# Classification of risk attitudes from customer behavior with machine learning

Mr. Teeranai Sriparkdee



A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2018

# การจำแนกทัศนคติต่อความเสี่ยงจากพฤติกรรมของผู้บริโภคด้วยการเรียนรู้ของเครื่อง

นายธีรนัย ศรีภักดี



วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561

| | |
|---|---|
| Thesis Title | Classification of risk attitudes from customer behavior with machine learning |
| By | Mr. Teeranai Sriparkdee |
| Field of Study | Computer Science |
| Thesis Advisor | Professor PRABHAS CHONGSTITVATANA |

Accepted by the Faculty of Engineering, Chulalongkorn University in Partial Fulfillment of the Requirement for the Master of Science

..... Dean of the Faculty of Engineering

(Associate Professor SUPOT TEACHAVORASINSKUN)

THESIS COMMITTEE

..... Chairman

(Assistant Professor SUKREE SINTHUPINYO)

..... Thesis Advisor

(Professor PRABHAS CHONGSTITVATANA)

..... External Examiner

(Associate Professor Worasait Suwannik)

ธีรนัย ศรีภักดี : การจำแนกทัศนคติต่อความเสี่ยงจากพฤติกรรมของผู้บริโภคด้วยการเรียนรู้ของเครื่อง. ( Classification of risk attitudes from customer behavior with machine learning) อ.ที่ปรึกษาหลัก : ศ. ดร.ประภาส จงสถิตย์วัฒนา

ทุกผลิตภัณฑ์และบริการในตลาดมีลักษณะที่มีผลกระทบต่อการตัดสินใจของผู้บริโภคในการซื้อหรือใช้งาน ความเสี่ยงเป็นลักษณะพิเศษของผลิตภัณฑ์ทางการเงินดังนั้นในการออกแบบผลิตภัณฑ์และบริการทางการเงินต้องใช้ความเสี่ยงเป็นปัจจัยสำคัญ ในทางกลับกันผู้บริโภคมีทัศนคติต่อความเสี่ยงที่แตกต่างกันซึ่งสามารถแยกความแตกต่างใน 3 ประเภท ได้แก่ การหลีกเลี่ยงความเสี่ยงความเป็นกลางและความเสี่ยง ดังนั้นการรู้ทัศนคติความเสี่ยงของผู้บริโภคที่เป็นตลาดเป้าหมายจึงเป็นกุญแจสำคัญในการกำหนดกลยุทธ์ทางการตลาดเช่นการออกแบบบริการและผลิตภัณฑ์การรณรงค์และการส่งเสริมการขายซึ่งจะเสนอให้พวกเขา มีสองวิธีในการทราบทัศนคติความเสี่ยงของผู้บริโภค วิธีแรกคือผ่านแบบสอบถามที่ผู้บริโภคทำด้วยตนเองและวิธีที่สองคือผ่านพฤติกรรมของพวกเขาที่สะท้อนทัศนคติเสี่ยงของพวกเขาจากกิจกรรมในชีวิตประจำวัน ด้วยวิธีที่สองการเรียนรู้ของเครื่องมีบทบาทสำคัญในการจำแนกทัศนคติความเสี่ยงของผู้บริโภคแต่ละรายและการเรียนรู้ของเครื่องบางอย่างเช่น Ensemble สามารถระบุคุณลักษณะหรือพฤติกรรมของผู้บริโภคที่มีความเสี่ยงต่อทัศนคติของพวกเขา ในบทความนี้เราศึกษาและทดลองเพื่อจำแนกทัศนคติความเสี่ยงของผู้บริโภคจากพฤติกรรมของพวกเขาและระบุคุณสมบัติที่สำคัญด้วยวิธี Ensemble



สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่อนิสิต .....
ปีการศึกษา 2561 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

# # 6070927021 : MAJOR COMPUTER SCIENCE

KEYWORD: Risk Attitudes, Importance feature, customer behavior, Ensemble Model, Random Forest, Gradient Boosting, XGBoost

Teeranai Sriparkdee : Classification of risk attitudes from customer behavior with machine learning. Advisor: Prof. Dr PRABHAS CHONGSTITVATANA

Every product and service in the market has its characteristic which has an impact on a consumer's decision to buy or use them. The risk is a special characteristic of financial products, so in financial product and service design must use risk as a critical factor. On the other hand, the consumer has different attitudes to the risk which can distinguish in 3 categories viz risk aversion, risk neutral and risk seeking. Therefore, knowing risk attitudes of consumer who is the target market is an important key to define marketing strategy such as designing service and product, campaign, and promotion which going to be offered to them. There are two ways to know the consumer’s risk attitudes. The first way, is via a questionnaire which consumer do it by themselves and the second way, is via their behaviors which reflect their risk attitude from their activities in everyday life. With the second way, machine learning takes a vital role to classify risk attitudes of each consumer and some machine learning such as Ensemble can specify the features or consumer’s behaviors which dominant to their risk attitudes. In this paper, we study and experiment to classify consumer’s risk attitudes from their behaviors and specify importance features with Ensemble method.



Field of Study: Computer Science Student's Signature .....

Academic Year: 2018 Advisor's Signature .....

# ACKNOWLEDGEMENTS

Firstly, I would like to express my sincere gratitude to my advisor Prof. Dr. Prabhas Chongstitvatana for the continuous support of my thesis study, give good suggestions and great ideas which I can use them in my thesis and in the real life.

Thank you, Assistant Professor Sukree Sinthupinyo who is the chairperson of a thesis committee and Assistant Professor Worasait Suwannik who is a thesis committee for terrific suggestions and guidelines for this thesis.

Thank you all of my teachers to give me the knowledge and give me the chance me to grow up. You have made me what I am today.

Thank you all of the obstacle which I faced for teaching me to learn how to solve the problem and how to overcome obstacles.

Finally, I must express my very profound gratitude to my family and to my colleagues for providing me with the great support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them. Thank you.

