videogamesales

Kabir Kathuria

1/15/2022

Introduction/Overview

The data set used in this project is called "Video Game Sales with Ratings," and it was created by Rush Kirubi. The video game sale data originates from Vgchartz, and the corresponding ratings come from Metacritic. The raw version of the data set contains data on a total of 16,719 video games. The variables listed below are provided for each game in the set:

Name: Name of the video game Platform: Video game console that released the video game Year_of_Release: Year that the video game was released Genre: Category of video game Publisher: Publisher of video game Developer: Developer of video game NA_Sales: Video game sales in North America EU_Sales: Video game sales in Europe JP_Sales: Video game sales in Japan Other_Sales: Video game sales in other countries Global_Sales: Total worldwide video game sales Critic_Score: Aggregate score of video game compiled by Metacritic staff Critic_Count: Number of critics used in formulating critic score User_Score: Video game score given by Metacritic's subscribers User_Count: Number of users used in formulating user score Rating: ESRB rating of video game

My aim for this project was to analyze trends in the video games for each variable. Although I explored and analyzed every variable listed, I maintained a focus on the critic score and user score in order to eventually determine a correlation between the two. It is important to mention that I eliminated games that had incomplete information from the data set in order to simplify the numerous analyses.

Link to the data set: https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings

Installing/Loading Necessary Packages

The packages needed for this analysis must be installed first and loaded.

```r
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tidyverse
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: caret

## Warning: package ’caret’ was built under R version 4.1.1

## Loading required package: lattice

##
## Attaching package: ’caret’

## The following object is masked from ’package:purrr’:
##
##     lift
```

```
if(!require(data.table)) install.packages("data.table", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: data.table

##
## Attaching package: ’data.table’

## The following objects are masked from ’package:dplyr’:
##
##     between, first, last

## The following object is masked from ’package:purrr’:
##
##     transpose
```

```
if(!require(ggplot2)) install.packages("ggplot2", repos = "http://cran.us.r-project.org")
if(!require(readr)) install.packages("readr", repos = "http://cran.us.r-project.org")
if(!require(tidyr)) install.packages("tidyr", repos = "http://cran.us.r-project.org")
if(!require(dplyr)) install.packages("dplyr", repos = "http://cran.us.r-project.org")
if(!require(knitr)) install.packages("knitr", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: knitr
```

```
library(tidyverse)
library(caret)
library(data.table)
library(ggplot2)
library(readr)
library(tidyr)
library(dplyr)
library(knitr)
```

Data Exploration/Cleaning

The data is accessible through a CSV file, so it must be read into a variable which we will call vgSales.

```
vgSales <- read_csv("Video_Games_Sales_as_at_22_Dec_2016.csv")
```

```
##
## -- Column specification --------------------------------------------------------
## cols(
##   Name = col_character(),
##   Platform = col_character(),
##   Year_of_Release = col_character(),
##   Genre = col_character(),
##   Publisher = col_character(),
##   NA_Sales = col_double(),
##   EU_Sales = col_double(),
##   JP_Sales = col_double(),
##   Other_Sales = col_double(),
##   Global_Sales = col_double(),
##   Critic_Score = col_double(),
##   Critic_Count = col_double(),
##   User_Score = col_character(),
##   User_Count = col_double(),
##   Developer = col_character(),
##   Rating = col_character()
## )
```

```
summary(vgSales)
```

```
##      Name             Platform         Year_of_Release       Genre
##  Length:16719      Length:16719      Length:16719      Length:16719
##  Class :character  Class :character  Class :character  Class :character
##  Mode  :character  Mode  :character  Mode  :character  Mode  :character
##
##
##
##
##   Publisher            NA_Sales          EU_Sales         JP_Sales
##  Length:16719      Min.   : 0.0000   Min.   : 0.000   Min.   : 0.0000
##  Class :character  1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.: 0.0000
##  Mode  :character  Median : 0.0800   Median : 0.020   Median : 0.0000
##                    Mean   : 0.2633   Mean   : 0.145   Mean   : 0.0776
##                    3rd Qu.: 0.2400   3rd Qu.: 0.110   3rd Qu.: 0.0400
##                    Max.   :41.3600   Max.   :28.960   Max.   :10.2200
##
##   Other_Sales        Global_Sales      Critic_Score    Critic_Count
##  Min.   : 0.00000   Min.   : 0.0100   Min.   :13.00   Min.   :  3.00
##  1st Qu.: 0.00000   1st Qu.: 0.0600   1st Qu.:60.00   1st Qu.: 12.00
##  Median : 0.01000   Median : 0.1700   Median :71.00   Median : 21.00
##  Mean   : 0.04733   Mean   : 0.5335   Mean   :68.97   Mean   : 26.36
##  3rd Qu.: 0.03000   3rd Qu.: 0.4700   3rd Qu.:79.00   3rd Qu.: 36.00
##  Max.   :10.57000   Max.   :82.5300   Max.   :98.00   Max.   :113.00
##                                       NA’s   :8582    NA’s   :8582
##   User_Score          User_Count       Developer            Rating
##  Length:16719       Min.   :    4.0   Length:16719       Length:16719
##  Class :character   1st Qu.:   10.0   Class :character   Class :character
##  Mode  :character   Median :   24.0   Mode  :character   Mode  :character
##                     Mean   :  162.2
##                     3rd Qu.:   81.0
##                     Max.   :10665.0
##                     NA's   :9129
```

