Predicting Full Funding for Donors Choose Projects

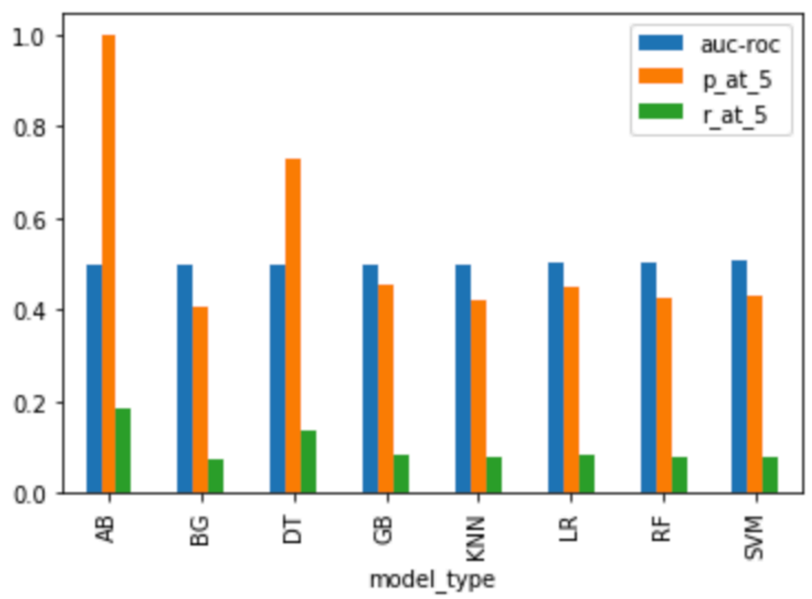
Given the high cost of school supplies, and decreasing funding in public schools across the US, many teachers struggle with the day-to-day challenge of providing basic resources to their students. An organization known as DonorsChoose connects teachers in high-need communities with donors who want to help. In order for this organization to be effective, it is helpful to know what factors increase the likelihood that a project receives funding. In order to investigate this issue, we ran various models to predict the likelihood that a project was funded within 60 days of it being posted. The purpose of the analysis was to classify observations as either being fully funded in 60 days, or not being fully funded in 60 days. Running 196 models over 9 different model types, we compared the model predictions to the actual outcome. Using various evaluation metrics, we compared the performance of the models. The metric we used to find the best model was the AUC-ROC metric, which is the likelihood that a positive result is correctly identified as being positive, and likelihood that a negative result is correctly identified as being negative. Based on our results, we recommend that DonorsChoose select the Logistic Regression classifier model to predict the likelihood of a project being fully funded in 60 days.

Model and Evaluation Details

In order to suggest an appropriate model to predict successfully funded projects, we had to run and compare the results of various models. For this particular analysis, we ran the following models: K-Nearest Neighbors, RandomForest, ADA Boosting, Support Vector Machines, Logistic Regression, Decision Trees, Gradient Boosting, and Bagging. We trained the models using a rolling 6-month window, which resulted in 3 different timeframes, and 3 training sets and 3 testing sets, respectively. We varied the parameters to find the model with the optimal parameters for prediction. Overall, none of the models performed particularly well it came to their AUC-ROC measure. All the models had an AUC that hovered at around 50%, and even the best model (measured by having the highest AUC) only had an AUC of 51.3%. This is still significantly better than the 26% accuracy of the baseline model, which measures the accuracy of a random prediction. Basically, this suggests that the models trained do a better job predicting whether a project would receive full funding within 60 days than a random prediction would do.

The figure below shows the relative performance of the models across a few metrics, including AUC, precision, and recall. The graph shows metrics measured for the bottom 5% of projects (those least likely to get funded). Precision tells us the percent of projects classified as not having been fully funded that were actually not fully funded. Recall tells us the percent of projects that were not fully funded in 60 days that the model correctly classified. As the figure shows, all the models do significantly better when it comes to precision when compared to recall. Typically, whether the organization is more interested in optimizing precision or recall is dependent on the resources that an organization has to serve people, and the target group that they are hoping to serve. In this particular case, if we are interested in identifying the 5% of projects that are at highest risk of not getting fully funded, we would be interested in precision at 5%, because with limited resources, the organization would likely want to identify and target the projects in most need of help, and divert time away from projects in less need of help to achieve funding.