Teeranai Sriparkdee

# TABLE OF CONTENTS

Page

# LIST OF TABLES

Page

# LIST OF FIGURES

Page

# Chapter I

# Introduction

## 1.1 Statement of the problems

We live in a world full of information. Every second, we receive gigantic data via our notebook, mobile phone, and social network. Sometimes we call this phenomenon that information overload and we cannot memorize all information which is received so we will try to choose the information for remembering. The key to success is the firm need to gain a 360-degree view of customers which include their daily life and changes that occur during their lifetimes for offering the right product to the right person at the right time. (Philip Kotler, 2012)

Seth Godin said that “You don't find customers for your products. You find products for your customers.” this quote remind us that the firm needs to produce and offer products or services which deliver the value to customers and to do that the firm must know them. Financial products are products and services provided by financial firms to customers such as deposit products, debt instruments, funds, equities, stocks, and derivatives, etc. Which each customer have his attitude toward risk, and each product has a different level of risk, such as the fund type which can be divided into eight levels of risk according to Table 1.



*Table 1 Type of funds and risk level.*

| Type of investor | Risk level | |
|---|---|---|
| Low risk | 1 | Domestic money market fund |
| Low to moderate risk | 2 | International money market fund |
| | 3 | Bond fund |
| | 4 | Debenture fund |
| Moderate to high risk | 5 | Mixed fund |
| High risk | 6 | Equity fund |
| | 7 | Sector fund |
| Very high risk | 8 | Alternative fund |

With the different risks of each financial product and the attitude toward risk or risk attitudes of each customer affects their decision to choose the firm's financial products and services. Therefore, if the firm can know the attitude towards the risk of each customer, it will be able to develop the service and commercial products, including campaigns and valid promotion to each customer in line with the attitude towards risk of target customers.

Generally, in finance and economics, there are three types of attitudes towards risk:

1. risk aversion
2. risk neutral
3. risk seeking

To be aware of the risk attitude of the customer, the firm can do so by doing the questionnaire about attitude to investment risk by customers when they purchase investment products from the firm. Therefore, it is possible to evaluate the risk attitude only when the customer comes and purchases the investment product from the firm. While if customers do not have a history of buying investment products from the firm, they will not be able to evaluate the risk attitude of such customers at all. Another disadvantage of this approach is the bias of customers when they do questionnaires for self-assessment; thus, the result from this approach may not reflect real risk attitude of customers. Using customer behavior can handle these issues with this approach the firm can specific or classified their customers' risk attitude from their act which frees from customers' bias and can use customer behavior of historical to specify the risk attitude of the new customer who never does the questionnaire.

Due to many factors of consumer behavior, machine learning is used to classify risk attitudes and specify features or consumer behavior that affects the risk attitude. With this method, the firm can classify customers according to the level of risk attitude and specify it although they never purchase any financial products from the firm and can introduce products and services according to consumers' risk attitudes which reflect from their behaviors. The machine learning method with ensemble such as random forest or boosting such gradient boosting and XGBoost can be used for this

problem. They can classify customer into each risk attitude due to their behavior information, and at the same time, they can identify important features that effect to the classification of customer. Machine learning methods require positive and negative training data. For the result to be accurate, both type of data should be balanced, that is, they are approximately equal in the number of data. However data in the real world can be imbalance so to select the method and create a model which has the best performance for the problem it necessary to address data imbalance problem. The resampling technique is one way to solve this problem which we will perform the experiment to find the best resampling technique and that is suitable for the model to give the best performance in the classification task.

## 1.2 Objective

To study and present machine learning method to create a model for classify and identify patterns from consumer behavior to specify customers' risk attitude and find factors that have a significant impact on it.

## 1.3 Scope of study

1.3.1 Data which use in this research comprise 500 features.

1.3.2 Dividing the risk attitude into eight levels with risk levels ranging from one to eight, where one is the lowest level of the risk attitude (risk aversion) and eight is the level of the highest risk attitude (risk seeking).

## 1.4 Expected or anticipated benefit gain

1.4.1 Get an effective and productive model in identifying risk attitudes by using consumer behavior for the firm can offer or design financial products and services which respond to the target consumer precisely and correctly.

1.4.2 The result reflects the characteristics of consumer behavior, which is an important factor affecting the risk attitude.

1.4.3 Apply knowledge and principles from modeling adapt to other business operations.

## 1.5 Research methodology

1.5.1 Study related documents and research.

1.5.2 Data exploratory analysis.

1.5.3 Study the machine learning processes to create models for classifying risk attitudes by using consumer behavior.

1.5.4 Design the experimental process and measurements to evaluate the efficiency and effectiveness of the model.

1.5.5 Develop the machine learning process and create models for use in classifying risk attitudes towards by using consumer behavior.

1.5.6 Evaluate the model of classification using consumer behavior and collecting data from the test.

1.5.7 Conclusions.

1.5.8 Compiled and prepared academic articles.

1.5.9 Compilation and dissertation.

# Chapter II

# Literature Review

## 2.1 Related theories

### 2.1.1 Machine learning

Machine learning is the science and art of writing programs for computers to be able to learn from the information. Arthur Lee Samuel, an American computer scientist, has given the machine learning definitions that machine learning is subjects that make computers capable of learning without having to write additional programs. Tom Michael Mitchell, an American computer scientist, has given another definition of machine learning in an engineering perspective that it is computer programs that can learn from experiences which relevance to mission. For example, a task such as filtering electronic spam, machine learning will have a mission to identify spam when new electronic mail arrives. The information in the past that has been identified as electronic mail as spam or not is used as training data. The performance measurement will be defined as the accuracy of classification result. Machine learning can be categorized into four main categories:

1. Supervised learning is a learning process which the training data that will be used in the modeling process has a feature which tells the class label value. The model uses such features in learning patterns to classify information in the future. Examples of learning that use supervised learning such as classification, prediction, and regression.
2. Unsupervised learning is a method that Training data, which is used to create models, does not have a feature that tells the sample type value. Therefore it cannot determine which data values in each row are classified or how many groups are there. Modeling of such learning method will use the features which exist in the sample data to identify the patterns of each group and assign the group identity to the data. With this method, the model divide data into groups. Examples unsupervised learning are clustering and learning to find association rules.

3. Semi-supervised learning uses the characteristics of the data set that has the mix of both known and unknown type information in the same data set. This method use the supervised learning and unsupervised learning together by using unsupervised learning to specify the type of sample value for the data, which is not specified. After using unsupervised learning, every row of data has been specified, supervised learning method will be used to specifying the type of the rest of sample.
4. Reinforcement learning is a learning method that has an agent which can observe the environment and then select and perform actions that will get a reward in the event that the action is correct but if the action which agent chooses is incorrect, it will receive a penalty. So, the agent will try to take action which give the most reward to it.