vgSales must be examined for any titles with N/A values to delete. There are many null values because the data set is a combination of two different subsets, which means there is a lot of games with missing data.

```
colSums(is.na(vgSales))
```

```
##            Name        Platform Year_of_Release           Genre       Publisher
##               2               0               0               2               0
##        NA_Sales        EU_Sales        JP_Sales     Other_Sales    Global_Sales
##               0               0               0               0               0
##    Critic_Score    Critic_Count      User_Score      User_Count       Developer
##            8582            8582            6704            9129            6623
##          Rating
##            6769
```

```
vgSales <- vgSales[complete.cases(vgSales), ]

str(vgSales)
```

```
## tibble [6,947 x 16] (S3: tbl_df/tbl/data.frame)
##  $ Name           : chr [1:6947] "Wii Sports" "Mario Kart Wii" "Wii Sports Resort" "New Super Mario
##  $ Platform       : chr [1:6947] "Wii" "Wii" "Wii" "DS" ...
##  $ Year_of_Release: chr [1:6947] "2006" "2008" "2009" "2006" ...
##  $ Genre          : chr [1:6947] "Sports" "Racing" "Sports" "Platform" ...
##  $ Publisher      : chr [1:6947] "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
##  $ NA_Sales       : num [1:6947] 41.4 15.7 15.6 11.3 14 ...
##  $ EU_Sales       : num [1:6947] 28.96 12.76 10.93 9.14 9.18 ...
##  $ JP_Sales       : num [1:6947] 3.77 3.79 3.28 6.5 2.93 4.7 4.13 3.6 0.24 2.53 ...
##  $ Other_Sales    : num [1:6947] 8.45 3.29 2.95 2.88 2.84 2.24 1.9 2.15 1.69 1.77 ...
##  $ Global_Sales   : num [1:6947] 82.5 35.5 32.8 29.8 28.9 ...
##  $ Critic_Score   : num [1:6947] 76 82 80 89 58 87 91 80 61 80 ...
##  $ Critic_Count   : num [1:6947] 51 73 73 65 41 80 64 63 45 33 ...
##  $ User_Score     : chr [1:6947] "8" "8.3" "8" "8.5" ...
##  $ User_Count     : num [1:6947] 322 709 192 431 129 594 464 146 106 52 ...
##  $ Developer      : chr [1:6947] "Nintendo" "Nintendo" "Nintendo" "Nintendo" ...
##  $ Rating         : chr [1:6947] "E" "E" "E" "E" ...
```

```
sum(vgSales$Year_of_Release != "N/A")
```

```
## [1] 6826
```

```
sum(vgSales$Publisher != "N/A")
```

```
## [1] 6943
```

```
sum(vgSales$Developer != "N/A")
```

```
## [1] 6947
```

```
sum(vgSales$Rating != "N/A")
```

```
## [1] 6947
```

```
vgSales_YOR <- vgSales[vgSales$Year_of_Release != "N/A", ]
vgSales_Publisher <- vgSales[vgSales$Publisher != "N/A", ]
unique(vgSales$Year_of_Release)
```

```
##  [1] "2006" "2008" "2009" "2005" "2007" "2010" "2013" "2004" "2002" "2001"
## [11] "2011" "2012" "2014" "1997" "1999" "2015" "2016" "2003" "1998" "1996"
## [21] "2000" "N/A"  "1994" "1985" "1992" "1988"
```

```
unique(vgSales$Publisher)
```

```
##   [1] "Nintendo"
##   [2] "Microsoft Game Studios"
##   [3] "Take-Two Interactive"
##   [4] "Sony Computer Entertainment"
##   [5] "Activision"
##   [6] "Ubisoft"
##   [7] "Bethesda Softworks"
##   [8] "Electronic Arts"
##   [9] "SquareSoft"
##  [10] "GT Interactive"
##  [11] "Konami Digital Entertainment"
##  [12] "Square Enix"
##  [13] "Sony Computer Entertainment Europe"
##  [14] "Virgin Interactive"
##  [15] "LucasArts"
##  [16] "505 Games"
##  [17] "Capcom"
##  [18] "Warner Bros. Interactive Entertainment"
##  [19] "Universal Interactive"
##  [20] "RedOctane"
##  [21] "Atari"
##  [22] "Eidos Interactive"
##  [23] "Namco Bandai Games"
##  [24] "Vivendi Games"
##  [25] "MTV Games"
##  [26] "Sega"
##  [27] "THQ"
##  [28] "Disney Interactive Studios"
##  [29] "Acclaim Entertainment"
##  [30] "Midway Games"
##  [31] "Deep Silver"
##  [32] "NCSoft"
##  [33] "Tecmo Koei"
## [34] "Valve Software"
## [35] "Infogrames"
## [36] "Mindscape"
## [37] "Valve"
## [38] "Hello Games"
## [39] "Global Star"
## [40] "Gotham Games"
## [41] "Crave Entertainment"
## [42] "Hasbro Interactive"
## [43] "Codemasters"
## [44] "TDK Mediactive"
## [45] "Zoo Games"
## [46] "Sony Online Entertainment"
## [47] "RTL"
## [48] "D3Publisher"
## [49] "Unknown"
## [50] "Black Label Games"
## [51] "SouthPeak Games"
## [52] "Zoo Digital Publishing"
## [53] "City Interactive"
## [54] "Empire Interactive"
## [55] "Russel"
## [56] "Atlus"
## [57] "Mastertronic"
## [58] "Slightly Mad Studios"
## [59] "Play It"
## [60] "Tomy Corporation"
## [61] "Focus Home Interactive"
## [62] "Game Factory"
## [63] "Titus"
## [64] "Marvelous Entertainment"
## [65] "Genki"
## [66] "TalonSoft"
## [67] "SCi"
## [68] "Rage Software"
## [69] "Ubisoft Annecy"
## [70] "Rising Star Games"
## [71] "Enix Corporation"
## [72] "Level 5"
## [73] "Koch Media"
## [74] "Square EA"
## [75] "Touchstone"
## [76] "Spike"
## [77] "Nippon Ichi Software"
## [78] "Sony Computer Entertainment America"
