Data Analysis for TMDB Top 5000 Movies

Chenyuan Li and Anton Bauer

Python for Data Management and Analytics

Dmitriy Babichenko

Dec 10, 2018

Data Analysis for TMDB Top 5000 Movies

Abstract

Using the Kaggle dataset on TMDB Top 5000 Movies, we decided to research what factors largely contribute to box office success of films. We both found movies to be an interesting topic to research about, and it seems very insightful to analyze what variables, such as genre, lead actors, director, budget, runtime, etc. lead to a box office hit. Our plan is to first analyze which categories most of the top movies fell under (high budget, top name directors, etc.) and then train a model based on several of the aforementioned variables to predict what makes a random movie both fiscally and democratically successful.

Introduction

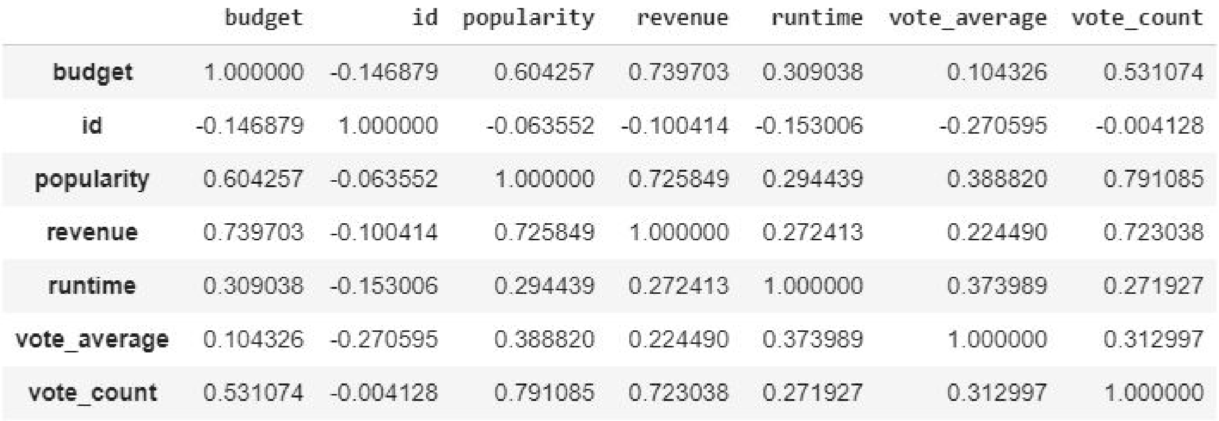
Everyone loves watching movies, but movies selection can be a very troublesome process for some people. We sometimes cannot really decide which movie to watch. But, most people often want to check out the most popular movies. So for our project, we decided to focus on this research question: what factors in a movie are accounted for a movie's popularity? We also hope this analysis can give some insight for movie makers as to what factors they can implement into their films to maximize potential viewership.

Methodology

Going in this project, we had a decent idea of which parts of the dataset would be most beneficial to our analysis. Logically, highly variable numeric columns, like budget, runtime, and vote count, are generally much more reliable predictors of our target variable, popularity, than discrete columns, like genre or language. Since those discrete columns have so many entries in each discrete value, there's bound to be a large variation in outcomes from them and, thus, makes it very hard to predict the outcome just given that discrete information. For example, just knowing a movie is sci-fi gives no indication of its popularity, as probably thousands of sci-fi movies were made that never caught any attention, while many also turned out to be massive hits. We also quickly discovered that this dataset is not particularly conducive to quick and easy analysis and training using most of the columns. All of the non-numeric columns are inputted in a database-style format, as an array of an id and a name. For example, for genre, rather than just listing one as "Action", it lists it as {"id": 28, "name": "Action"}. This made it rather difficult to use these for analysis and model training, as you can't just throw those columns into a model and expect it to understand what to do with them. Another big problem we discovered was that a large number of rows in several columns had 0's as their values for seemingly no reason. For example, we figured out that 1,100 rows in the budget column were 0's, which is a whopping quarter of the total rows. This put us in a real dilemma, as removing them would take out a large chunk of our data, but assigning them to other values would also skew and misrepresent the dataset as whole. Ultimately, we decided to convert them to null's instead, so that we could still use the other columns for those rows for other analysis purposes. We also normalized all continuous variables to reduce the impact of outliers and create somewhat normal distributions of the data. And, with that, we were able to start visualizing and training models on the data to finally start answering our initial research question.

