VIB Datarathon

SAY CŨNG THÀNH ĐÚNG TEAM

Data Track - Round 1

Team Members:
Nguyen Van Duc
Bui Thi Ngoc Tram
Pham Van Ngoan
Nguyen Hong Minh Nhat

May 25, 2021

Contents

1 Introduction

With the purpose of improving customers' experience on myVIB application, it is important to predict the customers who are potential to be churned the application. This customer group will provide us a set of behaviours which can help to find the weakness points of the current platform and even the current business and marketing campaigns. In this report, we aim to propose a reasonable definition of churned customers and also a strategy for early detecting these customers based on the provided VIB customers data. Divide the overall goal into subgoals with corresponding relevant questions In our attempt to solve aforementioned problem, we divide it into 2 sub-goals, which are (1) identify appropriate definition of churned customers and (2) identify rational factors that explain such behavior. Therefore, we employ following sub-questions which this report will solve it case by case:

- What are appropriate definition candidates for identifying the customers who leave our mobile application?
- Which data strategies do we intend to get first data insights?
- Which data insights are valuable to VIB mobile banking?
- Which is the best functionality of data mining can be used to solve the "big" problem? For each functionality, which algorithm do we intend to use?
- What are our evaluation strategies? Based on the analytical result, are there any practical/ feasible recommendations?
- ...

Based on our theoretical researches and practical background, we suggest some possible definitions as our candidates as follows.

1. **Definition 1:** Customers who don't use MyVIB (no activities) from X days since e-bank register date.
2. **Definition 2:** Customers who have off time (from register E Banking to the last activity date) greater than Y days.
3. **Definition 3:** Customers who don't use MyVIB from X days since e-bank register date and/or have off time greater than Y days

2 Data Overview

Before choosing any definitions also proposing any strategy for processing data or features engineering, we would like to take a look at all the table data and provide some general views for each data table as below.

2.1 Customer

There are 290223 records including 290223 unique customer numbers which were created in the period from 01-01-2019 to 31-12-2019. As observing, there are several missing values in customer data and they are mostly in columns VERIFY_METHOD, SMS, EB_REGISTER_CHANNEL, CLIENT_SEX and DATE_OF_BIRTH.

EB_REGISTER_CHANEL and IB_REGISTER_DATE are null which mean customers did not register E-banking and myVIB, so these customers should not be considered.

After removing the above customer, we get 158975 customers who register internet banking. The below pie chart shows the proportion of customers opening e-bank account based on their registered channel. We observed that customers mostly go to a branch to register an account (about 73.7%).

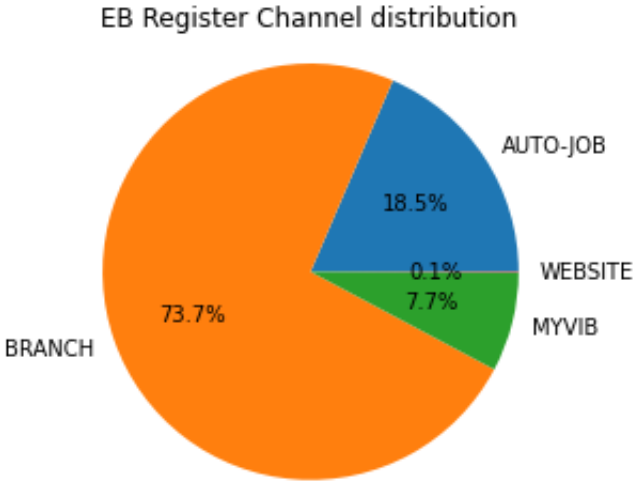


Figure 1: EB Register Channel Proportion Distribution

For customers who have missing or invalid values at DATE_OF_BIRTH, CLIENT_SEX and VERIFY_METHOD we will consider them as noises and remove from the table in order to analyse these features.

We suppose to compute AGE to analyse easier. According to age values, we observed that the average age is 33 and 75% customers are less that 39 years old. However, we will divide age into age range to explore edge cases and get more meaning.

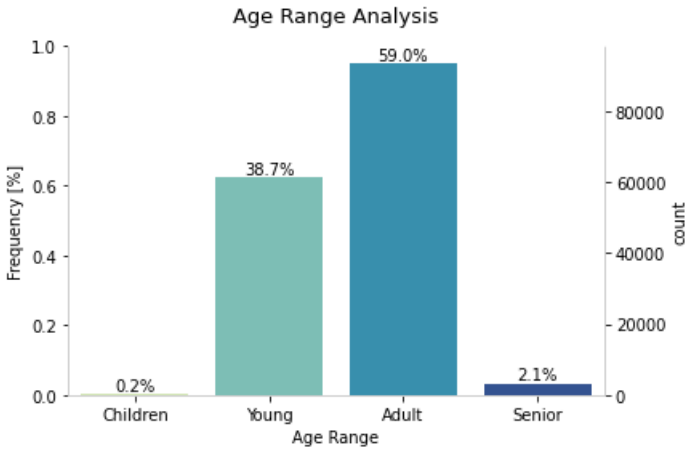


Figure 2: Histogram of age range

We split age into 4 groups for easier observing. In detail:

- **Children:** customers are smaller than 18 years old.
- **Young:** customers who are from 18 to 30 years old.
- **Adult:** customers are from 30 to 59 years old.
- **Senior:** who are older than 59 years old.

We observed about 364 customers, who have registered Internet Banking, are smaller than 18 years old. There exists a customer who is just 2 years old, we assume that this customer probably inherits banking card from parents.

Let us take a look at the distribution of genders. The figure below show that male customers get bigger proportion of total customers (about 58.2%).

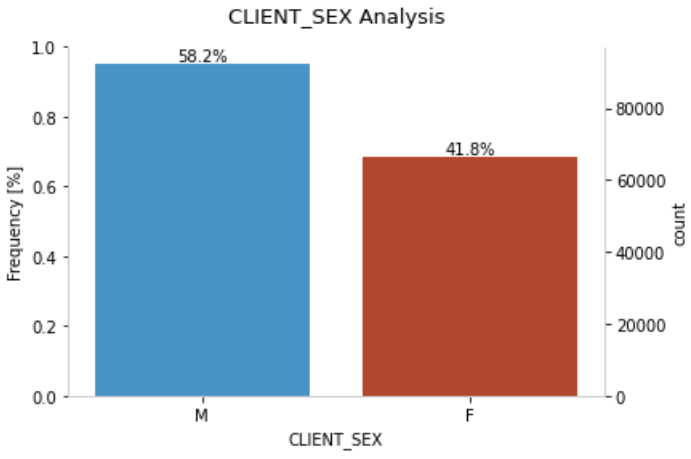


Figure 3: Proportion of clients genders

On the other hand, there is a mismatch between CLIENT_CREATE_DATE and IB_REGISTER_DATE. In detail, the period of register date lasts from 02-01-2019 to 10-03-2021, whether the

period of create date is from 01-01-2019 to 31-12-2019. For getting better understanding, we consider the gap days between the day that bank account created and the day the e-bank account have registered.

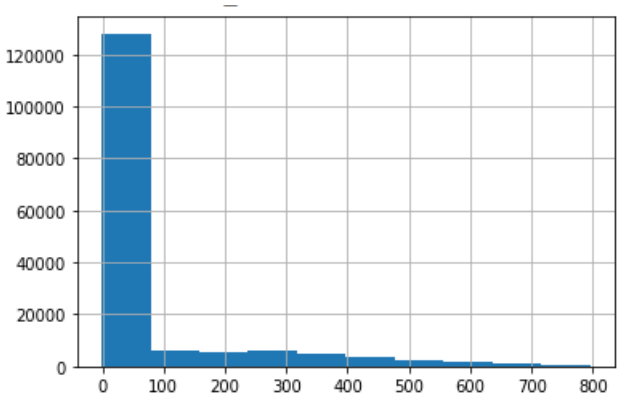
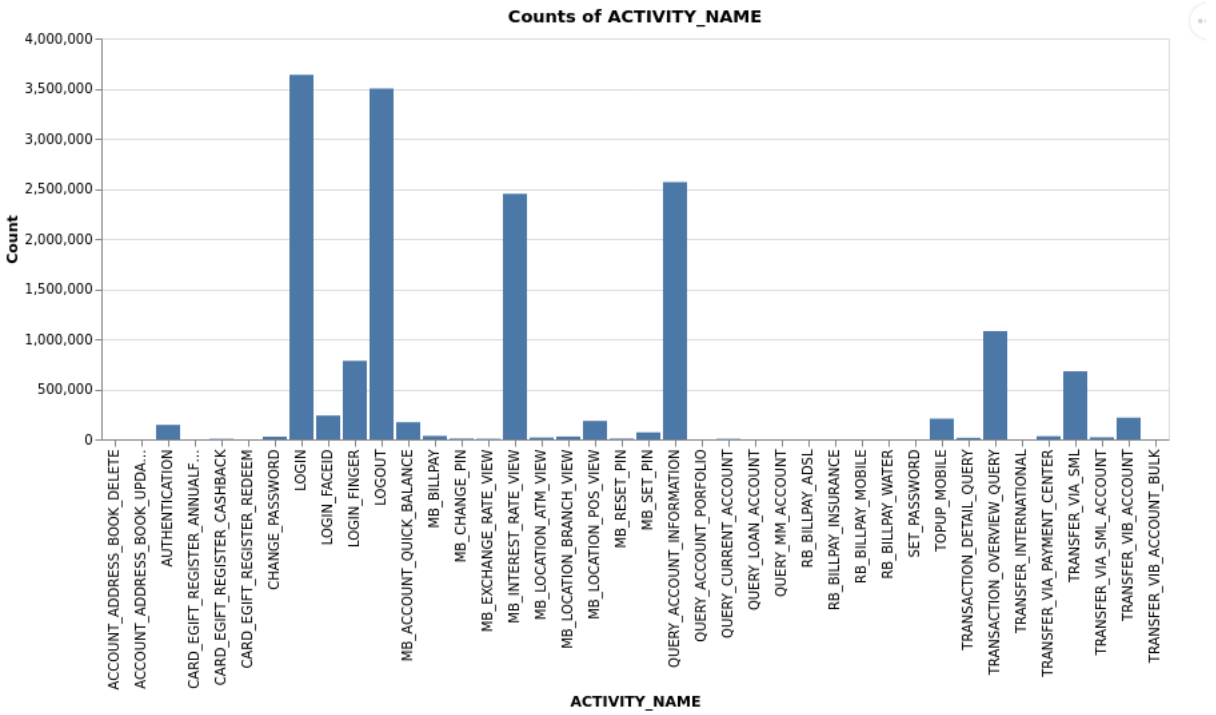


Figure 4: Registered gap days distribution

The figure above shows that most of all customers will register internet banking account in the same day they create their bank account (about 50.3%). There are 77 customers who registered internet banking before create banking account, it is probably because of the system.

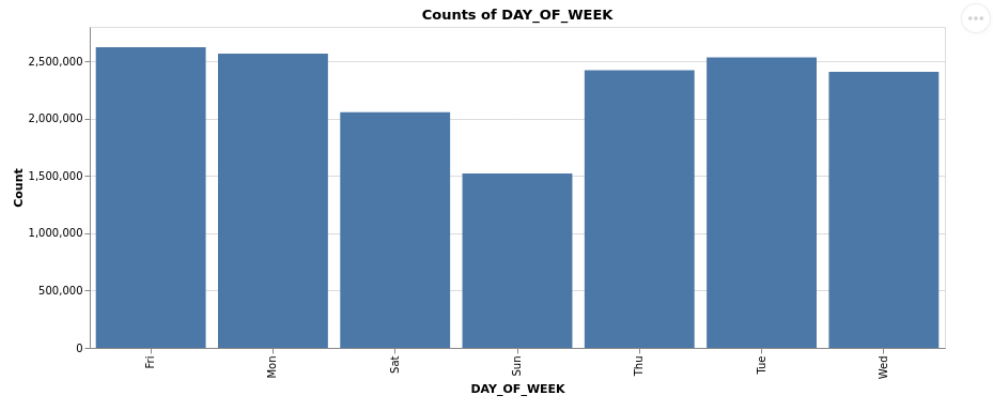
2.2 MyVIB Activity

This table recorded customers' activities from 01-01-2019 to 31-12-2019 on myVIB platform. According to our observation, this table has no missing values. The chart below show the number of customers based on its name.



It is obvious that LOG_IN and LOG_OUT are the most frequent activities. Besides, there are other activities also appear frequently such as QUERRY_ACCOUNT_INFORMATION or INTEREST_RATE_VIEW. They are all normal activities on banking application of customers.

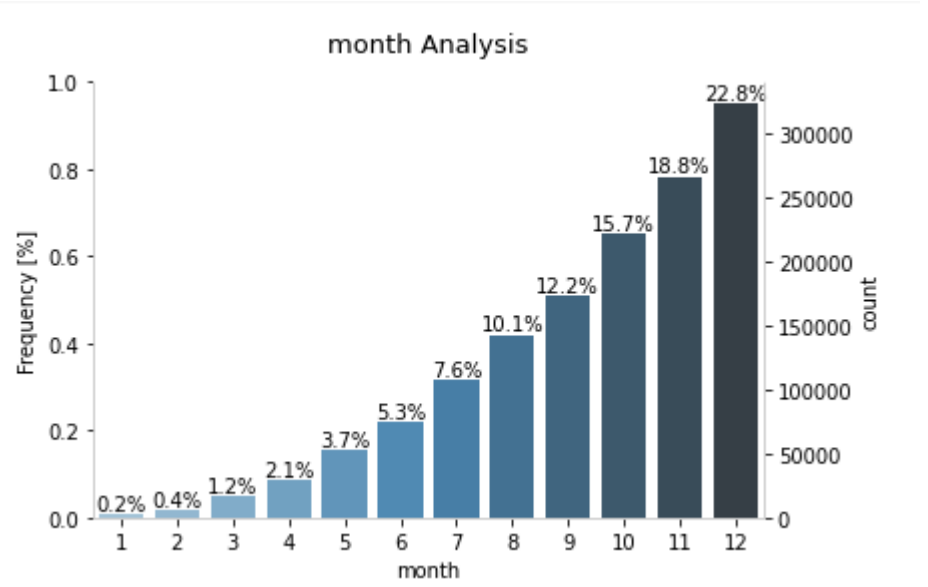
Take a look at number of activities by days of week, we aware that customers have trend to do activities on work days instead of weekend.



According to activity table, there are **77741 customers** who were recorded activities on application from 01-01-2019 to 31-12-2019, whether the number of customers who registered e-bank account in customer table above. Therefore, we check IB_REGISTER_DATE of CUSTOMER_NUMBER in activity table on customer table for sure that all of them have registered e-bank account. However, there is a customer who does not have register date date but have an activity on app recorded. It is an abnormal case, however, this customer just has 1 activity and does not have any transactions, therefore we consider as a noise and remove it. Finally, we will focus on analysing behaviours of **77740 customers** who registered e-bank account and have recorded activities on app.

2.3 Transaction

In this table, there are 1418030 transactions were recorded from 02-01-2019 to 31-12-2019. The figure below shows the proportion of number transactions in each month compare with the whole year.



It is easy to observe that the number of recorded transactions have trend to increase each month because the number of customers are increase. Therefore, we consider the distribution of average transaction numbers per month as the figure below.

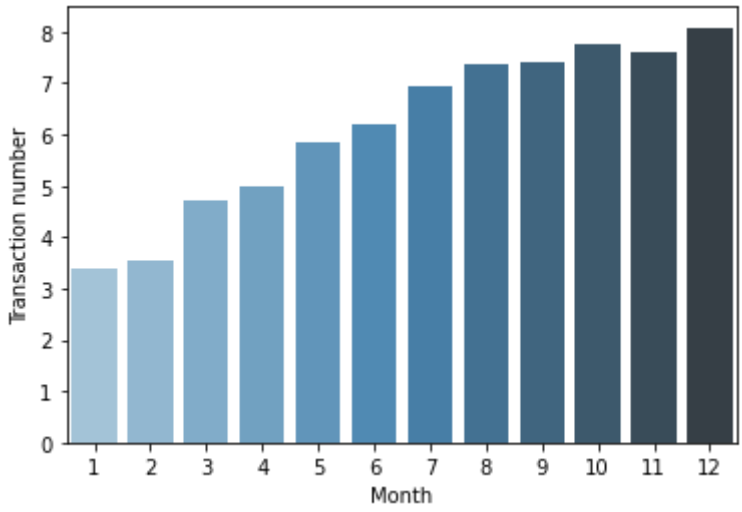


Figure 5: Average Transactions Per Month

We observed that, averagely, customers do more transactions in the last months of the year than the early months. Especially in period from April 2019 to August 2019, the average transaction numbers per month of each customers increase significantly by month, there probably had a promotion in this period that attracts customers make transactions on app or simply that the demand of transactions is increase in this period.

There is an extraordinary point in this table that there exists until 931 transactions with value 1 VND, so that we would like to raise a question here for these transactions. Besides, after comparing with activity table, we got some customers have transactions but have no activity recorded. For these customers, they did transactions on a third-platform, not in myVIB platform, so we will drop them also.

2.4 Deposit

This table recorded deposit information of customers from 31-01-2019 to 31-12-2019. In our considering data, almost customers have a deposit, however, there are 2480 customers do not have any deposits. We showed below a bar chart of average of AVG_CA_BALANCE per month of all customers.

