Data Science Project

❖ Problem statement

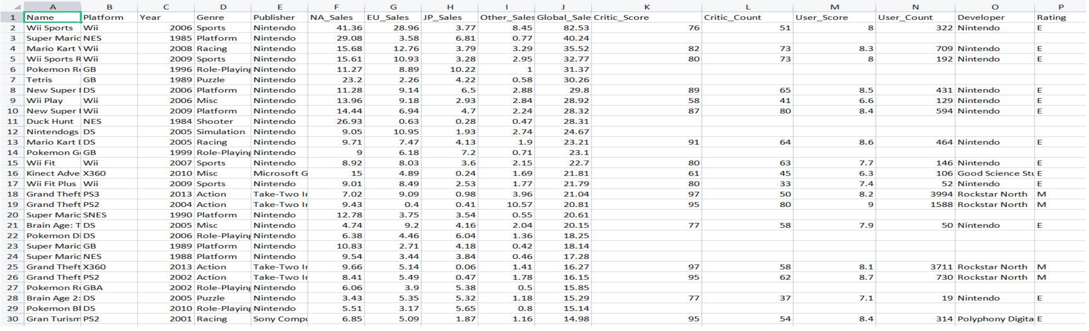
In today's world due to this pandemic, there is a huge rise in the gaming sector, But when and where to launch a particular game so that it gets the maximum sales is the challenging part
So,
To Predict which game to release at a certain platform at a particular Genre so that there is an increase in sales.

❖ Objectives

1)To find a trend where the sales can be maximized.
2)To Compare the sales in between certain countries
3)Comparing Platform and Genre with Critic Score/Count and User Score/Count
4)We will be using Multiple Regression.

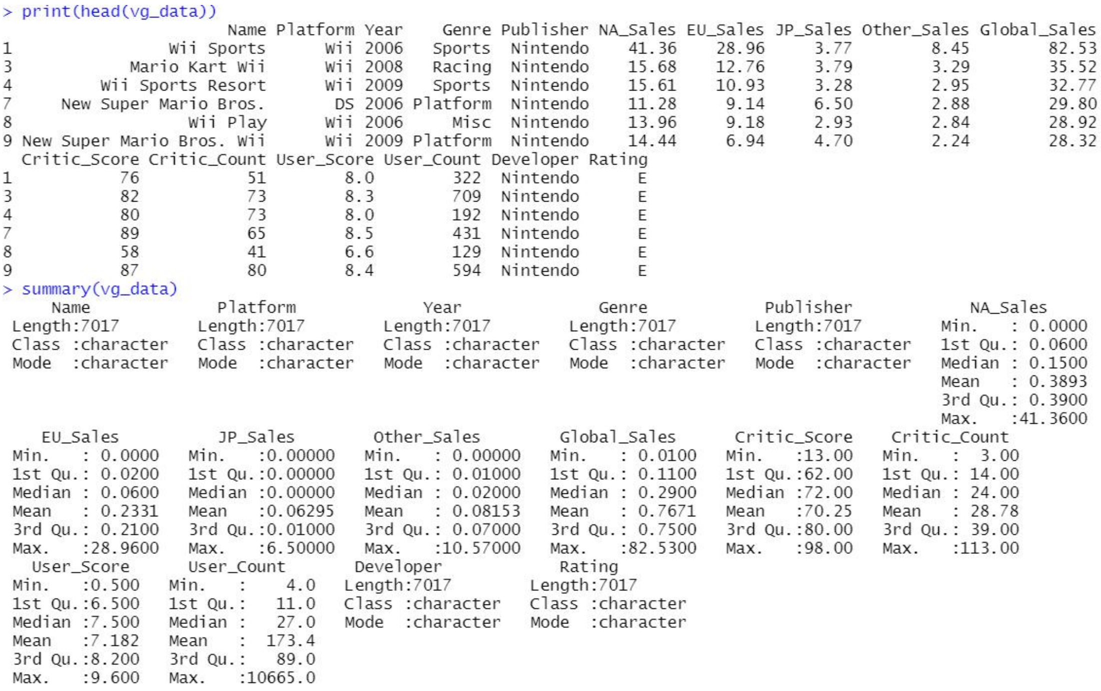
❖ Data-Set Description

This data-set contains a list of video games with sales greater than 100,000 copies. It was generated by a scrape of vgchartz.com.
Fields included
Rank - Ranking of overall sales
Name - The games name
Platform - Platform of the games release (i.e. PC,PS4, etc.)
Year - Year of the game's release
Genre - Genre of the game
Publisher - Publisher of the game
NA_Sales - Sales in North America (in millions)
EU_Sales - Sales in Europe (in millions)
JP_Sales - Sales in Japan (in millions)
Other_Sales - Sales in the rest of the world (in millions)
Global_Sales - Total worldwide sales.
Critic Score - Aggregate score compiled by Metacritic staff Critic Count - The number of critics used in coming up with the Critic Score User Score - Score by Metacritic's subscribers
User Count - Number of users who gave the userscore
Developer - Party responsible for creating the game

