Proj 2 temp

ArunK

10/18/2020

Videogames represent a large and growing proportion of the economy and consumer spending, largely due to the rise of E-sports; E-sports has drawn in competitive talent from across the world, as well as the attention of entrepreneurs, game developers, and viewers. As an avid E-sports fan, I've always been shocked by the huge cash prizes given out at tournaments, and wondered what contributes to a game's monetization. I wanted to explore two other game performance metrics I felt would be intimately related with E-Sports prize pools: consumer sales and game reviews. In order to get some insight to the relationship between a game's consumer success, audience reception, and E-sports monetization, I found three datasets dealing with individual metrics a number of game titles. The videogame sales dataset includes information on genre, publisher and number of copies sold by region (in millions) for over 15,000 games across over 100 platforms. The E-sports dataset has information on number of E-Sports tournaments held, total cash prize pool and total online cash prize pool for 492 unique games. The review dataset has both Metacritic and User score for 10,451 unique titles. All 3 datasets were found on kaggle's data repository, posted by different individuals.

I initially expected to see a strong positive association between both consumer sales and reception with E-sports distributions; as more people purchase and enjoy a game, I believe that the competitive pool would grow, enabling higher-skill competition, and more viewership of potential events; both of these factors should incentivize tournament hosts and sponsors to offer larger prize-pools, thereby solidifying their own spot in the market and attracting new talent. Despite this belief, I feel that certain games might be more conducive than others to high E-sports prize pools; factors like genre or platform might inherently make games more competitive or appealing to viewers, which would result in increased prize distributions. I also believe that E-sports distributions and review score would increase with publishing date; I think modern publishers are more capable of incorporating user feedback when making games, and would be able to quickly iron out poor reviews. Additionally, given the cash inflow to the E-Sports industry as awareness of the industry improves, I'd expect prize-pools to increase to accomodate larger venues, higher competition, greater viewership, and overall attract new talent to a game. Modern E-Sports players are often able to compete in multiple games within a genre, and publishers would be hesitant to let competitors steal their player-base; the easiest way to combat this would be to offer larger prize pools than any prior game in the same genre. However, there are several cult-classic games that maintain their prevalence in E-Sports and the videogame community despite their age and relatively low prize pools, suggesting that E-sports prize pools and game sales may be unrelated in certain cases.

```
library(tidyverse)
```

```
## -- Attaching packages ---------------------------------------------------------------------------
```

```
## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0
```

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

```
esports <- read.csv("GeneralEsportData.csv")
vg_sales <- read.csv("vg_sales.csv")
vg_reviews <- read.csv("result.csv")
inner_join(esports, vg_sales, by = c("Game" = "Name")) -> vg_dat

vg_dat %>% mutate(Name_Console = paste(Game, Platform, sep = "_")) -> vg_dat_cl
vg_reviews %>%mutate(Name_Console = paste(name, console, sep = "_")) -> vg_reviews_cl

inner_join(vg_dat_cl, vg_reviews_cl, by = "Name_Console") -> vg_dat_full
length(unique(esports$Game))
```

```
## [1] 492
```

```
glimpse(vg_reviews)
```

```
## Rows: 15,647
## Columns: 5
## $ metascore <int> 97, 97, 95, 94, 94, 93, 93, 92, 92, 91, 91, 91, 91, 90, 9...
## $ name      <chr> "Grand Theft Auto V", "Grand Theft Auto V", "The Last of ...
## $ console   <chr> "PS3", "X360", "PS3", "PS3", "PC", "X360", "WIIU", "3DS",...
## $ userscore <chr> "8.3", "8.3", "9.2", "8.5", "8.6", "8.5", "8.9", "9.1", "...
## $ date      <chr> "17-Sep-13", "17-Sep-13", "14-Jun-13", "26-Mar-13", "25-M...
```

```
vg_dat_full %>% select(-Year, -Genre.y, -Name_Console, -name, -date, -console) -> video_games
```

All three datasets were in wide format to begin which was kept for ease of joining; the sales and esports datasets were joined first on game to produce the `vg_dat` dataset. I then used the game and console variables in the `vg_dat` dataset and the review dataset to create the temporary variable `Name_Console`, which was then used to join the datasets; this allowed me to retain granularity in console variable, since I felt this may be important for game performance. The original E-sports dataset includes information on 492 games, while the videogame sales dataset includes data on 16,598 games, and the review dataset contains information for 15,647 games. Since I'm primarily interested in the interaction of E-sports prize pools, game reception, and consumer sales, I chose to perform an inner join for each merge step; the resulting dataset includes games with E-sports prize pool, review and sales information. Including games that only appeared in two or fewer datasets would not be useful for my main topic of interest, and would unnecessarily increase the dataset size. The merge step produced a complete dataset that would be useful in examining the interaction of consumer success and E-sports prize pools; this dataset contained tournament and prize distribution data, as well as sales data by region and platform, for 111 unique games. The choice of data introduced a significant error to the data: the videogame sales data listed the `publisher` of `World of Tanks` for the XBox-360 as `"N/A"`, instead of Wargaming.net, which was manually verified and fixed. The dataset was then pivoted longer for relevant numerical columns.

```
## Conversion of Sales Data
video_games %>% mutate(NA_Sales = NA_Sales * 1000000, EU_Sales = EU_Sales * 1000000, JP_Sales = JP_Sales

video_games %>% mutate(userscore = as.numeric(userscore)*10, score_dev = (userscore - metascore)^2) -> v
video_games %>% mutate(ReleaseDate = as.character(ReleaseDate)) -> video_games
video_games %>% mutate(Dist_per_Tournament = TotalEarnings/TotalTournaments) -> video_games
video_games %>% pivot_longer(c(TotalEarnings:Rank, NA_Sales:Global_Sales, metascore, userscore, score_de

## Global Sales Breakdown
opts <- options(knitr.kable.NA = "")

video_games_long %>% select(Publisher, Metric, Value) %>% filter(Metric == "Global_Sales") %>% group_by
```

Table 1: Global Sales Metrics for Top 10 Publishers

Publisher	Total	StDev	Minimum	Maximum
Activision	155,600,000	4,946,251.51	20,000	14,760,000
Electronic Arts	92,250,000	2,299,688.73	30,000	8,490,000
Microsoft Game Studios	66,300,000	4,126,286.16	20,000	12,140,000
Sony Computer Entertainment	30,840,000	5,520,905.72	4,200,000	14,980,000
Namco Bandai Games	18,060,000	1,522,333.73	320,000	4,050,000
Konami Digital Entertainment	12,500,000	934,930.61	40,000	3,630,000
Capcom	9,890,000	1,340,749.17	20,000	4,190,000
Nintendo	7,070,000		7,070,000	7,070,000
Warner Bros. Interactive Entertainment	5,410,000	1,084,139.44	50,000	2,700,000
Midway Games	3,930,000	49,497.47	1,930,000	2,000,000

```
video_games_long %>% select(Platform, Metric, Value) %>% filter(Metric == "Global_Sales") %>% group_by(
```

