CS 4780 Final Project Propsal

1. Team

This project will be completed by a team of 4 students.

2. Motivation

DonorsChoose is a crowd funding website that helps public school teachers request and receive funding. Almost 70 percent of campaigns on DonorsChoose are successfully funded. While this is a healthy margin, it could be vastly improved. It would be valuable for teachers to learn what factors may affect the success of their campaign.

Extensive research about Kickstarter and other similar crowd-funding sources has already been done to investigate causes of successful commercial campaigns. It would be interesting to compare and contrast factors that influence the success of a crowd-funding campaign in the commercial realm versus in the philanthropic realm.

3. Problem Statement

The main goal of this project is to help teachers determine what they can do to give their project the best chance of being funded. To do this we will determine what factors have the greatest influence on whether a project will be fully funded or not. in particular, we are interested in investigating whether characteristics of a project such as the location of the school, the poverty level, the grade level, or area of study (such as english versus chemistry) affect the likelihood of funding.

DonorChoose enables various promotions such as having a corporation match donations. We would like to investigate if these promotions affect the number or size of donation as well as if they affect the likelihood of a project being funded.

4. Approach

Using the ?fully funded? attribute as our label, we want to highlight which "features" are most important. We will use various linear classifiers such as SVM light, K-nearest-neighbor, Decision Trees, and others. We will experiment with different scaling of our data as well as omitting certain fields to achieve the best outcome for each model. As for identifying the most important features we will look at the weighting of the feature vectors in the optimal hyperplane as well as run some model selection tests where we omit certain features and observe the change in accuracy on our test set. We will run a full train-validate-test environment to try to find the most predictive model in the hopes of identifying the most important attributes of a new project.

5. Resources

We will use existing linear classification software provided publicly online, starting with SVM light and expanding to other software as necessary. File reading and other custom code written for this project is written in Scala. The full dataset for this project is provided publicly by Kaggle. It is available here:

https://www.kaggle.com/c/kdd-cup-2014-predicting-excitement-at-donors-choose/data.

The data consists of five files each with different information for each funding campaign: projects, essays, resources, donations and outcomes. The main focus of our project is projects.csv which gives the characteristics of each project that is requesting funding. These characteristics include who the teacher is, where the school is and how much is being requested.

We use the isFullyFunded feature in outcomes.csv as our label. Essays contains a short essay that teachers submit explaining what they are requesting and why. resources and donations are lists of resources requested and donations received for each project. In total the dataset includes over 600,000 projects and we will be learning using 100,000 randomly selected projects which should be an accurate representation of all of the data.

6. Progress

The raw data provided by Kaggle is in csv format, and in numeric, enum, boolean, and string formats. We created an OO model of the data in Scala and wrote file reading code to read data from the csv files. Lists of data can then be written to output files in any format as necessary, starting with data for SVMlight. To save time later, we implemented a set of data analysis tools, such as finding the mean or standard deviation of a given data index, separating a list of data by its label, and similar functionality.

In order to make the data manageable, we randomly selected ten sets with 10,000 instance to represent our data, for a total of 100,000 instances. This set is comparable in size to the original data set and certainly large enough to draw conclusions from, while being small enough to process in a timely manner. One important note about the data we found while processing it is that approximately 70% of the projects in the input are fully funded (thus confirming DonorChoose?s claim). Any classifier that uniformly predicts true (funded) will therefore achieve 70% accuracy.

We wrote the code to use svm light on our data. Initially when trained on svm light with default values all training sets produced the same classification of classify all samples as positive. By adjusting j values we were able to improve this accuracy. We were able to find the primal weight vectors for various sample training sets. We were already able to identify some trends from these preliminary results. We found that poverty level and if the school is rural or suburban to have a large impact. Currently we are not doing validation, but the code is written in modular fashion that will make validation easy to add in and that will be one of the next steps.

Another classifier we are using are decision trees. The ID3 algorithm with early termination is used for construction of trees. Very simple preliminary analysis shows that the optimal depth of a decision tree is around six to eight nodes deep, achieving prediction accuracy at around 72%. This isn?t notably better than the base predict-everything-as-positive accuracy of 70%, but approximately parallels the results of SVM results.

7. Moving Forward and Feasibility

We were also originally interested in looking at how characteristics of already pledged donations affect likelihood of future donations, and thus success of projects in a time-series framework. If there is time will analyze this problem using markov chains and the Viterbi algorithm for analysis.

The dataset also includes project description essays, written by the creating teachers. Given the time frame, we will likely not be able to address the impact of these essays on the probability of being funded, but it would be an interesting problem to return to in the future.

The DonorsChoose.org website has an option to randomly select a project to donate to. The site chooses a project that is from a teacher who has never been funded before in a high-poverty community and which has a quick approaching deadline. Presumably it uses these criteria because they are representative of projects that have a low success rate and hope to boost their success by making these projects more salient. We will examine whether these characteristics are highly negatively correlated with success in funding or if, perhaps, there are other factors that better predict low success rates. If this is the case then DonorsChoose can filter based on these factors to better improve the success rate of its worst funded projects.

8. Encountered Problems

In the SVM environment, we found some of the largest factors were longitude, latitude, monetary goal and date posted. While these factors may be influential, we noticed that these values on the whole had a much larger values compared to most other categories that were boolean values. The decision tree id3 algorithm showed similar issues.

To address this, we plan on creating a normalized version of data to test on, to see if putting all of the attributes on the same scale can improve the accuracy of the linear classifiers.

Our current implementation of the id3 algorithm treats enum values as real numbered values by simply assigning a real number to each enum. This functions correctly and allows the creation of decision trees, but imposes an arbitrary ordering on the different enum values that gives significance where there is none. We plan on fixing this problem by altering the id3 algorithm to treat enum values correctly.

9. Schedule

- 25th October: Process and organize data
- 4th November: Begin classification
- 8th November: Compile results, begin comparison of different results
- 14th November: Compile models, begin comparison of different models
- 18th November: Model comparison, hypothesis testing
- 24th November: Begin writing poster and report.
- 4th December: Poster presentation.
- 10th December: Final project report (and code) due.