Final Project – Video Game Sales Analysis

IST687 – Thursdays – 9 EST

Rohini Shrivastava
Mike Coomber
Rodney Hur

Introduction and Data Acquisition

The first video game was thought to be created by a physicist, William Higinbotham, in 1958. A little over a decade later, video games became much more mainstream and available to individuals. Today, it is projected that over a billion people around the globe play videogames in one form or another.

For the purposes of our analysis, we acquired our dataset, *Video Game Sales*, by Gregory Smith on Kaggle[i]. We decided to investigate video game sales because we are all avid gamers and interested in how the patterns and trends have developed over time. By following the trends, we will be able to leverage the data to see what games, regions, genre, etc will have the greatest rate of return on investment.

The dataset has 16,599 rows and 11 columns for a total of 182,589 datapoints. We will be looking at all the data points, except for the id, as we are interested in looking at the trends of several different variables, like console type, region, etc., over time.

Few limitations of this dataset are as following: this data splits the global market into regions; therefore, we are unable to investigate sales by countries. Second limitation is that this dataset only collects data from 1980 to 2016, therefore, we are unable to investigate the trends and queries outside the time frame. The third limitation is that majority of the datasets are of big sellers and not on independent sub-genre of games.

Business Questions:

After a quick overview of the data, the team came up with several questions that we wanted to understand about the data. The following eight questions were developed to explore the data.

- What console released the most games overall?
- What year had the most games released?
- What genre of game was the highest in sales overall?
- Which region has the most sales?
- Which publisher published the most games?
- What is the trend of global video game sales over the years?
- What are the average sales per game released in a given year?
- Can we create a model that predicts global sales by utilizing only one of the three regions?
- Can we create a model that predicts if we will have high or low sales for a video game?

Data:

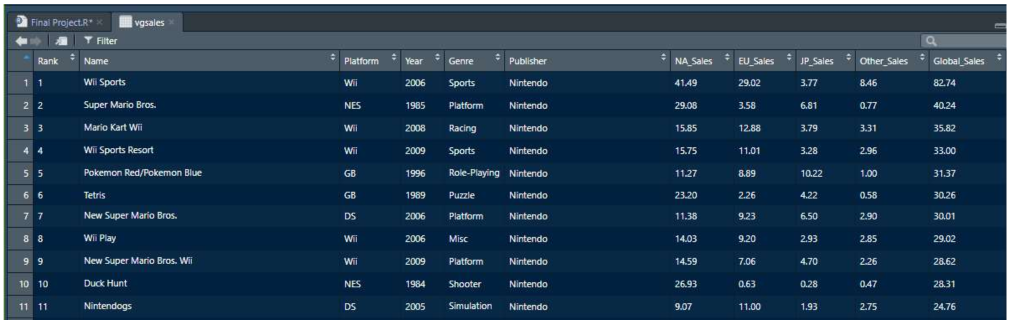
The *Video Game Sales* dataset by Gregory Smith, had several attributes that needed to be understood before we continued with the analysis. The following Data Dictionary table explains the data type and the definition of each of the attributes in the dataset.

Data dictionary

Attribute	Data Type	Definition
Name	Factor	Name of the game
Platform	Factor	Name of gaming console
Year	Factor	Year released
Genre	Factor	Genre of the game
Publishers	Factor	Name of game publisher
NA Sales	Num	North-American copies sold in millions
EU Sales	Num	European copies sold in millions
JP Sales	Num	Japanese copies sold in millions
Other Sales	Num	Other copies sold in millions
Global Sales	Num	Total copies sold in millions

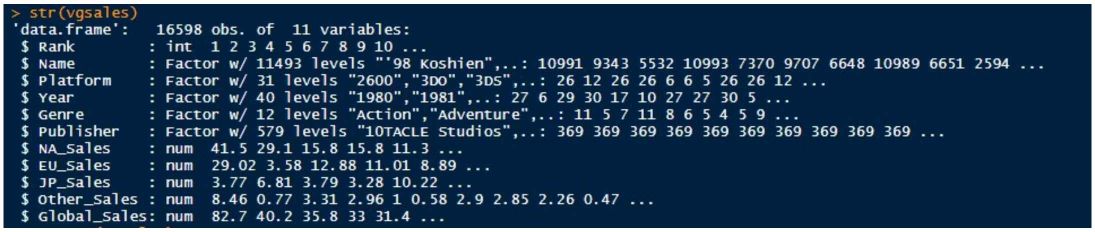
Before doing any modifications and analysis on the data, we needed to bring in the data from the csv file and have a R read and create a data frame from it. We started our code with a simple read.csv function. After import, we took some beginning looks at what the data was in its raw format. The following two images are samples of the data set after importing into R and the summary of the data frame and its attributes.

Sample of Uncleaned Data:

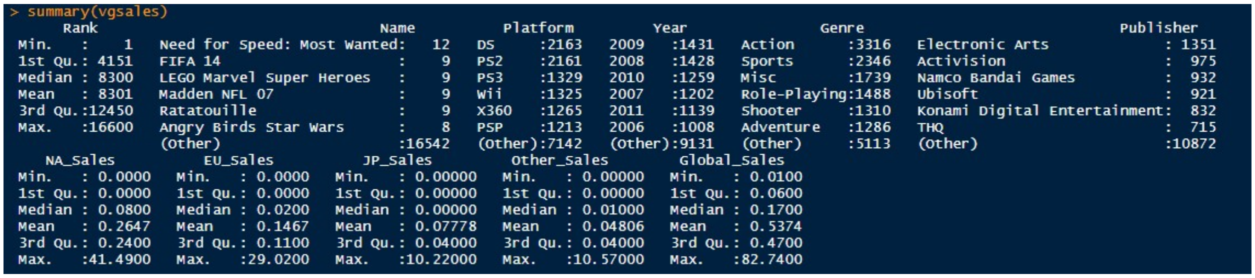


	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
1	1	Wii Sports	Wii	2006	Sports	Nintendo	41.49	29.02	3.77	8.46	82.74
2	2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08	3.58	6.81	0.77	40.24
3	3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85	12.88	3.79	3.31	35.82
4	4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75	11.01	3.28	2.96	33.00
5	5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27	8.89	10.22	1.00	31.37
6	6	Tetris	GB	1989	Puzzle	Nintendo	23.20	2.26	4.22	0.58	30.26
7	7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11.38	9.23	6.50	2.90	30.01
8	8	Wii Play	Wii	2006	Misc	Nintendo	14.03	9.20	2.93	2.85	29.02
9	9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14.59	7.06	4.70	2.26	28.62
10	10	Duck Hunt	NES	1984	Shooter	Nintendo	26.93	0.63	0.28	0.47	28.31
11	11	Nintendogs	DS	2005	Simulation	Nintendo	9.07	11.00	1.93	2.75	24.76