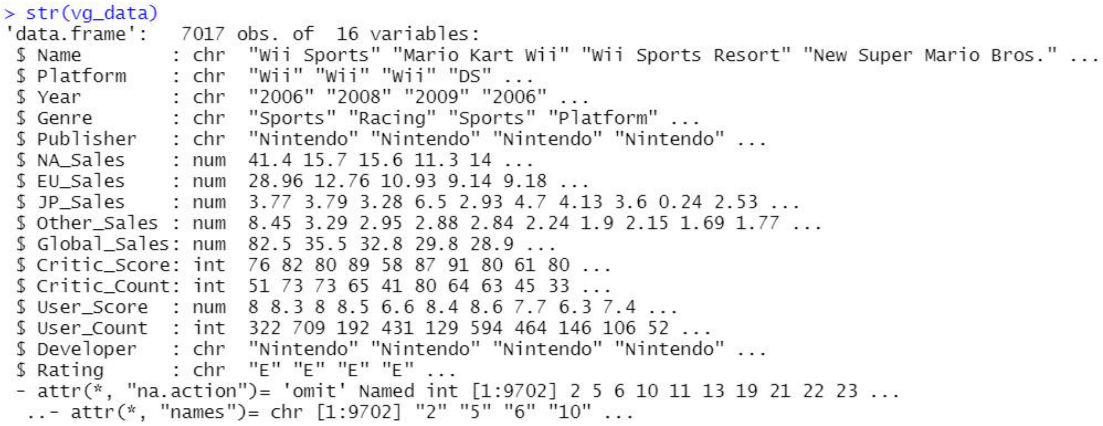


	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sale	Critic_Score	Critic_Count	User_Score	User_Count	Developer	Rating
2	Wii Sports	Wii	2006	Sports	Nintendo	41.36	28.96	3.77	8.45	82.53	76	51	8	322	Nintendo	E
3	Super Maric	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24						
4	Mario Kart \	Wii	2008	Racing	Nintendo	15.68	12.76	3.79	3.29	35.52	82	73	8.3	709	Nintendo	E
5	Wii Sports R	Wii	2009	Sports	Nintendo	15.61	10.93	3.28	2.95	32.77	80	73	8	192	Nintendo	E
6	Pokemon Re	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1	31.37						
7	Tetris	GB	1989	Puzzle	Nintendo	23.2	2.26	4.22	0.58	30.26						
8	New Super I	DS	2006	Platform	Nintendo	11.28	9.14	6.5	2.88	29.8	89	65	8.5	431	Nintendo	E
9	Wii Play	Wii	2006	Misc	Nintendo	13.96	9.18	2.93	2.84	28.92	58	41	6.6	129	Nintendo	E
10	New Super I	Wii	2009	Platform	Nintendo	14.44	6.94	4.7	2.24	28.32	87	80	8.4	594	Nintendo	E
11	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31						
12	Nintendogs	DS	2005	Simulation	Nintendo	9.05	10.95	1.93	2.74	24.67						
13	Mario Kart I	DS	2005	Racing	Nintendo	9.71	7.47	4.13	1.9	23.21	91	64	8.6	464	Nintendo	E
14	Pokemon G	GB	1999	Role-Playing	Nintendo	9	6.18	7.2	0.71	23.1						
15	Wii Fit	Wii	2007	Sports	Nintendo	8.92	8.03	3.6	2.15	22.7	80	63	7.7	146	Nintendo	E
16	Kinect Adve	X360	2010	Misc	Microsoft G	15	4.89	0.24	1.69	21.81	61	45	6.3	106	Good Science Stu	E
17	Wii Fit Plus	Wii	2009	Sports	Nintendo	9.01	8.49	2.53	1.77	21.79	80	33	7.4	52	Nintendo	E
18	Grand Theft	PS3	2013	Action	Take-Two Ir	7.02	9.09	0.98	3.96	21.04	97	50	8.2	3994	Rockstar North	M
19	Grand Theft	PS2	2004	Action	Take-Two Ir	9.43	0.4	0.41	10.57	20.81	95	80	9	1588	Rockstar North	M
20	Super Maric	SNES	1990	Platform	Nintendo	12.78	3.75	3.54	0.55	20.61						
21	Brain Age: T	DS	2005	Misc	Nintendo	4.74	9.2	4.16	2.04	20.15	77	58	7.9	50	Nintendo	E
22	Pokemon Di	DS	2006	Role-Playing	Nintendo	6.38	4.46	6.04	1.36	18.25						
23	Super Maric	GB	1989	Platform	Nintendo	10.83	2.71	4.18	0.42	18.14						
24	Super Maric	NES	1988	Platform	Nintendo	9.54	3.44	3.84	0.46	17.28						
25	Grand Theft	X360	2013	Action	Take-Two Ir	9.66	5.14	0.06	1.41	16.27	97	58	8.1	3711	Rockstar North	M
26	Grand Theft	PS2	2002	Action	Take-Two Ir	8.41	5.49	0.47	1.78	16.15	95	62	8.7	730	Rockstar North	M
27	Pokemon Ru	GBA	2002	Role-Playing	Nintendo	6.06	3.9	5.38	0.5	15.85						
28	Brain Age 2:	DS	2005	Puzzle	Nintendo	3.43	5.35	5.32	1.18	15.29	77	37	7.1	19	Nintendo	E
29	Pokemon Bl	DS	2010	Role-Playing	Nintendo	5.51	3.17	5.65	0.8	15.14						
30	Gran Turism	PS2	2001	Racing	Sony Compu	6.85	5.09	1.87	1.16	14.98	95	54	8.4	314	Polyphony Digita	E

❖ Analysis of data-set(R/Python)



```
> print(head(vg_data))
                      Name Platform Year    Genre Publisher NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales
1               Wii Sports      Wii 2006   Sports  Nintendo    41.36    28.96     3.77        8.45        82.53
3           Mario Kart Wii      Wii 2008   Racing  Nintendo    15.68    12.76     3.79        3.29        35.52
4        Wii Sports Resort      Wii 2009   Sports  Nintendo    15.61    10.93     3.28        2.95        32.77
7    New Super Mario Bros.       DS 2006 Platform  Nintendo    11.28     9.14     6.50        2.88        29.80
8                 Wii Play      Wii 2006     Misc  Nintendo    13.96     9.18     2.93        2.84        28.92
9 New Super Mario Bros. Wii      Wii 2009 Platform  Nintendo    14.44     6.94     4.70        2.24        28.32
  Critic_Score Critic_Count User_Score User_Count Developer Rating
1           76           51        8.0        322  Nintendo      E
3           82           73        8.3        709  Nintendo      E
4           80           73        8.0        192  Nintendo      E
7           89           65        8.5        431  Nintendo      E
8           58           41        6.6        129  Nintendo      E
9           87           80        8.4        594  Nintendo      E
> summary(vg_data)
     Name             Platform              Year              Genre            Publisher            NA_Sales      
 Length:7017        Length:7017        Length:7017        Length:7017        Length:7017        Min.   : 0.0000  
 Class :character   Class :character   Class :character   Class :character   Class :character   1st Qu.: 0.0600  
 Mode  :character   Mode  :character   Mode  :character   Mode  :character   Mode  :character   Median : 0.1500  
                                                                                                Mean   : 0.3893  
                                                                                                3rd Qu.: 0.3900  
                                                                                                Max.   :41.3600  
    EU_Sales          JP_Sales         Other_Sales        Global_Sales      Critic_Score    Critic_Count   
 Min.   : 0.0000   Min.   :0.00000   Min.   : 0.00000   Min.   : 0.0100   Min.   :13.00   Min.   :  3.00  
 1st Qu.: 0.0200   1st Qu.:0.00000   1st Qu.: 0.01000   1st Qu.: 0.1100   1st Qu.:62.00   1st Qu.: 14.00  
 Median : 0.0600   Median :0.00000   Median : 0.02000   Median : 0.2900   Median :72.00   Median : 24.00  
 Mean   : 0.2331   Mean   :0.06295   Mean   : 0.08153   Mean   : 0.7671   Mean   :70.25   Mean   : 28.78  
 3rd Qu.: 0.2100   3rd Qu.:0.01000   3rd Qu.: 0.07000   3rd Qu.: 0.7500   3rd Qu.:80.00   3rd Qu.: 39.00  
 Max.   :28.9600   Max.   :6.50000   Max.   :10.57000   Max.   :82.5300   Max.   :98.00   Max.   :113.00  
   User_Score      User_Count       Developer            Rating         
 Min.   :0.500   Min.   :    4.0   Length:7017        Length:7017       
 1st Qu.:6.500   1st Qu.:   11.0   Class :character   Class :character  
 Median :7.500   Median :   27.0   Mode  :character   Mode  :character  
 Mean   :7.182   Mean   :  173.4                                        
 3rd Qu.:8.200   3rd Qu.:   89.0                                        
 Max.   :9.600   Max.   :10665.0                                        
```

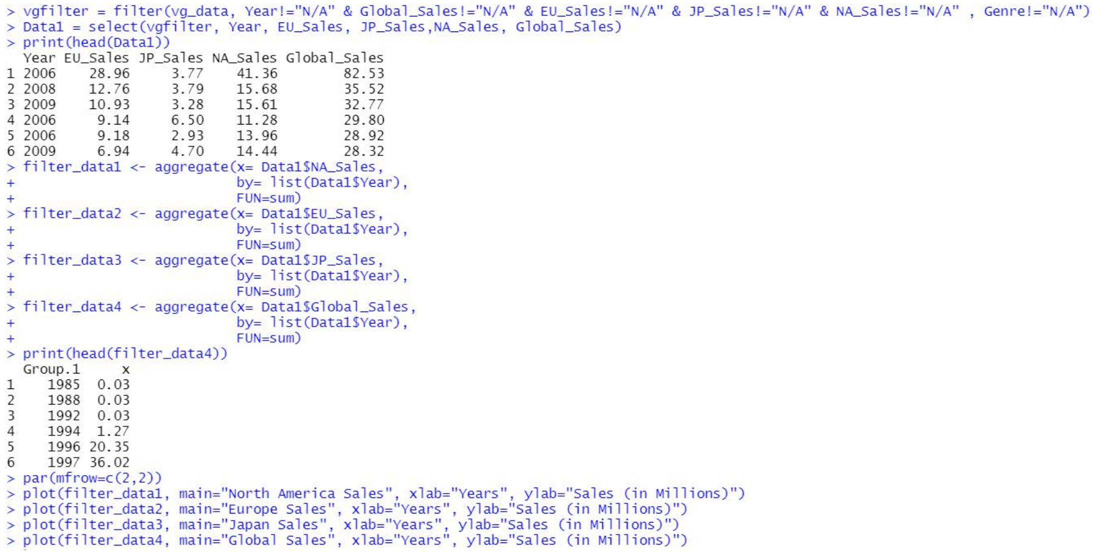


```
> str(vg_data)
'data.frame':	7017 obs. of  16 variables:
 $ Name        : chr  "Wii Sports" "Mario Kart Wii" "Wii Sports Resort" "New Super Mario Bros." ...
 $ Platform    : chr  "Wii" "Wii" "Wii" "DS" ...
 $ Year        : chr  "2006" "2008" "2009" "2006" ...
 $ Genre       : chr  "Sports" "Racing" "Sports" "Platform" ...
 $ Publisher   : chr  "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
 $ NA_Sales    : num  41.4 15.7 15.6 11.3 14 ...
 $ EU_Sales    : num  28.96 12.76 10.93 9.14 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 2.24 1.9 2.15 1.69 1.77 ...
 $ Global_Sales: num  82.5 35.5 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  322 709 192 431 129 594 464 146 106 52 ...
 $ Developer   : chr  "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
 $ Rating      : chr  "E" "E" "E" "E" ...
 - attr(*, "na.action")= 'omit' Named int [1:9702] 2 5 6 10 11 13 19 21 22 23 ...
  ..- attr(*, "names")= chr [1:9702] "2" "5" "6" "10" ...
```

