82 89 90 91 95 97 98 103

104 105 107 109 110 112 115 116
117 118 119 120 122 124 127 133 135
137 139 140 142 144 148 153 154
160 162 165 174 177 178 186 194 197
199 200 202 203 211 213 215 222
223 228 229 230 235 240 243 251
255 261 263 264 265 270 278 283
285 291 295 297 302 313 314 316
321 328 330 332 333 338 343 346
351 352 353 354 359 361 362 364
368 369 370 376 383 385 386 387
388 394 395 399 404 408 410 411 416
417 418 419 420 421 423 427 428
436 442 457 458 467 469 471 473
475 476 484 485 486 488 490 493
498 505 506 508 512 518 519 522
524 526 528 531 532 542 543 546
548 550 551 553 555 556 561 565
568 569 571 573 575 582 584 587
589 590 593 597 598 602 604 606
609 614 619 623 625 630 631 633
639 641 643 644 647 650 652 656
658 659 661 663 664 665 669
673 674 675 678 680 681 682 685 694
695 697 700 701 706 710 711 712
713 715 722 724 730 731 737 738
740 746 747 755 756 760 761 762
763 764 774 776 777 782 786 791
792 800 808 809 811 813 816 820
822 829 831 834 836 840 841 845
846 849 853 854 856 857 858
859 860 861 864 868 871 874 876
879 883 885 889 893 896 898 899

904 907 909 911 912 922 924 927
928 930 935 936 938 940 944 947
949 951 952 953 954 955 956 958
961 963 964 966 967 976 977
978 983 985 986 991 993 994 999
1000 1002 1010 1012 1015 1018 1019
1023 1027 1028 1031 1035 1038 1044 1050 1051
1055 1056 1060 1062 1064 1066 1073
1076 1080 1082 1084 1085 1091 1096
1097 1098 1099 1101 1103 1108 1109 1111
1114 1115 1117 1118 1119 1127 1128
1131 1134 1142 1144 1146 1150 1154 1156
1157 1158 1163 1166 1168 1176 1180
1181 1184 1185 1186 1191 1193 1194
1195 1197 1198 1201 1202 1203 1204 1205
1206 1212 1213 1215 1224 1226 1228
1237 1239 1243 1244 1249 1251 1252
1259 1265 1267 1277 1278 1282 1287
1303 1306 1307 1309 1311 1313 1318 1319
1321 1322 1323 1325 1327 1335
1337 1344 1345 1349 1358 1377 1378
1384 1389 1394 1398 1399 1405 1407 1408
1409 1415 1417 1418 1422 1424 1426
1430 1433 1437 1447 1450 1455 1456
1463 1464 1465 1470 1471 1481 1483
1485 1489 1495 1497 1498 1502 1507
1510 1516 1518 1519 1521 1523 1527
1529 1535 1539 1541 1545 1547 1550
1553 1554 1555 1556 1558 1561 1562
1563 1566 1570 1579 1580 1581 1586
1588 1590 1592 1593 1596 1601 1602
1609 1612 1616 1617 1621 1622 1627
1629 1635 1641 1642 1644 1645 1646

1653 1654 1655 1656 1665 1666 1672
1673 1675 1677 1680 1683 1685 1686
1693 1694 1696 1707 1708 1715 1717
1719 1721 1722 1725 1730 1732 1734
1735 1743 1746 1749 1752 1753 1755
1756 1757 1758 1763 1766 1777 1778 1784
1785 1786 1787 1793 1795 1802 1804
1806 1807 1810 1812 1816 1819 1826
1833 1835 1841 1843 1847 1852 1857
1861 1864 1865 1867 1868 1869
1870 1871 1872 1885 1886 1887 1888
1899 1900 1902 1903 1904 1906 1907
1910 1916 1917 1918 1923 1926 1930
1940 1941 1944 1945 1946 1947 1962
1963 1967 1969 1973 1975 1976 1977
1978 1980 1982 1983 1984 1988 1990
1998 2004 2011 2013 2014 2016 2020
2022 2024 2026 2027 2029 2030 2033
2034 2038 2042 2043 2044 2045 2048
2050 2052 2053 2054 2055 2059 2062
2066 2069 2072 2075 2077 2078 2079
2083 2086 2090 2093 2094 2098 2099
2103 2107 2113 2115 2122 2125 2128
2135 2136 2137 2138 2139 2142 2145
2148 2149 2151 2155 2162 2165 2173
2175 2177 2178 2181 2186 2192 2193
2195 2196 2199 2202 2210 2213 2215
2217 2218 2223 2224 2227 2231 2232
2233 2235 2240 2241 2244 2245 2246
2254 2259 2262 2263 2269 2277 2282
2287 2288 2294 2296 2297 2298 2299
2304 2311 2313 2317 2318 2322 2323
2325 2327 2339 2340 2343 2347 2353

