HarvardX Capstone: Choose Your Own

Most Popular Video Game Console: 1980-2017

By

Jillian Combest

Introduction

In this project we will be using a dataset that consists of about several video games and information regarding their sales, customer ratings, game consoles, and producer details

Dataset link is below for more detail:
https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings

The goal of this analysis is to determine which video game console was most preferred among players between the years 1980 to 2017. We will be analyzing sales, game release quantities, and user scores.

First, we will begin by importing the libraries we will be using.

```
# import of several libraries that will be used in project
library(readr)
library(dplyr)
library(ggplot2)
library(tidyr)

options(warn = -1)
```

Next, let's import the dataset and the generic functions we will be using.

We will be looking at type of data, categorical and numeric variables, and statistical distributions.

```
# Import and read dataset
Video_Games_Sales_as_at_22_Dec_2016 <-
read.csv("~/Downloads/Video_Games_Sales_as_at_22_Dec_2016.csv")
View(Video_Games_Sales_as_at_22_Dec_2016)

# Examine the data types of every variable
 str(Video_Games_Sales_as_at_22_Dec_2016)
```

```
'data.frame':  16719 obs. of  16 variables:
 $ Name          : chr  "Wii Sports" "Super Mario Bros." "Mario Kart Wii" "Wii Sports
Resort" ...
 $ Platform      : chr  "Wii" "NES" "Wii" "Wii" ...
```

```
$ Year_of_Release: chr  "2006" "1985" "2008" "2009" ...
$ Genre          : chr  "Sports" "Platform" "Racing" "Sports" ...
$ Publisher      : chr  "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
$ NA_Sales       : num  41.4 29.1 15.7 15.6 11.3 ...
$ EU_Sales       : num  28.96 3.58 12.76 10.93 8.89 ...
$ JP_Sales       : num  3.77 6.81 3.79 3.28 10.22 ...
$ Other_Sales    : num  8.45 0.77 3.29 2.95 1 0.58 2.88 2.84 2.24 0.47 ...
$ Global_Sales   : num  82.5 40.2 35.5 32.8 31.4 ...
$ Critic_Score   : int  76 NA 82 80 NA NA 89 58 87 NA ...
$ Critic_Count   : int  51 NA 73 73 NA NA 65 41 80 NA ...
$ User_Score     : chr  "8" "" "8.3" "8" ...
$ User_Count     : int  322 NA 709 192 NA NA 431 129 594 NA ...
$ Developer      : chr  "Nintendo" "" "Nintendo" "Nintendo" ...
$ Rating         : chr  "E" "" "E" "E" ...
```

```
dim(Video_Games_Sales_as_at_22_Dec_2016)
```

```
[1] 16719    16
```

```
# Examine statistical details of non-categorical variables
summary(Video_Games_Sales_as_at_22_Dec_2016)
```

```
    Name           Platform        Year_of_Release       Genre          
 Length:16719       Length:16719       Length:16719       Length:16719      
 Class :character   Class :character   Class :character   Class :character  
 Mode  :character   Mode  :character   Mode  :character   Mode  :character  
```

```
  Publisher          NA_Sales         EU_Sales         JP_Sales       
 Length:16719       Min.   : 0.0000   Min.   : 0.000   Min.   : 0.0000  
 Class :character   1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.: 0.0000  
 Mode  :character   Median : 0.0800   Median : 0.020   Median : 0.0000  
                    Mean   : 0.2633   Mean   : 0.145   Mean   : 0.0776  
                    3rd Qu.: 0.2400   3rd Qu.: 0.110   3rd Qu.: 0.0400  
                    Max.   :41.3600   Max.   :28.960   Max.   :10.2200  

 Other_Sales        Global_Sales      Critic_Score    Critic_Count   
 Min.   : 0.00000   Min.   : 0.0100   Min.   :13.00   Min.   :  3.00  
 1st Qu.: 0.00000   1st Qu.: 0.0600   1st Qu.:60.00   1st Qu.: 12.00  
```

```
 Median : 0.01000   Median : 0.1700   Median :71.00   Median : 21.00
 Mean   : 0.04733   Mean   : 0.5335   Mean   :68.97   Mean   : 26.36
 3rd Qu.: 0.03000   3rd Qu.: 0.4700   3rd Qu.:79.00   3rd Qu.: 36.00
 Max.   :10.57000   Max.   :82.5300   Max.   :98.00   Max.   :113.00
                                      NA's   :8582    NA's   :8582
  User_Score         User_Count       Developer          Rating
 Length:16719       Min.   :   4.0   Length:16719       Length:16719
 Class :character   1st Qu.:  10.0   Class :character   Class :character
 Mode  :character   Median :  24.0   Mode  :character   Mode  :character
                    Mean   : 162.2
                    3rd Qu.:  81.0
                    Max.   :10665.0
                    NA's   :9129
```

```
# Examine the details of categorical variables
categorical <-
names(Video_Games_Sales_as_at_22_Dec_2016)[sapply(Video_Games_Sales_as_at_22_De
c_2016, is.character)]
print(categorical)
[1] "Name"          "Platform"      "Year_of_Release" "Genre"
[5] "Publisher"     "User_Score"    "Developer"       "Rating"
```

```
summary(Video_Games_Sales_as_at_22_Dec_2016[categorical])
     Name            Platform        Year_of_Release       Genre
 Length:16719       Length:16719       Length:16719       Length:16719
 Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character
  Publisher          User_Score         Developer          Rating
 Length:16719       Length:16719       Length:16719       Length:16719
 Class :character   Class :character   Class :character   Class :character
 Mode  :character   Mode  :character   Mode  :character   Mode  :character
```

Data Cleaning

Now, before we begin visualization, let's go through the statistical information of the dataset to gather more details.

