CSCI 567 - Spring 16 - Mini Project
Team: MLCLASS_SP16_nilay_suraj

Nilay Chheda
Department of Computer Science
University of Southern California
Los Angeles, 90007
nilayvac@usc.edu

Suraj Rajasekhar
Department of Computer Science
University of Southern California
Los Angeles, 90007
rajasekh@usc.edu

Abstract

In this report we describe our approach for building data models to predict which customers are happy customers. The raw dataset and the problem statement are taken from the Santander Customer Satisfaction competition on Kaggle [1]. First we introduce the problem statement and description of the dataset. Then we identify important pre-processing steps to transform the data. After that we describe few Machine Learning Models used for this classification problem. Finally we discuss the results obtained along with some lessons learnt on overfitting the data. Few insights are incorporated from Kaggle [2]-[4].

1 Introduction

Machine learning and data mining have become a part of our daily lives. Our emails are protected by smart spam classifiers, which learn from massive amounts of spam data and user feedback. As we shop online, a recommender system helps us find products that match our taste by learning from our shopping history. Besides improving personal life, machine learning also plays a key role in helping companies make smart decisions and generate revenue: advertising systems match the right ads with the right users at the right time. Demand forecasting systems predict product demand in advance, allowing sellers to be prepared. Fraud detection systems protect banks from malicious attackers. There are two important factors behind the success of these applications effective machine learning models that can capture the complex dependence between variables and scalable learning systems that effectively learn the model of interest with large amount of collected data. Tree boosting is one of the most important and widely used machine learning models. Variants of the model have been applied to problems such as classification and ranking. These types of models are used by many winning solutions for machine learning challenges. Despite its great success, the existing public practice of tree boosting algorithms are still limited to million scale datasets. In this paper, we intro-

duce XGBoost, a novel machine learning system that reliably scales tree boosting algorithms to billions of samples with fault tolerance guarantees.

2 Data

The data consists of multiple sets of customer features uniquely identified by an ID. The training dataset consists of 76020 observations with 371 observable features. Target column is the variable to predict. It equals one for unsatisfied customers and zero for satisfied customers. The test set consists of 75818 observations with no Target column, 370 observations and our task is to predict the probability of each customer in the test set is an unsatisfied customer.

Its very important to learn the features [5] before applying cleaning methods and since the feature names were not clear, it became very difficult to predict which features were important.

- Var15 is one of the most important feature in the dataset and its learnt that its the age of the customer.
- According to dmi3kno, var_num3 is the number of bank products. After plotting the histogram, we found that unhappy customers has less products.
- Var3 is the nationality of the customer.
- Var38 is suspected as mortage value with the bank as per one of the discussion forms [6].

2.1 Data Preprocessing and Cleaning

The dataset provided with this competition posted lot of challenges like high percentage of missing values, outliers, orthodox structure and size of the training set.

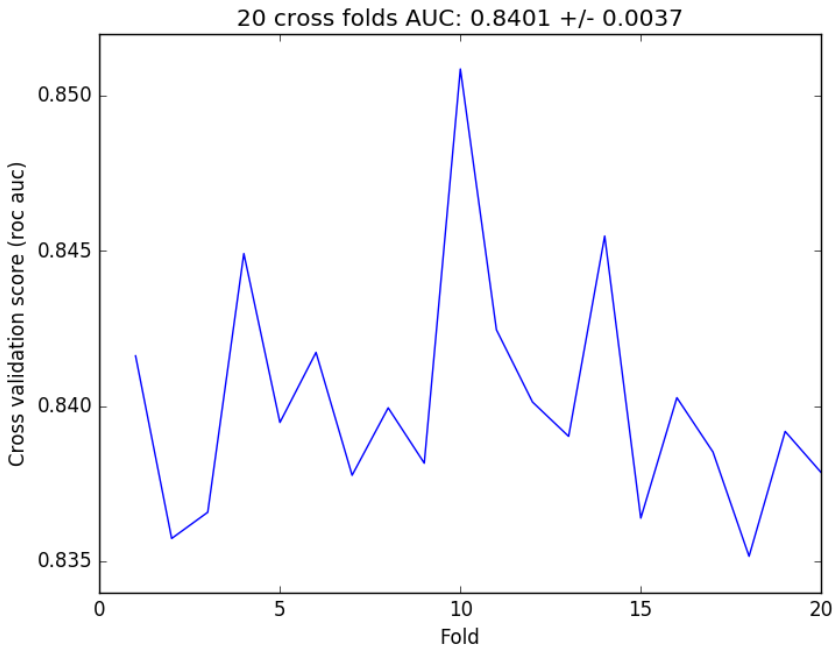
- Constant features can lead to errors in some models and it provides no information in the training set that can be learned from. With the argument perc, there is a possibility to remove features for which less than perc percent of the observations differ from the mode value.
- Apply band pass filter in removing outliers from the dataset. Also feature labeled var38 was most skewed. Hence applied logarithmic transformation to make this feature less skewed.
- Another important factor is abundance of missing values. This made lot of statistical derived statistical features as NA (Missing). To reduce this, we combined saldo_medio_var33_ult1 and saldo_medio_var44_ult1 into one and its one of the most important features in our model.

