IGN Video Game Reviews

Data Story

by Ellen A Savoye

May 8, 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 the 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

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.

```
## # 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   Plat~ Y
## 2     1 Amazing      LittleB~ /games/~ PlaySta~   9   Plat~ Y
## 3     2 Great        Splice:~ /games/~ iPad       8.5 Puzz~ N
## 4     3 Great        NHL 13   /games/~ Xbox 360   8.5 Spor~ N
## 5     4 Great        NHL 13   /games/~ PlaySta~   8.5 Spor~ N
## # ... with 3 more variables: release_year <int>, release_month <int>,
## #   release_day <int>
```

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

```
## # 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   8.5 Adve~ N
## # ... with 3 more variables: release_year <int>, release_month <int>,
## #   release_day <int>
```

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.

```
## # A tibble: 59 x 3
## # Groups:   platform [?]
##    platform            platform_group `n_distinct(platform_group)`
##    <chr>               <chr>                                <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.

```
## # A tibble: 36 x 12
##       X1 score_phrase title    url     platform score genre editors_choice
##    <int> <chr>        <chr>    <chr>   <chr>    <dbl> <chr> <chr>
##  1    12 Good         Wild Bl~ /games~ iPhone     7   <NA>  N
##  2   113 Good         Retro/G~ /games~ PlaySta~   7   <NA>  N
##  3   160 Good         10000000 /games~ iPhone     7.5 <NA>  N
##  4   176 Okay         Colour ~ /games~ PC         6.2 <NA>  N
##  5  9375 Great        Duke Nu~ /games~ Wireless   8   <NA>  Y
##  6  9488 Okay         Rengoku  /games~ Wireless   6.5 <NA>  N
##  7  9767 Good         Super S~ /games~ Wireless   7.5 <NA>  N
##  8  9774 Amazing      Critter~ /games~ Wireless   9   <NA>  Y
##  9 10494 Awful        Clue / ~ /games~ Nintend~   3.5 <NA>  N
## 10 11367 Painful      Jeep Th~ /games~ PlaySta~   2   <NA>  N
## # ... with 26 more rows, and 4 more variables: release_year <int>,
## #   release_month <int>, release_day <int>, platform_group <chr>
```