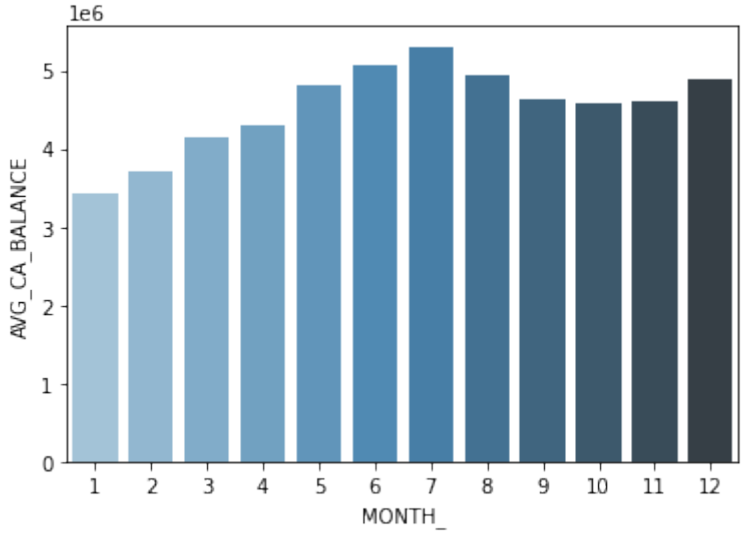


Figure 6: Average Deposit of customers per month

We observed that deposit amount of customers tend to increase rapidly in first half of the year, slightly decrease in August and stable in last months of the year.

2.5 Lending

Based on lending table, we observed that the average loan amount of each customer in each month last from 1VND to 300 billion VND. It is weird to lend 1VND however, there is just 1 customer and this customer did not register e-bank account also. Therefore, we will not consider to him. Similar to deposit table, we show below the average of AVG_LOAN_AMOUNT in each month of all the customers.

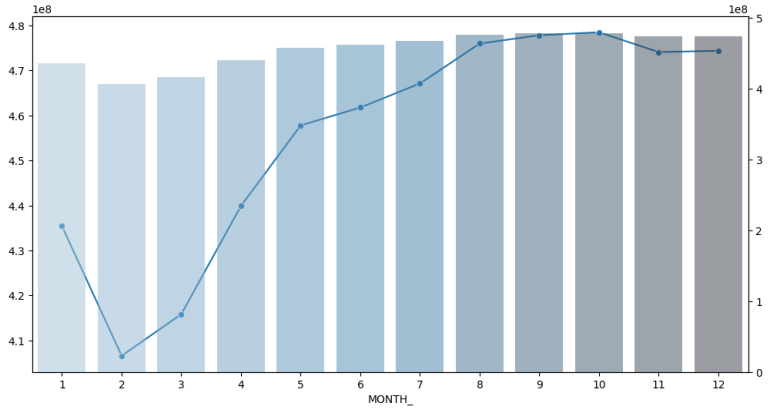


Figure 7: Average loan of customers per month

2.6 Card

Card table recorded debit card and credit card information of customers each month from their register date to 31-12-2019. From the data, we observed that each customer have at least a credit or debit card. There are some customers who have until 19 debit card or 13 credit card. It is a little unusual, so that, we would like to generally review these customers. For customers who have many credit card, they did not register e-bank account so we will not consider them. On the other hand, customer **390999**, who owns 19 debit card, has e-bank account and have a lot of activities on app. It is easy to understand because his demand for query data of each card is high. However, behaviour of this customer about creating a lot of credit cards (as showed in Figure. 8) each month is not normal.

```
1  card[card.CUSTOMER_NUMBER == 390999]
```

	MONTH	COUNT_CREDITCARD	COUNT_DEBITCARD	CUSTOMER_NUMBER
303162	10/31/2019	0	18	390999
303163	6/30/2019	0	9	390999
303164	9/30/2019	0	15	390999
303165	8/31/2019	0	12	390999
303166	5/31/2019	0	6	390999
303167	12/31/2019	0	19	390999
303168	7/31/2019	0	9	390999
303169	11/30/2019	0	19	390999

Figure 8: Record of card information of a customer who has unusual behaviours.

3 Analysis Strategy

3.1 Customers segmentation

Customer segmentation is similarly the process of dividing an organisation's customer bases into different sections or segments based on various customer attributes. The process of customer segmentation is based on the premise of finding differences among the customers' behaviour and patterns.

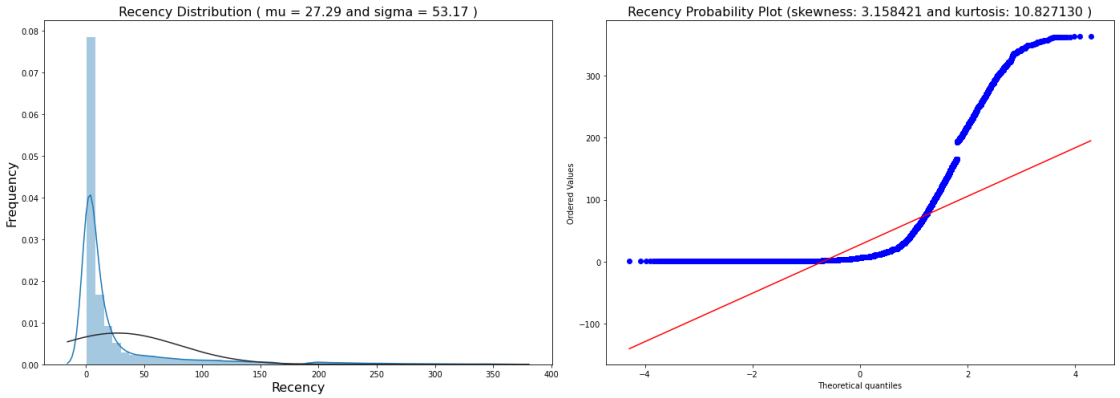
There are many objectives and benefits behind the motivation for customer segmentation such as: get higher revenue, understand customers, target marketing, optimal product placement, etc. . . . For this problem, we expect to find the customer segment who are potential to leave the application and analyse their behaviours.

Based on our assumptions about churned definitions in section II and overview all the data, we propose a customer segmentation method, which is efficient for segment customers data by combining three features in their behaviours: recency, frequency and monetary. In detail, the customers will be segmented based on:

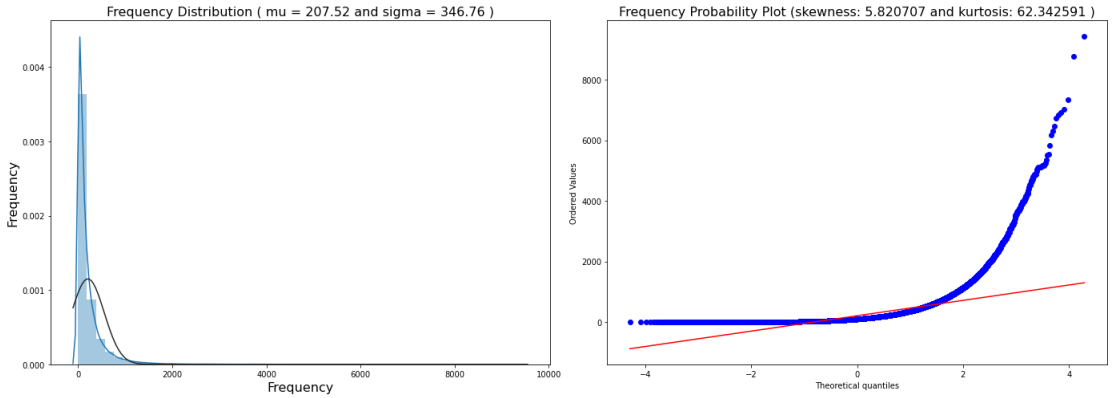
- **Recency:** The value of how recently a customer has an activity on the application.
- **Frequency:** How frequent the customer has activities on the application.
- **Monetary:** The value of all the customer's transactions.

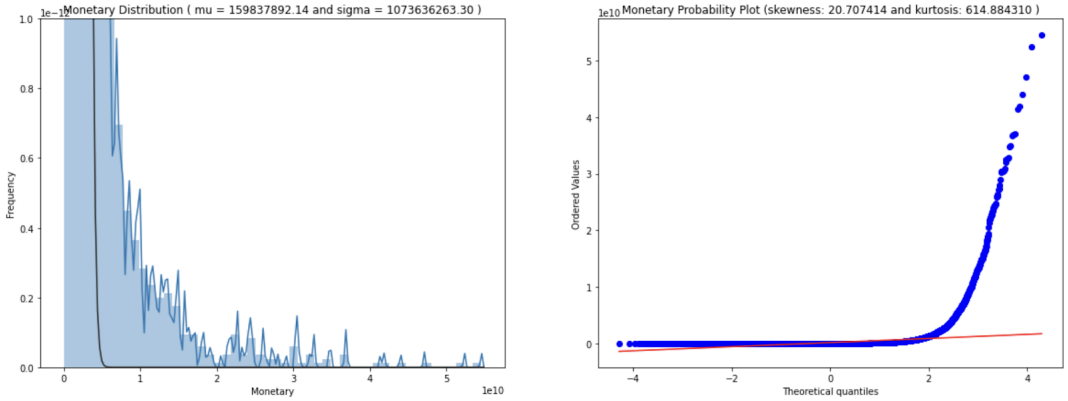
All three of these measures have proven to be effective predictors of a customer's willingness to engage with the application and even with the bank. We will use table ACTIVITY for calculating recency and frequency and table TRANSACTION for calculating monetary value.

With this data, we will compute recency value by getting the number of days between the last ACTIVITY_DATE of the customer and the last day of all the activities (31-12-2019). The distribution of recency is shown in the figure below.



In like manner, we can compute frequency value equals to the total number of customer's activities in a year and monetary value is the total transaction values of a customer. The distributions of computed frequency and monetary values are shown below.

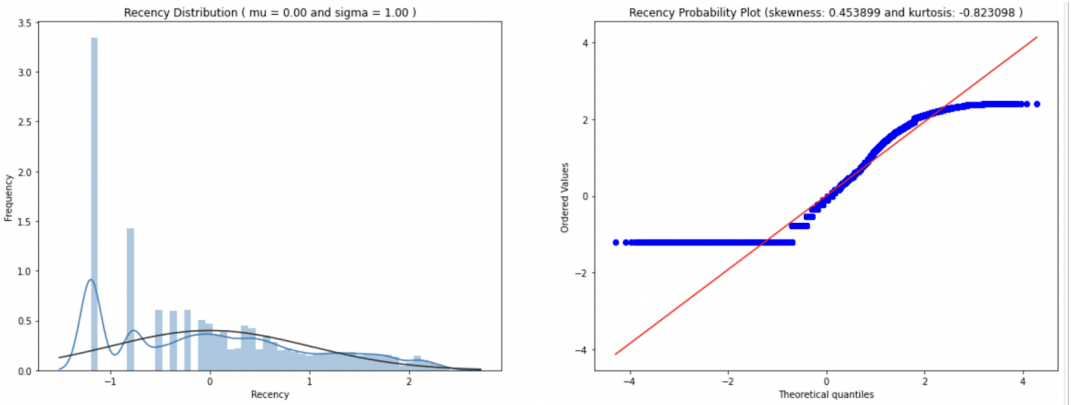




From these figures above, we observed that the values of frequency, recency and monetary are not aligned to normal distribution. Therefore, segmentation by these values will lead to several noises. In order to achieve a better effectiveness of clustering algorithms we will apply later, it is necessary to do some normalisation steps for these values to convert them to standard normal distribution.

3.2 Data Preprocessing

We need to segment current customer set with RFM values with proper strategies, so we chose K-Means Clustering Algorithm to do customer segmentation. First we normalise data by apply logarithmic function to Recency, Frequency and Monetary values. Then, we normalise them with z-score normalisation. We then have the following results:



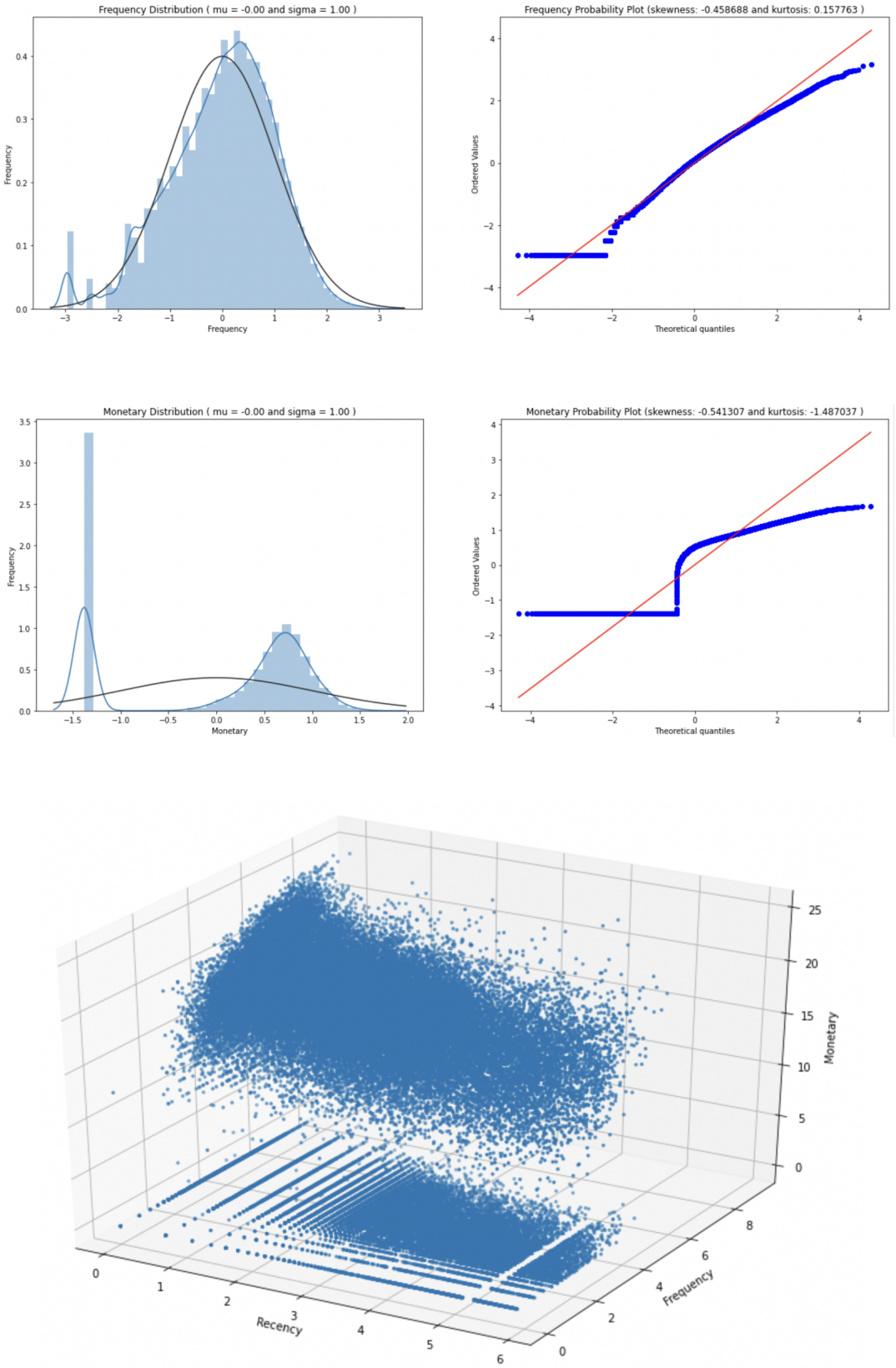
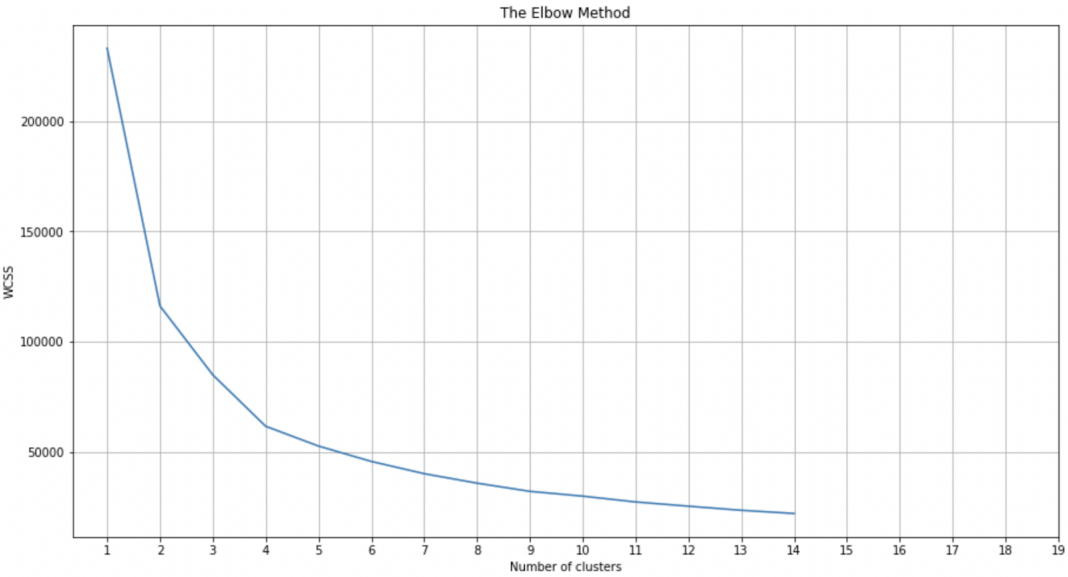
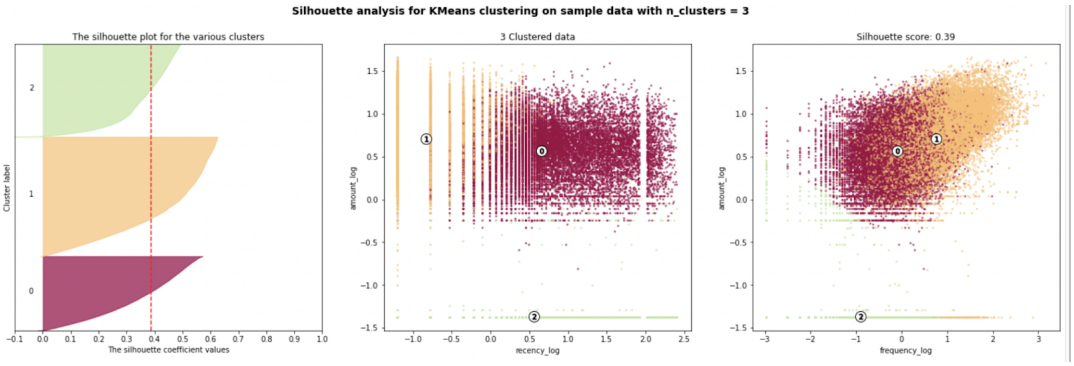