2354 2360 2361 2362 2367 2379 2382
2383 2384 2387 2389 2390 2398 2399
2403 2407 2410 2412 2416 2425 2427
2436 2440 2442 2443 2445 2452 2453
2457 2458 2460 2466 2468 2473 2474
2476 2477 2486 2488 2490 2497 2498
2504 2506 2508 2509 2513 2516 2522
2523 2525 2530 2533 2534 2537 2538
2539 2540 2554 2556 2561 2564
2565 2566 2569 2571 2572 2574 2575
2576 2579 2580 2582 2583 2586 2587
2590 2591 2593 2594 2597 2607 2608
2612 2613 2614 2615 2620 2621 2624
2626 2628 2634 2635 2639 2643 2646
2647 2651 2652 2657 2661 2664 2666
2668 2669 2674 2677 2681 2686 2687
2691 2692 2693 2697 2698 2704 2705
2706 2707 2710 2716 2717 2718 2720
2727 2731 2733 2734 2735 2737 2738
2742 2743 2744 2745 2748 2750 2751
2752 2763 2764 2767 2768 2773
2776 2781 2782 2785 2786 2791
2799 2800 2804 2807 2812 2816 2821
2822 2824 2825 2830 2831 2834 2835
2838 2839 2841 2843 2845 2847
2848 2857 2859 2865 2866 2867 2872
2873 2875 2877 2887 2894 2896 2898
2899 2902 2904 2905 2906 2907 2908
2911 2912 2913 2914 2917 2920 2921
2927 2929 2930 2932 2935 2936 2939
2940 2942 2943 2945 2948 2949 2950
2951 2955 2961 2962 2964 2980 2982
2986 2987 2988 2993 2995 2997 2998

2999 3000 3004 3006 3007 3010 3011
3012 3013 3014 3017 3018 3021
3025 3028 3034 3037 3038 3040 3041
3042 3043 3044 3047 3051 3056 3057
3059 3060 3061 3063 3064 3066 3068
3072 3074 3077 3079 3083 3087 3091
3095 3096 3100 3105 3108 3112 3113
3114 3117 3121 3122 3131 3132 3133
3137 3141 3142 3149 3150 3152 3153
3154 3156 3159 3160 3163 3165
3169 3172 3173 3174 3175 3178 3181
3182 3185 3190 3192 3193 3194 3195
3196 3201 3205 3207 3212 3221 3225
3227 3228 3234 3238 3241 3244
3251 3253 3256 3258 3268 3270
3273 3274 3277 3278 3279 3280 3288
3297 3300 3309 3316 3317 3326 3328
3331 3335 3348 3353 3355 3357 3362
3367 3369 3370 3375 3377 3382
3387 3390 3392 3394 3396 3399 3400
3401 3411 3413 3417 3419 3420
3422 3424 3425 3426 3427 3429 3432
3433 3435 3438 3440 3442 3448
3449 3450 3458 3459 3461 3462 3464
3470 3472 3473 3474 3477 3478
3480 3486 3487 3488 3489 3490
3492 3500 3502 3504 3507 3508 3515
3518 3521 3528 3533 3534 3537
3539 3540 3541 3544 3548 3550
3553 3559 3567 3570 3574 3577
3583 3591 3592 3593 3596 3598
3602 3607 3608 3612 3618 3621 3623
3624 3625 3632 3636 3644 3650

