Exploratory Data Analysis - VIDEO GAME SALES

Yipeng CHEN

Contents

Here, I just combined two excellent Kaggle Notebooks of
Exploratory Data Analysis (EDA) on video games sales.

Description

The Original Dataset - Video Game Sales

This dataset contains a list of video games with sales greater than 100 000 copies.

It was generated by a scrape of VGChartz.

Fields include :
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.

Many thanks for the original notebooks :

EDA - VIDEO GAME SALES

Video Games Sales Analysis And Visualization

1.EDA - VIDEO GAME SALES using R

The data used in the first part ("EDA - VIDEO GAME SALES using R")
contains information only from 1980 to 2016.

1.1 Libraries & Data loading

1.1 A. Database Loading

```r
# Loading the database
data <- read.csv("vgsales.csv", stringsAsFactors = FALSE)

# Removing the Rank column
data$Rank <- NULL

# Filtering only the records of interest for this study,
# removing the records with Year = NaN and records with the year above 2016
data <- data[data$Year != "N/A" & data$Year != "2017"
             & data$Year != "2020", ]
data$Year <- factor(data$Year)
```

Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37

```r
# Viewing the first 5 DataFrame records
pacman::p_load(knitr,kableExtra)
kable(head(data, 5)) %>% kable_styling(font_size = 7)
```

1.1 B. Summary of data

```r
summary(data)
```

```
##      Name             Platform              Year          Genre
##  Length:16323       Length:16323       2009   :1431   Length:16323
##  Class :character   Class :character   2008   :1428   Class :character
##  Mode  :character   Mode  :character   2010   :1259   Mode  :character
##                                        2007   :1202
##                                        2011   :1139
##                                        2006   :1008
##                                        (Other):8856
##   Publisher            NA_Sales          EU_Sales          JP_Sales
##  Length:16323       Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.00000
##  Class :character   1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.00000
##  Mode  :character   Median : 0.0800   Median : 0.0200   Median : 0.00000
##                     Mean   : 0.2655   Mean   : 0.1476   Mean   : 0.07868
##                     3rd Qu.: 0.2400   3rd Qu.: 0.1100   3rd Qu.: 0.04000
##                     Max.   :41.4900   Max.   :29.0200   Max.   :10.22000
##
##   Other_Sales        Global_Sales
##  Min.   : 0.00000   Min.   : 0.0100
##  1st Qu.: 0.00000   1st Qu.: 0.0600
##  Median : 0.01000   Median : 0.1700
##  Mean   : 0.04834   Mean   : 0.5403
##  3rd Qu.: 0.04000   3rd Qu.: 0.4800
##  Max.   :10.57000   Max.   :82.7400
##
```

1.2 Descriptive Analysis

1.2 A. Frequency Distribution

Year => Year of the game's release

```
freq_year <- data.frame(cbind(Frequency = table(data$Year),
                  Percent = prop.table(table(data$Year)) * 100))
freq_year <- freq_year[order(freq_year$Frequency, decreasing=TRUE), ]
```

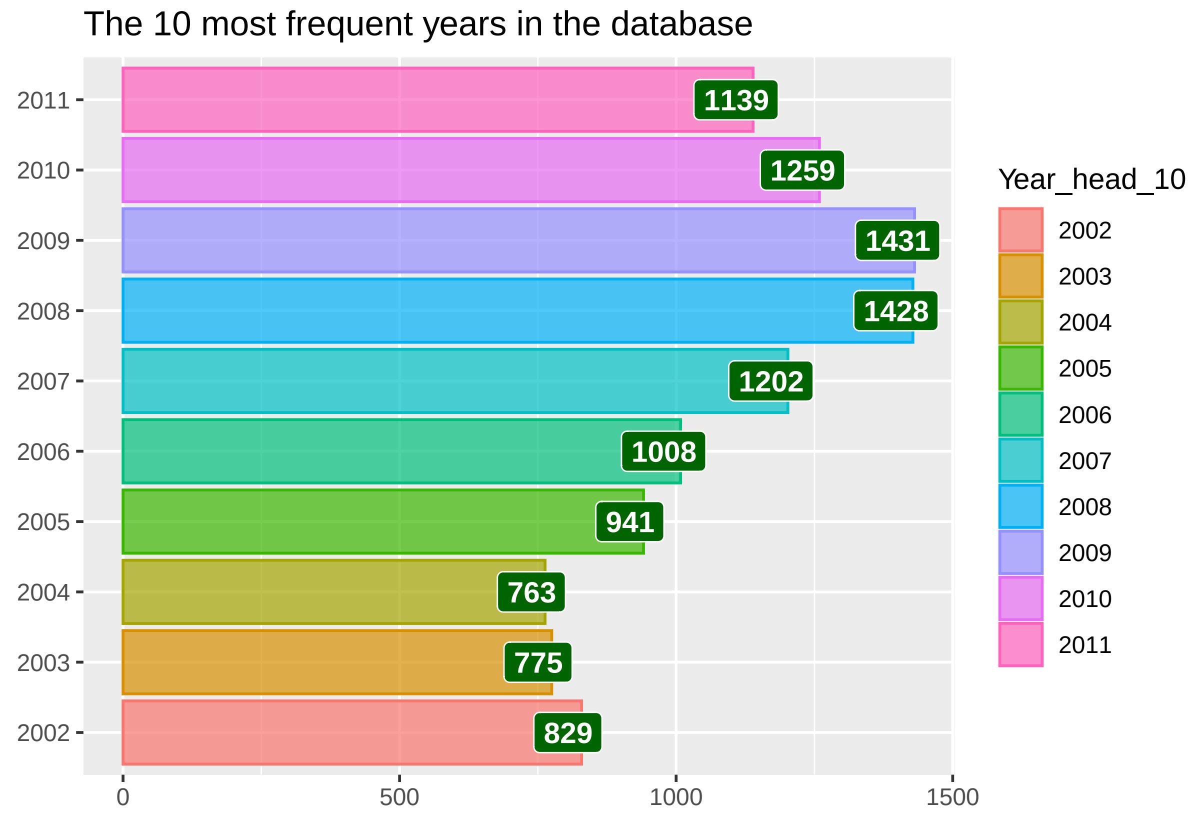
```
kable(head(freq_year, 10))
```

	Frequency	Percent
2009	1431	8.766771
2008	1428	8.748392
2010	1259	7.713043
2007	1202	7.363842
2011	1139	6.977884
2006	1008	6.175335
2005	941	5.764872
2002	829	5.078723
2003	775	4.747902
2004	763	4.674386

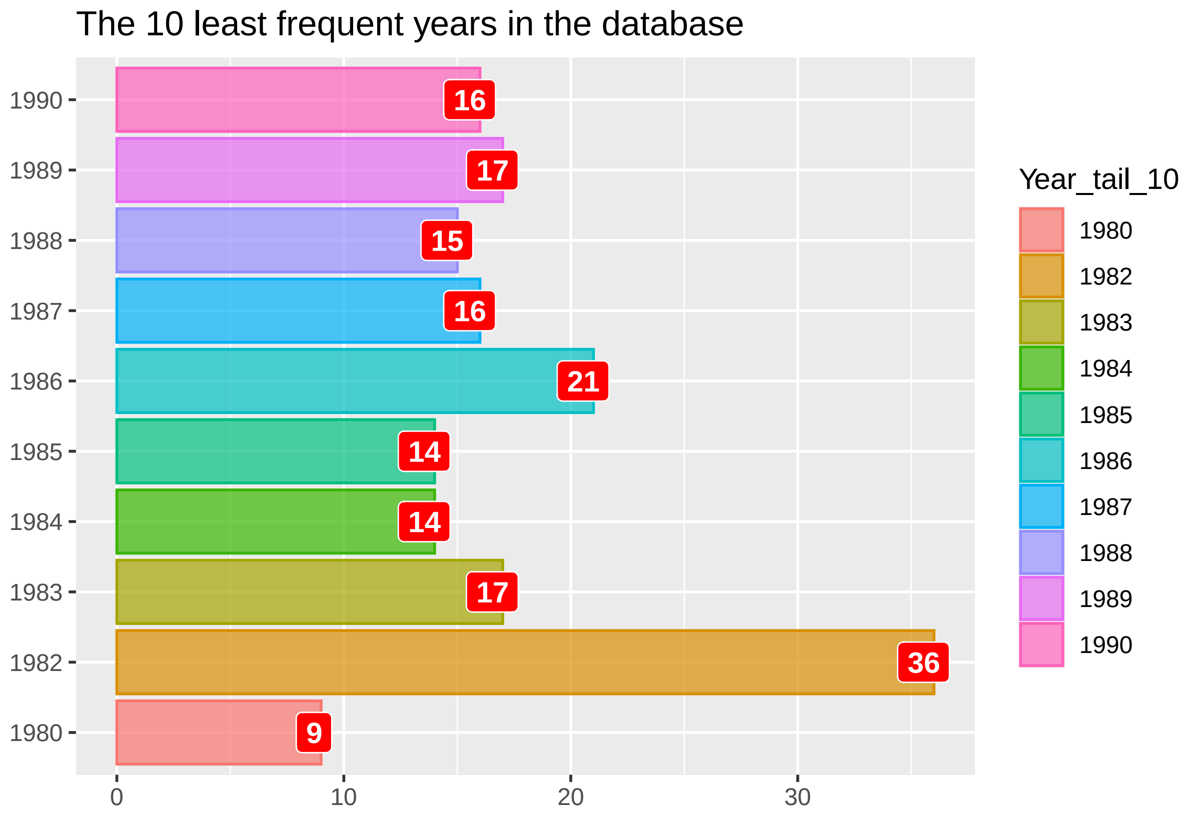
```
kable(tail(freq_year, 10))
```

	Frequency	Percent
1982	36	0.2205477
1986	21	0.1286528
1983	17	0.1041475
1989	17	0.1041475
1987	16	0.0980212
1990	16	0.0980212
1988	15	0.0918949
1984	14	0.0857685
1985	14	0.0857685
1980	9	0.0551369

```
# Visualizations
pacman::p_load(hrbrthemes,gganimate,gapminder,babynames,ggthemes,cowplot,ggplot2)

df <- head(freq_year, 10)

Year_head_10 = row.names(df)

ggplot(data = df, mapping = aes(x = Frequency, y = Year_head_10)) +
         geom_bar(stat = "identity", mapping = aes(fill = Year_head_10,
                                      color = Year_head_10), alpha = .7) +
         geom_label(mapping = aes(label=Frequency), fill = "#006400",
                    color = "white", fontface = "bold", hjust=.7) +
         ggtitle("The 10 most frequent years in the database") +
         xlab(" ") +
         ylab("")
```

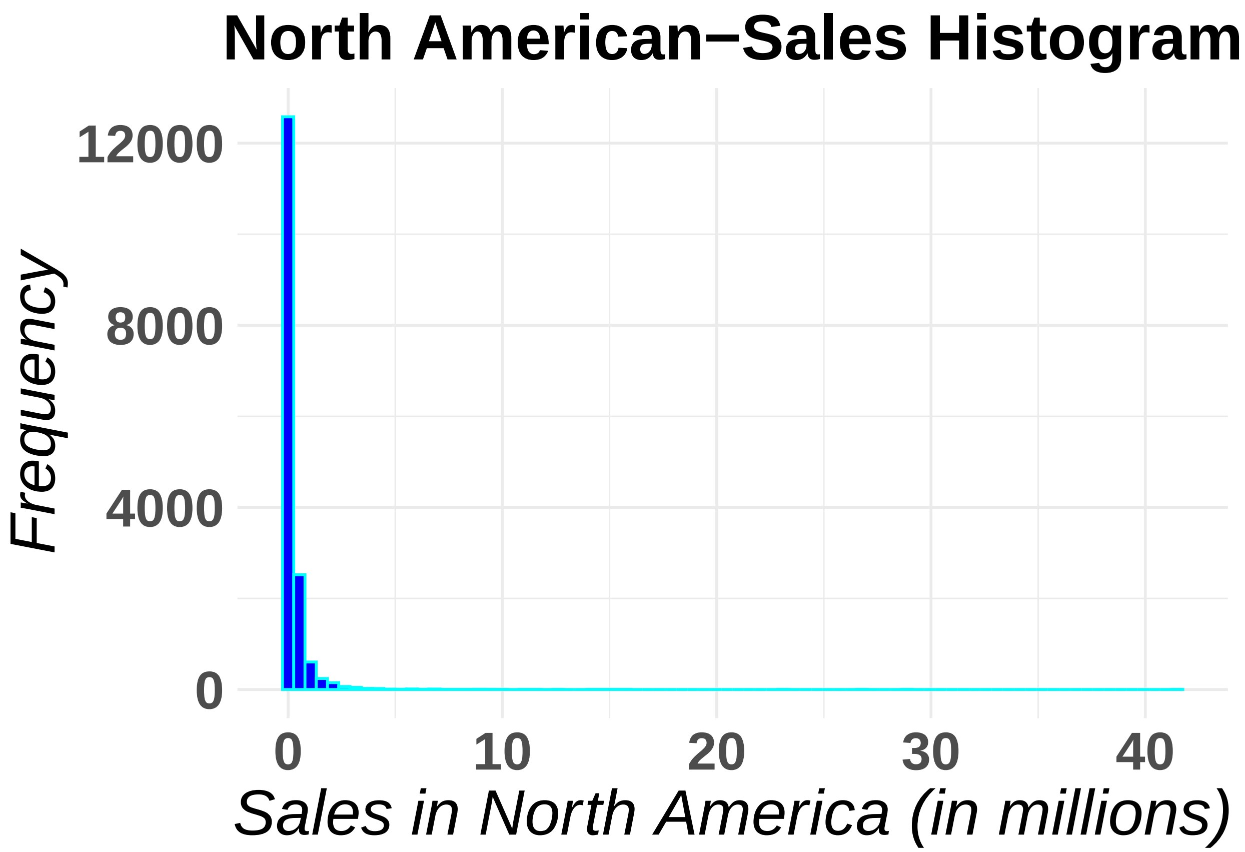


```
# Visualizations
pacman::p_load(hrbrthemes,gganimate,gapminder,babynames,ggthemes,cowplot,ggplot2)

df1 <- tail(freq_year, 10)

Year_tail_10 = row.names(df1)

ggplot(data = df1, mapping = aes(x = Frequency, y = Year_tail_10)) +
        geom_bar(stat = "identity", mapping = aes(fill = Year_tail_10,
                                      color = Year_tail_10), alpha = .7) +
        geom_label(mapping = aes(label=Frequency), fill = "red",
                   color = "white", fontface = "bold", hjust=.7) +
        ggtitle("The 10 least frequent years in the database") +
        xlab(" ") +
        ylab("")
```

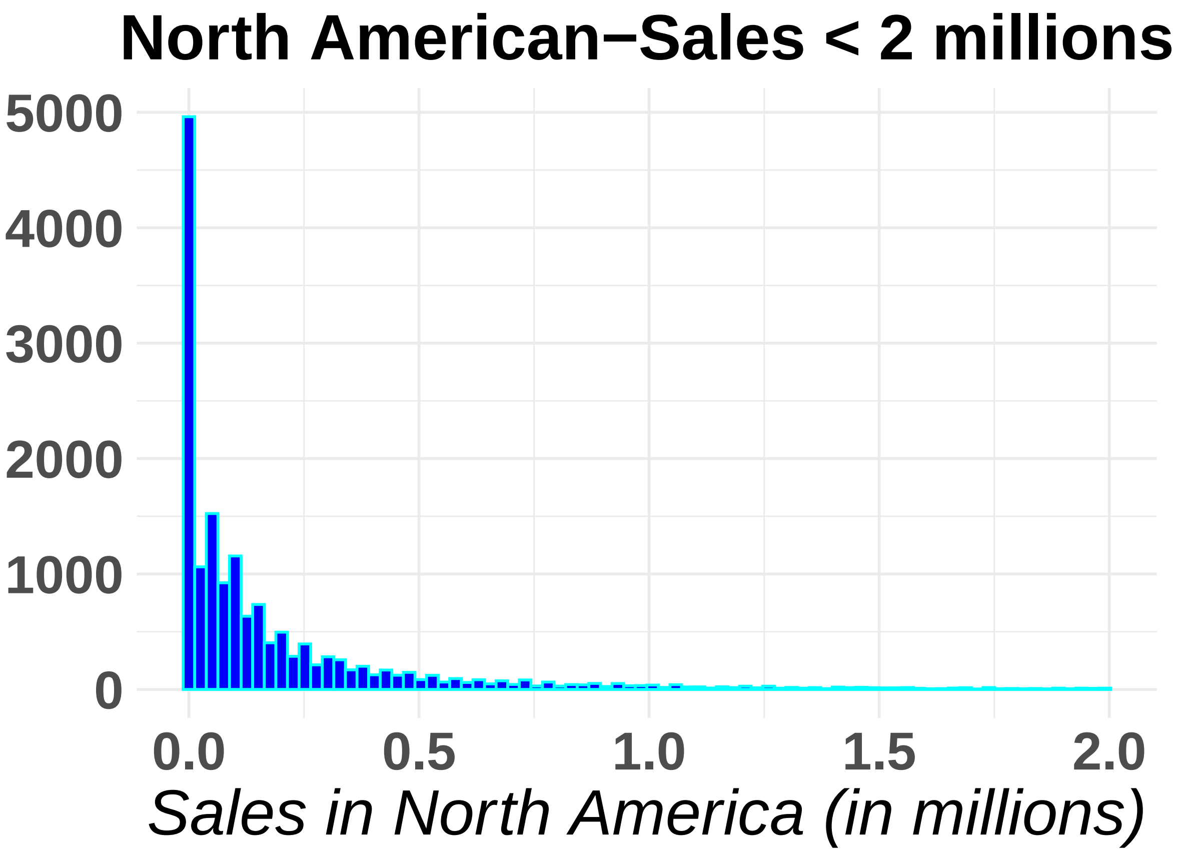


NA_Sales => Sales in North America (in millions)

```
ggplot(data = data, mapping = aes(x = NA_Sales)) +
         geom_histogram(bins = 80, fill = "blue", color = "cyan") +
         xlab("Sales in North America (in millions)") +
         ylab("Frequency") +
         ggtitle("North American-Sales Histogram") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

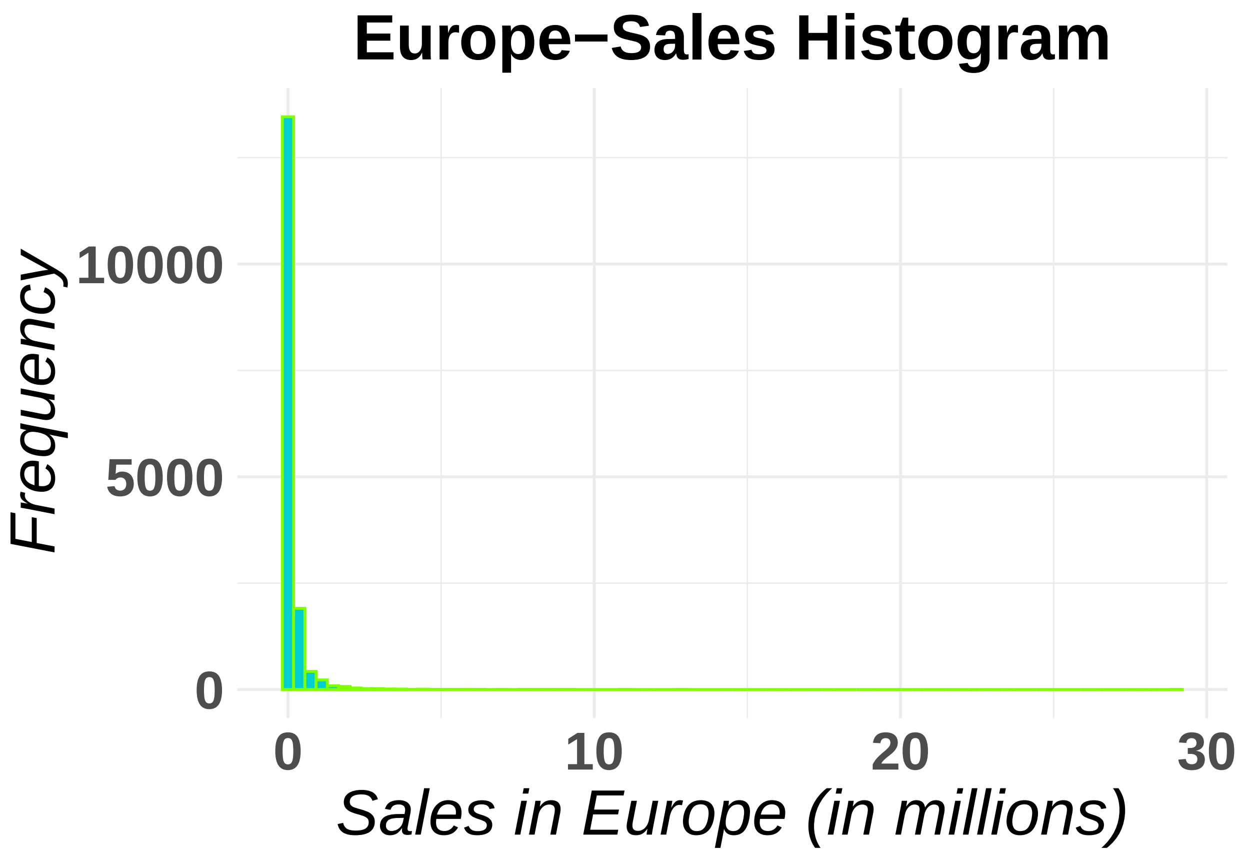


```
df2 <- data[data$NA_Sales < 2, ]
ggplot(data = df2, mapping = aes(x = NA_Sales)) +
         geom_histogram(bins = 80, fill = "blue", color = "cyan") +
         xlab("Sales in North America (in millions)") +
         ylab("") +
         ggtitle("North American-Sales < 2 millions") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

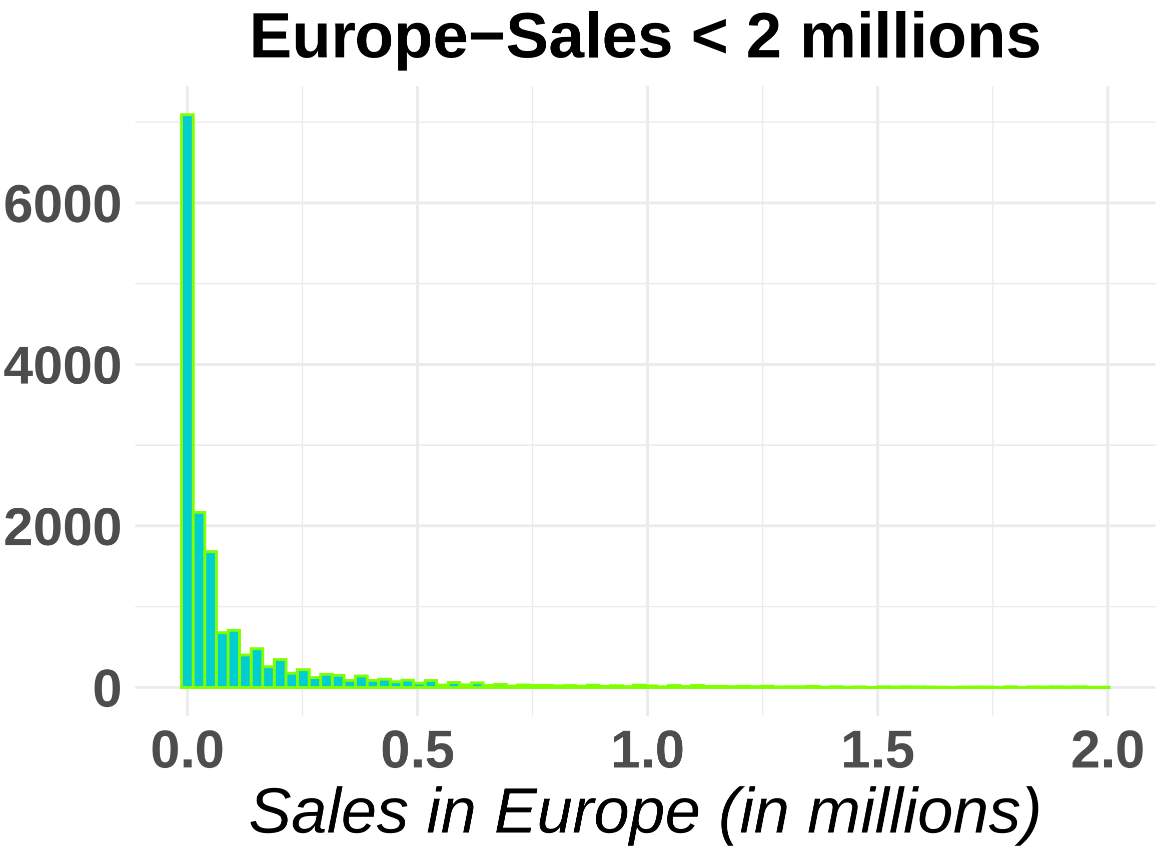


EU_Sales => Sales in Europe (in millions)

```
ggplot(data = data, mapping = aes(x = EU_Sales)) +
         geom_histogram(bins = 80, fill = "#00CED1", color = "#7FFF00") +
         xlab("Sales in Europe (in millions)") +
         ylab("Frequency") +
         ggtitle("Europe-Sales Histogram") +
         theme_minimal() +
         theme(plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

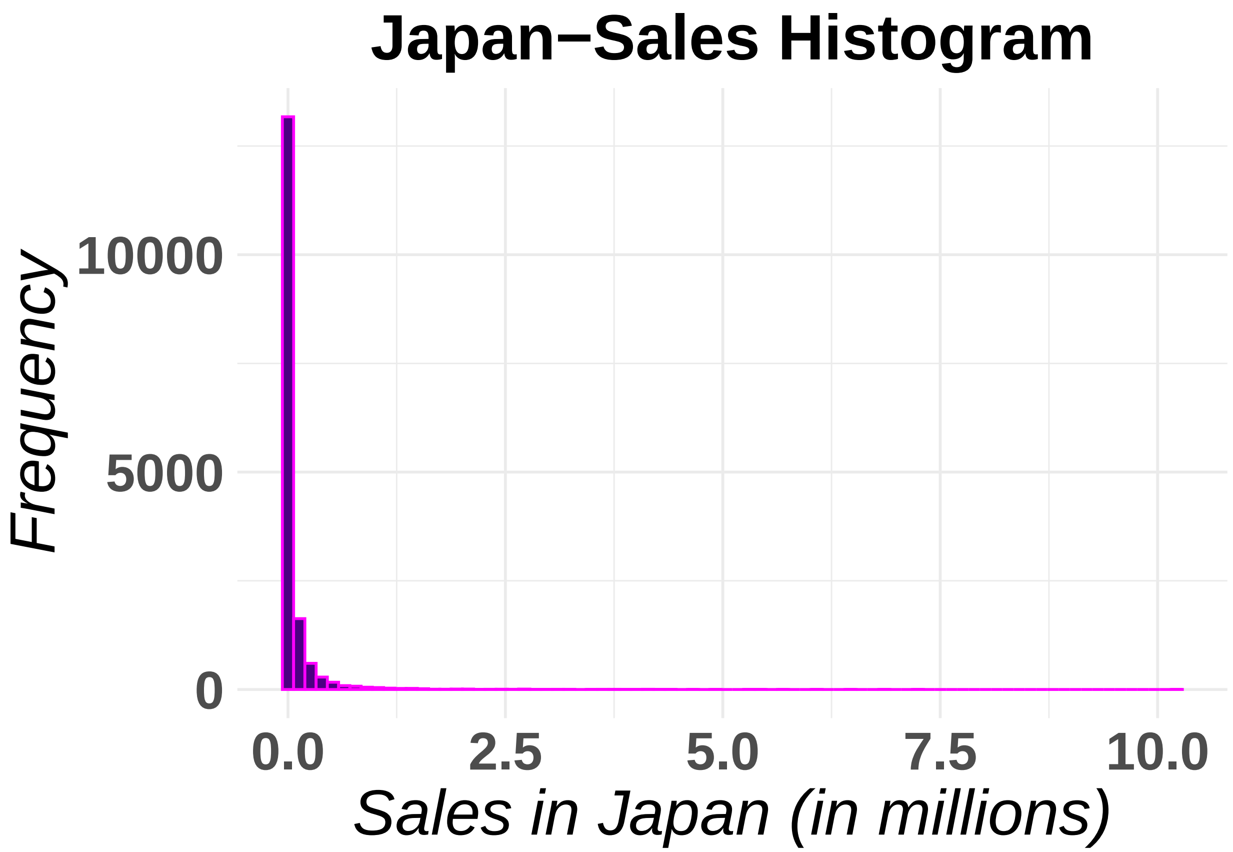


```
df3 <- data[data$EU_Sales < 2, ]
ggplot(data = df3, mapping = aes(x = EU_Sales)) +
         geom_histogram(bins = 80, fill = "#00CED1", color = "#7FFF00") +
         xlab("Sales in Europe (in millions)") +
         ylab("") +
         ggtitle("Europe-Sales < 2 millions") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