```
##  [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
```

```
## [1] 22
```

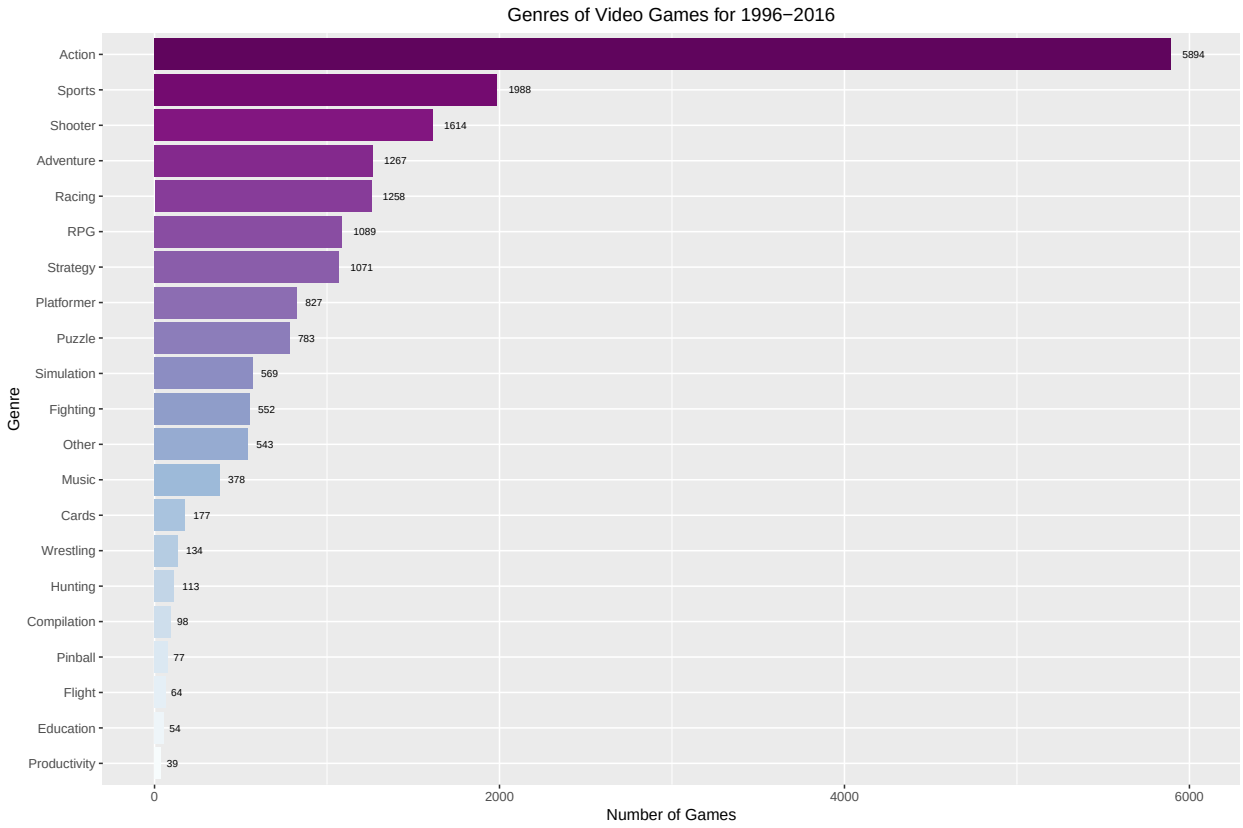
After cleaning up the variables that I will be using in my analysis, I wrote the wrangled data to a new file called "ign_clean.csv" for further use in creating insights.

Exploratory Data Analysis

Genre

Using the condensed genre field, I'm looking to see what the top genre fields are in terms of video game releases and whether or not the scores correspond to the top genres.

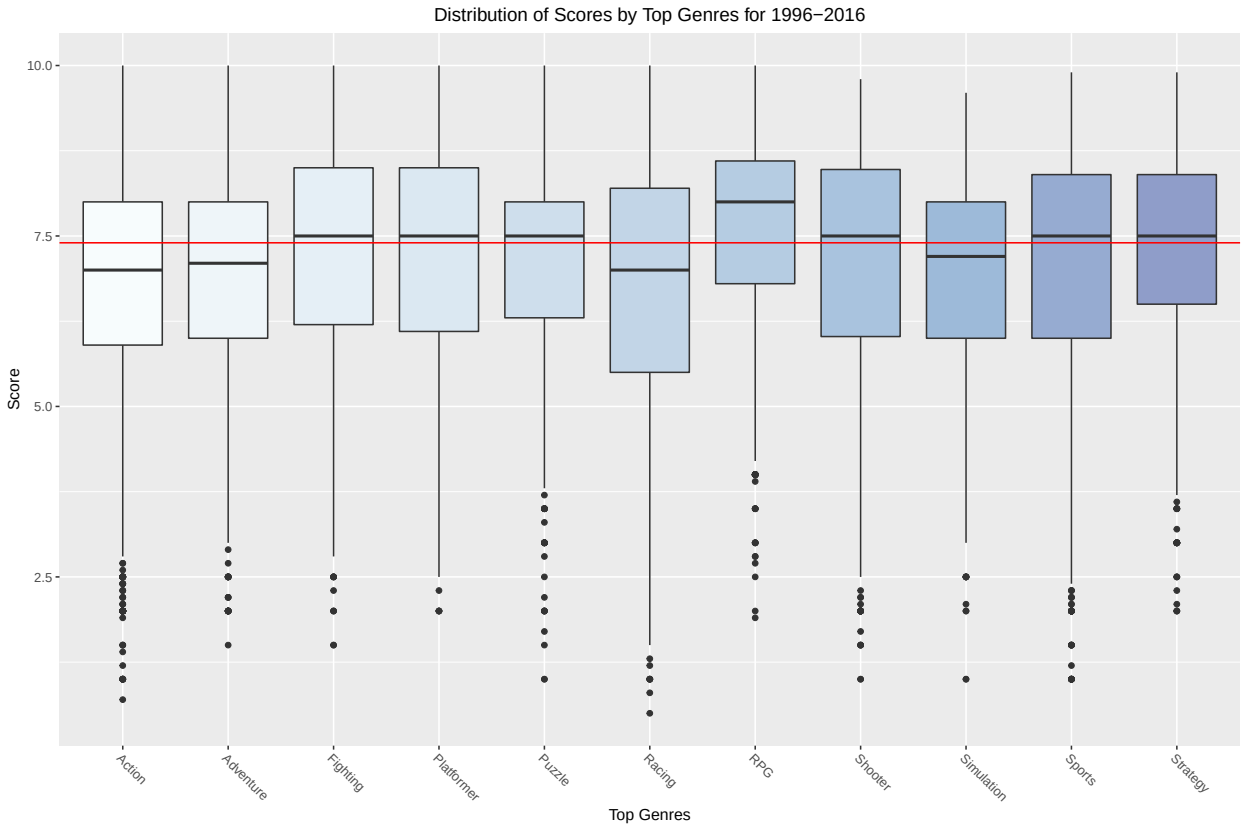
Top Genres



Cumulatively, 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 therefore being counted multiple times. Now that we have the top genres, we can see if there are any insights to glean.

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. However, given that 'other' is a catch-all bucket, the miminum will instead be 550 to exclude 'other'.



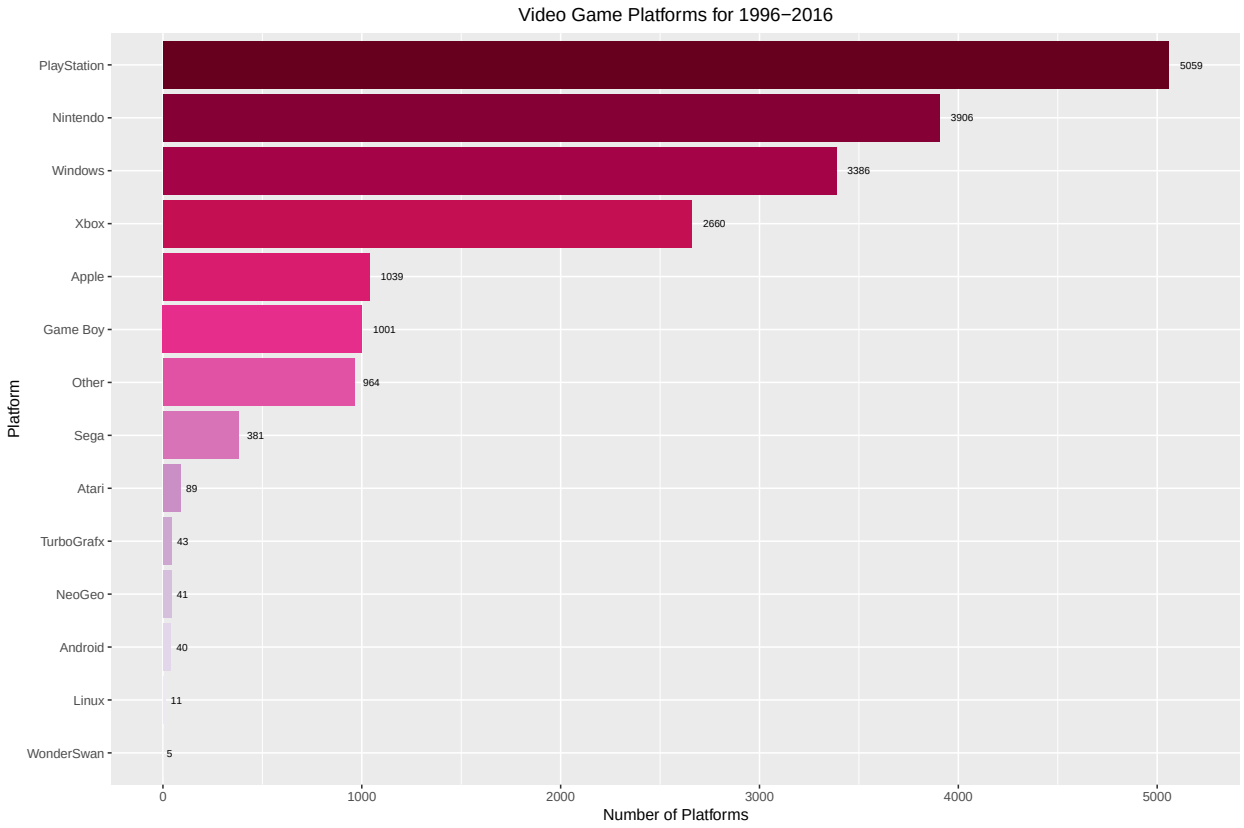