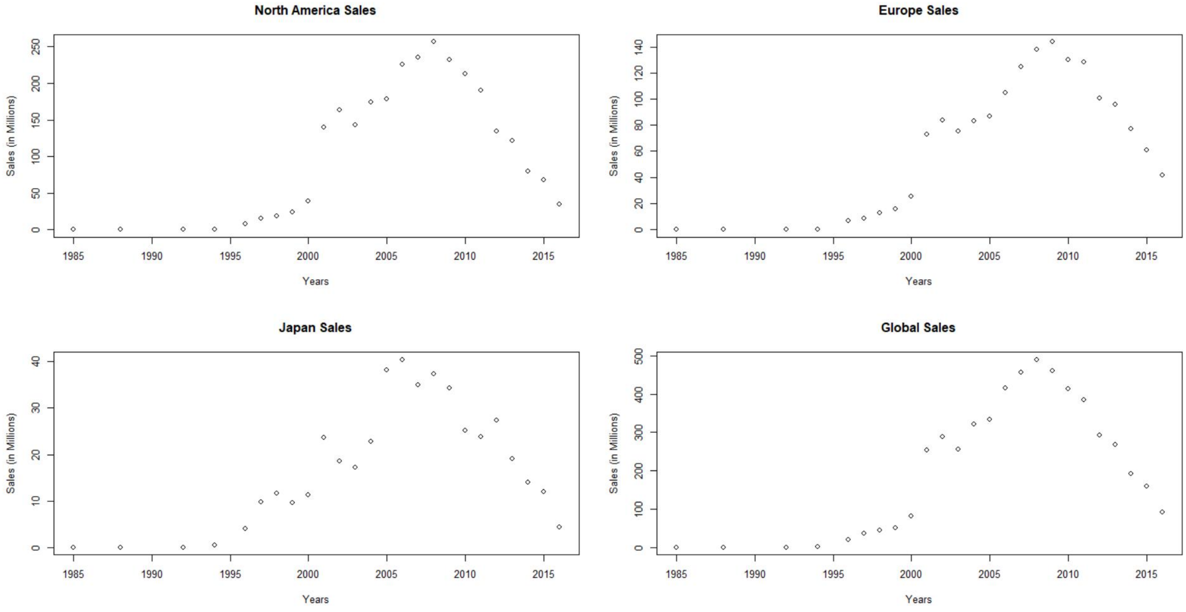
❖ Analysis graph of the data-set(R/Python)

1) Comparing Sales of various Countries and the Global Sales Graph

Removing the Null Values and filtering data-set to get various sales and year column. Then adding the duplicate Year rows to get total sales in a particular Year by Aggregate function.



```
> vgfilter = filter(vg_data, Year!="N/A" & Global_Sales!="N/A" & EU_Sales!="N/A" & JP_Sales!="N/A" & NA_Sales!="N/A" , Genre!="N/A")
> Data1 = select(vgfilter, Year, EU_Sales, JP_Sales,NA_Sales, Global_Sales)
> print(head(Data1))
  Year EU_Sales JP_Sales NA_Sales Global_Sales
1 2006    28.96     3.77    41.36        82.53
2 2008    12.76     3.79    15.68        35.52
3 2009    10.93     3.28    15.61        32.77
4 2006     9.14     6.50    11.28        29.80
5 2006     9.18     2.93    13.96        28.92
6 2009     6.94     4.70    14.44        28.32
> filter_data1 <- aggregate(x= Data1$NA_Sales,
+                          by= list(Data1$Year),
+                          FUN=sum)
> filter_data2 <- aggregate(x= Data1$EU_Sales,
+                          by= list(Data1$Year),
+                          FUN=sum)
> filter_data3 <- aggregate(x= Data1$JP_Sales,
+                          by= list(Data1$Year),
+                          FUN=sum)
> filter_data4 <- aggregate(x= Data1$Global_Sales,
+                          by= list(Data1$Year),
+                          FUN=sum)
> print(head(filter_data4))
  Group.1     x
1    1985  0.03
2    1988  0.03
3    1992  0.03
4    1994  1.27
5    1996 20.35
6    1997 36.02
> par(mfrow=c(2,2))
> plot(filter_data1, main="North America Sales", xlab="Years", ylab="Sales (in Millions)")
> plot(filter_data2, main="Europe Sales", xlab="Years", ylab="Sales (in Millions)")
> plot(filter_data3, main="Japan Sales", xlab="Years", ylab="Sales (in Millions)")
> plot(filter_data4, main="Global Sales", xlab="Years", ylab="Sales (in Millions)")
```



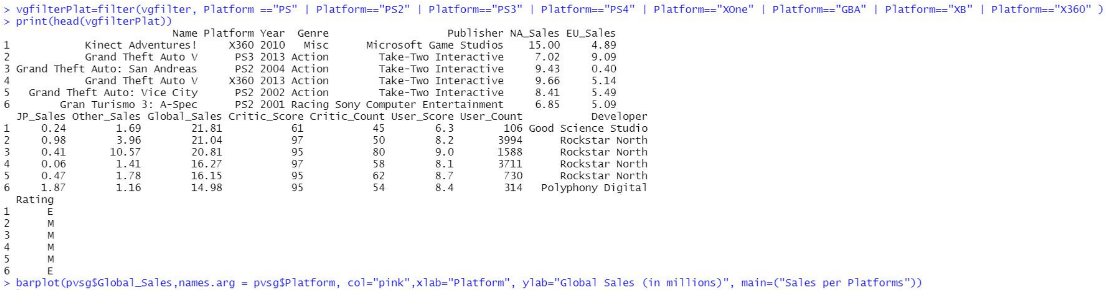
Comparing Sales

2)

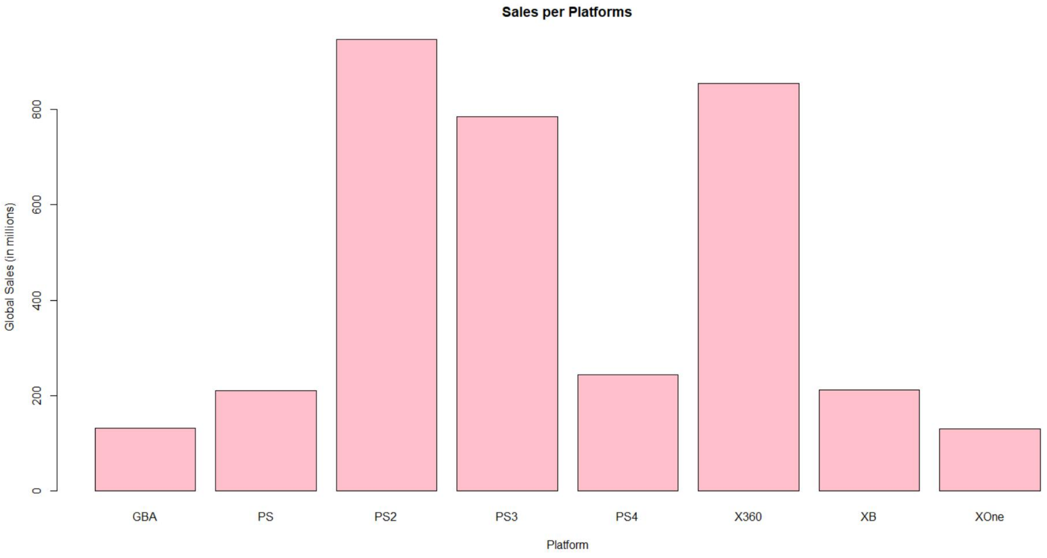
i) ***Analyzing Total Sales Per Platform***

First we filter the data to get only a certain famous Platforms, example :- we included PS and XB Series.

