Predicting Commercial Bank Customer Retention (Churn) Rate using Supervised Machine Learning

APSTA-GE 2041: Statistical Consulting Research Seminar

Tong Jin (tj1061), Zixuan Zhou (zz2478), Zheng Tan (zt654)

December 27, 2020

GitHub Repository

Table of Contents

Introduction

If you have a bank account, you probably know that banks offer opening bonuses in the form of cashback or reward points in order to encourage potential clients. A typical opening bonus works in the way that if you agree to put a certain amount of assets in the bank, you will receive direct discounts or promotions of other services with the bank. If you are an experienced bank client, you may have been contacted by banks about retention offers. Banks give various financial offers to current clients to prevent them from leaving. There are two main reasons why banks are so obsessed with offers. First, banks need to maintain a healthy client relationship in order to keep a consistent increase in revenues. Second, the finance industry is highly competitive. Banks constantly launch new discount programs to attract competitors' clients and to stay afloat. After decades of improvement, nowadays, most commercial banks have adopted a retention-based client relationship model.

This project applies supervised machine learning techniques to develop a prediction model that proactively predicts whether or not clients of a commercial bank will stay with the bank (not churn). The data come from real transaction records of Xiamen International Bank, a major financial institute in China. The project starts with an Exploratory Data Analysis (EDA) of the data, followed by model development and implementation. The project then estimates the client's retention rate through the model. Finally, the project provides a business solution regarding how to precisely and efficiently target clients who are likely to churn.

This project also participates in an active competition on Data Castle: 2020 Financial Modeling Competition by Xiamen International Bank.

Business Understanding

As we enter a data-driven era, commercial banks are challenged with increasing competitions worldwide. The pandemic in 2020 adds an extra heavy load on banks' already overwhelmed client relationship systems. To keep revenue increases, banks need a better understanding and estimation of client's demands as well as preferences. Specifically, banks are interested in predicting client's churn rate and the change of their financial interests. Through targeting and marketing, banks can alleviate revenue losses by retaining clients who are leaving.

To effectively retain clients, banks established various retention-based business models to maintain client loyalty. These models start by launching various initial incentives to

attract potential clients. Banks use marketing techniques to precisely target clients who are interested in their products. Once the bank acquires a client, it starts to collect as much information about the client as possible. Information categories range from assets and behaviors to personal information. Then, the bank predicts whether or not the client will stay, given the collected information. In other words, banks need to predict whether or not a client will leave and embrace a competitor, or “churn”. Churn not only directly cuts revenue sources but also discourages potential clients. If a client decides to churn, he/she will likely persuade people he/she knows to not choose products or services from the bank. Banks suffer multiplied revenue losses from a single churn activity. Therefore, to prevent this, banks are willing to make a controllable sacrifice on profits by offering retention incentives to clients who may churn.

At face value, the model is effective: the bank increases marketing budgets to retain clients who are likely to churn but eventually generates enough profits to cover additional costs. However, with a closer look, the retention model incubates new issues. First, not all clients at risk will churn. Some clients may just test the water and see if they can get extra benefits. Some may stay for another year even though their measurements indicate that they are very likely to leave. Second, some clients will churn no matter what financial offers they receive. For example, if a client decides to forever leave the country, then he/she will probably close the account, regardless of any retention offers. If a client has a large amount of money and decides to churn, he/she will be less interested in financial incentives. The effect of retention offers will then become marginal.

It is important for banks to design and implement a prediction mechanism that actively estimates a client’s churn risk and proactively intervenes before he/she makes the final decision. This is the main purpose of this project: apply advanced machine learning techniques to build a prediction model.

Research Questions

This project focuses on determining the most important features of a churn prediction model. The research questions are:

1. What are the most important features of the data? Are there any interesting findings regarding the data?
2. Which algorithm has the best performance in predicting churns? How do we estimate the performance?
3. How to apply model results to solve real-world churn problems?

Data and Sample

The data come from Xiamen International Bank, a major commercial bank in China. The available sample contains daily transactions records, in multiple categories, of the third and fourth quarter of 2019. There are three data sets:

- Train set:
  - `x_train`: this is the train set. It contains all available features (predictors)
- Test set:
  - `x_test`: this is the test set. It contains the same features as `x_train`.
- Validation set:
  - `Y_train`: this is the validation set. It contains the results (also known as the label) of the `x_train`. In this case, the results are indicators of whether or not clients churn.

The train set and the validation set were randomly sampled from transaction records of the third and fourth quarters of 2019. The test set was randomly sampled from the first quarter of 2020.

