HMDA Loan Group Prediction Model Tutorial

Jiafan(Kevin) Deng

Introduction

As Change Financial is a small regional bank, it might not have very strong finance support as those big banks. Thus, Change Financial might need smart strategies to prompt their loan products to proper potential customers. Nowadays, building machine learning models in marketing analytics is very popular and useful. As a data analyst at Change Financial, I decided to build a model to predict which kind of loan a customer might want to get.

Data Preparation

In our HMDA dataset, first we need to filter all the missing values in our data set. Next, we need to select variables for our further analysis.

Variables Not Selected

Variable Name	Reasons
Census_Tract_Number	Repeated geographic information
Loan_Amount_000	The data used for response variable
State	Repeated geographic information with State_Code
County_Name	Repeated geographic information with County_Code
Repondent_Name_TS	Repeated information with Respondent_ID
Repondent_City_TS	As we already have much geographic data in our dataset, and the locations of Respondents and Parents do not seem like the variables having big influence on our response data, I decided to filter these data.
Repondent_State_TS	
Repondent_ZIP_Code	
Parent_Name_TS	
Parent_City_TS	
Parent_State_TS	
Parent_ZIP_Code	

Model Selection

As we need to predict which kind of loan a customer will apply, Loans_Group should be our response variable. And there are three levels in Loan_Group, L, M, and H, so we need a multi-classification model here. Also, the accuracy of our prediction model is definitely the most important. In the context of machine learning, decision trees, random forest, and neural network are good choices for a multi-classification model. However, when I tried neural network on the data, my laptop always crashed because the algorithm's requirement on large memory and high computation ability. So I decided to use SVM, Decision Trees, Random Forest and XGBoost for my models.

Model Comparation

First, I tried SVM and the classification report is below:

SVM

	precision	recall	f1-score	support
0	1.00	0.00	0.00	4397
1	0.79	0.82	0.80	146781
2	0.76	0.75	0.75	127219
avg / total	0.78	0.77	0.77	278397

Actually, SVM did a quiet good job and achieve 78% accuracy. Then let's try Decision Trees.

Decision Trees

	precision	recall	f1-score	support
0	0.56	0.57	0.56	4397
1	0.76	0.76	0.76	146781
2	0.71	0.71	0.71	127219
avg / total	0.73	0.73	0.73	278397

It turns out that Decision Trees does not perform better than SVM so I move forward to Random Forest.

Random Forest

	precision	recall	f1-score	support
0	0.63	0.63	0.63	4397
1	0.79	0.83	0.81	146781
2	0.78	0.73	0.75	127219
avg / total	0.78	0.78	0.78	278397

Random Forest gives us the same accuracy as SVM equaling to 78%.

XGBoost

	precision	recall	f1-score	support
0	0.76	0.48	0.59	4397
1	0.83	0.82	0.82	146781
2	0.78	0.80	0.79	127219
avg / total	0.80	0.80	0.80	278397

In the end, our XGBoost model gives us the best accuracy which is 80%.

Then I tried to use PCA(Principle Components Analysis) on my models but I could not get better accuracy, so the best model is still the XGBoost model without using PCA.

Conclusion

By trying several different multi-classification models, we found that XGBoost performed the best on our data set and reached 80% accuracy. With this model, we could be very likely to prompt the right kind of loans to our potential customers, thus attracting more customers to our Change Financial.