So we can compare which Platform got higher sales.



```
> vgfilterPlat=filter(vgfilter, Platform =="PS" | Platform=="PS2" | Platform=="PS3" | Platform=="PS4" | Platform=="XOne" | Platform=="GBA" | Platform=="XB" | Platform=="X360" )
> print(head(vgfilterPlat))
                       Name Platform Year  Genre                    Publisher NA_Sales EU_Sales
1           Kinect Adventures!     X360 2010   Misc      Microsoft Game Studios    15.00     4.89
2           Grand Theft Auto V      PS3 2013 Action        Take-Two Interactive     7.02     9.09
3 Grand Theft Auto: San Andreas      PS2 2004 Action        Take-Two Interactive     9.43     0.40
4           Grand Theft Auto V     X360 2013 Action        Take-Two Interactive     9.66     5.14
5   Grand Theft Auto: Vice City      PS2 2002 Action        Take-Two Interactive     8.41     5.49
6        Gran Turismo 3: A-Spec      PS2 2001 Racing Sony Computer Entertainment     6.85     5.09
  JP_Sales Other_Sales Global_Sales Critic_Score Critic_Count User_Score User_Count           Developer
1     0.24        1.69        21.81           61           45        6.3        106 Good Science Studio
2     0.98        3.96        21.04           97           50        8.2       3994      Rockstar North
3     0.41       10.57        20.81           95           80        9.0       1588      Rockstar North
4     0.06        1.41        16.27           97           58        8.1       3711      Rockstar North
5     0.47        1.78        16.15           95           62        8.7        730      Rockstar North
6     1.87        1.16        14.98           95           54        8.4        314   Polyphony Digital
  Rating
1      E
2      M
3      M
4      M
5      M
6      E
> barplot(pvsg$Global_Sales,names.arg = pvsg$Platform, col="pink",xlab="Platform", ylab="Global Sales (in millions)", main=("Sales per Platforms"))
```

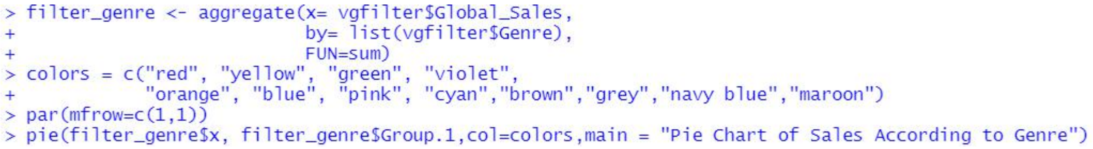


v1.01

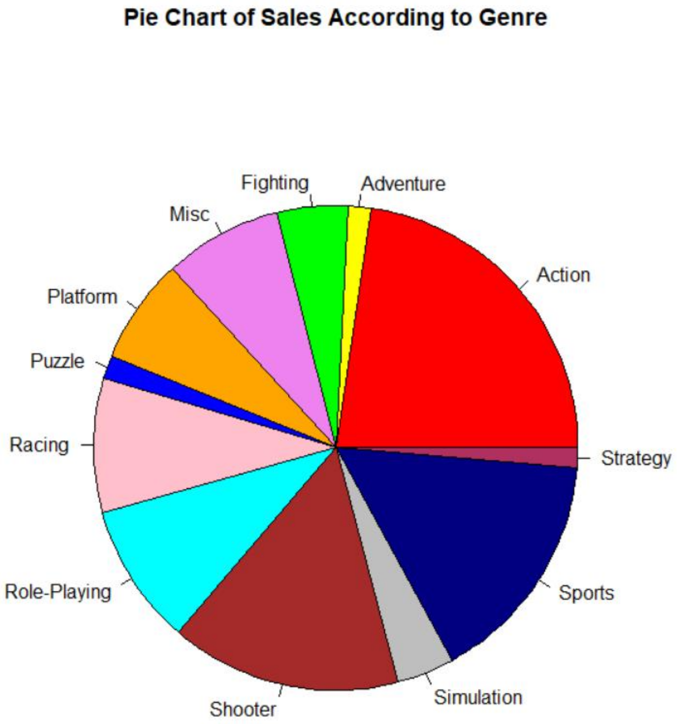
Here we can observe that PS2, PS3 and X360 got highest Sales Globally.

ii) ***Analyzing Total Sales Per Genre***

Filtering Data to add the sales of duplicate Genre to get total Sales of each Genre.

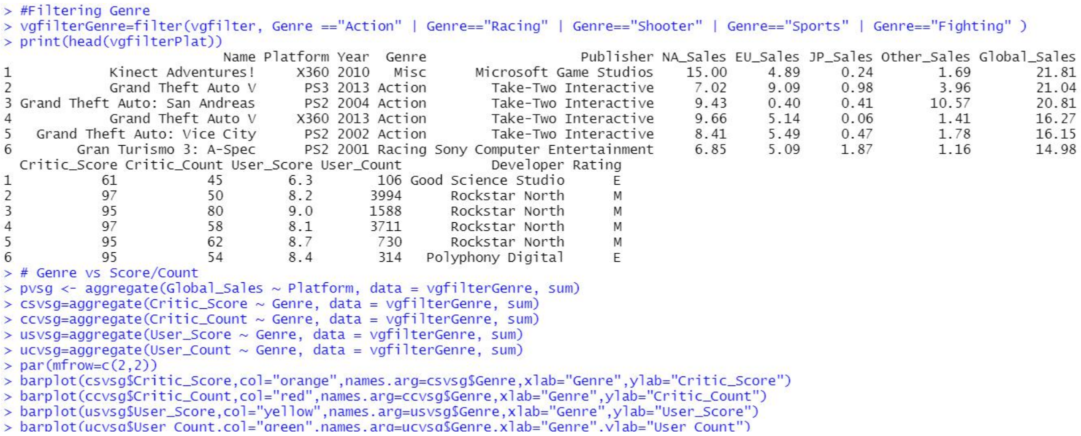


```
> filter_genre <- aggregate(x= vgfilter$Global_Sales,
+                           by= list(vgfilter$Genre),
+                           FUN=sum)
> colors = c("red", "yellow", "green", "violet",
+            "orange", "blue", "pink", "cyan","brown","grey","navy blue","maroon")
> par(mfrow=c(1,1))
> pie(filter_genre$x, filter_genre$Group.1,col=colors,main = "Pie Chart of Sales According to Genre")
```



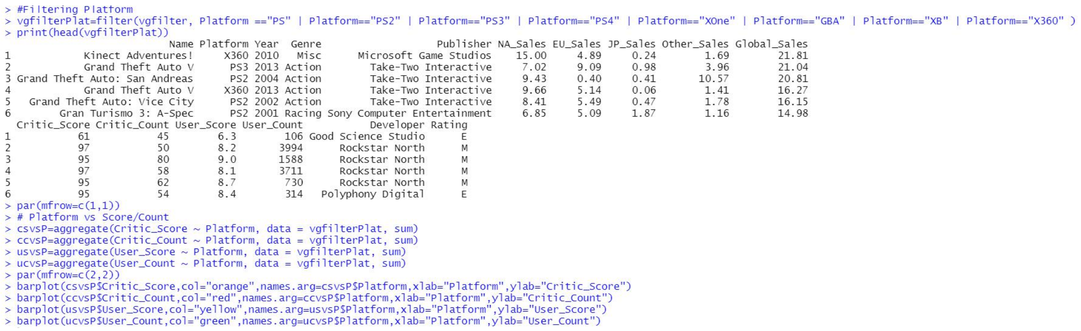
3) Comparing Platform and Genre with the Critic and Users Score/Count

