Proposal

March 6, 2022

1 Project Proposal

1.0.1 1. Introduction

The video game industry began in the 1950s as simple games and simulations. Pixelated screens and limited sound has become a distant memory as video games are offering photorealistic graphics and pushing the frontier of stimulational reality. Video games have become one of the largest sectors in the entertainment market. With the fast growing market, the gaming industry requires marketing data to help predict the sales for their new games. However, in recent years, the emergence of social networks and the developments of mobile games have greatly impacted traditional video games. Careful marketing planning is crucial when a new game is introduced to the market. Therefore, our research question is to predict the sales in the European market for a new video game given North America and other regional sales. To achieve this, we used a dataset generated by scraping of vgchartz.com. It contains a list of video games with sales greater than 100,000 copies from 1980 to 2017.

Dataset: Our dataset can be found at this link. Dataset is scraped from Vgchartz website. List of the fields included in the data are: * `Name`: name of the game * `Platform`: platform of the game release * `Year`: year that the game is released * `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 other countries (in millions) * `Global_sales`: total worldwide sales

Reference can be found here.

[1]:
```
library(tidyverse)
library(dplyr)
library(RColorBrewer)
library(tidyr)
library(tidymodels)
library(repr)
```

```
 Attaching packages                           tidyverse
1.3.0

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

```
Warning message:
"package ‘ggplot2’ was built under R version 4.0.1"
Warning message:
"package ‘tibble’ was built under R version 4.0.2"
Warning message:
"package ‘tidyr’ was built under R version 4.0.2"
Warning message:
"package ‘dplyr’ was built under R version 4.0.2"
  Conflicts
tidyverse_conflicts()
 dplyr::filter() masks stats::filter()
 dplyr::lag()    masks stats::lag()

Warning message:
"package ‘tidymodels’ was built under R version 4.0.2"
  Attaching packages                         tidymodels
0.1.1

 broom     0.7.0       recipes
0.1.13
 dials     0.0.9       rsample   0.0.7
 infer     0.5.4       tune      0.1.1
 modeldata 0.0.2       workflows 0.2.0
 parsnip   0.1.3       yardstick 0.0.7

Warning message:
"package ‘broom’ was built under R version 4.0.2"
Warning message:
"package ‘dials’ was built under R version 4.0.2"
Warning message:
"package ‘infer’ was built under R version 4.0.3"
Warning message:
"package ‘modeldata’ was built under R version 4.0.1"
Warning message:
"package ‘parsnip’ was built under R version 4.0.2"
Warning message:
"package ‘recipes’ was built under R version 4.0.1"
Warning message:
"package ‘tune’ was built under R version 4.0.2"
Warning message:
"package ‘workflows’ was built under R version 4.0.2"
Warning message:
"package ‘yardstick’ was built under R version 4.0.2"
  Conflicts
tidymodels_conflicts()
 scales::discard() masks
purrr::discard()
 dplyr::filter()   masks
```

```
stats::filter()
 recipes::fixed()  masks
stringr::fixed()
 dplyr::lag()      masks stats::lag()
 yardstick::spec() masks readr::spec()
 recipes::step()   masks stats::step()
```

Load data onto Jyputer notebook

[37]:
```
ovg <- read_csv("vgsales.csv")
summary(ovg)
```

```
Parsed with column specification:
cols(
  Rank = col_double(),
  Name = col_character(),
  Platform = col_character(),
  Year = 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()
)
```

```
      Rank            Name              Platform              Year
 Min.   :    1   Length:16598       Length:16598       Length:16598
 1st Qu.: 4151   Class :character   Class :character   Class :character
 Median : 8300   Mode  :character   Mode  :character   Mode  :character
 Mean   : 8301
 3rd Qu.:12450
 Max.   :16600
    Genre              Publisher              NA_Sales           EU_Sales
 Length:16598       Length:16598       Min.   : 0.0000   Min.   : 0.0000
 Class :character   Class :character   1st Qu.: 0.0000   1st Qu.: 0.0000
 Mode  :character   Mode  :character   Median : 0.0800   Median : 0.0200
                                       Mean   : 0.2647   Mean   : 0.1467
                                       3rd Qu.: 0.2400   3rd Qu.: 0.1100
                                       Max.   :41.4900   Max.   :29.0200
    JP_Sales          Other_Sales          Global_Sales
 Min.   : 0.00000   Min.   : 0.00000   Min.   : 0.0100
 1st Qu.: 0.00000   1st Qu.: 0.00000   1st Qu.: 0.0600
 Median : 0.00000   Median : 0.01000   Median : 0.1700
 Mean   : 0.07778   Mean   : 0.04806   Mean   : 0.5374
 3rd Qu.: 0.04000   3rd Qu.: 0.04000   3rd Qu.: 0.4700
```

