STEAMCONOMICS- A TOOL FOR ANALYZING THE ECONOMICS OF IN-GAME PURCHASES

Stefanos Logothetis

April 30th, 2018

Trading and Gaming

- A recent trend has been growing in the gaming industry- user-driven transactions within games
- Users can trade virtual items with each other for real-world currency, or buy in-game items with said currency
- Recent hit games that include this mechanic-
  - *PLAYERUNKNOWN BATTLEGROUNDS (produced by PUBG corporation)*
  - *Team Fortress 2 (produced by Valve)*
  - *Overwatch (produced by Activision/Blizzard)*

Why study it?

- Naturally, the question arises, what is so important about these transactions if they're user to user?
- Several of these games are what are categorized as "Free to Play", meaning the user does not pay any up-front cost for the game
  - *E.g. Team Fortress 2, DOTA 2*
  - *That means all revenue comes from in-game purchases and trading*
- Usually, for real-world currency, the trading platform will take some money out of what the user sells an item for.
- Despite the "free" nature of these games, the microtransaction business has an estimated value of 22 Billion USD[1].

[1]https://www.pcgamer.com/revenue-from-pc-free-to-play-microtransactions-has-doubled-since-2012/

Why Steam?

- To study these transactions, I used data from an online game distribution and trading platform called Steam.
- Steam is one of the largest distribution platforms for games on PC.
  - *~ 16 Million users and rising*
  - *~ Host to the Steam Community market, a one-of-a-kind platform where users sell in-game items for currency*
  - *Valued at $2.5 billion*[2]
- Unfortunately, one of the only platforms where such data is available.

[2]https://www.nytimes.com/2012/09/09/technology/valve-a-video-game-maker-with-few-rules.html?_r=2&pagewanted=all

How to get the data

- All of the market data was taken off of Steam, which allows users to take market data off their site in JSON format for free without a key.
- However, Steam's game data is stored on an entirely different part of the site, meaning we need to combine data from two sets to get a full picture
  - *1: Get list of apps from* *http://api.steampowered.com/ISteamApps/GetAppList/v0002/*
  - *2: Search list to find id for our game*
  - *3: Feed the ID into Steam's market API to get trade data*
- The code for both of these functions mainly uses the libraries jsonlite, curl and httr

Code Sample

```
get_items_and_prices <- function(app.name, num_items = 100){

  if(app.name %in% game_list$applist.apps.name){

    appid = game_list$applist.apps.appid[which(game_list$applist.apps.name == app.name)]

  }

  else{

    stop("Your game isn't in Steam's Library. Check your spelling or try another game")

  }

  for(i in 0:(num_items/100 - 1)){

    url = paste("https://steamcommunity.com/market/search/render/?query=&start=", 100*i, "&count=100&norender=1&appid=", appid, sep = "")

    new_data = url %>% curl() %>% readLines() %>% fromJSON(flatten = TRUE) %>% as.data.frame() %>%  subset(select = c("results.name", "results.sell_listings","results.sell_price",
"results.sell_price_text", "results.app_icon","results.app_name"))

    if(i == 0){

      out = new_data

    }

    else{

      out = rbind(out, new_data)

    }

  }

  out

}
```

The end result

- The main idea of this project was to be able to quickly gather data from steam and market resources, and display all the information in one place.
  - *Best way to do this was through R's shiny package*
- Have one area where one can search through steam and look at community market data, and another to look at stock market data
- Ideally, user only has to put in one parameter, the name of the game, and the app would do the rest, including
  - *Getting the publisher of the game*
  - *List of items being sold currently and prices*
  - *Displaying stock data of the publisher*

The Shiny App in Action

Issues

- Getting the stock quote from the name of the publisher is hard
  - *There simply isn't any good way of searching the name of a publisher and getting back the stock symbol*
- The stock market might not be the best way of observing the impact of in-game transactions on the value of a company
  - *Many publishers are parts of larger corporations, some of which make multiple games.*
  - *The impact of a single game can be hard to determine*
- The data we have is a snapshot at one instance in time
  - *To get a good idea of what's happening, we need to see the number and price of items over time*
  - *Although Steam keeps records of this, there's no way to parse it off of steam's website or API*

Future Goals

- Try to implement some way to get the stock name from just a developer name.
- Make a part of the app persistent to allow for acquisition of data over time
- Make another interface that allows for plotting of all games from a given publisher/developer.
- Make tools to analyze any sort of correlation between the market value and the value of in-game items.

THANK YOU