The data contains 55 features. There are five feature categories in the train set and the test set:

1. "X" (8 features): this category includes information regarding client's assets at the end of each month. Features include structured deposit balance, loan balance, financial products balance, and so on.
2. "B" (7 features): this category records client's behaviors in each month. Features include number of transfers, latest transfer date/time, transfer amounts, and so on.
3. "E" (18 features): this category records client's important behaviors in each season, such as first time loan date/time, first overdue date, first online banking login date, and so on.
4. "Y" (2 features): this category contains client's deposits in each month.
5. "I" (20 features): this category contains client's information (trivias) in each season. Features include gender, age, occupation, education level, and so on.

The test set has two columns:

1. "Cust_no": customer's unique ID

2. “label”: whether or not a customer churns. There are three possible values:
    a. 1: indicates churn.
    b. -1: indicates not churn
    c. 0: indicates no preference.

(A full codebook is available here on the GitHub repository.)

Dimensions

The raw train set contains 465,441 rows and 56 columns (1 index and 55 features). The train validation set contains 145,296 rows and 56 columns. The test set contains 76,722 rows and 1 index column.

Data Preprocessing

Validation Set

We started from examining the validation set, **`y_train`**, because it contains the only source for validation, the labels. The validation set is formatted in two data files, y_Q3 and y_Q4, separated by quarters. We first validated that all customer IDs are unique. We then confirmed that there are duplicates in both files. To deal with this, we followed the default quarter separation and applied two independent data preprocessing processes.

Out of 145,296 rows of data, the third quarter file contains 69,126 rows (47.58%) and the fourth quarter file contains 76,170 rows. There are 62,397 (42.94%) duplicate records. There are also 40,090 completely identical records (same ID, same label). We confirmed that two samples are heavily overlapping. 22,307 (32.27%) clients changed their churn preference from the third quarter to the fourth.

Train Set

Based on the quarterly-separated validation set, we merged the raw train set accordingly. For each quarter, we dropped duplicate records except for the last occurrence. We kept cross-quarter duplicates to maximize retained information. For example, if a client has three identical records in the third quarter, we only keep the last one. However, if a client has two identical records, one in September (the third quarter) and the other in November (the fourth quarter), we keep both of them.

After preprocessing, we have three sets:

1. “X”, the train set, contains 145,296 rows and 56 columns (55 features and 1 index)
2. “y”, the validation set, contains 145,296 rows and 1 label column.
3. “X_true”, the test set, contains 76,722 rows and 55 columns (features).

(The script of data preprocessing is available here on the GitHub repository.)

Data Processing

Feature Engineering

After data preprocessing, we confirmed that both train and validation sets have the same length: 145,296. We also confirmed that the test set has 76,722 records. During data processing, we found multiple columns containing missing values. The percentage of missing values in each column ranges from 0.005% to 100%. We dropped columns containing large portions (>70%) of missing values. For columns with small portions of missing values, we replaced them based on characteristics. For example, for date-specific columns, such as the first online banking login date, we selected the latest date to replace missing values. For quantitative columns, such as the maximum amount transferred out, we replaced them with zero. For the gender column, we replaced missing value with “female”. For the occupation column, we replaced missing values with “Unknown”.

We then inspected features. Given that there are 55 features in the dataset, we need to apply feature reduction procedures in order to reduce the dimensionality of the data. After mining into the data, we determined that the following columns contain meaningless information and, therefore, we dropped them:

- Constellation. We don’t believe constellations can alter customer behaviors.
- Field description. This column only contains 1 different value.
- QR code recipient.

After feature selection, we reduced the feature size from 55 to 42. We then dummy coded categorical columns using the one-hot encoding method in order to prevent machine learning algorithms from interpreting hierarchy in categorical columns. For date-time columns, we first converted all string-like inputs as data time format. We then created dummy columns for each year, month and day of each date-time column. Finally, we dropped the original columns. After dummy coding, we have 77 features.

Labels

For the label column, out of 145,296 records, 92,818 (63.88%) of them are labeled as 1, 30,237 (20.82%) are labeled as 0, and 22,241 (15.31%) are labeled as -1.

After data processing, we have three sets:

4. "X_train", the train set, contains 145,296 rows and 78 columns (77 features and 1 index)
5. "y_train", the validation set, contains 145,296 rows and 1 label column.
6. "X_test", the test set, contains 76,722 rows and 77 columns (features).

(The script of data processing is available here on the GitHub repository.)

Measurements

The evaluation metrics we used include: (1) Receiver Operating Characteristic (ROC); (2) Area Under the Curve (AUC). The ROC curve visualizes the ratio between true positive rate and false positive rate. The AUC is the area under the ROC curve.