Table 2: Global Sales Metrics for Top 10 Platforms

Platform	Total	StDev	Minimum	Maximum
X360	174,690,000	4,242,402.7	70,000	14,760,000
PS3	101,760,000	3,764,458.6	100,000	13,460,000
PS2	61,090,000	4,046,243.5	40,000	14,980,000
PS4	52,320,000	2,548,354.0	40,000	8,490,000
PC	18,890,000	586,147.6	10,000	2,790,000
GC	8,690,000	2,756,043.7	220,000	7,070,000
PSP	2,190,000	390,843.1	130,000	980,000
DS	1,690,000	304,055.9	630,000	1,060,000
GBA	460,000	113,137.1	150,000	310,000
PS	420,000		420,000	420,000

```
video_games_long %>% select(Genre.x, Metric, Value) %>% filter(Metric == "Global_Sales") %>% group_by(G
```

Table 3: Global Sales Metrics for Top 10 Genres

Genre.x	Total	StDev	Minimum	Maximum
First-Person Shooter	213,820,000	4,579,390.3	10,000	14,760,000
Sports	72,790,000	2,154,401.0	10,000	8,490,000
Racing	57,790,000	3,653,802.5	30,000	14,980,000
Fighting Game	53,700,000	1,487,525.4	20,000	7,070,000
Third-Person Shooter	19,130,000	3,187,395.8	20,000	6,760,000
Role-Playing Game	2,300,000		2,300,000	2,300,000
Strategy	1,960,000	163,892.1	30,000	470,000
Puzzle Game	1,040,000	282,842.7	320,000	720,000

```
video_games_long %>% select(ReleaseDate, Metric, Value) %>% filter(Metric == "Global_Sales") %>% group_
```

Table 4: Global Sales Metrics for Top 10 Release Years

ReleaseDate	Total	StDev	Minimum	Maximum
2011	65,290,000	4,413,810	60,000	14,760,000
2007	61,370,000	2,952,773	10,000	12,140,000
2013	55,790,000	3,035,298	110,000	10,210,000
2009	46,200,000	4,019,136	60,000	13,510,000
2010	40,040,000	6,287,898	110,000	14,640,000
2001	27,110,000	5,748,921	40,000	14,980,000
2004	19,560,000	3,796,581	20,000	11,660,000
2014	17,900,000	2,642,708	30,000	7,600,000
2012	17,640,000	2,687,983	20,000	9,760,000
2015	13,660,000	2,530,643	20,000	8,490,000

```
## Publisher Contribution for platforms
video_games_long %>% select(Publisher, Platform, Metric, Value) %>% filter(Metric == "Global_Sales") %>%
```

Table 5: Global Sales Metrics for Top 10 Publisher-Platform Pairs

Publisher	Platform	Total	StDev	Minimum	Maximum
Activision	X360	70,960,000	6,041,794	380,000	14,760,000
Microsoft Game Studios	X360	65,570,000	3,347,812	470,000	12,140,000
Activision	PS3	56,900,000	5,296,440	350,000	13,460,000
Electronic Arts	PS4	26,960,000	3,027,042	200,000	8,490,000
Sony Computer Entertainment	PS2	26,640,000	2,347,595	11,660,000	14,980,000
Electronic Arts	PS3	23,570,000	2,775,438	570,000	7,230,000
Electronic Arts	X360	23,540,000	2,143,010	540,000	7,340,000
Activision	PS4	12,960,000	2,995,213	1,730,000	7,600,000
Namco Bandai Games	PS2	11,360,000	313,422	3,440,000	4,050,000
Electronic Arts	PS2	9,490,000	1,921,516	1,160,000	5,230,000

```
## Genre Contribution for platforms
video_games_long %>% select(Genre.x, Platform, Metric, Value) %>% filter(Metric == "Global_Sales") %>%
```

Table 6: Global Sales Metrics for Top 10 Genre-Platform Pairs

Genre.x	Platform	Total	StDev	Minimum	Maximum
First-Person Shooter	X360	110,370,000	5,528,375.5	340,000	14,760,000
First-Person Shooter	PS3	66,340,000	4,973,378.5	350,000	13,460,000
Racing	PS2	30,360,000	6,720,113.1	1,160,000	14,980,000
Sports	PS4	25,580,000	3,035,551.6	200,000	8,490,000
First-Person Shooter	PS4	21,540,000	2,319,853.4	950,000	7,600,000
Racing	X360	20,660,000	1,732,016.4	1,290,000	5,510,000
Third-Person Shooter	X360	19,110,000	343,947.7	6,110,000	6,760,000
Fighting Game	PS2	16,310,000	1,584,653.5	40,000	4,050,000
Sports	PS3	16,200,000	1,993,276.2	570,000	6,900,000
Fighting Game	PS3	14,560,000	1,226,860.0	100,000	4,190,000

```
## E-Sports Distribution Breakdown
video_games_long %>% select(Publisher, Metric, Value) %>% filter(Metric == "TotalEarnings") %>% group_by
```

Table 7: Total E-Sports Earnings for Top 10 Publishers

Publisher	Average	StDev	Minimum	Maximum
505 Games	4,598,705.1	5,202,202.4	4,901.26	9,103,504.9
Wargaming.net	3,656,496.6		3,656,496.59	3,656,496.6
Nintendo	3,208,949.7		3,208,949.74	3,208,949.7
Activision	1,696,628.0	4,912,198.9	50.00	22,049,333.3
DreamCatcher Interactive	1,015,000.0		1,015,000.00	1,015,000.0
Vivendi Games	941,640.1	1,451,406.9	178,000.00	3,117,222.3
Capcom	477,668.3	835,664.2	300.00	2,202,726.4
NCSoft	476,005.5		476,005.50	476,005.5
Microsoft Game Studios	370,260.8	597,042.0	132.37	2,158,079.4
Warner Bros. Interactive Entertainment	279,354.7	281,952.2	35,176.89	523,532.4

```
video_games_long %>% select(Platform, Metric, Value) %>% filter(Metric == "TotalEarnings") %>% group_by
```

Table 8: Total E-Sports Earnings for Top 10 Platforms

Platform	Average	StDev	Minimum	Maximum
PS4	1,822,807.59	4,816,584.62	0.00	22,049,333.28
PC	979,417.91	3,193,907.53	0.00	22,049,333.28
GC	543,405.85	1,305,912.30	50.00	3,208,949.74
DS	398,062.48	31,493.11	375,793.48	420,331.47
X360	212,674.56	423,495.73	0.00	2,158,079.40
PS3	151,186.80	336,921.54	0.00	1,594,185.25
PSP	39,262.45	54,537.70	2,084.15	119,874.64
3DS	24,591.01		24,591.01	24,591.01
PS2	14,253.44	23,135.13	0.00	89,000.00
PS	7,679.25		7,679.25	7,679.25

```
video_games_long %>% select(Genre.x, Metric, Value) %>% filter(Metric == "TotalEarnings") %>% group_by(
```

Table 9: Total E-Sports Earnings for Top 10 Genres

Genre.x	Average	StDev
First-Person Shooter	1,245,672.74	3,929,595.25
Sports	521,741.72	1,920,184.53
Role-Playing Game	476,005.50	
Strategy	353,366.41	1,043,425.51
Fighting Game	251,067.63	652,911.35
Third-Person Shooter	161,661.74	116,825.78
Racing	34,111.76	53,134.08
Puzzle Game	1,545.00	0.00

```
## Publisher + Platform Distribution Breakdown
video_games_long %>% select(Publisher, Platform, Metric, Value) %>% filter(Metric == "TotalEarnings") %
```

