AML Group 18
Ji Qi(jq2365), Yizhi Liu(yl4993), Wen Song(ws2685)

AML Final Project Report

Introduction

In light of the changing consumer attitudes and preferences, personal loan has become an increasingly popular topic in recent years. The topics related to Personal Loan can imply the business decisions of banks, playing an important role in the banking industry. The objective of this project is to utilize various machine learning techniques to figure out an optimal model that can predict whether a customer will take a personal loan or not (binary classification problem). The major steps that are potentially included are data preprocessing, exploratory data analysis, selecting appropriate features, and training and evaluating the models. It will be a valuable exercise in applying machine learning techniques to a real-world banking problem by helping increase customer satisfaction, improve business performance, and reduce risks for the bank.

About the Dataset

The dataset from Kaggle includes 5000 observations with 13 features classified into two categories (numerical or categorical features). The numerical features contain 5 variables: age, experience, income, CC AVG(Avg. spending on credit cards per month), and mortgage. For categorical variables, there are 2 ordinal variables including family and education, and 5 boolean variables including securities account, CD account, online banking, and credit card. In addition, Zipcode and ID are also classified as categorical features. Our target label is Personal Loan, indicating whether the potential customers have a higher probability of purchasing the loan based on their demographic characteristics. The '0' in the personal loan means the customers didn't accept the personal loan offered in the last campaign and '1' means the customers accepted the personal loan.

Data Preprocessing

Three major steps will be applied for the data processing, including Data Cleaning, Feature Engineering, and Addressing the Imbalanced Dataset. After the initial data exploration, we found that the dataset contains no missing or duplicate values. Based on the Univariate Analysis, we needed to drop all the noise records whose 'Experience' <0 or 'ZIP Code' < 5 digits. We dropped the variable ID (no useful information) and 'Experience' (strongly correlated with Age.) For Feature Engineering, we used Log transformation will be applied to remove the right skewness in 'Income', ''CCAvg', and 'Mortgage'.In addition, we convert the 'CCAvg' average monthly credit card spending to annual spending by times 12. Category Encoding will be selected to convert 'Family', 'Education', and 'Zip Code' into numerical values. The dataset is highly imbalanced (Only 9.6% of customers bought loans in the last campaign) and we chose Synthetic Minority Oversampling Technique and Train-Test Split (80:20) to reduce overfitting incurred by the biased model.

Evaluation Metrics

Since the dataset is highly imbalanced, we will use precision, recall, and F1 scores to evaluate the performance of our models. The precision score for Class '1' is a more crucial evaluation metric, since we would like to devise campaigns with better target marketing to increase the success ratio with minimal

budget. Therefore, the cost of a False Positive (**cost of the marketing campaign**) is much higher than that of a False Negative. Moreover, the F1 score is an important metric for an imbalanced dataset because it tells us how well our model learns both classes. Finally, we trained models on both datasets with and without SMOTE for comparison.

Model Performance Summary

Model Performance Summary (Test Dataset)

Model	Accuracy	Precision (macro average)	Recall (macro average)	F1 (macro average)
Logistic Regression w/ SMOTE	0.92	0.77	0.92	0.82
Logistic Regression w/o SMOTE	0.97	0.93	0.90	0.91
Random Forest w/ SMOTE	0.99	0.95	0.97	0.96
Random Forest w/o SMOTE	0.99	0.99	0.95	0.97
XGBoost w/ SMOTE	0.99	0.96	0.99	0.97
XGBoost w/o SMOTE	0.99	0.97	0.97	0.97
SVM w/ SMOTE	0.85	0.60	0.61	0.61
SVM w/o SMOTE	0.88	0.68	0.70	0.69

This project evaluated several Machine Learning (ML) models, including Logistic Regression, Random Forest, XGBoost, and SVM, on a personal loan dataset. As we mentioned in previous sections, one important consideration in this analysis is the class imbalance in the selected dataset. To mitigate this problem, we used the SMOTE technique to oversample the minority class and balance the dataset. Then, we used the SMOTE-enhanced dataset to train the selected ML classifiers. Moreover, we applied the original dataset (without SMOTE technique) to train the ML classifiers. By doing this, we planned to check if the SMOTE technique can improve the performance of the ML classifiers in classifying personal loans. We reported the classification performances of all ML classifiers ***on the test dataset***. As per the "Accuracy", XGBoost and Random Forest models performed well with and without SMOTE, achieving an accuracy of 99%. Logistic Regression also performed well, but only without SMOTE (97% accuracy). SVM, however, did not perform well, with accuracy scores below 90%. According to the measurement metric, "Precision": XGBoost and Random Forest models performed well with and without SMOTE, achieving a precision of over 95%. Logistic Regression also performed well, but only without SMOTE (93% precision). SVM had poor precision scores, below 70%. In terms of "Recall": Similarly, XGBoost and Random Forest models performed well with and without SMOTE, achieving recall values over 95%. Logistic Regression with and without SMOTE also had efficient performances, with scores over 90%.