```
##       score_phrase                                    title platform score
## 891       Disaster                    Extreme PaintBrawl       PC   0.7
## 5243      Disaster Looney Tunes: Back in Action: Zany Race Wireless   0.5
## 12514     Disaster                   Action Girlz Racing      Wii   0.8
##        genre editors_choice release_year release_month release_day
## 891   Action              N         1998            10          29
## 5243  Racing              N         2003            10          28
## 12514 Racing              N         2009             2          11
##       platform_group genre_group
## 891          Windows      Action
## 5243           Other      Racing
## 12514       Nintendo      Racing
```

When looking at the distribution of scores for the top 11 genres, we can see that RPG has higher scores than the other genres eve though it sits in sixth place for volumne of games released. Given the darth of games released in the Action genre, I expected it to have the higher scores across the board from a slightly volume influenced stand-point. Action, Adventure, and Racing have lower scores with Racing having the largest IQR. Both Action and Racing have games with a score less than 1. Two of the games are in Racing under while the third is in Action.

Platforms

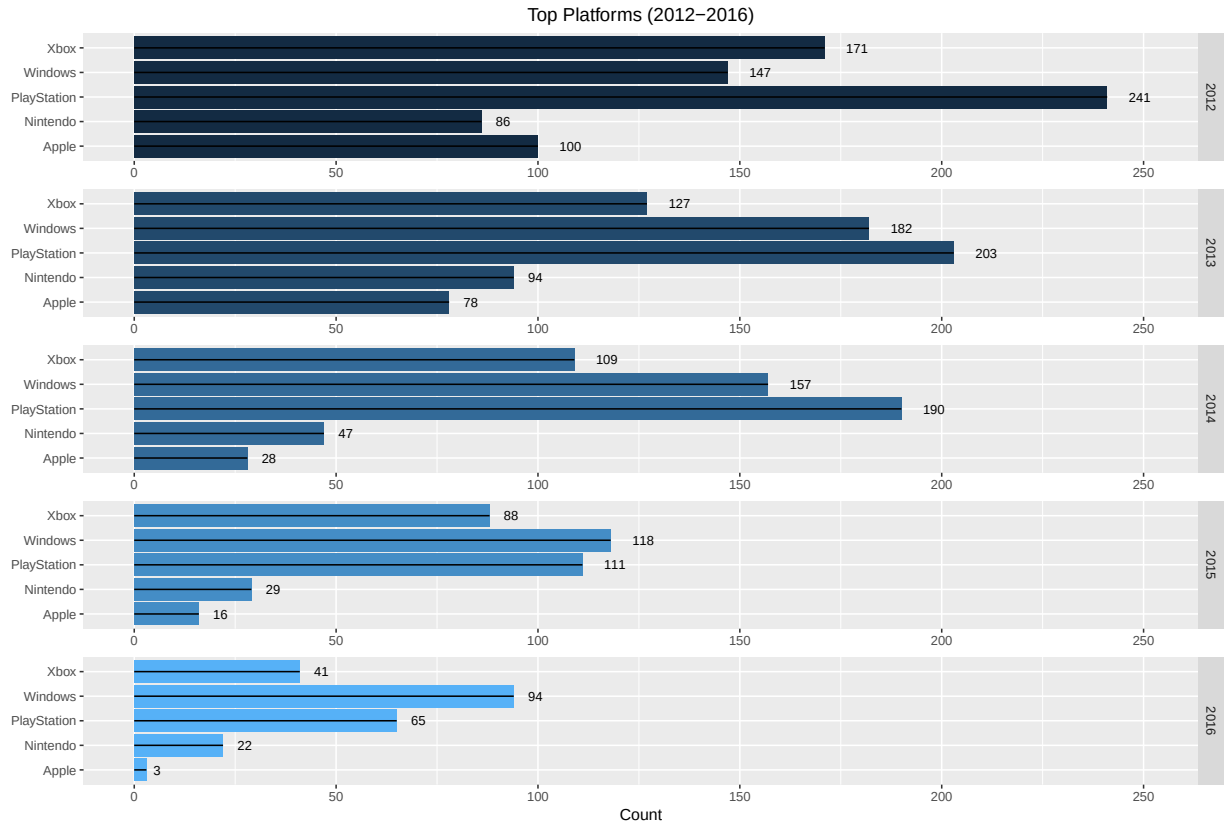
I want to see which platform reigned supreme over the last twenty years and if that holds true when looking at the past five years.

Top Platforms



While some of these platforms have had multiple versions/evolutions over the last 20 years, Nintendo for example, they have been grouped together for a simpler analysis on the major gaming platforms. Even though Playstation was released after Nintendo, Playstation is still the top system followed by Nintendo, Windows, and Xbox. Given the previous graph consisting of all 20 years worth of data, I put together a graph showing the last 5 years of data to see if the top systems held consistent.

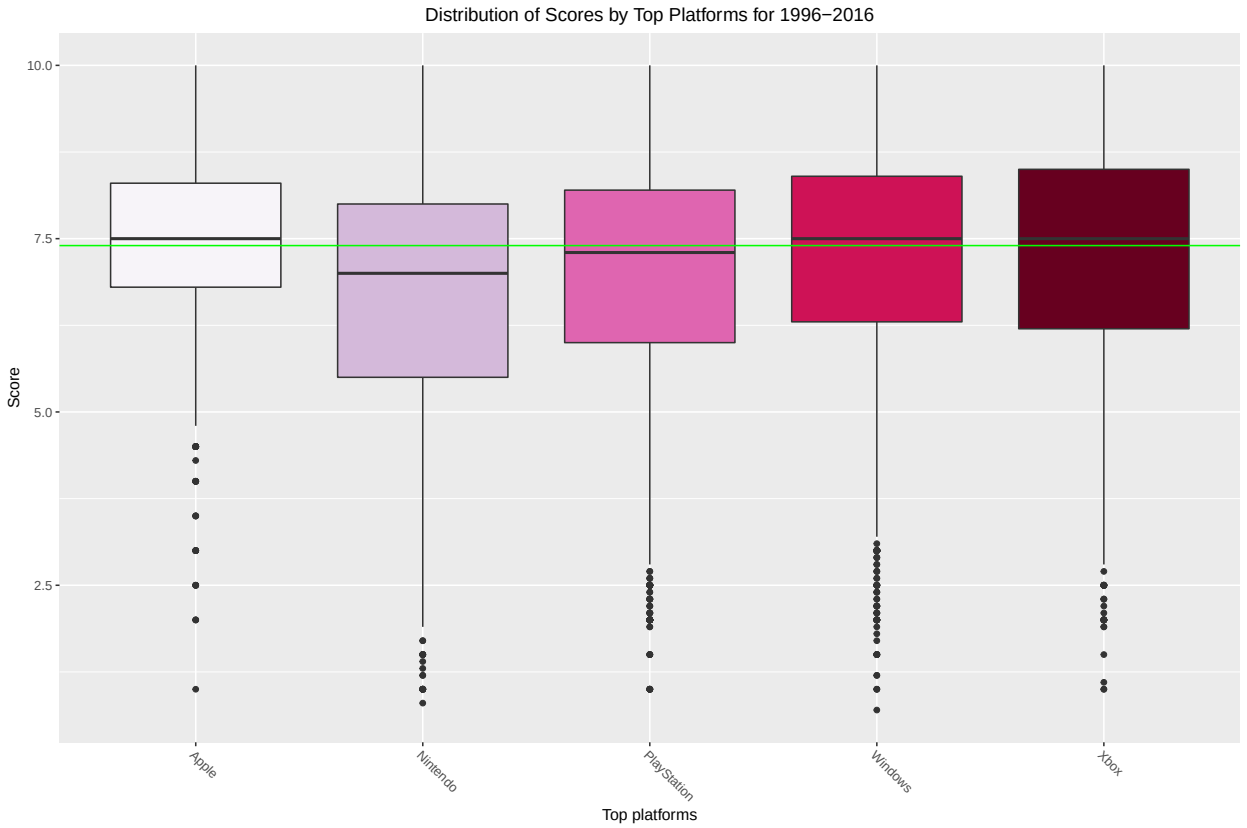
Top Platforms for Past 5 Years