## [79] "Illusion Softworks"
## [80] "Interplay"
## [81] "Metro 3D"
## [82] "Rondomedia"
## [83] "Ghostlight"
## [84] "Majesco Entertainment"
## [85] "PQube"
## [86] "Trion Worlds"
## [87] "Xseed Games"
##  [88] "Ignition Entertainment"
##  [89] "Kadokawa Shoten"
##  [90] "Natsume"
##  [91] "Square"
##  [92] "Gamebridge"
##  [93] "Midas Interactive Entertainment"
##  [94] "ASCII Entertainment"
##  [95] "Rebellion"
##  [96] "N/A"
##  [97] "Harmonix Music Systems"
##  [98] "Activision Blizzard"
##  [99] "Xplosiv"
## [100] "System 3 Arcade Software"
## [101] "Wanadoo"
## [102] "NovaLogic"
## [103] "BAM! Entertainment"
## [104] "Tetris Online"
## [105] "Psygnosis"
## [106] "Screenlife"
## [107] "GungHo"
## [108] "Jester Interactive"
## [109] "Black Bean Games"
## [110] "3DO"
## [111] "Takara Tomy"
## [112] "Sammy Corporation"
## [113] "Kalypso Media"
## [114] "Hudson Soft"
## [115] "Marvelous Interactive"
## [116] "Home Entertainment Suppliers"
## [117] "Arc System Works"
## [118] "Banpresto"
## [119] "Wargaming.net"
## [120] "Destineer"
## [121] "Pacific Century Cyber Works"
## [122] "PopCap Games"
## [123] "Indie Games"
## [124] "FuRyu"
## [125] "Nihon Falcom Corporation"
## [126] "Gathering of Developers"
## [127] "Oxygen Interactive"
## [128] "DTP Entertainment"
## [129] "Falcom Corporation"
## [130] "Kemco"
## [131] "Milestone S.r.l."
## [132] "AQ Interactive"
## [133] "Agetec"
## [134] "XS Games"
## [135] "Activision Value"
## [136] "Telltale Games"
## [137] "Zushi Games"
## [138] "CCP"
## [139] "Agatsuma Entertainment"
## [140] "Compile Heart"
## [141] "Mad Catz"
## [142] "Gust"
## [143] "Media Rings"
## [144] "JoWood Productions"
## [145] "Brash Entertainment"
## [146] "Funcom"
## [147] "Jaleco"
## [148] "Playlogic Game Factory"
## [149] "Human Entertainment"
## [150] "Fox Interactive"
## [151] "Scholastic Inc."
## [152] "System 3"
## [153] "Nordic Games"
## [154] "White Park Bay Software"
## [155] "EA Games"
## [156] "Acquire"
## [157] "Paradox Interactive"
## [158] "Yacht Club Games"
## [159] "Swing! Entertainment"
## [160] "Hip Interactive"
## [161] "Tripwire Interactive"
## [162] "Enterbrain"
## [163] "Havas Interactive"
## [164] "Sting"
## [165] "Idea Factory"
## [166] "Funsta"
## [167] "Tru Blu Entertainment"
## [168] "Moss"
## [169] "From Software"
## [170] "NDA Productions"
## [171] "Bigben Interactive"
## [172] "Idea Factory International"
## [173] "O-Games"
## [174] "Funbox Media"
## [175] "Valcon Games"
## [176] "PM Studios"
## [177] "Bohemia Interactive"
## [178] "Aqua Plus"
## [179] "Ackkstudios"
## [180] "HMH Interactive"
## [181] "inXile Entertainment"
## [182] "Cave"
## [183] "Microids"
## [184] "Phantom EFX"
## [185] "Evolved Games"
## [186] "O3 Entertainment"
## [187] "Aspyr"
## [188] "Nobilis"
## [189] "Sunsoft"
## [190] "DSI Games"
## [191] "Little Orbit"
## [192] "Telegames"
## [193] "The Adventure Company"
## [194] "Popcorn Arcade"
## [195] "Insomniac Games"
## [196] "Aksys Games"
## [197] "Taito"
## [198] "Reef Entertainment"
## [199] "Irem Software Engineering"
## [200] "Myelin Media"
## [201] "Success"
## [202] "SNK"
## [203] "Avalon Interactive"
## [204] "Revolution Software"
## [205] "Gamecock"
## [206] "Groove Games"
## [207] "Hudson Entertainment"
## [208] "Mercury Games"
## [209] "Ascaron Entertainment GmbH"
## [210] "Mastiff"
## [211] "Destination Software, Inc"
## [212] "Graffiti"
## [213] "1C Company"
## [214] "Phantagram"
## [215] "DreamCatcher Interactive"
## [216] "Dusenberry Martin Racing"
## [217] "Navarre Corp"
## [218] "ESP"
## [219] "Team17 Software"
## [220] "Max Five"
## [221] "Conspiracy Entertainment"
## [222] "Milestone S.r.l"
## [223] "Rebellion Developments"
## [224] "Kool Kizz"
## [225] "Monte Christo Multimedia"
## [226] "5pb"
## [227] "Cloud Imperium Games Corporation"
## [228] "Flashpoint Games"
## [229] "Alternative Software"
## [230] "DHM Interactive"
## [231] "Iceberg Interactive"
## [232] "MC2 Entertainment"
## [233] "2D Boy"
## [234] "Gearbox Software"
## [235] "Global A Entertainment"
## [236] "Just Flight"
## [237] "bitComposer Games"
## [238] "Introversion Software"
## [239] "Sold Out"
## [240] "Sunflowers"
## [241] "id Software"
## [242] "Maxis"
## [243] "Pinnacle"
## [244] "Xicat Interactive"
## [245] "Devolver Digital"
## [246] "Number None"
## [247] "TopWare Interactive"
## [248] "Strategy First"
## [249] "Lexicon Entertainment"
## [250] "GOA"
## [251] "Avanquest"
## [252] "Graphsim Entertainment"
## [253] "Codemasters Online"
## [254] "Stainless Games"
## [255] "10TACLE Studios"
## [256] "FuRyu Corporation"
## [257] "Visco"
## [258] "Crimson Cow"
## [259] "Lighthouse Interactive"
## [260] "CDV Software Entertainment"
## [261] "Encore"
## [262] "Blue Byte"
## [263] "NewKidCo"
```