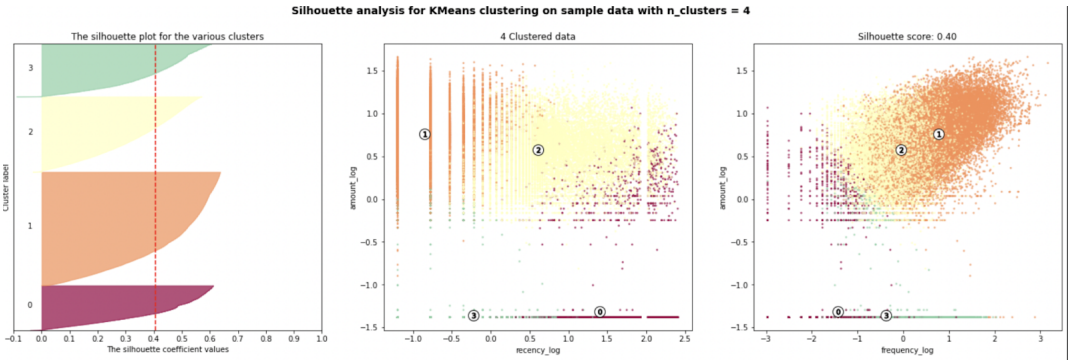
Figure 9: Data points on 3-D space: Recency, Frequency and Monetary

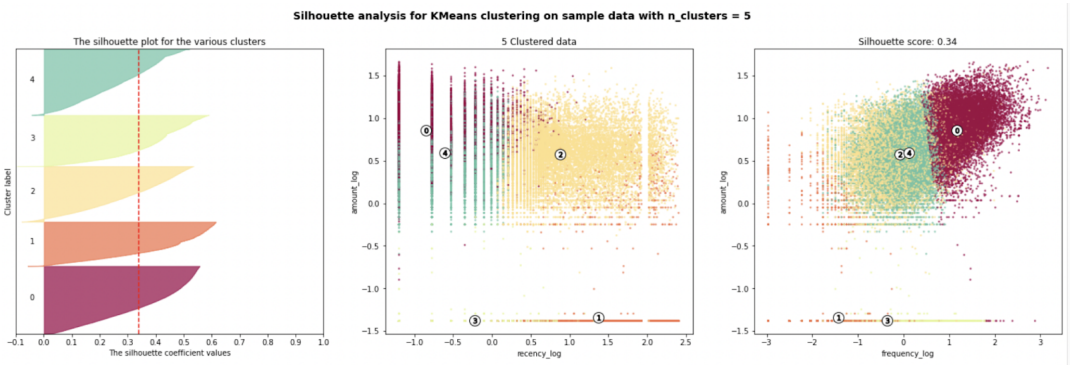
K-Means clustering algorithm requires a proper hyperparameter k (clusters). We used Elbow Method to find the optimum k with k in range 1-14



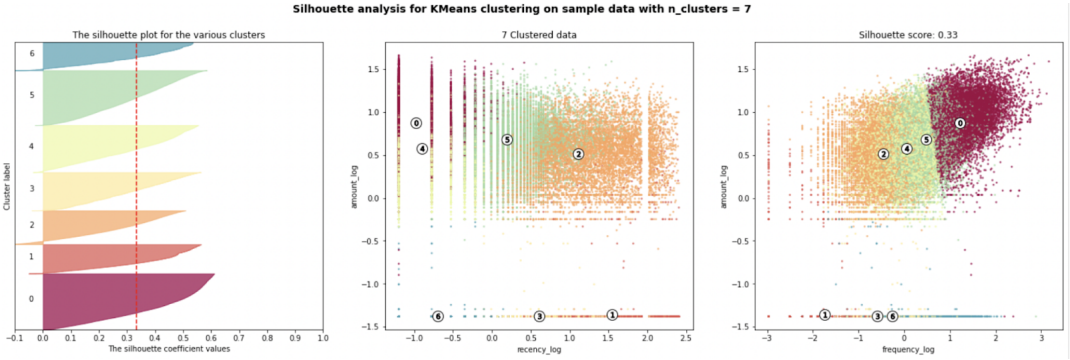
As we can see in the Elbow Curve above, the optimum k value is 4. We want to make sure that k = 4 is the best answer, so we try calculating Silhouette score, with k in range 3-9, and get the following results:

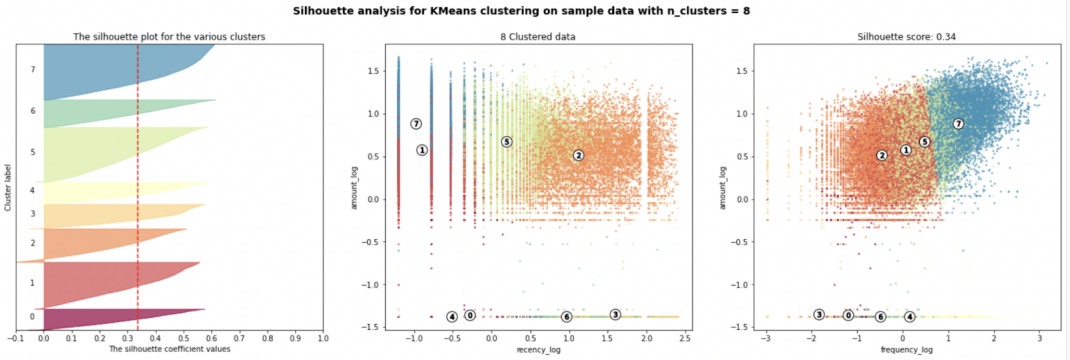


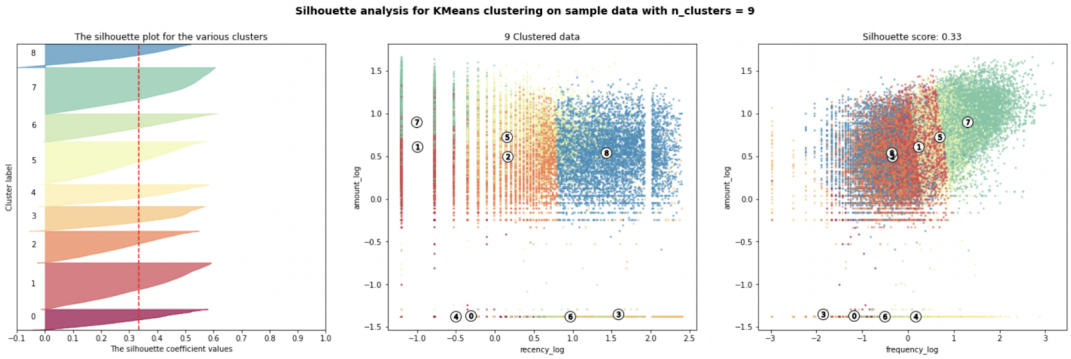



Silhouette analysis for KMeans clustering on sample data with n_clusters = 5
The silhouette plot for the various clusters
Cluster label
The silhouette coefficient values
5 Clustered data
amount_log
recency_log
Silhouette score: 0.34
amount_log
frequency_log


Silhouette analysis for KMeans clustering on sample data with n_clusters = 6
The silhouette plot for the various clusters
Cluster label
The silhouette coefficient values
6 Clustered data
amount_log
recency_log
Silhouette score: 0.35
amount_log
frequency_log


Silhouette analysis for KMeans clustering on sample data with n_clusters = 7
The silhouette plot for the various clusters
Cluster label
The silhouette coefficient values
7 Clustered data
amount_log
recency_log
Silhouette score: 0.33
amount_log
frequency_log


Silhouette analysis for KMeans clustering on sample data with n_clusters = 8
The silhouette plot for the various clusters
Cluster label
The silhouette coefficient values
8 Clustered data
amount_log
recency_log
Silhouette score: 0.34
amount_log
frequency_log



Silhouette score of 4 clusters get the best results, on par with Elbow Method. The more clusters we have, the more clusters are overlapped, so 4 clusters is the optimum result for the current dataset.

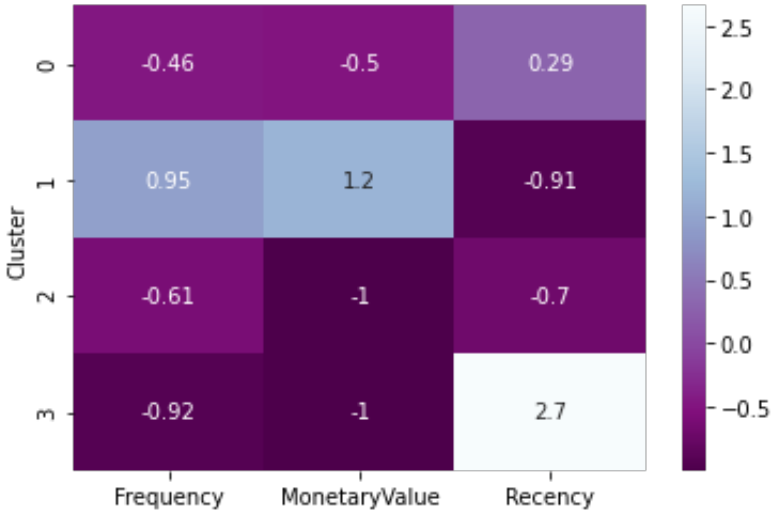
These are the 'centers' of each cluster (4 clusters with silhouette score = 0.4)

```
for 4 clusters the silhouette score is 0.40
Centers of each cluster:
   MonetaryValue    Recency   Frequency
0   1.181927e+00   4.918218   48.025184
1   8.264728e+06  18.938138   77.446728
2   3.627124e+07   1.764248  265.742023
3   1.806926e+00  69.808249    9.879429
```

clusters_4	Frequency	Recency	Monetary
0	81.410393	7.222635	1.316884e+03
1	112.613002	34.023618	8.052053e+07
2	404.713891	1.321478	3.501036e+08
3	17.123140	98.732100	2.262380e+05

Figure 10: Mean value of each cluster.

The following heatmap show the relationship of R, F, M values more visually.



We can clearly see that each cluster has notably characteristics:

- **Cluster 0:** Customers that recently active, have average activity frequency and value of transactions.
- **Cluster 1:** Customers with high activity frequency, high value of transaction and recently active. These are customers that every application wants to acquire.
- **Cluster 2:** Customers have average recency value, low transaction value and average activity frequency. These are usually new customers who try using the app and have no significant transaction.
- **Cluster 3:** Customers have large recency values, low transaction values and low activity frequency. These customers probably stopped using the app.

We assumed cluster 0 and 3 are clusters with high churn probabilities. Cluster 3 is almost churned, and cluster 0 is potential churn. We will concentrate on these clusters.

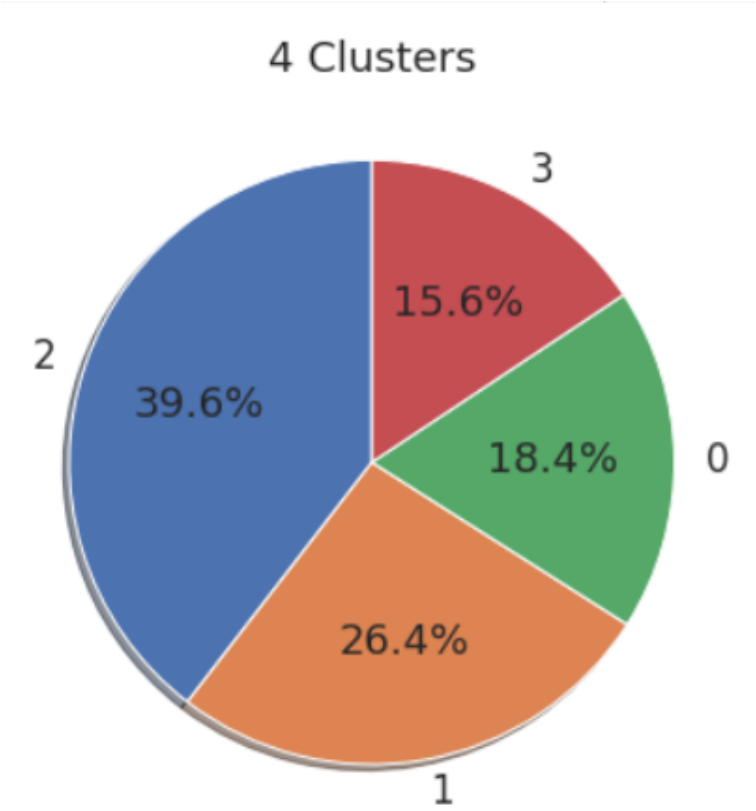


Figure 11: The proportion of each cluster

3.3 Data Insight

After merging CUSTOMER_NUMBER in cluster 3 with customer data, we have some insight about churned customer AGE and GENDER:

- There are 49 customers aged from 0 to 18.
- There are 5494 customers aged from 18 to 30, 59.25% are males.
- There are 6369 customers aged from 30 to 59, 57.39% are males.
- There are 252 customers aged from 59 and above, 50.4% are females.

LifeTime Value

Customer's Life-Time Values is defined as the date number between the register date and the last activity date.

As mentioned in Section 2.2, we will not consider 413 customers have e-bank register date after the first activity date. So that, we analyse here total 77327 customers. The figure below will perform the life-time value distribution of cluster 3.

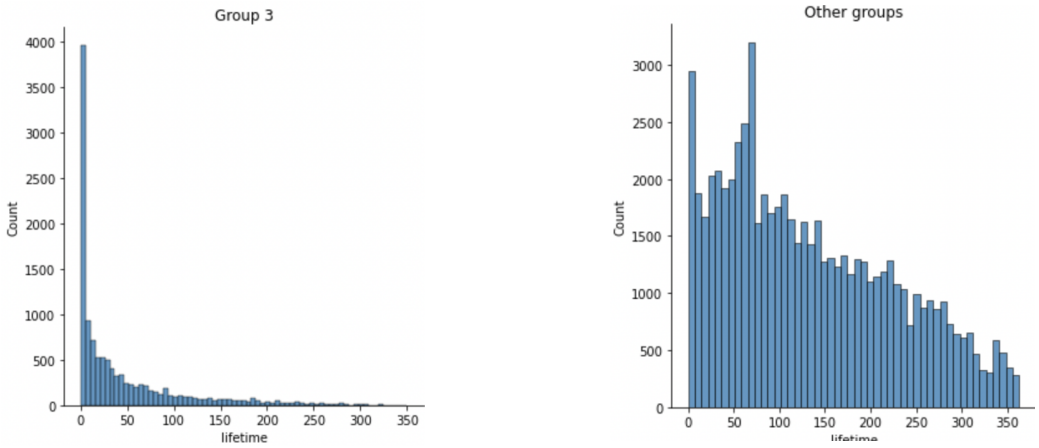


Figure 12: Distribution of life-time value of cluster 3 and other clusters.

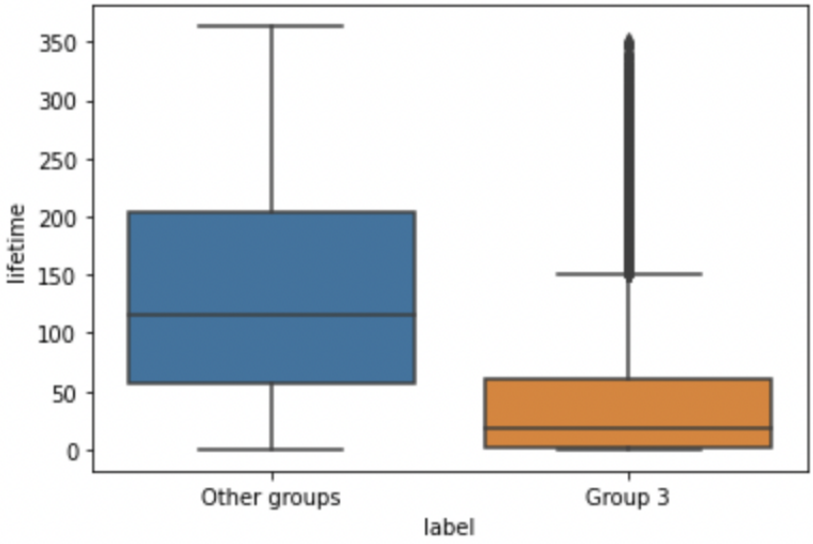


Figure 13: Boxplot perform distribution of other clusters and cluster 3.

We suppose to analyse cluster 3 separately because customers in this cluster almost churned. There are significant differences between cluster 3 and other clusters in life-time value. Cluster 3's life-time value is just under 50 days, and others are well distributed and mostly over 50 days. Besides, most customers (53%) in cluster 3 have life-time value less than 30 days. If we chose customers that have value below the 30 days threshold, we would have over 12,000 customers, and churn rate in cluster 3 would be 50%. If we did not use the 30 days threshold, cluster 3 churn rate will just be 15.6%. Therefore, we can use life-time value as a feature when defining churned customers.

Off-Time Value

Customer's Off-Time Value is defined as the largest time gap between 2 customer's activities.

Similar to life-time value, we will compare between cluster 3 and other clusters.