3655 3657 3662 3663 3664 3671 3673
3675 3678 3679 3681 3682 3687 3688
3690 3691 3692 3698 3699 3704
3714 3719 3720 3721 3722 3723
3724 3726 3727 3732 3735 3745 3751
3752 3767 3768 3776 3781 3784
3785 3794 3795 3801 3803 3804
3806 3810 3812 3820 3822 3826
3832 3836 3837 3842 3845 3850
3853 3854 3856 3861 3863 3867
3871 3873 3875 3876 3880 3881
3882 3886 3887 3888 3893 3894
3895 3899 3901 3906 3907 3914
3920 3923 3924 3926 3927 3931 3933
3942 3944 3946 3947 3951 3959
3960 3962 3963 3967 3976 3977
3981 3985 3988 3989 3993 3996 3999
4002 4004 4008 4013 4014 4017
4018 4024 4025 4029 4034 4041
4042 4052 4053 4054 4061 4064
4065 4068 4074 4080 4082 4085 4090
4091 4092 4094 4097 4099 4100
4101 4103 4105 4107 4108 4113
4115 4118 4119 4121 4126 4130 4136
4137 4139 4141 4142 4144 4145
4148 4157 4162 4163 4167 4168
4170 4175 4176 4179 4180 4184
4187 4191 4197 4201 4208 4210 4211
4213 4214 4216 4220 4228 4230
4231 4233 4239 4244 4247 4248
4250 4252 4253 4254 4259 4263
4264 4265 4266 4272 4275 4276
4277 4279 4286 4290 4293 4296

4300	4303	4306	4312	4318	4321	4324
4329	4331	4332	4334	4335	4339	
4341	4342	4344	4347	4350	4351	4355
4356	4364	4366	4368	4375	4386	
4387	4391	4394	4395	4397	4400	4403
4405	4408	4410	4412	4413	4415	
4417	4418	4420	4427	4429	4430	
4432	4434	4435	4436	4438	4439	
4445	4446	4449	4456	4463	4464	
4466	4467	4470	4472	4473	4477	
4481	4491	4492	4494	4496	4498	
4502	4503	4505	4506	4508	4511	
4513	4516	4520	4521	4525	4528	
4529	4530	4537	4538	4542	4545	
4548	4555	4558	4568	4576	4577	
4579	4580	4583	4585	4586	4587	
4592	4597	4598	4599	4603	4605	
4607	4609	4611	4614	4619	4623	
4624	4628	4631	4632	4633	4634	4636
4640	4643	4644	4645	4647	4648	
4651	4652	4655	4657	4658	4665	
4667	4670	4672	4675	4677	4678	
4680	4681	4687	4689	4694	4701	
4705	4706	4707	4709	4710	4712	
4719	4723	4729	4734	4735	4739	
4743	4746	4747	4750	4752	4753	
4755	4762	4763	4766	4770	4773	
4774	4779	4786	4789	4794	4795	
4796	4797	4799	4808	4809	4811	
4812	4814	4815	4818	4819	4820	
4821	4824	4826	4829	4832	4835	4837
4838	4847	4853	4858	4860	4862	
4863	4866	4876	4886	4888	4891	

