Video Game Sales Submission

GL

3/2/2020

Introduction

The following report will describe the predictive data model created for Video Game Sales using the data provided from Kaggle.com using the URL below. The data is a web scrape of Metacritic ratings and information for video games. There are several columns detailing the level of unit sales, split by each major video game buying region: North America, Europe and Japan, as well as the Genre, Publisher, Platform and other fields. The list below shows all the columns included in the web scraped dataset.
Data URL: https://www.kaggle.com/rush4ratio/video-game-sales-with-ratings

```
[1] "Column Names"

 [1] "Name"            "Platform"        "Year_of_Release" "Genre"
 [5] "Publisher"       "NA_Sales"        "EU_Sales"        "JP_Sales"
 [9] "Other_Sales"     "Global_Sales"    "Critic_Score"    "Critic_Count"
[13] "User_Score"      "User_Count"      "Developer"       "Rating"
[17] "Sales_Rank"
```

See below for an example of the data set and its contents:

```
                     Name Platform Year_of_Release        Genre Publisher
1               Wii Sports      Wii            2006       Sports  Nintendo
2        Super Mario Bros.      NES            1985     Platform  Nintendo
3           Mario Kart Wii      Wii            2008       Racing  Nintendo
4        Wii Sports Resort      Wii            2009       Sports  Nintendo
5 Pokemon Red/Pokemon Blue       GB            1996 Role-Playing  Nintendo
6                  Tetris       GB            1989       Puzzle  Nintendo
  NA_Sales EU_Sales JP_Sales Other_Sales Global_Sales Critic_Score Critic_Count
1    41.36    28.96     3.77        8.45        82.53           76           51
2    29.08     3.58     6.81        0.77        40.24           NA           NA
3    15.68    12.76     3.79        3.29        35.52           82           73
4    15.61    10.93     3.28        2.95        32.77           80           73
5    11.27     8.89    10.22        1.00        31.37           NA           NA
6    23.20     2.26     4.22        0.58        30.26           NA           NA
  User_Score User_Count Developer Rating Sales_Rank
1          8        322  Nintendo      E          6
2                    NA                           6
3        8.3        709  Nintendo      E          6
4          8        192  Nintendo      E          6
5                    NA                           6
6                    NA                           6
```

Since the sales figures from each region will inevitably feed into the Global sales, this predictive model will be looking at NA sales as a final predicted value.

The following table is a grouping of ranks the games by the number of units sold. The top 10 sellers in the data set are shown below.

	Sales_Rank	n
Above 1 million units sold	6	918
500,000 - 1 million	5	1169
250,000 - 500,000	4	1986
100,000 - 250,000	3	3490
50,000 - 100,000	2	2414
Below 50,000	1	6742

	Name	Platform	Year_of_Release	NA_Sales
1	Wii Sports	Wii	2006	41.36
2	Super Mario Bros.	NES	1985	29.08
3	Mario Kart Wii	Wii	2008	15.68
4	Wii Sports Resort	Wii	2009	15.61
5	Tetris	GB	1989	23.20
6	Wii Play	Wii	2006	13.96
7	New Super Mario Bros. Wii	Wii	2009	14.44
8	Duck Hunt	NES	1984	26.93
9	Kinect Adventures!	X360	2010	15.00
10	Super Mario World	SNES	1990	12.78

We can see that the vast majority of the population sold less than 1 million units. The data models will attempt to see which variables tend to influence the level of sales in North America and if predictions can be made to see how many sales a video game will have.

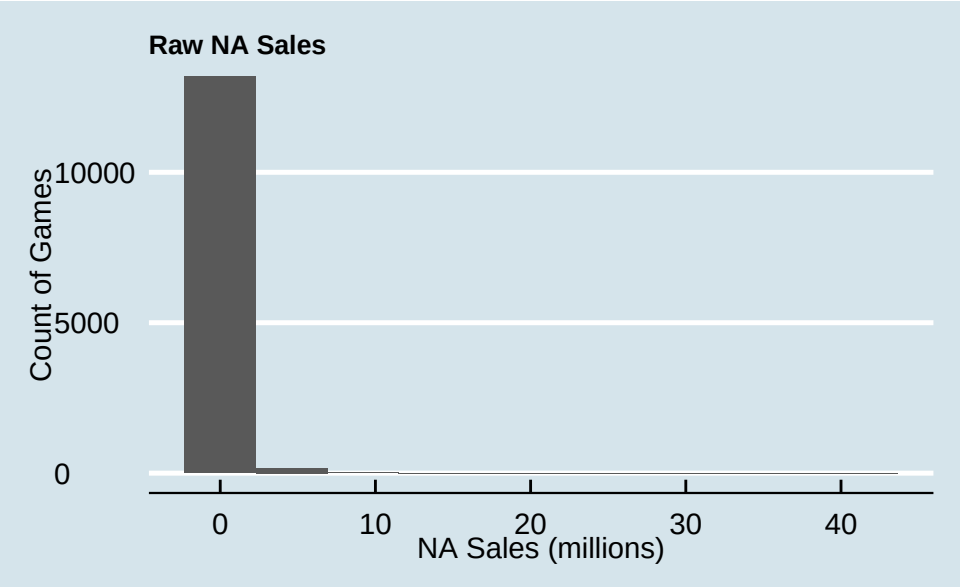
The table below gives a breakdown of the data set structure to see if we can even use some of the columns, or if we need to clean up the data. We will have to coerce the User and Critic Scores/Counts and Year of Release columns into numerics, which will end up introducing some "NAs" into the list, but it will allow for consistency in the data model.