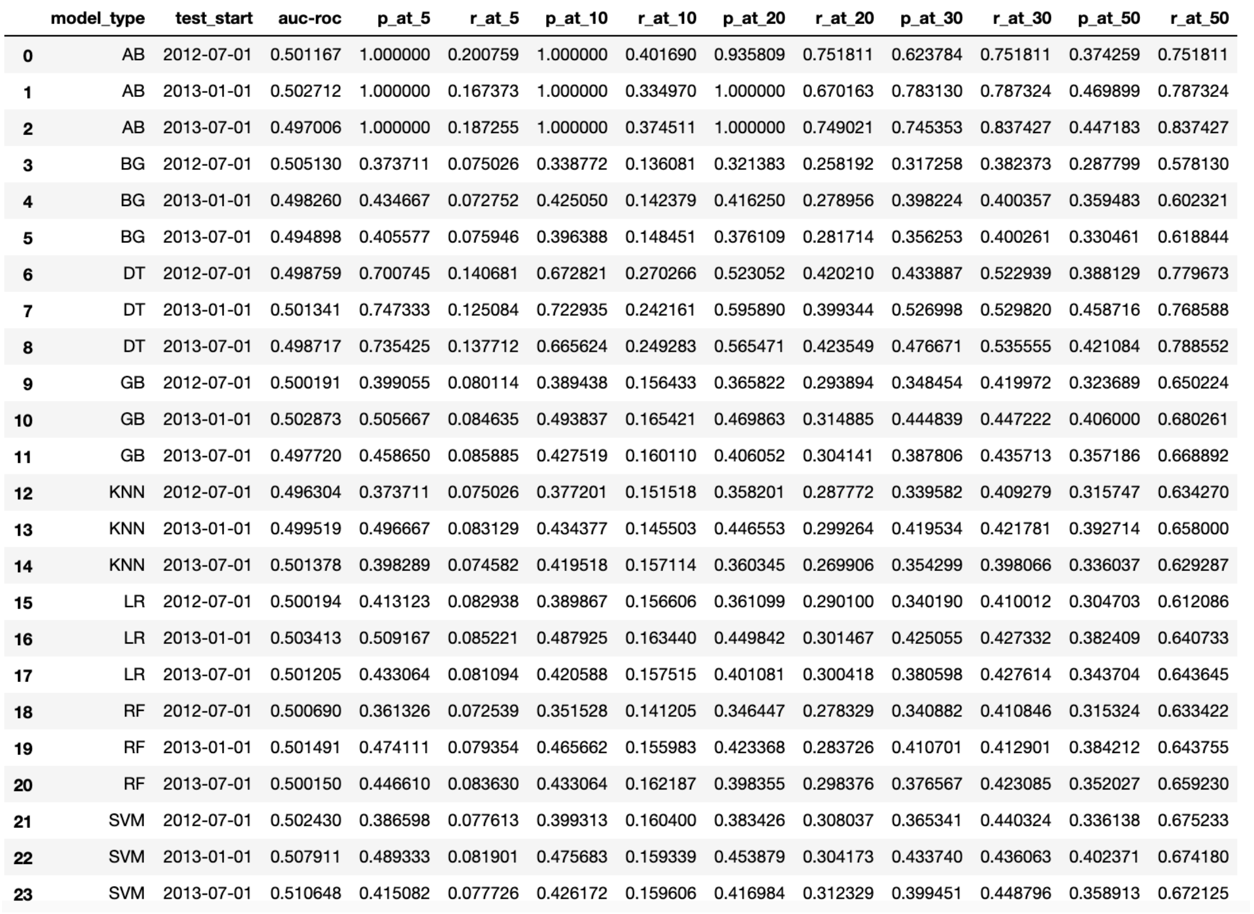
Figure 1: AUC, Precision, Recall at Bottom 5%



The boosting model appears to perform the best when it comes to precision, and the K nearest neighbor model appears to perform the worst when it comes to precision (and the other two metrics as well). The relative slump in performance for the K-Nearest Neighbor model could be due to the large number of features included in the model. The KNN model is typically sensitive to including inconsequential features. It is possible that pruning features could result in better performance of the KNN model. Given the poor overall metrics, it is likely that the organization would need to further analyze the data and tweak the models to more accurately predict the success/failure of funding.

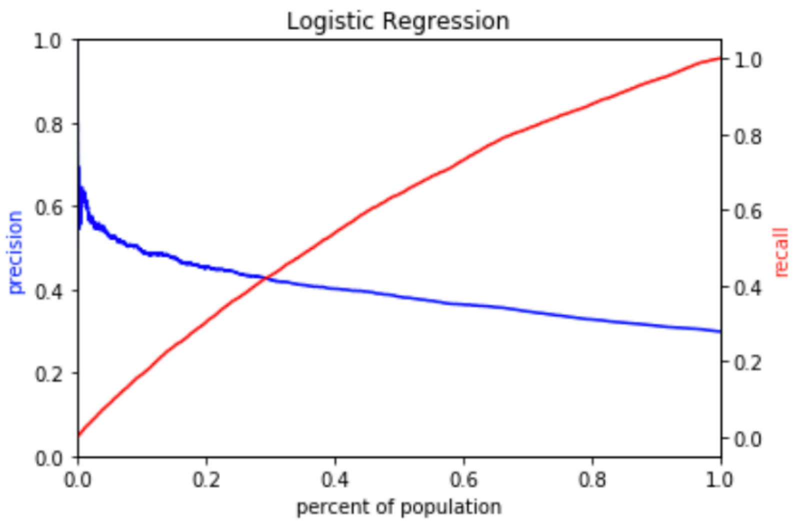
The full tables (in the appendix) show precision and recall for the bottom 1%, 2%, 5%, 10%, 20%, 30%, and 50% of the projects. An abridged table below shows the trends across models over time. Precision is particularly high for boosting and decision trees. In general, the models seem to show higher precision for the lower percent of percent of the population, and higher recall for the higher percent of the population.