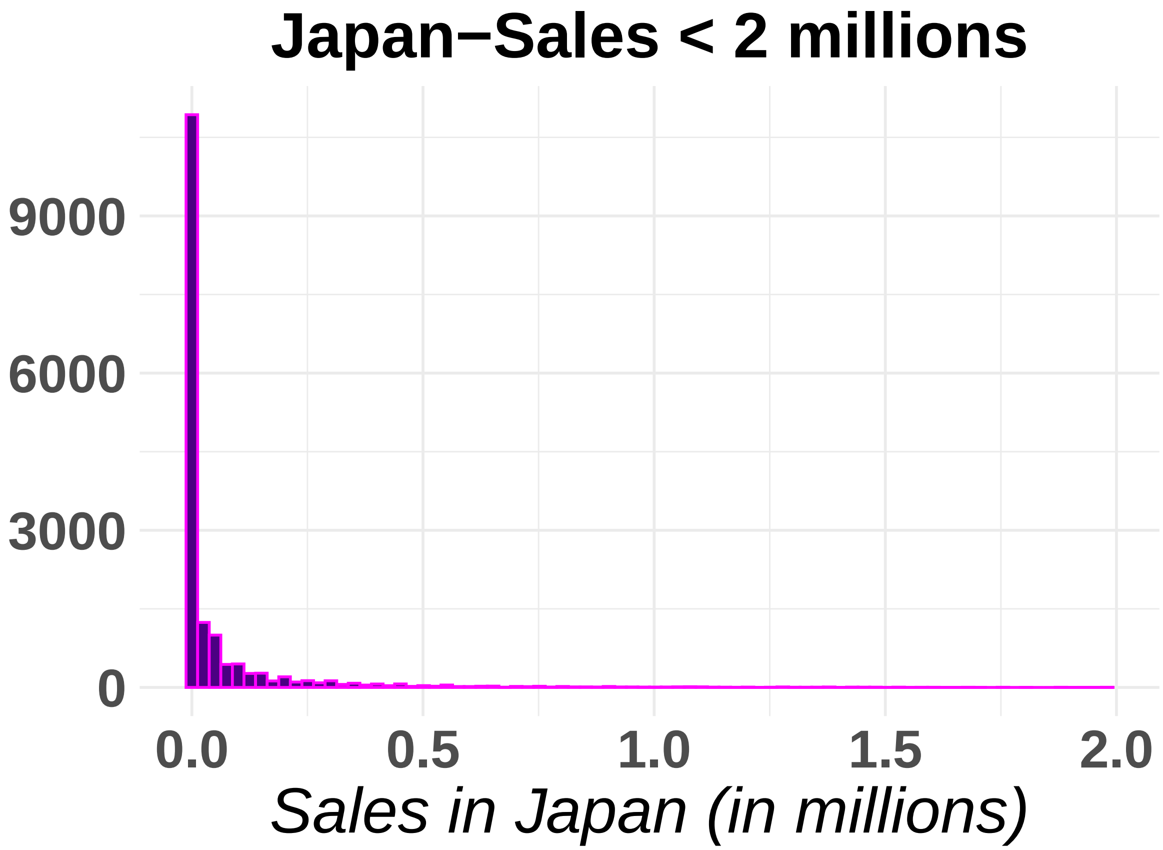


JP_Sales => Sales in Japan (in millions)

```r
ggplot(data = data, mapping = aes(x = JP_Sales)) +
         geom_histogram(bins = 80, fill = "#4B0082", color = "#FF00FF") +
         xlab("Sales in Japan (in millions)") +
         ylab("Frequency") +
         ggtitle("Japan-Sales Histogram") +
         theme_minimal() +
         theme(plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

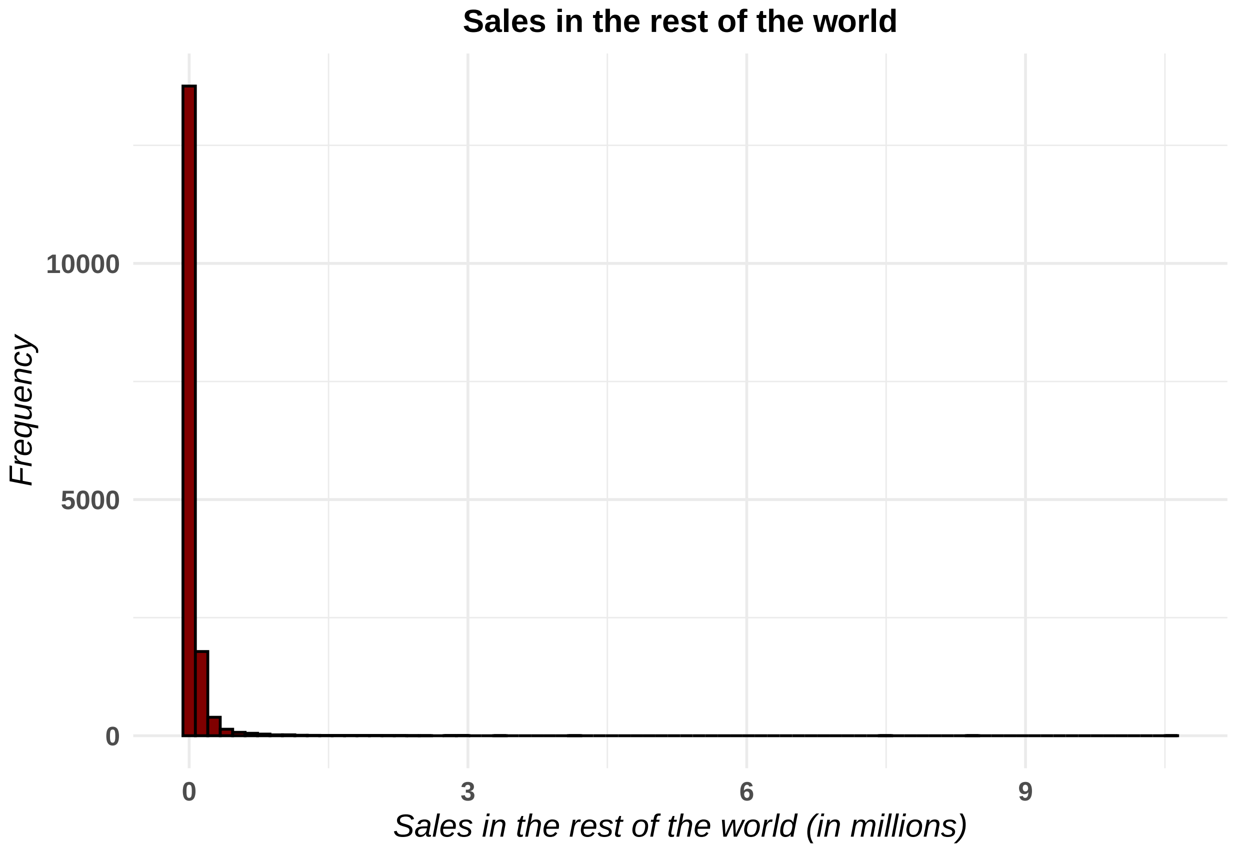


```r
df4 <- data[data$JP_Sales < 2, ]
ggplot(data = df4, mapping = aes(x = JP_Sales)) +
         geom_histogram(bins = 80, fill = "#4B0082", color = "#FF00FF") +
         xlab("Sales in Japan (in millions)") +
         ylab("") +
         ggtitle("Japan-Sales < 2 millions") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 20, face = "bold"),
               legend.position = "none")
```

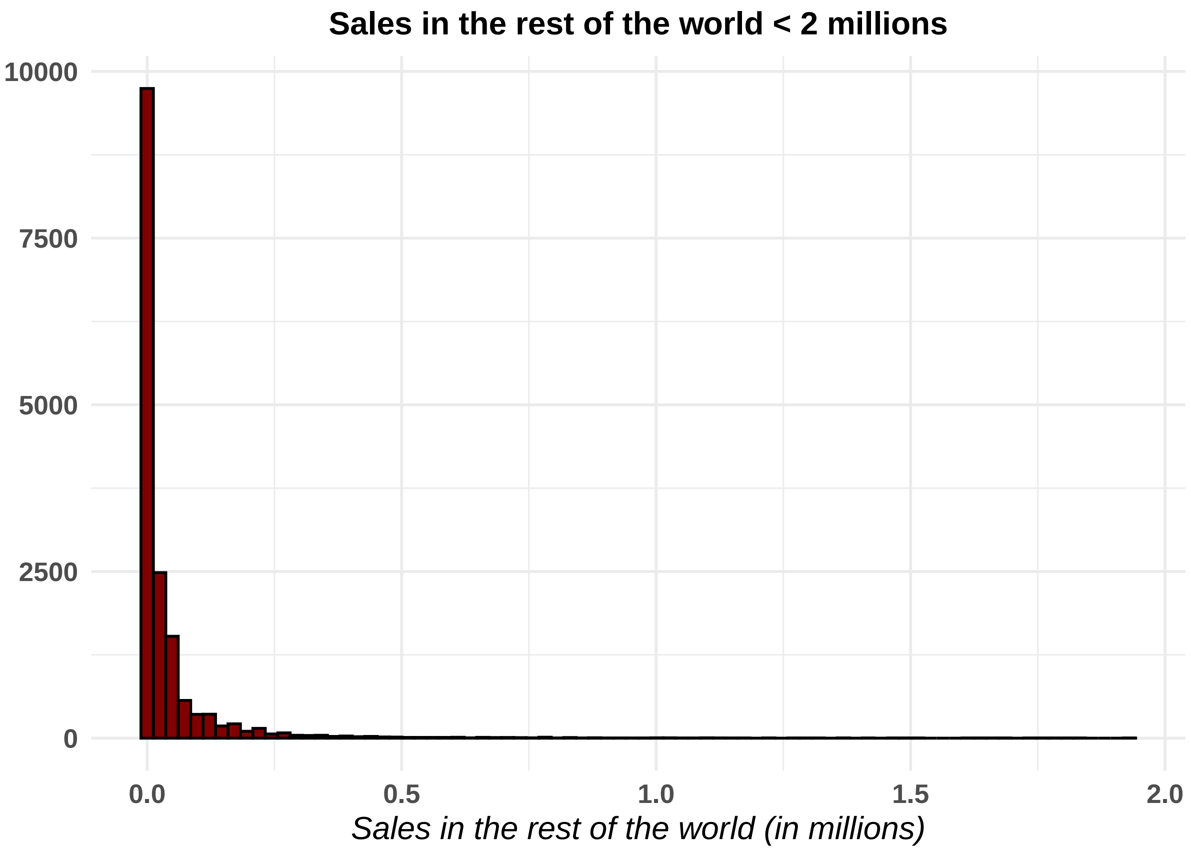


Other_Sales => Sales in the rest of the world (in millions)

```
ggplot(data = data, mapping = aes(x = Other_Sales)) +
         geom_histogram(bins = 80, fill = "#800000", color = "black") +
         xlab("Sales in the rest of the world (in millions)") +
         ylab("Frequency") +
         ggtitle("Sales in the rest of the world") +
         theme_minimal() +
         theme(plot.title = element_text(size = 12, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 12, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 12, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```

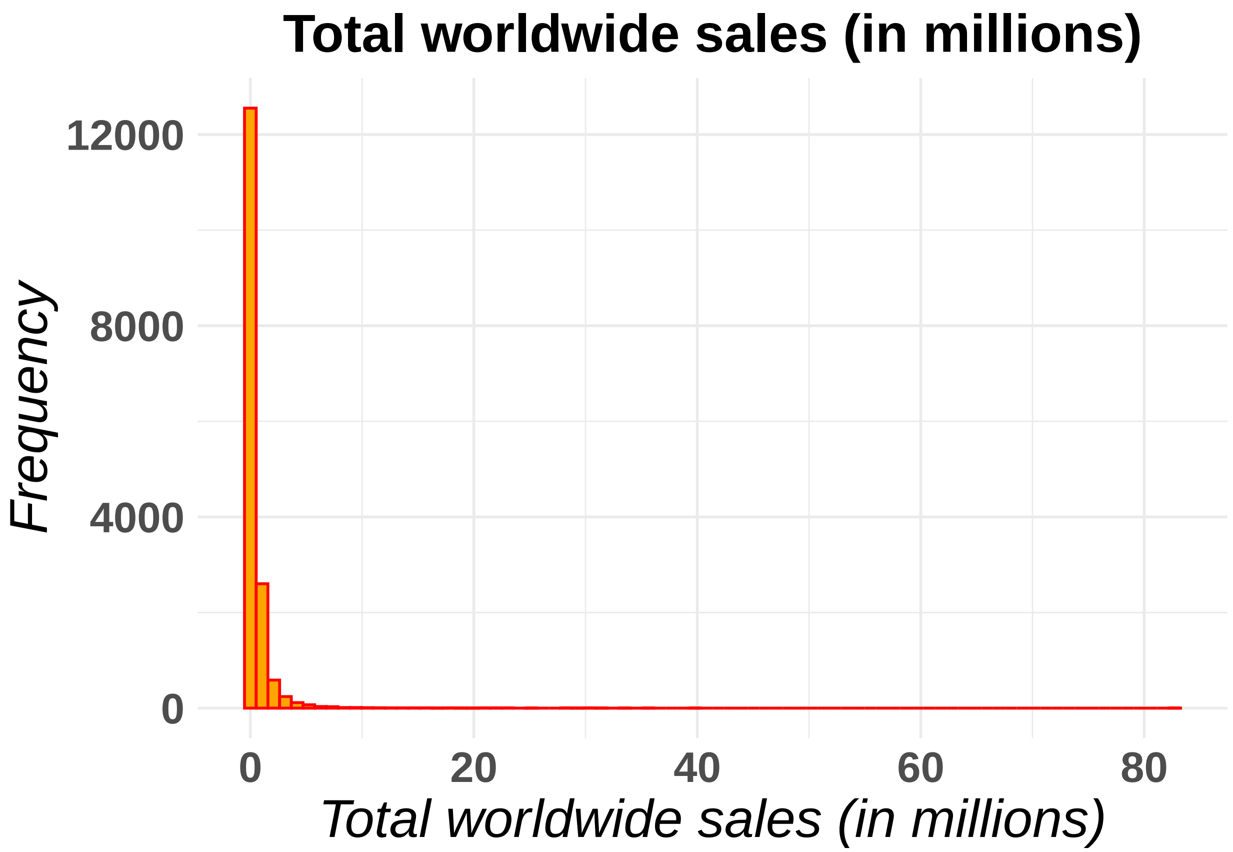


```r
df5 <- data[data$Other_Sales < 2, ]
ggplot(data = df5, mapping = aes(x = Other_Sales)) +
         geom_histogram(bins = 80, fill = "#800000", color = "black") +
         xlab("Sales in the rest of the world (in millions)") +
         ylab("") +
         ggtitle("Sales in the rest of the world < 2 millions") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 12, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 12, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 12, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```

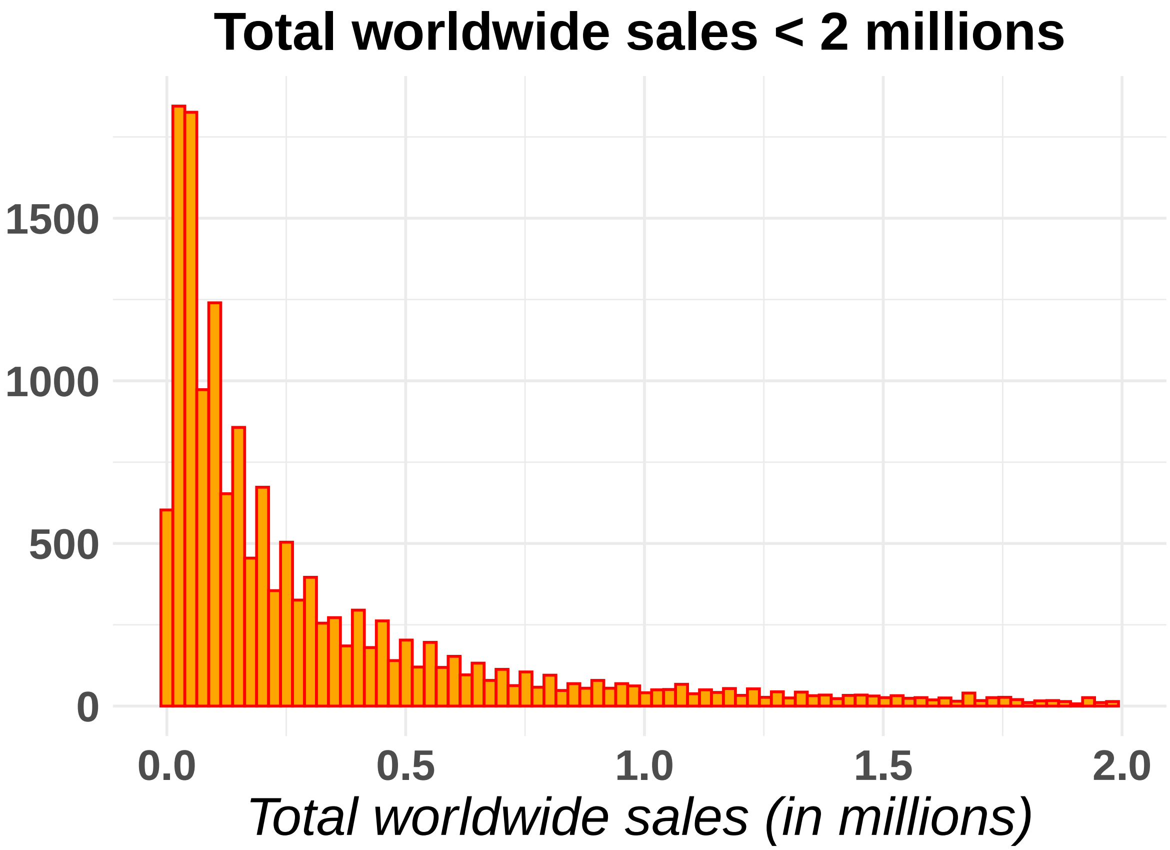


Global_Sales => Total worldwide sales

```
ggplot(data = data, mapping = aes(x = Global_Sales)) +
         geom_histogram(bins = 80, fill = "orange", color = "#FF0000") +
         xlab("Total worldwide sales (in millions)") +
         ylab("Frequency") +
         ggtitle("Total worldwide sales (in millions)") +
         theme_minimal() +
         theme(plot.title = element_text(size = 20, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 20, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 20, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 16, face = "bold"),
               axis.text.y = element_text(size = 16, face = "bold"),
               legend.position = "none")
```



```
df6 <- data[data$Global_Sales < 2, ]
ggplot(data = df6, mapping = aes(x = Global_Sales)) +
         geom_histogram(bins = 80, fill = "orange", color = "#FF0000") +
         xlab("Total worldwide sales (in millions)") +
         ylab("") +
         ggtitle("Total worldwide sales < 2 millions") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 20, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 20, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 20, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 16, face = "bold"),
               axis.text.y = element_text(size = 16, face = "bold"),
               legend.position = "none")
```



With these graphical analysis,
we can overlook that the sales of video games is mostly below 2 million dollars.

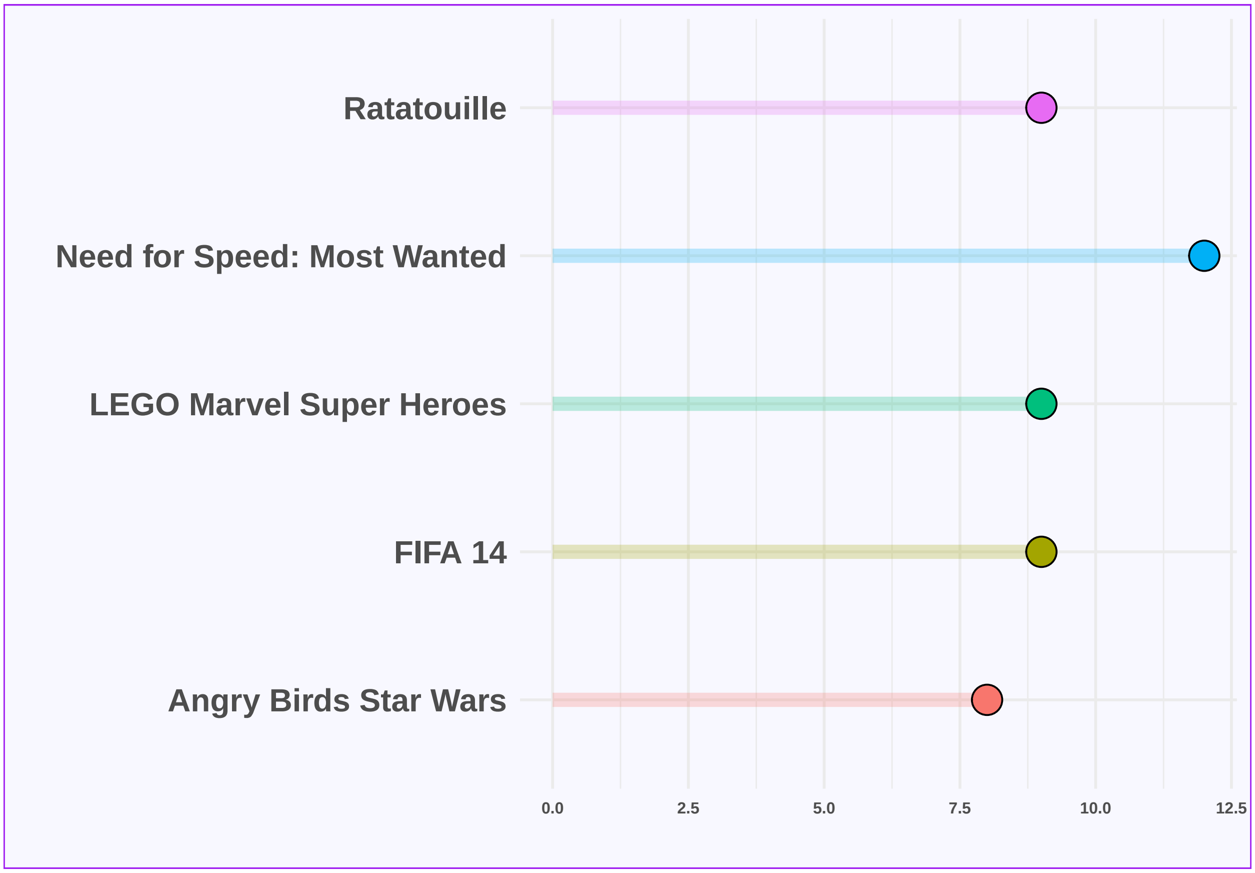
Name -> The games name

The 5 most frequent games in the database

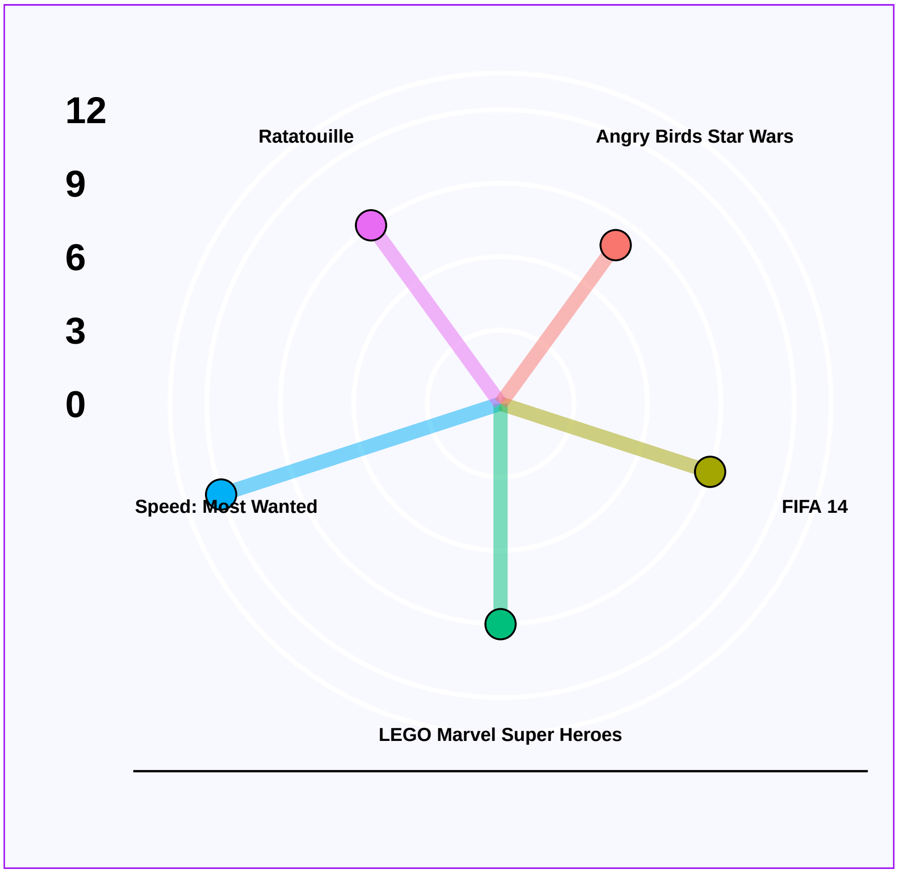
```
freq_name <- data.frame(cbind(Frequency = table(data$Name),
                              Percent = prop.table(table(data$Name)) * 100))
freq_name <- head(freq_name[order(freq_name$Frequency, decreasing = T), ], 5)
kable(freq_name)
```

	Frequency	Percent
Need for Speed: Most Wanted	12	0.0735159
FIFA 14	9	0.0551369
LEGO Marvel Super Heroes	9	0.0551369
Ratatouille	9	0.0551369
Angry Birds Star Wars	8	0.0490106

```
ggplot(data = freq_name, mapping = aes(x = row.names(freq_name), y = Frequency)) +
         geom_segment(aes(xend=row.names(freq_name), yend=0,
                          color = row.names(freq_name)),
                      linewidth = 2.5, alpha = .25) +
         geom_point(mapping = aes(fill = row.names(freq_name)),
                    size = 5, shape = 21) +
         coord_flip() +
         theme_minimal() +
         xlab("") +
         ylab("") +
         theme(plot.background = element_rect(fill = "#F8F8FF", color = "purple"),
               axis.title.x = element_text(size = 12, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 12, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 6, face = "bold"),
               axis.text.y = element_text(size = 12, face = "bold"),
               legend.position = "none")
```



```
ggplot(data = freq_name, mapping = aes(x = row.names(freq_name), y = Frequency)) +
         geom_segment(aes(xend=row.names(freq_name), yend=0,
                          color = row.names(freq_name)),
                      linewidth = 2.5, alpha = .5) +
         geom_point(mapping = aes(fill = row.names(freq_name)),
                    size = 5, shape = 21) +
         theme_economist() +
         xlab("") +
         ylab("") +
         coord_polar() +
         theme(plot.background = element_rect(fill = "#F8F8FF", color = "purple"),
               axis.title.x = element_text(size = 14, face = "italic"),
               axis.title.y = element_text(size = 14, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 7, face = "bold"),
               axis.text.y = element_text(size = 14, face = "bold"),
               legend.position = "none")
```



Platform -> Platform of the games release (i.e. PC, PS4, etc.)

```r
kable(unique(data$Platform),col.names ='Platform')
```

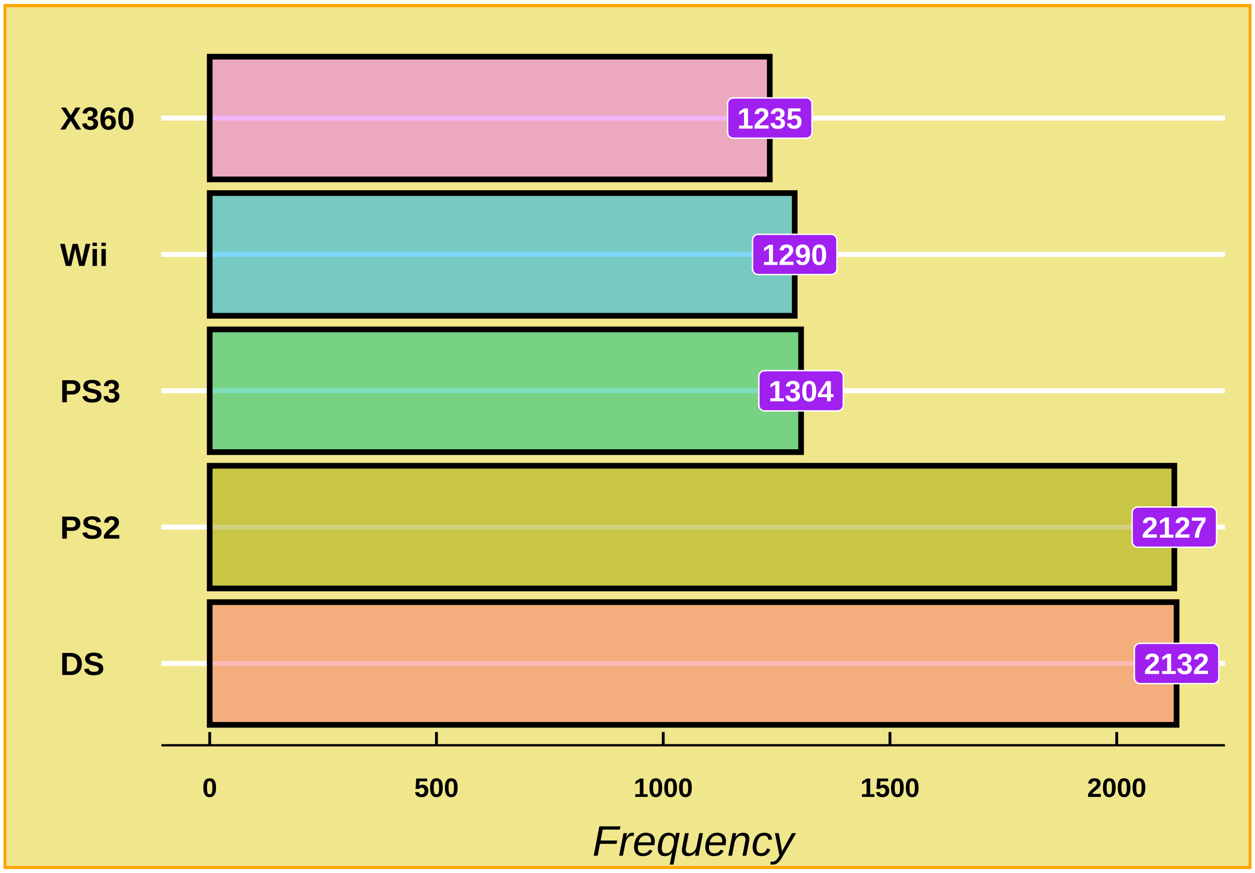
Platform
Wii
NES
GB
DS
X360
PS3
PS2
SNES
GBA
3DS
PS4
N64
PS
XB
PC
2600
PSP
XOne
GC
WiiU
GEN
DC
PSV
SAT
SCD
WS
NG
TG16
3DO
GG
PCFX

The 5 most frequent gaming platforms in the database

```
freq_platform <- data.frame(cbind(Frequency = table(data$Platform),
                                  Percent = prop.table(table(data$Platform)) * 100))
freq_platform <- head(freq_platform[order(freq_platform$Frequency,
                                          decreasing = T), ], 5)
kable(freq_platform)
```

	Frequency	Percent
DS	2132	13.061324
PS2	2127	13.030693
PS3	1304	7.988728
Wii	1290	7.902959
X360	1235	7.566011

```
ggplot(data = freq_platform, mapping = aes(x = row.names(freq_platform),
                                           y = Frequency)) +
         geom_bar(stat = "identity", aes(fill = row.names(freq_platform)),
                  linewidth = 1, alpha = .5, color = "black") +
         geom_label(mapping = aes(label = Frequency), fill = "purple",
                    color = "white", size = 4, fontface = "bold") +
         coord_flip() +
         theme_economist() +
         ylab("Frequency") +
         xlab("") +
         theme(plot.background = element_rect(fill = "#F0E68C",
                                              color = "orange", linewidth = 1),
               axis.title.y = element_text(size = 12, hjust = .5, face = "italic"),
               axis.title.x = element_text(size = 16, hjust = .5,
                                           vjust = -2, face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold"),
               axis.text.y = element_text(size = 12, face = "bold"),
               legend.position = "none")
```

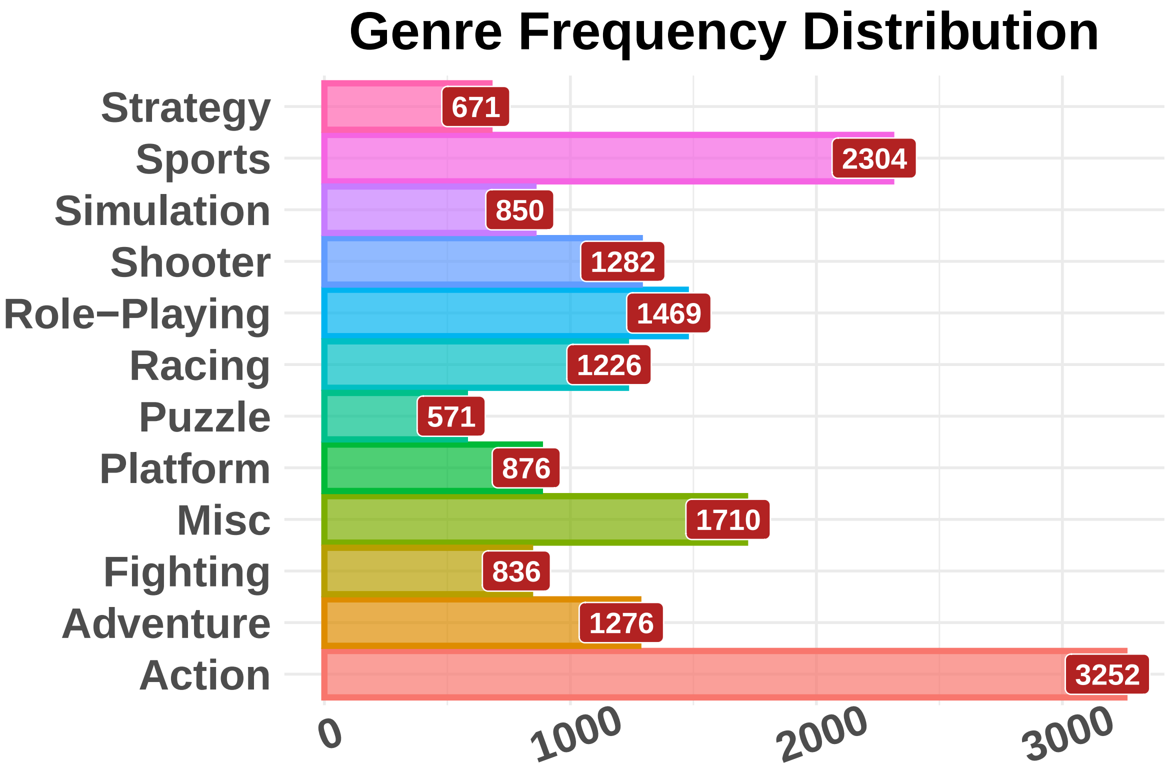


Genre -> Genre of the game

```
freq_genre <- data.frame(cbind(Frequency = table(data$Genre),
                               Percent = prop.table(table(data$Genre)) * 100))
freq_genre <- freq_genre[order(freq_genre$Frequency, decreasing = T), ]
kable(freq_genre)
```

	Frequency	Percent
Action	3252	19.922808
Sports	2304	14.115052
Misc	1710	10.476015
Role-Playing	1469	8.999571
Shooter	1282	7.853948
Adventure	1276	7.817190
Racing	1226	7.510874
Platform	876	5.366661
Simulation	850	5.207376
Fighting	836	5.121607
Strategy	671	4.110764
Puzzle	571	3.498131