Summary of Uncleaned Data:



```
> str(vgsales)
'data.frame':   16598 obs. of  11 variables:
 $ Rank        : int  1 2 3 4 5 6 7 8 9 10 ...
 $ Name        : Factor w/ 11493 levels "'98 Koshien",..: 10991 9343 5532 10993 7370 9707 6648 10989 6651 2594 ...
 $ Platform    : Factor w/ 31 levels "2600","3DO","3DS",..: 26 12 26 26 6 6 5 26 26 12 ...
 $ Year        : Factor w/ 40 levels "1980","1981",..: 27 6 29 30 17 10 27 27 30 5 ...
 $ Genre       : Factor w/ 12 levels "Action","Adventure",..: 11 5 7 11 8 6 5 4 5 9 ...
 $ Publisher   : Factor w/ 579 levels "10TACLE Studios",..: 369 369 369 369 369 369 369 369 369 369 ...
 $ NA_Sales    : num  41.5 29.1 15.8 15.8 11.3 ...
 $ EU_Sales    : num  29.02 3.58 12.88 11.01 8.89 ...
 $ JP_Sales    : num  3.77 6.81 3.79 3.28 10.22 ...
 $ Other_Sales : num  8.46 0.77 3.31 2.96 1 0.58 2.9 2.85 2.26 0.47 ...
 $ Global_Sales: num  82.7 40.2 35.8 33 31.4 ...
```



```
> summary(vgsales)
      Rank                                 Name            Platform          Year              Genre                                    Publisher
 Min.   :    1   Need for Speed: Most Wanted:   12   DS      :2163   2009    :1431   Action      :3316   Electronic Arts             : 1351
 1st Qu.: 4151   FIFA 14                    :    9   PS2     :2161   2008    :1428   Sports      :2346   Activision                  :  975
 Median : 8300   LEGO Marvel Super Heroes   :    9   PS3     :1329   2010    :1259   Misc        :1739   Namco Bandai Games          :  932
 Mean   : 8301   Madden NFL 07              :    9   Wii     :1325   2007    :1202   Role-Playing:1488   Ubisoft                     :  921
 3rd Qu.:12450   Ratatouille                :    9   X360    :1265   2011    :1139   Shooter     :1310   Konami Digital Entertainment:  832
 Max.   :16600   Angry Birds Star Wars      :    8   PSP     :1213   2006    :1008   Adventure   :1286   THQ                         :  715
                 (Other)                    :16542   (Other):7142    (Other):9131    (Other)     :5113   (Other)                     :10872
    NA_Sales          EU_Sales           JP_Sales          Other_Sales         Global_Sales
 Min.   : 0.0000   Min.   : 0.0000   Min.   : 0.00000   Min.   : 0.00000   Min.   : 0.0100
 1st Qu.: 0.0000   1st Qu.: 0.0000   1st Qu.: 0.00000   1st Qu.: 0.00000   1st Qu.: 0.0600
 Median : 0.0800   Median : 0.0200   Median : 0.00000   Median : 0.01000   Median : 0.1700
 Mean   : 0.2647   Mean   : 0.1467   Mean   : 0.07778   Mean   : 0.04806   Mean   : 0.5374
 3rd Qu.: 0.2400   3rd Qu.: 0.1100   3rd Qu.: 0.04000   3rd Qu.: 0.04000   3rd Qu.: 0.4700
 Max.   :41.4900   Max.   :29.0200   Max.   :10.22000   Max.   :10.57000   Max.   :82.7400
```

Cleansing and Quality assessment:

As we reviewed the raw data after import we noticed that more recent data has higher frequency of N/As. It was decided that these could be removed easily via na.omit() once it has been converted into the correct format.

Additionally, found the attribute “Rank” redundant because t only ranks based on global sales, which is the last attribute in the data frame. We believed instead of having rank, we can just sort the global sales column for the same effect.

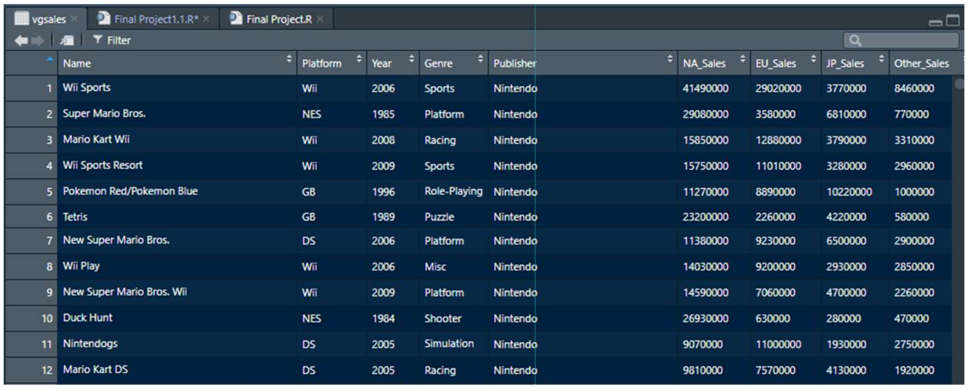
Further cleansing of the data was needed as we discovered that there are various platforms that are very specific but only a few game records. It was decided to lump them with other platforms of similar type. The following table illustrates the new platform name used to replace the one-off platforms.

<u>Platform Renaming</u>

Old Platform	New Platform
2600	Atari
GEN	SNES
SAT	Sega
SCD	Sega
NG	Misc
TG16	PC
3DO	Misc
PCFX	Misc
GG	Misc
WS	Misc

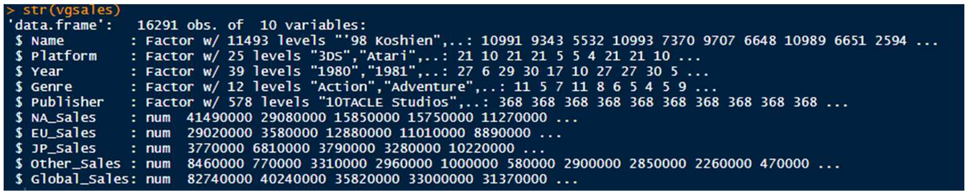
The following figures show the data after the cleansing steps we performed. Each of the cleansing steps were added to the function that imported the data. Additionally, in the same function, we performed a gsub to replace all “N/A” records with NA, so the na.omit() function worked correctly and recognized the value as a true NA.

Sample of cleansed Data:

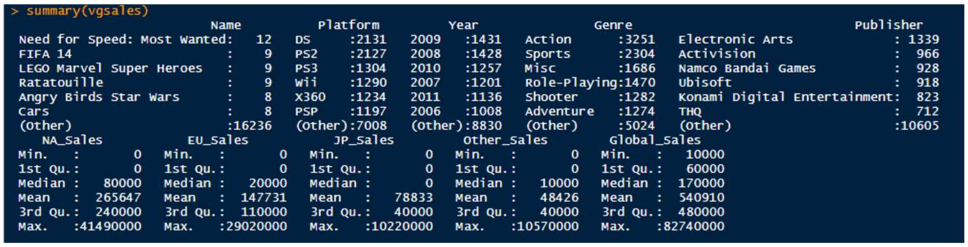


	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales
1	Wii Sports	Wii	2006	Sports	Nintendo	41490000	29020000	3770000	8460000
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29080000	3580000	6810000	770000
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15850000	12880000	3790000	3310000
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15750000	11010000	3280000	2960000
5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11270000	8890000	10220000	1000000
6	Tetris	GB	1989	Puzzle	Nintendo	23200000	2260000	4220000	580000
7	New Super Mario Bros.	DS	2006	Platform	Nintendo	11380000	9230000	6500000	2900000
8	Wii Play	Wii	2006	Misc	Nintendo	14030000	9200000	2930000	2850000
9	New Super Mario Bros. Wii	Wii	2009	Platform	Nintendo	14590000	7060000	4700000	2260000
10	Duck Hunt	NES	1984	Shooter	Nintendo	26930000	630000	280000	470000
11	Nintendogs	DS	2005	Simulation	Nintendo	9070000	11000000	1930000	2750000
12	Mario Kart DS	DS	2005	Racing	Nintendo	9810000	7570000	4130000	1920000