Table 10: Total E-Sports Earnings for Top 10 Publisher-Platform Pairs

Publisher	Platform	Average	StDev	Minimum	Maximum
505 Games	PC	9,103,504.9		9,103,504.89	9,103,505
Activision	PS4	8,750,775.8	11,528,056.6	1,594,185.25	22,049,333
505 Games	PS4	4,643,207.1	6,307,813.6	182,909.30	9,103,505
Wargaming.net	PC	3,656,496.6		3,656,496.59	3,656,497
Nintendo	GC	3,208,949.7		3,208,949.74	3,208,950
Activision	PC	2,660,391.5	6,478,645.9	60,923.12	22,049,333
Vivendi Games	PC	1,705,280.3	1,996,787.6	293,338.23	3,117,222
Capcom	PS4	1,509,719.9	980,059.2	816,713.46	2,202,726
DreamCatcher Interactive	PC	1,015,000.0		1,015,000.00	1,015,000
Capcom	PC	764,918.4	1,245,760.3	7,938.22	2,202,726

```
## Platform + Genre Distribution Breakdown
video_games_long %>% select(Genre.x, Platform, Metric, Value) %>% filter(Metric == "TotalEarnings") %>%
```

Table 11: Total E-Sports Earnings for Top 10 Genre-Platform Pairs

Genre.x	Platform	Average	StDev	Minimum	Maximum
First-Person Shooter	PS4	4,445,973.0	8,683,319.5	46,235.07	22,049,333.3
Fighting Game	GC	1,604,499.9	2,269,034.8	50.00	3,208,949.7
First-Person Shooter	PC	1,440,032.1	4,372,846.0	4,901.26	22,049,333.3
Sports	PC	1,216,724.3	2,995,529.6	0.00	9,103,504.9
Sports	PS4	1,151,631.1	2,830,778.9	0.00	9,103,504.9
Fighting Game	PS4	709,903.3	904,616.2	1,994.00	2,202,726.4
Fighting Game	PC	704,571.9	1,024,294.5	7,938.22	2,202,726.4
First-Person Shooter	X360	514,844.3	627,620.1	5,000.00	2,158,079.4
Role-Playing Game	PC	476,005.5		476,005.50	476,005.5
First-Person Shooter	PS3	417,908.3	534,327.2	5,000.00	1,594,185.2

```
## Average Distribution per Tournament
video_games_long %>% select(Publisher, Platform, Metric, Value) %>% filter(Metric == "Dist_per_Tournamer
```

Table 12: Distribution per Tournament for top publishers

Publisher	Platform	Average	StDev
DreamCatcher Interactive	PC	78,076.92	
Wargaming.net	PC	73,129.93	
Sony Computer Entertainment	PS4	46,235.07	
Tecmo Koei	X360	38,288.10	
Vivendi Games	PS3	35,600.00	
Vivendi Games	X360	35,600.00	
Microsoft Game Studios	X360	27,111.22	33,469.42
Konami Digital Entertainment	PS4	21,084.67	14,600.88
Konami Digital Entertainment	PC	20,165.22	13,606.92
Activision	PS4	19,076.66	11,784.85

```
## Review Breakdown
video_games_long %>% select(Publisher, Metric, Value) %>% filter(Metric %in% c("userscore")) %>% group_b
```

Table 13: Userscore for top 10 Publishers

Metric	Publisher	Average	StDev	Minimum	Maximum
userscore	Nintendo	91.00		91	91
userscore	Virgin Interactive	86.00		86	86
userscore	Unknown	85.50	2.12	84	87
userscore	Atari	85.33	4.62	80	88
userscore	Sony Computer Entertainment	84.33	1.53	83	86
userscore	DreamCatcher Interactive	81.00		81	81
userscore	NCSoft	81.00		81	81
userscore	Sega	81.00	11.31	73	89
userscore	Deep Silver	80.50	0.71	80	81
userscore	Namco Bandai Games	80.00	5.52	70	87

```
video_games_long %>% select(Publisher, Metric, Value) %>% filter(Metric %in% c("metascore")) %>% group_b
```

Table 14: Critic Score for top 10 Publishers

Metric	Publisher	Average	StDev	Minimum	Maximum
metascore	Sony Computer Entertainment	92.33	3.06	89	95
metascore	Nintendo	92.00		92	92
metascore	NCSoft	90.00		90	90
metascore	Sega	89.00	5.66	85	93
metascore	Microsoft Game Studios	85.59	8.67	66	94
metascore	Atari	85.33	8.02	77	93
metascore	Vivendi Games	83.50	7.85	72	89
metascore	THQ	83.33	2.66	80	86
metascore	Capcom	82.33	7.28	71	94
metascore	Namco Bandai Games	81.89	3.10	79	88

```
video_games_long %>% select(Platform, Metric, Value) %>% filter(Metric %in% c("userscore")) %>% group_by
```

Table 15: Userscore for top 10 publishers

Metric	Platform	Average	StDev	Minimum	Maximum
userscore	DC	86.00		86	86
userscore	PS2	83.00	6.48	64	89
userscore	GBA	72.50	9.19	66	79
userscore	GC	71.83	15.33	46	91
userscore	X360	70.35	13.66	26	85
userscore	PSP	69.25	8.54	64	82
userscore	PS	69.00		69	69
userscore	DS	66.50	7.78	61	72
userscore	PS3	66.30	14.22	26	85
userscore	PC	66.21	19.38	21	89

```
video_games_long %>% select(Platform, Metric, Value) %>% filter(Metric %in% c("metascore")) %>% group_by
```

Table 16: Critic score for top 10 publishers

Metric	Platform	Average	StDev	Minimum	Maximum
metascore	X360	83.12	7.78	56	94
metascore	PS2	83.00	7.49	67	95
metascore	PC	82.39	5.80	68	93
metascore	GC	81.83	9.95	70	94
metascore	PS3	81.27	7.19	60	94
metascore	PS4	80.43	7.25	60	93
metascore	DC	80.00		80	80
metascore	PSP	77.00	5.72	69	82
metascore	GBA	76.50	9.19	70	83
metascore	DS	74.50	0.71	74	75

```
video_games_long %>% select(Platform, Metric, Value) %>% filter(Metric %in% c("score_dev")) %>% group_by
```

Table 17: Squared Deviation of Critics and Users

Metric	Platform	Average	StDev
score_dev	PC	524.14	784.88
score_dev	PS4	518.00	549.06
score_dev	PS3	448.36	712.30
score_dev	X360	345.94	612.21
score_dev	GC	257.67	281.44
score_dev	PSP	95.25	101.56
score_dev	DS	89.00	113.14
score_dev	DC	36.00	
score_dev	PS2	35.88	43.03
score_dev	GBA	16.00	0.00