Research Design

The strategy of this project is to train supervised machine learning models with different algorithms using the provided train set. During the training process, we improve model performance through recursively validating model prediction results with the true results in the validation set. After training, we fit the model to the test set and generate actual prediction results.

Analytic Methods

There are various data mining algorithms we can apply for this type of problem. In lieu of the logistic regression, naïve Bayes, neural network, and random forest approaches, Amjad et. al. used a Scatter Search and K-Nearest Neighbors algorithm (Amjad, Gharehchopogh, 2019). Wang et. al. adopted the Support Vector Machine algorithm (Wang, Yu, Liu, 2005). Shi et. al. took the decision trees ensemble (Wang, Ma, Weng, Qiao, 2012).

The advantage of black-box algorithms, such as random forest and neural networks, is their ability to efficiently predict based on the learning of big size data. However, these

models are highly complex and are hard to interpret and translate. Logistic regression and Naive Bayes, are more intuitive. Especially Naive Bayes, it has been proved to be powerful yet simple. With the potential of being used to train data based on a per-user characteristics, Naive Bayes can also lower false positive rate effectively due to the probability estimation ability. The limitation of logistic regression models, however, lie in the assumption of linearity between the target variable and features. As a result, when selecting models, we decided to use the following algorithms:

- Logistic Regression with Elastic Net as the baseline model
- Random Forest
- Gradient Boosting Machine

For each model, we applied grid search and randomized search to fine tune the hyperparameters. We also used 5-fold cross validation to increase model accuracy.

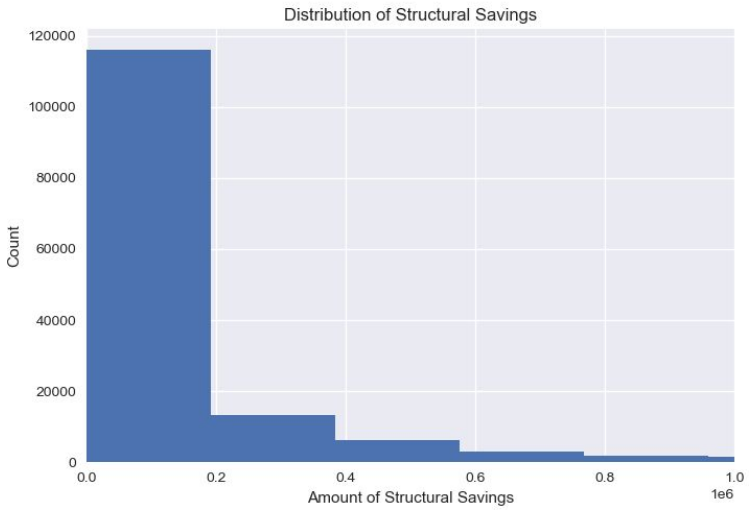
Before fitting the model, we down sampled the data in order to reduce computational burdens. We achieved this by keeping all labels marked as 0 and -1 while randomly sampling a reduced amount of label 1.

Results

In this section, we discussed the results from the EDA. We also covered the main results from machine learning models.

Descriptive Results

The processed training set contains 145,296 rows and 77 features. For clients' assets, the average structured deposit balance is 21,001.53 (in RMB). The average amount of savings is 1,564.47. The average checking balance is 3,845.89. The distribution of these features is highly right-skewed, given the fact that there are people who possess a large amount of assets.