i)Analyzing Genre with Critic Scores/Count and Users Scores/Count



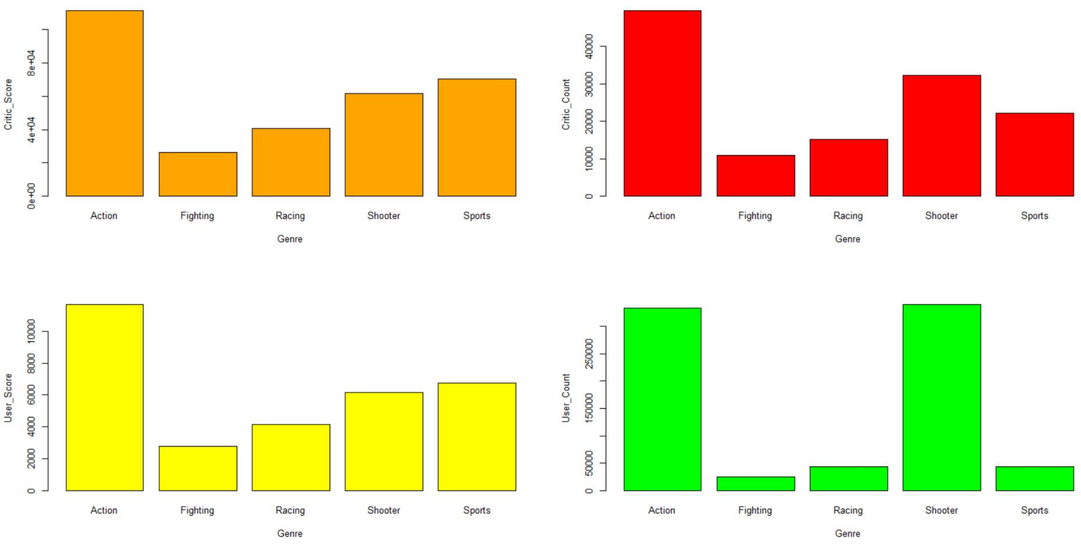
```
> #Filtering Genre
> vgfilterGenre=filter(vgfilter, Genre =="Action" | Genre=="Racing" | Genre=="Shooter" | Genre=="Sports" | Genre=="Fighting" )
> print(head(vgfilterPlat))
                         Name Platform Year  Genre                    Publisher NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales
1          Kinect Adventures!     X360 2010   Misc       Microsoft Game Studios    15.00     4.89     0.24        1.69        21.81
2          Grand Theft Auto V      PS3 2013 Action         Take-Two Interactive     7.02     9.09     0.98        3.96        21.04
3 Grand Theft Auto: San Andreas      PS2 2004 Action         Take-Two Interactive     9.43     0.40     0.41       10.57        20.81
4          Grand Theft Auto V     X360 2013 Action         Take-Two Interactive     9.66     5.14     0.06        1.41        16.27
5   Grand Theft Auto: Vice City      PS2 2002 Action         Take-Two Interactive     8.41     5.49     0.47        1.78        16.15
6        Gran Turismo 3: A-Spec      PS2 2001 Racing Sony Computer Entertainment     6.85     5.09     1.87        1.16        14.98
  Critic_Score Critic_Count User_Score User_Count           Developer Rating
1           61           45        6.3        106 Good Science Studio      E
2           97           50        8.2       3994      Rockstar North      M
3           95           80        9.0       1588      Rockstar North      M
4           97           58        8.1       3711      Rockstar North      M
5           95           62        8.7        730      Rockstar North      M
6           95           54        8.4        314   Polyphony Digital      E
> # Genre vs Score/Count
> pvsg <- aggregate(Global_Sales ~ Platform, data = vgfilterGenre, sum)
> csvsg=aggregate(Critic_Score ~ Genre, data = vgfilterGenre, sum)
> ccvsg=aggregate(Critic_Count ~ Genre, data = vgfilterGenre, sum)
> usvsg=aggregate(User_Score ~ Genre, data = vgfilterGenre, sum)
> ucvsg=aggregate(User_Count ~ Genre, data = vgfilterGenre, sum)
> par(mfrow=c(2,2))
> barplot(csvsg$Critic_Score,col="orange",names.arg=csvsg$Genre,xlab="Genre",ylab="Critic_Score")
> barplot(ccvsg$Critic_Count,col="red",names.arg=ccvsg$Genre,xlab="Genre",ylab="Critic_Count")
> barplot(usvsg$User_Score,col="yellow",names.arg=usvsg$Genre,xlab="Genre",ylab="User_Score")
> barplot(ucvsg$User_Count,col="green",names.arg=ucvsg$Genre,xlab="Genre",ylab="User_Count")
```

ii)Analyzing Platform with Critic Scores/Count and Users Scores/Count



```
> #Filtering Platform
> vgfilterPlat=filter(vgfilter, Platform =="PS" | Platform=="PS2" | Platform=="PS3" | Platform=="PS4" | Platform=="XOne" | Platform=="GBA" | Platform=="XB" | Platform=="X360" )
> print(head(vgfilterPlat))
                         Name Platform Year  Genre                    Publisher NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales
1          Kinect Adventures!     X360 2010   Misc       Microsoft Game Studios    15.00     4.89     0.24        1.69        21.81
2          Grand Theft Auto V      PS3 2013 Action         Take-Two Interactive     7.02     9.09     0.98        3.96        21.04
3 Grand Theft Auto: San Andreas      PS2 2004 Action         Take-Two Interactive     9.43     0.40     0.41       10.57        20.81
4          Grand Theft Auto V     X360 2013 Action         Take-Two Interactive     9.66     5.14     0.06        1.41        16.27
5   Grand Theft Auto: Vice City      PS2 2002 Action         Take-Two Interactive     8.41     5.49     0.47        1.78        16.15
6        Gran Turismo 3: A-Spec      PS2 2001 Racing Sony Computer Entertainment     6.85     5.09     1.87        1.16        14.98
  Critic_Score Critic_Count User_Score User_Count           Developer Rating
1           61           45        6.3        106 Good Science Studio      E
2           97           50        8.2       3994      Rockstar North      M
3           95           80        9.0       1588      Rockstar North      M
4           97           58        8.1       3711      Rockstar North      M
5           95           62        8.7        730      Rockstar North      M
6           95           54        8.4        314   Polyphony Digital      E
> par(mfrow=c(1,1))
> # Platform vs Score/Count
> csvsP=aggregate(Critic_Score ~ Platform, data = vgfilterPlat, sum)
> ccvsP=aggregate(Critic_Count ~ Platform, data = vgfilterPlat, sum)
> usvsP=aggregate(User_Score ~ Platform, data = vgfilterPlat, sum)
> ucvsP=aggregate(User_Count ~ Platform, data = vgfilterPlat, sum)
> par(mfrow=c(2,2))
> barplot(csvsP$Critic_Score,col="orange",names.arg=csvsP$Platform,xlab="Platform",ylab="Critic_Score")
> barplot(ccvsP$Critic_Count,col="red",names.arg=ccvsP$Platform,xlab="Platform",ylab="Critic_Count")
> barplot(usvsP$User_Score,col="yellow",names.arg=usvsP$Platform,xlab="Platform",ylab="User_Score")
> barplot(ucvsP$User_Count,col="green",names.arg=ucvsP$Platform,xlab="Platform",ylab="User_Count")
```

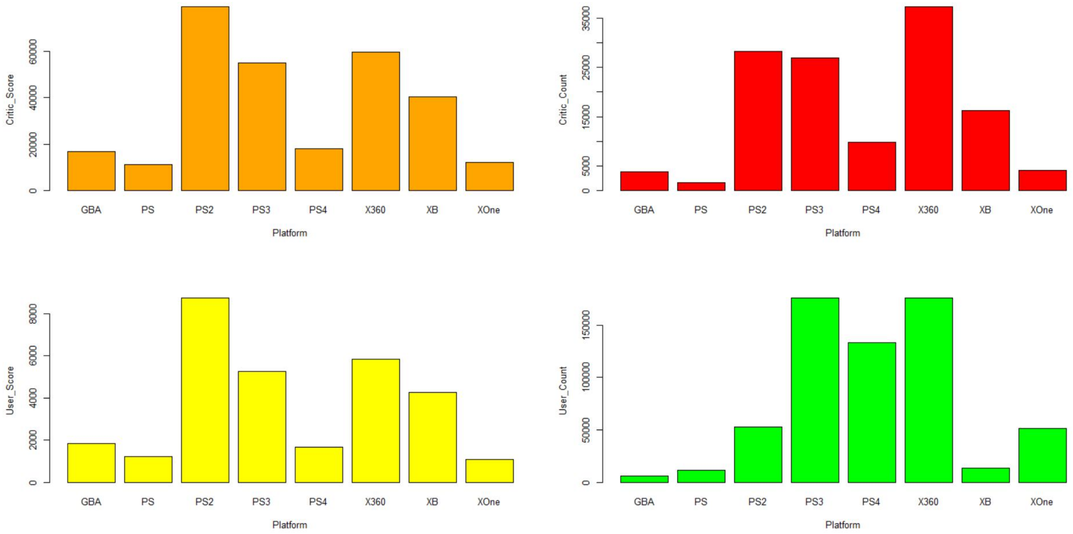
Graphs of Genre to Critic and User Values