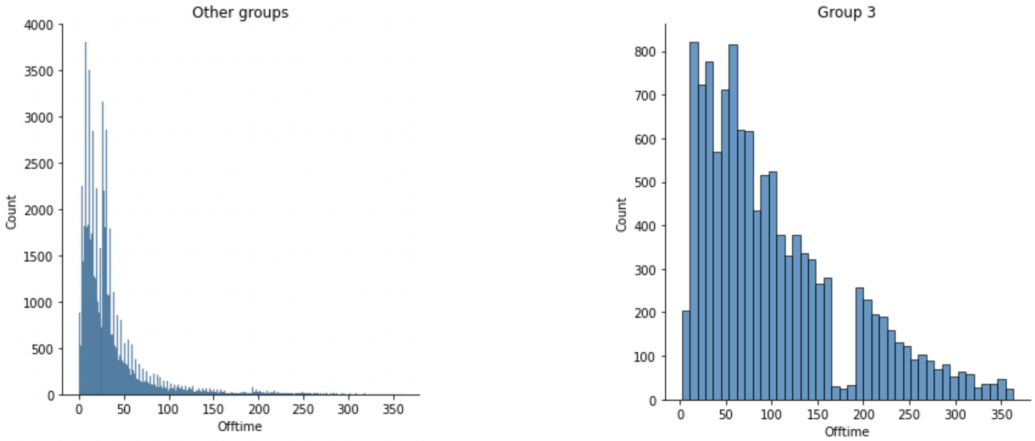


Figure 14: Distribution of off-time value of cluster 3 and other clusters.

50% of customers in cluster 3 have off-time value over 80 days, while other clusters' off-time value are mostly under 50 days, of which 75% have off-time value below 40 days. We can say that if we took customers that have off-time value over 43 days, we will have 22589 customers, in which churn rate of cluster 3 would be 39%. Therefore, we can consider off-time value as a feature when defining churn customers.

In addition, if combining life-time with off-time value:

- Consider customers which have off-time value larger than 43 days and life-time value less than 30 days, we have 4836 customers and churn rate is 86%.
- Consider customers which have off-time value larger than 43 days or lifetime less than 30 days, we have 29760 customers and churn rate is 36%.

If we consider cluster 3 as "churned", we would have some insights.

Cluster 3 and 0 (3+0) are total 32654, about 34% total considered customers. We suppose to combine these 2 groups of customers and observed that 63.2% of customer numbers have life-time value less than 69 days. We show below life-time value distribution of cluster 3+0 and other clusters.

```
1   dflt.LIVE_TIME.describe()
```

```
count    26031.000000
mean        69.602897
std         79.356479
min          0.000000
25%          8.000000
50%         39.000000
75%        105.000000
max        363.000000
Name: LIVE_TIME, dtype: float64
```

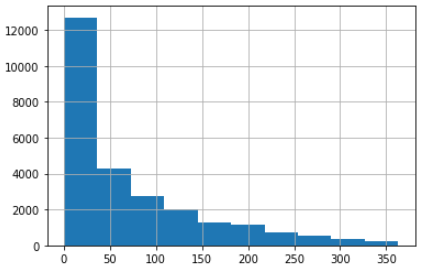


Figure 15: Life-time value Distribution of cluster 3+0.

```
1   df_not_churn.LIVE_TIME.describe()
```

```
count    51296.000000
mean       145.440619
std         91.982720
min          0.000000
25%         67.000000
50%        131.000000
75%        216.000000
max        363.000000
Name: LIVE_TIME, dtype: float64
```

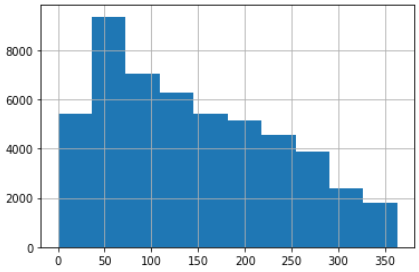


Figure 16: Life-time value Distribution of other clusters.

Consider the threshold of life-time value about 25 days, based on the below chart, we get the number of customers who have life-time value less than threshold is 14004, in which number of churned customers occupy about 74% where 54% from cluster 0 and 45% from cluster 3. It is clear that life-time value is efficient criteria for defining churned customer.

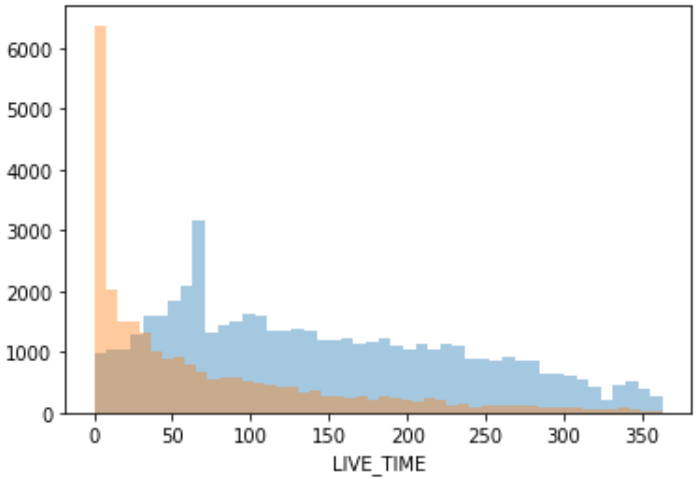


Figure 17: Comparison between distribution of cluster 3+0 (red) and other clusters(blue).

3.4 Final Churn Definition

Based on above analyse result, the value of X, Y we suggest for the suggested definitions in Section 1 are:

- **Definition 1:** Customers who do not use myVIB (no activities) after 30 days since e-bank register date.
- **Definition 2:** Customers who have off time(from register E Banking to the last activity date) greater than 45 days.
- **Definition 3:** Customers who do not use myVIB from 30 days since e-bank register date *or* have off time greater than 45 days.

4 Feature Engineering

4.1 Customer

When we look at age distribution, we can see that customers which have age greater than 35 are likely to be churned, make this an important feature to be considered.

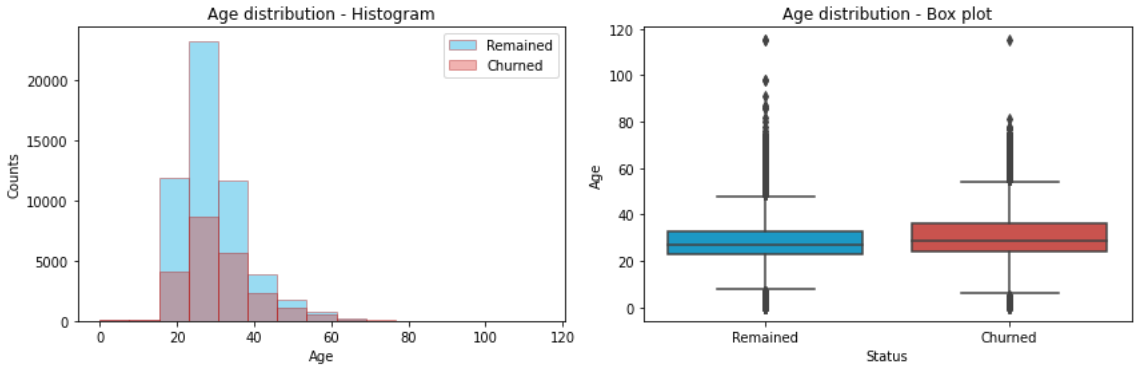


Figure 18: Age Distribution

Considering EB_REGISTER_CHANNEL, churn rate is higher when its value is ”AUTO-JOB”.

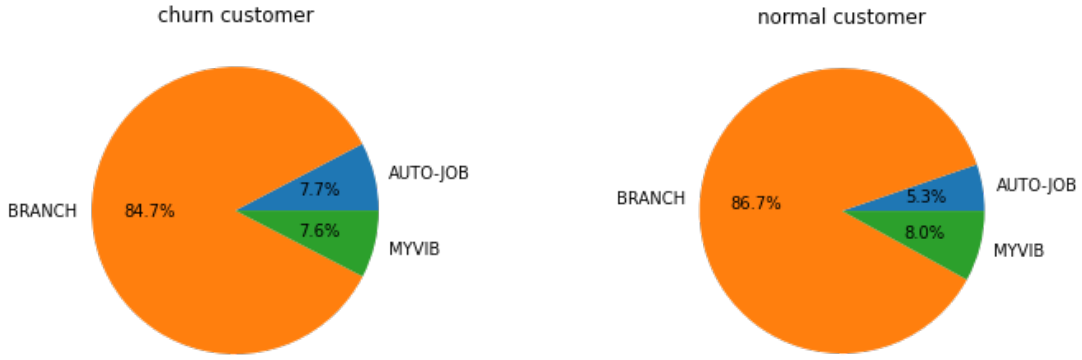


Figure 19: Proportion of EB Register Channel

4.2 MyVIB Activity

Number of transactions in a day of a churned customer is 50% lower than a normal customer.

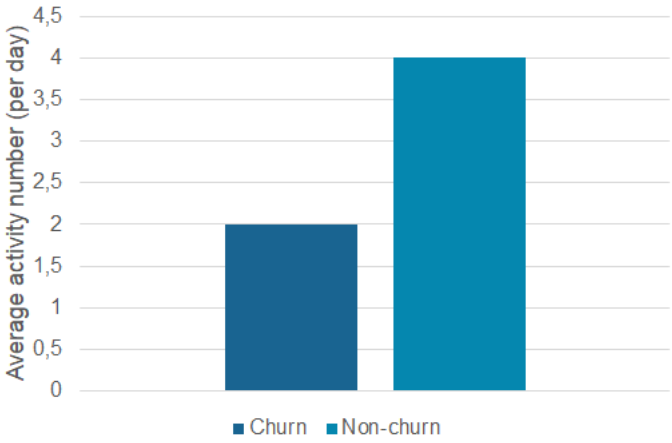


Figure 20: Average activity number (per day) between Churn and Non-churn

When look at login and logout distribution, we can see that churned customers have very small login and logout times (mostly under 20 times) in the first 30 days. On the other hand, normal customers have high login and logout times (mostly around 50 times).

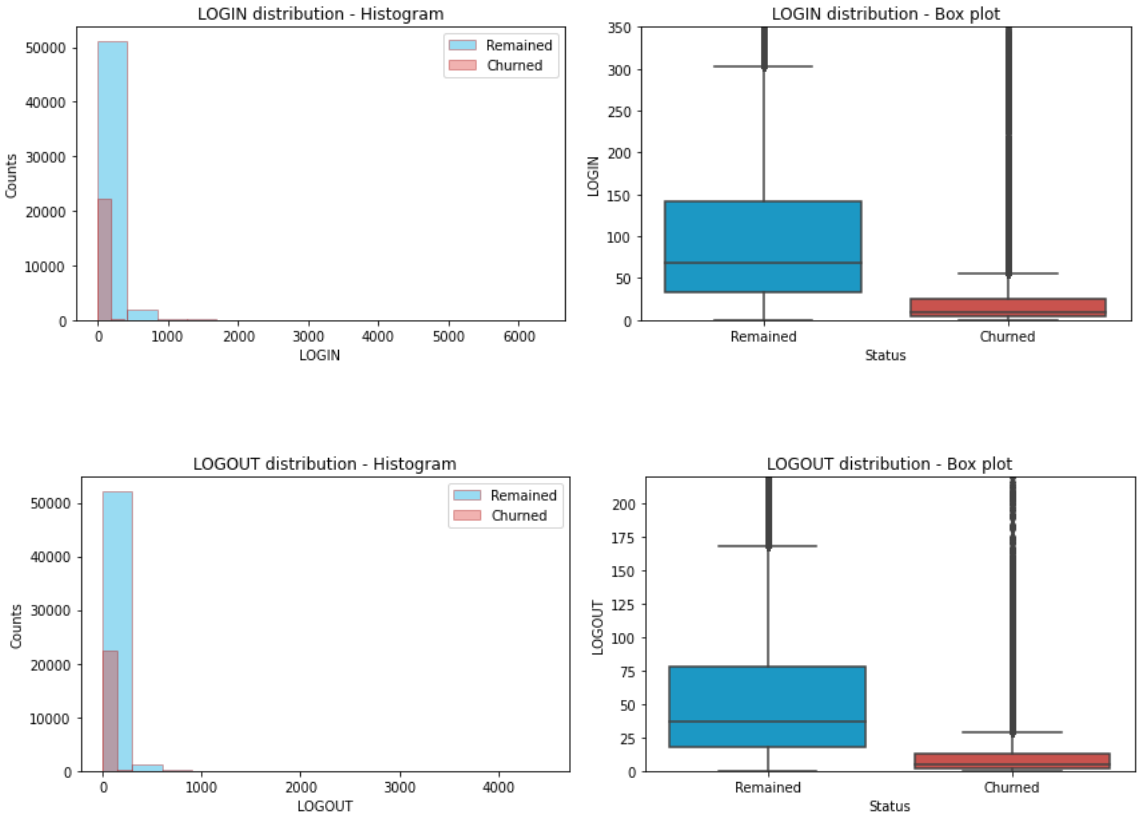


Figure 21: Login and Logout Distribution

Similar to login and logout distribution, when we considered QUERY_ACCOUNT_INFORMATION, we could see that churned customers has small number of queries in the first 30 days (mostly under 10 queries). On the other hand, normal customers have high number of queries (around 50 queries).

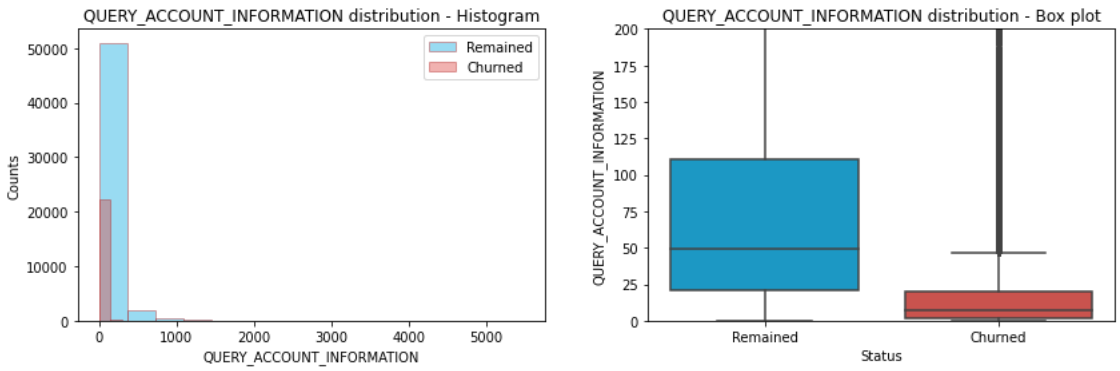
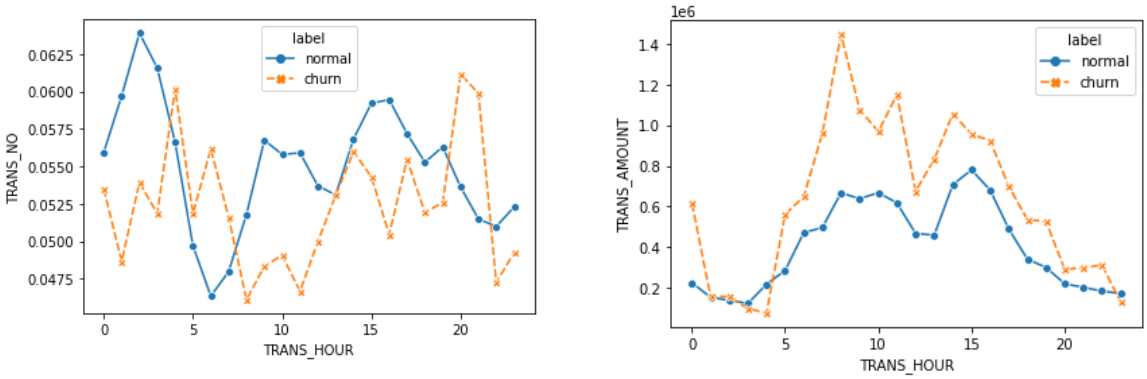


Figure 22: QUERY_ACCOUNT_INFORMATION Distribution

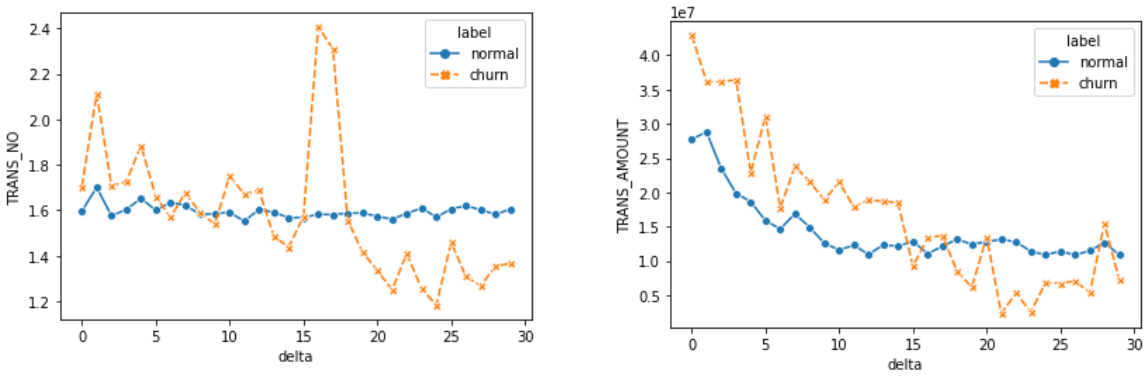
4.3 Transaction

Considering transactions after the first 30 days since the register date of each customer, we got some interesting insights:

- The number and value of transactions per hour have distinctions. In detail, normal customers usually have transactions in the work time, otherwise, churn customer tend to have transactions after work time.
- However, in working time, the value of transactions of churned customers are much higher than normal customers. This is an extraordinary point.



Besides, the number of transactions of normal customers are stable per day, whereas, they decrease with churned customers after 17-th day. The value of transactions is more significantly decrease compare with normal customers.



4.4 Deposit

Considering customers in this table, we have some insights:

- Customers who do not have Current Account have higher churn rate compare to customers who do (about 20%).