Critic_Score and User_Score are character fields, so they must be converted to numeric fields. Critic_Count and User_Count must also be converted to numeric fields. Dividing the Critic_Score variable by 10 will allow the critic score to be comparable to the user score as they will be in the same decimal place.

```
vgSales$Critic_Score <- as.numeric(as.character(vgSales$Critic_Score))
vgSales$Critic_Score <- vgSales$Critic_Score / 10

vgSales$User_Score <- as.numeric(as.character(vgSales$User_Score))

vgSales$Critic_Count <- as.numeric(vgSales$Critic_Count)

vgSales$User_Count <- as.numeric(vgSales$User_Count)
```

It is important to remember to look for outliers in the sales variables as well. After viewing each summary, there seems to be no extreme outliers with potential to severely alter the data.

```
summary(vgSales$EU_Sales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0200  0.0600  0.2346  0.2100 28.9600
```

```
summary(vgSales$JP_Sales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.00000 0.00000 0.00000 0.06324 0.01000 6.50000
```

```
summary(vgSales$NA_Sales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0600  0.1500  0.3928  0.3900 41.3600
```

```
summary(vgSales$Global_Sales)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0100  0.1100  0.2900  0.7731  0.7500 82.5300
```

```
summary(vgSales$Other_Sales)
```

```
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
##  0.00000  0.01000  0.02000  0.08219  0.07000 10.57000
```

The rating groups “AO,” “K-A,” and “RP” contain only 1-2 records. Therefore, categorizing the rating groups together as shown below will not only simplify the data, but it will prevent data distortion caused by these three ratings with very few records in particular.

```
table(vgSales$Rating)
```

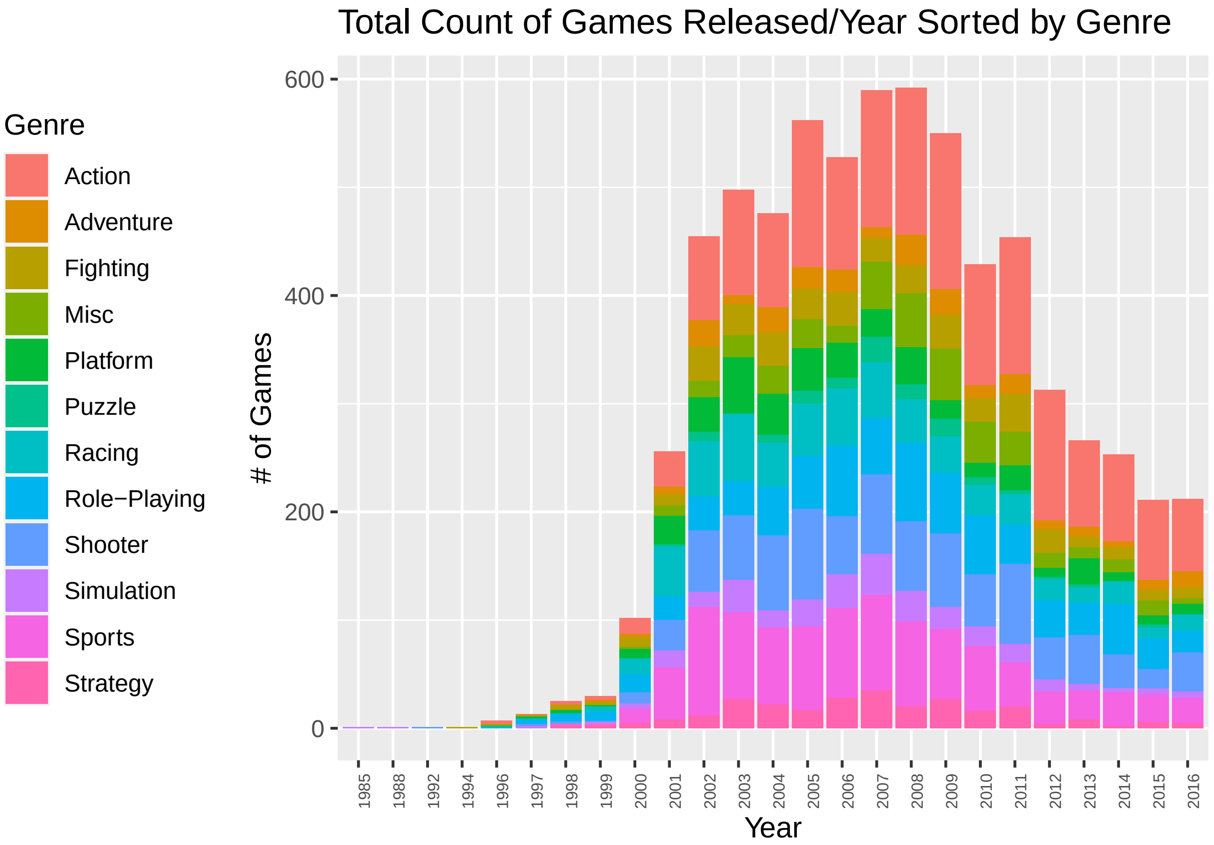
```
##
##   AO    E E10+  K-A    M   RP    T
##    1 2118  946    1 1459    2 2420
```

```
vgSales <- vgSales %>%
  mutate(Rating = ifelse(Rating == "AO", "M", Rating))
vgSales <- vgSales %>%
  mutate(Rating = ifelse(Rating == "K-A", "E", Rating))
vgSales <- vgSales %>%
  mutate(Rating = ifelse(Rating == "RP", "E", Rating))
```