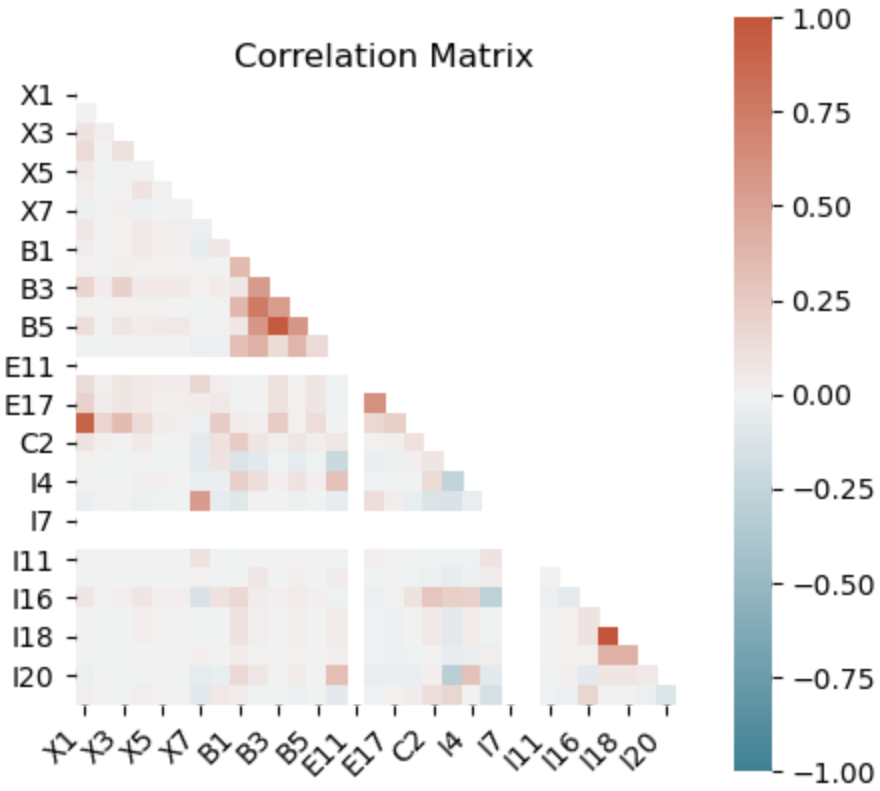
For the transaction data, on average, clients made 3 transfers per month. The average transfer-in amount is 23,729. Similar to the saving distribution, the distribution of transfer is also highly right skewed.

The average of monthly assets is 337,523.7. The distribution is also highly right-skewed.

For client's personal information, out of 145,926 observations, 78,912 are female. 66,320 are male. The average age is 49, with a standard deviation of 15.95. 71,525 are normal clients. 54,142 of them are gold. 17,996 are platinum. 1,633 are diamonds (top-tier). 6,422 (4.42%) are marked as employees. 27,150 of them work in the business industry. Only 4.86% of them hold a bachelor (or above) degree.

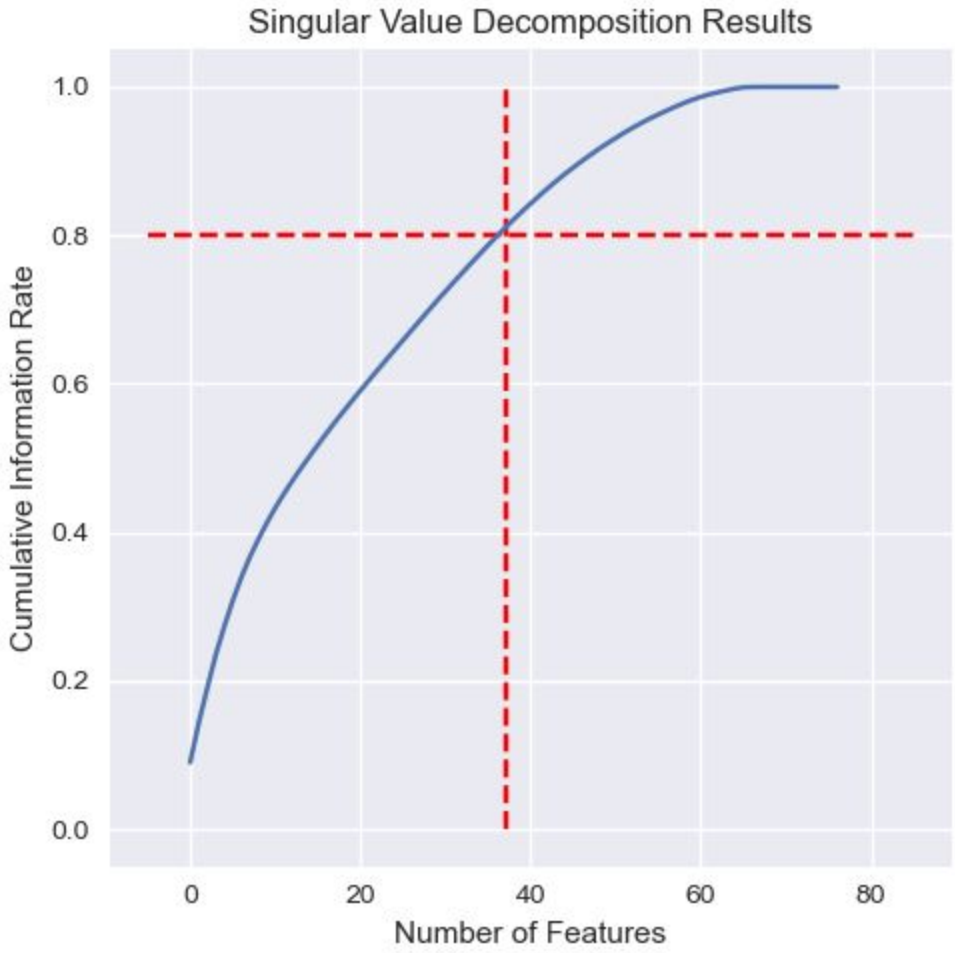
Correlations

We further examined the correlation between feature columns. Results indicate that the transfer-in amount is highly correlated with the transfer-out amount. This makes sense because clients are likely to transfer a similar amount of money for a certain period of time.