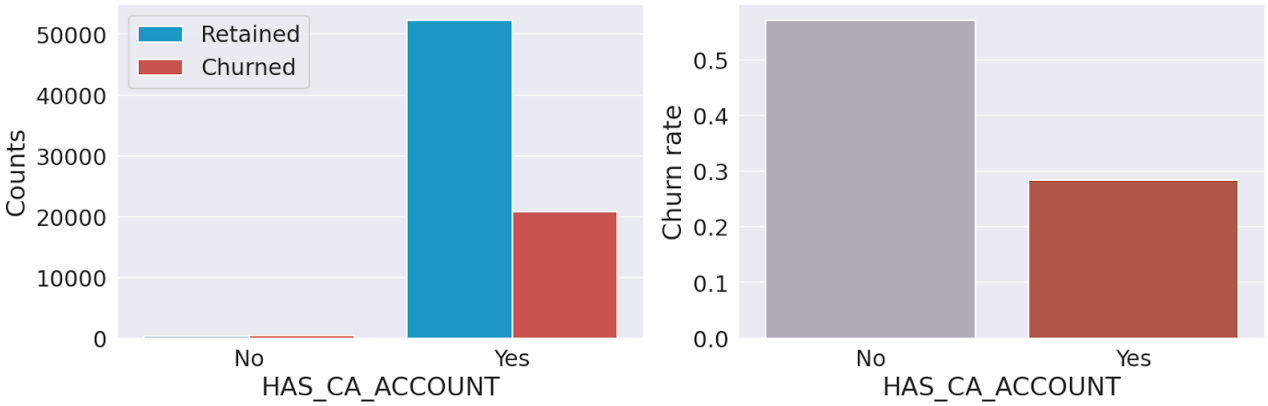


Figure 25: HAS_CA_ACCOUNT and Churned Customer relationship

- Customers who have fluctuations in Term Deposit Account also have higher churn rate than customers who don't have any fluctuations in Term Deposit Account.

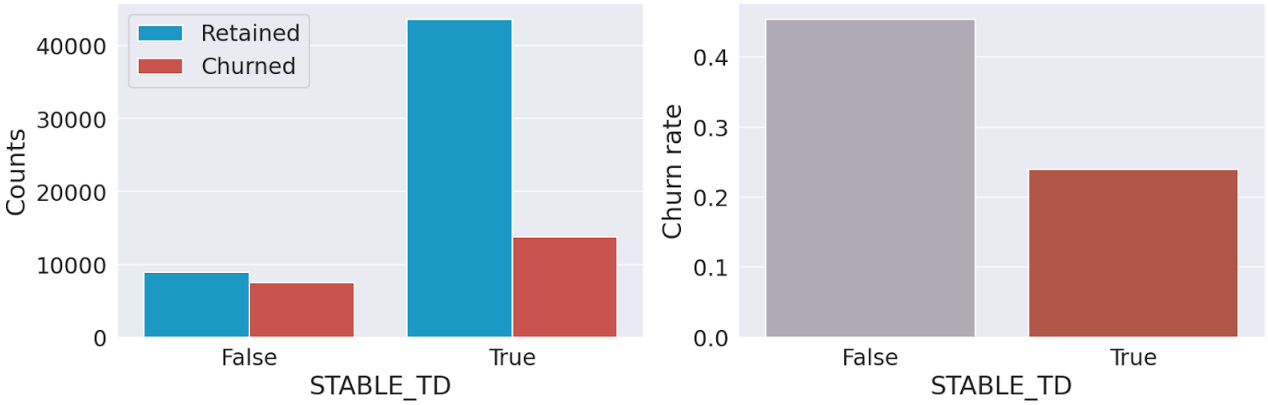
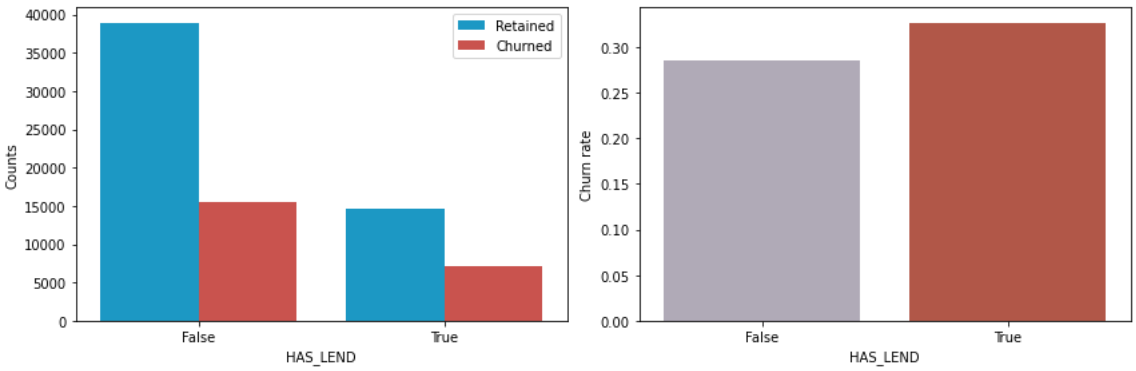


Figure 26: STABLE_TD and Churned Customer relationship

4.5 Lending

With the given lending data, we observed that it does not affect to the behaviours of churned customers.



4.6 Card

Whether customers have Credit Cards, Debit Cards or not, it is not a factor that make customers churn.

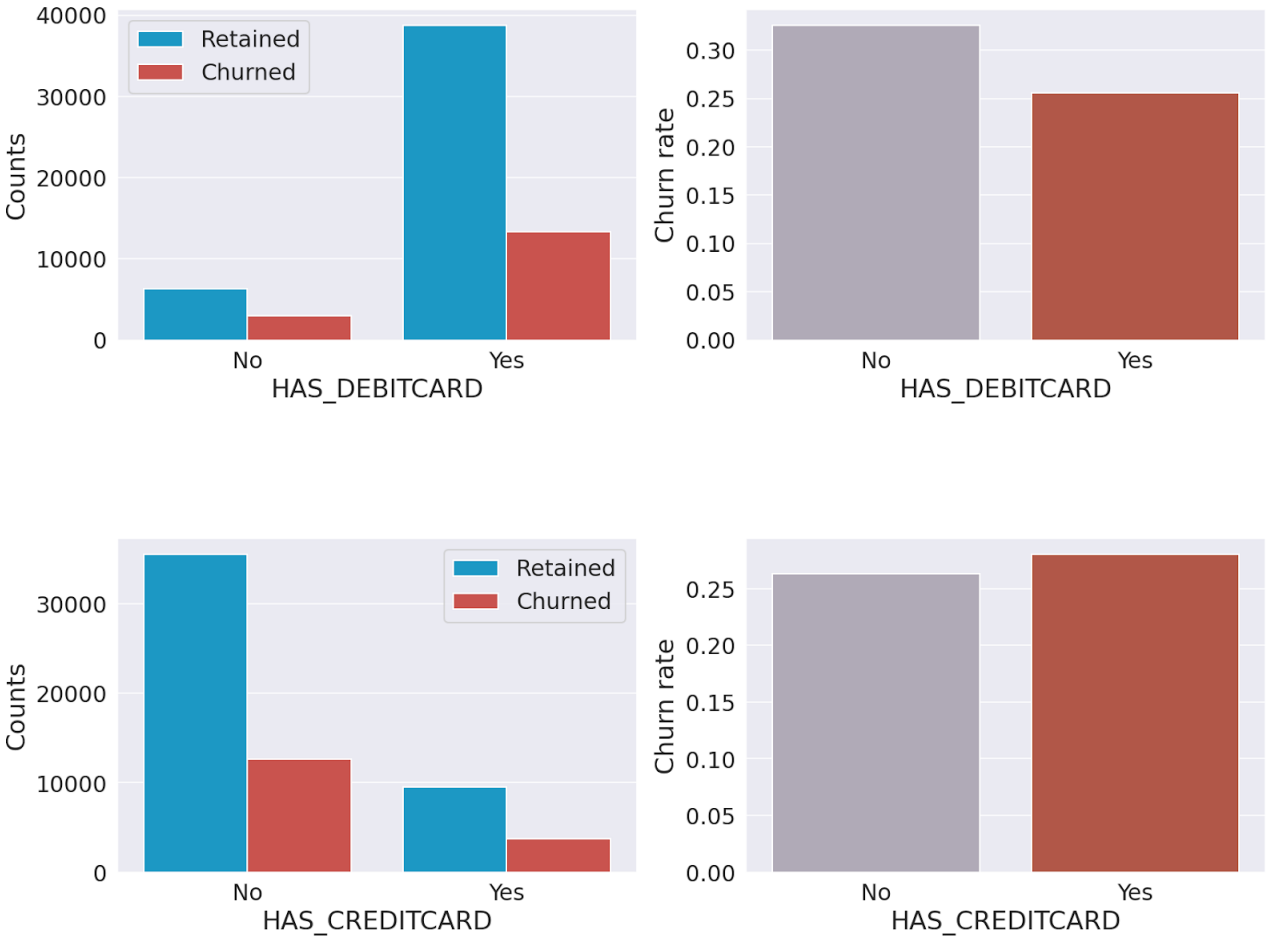


Figure 27: HAS_CREDIT_CARD and HAS_DEBIT_CARD Distribution

5 Train Test Strategies

5.1 Feature observation and labeling

- Creating features in the first 30 days since register date for all customers.
- Label customers based on their behaviours from the 31^{st} day to the 60^{th} day since register date.
  - Churn customers: Customers have no activities during this time.
  - Normal customers: Customers have activities during this time.
- Combining features and labels, we would have a dataset ready for training models.

5.2 Random Strategies

We use two methods when training and testing model.

- Method 1: Randomly splitting the dataset into three smaller sets:
  - Train set: 60% of the dataset, used to train and fine-tuning model.
  - Validation set: 20% of the dataset, used to test and compare performance between different models, then select the best model for testing.
  - Test set: The remaining 20% of the dataset, used to test performance of the selected model on new data.
- Method 2: Randomly splitting the dataset into two smaller sets:
  - Train set: 75% of the dataset, used to train and compare performance between different models. In the training step, we split the train set into 5 sets, use 4 sets to train and use 1 set to validate, then select the best model.
  - Test set: The remaining 25% of the dataset, used to test performance of the selected model on new data.

5.3 Selection Strategies

Assuming we were at the end of August, we would have the dataset and labels for customers who registered 7 months ago. Use that dataset to train model, then use customers data from September to December for testing. In particular:

- Train set: Including customers who registered from January to July, we would have labels and features in 30 days for these customers. Using k-fold cross validation to train and validate, and then select model with the best parameters.
- Test set: Including customers who registered from August to October, we would have data to the end of December, then extract features and labels to test perfomance of trained model.

6 Model Strategies

After labelling data by churned definition above, we suppose to use Supervised Learning for solving this problem and treat it as a classification problem.

We can see that during data overview and analyse steps, we got a lot of features, which has significant impact on the decision whether a customer is churned or not, such as Recency, Frequency, Monetary, Life-time value, Off-time value, Age, Gender, Average deposit/lending amount, etc... .

For data, we decided to split into training and testing set based on these criteria:

- **Training:** features and label of customers who registered e-bank in the first 9 months of the year.
- **Testing:** label of customers who registered e-bank in the last 3 months of the year.

During training process, we will use *K-fold validation*.

After features selection, balancing data (in case data is imbalance), we suppose to do modelling step, we probably try with some statistical modelling approaches such as: Logistic Regression, K-Nearest Neighbor, Random Forest, SVM, Ridge Classifier, Decision Tree and Gradient Boosting, etc... .

However, with these traditional machine learning approaches, one challenge is choosing optimal set of hyper-parameters. For this purpose, we will first define which hyper-parameters we want to experiment with, and what values to try out. We will pass this information to Scikit-Learn's GridSearchCV which then evaluates all the possible combinations of hyper-parameter values.

GridSearchCV evaluates performance by performing k-fold cross-validation. The idea behind k-fold cross-validation, which is illustrated in this figure, is simple: it splits the (training) set into k subsets/folds, trains the models using k-1 folds, and evaluates the model on the remaining one fold. This process is repeated until every fold is tested once.

7 Modeling

7.1 Data Processing

7.1.1 Encoding Categorical Features

- Label encoding: Label encoding is a popular encoding technique for handling categorical variables. In this technique, each label is assigned a unique integer based on alphabetical ordering.
- One-hot encoding: One-Hot Encoding is another popular technique for treating categorical variables. It simply creates additional features based on the number of unique values in the categorical feature. Every unique value in the category will be added as a feature.

7.1.2 Features description

- We have a total of 222 features from all tables. Some main features from these tables are:
  - Customer: gender, register channel, age, e-banking register channel,...
  - Activity: has activity or not, activity number for login, logout, query type, hourly, daily,...
  - Transaction: has transaction or not, transaction number and transaction amount for hourly, daily,...
  - Deposit: has current account or not, has temp deposit account or not, current account stability,...
  - Lending: has lending or not, lending amount, monthly payment stable,...
  - Card: has debit cards or not, has credit cards or not, max number of cards,...

7.1.3 Scaling

Feature scaling is a technique used to normalise the range of features in a dataset. Some algorithms are sensitive to feature scaling (e.g. SVMs) while others are invariant to it (e.g. Random Forests).

I decided to use **StandardScaler()** which standardises features by subtracting the mean and dividing by the standard deviation. This results in features with zero mean and unit variance.

7.1.4 Addressing Class Imbalance

As we have seen previously, there is an imbalance in the classes to be predicted, with one class (0 – remained) much more prevalent than the other (1 - churned).

Class imbalance is usually a problem and occurs in many real-world tasks. Classification using imbalanced data is biased in favor of the majority class, meaning that machine learning algorithms will likely result in models that do little more than predict the most common class. Additionally, common metrics can be misleading when handling class-imbalanced data (e.g. if a dataset contains 99.9% 0s and 0.01% 1s, a classifier that always predicts 0 will have 99.9% accuracy).

Thankfully, some strategies can address this problem. We decided to use the SMOTE ('Synthetic Minority Oversampling Technique') algorithm which 'finds a record that is similar to

the record being upsampled and creates a synthetic record that is a randomly weighted average of the original record and the neighboring record, where the weight is generated separately for each predictor.

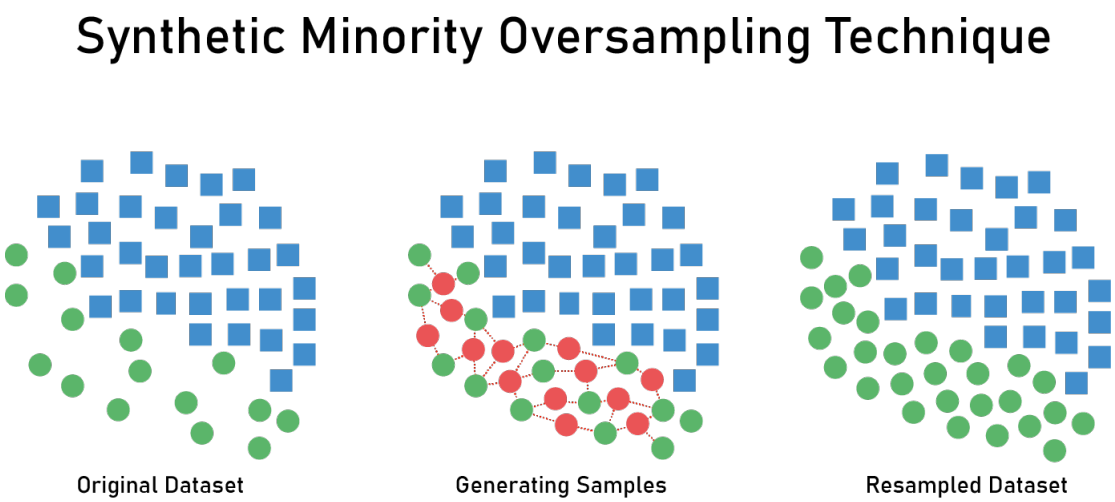


Figure 28: Synthetic Minority Over-sampling Technique

7.2 Building Machine Learning Models

We are now ready to start building machine learning models. The 6 classifiers we have selected are the following:

- Logistic Regression
- K-Nearest Neighbors (KNN)
- Adaptive Boosting (AdaBoost)
- Gradient Boosting Classifier
- Xtreme Gradient Boosting Classifier
- Light Gradient Boosting Machine
- Random Forest Classifier

Using default hyperparameters usually results in non-optimised models that overfit or underfit the dataset. Hyperparameter tuning is the process of finding the set of hyperparameter values that achieves optimal performance. For this purpose, we will first define which hyperparameters we want to experiment with, and what values to try out. We will pass this information to Scikit-Learn's **GridSearchCV** which then evaluates all the possible combinations of hyperparameter values. As mentioned in the Objective, recall will be used as the scoring metric for optimising our models. We use recall since correctly classifying elements of the positive class (customers who churned) is more important for the bank.

Beside that, we also consider other metrics to generally evaluate a churn model, such as Precision, F1-Score, Accuracy, MCC.

GridSearchCV evaluates performance by performing **k-fold cross-validation**. The idea behind k-fold cross-validation is simple: it splits the (training) set into k subsets/folds, trains the

models using k-1 folds, and evaluates the model on the remaining one fold. This process is repeated until every fold is tested once.