For 2012-2014, Playstation held down the top spot which is consistent with all years combined. However, 2015-2016, Windows is the top platform. So even though Playstation holds the top spot in number of games released overall, it doesn't necessarily hold true on a year-by-year basis.

Top Platforms by Score

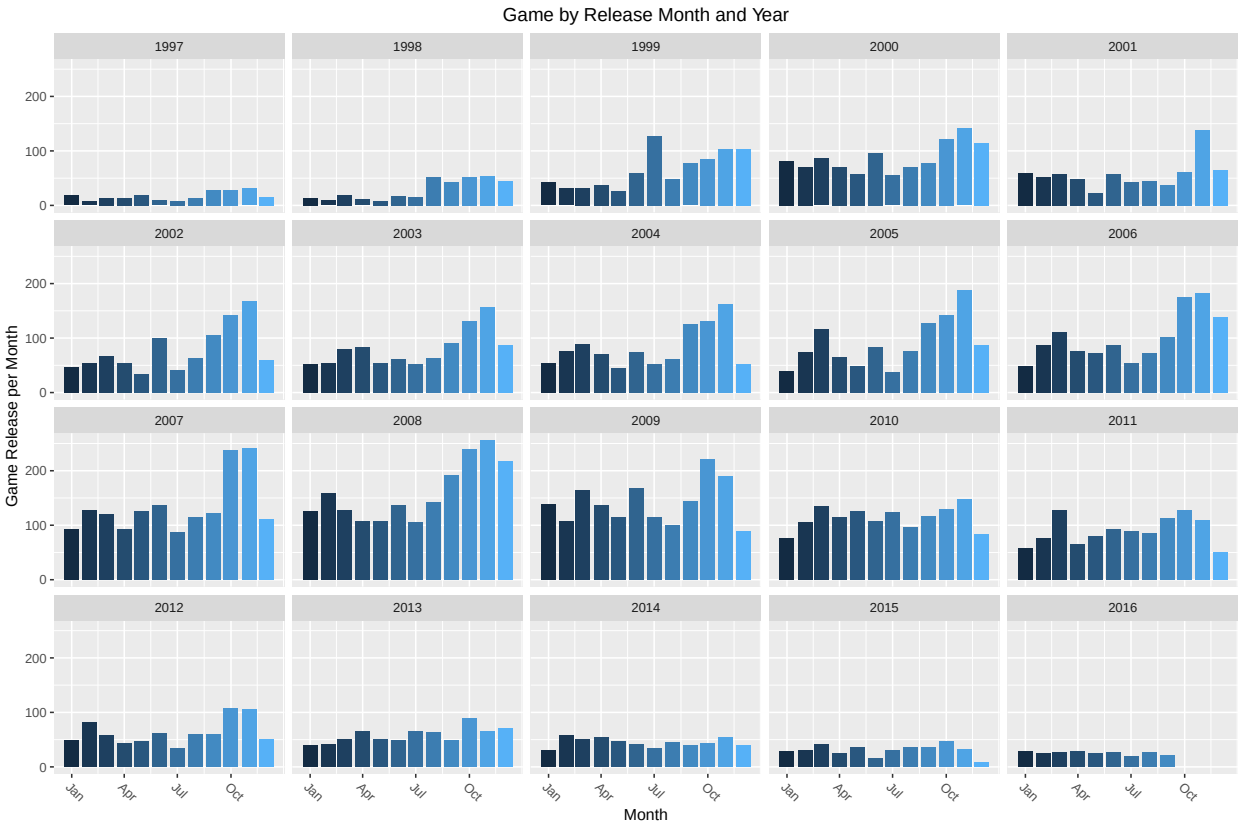
Taking the top 5 platforms found in our "Top Platforms" exhibit above, I want to see how the scores correspond to platform volume.



Given the number of game releases that Playstation has had, there was some expectation for a better score performance. However, this doesn't appear to be the case. Both Xbox and Windows have higher scores for the 25th, 50th, and 75th IQR values. Windows has the top number of released games for 2015 and 2016 so the higher scores are not completely a surprise.

Important Dates

We want to see if a particular month and year stands out as significant. To do so, I looked at month and year together in a grid.



From 1997 - 2008, the number of games released largely increases with 2008 having the most reviewed and released in video game history. Not so much in the more recent years with more consistent releases over the year, in the prior years, there is the larger amount of released video games in Octover and November over other months. The decreased number of games released could be due to video games becoming significantly more intricate and graphics intensive. With the capailities of T.V. and graphics cards now, immerisive video game graphics seem like a must-have for a game to be released.

Predictive Models

Subsetting the Data

Prior to any ventures into modeling, a data set was created to remove any unnecessary features and any blank records. 36 records with a 'blank' genre were removed. Given that some of the kept features are categorical, a conversion to binary/dummy variables was needed using model matrix. I chose to not combine release day, month, and year into a combined date field as I don't believe it would have given any valuable insight as unique a combined value.