Summary of cleansed Data:



```
> str(vgsales)
'data.frame':	16291 obs. of  10 variables:
 $ Name        : Factor w/ 11493 levels "'98 Koshien",..: 10991 9343 5532 10993 7370 9707 6648 10989 6651 2594 ...
 $ Platform    : Factor w/ 25 levels "3DS","Atari",..: 21 10 21 21 5 5 4 21 21 10 ...
 $ Year        : Factor w/ 39 levels "1980","1981",..: 27 6 29 30 17 10 27 27 30 5 ...
 $ Genre       : Factor w/ 12 levels "Action","Adventure",..: 11 5 7 11 8 6 5 4 5 9 ...
 $ Publisher   : Factor w/ 578 levels "10TACLE Studios",..: 368 368 368 368 368 368 368 368 368 368 ...
 $ NA_Sales    : num  41490000 29080000 15850000 15750000 11270000 ...
 $ EU_Sales    : num  29020000 3580000 12880000 11010000 8890000 ...
 $ JP_Sales    : num  3770000 6810000 3790000 3280000 10220000 ...
 $ Other_Sales : num  8460000 770000 3310000 2960000 1000000 580000 2900000 2850000 2260000 470000 ...
 $ Global_Sales: num  82740000 40240000 35820000 33000000 31370000 ...
```

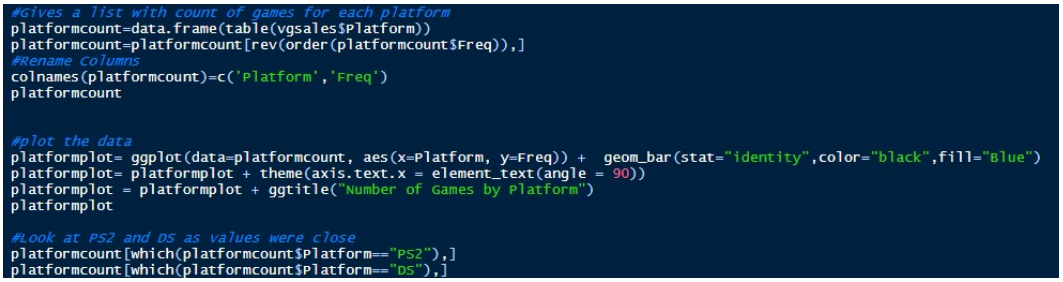


```
> summary(vgsales)
                          Name          Platform         Year              Genre                               Publisher
 Need for Speed: Most Wanted:   12   DS      :2131   2009   :1431   Action       :3251   Electronic Arts            : 1339
 FIFA 14                    :    9   PS2     :2127   2008   :1428   Sports       :2304   Activision                 :  966
 LEGO Marvel Super Heroes   :    9   PS3     :1304   2010   :1257   Misc         :1686   Namco Bandai Games         :  928
 Ratatouille                :    9   Wii     :1290   2007   :1201   Role-Playing:1470   Ubisoft                    :  918
 Angry Birds Star Wars      :    8   X360    :1234   2011   :1136   Shooter      :1282   Konami Digital Entertainment:  823
 Cars                       :    8   PSP     :1197   2006   :1008   Adventure    :1274   THQ                        :  712
 (Other)                    :16236   (Other):7008   (Other):8830   (Other)      :5024   (Other)                    :10605
    NA_Sales           EU_Sales           JP_Sales          Other_Sales        Global_Sales
 Min.   :       0   Min.   :       0   Min.   :       0   Min.   :       0   Min.   :   10000
 1st Qu.:       0   1st Qu.:       0   1st Qu.:       0   1st Qu.:       0   1st Qu.:   60000
 Median :   80000   Median :   20000   Median :       0   Median :   10000   Median :  170000
 Mean   :  265647   Mean   :  147731   Mean   :   78833   Mean   :   48426   Mean   :  540910
 3rd Qu.:  240000   3rd Qu.:  110000   3rd Qu.:   40000   3rd Qu.:   40000   3rd Qu.:  480000
 Max.   :41490000   Max.   :29020000   Max.   :10220000   Max.   :10570000   Max.   :82740000
```

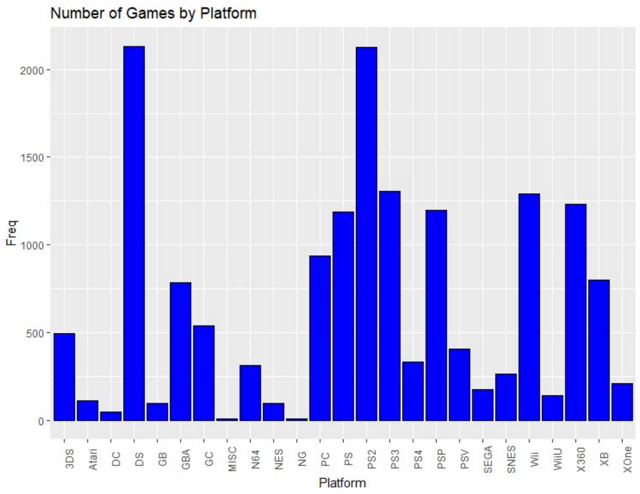
Results:

<u>Question 1: What console released the most games overall?</u>

When plotting the Number of games by Platform, the bar graph shows a tie between Nintendo DS and Playstation PS2. The number of different games sold between both Nintendo DS and Playstation were indeed very close when looking at the specific records in the data frame. Nintendo DS sold 2131 different games where PS2 sold 2127.

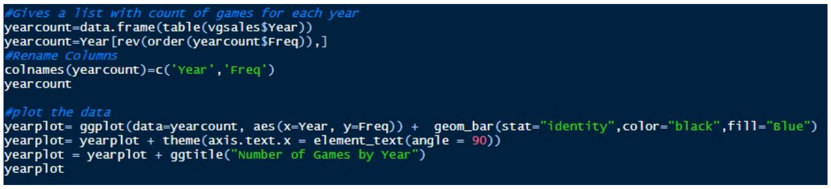


```
#Gives a list with count of games for each platform
platformcount=data.frame(table(vgsales$Platform))
platformcount=platformcount[rev(order(platformcount$Freq)),]
#Rename Columns
colnames(platformcount)=c('Platform','Freq')
platformcount


#plot the data
platformplot= ggplot(data=platformcount, aes(x=Platform, y=Freq)) +  geom_bar(stat="identity",color="black",fill="Blue")
platformplot= platformplot + theme(axis.text.x = element_text(angle = 90))
platformplot = platformplot + ggtitle("Number of Games by Platform")
platformplot

#Look at PS2 and DS as values were close
platformcount[which(platformcount$Platform=="PS2"),]
platformcount[which(platformcount$Platform=="DS"),]
```

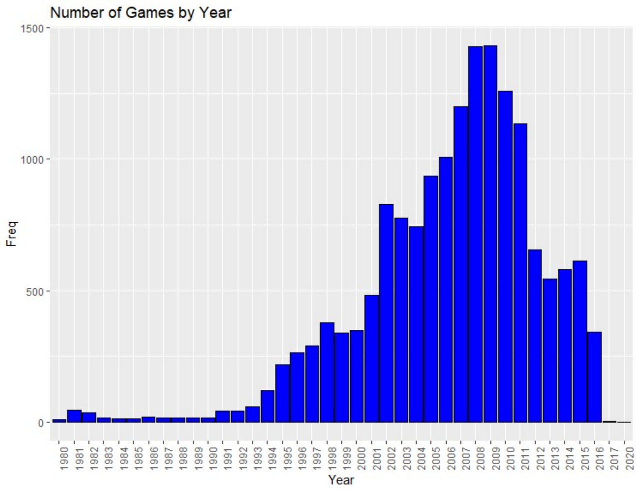


Question 2: What year had the most games released?

