Bank Customers Churn Classification

Abstract

The goal of this project is to use predictive analysis to determine customers of the bank who are most likely to churn (close their bank account) based on the information of 10,000 bank customers (Balance, Age, Gender, etc.). The used data was found on a website called 'Kaggle' (kaggle.com). The data did not need much cleaning, so I started exploratory data analysis. After that, I split the data into train/validation, and test data to start experimenting with different models in order to find the best model to achieve the goal of this project. Finally, I communicated my work using a PowerPoint presentation to show my findings.

Design

By training the data using Classification models, we could classify the customers of the bank based on the information of that customer (Balance, Age, Gender, etc.).

Data

The dataset contains the information of 10,000 customers who are withdrawing their account from the bank.

Algorithms

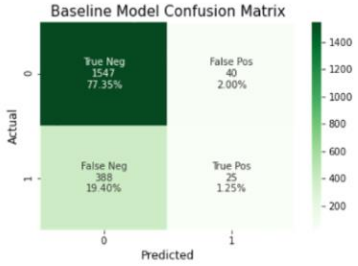
The methodology used in this project is: Problem understanding, Data validation, Data exploration, Data visualization, Feature engineering, Feature selection, Training and modeling the data.

Tools

- Numpy and Pandas for data manipulation
- Matplotlib and Seaborn for plotting

Communication

Baseline Model



- This is the confusion matrix of the baseline model, which was created using Logistic Regression. The accuracy of the baseline model is: **0.786**, which is not bad. But the F1 score is **0.104**, which is very low.

Solving class imbalance

Method	F1 Score using Logistic Regression
Oversampling	0.455
SMOTE	0.469
Undersampling	0.453
Combining Undersampling and Oversampling	0.456

The method that produce the best F1 score is Oversampling SMOTE.

Experimenting with different Models

Model	F1 Score
Logistic Regression	0.469
Decision Tree	0.505
Random Forest	0.603
XGBoost	0.599
Stacking models Using Max Voting	0.574
Stacking models Using Average Voting	0.574
Stacking models Using Weighted Voting	0.553
Stacking models Using Stacked Classifiers	0.583

The best performing model is Random Forest with F1 score equal to 0.603

Final Result

- After applying feature selection, the F1 score using Random Forest has increased from **0.603** to **0.618**