```
##   score_phrase          title       platform          score          genre
##              0              0              0              0             36
## editors_choice   release_year  release_month    release_day platform_group
##              0              0              0              0              0
##    genre_group
##             36
```

```
## 'data.frame':    18589 obs. of  7 variables:
##  $ score         : num  9 9 8.5 8.5 8.5 7 3 9 3 7 ...
```

```
##  $ editors_choice: Factor w/ 2 levels "N","Y": 2 2 1 1 1 1 1 2 1 1 ...
##  $ release_year  : Factor w/ 21 levels "1996","1997",..: 17 17 17 17 17 17 17 17 17 17 ...
##  $ release_month : Factor w/ 12 levels "1","2","3","4",..: 9 9 9 9 9 9 9 9 9 9 ...
##  $ release_day   : Factor w/ 31 levels "1","2","3","4",..: 12 12 12 11 11 11 11 11 11 11 ...
##  $ platform_group: Factor w/ 14 levels "Android","Apple",..: 9 9 2 14 9 2 14 12 9 12 ...
##  $ genre_group   : Factor w/ 21 levels "Action","Adventure",..: 12 12 14 19 19 20 6 16 6 20 ...
```

```
## [1] 18589    76
```

```
##  [1]  1  4  6  7 11 12 14 17 18 19 20 21 22 23 24 25 26 27 28 32 35 36 37
## [24] 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60
## [47] 61 62 63 64 65
```

```
## [1] 18589    25
```

```
## integer(0)
```

```
## reg_data
##
##  26  Variables      18589  Observations
## ---------------------------------------------------------------------------
## editors_choiceY
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2     0.46     3515   0.1891   0.3067
##
## ---------------------------------------------------------------------------
## platform_groupApple
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.156     1025  0.05514   0.1042
##
## ---------------------------------------------------------------------------
## platform_groupGame Boy
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.153     1001  0.05385   0.1019
##
## ---------------------------------------------------------------------------
## platform_groupNintendo
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.497     3896   0.2096   0.3313
##
## ---------------------------------------------------------------------------
## platform_groupOther
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.147      959  0.05159  0.09786
##
## ---------------------------------------------------------------------------
## platform_groupPlayStation
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.594     5055   0.2719    0.396
##
## ---------------------------------------------------------------------------
## platform_groupWindows
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.447     3383    0.182   0.2978
##
## ---------------------------------------------------------------------------
```

```
## platform_groupXbox
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.368     2660   0.1431   0.2453
##
## ------------------------------------------------------------------------
## genre_groupAdventure
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.191     1267  0.06816    0.127
##
## ------------------------------------------------------------------------
## genre_groupRacing
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.189     1258  0.06767   0.1262
##
## ------------------------------------------------------------------------
## genre_groupRPG
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.165     1089  0.05858   0.1103
##
## ------------------------------------------------------------------------
## genre_groupShooter
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.238     1614  0.08683   0.1586
##
## ------------------------------------------------------------------------
## genre_groupSports
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.287     1988   0.1069    0.191
##
## ------------------------------------------------------------------------
## genre_groupStrategy
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.163     1071  0.05761   0.1086
##
## ------------------------------------------------------------------------
## release_month2
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.199     1327  0.07139   0.1326
##
## ------------------------------------------------------------------------
## release_month3
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.231     1565  0.08419   0.1542
##
## ------------------------------------------------------------------------
## release_month4
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2     0.19     1263  0.06794   0.1267
##
## ------------------------------------------------------------------------
## release_month5
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.173     1141  0.06138   0.1152
##
```