First, null values will be controlled in the data;

```
 # Count of NA values
sum(is.na(Video_Games_Sales_as_at_22_Dec_2016))
[1] 26293

# Sum of NA values per column
colSums(is.na(Video_Games_Sales_as_at_22_Dec_2016))
        Name     Platform Year_of_Release        Genre    Publisher
           0            0            0            0            0
    NA_Sales     EU_Sales     JP_Sales  Other_Sales Global_Sales
           0            0            0            0            0
Critic_Score Critic_Count   User_Score   User_Count    Developer
        8582         8582            0         9129            0
      Rating
           0
```

According to the null value check, null values generally seem to fill in some of the columns. We are not planning to use it in our analysis so let us get rid of all the null values. There is no need to fill them in with mean, median or any other type of generic data.

```
url <- na.omit(Video_Games_Sales_as_at_22_Dec_2016)
```

.

```
# Convert "Year of Release" variable to an object
url$Year_of_Release <- as.character(url$Year_of_Release)
head(url$Year_of_Release)
[1] "2006" "2008" "2009" "2006" "2006" "2009"
```

According to unique variables of "Platform" it will be useful to add one more "Platform_General" column to group some of these platform types under larger groups such as PS,PS3 and PS4 under Playstation and X360,XB and Xone under Xbox categories. We should assume this categorization will lead us to perform comparisons between the two biggest game consoles..

After we create the Platform_General column, let's group and name the three main console brands under the same name, such as Playstation, Nintendo and Microsoft(XBox).

```
# Check unique values of Platform
unique(url$Platform)
 [1] "Wii"  "DS"   "X360" "PS3"  "PS2"  "3DS"  "PS4"  "PS"   "XB"   "PC"   "PSP"
[12] "WiiU" "GC"   "GBA"  "XOne" "PSV"  "DC"

# Copy and add one more platform column to the end of dataset
url$Platform_General <- url$Platform

#Convert console subnames to the general names
 url$Platform_General[url$Platform == 'PS3'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'PS'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'PS2'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'PS4'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'PSP'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'PSV'] <- 'Sony_Playstation'
 url$Platform_General[url$Platform == 'Wii'] <- 'Nintendo'
 url$Platform_General[url$Platform == 'DS'] <- 'Nintendo'
 url$Platform_General[url$Platform == 'GBA'] <- 'Nintendo'
url$Platform_General[url$Platform == '3DS'] <- 'Nintendo'
url$Platform_General[url$Platform == 'WiiU'] <- 'Nintendo'
url$Platform_General[url$Platform == 'NES'] <- 'Nintendo'
url$Platform_General[url$Platform == 'SNES'] <- 'Nintendo'
url$Platform_General[url$Platform == 'N64'] <- 'Nintendo'
url$Platform_General[url$Platform == 'GB'] <- 'Nintendo'
url$Platform_General[url$Platform == 'GC'] <- 'Nintendo'
url$Platform_General[url$Platform == 'X360'] <- 'Microsoft_Xbox'
url$Platform_General[url$Platform == 'XB'] <- 'Microsoft_Xbox'
url$Platform_General[url$Platform == 'XOne'] <- 'Microsoft_Xbox'
url$Platform_General[url$Platform == '2600'] <- 'Atari'
url$Platform_General[url$Platform == 'DC'] <- 'Sega'
url$Platform_General[url$Platform == 'SAT'] <- 'Sega'
url$Platform_General[url$Platform == 'GG'] <- 'Sega'
url$Platform_General[url$Platform == 'WS'] <- 'Bandal'
url$Platform_General[url$Platform == 'TG16'] <- 'Nec'
url$Platform_General[url$Platform == 'PCFX'] <- 'Nec'
url$Platform_General[url$Platform == '3DO'] <- 'Panasonic'
```

```
# Check uniq values of Platform_General
unique(url$Platform_General)
[1] "Nintendo"         "Microsoft_Xbox"   "Sony_Playstation" "PC"
[5] "Sega"
```

