Customer Transaction Prediction

Rutgers University has made this article freely available. Please share how this access benefits you.
Your story matters. [https://rucore.libraries.rutgers.edu/rutgers-lib/61717/story/]

This work is the AUTHOR'S ORIGINAL (AO)
This is the author's original version of a work, which may or may not have been subsequently published. The author accepts full responsibility for the article. Content and layout is as set out by the author.


Terms of Use: Copyright for scholarly resources published in RUcore is retained by the copyright holder. By virtue of its appearance in this open access medium, you are free to use this resource, with proper attribution, in educational and other non-commercial settings. Other uses, such as reproduction or republication, may require the permission of the copyright holder.

Article begins on next page
Customer Transaction Prediction

Saachi Shah
Department of MSIS, Rutgers University
E-mail: saachi.shah@rutgers.edu

Kemal Gursoy
Department of MSIS, Rutgers University,
E-mail: kgursoy@business.rutgers.edu

Abstract
At Santander their mission is to help people and businesses prosper. They are always looking for ways to help customers understand their financial health and identify which products and services might help them achieve their monetary goals. The goal of this project is to predict which customers will make a specific transaction in the future. The dataset contains numeric feature variables, the target column, and a string Id code column. The task is to predict the value of target column in the test set.
In the project we will explore the data, prepare it for a model, train a model and predict the target value for the test set.
In Santander Customer Transaction Prediction project we have a binary classification task. Train and test data have 200k samples each and we have 200 anonymized numerical columns. It would be interesting to try good models without overfitting and knowing the meaning of the features. The data is anonymized, each row containing 200 numerical values identified just with a number.

1. Introduction
According to Epsilon research, 80% of customers are more likely to do business with you if you provide personalized service. Banking is no exception. Santander Group aims to go a step beyond recognizing that there is a need to provide a customer a financial service and intends to determine the amount or value of the customer’s transaction. This means anticipating customer needs in a more concrete, but also simple and personal way.
Santander Bank was founded in 1902 as Sovereign Bank, savings and loan in Wyomissing, Pennsylvania. The company’s earliest customers were largely textile workers. Sovereign expanded rapidly during the savings and loan crisis of 1980s and 1990s, acquiring numerous other banks. It is based in Boston and its principal market is the north eastern United States.
This paper focuses on Santander Bank, a large corporation focusing principally on the market in the northeast United States. Through means of a Kaggle competition (Santander, 2015), the objective is to find an appropriate model to predict whether a
client will make a transaction in the future. Having this model in place can ensure that Santander can take proactive steps to improve a customer’s happiness before they would take their business elsewhere. First the paper will discuss related work done on this. Secondly it delves into the data, analysing groups of variables and individual features to give us insight in what is relevant. Thirdly several cleaning procedures that were employed to lead to better results are outlined. Fourthly we explain the performance measure of this competition and the three models: Logistic Regression, Random Forest, Naive Bayes and Decision Tree that we utilize to tackle the problem.

2. Today’s Approach

Santander Group aims to go a step beyond recognizing that there is a need to provide a customer financial service and intends to determine the amount of the customer transaction. This means anticipating customer needs in a more concrete, but also simple and personal way. With so many choices for financial services, this need is greater now than ever before. Their data science team is continually challenging machine learning algorithms, working with the global data science community to make sure they can more accurately identify new ways to solve our most common challenge, binary classification problems such as: is a customer satisfied? Will a customer buy this product? Can a customer pay this loan? In this project, Santander Group is asking to help them identify the prediction of transactions for each potential customer. This is a first step that Santander needs to nail in order to personalize their services at scale. First we checked if there were any missing data. Which there wasn’t. We also checked the data types, which are float64s. So no categorical data and all numeric data. With this in mind, we decided to look at the correlations between the features. We see that the features are uncorrelated because the highest ones are only at 0.01. This is usually a good thing for model performance because uncorrelated features contain more information potentially. Later we did the exploratory analysis part where we checked mean, std, min, max

<table>
<thead>
<tr>
<th></th>
<th>Target</th>
<th>Var_0</th>
<th>Var_1</th>
<th>Var_3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Count</td>
<td>198264.0</td>
<td>198264.0000</td>
<td>198264.0000</td>
<td>198264.0000</td>
</tr>
<tr>
<td>mean</td>
<td>0.100412</td>
<td>10.679648</td>
<td>-1.627397</td>
<td>10.714957</td>
</tr>
<tr>
<td>std</td>
<td>0.30054</td>
<td>3.03</td>
<td>4.04</td>
<td>2.069</td>
</tr>
<tr>
<td>min</td>
<td>0.000</td>
<td>0.5000</td>
<td>-15.50</td>
<td>2.640</td>
</tr>
<tr>
<td>25%</td>
<td>0.000</td>
<td>8.45</td>
<td>-4.7</td>
<td>8.722</td>
</tr>
<tr>
<td>50%</td>
<td>0.000</td>
<td>10.52</td>
<td>-1.6</td>
<td>10.56</td>
</tr>
<tr>
<td>75%</td>
<td>0.000</td>
<td>12.75</td>
<td>1.35</td>
<td>12.51</td>
</tr>
<tr>
<td>max</td>
<td>1.00</td>
<td>20.31</td>
<td>10.37</td>
<td>19.353</td>
</tr>
</tbody>
</table>

After performing the data exploration, we started applying the machine learning algorithms where we started with supervised learning Naive Bayes (NB) Gaussian Naive Bayes (GNB), XGBoost (XGB) LightGBM (LGBM). The tree based gradient boosted methods XGB, LGBM are some of the most popular methods for tackling tabular supervised learning problems and getting good performance quickly without specifying a particular architecture such as with neural networks. Neural networks are the most powerful methods, being able to approximate almost any function, with the downside of having a large number of options for defining the architecture.