```
 Max.   :10.22000   Max.   :10.57000   Max.   :82.7400
```

Dataset is in tidy format, therefore, no additional cleaning and wrangling is necessary. However, missing data (NAs) is removed by using `omit.na` function assuming they are missing at random. Moreover, we focused on games published prior to 2017 since the sales data is incomplete in 2017.

[3]:
```
vg <- na.omit(ovg) %>%
      filter(Year<2017)

head(vg)
```

A tibble: 6 × 11

Rank	Name	Platform	Year	Genre	Publisher	NA_Sa
<dbl>	<chr>	<chr>	<chr>	<chr>	<chr>	<dbl>
1	Wii Sports	Wii	2006	Sports	Nintendo	41.49
2	Super Mario Bros.	NES	1985	Platform	Nintendo	29.08
3	Mario Kart Wii	Wii	2008	Racing	Nintendo	15.85
4	Wii Sports Resort	Wii	2009	Sports	Nintendo	15.75
5	Pokemon Red/Pokemon Blue	GB	1996	Role-Playing	Nintendo	11.27
6	Tetris	GB	1989	Puzzle	Nintendo	23.20

Split Training/Testing Tests

[4]:
```
set.seed(9999)

vg_split <- initial_split(vg, prop = 0.75, strata = EU_Sales)
vg_train <- training(vg_split)
vg_test <- testing(vg_split)
```

1.0.2 Exploratory Data Analysis

Visualization

[31]:
```
vg_genre <- vg_train %>%
  group_by(Genre) %>%
  summarise(n=n())%>%
  arrange(desc(n))

vg_genre

#Graph 1
#visualization on the number of games in each genre
vg_genre_plot <- vg_genre%>%
  ggplot(aes(x = reorder(Genre, -n), y = n, fill = Genre))+
  geom_bar(stat = 'identity')+
  labs(x = "Genre of the game",
       y = "Count",
       fill = "Genre",
       title = "Total Number of Games of Genre")+
  scale_color_brewer(palette = "Set3")+
```

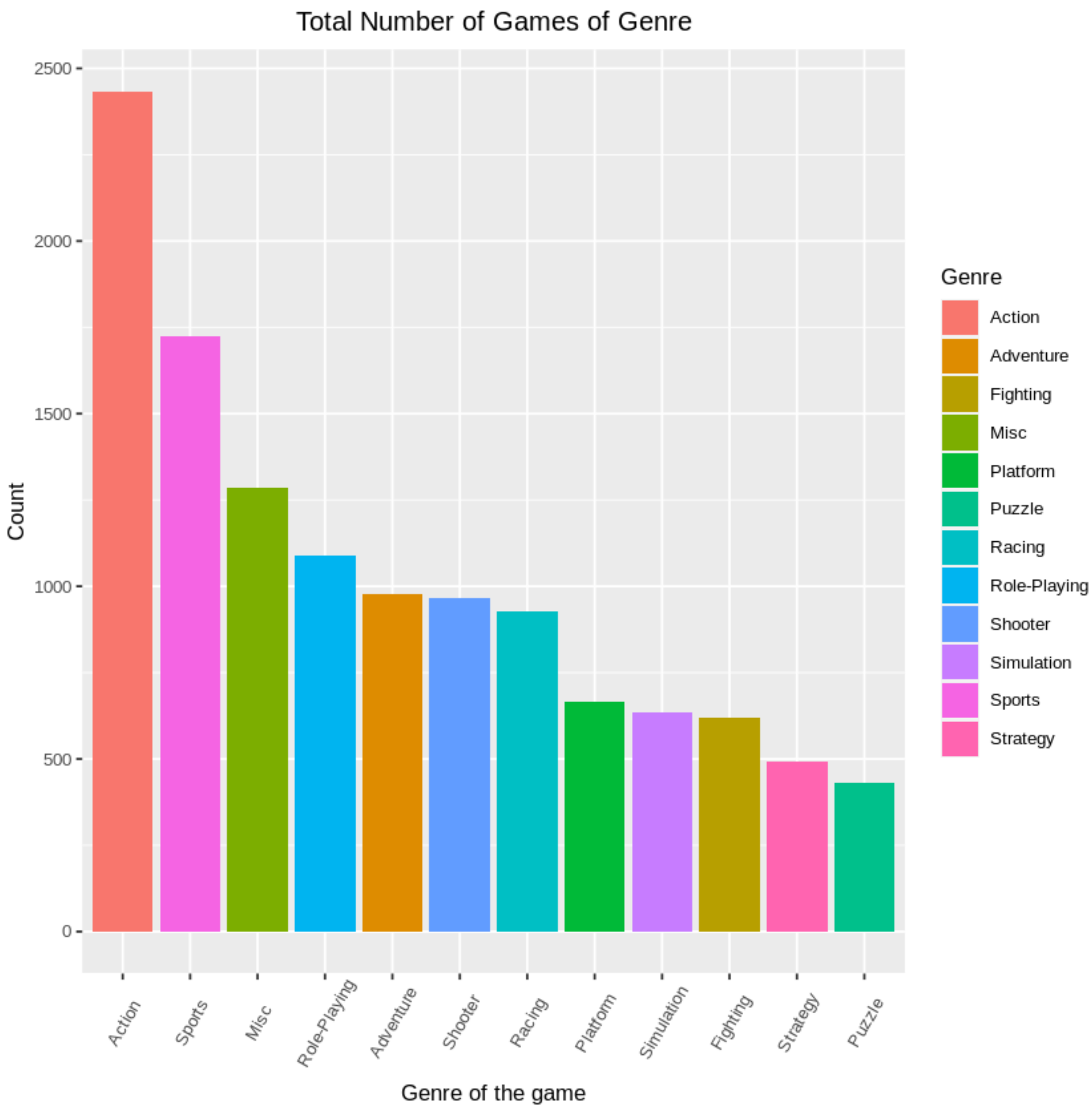
```
  theme(axis.text.x = element_text(angle = 60, vjust = 0.6, hjust=0.5),
        text = element_text(size = 10))+
  theme(plot.title = element_text(hjust = 0.5))