4897 4898 4902 4905 4909 4910 4913
4915 4917 4920 4922 4925 4930
4931 4932 4939 4942 4943 4947
4957 4960 4964 4966 4968 4971
4973 4974 4977 4978 4979 4980
4985 4986 4988 4991 4993 4994
4997 4999 5001 5002 5009 5013
5019 5020 5021 5023 5029 5032
5034 5036 5037 5038 5039 5045
5046 5049 5056 5061 5063 5066
5073 5075 5078 5084 5090 5091
5103 5113 5116 5121 5123 5130
5131 5133 5137 5140 5143 5144
5145 5150 5152 5155 5157
5164 5170 5172 5175 5179
5183 5185 5188 5190 5193 5194
5196 5202 5203 5205 5207 5209
5210 5212 5213 5218 5221 5227
5229 5233 5234 5239 5243 5244
5246 5249 5252 5254 5255 5257
5258 5259 5260 5262 5267 5270
5271 5273 5274 5278 5282 5283
5284 5289 5291 5293 5296 5298
5305 5308 5314 5317 5322 5325
5326 5327 5329 5332 5339 5340
5345 5347 5355 5356 5360 5361
5362 5367 5368 5370 5372 5375
5376 5377 5381 5385 5388 5389
5392 5393 5394 5395 5403 5406
5407 5413 5416 5423 5425 5427
5428 5431 5434 5436 5437 5439
5447 5451 5452 5453 5458 5462
5463 5464 5466 5472 5473 5476

5478 5479 5483 5487 5488 5493
5496 5497 5500 5502 5503 5505
5507 5511 5515 5517 5521
5523 5527 5529 5533 5537 5538
5540 5541 5548 5555 5557 5565
5566 5568 5571 5573 5577 5578
5579 5584 5585 5586 5593 5595
5603 5605 5610 5613 5623 5626
5635 5640 5647 5650 5652 5653
5656 5657 5659 5660 5662 5664
5665 5669 5670 5673 5675 5681
5683 5688 5690 5692 5693 5699
5707 5708 5709 5714 5715 5720
5722 5724 5726 5730 5738 5739
5740 5741 5745 5749 5750 5754
5756 5758 5763 5764 5768 5773
5774 5777 5783 5786 5788 5789
5791 5794 5796 5798 5799 5802
5803 5805 5812 5813 5817 5827
5829 5831 5836 5842 5844 5845
5846 5850 5853 5860 5862 5868
5870 5871 5873 5875 5881 5887
5890 5901 5902 5903 5906 5907
5909 5911 5914 5918 5923 5926
5927 5928 5929 5930 5931 5941
5945 5946 5954 5955 5956 5957
5960 5963 5965 5968 5969 5972
5975 5977 5985 5986 5991 5992
5998 6000 6003 6007 6008 6009
6010 6013 6016 6025 6029 6038
6041 6042 6045 6048 6050
6051 6052 6053 6056 6063 6064
6071 6076 6078 6082 6086 6090

6095 6100 6104 6106 6108 6109
6111 6112 6114 6123 6124 6126
6131 6134 6139 6142 6146 6147
6148 6149 6150 6151 6152 6153
6154 6156 6157 6160 6165 6169
6170 6171 6174 6175 6180 6187
6189 6191 6194 6202 6203
6211 6213 6217 6219 6221 6231
6235 6236 6238 6245 6248 6249
6251 6253 6254 6255 6264 6265
6268 6269 6270 6271 6278 6279
6285 6288 6290 6302 6303 6304
6306 6315 6320 6325 6326 6331
6335 6337 6339 6340 6341 6345
6346 6350 6352 6355 6360 6364
6365 6367 6368 6369 6375 6376
6380 6381 6382 6383 6385 6387
6391 6392 6394 6395 6396 6400
6405 6409 6411 6413 6414 6416
6417 6421 6423 6424 6425 6426
6429 6430 6431 6439 6441
6443 6447 6448 6453 6458 6466
6467 6468 6471 6474 6476 6482
6484 6490 6492 6493 6498 6503
6504 6505 6510 6512 6514 6517
6518 6519 6524 6527 6528 6531
6534 6536 6537 6538 6540 6546
6555 6558 6562 6563 6567
6569 6575 6576 6590 6595 6596
6597 6598 6602 6603 6607
6609 6616 6622 6626 6632 6633
6636 6641 6643 6644 6645 6649
6651 6654 6655 6656 6660 6662