Modeling Analysis/Results

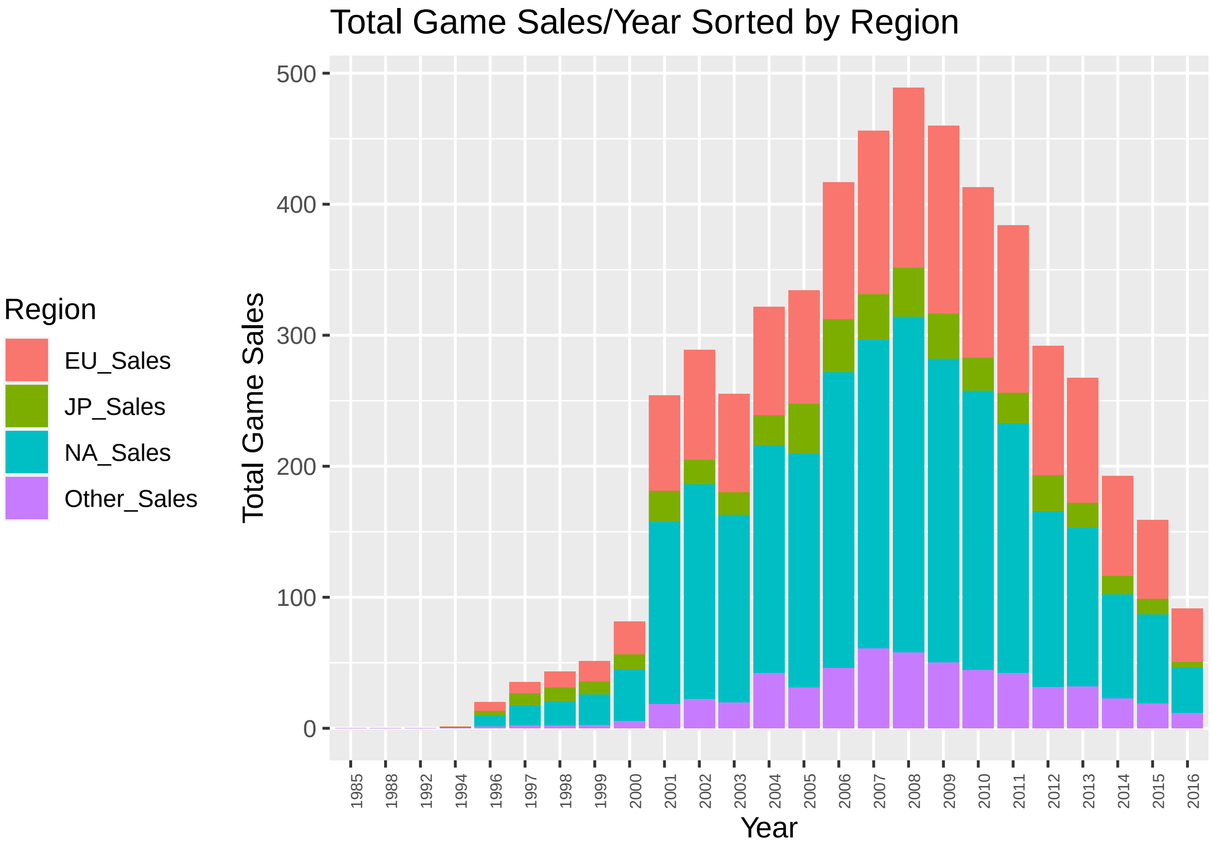
Based on the plot, the years with the most games released were 2007 and 2008. Both of these years were approaching 600 million units, but were unable to reach it. It seems that the genre that consistently sold the most is action.

```
gameGenre <- ggplot(vgSales_YOR, aes(Year_of_Release)) +
  geom_bar(stat = "count", aes(fill = Genre)) +
  labs(title = "Total Count of Games Released/Year Sorted by Genre", x = "Year", y = "# of Games") +
  theme(legend.position = "left", axis.text.x = element_text(angle = 90, hjust = 0.8, size = 6))
gameGenre
```