```r
ggplot(data = freq_genre, mapping = aes(x = Frequency, y = row.names(freq_genre))) +
         geom_bar(stat = "identity", mapping = aes(fill = row.names(freq_genre),
                                                    color = row.names(freq_genre)),
                                                    alpha = .7, linewidth = 1.1) +
         geom_label(mapping = aes(label=Frequency), fill = "#B22222", size = 4,
                    color = "white", fontface = "bold", hjust=.7) +
         ggtitle("Genre Frequency Distribution") +
         xlab(" ") +
         ylab("") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 20, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 20, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 20, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 16, face = "bold", angle = 20),
               axis.text.y = element_text(size = 16, face = "bold"),
               legend.position = "none")
```



Publisher -> Publisher of the game

```
kable(head(unique(data$Publisher), 25),col.names ='Publisher')
```

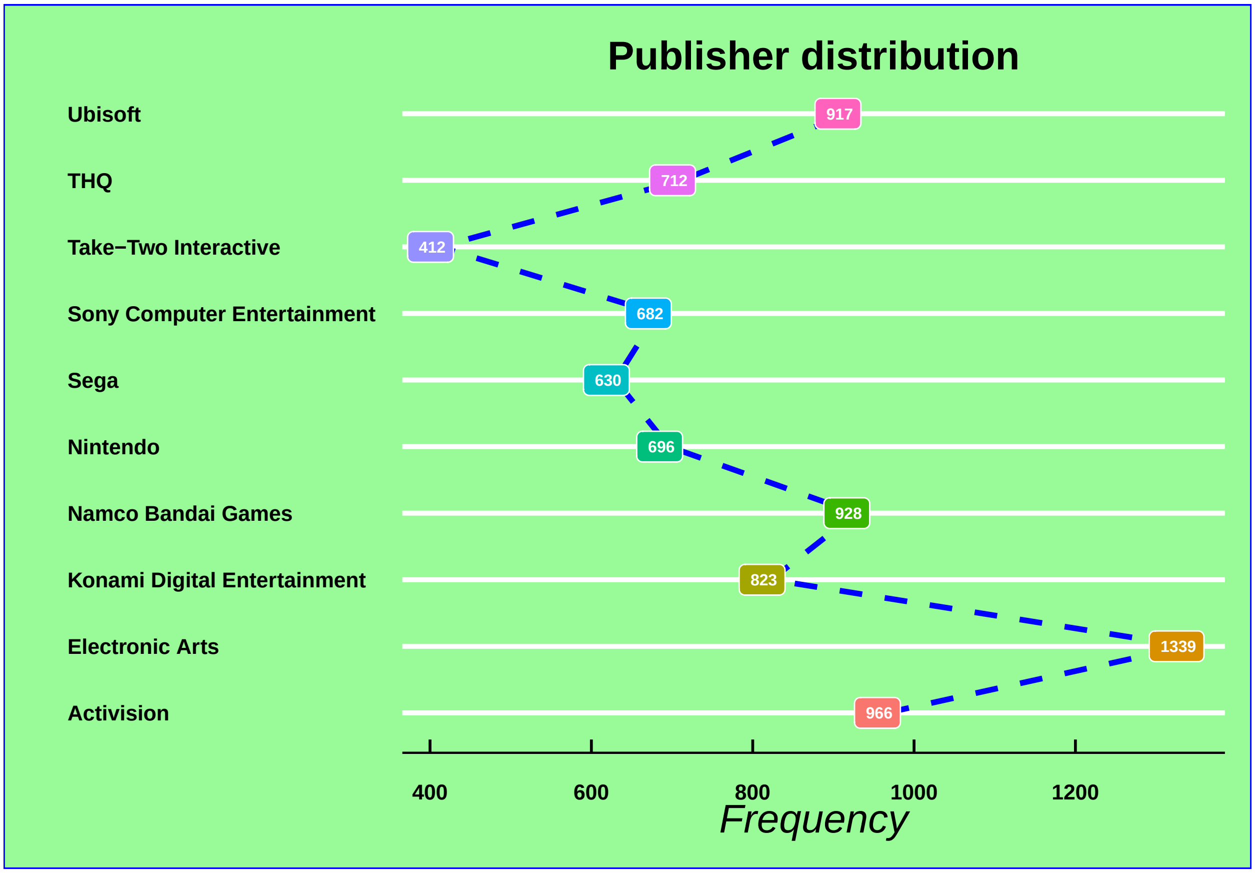
Publisher
Nintendo
Microsoft Game Studios
Take-Two Interactive
Sony Computer Entertainment
Activision
Ubisoft
Bethesda Softworks
Electronic Arts
Sega
SquareSoft
Atari
505 Games
Capcom
GT Interactive
Konami Digital Entertainment
Sony Computer Entertainment Europe
Square Enix
LucasArts
Virgin Interactive
Warner Bros. Interactive Entertainment
Universal Interactive
Eidos Interactive
RedOctane
Vivendi Games
Enix Corporation

The 10 most frequent Publisher in the database

```
freq_published <- data.frame(cbind(Frequency = table(data$Publishe),
                                   Percent = prop.table(table(data$Publishe)) * 100))
freq_published <- head(freq_published[order(freq_published$Frequency,
                                            decreasing = T), ], 10)
kable(freq_published)
```

	Frequency	Percent
Electronic Arts	1339	8.203149
Activision	966	5.918030
Namco Bandai Games	928	5.685229
Ubisoft	917	5.617840
Konami Digital Entertainment	823	5.041965
THQ	712	4.361943
Nintendo	696	4.263922
Sony Computer Entertainment	682	4.178153
Sega	630	3.859585
Take-Two Interactive	412	2.524046

```
ggplot(data = freq_published, mapping = aes(x = Frequency,
                                            y = row.names(freq_published))) +
         geom_line(group = 1, linewidth = 1, color = "blue",
                   linetype = "dashed") +
         geom_label(mapping = aes(label=Frequency,
                                  fill = row.names(freq_published)),
                    size = 2.25, color = "white", fontface = "bold", hjust=.7) +
         ggtitle("Publisher distribution") +
         xlab("Frequency") +
         ylab("") +
         theme_economist() +
         theme(plot.background = element_rect(fill = "#98FB98", color = "blue"),
               plot.title = element_text(size = 15, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 15, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 15, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 8, face = "bold"),
               axis.text.y = element_text(size = 8, face = "bold"),
               legend.position = "none")
```



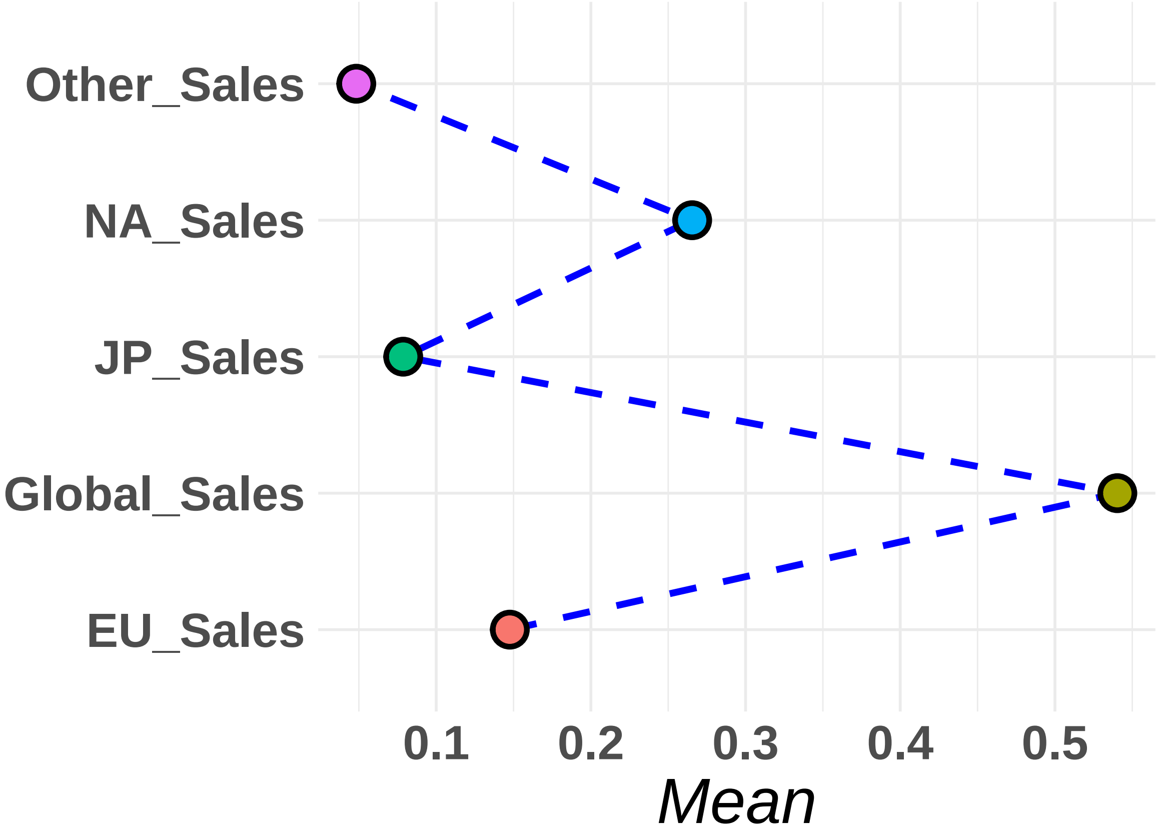
1.2 B. Central Trend Measures

Mean

```
df_means <- data.frame(Mean = c(mean(data$NA_Sales),
                                mean(data$EU_Sales),
                                mean(data$JP_Sales),
                                mean(data$Other_Sales),
                                mean(data$Global_Sales)))
row.names(df_means) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                         "Other_Sales", "Global_Sales")
kable(df_means)
```

	Mean
NA_Sales	0.2654635
EU_Sales	0.1475905
JP_Sales	0.0786773
Other_Sales	0.0483361
Global_Sales	0.5403431

```
ggplot(data = df_means, mapping = aes(x = Mean, y = row.names(df_means))) +
         geom_line(group = 1, linewidth = 1.2,
                   linetype = "dashed", color = "blue") +
         geom_point(size = 5, shape = 21, stroke = 1.5,
                    mapping = aes(fill = row.names(df_means))) +
         theme_minimal() +
         ylab("") +
         theme(plot.title = element_text(size = 24, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 24, hjust = .5, face = "italic"),
               axis.title.y = element_text(size = 24, hjust = .5, face = "italic"),
               axis.text.x = element_text(size = 18, face = "bold"),
               axis.text.y = element_text(size = 18, face = "bold"),
               legend.position = "none")
```



Median

```r
df_median <- data.frame(Median = c(median(data$NA_Sales),
                                   median(data$EU_Sales),
                                   median(data$JP_Sales),
                                   median(data$Other_Sales),
                                   median(data$Global_Sales)))
row.names(df_median) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                          "Other_Sales", "Global_Sales")
kable(df_median)
```

	Median
NA_Sales	0.08
EU_Sales	0.02
JP_Sales	0.00
Other_Sales	0.01
Global_Sales	0.17

Mode

```
Mode <- function(x){

    freq <- table(x)
    return(names(freq)[freq == max(freq)])

}

df_mode <- data.frame(Mode = c(Mode(data$NA_Sales),
                               Mode(data$EU_Sales),
                               Mode(data$JP_Sales),
                               Mode(data$Other_Sales),
                               Mode(data$Global_Sales)))
row.names(df_mode) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                        "Other_Sales", "Global_Sales")
kable(df_mode)
```

	Mode
NA_Sales	0
EU_Sales	0
JP_Sales	0
Other_Sales	0
Global_Sales	0.02

Mean + Median + Mode

```
df_mmm <- data.frame(Mean = df_means$Mean, Median = df_median, Mode = df_mode)
kable(df_mmm)
```

	Mean	Median	Mode
NA_Sales	0.2654635	0.08	0
EU_Sales	0.1475905	0.02	0
JP_Sales	0.0786773	0.00	0
Other_Sales	0.0483361	0.01	0
Global_Sales	0.5403431	0.17	0.02

1.2 C. Separating Measures

Percentile

```
percentile <- c()

for(i in 1:99){

    percentile <- c(percentile, i / 100)

}

df_percentiles <- data.frame(NA_Sales = quantile(data$NA_Sales, percentile),
                             EU_Sales = quantile(data$EU_Sales, percentile),
                             JP_Sales = quantile(data$JP_Sales, percentile),
                             Other_Sales = quantile(data$Other_Sales, percentile),
                             Global_Sales = quantile(data$Global_Sales, percentile))

kable(df_percentiles[c('25%','50%','75%','99%'),])
```

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
25%	0.0000	0.00	0.00	0.00	0.0600
50%	0.0800	0.02	0.00	0.01	0.1700
75%	0.2400	0.11	0.04	0.04	0.4800
99%	2.8156	1.94	1.27	0.65	5.4678

1.2 D. Dispersion Measures

Mean-Absolute Deviation

```r
# https://search.r-project.org/CRAN/refmans/ie2misc/html/madstat.html
library("ie2misc")

df_mad <- data.frame(MeanAbsoluteDeviation = c(madstat(data$NA_Sales),
                                                 madstat(data$EU_Sales),
                                                 madstat(data$JP_Sales),
                                                 madstat(data$Other_Sales),
                                                 madstat(data$Global_Sales)))

row.names(df_mad) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                      "Other_Sales", "Global_Sales")

kable(df_mad)
```

	MeanAbsoluteDeviation
NA_Sales	0.3094731
EU_Sales	0.1912648
JP_Sales	0.1167081
Other_Sales	0.0617212
Global_Sales	0.5945282

Variance

```
df_var <- data.frame(Variance = c(var(data$NA_Sales),
                                  var(data$EU_Sales),
                                  var(data$JP_Sales),
                                  var(data$Other_Sales),
                                  var(data$Global_Sales)))
row.names(df_var) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                       "Other_Sales", "Global_Sales")
kable(df_var)
```

	Variance
NA_Sales	0.6751640
EU_Sales	0.2589006
JP_Sales	0.0970904
Other_Sales	0.0360648
Global_Sales	2.4520629

Standard deviation

```r
df_std <- data.frame(StandardDeviation = c(sqrt(var(data$NA_Sales)),
                                 sqrt(var(data$EU_Sales)),
                                 sqrt(var(data$JP_Sales)),
                                 sqrt(var(data$Other_Sales)),
                                 sqrt(var(data$Global_Sales))))
row.names(df_std) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                       "Other_Sales", "Global_Sales")
kable(df_std)
```

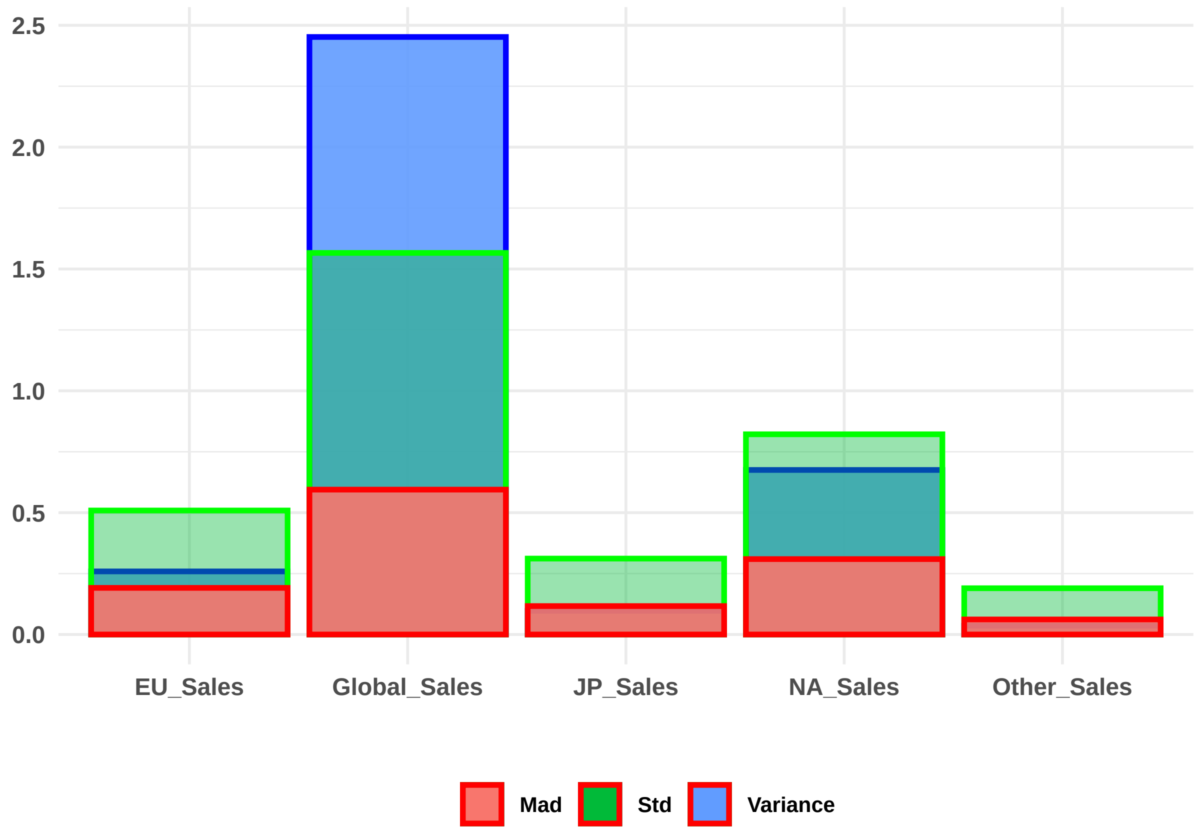
	StandardDeviation
NA_Sales	0.8216836
EU_Sales	0.5088228
JP_Sales	0.3115934
Other_Sales	0.1899074
Global_Sales	1.5659064

Mad + Var + Std

```r
df_dispersion <- data.frame(MeanAbsoluteDeviation = df_mad$MeanAbsoluteDeviation,
                            Variance = df_var$Variance,
                            StandardDeviation = df_std$StandardDeviation)

row.names(df_dispersion) <- c("NA_Sales", "EU_Sales", "JP_Sales",
                              "Other_Sales", "Global_Sales")

kable(df_dispersion)
```

	MeanAbsoluteDeviation	Variance	StandardDeviation
NA_Sales	0.3094731	0.6751640	0.8216836
EU_Sales	0.1912648	0.2589006	0.5088228
JP_Sales	0.1167081	0.0970904	0.3115934
Other_Sales	0.0617212	0.0360648	0.1899074
Global_Sales	0.5945282	2.4520629	1.5659064

```
ggplot(data = df_dispersion) +
    geom_bar(stat = "identity", mapping = aes(x = row.names(df_dispersion),
                                              y = Variance, fill = "Variance"),
             alpha = .9, linewidth = 1, color = "blue") +
    geom_bar(stat = "identity", mapping = aes(x = row.names(df_dispersion),
                                          y = StandardDeviation, fill = "Std"),
             alpha = .4, linewidth = 1, color = "green") +
    geom_bar(stat = "identity", mapping = aes(x = row.names(df_dispersion),
                                      y = MeanAbsoluteDeviation, fill = "Mad"),
             alpha = .9, linewidth = 1, color = "red") +
    xlab("") +
    ylab("") +
    theme_minimal() +
    theme(plot.title = element_text(size = 12, hjust = .5, face = "bold"),
          axis.title.x = element_text(size = 12, hjust = .5, face = "italic"),
          axis.title.y = element_text(size = 12, hjust = .5, face = "italic"),
          axis.text.x = element_text(size = 9, face = "bold"),
          axis.text.y = element_text(size = 9, face = "bold"),
          legend.position = "bottom",
          legend.title = element_text(color = "white"),
          legend.text = element_text(size = 8, face = "bold"))
```



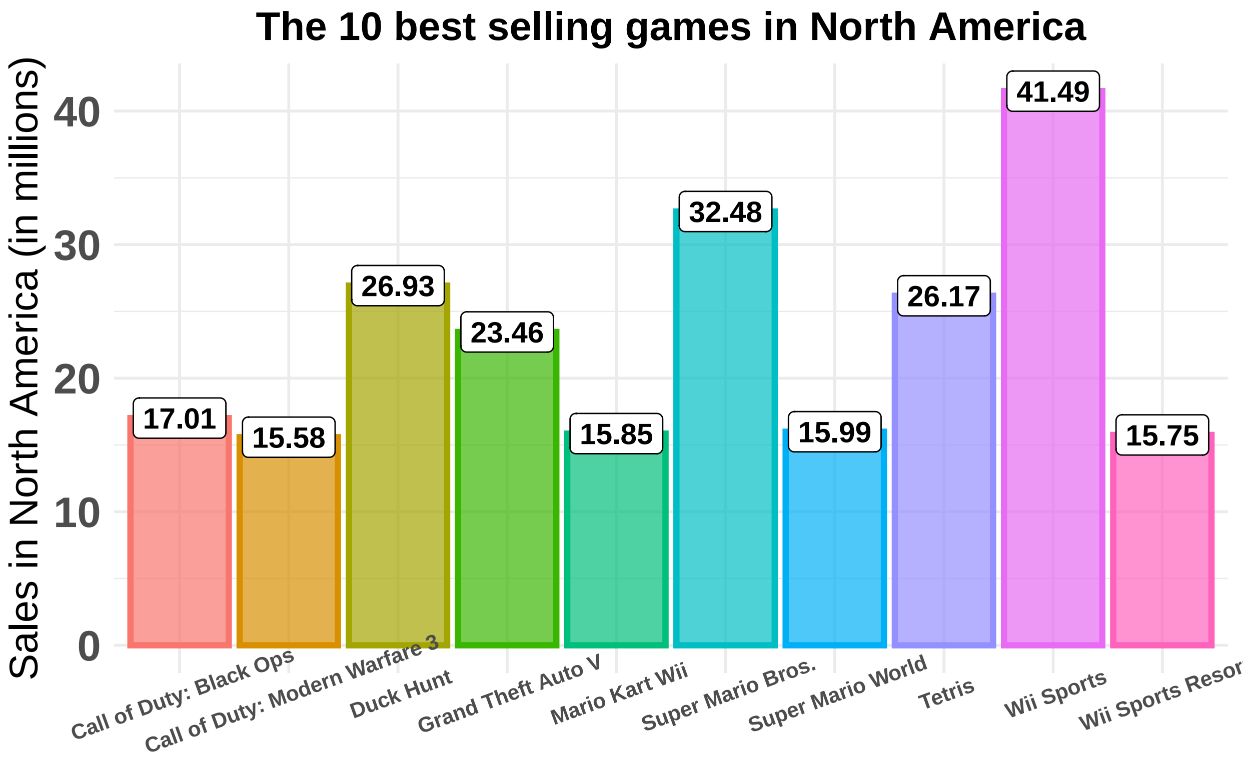
1.3 Exploratory Analysis

1.3 A. Analysis of the world’s best-selling games

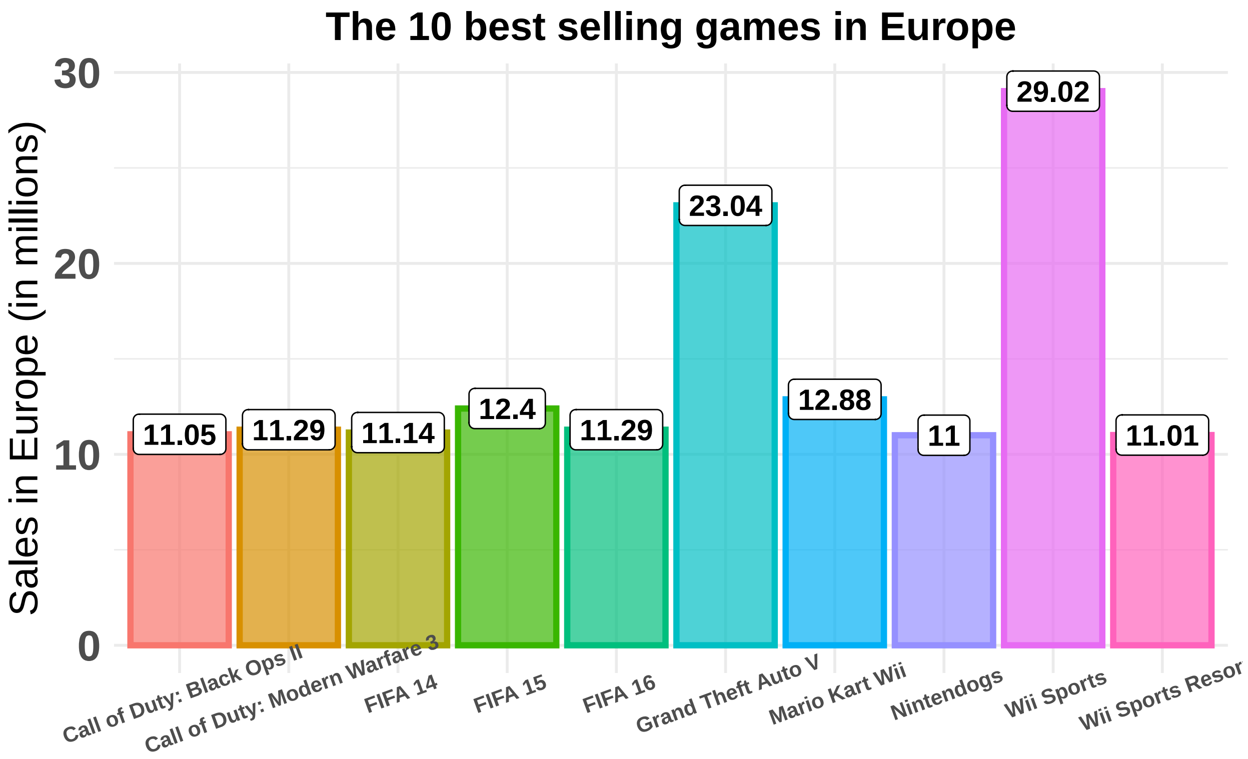
The best selling games in North America, Europe, Japan and the rest of the world

```r
# NA_Sales
t_v_name_NA <- aggregate(list(NA_Sales = data$NA_Sales),
                         list(Name = data$Name), sum)
t_v_name_NA <- t_v_name_NA[order(t_v_name_NA$NA_Sales,
                                 decreasing = T), ]

# EU_Sales
t_v_name_EU <- aggregate(list(EU_Sales = data$EU_Sales),
                         list(Name = data$Name), sum)
t_v_name_EU <- t_v_name_EU[order(t_v_name_EU$EU_Sales,
                                 decreasing = T), ]

# JP_Sales
t_v_name_JP <- aggregate(list(JP_Sales = data$JP_Sales),
                         list(Name = data$Name), sum)
t_v_name_JP <- t_v_name_JP[order(t_v_name_JP$JP_Sales,
                                 decreasing = T), ]

# Other_Sales
t_v_name_Other <- aggregate(list(Other_Sales = data$Other_Sales),
                            list(Name = data$Name), sum)
t_v_name_Other <- t_v_name_Other[order(t_v_name_Other$Other_Sales,
                                       decreasing = T), ]

# Global_Sales
t_v_name_Global <- aggregate(list(Global_Sales = data$Global_Sales),
                             list(Name = data$Name), sum)
t_v_name_Global <- t_v_name_Global[order(t_v_name_Global$Global_Sales,
                                         decreasing = T), ]
```

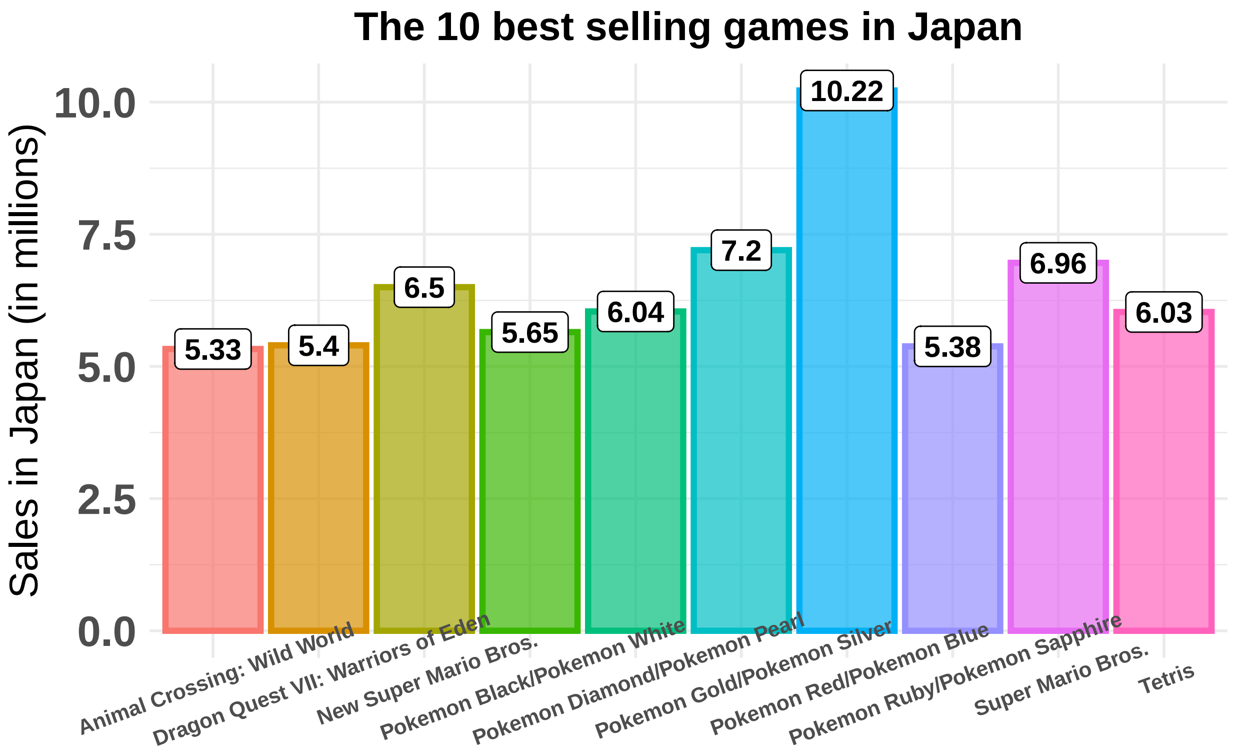
```
ggplot(data = head(t_v_name_NA, 10), mapping = aes(x = Name, y = NA_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
                  linewidth = 1.1, alpha = .7) +
         geom_label(mapping = aes(label = NA_Sales), size = 4, fontface = "bold") +
         xlab("") +
         ylab("Sales in North America (in millions)") +
         ggtitle("The 10 best selling games in North America") +
         theme_minimal() +
         theme(legend.position = "none",
               plot.title = element_text(size = 15, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 8, face = "bold", angle = 20),
               axis.text.y = element_text(size = 16, face = "bold"),
               axis.title.y = element_text(size = 15))
```



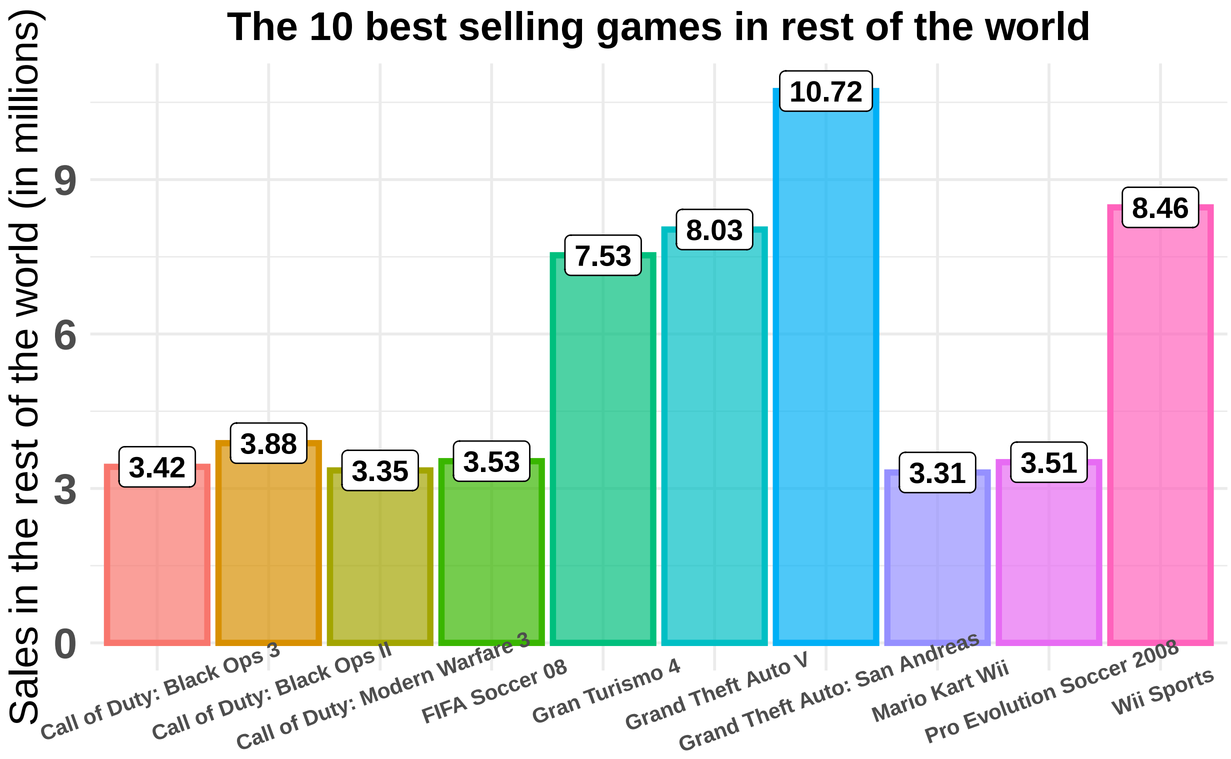
```
ggplot(data = head(t_v_name_EU, 10), mapping = aes(x = Name, y = EU_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
                  linewidth = 1.1, alpha = .7) +
         geom_label(mapping = aes(label = EU_Sales), size = 4, fontface = "bold") +
         xlab("") +
         ylab("Sales in Europe (in millions)") +
         ggtitle("The 10 best selling games in Europe") +
         theme_minimal() +
         theme(legend.position = "none",
               plot.title = element_text(size = 15, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 8, face = "bold", angle = 20),
               axis.text.y = element_text(size = 16, face = "bold"),
               axis.title.y = element_text(size = 15))
```