6666 6671 6678 6680 6682 6692
6694 6698 6713 6715 6718
6719 6720 6722 6726 6728 6729
6734 6735 6736 6738 6740 6741
6744 6745 6746 6748 6749 6750
6752 6753 6755 6759 6765
6766 6770 6780 6782 6784 6785
6794 6796 6798 6801 6804 6806
6811 6814 6819 6820 6821 6822
6826 6827 6829 6835 6836 6838
6841 6843 6844 6845 6848
6850 6851 6854 6855 6858
6860 6863 6866 6867 6881 6886
6888 6889 6895 6898 6899 6902
6903 6904 6906 6915 6916 6922
6926 6933 6934 6936 6939 6940
6941 6946 6947 6948 6949
6951 6952 6956 6959 6960 6961
6962 6968 6969 6971 6972
6976 6977 6978 6979 6980 6991
6998 7001 7002 7004 7005
7007 7008 7012 7014 7016 7017
7019 7020 7024 7025 7026
7029 7033 7039 7040 7043
7045 7050 7052 7056 7057
7064 7066 7068 7069 7072
7073 7077 7078 7080 7084
7089 7092 7095 7098 7101 7104
7106 7108 7111

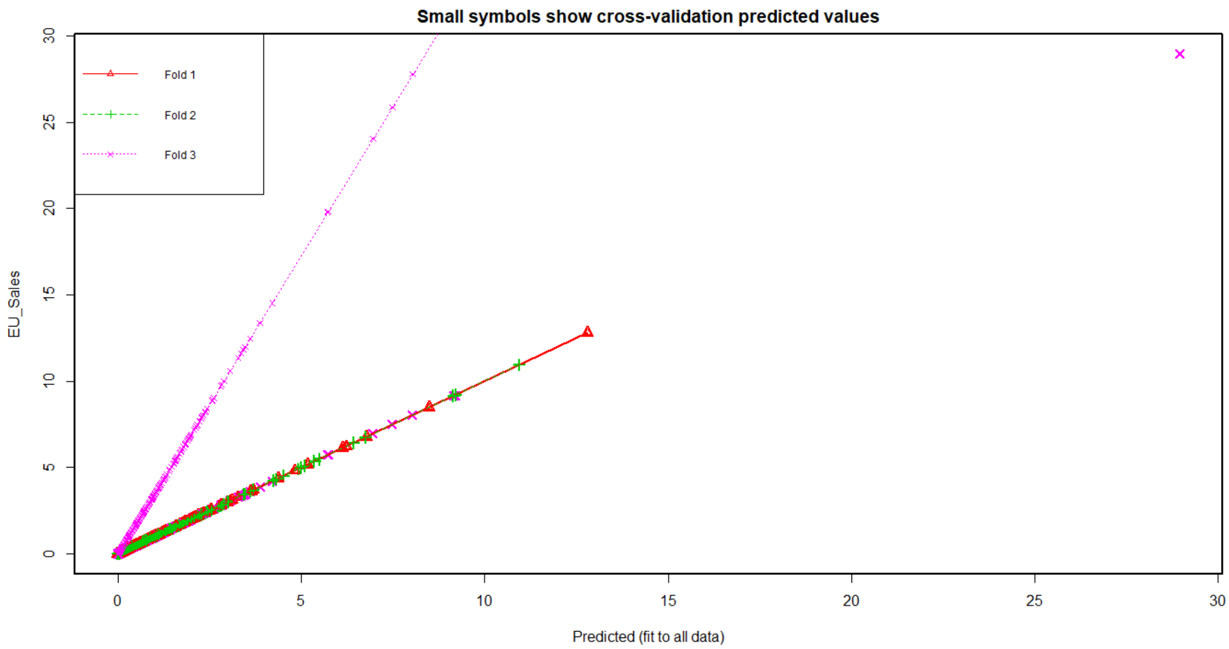
[reached getOption("max.print") -- omitted 4 rows]