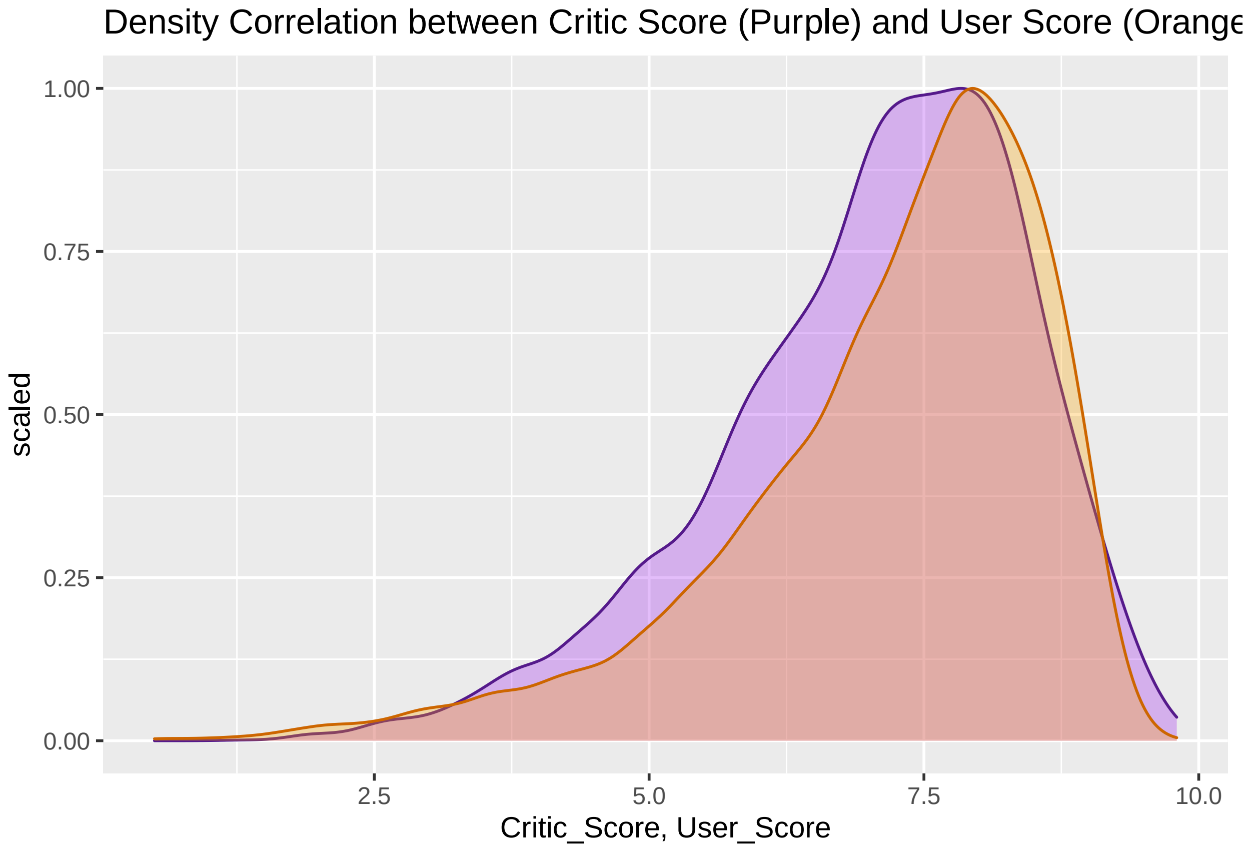
Based on the plot, the year with the most game sales was 2008. This plot is unimodal, so it peaks at only 2008 instead of 2 different years like the previous graph. At this peak, the total game sales were approaching 500 million units, but were unable to reach it. It seems that the region in which the most games consistently sold was North America.

```
vgSales_wwSales <- vgSales_YOR %>%
  select(Year_of_Release, EU_Sales, JP_Sales, NA_Sales, Other_Sales)

vgSales_byRegion <- gather(vgSales_wwSales, Region, TotalGameSales, EU_Sales:Other_Sales)

wwSales <- ggplot(vgSales_byRegion, aes(x = Year_of_Release, y = TotalGameSales, fill = Region)) +
  geom_bar(stat = "identity") +
  labs(title = "Total Game Sales/Year Sorted by Region", x = "Year", y = "Total Game Sales") +
  theme(legend.position = "left", axis.text.x = element_text(angle = 90, hjust = 0.8, size = 6))
wwSales
```



Overall, the critic score is slightly lower than the user score. Also, there is clearly a positive correlation between critic and user score.