### 2.1.2 Decision tree (Ian H. Witten, 2011)

The decision tree is a machine learning process which is supervised. It aims to classify data into classes by using features of data. The decision tree consists of the following structure.

- The internal node is a feature of data which used to divide data.
- The root node is the internal node which is the beginning of the decision tree.
- The leaf node is the result or class from using decision trees for data classification.
- The branch is the possible value of the characteristics of the internal node and the number of branches is equal to the number of attributes of the internal node

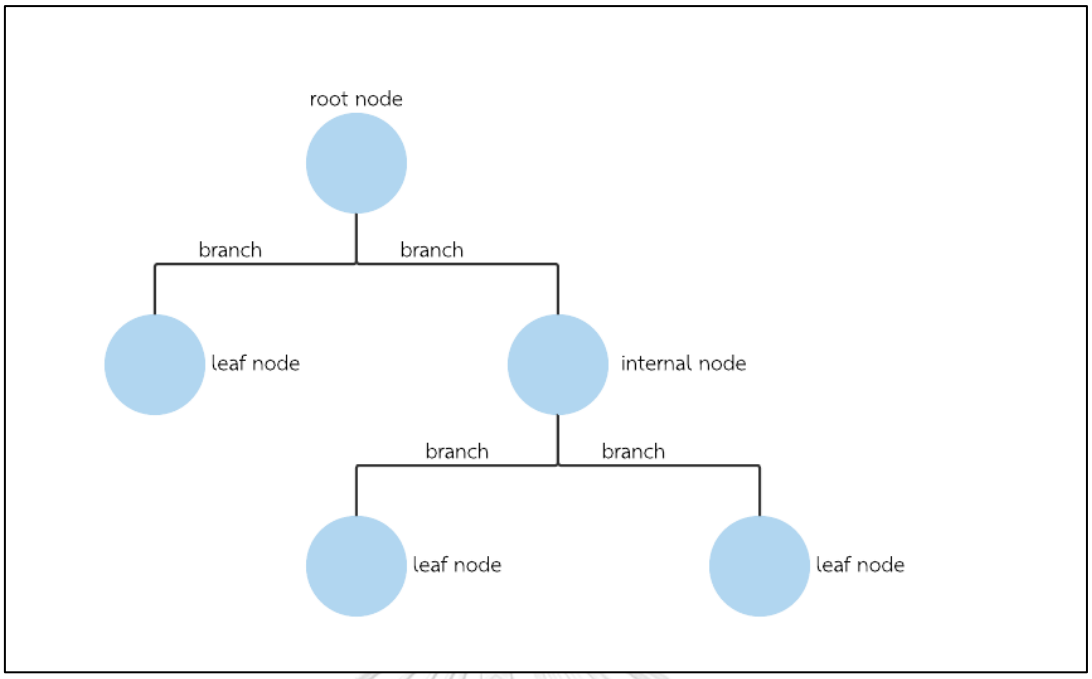



*Figure 1 Components of the Decision Tree.*

In practice, the nodes in the decision tree are created by using greedy search and top-down approach. The decision trees are created by starting with choosing the best feature to use as a root node. Then the data is divided by the root node. The next step is to find the best features to split the data further until when the data is fully classified, or the amount of data in one branch is less than the predefined. There are three popular algorithms, namely ID3 (Iterative Dichotomiser), C4.5 and CART (Classification and Regression Trees). All three algorithms have different methods to choose a feature or attribute selection to be used in the creation of trees.



### *A. Gini Criteria*

Gini Criteria is a selection process for features that will be used to split the data in the decision tree by using the Gini index or Gini coefficient as a measure. The Gini index as a statistical measure for the distribution of information commonly used to indicate the Inequality of income distribution. Corrado Gini invented the Gini index. If the Gini index is high, it shows the inequality of high distribution of income or data is very distributed. Formula to calculate the Gini index shown below.

$$Gini(D) = 1 - \sum_{i=1}^{m} p_i^2$$

Gini (D) = Gini value of data D

$p_i$ = probability that data of D is in class i

m = the total number of classes

In the case of data samples, D is binary split with feature A. D is divided into D1 and D2. Gini index values from splitting D data by feature A will be

$$Gini_A(D) = \frac{|D_1|}{|D|} Gini(D_1) + \frac{|D_2|}{|D|} Gini(D_2)$$

$Gini_A$ (D) = Gini value after dividing D with feature A

| D | = data size before splitting

| $D_1$ | = the size of the data that is divided into class 1

| $D_2$ | = The size of the data that is divided into class 2

Gini ($D_1$) = Gini value of data that is divided into class 1

Gini ($D_2$) = Gini value of data that is divided into class 2

Find impurity value or Gini value that change from using the feature A split data D.



$$\Delta Gini(A) = Gini(D) - Gini_A(D)$$

$\Delta$Gini(A) = Gini values that have been changed after splitting data D with feature A

Gini (D) = Gini value of data D

$Gini_A$ (D) = Gini value after splitting D data with feature A

The feature that makes the most change value of impurity values or has the least Gini index value will be selected is a feature that will be used to split the data in the decision tree. The feature selection process using the Gini index is the process which applies in the CART algorithm.

*B. Gain value*

Gain is a feature selection process used in the ID3 algorithm. It is based on the information theory of Claude Shannon by using information value of data, the entropy of data and probability.

$$\text{Info}(D) = -\sum_{i=1}^{m} p_i \log_2 (p_i)$$

info (D) = Information value of data D

$p_i$ = probability that data of D is in class i

m = the total number of classes

Using log2 to find out how many bits to use for encoding information when using the feature A to split data D.

$$\text{Info}_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \text{Info}(D_j)$$

$\text{Info}_A$ (D) = Information value from splitting data D with feature A

$|D_j|$ = size of data that is splitted into class j

|D| = data size

Info ($D_j$) = Information value of data that is splitted into class j

v = the total number of classes



The gain is the value of information values that change after using the feature A in splitting data D are calculated from

$$\text{Gain}(A) = \text{Info}(D) - \text{Info}_A(D)$$

Gain (A) = gain from using feature A for splitting data D

Info (D) = Information value of data D

$\text{Info}_A$ (D) = Information value from splitting data D with feature A

The feature which gives the highest gain value will be selected for split data in the decision tree.