Sum of squares = 10362 Mean square = 4.37 n = 2371

```
Overall (Sum over all 2371 folds)
  ms
1.46
Warning messages:
1: In predict.lm(subs.lm, newdata = data[rows.out, ]) :
  prediction from a rank-deficient fit may be misleading
2: In predict.lm(subs.lm, newdata = data[rows.out, ]) :
  prediction from a rank-deficient fit may be misleading
3: In predict.lm(subs.lm, newdata = data[rows.out, ]) :
  prediction from a rank-deficient fit may be misleading
4: In cv.lm(data = vg, form.lm = formula(EU_Sales ~ .), plotit = "Observed",  :


 As there is >1 explanatory variable, cross-validation
 predicted values for a fold are not a linear function
 of corresponding overall predicted values.  Lines that
 are shown for the different folds are approximate

>
>
> #results for first fold
> # Sum of squares = 0.09    Mean square = 0    n = 2370
>
> #results for second fold
> #Sum of squares = 0.08    Mean square = 0    n = 2371
>
>
> #results for third fold
> # Sum of squares = 10362    Mean square = 4.37    n = 2371
>
>
> # Avg mean squre error
> # 1.46
```



#-----Stepwise Regression-----

install.packages("MASS")
library(MASS)

#AIC - stepwise Algorithum

#=====Backword Elimination ========

> vg <- Videogames[,-c(1,2,3,4,5,15)]
>
> model_full<- lm(EU_Sales~., data= vg)
>
> step<- stepAIC(model_full, direction = "backward")
Start: AIC=-72908
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales +

```
    Critic_Score + Critic_Count + User_Score + User_Count

              Df Sum of Sq RSS    AIC
- Critic_Score  1         0   0 -72910
- User_Score    1         0   0 -72910
- Critic_Count  1         0   0 -72910
- User_Count    1         0   0 -72909
<none>                        0 -72908
- Other_Sales   1       141 141 -27879
- JP_Sales      1       217 217 -24791
- NA_Sales      1       349 349 -21418
- Global_Sales  1       818 818 -15361

Step:  AIC=-72910
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales + 
    Critic_Count + User_Score + User_Count

              Df Sum of Sq RSS    AIC
- User_Score    1         0   0 -72912
- Critic_Count  1         0   0 -72912
- User_Count    1         0   0 -72911
<none>                        0 -72910
- Other_Sales   1       141 141 -27881
- JP_Sales      1       217 218 -24788
- NA_Sales      1       349 349 -21419
- Global_Sales  1       819 819 -15360

Step:  AIC=-72912
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales + 
    Critic_Count + User_Count

              Df Sum of Sq RSS    AIC
- Critic_Count  1         0   0 -72914
```

```
- User_Count    1         0   0 -72913
<none>                       0 -72912
- Other_Sales   1       141 141 -27881
- JP_Sales      1       220 220 -24713
- NA_Sales      1       350 350 -21398
- Global_Sales  1       821 821 -15340

Step:  AIC=-72914
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales +
    User_Count

               Df Sum of Sq RSS    AIC
- User_Count    1         0   0 -72914
<none>                       0 -72914
- Other_Sales   1       141 141 -27879
- JP_Sales      1       220 220 -24695
- NA_Sales      1       351 352 -21377
- Global_Sales  1       821 822 -15340

Step:  AIC=-72914
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales

               Df Sum of Sq RSS    AIC
<none>                       0 -72914
- Other_Sales   1       141 141 -27879
- JP_Sales      1       225 225 -24549
- NA_Sales      1       355 355 -21314
- Global_Sales  1       839 839 -15189
> step$anova   # Display results
Stepwise Model Path
Analysis of Deviance Table

Initial Model:
```

```
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales + 
    Critic_Score + Critic_Count + User_Score + User_Count

Final Model:
EU_Sales ~ NA_Sales + JP_Sales + Other_Sales + Global_Sales


            Step Df Deviance Resid. Df Resid. Dev    AIC
1                                 7103      0.250 -72908
2 - Critic_Score  1 4.27e-07      7104      0.250 -72910
3   - User_Score  1 1.47e-06      7105      0.250 -72912
4 - Critic_Count  1 1.46e-05      7106      0.250 -72914
5   - User_Count  1 5.24e-05      7107      0.251 -72914
> 
> 
> #Checking empty and full models
> model_full<- lm(EU_Sales~., data= vg)
> model_empty<- lm(EU_Sales~ 1, data=vg)
> 
> summary(model_empty)   # The empty model has only one yinterscept

Call:
lm(formula = EU_Sales ~ 1, data = vg)

Residuals:
   Min     1Q Median     3Q    Max 
-0.233 -0.213 -0.173 -0.030 28.727 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  0.23254    0.00806    28.8   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Residual standard error: 0.68 on 7111 degrees of freedom