Correlation Matrix
X1
X3
X5
X7
B1
B3
B5
E11
E17
C2
I4
I7
I11
I16
I18
I20
1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00

Feature Importance

To reduce the feature size, we applied singular value decomposition (SVD) and selected the top 80% of features.



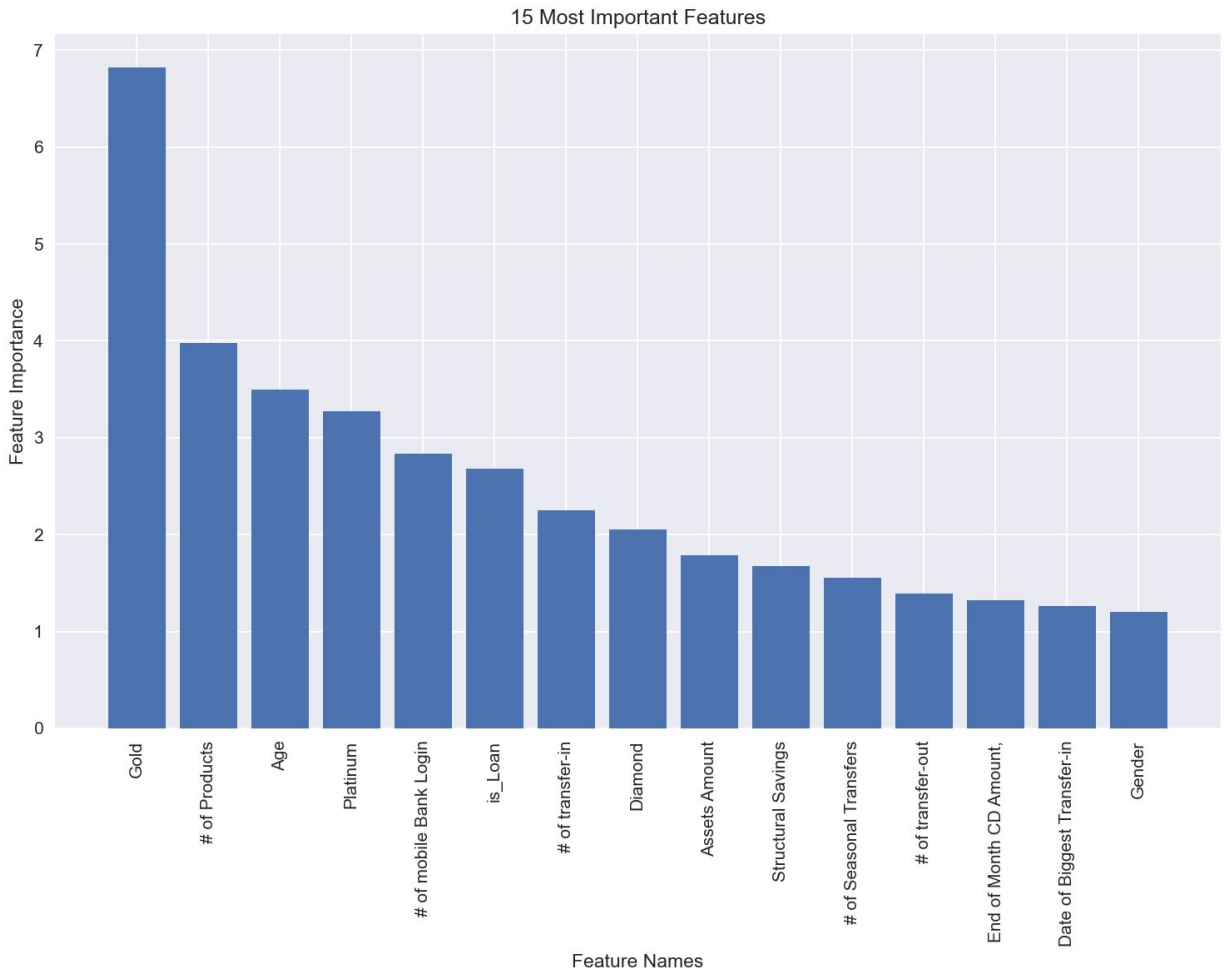
From the SVD, we selected the top 37 features to keep 80% of the information from the full data.

