Machine Learning Engineer Nanodegree

Capstone Proposal

Adriel Vieira
April 3rd, 2018

Proposal

Domain Background

This project is based on a Kaggle competition posted by Santander bank. The goal of this competition is to help Santander predict whether a given customer is a satisfied or unsatisfied customer. Customer satisfaction is a key measure of success. Unhappy customers don't stick around. What's more, unhappy customers rarely voice their dissatisfaction before leaving.

By helping to identify dissatisfied customers early in their relationship, allows Santander to take proactive steps to improve a customer's happiness before it's too late. Reducing churn is something really important for most companies, considering that acquiring new clients is much more expensive than working on the current relationships and keeping them active and profitable.

Problem Statement

In this competition, the task is to work with hundreds of anonymized features to predict if a customer is satisfied or dissatisfied with their banking experience. It is a binary classification problem in which the goal is to predict the probability that each customer in the test set is an unsatisfied customer.

Datasets and Inputs

As mentioned on Kaggle, Santander provided an anonymized dataset containing a large number of numeric variables (370 features and 1 target variable). The "TARGET" column is the variable to predict. It equals one for unsatisfied customers and 0 for satisfied customers. The dataset provided has 76020 rows or data points for training. The testing set provided contains roughly as much data points as the training set (75818 rows), but lacks of the “TARGET” column. This dataset will be used as validation set.

Since we have hundreds of features to work with, and we don’t have information about the meaning behind these features, the steps of feature selection and feature engineering will be really important for this task.

The datasets are available at www.kaggle.com/c/santander-customer-satisfaction/data.

Solution Statement

The solution to this machine learning problem is a binary classifier that predicts a value of 1 or 0 (and/or the probabilities of these labels being true), representing unsatisfied and satisfied customers. After working on features, we’ll try different algorithms (e.g. linear regression, random forest and XGBoost), with different sets of parameters. The combination that performs better will be used to predict on the testing set, generating our submission to Kaggle.

Benchmark Model

Kaggle provides a Sample Submission to be used as benchmark for the project. In our case, the benchmark is an all-zeros submission. Our training set has much more ones than zeros and it also makes sense to think of an all-zeros scenario as the banking taking no action on customers’ behavior and considering them all “satisfied” customers.

Evaluation Metrics

This competition’s submissions are evaluated on area under the ROC curve between the predicted probability and the observed target.

The model estimates a probability of the target variable being True (or 1) and a threshold value is then applied to separate Trues and Falses. The ROC curve is the interpolated curve made of points whose coordinates are functions of the threshold. It is obtained by plotting the True Positive Rates (True Positives / All Positives) on the Y axis and False Positive Rates (False Positives / All Negative) for every probability threshold value possible to be used to classify a datapoint as Positive or Negative.

Project Design

To solve the task proposed by Santander bank the main duty is to train a binary classifier, but before doing so, we need to process and clean our data, work on the features in order to decide which ones to use and test different models to be trained and the parameters sets that best fit the data and can still generalize well.

The first thing to do is to get a sense of the data we are working with. For that, we'll take a look at the meta description of our data, analyse target distribution and missing values.

Another part of preparing the data set involves analysing and, possibly changing, the data distribution so it can be better consumed by some algorithms. We shall visualize the distributions of features and decide whether to apply a feature scaler or not, aiming to have distributions closer to normal.

Because we have hundreds of features, we first need to decide the ones that will feed our classifier, so that we don't suffer too much from the curse of dimensionality. We might transform our features using PCA or another dimensionality reduction technique. One hot encoding might also be useful and should be tested and/or used.

The next step is to work with outliers and missing values. Missing values might be substituted by -1 at the beginning, and optimised later on. On the other hand, outliers need a rule to be detected. One to be used is the Turkey's Method.

With our dataset processed we're ready to begin training our baseline model. We will use scikit learn to train a Random Forest Classifier. Like Decision Trees, they're simple to understand, can handle numerical and categorical features and doesn't make assumptions over the data distribution. Also, being an ensemble method helps to avoid overfitting

Other algorithms, such as Linear Regression and XGBoost should also be tested before moving into parameter optimization, always validating the model with scikit learn implementation of K Fold Validation. Stacking models may also be tested and used at the end and we can use Grid Search to decide the hyper parameters of the model.

Reference

https://www.kaggle.com/c/santander-customer-satisfaction
https://www.youtube.com/watch?v=mDGNE-LYDiY
https://www.joyofdata.de/blog/illustrated-guide-to-roc-and-auc/