Similarly to the games by platform bar chart, the games released by year were put into a frequency data frame, ordered by year and plotted. From the review of the bar chart output, 2009 had the most game releases which 2008 coming in as a close second.

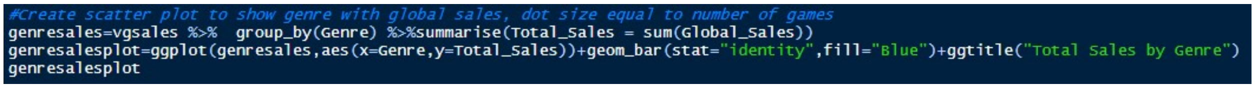


```
#Gives a list with count of games for each year
yearcount=data.frame(table(vgsales$Year))
yearcount=Year[rev(order(yearcount$Freq)),]
#Rename Columns
colnames(yearcount)=c('Year','Freq')
yearcount

#plot the data
yearplot= ggplot(data=yearcount, aes(x=Year, y=Freq)) +  geom_bar(stat="identity",color="black",fill="Blue")
yearplot= yearplot + theme(axis.text.x = element_text(angle = 90))
yearplot = yearplot + ggtitle("Number of Games by Year")
yearplot
```

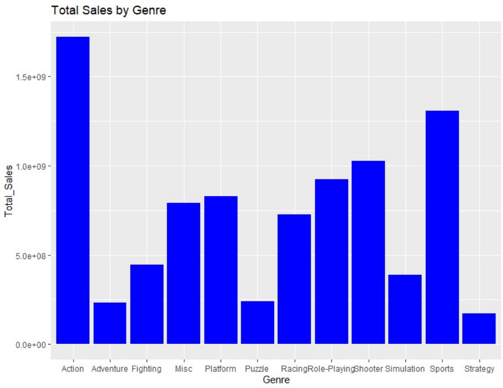


Question 3: What genre of game was the highest in sales overall?

This question looks at total global sales by genre. To accomplish this, it was necessary to create a data frame grouped by genre. In addition to the grouping, the total global sales from the original vgsales data frame needed to be summarized into this new data frame. The graph clearly shows that the genre "Action" is by far the leader in global sales vs all the other genres.

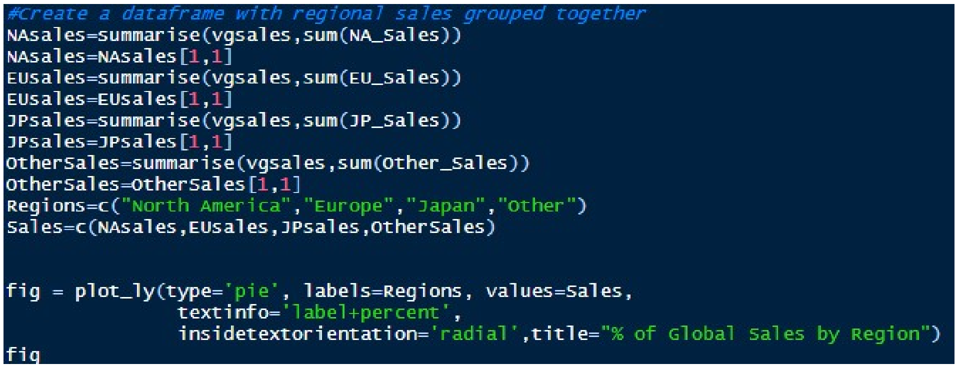


```
#Create scatter plot to show genre with global sales, dot size equal to number of games
genresales=vgsales %>%  group_by(Genre) %>%summarise(Total_Sales = sum(Global_Sales))
genresalesplot=ggplot(genresales,aes(x=Genre,y=Total_Sales))+geom_bar(stat="identity",fill="Blue")+ggtitle("Total Sales by Genre")
genresalesplot
```

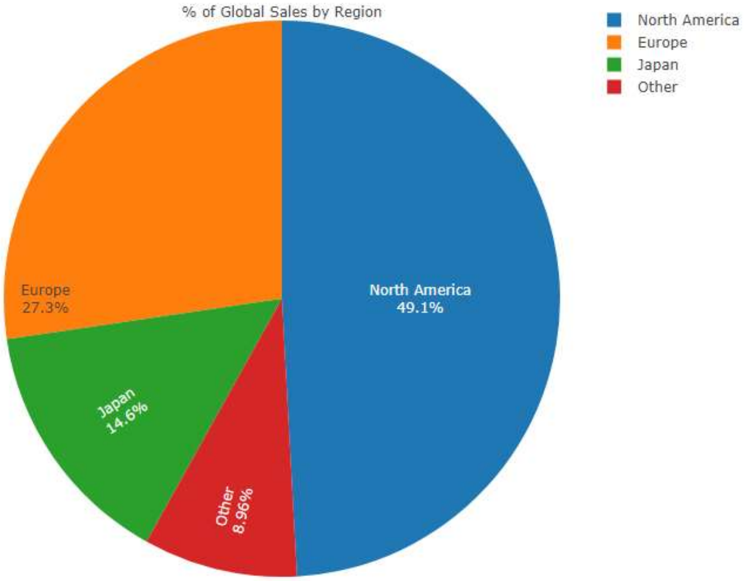


Question 4: Which region has the most sales?

For this question, we took a different approach at visualizing the data. Since there were only four regions listed, it was easier to understand this data in a percentage of global sales and view them in a pie chart. From the pie chart, North America has nearly 50% of all video game sales in the world.

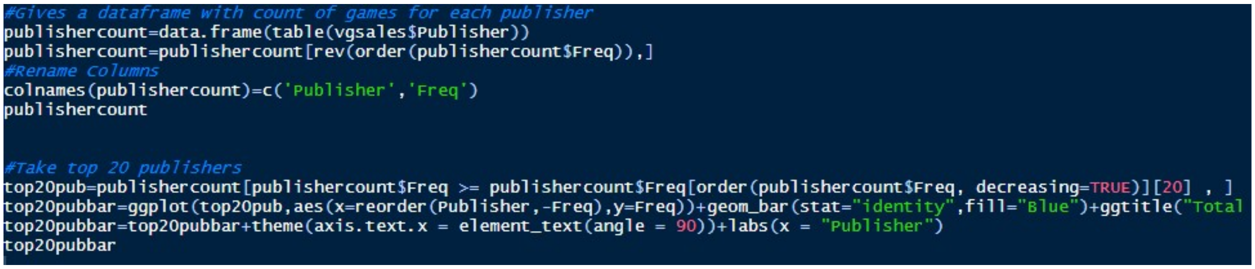


```
#Create a dataframe with regional sales grouped together
NAsales=summarise(vgsales,sum(NA_Sales))
NAsales=NAsales[1,1]
EUsales=summarise(vgsales,sum(EU_Sales))
EUsales=EUsales[1,1]
JPsales=summarise(vgsales,sum(JP_Sales))
JPsales=JPsales[1,1]
OtherSales=summarise(vgsales,sum(Other_Sales))
OtherSales=OtherSales[1,1]
Regions=c("North America","Europe","Japan","Other")
Sales=c(NAsales,EUsales,JPsales,OtherSales)


fig = plot_ly(type='pie', labels=Regions, values=Sales,
               textinfo='label+percent',
               insidetextorientation='radial',title="% of Global Sales by Region")
fig
```

