IGN Video Game Reviews

Data Story

by Ellen A Savoye

April 21 2018

Introduction

Over the last 20 years, a plethora of video games have been released in an ever growing market of gaming consoles. In tandem With the creation of these games, a website called IGN began releasing reviews and ratings of the aforementioned games. These ratings and reviews could potentially be compiled and analyzed to create insights that would be useful to console and video game creators. With analysis, console and video game creators would be able to determine their market/popularity standing in comparison to other consoles and games. Furthermore, they would be able to determine if there is a particular genre that needs more development in order to potentially increase their rating and market standpoint. To do such an analysis, data was sourced from Kaggle (https://www.kaggle.com/egrinstein/20-years-of-games) and IGN (http://ign.com/games/reviews), via a crawl, consisting of 20 years worth of video game data.

Caveats

The data does not contain any financial information relating to volume of games sold or the monetary amount it was sold for. Any potential insights are solely related to the number of games released, their rating, and the console the game was released on. Some games are released across multiple consoles. I have not adjusted the data to constrain games from multiple platforms down to one.

The Data

The data, 20 years' worth of IGN game reviews, consists of 18,625 records. The raw dataset had 10 columns (listed below).

Variable	Description
score_phrase	Phrase given to describe overall score
title	Game title
url	IGN Game URL
platform	Game Console
genre	Video game genre
score	Overall rating for video game
editors_choice	Editor Recommended (Y/N)
release_year	Year of game release
release_month	Month of game release
release_day	Day of game release

Table 2: IGN Data (continued below)

X1	score_phrase	title
0	Amazing	LittleBigPlanet PS Vita
1	Amazing	LittleBigPlanet PS Vita – Marvel Super Hero Edition
2	Great	Splice: Tree of Life
3	Great	NHL 13
4	Great	NHL 13

Table 3: Table continues below

url
/games/littlebigplanet-vita/vita-98907
/games/littlebigplanet-ps-vita-marvel-super-hero-edition/vita-20027059
/games/splice/ipad-141070
/games/nhl-13/xbox-360-128182
/games/nhl-13/ps3-128181

Table 4: Table continues below

platform	score	genre	editors_choice	release_year
PlayStation Vita	9	Platformer	Y	2012
PlayStation Vita	9	Platformer	Y	2012
iPad	8.5	Puzzle	N	2012
Xbox 360	8.5	Sports	N	2012
PlayStation 3	8.5	Sports	N	2012

release_month	release_day
9	12
9	12
9	12
9	11
9	11

Data Wrangling

To take a proper look at the data, I loaded the original dataset as a CSV file and the necessary libraries. Of the variables available for use, score_phrase, platform, score, genre, editors_choice, release_year, release_month, and release_day are the ones I used in my analysis. As such, I analyzed them for missing values, outliers, and whether or not the number of distinct factors in each was usable. Editors_choice, score_phrase, and score did not need cleaning. However, when checking release_year, I noticed an outlier titled "The Walking Dead: The Game – Episode 1: A New Day". This record had a release date of 1/1/1970. Given the dataset is spanning 1996 - 2016, I chose to correct the outlier to the correct release date of 4/24/2012.

```
head(tbl_df(IGN_data), 5)
```

```
## # A tibble: 5 x 11
```

```
##      X1 score_phrase title    url      platform score genre editors_choice
##   <int> <chr>        <chr>    <chr>    <chr>    <dbl> <chr> <chr>
## 1     0 Amazing      LittleB~ /games/~ PlaySta~  9.00 Plat~ Y
## 2     1 Amazing      LittleB~ /games/~ PlaySta~  9.00 Plat~ Y
## 3     2 Great        Splice:~ /games/~ iPad      8.50 Puzz~ N
## 4     3 Great        NHL 13   /games/~ Xbox 360  8.50 Spor~ N
## 5     4 Great        NHL 13   /games/~ PlaySta~  8.50 Spor~ N
## # ... with 3 more variables: release_year <int>, release_month <int>,
## #   release_day <int>
```

```r
# Release year is supposed to be higher than 1995

head(IGN_data %>% distinct(release_year), 5) %>% arrange(release_year)
```

```
## # A tibble: 5 x 1
##   release_year
##          <int>
## 1         1970
## 2         1996
## 3         1997
## 4         2012
## 5         2013
```

```r
IGN_data[IGN_data$release_year == "1970", ]
```

```
## # A tibble: 1 x 11
##      X1 score_phrase title    url      platform score genre editors_choice
##   <int> <chr>        <chr>    <chr>    <chr>    <dbl> <chr> <chr>
## 1   516 Great        The Wal~ /games/~ Xbox 360  8.50 Adve~ N
## # ... with 3 more variables: release_year <int>, release_month <int>,
## #   release_day <int>
```

```r
IGN_data <- IGN_data %>% mutate(release_year = if_else(title ==
    "The Walking Dead: The Game -- Episode 1: A New Day", as.integer(2012),
    release_year)) %>% mutate(release_month = if_else(title ==
    "The Walking Dead: The Game -- Episode 1: A New Day", as.integer(4),
    release_month)) %>% mutate(release_day = if_else(title ==
    "The Walking Dead: The Game -- Episode 1: A New Day", as.integer(24),
    release_day))
```