```
ggplot(data = head(t_v_name_JP, 10), mapping = aes(x = Name, y = JP_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
                  linewidth = 1.1, alpha = .7) +
         geom_label(mapping = aes(label = JP_Sales), size = 4, fontface = "bold") +
         xlab("") +
         ylab("Sales in Japan (in millions)") +
         ggtitle("The 10 best selling games in Japan") +
         theme_minimal() +
         theme(legend.position = "none",
               plot.title = element_text(size = 15, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 8, face = "bold", angle = 20),
               axis.text.y = element_text(size = 16, face = "bold"),
               axis.title.y = element_text(size = 15))
```



```
ggplot(data = head(t_v_name_Other, 10), mapping = aes(x = Name, y = Other_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
                  linewidth = 1.1, alpha = .7) +
         geom_label(mapping = aes(label = Other_Sales), size = 4, fontface = "bold") +
         xlab("") +
         ylab("Sales in the rest of the world (in millions)") +
         ggtitle("The 10 best selling games in rest of the world") +
         theme_minimal() +
         theme(legend.position = "none",
               plot.title = element_text(size = 15, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 8, face = "bold", angle = 20),
               axis.text.y = element_text(size = 16, face = "bold"),
               axis.title.y = element_text(size = 15))
```

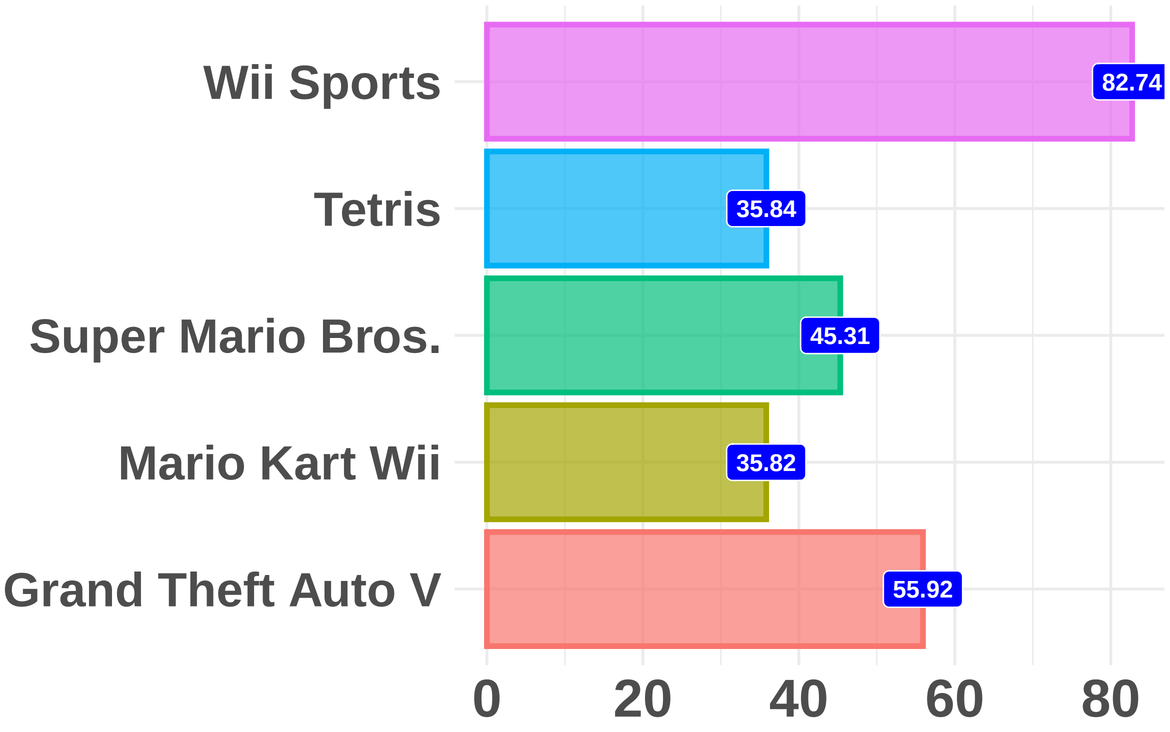


The best selling games in the world from 1980 to 2016

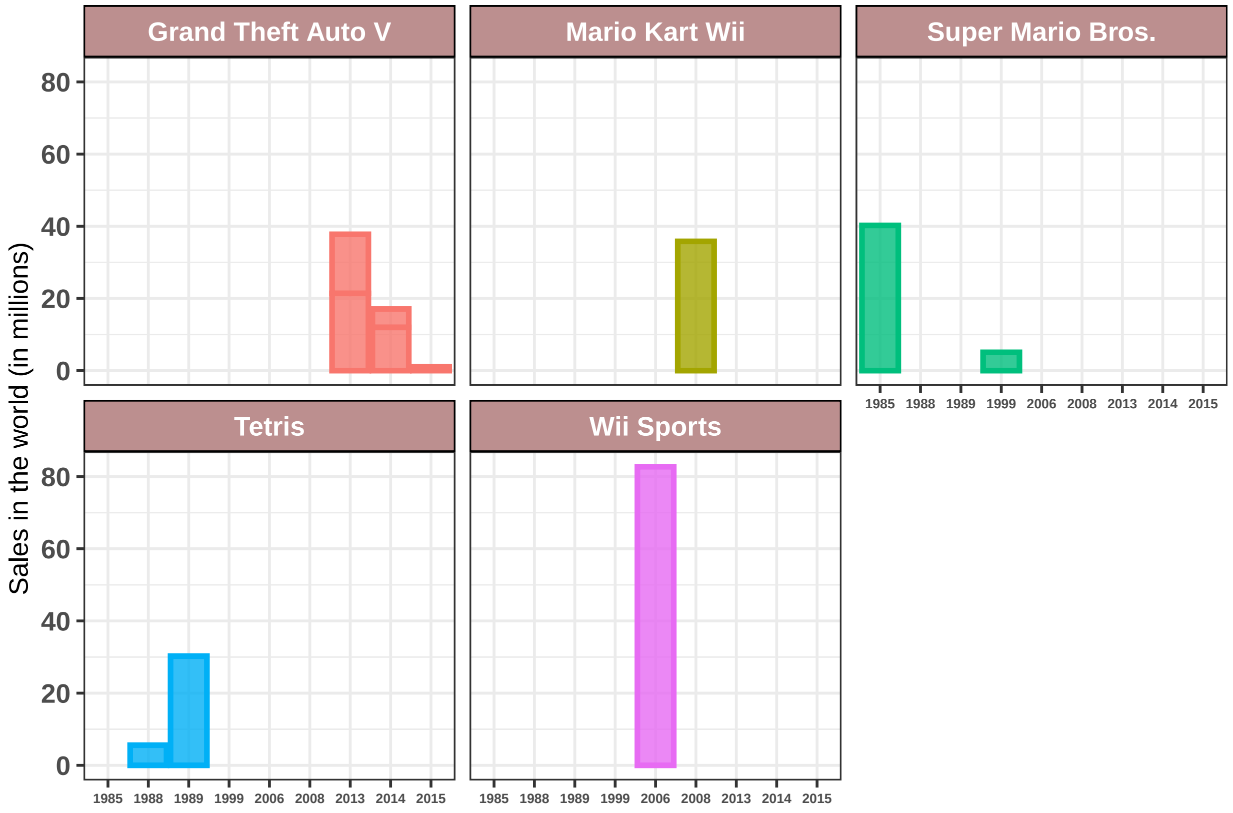
```r
a <- c()

for(i in 1:nrow(t_v_name_Global)){
    a <- c(a, i)
}

row.names(t_v_name_Global) <- a

kable(head(t_v_name_Global, 10))
```

Name	Global_Sales
Wii Sports	82.74
Grand Theft Auto V	55.92
Super Mario Bros.	45.31
Tetris	35.84
Mario Kart Wii	35.82
Wii Sports Resort	33.00
Pokemon Red/Pokemon Blue	31.37
Call of Duty: Modern Warfare 3	30.83
New Super Mario Bros.	30.01
Call of Duty: Black Ops II	29.72

```
ggplot(data = head(t_v_name_Global, 5),
       mapping = aes(x = Name, y = Global_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
                  linewidth = 1, alpha = .7) +
         geom_label(mapping = aes(label = Global_Sales),
                color = "white", fill = "blue", size = 3, fontface = "bold") +
         xlab("") +
         ylab("") +
         ggtitle("The best selling games in the world from 1980 to 2016") +
         theme_minimal() +
         coord_flip() +
         theme(legend.position = "none",
               plot.title = element_text(size = 25, face = "bold",
                                         hjust = -2, vjust = 4),
               axis.text.x = element_text(size = 20, face = "bold"),
               axis.text.y = element_text(size = 18, face = "bold"),
               axis.title.y = element_text(size = 20))
```



```r
df_top_5 <- data[data$Name == "Wii Sports" | data$Name == "Grand Theft Auto V" |
                   data$Name == "Super Mario Bros." | data$Name == "Tetris" |
                   data$Name == "Mario Kart Wii", ]

ggplot(data = df_top_5, mapping = aes(x = Year, y = Global_Sales)) +
    geom_bar(stat = "identity", mapping = aes(fill = Name, color = Name),
             linewidth = 1, alpha = .8) +
    facet_wrap(~Name) +
    theme_bw() +
    xlab("") +
    ylab("Sales in the world (in millions)") +
    theme(
          legend.position = "none",
          strip.text.x = element_text(margin = margin(5, 5, 5, 5),
                                      size = 10, face = "bold", color = "white"),
          strip.background = element_rect(fill = "#BC8F8F", color = "black"),
          plot.title = element_text(size = 10, face = "bold", hjust = .5),
          axis.text.x = element_text(size = 5, face = "bold"),
          axis.text.y = element_text(size = 10, face = "bold"),
          axis.title.y = element_text(size = 10))
```

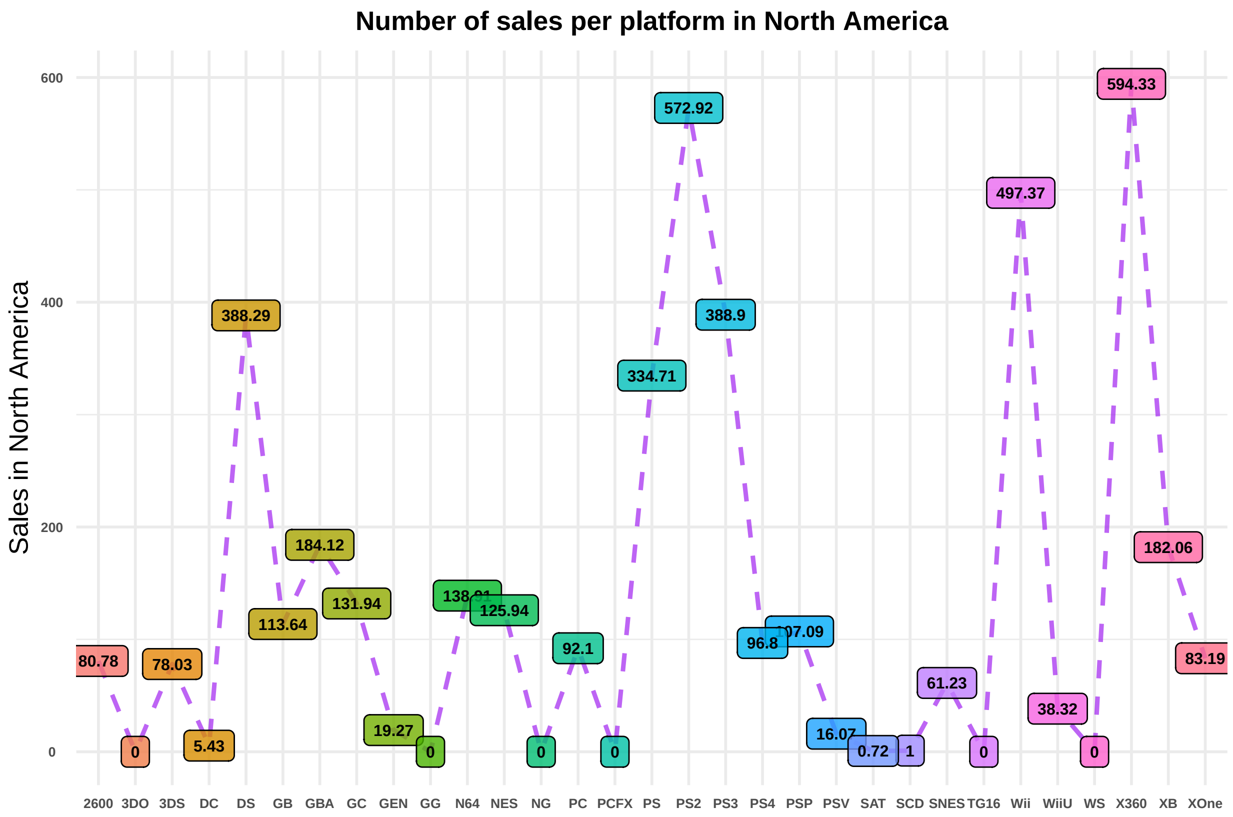


1.3 B. Number of Sales per platform

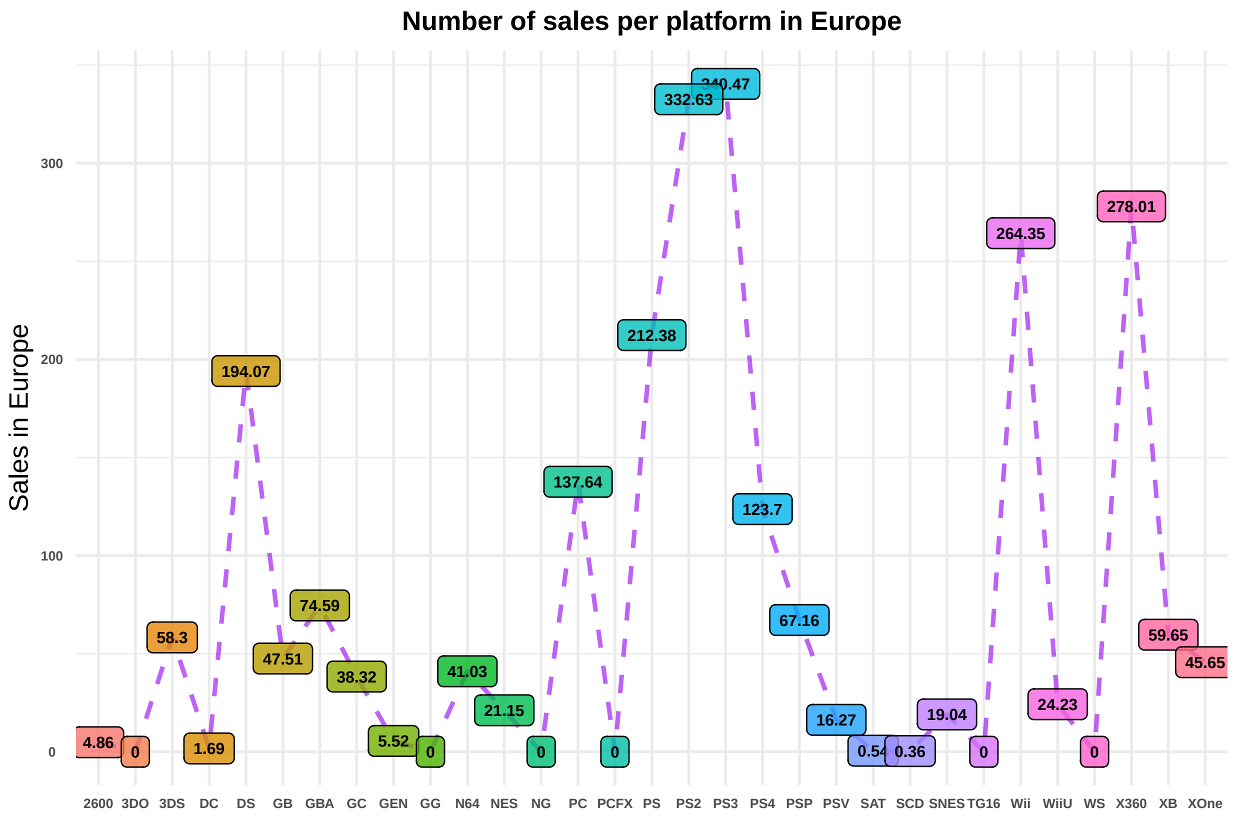
Number of Sales per platform in North America, Europe, Japan and the rest of the world

```r
# NA_Sales
p_name_NA <- aggregate(list(NA_Sales = data$NA_Sales),
                       list(Platform = data$Platform), sum)
p_name_NA <- p_name_NA[order(p_name_NA$NA_Sales, decreasing = T), ]

# EU_Sales
p_name_EU <- aggregate(list(EU_Sales = data$EU_Sales),
                       list(Platform = data$Platform), sum)
p_name_EU <- p_name_EU[order(p_name_EU$EU_Sales, decreasing = T), ]


# JP_Sales
p_name_JP <- aggregate(list(JP_Sales = data$JP_Sales),
                       list(Platform = data$Platform), sum)
p_name_JP <- p_name_JP[order(p_name_JP$JP_Sales, decreasing = T), ]

# Other_Sales
p_name_Other <- aggregate(list(Other_Sales = data$Other_Sales),
                          list(Platform = data$Platform), sum)
p_name_Other <- p_name_Other[order(p_name_Other$Other_Sales,
                                   decreasing = T), ]

# Global_Sales
p_name_Global <- aggregate(list(Global_Sales = data$Global_Sales),
                           list(Platform = data$Platform), sum)
p_name_Global <- p_name_Global[order(p_name_Global$Global_Sales,
                                     decreasing = T), ]
```

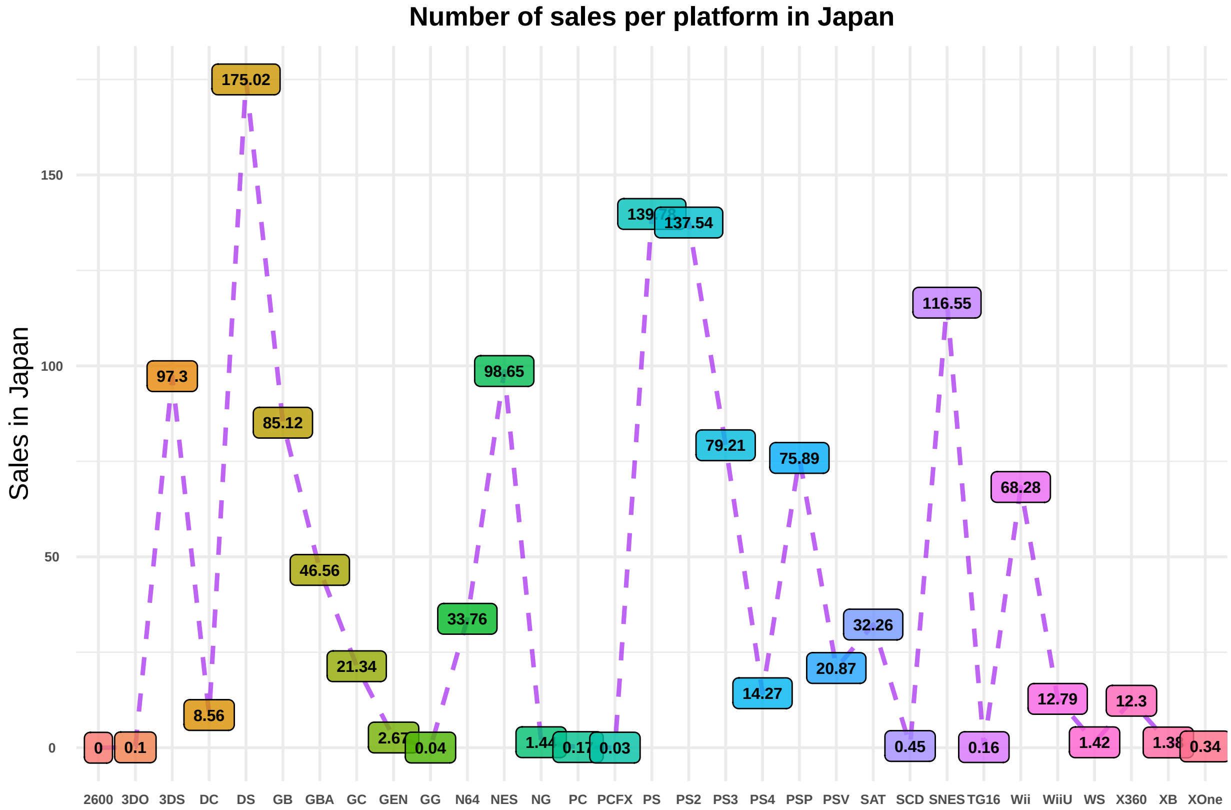
```
ggplot(data = p_name_NA, 10, mapping = aes(x = Platform, y = NA_Sales)) +
         geom_line(linewidth = .8, alpha = .7, group = 1,
                   linetype = 2, color = "purple") +
         geom_label(mapping = aes(label = NA_Sales, fill = Platform),
                    color = "black", size = 2, fontface = "bold", alpha = .8) +
         xlab("") +
         ylab("Sales in North America") +
         ggtitle("Number of sales per platform in North America") +
         theme_minimal() +
         theme(
               legend.position = "none",
               plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 5, face = "bold", hjust = .5),
               axis.title.y = element_text(size = 10, hjust = .5))
```



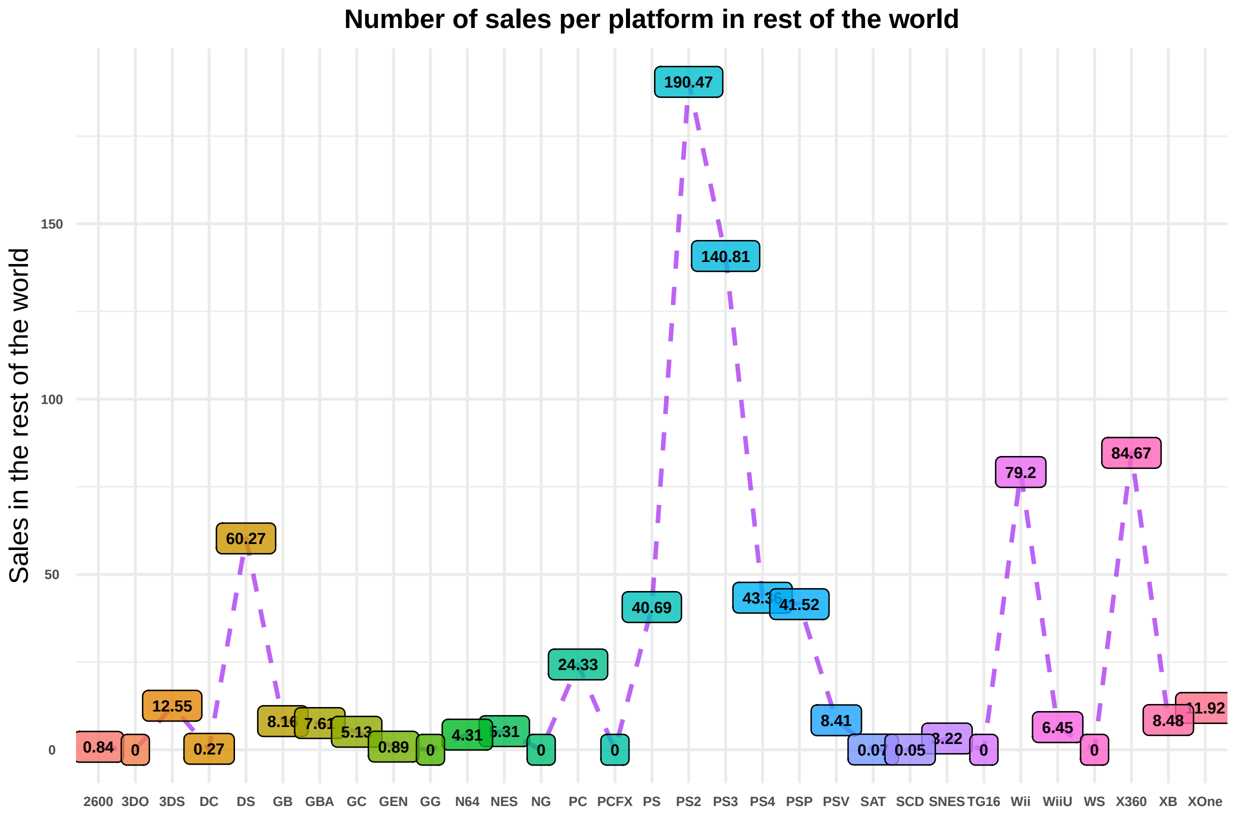
```
ggplot(data = p_name_EU, 10, mapping = aes(x = Platform, y = EU_Sales)) +
         geom_line(linewidth = .8, alpha = .7, group = 1,
                   linetype = 2, color = "purple") +
         geom_label(mapping = aes(label = EU_Sales, fill = Platform),
                    color = "black", size = 2, fontface = "bold", alpha = .8) +
         xlab("") +
         ylab("Sales in Europe") +
         ggtitle("Number of sales per platform in Europe") +
         theme_minimal() +
         theme(
               legend.position = "none",
               plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 5, face = "bold", hjust = .5),
               axis.title.y = element_text(size = 10, hjust = .5))
```



```
ggplot(data = p_name_JP, 10, mapping = aes(x = Platform, y = JP_Sales)) +
         geom_line(linewidth = .8, alpha = .7, group = 1,
                   linetype = 2, color = "purple") +
         geom_label(mapping = aes(label = JP_Sales, fill = Platform),
                    color = "black", size = 2, fontface = "bold", alpha = .8) +
         xlab("") +
         ylab("Sales in Japan") +
         ggtitle("Number of sales per platform in Japan") +
         theme_minimal() +
         theme(
               legend.position = "none",
               plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 5, face = "bold", hjust = .5),
               axis.title.y = element_text(size = 10, hjust = .5))
```



```r
ggplot(data = p_name_Other, 10, mapping = aes(x = Platform, y = Other_Sales)) +
         geom_line(linewidth = .8, alpha = .7, group = 1,
                   linetype = 2, color = "purple") +
         geom_label(mapping = aes(label = Other_Sales, fill = Platform),
                    color = "black", size = 2, fontface = "bold", alpha = .8) +
         xlab("") +
         ylab("Sales in the rest of the world") +
         ggtitle("Number of sales per platform in rest of the world") +
         theme_minimal() +
         theme(
               legend.position = "none",
               plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 5, face = "bold", hjust = .5),
               axis.title.y = element_text(size = 10, hjust = .5))
```

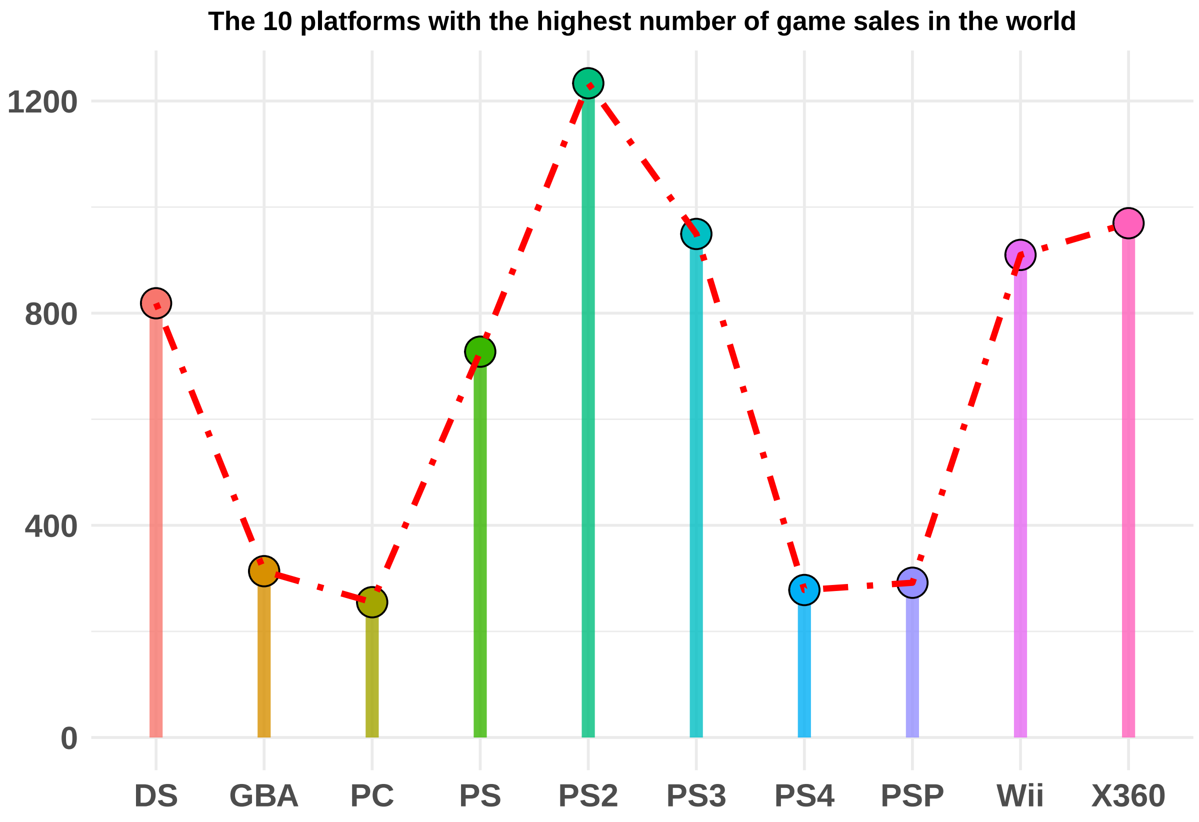


The 10 platforms with the highest number of game sales in the world