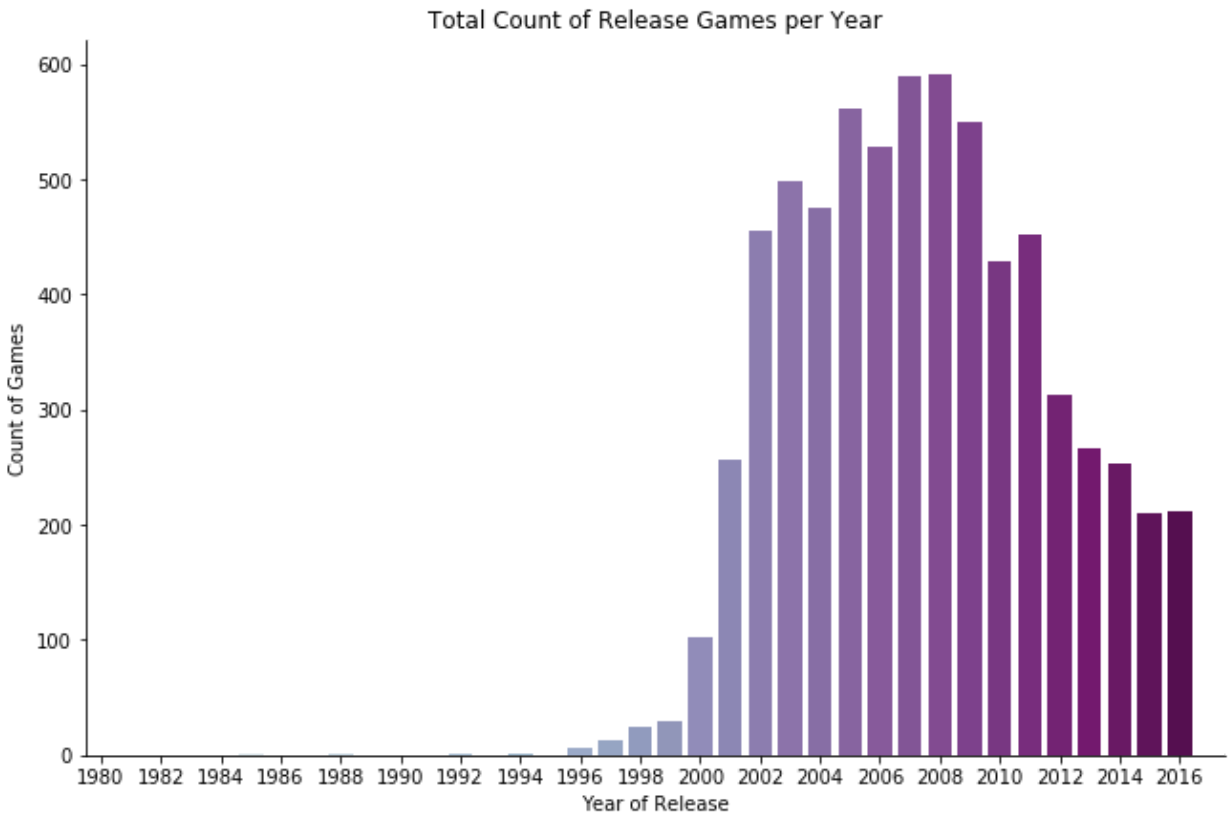
Generic Data Analysis

As we start our analysis, let's see how the game industry evolved based on the number of games published every year between 1980 and 2017.

The Number of games published started to increase in the mid 1990's and it is estimated that one of the biggest reasons for this is due to the release of first versions of the Playstation in 1994 and the Xbox in 2001.

```
c(1980:2017)
 [1] 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991 1992 1993 1994 1995
[17] 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005 2006 2007 2008 2009 2010 2011
[33] 2012 2013 2014 2015 2016 2017
```

```
ggplot(url, aes(x = Year_of_Release)) +
  geom_bar(fill = BuPu) +
  scale_x_discrete(breaks = seq.default(min(url$Year_of_Release), max(url$Year_of_Release),
by = 2)+
  labs(x = Year_of_Release, y = Count_of_Games, title =
Total_Count_of_Release_Games_per_Year)
```



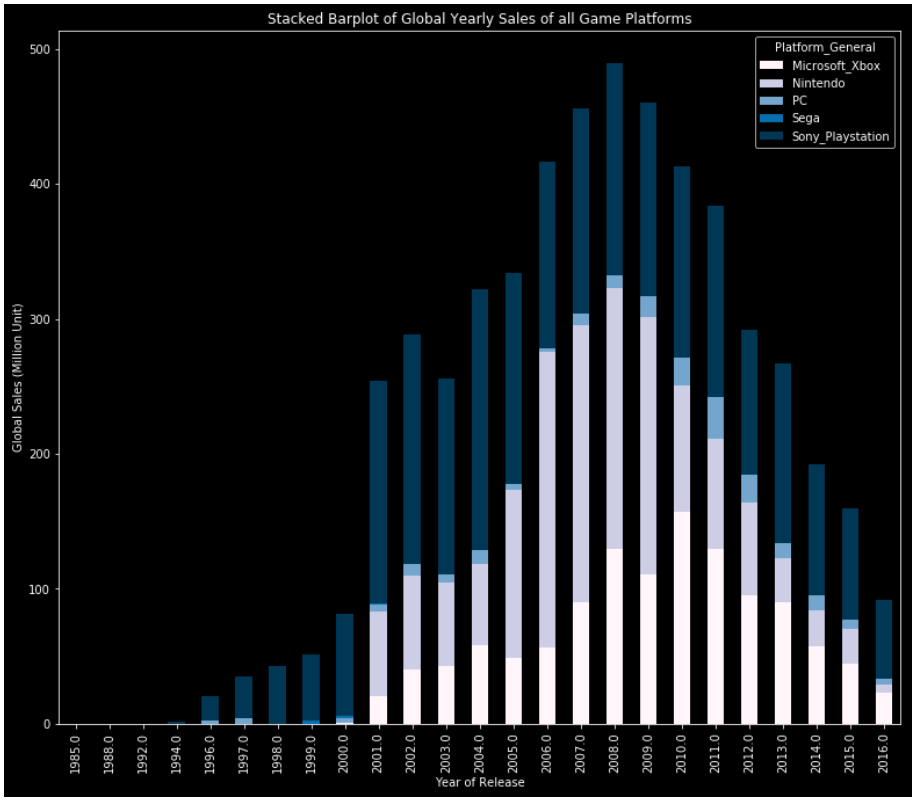
Let us dig into the details. First we will go through the platform distribution of released games per year and global sales per year between 1994 - 2017. To do this, we will check for all consoles and after that we will pick the top three to continue our analysis.