*C. Gain ratio*

The problem of using information gain in selecting the features to split the data in the decision tree is the bias to select features that have many possible values, such as the identification number of employee. The result is that data in each class or node after splitting will consist of one sample. This result will give the information gain the highest value because of pi is one that makes log2 ($p_i$) is zero. So, the algorithm will choose this feature as the feature for split data in the decision tree, which is not useful for classification at all.

In order to resolve the bias, the C4.5 algorithm uses the gain ratio process to select features that will be used to split the data by finding a gain ratio and use standardization (normalization) with split information.

$$\text{SplitInfo}_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right)$$

$\text{SplitInfo}_A$ (D) = split information value from splitting data D with feature A

| D | = data size D

| $D_j$ | = the size of the data that is splited into class j

v = the total number of classes

And normalize the information gain with split information.

$$\text{GainRatio}(A) = \frac{\text{Gain}(A)}{\text{SplitInfo}_A(D)}$$

GainRatio (A) = the gain ratio from splitting data D with feature A

Gain (A) = value of information gain of feature A

SplitInfoA (D) = split information value from splitting data D with feature A

With this process, the features that have many possible values will increase the value of split information which results in a decrease in the gain ratio. The feature with the highest gain ratio will be selected as the splitter in the decision tree.

### 2.1.3 Ensemble Model ("Ensemble methods ", 2018; Geron, 2017)

The ensemble is a learning method that uses multiple learning models together. In classification, we called these learning models which are used in the ensemble model *base classifiers*. The ensemble receives a class prediction or classification from each base classifier and use a majority vote to choose the answer. With this method, the mistake from some base classifier will not affect the classification as long as the majority of base classifier can still classify correctly.

The ensemble method works well when the base classifiers have diversity or each base classifiers have less correlation to each other. Because of using non-diversity base classifiers will have the same or resemble algorithm and structure that will give a tendency to predict the same result in classification. Therefore the result will not be different from using only one classifier.

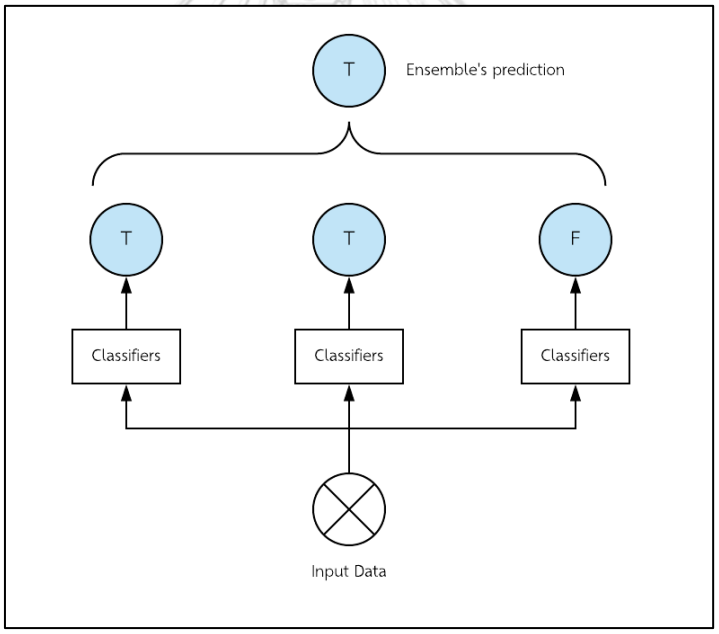



*Figure 2 Bagging method*

The way to increase the diversity of base classifiers are:

1. Base classifiers must have different algorithm from each other. For example, In one Ensemble model may be consist of base classifiers which use algorithm SVM, Logistic Regression, and Random Forest works together.

2. Using different training data by using the bootstrap aggregation (bagging) process.

### 2.1.4 Random Forest

Random Forest is an algorithm for the learning of ensemble which is based on decision trees. Because the decision tree has difficulty when working with anomalies data. Such anomalies dominate classification with decision trees and caused overfitting. To resolve this limitation, Random Forest using the average method for numerical classes and the majority voting for categorical classes. Also, data which is used as training data in each decision tree in Random Forest will be randomly drawn from all data or each decision tree in Random Forest will use different data to create. With this way, Random Forest can solve dominant from anomalies data.

#### *A. Bootstrapping*

Bootstrapping is a technique which uses to random the new data set and still, maintains the size of the new data set equal to the size of the original data by using the random sample with replacement. Such as a set of original data consisting of ('A', 'B', 'C', 'D', 'E') which size is five, in bootstrapping to create a decision tree will use the random sample with replacement from original data five times. Therefore, the information that will be used may be (‘B’, ’B’, ’D’, ’C’, ’C’). In this way, each decision tree in Random Forest is created by different data but have the same size.

With this process, the base classifier will have diversity and when combined with the majority vote in classification, the influence of anomalies data that dominates classifications has decreased.

#### *B. Criteria Selection*

In the algorithm to create the decision tree, the features selection will be used to split data to each node in the decision tree will choose the feature which gives the highest gain value. So, if in the decision tree creation in Random Forest, each decision tree uses all feature or all of decision tree use the same features. It means that all decision tree in Random Forest will have the same structure.

In order to avoid this problem, Random Forest uses random selection some features for use them to create each decision tree to prevent the creation of all decision tree from the same features. So, each decision tree of Random Forest will have a different structure from each other.

### *C. Out Of Bag Errors*

In creating a machine learning model, the data will be divided into two parts: training data and test data. After training the machine learning algorithms with training data. Test data will be used to measure the performance of the model. This process is the cross-validation process

In the case of small data affect the cross-validation process. Since the process must divide data into two parts this makes data after divide too little to use for train or test model effectively.

Random Forest uses the bootstrapping method to generate data for each decision tree randomly. From this method, there will be some data which is not selected called these data that out of bags of decision tree creation and Random Forest use these data as the test data of that decision tree and collect the error from each decision tree of Random Forest. The result of average the error is out of bag errors which are used as indicators for the performance of Random Forest by if the value of out of bag errors increase it means that the performance of Random Forest decrease.