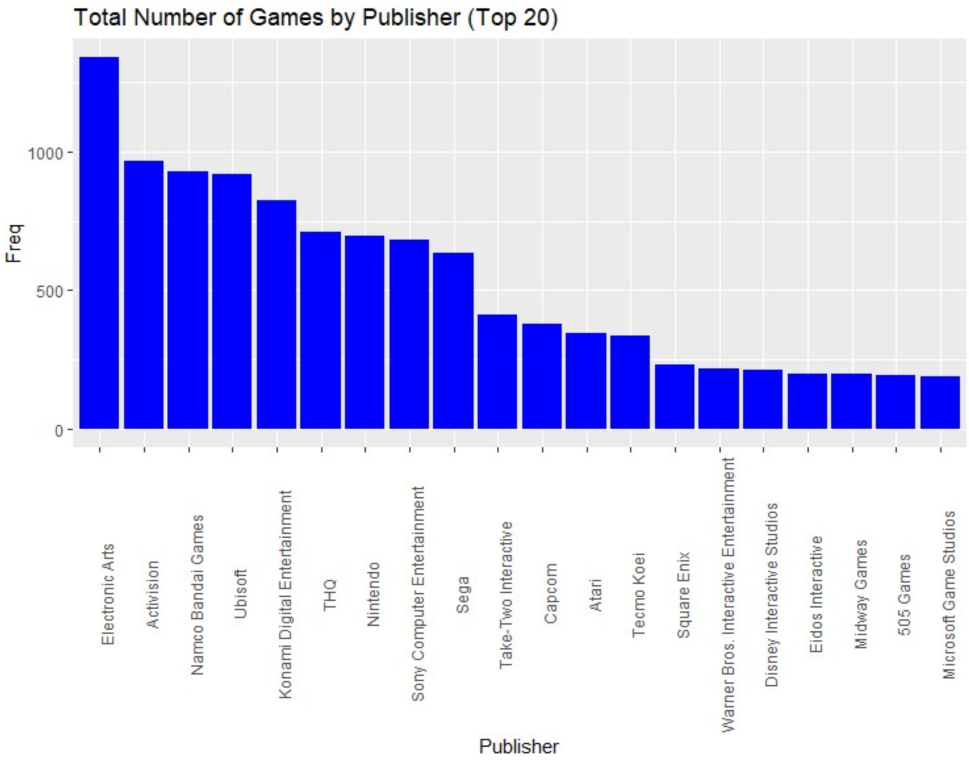


Question 5: Which publisher published the most games?

In review of this question, it was found that there were over 500 different publishers that have contributed to producing the video games in the data set. In the effort for visualization and to ensure an eye-chart wasn't created, the top 20 publishers were used for the analysis.

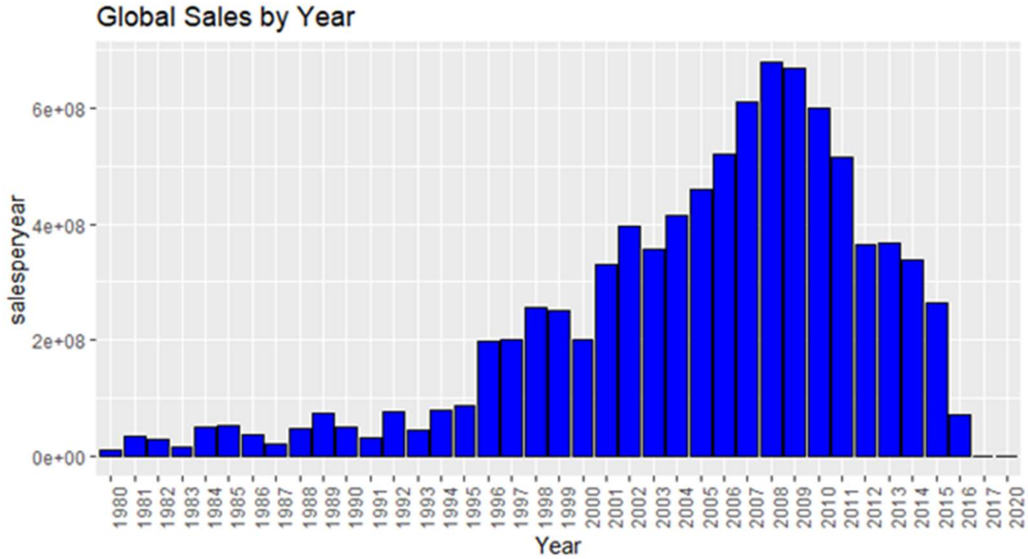


```r
#Gives a dataframe with count of games for each publisher
publishercount=data.frame(table(vgsales$Publisher))
publishercount=publishercount[rev(order(publishercount$Freq)),]
#Rename Columns
colnames(publishercount)=c('Publisher','Freq')
publishercount


#Take top 20 publishers
top20pub=publishercount[publishercount$Freq >= publishercount$Freq[order(publishercount$Freq, decreasing=TRUE)][20] , ]
top20pubbar=ggplot(top20pub,aes(x=reorder(Publisher,-Freq),y=Freq))+geom_bar(stat="identity",fill="Blue")+ggtitle("Total
top20pubbar=top20pubbar+theme(axis.text.x = element_text(angle = 90))+labs(x = "Publisher")
top20pubbar
```



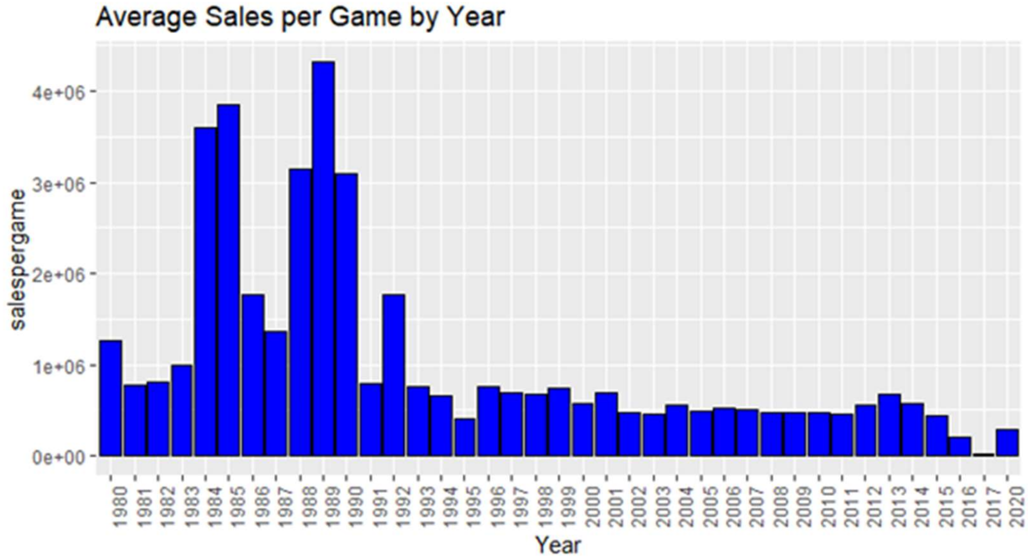
Question 6: What is the trend of global video games sales by year?

We can see a trend of the chart that matches what we have previously seen with the chart for "Number of Games by Year". We saw the most sales in 2008 with 2009 being in a close second.



Question 7: What are the average sales per game released on a given year?

We calculated average sales per year by taking the number of global sales divided by the number of games released for the year. The highest average was seen in 1988. After the mid-90s the averages were very similar.