With the outlier corrected, platform and genre variables remained. The original platform variable consisted of 59 distinct factors. Because platform spanned multiple generations of systems (e.g., PlayStation 1-3) and because not all manufacturers kept system naming consistent, I chose to combine the values into a condensed version based on system name/manufacturer and created a new variable named platform_group. To do so, I loaded a 'platform map' CSV file to merge the new platform_group variable onto the original dataset. After comparing the original platform variable against the new platform_group to ensure no misplaced systems, I moved onto the genre variable.

```r
# 59 variables in original platform column

IGN_data %>% distinct(platform) %>% arrange(platform)

Platform_Map <- read.csv("platform_map.csv")
as.character(Platform_Map$platform)

IGN_data <- IGN_data %>% left_join(Platform_Map, by = c(platform = "platform"))
```

```
## Warning: Column `platform` joining character vector and factor, coercing
## into character vector
```

```
IGN_data %>% group_by(platform, platform_group) %>% summarise(n_distinct(platform_group))
```

```
## # A tibble: 59 x 3
## # Groups:   platform [?]
##    platform            platform_group `n_distinct(platform_group)`
##    <chr>               <fct>                                 <int>
##  1 Android             Android                                   1
##  2 Arcade              Other                                     1
##  3 Atari 2600          Atari                                     1
##  4 Atari 5200          Atari                                     1
##  5 Commodore 64/128    Other                                     1
##  6 Dreamcast           Sega                                      1
##  7 Dreamcast VMU       Sega                                      1
##  8 DVD / HD Video Game Other                                     1
##  9 Game Boy            Game Boy                                  1
## 10 Game Boy Advance    Game Boy                                  1
## # ... with 49 more rows
```