According to the graph below, Sony Playstation, Xbox and Nintendo dominate the market in reference to the amount of published and sold games.

```
yearlySales <- url%>%
    group_by(Year_of_Release, Platform_General) %>%
    summarize(Global_Sales = sum(Global_Sales))
```

```
yearlySales %>%
  pivot_wider(names_from = url$Platform_General, values_from = Global_Sales) %>%
  ggplot(aes(x = Year_of_Release)) +
  geom_bar(aes(fill = url$Platform_General), stat = "identity", position = "stack") +
  scale_fill_brewer(palette = "PuBu") +
  theme(panel.grid = element_blank()) +
  labs(title = "Stacked Barplot of Global Yearly Sales of all Game Platforms",
       y = "Global Sales (Million Unit)",
       x = "Year of Release")
```

Text(0.5,0,'Year of Release')



From this point moving forward, we will continue our analysis using these three consoles only. To perform this we need to create a dataset that contains information of these selected game consoles.

First, let's make a statistical analysis of the dataset to define outliers or any other possible data that may affect the solution.

Statistical Analysis

Because we are only going to analyse the top 3 consoles, we need to create a subset of Sony_Playstation, Microsoft_Xbox and Nintendo.

```
url_platform <- url %>%
  filter(Platform_General %in% c('Sony_Playstation', 'Microsoft_Xbox', 'Nintendo')) %>%
  mutate(dummy_count = as.numeric(1),
         User_Score = as.numeric(User_Score)) %>%
  summary()
```

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales	Critic_Score	Critic_Count	User_Score	User_Count	dummy_count
count	6160.000000	6160.000000	6160.000000	6160.000000	6160.000000	6160.000000	6160.000000	6160.000000	6160.000000	6160.0
mean	0.426469	0.244513	0.070625	0.088703	0.830419	69.600325	28.979383	7.195812	119.513961	1.0
std	1.006861	0.708645	0.301780	0.282232	2.047041	13.971186	19.397787	1.429799	417.314998	0.0
min	0.000000	0.000000	0.000000	0.000000	0.010000	13.000000	3.000000	0.500000	4.000000	1.0
25%	0.080000	0.020000	0.000000	0.010000	0.130000	61.000000	14.000000	6.500000	10.000000	1.0
50%	0.170000	0.060000	0.000000	0.030000	0.330000	72.000000	25.000000	7.500000	23.000000	1.0
75%	0.430000	0.220000	0.020000	0.080000	0.820000	80.000000	39.000000	8.200000	69.000000	1.0
max	41.360000	28.960000	6.500000	10.570000	82.530000	98.000000	113.000000	9.600000	10179.000000	1.0

Now, let's check the statistical details of Global_Sales quantities. It is clear that there are some outliers because of the difference between mean (0,83) and median (0,33).

According to the quantile details below, 90% of global sales are under 1.81 M and 95% are under 3.01 M. So we can consider the 95% quantile of Global Sales for a more feasible analysis in terms of sales amounts.

```
url_platform$`Global_Sales`[c(0, 0.01, 0.25, 0.5, 0.75, 0.95, 1)]
```

```
0.00     0.0100
0.01     0.0200
0.25     0.1300
0.50     0.3300
0.75     0.8200
0.95     3.0105
1.00    82.5300
Name: Global_Sales, dtype: float64
```

```
url_platform <- url_platform[url_platform$Global_Sales < quantile(url_platform$Global_Sales, 0.95)]
```

```
quantile(url_platform$Global_Sales, c(0, 0.01, 0.25, 0.5, 0.75, 0.95, 1))
```

```
0.00    0.01
0.01    0.02
0.25    0.13
0.50    0.30
0.75    0.70
0.95    1.86
1.00    3.01
Name: Global_Sales, dtype: float64
```

Global Sales & Games Released per Year

Let us look at the correlation between the number of games released per year and global sales according to the game consoles. To be able to do this analysis we need to create a subset including the sales per game in terms of consoles.

```
# First group by the data

urlgc <- urla_platform %>%
  group_by(Year_of_Release, Platform_General, Platform)

# Second add counts according to group_by info and then join with the statistical information required

counts <- vdpg %>%
  summarise(counts = n()) %>%
  as.data.frame()

global_sales <- vdpg %>%
  summarise(Global_Sales_M = sum(Global_Sales)) %>%
  as.data.frame()

url_df_pf_grouped <- merge(counts, global_sales, by = c("Year_of_Release",
"Platform_General", "Platform"))

url_df_pf_grouped$GS_Amount_per_Game <- url_df_pf_grouped$Global_Sales_M /
url_df_pf_grouped$counts
url_df_pf_grouped[] <- lapply(url_df_pf_grouped, function(x) as.numeric(as.character(x)))
```

Now let's check the descriptive statistics of each platform separately before visualization. The below tables have been created for Playstation, Xbox and Nintendo. Playstation dominates the market both in terms of games released and global sales.

```
url_df_pf_grouped %>%
  filter(Platform_General == 'Sony_Playstation') %>%
  summary()
```

	Year_of_Release	counts	Global_Sales(M#)	GS_Amount_per_Game
count	48.000000	48.000000	48.000000	48.000000
mean	2007.479167	55.083333	31.262500	0.545623s
std	5.786961	51.544995	32.331395	0.280213
min	1994.000000	1.000000	0.020000	0.020000
25%	2003.750000	12.750000	5.097500	0.370187
50%	2008.000000	38.500000	19.050000	0.546404
75%	2012.250000	87.250000	45.830000	0.682222
max	2016.000000	191.000000	120.400000	1.356923d

```
url_df_pf_grouped %>%
  filter(Platform_General == 'Microsoft_Xbox') %>%
  summary()
```