Results

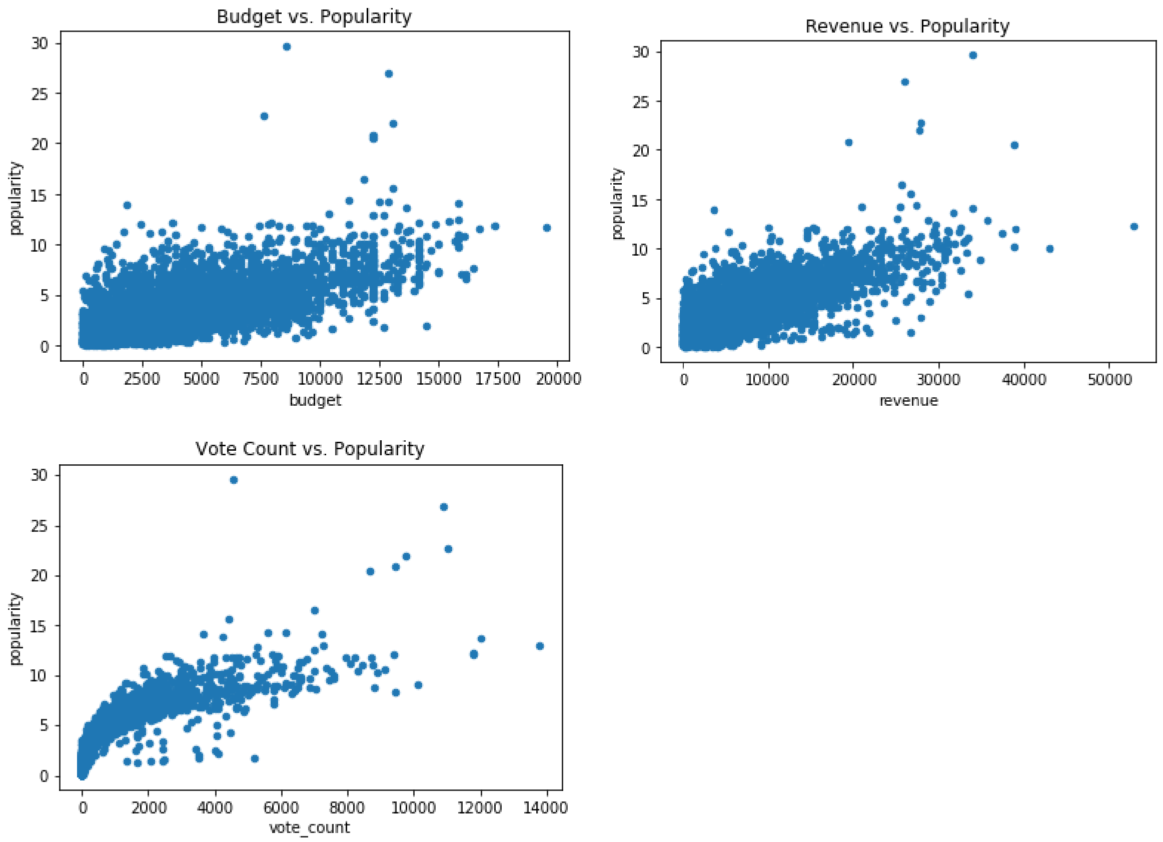
Our first step in answering our research question was to get an idea of how our variables are related to one another. We found the most effective way to do this was to look at a correlation table to find the highest correlation coefficients between variables, then visualize them through a scatter plot. The aforementioned correlation table can be seen below.



	budget	id	popularity	revenue	runtime	vote_average	vote_count
budget	1.000000	-0.146879	0.604257	0.739703	0.309038	0.104326	0.531074
id	-0.146879	1.000000	-0.063552	-0.100414	-0.153006	-0.270595	-0.004128
popularity	0.604257	-0.063552	1.000000	0.725849	0.294439	0.388820	0.791085
revenue	0.739703	-0.100414	0.725849	1.000000	0.272413	0.224490	0.723038
runtime	0.309038	-0.153006	0.294439	0.272413	1.000000	0.373989	0.271927
vote_average	0.104326	-0.270595	0.388820	0.224490	0.373989	1.000000	0.312997
vote_count	0.531074	-0.004128	0.791085	0.723038	0.271927	0.312997	1.000000

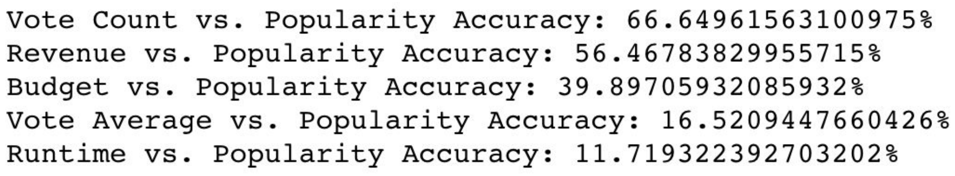
Looking along the popularity column (since it's our response variable), you can see the variables with the highest correlations to it are budget, revenue, and vote_count.