```
a <- c()

for(i in 1:nrow(p_name_Global)){
    a <- c(a, i)
}

row.names(p_name_Global) <- a

kable(head(p_name_Global, 10))
```

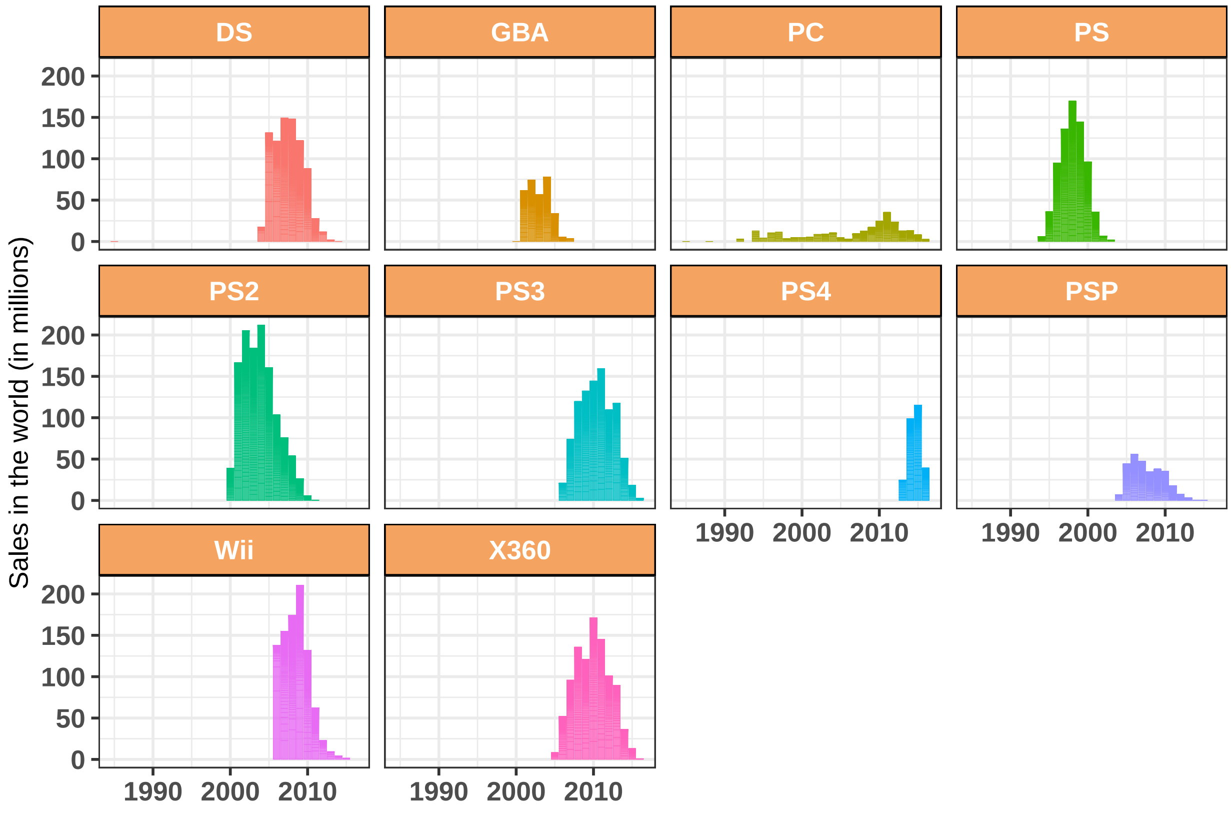
Platform	Global_Sales
PS2	1233.46
X360	969.61
PS3	949.35
Wii	909.81
DS	818.67
PS	727.39
GBA	313.56
PSP	291.71
PS4	278.07
PC	255.05

```r
ggplot(data = head(p_name_Global, 10),
   mapping = aes(x = Platform, y = Global_Sales)) +
   geom_segment(aes(xend=Platform, yend=0,
                    color = Platform), linewidth = 2.3, alpha = .8) +
   geom_point(mapping = aes(fill = Platform), size = 5, shape = 21) +
   geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
   xlab("") +
   ylab("") +
   ggtitle("The 10 platforms with the highest number of game sales in the world") +
   theme_minimal() +
   theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
         axis.title.x = element_text(size = 10, hjust = .5, face = "italic"),
         axis.title.y = element_text(size = 10, hjust = .5, face = "italic"),
         axis.text.x = element_text(size = 12, face = "bold"),
         axis.text.y = element_text(size = 12, face = "bold"),
         legend.position = "none")
```



```
d_top_10 <- data[data$Platform == "PS2"
                 | data$Platform == "X360"
                 | data$Platform == "PS3"
                 | data$Platform == "Wii"
                 | data$Platform == "DS"
                 | data$Platform == "PS"
                 | data$Platform == "GBA"
                 | data$Platform == "PSP"
                 | data$Platform == "PS4"
                 | data$Platform == "PC", ]
d_top_10$Year <- as.numeric(levels(d_top_10$Year))[d_top_10$Year]
```

```r
ggplot(data = d_top_10, mapping = aes(x = Year, y = Global_Sales)) +
    geom_bar(stat = "identity",
             mapping = aes(fill = Platform, color = Platform),
             linewidth = .1, alpha = .8) +
    facet_wrap(~Platform) +
    theme_bw() +
    xlab("") +
    ylab("Sales in the world (in millions)") +
    theme(
          legend.position = "none",
          strip.text.x = element_text(margin = margin(5, 5, 5, 5), size = 10,
                                      face = "bold", color = "white"),
          strip.background = element_rect(fill = "#F4A460", color = "black"),
          plot.title = element_text(size = 12, face = "bold", hjust = .5),
          axis.text.x = element_text(size = 10, face = "bold"),
          axis.text.y = element_text(size = 10, face = "bold"),
          axis.title.y = element_text(size = 10))
```

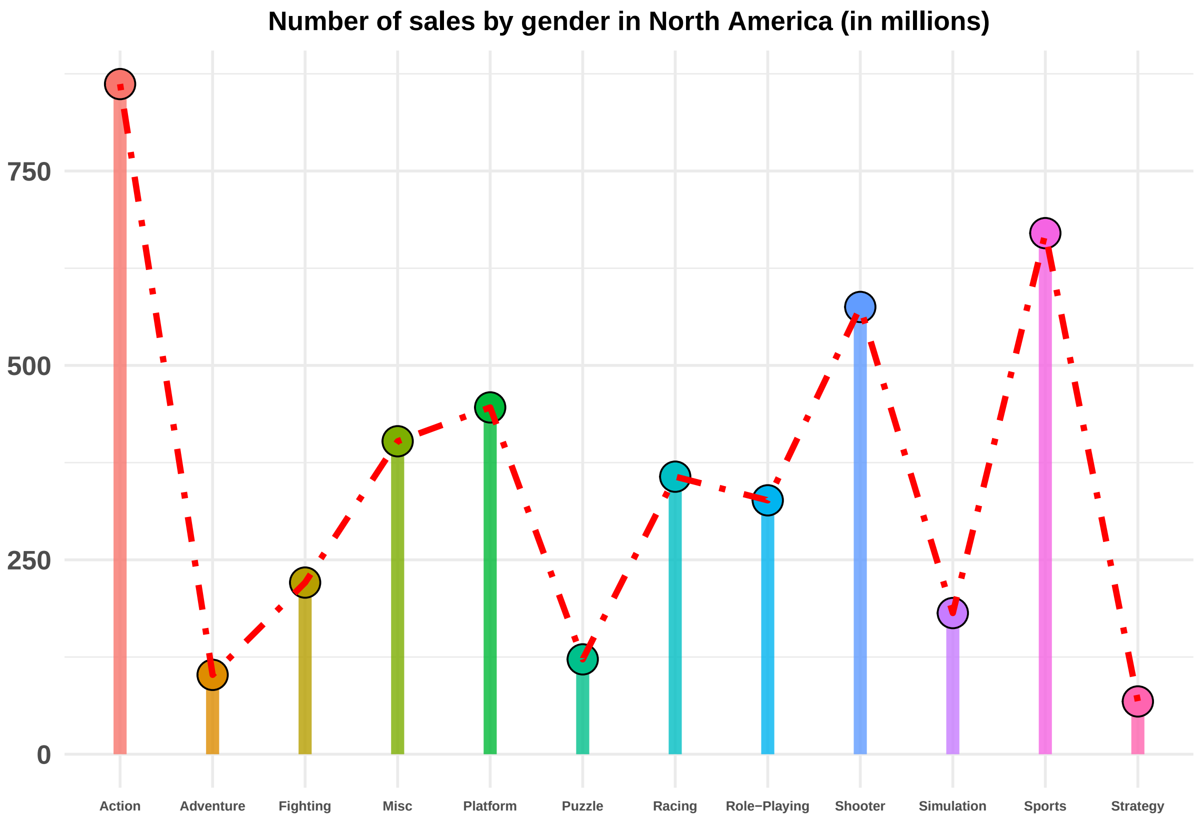


1.3 C. Game Sales by Genre

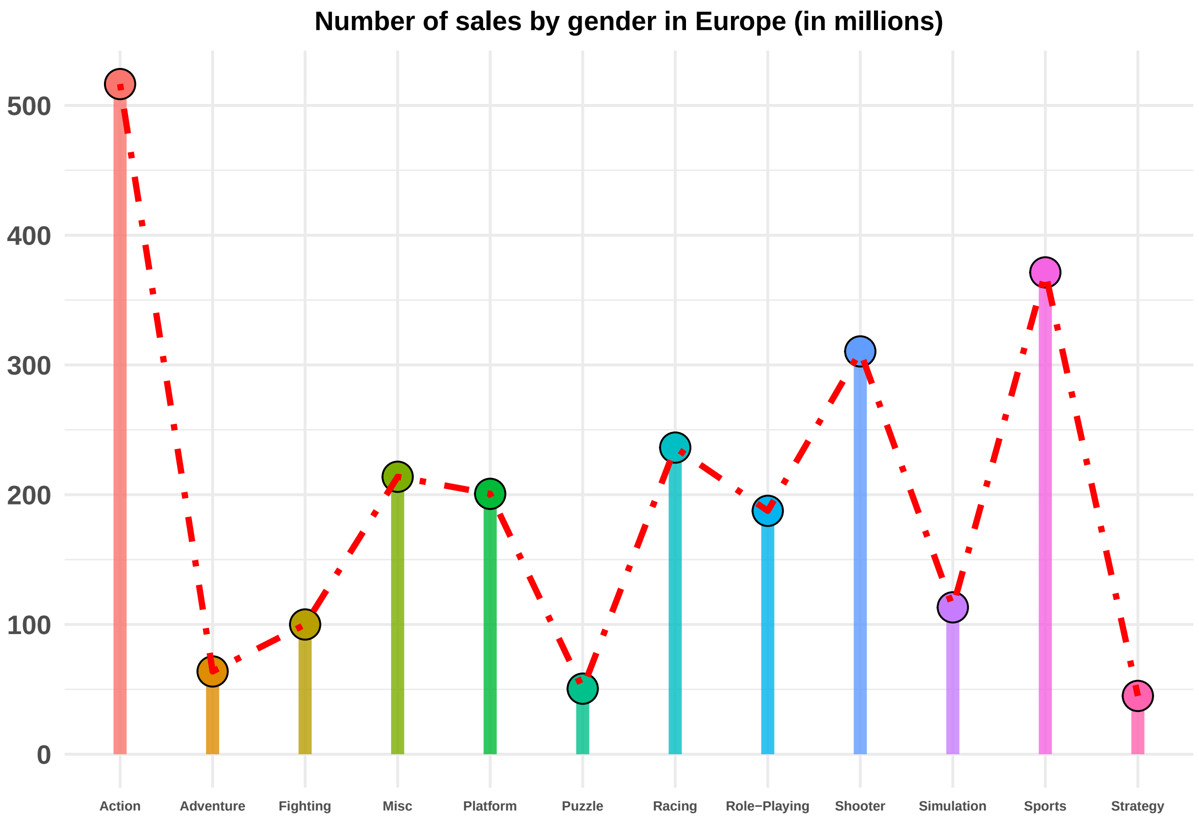
Best-selling people from 1980 to 2016 by features

```
# NA_Sales
g_name_NA <- aggregate(list(NA_Sales = data$NA_Sales),
                       list(Genre = data$Genre), sum)
g_name_NA <- g_name_NA[order(g_name_NA$NA_Sales, decreasing = T), ]

# EU_Sales
g_name_EU <- aggregate(list(EU_Sales = data$EU_Sales),
                       list(Genre = data$Genre), sum)
g_name_EU <- g_name_EU[order(g_name_EU$EU_Sales, decreasing = T), ]


# JP_Sales
g_name_JP <- aggregate(list(JP_Sales = data$JP_Sales),
                       list(Genre = data$Genre), sum)
g_name_JP <- g_name_JP[order(g_name_JP$JP_Sales, decreasing = T), ]

# Other_Sales
g_name_Other <- aggregate(list(Other_Sales = data$Other_Sales),
                          list(Genre = data$Genre), sum)
g_name_Other <- g_name_Other[order(g_name_Other$Other_Sales,
                                   decreasing = T), ]

# Global_Sales
g_name_Global <- aggregate(list(Global_Sales = data$Global_Sales),
                           list(Genre = data$Genre), sum)
g_name_Global <- g_name_Global[order(g_name_Global$Global_Sales,
                                     decreasing = T), ]
```

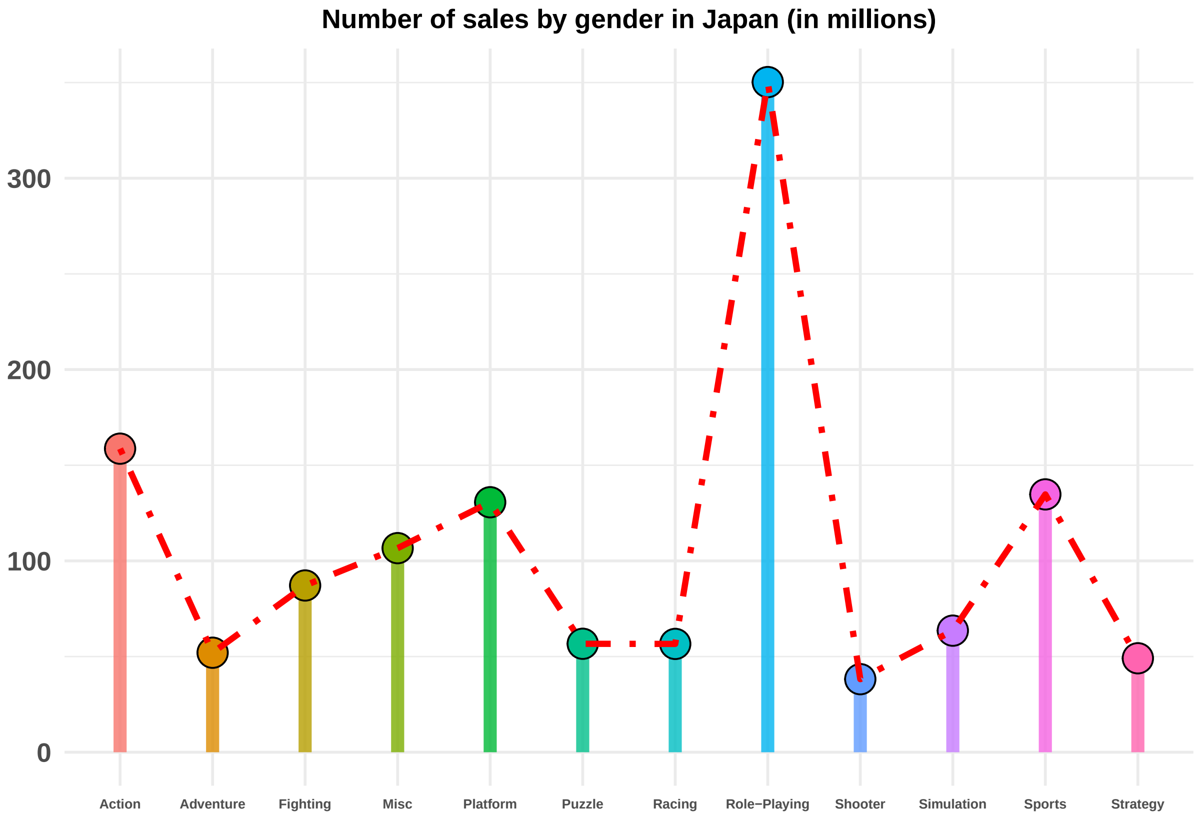
```
ggplot(data = g_name_NA, mapping = aes(x = Genre, y = NA_Sales)) +
         geom_segment(aes(xend=Genre, yend=0, color = Genre),
                      linewidth = 2.3, alpha = .8) +
         geom_point(mapping = aes(fill = Genre), size = 5, shape = 21) +
         geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
         xlab("") +
         ylab("") +
         ggtitle("Number of sales by gender in North America (in millions)") +
         theme_minimal() +
         theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



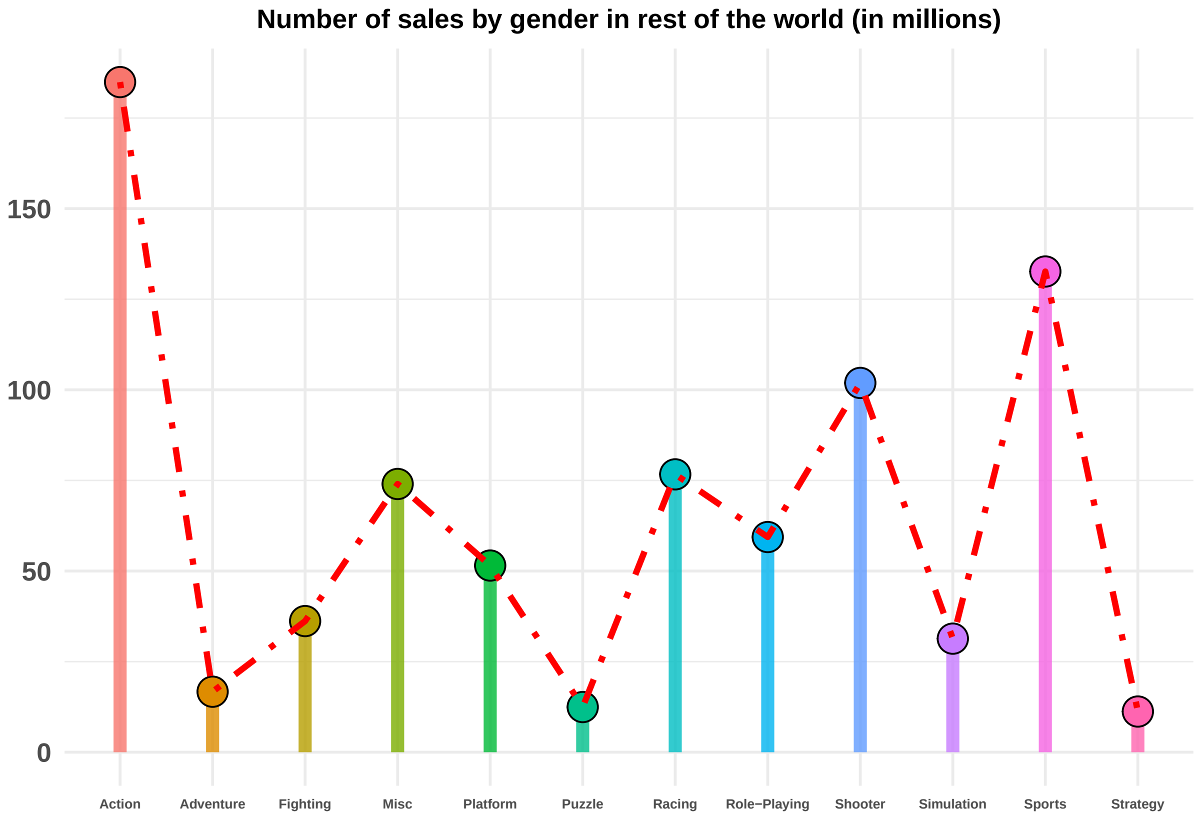
```
ggplot(data = g_name_EU, mapping = aes(x = Genre, y = EU_Sales)) +
         geom_segment(aes(xend=Genre, yend=0, color = Genre),
                      linewidth = 2.3, alpha = .8) +
         geom_point(mapping = aes(fill = Genre), size = 5, shape = 21) +
         geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
         xlab("") +
         ylab("") +
         ggtitle("Number of sales by gender in Europe (in millions)") +
         theme_minimal() +
         theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```r
ggplot(data = g_name_JP, mapping = aes(x = Genre, y = JP_Sales)) +
         geom_segment(aes(xend=Genre, yend=0, color = Genre),
                      linewidth = 2.3, alpha = .8) +
         geom_point(mapping = aes(fill = Genre), size = 5, shape = 21) +
         geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
         xlab("") +
         ylab("") +
         ggtitle("Number of sales by gender in Japan (in millions)") +
         theme_minimal() +
         theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```
ggplot(data = g_name_Other, mapping = aes(x = Genre, y = Other_Sales)) +
         geom_segment(aes(xend=Genre, yend=0, color = Genre),
                      linewidth = 2.3, alpha = .8) +
         geom_point(mapping = aes(fill = Genre), size = 5, shape = 21) +
         geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
         xlab("") +
         ylab("") +
         ggtitle("Number of sales by gender in rest of the world (in millions)") +
         theme_minimal() +
         theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 5, face = "bold"),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```

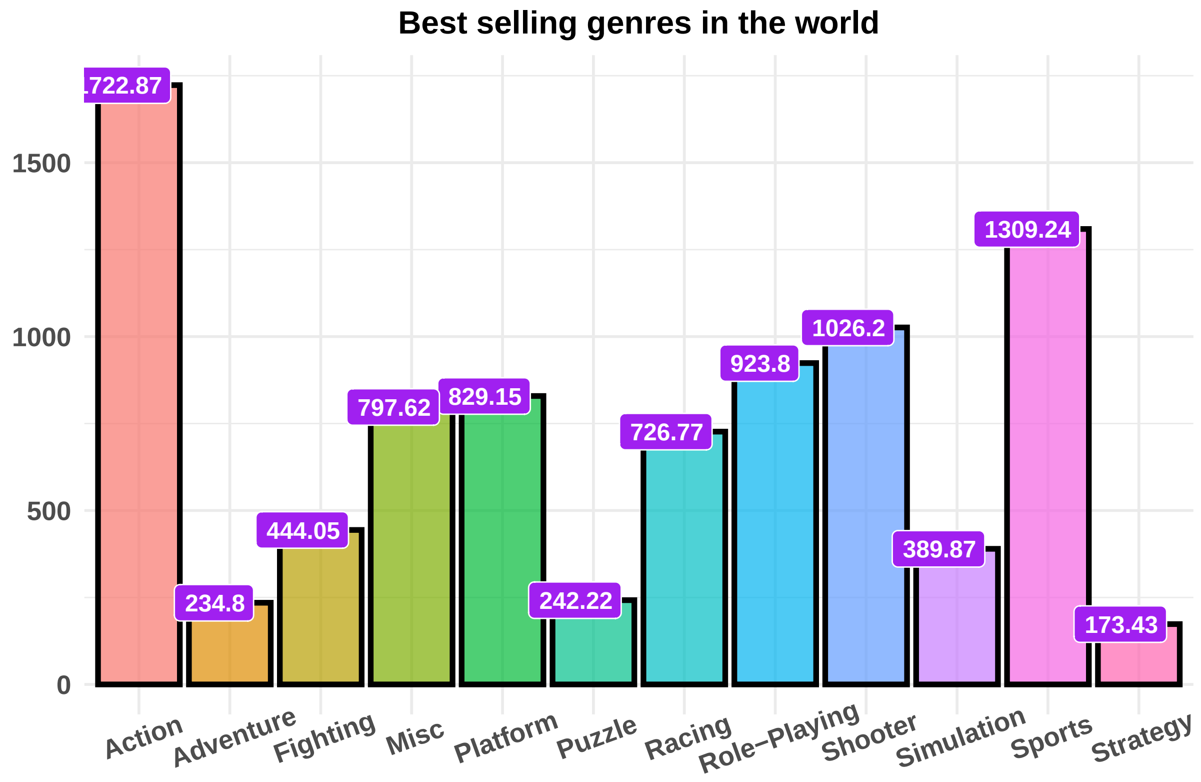


Best selling genres in the world

```
a <- c()

for(i in 1:nrow(g_name_Global)){
    a <- c(a, i)
}

row.names(g_name_Global) <- a

kable(g_name_Global)
```

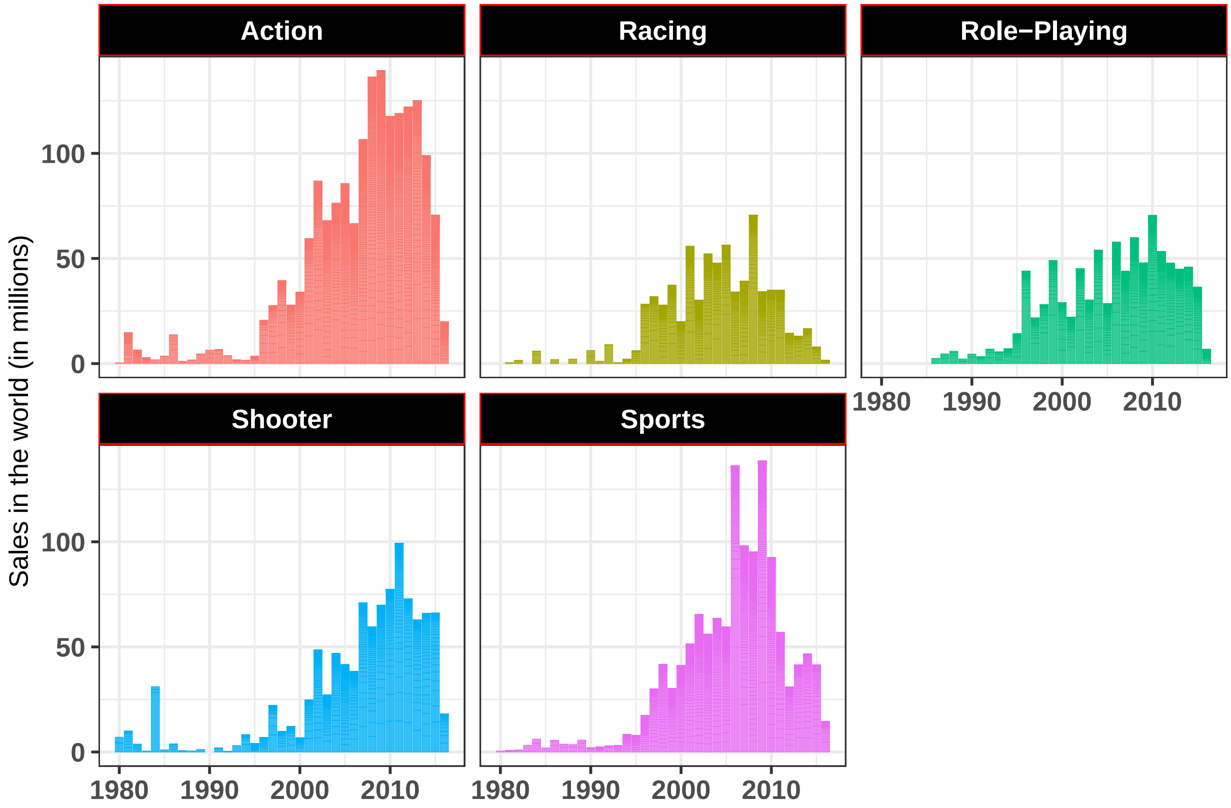
Genre	Global_Sales
Action	1722.87
Sports	1309.24
Shooter	1026.20
Role-Playing	923.80
Platform	829.15
Misc	797.62
Racing	726.77
Fighting	444.05
Simulation	389.87
Puzzle	242.22
Adventure	234.80
Strategy	173.43

```
ggplot(data = g_name_Global, mapping = aes(x = Genre, y = Global_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Genre),
                  alpha = .7, linewidth = 1, color = "black") +
         geom_label(mapping = aes(label=Global_Sales), fill = "purple",
                    size = 3, color = "white", fontface = "bold", hjust=.7) +
         ggtitle("Best selling genres in the world") +
         xlab(" ") +
         ylab("") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 12, hjust = .5,
                                         face = "bold"),
               axis.title.x = element_text(size = 12, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 12, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold",
                                          angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```r
g_top_10 <- data[data$Genre == "Action"
                 | data$Genre == "Sports"
                 | data$Genre == "Shooter"
                 | data$Genre == "Role-Playing"
                 | data$Genre == "Racing", ]
g_top_10$Year <- as.numeric(levels(g_top_10$Year))[g_top_10$Year]
```

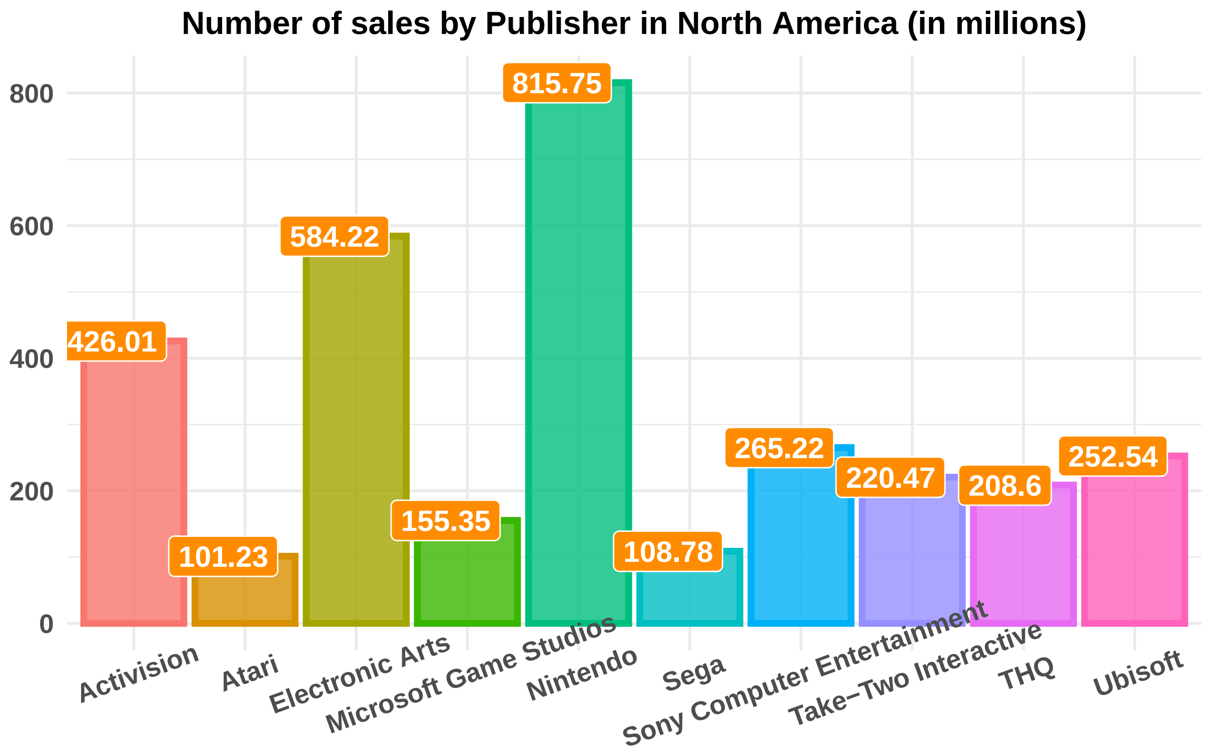
```
ggplot(data = g_top_10, mapping = aes(x = Year, y = Global_Sales)) +
    geom_bar(stat = "identity", mapping = aes(fill = Genre,
                        color = Genre), linewidth = .1, alpha = .8) +
    facet_wrap(~Genre) +
    theme_bw() +
    xlab("") +
    ylab("Sales in the world (in millions)") +
    theme(
        legend.position = "none",
        strip.text.x = element_text(margin = margin(5, 5, 5, 5),
                            size = 10, face = "bold", color = "white"),
        strip.background = element_rect(fill = "black", color = "red"),
        plot.title = element_text(size = 11, face = "bold", hjust = .5),
        axis.text.x = element_text(size = 10, face = "bold"),
        axis.text.y = element_text(size = 10, face = "bold"),
        axis.title.y = element_text(size = 10))
```



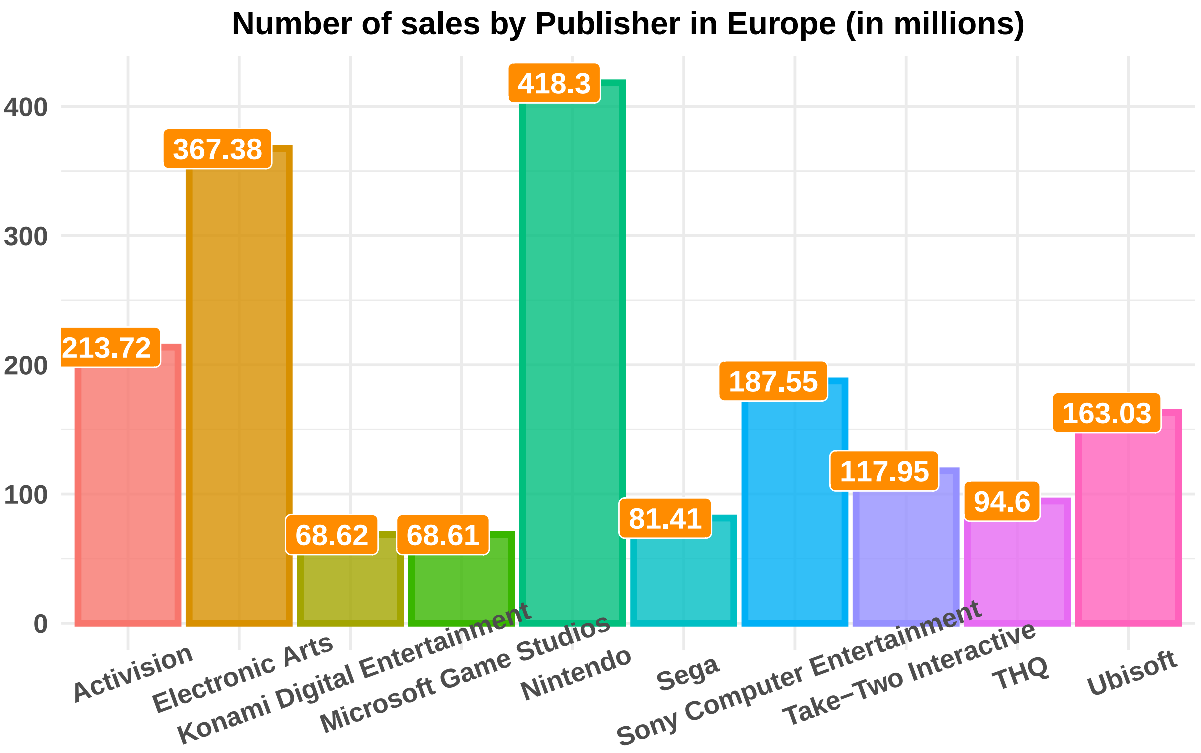
1.3 D. Number of sales per publisher