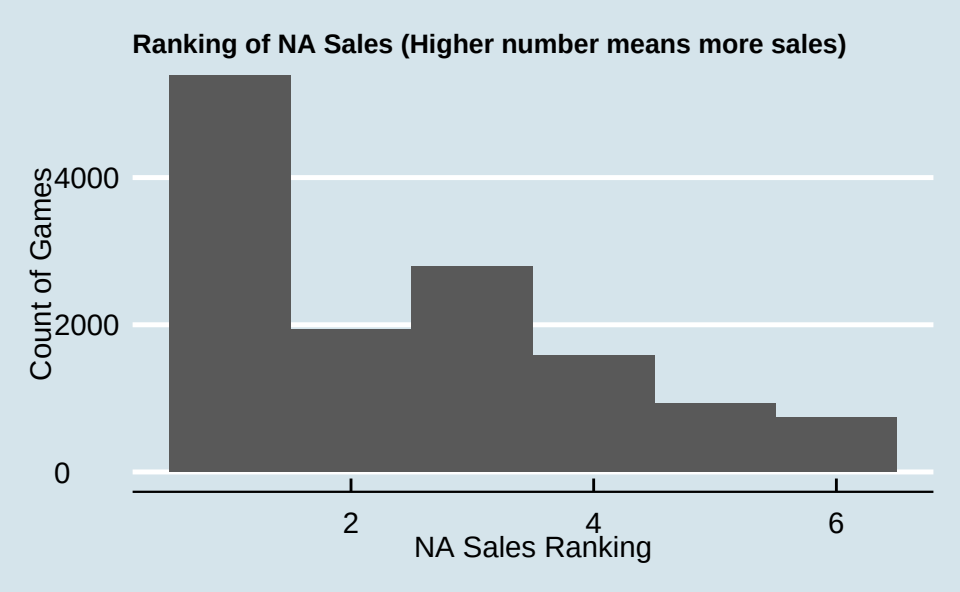
```
    Name              Platform         Year_of_Release       Genre
Length:16719       Length:16719       Length:16719       Length:16719
Class :character   Class :character   Class :character   Class :character
Mode  :character   Mode  :character   Mode  :character   Mode  :character




 Publisher            NA_Sales          EU_Sales         JP_Sales
Length:16719       Min.   : 0.0000   Min.   : 0.000   Min.   : 0.0000
Class :character   1st Qu.: 0.0000   1st Qu.: 0.000   1st Qu.: 0.0000
Mode  :character   Median : 0.0800   Median : 0.020   Median : 0.0000
                   Mean   : 0.2633   Mean   : 0.145   Mean   : 0.0776
                   3rd Qu.: 0.2400   3rd Qu.: 0.110   3rd Qu.: 0.0400
                   Max.   :41.3600   Max.   :28.960   Max.   :10.2200

 Other_Sales        Global_Sales      Critic_Score    Critic_Count
Min.   : 0.00000   Min.   : 0.0100   Min.   :13.00   Min.   :  3.00
1st Qu.: 0.00000   1st Qu.: 0.0600   1st Qu.:60.00   1st Qu.: 12.00
Median : 0.01000   Median : 0.1700   Median :71.00   Median : 21.00
Mean   : 0.04733   Mean   : 0.5335   Mean   :68.97   Mean   : 26.36
3rd Qu.: 0.03000   3rd Qu.: 0.4700   3rd Qu.:79.00   3rd Qu.: 36.00
Max.   :10.57000   Max.   :82.5300   Max.   :98.00   Max.   :113.00
                                     NA's   :8582    NA's   :8582
 User_Score           User_Count       Developer            Rating
Length:16719       Min.   :    4.0   Length:16719       Length:16719
Class :character   1st Qu.:   10.0   Class :character   Class :character
Mode  :character   Median :   24.0   Mode  :character   Mode  :character
                   Mean   :  162.2
                   3rd Qu.:   81.0
                   Max.   :10665.0
                   NA's   :9129
  Sales_Rank
Min.   :1.000
1st Qu.:1.000
Median :2.000
Mean   :2.472
3rd Qu.:3.000
Max.   :6.000
```