We also applied downsampling to continue reducing data dimensions. After downsampling, the model-ready data contain 111,772 rows with 37 features.

Main Results

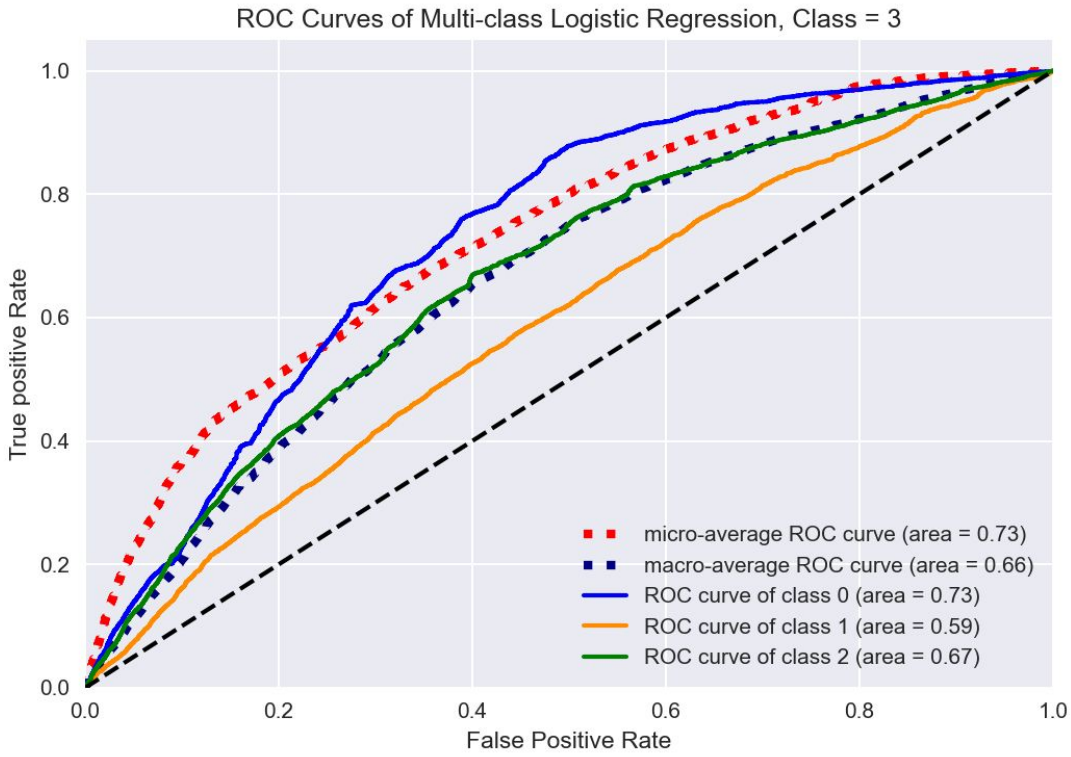
Top 15 Features

We first summarized the top 15 most important features in the data using Truncated SVD.



