Machine Learning Engineer Nanodegree

Capstone Proposal – Give Me Some Credit

Brian Wozniak
January 9th, 2019

Domain Background

I am investigating a past Kaggle competition called "Give Me Some Credit." In this competition, participants predicted how likely a person was to experience financial distress in the following two years. The following is a summary of the "Overview" section of the contest: Banks play a crucial role in deciding who gets credit. Credit is needed for markets to function. Machine learning techniques can be used to improve the way credit is given to consumers.

As stated in the summary, it is important that the credit granting process be as efficient as possible so that markets can function properly. Also, it is important to allow consumers who are financially responsible to get access to credit. Getting credit to the right people and keeping it out of the hands of the wrong people helps the economy, banks, and the individuals that are given the credit. I am interested in this problem because I have worked for a credit card company for 5 years and this is a problem, we as a company also face.

This is a link to the competition details - https://www.kaggle.com/c/GiveMeSomeCredit

The following is a citation to a paper that addresses using machine learning in credit scoring; specifically, utilizing decision tree information to inform a logistic regression:

Hué, Sullivan & Christophe, Hurlin & Tokpavi, Sessi. (2017). Machine Learning for Credit Scoring: Improving Logistic Regression with Non Linear Decision Tree Effects. https://www.researchgate.net/publication/318661593_Machine_Learning_for_Credit_Scoring_Improving_Logistic_Regression_with_Non_Linear_Decision_Tree_Effects

Problem Statement

The problem to be solved is improving the performance of a credit scoring model from the benchmark model. Specifically, it is to build a model that will predict the probability of a person experiencing financial stress in the next two years. The inputs are several financial variables about each individual and the output is the probability of the financial stress over the next two years. Since this is predicting a probability of a class, this is a classification problem. Quantifiably, the problem is to improve the area under the curve score from the benchmark model.

Datasets and Inputs

The dataset is provided by the company that hosted the competition. It is a collection of 250,000 borrowers. It contains various financial variables about a borrower's history as well as some personal information such as age and number of dependents.

The financial variables are appropriate because these variables generally tell the credit worthiness of an individual. However, I will not use age or number of dependents for this model, as these are beyond the

control of the borrower and I do not believe it to be ethical to punish an individual for variables beyond their control.

There are nine financial variables that I will use in the models. They are all continuous (some are discrete counts). There are no categorical variables in the dataset. The output variable is a binary outcome, either 0 or 1, if they experienced financial distress in the next two years or not. The variable has severe class imbalance, as the output variable has only 10,026 of those with financial distress out of the 150,000. Since the provided test set does not have labels for the output variable, I will split the provided training dataset into a training, validation, and test set. When I preprocess the dataset, before building my selected models on them, I will balance the classes for each of the different datasets.

Solution Statement

The solution to the problem will be to turn to machine learning. I will use three supervised learning methods: Random Forest, Support Vector Machine, and Gradient Boosting Machine through Xgboost. I will also do some preprocessing such as examining outliers, normalizing and scaling variables, and using SMOTE to counteract class imbalance. I will predict who will experience financial distress over the next two years using each model and compute the area under the curve. I will then predict on the test dataset provided by the company on Kaggle and submit to verify how well the models predict.

Benchmark Model

For the benchmark model I created a logistic regression utilizing all of the financial variables. I filled in missing values with the mean of the respective columns. The area under the curve for the model is 0.654. When predicting on the actual test data set from the competition, the area under the curve is 0.674.

Evaluation Metrics

The evaluation metric that was used for the competition and that I will be using to quantify the performance of the benchmark and solution models is the area under the curve metric. I will be using the area under the ROC (Receiver Operating Characteristic) curve as the Kaggle leaderboard indicates that the area under the curve is between 0 and 1.

Project Design

I will begin the project by reading in the dataset. I will split the dataset into training, validation, and test datasets. I will then do some data analysis (data manipulation and visualizations) to understand the financial variables. This will help me to determine which variables should be used and if they need to be scaled or normalized in any way before being used in the model. Also, I can identify and remove outliers as appropriate. For missing values, I will use either the mean, median, or mode to impute the values. I will also balance the classes for the output variable using SMOTE, which is the most common method for oversampling according to Wikipedia: https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis#SMOTE.

Once the variables are prepared, I will train each model on the training dataset. I will use a Random Forest, a Support Vector Machine, and an Xgboost. I will use a grid search on each model on a few variables to maximize performance of the models. I will then predict on the test dataset I created from the beginning of the process and evaluate performance based on the roc_auc_score in the

sklearn.metrics package of Python. The goal is that at least one of the models will outperform the benchmark logistic model.