Similar to the platform variable, the genre variable has a multitude of factors which makes intelligent analysis a bit difficult. There are 113 unique genres within the field. I chose my grouping based on an overall description (e.g., Sports, Cards, Action, etc.) given the numerous distinct factors. Before cleaning up the column, I checked for any blank cells. Out of 18,625 observations, 36 do not have a genre which is .19%. Due to the blank records being less than 1% of the overall genre column, I chose not to populate them but instead mapped them to 'Other.' To map genre, I loaded a 'genre map' CSV file to merge the new genre_group variable onto the original dataset. In doing so, I brought the number of unique genres from 113 to 21.

```
# Check for blanks in genre column

IGN_data %>% distinct(genre) %>% arrange(genre)
```

```
group_by(IGN_data[IGN_data$genre == "", ])
```

```
## # A tibble: 36 x 12
##       X1 score_phrase title url   platform score genre editors_choice
##    <int> <chr>        <chr> <chr> <chr>    <dbl> <chr> <chr>
##  1    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  2    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  3    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  4    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  5    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  6    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  7    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  8    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
##  9    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
## 10    NA <NA>         <NA>  <NA>  <NA>        NA <NA>  <NA>
## # ... with 26 more rows, and 4 more variables: release_year <int>,
## #   release_month <int>, release_day <int>, platform_group <fct>
```

```
# 113 unique factors in genre column

IGN_data %>% distinct(genre) %>% arrange(genre)

Genre_Map <- read.csv("genre_map.csv")
as.character(Genre_Map$genre)
```

```r
IGN_data <- IGN_data %>% left_join(Genre_Map, by = c(genre = "genre"))
```

```
## Warning: Column `genre` joining character vector and factor, coercing into
## character vector
```

```r
unique(IGN_data$genre_group)
```

```
##  [1] Platformer   Puzzle       Sports       Strategy     Fighting
##  [6] RPG          <NA>         Action       Adventure    Shooter
## [11] Music        Other        Racing       Simulation   Education
## [16] Wrestling    Productivity Cards        Compilation  Flight
## [21] Pinball      Hunting
## 21 Levels: Action Adventure Cards Compilation Education ... Wrestling
```

After cleaning up the variables that I will be using in my analysis, I wrote the wrangled data to a new file called "ign_clean.csv" for further use in creating insights.

```r
write.csv(IGN_data, "ign_clean.csv")
IGN_data_cleaned <- read.csv("ign_clean.csv")
```