```
# NA_Sales
pu_name_NA <- aggregate(list(NA_Sales = data$NA_Sales), list(Publisher
                                                             = data$Publisher), sum)
pu_name_NA <- pu_name_NA[order(pu_name_NA$NA_Sales, decreasing = T), ]

# EU_Sales
pu_name_EU <- aggregate(list(EU_Sales = data$EU_Sales), list(Publisher
                                                             = data$Publisher), sum)
pu_name_EU <- pu_name_EU[order(pu_name_EU$EU_Sales, decreasing = T), ]


# JP_Sales
pu_name_JP <- aggregate(list(JP_Sales = data$JP_Sales), list(Publisher
                                                             = data$Publisher), sum)
pu_name_JP <- pu_name_JP[order(pu_name_JP$JP_Sales, decreasing = T), ]

# Other_Sales
pu_name_Other <- aggregate(list(Other_Sales = data$Other_Sales), list(Publisher
                                                             = data$Publisher), sum)
pu_name_Other <- pu_name_Other[order(pu_name_Other$Other_Sales, decreasing = T), ]

# Global_Sales
pu_name_Global <- aggregate(list(Global_Sales = data$Global_Sales), list(Publisher
                                                             = data$Publisher), sum)
pu_name_Global <- pu_name_Global[order(pu_name_Global$Global_Sales, decreasing = T), ]
```

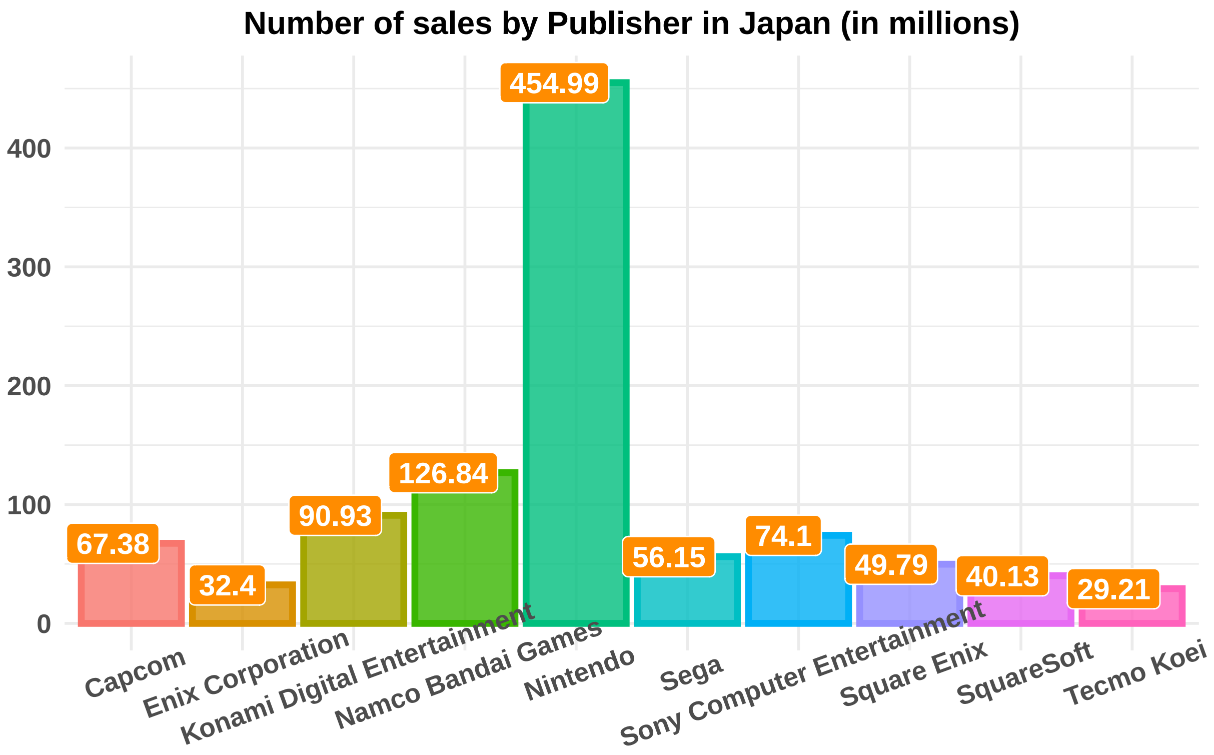
```r
ggplot(data = head(pu_name_NA, 10), mapping = aes(x = Publisher, y = NA_Sales))+
         geom_bar(stat = "identity", aes(fill = Publisher, color = Publisher),
                  linewidth = 1.2, alpha = .8)+
         geom_label(mapping = aes(label=NA_Sales), fill = "#FF8C00",
                    size = 4, color = "white", fontface = "bold", hjust=.7)+
         xlab("")+
         ylab("")+
         ggtitle("Number of sales by Publisher in North America (in millions)")+
         theme_minimal()+
         theme(plot.title = element_text(size = 12, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 8, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 8, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold", angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



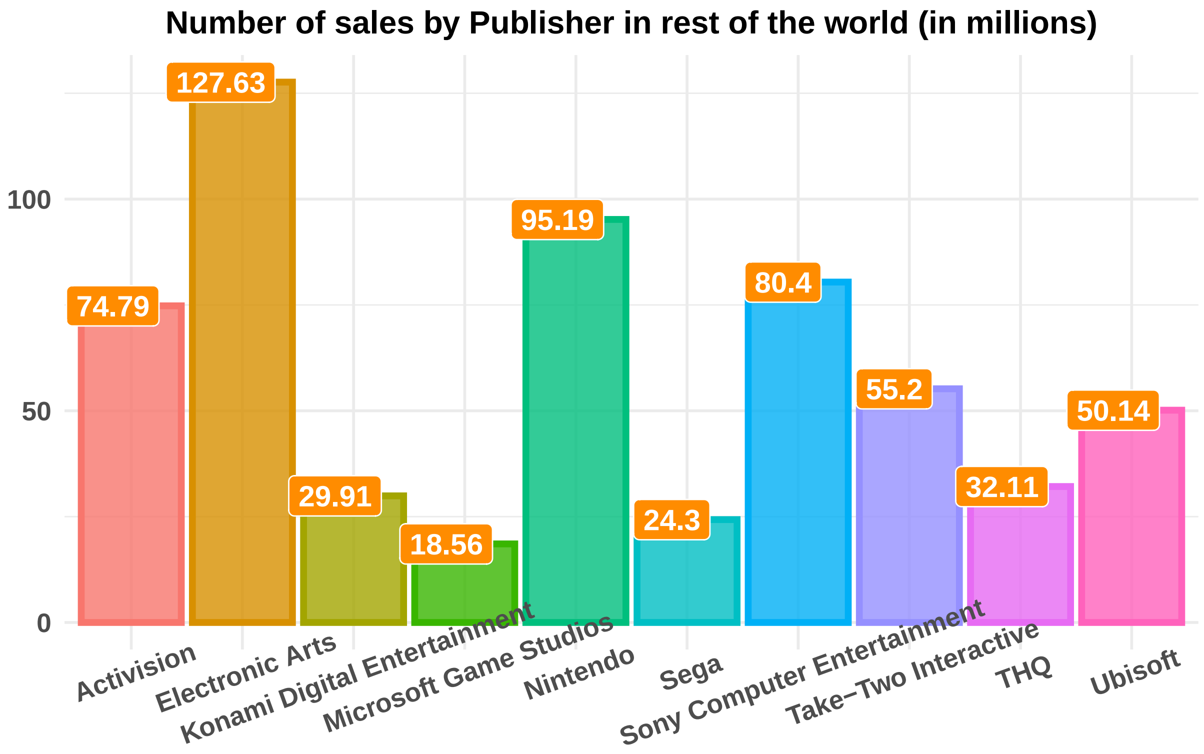
```
ggplot(data = head(pu_name_EU, 10), mapping = aes(x = Publisher, y = EU_Sales))+
         geom_bar(stat = "identity", aes(fill = Publisher, color = Publisher),
                  linewidth = 1.2, alpha = .8)+
         geom_label(mapping = aes(label=EU_Sales), fill = "#FF8C00",
                    size = 4, color = "white", fontface = "bold", hjust=.7)+
         xlab("")+
         ylab("")+
         ggtitle("Number of sales by Publisher in Europe (in millions)")+
         theme_minimal()+
         theme(plot.title = element_text(size = 12, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 8, hjust = .5,
                                            face = "italic"),
               axis.title.y = element_text(size = 8, hjust = .5,
                                            face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold", angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```
ggplot(data = head(pu_name_JP, 10), mapping = aes(x = Publisher, y = JP_Sales))+
         geom_bar(stat = "identity", aes(fill = Publisher, color = Publisher),
                  linewidth = 1.2, alpha = .8)+
         geom_label(mapping = aes(label=JP_Sales), fill = "#FF8C00",
                    size = 4, color = "white", fontface = "bold", hjust=.7)+
         xlab("")+
         ylab("")+
         ggtitle("Number of sales by Publisher in Japan (in millions)")+
         theme_minimal()+
         theme(plot.title = element_text(size = 12, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 8, hjust = .5,
                                            face = "italic"),
               axis.title.y = element_text(size = 8, hjust = .5,
                                            face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold", angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```
ggplot(data = head(pu_name_Other, 10), mapping = aes(x = Publisher,
                                                     y = Other_Sales))+
         geom_bar(stat = "identity", aes(fill = Publisher, color = Publisher),
                  linewidth = 1.2, alpha = .8)+
         geom_label(mapping = aes(label=Other_Sales), fill = "#FF8C00",
                    size = 4, color = "white", fontface = "bold", hjust=.7)+
         xlab("")+
         ylab("")+
         ggtitle("Number of sales by Publisher in rest of the world (in millions)")+
         theme_minimal()+
         theme(plot.title = element_text(size = 12, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 8, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 8, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold", angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```

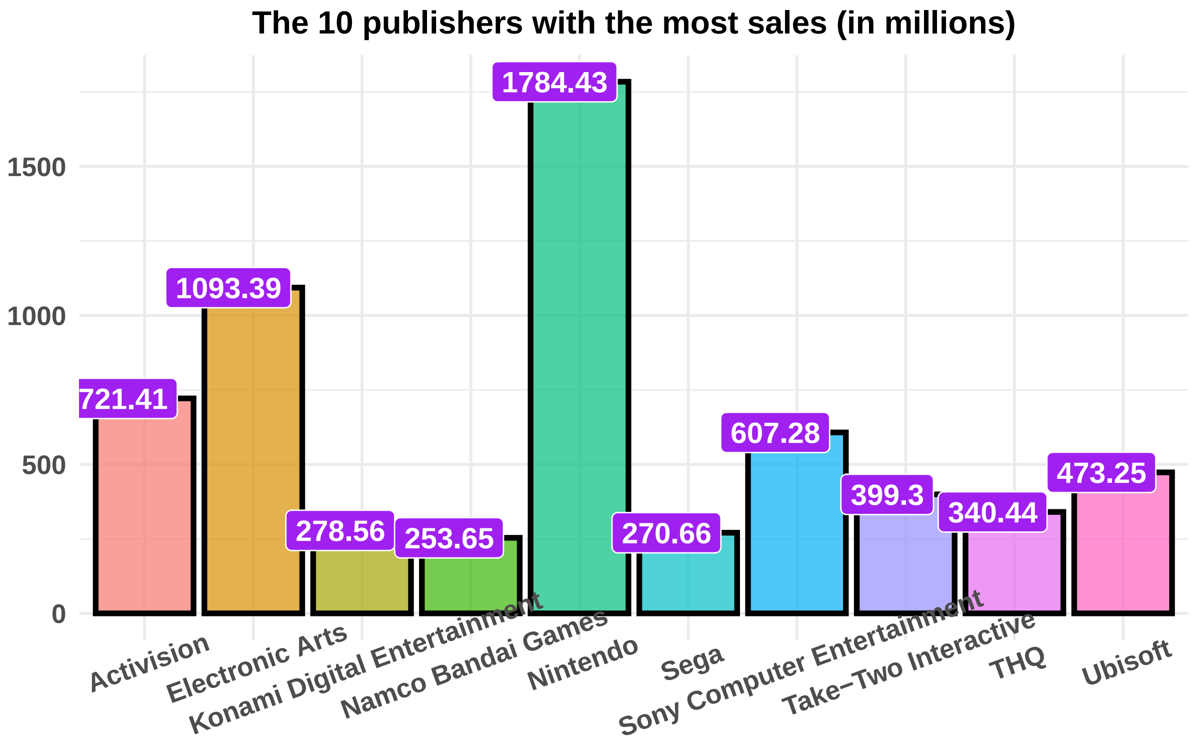


The 10 publishers with the most sales

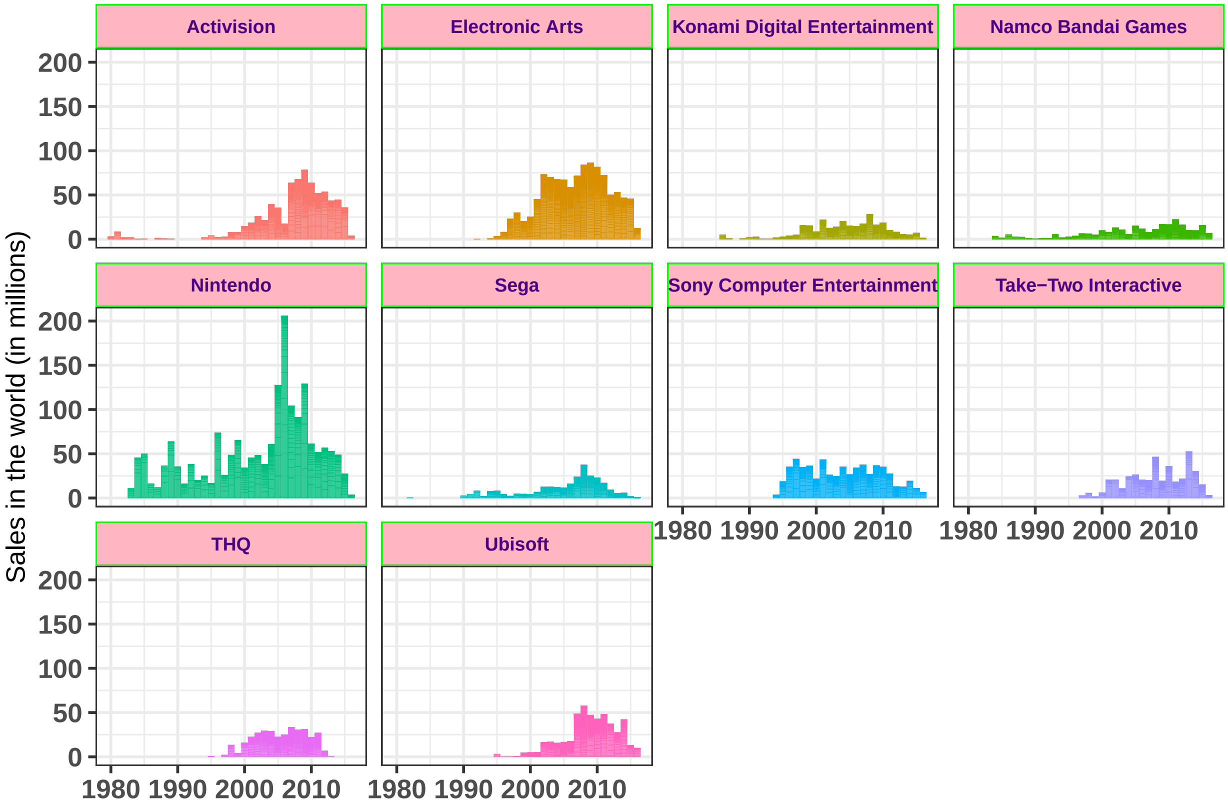
```
a <- c()

for(i in 1:nrow(pu_name_Global)){
    a <- c(a, i)
}

row.names(pu_name_Global) <- a
head(pu_name_Global, 10)
```

```
##                        Publisher Global_Sales
## 1                       Nintendo      1784.43
## 2                Electronic Arts      1093.39
## 3                     Activision       721.41
## 4    Sony Computer Entertainment       607.28
## 5                        Ubisoft       473.25
## 6           Take-Two Interactive       399.30
## 7                            THQ       340.44
## 8   Konami Digital Entertainment       278.56
## 9                           Sega       270.66
## 10            Namco Bandai Games       253.65
```

```
ggplot(data = head(pu_name_Global, 10), mapping = aes(x = Publisher,
                                                      y = Global_Sales)) +
         geom_bar(stat = "identity", mapping = aes(fill = Publisher),
                  alpha = .7, linewidth = 1, color = "black") +
         geom_label(mapping = aes(label=Global_Sales), fill = "purple",
                    size = 4, color = "white", fontface = "bold", hjust=.7) +
         ggtitle("The 10 publishers with the most sales (in millions)") +
         xlab(" ") +
         ylab("") +
         theme_minimal() +
         theme(
               plot.title = element_text(size = 12, hjust = .5, face = "bold"),
               axis.title.x = element_text(size = 12, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 12, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 10, face = "bold", angle = 20),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```
pu_top_10 <- data[data$Publisher == "Nintendo"
                  | data$Publisher == "Electronic Arts"
                  | data$Publisher == "Activision"
                  | data$Publisher == "Sony Computer Entertainment"
                  | data$Publisher == "Ubisoft"
                  | data$Publisher == "Take-Two Interactive"
                  | data$Publisher == "THQ"
                  | data$Publisher == "Konami Digital Entertainment"
                  | data$Publisher == "Sega"
                  | data$Publisher == "Namco Bandai Games", ]
pu_top_10$Year <- as.numeric(levels(pu_top_10$Year))[pu_top_10$Year]

ggplot(data = pu_top_10, mapping = aes(x = Year, y = Global_Sales)) +
    geom_bar(stat = "identity",
             mapping = aes(fill = Publisher, color = Publisher),
             linewidth = .1, alpha = .8) +
    facet_wrap(~Publisher) +
    theme_bw() +
    xlab("") +
    ylab("Sales in the world (in millions)") +
    theme(
          legend.position = "none",
          strip.text.x = element_text(margin = margin(5, 5, 5, 5),
                                      size = 7, face = "bold",
                                      color = "#4B0082"),
          strip.background = element_rect(fill = "#FFB6C1", color = "green"),
          plot.title = element_text(size = 10, face = "bold", hjust = .5),
          axis.text.x = element_text(size = 10, face = "bold"),
          axis.text.y = element_text(size = 10, face = "bold"),
          axis.title.y = element_text(size = 10))
```

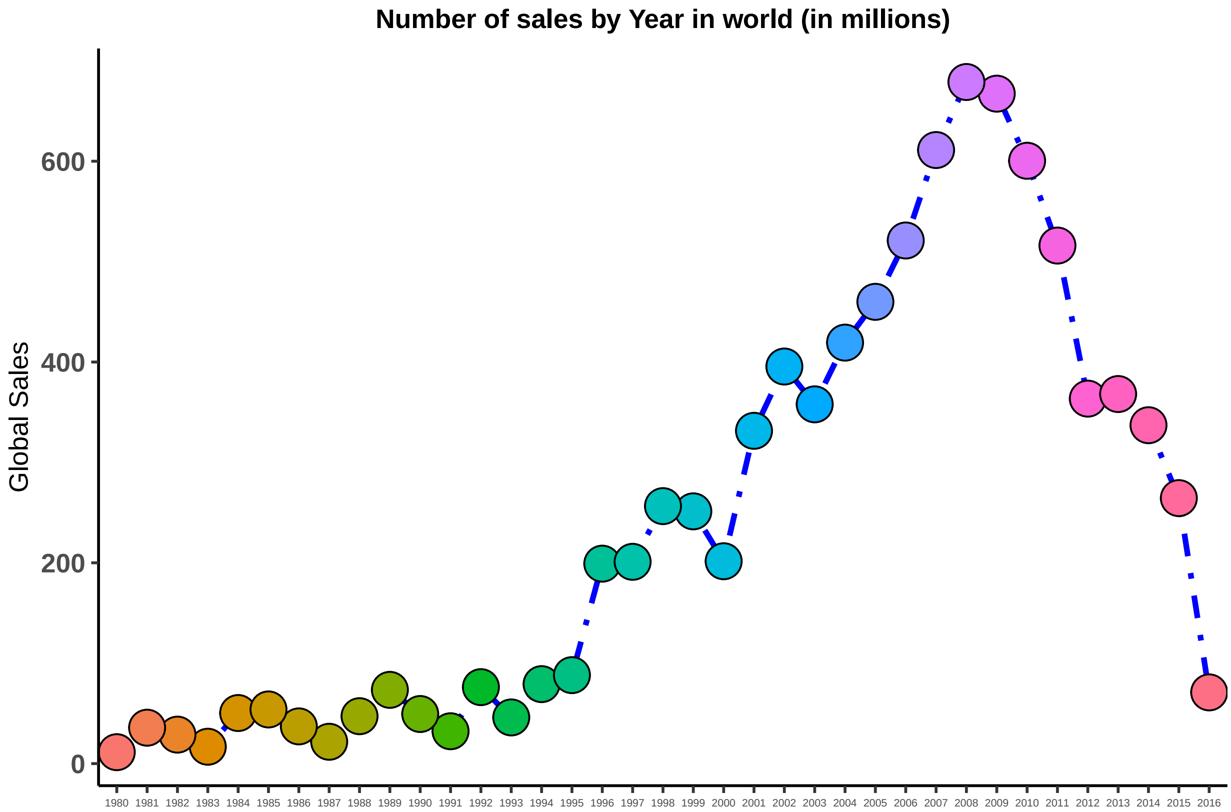

Activision
Electronic Arts
Konami Digital Entertainment
Namco Bandai Games
Nintendo
Sega
Sony Computer Entertainment
Take−Two Interactive
THQ
Ubisoft
Sales in the world (in millions)
0
50
100
150
200
1980
1990
2000
2010

1.3 E. Global Sales Number per Year

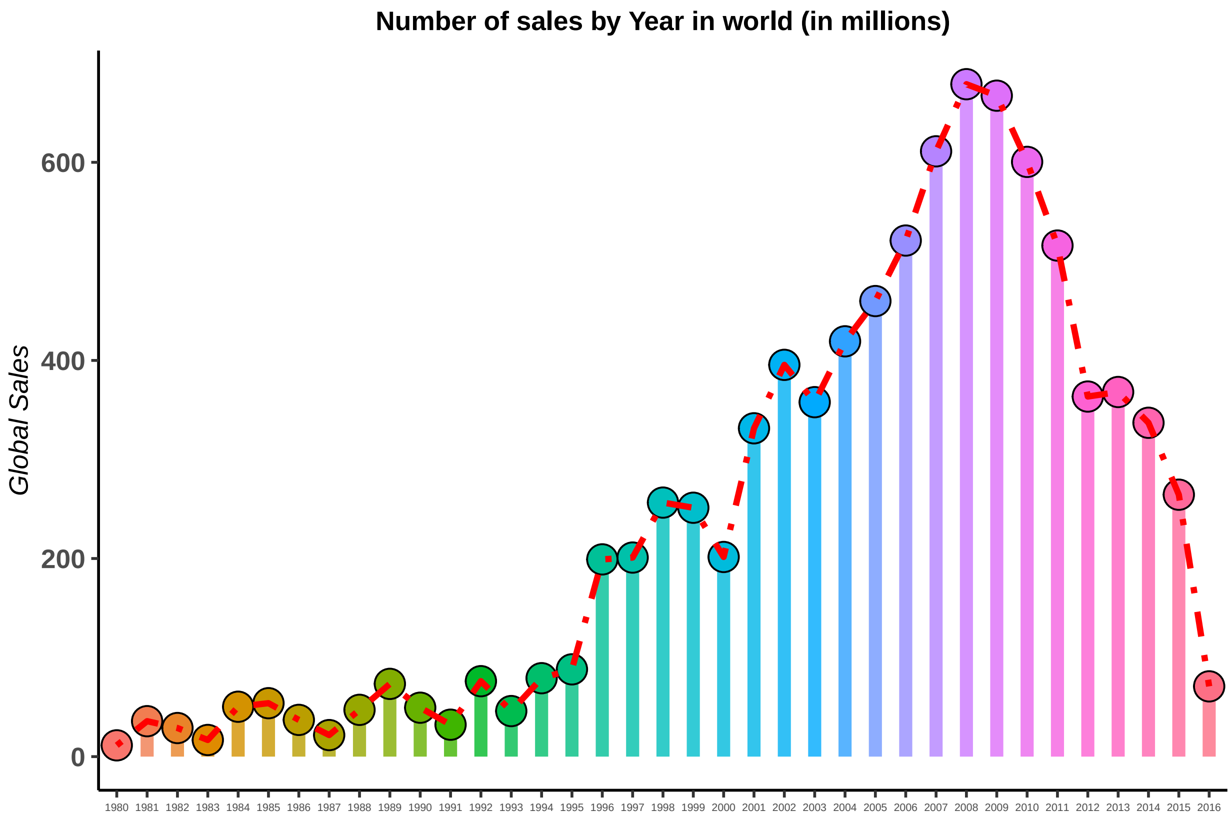
```
df_global <- aggregate(list(Global_Sales = data$Global_Sales),
                       list(Year = data$Year), sum)
df_global <- df_global[order(df_global$Global_Sales), ]

a <- c()

for(i in 1:nrow(df_global)){
    a <- c(a, i)
}

row.names(df_global) <- a
df_global
```

```
##    Year Global_Sales
## 1  1980        11.38
## 2  1983        16.79
## 3  1987        21.74
## 4  1982        28.86
## 5  1991        32.23
## 6  1981        35.77
## 7  1986        37.07
## 8  1993        45.98
## 9  1988        47.22
## 10 1990        49.39
## 11 1984        50.36
## 12 1985        53.94
## 13 2016        70.93
## 14 1989        73.45
## 15 1992        76.16
## 16 1994        79.17
## 17 1995        88.11
## 18 1996       199.15
## 19 1997       200.98
## 20 2000       201.56
## 21 1999       251.27
## 22 1998       256.47
## 23 2015       264.44
## 24 2001       331.47
## 25 2014       337.05
## 26 2003       357.85
## 27 2012       363.54
## 28 2013        368.11
## 29 2002        395.52
## 30 2004        419.31
## 31 2005        459.94
## 32 2011        515.99
## 33 2006        521.04
## 34 2010        600.45
## 35 2007        611.13
## 36 2009        667.30
## 37 2008        678.90
```

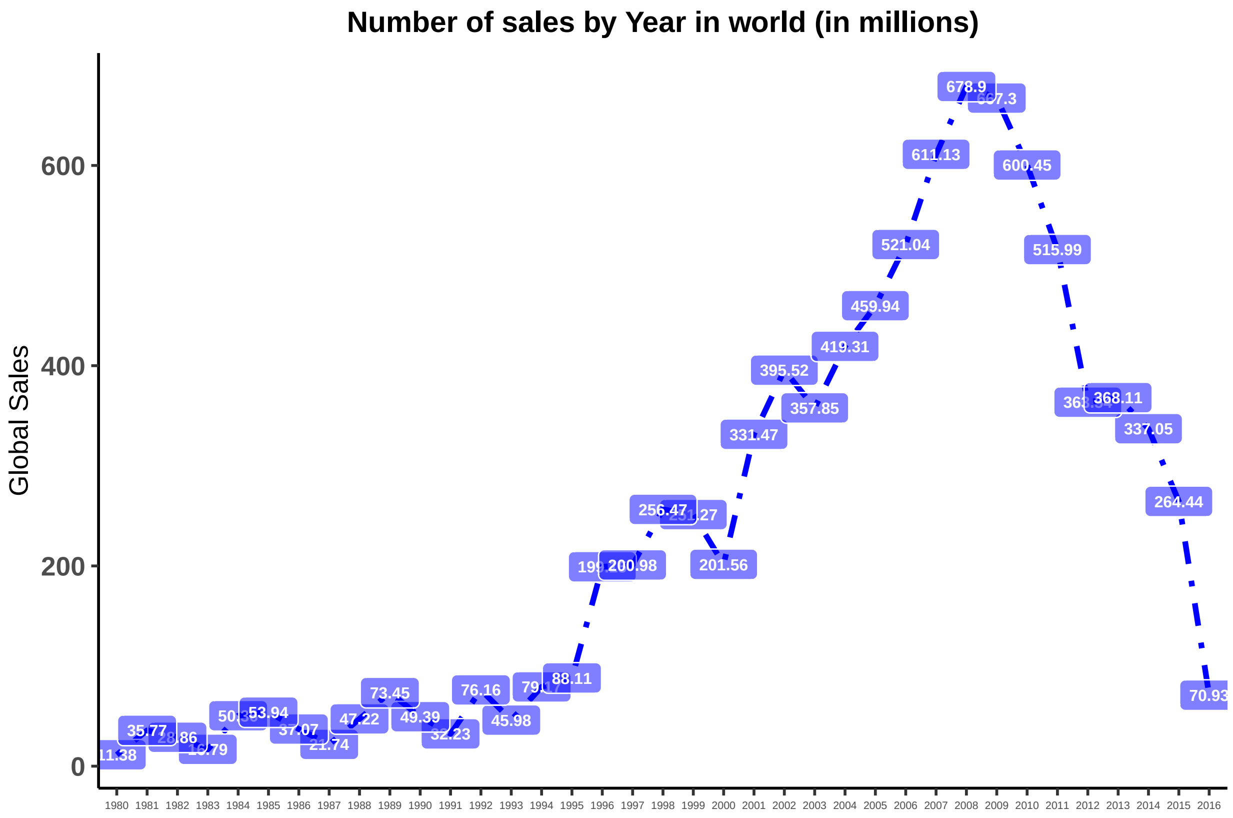
```r
ggplot(data = df_global, mapping = aes(x = Year, y = Global_Sales)) +
         geom_line(linewidth = 1, linetype = 10, color = "blue", group = 1) +
         geom_point(size = 6, shape = 21, mapping = aes(fill = Year)) +
         xlab("") +
         ylab("Global Sales") +
         ggtitle("Number of sales by Year in world (in millions)") +
         theme_classic() +
         theme(legend.position = "none",
               strip.text.x = element_text(margin = margin(7, 7, 7, 7),
                              size = 20, face = "bold", color = "#4B0082"),
               strip.background = element_rect(fill = "#FFB6C1",
                                               color = "green"),
               plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.text.x = element_text(size = 4),
               axis.text.y = element_text(size = 10, face = "bold"),
               axis.title.y = element_text(size = 10))
```