Question 8: Which one of the major regions' sales is the best predictor of global sales?

Linear Modeling

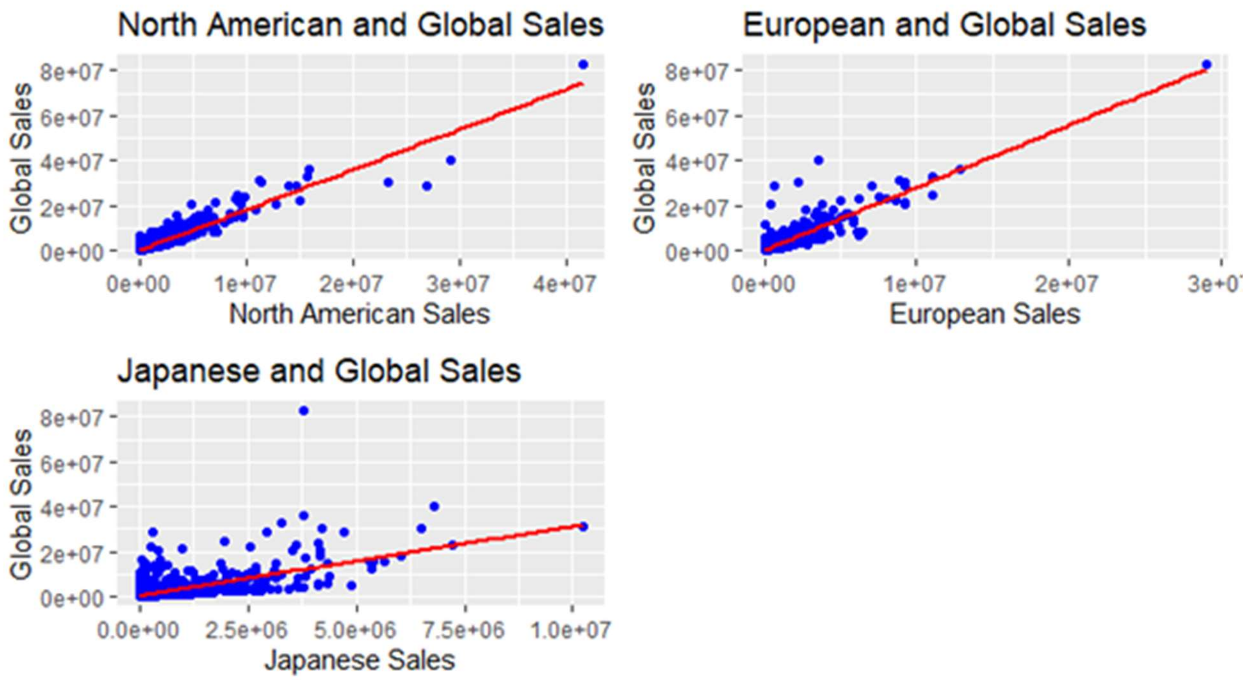
Videogame publishers often incorporate primary research as the main driver of their future sales projections of a new release. As videogames are widely adopted around the world, it is important for publishers to be able to forecast near-accurate global demand; in a perfect world, publishers would be able to conduct research around the world to do so, but this is often not feasible due to the constraints posed by money, time, and resources.

To solve this problem, we ask the question: **can we create a model that can predict global sales by utilizing only one of the three major regions?**

First, we created three separate linear models and plots with the predictor variables being sales in the three major regions, North America, Europe, and Japan, and the outcome variable being global sales. Then we assessed the results of each model and plots to assess the fit of the line and see if there are any unusual variability. For ease of view, we have arranged our plots into a single visual.

Our results are as follows:

Region (significance)	Coefficient	R-Squared
North America (***)	1.794	0.8860
Europe (***)	2.780	0.8159
Japan (***)	3.079	0.3775



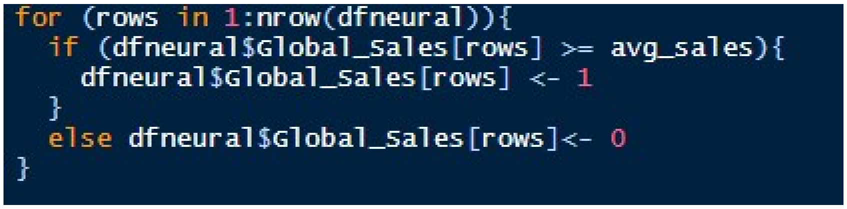
In conclusion, the best predictor of global sales is North American sales because it had a high degree of significance paired with the highest R squared value of 0.8860; this means that roughly 88.60% of variability in global sales can be explained by North American sales.

Question 9: Can we create a model that predicts if we will have high or low sales for a video game?

SVM

We used SVM to see if we can predict if there will be high sales or low for a video game based off Publisher, Genre, Year, and Platform.

First, we converted the categorical variables into numbers for our model. Global Sales was updated to be a binary where 0 is for low sales and 1 are for high sales. This was calculated by seeing whether Global Sales were higher or lower than the average.



```
for (rows in 1:nrow(dfneural)){
  if (dfneural$Global_Sales[rows] >= avg_sales){
    dfneural$Global_Sales[rows] <- 1
  }
  else dfneural$Global_Sales[rows]<- 0
}
```

We created two datasets, training and testing. The SVM Model was run using our training dataset. We wanted to see Global Sales based off Genre, Publisher, Platform, and Year. This model was then used to predict high or low sales in our testing dataset.

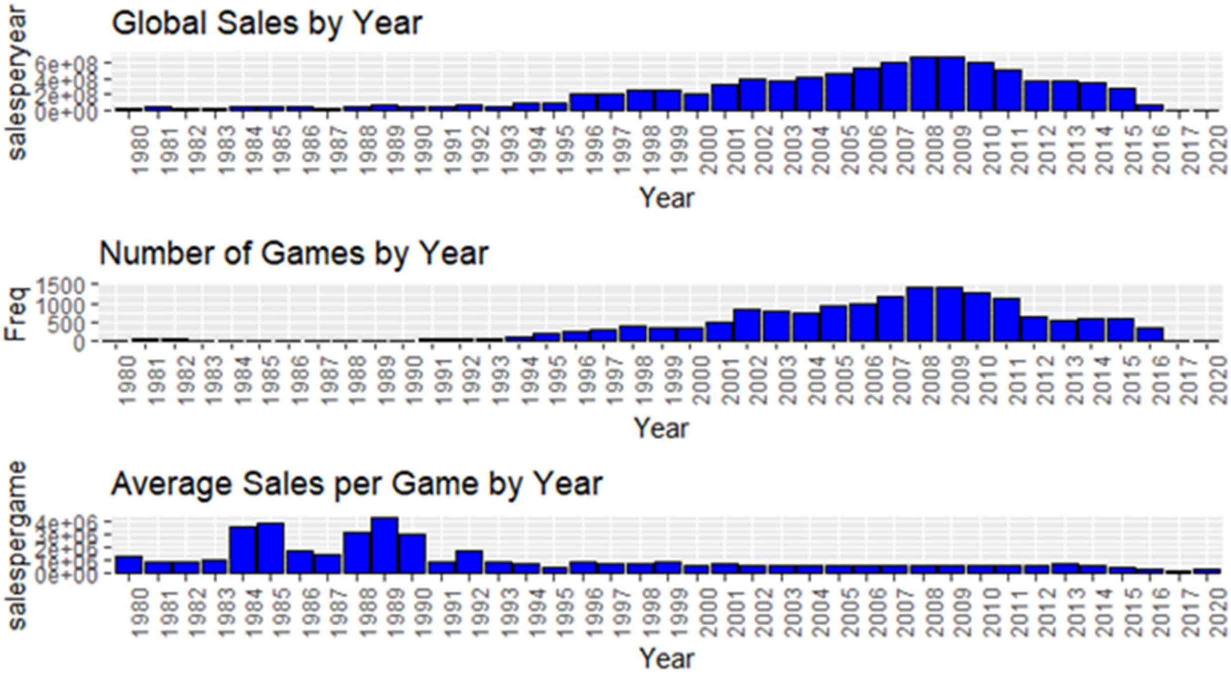
We took the difference between our predicted output and the actual output to see whether the model was accurate. We gave 0.05 as a margin of error in our calculations. We found our model had a **75.47%** chance of being accurate in determining amount of sales.