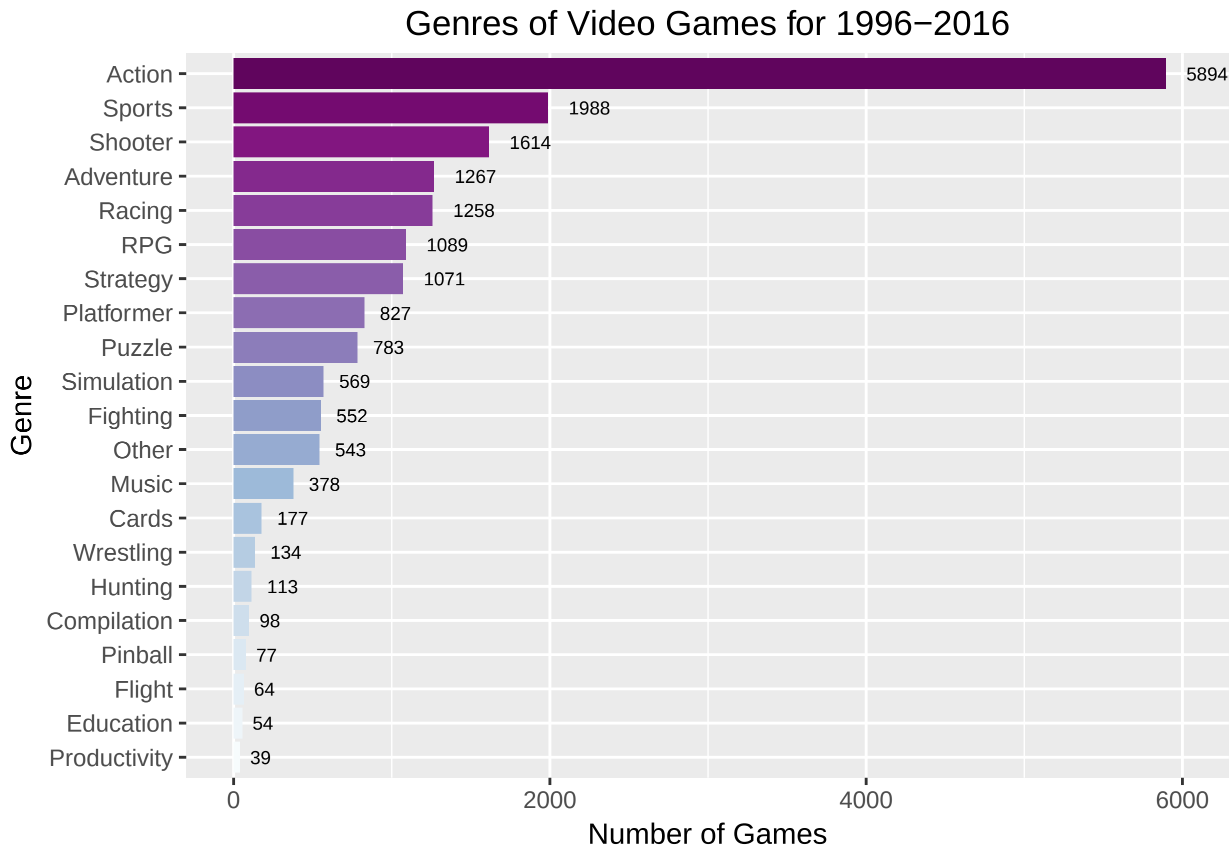
Exploratory Data Analysis

For the first look, I'd like to determine what are the top genres and what the top platforms are.

Genre

Top Genres

```r
top_genres <- IGN_data_cleaned %>% group_by(genre_group) %>% summarize(genres_count = n()) %>%
    arrange(desc(genres_count))

top_genres$genre_group <- factor(top_genres$genre_group, levels = top_genres$genre_group[order(top_genre

colourCount = length(unique(top_genres$genre_group))
fill_purple <- colorRampPalette(brewer.pal(9, "BuPu"))

top_genres_plot <- top_genres %>% filter(genre_group != "NA") %>% ggplot(aes(x = genre_group,
    y = genres_count, fill = genre_group)) + geom_bar(stat = "identity") + coord_flip() +
    geom_text(aes(label = genres_count), size = 2.5, color = "black", hjust = -0.5) +
    labs(x = "Genre", y = "Number of Games", title = "Genres of Video Games for 1996-2016") +
    theme(legend.position = "none", plot.title = element_text(hjust = 0.5)) +
    ylim(0, max(top_genres$genres_count + 100)) + scale_fill_manual(values = fill_purple(colourCount))
top_genres_plot
```