AML Group 18
Ji Qi(jq2365), Yizhi Liu(yl4993), Wen Song(ws2685)

SVM had recall scores below 70%. Finally, we reported the F1 scores of all ML classifiers. F1 score is a combined metric that takes into account both precision and recall,l and can be a useful overall performance measure. Based on F1 scores, Random Forest and XGBoost with SMOTE appear to be the best-performing models, with scores of 96% and 97%, respectively. To summarize the findings, 1) XGBoost and Random Forest achieved the best classification performance for the personal loan dataset, with Logistic Regression also performing well. SVM had poor performance based on precision, recall, and F1 scores. 2) Comparing the classification performance of each classifier trained with and without the SMOTE technique, the results suggest that the SMOTE technique did not significantly enhance the classification performance of the ML classifiers for the personal loan dataset. In the next paragraph, we will provide further details on the Logistic Regression, Random Forest, and XGBoost classifiers.

1) Logistic Regression

Based on the Logistic Regression classifier, the hyperparameters that we chose to tune include {"solver", "penalty", "C" "l1_ratio"}. More specifically, we selected different values for each hyperparameter: "solver" with ["lbfgs", "liblinear", "newton", "saga"]; "penalty" with ["l1", "l2", "elasticnet"]; "c" with 10 numbers that are evenly spaced on a logarithmic scale between 10^{-1} and 10^{1}; "l1_ratio" with [0, 0.1, 0.2, 0.3]. 5-fold cross-validation was used to tune these hyperparameters based on F1-score and obtained the optimal Logistic Regression classifiers trained with and without SMOTE. After applying these two models with the optimal hyperparameters to the test dataset, we noticed that SMOTE did not improve precision, accuracy, precision, and F1-score in Logistic Regression model. Thus, we could conclude that the SMOTE technique did not do well in improving the performance of the Logistic Regression classifier.

Optimal Hyperparameters:

- Losgistic Regression with SMOTE: {'C': 5.994842503189409, 'l1_ratio': 0.0, 'penalty': 'l1', 'solver': 'liblinear'}
- Losgistic Regression without SMOTE: {'C': 10.0, 'l1_ratio': 0.0, 'penalty': 'l1', 'solver': 'liblinear'}

2) Random Forests

According to the Random Forest classifier, we chose to tune some hyperparameters, including {"max_depth", "min_samples_leaf", "min_samples_split", "n_estimators"}. Specifically, we compared different values for each hyperparameter: "max_depth" with a tuning set [4, 5, 6, 7, 8, 9]; "n_estimators" with [10, 20, 50, 100]; "min_samples_split" with [2, 3, 4]; "min_samples_leaf" with [1, 2, 3]. We used 5-fold cross-validation to tune these hyperparameters based on F1-score and obtained the optimal Random Forest classifiers trained with and without SMOTE. When we applied these two optimal models to the test dataset, we noticed that SMOTE only improved recall score, and did not improve accuracy, precision, and F1-score in Random Forest. Thus, we could conclude that the SMOTE technique could not efficiently improve the performance of the Random Forest classifier.

Optimal Hyperparameters:

- Random Forest with SMOTE: {'max_depth': 9, 'min_samples_leaf': 1, 'min_samples_split': 4, 'n_estimators': 100}
- Random Forest without SMOTE: {'max_depth': 9, 'min_samples_leaf': 1, 'min_samples_split': 4, 'n_estimators': 100}

AML Group 18
Ji Qi(jq2365), Yizhi Liu(yl4993), Wen Song(ws2685)

3) XGBoost

Regarding the XGBoost classifier, we tuned several hyperparameters, including “learning_rate”, “n_estimators”, “reg_alpha”, “reg_lambda”, and “subsample”. Specifically, we tested different values for each hyperparameter: “learning_rate” with a tuning set [0.05, 0.1, 0.2]; “n_estimators” with [30, 50]; “reg_alpha” with [0, 0.1, 1]; “reg_lambda” with [0, 0.1, 1]; and “subsample” with [0.6, 0.8, 1.0]. We used 5-fold cross-validation to tune these hyperparameters based on F1-score and obtained the optimal XGBoost classifiers trained with and without SMOTE. When we applied these two optimal models to the test dataset, we noticed that SMOTE only improved recall score, and did not improve accuracy, precision, and F1-score in XGBoost. Therefore, we concluded that the SMOTE technique could not efficiently improve the classification of the XGBoost classifier.

Optimal Hyperparameters:

- XGBoost with SMOTE: { 'learning_rate': 0.2, 'n_estimators': 50, 'reg_alpha': 0, 'reg_lambda': 1, 'subsample': 1.0 }
- XGBoost without SMOTE: {'learning_rate': 0.2, 'n_estimators': 30, 'reg_alpha': 0, 'reg_lambda': 0, 'subsample': 0.6}

Limitations and Future Steps

- **Gather more features:** we could find more information such as customer job, card current balance, transaction type, and so on. Gathering those information as new features might be able to improve our model performance.
- **Model interpretation with visualization tools:** we could perform SHAP and LIME to globally and locally interpret our model. For instance, SHAP produces feature importance values that show how much each feature contributes to the output of a model, both in terms of direction and magnitude. Those techniques would be helpful to propose some recommendations based on the insights (Customer Profiling) shown from the visualization graphs.

Github Submission Link

https://github.com/W4995-AML/aml-spring2023-project-Wennns