	Year_of_Release	counts	Global_Sales(M#)	GS_Amount_per_Game
count	22.000000	22.000000	22.000000	22.000000
mean	2008.227273	68.863636	35.737273	0.551475
std	4.790187	45.437865	24.729521	0.227895
min	2000.000000	1.000000	0.550000	0.183333
25%	2005.000000	29.000000	14.420000	0.390407
50%	2007.500000	64.500000	37.250000	0.547944

75%	2012.750000	112.250000	52.902500	0.686163
max	2016.000000	133.000000	79.020000	1.064286

```
url_df_pf_grouped %>%
  filter(Platform_General == 'Nintendo') %>%
  summary()
```

	Year_of_Release	counts	Global_Sales(M#)	GS_Amount_per_Game
count	40.000000	40.000000	40.000000	40.000000
mean	2008.375000	42.325000	19.921000	0.485024
std	4.406798	33.933863	16.880217	0.196187
min	2001.000000	1.000000	0.220000	0.110000
25%	2005.000000	17.000000	7.525000	0.371885
50%	2008.000000	30.500000	17.875000	0.454232
75%	2012.000000	59.000000	28.395000	0.562500
max	2016.000000	121.000000	71.200000	1.113333

In the graph below we visualized the statistical information above to support the idea about Playstation global sales per game. In these graphs it can be seen that Nintendo and Xbox have a more straightforward trend while Playstation has a decreasing trend. We can make some assumptions regarding to Playstation's trend:

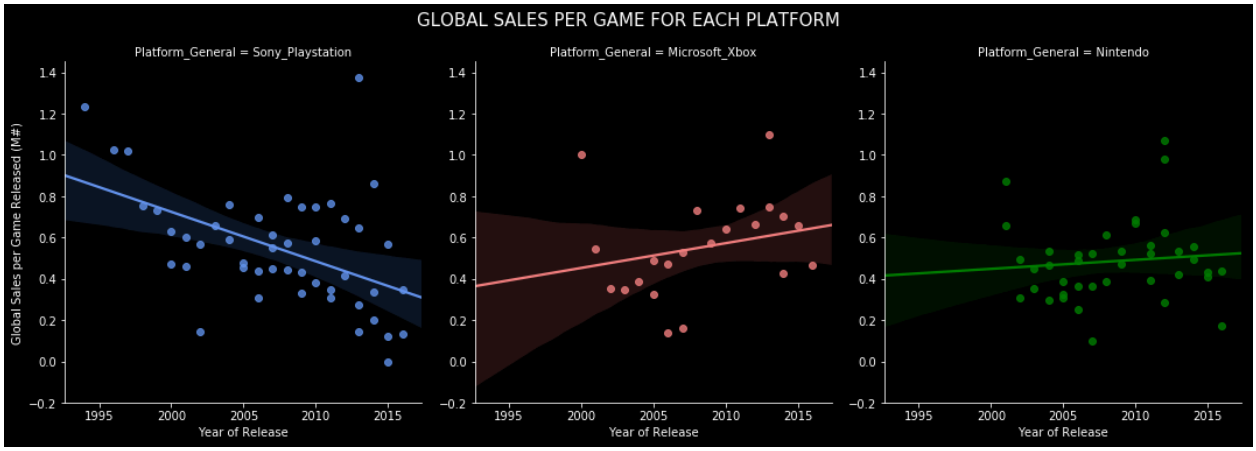
- Because Playstation is the most sold console around the world, game companies publish much more games for Playstation than Xbox and Nintendo. This most likely will result in a lot of games with low sales amounts.

- Nintendo and Xbox game companies may publish higher quality, but a less amount of games which may lead to higher sales per game than Playstation.

Now we will have to check these assumptions and see whether they are true or not.

```
# Make a custom palette with platform colors
pal <- list(Sony_Playstation = "#6495ED", Microsoft_Xbox = "#F08080", Nintendo = "Green")

# Show the survival proability as a function of platforms
g <- ggplot(url_df_pf_grouped, aes(x = Year_of_Release, y = GS_Amount_per_Game, color =
Platform_General)) +
  geom_point(position = position_jitter(height = 0.05)) +
  geom_smooth(method = "lm") +
  facet_wrap(~ Platform_General) +
  scale_color_manual(values = pal)

# Use more informative axis labels than are provided by default
g <- g + labs(x = "Year of Release", y = "Global Sales per Game Released (M#)")
g <- g + ggtitle("GLOBAL SALES PER GAME FOR EACH PLATFORM") + theme(plot.title =
element_text(size = 15, vjust = 1.5))
```