vg_genre_plot
```

```
`summarise()` ungrouping output (override with `.groups` argument)
```

A tibble: 12 × 2

Genre	n
<chr>	<int>
Action	2433
Sports	1723
Misc	1286
Role-Playing	1090
Adventure	979
Shooter	965
Racing	928
Platform	665
Simulation	634
Fighting	619
Strategy	492
Puzzle	430



[33]:

```
#summarize the different game genres' global sales
genre_gbsales <- vg_train %>%
  filter(Genre %in% c("Action","Sports","Role-Playing","Shooter",
                      "Adventure","Racing"))%>%
    group_by(Year,Genre)%>%
    summarize(total_sales = sum(Global_Sales))

head(genre_gbsales)

#plot top 7 genre global sales vs yr
#the customers' preference shifts in all genres over years
#in recent years, sales are decreasing among all 7 genres
options(repr.plot.width = 15, repr.plot.height = 10)
```

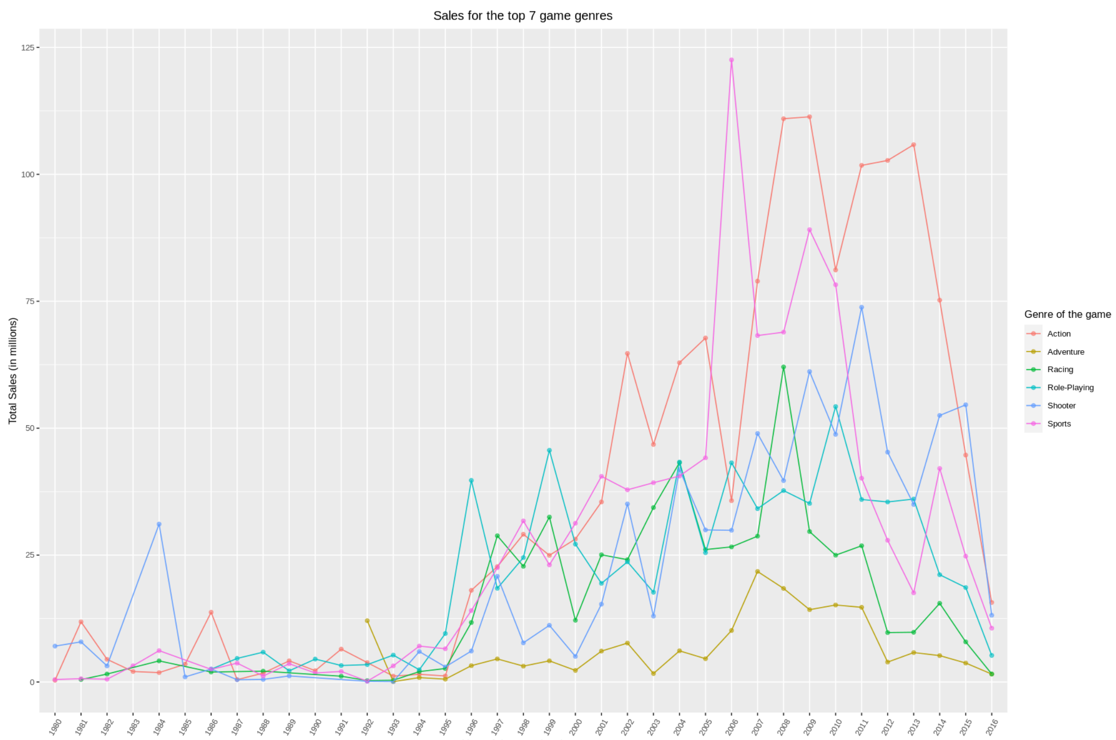
```r
genre_gbsales_plot <- genre_gbsales %>%
  ggplot(aes(x = Year, y = total_sales, colour = Genre, group = Genre))+
  geom_point(alpha = 0.6)+
  geom_line(alpha = 0.9)+
    labs(x = "Year of the game's release",
         y = "Total Sales (in millions)",
         colour = "Genre of the game",
         title = "Sales for the top 7 game genres")+
    theme(axis.text.x = element_text(angle = 60, vjust = 0.5, hjust=0.5),
          text = element_text(size = 10))+
    theme(plot.title = element_text(hjust = 0.5))