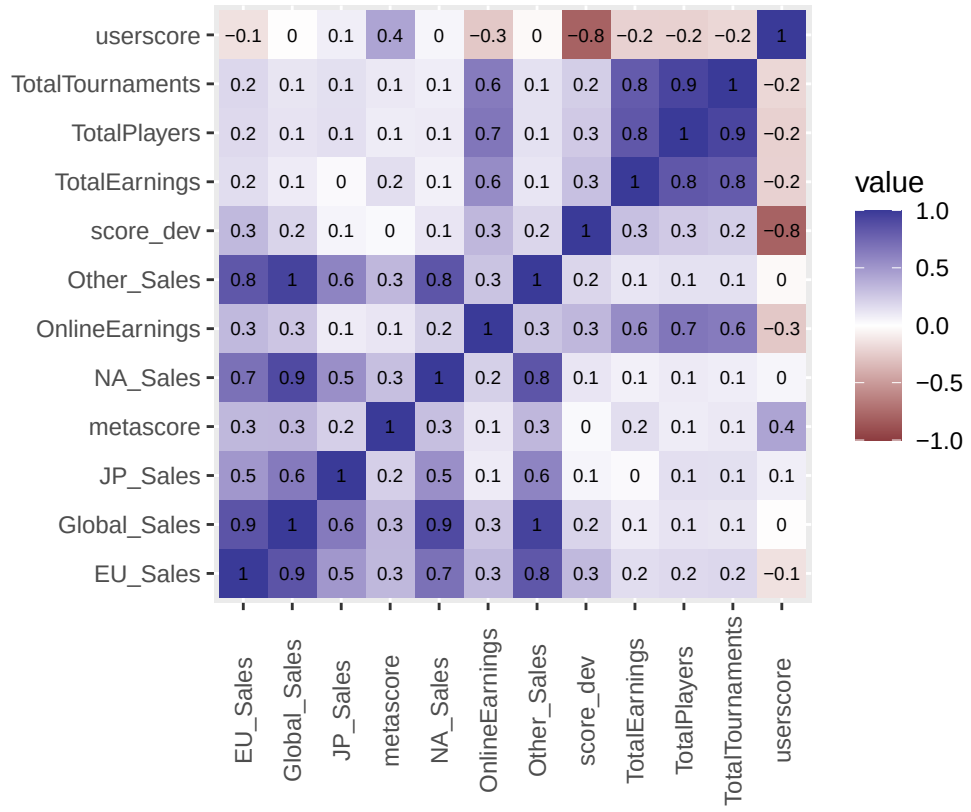
```
## Correlation Matrix
video_games_cordat <- video_games %>% select_if(is.numeric) %>% select(-Dist_per_Tournament,-Rank)
video_games_cor <- video_games_cordat %>% cor(use = "pairwise.complete.obs", method = "spearman")
```

Taking a look first at which publishers dominate the consumer market, well-known players like Activision, EA, and Microsoft Games Ranks highly on global sales, followed by smaller producers like Namco-Bandai and Capcom. It's also clear that the XBox-360 and PS3 are the two most significant gaming platforms by total sales, potentially due to the age of the datasets used; First-Person Shooters, Sports, Fighting, and Racing Games all rank highly for global sales. Activision seems to produce the most consumer-relevant games, making up a significant portion of sales for both the XBox-360 and the PS3.

While 505 Games has the highest average E-Sports prize pool of any publisher, the high standard deviation corresponds to the relative commercial failure of all games except Rocket League, which has become a staple in E-Sports competitions. It seems like the highest average prize pools correspond to games made on the Xbox-One and PS4; the high average prize pool might suggest an increase in spending by sponsors and hosts for tournaments played on modern game platforms, or could result from general increases in viewership. Regardless of platform, First-Person Shooter, Strategy, Fighting, and Sports Games rank highly for average prize pools; this similarity with the genres that had high consumer success suggest that certain genres are more likely to appeal to videogamers, who then drive viewership and prize pools for the game up. Surprisingly,

there was high variability in the average prize pool for each publisher scaled by the number of consumer sales made; this somewhat discredits the stated rationale for increases in prize pools, since certain game's prize pools seem to be dissproportionately large compared to their player base.

```
library(ggpubr)
video_games_cor %>% as.data.frame %>%  rownames_to_column %>% pivot_longer(-1) %>%  na.omit %>%  ggplot
scale_fill_gradient2(low="#8d3c40",high="#3a3a98", limits = c(-1,1)) + theme(axis.text.x = element_text
```



The shown correlation plot indicates weak relationships between metrics for E-Sports prize pools and consumer success; this suggests that the two metrics of game performance may not be related in the way I initially thought, or that there are other factors which contribute to E-Sports prize pool which are independent of consumer sales.

```
library(ggridges)
```

```
## Warning: package 'ggridges' was built under R version 4.0.3
```

```
library(gridExtra)
```

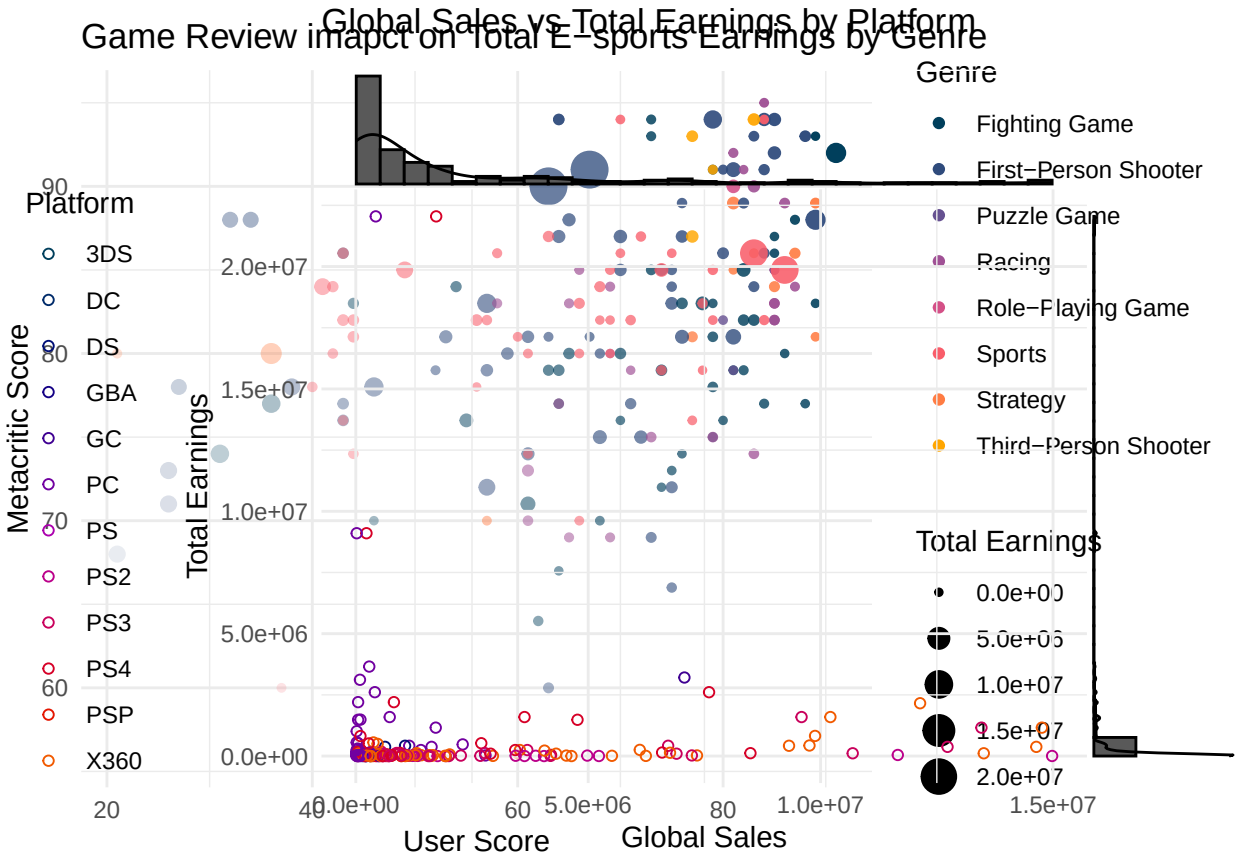
```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(gtable)
library(grid)
require(scales)
```

