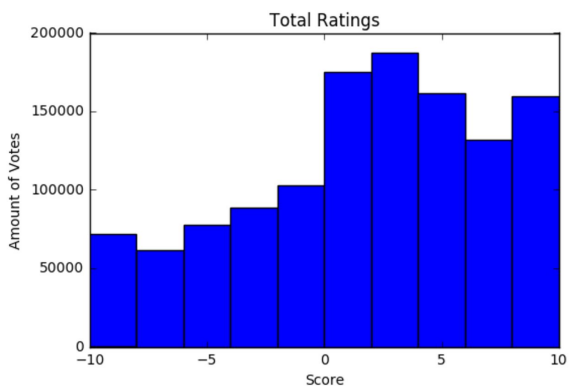
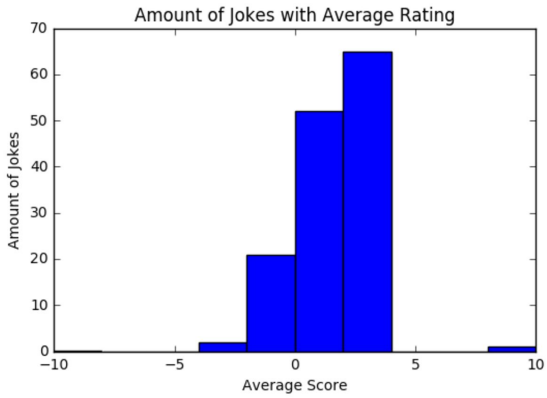
Case Study: Recommender

Date: 3/7/2017
Team: John Zhu, Nicholas Halpern, Jonathan Ng, Megan O'Rorke, Alexander He

Analysis





From the Total Ratings graph, we noticed there were more positive votes and a left skewed histogram. After looking at the Average Rating graph of each joke, we notice a similar leftward skew, however most jokes fall near the average rating of 1.7 rating.

Parameter Tuning and Other Improvements

We used grid search for parameter tuning. Parameter optimization with 60 features improved RMSE from ~3 to .9, but the test RMSE was very high (5). When we tried optimizing 3 features our train RMSE was 3.9 and test was 4.3. Standard deviation of ratings if 5.3 so we made a marginal improvement. Other improvements include assigning a rating of -10 to any value below -10, and above 10 to -10 and 10.

Choices We Made

In an effort to increase model performance we tried incorporating metadata (item content data). We hypothesized joke format: question vs answer, how many breaks, and joke length could impact rating. We noticed a theme of Clinton jokes, and tried that as a categorical feature. We rounded values above 10 and below -10 to ensure our model predictions are between -10 and 10, since a user can only rate a joke -10 to 10. We tried TF-IDF to uncover topics significantly contributing to joke rating. And engineered a feature that captured most variance across jokes.

What We Learned

Features we engineered had little or no effect. The biggest learning for us was to not get seduced by including a ton of parameters in grid search.

Github: https://github.com/souljourner/case-study-recommender