We then visualized these relationships with scatter plots, which can also be seen below.



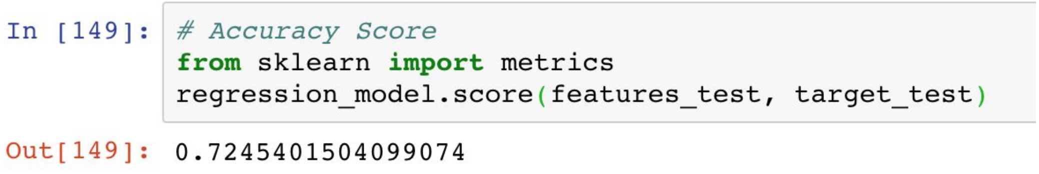
From these visualizations, you can see both variables have a fairly weak positive correlation to popularity, but still show some indication of a pattern, showing higher budget and revenues are generally conducive to a higher popularity of a film. Vote count has a much stronger correlation to popularity, which makes perfect sense, as more popular movies would have more people rating them.

Knowing all this information, when then began training models on this data. We started with simple regression models, directly relating one variable to popularity. We did this with five different variables, budget, revenue, runtime, vote count, and vote average, and recorded the accuracy of these models. Unsurprisingly, we found that budget, revenue, and vote count got the highest accuracies scores, shown in the image below.



```
Vote Count vs. Popularity Accuracy: 66.64961563100975%
Revenue vs. Popularity Accuracy: 56.46783829955715%
Budget vs. Popularity Accuracy: 39.89705932085932%
Vote Average vs. Popularity Accuracy: 16.5209447660426%
Runtime vs. Popularity Accuracy: 11.719322392703202%
```

But we thought we could do better than that, so we resorted to using a PCA model, as well. While this type of model doesn't tell us exactly which variables were most indicative of popularity, since it groups them into components, that wasn't too big of an issue, as we already figured out which those were, and ultimately the higher success rate of the model made up for this. You can see from the snippet below that we were able to achieve a 72% accuracy rate of a movie's popularity, where we used the same top 5 variables mentioned above as our input columns for the PCA transformation. Considering how many variables are at play in the success of a movie and how volatile the movie industry, this score is fairly impressive.



```
In [149]: # Accuracy Score
          from sklearn import metrics
          regression_model.score(features_test, target_test)

Out[149]: 0.7245401504099074
```

Discussion

So can we just say that you just need a ton of money with big names to create a hit movie? Well, while those certainly will help and high budget movies are positively correlated with popularity, those are not necessarily the only factors. Longer movies may captivate viewers and tend to have a more niche, but dedicated fanbase, however they may also turn away many skeptical viewers. Using more popular genres can intrigue audiences interested in that genre, but obviously the quality of that film plays a much bigger role. Well, in an extreme case, if a customer really wants to know the top variable which indicates a movie's success, *vote count* stands out based on our accuracy score test with 66.6% accuracy score. Obviously, a popular movie will have a large amount of votes, regardless of whether the votes are positive or negative. The vote count will directly reflect how many people saw the movie, and thus, it would be a reasonable indicator that reflects the popularity of a movie. But all in all, like what previously stated, the quality of a film plays a much bigger role because in the end, a movie with good quality would be heavily invested, which means a high budget, leading to a large number of vote count, and ultimately generating a good revenue for the production crew.

As for each group member's contribution to this project, the group quickly came to an agreement and was able to agree, and if not, propose better solutions to obstacles during different stages of our project. All decisions about project was discussed before they are decided and the workload was split reasonably. Finally, we are very pleased and honored to work with each other. It was a pleasant and smooth collaboration.

References

The Movie Database (TMDb): TMDB 5000 Movie Dataset, 2017 [Data set].

https://www.kaggle.com/tmdb/tmdb-movie-metadata