Interesting Findings

Game releases and sales VS. Sales per game released

If we were to focus our attention to only the frequencies of new releases and global sales over the years, it is fair to conclude that 2008 and 2009 were the golden years for publishers to produce new games as the number of new releases and global sales have peaked; but that would be a mistake.

Since 1990, the average sales per release in dollar amount has steadily declined. This means that individual games are not attracting as many buyers. This may be due to the sheer variety presented by the increasing number of games released per year among other factors.



Wii Sports, not as popular in Japan?

The video game that set the world record for sales is the Nintendo Wii Sports with $82.7 million. In both North America and Europe, Wii sports is king, and it outsold their respective second place games by $25.6 million and $16.1 million. However, in Japan, Wii sports' sales is placed as the 26th most sold game. This is interesting because Wii Sports was the most sold video game in all the regions except for Japan, their country of production.

Conclusions:

Average video game sales have gone down since the late 80s/early 90s, however we have seen an increase in overall video game sales and amount of games released. This may be showing that the market has become oversaturated. Video game sales peaked in 2008- which follows trend of the number of games sold by year as well.

According to our SVM model- 75% of the time we can accurately predict if a video game will be a high seller by knowing the Genre, Publisher, Year, and Platform of the game. With our Linear Model we can determine what the overall sales of the games are based off the region sales.

Through all the analyses performed, we can assume that the data taken from Gregory Smith did not accurately reflect the correct numbers of video games and sales in the later years. Starting 2016 there was a decline in the data collected- which may be causing our findings to be slightly skewed.