```
## ---------------------------------------------------------------------------
## release_month6
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2     0.22     1481  0.07967   0.1467
##
## ---------------------------------------------------------------------------
## release_month7
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.179     1188  0.06391   0.1197
##
## ---------------------------------------------------------------------------
## release_month8
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2      0.2     1334  0.07176   0.1332
##
## ---------------------------------------------------------------------------
## release_month9
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.249     1701  0.09151   0.1663
##
## ---------------------------------------------------------------------------
## release_month10
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.326     2306   0.1241   0.2173
##
## ---------------------------------------------------------------------------
## release_month11
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.367     2655   0.1428   0.2449
##
## ---------------------------------------------------------------------------
## release_month12
##        n  missing distinct     Info      Sum     Mean      Gmd
##    18589        0        2    0.223     1504  0.08091   0.1487
##
## ---------------------------------------------------------------------------
## score
##        n  missing distinct     Info     Mean      Gmd      .05      .10
##    18589        0       93    0.998    6.951    1.889      3.5      4.5
##      .25      .50      .75      .90      .95
##      6.0      7.3      8.2      8.9      9.1
##
## lowest :  0.5  0.7  0.8  1.0  1.1, highest:  9.6  9.7  9.8  9.9 10.0
## ---------------------------------------------------------------------------
```

Linear Regression

```
##
## Call:
## lm(formula = score ~ ., data = reg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9827 -0.6511  0.2158  1.0260  3.4121
```

```
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 6.61710    0.07086  93.378  < 2e-16 ***
## editors_choiceY             2.29780    0.02693  85.334  < 2e-16 ***
## platform_groupApple         0.04548    0.07301   0.623 0.533318
## platform_groupGame.Boy     -0.63079    0.07332  -8.603  < 2e-16 ***
## platform_groupNintendo     -0.58438    0.06221  -9.393  < 2e-16 ***
## platform_groupOther        -0.33464    0.07393  -4.526 6.05e-06 ***
## platform_groupPlayStation -0.33149    0.06135  -5.403 6.63e-08 ***
## platform_groupWindows      -0.20208    0.06353  -3.181 0.001470 **
## platform_groupXbox         -0.20217    0.06425  -3.147 0.001654 **
## genre_groupAdventure        0.05884    0.04268   1.379 0.168006
## genre_groupRacing          -0.15182    0.04251  -3.572 0.000356 ***
## genre_groupRPG              0.42068    0.04576   9.194  < 2e-16 ***
## genre_groupShooter          0.07990    0.03853   2.074 0.038113 *
## genre_groupSports           0.04943    0.03503   1.411 0.158236
## genre_groupStrategy         0.21563    0.04755   4.535 5.80e-06 ***
## release_month2              0.16384    0.05764   2.843 0.004478 **
## release_month3              0.16889    0.05565   3.035 0.002412 **
## release_month4              0.15263    0.05829   2.619 0.008837 **
## release_month5              0.11642    0.05978   1.947 0.051499 .
## release_month6              0.22268    0.05622   3.961 7.51e-05 ***
## release_month7              0.12258    0.05923   2.069 0.038518 *
## release_month8              0.26927    0.05757   4.678 2.92e-06 ***
## release_month9              0.35055    0.05473   6.406 1.53e-10 ***
## release_month10             0.26770    0.05179   5.169 2.38e-07 ***
## release_month11             0.19154    0.05070   3.777 0.000159 ***
## release_month12             0.02856    0.05603   0.510 0.610195
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.42 on 18563 degrees of freedom
## Multiple R-squared:  0.3132, Adjusted R-squared:  0.3123
## F-statistic: 338.6 on 25 and 18563 DF,  p-value: < 2.2e-16
```