V1.2

In the Following Graphs we can see that Action, Sports and Shooter games got Higher Scores and Count

Graphs of Platform to Critic and User Values



V1.3

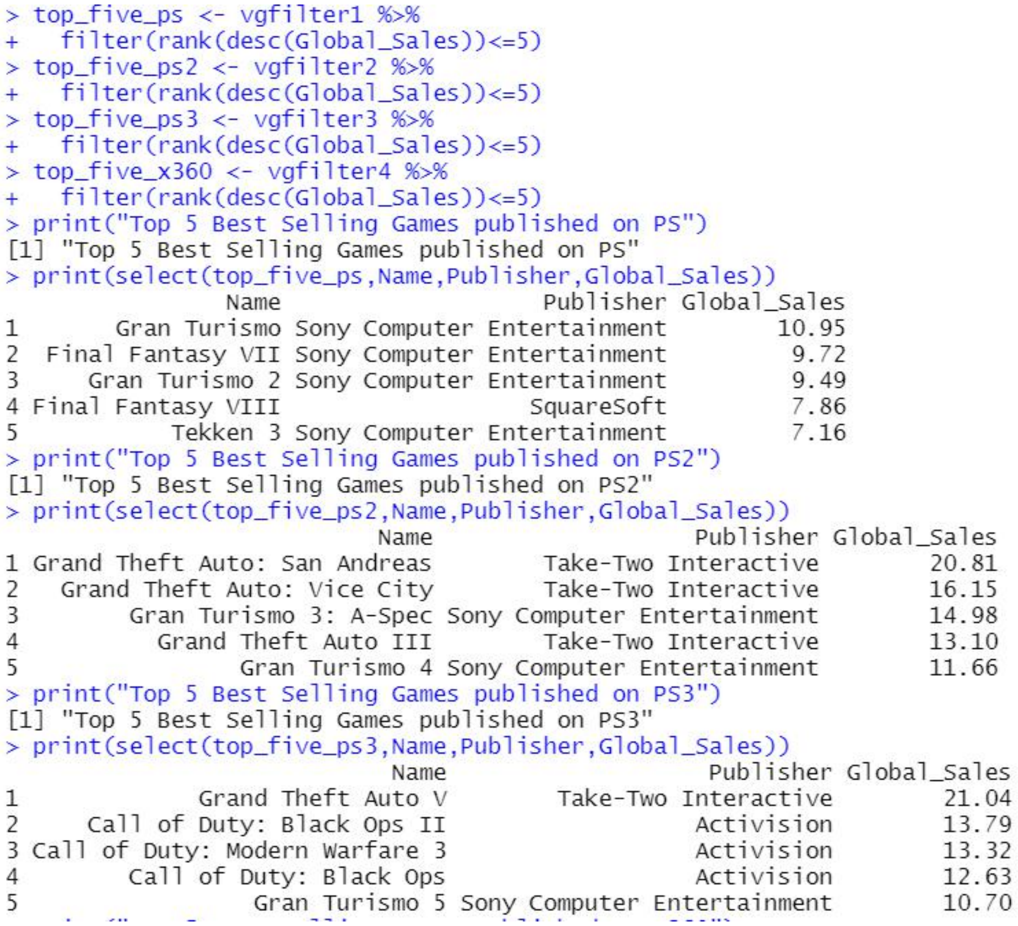
In the Following Graphs we can see that PS2 ,PS3 and X360 Got Higher Scores and Count

Therefore , according to Graph v1.01 :- PS2 ,PS3 and X360 got globally high sales. Graph v1.02/v1.2 and v1.3 shows that Genre : Action, Sports and Shooter and Platform : PS2 ,PS3 and X360 got higher values for User and Critic - Scores and Count.

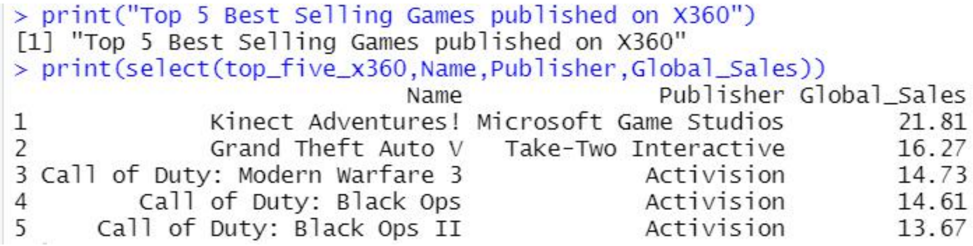
So,we can say that if Action, Sports and Shooter games were Launched on (PS2/PS3/X360) their Sales would be higher.

4) Printing Top 5 Sales Per Platform

Filtering data to get top 5 Best Selling Games Data on Biggest Platform (PS/XBOX).



```
> top_five_ps <- vgfilter1 %>%
+   filter(rank(desc(Global_Sales))<=5)
> top_five_ps2 <- vgfilter2 %>%
+   filter(rank(desc(Global_Sales))<=5)
> top_five_ps3 <- vgfilter3 %>%
+   filter(rank(desc(Global_Sales))<=5)
> top_five_x360 <- vgfilter4 %>%
+   filter(rank(desc(Global_Sales))<=5)
> print("Top 5 Best Selling Games published on PS")
[1] "Top 5 Best Selling Games published on PS"
> print(select(top_five_ps,Name,Publisher,Global_Sales))
                Name                    Publisher Global_Sales
1       Gran Turismo Sony Computer Entertainment        10.95
2  Final Fantasy VII Sony Computer Entertainment         9.72
3     Gran Turismo 2 Sony Computer Entertainment         9.49
4 Final Fantasy VIII                   SquareSoft         7.86
5          Tekken 3 Sony Computer Entertainment         7.16
> print("Top 5 Best Selling Games published on PS2")
[1] "Top 5 Best Selling Games published on PS2"
> print(select(top_five_ps2,Name,Publisher,Global_Sales))
                           Name                   Publisher Global_Sales
1 Grand Theft Auto: San Andreas        Take-Two Interactive        20.81
2   Grand Theft Auto: Vice City        Take-Two Interactive        16.15
3        Gran Turismo 3: A-Spec Sony Computer Entertainment        14.98
4          Grand Theft Auto III        Take-Two Interactive        13.10
5                Gran Turismo 4 Sony Computer Entertainment        11.66
> print("Top 5 Best Selling Games published on PS3")
[1] "Top 5 Best Selling Games published on PS3"
> print(select(top_five_ps3,Name,Publisher,Global_Sales))
                            Name                   Publisher Global_Sales
1             Grand Theft Auto V        Take-Two Interactive        21.04
2      Call of Duty: Black Ops II                  Activision        13.79
3 Call of Duty: Modern Warfare 3                  Activision        13.32
4         Call of Duty: Black Ops                  Activision        12.63
5                 Gran Turismo 5 Sony Computer Entertainment        10.70
```