```r
ggplot(data = df_global, mapping = aes(x = Year, y = Global_Sales)) +
         geom_segment(aes(xend=Year, yend=0, color = Year),
                      linewidth = 2.3, alpha = .8) +
         geom_point(mapping = aes(fill = Year), size = 5, shape = 21) +
         geom_line(group = 1, linewidth = 1.1, linetype = 10, color = "red") +
         xlab("") +
         ylab("Global Sales") +
         ggtitle("Number of sales by Year in world (in millions)") +
         theme_classic() +
         theme(plot.title = element_text(size = 10, face = "bold", hjust = .5),
               axis.title.x = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.title.y = element_text(size = 10, hjust = .5,
                                           face = "italic"),
               axis.text.x = element_text(size = 4),
               axis.text.y = element_text(size = 10, face = "bold"),
               legend.position = "none")
```



```
ggplot(data = df_global, mapping = aes(x = Year, y = Global_Sales)) +
        geom_line(linewidth = 1, linetype = 10, color = "blue", group = 1) +
        geom_label(mapping = aes(label=Global_Sales), fill = "blue",
                   size = 2, color = "white", fontface = "bold", alpha = .5) +
        xlab("") +
        ylab("Global Sales") +
        ggtitle("Number of sales by Year in world (in millions)") +
        theme_classic() +
        theme(legend.position = "none",
             strip.text.x = element_text(margin = margin(7, 7, 7, 7),
                          size = 20, face = "bold", color = "#4B0082"),
             strip.background = element_rect(fill = "#FFB6C1",
                                        color = "green"),
             plot.title = element_text(size = 11, face = "bold", hjust = .5),
             axis.text.x = element_text(size = 4),
             axis.text.y = element_text(size = 10, face = "bold"),
             axis.title.y = element_text(size = 10))
```



2.EDA - VIDEO GAME SALES using Python

```
library(reticulate)
```

```
import numpy as np
import pandas as pd
import scipy.stats as st

import math

import matplotlib.pyplot as plt

import seaborn as sns
sns.set_style('whitegrid')

import missingno as msno

from sklearn.preprocessing import StandardScaler
from scipy import stats
```

```
data = pd.read_csv('vgsales.csv')
data = data[data['Year'] <= 2015]
data.head()
```

```
##    Rank                      Name Platform  ...  JP_Sales Other_Sales Global_Sales
## 0     1                Wii Sports      Wii  ...      3.77        8.46        82.74
## 1     2         Super Mario Bros.      NES  ...      6.81        0.77        40.24
## 2     3            Mario Kart Wii      Wii  ...      3.79        3.31        35.82
## 3     4         Wii Sports Resort      Wii  ...      3.28        2.96        33.00
## 4     5  Pokemon Red/Pokemon Blue       GB  ...     10.22        1.00        31.37
##
## [5 rows x 11 columns]
```

```
data.shape
```

```
## (15979, 11)
```

```
data.info()
```

```
## <class 'pandas.core.frame.DataFrame'>
## Int64Index: 15979 entries, 0 to 16597
## Data columns (total 11 columns):
##  #   Column        Non-Null Count  Dtype
## ---  ------        --------------  -----
##  0   Rank          15979 non-null  int64
##  1   Name          15979 non-null  object
##  2   Platform      15979 non-null  object
##  3   Year          15979 non-null  float64
##  4   Genre         15979 non-null  object
##  5   Publisher     15945 non-null  object
##  6   NA_Sales      15979 non-null  float64
##  7   EU_Sales      15979 non-null  float64
##  8   JP_Sales      15979 non-null  float64
##  9   Other_Sales   15979 non-null  float64
##  10  Global_Sales  15979 non-null  float64
## dtypes: float64(6), int64(1), object(4)
## memory usage: 1.5+ MB
```

```
data.isnull().sum()
```

```
## Rank             0
## Name             0
## Platform         0
## Year             0
## Genre            0
## Publisher       34
## NA_Sales         0
## EU_Sales         0
## JP_Sales         0
## Other_Sales      0
## Global_Sales     0
## dtype: int64
```

```
data = data[data['Publisher'].isnull()!=True]
data.isnull().sum()
```

```
## Rank            0
## Name            0
## Platform        0
## Year            0
## Genre           0
## Publisher       0
## NA_Sales        0
## EU_Sales        0
## JP_Sales        0
## Other_Sales     0
## Global_Sales    0
## dtype: int64
```

```
data.info()
```

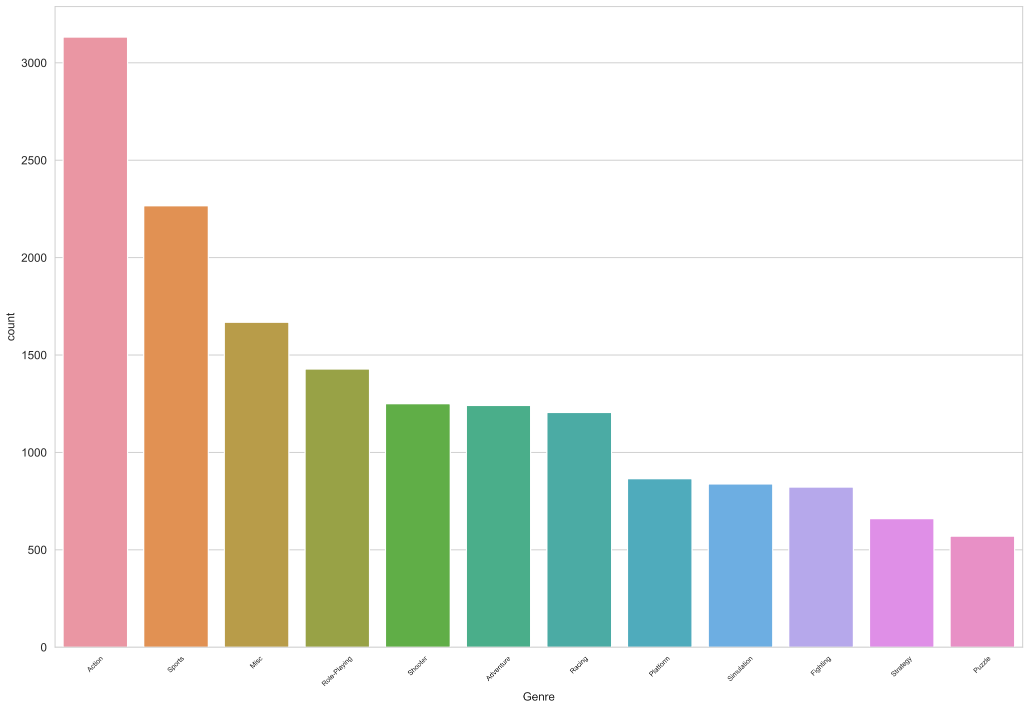
```
## <class 'pandas.core.frame.DataFrame'>
## Int64Index: 15945 entries, 0 to 16597
## Data columns (total 11 columns):
##  #   Column        Non-Null Count  Dtype  
## ---  ------        --------------  -----  
##  0   Rank          15945 non-null  int64  
##  1   Name          15945 non-null  object 
##  2   Platform      15945 non-null  object 
##  3   Year          15945 non-null  float64
##  4   Genre         15945 non-null  object 
##  5   Publisher     15945 non-null  object 
##  6   NA_Sales      15945 non-null  float64
##  7   EU_Sales      15945 non-null  float64
##  8   JP_Sales      15945 non-null  float64
##  9   Other_Sales   15945 non-null  float64
##  10  Global_Sales  15945 non-null  float64
## dtypes: float64(6), int64(1), object(4)
## memory usage: 1.5+ MB
```

2.1 What genre games have been made the most?

```
data['Genre'].value_counts()
```

```
## Action          3132
## Sports          2266
## Misc            1668
## Role-Playing    1428
## Shooter         1250
## Adventure       1241
## Racing          1205
## Platform         865
## Simulation       838
## Fighting         822
## Strategy         660
## Puzzle           570
## Name: Genre, dtype: int64
```

```
plt.figure(figsize=(15, 10))
sns.countplot(x="Genre", data=data, order = data['Genre'].value_counts().index)
plt.xticks(rotation=45,fontsize=6)
plt.show()
```

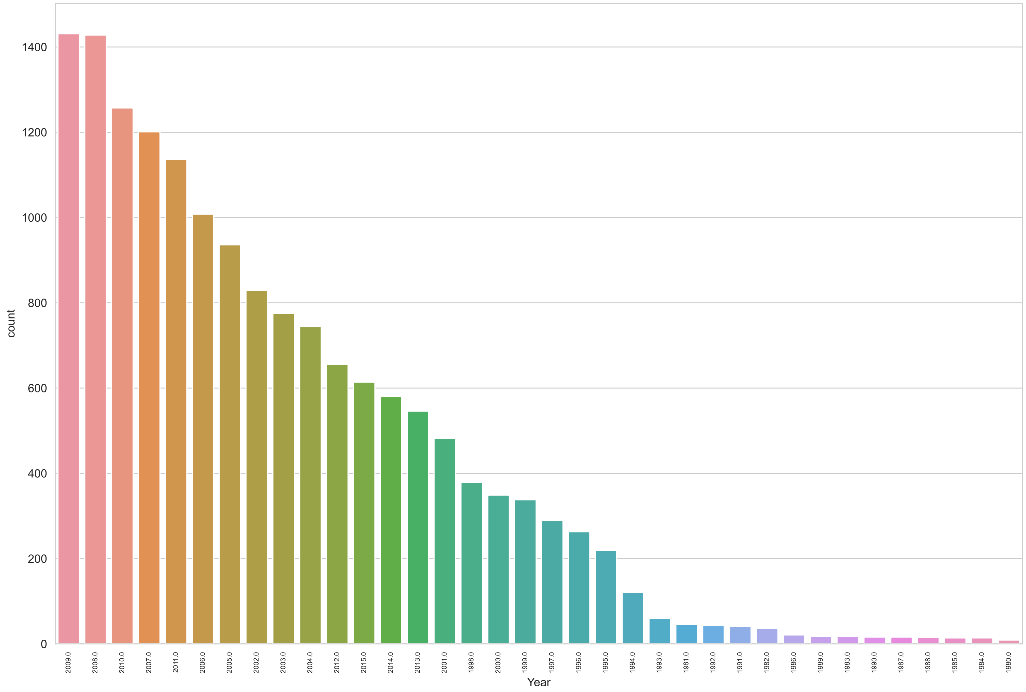


Answer is => “Action” and “Sports”

2.2 Which years had the most game release?

```
plt.figure(figsize=(15, 10))
sns.countplot(x="Year", data=data,
order = data.groupby(by=['Year'])['Name'].
count().sort_values(ascending=False).index)
plt.xticks(rotation=90,fontsize=6)
# data.groupby(by=['Year'])['Name'].count().sort_values(ascending=False)
```

```
plt.show()
```



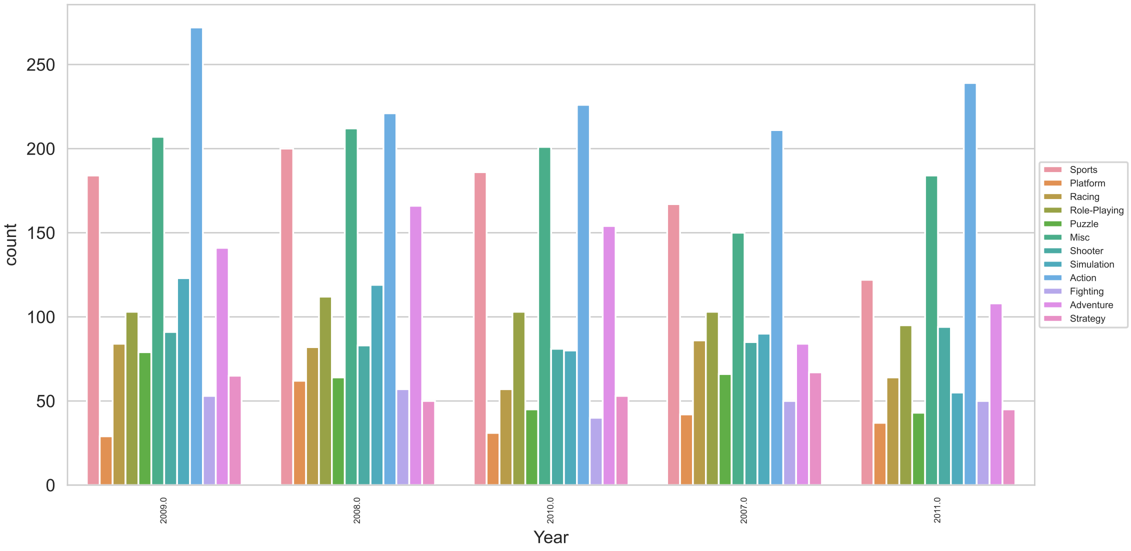
Answer is =>

1. **2009.0 -> 1431**
2. **2008.0 -> 1428**
3. **2010.0 -> 1257**
4. **2007.0 -> 1201**
5. **2011.0 -> 1136**

2.3 Top 5 years games release by genre

```
plt.figure(figsize=(10, 5))
sns.countplot(x="Year", data=data, hue='Genre',
order=data.Year.value_counts().iloc[:5].index)
# Move the legend outside the plot area
plt.legend(loc='center left', bbox_to_anchor=(1, 0.5), fontsize=5.5)
plt.xticks(fontsize=5, rotation=90)
# data.Year.value_counts().iloc[:5]
```

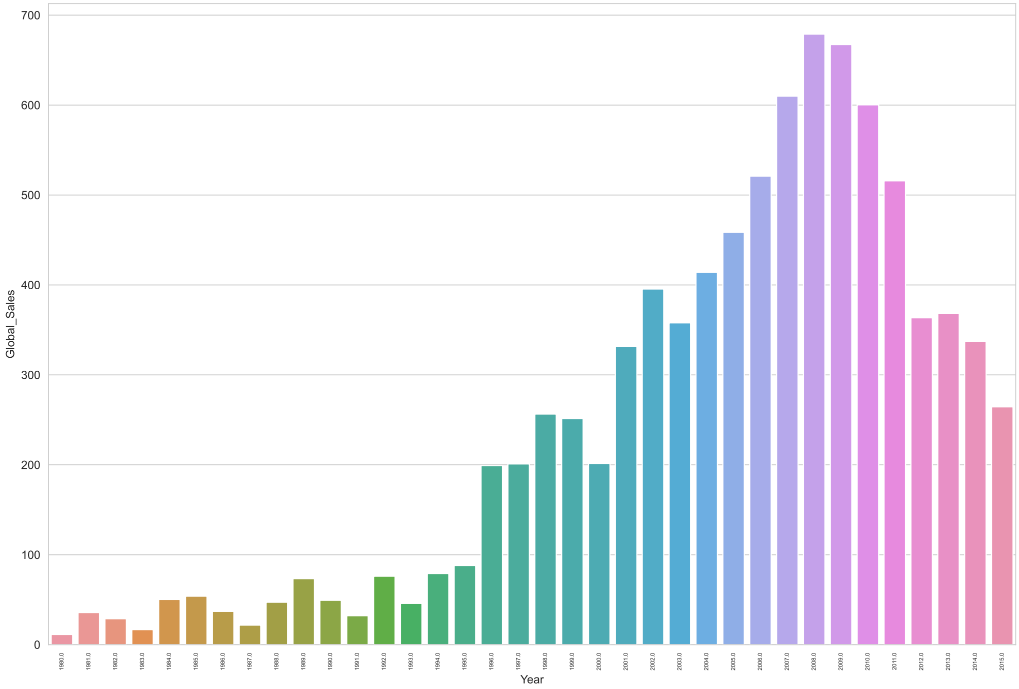
```
plt.show()
```



2.4 Which years had the highest sales worldwide?

```
data_year = data.groupby(by=['Year'])['Global_Sales'].sum()
data_year = data_year.reset_index()
# data_year.sort_values(by=['Global_Sales'], ascending=False)
```

```
plt.figure(figsize=(15, 10))
sns.barplot(x="Year", y="Global_Sales", data=data_year)
plt.xticks(rotation=90,fontsize=5)
plt.show()
```

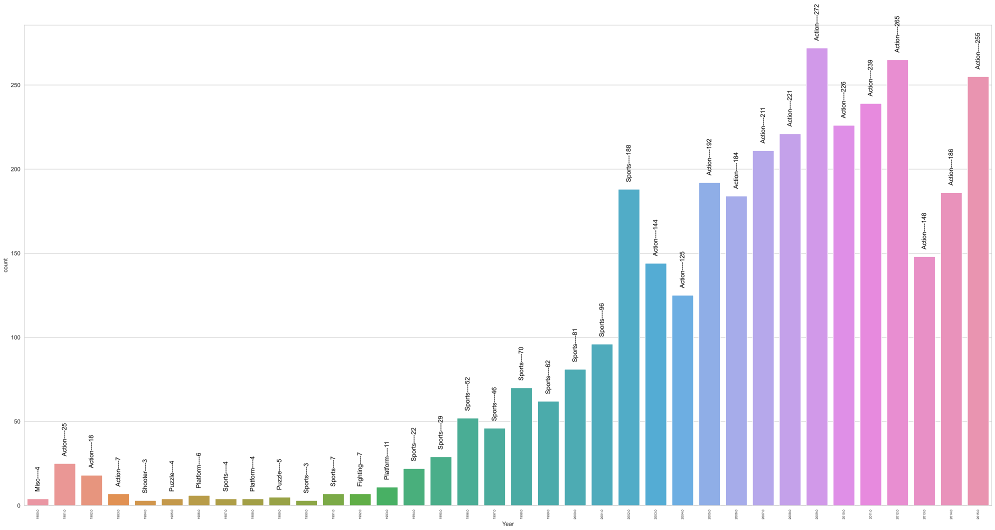


Answer is =>

1. **2008.0 -> 678.90**
2. **2009.0 -> 667.30**
3. **2007.0 -> 609.92**
4. **2010.0 -> 600.29**
5. **2006.0 -> 521.04**

2.5 Which genre game has been released the most in a single year?

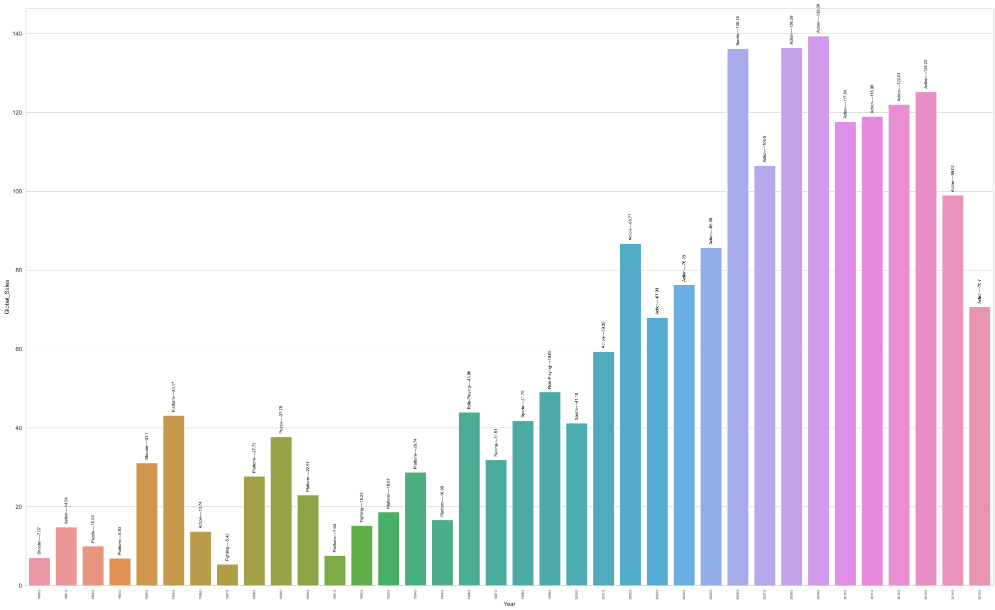
```python
year_max_df = data.groupby(['Year', 'Genre']).size().reset_index(name='count')
year_max_idx = year_max_df\
.groupby(['Year'])['count'].transform(max) == year_max_df['count']
year_max_genre = year_max_df[year_max_idx].reset_index(drop=True)
year_max_genre = year_max_genre.drop_duplicates(subset=["Year", "count"],
keep='last').reset_index(drop=True)
# year_max_genre
genre = year_max_genre['Genre'].values
# genre[0]
plt.figure(figsize=(30, 15))
g = sns.barplot(x='Year', y='count', data=year_max_genre)
index = 0
for value in year_max_genre['count'].values:
    #print(asd)
    g.text(index, value + 5, str(genre[index] + '----' +str(value)),
    color='#000', size=11, rotation= 90, ha="center")
    index += 1
plt.xticks(rotation=90,fontsize=5.5)
plt.show()
```



Answer is =>
2009 Action —> 272
2012 Action —> 266

2.6 Which genre game has been sold the most in a single year?

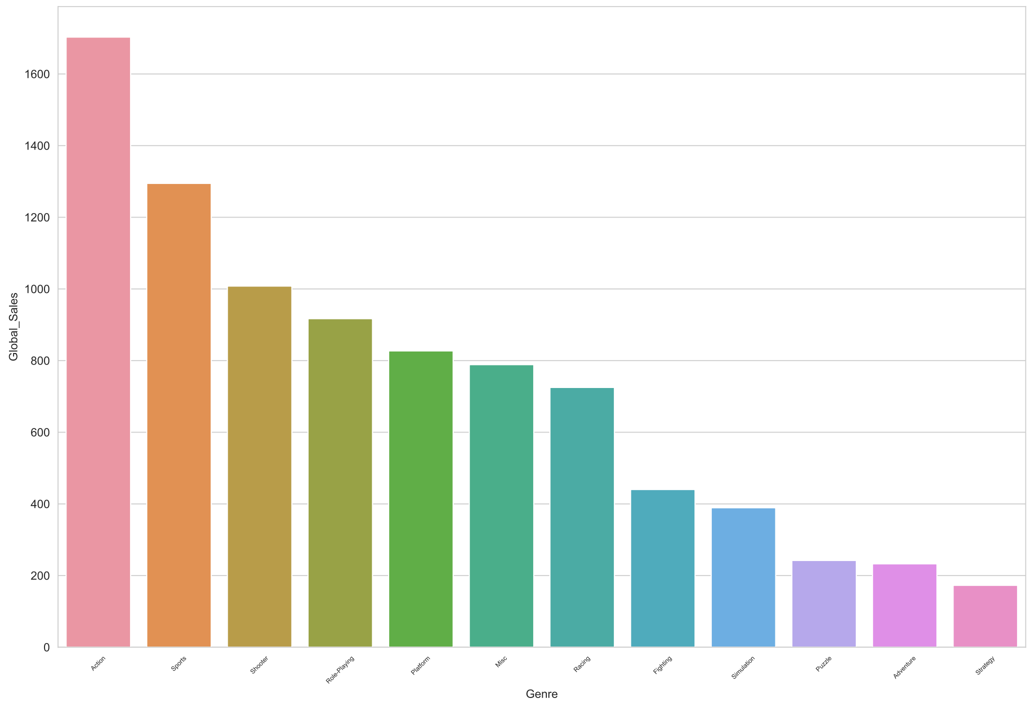
```
year_sale_dx = data.groupby(by=['Year', 'Genre'])\
['Global_Sales'].sum().reset_index()
year_sale = year_sale_dx.groupby(by=['Year'])\
['Global_Sales'].transform(max) == year_sale_dx['Global_Sales']
year_sale_max = year_sale_dx[year_sale].reset_index(drop=True)
# year_sale_max
genre = year_sale_max['Genre']
plt.figure(figsize=(30, 18))
g = sns.barplot(x='Year', y='Global_Sales', data=year_sale_max)
index = 0
for value in year_sale_max['Global_Sales']:
    g.text(index, value + 1, str(genre[index] + '----' +str(round(value, 2))),
    color='#000', size=7, rotation= 90, ha="center")
    index += 1
plt.xticks(rotation=90,fontsize=5.5)
plt.show()
```



Answer is =>
2009 Action —> 139.36 million
2008 Action —> 136.39 miliion

2.7 Which genre games have the highest sale price globally?

```
data_genre = data.groupby(by=['Genre'])['Global_Sales'].sum()
data_genre = data_genre.reset_index()
data_genre = data_genre.sort_values(by=['Global_Sales'], ascending=False)
# data_genre
plt.figure(figsize=(15, 10))
sns.barplot(x="Genre", y="Global_Sales", data=data_genre)
plt.xticks(rotation=45,fontsize=5.5)
plt.show()
```

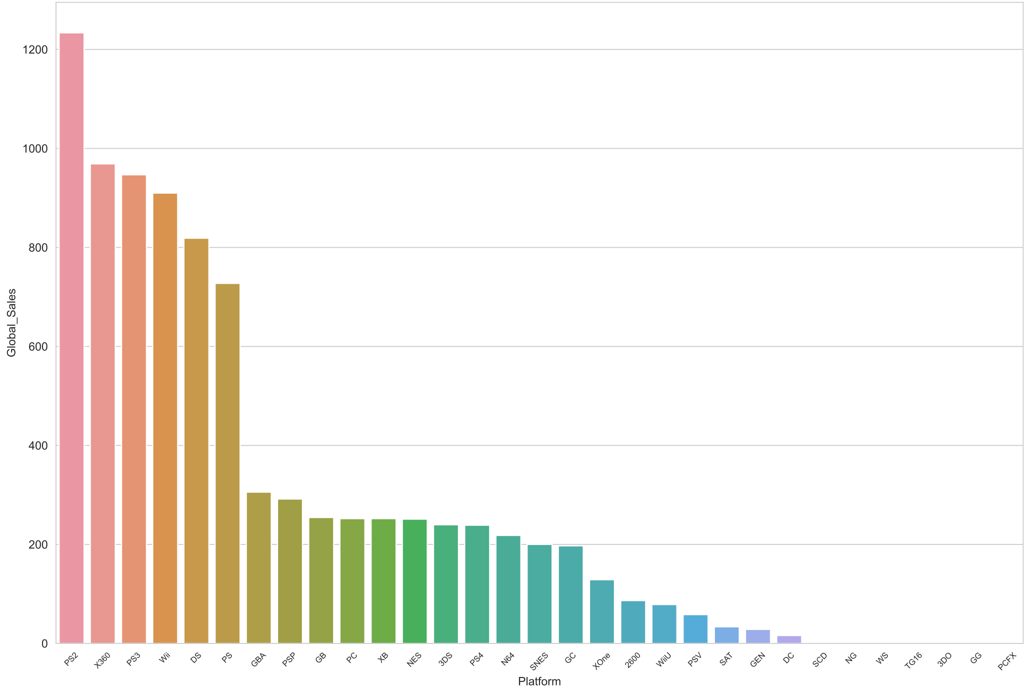


Action and Sports are always in top

2.8 Which platfrom has the highest sale price globally?

```
data_platform = data.groupby(by=['Platform'])['Global_Sales'].sum()
data_platform = data_platform.reset_index()
data_platform = data_platform.sort_values(by=['Global_Sales'], ascending=False)
# data_platform
plt.figure(figsize=(15, 10))
sns.barplot(x="Platform", y="Global_Sales", data=data_platform)
plt.xticks(rotation=45,fontsize=7)
```

```
plt.show()
```



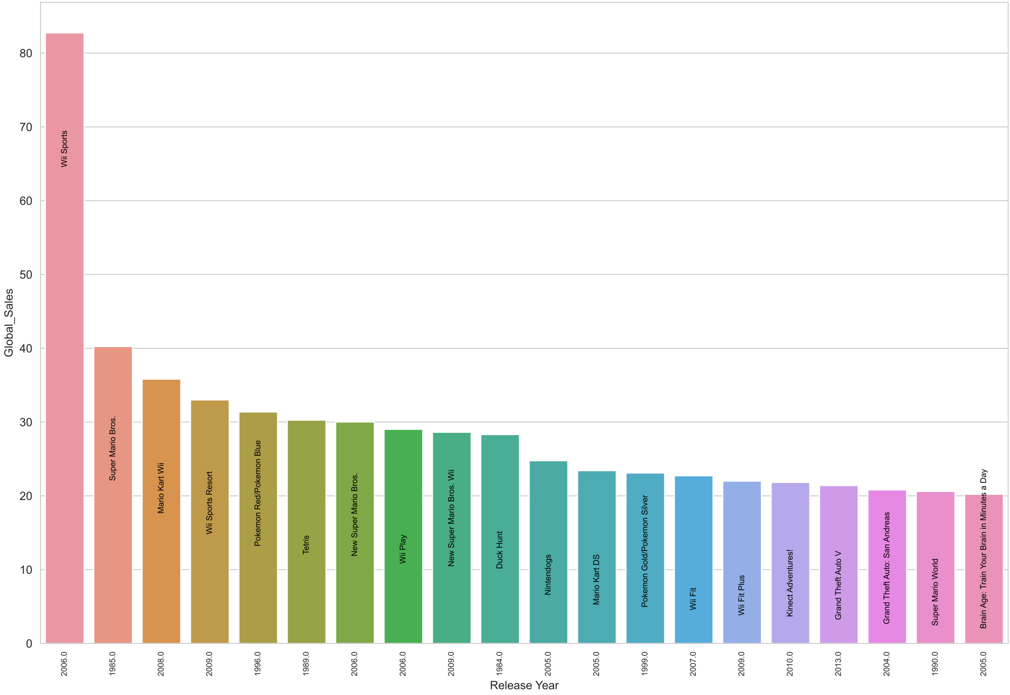
The winner is PS2

2.9 Which individual game has the highest sale price globally?

The winner is Wii Sports

```
top_game_sale = data.head(20)
top_game_sale = top_game_sale[['Name', 'Year', 'Genre', 'Global_Sales']]
top_game_sale = top_game_sale.sort_values(by=['Global_Sales'], ascending=False)
# top_game_sale
name = top_game_sale['Name']
year = top_game_sale['Year']
y = np.arange(0, 20)
plt.figure(figsize=(15, 10))
g = sns.barplot(x='Name', y='Global_Sales', data=top_game_sale)
index = 0
for value in top_game_sale['Global_Sales']:
    g.text(index, value - 18, name[index],
    color='#000', size=7, rotation= 90, ha="center")
    index += 1
plt.xlabel('Release Year')
plt.xticks(y, top_game_sale['Year'], fontsize=7, rotation=90)
```

```
plt.show()
```

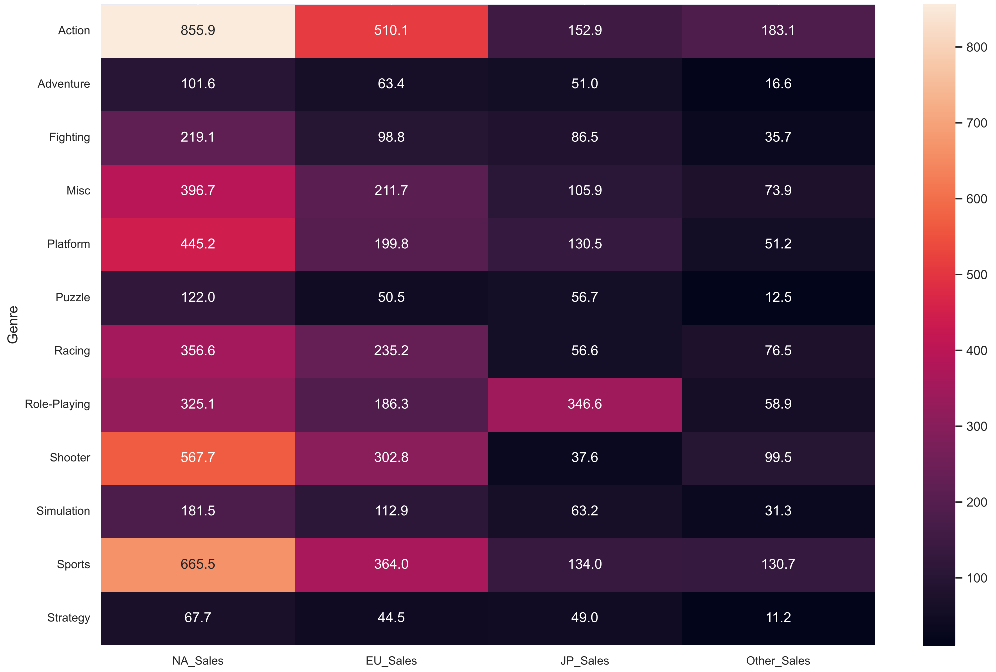


2.10 Sales comparison by genre