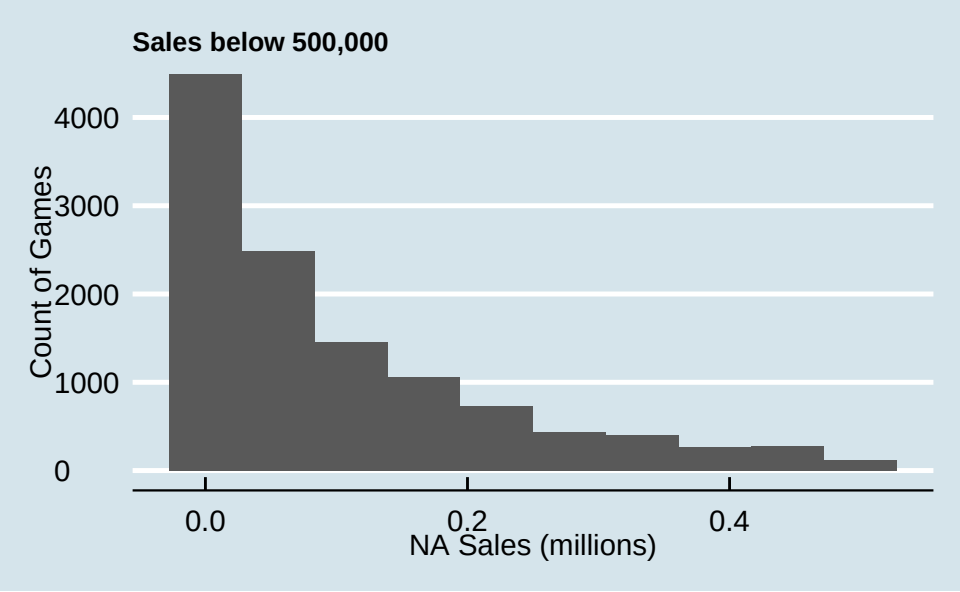
Methods and Analysis

First we start off with viewing the population of the training set. The black highlighting indicates missing data in the training set. We will still attempt to use the User/Critic Scores in predicting NA Sales.

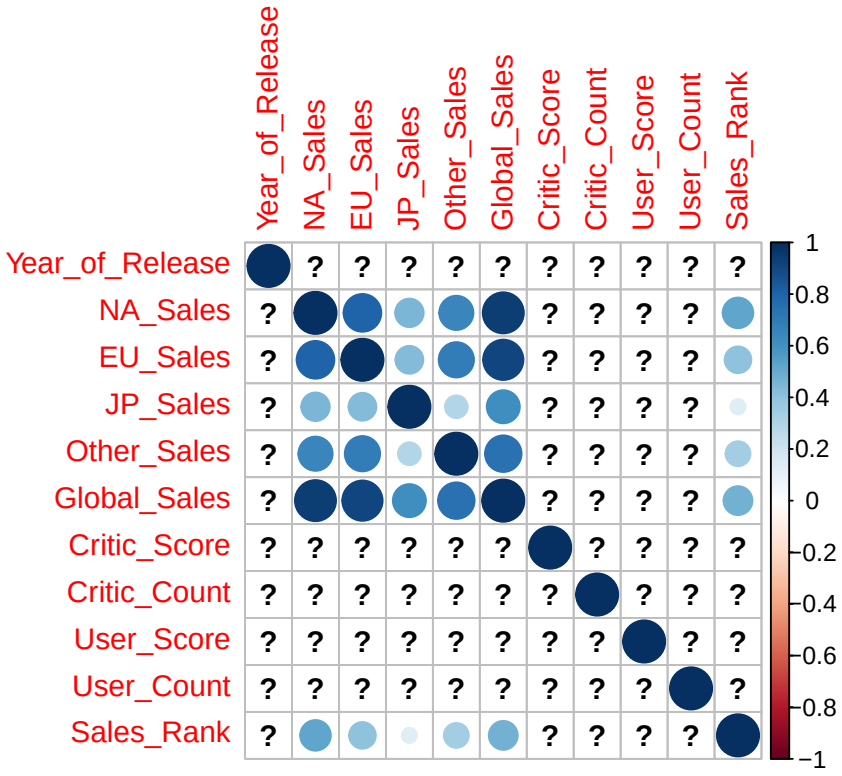
The following graphs show the distribution of the video games by NA Sales. The definition of rankings is provided in the Introduction.