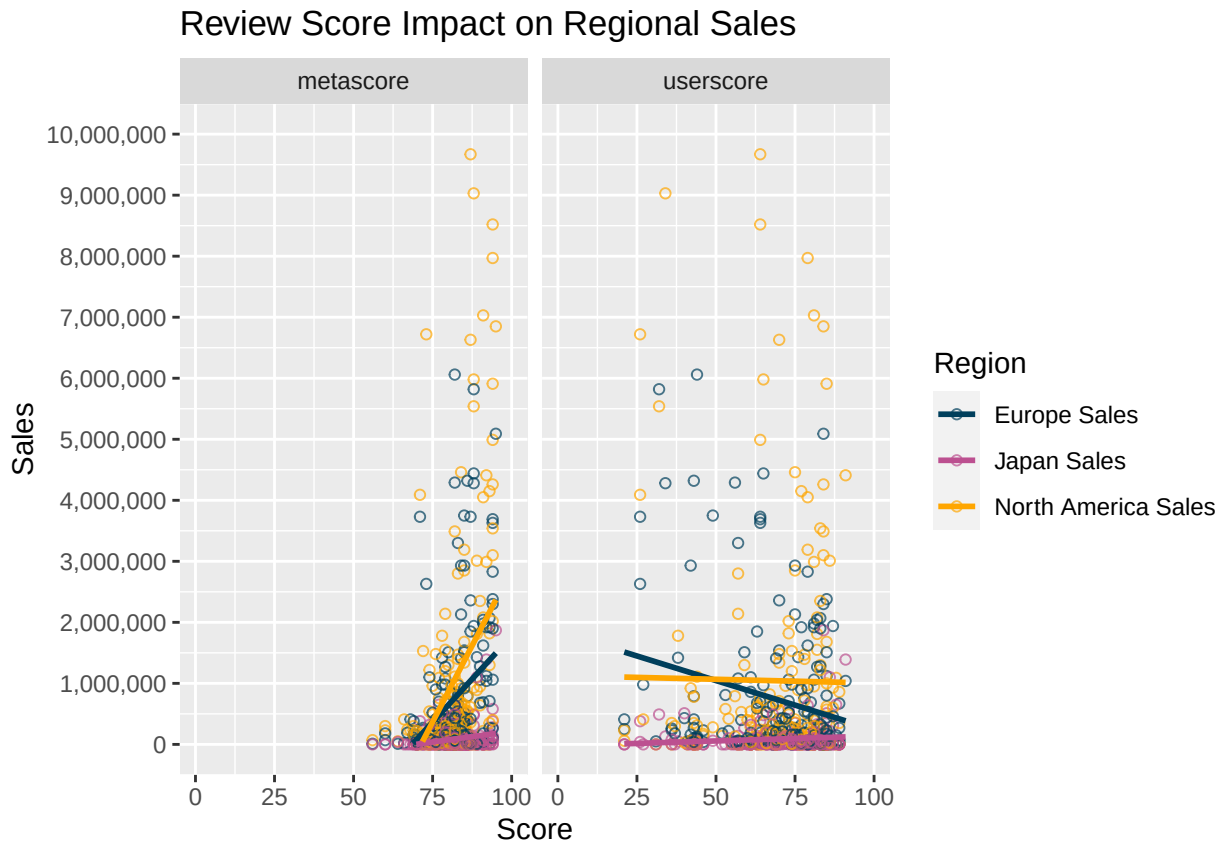
```
## Loading required package: scales
##
## Attaching package: 'scales'
## The following object is masked from 'package:purrr':
##
##     discard
## The following object is masked from 'package:readr':
##
##     col_factor
```

```
cust_pallete13 <- c("#003f5c","#002b6a","#000f77","#140085","#3f0092","#7100a0","#ab00ae","#bb008a","#c
cust_pallete8 <- c("#003f5c","#2f4b7c","#665191","#a05195","#d45087","#f95d6a","#ff7c43","#ffa600")
cust_pallete3 <- c("#003f5c","#bc5090","#ffa600")

video_games %>% group_by(Genre.x) %>% ggplot(aes(x = userscore, y = metascore, color = Genre.x)) + geom_

video_games %>% group_by(Platform) %>% select(Platform, TotalEarnings, Global_Sales) %>% ggplot(aes(y =
scatter %>% ggExtra::ggMarginal(type = "densigram")
```




```
video_games %>% select(userscore, metascore, NA_Sales, EU_Sales, JP_Sales) %>% pivot_longer(userscore:me
```

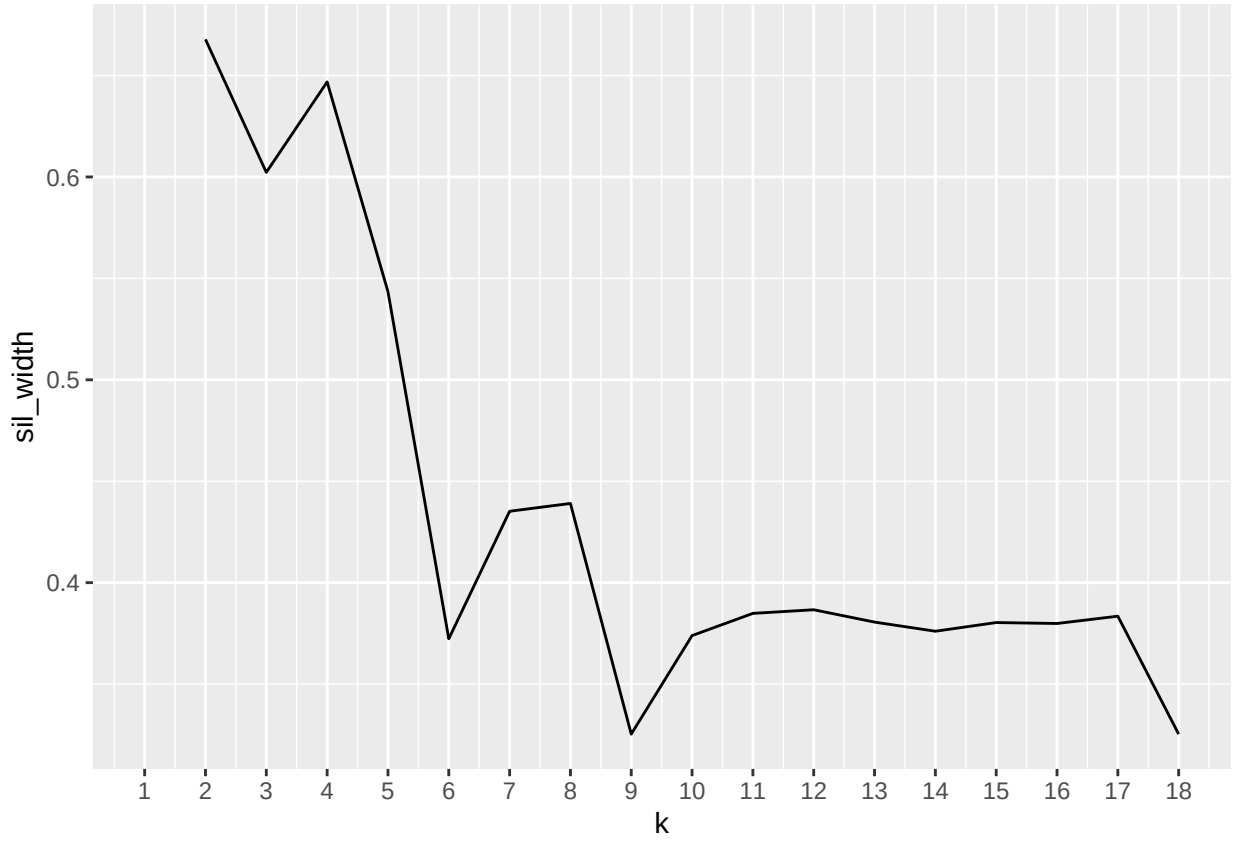
```
## `geom_smooth()` using formula 'y ~ x'
## Warning: Removed 86 rows containing missing values (geom_smooth).
```



