Data Exploration

Before building models to classify whether or not a project would fail to get full funding within sixty days, I began by exploring the data to gain an understanding of the different information available and which variables to include in the classifiers. After this process, I decided to use nineteen predictor variables to generate features for each of my classifiers. These predictors describe the characteristics of the projects and the students, classrooms, and schools that these projects would benefit. Through this initial analysis, it was clear that the number of days it took for a project to get funded did vary along with many of these variables, however, simple correlation analysis did not reveal any exceedingly strong links between the potential predictor variables and the number of days it took a project to reach full funding. Not all of the the nineteen predictor variables that I included in my classifiers were included directly. For example, in an attempt to reduce overfitting, I converted the column containing teacher's account ID to represent the number of times a particular teacher account ID appeared in the data, reasoning that teachers who have submitted projects in the past may improve at writing project proposals and that this information was more important and more applicable to unseen instances than the random ID assigned to each teacher's account. I applied similar reasoning to the schoolid, school_district, and school_county variables, reasoning that teachers in the same geographic areas may share their experience with the platform. Ultimately, I also decided to drop a some provided variable, namely schools' longitude and latitude coordinates. I believe that information conveyed by a schools geographic coordinates is better captured by other variables in the dataset, such as a schools metropolitan surrounds, county, city, state, or ID and that models would struggle to interpret geographic coordinates correctly. I also dropped NCES ID because that column was missing for many schools and, based on external reading, seemed not to provide any additional information beyond schoolid.

Classifiers

This project includes twelve classifiers and two baselines to compare the other classifiers against. The twelve classifiers are: (1) two decision trees, (2) two logistic regressions, (3) two support vector machines models, (4) two random forests, (5) two boosting models, (6) one bagging model, and (7) one k-nearest neighbors model. The first baseline is a short decision tree with a maximum depth of three levels, and the baseline simply considers whether the majority of projects in the data used to build a model did or did not receive full funding within sixty days and then predicts that value for every observation. In effect, these baseline help assess whether any of the six classifiers are better than a very simple model or a (well-informed) guess.

Evaluation Methodology

To evaluate the classifiers built for this project, I simulate the actual problem that we hope to solve by splitting up our full dataset by time. In particular, I split the dataset up into six month intervals based on the time that a project was posted. I then select one of these six month periods of data to use as a "hold-out," meaning it is not used to build the models and instead only use data from before the selected interval to build the model. To evaluate the built models, they are each used to predict whether or not the projects in our "hold-out" will get fully funded within sixty days and these predictions are compared to the ground truth about whether projects actually got full funding within sixty days.

Each model is evaluated on a number of metrics at different thresholds. When one of the trained classifiers is applied to a set of data, it generates a score assessing how likely it is that a given project will fail to receive full funding within sixty days of being posted. The threshold then gives the standard to differentiate between projects that are and are not predicted to fail to gain full funding within 60 days. For example, if the threshold is 5% then observations within the top fifth percentile of the generate score will be predicted not to gain full funding within sixty days and all others will be predicted to gain full funding with sixty days.

Using these predictions, the program generates five evaluation metrics at a variety of threshold. The first metric is accuracy, which compares the total number of projects that a model correctly predicted the funding status of to the total number of predictions the model made. The second is precision, which represents the proportion of projects predicted not to receive full funding within sixty days that actually did not receive full funding within sixty days. Recall is the third metric, and it conveys the proportion of projects that actually did not receive full funding within sixty days that the model accurately predicts as failing to receive funding within sixty days. The program also produces a graph visualizing the tradeoff between precision and recall for each model. The fourth metric, F1, acts as a weighted average of the previous two, precision and recall. The final metric is area under the receiver operator curve which does not depend on the threshold applied and gives the probability that a model will score a randomly selected project that did not receive full funding within sixty days higher than a randomly selected model that did receive full funding within sixty days.

Comparing Classifiers

Compared to the baselines, all of the classifiers in this model did a poor job on the accuracy metric. The short decision tree baseline, which represents a very simple model, achieved accuracy scores between 0.56 and 0.74 across different thresholds. The naive baseline that predicts that every project will receive full funding within sixty days will achieve accuracy ratings between 0.69 and 0.74 across the three training sets, meaning that between 69% and 74% of the predictions it made were correct. Among the classifiers developed in this project, multiple classifiers reach an accuracy rating of 0.74 or slightly higher for one or more of the training sets. However, there is still very little benefit to using these other models from this project even if accuracy is the most important metric due to the extra resources required to develop and maintain them versus the short decision tree or naive baseline.