Next we will view if there is any correlation with the numeric variables.



We can see that the Sales figures will have a clear correlation but User or Critic scores and the number of reviews don't directly impact other factors. Creating a predictive model will have some level of uncertainty, especially with the large number of video games with very low sales and a comparatively smaller selection of games that had enormous success.

Moving on to testing different models, we begin with seeing if just taking the average NA Sales will allow us to closely predict sales of other games

```
# A tibble: 1 x 2
  method            RMSE
  <chr>            <dbl>
1 Just the average 0.912
```

The RMSE of taking just the average is quite high, since the 75th percentile of NA Sales is 0.240. We will try taking different factors into account, specifically, the Platform and Genres to see if they have any impact on the level of sales in North America.

```
# A tibble: 3 x 2
  method                          RMSE
  <chr>                          <dbl>
1 Just the average               0.912
2 Platform Effect Model          0.882
3 Platform + Genre Effect Model 0.917
```

An attempt was then made to regularize the data sets and then evaluate predictive power of several options:

- Regularized Platform Effect Model
- Regularized Platform + Genre Effect Model
- Regularized Platform + Genre + Critic Score + User Score Effect Model
- Regularized Platform + Genre + Critic Score + User Score + Year of Release Effect Model

Example code for the first regularized model is below:

```
############################
# Regularized Platform + Genre Effect Model
# Regularizing the original model that relies on Platform and Genre variables to predict NA Sales
# Check to see if regularizing will improve predictions
# Train to find the best lambda
lambdas <- seq(0, 10, 0.25)

rmses <- sapply(lambdas, function(l){
  mu <- mean(train_set$NA_Sales)
  b_i <- train_set %>%
    group_by(Platform) %>%
    summarize(b_i = sum(NA_Sales - mu)/(n()+l))
  b_g <- train_set %>%
    left_join(b_i, by = "Platform") %>%
    group_by(Genre) %>%
    summarize(b_g = sum(NA_Sales - b_i - mu)/(n()+l))
  predicted_ratings <- train_set %>%
    left_join(b_i, by = "Platform") %>%
    left_join(b_g, by = "Genre") %>%
    mutate(pred = mu + b_i + b_g) %>%
    .$pred
  return(RMSE(predicted_ratings, train_set$NA_Sales))
})

lambda <- lambdas[which.min(rmses)]

# Test prediction capability with the lambda from the previous step
# Using test set for this function
rmses <- sapply(lambda, function(l){
  mu <- mean(test_set$NA_Sales)
  b_i <- test_set %>%
    group_by(Platform) %>%
    summarize(b_i = sum(NA_Sales - mu)/(n()+l))
  b_g <- test_set %>%
    left_join(b_i, by = "Platform") %>%
    group_by(Genre) %>%
    summarize(b_g = sum(NA_Sales - b_i - mu)/(n()+l))
  predicted_ratings <- test_set %>%
    left_join(b_i, by = "Platform") %>%
    left_join(b_g, by = "Genre") %>%
    mutate(pred = mu + b_i + b_g) %>%
    .$pred
  return(RMSE(predicted_ratings, test_set$NA_Sales))
})
```

Results

```
[1] "Table of RMSEs"
```

method	RMSE
Just the average	0.9120406
Platform Effect Model	0.8819838
Platform + Genre Effect Model	0.9172674
Regularized Platform Effect Model	0.8819838
Regularized Platform + Genre Effect Model	0.8774207
Regularized Platform + Genre + Critic Score + User Score Effect Model	0.8461337
Regularized Platform + Genre + Critic Score + User Score + Year of Release Effect Model	0.7872807

Based on RMSE, the best model is:

```
[1] "Regularized Platform + Genre + Critic Score + User Score + Year of Release Effect Model"
```

```
[1] "It has an RMSE score of: 0.787281"
```

Conclusion

In conclusion, the best data model has a RMSE of ~787,000 North America Sales units. When judging by the range of values (0 - 41.6), that seems very powerful, but unfortunately the NA Sales population of video games is heavily skewed towards the lower value. As we can see, the 75th percentile is still below a quarter of a million units sold.

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 0.0000  0.0000  0.0800  0.2633  0.2400 41.3600
```

With the hit or miss nature of video games in general where they either become very popular and sell extremely well or have low sales, we would have to make sacrifices in one direction or another: either keeping the games that sold extremely well (>75th percentile) or keep only the low selling games (below 75th percentile) to allow for for stronger predictive capability. Using the current method favors the huge outliers and skews the results to a higher predictive unit sale figure.