Figure 2: AUC, Precision, Recall Over Time



	model_type	test_start	auc-roc	p_at_5	r_at_5	p_at_10	r_at_10	p_at_20	r_at_20	p_at_30	r_at_30	p_at_50	r_at_50
0	AB	2012-07-01	0.501167	1.000000	0.200759	1.000000	0.401690	0.935809	0.751811	0.623784	0.751811	0.374259	0.751811
1	AB	2013-01-01	0.502712	1.000000	0.167373	1.000000	0.334970	1.000000	0.670163	0.783130	0.787324	0.469899	0.787324
2	AB	2013-07-01	0.497006	1.000000	0.187255	1.000000	0.374511	1.000000	0.749021	0.745353	0.837427	0.447183	0.837427
3	BG	2012-07-01	0.505130	0.373711	0.075026	0.338772	0.136081	0.321383	0.258192	0.317258	0.382373	0.287799	0.578130
4	BG	2013-01-01	0.498260	0.434667	0.072752	0.425050	0.142379	0.416250	0.278956	0.398224	0.400357	0.359483	0.602321
5	BG	2013-07-01	0.494898	0.405577	0.075946	0.396388	0.148451	0.376109	0.281714	0.356253	0.400261	0.330461	0.618844
6	DT	2012-07-01	0.498759	0.700745	0.140681	0.672821	0.270266	0.523052	0.420210	0.433887	0.522939	0.388129	0.779673
7	DT	2013-01-01	0.501341	0.747333	0.125084	0.722935	0.242161	0.595890	0.399344	0.526998	0.529820	0.458716	0.768588
8	DT	2013-07-01	0.498717	0.735425	0.137712	0.665624	0.249283	0.565471	0.423549	0.476671	0.535555	0.421084	0.788552
9	GB	2012-07-01	0.500191	0.399055	0.080114	0.389438	0.156433	0.365822	0.293894	0.348454	0.419972	0.323689	0.650224
10	GB	2013-01-01	0.502873	0.505667	0.084635	0.493837	0.165421	0.469863	0.314885	0.444839	0.447222	0.406000	0.680261
11	GB	2013-07-01	0.497720	0.458650	0.085885	0.427519	0.160110	0.406052	0.304141	0.387806	0.435713	0.357186	0.668892
12	KNN	2012-07-01	0.496304	0.373711	0.075026	0.377201	0.151518	0.358201	0.287772	0.339582	0.409279	0.315747	0.634270
13	KNN	2013-01-01	0.499519	0.496667	0.083129	0.434377	0.145503	0.446553	0.299264	0.419534	0.421781	0.392714	0.658000
14	KNN	2013-07-01	0.501378	0.398289	0.074582	0.419518	0.157114	0.360345	0.269906	0.354299	0.398066	0.336037	0.629287
15	LR	2012-07-01	0.500194	0.413123	0.082938	0.389867	0.156606	0.361099	0.290100	0.340190	0.410012	0.304703	0.612086
16	LR	2013-01-01	0.503413	0.509167	0.085221	0.487925	0.163440	0.449842	0.301467	0.425055	0.427332	0.382409	0.640733
17	LR	2013-07-01	0.501205	0.433064	0.081094	0.420588	0.157515	0.401081	0.300418	0.380598	0.427614	0.343704	0.643645
18	RF	2012-07-01	0.500690	0.361326	0.072539	0.351528	0.141205	0.346447	0.278329	0.340882	0.410846	0.315324	0.633422
19	RF	2013-01-01	0.501491	0.474111	0.079354	0.465662	0.155983	0.423368	0.283726	0.410701	0.412901	0.384212	0.643755
20	RF	2013-07-01	0.500150	0.446610	0.083630	0.433064	0.162187	0.398355	0.298376	0.376567	0.423085	0.352027	0.659230
21	SVM	2012-07-01	0.502430	0.386598	0.077613	0.399313	0.160400	0.383426	0.308037	0.365341	0.440324	0.336138	0.675233
22	SVM	2013-01-01	0.507911	0.489333	0.081901	0.475683	0.159339	0.453879	0.304173	0.433740	0.436063	0.402371	0.674180
23	SVM	2013-07-01	0.510648	0.415082	0.077726	0.426172	0.159606	0.416984	0.312329	0.399451	0.448796	0.358913	0.672125

In other words, recall is better for the bottom 50% of projects compared to the bottom 5% of projects. There do not appear to be any noticeable differences in performance over time. The precision-recall curve for the best performing model, the Logistic regression model with C of 0.1, and penalty 'l1' is shown below.


Logistic Regression
precision
recall
percent of population