### *D. Importance Feature*

One random forest capability is to be able to determine which features are important features of the data. It finds important features by using the gain value, with the following steps

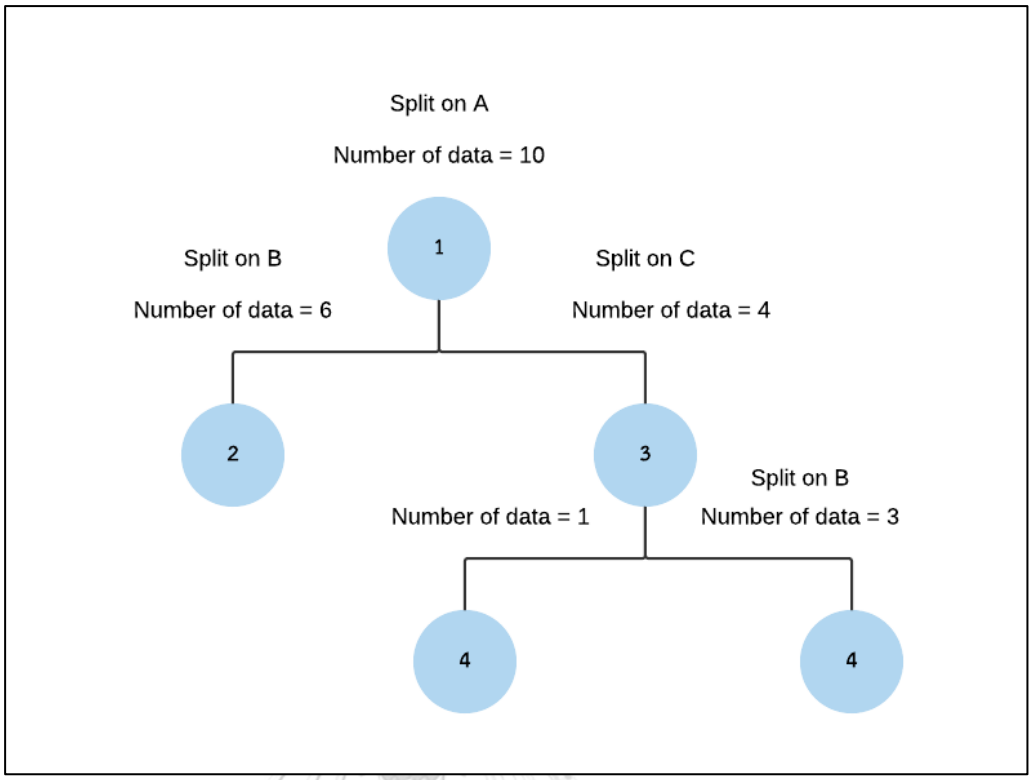



*Figure 3 The Decision Tree*

From the figure, within the decision tree, there are 10 records, and the feature which will be used to split data 3 features, namely A, B, and C, with the decision tree splitting all data 4 times. Calculating the value of features importance are as follows.

*Table 2 Importance value calculation*

| Times Of splitting | Size Of Data (1) | Feature | Gain (2) | Importance Value (1) X (2) |
|---|---|---|---|---|
| 1 | 10 | A | 0.26 | 2.6 |
| 2 | 6 | B | 0.4 | 2.4 |
| 3 | 4 | C | 0.3 | 1.2 |
| 4 | 3 | B | 0.1 | 0.3 |

Find that feature B is used to split the data twice. So, we will sum the importance value of feature B. Therefore, the importance value of feature B is 2.7. After that, we will normalize the importance value of each feature with the sum of importance value of all feature in the decision tree.

*Table 3 Normalization importance value calculation*

| Feature | Importance Value | Importance Value After normalization |
|---|---|---|
| A | 2.6 | 0.4 |
| B | 2.4 | 0.42 |
| C | 1.2 | 0.18 |

From the above table, it can be determined that feature B is an important feature for splitting data rather than other features and to find the value of the important feature of Random Forest we use the average value of the important features of all features within every decision tree in the random forest.

### 2.1.5 Boosting

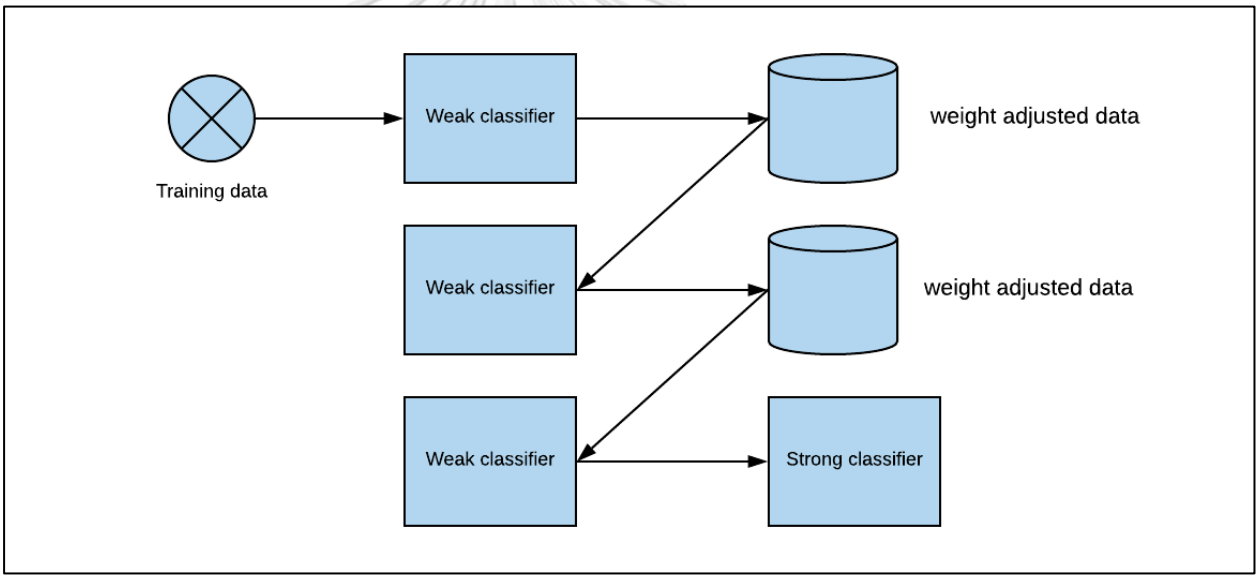



*Figure 4 Boosting method*

Boosting or hypothesis boosting is another ensemble method which combines several weak learners into a strong learner. The concept of this method is to train classifier or predictor sequentially and each classifier or predictor try to correct its predecessor. A classifier must pay a bit more attention to the data which underfit by a predecessor. Adaboost (Adaptive Boosting) uses this approach as follow, the first base classifier is trained and after this step weight of data which is misclassified will be adjusted or the weight get a boost and then the second base classifier will use this data as the training set and so on. The equation of Adaboost is shown below.