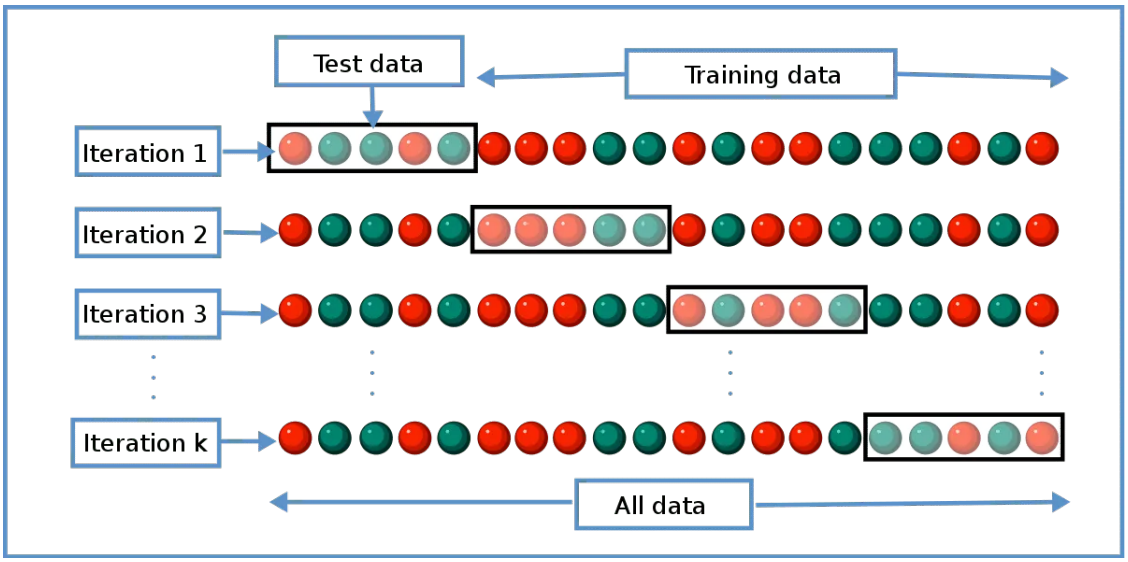


Figure 29: K-fold cross validation

7.3 Ensemble Learning

We will use a voting-based ensemble classifier. So, we implement a voting-based machine learning model using **VotingClassifier**

8 Evaluation Strategies

For evaluation, we suggest using Precision, Recall, F1-score value and Matthew's Correlation Coefficient. Let us see the confusion matrix below:

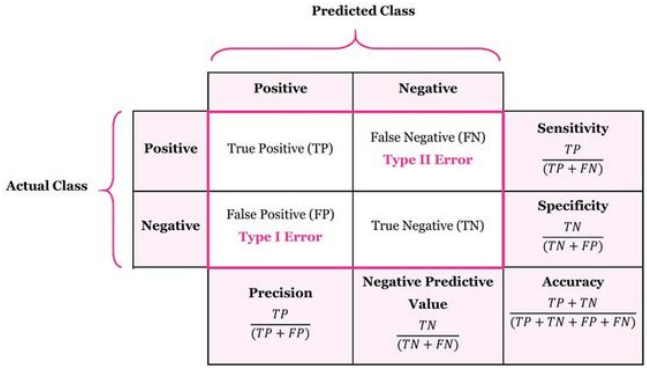


Figure 30: Confusion Matrix

8.1 Precision

Precision measures how many percent of the actual successes the model correctly classifies. Precision uses information provided by the confusion matrix and is calculated using the formula:

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}}$$

8.2 Recall

Recall is a metric that quantifies the number of correct positive predictions made out of all positive predictions that could have been made.

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}}$$

8.3 F1-Score

The F-score is a way of combining the precision and recall of the model, and it is defined as the harmonic mean of the model's precision and recall.

$$\text{F1} = \frac{2 \times \text{precision} \times \text{recall}}{\text{precision + recall}}$$

8.4 Matthew's Correlation Coefficient

Matthew's correlation coefficient (MCC) is used as a reference performance measurement in addition to κ when evaluating predictive models. MCC is commonly used for unbalanced datasets as it is unaffected by the issues caused by disproportions in the dependent variable. MCC considers how well the model performs in all categories of the confusion matrix adjusted for chance and is thus a more reliable statistical evaluation measure.

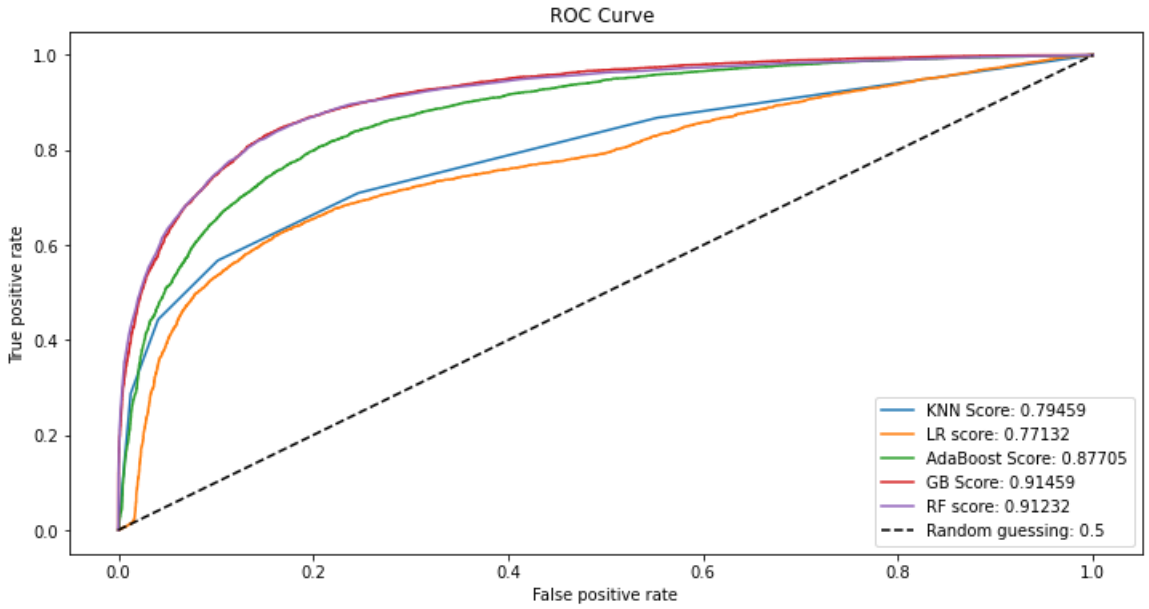
$$\text{MCC} = \frac{\text{TP} \times \text{TN} - \text{FP} \times \text{FN}}{\sqrt{(\text{TP+FP}) \times (\text{TP+FN}) \times (\text{TN + FP}) \times (\text{TN+FN})}}$$

A high MCC score, i.e. close to 1, indicates that the prediction obtains good results for true positives, false negatives, true negatives and false positives, independently of the ratios in the dataset.

9 Result

9.1 Performance

Validations results of trained models after run **GridSearch** is shown below:



We use ROC Curve to visualize and ROC_AUC_SCORE to represents the degree of separability between models. We chose this metric to generalize other concerned metrics.

The figure shows that Gradient Boosting and Random Forest give the best results, and we would ensemble these 2 models using soft voting technique, give us ROC_AUC_SCORE = 0.9125

For having an overall view of all metrics, we have a detailed result below, with class 1 represent churned customers.

	precision	recall	f1-score	support
0	0.86	0.95	0.90	13317
1	0.84	0.63	0.72	5737
accuracy			0.85	19054
macro avg	0.85	0.79	0.81	19054
weighted avg	0.85	0.85	0.85	19054

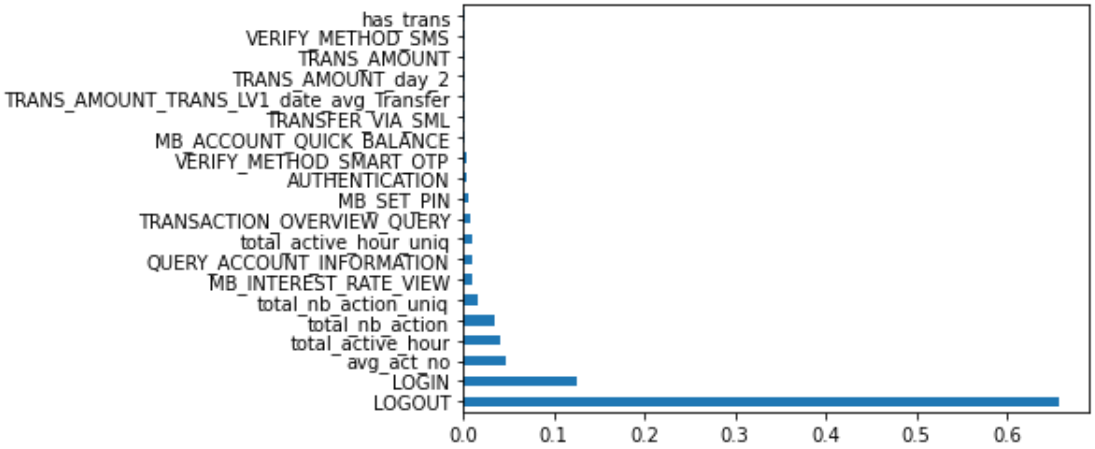
We have Matthew's Correlation Coefficient, MCC = 0.709

These results could be improved if we spent more time on feature engineering, **GridSearch** with more hyperparameters to find the optimal values, or implement other algorithms.

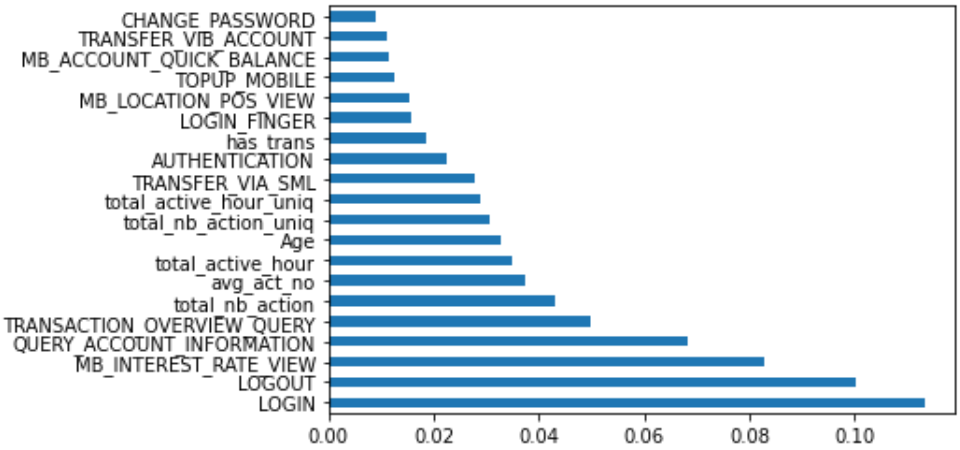
9.2 Feature importance

Below are the top 20 feature importance results of the top 2 models. Login, Logout and avg_act_no are important features for both models. Other features about transaction and age also contribute to model.

- Gradient Boosting



- Random Forest



10 Conclusion

According to this report, we proposed 3 definitions of a churned customer on myVIB platform based on their behaviours. All our strategies from preprocessing data, analysing to modelling and evaluation are fully presented in above Sections. Having an efficient customers churn detection will help marketing and customer service team to correctly target group of customers who need to take care. Therefore making customer retention more effective.

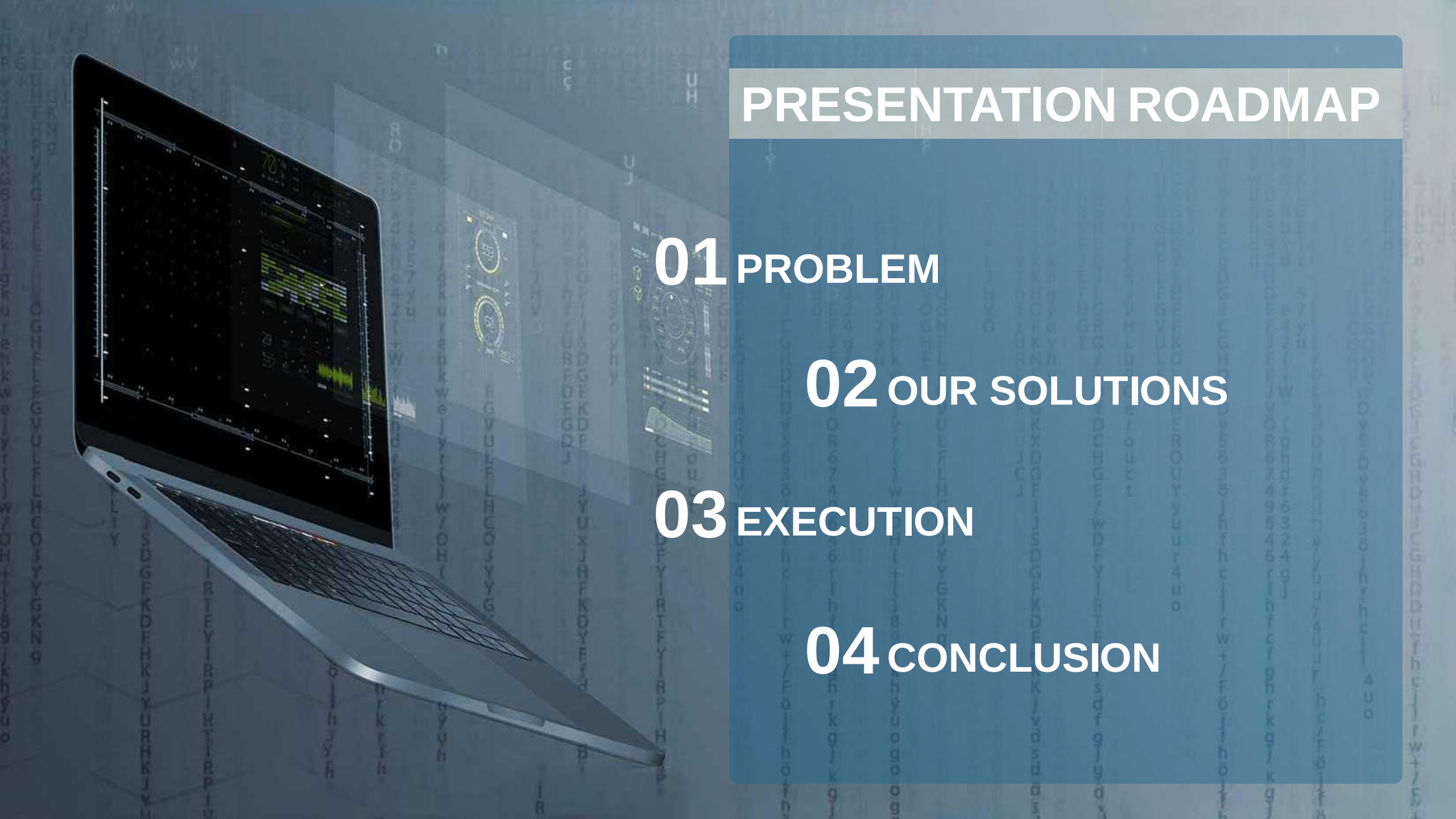
By applying state-of-the-art machine learning models for binary classification problem, we have a promising model for detecting which customer will churn in the future, helping bank increases customer retention rate.

Our technical solutions can be easy to interpret, require small computation power, and deliver result fast, helping bank cut costs and receive more benefits.

From data insights, we have a suggestion to help bank understand more about their customers, is that the need to record details about what kind of transaction, for example online shopping, groceries or real estate,etc. These information could help analyze customers behaviours. We could build a customer profiles database, which help serve the customers better.

Customers churn prediction for myVIB application

SAY Cũng Thành Đúng



PRESENTATION ROADMAP

01 PROBLEM

02 OUR SOLUTIONS

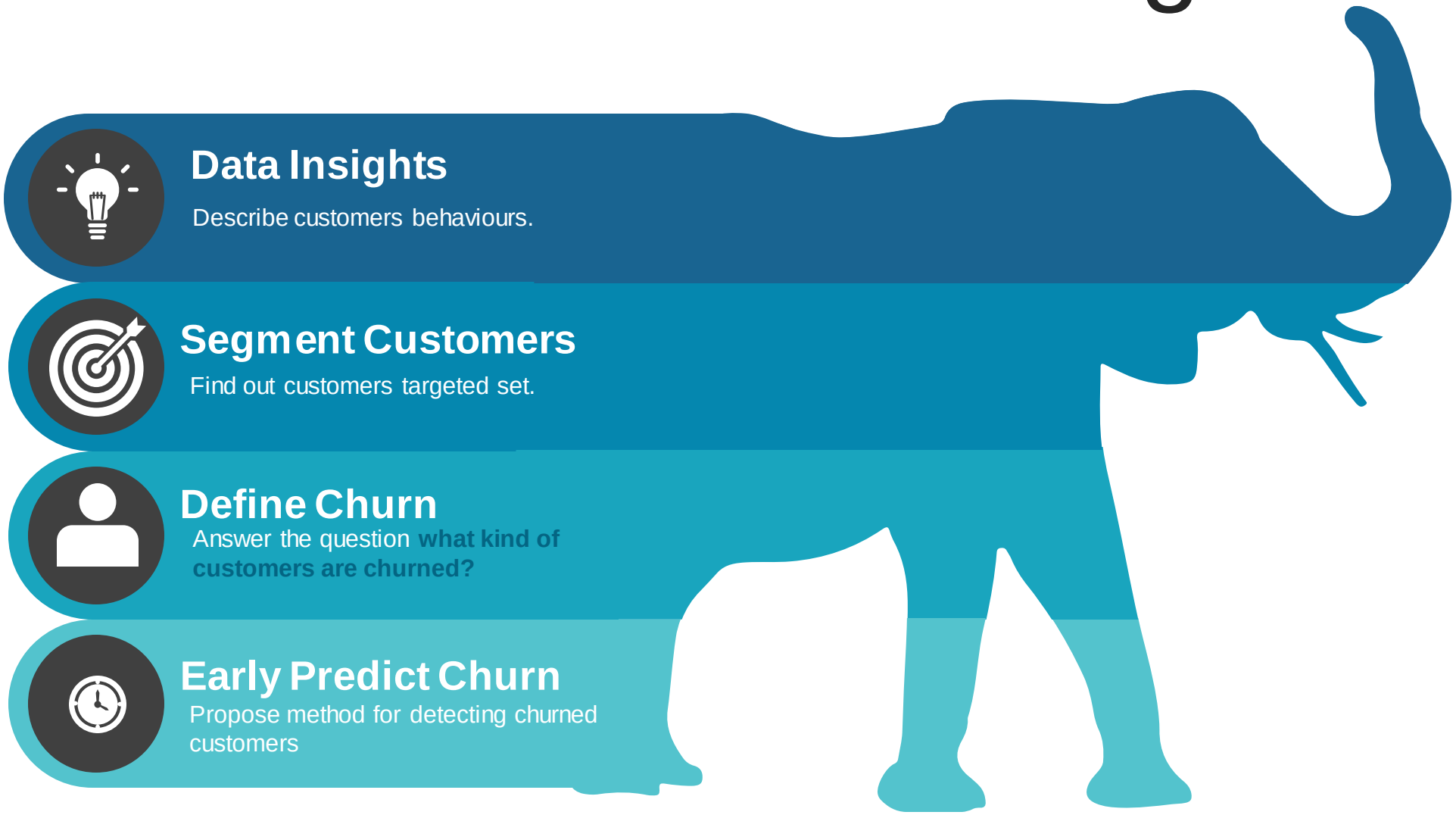
03 EXECUTION

04 CONCLUSION

The Problem

Problem

What are we delivering?



