

# IGN Data: Data Wrangling

by Ellen A. Savoye

The IGN dataset sourced from Kaggle (<https://www.kaggle.com/egrinstein/20-years-of-games>) and IGN (<http://ign.com/games/reviews>), via a crawl, consists of 20 years worth of video-game data. To take a proper look at the data, I loaded the original dataset as a CSV file and the necessary libraries.

```
tbl_df(IGN_data)
```

Of the variables available for use, `score_phrase`, `platform`, `score`, `genre`, `editors_choice`, `release_year`, `release_month`, and `release_day` are the ones I am using 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.

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

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

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

```
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/manufacture 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.

```
# 59 variables in original platform column
```

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

```
Platform_Map <- read.csv("platform_map.csv")
```

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

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

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)

IGN_data[IGN_data$genre == "", ]

# 113 unique factors in genre column

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

Genre_Map <- read.csv("genre_map.csv")

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

unique(IGN_data$genre_group)
```

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 later in the course.

```
write.csv(IGN_data, "ign_clean.csv")
```