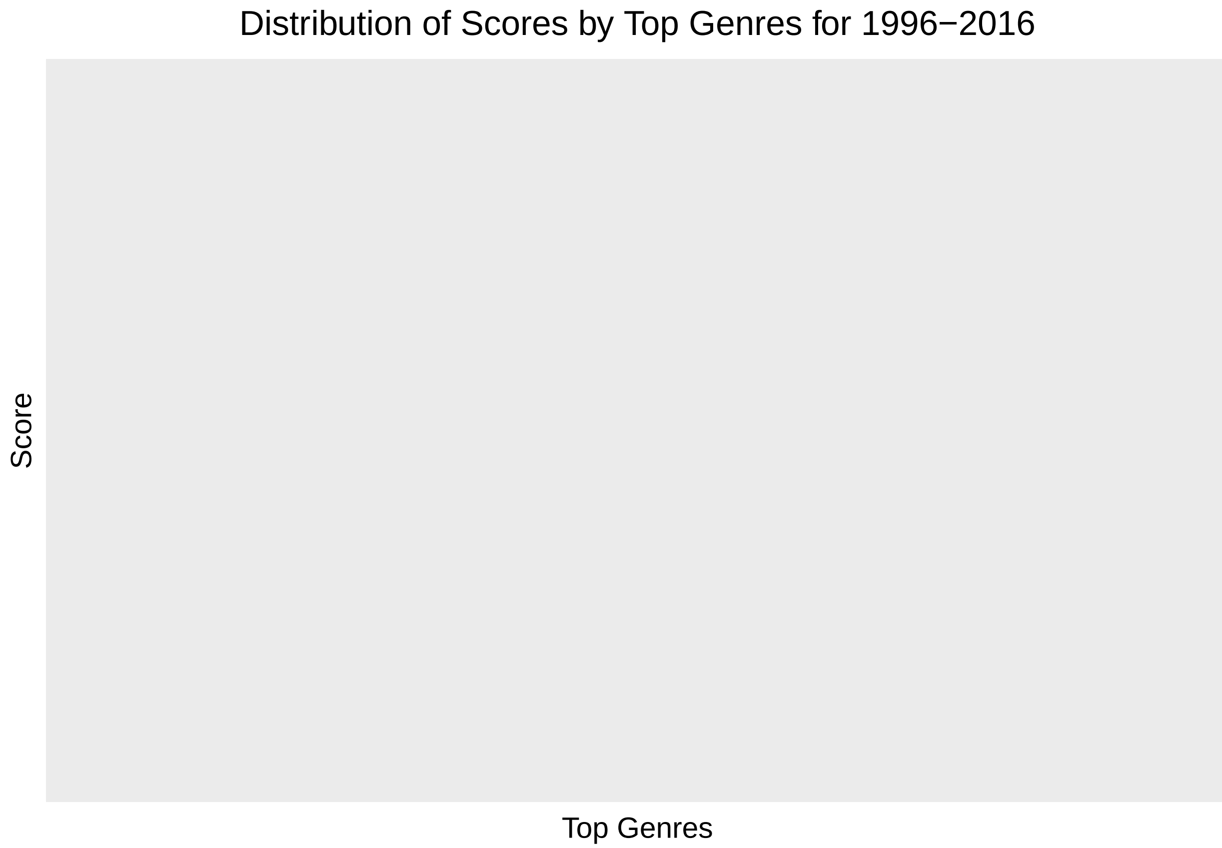


The top genre is Action followed by Sports, Shooter, Adventure, and Racing. The number of Action games is more than double the next genre, Sports. As I mentioned in the caveats section, this may be due to some games applying across platforms and are therefor being counted multiple times.

Top Genres by Score

Due to a little more than half of the genres having a count of 500+, I'm using 500 as my minimum number of games to filter my data. **Needs to be fixed**

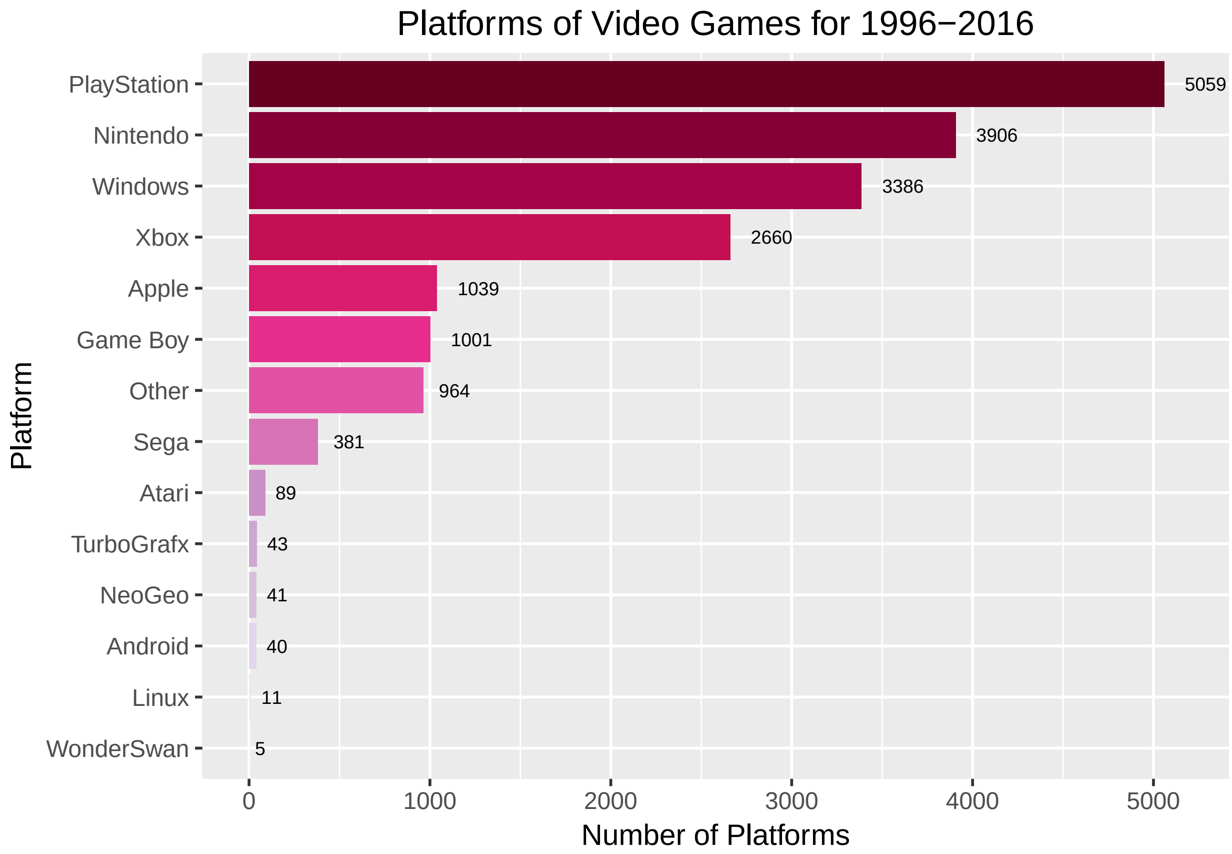
```
top_genre_scores <- IGN_data_cleaned %>% group_by(genre_group) %>% summarize(sum_genres_count = n()) %>%
    arrange(desc(sum_genres_count)) %>% filter(sum_genres_count > 499)
top_genre_scores <- top_genre_scores[, 1]

colourCount = length(unique(top_genres$genre_group))
fill_purple <- colorRampPalette(brewer.pal(9, "BuPu"))

top_genres_score_plot <- IGN_data_cleaned %>% filter(top_genre_scores %in% genre_group) %>%
    ggplot(aes(x = genre_group, y = score, fill = genre_group, color = genre_group)) +
    geom_boxplot(alpha = 0.5) + labs(x = "Top Genres", y = "Score", title = "Distribution of Scores by
    theme(legend.position = "none", plot.title = element_text(hjust = 0.5))
top_genres_score_plot
```