3 Model and Learning Algorithms

Before we delve into the models we used for this competition, its important to understand the evaluation metric for this competition which is Area under ROC curve between predicted probability and observed target.

3.1 Logistic Regression

We started with the basic approach by applying logistic regression on the scaled data with both L1 and L2 norm as regularization. This model didnt perform well when compared to other models. We got an average CV score of 0.78954. One of the reasons for it was that this algorithm did not know how to handle missing values. Hence we used mean value wherever missing values were found.



3.2 XGBoost (eXtreme Gradient Boosting)

XGB is an advanced implementation of gradient boosting algorithm. It has several advantages over gradient boosting algorithms. It implements parallel processing and is blazingly faster as compared to GBM. XGBoost allows user to run a cross-validation at each iteration of the boosting process and thus it is easy to get the exact optimum number of boosting iterations in a single run. We got an CV of 0.824242 over the test set. We followed the below approach in order to learn several parameters for this model.

- Choose a relatively high learning rate. Generally a learning rate of 0.1 works but somewhere between 0.05 to 0.3 should work for different problems. XGBoost has a very useful function called as cv which performs cross-validation at each boosting iteration and thus returns the optimum number of trees required.

- Tune tree-specific parameters (max_depth, min_child_weight, subsample, colsample_bytree) for decided learning rate and number of trees.
- Tune regularization parameters (lambda, alpha) for xgboost which can help reduce model complexity and enhance performance.

3.3 Other Hybrid techniques

We also applied other advanced techniques like Random Forest and classification algorithm to classify which range the target value belongs in and then using a regression algorithm to find if it belongs to happy or unhappy customer. While in theory, this algorithm works well with good CV, but given this dataset it performed poorly on private leaderboard and fantastic on public leaderboard. We believe low CV is because of overfitting the trained data and our CV was good in public leaderboard because it was capable of generalizing the model.

4 Results

4.1 Execution of code

We decided to use R for our analysis since it provides powerful computational abilities for data analysis with small lines of code. To run this script, you need to install R studio with all the required Machine Learning packages installed like Matrix, XGBoost and pRoc. Place the test and training dataset in the same path as the script and run to generate submission.csv.

4.2 Final scores

We achieved our best score with XGBoost library and obtained a public leaderboard score of 0.842839 and private leaderboard score of 0.824242.

5 References

[1] Kaggle - https://www.kaggle.com/c/santander-customer-satisfaction/

[2] working of XGBoost - http://xgboost.readthedocs.io/en/latest/model.html

[3] Kaggle ensembling guide - http://mlwave.com/kaggle-ensembling-guide/

[4] Statistical Measures in R - http://www.r-tutor.com/elementary-statistics/numerical-measures

[5] Exploring features - https://www.kaggle.com/cast42/santander-customer-satisfaction/exploring-features

[6] https://www.kaggle.com/c/santander-customer-satisfaction/forums/t/19895/var38-is-mortgage-value