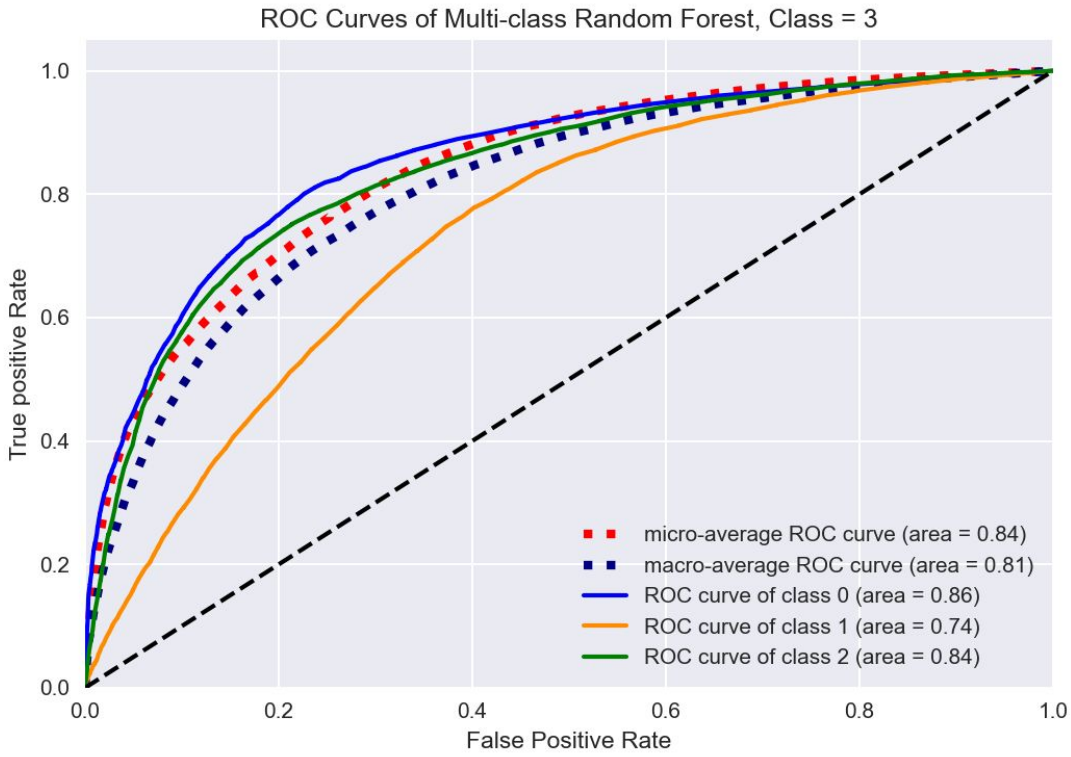
Results indicate that client loyalty level is one of the most important features in predicting churns. Clients in “gold”, “platinum” or “diamond” are easier to be tagged, as compared to normal clients. The number of products associated with the bank is another important feature. Additionally, transfer frequency is a major category in prediction. Furthermore, age and gender are two personal information categories that contain important prediction information.

Logistic Regression (Baseline)



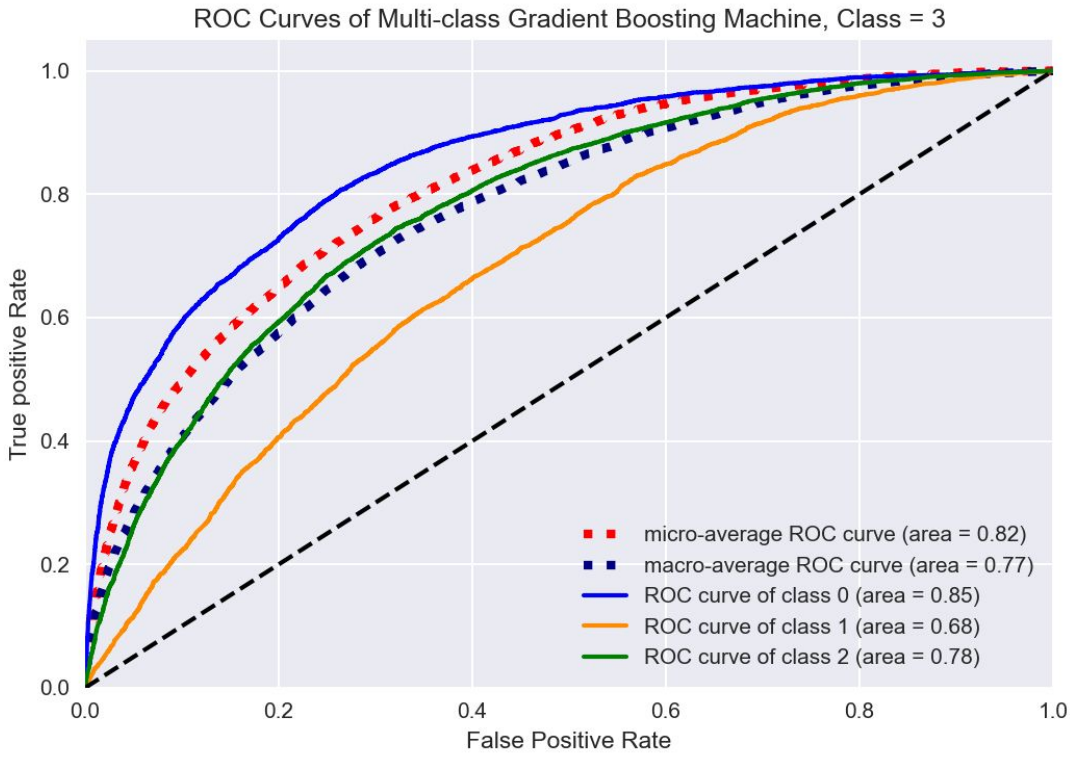
The above plot displays the result of the baseline logistic regression model. The blue solid line is the ROC curve of class 0 (labelled as no churn). It's AUC score is 0.73. The orange solid line is the ROC curve of class 1 (labelled as no preference). It's AUC is 0.59. The green solid line represents the ROC of class 2 (labelled as churn). It has an AUC score of 0.67. The red dashed line is the micro-average ROC curve. The navy dashed line is the macro-average ROC curve. The macro-average is calculated by independently computing scores for each class and then take the average. The micro-average is computed by aggregating the contributions of all classes. Given that the classes in our data are imbalanced, we referred to the micro-average score for our final result. In this case, the Logistic Regression baseline model achieved a 0.73 prediction accuracy.