Platforms

Top Platforms

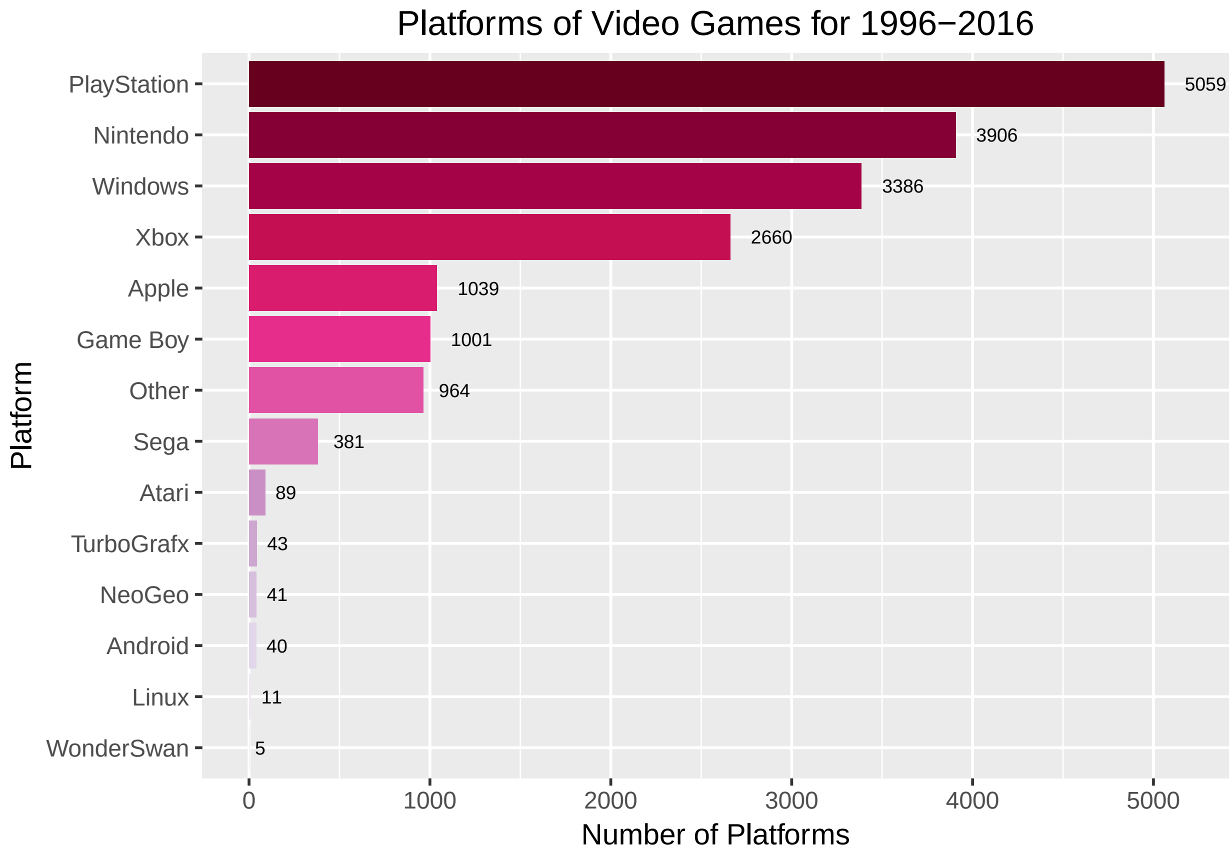
```
top_platform <- IGN_data_cleaned %>% group_by(platform_group) %>% summarize(platform_count = n()) %>%
    arrange(desc(platform_count))

top_platform$platform_group <- factor(top_platform$platform_group, levels = top_platform$platform_group

colourCount = length(unique(top_platform$platform_group))
fill_purple <- colorRampPalette(brewer.pal(9, "PuRd"))

top_platform_plot <- top_platform %>% filter(platform_group != "NA") %>% ggplot(aes(x = platform_group,
    y = platform_count, fill = platform_group)) + geom_bar(stat = "identity") +
    coord_flip() + geom_text(aes(label = platform_count), size = 2.5, color = "black",
    hjust = -0.5) + labs(x = "Platform", y = "Number of Platforms", title = "Platforms of Video Games fo
    theme(legend.position = "none", plot.title = element_text(hjust = 0.5)) +
    ylim(0, max(top_platform$platform_count + 100)) + scale_fill_manual(values = fill_purple(colourCoun
top_platform_plot
```



While some of these platforms have had multiple versions/evolutions over the last 20 years, Nintendo for example, those versions have been grouped together for a simpler analysis on the major gaming platforms. As such, Playstation is the top system followed by Nintendo, Windows, and Xbox. Taking this information, we can take the top 6 platforms and see how they performed over the last 10 years by their aggregated scores.

Top Platforms by Score (to be updated)