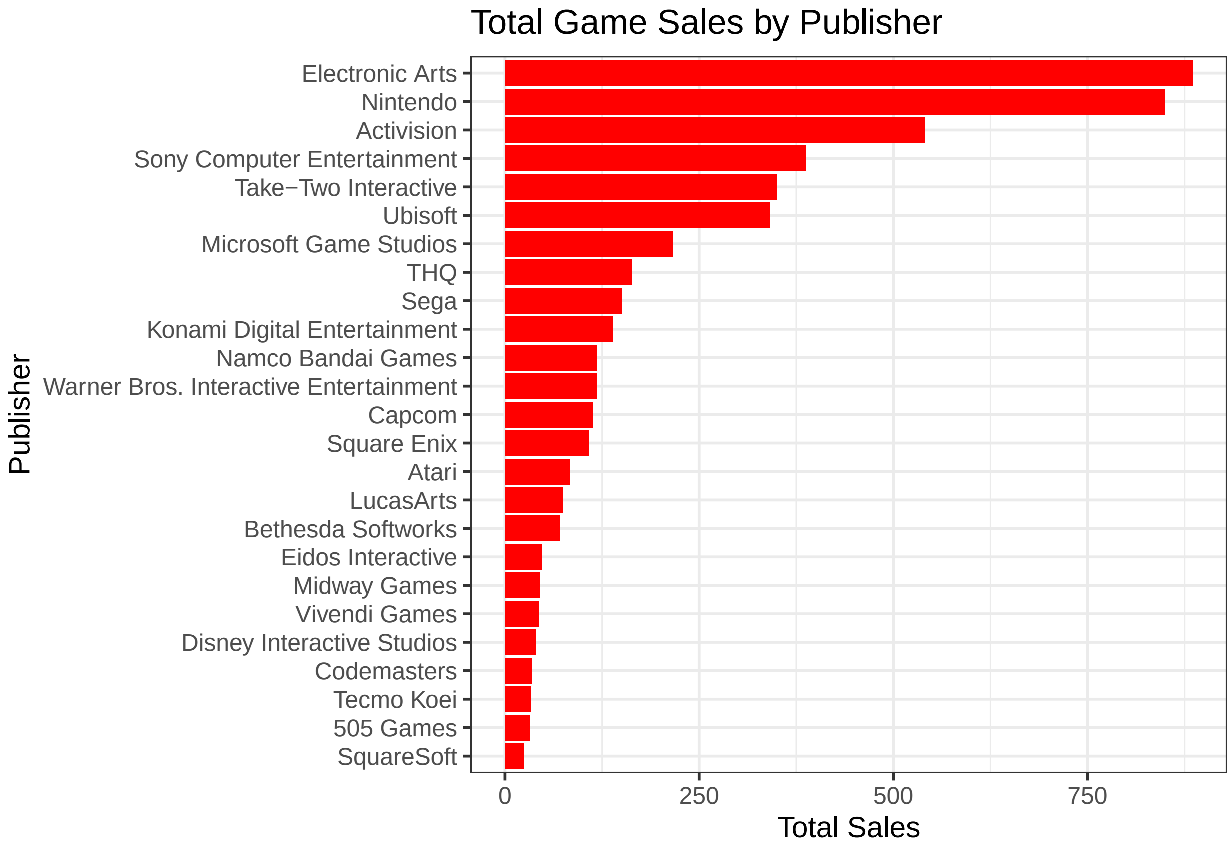
```
scoresPlot <- ggplot() +
  geom_density(data = vgSales, aes(x = Critic_Score, y = ..scaled..), color = "purple4", fill = "purple
  geom_density(data = vgSales, aes(x = User_Score, y = ..scaled..), color = "darkorange3", fill = "orang
  labs(title = "Density Correlation between Critic Score (Purple) and User Score (Orange)", x = "Critic
scoresPlot
```