```
>
>
>
> # Now we will do forward
>
> step<- stepAIC(model_empty, direction = "forward", scope = list(upper= model_full,
lower= model_empty))
Start:  AIC=-5484
EU_Sales ~ 1

               Df Sum of Sq  RSS    AIC
+ Global_Sales  1      2896  392 -20604
+ NA_Sales      1      2310  978 -14107
+ Other_Sales   1      1696 1592 -10641
+ JP_Sales      1       884 2404  -7709
+ User_Count    1       266 3022  -6082
+ Critic_Count  1       236 3053  -6011
+ Critic_Score  1       148 3141  -5809
+ User_Score    1        10 3278  -5504
<none>                      3288  -5484

Step:  AIC=-20604
EU_Sales ~ Global_Sales

               Df Sum of Sq RSS    AIC
+ NA_Sales      1     125.3 267 -23339
+ JP_Sales      1      16.3 376 -20904
+ Other_Sales   1      13.2 379 -20846
+ User_Count    1       4.5 388 -20684
+ User_Score    1       2.5 390 -20647
```

```
+ Critic_Score  1       0.4 392 -20609
+ Critic_Count  1       0.2 392 -20605
<none>                      392 -20604

Step:  AIC=-23339
EU_Sales ~ Global_Sales + NA_Sales

               Df Sum of Sq RSS    AIC
+ JP_Sales      1     126.0 141 -27879
+ Other_Sales   1      41.8 225 -24549
+ User_Count    1       3.4 264 -23429
+ User_Score    1       2.3 265 -23399
+ Critic_Score  1       0.1 267 -23341
<none>                      267 -23339
+ Critic_Count  1       0.0 267 -23339

Step:  AIC=-27879
EU_Sales ~ Global_Sales + NA_Sales + JP_Sales

               Df Sum of Sq   RSS    AIC
+ Other_Sales   1     140.7   0.3 -72914
+ User_Score    1       0.1 140.9 -27881
+ Critic_Count  1       0.0 140.9 -27879
<none>                      140.9 -27879
+ User_Count    1       0.0 140.9 -27879
+ Critic_Score  1       0.0 140.9 -27879

Step:  AIC=-72914
EU_Sales ~ Global_Sales + NA_Sales + JP_Sales + Other_Sales

               Df Sum of Sq   RSS    AIC
<none>                      0.250 -72914
+ User_Count    1  5.24e-05 0.250 -72914
```

```
+ Critic_Count  1  3.47e-05 0.250 -72913
+ Critic_Score  1  8.20e-06 0.250 -72913
+ User_Score    1  3.90e-06 0.250 -72913
> summary(step)

Call:
lm(formula = EU_Sales ~ Global_Sales + NA_Sales + JP_Sales +
    Other_Sales, data = vg)

Residuals:
      Min        1Q    Median        3Q       Max
-0.020098  0.000147  0.000228  0.000244  0.020221

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  -2.43e-04   7.61e-05    -3.2   0.0014 **
Global_Sales  1.00e+00   2.05e-04  4879.1   <2e-16 ***
NA_Sales     -1.00e+00   3.15e-04 -3171.2   <2e-16 ***
JP_Sales     -1.00e+00   3.96e-04 -2525.6   <2e-16 ***
Other_Sales  -1.00e+00   5.00e-04 -1997.8   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.00594 on 7107 degrees of freedom
Multiple R-squared:      1,      Adjusted R-squared:      1
F-statistic: 2.33e+07 on 4 and 7107 DF,  p-value: <2e-16
```