Customer Segmentation

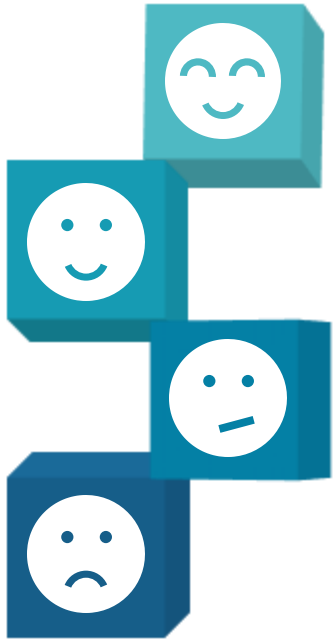
Customer Segmentation

Champion Users

Customers with high activity frequency, high value of transactions and recently active.

Potential Loyalist

Customers have average recency value, low transaction value and average activity frequency.



Promising Users

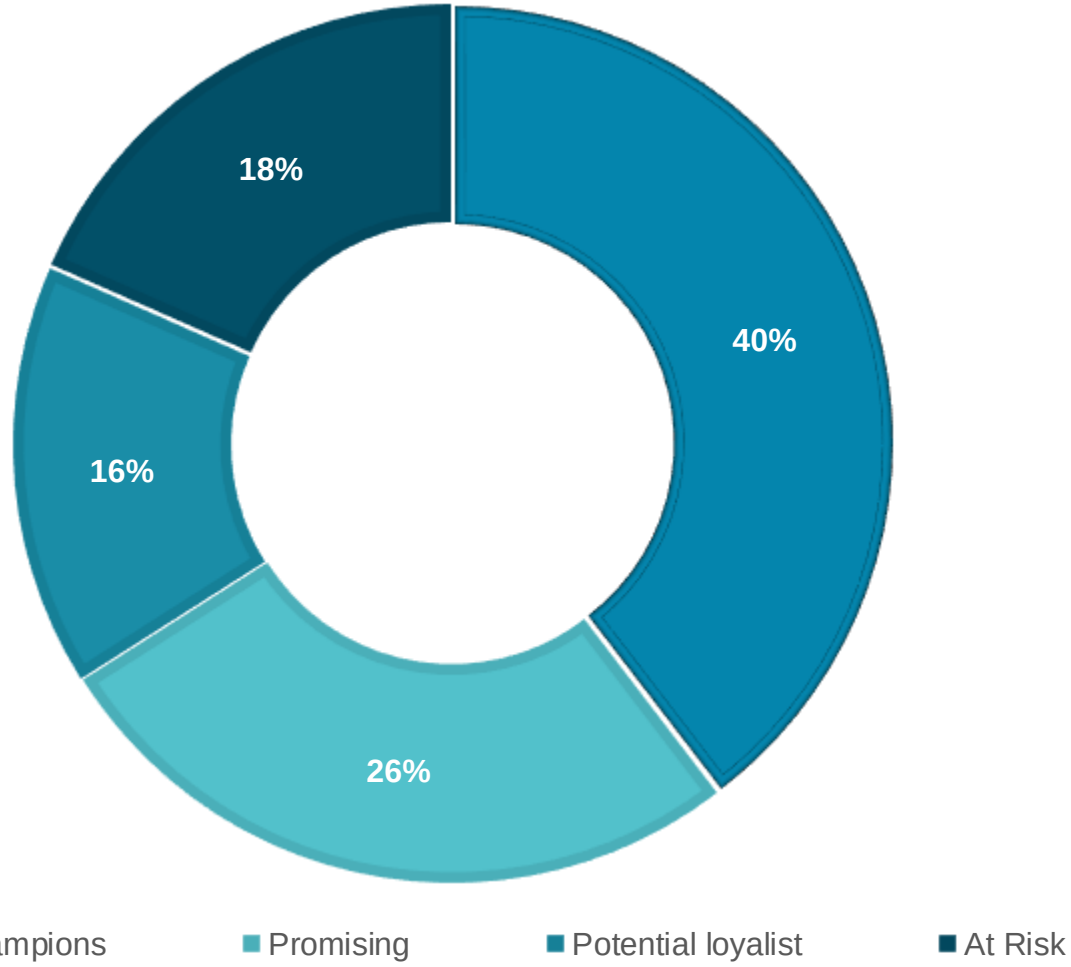
Customers that recently active and high frequency but low value of transactions.

At-Risk Users

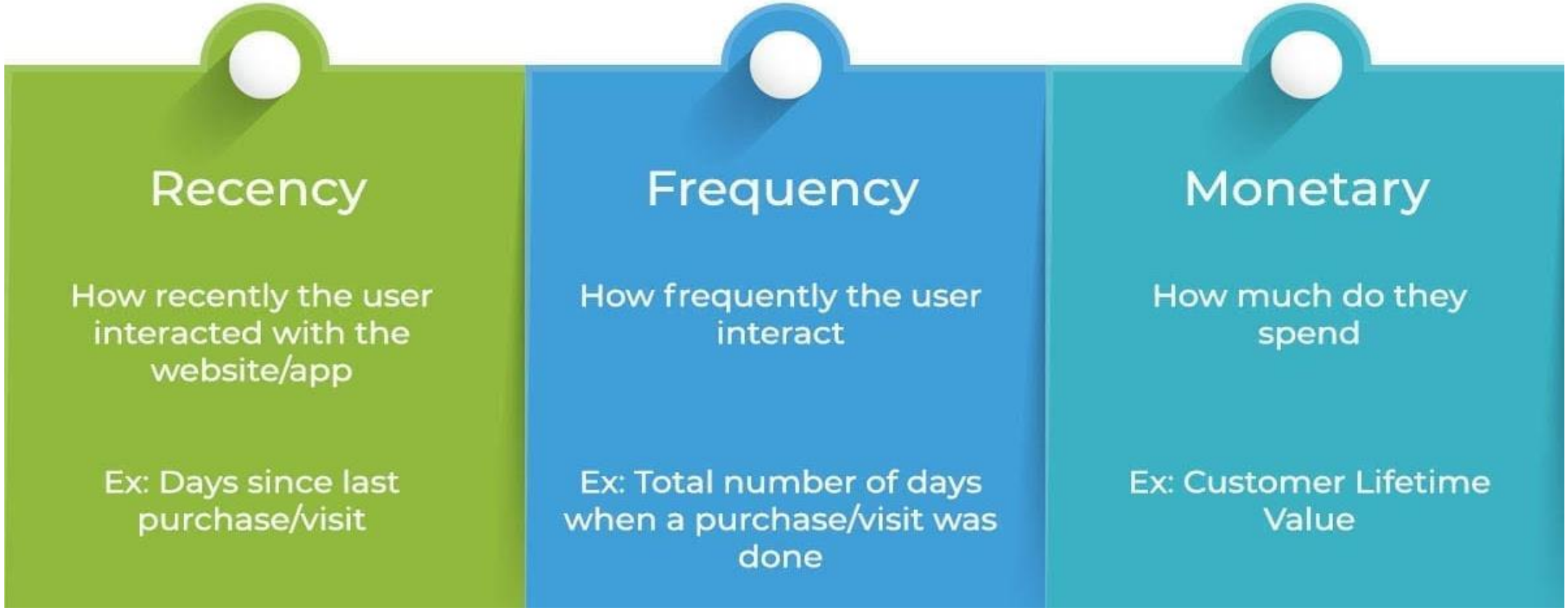
Customers recently not active, low values of transaction and low activity frequency.

Customer Segmentation

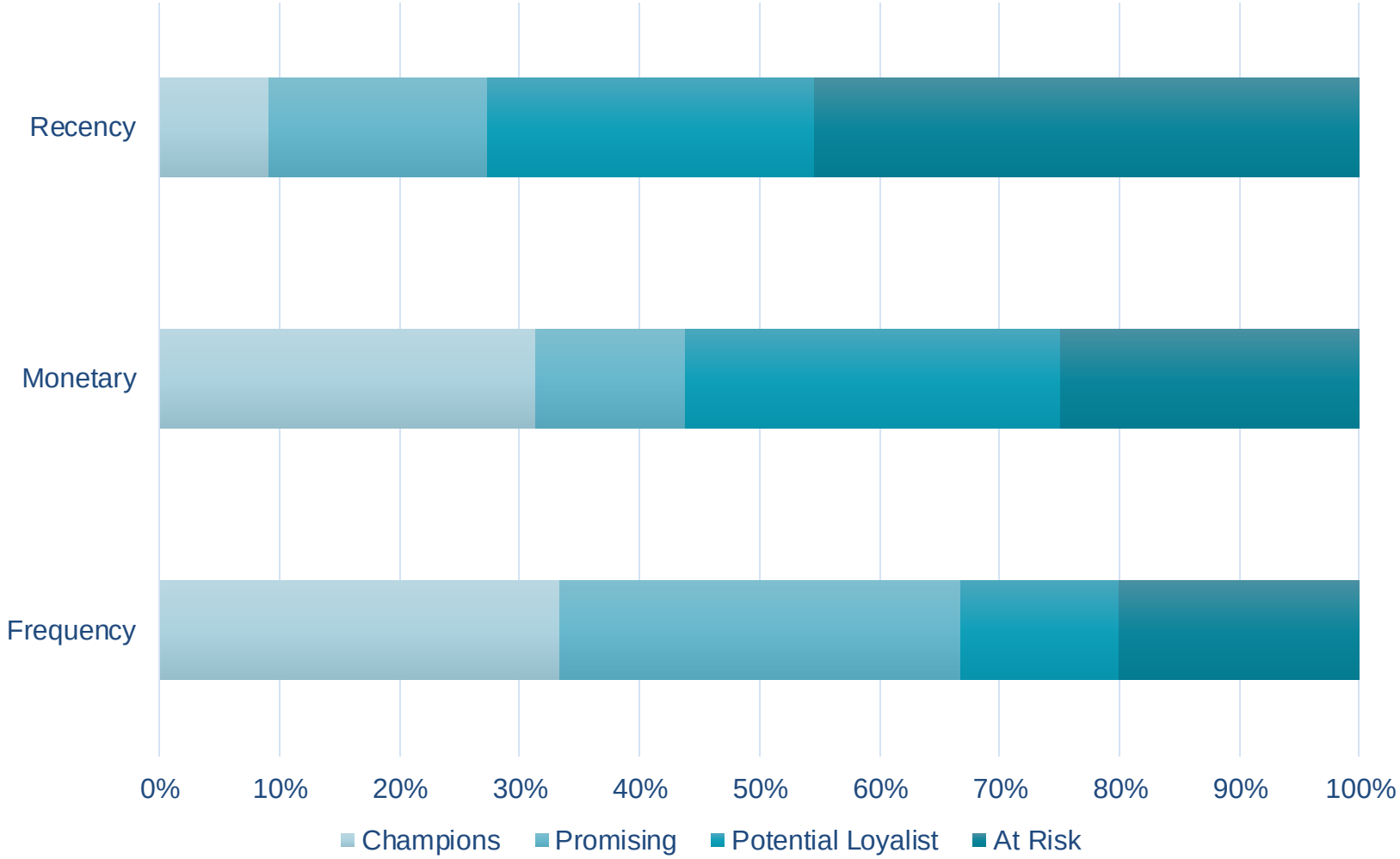
18% belongs to **At-risk** group



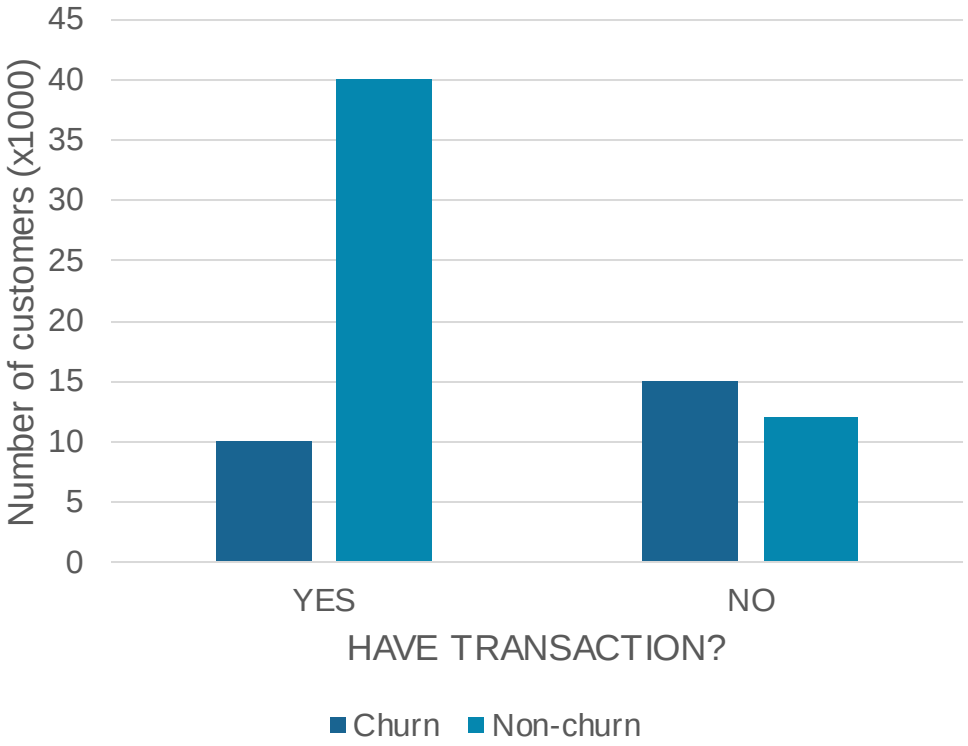
Segmentation Method

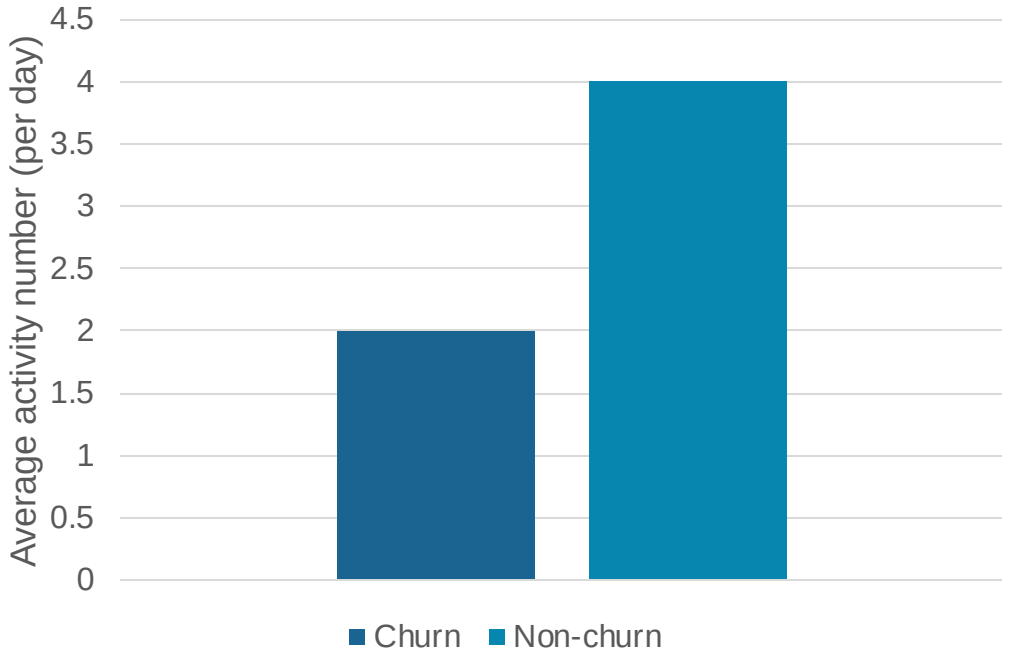


Segmentation Method

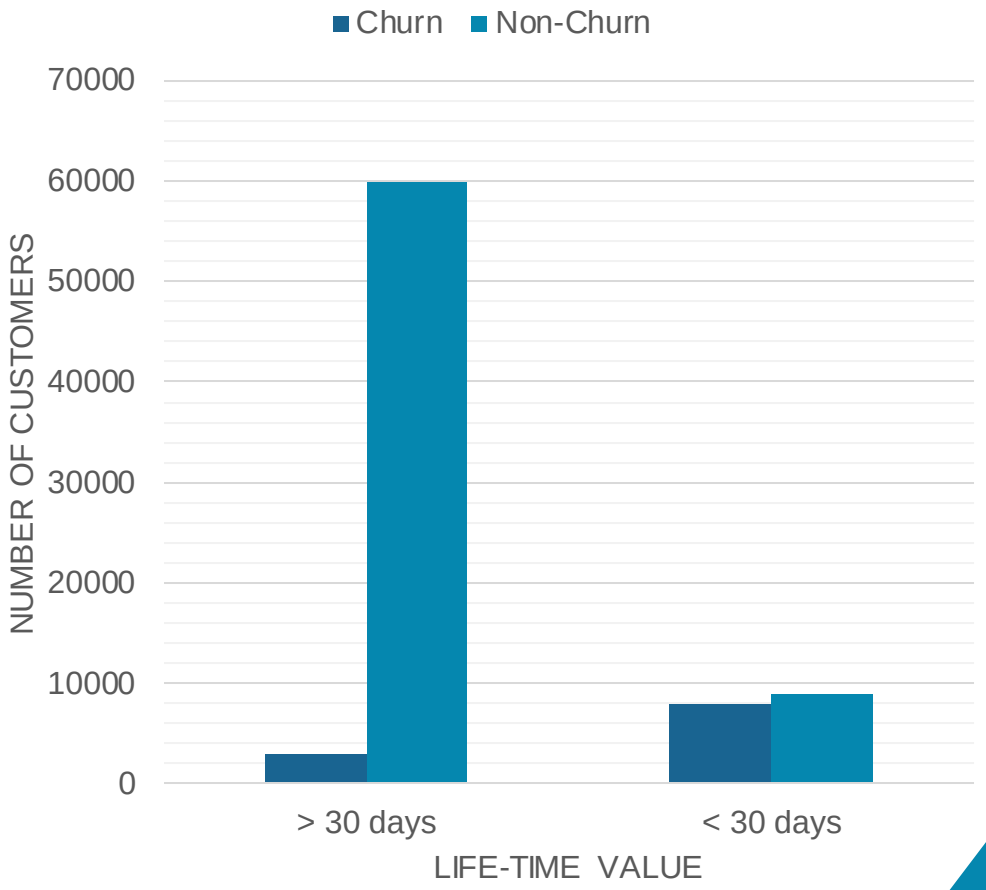


Data Insights





- Customers **with transactions** in the churn group are **lower** than those **without transactions**.
- Customers in **churn** group have **less activity number per day** than others.