Conclusion

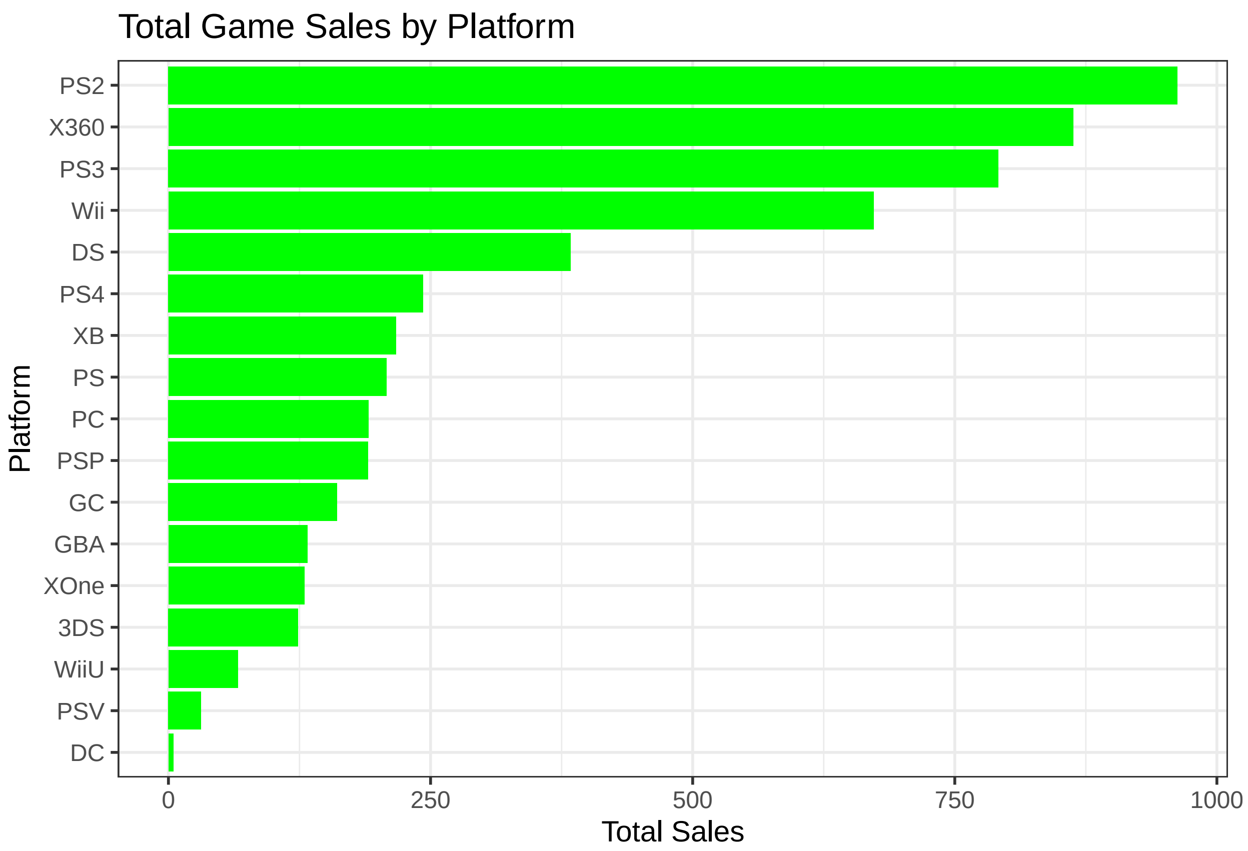
This graph displays the top 25 game publishers based on total game sales. The most successful game publishers are Electronic Arts and Nintendo by a large margin. Although Electronic Arts is slightly ahead in total sales, both publishers are comparable at approximately 875 million units sold.

```
salesbyPublisher <- vgSales %>%
  group_by(Publisher) %>%
  summarise(pubSales = sum(Global_Sales)) %>%
  arrange(desc(pubSales)) %>%
  head(n = 25)

pubPlot <- ggplot(salesbyPublisher, aes(x = reorder(Publisher, pubSales), y = pubSales)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(title = "Total Game Sales by Publisher", x = "Publisher", y = "Total Sales") +
  theme_bw() +
  coord_flip()
pubPlot
```



This graph displays the top 25 game platforms based on total game sales. The most successful game platform is PS2, with Xbox 360 close behind it. This graph is distributed much more evenly than the previous graph. PS2 has successfully sold approximately 960 million units, which is about 100 million more units sold than Xbox 360.

```r
salesbyPlatform <- vgSales %>%
  group_by(Platform) %>%
  summarise(platSales = sum(Global_Sales)) %>%
  arrange(desc(platSales)) %>%
  head(n = 25)

platPlot <- ggplot(salesbyPlatform, aes(x = reorder(Platform, platSales), y = platSales)) +
  geom_bar(stat = "identity", fill = "green") +
  labs(title = "Total Game Sales by Platform", x = "Platform", y = "Total Sales") +
  theme_bw() +
  coord_flip()
platPlot
```



Based on the results of this t-test, the mean of the critic score is approximately 7.026357 and the mean of the user score is approximately 7.183360. Also, $p < 0.05$, meaning that it can be said with 95% confidence that the difference between the means of the critic score and user score are significant.

```
t.test(vgSales$Critic_Score, vgSales$User_Score)
```

```
##
##  Welch Two Sample t-test
##
## data:  vgSales$Critic_Score and vgSales$User_Score
## t = -6.5361, df = 13872, p-value = 6.533e-11
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -0.2040869 -0.1099192
## sample estimates:
## mean of x mean of y
##  7.026357  7.183360
```

This exploratory data analysis served useful in analyzing the sales and distribution of the video games data set. My primary goal for this project was to determine the correlation between critic scores and user scores, which I successfully accomplished using statistical concepts. After converting the scores to numeric values, running a t-test and plotting a corresponding density plot allowed me to observe that the difference between the means of user and critic scores was very minor (less than 0.05). Additionally, it was visible that the two scores had a positive correlation as their graphs were of very similar shape and followed the same curving patterns.

I was also intrigued by this project because of the other unexpected trends between variables besides User_Score and Critic_Score. For example, it surprised me that the two years with the most video games released were 2008 and 2009. Additionally, I learned that action has consistently been the best selling video game genre, contrary to my initial assumption, sports.

Using ggplot2, I created horizontal bar charts for the top 25 grossing video game publishers and platforms in terms of global sales. I noticed that in the publishers' plot, Electronic Arts and Nintendo were the most successful publishers in terms of sales by a large margin. The plot was very skewed and unevenly distributed. However, the platform plot was more evenly distributed and contained numerous close competitors. Although PS2 and Xbox 360 were the leading platforms, PS3, Wii, and a few others were close. I expected Wii to be the most popular platform as Nintendo is very popular in Japan, which is what I considered the hub of gaming. To my surprise, the Americans are taking over the gaming scene as their platforms are the highest grossing, and a majority of video game sales also occur in North America compared to Europe, Japan, and other countries.

Future work definitely needs to be done on this data set to account for the games that were eliminated because they contained N/A values. With the addition of games that were missing information, I believe that the results and observed trends would have been completely different. Instead of inputting 100% accurate values for missing categories, perhaps a good idea would be to input a projected value in order to make sure all 16,000+ games can be used in the data analysis.