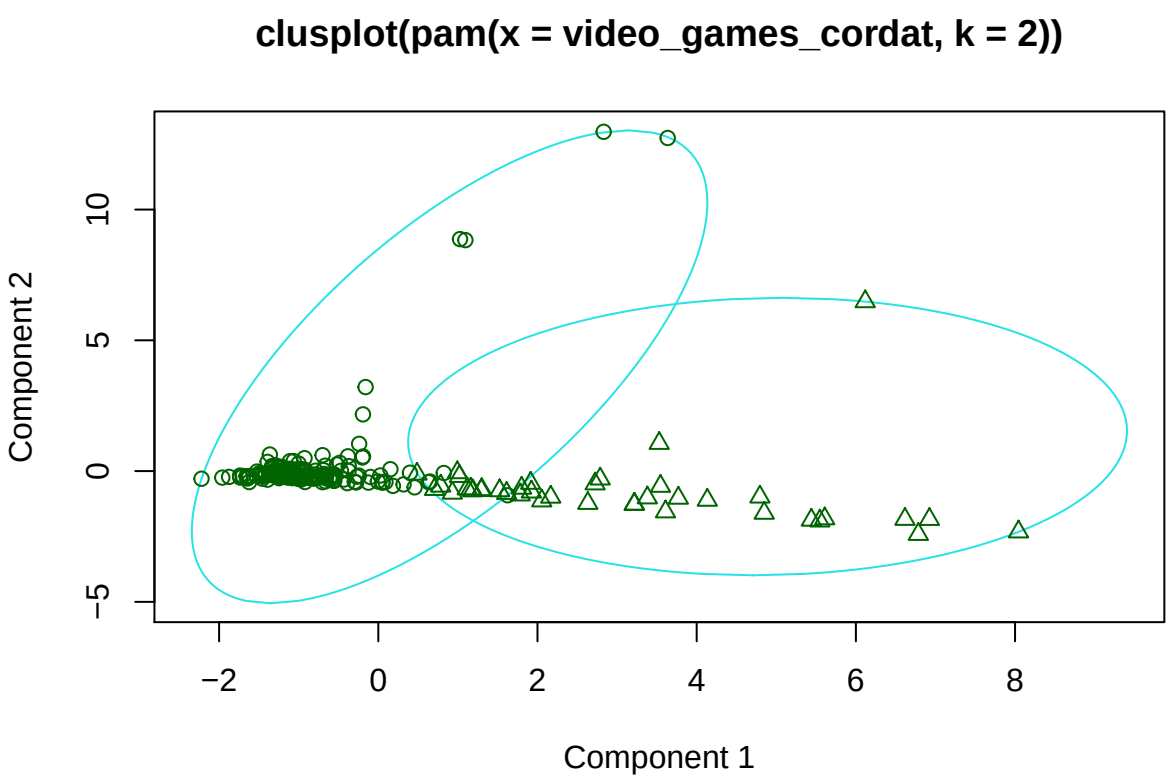
As shown above, there seems to be a general increase in E-Sports prize pools for the highest

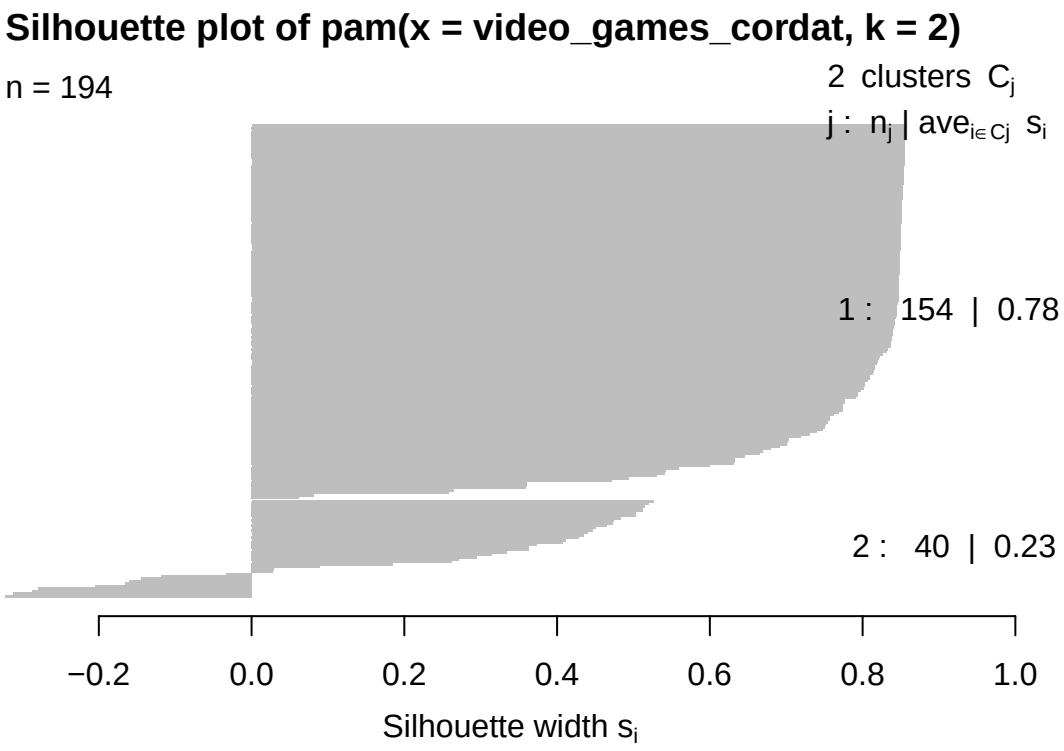
```
library(cluster)
#PAM
sil_width <- c()

for (i in 1:18) {
  pam1 <- pam(video_games_cordat, k = i)
  sil_width[i] <- pam1$silinfo$avg.width
}
ggplot() + geom_line(aes(x=1:18, y = sil_width)) + scale_x_continuous(name = "k", breaks = 1:18)
```

```
## Warning: Removed 1 row(s) containing missing values (geom_path).
```




```
plot(pam(video_games_cordat, 2))
```

