

Credit Card Lead Prediction

Ayan Maity


Problem Statement

Happy Customer Bank is a mid-sized private bank that deals in all kinds of banking products, like Savings accounts, Current accounts, investment products, credit products, among other offerings.

The bank also cross-sells products to its existing customers and to do so they use different kinds of communication like tele-calling, e-mails, recommendations on net banking, mobile banking, etc.

In this case, the Happy Customer Bank wants to cross sell its credit cards to its existing customers. The bank has identified a set of customers that are eligible for taking these credit cards.

Now, the bank is looking for your help in identifying customers that could show higher intent towards a recommended credit card, given:

- Customer details (gender, age, region etc.)
 - Details of his/her relationship with the bank (Channel_Code, Vintage, 'Avg_Asset_Value etc.)
- 

Understanding the business problem

- There is huge competition. To keep up with the fluctuating customer demand, we need to identify those customer who might be interested in Credit Cards.
- Manually analyzing these leads will be error-prone and a time consuming process, with the help of machine learning this task can be automated.



Dataset Overview

Dataset	Features
train.csv	ID, Gender, Age, Region_Code, Occupation, Channel_code, Vintage, Credit_Product, Avg_Account_balance, Is_Active, Is_Lead
test.csv	ID, Gender, Age, Region_Code, Occupation, Channel_code, Vintage, Credit_Product, Avg_Account_balance, Is_Active
sample_submission	ID, Is_Lead

Problem Objective

Build a classifier that predicts if the customer is a lead or not.

Error Metric

The evaluation metric for this problem is `roc_auc_score` across all entries in the test set.



Exploratory Data Analysis

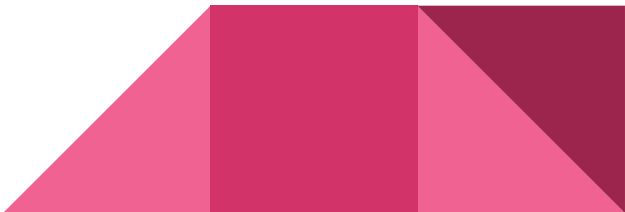
Exploratory Data Analysis

To speed up the model building process. I took advantage of the [Pandas Profiling Library](#), which is an open source Python module with which we can quickly do an exploratory data analysis with just a few lines of code.

Generated a detailed [report](#) for our train dataset. (To view the report please download and open it in your local browser.



Final Thoughts on EDA

- There are 6 categorical 3 numerical and 2 boolean features in our dataset.
 - All the numerical columns i.e. Age, Vintage, Avg_Account_Balance are skewed in nature. So we need to do some transformation.
 - There are some missing values in the Credit_Product Column. Which consists of 1.1% of the entire dataset.
 - Age and Vintage feature are highly correlated.
 - Region_Code is a high cardinal categorical variable.
 - Target Variable is imbalanced.
- 


Model Building

Data Preprocessing

- Missing Value Imputation:
 - Filled missing values with a string value (“UA”, UA = Unavailable). Treated them as a separate category.
- Outlier Removal:
 - Removed few rows where ‘Avg_Account_Balance’ is greater than max value of ‘Avg_Account_Balance’ in test data.



Feature Extraction / Feature Engineering

- Log transformed 'Avg_Account_Balance' and created a separated feature named 'Avg_Account_Balance'.
 - Applied 'boxcox' transformation to the vintage column and created 'Vintage_box' feature.
 - Using "KBinsDiscretizer", binned 'Age' and 'Avg_Account_Balance' feature separately and created new features from it.
 - Label Encoded the categorical variables.
 - Frequency Encoded the 'Region_Code' feature because of its high cardinality.
 - Dropped ID, 'Vintage', 'Avg_Account_Balance' while building the model.
- 

Final Model Building

- Used XGBoost Classifier, Random Forest Classifier , LGBMClassifier and ANN
- For final model building used “CatBoostClassifier” .
- Finely tuned hyperparameters like - loss function ,depth ,iterations ,learning_rate.



Future Work

- Apply K-fold Cross Validation and check model performance.
- Handle categorical columns with different encoding methods.
- Create more features.
- Build ANN model with an extra layer and add dropout and batch normalization to avoid overfitting.





Thank You

Ayan Maity