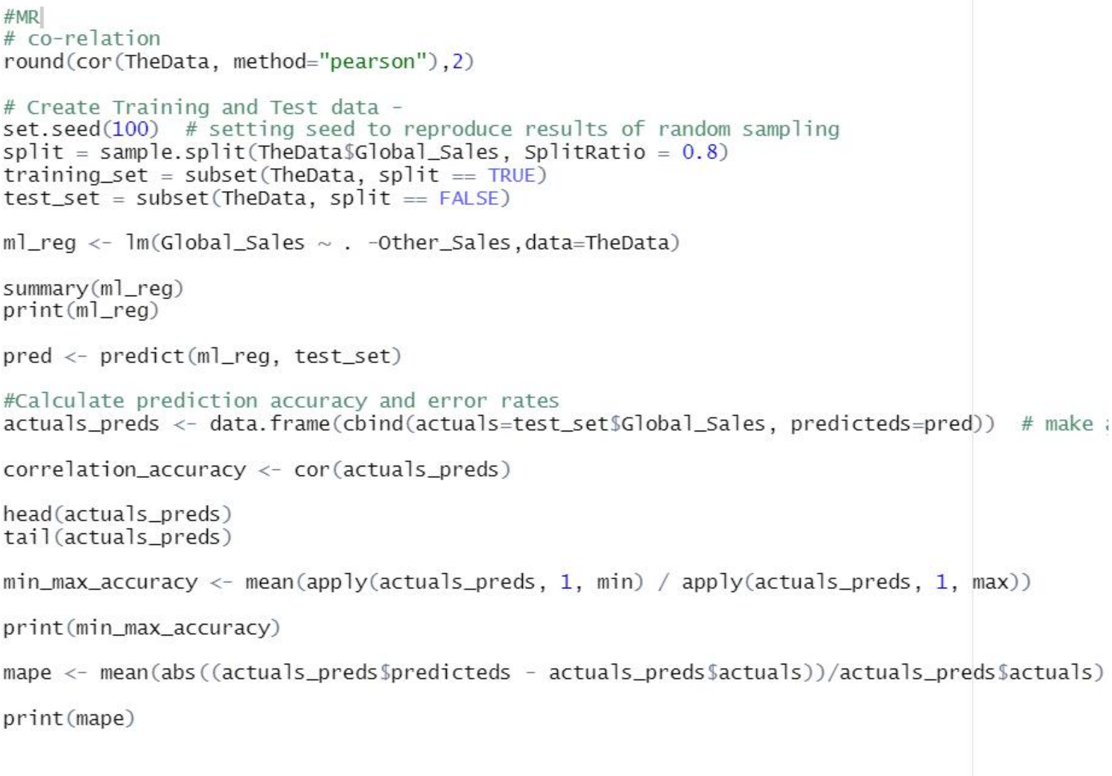
```
> print("Top 5 Best Selling Games published on X360")
[1] "Top 5 Best Selling Games published on X360"
> print(select(top_five_x360,Name,Publisher,Global_Sales))
                            Name              Publisher Global_Sales
1             Kinect Adventures! Microsoft Game Studios        21.81
2             Grand Theft Auto V   Take-Two Interactive        16.27
3 Call of Duty: Modern Warfare 3             Activision        14.73
4        Call of Duty: Black Ops             Activision        14.61
5     Call of Duty: Black Ops II             Activision        13.67
```

❖ Data Cleaning if required (R/Python)

1. Null Data was dropped.
2. Filtered to get numeric Columns.
3. Some of the Platforms and Genre were removed to keep graph clean.

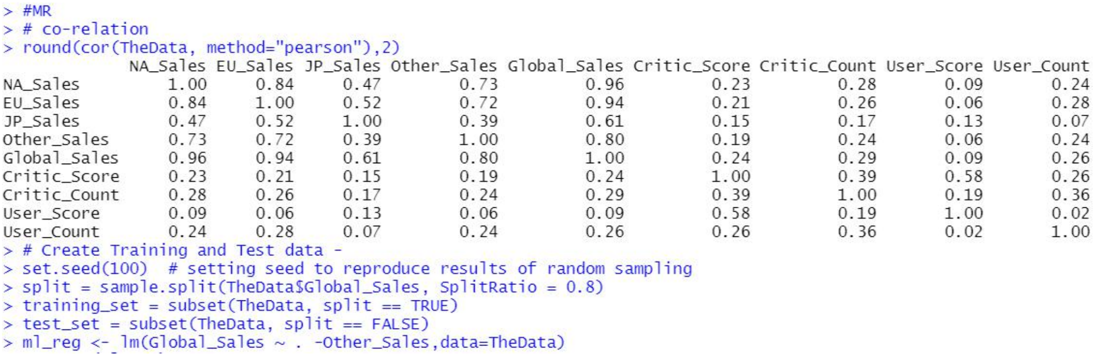
❖ Code (Multiple models) (R/Python)

MULTIPLE REGRESSION

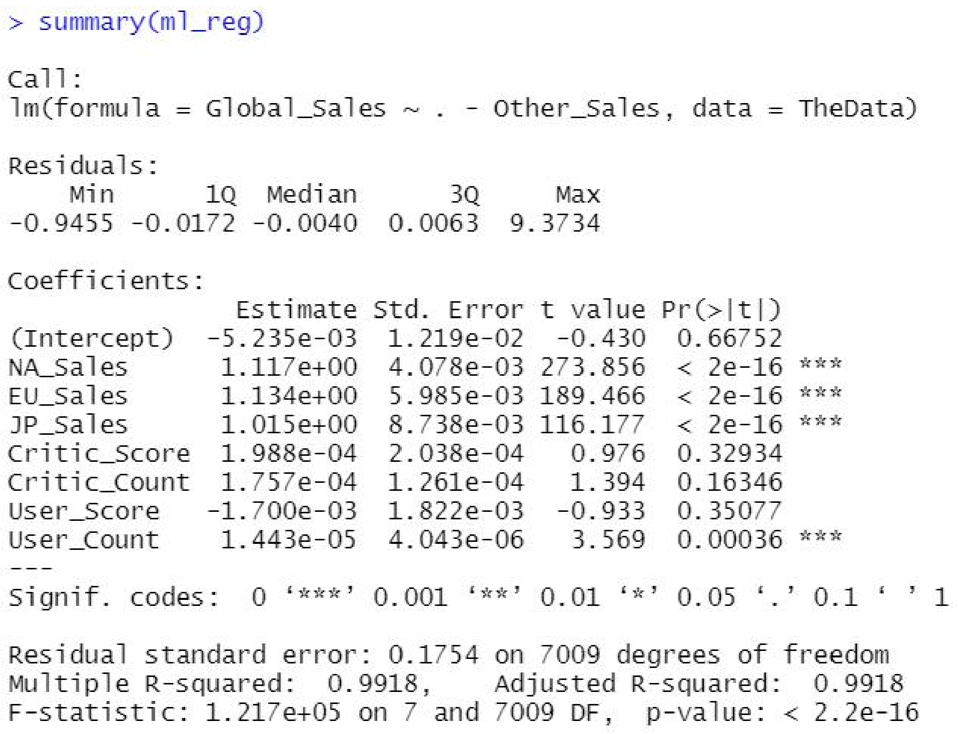


```
#MR
# co-relation
round(cor(TheData, method="pearson"),2)

# Create Training and Test data -
set.seed(100)  # setting seed to reproduce results of random sampling
split = sample.split(TheData$Global_Sales, SplitRatio = 0.8)
training_set = subset(TheData, split == TRUE)
test_set = subset(TheData, split == FALSE)

ml_reg <- lm(Global_Sales ~ . -Other_Sales,data=TheData)

summary(ml_reg)
print(ml_reg)

pred <- predict(ml_reg, test_set)

#Calculate prediction accuracy and error rates
actuals_preds <- data.frame(cbind(actuals=test_set$Global_Sales, predicteds=pred))  # make a

correlation_accuracy <- cor(actuals_preds)

head(actuals_preds)
tail(actuals_preds)

min_max_accuracy <- mean(apply(actuals_preds, 1, min) / apply(actuals_preds, 1, max))

print(min_max_accuracy)

mape <- mean(abs((actuals_preds$predicteds - actuals_preds$actuals))/actuals_preds$actuals)

print(mape)
```