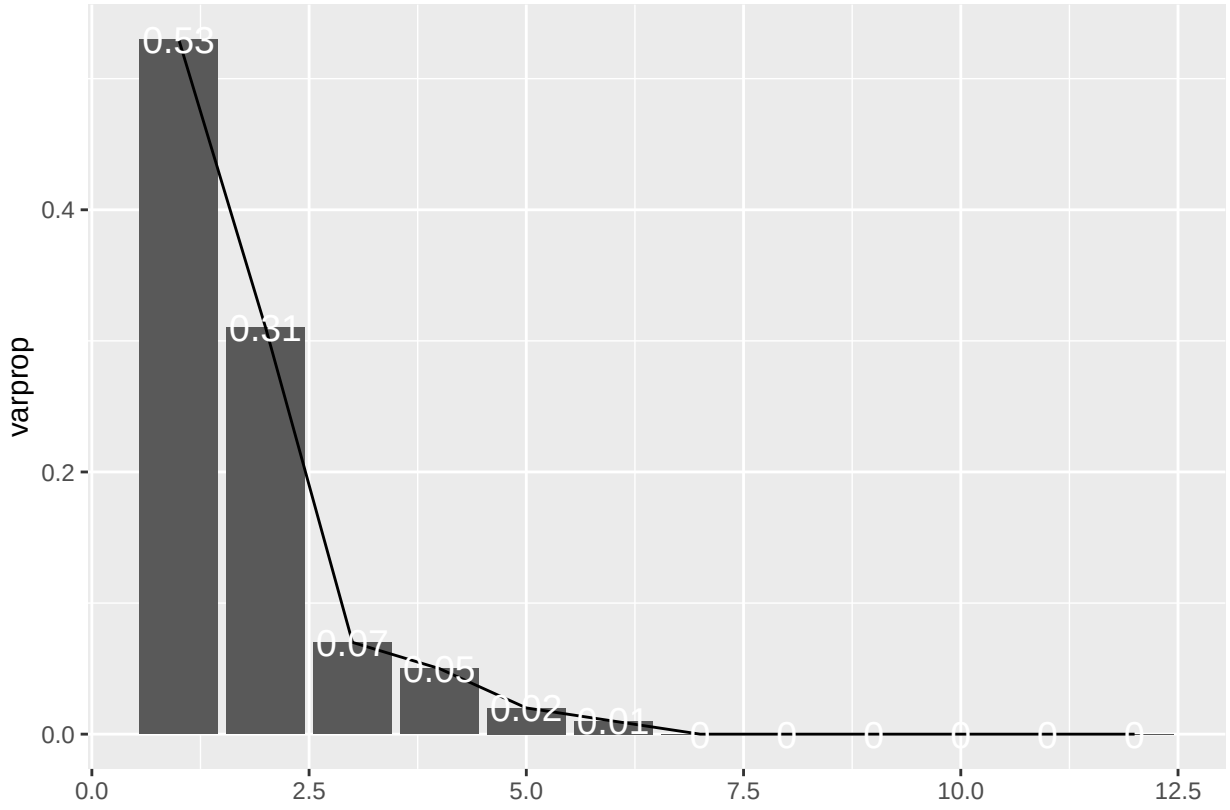

clusplot(pam(x = video_games_cordat, k = 2))
Component 2
10
5
0
−5
−2
0
2
4
6
8
Component 1
These two components explain 57.66 % of the point variability.




```
video_games_cor %>%  scale() -> vg_pca
vg_pca1 <- princomp(vg_pca)
summary(vg_pca1)
```

```
## Importance of components:
##                           Comp.1    Comp.2     Comp.3    Comp.4     Comp.5
## Standard deviation     2.4050153 1.8443720 0.88670100 0.7632699 0.50092244
## Proportion of Variance 0.5258271 0.3092462 0.07147624 0.0529619 0.02281121
## Cumulative Proportion  0.5258271 0.8350733 0.90654959 0.9595115 0.98232270
##                           Comp.6      Comp.7      Comp.8       Comp.9
## Standard deviation     0.34836167 0.216555935 0.130959831 0.0805001718
## Proportion of Variance 0.01103235 0.004263316 0.001559134 0.0005891162
## Cumulative Proportion  0.99335505 0.997618364 0.999177499 0.9997666147
##                            Comp.10      Comp.11 Comp.12
## Standard deviation     0.0445760109 2.408770e-02       0
## Proportion of Variance 0.0001806382 5.274702e-05       0
## Cumulative Proportion  0.9999472530 1.000000e+00       1
```

```
eigval <- vg_pca1$sdev^2
varprop = round(eigval/sum(eigval), 2)

ggplot() + geom_bar(aes(y = varprop, x = 1:12), stat = "identity") + xlab("") + geom_path(aes(x = 1:12,y
```

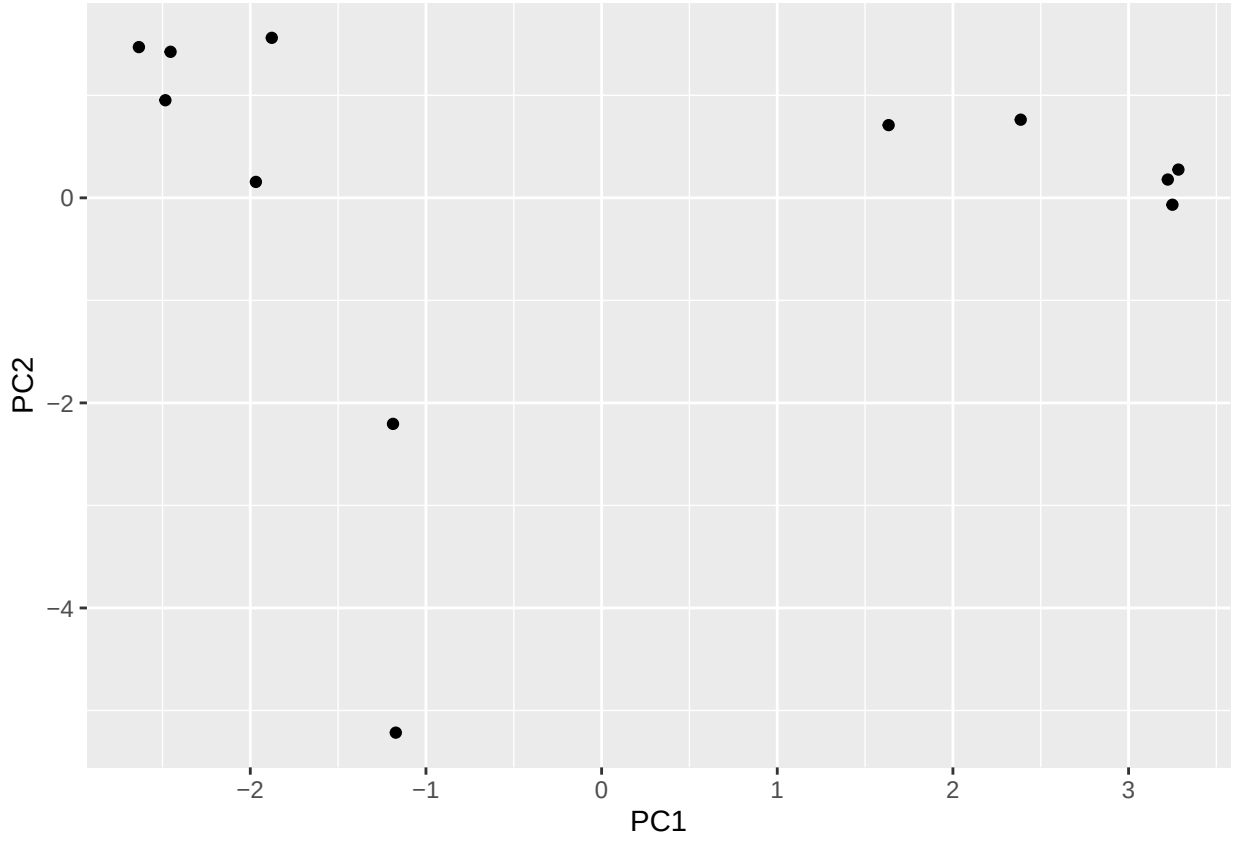



```
summary(vg_pca1, loadings = T)
```

