Assignment 3: Report

We were given two datasets namely the projects dataset and outcome dataset. Projects dataset has information a school’s geographical location, type of school, and other details, while the outcome dataset has information on if a project was fully funded, if the donor conversation was great, and other funding related details.

As a first step, I filtered the project dataset for information between 2011 and 2013 year. I then merged the two datasets and selected relevant variables for the analysis. This included a total of 33 variables. Out of these 33 variables I chose 17 independent variables for running the model. Fully\_funded was the dependent variable. These 17 independent variables were choses based on their relevance as well as if we had enough information on them i.e. we didn’t have a lot of missing values.

Before running the model, I first divided the data into training and test dataset. A training dataset is a data from which a machine learning model learns the pattern between the dependent and independent variables. The model uses these patterns to predict the outcome for the dependent variable in the test dataset.

For my analysis, I first looked at the first year i.e. 2011-01-01 to 2011-12-31 and divided this dataset equally, I established a naïve baseline for my model. Naïve baseline is basically a guess against which one can compare the machine learning model. If the machine learning model will not do better than the established naïve baseline then it is not relevant to use machine learning approach. For the naïve baseline, I used mean absolute error metrics which is a measure average absolute error on the predictions. The baseline performance on mean absolute error was .259

I ran a pipeline of Logistic Regression, SVM, KNN, Decision tree, random forest, boosting, and bagging classifier. In the first step, I compared all the classifier on a common metrics of mean absolute error to assess which models will be good for the predictions and if they are doing better than the baseline performance. Decision Tree, and Bagging classifier did the best among all the classifiers. After this I evaluated all the three classifiers on different performance metric. Decision tree performed better than bagging for accuracy, test\_recall, and F1 score, while bagging performed better for precision and roc\_auc score.

For the data set from 2011-01-01 to 2012-12-31, the Baseline Performance mean absolute error was 0.258. Decision tree and Bagging perform best, with mean absolute error of .08. Bagging performs better than Decision tree for all the performance metrics.

Given, these results I would recommend using Bagging classifier for this problem if we a large training dataset, while using decision tree if the training data is small.