$$F(x) = \propto_1 f_1(x) + \propto_2 f_2(x) + \cdots + \propto_n f_n(x)$$

F(x) = a strong classifier

f(x) = a weak classifier

$\alpha$ = weight

Another popular Boosting algorithm is Gradient Boosting. It is different from Adaboost because it does not adjust the weight of misclassified data but it tries to fit the new classifier by using the residual errors made by a predecessor or calculate the gradients in the loss function from the predecessor. The Gradient Boosting use gradient descent to minimize loss when adding new weak learner into the model.

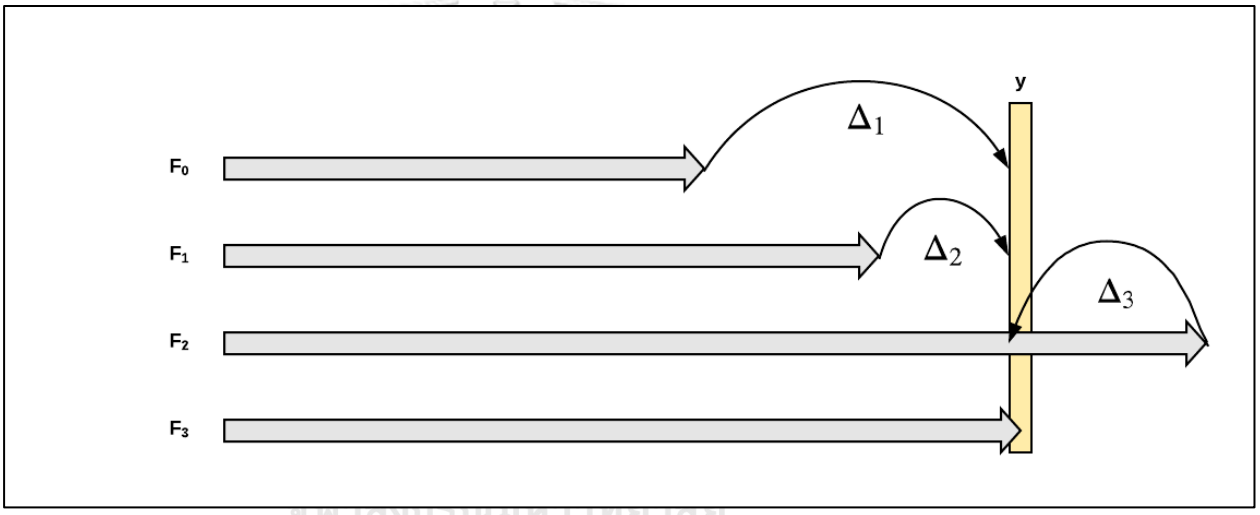



*Figure 5 Gradient descent process*



Figure 5 shows the learning process of Gradient Boosting by use error from predecessor weak learner to create better learner from the last learner and we can describe this process in the form of an equation as below.

$$F_m(X) = F_{m-1}(X) + \eta \Delta_m(X)$$

m = stage or the sequence of weak learner

$F_m(X)$ = prediction of weak learner number m

$F_{m-1}(X)$ = prediction of weak learner number m - 1

$\eta$ = learning rate

$\Delta_m(X)$ = the function to measure the quality of a split

The parameter $\eta$ is the learning rate. It has a strong relationship with the number of the weak learner in Boosting. If we set small $\eta$ it means that we need more iteration or the number of the weak learner in the sequence of learning. This technique is the regularization technique that makes the model more general. We called this technique *shrinkage*.

### 2.1.7 Bias and variance (NG, 2019)

Bias and variance are significant sources of error in machine learning. To improve model performance, it necessary to understand them. The bias is an error rate when the algorithm does on the training set, while variance is an error rate when the algorithm does on the test data. So, from their definition, we can use them to specify procreation of overfitting and underfitting. When we found that result from the algorithm has high variance and low bias. It means that this algorithm works well on training data but fail to generalize to the test data that indicate that this model is overfitting whereas if the result has low variance and high bias this means that this algorithm works better on test data than training data. In this situation, we can specify that this model is underfitting. In case of low bias and low variance is the ideal situation, it means that the algorithm works well on both training data and test data or it has an excellent performance.

Bias consists of two components : avoidable and unavoidable bias. Unavoidable bias or optimal error rate also called Bayes error rate. It means the lowest possible error rate for any classifier. So it means that we cannot improve this type of bias, but in case we found high avoidable, we can improve by these approaches.

1. Increase the size of the model, such as adding layers or neurons in the neural network, although this approach can reduce bias, it is the cause of the increasing variance lead to overfitting problem because the model tries to fit training data. So, we must use regularization, which eliminates the increasing of the variance along with this approach.
2. Add more features which help algorithm eliminate bias like the first approach with this approach we can eliminate the bias, but it increases

variant. So regularization should be used when we found that adding more features lead to overfitting problem.

3. Add more features which help algorithm eliminate bias like the first approach. With this approach we can eliminate the bias, but it increases variance. So regularization should be used when we found that adding more features lead to overfitting problem.
4. Reduce or eliminate regularization. With this approach, we can reduce bias but increase variance simultaneously.
5. Modify model architecture. This makes the model appropriate for the problem.
6. Add more training data. This approach might help reduce variance more than the effect on bias.

Whereas, if we found that the model gets a high variance, we can use approaches below to eliminate it.