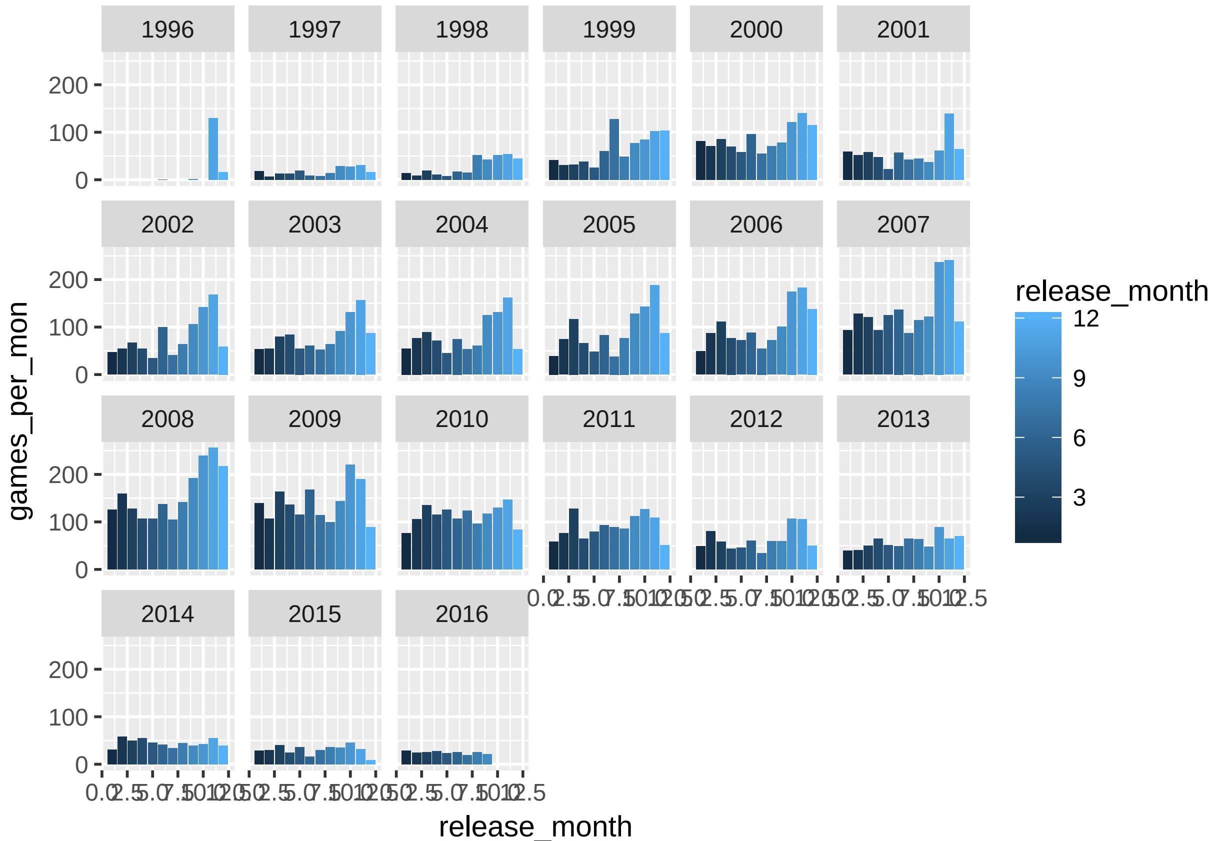
```
top_platform <- IGN_data_cleaned %>% group_by(platform_group) %>% summarize(platform_count = n()) %>%
    arrange(desc(platform_count))

top_platform$platform_group <- factor(top_platform$platform_group, levels = top_platform$platform_group

colourCount = length(unique(top_platform$platform_group))
fill_purple <- colorRampPalette(brewer.pal(9, "PuRd"))

top_platform_plot <- top_platform %>% filter(platform_group != "NA") %>% ggplot(aes(x = platform_group,
    y = platform_count, fill = platform_group)) + geom_bar(stat = "identity") +
    coord_flip() + geom_text(aes(label = platform_count), size = 2.5, color = "black",
    hjust = -0.5) + labs(x = "Platform", y = "Number of Platforms", title = "Platforms of Video Games fo
    theme(legend.position = "none", plot.title = element_text(hjust = 0.5)) +
    ylim(0, max(top_platform$platform_count + 100)) + scale_fill_manual(values = fill_purple(colourCoun
top_platform_plot
```



Important Dates

We want to see if a particular day, month, and/or year stands out as significant. To do so, I first looked at month and year together.

```
Mon_Yr_Ct <- IGN_data_cleaned %>% group_by(release_month, release_year) %>%
    summarize(games_per_mon = n()) %>% arrange(desc(games_per_mon))
Mon_Yr_Ct %>% ggplot(aes(x = release_month, y = games_per_mon, fill = release_month)) +
    geom_bar(stat = "identity") + facet_wrap(~release_year, ncol = 6)
```



1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
200
100
0
games_per_mon
release_month
release_month
12
9
6
3