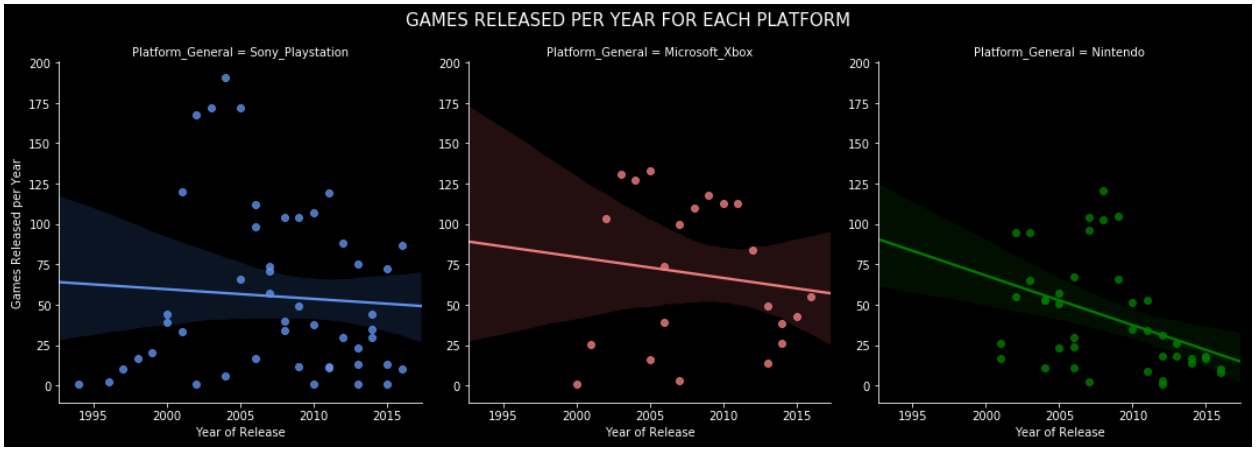
Text(0.5,1.05,'GLOBAL SALES PER GAME FOR EACH PLATFORM')

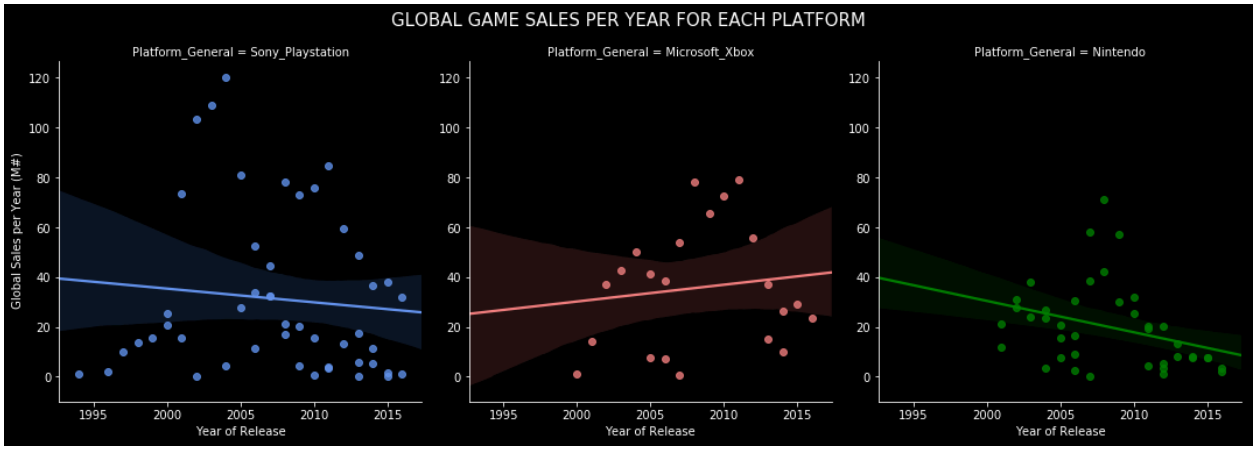



```
# Make a custom palette with platform colors
pal <- list(Sony_Playstation = "#6495ED", Microsoft_Xbox = "#F08080", Nintendo = "Green")

# Show the survival proability as a function of platforms
g <- ggplot(url_df_pf_grouped, aes(x = Year_of_Release, y = counts, color = Platform_General))
+
  geom_point(position = position_jitter(height = 0.05)) +
```

```
  geom_smooth(method = "lm") +
  facet_wrap(~ Platform_General) +
  scale_color_manual(values = pal)


# Use more informative axis labels than are provided by default
ggplot2::ggtitle("GAMES RELEASED PER YEAR FOR EACH PLATFORM", size = 15) +
  xlab("Year of Release") +
  ylab("Games Released per Year")

# Make a custom palette with platform colors
pal <- c(Sony_Playstation = "#6495ED", Microsoft_Xbox = "#F08080", Nintendo = "Green")

# Show the survival proability as a function of platforms
ggplot(vgs_df_pf_grouped, aes(x = Year_of_Release, y = Global_Sales.M., color =
Platform_General, group = Platform_General)) +
  geom_point(position = position_jitter(width = 0, height = 0.05)) +
  geom_smooth(method = "lm") +
  facet_wrap(~ Platform_General) +
  scale_color_manual(values = pal)

# Use more informative axis labels than are provided by default
ggplot(data, aes(x = Year_of_Release, y = Global_Sales_per_Year)) +
  geom_line() +
  labs(x = "Year of Release", y = "Global Sales per Year (M#)") +
  ggtitle("GLOBAL GAME SALES PER YEAR FOR EACH PLATFORM") +
  theme(plot.title = element_text(size = 15, vjust = 1.05))
```

```
Text(0.5,1.05,'GLOBAL GAME SALES PER YEAR FOR EACH PLATFORM')
```



