Project Luther: Analyzing and Predicting Movie Ratings on IMDb

Project goal. The goal of my project is to predict a movie's rating based on a set of features. The data set was scraped from IMDb.com and contained nearly fifty thousand of US feature film releases between 1990 and 2017, with only 4455 of them being valid for modeling.

Data Sets. Although I scraped a variety of features, I removed some of them because I thought they would have no effect on my predictions. For example, I dropped release year and movie title. In hindsight, I could have kept release years for feature engineering. Since I collected nominal values for total domestic gross over the past twenty seven years, I introduced an inflation error into my data, and perhaps the years could be used to do adjust domestic gross values to get more accurate results.

I split my data into four sets: 1. a set with only numerical data (runtime, budget, total gross, number of IMDb votes) as my baseline set; 2. a set which added genres and MPAA ratings; 3. another set which added directors and languages on top of the previous set; 4. the last set included everything (previous set data with countries, actors, and writers). The splitting was based on my intuition rather than intrinsic data science knowledge.

Models. For my models, I chose to perform a regular linear regression, LASSO regularization, Ridge regularization on all sets, and polynomial regression on the first two sets (numeric and numeric with genres and MPAA ratings; I did not perform polynomial regression on the other two sets because they contained sparse data and 0-1 values and took a lot of computational power).

Once again, I did not have data science intuition to select the models, and I simply ran my data sets through all of them and compared the results. However, I thought that adding complexity to the very first set of four features was a necessary due to its simplicity, and I did not think that adding complexity to the other three data sets would be a good idea, since the number of features increased significantly to a thousand, and I had only 4455 data points.

Regularization on more complex data sets proved to be more useful because the coefficients showed which features were more influential than others. However, because there were many coefficients, and all of them were small, I did not know where to make the cut and which coefficients to drop. I did not perform regularization on my polynomial models.

Results. When running models on more complex data sets, I found that my metrics (r-squared and RMSE) were improving very marginally, while the models were dramatically increasing in complexity. Hence I selected my first two data sets, one of which was purely numerical, and the other with a few categorical values for genres and MPAA ratings. Again, I did not have a good intuition for where to make the cut, so I looked at RMSE and selected the data set with polynomial regression, since the numbers seemed "good enough," and the model was simple.