Random Forest



As a highly complicated algorithm, Random Forest is known for its performance when dealing with complex problems. In our case, the prediction using the random forest algorithm achieved a much higher accuracy than the prediction using logistic regression. The micro-average AUC score is 0.84. The best performed class achieved an AUC score of 0.86.

Gradient Boosting Machine



As another strong ensemble algorithm, Gradient Boosting Machine uses a similar mechanism as random forest but with different methods. Results indicate that the micro-average AUC is 0.82. The best performed class achieved an AUC score of 0.85.

Here is a summary table of all model performances. The random forest model out-performed the rest two in all categories.

Models	Micro-Avg	Macro-Avg	Class 0	Class 1	Class 2
Log Reg	0.73	0.66	0.73	0.59	0.67
RF	**0.84**	**0.81**	**0.86**	**0.74**	**0.84**
GBM	0.82	0.77	0.85	0.68	0.78

Conclusion

The result of this project proves that modern machine learning algorithms can reach a high prediction accuracy. To conclude, in this project, we explored all the features in the data and selected the top 15 most important ones. Among them, client loyalty and transfer frequency are two of the crucial features. After applying machine learning models, we managed to achieve a high accuracy using the random forest algorithm.

Since this study is still at the alpha stage, it contains many potentials. For example, by expanding the data samples, we can reduce model variance and get stable results over time. We can also fine tune the hyperparameters of machine learning algorithms in order to prevent overfitting. Moreover, we can conduct more detailed data mining and EDA so that the data can be processed more precisely. This will reduce model bias and, consequently, increase the accuracy.

References

1. Amjad, S., & Soleimanian Gharehchopogh, F. (2019, August 01). A Novel Hybrid Approach for Email Spam Detection based on Scatter Search Algorithm and K-Nearest Neighbors. Retrieved from http://journals.srbiau.ac.ir/article_14397.html
2. Attack landscape update: Facebook phishing, COVID-19 spam, and more - F-Secure Blog. (2020, September 17). Retrieved from https://blog.f-secure.com/attack-landscape-h1-2020/
3. Dada, E. G., Bassi, J. S., Chiroma, H., Abdulhamid, S. M., Adetunmbi, A. O., & Ajibuwa, O. E. (2019, June 10). Machine learning for email spam filtering: Review, approaches and open research problems. Retrieved from https://www.sciencedirect.com/science/article/pii/S2405844018353404
4. Buerck, J. P., Fisher, J. E., Mathieu, R. G. (2011, October 24). Ethical dimensions of spam.Retrieved from https://www.inderscienceonline.com/doi/abs/10.1504/IJEB.2011.043255
5. Gaikwad, B. U., Halkarnikar, P., & Student, M. T. (2014). Random Forest Technique for E-mail Classification: Semantic Scholar. Retrieved from https://www.semanticscholar.org/paper/Random-Forest-Technique-for-E-mail-Classification-Gaikwad-Halkarnikar/e0c37ec1359268e4431e49ee3729227489bd7ce4
6. Spam Email Classification using Decision Tree Ensemble Retrieved from http://jof-cis.com/article/spam-email-classification-using-decision-tree-ensemble/
7. Metsis, Vangelis, Androutsopoulos, Ion, Paliouras Georgios (2006). Spam Filtering with Naive Bayes -- Which Naive Bayes?. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.5542
8. Wang, H., Yu, Y., & Liu, Z. (2005, December 06). SVM Classifier Incorporating Feature Selection Using GA for Spam Detection. Retrieved from https://link.springer.com/chapter/10.1007/11596356_113
9. Yang Song Department of Computer Science and Engineering. (2009, August 01). Better Naive Bayes classification for high-precision spam detection. Retrieved from https://dl.acm.org/doi/10.5555/1568514.1568517
10. S, T. (2020, May 13). What Does it Mean to Deploy A Machine Learning Model? Retrieved December 05, 2020, from https://towardsdatascience.com/what-does-it-mean-to-deploy-a-machine-learning-model-dddb983ac416