The classifiers built in this project do outperform the baseline for other goals, however. Since the naive baseline predicts that all future projects would be fully funded within sixty days, it fails to correctly identify any projects that would not achieve full funding within sixty days, giving it a precision and a recall score of 0 across all training sets and thresholds. The short decision tree's maximum precision score is 0.42 when predictions are made for the third testing set at the 0.05 threshold. Overall, precision scores for this baseline model, however, fall between 0.33 and 0.42. The short decision tree's maximum recall score is 0.68, achieved for the third testing set at the 0.50 threshold. Multiple other machine learning models developed for this project are able to achieve precision scores over 0.50 depending on the threshold established, and can achieve recall scores that are consistently slightly above the short decision tree model. It should be noted, however, that the models developed for this purpose cannot simultaneously achieve high precision and recall scores at the same threshold. In general, as the threshold

increases, these models' precision falls while their recall rises, and as the threshold decreases, precision rises while recall falls.

If a program requires predictions where the projects predicted not to be fully funded within sixty days most often actually do fail to obtain full funding within sixty days, the boosting (AdaBoosting) classifier with twenty estimators developed in this project is most likely the best choice for predictions. Relative to the other models in this project, this classifier maintains consistently high precision scores across all the specified thresholds and training sets. This means that the projects it predicts not to be fully funded within sixty days actually do fail to be fully funded with sixty days at rates above the other classifiers in this project. Thus, one example of where this boosting model may be helpful is a resource-constrained initiative where the group supporting the intervention wants to ensure that their limited support goes to projects that are actually not likely to achieve full funding within sixty days.

For programs that require identifying a high proportion of all the projects that will fail to achieve full funding within sixty days are identified without regard for the proportion of those projects that actually do fail to achieve funding within sixty days (perhaps those with less severe resource constraints), the same boosting classifier as before generally outperforms the other models in this project by achieving generally higher recall scores across the three training datasets and a variety of thresholds. It should be noted, however, that this classifier, along with all the others developed in this project only slightly outperform the short decision tree baseline on the recall metric and that model may actually be preferable to implement as it is significantly less resource intensive and easier to understand.

Finally, if a balance of the two above goals is required, then the boosting model with twenty estimators is also the best choice, as it has consistently high F1 scores across all training sets and thresholds. F1 scores are a weighted average of recall and precision and thus help convey the extent to which a model identifies a high proportion of the projects that actually fail to achieve full funding within sixty days while simultaneously having a high proportion of the projects that it predicts will fail to receive full funding within sixty days actually fail to receive full funding within sixty days.

Change over Time/Datasets/Temporal Splits

For the later testing/training sets, all of the evaluation statistics tended to increase, though not by large amounts. The implications of this trend are twofold. First, it likely indicates that the models did benefit slightly from having larger training sets (the later training sets contain more data since they include all observations that would have been available at start of the associated testing set) and that these larger datasets allowed the models to more accurately detect patterns and in turn produce more accurate prediction. The fact that the increase tended to be so small, however, likely indicates that the additional data is only slightly improving the models' predictive abilities and that any further improvements from an increase in data size will require extremely large increases in dataset size.

Recommendations

Given the above evaluations of the generated classifiers, I recommend that a group hoping to identify and intervene with 5% of posted projects use an AdaBoosing classifier with twenty estimators to identify projects with which to intervene. This recommendation is based primarily on this classifier's performance on the precision metric with a threshold of 5%. Given the constraints, namely that the group will only intervene on 5% of posted projects, precision at the 5% threshold is the proper evaluative metric to use in this case because it conveys how many of the projects a model scored within the top 5% on their

likelihood of not receiving full funding within sixty days actually failed to receive funding within sixty days. Compared to all the other classifiers designed for this project, this boosting classifier had a strictly higher precision value at the 5% threshold across all three training sets.

Implications and Caveats

Based on the recommended classifier's precision metric at the 5% threshold, I expect that somewhere around 50% of the projects that this group decides to intervene with will be projects that actually would not have received full funding within 60 days. This expectation, however, may not hold if a structural change in the factors that predict which projects will and will not be funded within 60 days has occurred since 2012-2013, the years covered by this project's datasets. Some possible factors that could have driven such a change include changes to DonorsChoose's website design, changes to the project approval process, or differences in donors' philanthropic preferences.

Furthermore, this analysis does not suggest that any of the predictor variables have a causal relationship with whether or not a project receives full funding within sixty days. Answering that question requires work from more traditional statistics. Finally, identifying projects that are most likely not to receive full funding within sixty days and targeting them for intervention does not guarantee that the intervention will lead these projects to receive full funding within sixty days, and more work should be done to justify why the proposed intervention will help projects that are otherwise relatively unlikely to receive full funding within sixty days.