Yu-Gi-Oh! Meets Data Analysis!

Gabriel SH (yeye), tutored by Geeked.id

9/9/2021

Table of Contents

# 1. Introduction

## 1.1 A Bit About Me

Hellow, I’m Gabriel, but you may call me Yeye for convenience, and I would prefer you to call me that :)

## 1.2 A bit about Yu-Gi-Oh!

In this portofolio builder, I would like to to analyze **Yu-Gi-Oh!** card game, spesifically the TCG (English) version. YGO is in my kind of *biased* opinion, the best card game I’ve ever played, and I stumbled upon this dataset by chance.

Sadly though, I have left playing it since long ago because of **money** issues. If you don’t know, a single card can cost a maximum of **Rp 1.000.000**, or maybe more.

## 1.3 A Little Note About This Portofolio

In this portofolio, I want to explore tidyverse, which is R package (more like a bundle), that can generalize the style of data analysis using R, since many people have different coding styles / approach, using tidyverse makes the code style uniform and much more simple and easy to understand.

#install.packages("tidyverse")  
library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

# 2. Meet The Data

## 2.1 Dataset Brief

The datasets are provided by [James Palmer](https://www.kaggle.com/jpalmer2) from kaggle.com, and the data itself was scraped from [Yu-Gi-Oh! Fandom Wiki](https://yugioh.fandom.com/wiki/Yu-Gi-Oh!). It consist of 4 files, which upon slight inspection, it’s actually quite similar.

## 2.2 Dataset Characteristics

### 2.2.1 Parameters

* **Name** : Card name
* **Card.Type** : Monster / Spell / Trap card
* **Attribute** : Monster Attribute (LIGHT, DARK, EARTH, WATER, etc)
* **Monster.Type** : Monster type (Fairy, Zombie, Fiend, Pyro, etc), also represents the Monster Type (Synchro, Effect, Xyz, etc)
* **Level.Rank** : Monster Level / Xyz Monster Rank
* **ATK.DEF** : ATK & DEF of the monster
* **Passcode** : Card code printed on the bottom left side
* **Materials…Ritual.spell** : Material for Synchro Monsters / Fusion Monsters / Xyz Monsters / Link Monsters
* **Effect.Type** : This is a bit complex, but I can give you an easy example, suppose it’s *Continuous* , then as long as that card is face up on the field, the effect will remain.
* **Effect** : The actual effect of the card printed on it.
* **Spell.Trap.type** : The type of the Spell/Trap card, like *Equip* / *Continuous* / *Ritual* / etc
* **TCG.sets** : This represents the package code where the card originated from. A package can be a *Structure Deck* , *Collector Tin* , *Booster Pack* , etc. A card may have more than 1 package code because there’s a possibility of “Reprinting” the card to introduce lower money-cost
* **Ban\_list** : Banlist status of the card, either *Unlimited* / *Semi Limited* / *Limited* / *Forbidden*
* **Number.of.sets** : If it’s more than 1, then the card has already been reprinted.
* **Link.Arrows** : The direction of *Link Arrow* for Link Monster, it’s octagonal (8-way) and may consist of more than 1 arrow. The number of arrows also represents the **Link Rating**, which is like Level / Rank for other Monster.
* **Pendulum.Scale** : The pendulum scale of Pendulum Monsters.
* **Set.Name** : The card package name
* **Relase.Date** : Well… the release date.

names(yugi)

## [1] "X" "Name"   
## [3] "Card.type" "Attribute"   
## [5] "Monster.Type" "Level.Rank"   
## [7] "ATK.DEF" "Passcode"   
## [9] "Materials...Ritual.spell" "Effect.type"   
## [11] "Effect" "Spell.Trap.type"   
## [13] "TCG.sets" "Ban\_list"   
## [15] "Number.of.sets" "Link.Arrows"   
## [17] "Pendulum.Scale" "Rarities"   
## [19] "Set.Name" "Release.Date"

### 2.2.2 A Little Peek to the Data

str(yugi)

## 'data.frame': 22293 obs. of 20 variables:  
## $ X : int 0 1 2 3 4 5 6 7 8 9 ...  
## $ Name : chr "\"A\" Cell Breeding Device" "\"A\" Cell Incubator" "\"A\" Cell Recombination Device" "\"A\" Cell Scatter Burst" ...  
## $ Card.type : chr "Spell " "Spell " "Spell " "Spell " ...  
## $ Attribute : chr "" "" "" "" ...  
## $ Monster.Type : chr "" "" "" "" ...  
## $ Level.Rank : num NA NA NA NA NA NA NA NA NA NA ...  
## $ ATK.DEF : chr "" "" "" "" ...  
## $ Passcode : chr "34541863" "64163367" "91231901" "73262676" ...  
## $ Materials...Ritual.spell: chr "" "" "" "" ...  
## $ Effect.type : chr "Trigger-like" "Continuous-like, Trigger-like" "Effect, Ignition-like" "Effect" ...  
## $ Effect : chr "During each of your Standby Phases, put 1 A- Counter on 1 face-up monster your opponent controls" "Each time an A- Counter(s) is removed from play by a card effect, place 1 A- Counter on this card. When this ca"| \_\_truncated\_\_ "Target 1 face-up monster on the field; send 1 \" Alien\" monster from your Deck to the Graveyard, and if you do"| \_\_truncated\_\_ "Select 1 face-up \" Alien\" monster you control. Destroy it and distribute new A- Counters equal to its Level a"| \_\_truncated\_\_ ...  
## $ Spell.Trap.type : chr "Continuous " "Continuous " "Quick-Play " "Quick-Play " ...  
## $ TCG.sets : chr "FOTB-EN043" "GLAS-EN062" "INOV-EN063" "STON-EN041" ...  
## $ Ban\_list : chr "Unlimited " "Unlimited " "Unlimited " "Unlimited " ...  
## $ Number.of.sets : int 1 1 1 1 1 1 3 3 3 1 ...  
## $ Link.Arrows : chr "" "" "" "" ...  
## $ Pendulum.Scale : num NA NA NA NA NA NA NA NA NA NA ...  
## $ Rarities : chr "Common" "Common" "Common" "Common" ...  
## $ Set.Name : chr "Force of the Breaker Sneak Peek Participation Card" "Gladiator's Assault Sneak Peek Participation Card" "Invasion: Vengeance Sneak Peek Participation Card" "Strike of Neos Sneak Peek Participation Card" ...  
## $ Release.Date : chr "May 16, 2007" "November 14, 2007" "November 4, 2016" "February 28, 2007" ...

## 2.3 Problem

Firstly, of course I will need the name of the card.

# 3. Processing the Data

## 3.1 Data Wrangling

## 3.2 Data Visualization

# 4. Infering the Data

## 4.1 What We Have Got So Far

## 4.2 Conclusion