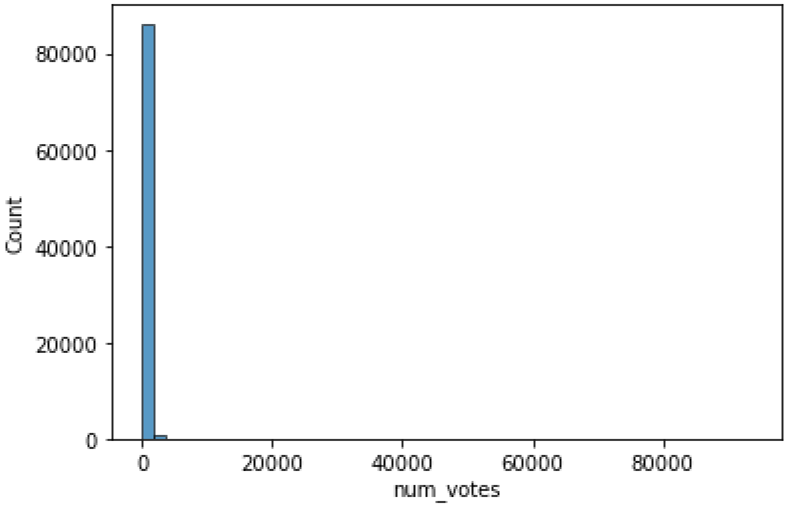
Capstone 2 Final Project Report

Analyzing BoardGameGeek.com rankings to find the formula for the board game “killer app”

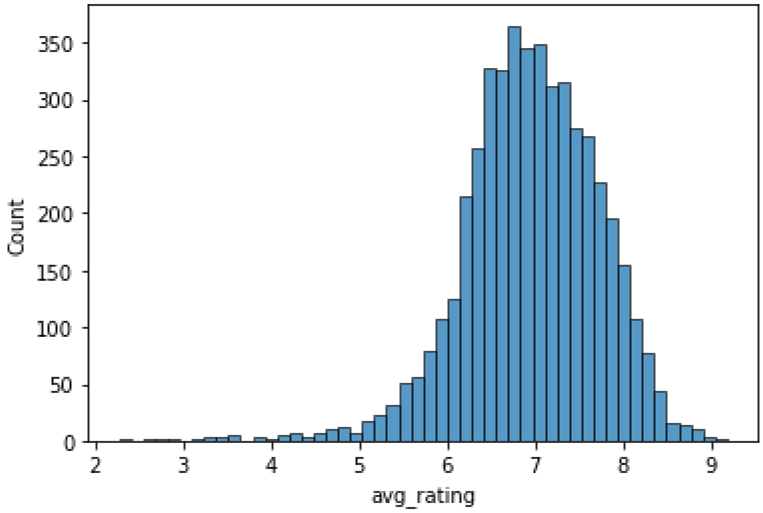
A group of friends has gotten together to create a new [fictitious] board game company. Because they are self funded, they need to make sure their first game is successful. They decided to analyse the user rankings on BoardGameGeek.com. The assumption was that the user popularity was a proxy for how much the game would sell and how successful it would be. If they could find out what features lead to a highly ranked game, they could ensure that their first project obtained enough notoriety (and revenue!) to sustain the company on further ventures. The plan was to run a regression analysis of the board game features against the listed user rating. Once the most popular features were determined, the team could include those features in its first game creation.

The BoardGameGeek (“BGG”) database was downloaded and examined. Actual file was downloaded from Kaggle.com and was obtained from BGG by user “phizzuela” using script from “https://github.com/mrpantherson/bgg_pull” which uses the API provided by the BGG website. The initial data frame had 23 columns, some of which were determined not to be useful for the analysis and removed. These were columns that link to the original BGG website page, images, ranks, and min/max time to play metrics along with a column that shows if a game was a reimplementation of an earlier game. As part of clearing, empty values in categorical features were filled with stock values (e.g. “unknown”). Examination of the data showed that a large number of the games listed had very few votes and/or a zero rating. The rows with a zero rating also had zero votes. These were discarded. Of the remaining games, the 75th percentile of number of votes was still 41, but the maximum was 93524. There are a lot of games in the database that are irrelevant because no one has bother voting for them. Fifty percent (50%) of the rows have 8 or fewer votes.