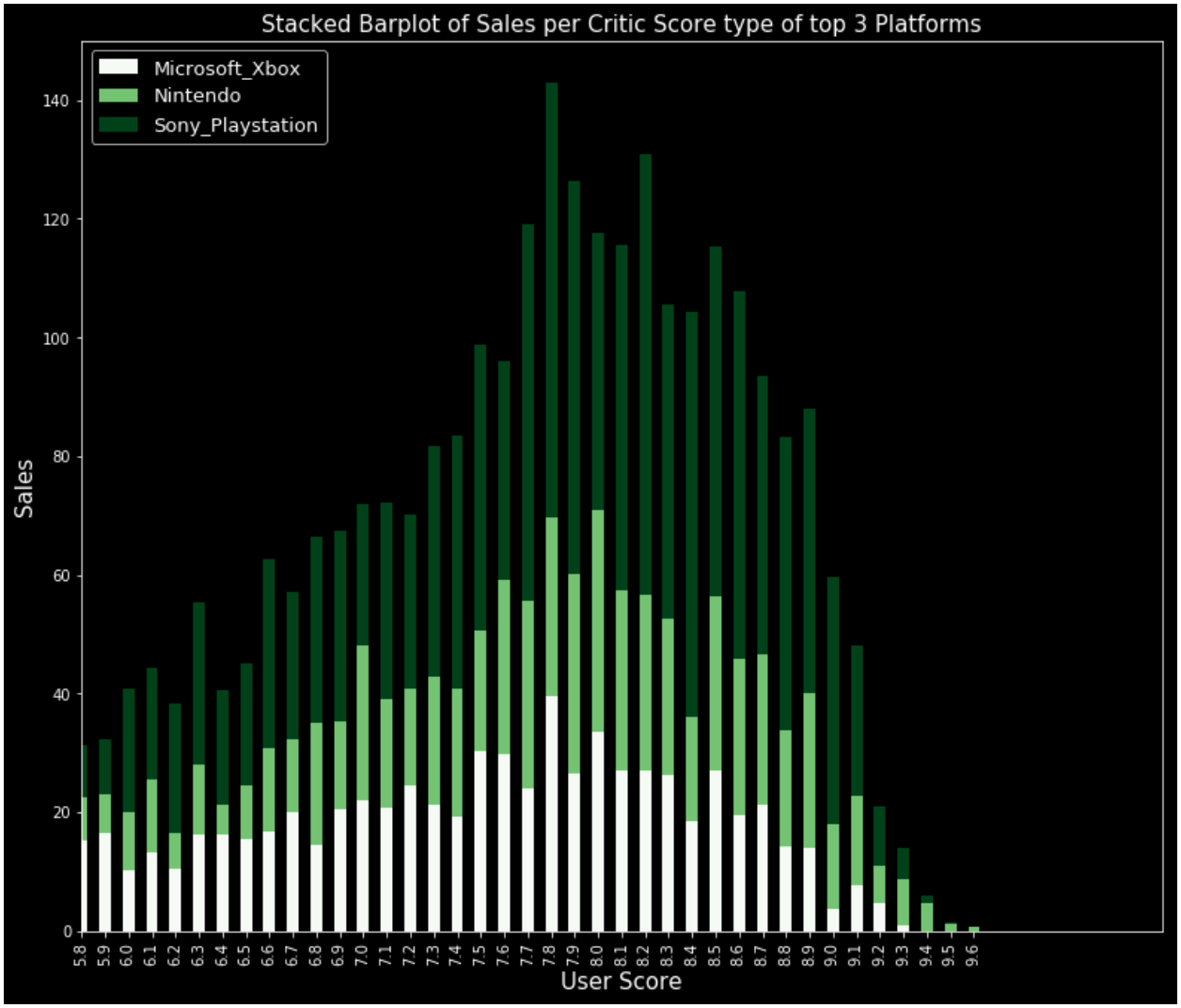

GLOBAL GAME SALES PER YEAR FOR EACH PLATFORM
Platform_General = Sony_Playstation
Platform_General = Microsoft_Xbox
Platform_General = Nintendo
Global Sales per Year (M#)
Year of Release
Year of Release
Year of Release

Critic Scores comparison between platforms

According to the comparison below, Playstation games not only get higher scores than Nintendo and Xbox, but are also selling a lot more games than its competitors. From the graph below we can assume that customer satisfaction of Playstation is much more preferred than its competitors. For specific cases we may encounter some outliers, but they won't be changing the result.

```
url_platform$Platform_General <- as.factor(url_data_platform$Platform_General)
url_data_platform$User_Score <- as.numeric(url_data_platform$User_Score)

ratingSales <- url_platform %>%
  group_by(User_Score, Platform_General) %>%
  summarize(Global_Sales = sum(Global_Sales))

ratingSales_wide <- ratingSales %>%
  pivot_wider(names_from = Platform_General, values_from = Global_Sales)

ggplot(ratingSales_wide, aes(x = User_Score)) +
  geom_bar(aes(y = ``, fill = `Platform_General`), stat = "identity", position = "stack") +
  scale_fill_brewer(palette = "Greens") +
  theme_dark() +
  labs(title = "Stacked Barplot of Sales per Critic Score type of top 3 Platforms",
       x = "User Score",
       y = "Sales") +
  xlim(50, 96) +
  theme(legend.position = "top", legend.text = element_text(size = 13))
```

```
<matplotlib.legend.Legend at 0x1131ae390>
```