1. Add more training data. It is the easiest but reliable way — cons of this approach is that it affects computational power to process data.
2. Add regularization (L2 regularization, L1 regularization, dropout). Though this approach can reduce the variance, it is the cause of bias increase.
3. Feature selection or decrease number of feature. This approach might help decrease variance, but it increases bias at the same time.
4. Decrease the model size. This approach decreases the model complexity, and it affects to decrease variance, but when model complexity decrease, it means that bias might increase.

Besides, we can use modify or add more feature and use regularization simultaneously. Modifying model architecture is a great approach because it can decrease variance and bias at the same time.

### 2.1.8 Learning curves

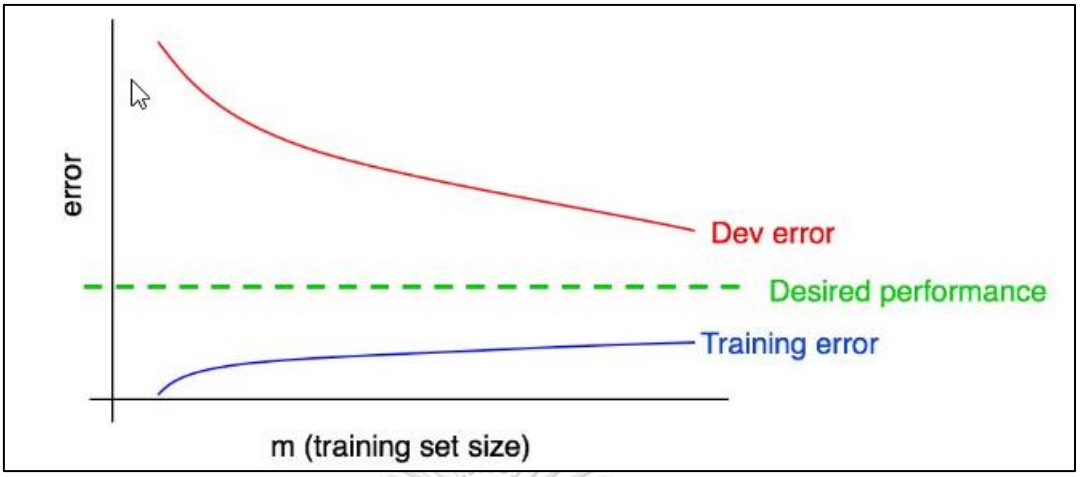



*Figure 6 (NG, 2019) Learning curve*

A learning curve is a graph which plot relation between an error from the model and the number of the training set. To plot the learning curve, we must run the algorithm with different training set sizes.

The learning curve above shows that when the training set size increase, the variance or error from the test set should decrease. Sometimes we can define the desired error rate that is the target of the learning algorithm to achieve it might be the human-level performance. if we found that curve from the training set has plateaued, it means that adding more data will not improve the learning algorithm.



Increasing of training set size means by usually will decrease variance but increase bias. So, if we plot the error of the training set or bias in the learning curve, it will increase when the training set size increase. Furthermore, the learning algorithm usually does better on training data than test data, so the curve of bias often lies above the curve of variance.

### 2.1.9 Correlation analysis (Agresti, 2018)

When we want to study the relation between two variables we often create a scatter diagram and analyze relation from that diagram. In addition, there is a group of statistics which is used to measure the relation of two variables called correlation analysis.

The correlation coefficient is one of correlation analysis originated by Karl Pearson about 1900. It is a measure of the strength of the linear relationship between two variables. It describes the strength of the relation between two variables with Pearson's r. Pearson's r has a value between -1 to 1. A correlation coefficient of -1 or 1 indicates perfect correlation. If a correlation coefficient has negative value, it means that both variables have negative relation when variable increase another will decrease and if a correlation coefficient has a positive value means that both variables have positive relation when variable increase another will increase too. When both variables are not related to another, the correlation coefficient is 0.

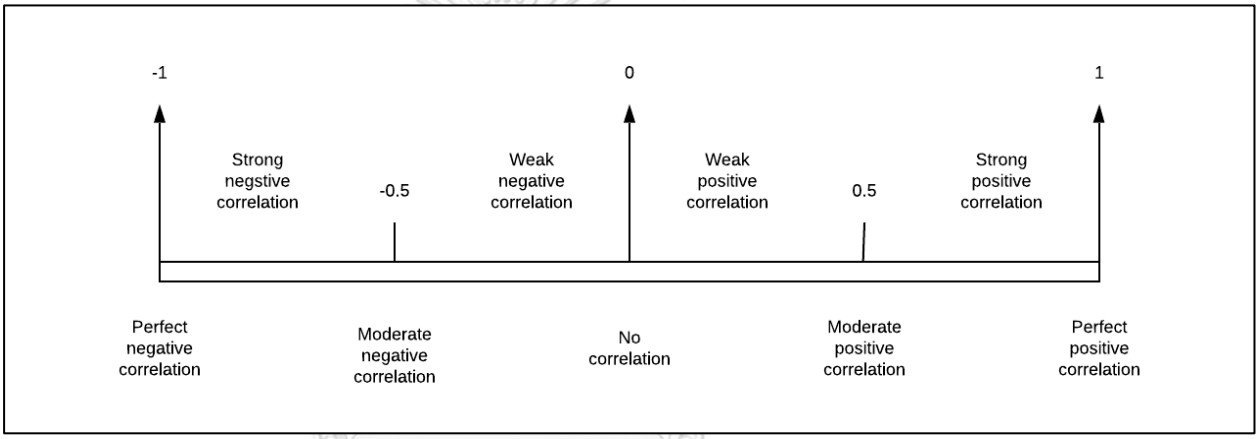



*Figure 7 (Lind, 2018) Range of Pearson correlation coefficient*

We can compute Pearson's r from standard deviations and mean of variables as the equation below.



$$r = \frac{\sum(x - \bar{x})(y - \bar{y})}{(n-1)s_x s_y}$$

n = number of example

$\bar{x}$ = mean of variable x

$\bar{y}$ = mean of variable y

$s_x$ = standard deviations of variable x

$s_y$ = standard deviations of variable y

## 2.2 Related works

### 2.2.1 Large-scale Ensemble Model for Customer Churn Prediction in Search Ads (Hussain., 2018)

This research presents the use of a gradient boosting decision tree (GBDT), which is learned by the Ensemble method to predict the cancellation of the service (churn) platform so that the company who owns the platform knows how to prevent customers from canceling the service. The company needs to know which features are essential for deciding to stop using services. Therefore, in the research, therefore, choose the gradient boosting decision method which can identify and rank the importance of the feature instead of using deep learning, which can be more accurately predicted but cannot identify features or factors that affect the decision.