```
##
## Call:
## lm(formula = score ~ ., data = reg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -5.9962 -0.6504  0.2144  1.0254  3.3865
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 6.70192    0.04108 163.145  < 2e-16 ***
## editors_choiceY             2.29975    0.02690  85.502  < 2e-16 ***
## platform_groupGame.Boy     -0.66284    0.05727 -11.574  < 2e-16 ***
## platform_groupNintendo     -0.61250    0.04208 -14.555  < 2e-16 ***
## platform_groupOther        -0.36223    0.05792  -6.254 4.09e-10 ***
## platform_groupPlayStation -0.35633    0.04073  -8.748  < 2e-16 ***
## platform_groupWindows      -0.22645    0.04344  -5.213 1.88e-07 ***
## platform_groupXbox         -0.22708    0.04494  -5.053 4.39e-07 ***
```

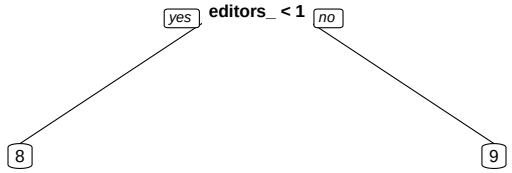
```
## genre_groupRacing         -0.16459    0.04191  -3.927 8.63e-05 ***
## genre_groupRPG             0.40826    0.04517   9.038  < 2e-16 ***
## genre_groupShooter         0.06955    0.03783   1.838  0.06603 .
## genre_groupStrategy        0.20393    0.04690   4.348 1.38e-05 ***
## release_month2             0.11781    0.04539   2.595  0.00945 **
## release_month3             0.12292    0.04283   2.870  0.00411 **
## release_month4             0.10751    0.04620   2.327  0.01998 *
## release_month6             0.17529    0.04358   4.022 5.79e-05 ***
## release_month7             0.07613    0.04734   1.608  0.10781
## release_month8             0.22506    0.04528   4.971 6.74e-07 ***
## release_month9             0.30486    0.04161   7.327 2.45e-13 ***
## release_month10            0.22076    0.03769   5.857 4.80e-09 ***
## release_month11            0.14380    0.03615   3.977 7.00e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.42 on 18568 degrees of freedom
## Multiple R-squared:  0.3129, Adjusted R-squared:  0.3121
## F-statistic: 422.8 on 20 and 18568 DF,  p-value: < 2.2e-16
```

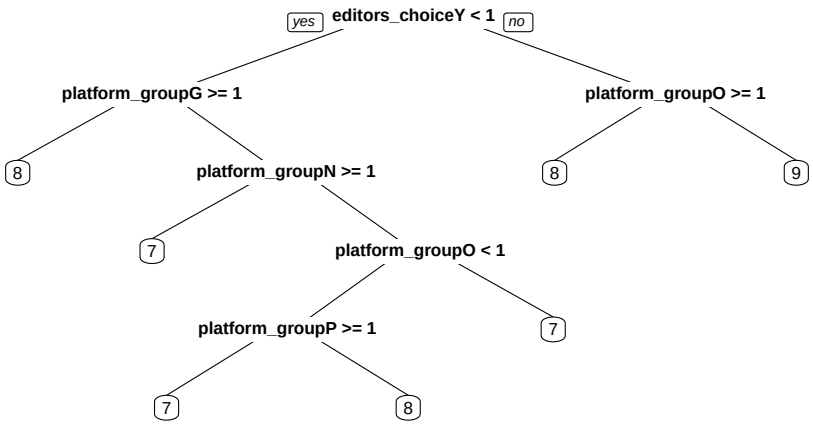
```
##
## Call:
## lm(formula = score ~ ., data = reg_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.0058 -0.6557  0.2087  1.0293  3.4429
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 6.72969    0.03872 173.784  < 2e-16 ***
## editors_choiceY             2.30134    0.02689  85.593  < 2e-16 ***
## platform_groupGame.Boy     -0.67136    0.05717 -11.744  < 2e-16 ***
## platform_groupNintendo     -0.61872    0.04200 -14.732  < 2e-16 ***
## platform_groupOther        -0.36714    0.05788  -6.343 2.31e-10 ***
## platform_groupPlayStation -0.36001    0.04068  -8.850  < 2e-16 ***
## platform_groupWindows      -0.22569    0.04333  -5.209 1.92e-07 ***
## platform_groupXbox         -0.22600    0.04486  -5.038 4.75e-07 ***
## genre_groupRacing          -0.17263    0.04172  -4.138 3.52e-05 ***
## genre_groupRPG              0.40065    0.04496   8.912  < 2e-16 ***
## genre_groupStrategy         0.19513    0.04661   4.186 2.85e-05 ***
## release_month2              0.10024    0.04397   2.280 0.022630 *
## release_month3              0.10592    0.04132   2.563 0.010379 *
## release_month4              0.09026    0.04480   2.015 0.043954 *
## release_month6              0.15790    0.04210   3.751 0.000177 ***
## release_month8              0.20687    0.04384   4.719 2.39e-06 ***
## release_month9              0.28605    0.04005   7.142 9.52e-13 ***
## release_month10             0.20184    0.03598   5.611 2.05e-08 ***
## release_month11             0.12746    0.03437   3.709 0.000209 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.42 on 18570 degrees of freedom
## Multiple R-squared:  0.3127, Adjusted R-squared:  0.312
```

```
## F-statistic: 469.3 on 18 and 18570 DF,  p-value: < 2.2e-16
```

The first iteration of the linear regression showed multicollinearity between score and score phrase with a multiple R-squared of 0.9726. After removing score_phrase, R-squared drops to 0.3187. In an attempt to improve R-squared, near zero variance predictors (NZP) were identified and removed because they are non-informative and tend to occur when breaking categorical variables into dummy variables, as was done above. After removing NZP and checking for high collinearity, a few iterations of the linear model were run in order to narrow down which features are not significant. A few of the fields removed are platform_groupApple, genre_groupAdventure, and genre_groupSports. After removing non-significant fields, the adjusted R-squared is now 0.3127.

CART/Random Forest






```
##
## Call:
##  randomForest(formula = score ~ ., data = Train, ntree = 500)
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 6
##
##           Mean of squared residuals: 2.026652
##                     % Var explained: 30.95
```

```
## [1] 0.3269865
```

```
##          [,1]
## 0.5 0.3039893
## 0.7 0.3039893
## 0.8 0.3039893
## 1   0.3039893
## 1.1 0.3039893
## 1.2 0.3039893
## 1.3 0.3039893
## 1.4 0.3039893
## 1.5 0.3039893
## 1.7 0.3039893
## 1.8 0.3039893
## 1.9 0.3039893
## 2   0.3039893
## 2.1 0.3039893
## 2.2 0.3039893
```