3. Models

- **Logistic regression**

This is the classification problem. We can use logistic regression for this problem. Logistic Regression is used when the dependent variable(target) is categorical. It has a bias towards classes which have large number of instances. It tends to only predict the majority class data. The features of the minority class are treated as noise and are often ignored. Thus, there is a high probability of misclassification of the
minority class as compared to the majority class

- **Naive Bayes**
  Naive Bayes model is easy to build and particularly useful for very large data sets. Along with simplicity, Naive Bayes is known to outperform even highly sophisticated classification methods. It performs well in case of categorical input variables compared to numerical variable Santander Customer Transaction Prediction curve, which is a strong assumption. Also numerical variables are very less correlated.

- **Random Forest**
  Random Forest is a bagging based ensemble learning model. Random forests is slow in generating predictions because it has multiple decision trees. Whenever it makes a prediction, all the trees in the forest have to make a prediction for the same given input and then perform voting on it. This whole process is time-consuming. Thus result (AUC-score) shown for random Forest is not modified. These results are shown for default values of parameters.

- **LightGBM**
  LightGBM is Gradient Boosting ensemble model which is faster in speed and accuracy as compared to bagging and adaptive boosting. It is capable of performing equally good with large datasets with a significant reduction in training time as compared to XGBOOOST. But parameter tuning in LightGBM should be done carefully.

- **XGBoost**
  XGBoost is the leading model for working with standard tabular data (the type of data you store in Pandas Data Frames, as opposed to data like images and videos). XGBoost models dominate many Kaggle competitions. The implementation of the algorithm is such that the compute time and memory resources are very efficient. A design goal was to make the best use of available resources to train the model. XGBoost has a few parameters that can change accuracy and speed of your model significantly - n_estimators and early_stopping_rounds.

XGBoost directly avoids overfitting by promoting simplicity of models in the objective via regularization, unlike Random Forest that only limits the way trees can grow by imposing restraints. (Chen, 2014)

### 4. Experiments

#### Datasets

Both train and test data have 200,000 entries and 202, respectively 201 columns.

**Train contains:**

- ID_code (string);
- target;
- 200 numerical variables, named from var_0 to var_199;

**Test contains:**

- ID_code (string);
- 200 numerical variables, named from var_0 to var_199;

There are no missing data in train and test datasets. Standard deviation is relatively large for both train and test variable data;

- Min, max, mean, Sd values for train and test data looks quite closely;
- Mean values are distributed over a large range.
- We can observe that there is a considerable number of features with significant different distribution for the two target values.
  Foreexample, var_0, var_1, var_2, var_5, var_9, var_13, var_106, var_109, var_139 and many others.
- Also some features, like var_2, var_13, var_26, var_55, var_175, var_184, var_196 shows a distribution that resembles to a bivariate distribution.
- We will take this into consideration in the future for the selection of the features for our prediction model.
5. Comparison of Methods

The performance measure of this Kaggle competition is the area under the receiver operating characteristic curve, AUC for short. This metric deals well with the imbalance that is typical in prediction. We use 4 different methods to come to a solution: Logistic Regression, Random Forest from Scikit Learn, LightGBM and XGBoost (Chen and Guestrin, 2016). Consequently, fit the best model on the entirety of the train data and then predict labels for the test set and validate these for the final score. The baseline performance achieved an average AUC score of 0.89. After a series of tuning trials, the top result from the training data was an AUC score of 1.

<table>
<thead>
<tr>
<th>Models</th>
<th>Accuracy Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random Forest Classifier</td>
<td>1</td>
</tr>
<tr>
<td>LightGBM</td>
<td>0.94</td>
</tr>
<tr>
<td>XGBoost</td>
<td>0.92</td>
</tr>
<tr>
<td>Logistic Regression</td>
<td>0.91</td>
</tr>
</tbody>
</table>

6. Results and Discussion

It is a performance measurement for machine learning classification problem where output can be two or more classes. It is a table with 4 different combinations of predicted and actual values. Standard deviation is relatively large for both training and test variable data. However, the min, max, mean, median, standard deviation values for training and test data looks quite close. Mean values are distributed over a large range. Moreover, mean and median have similar distribution. Both training and test data is negatively skewed. After modelling LightGBM with standard parameters, the Area under the ROC Curve (AUC) Score was 0.88576

Further Improvements
- Further improvements in model can be done by-
  - Using Parallel Processing with LightGBM Algorithm.
  - Selecting important features and then modelling them.
  - Using Stratified Folding for train and test splits.
  - Trying for XGBoost for faster speeds.

Conclusion

This paper researched how to preemptively understand if customers of Santander will be dissatisfied using Machine Learning. A semi-anonymized dataset, to protect the privacy of the customers, provided difficulties in asserting what could be relevant or not, especially in light of a huge feature set. However, a thorough data analysis discerned the meaning and interpretation of several features. A Python implementation utilized the Logistic Regression, Random Forest and XGBoost algorithms, carefully tuned, in order to lead to predictions. Further research could for example employ different solution methods, apply more feature engineering or combine several models instead of trying singular models. More specifically they can increase the computation time that goes into tuning and for example make the correlation filtering dependent on the correlation with the target. This was a classification problem on a typically unbalanced dataset with no missing values. Predictor variables are anonymous and numeric and target variable is categorical. Visualizing descriptive features and finally we got to know that these variables are not correlated among themselves. After that we decided to treat imbalanced dataset and built different models with original data and choose LightGBM as my final model then using the same model with feature engineered data, we got AUC-Score of 0.89.

References
5. https://medium.com/analytcs-vidhya