Anti-money laundering (AML) client risk rating (CRR) model

Yiqu Ding, Shaohong Dong, Tongfei Zhou

November 30, 2022

1 Data Pre-processing

1.1 Data quality analysis

We imported the data set and checked no missing values. Since all data have been normalized, the data is incomprehensible for us to check for any bad/unreasonable data. However, there is one duplicated feature in the data set, which is `tot_acct_num`, so we drop this feature to avoid the perfect co-linearity issue. Also, as we decided to use tree models for the tabular data, we did not remove outliers since tree models are robust to outliers.

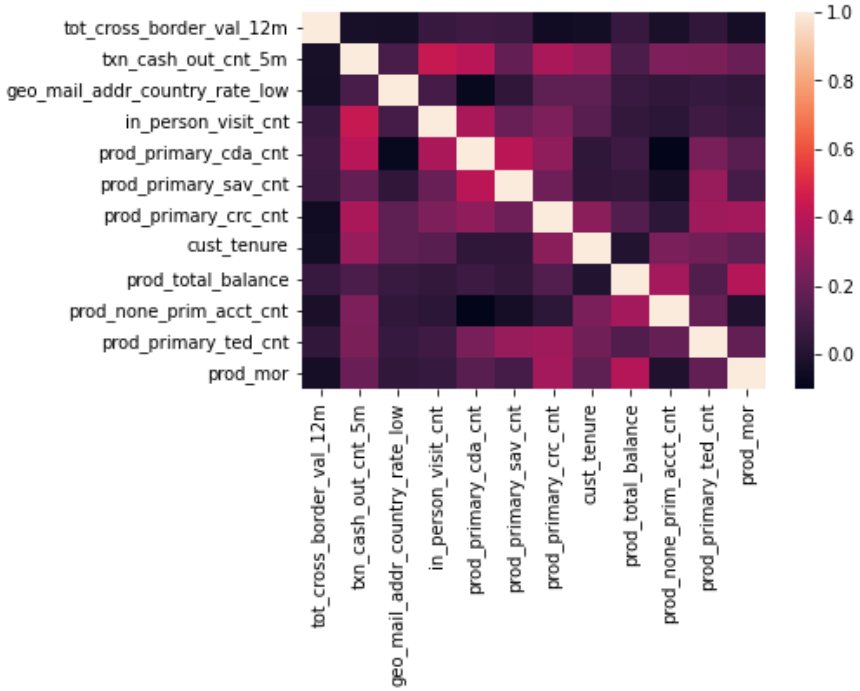
1.2 Data Transformation

It is unnecessary to transform the data because we have observed that the data is normalized. Meanwhile, we will be incorporating tree models in this report, which have the characteristics of not requiring data transformation.

1.3 Exploratory data analysis

The data set has 588 rows in total with 149 features. As the number of features is quite large, we would conduct feature selection in the first place and then develop the relevant models. Since we are solving a classification problem, we examined the distribution of our target classes, and the "rating" has a ratio of 2:1. So, so there is no imbalanced data issue.

Figure 1: features' correlation heatmap



2 Model Development

2.1 Feature selection

First, we observed the following heat map of the correlation between variables and excluded those with a high correlation (close to 1). Because including multiple variables with a high correlation induces over-fitting. Then we started modelling with these selected variables and refined our feature selection based on the results. We have used the feature importance method to determine the top 12 features and included them in our final model.

2.2 Train test split

We conducted a $70 - 10 - 20$ split ratio on the training, validation and testing date set. Since there is no imbalanced data issue, we did not consider sample weight.

2.3 Cross validation

We have applied 5-fold cross-validation to train our models. Cross-validation on a limited data sample can improve the skill of the model on unseen data (prediction power) and avoid overfitting issues.

2.4 Modelling technique

We have applied Autogluon (AutoML) to train our model. This automatic machine learning framework would train multiple models with the ensemble, boosting and stacking techniques. The models include K-Nearest Neighbors (KNN), Random Forest, LightGBM, CatBoost, XGBoost and Neural Network. After training these basic models, Autogluon would use the stacking technique to integrate all of the outputs of the basic models stated above and produce a best-performed model. We used this integrated model as our finalized model. For each basic model, Autogluon would use "Grid Search" to tune the model hyperparameters and also the required feature engineering automatically.
Notice that, since we are using the AutoML framework, we did not need to set the threshold by hand. We directly used the default threshold in this framework, which is 0.5, and we belive this is a fair threshold since we hold no bias (i.e, we do not put a very harsh threshold to maximize the model's sensitivity) to the data set. This belief also explains why we choose accuracy as our evaluation metric.

3 Model Validation

Q6. How do you interpret the model? Which are the important features, and how do they drive the model outcome? Is there any feature with counterintuitive behaviour? For example, a higher risk occupation should indicate a higher client risk rating, but the data or trained model might indicate the opposite. Can you think of a reason for this kind of problem, and how do you address it?

3.1 Model Performance Assessment - Evaluation metrics

We have chosen to evaluate our model based on accuracy, which is calculated by

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN}$$

The accuracy of our finalized model is 88.14%. Notice that we did not use balanced accuracy as our evaluation metric since the data has no imbalanced

issue.

3.2 Model interpretation

We have selected the independent variables in 1 to predict our target variable, the customer's credit rating. Reading off the table, we can roughly interpret the features that the model determines to identify riskiness. For example, all else being equal, a customer with a higher amount of total cross-border transactions in the last 12 months, a smaller average cash value in the last five months and a greater total balance will have higher customer credit risk. The features in 1 are ranked in the order of their feature importance score from the random forest modelling, indicating that the most predictive feature for customer riskiness is the total value of cross-border transactions over the last twelve months.

Features	Prediction (closer to 1)
tot_cross_border_val_12m	greater
tot_cash_val_5m	smaller
geo_mail_addr_country_rate_low	smaller
in_person_visit_cnt	greater
prod_primary_cda_cnt	greater
prod_primary_sav_cnt	greater
prod_primary_crc_cnt	greater
cust_tenure	smaller
prod_total_balance	greater
prod_none_prim_acct_cnt	smaller
prod_primary_ted_cnt	smaller
prod_mor	greater

Table 1: Features selected for model development

4 Issues/Future Improvement

There are several issues that can be improved further while we were training the model. First is that, since we only have 588 rows for our data set but 12 selected features in total, the models we have trained may be under-fitting. Therefore, we could have found more data or used a synthetic data-generating method to solve the lacking data issue. Secondly, Even though we conducted feature selection in the first place through correlation check, we also tried the method of feature importance by fitting an XGBoost model on the entire data set in the first place to make the feature selection. However, this method would be restricted by the model we choose. Therefore, more advanced techniques could be implemented to explore 149 features to find undiscovered useful features.

Third, since we chose to use an AutoML framework, we could not incorporate our belief in threshold settings or the weights in stacking the models.