“

Customers who do not have any activity on app after 30 days since their e-bank register date are defined as churned.

”

Churn

Definition

Customers Churn Prediction

Data Processing

Encode Categorical

Label encoding
One-hot encoding

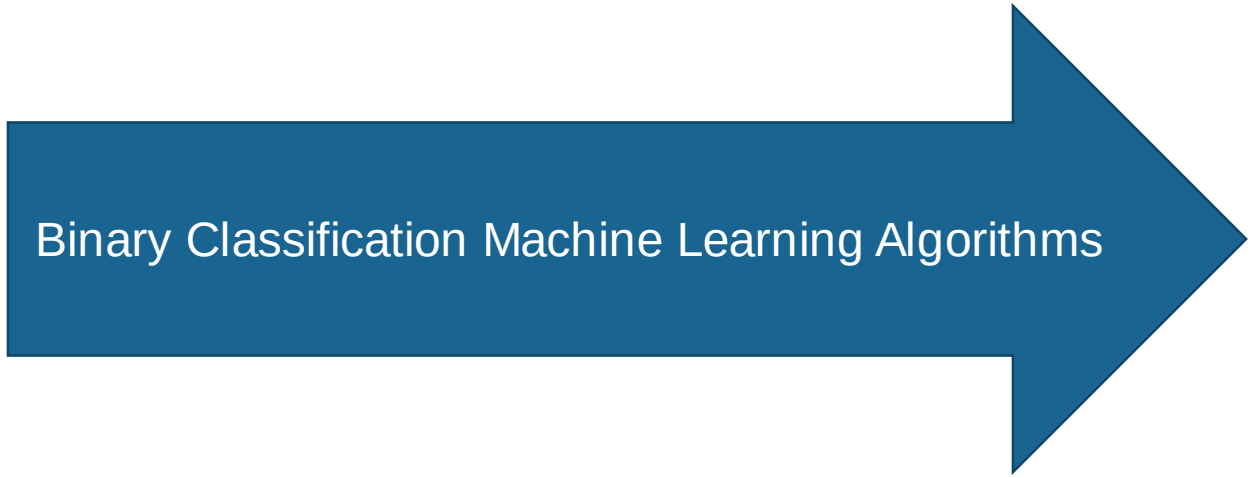
Address Class Imbalance

use the SMOTE ('Synthetic Minority Oversampling Technique') algorithm

Scale

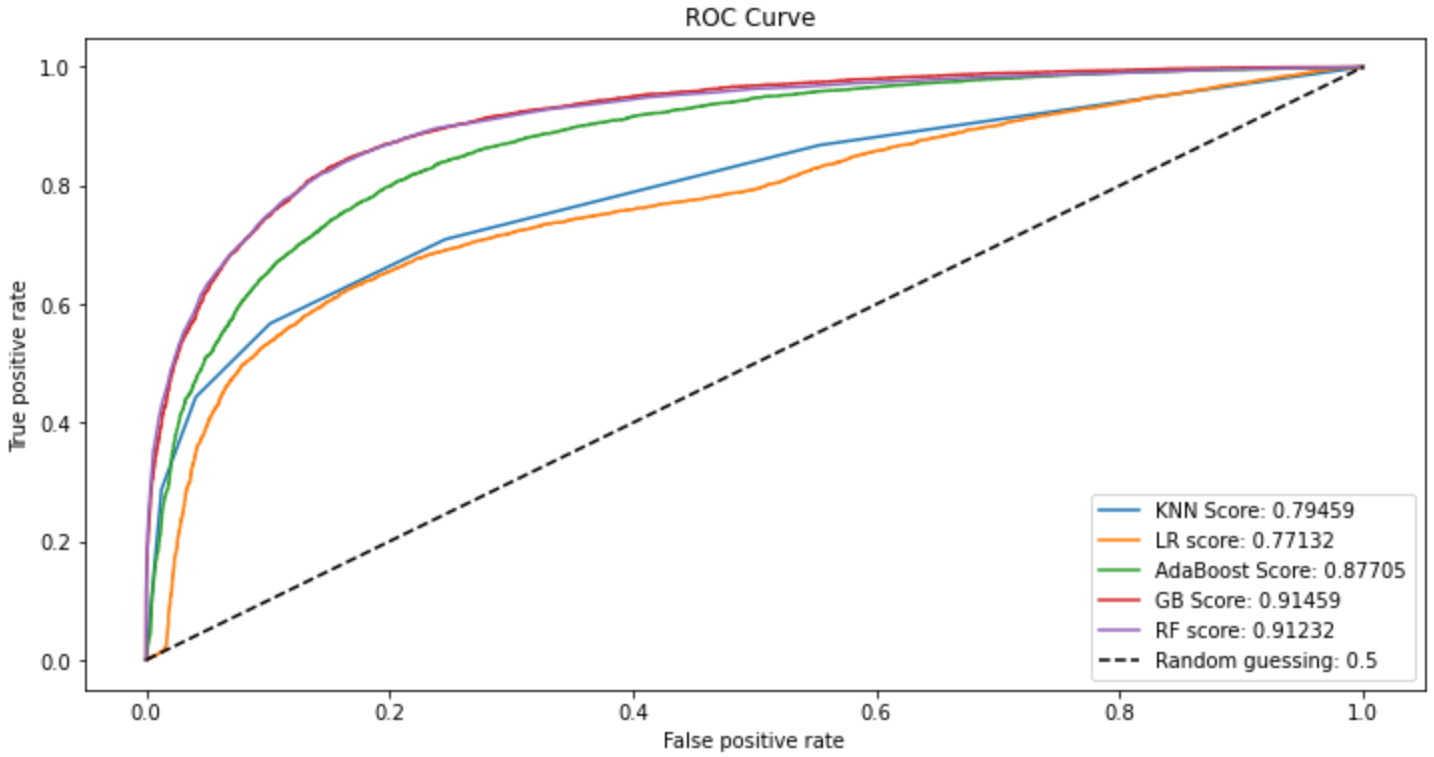
normalise the range of features

Modeling



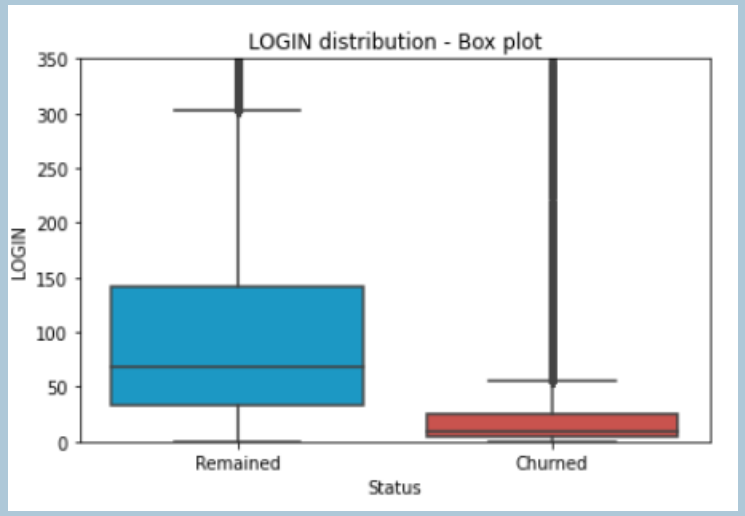
- Logistic Regression
- K-Nearest Neighbors (KNN)
- Adaptive Boosting (AdaBoost)
- Gradient Boosting Classifier
- Xtreme Gradient Boosting Classifier
- Random Forest Classifier

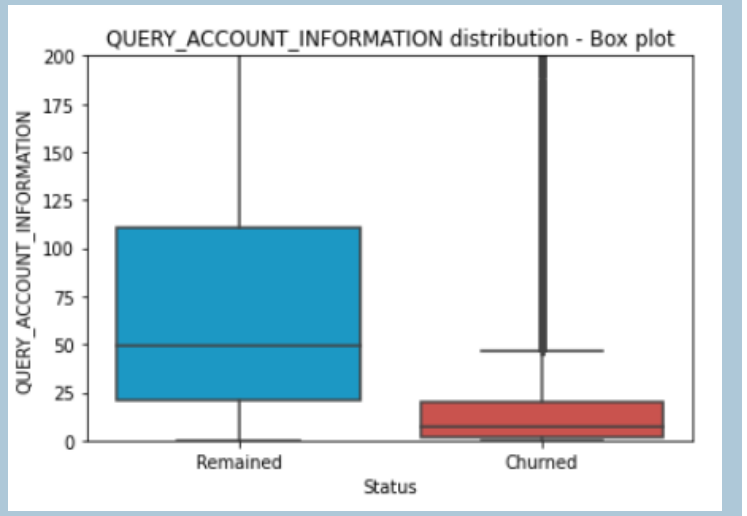
Results

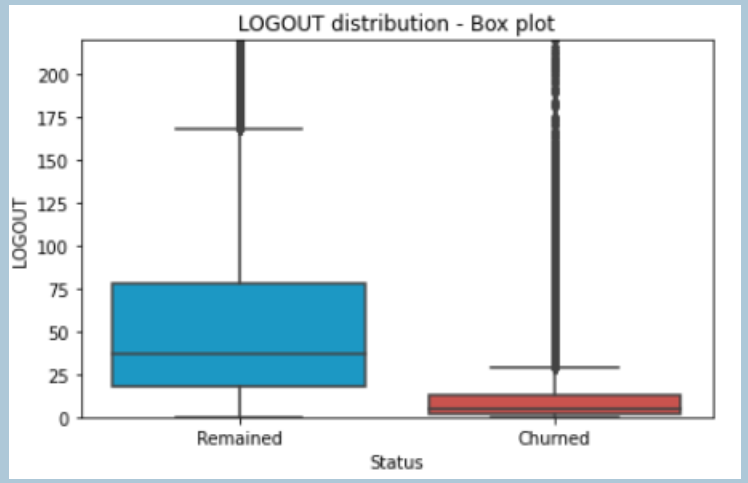


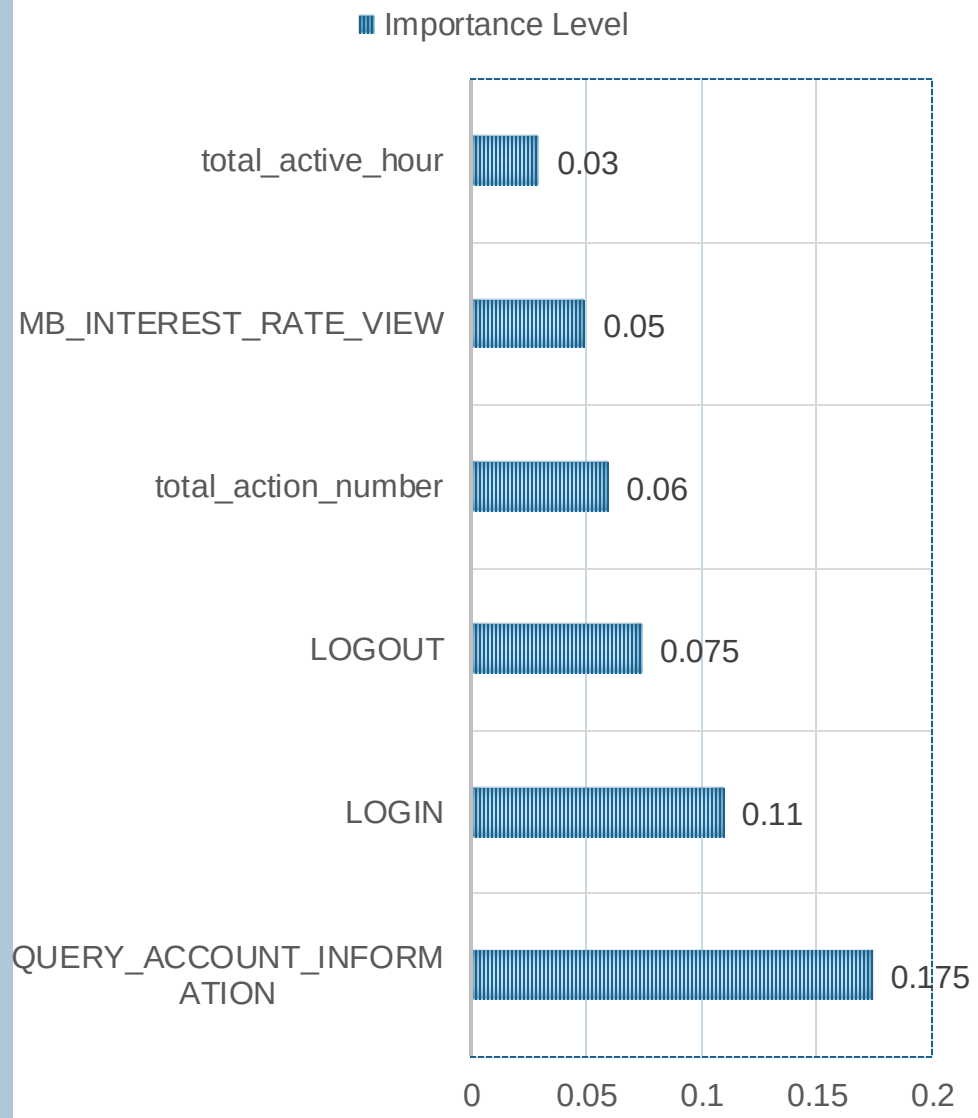
	precision	recall	f1-score	support
0	0.86	0.95	0.90	13317
1	0.84	0.63	0.72	5737
accuracy			0.85	19054
macro avg	0.85	0.79	0.81	19054
weighted avg	0.85	0.85	0.85	19054

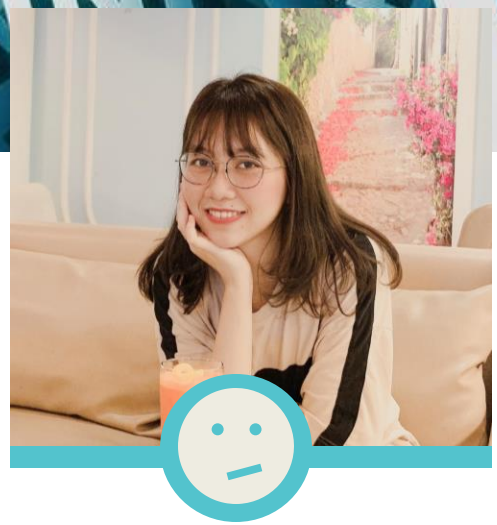
Important Features





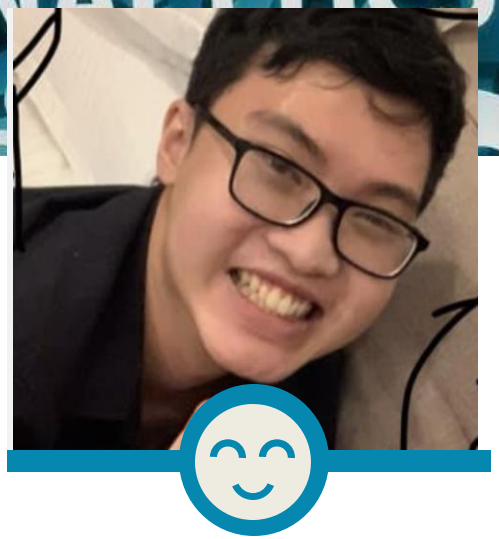




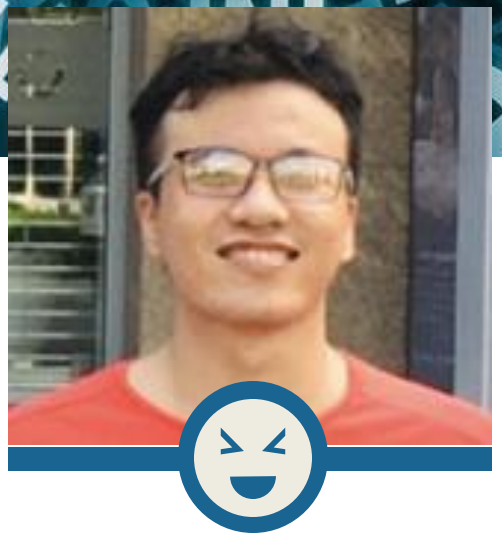


Bùi Thị Ngọc Trâm

Nguyễn Văn Đức



Phạm Văn Ngoan



Nguyễn Hồng Minh Nhật

Who we are?


Thank you!