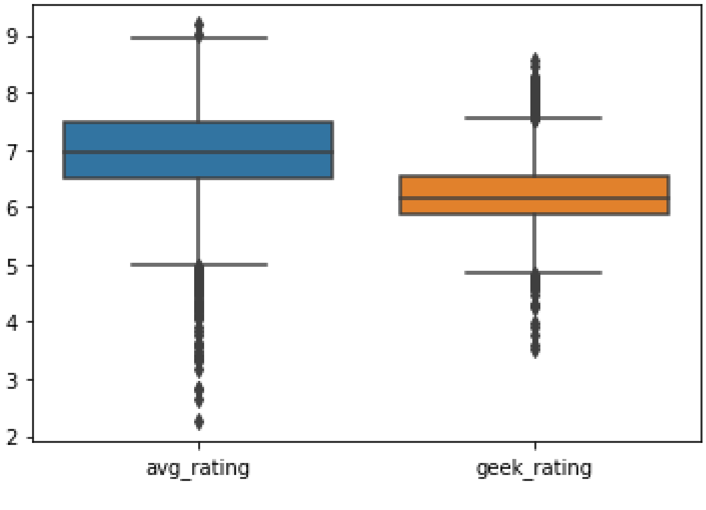


In looking at the top 25% of the Number of Votes column, the 75th percentile was still 408 votes, 500 votes was arbitrarily chosen as a level of votes that indicated a game that was well enough reviewed to have a reliable user rating. This left 4768 games in the list to analyse.

This filter left a reasonably large sample with a good distribution of ratings. The rating scale is 1 to 10.

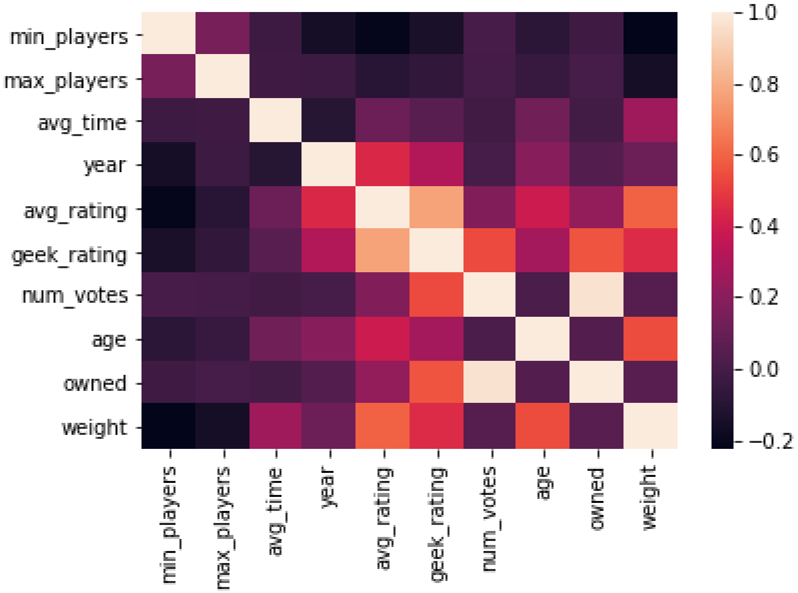


Now that there is a good sample to analyse, the existing numerical features were examined. There are two ratings in the database. One, the “geek rating” is a Bayesian average of the other, so they are similar for well reviewed games. Only the “user average” will be used as a dependent variable.



There were a few games where some values were not reported. Because of the relatively large sample size, these games were dropped from the analysis. Examination of the year variable shows a problem of outliers. The minimum value for the year the game was published is “-3500” but the 25th percentile was 2004. Different cutoff points were examined, and it was decided to limit the analysis to only games published since 1960. This still left a sample size of 4630 games to examine.