Appendix:

```
###########
# Course: IST 687
# Assignment: Final Project
# Name: Mike Coomber, Rohini Shrivastava, Rodney Hur
# Date: 8/26/20
# Notes: Rev 1.1


#########
#Libraries
EnsurePackage=function(x)
{
  x=as.character(x)
  if (!require(x,character.only=TRUE))
  {
    install.packages(pkgs=x,repos="http://cran.r-project.org")
    require(x,character.only = TRUE)
  }
}

EnsurePackage("ggplot2")
EnsurePackage("dplyr")
EnsurePackage("plotly")
EnsurePackage("kernlab")
EnsurePackage("Metrics")
EnsurePackage("e1071")
EnsurePackage("gridExtra")


#Create Numberize function
Numberize = function(inputVector)
{
  #Remove Commas
  inputVector = gsub(',','',inputVector)
  #Remove spaces
  inputVector = gsub(' ','',inputVector)
  return(as.numeric(inputVector))
}



inputfile="C:/Users/frogg/OneDrive/SU Files/Intro to Data Science/Final
Project/Final Project/vgsales.csv"
#read in the data set and
readdata = function(fileloc)
{
  tempfile=read.csv(fileloc)
  #convert all N/As to NA for removal
  tempfile$Name=as.factor(gsub("N/A",NA,tempfile$Name))
  tempfile$Platform=as.factor(gsub("N/A",NA,tempfile$Platform))
  tempfile$Year=as.factor(gsub("N/A",NA,tempfile$Year))
  tempfile$Genre=as.factor(gsub("N/A",NA,tempfile$Genre))
  tempfile$Publisher=as.factor(gsub("N/A",NA,tempfile$Publisher))
  tempfile$NA_Sales=gsub("N/A",NA,tempfile$NA_Sales)
  tempfile$EU_Sales=gsub("N/A",NA,tempfile$EU_Sales)
  tempfile$JP_Sales=gsub("N/A",NA,tempfile$JP_Sales)
  tempfile$Other_Sales=gsub("N/A",NA,tempfile$Other_Sales)
  tempfile$Global_Sales=gsub("N/A",NA,tempfile$Global_Sales)

  #set sales to numeric using numberize and make value in millions

  tempfile$NA_Sales=Numberize(tempfile$NA_Sales)*1000000
  tempfile$EU_Sales=Numberize(tempfile$EU_Sales)*1000000
  tempfile$JP_Sales=Numberize(tempfile$JP_Sales)*1000000
  tempfile$Other_Sales=Numberize(tempfile$Other_Sales)*1000000
  tempfile$Global_Sales=Numberize(tempfile$Global_Sales)*1000000

  #update platform names to match


  tempfile$Platform=as.factor(gsub("2600","Atari",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("GEN","SNES",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("SAT","SEGA",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("SCD","SEGA",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("3DO","MISC",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("PCFX","MISC",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("GG","MISC",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("WS","MISC",tempfile$Platform))
  tempfile$Platform=as.factor(gsub("TG16","PC",tempfile$Platform))


  #Omit NA rows
  tempfile=na.omit(tempfile)
  #Remove Rank column
  tempfile=tempfile[,2:11]
  return(tempfile)
}

#store output into dataframe
vgsales = readdata(inputfile)

#dataframe
str(vgsales)
summary(vgsales)

#Gives a dataframe with count of games for each platform
platformcount=data.frame(table(vgsales$Platform))
platformcount=platformcount[rev(order(platformcount$Freq)),]
#Rename Columns
colnames(platformcount)=c('Platform','Freq')
platformcount


#plot the data
platformplot= ggplot(data=platformcount, aes(x=Platform, y=Freq)) +
geom_bar(stat="identity",color="black",fill="Blue")
platformplot= platformplot + theme(axis.text.x = element_text(angle = 90))
platformplot = platformplot + ggtitle("Number of Games by Platform")
platformplot

#Look at PS2 and DS as values were close
platformcount[which(platformcount$Platform=="PS2"),]
```

```
platformcount[which(platformcount$Platform=="DS"),]



#Gives a dataframe with count of games for each genre
genrecount=data.frame(table(vgsales$Genre))
genrecount=genrecount[rev(order(genrecount$Freq)),]
#Rename Columns
colnames(genrecount)=c('Genre','Freq')
genrecount

#plot the data
genrerplot= ggplot(data=genrecount, aes(x=Genre, y=Freq)) +
geom_bar(stat="identity",color="black",fill="Blue")
genrerplot= genrerplot + theme(axis.text.x = element_text(angle = 90))
genrerplot = genrerplot + ggtitle("Number of Games by Genre")
genrerplot

#Gives a dataframe with count of games for each year
yearcount=data.frame(table(vgsales$Year))
yearcount=Year[rev(order(yearcount$Freq)),]
#Rename Columns
colnames(yearcount)=c('Year','Freq')
yearcount

#plot the data
yearplot= ggplot(data=yearcount, aes(x=Year, y=Freq)) +
geom_bar(stat="identity",color="black",fill="Blue")
yearplot= yearplot + theme(axis.text.x = element_text(angle = 90))
yearplot = yearplot + ggtitle("Number of Games by Year")
yearplot

#Create scatter plot to show genre with global sales, dot size equal to
number of games
genresales=vgsales %>%  group_by(Genre) %>%summarise(Total_Sales =
sum(Global_Sales))
genresalesplot=ggplot(genresales,aes(x=Genre,y=Total_Sales))+geom_bar(stat="i
dentity",fill="Blue")+ggtitle("Total Sales by Genre")
genresalesplot

#Create a dataframe with regional sales grouped together
NAsales=summarise(vgsales,sum(NA_Sales))
NAsales=NAsales[1,1]
EUsales=summarise(vgsales,sum(EU_Sales))
EUsales=EUsales[1,1]
JPsales=summarise(vgsales,sum(JP_Sales))
JPsales=JPsales[1,1]
OtherSales=summarise(vgsales,sum(Other_Sales))
OtherSales=OtherSales[1,1]
Regions=c("North America","Europe","Japan","Other")
Sales=c(NAsales,EUsales,JPsales,OtherSales)


fig = plot_ly(type='pie', labels=Regions, values=Sales,
              textinfo='label+percent',
              insidetextorientation='radial',title="% of Global Sales by
Region")
```

```
fig


#Gives a dataframe with count of games for each publisher
publishercount=data.frame(table(vgsales$Publisher))
publishercount=publishercount[rev(order(publishercount$Freq)),]
#Rename Columns
colnames(publishercount)=c('Publisher','Freq')
publishercount


#Take top 20 publishers
top20pub=publishercount[publishercount$Freq >=
publishercount$Freq[order(publishercount$Freq, decreasing=TRUE)][20] , ]
top20pubbar=ggplot(top20pub,aes(x=reorder(Publisher,-
Freq),y=Freq))+geom_bar(stat="identity",fill="Blue")+ggtitle("Total Number of
Games by Publisher (Top 20)")
top20pubbar=top20pubbar+theme(axis.text.x = element_text(angle = 90))+labs(x
= "Publisher")
top20pubbar

#Q6: What is the trend of global video game sale by year?
#Create a vector
salesperyear<-tapply(vgsales$Global_Sales ,as.factor(vgsales$Year), sum)

#Create new df
salesperyeardf<-data.frame(yearcount,salesperyear)

#Plot the data
salesperyearplot=ggplot(data=salesperyeardf, aes(x=Year, y=salesperyear)) +
geom_bar(stat="identity",color="black",fill="Blue")
salesperyearplot=salesperyearplot + theme(axis.text.x = element_text(angle =
90))
salesperyearplot=salesperyearplot + ggtitle("Global Sales by Year")
salesperyearplot


#Q6: What is the average sales per game released in a given year?
#Create a new variable
salesperyeardf$salespergame=(salesperyeardf$salesperyear/salesperyeardf$Freq)

#Plot the data
salespergameplot=ggplot(data=salesperyeardf, aes(x=Year, y=salespergame)) +
geom_bar(stat="identity",color="black",fill="Blue")
salespergameplot=salespergameplot + theme(axis.text.x = element_text(angle =
90))
salespergameplot=salespergameplot + ggtitle("Average Sales per Game by Year")
salespergameplot


#Linear Modeling
#Which major regional sales is the best predictor of global sales?
#Create separate df for lm
lmdf<-vgsales

#What is the relationship between North American sales and Global Sales?
naglobal<-lm(Global_Sales ~ NA_Sales, data=lmdf)
```

```
summary(naglobal)
##Formula: Global_Sales = 1.794*NA_Sales + 6.439e+04
##Multiple R-Squared: 0.8860

#Plot and add line to naglobal
naglobalplot<-ggplot(lmdf,aes(NA_Sales,
Global_Sales))+geom_point(color="blue")
naglobalplot<-naglobalplot + geom_smooth(method="lm",col="red")
naglobalplot<-naglobalplot + ggtitle("North American and Global Sales")
naglobalplot<-naglobalplot + xlab("North American Sales") + ylab("Global
Sales")
naglobalplot

#What is the relationship between European sales and Global Sales?
euglobal<-lm(Global_Sales ~ EU_Sales, data=lmdf)
summary(euglobal)
##Formula: Global_Sales = 2.780*EU_Sales + 6.439e+04
##Multiple R-Squared: 0.8159

#Plot and add line to euglobal
euglobalplot<-ggplot(lmdf,aes(EU_Sales,
Global_Sales))+geom_point(color="blue")
euglobalplot<-euglobalplot + geom_smooth(method="lm",col="red")
euglobalplot<-euglobalplot + ggtitle("European and Global Sales")
euglobalplot<-euglobalplot + xlab("European Sales") + ylab("Global Sales")
euglobalplot


#What is the relationship between Japanese sales and Global Sales?
jpglobal<-lm(Global_Sales ~ JP_Sales, data=lmdf)
summary(jpglobal)
##Formula: Global_Sales = 3.0790*JP_Sales + 6.439e+04
##Multiple R-Squared: 0.3755

#Plot and add line to jpglobal
jpglobalplot<-ggplot(lmdf,aes(JP_Sales,
Global_Sales))+geom_point(color="blue")
jpglobalplot<-jpglobalplot + geom_smooth(method="lm",col="red")
jpglobalplot<-jpglobalplot + ggtitle("Japanese and Global Sales")
jpglobalplot<-jpglobalplot + xlab("Japanese Sales") + ylab("Global Sales")
jpglobalplot

#Put all in one frame

grid.arrange(naglobalplot,euglobalplot,jpglobalplot, ncol=2)


#Interesting Findings
#Game releases, sales vs. sales per game
#Interesting Fivndings
grid.arrange(salesperyearplot,yearplot,salespergameplot)

#create neural network



#split data into train and test sets
```

```
neural <- vgsales
avg_sales <- mean(neural$Global_Sales)

#Convert all categorical variables into numeric values for both test and
train

dm_neural<- data.matrix(neural)
dm_neural <- subset(dm_neural, select = -c(Name, NA_Sales, EU_Sales,
JP_Sales, Other_Sales, Global_Sales))
dfneural<- as.data.frame(dm_neural)
dfneural$Global_Sales <- neural$Global

#we want to see if the data will have high or low sales
#if global sales is less than the average it will have low sales
#if it's higher, it will have high sales
for (rows in 1:nrow(dfneural)){
  if (dfneural$Global_Sales[rows] >= avg_sales){
    dfneural$Global_Sales[rows] <- 1
  }
  else dfneural$Global_Sales[rows]<- 0
}

#create our training and test samples
randIn <- sample(1:dim(dfneural)[1])

cut1_3 <- floor(2*dim(dfneural)[1]/3)

trainData<- dfneural[randIn[1:cut1_3],]

testData <- dfneural[randIn[(cut1_3+1):dim(dfneural)[1]],]

#run the SVM Model and predict the values
MultLin <- svm(Global_Sales~Genre+Platform+Year+Publisher, data=trainData)
predict_test <- predict(MultLin, testData)

testData$Accuracy <- predict_test

#update our testData set with the predictions to see how far off we are
for (rows in 1:nrow(testData)){
  if (abs(as.double(testData$Accuracy[rows])-
as.double(testData$Global_Sales[rows])) <= 0.05){
    testData$Predicted[rows] <- 1
  }
  else testData$Predicted[rows]<- 0
}

#Calculate how often we are correct with this model
avg_pred = mean(testData$Predicted)
avg_pred
```

[i] Data acquired from Kaggle - https://www.kaggle.com/gregorut/videogamesales