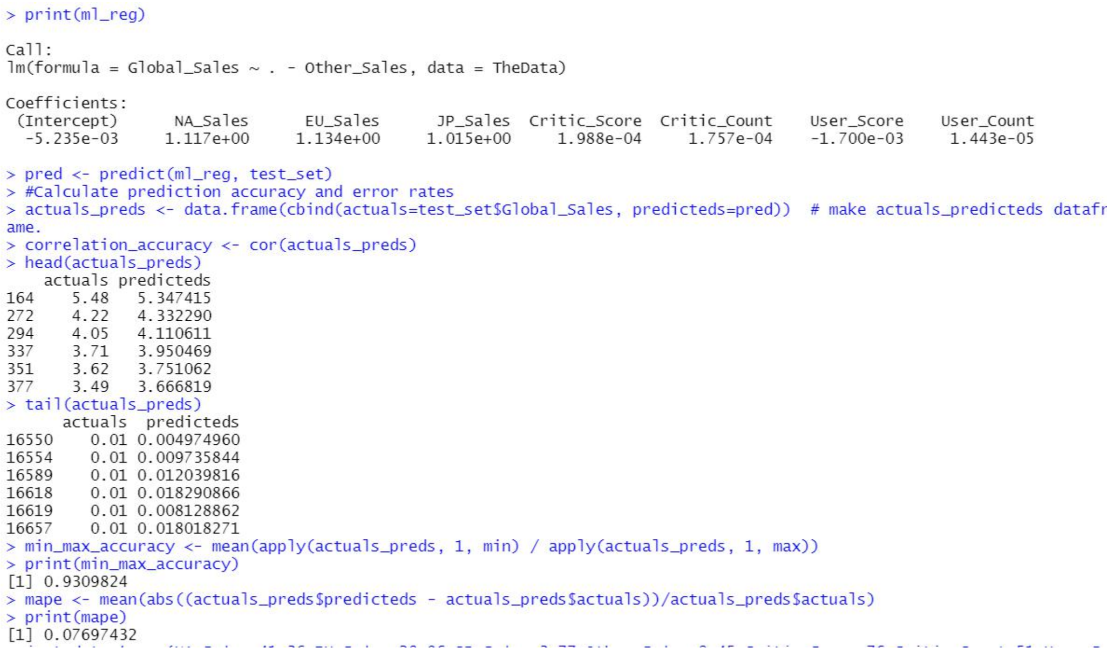
❖ Results



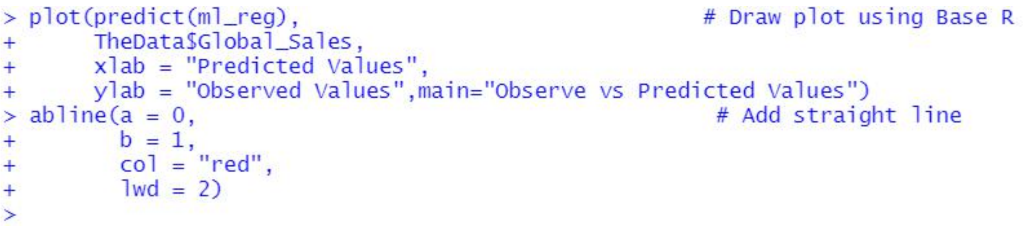
```
> #MR
> # co-relation
> round(cor(TheData, method="pearson"),2)
             NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales Critic_Score Critic_Count User_Score User_Count
NA_Sales         1.00     0.84     0.47        0.73         0.96         0.23         0.28       0.09       0.24
EU_Sales         0.84     1.00     0.52        0.72         0.94         0.21         0.26       0.06       0.28
JP_Sales         0.47     0.52     1.00        0.39         0.61         0.15         0.17       0.13       0.07
Other_Sales      0.73     0.72     0.39        1.00         0.80         0.19         0.24       0.06       0.24
Global_Sales     0.96     0.94     0.61        0.80         1.00         0.24         0.29       0.09       0.26
Critic_Score     0.23     0.21     0.15        0.19         0.24         1.00         0.39       0.58       0.26
Critic_Count     0.28     0.26     0.17        0.24         0.29         0.39         1.00       0.19       0.36
User_Score       0.09     0.06     0.13        0.06         0.09         0.58         0.19       1.00       0.02
User_Count       0.24     0.28     0.07        0.24         0.26         0.26         0.36       0.02       1.00
> # Create Training and Test data -
> set.seed(100)  # setting seed to reproduce results of random sampling
> split = sample.split(TheData$Global_Sales, SplitRatio = 0.8)
> training_set = subset(TheData, split == TRUE)
> test_set = subset(TheData, split == FALSE)
> ml_reg <- lm(Global_Sales ~ . -Other_Sales,data=TheData)
```



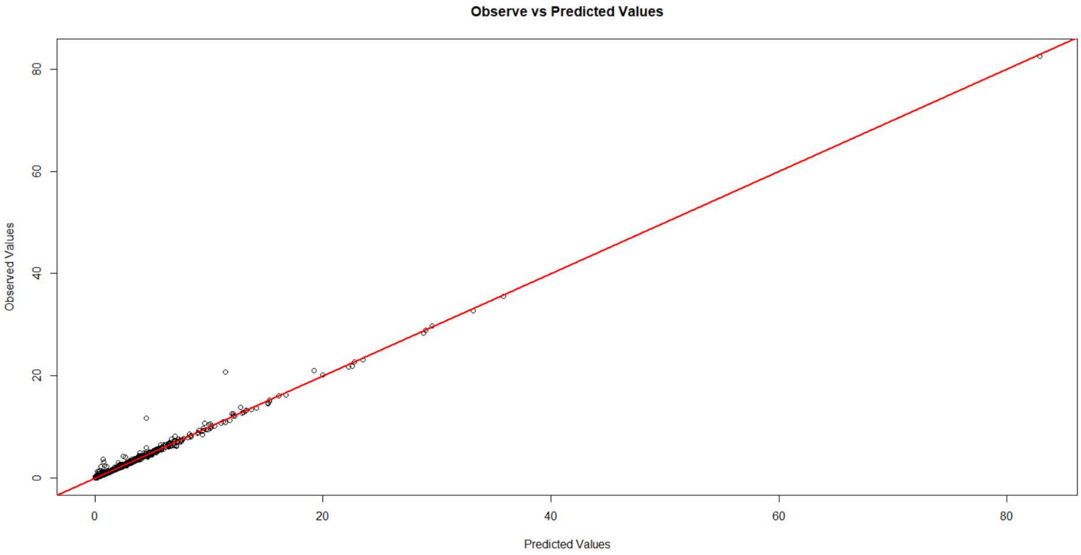
```
> summary(ml_reg)

Call:
lm(formula = Global_Sales ~ . - Other_Sales, data = TheData)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.9455 -0.0172 -0.0040  0.0063  9.3734 

Coefficients:
                Estimate Std. Error t value Pr(>|t|)    
(Intercept)   -5.235e-03  1.219e-02  -0.430  0.66752    
NA_Sales       1.117e+00  4.078e-03 273.856  < 2e-16 ***
EU_Sales       1.134e+00  5.985e-03 189.466  < 2e-16 ***
JP_Sales       1.015e+00  8.738e-03 116.177  < 2e-16 ***
Critic_Score   1.988e-04  2.038e-04   0.976  0.32934    
Critic_Count   1.757e-04  1.261e-04   1.394  0.16346    
User_Score    -1.700e-03  1.822e-03  -0.933  0.35077    
User_Count     1.443e-05  4.043e-06   3.569  0.00036 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1754 on 7009 degrees of freedom
Multiple R-squared:  0.9918,	Adjusted R-squared:  0.9918 
F-statistic: 1.217e+05 on 7 and 7009 DF,  p-value: < 2.2e-16
```



```
> print(ml_reg)

Call:
lm(formula = Global_Sales ~ . - Other_Sales, data = TheData)

Coefficients:
 (Intercept)      NA_Sales      EU_Sales      JP_Sales  Critic_Score  Critic_Count    User_Score    User_Count  
  -5.235e-03     1.117e+00     1.134e+00     1.015e+00     1.988e-04     1.757e-04    -1.700e-03     1.443e-05  

> pred <- predict(ml_reg, test_set)
> #Calculate prediction accuracy and error rates
> actuals_preds <- data.frame(cbind(actuals=test_set$Global_Sales, predicteds=pred))  # make actuals_predicteds datafr
ame.
> correlation_accuracy <- cor(actuals_preds)
> head(actuals_preds)
    actuals predicteds
164    5.48   5.347415
272    4.22   4.332290
294    4.05   4.110611
337    3.71   3.950469
351    3.62   3.751062
377    3.49   3.666819
> tail(actuals_preds)
      actuals  predicteds
16550    0.01 0.004974960
16554    0.01 0.009735844
16589    0.01 0.012039816
16618    0.01 0.018290866
16619    0.01 0.008128862
16657    0.01 0.018018271
> min_max_accuracy <- mean(apply(actuals_preds, 1, min) / apply(actuals_preds, 1, max))
> print(min_max_accuracy)
[1] 0.9309824
> mape <- mean(abs((actuals_preds$predicteds - actuals_preds$actuals))/actuals_preds$actuals)
> print(mape)
[1] 0.07697432
```



```
> plot(predict(ml_reg),                                 # Draw plot using Base R
+      TheData$Global_Sales,
+      xlab = "Predicted Values",
+      ylab = "Observed Values",main="Observe vs Predicted Values")
> abline(a = 0,                                          # Add straight line
+        b = 1,
+        col = "red",
+        lwd = 2)
> 
```



❖ Conclusion

Hence, we can conclude that if Action, Sports and Shooter games were Launched on (PS2/PS3/X360) their Sales would be higher.
Also Our Prediction model for Global Sales is 93% accurate.