Quinn Underriner
Policy Memo, Machine Learning for Public Policy HW3

DonorsChoose.org

Introduction

DonorsChoose is a US based nonprofit that allows teachers the ability to directly appeal to the public for donations for their public school classroom needs. The United States has high variance in how much it spends on public education as in many communities this funding is based on property taxes.[1] These leads to many severely underfunded communities. Further, the amount that the United States invests in education is declining.[2] In the absence of increased funding, better targeting of these types of charity will be more crucial to the educational system. Currently, 71 percent of projects posted on DonorsChoose get funding within 60 days. This analysis intends to predict which projects are not likely to get funded. With this analysis, policy makers can target interventions with projects that are not likely to get funded to help ensure they do. Further, having identified what are the attributes of good project proposals, DonorsChoose can provide great guidance to teachers before they put sites together, potentially saving them time on preparing their pleas (leaving them more time for teaching).

Result From Training the Model:

After applying 7 different models with a myriad of potential parameters, a decision tree with a max depth of 20, has the best f1 score, which means that of the models it did the best job of minimizing false positives and false negatives. This model has a high precision rate of 73 percent (at a threshold of 0.5), which compared to the baseline of 29 percent (the current percentage) means that this model is 250% percent better at identify projects that might fail than a random test. If DonorsChoose wants to do costly interventions it would want to focus on having a high rate of precision so that they know their resources are being used efficiently Notably, the random forest method has very similar scores, and given the number of features (24, including, for example, if the school is a charter or the focus area of the teacher) I might suggest that DonorsChoose use a random forest model. It shows less variability over different time periods than the decision tree.

---

[1] Bruce J. Biddle and David C. Berliner, "A Research Synthesis / Unequal School Funding in the United States" http://www.ascd.org/publications/educational-leadership/may02/vol59/num08/Unequal-School-Funding-in-the-United-States.aspx

[2] Jill Barshay, "While the rest of the world invests more in education, the U.S. spends less", https://hechingerreport.org/rest-world-invests-education-u-s-spends-less/

Recommendations for DonorsChoose

Note: These results are notably high and while they can be useful to guide policy, they should be treated with a healthy skepticism.

However, Donor's Choose could reach out to those who the model identified as likely not having their project be funded within 60 days and offer them support in advertising their project.
The model indicated certain features were related to a higher likelihood of being chosen. For example, having a project be eligible for a match was related with higher funding. Donor's Choose could recommend teachers more proactively try and seek these matching funds to boost overall funding.

Top evaluation results, sorted by f1_scores:

model	parameters	f1_score	auc	precision_at_0.5	recall_at_0.5
decision_tree	{'max_depth': 20}	0.545291	0.735545	0.741371	0.732040
bagging	{'n_estimators': 2}	0.543900	0.711505	0.742334	0.685972
random_forest	{'n_estimators': 1}	0.543246	0.732626	0.740327	0.726616
bagging	{'n_estimators': 20}	0.542575	0.767744	0.737493	0.802282
random_forest	{'n_estimators': 50}	0.542461	0.777961	0.736741	0.825577
bagging	{'n_estimators': 10}	0.542415	0.762648	0.737778	0.791192
random_forest	{'n_estimators': 25}	0.541519	0.774476	0.736422	0.817991
	{'n_estimators': 10}	0.540631	0.769060	0.736179	0.806107
logistic_regression	{'C': 10}	0.529895	0.824491	0.728237	0.952079
	{'C': 1}	0.529608	0.824747	0.728099	0.952950
	{'C': 0.1}	0.527655	0.826136	0.727174	0.958110
decision_tree	{'max_depth': 5}	0.526763	0.819946	0.727110	0.944480
	{'max_depth': 8}	0.526611	0.819675	0.726935	0.941844
boosting	{'n_estimators': 50}	0.521058	0.828013	0.724194	0.968303
	{'n_estimators': 100}	0.520604	0.828232	0.723993	0.969238
	{'n_estimators': 30}	0.519701	0.828618	0.723591	0.970933
logistic_regression	{'C': 0.01}	0.519186	0.828112	0.723411	0.969820
	{'C': 0.001}	0.503149	0.832748	0.716616	0.994514