A simple heat map was created to examine the variables for correlations.



This graphic shows that the number of users who own a particular game is correlate to the number of votes that game received. This makes sense. People vote on things they own and have played. Similarly, number of owners is correlated with rating. People who buy a game tend to rate it and rate it higher. This backs up the hypothesis that User Rating is a good proxy for financial success of a game. Furthermore, the perceived "weight" of a game (how complicated or subtle its strategies are) is correlated with the minimum recommended age. This also makes sense because kids games are less complicated.

Next the categorical columns were examined. This presented a problem. The categorical variable columns "category" and "mechanic" contained more than one item per game. These two links list the board game categories and mechanics from the BGG web site. The Category and Mechanic column are concatenations of multiples of these.

https://boardgamegeek.com/browse/boardgamecategory

https://boardgamegeek.com/browse/boardgamemechanic

To determine if one or more of these attributes contributed to a game's success, they needed to be split out. The strings in these two columns were split, but only the first 4 categories and 5 mechanics were kept for each row. Then the values of each item were counted and divided by the total to create a frequency encoded average variable for each game. This value was used in the final model. The dataset used to create the models contained the following features:

Minimum number of players
Maximum number of players
Average play time
Year of release
Average user rating

Recommended age
Weight
Category Average
Mechanic Average

Data were split with 25% being held out for testing. The training data were scaled using standardized scaling. The test set was then transformed using the same scaler. Four different regression models were employed. First a simple linear model was run as a baseline. Results are listed below. Second, a Lasso regression was performed. A cross validated grid search ("GridSearchCV") was used and a stock cross validation function ("LassoCV") to find the best alpha value for the Lasso regression. Both methods yielded an alpha value of 0.006. Third, a Grid Search was used with a Random Forest model. This showed the best number of estimators for the model to be 7. Finally, a Ridge regression was performed (using "RidgeCV") with different alphas, and an alpha of 1 was chosen. Of all of these, the Lasso Regression had the highest R-squared value and the lowest Mean Squared Error (MSE).

	Linear Model	Random Forest	Lasso	Ridge
R-squared	0.497	0.482	**0.498**	0.497
MSE	0.239	0.246	**0.238**	0.239

Conclusions:

The output of the Lasso regression showed that the "weight" of the game and the year of its release had the most significant effect on the User Rating.

Lasso Regression Coefficients

Feature	Coefficient
weight	0.359340
year	0.269700
Average Play Time	0.016254
Recommended Age	0.013141
Category (average frequency encoding)	0.009994
Maximum number of players	-0.006575
Minimum number of players	-0.029338
Mechanic (average frequency encoding)	-0.062645

The output of the Random Forest regression tells a similar story. The “weight” and “year” features are an order of magnitude more important than any of the other features.

Random Forest Feature Importances

Feature	Importance
weight	0.433587
year	0.244653
Mechanic (average frequency encoding)	0.089021
Category (average frequency encoding)	0.083413
Average Play Time	0.050433
Maximum number of players	0.044649
Recommended Age	0.033267
Minimum number of players	0.020978

So the conclusion of the analysis is thus, there is no magic bullet. Game designers cannot put together a perfect mix of mechanics and play time in the right category to have a guaranteed winner. Creativity still counts! Because BoardGameGeek is primarily a site for board game fans, it’s not a surprise that this audience would prefer weightier games. To make a successful game, the [fictitious] board game company should focus on marketing their first game as new and exciting, highlighting things like its strategic depth (“weightiness”).

Further study

Some further information could be determined by slicing the various games by individual features.Why is mechanic negatively correlated with popularity? Are there some mechanics that people really don’t like? Average play time may be correlated with “weight.” Should those two features be examined as confounding variables? Finally, since “weight” is such a large factor in the popularity, what makes up “weight?” Another regression could be performed with just the mechanics to see if any of them are “weightier.”