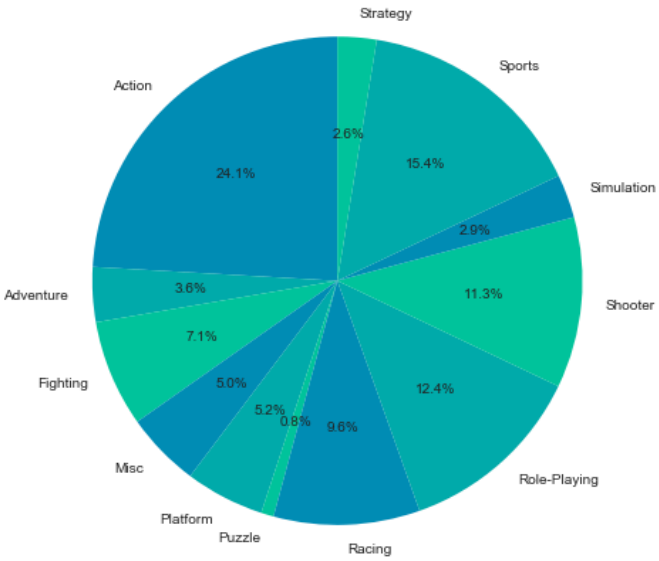

Stacked Barplot of Sales per Critic Score type of top 3 Platforms
Microsoft_Xbox
Nintendo
Sony_Playstation
Sales
User Score
0
20
40
60
80
100
120
140
5.8
5.9
6.0
6.1
6.2
6.3
6.4
6.5
6.6
6.7
6.8
6.9
7.0
7.1
7.2
7.3
7.4
7.5
7.6
7.7
7.8
7.9
8.0
8.1
8.2
8.3
8.4
8.5
8.6
8.7
8.8
8.9
9.0
9.1
9.2
9.3
9.4
9.5
9.6

Genre distribution of games

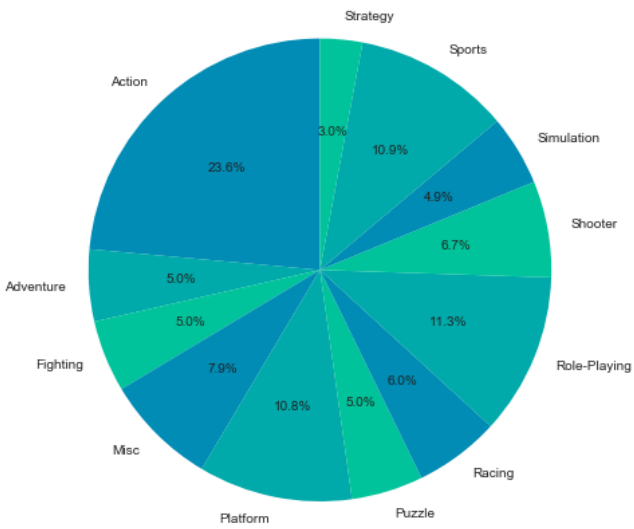
Genre distribution of games for the different consoles can be seen in the pie charts below. According to these charts action games are the most desirable genre in all platforms then it goes to sports, shooter games and role & playing.

```
colors <- c('#008DB8', '#00AAAA', '#00C69C')

# Sony Playstation
sony_data <- aggregate(dummy_count ~ Genre, data =
url_data_platform[url_data_platform$Platform_General == 'Sony_Playstation', ], sum)
par(mfrow=c(1, 2), mar=c(5, 5, 4, 2))
pie(sony_data$dummy_count,
    labels = sony_data$Genre,
    col = colors,
    main = 'Pie Chart of Genre Distribution of Playstation',
    init.angle = 90,
    radius = 1)

# Microsoft Xbox
xbox_data <- aggregate(dummy_count ~ Genre, data =
url_data_platform[url_data_platform$Platform_General == 'Microsoft_Xbox', ], sum)
pie(xbox_data$dummy_count,
    labels = xbox_data$Genre,
    col = colors,
    main = 'Pie Chart of Genre Distribution of Xbox',
    init.angle = 90,
    radius = 1)

# Nintendo
nintendo_data <- aggregate(dummy_count ~ Genre, data =
url_data_platform[url_data_platform$Platform_General == 'Nintendo', ], sum)
pie(nintendo_data$dummy_count,
    labels = nintendo_data$Genre,
    col = colors,
    main = 'Pie Chart of Genre Distribution of Nintendo',
    init.angle = 90,]    radius = 1)
```

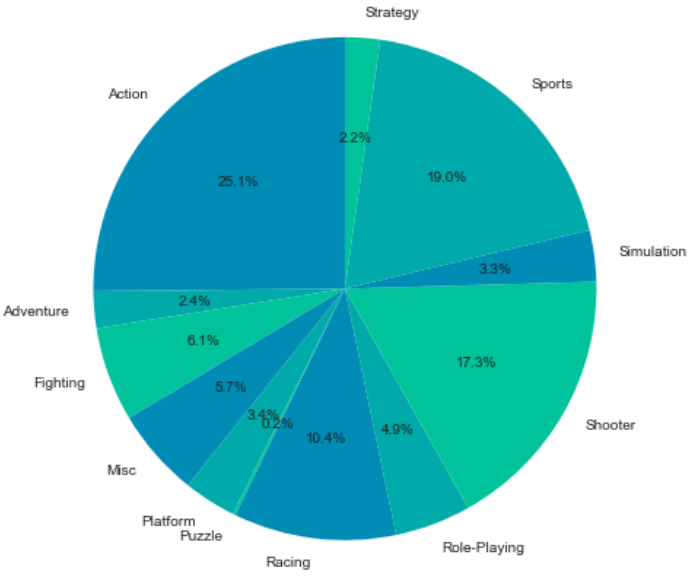
Pie Chart of Genre Distribution of Playstation



Pie Chart of Genre Distribution of Nintendo



Pie Chart of Genre Distribution of Xbox



Conclusion

As a result when compared, Playstation, Xbox and Nintendo, Playstation shows better results in terms of:

- Customer satisfaction
- Amount of games released for that platform
- Total sales amount
- Sales amount per game