Scraping for Games

Project Luther

Project description:

I used project Luther as an opportunity to scrape data from Boardgamegeek.com, in order to explore what features make a game popular.

Data overview:

Starting from a page on Boardgamegeek (BGG) that lists all the games by their community voted rating, I used Beautifulsoup to scrape an HTML table object for data. Beautifulsoup, a Python package for parsing HTML and XML documents, is the main tool I used in all of my scraping. I originally tinkered a bit with Selenium to open up links and gather data from individual game pages, however I realialized I could simplify and forgo it by using Beautifulsoup for the entirety of the scrape.

From the board game rank table, I not only scraped the "surface" text of the page, but also the URLs of each of the pages dedicated to individual games. At the end of the scraping process I had two pandas dataframes: one containing the rank, name, rating, and number of votes a game had, and a separate data frame for each game's particular parameters. After cleaning up, dropping null values, and merging the two data frames I had 699 individual observations with the following parameters:

- **Rank**: the rating a game hold on BGG
- **Title**: game title
- **Geek rating**: a Bayesian average
- **Avg Rating**: average user rating
- **Num Voters**: the number of users who voted
- **Num Player**: best number of players for the game, as voted by the community[1]
- **Complexity rating**: Score out of 5
- **Maxplaytime**: published established maximum duration of a session
- **Minplaytime**: published established minimum duration of a session

[1] At the times when there was no community voted best number of players, the model simply used the maximum number of players suggested by the publisher.

- **Yearpublished**: the year a game far published in[2]

Model overview:

I ended up reporting the finding for a basic OLS model. I tried using Elastic Net and Lasso, however the results weren't significantly different to justify the increased complexity of the resulting model.

Findings:

I believe the findings my model yielded, which changed drastically as I increased the size of the data set I was using, are flawed. Ultimately my R-squared was ~.88. It is a number I think is suspiciously high, however I ran into complications exploring the issue further due to time constraints.

One of the the implicit finding of this project, for me, is that setting up a successful scrape can take significantly more time that one initially anticipates.

Next Steps:

Moving forward, I would like to first go back to my model and investigate whether my findings are indeed reliable and can be trusted.

To fine tune the model, I would also like to include more parameters. For Project Luther I limited my scrape to numerical values only, however there are great insights to be gained from categorical variables available on each game's unique description page (URLs for which I already have available).

For instance, I believe that including the genre of the game will have a significant positive impact on the predictive capabilities of the model. Via a brief manual scan through the top games, it appears that many belong to a Strategy or Thematic category.

[2] The only game without a Yearpublished value was Go - a game invented in ancient China more than 2,500 years ago and believed to be the oldest board game continuously played today.