```
comp_genre = data[['Genre', 'NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales']]
# comp_genre
comp_map = comp_genre.groupby(by=['Genre']).sum()
# comp_map
plt.figure(figsize=(15, 10))
sns.set(font_scale=1)
sns.heatmap(comp_map, annot=True, fmt = '.1f')
plt.xticks(fontsize=10)
```

```
plt.yticks(fontsize=10)
```

```
plt.show()
```

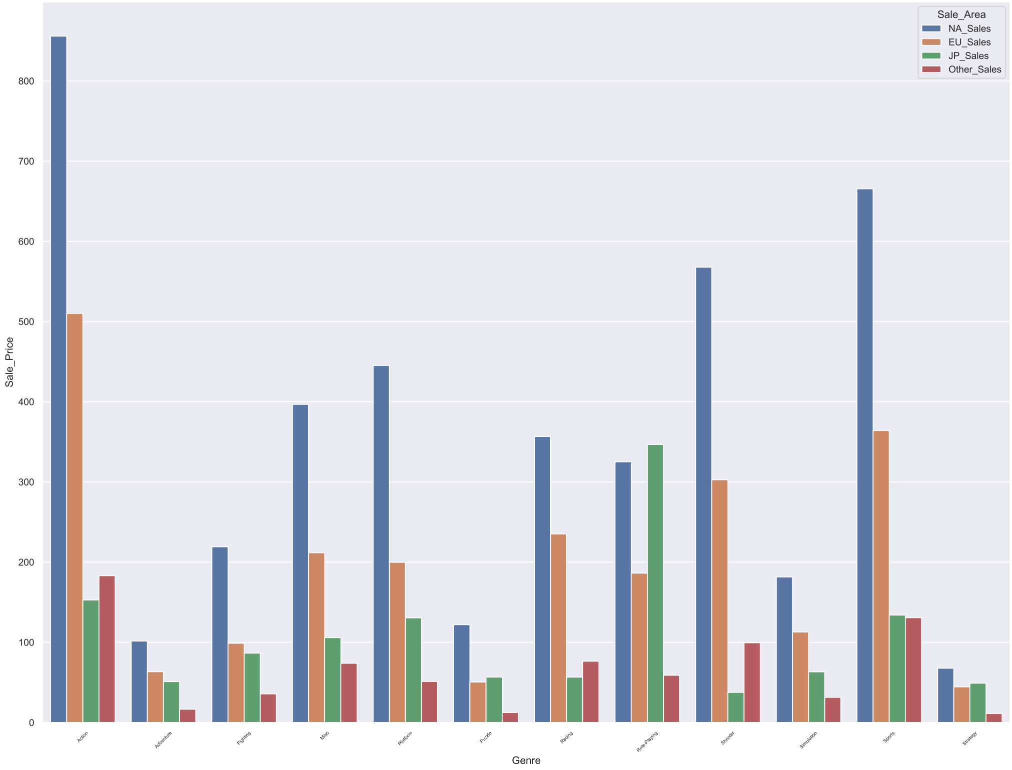


```
comp_table = comp_map.reset_index()
comp_table = pd.melt(comp_table, id_vars=['Genre'],
value_vars=['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales'],
var_name='Sale_Area', value_name='Sale_Price')
comp_table.head()
```

```
##         Genre Sale_Area  Sale_Price
## 0      Action  NA_Sales      855.90
## 1   Adventure  NA_Sales      101.59
## 2    Fighting  NA_Sales      219.14
## 3        Misc  NA_Sales      396.70
## 4    Platform  NA_Sales      445.20
```

```
plt.figure(figsize=(20, 15))
sns.barplot(x='Genre', y='Sale_Price', hue='Sale_Area', data=comp_table)
plt.xticks(rotation=45,fontsize=5.5)
```

```
plt.show()
```



2.11 Sales comparison by platform

```
comp_platform = data[['Platform', 'NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales']]
comp_platform.head()
```

```
##   Platform  NA_Sales  EU_Sales  JP_Sales  Other_Sales
## 0      Wii     41.49     29.02      3.77         8.46
## 1      NES     29.08      3.58      6.81         0.77
## 2      Wii     15.85     12.88      3.79         3.31
## 3      Wii     15.75     11.01      3.28         2.96
## 4       GB     11.27      8.89     10.22         1.00
```

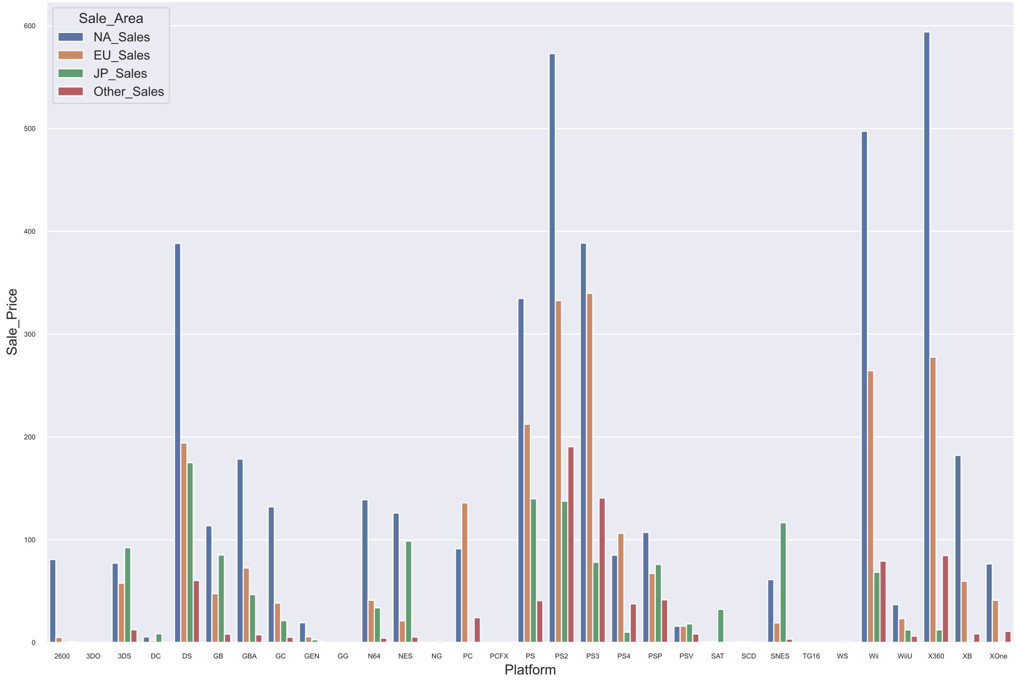
```
comp_platform = comp_platform.groupby(by=['Platform']).sum().reset_index()
# comp_table = comp_map.reset_index()
comp_table = pd.melt(comp_platform, id_vars=['Platform'],
value_vars=['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales'],
var_name='Sale_Area', value_name='Sale_Price')
comp_table.head()
```

```
##   Platform Sale_Area  Sale_Price
## 0     2600  NA_Sales       80.78
## 1      3DO  NA_Sales        0.00
## 2      3DS  NA_Sales       77.20
## 3       DC  NA_Sales        5.43
## 4       DS  NA_Sales      388.26
```

```
plt.figure(figsize=(15, 10))
sns.barplot(x='Platform', y='Sale_Price', hue='Sale_Area', data=comp_table)
plt.xticks(fontsize=6)
```

```
plt.yticks(fontsize=6)
```

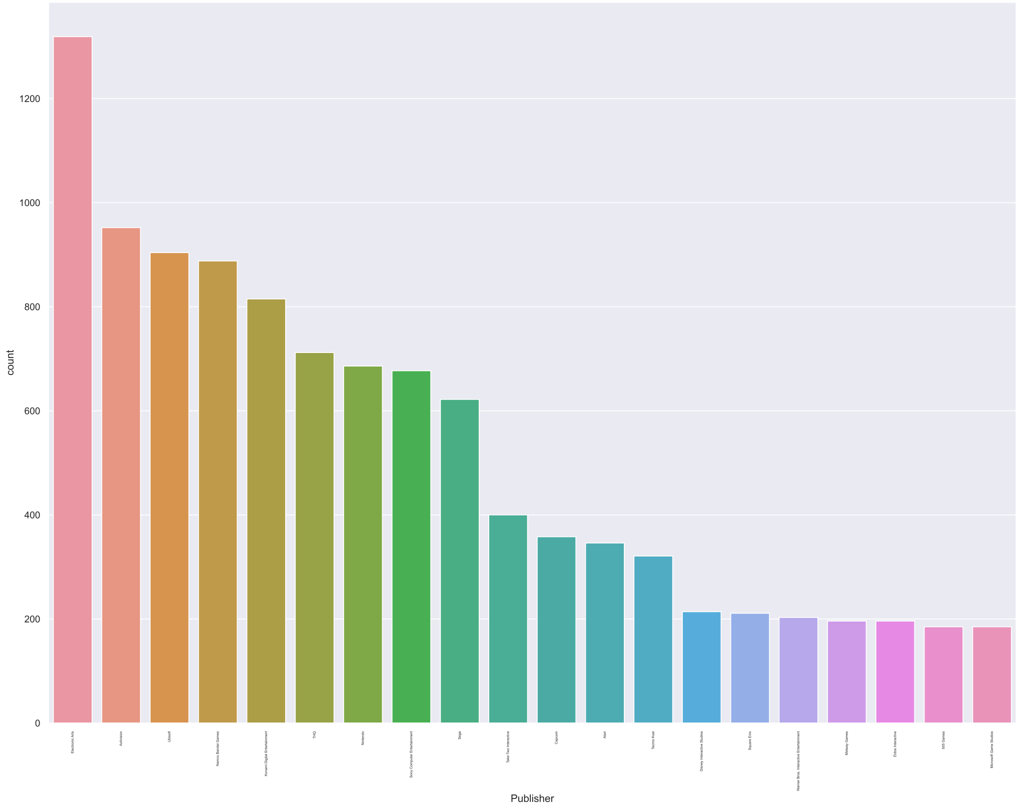
```
plt.show()
```



2.12 Top 20 Publisher

```
top_publisher = data.groupby(by=['Publisher'])\
['Year'].count().sort_values(ascending=False).head(20)
top_publisher = pd.DataFrame(top_publisher).reset_index()
# top_publisher
plt.figure(figsize=(20, 15))
sns.countplot(x="Publisher", data=data,
order = data.groupby(by=['Publisher'])\
['Year'].count().sort_values(ascending=False).iloc[:20].index)
plt.xticks(rotation=90,fontsize=4)
```

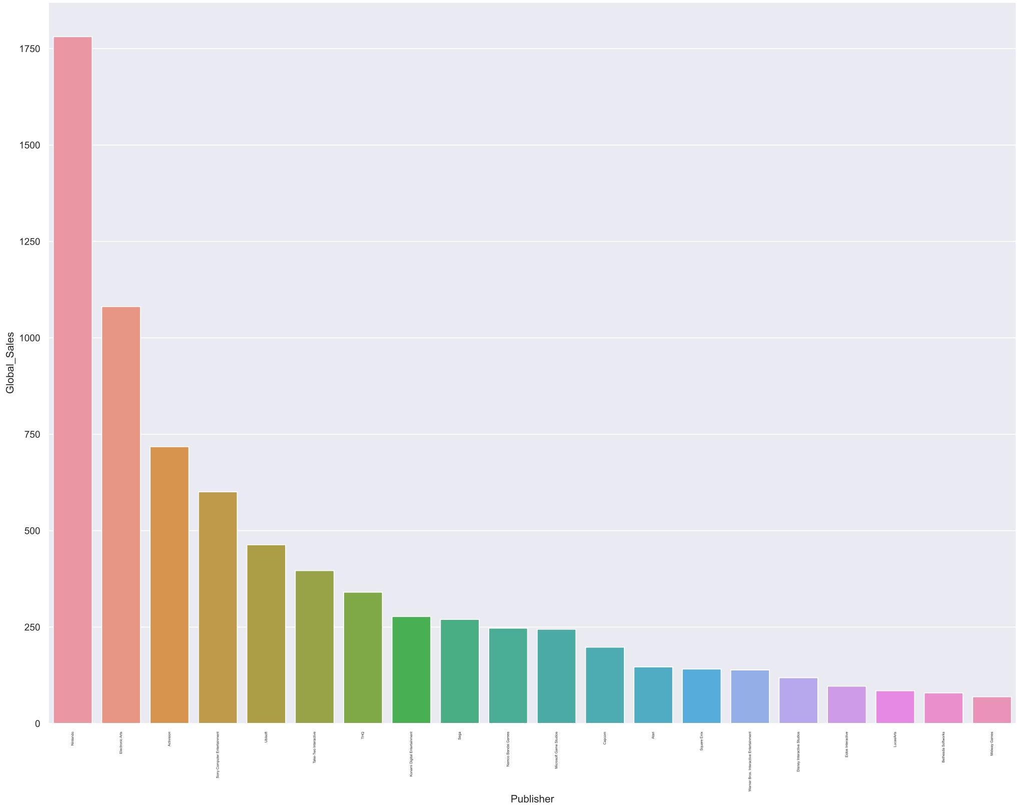
```
plt.show()
```



2.13 Top global sales by publisher

```
sale_pbl = data[['Publisher', 'Global_Sales']]
sale_pbl = sale_pbl.groupby('Publisher')\
['Global_Sales'].sum().sort_values(ascending=False).head(20)
sale_pbl = pd.DataFrame(sale_pbl).reset_index()
# sale_pbl
plt.figure(figsize=(20, 15))
sns.barplot(x='Publisher', y='Global_Sales', data=sale_pbl)
plt.xticks(rotation=90,fontsize=4)
```

```
plt.show()
```



2.14 Publisher comparison

```
comp_publisher = data[['Publisher',
'NA_Sales', 'EU_Sales',
'JP_Sales', 'Other_Sales',
'Global_Sales']]
comp_publisher.head()
```

```
##   Publisher  NA_Sales  EU_Sales  JP_Sales  Other_Sales  Global_Sales
## 0  Nintendo     41.49     29.02      3.77         8.46         82.74
## 1  Nintendo     29.08      3.58      6.81         0.77         40.24
## 2  Nintendo     15.85     12.88      3.79         3.31         35.82
## 3  Nintendo     15.75     11.01      3.28         2.96         33.00
## 4  Nintendo     11.27      8.89     10.22         1.00         31.37
```

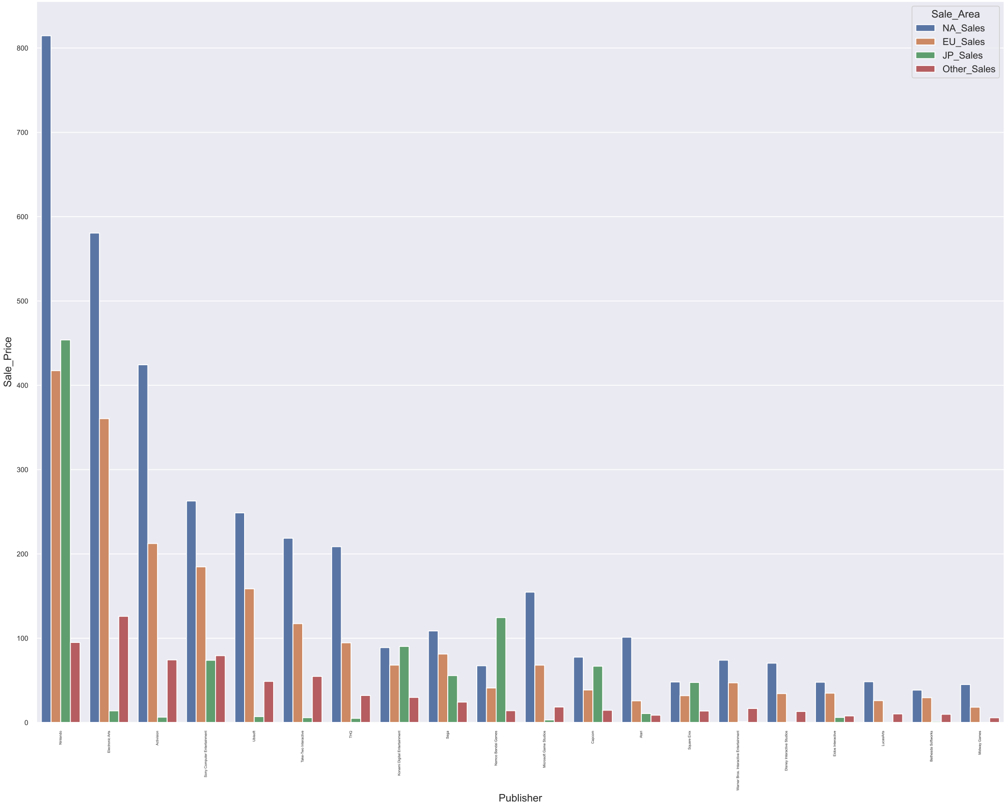
```
comp_publisher = comp_publisher.groupby(by=['Publisher'])\
.sum().reset_index().sort_values(by=['Global_Sales'], ascending=False)
comp_publisher = comp_publisher.head(20)
# comp_publisher
comp_publisher = pd.melt(comp_publisher, id_vars=['Publisher'],\
value_vars=['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales'],\
var_name='Sale_Area', value_name='Sale_Price')
comp_publisher
```

```
##                       Publisher    Sale_Area  Sale_Price
## 0                      Nintendo     NA_Sales      814.59
## 1               Electronic Arts     NA_Sales      580.58
## 2                    Activision     NA_Sales      424.34
## 3   Sony Computer Entertainment     NA_Sales      262.79
## 4                       Ubisoft     NA_Sales      248.69
## ..                          ...          ...         ...
## 75   Disney Interactive Studios  Other_Sales       13.15
## 76            Eidos Interactive  Other_Sales        7.90
## 77                    LucasArts  Other_Sales       10.28
## 78           Bethesda Softworks  Other_Sales        9.81
## 79                 Midway Games  Other_Sales        5.69
##
## [80 rows x 3 columns]
```

```
plt.figure(figsize=(20, 15))
sns.barplot(x='Publisher', y='Sale_Price',
hue='Sale_Area', data=comp_publisher)
plt.xticks(fontsize=4, rotation=90)
```

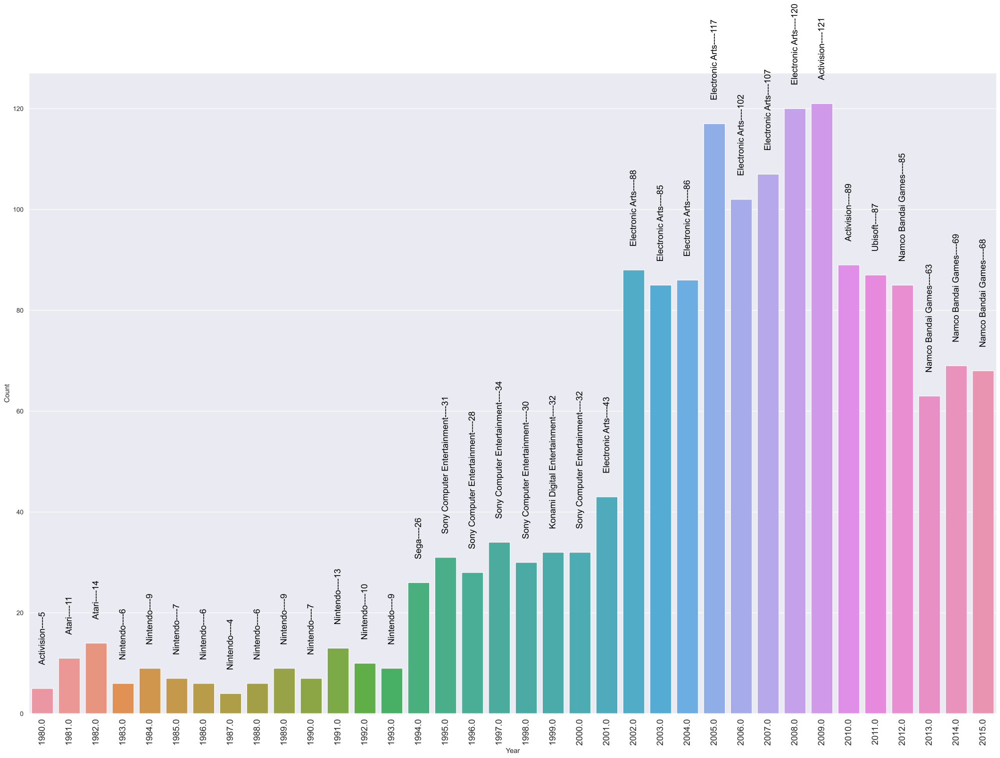
```
plt.yticks(fontsize=8)
```

```
plt.show()
```



2.15 Top publisher by count each year

```
top_publisher =  data[['Year', 'Publisher']]
top_publisher_df = top_publisher.groupby(by=['Year', 'Publisher'])\
.size().reset_index(name='Count')
top_publisher_idx =  top_publisher_df.groupby(by=['Year'])\
['Count'].transform(max) == top_publisher_df['Count']
top_publisher_count = top_publisher_df[top_publisher_idx]\
.reset_index(drop=True)
top_publisher_count  = top_publisher_count.\
drop_duplicates(subset=["Year", "Count"], keep='last').reset_index(drop=True)
# top_publisher_count
publisher= top_publisher_count['Publisher']
plt.figure(figsize=(30, 20))
g = sns.barplot(x='Year', y='Count', data=top_publisher_count)
index = 0
for value in top_publisher_count['Count'].values:
#     print(asd)
    g.text(index, value + 5, str(publisher[index] + '----' +str(value)),
    color='#000', size=15, rotation= 90, ha="center")
    index += 1
plt.xticks(rotation=90,fontsize=15)
plt.show()
```

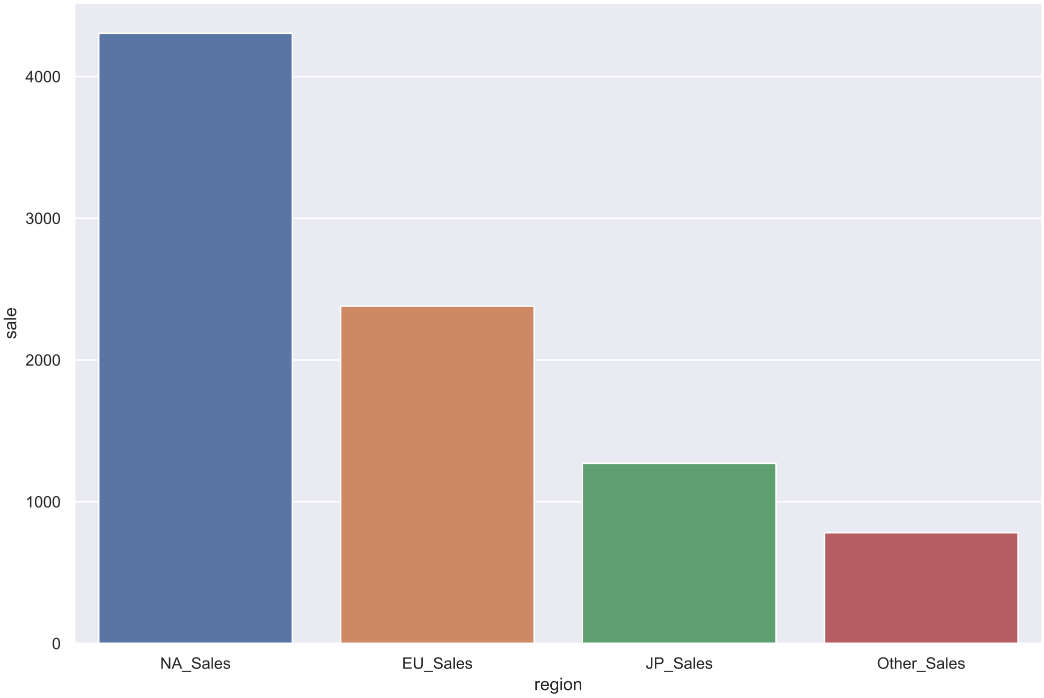

Count
0
20
40
60
80
100
120
Year
1980.0
1981.0
1982.0
1983.0
1984.0
1985.0
1986.0
1987.0
1988.0
1989.0
1990.0
1991.0
1992.0
1993.0
1994.0
1995.0
1996.0
1997.0
1998.0
1999.0
2000.0
2001.0
2002.0
2003.0
2004.0
2005.0
2006.0
2007.0
2008.0
2009.0
2010.0
2011.0
2012.0
2013.0
2014.0
2015.0
Activision----5
Atari----11
Atari----14
Nintendo----6
Nintendo----9
Nintendo----7
Nintendo----6
Nintendo----4
Nintendo----6
Nintendo----9
Nintendo----7
Nintendo----13
Nintendo----10
Nintendo----9
Sega----26
Sony Computer Entertainment----31
Sony Computer Entertainment----28
Sony Computer Entertainment----34
Sony Computer Entertainment----30
Konami Digital Entertainment----32
Sony Computer Entertainment----32
Electronic Arts----43
Electronic Arts----88
Electronic Arts----85
Electronic Arts----86
Electronic Arts----117
Electronic Arts----102
Electronic Arts----107
Electronic Arts----120
Activision----121
Activision----89
Ubisoft----87
Namco Bandai Games----85
Namco Bandai Games----63
Namco Bandai Games----69
Namco Bandai Games----68

2.16 Total revenue by region

```
top_sale_reg = data[['NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales']]
# pd.DataFrame(top_sale_reg.sum(), columns=['a', 'b'])
top_sale_reg = top_sale_reg.sum().reset_index()
top_sale_reg = top_sale_reg.rename(columns={"index": "region", 0: "sale"})
top_sale_reg
```

```
##         region     sale
## 0     NA_Sales  4304.72
## 1     EU_Sales  2379.93
## 2     JP_Sales  1270.55
## 3  Other_Sales   781.14
```

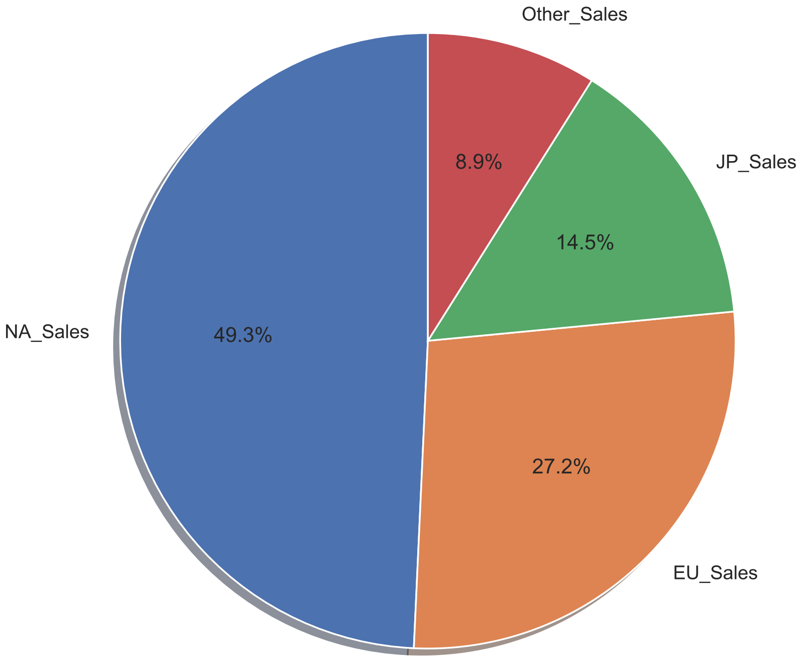
```
plt.figure(figsize=(12, 8))
sns.barplot(x='region', y='sale', data = top_sale_reg)
```



```
plt.show()
```

```
labels = top_sale_reg['region']
sizes = top_sale_reg['sale']
plt.figure(figsize=(10, 8))
plt.pie(sizes, labels=labels, autopct='%1.1f%%', shadow=True, startangle=90)
```

```
plt.show()
```

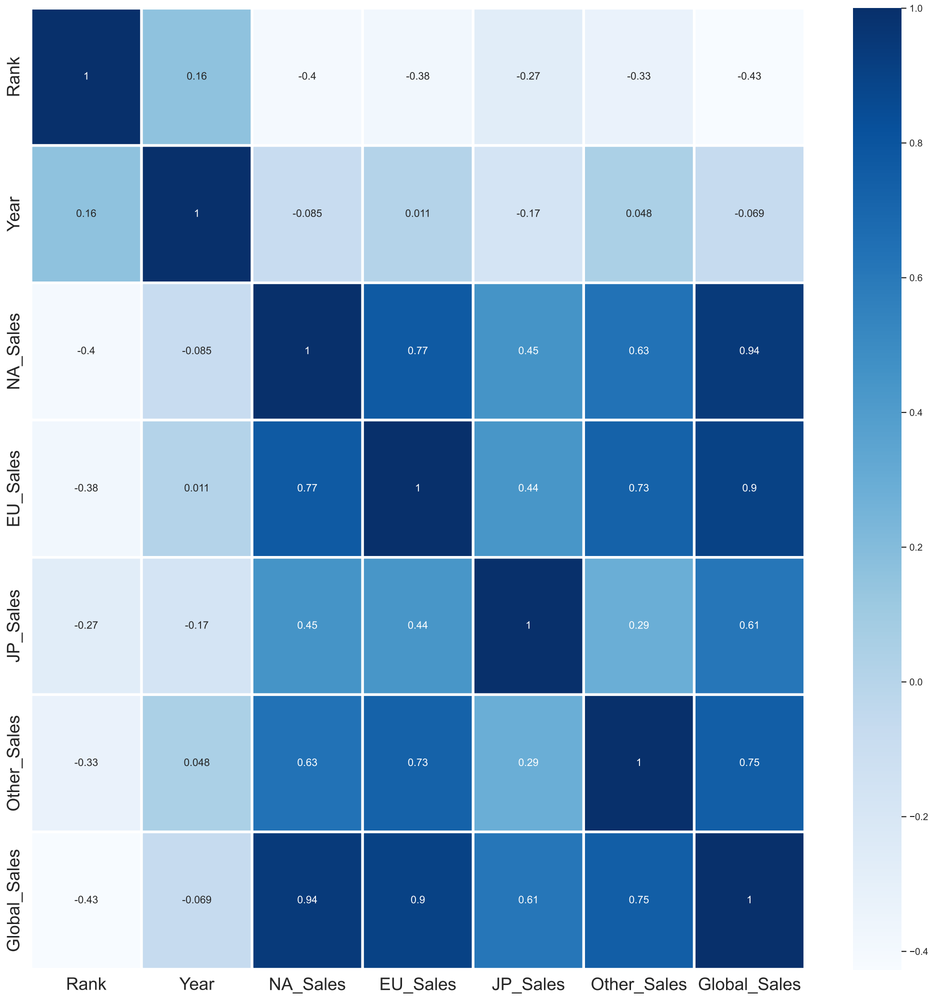


2.17 Relations of columns

```
plt.figure(figsize=(20,20))
sns.heatmap(data[['Rank','Year','NA_Sales',
       'EU_Sales', 'JP_Sales', 'Other_Sales',
       'Global_Sales']].corr(), cmap = "Blues", annot=True, linewidth=3)
plt.xticks(fontsize=20)
```

```
plt.yticks(fontsize=20)
```

```
plt.show()
```


Rank
Year
NA_Sales
EU_Sales
JP_Sales
Other_Sales
Global_Sales
1 0.16 -0.4 -0.38 -0.27 -0.33 -0.43
0.16 1 -0.085 0.011 -0.17 0.048 -0.069
-0.4 -0.085 1 0.77 0.45 0.63 0.94
-0.38 0.011 0.77 1 0.44 0.73 0.9
-0.27 -0.17 0.45 0.44 1 0.29 0.61
-0.33 0.048 0.63 0.73 0.29 1 0.75
-0.43 -0.069 0.94 0.9 0.61 0.75 1
Rank
Year
NA_Sales
EU_Sales
JP_Sales
Other_Sales
Global_Sales
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
−0.4

```
import sys
print(sys.version)
```

```
## 3.9.16 (main, Mar  8 2023, 10:39:24) [MSC v.1916 64 bit (AMD64)]
```

End