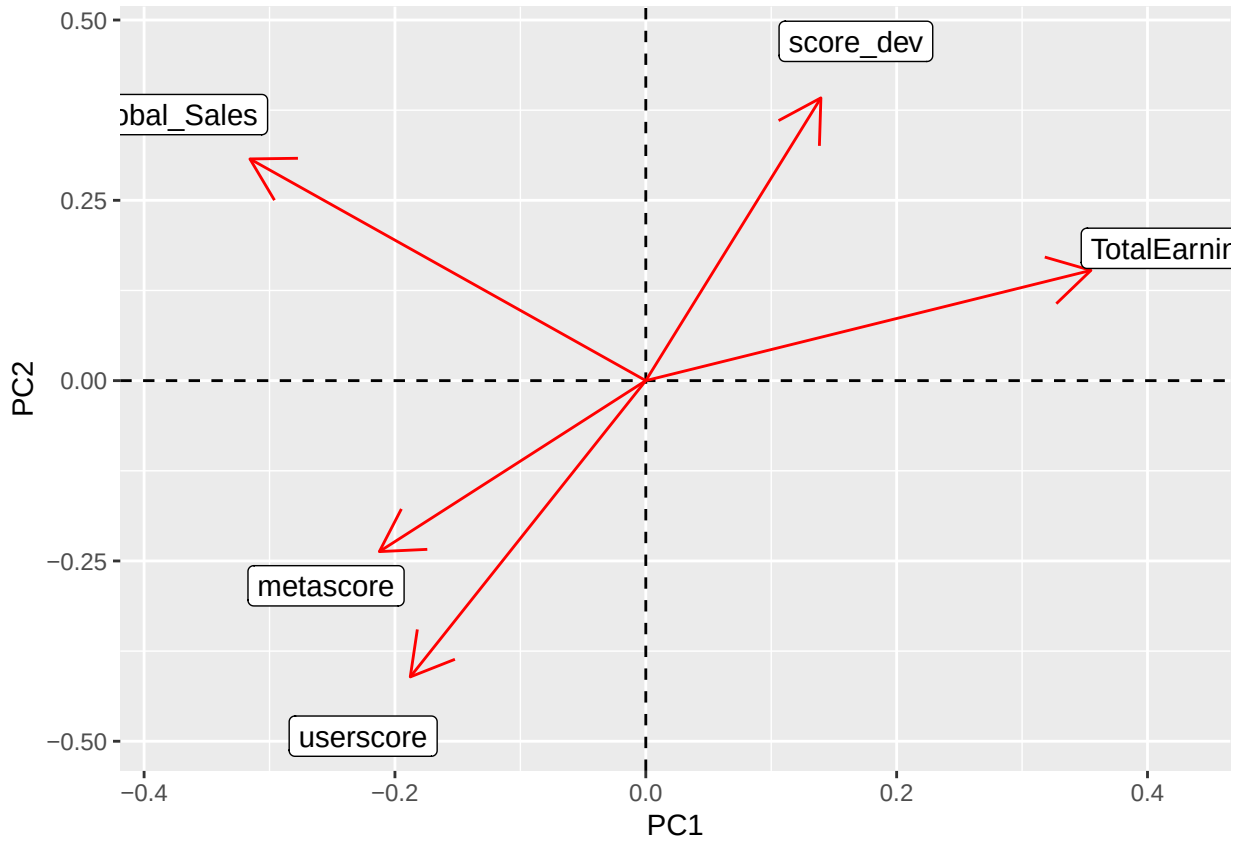
```
## Importance of components:
##                           Comp.1    Comp.2     Comp.3    Comp.4     Comp.5
## Standard deviation     2.4050153 1.8443720 0.88670100 0.7632699 0.50092244
## Proportion of Variance 0.5258271 0.3092462 0.07147624 0.0529619 0.02281121
## Cumulative Proportion  0.5258271 0.8350733 0.90654959 0.9595115 0.98232270
##                           Comp.6      Comp.7      Comp.8       Comp.9
## Standard deviation     0.34836167 0.216555935 0.130959831 0.0805001718
## Proportion of Variance 0.01103235 0.004263316 0.001559134 0.0005891162
## Cumulative Proportion  0.99335505 0.997618364 0.999177499 0.9997666147
##                            Comp.10      Comp.11 Comp.12
## Standard deviation     0.0445760109 2.408770e-02       0
## Proportion of Variance 0.0001806382 5.274702e-05       0
## Cumulative Proportion  0.9999472530 1.000000e+00       1
##
## Loadings:
##                  Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## TotalEarnings     0.355  0.153  0.284  0.117  0.137  0.440         0.721  0.110
## OnlineEarnings    0.282  0.296  0.214  0.156 -0.398 -0.742  0.106  0.199
## TotalPlayers      0.349  0.174  0.356         0.135                -0.334
## TotalTournaments  0.349  0.164  0.373         0.146  0.111         -0.507
## NA_Sales         -0.323  0.271  0.170        -0.241  0.228  0.609 -0.166  0.441
## EU_Sales         -0.255  0.379         0.183 -0.104        -0.745         0.404
## JP_Sales         -0.298  0.199  0.219 -0.400  0.696 -0.373         0.160
## Other_Sales      -0.306  0.319  0.153        -0.126  0.126               -0.772
```

```
## Global_Sales     -0.316  0.307  0.134        -0.108                        -0.105
## metascore        -0.212 -0.237  0.220  0.826  0.328 -0.119  0.104
## userscore        -0.188 -0.411  0.410        -0.146
## score_dev         0.140  0.392 -0.521  0.287  0.273         0.162
##                  Comp.10 Comp.11 Comp.12
## TotalEarnings
## OnlineEarnings
## TotalPlayers     -0.761
## TotalTournaments  0.633
## NA_Sales                 -0.278
## EU_Sales                 -0.133
## JP_Sales
## Other_Sales              -0.318   0.199
## Global_Sales              0.795  -0.339
## metascore                       -0.167
## userscore                 0.315   0.703
## score_dev                 0.245   0.554
```

```
data.frame(PC1 = vg_pca1$scores[,1], PC2 = vg_pca1$scores[,2] ) %>% ggplot(aes(PC1, PC2)) + geom_point()
```




```
vg_pca1$loadings[c(1,9,10,11,12), 1:2] %>% as.data.frame %>% rownames_to_column() %>% ggplot + geom_hli
  geom_vline(aes(xintercept = 0), lty = 2) + ylab("PC2") + xlab("PC1") +
  geom_segment(aes(x=0,y=0, xend = Comp.1, yend = Comp.2), arrow = arrow(), col = "red") +
  geom_label(aes(x = Comp.1 *1.2, y = Comp.2 *1.2, label = rowname))
```


0.50
0.25
0.00
−0.25
−0.50
PC2
−0.4
−0.2
0.0
0.2
0.4
PC1
score_dev
TotalEarnir
metascore
userscore