```
## 2.3 0.3039893
## 2.4 0.3039893
## 2.5 0.3039893
## 2.6 0.3039893
## 2.7 0.3039893
## 2.8 0.3039893
## 2.9 0.3039893
## 3   0.3039893
## 3.1 0.3039893
## 3.2 0.3039893
## 3.3 0.3039893
## 3.4 0.3039893
## 3.5 0.3039893
## 3.6 0.3039893
## 3.7 0.3039893
## 3.8 0.3039893
## 3.9 0.3039893
## 4   0.3039893
## 4.1 0.3039893
## 4.2 0.3039893
## 4.3 0.3039893
## 4.4 0.3039893
## 4.5 0.3039893
## 4.6 0.3039893
## 4.7 0.3039893
## 4.8 0.3039893
## 4.9 0.3039893
## 5   0.3039893
## 5.1 0.3039893
## 5.2 0.3039893
## 5.3 0.3039893
## 5.4 0.3039893
## 5.5 0.3039893
## 5.6 0.3039893
## 5.7 0.3039893
## 5.8 0.3039893
## 5.9 0.3039893
## 6   0.3039893
## 6.1 0.3039893
## 6.2 0.3039893
## 6.3 0.3039893
## 6.4 0.3039893
## 6.5 0.3039893
## 6.6 0.3039893
## 6.7 0.3039893
## 6.8 0.3039893
## 6.9 0.3039893
## 7   0.3039893
## 7.1 0.3039893
## 7.2 0.3039893
## 7.3 0.3039893
## 7.4 0.3039893
## 7.5 0.3039893
## 7.6 0.3039893
```

```
## 7.7 0.3039893
## 7.8 0.3039893
## 7.9 0.3039893
## 8   0.3039893
## 8.1 0.3039893
## 8.2 0.3039893
## 8.3 0.3039893
## 8.4 0.3039893
## 8.5 0.3039893
## 8.6 0.3039893
## 8.7 0.3039893
## 8.8 0.3039893
## 8.9 0.3039893
## 9   0.3039893
## 9.1 0.3039893
## 9.2 0.3039893
## 9.3 0.3039893
## 9.4 0.3039893
## 9.5 0.3039893
## 9.6 0.3039893
## 9.7 0.3039893
## 9.8 0.3039893
## 9.9 0.3039893
## 10  0.3039893
```

```
##           [,1]
## 0.5 0.002800549
## 0.7 0.008878185
## 0.8 0.050681353
## 1   0.168364769
## 1.1 0.008878185
## 1.2 0.074436848
## 1.3 0.050681353
## 1.4 0.050681353
## 1.5 0.170604906
## 1.7 0.050681353
## 1.8 0.008878185
## 1.9 0.141738963
## 2   0.188234501
## 2.1 0.098294830
## 2.2 0.095596742
## 2.3 0.068539370
## 2.4 0.141738963
## 2.5 0.206847665
## 2.6 0.040542881
## 2.7 0.183752866
## 2.8 0.114532262
## 2.9 0.031017862
## 3   0.209434249
## 3.1 0.115976027
## 3.2 0.128864287
## 3.3 0.111168899
## 3.4 0.106220543
## 3.5 0.270060070
```

```
## 3.6 0.134784624
## 3.7 0.156766723
## 3.8 0.186255350
## 3.9 0.147748589
## 4   0.228438652
## 4.1 0.154514978
## 4.2 0.153497017
## 4.3 0.163411769
## 4.4 0.135636163
## 4.5 0.314802332
## 4.6 0.170604906
## 4.7 0.180114702
## 4.8 0.183259583
## 4.9 0.182905655
## 5   0.254307880
## 5.1 0.272243153
## 5.2 0.144062721
## 5.3 0.198881102
## 5.4 0.204123980
## 5.5 0.297174699
## 5.6 0.166627655
## 5.7 0.147886183
## 5.8 0.225282896
## 5.9 0.196638929
## 6   0.288484908
## 6.1 0.209504802
## 6.2 0.207938385
## 6.3 0.196227515
## 6.4 0.138348341
## 6.5 0.304851280
## 6.6 0.137854307
## 6.7 0.125600725
## 6.8 0.187693565
## 6.9 0.216742135
## 7   0.304526719
## 7.1 0.157718903
## 7.2 0.143526487
## 7.3 0.114916755
## 7.4 0.136100400
## 7.5 0.250626873
## 7.6 0.176002450
## 7.7 0.166906413
## 7.8 0.232580019
## 7.9 0.164058437
## 8   0.068128708
## 8.1 0.033286442
## 8.2 0.025397658
## 8.3 0.010624332
## 8.4 0.073718808
## 8.5 0.311736210
## 8.6 0.316118824
## 8.7 0.309791793
## 8.8 0.314440048
## 8.9 0.306944876
```

```
## 9   0.294821162
## 9.1 0.298964548
## 9.2 0.296214965
## 9.3 0.284737042
## 9.4 0.284388680
## 9.5 0.300943109
## 9.6 0.283367433
## 9.7 0.285865991
## 9.8 0.286941378
## 9.9 0.283367433
## 10  0.298015403
```

When looking at the first tree, the only node shown is for editor's choice. If editor's choice is greater than 0.5, it is no or 1. One important note is that a high score can imply an editor's choice designation. However, a high score, say 8 or greater, does not automatically receive an editor's choice designation. Using the complexity parameter (cp), we can force more nodes to appear with cp = .0025. Looking at the tree, we see that editor's choice has remained as the first node. The first tree produces an R-squared of 30.4% which is similar to the linear regression model. When forcing the tree to have multiple nodes, the same R-squared calculation doesn't work as nicely nor is it the most efficient.

The random forest model, built on the training dataset, appears to be the best model based on a comparison of R-squared at 32.69% versus 30.4% (CART), and 31.27% (linear regression).

Conclusion

After looking at platform, genre, and release date in a myriad of ways, Action can be conclusively named the top pick in Genre for types of video games released. When looking at Genre by score, RPG is the favorite when it comes to review score. When looking at the data by Platform, the answer is not as clear. While Playstation dominated for all years combined, even though it was released years after Nintendo or Windows, the exhibit by score does not show Playstation to have the highest rated games. This would imply that while a platform can release a large amount of games, it does not necessarily mean that those games will be favorably reviewed. To truly determine what platform reigns supreme, if there is one, I would look into tying video game sale data with the video game reviews to look at sale volume versus platform, genre, date, and score.