genre_gbsales_plot
```

```
`summarise()` regrouping output by 'Year' (override with `.groups` argument)
```

A grouped_df: 6 × 3

Year <chr>	Genre <chr>	total_sales <dbl>
1980	Action	0.34
1980	Shooter	7.07
1980	Sports	0.49
1981	Action	11.86
1981	Racing	0.48
1981	Shooter	7.91



Exploratory Analysis

[34]:
```
vg_genre <- vg_train %>%
  group_by(Genre) %>%
  summarise(n=n())%>%
  arrange(desc(n))
```

`summarise()` ungrouping output (override with `.groups` argument)

[35]:
```
vg_action <- filter(vg_train, Genre == "Action")
vg_action_test <- filter(vg_test, Genre == "Action")

nrow(vg_action)
nrow(vg_action_test)

vg_action_sp <- filter(vg_train, Genre == "Sports" | Genre == "Action")
vg_action_sp_test <- filter(vg_test, Genre == "Sports" | Genre == "Action")

nrow(vg_action_sp)
nrow(vg_action_sp_test)

vg_action_shooter <- filter(vg_train, Genre == "Shooter" | Genre == "Action")
vg_action_shooter_test <- filter(vg_test, Genre == "Shooter" | Genre ==␣
↪"Action")
```

2433

819

4156

1400

Based on the filter above, we proved that there are enough data points.

[38]:
```
vg_cor2<- vg_action_sp %>%
  select(-(Rank:Publisher))

sales_cor_2 <- round(cor(vg_cor2),2)%>%
  as.matrix()

sales_cor_2

vg_cor3<- vg_action_shooter %>%
  select(-(Rank:Publisher))

sales_cor_3 <- round(cor(vg_cor3),2)%>%
```

```
  as.matrix()

sales_cor_3
```

A matrix: 5 × 5 of type dbl

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
NA_Sales	1.00	0.86	0.40	0.73	0.96
EU_Sales	0.86	1.00	0.41	0.70	0.94
JP_Sales	0.40	0.41	1.00	0.30	0.51
Other_Sales	0.73	0.70	0.30	1.00	0.80
Global_Sales	0.96	0.94	0.51	0.80	1.00

A matrix: 5 × 5 of type dbl

	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
NA_Sales	1.00	0.68	0.23	0.57	0.93
EU_Sales	0.68	1.00	0.26	0.64	0.87
JP_Sales	0.23	0.26	1.00	0.19	0.36
Other_Sales	0.57	0.64	0.19	1.00	0.74
Global_Sales	0.93	0.87	0.36	0.74	1.00

We chose to filter sports and action after proving that action and sports game had a high correlation value in terms of total global sales.

1.1 Methods

Three most popular games are action, adventure and fighting. If the gamemaker tries to maximize the revenue, choosing the most liked genre will increase the chance of maximizing the genre.

** Those two genres still look popular in lasst 10 years

1.2 Expected Outcomes

What do you expect to find? Our goal for this project is to predict the sales in Europe for a new video game using sales in NA and other regional sales over years. Based on the NA and other regional sales, we expect to predict the European sales using the regression model.

What impact could such findings have? Using the prediction of our model, it might be useful for video game publishers to predict the sales of new video games in certain regions. This could help gaming companies to focus their advertisements in one specific region, ultimately maximizing their revenue.

What future questions could this lead to? But the salesing value is different in different years. The value of unit money may change over time. But in this project we are mainly focusing on the trending of the game sales

[ ]:

```

```