With all 898 features that have been used in the experiment, the features are categorized into two types: static features such as customer type, customer address and the budget that will be used in the advertising of customers, etc. and dynamic features such as the number of times users have clicked on ads, the cost that customers are collected from the company and the number of times the ad appears on the platform ads. To avoid class imbalance due to the positive nature of the data, the number of customer who canceled the service is less than the negative class or customers who are still using the service. Therefore, resampling the negative class to reduce the ratio between the data with the positive class and negative classes.



In the performance measurement of the GBDT model in this experiment, they use the area under curve (AUC) which is calculated from the receiver operating characteristic curve (ROC). The ROC graph is constructed from true positive rate and false positive rate. AUC values are precious, reflecting the probability that the model will give higher accuracy results. After the experiment, using all the features, it was found that the AUC value was 0.8410 and if only one dynamic feature were used, the AUC value was 0.8236, and the only fixed feature would be the AUC of 0.8215.

### 2.2.2 Estimating Customer Lifetime Value Using Machine Learning

This research aims to use machine learning to predict values throughout the life span of a customer lifetime value and factors affecting the customer lifetime value

of passengers using airline services with feature more than 300 features. XGBoost system is a machine learning system that uses the gradient boosting the algorithm, which is the learning of the Ensemble uniform. There is a decision tree as base classifiers.

XGBoost is a machine learning system that is scalable can perform multithread and can work in parallel; therefore, it suitable for working with large amounts of data.

In the experiment, the data of 240,000 passengers was used, and by using XGBoost, it was able to identify essential features that affect customer lifetime value and in performance measurement using the root mean square error (RMSE) method. RMSE value is 13.02934.

2.2.3 Benchmarking sampling techniques for imbalance learning in churn prediction. (Bing Zhu, 2016)

Because in fact, customers who are canceling the service are tiny compared to all customers, such as in the telecom business, customers who will cancel the service are the events that very rare or the number of customers to cancel the service is different from customers who still use the service very much — resulting in an imbalance between the number of customers who cancel the use, which is a minority class and customers who still use the service, which is the majority class. The impact of unbalanced data usage on learning algorithms for classification will affect to accuracy and precision of classifying the minority class and may classify all data into the majority class.

There are many ways to solve the problem which can divide the solution into two types:

1. Data-level solutions try to solve problems by resampling in data preprocessing (data-level solutions) before learning to create models.
2. Problem-solving solutions try to solve at the algorithm level (algorithm-level solutions) by developing new algorithms or improve existing algorithms to handle unbalanced data.

Solving the problem at the algorithm level is necessary to have much knowledge about the learning algorithm, while data-level solutions do not require such knowledge which makes it easy to use. Therefore data-level solutions become more popular and are independent from the classifier, allowing it to be used with a variety of learning algorithms. In this research compares the use of resampling techniques with different classifiers for predicting customers to cancel the service.

1. Random oversampling (ROS) is a technique that will randomly select data which is a minority class to replicate it and added it to the original data. With this technique, there will be an increase in the number of minority classes and make the class of data balance. The advantages of this techniques are that they are easy to use, and easy to understand. The disadvantage is that when the data is replicated, it increases the possibility of overfitting problems, the model can accurately classify the data, but not when used with other data. This results in the model that has lower accuracy in data classification.
2. Random undersampling (RUS) is a technique which randomly selects the majority class to delete it from the original data. The advantages of this technique are that they are easy to use and to understand, but with a random method to remove the data from the original data, sometimes information that is useful or important is removed.
3. Synthetic minority oversampling technique (SMOTE) begins to randomly select the nearest neighbors of one or more data and then generate data based on linear interpolations between the initial data with nearest neighbors that are randomly generated. SMOTE will generate synthetic minority class in the same amount of every initial data that contains the class as a minority class. There is no check that the nearest neighbors are the majority class or not. So this technique may increase more overlap between classes.
4. Adaptive synthetic sampling (ADASYN) to solve SMOTE's weaknesses which cause overlap between classes. This technique uses density distribution.

Starting from finding the nearest neighbors of each data, which has minority class and then count the amount of the nearest neighbors which has majority class and create a ratio between the number of nearest neighbors which have minority class and the number of nearest neighbors which have majority class for uses this ratio to create the synthetic sample.

5. Borderline-SMOTE try to find minority samples located and find the borderline between majority and minority samples and to synthetic minority class on the borderline because models or classifier may misclassify them.
6. Most weighted minority oversampling technique (MWMOTE) gives weight to data which has a class as minority class according to the distance from the nearest data which has majority class and generates synthetic minority based on the weight that is assigned to the clustering approach.
7. SMOTE-Tomek will use the SMOTE method to create synthetic minority. After that, Tomek links are used to find the nearest pairs of samples from different classes and then eliminating the data which has the majority class from this pair.
8. SMOTE-ENN like SMOTE-Tomek but use Wilson's edited nearest neighbor (ENN) to delete data after random data with SMOTE.
9. Cluster-based undersampling (CLUS) group data using the same characteristics and then reduce the size (downsize) of data that has the majority class in each cluster.

# Chapter III

## Research methodology

Data used in this research includes 20,000 records and it has about 500 features. We use customers' data who have purchase fund profile from the firm and use it to calculate risk attitude that is used as a class or label. In this experiment, we group risk attitudes into three groups : risk-averse, risk neutral and risk seeker. After data pre-processing, we found that data is imbalanced. Learning on data like this leads to the classifier that classifies all majority examples correctly but misclassifies minority examples. Classifier's accuracy is high from the number of majority example more than minority example, but it cannot reflect the performance of classifier to classify minority samples.

In this research, we choose to solve the imbalance problem at data level because it is easier to implement and it can use multiple classifiers. We use this method in the data preparation process.

Overall of the experiment in this research consists of steps as the figure below.

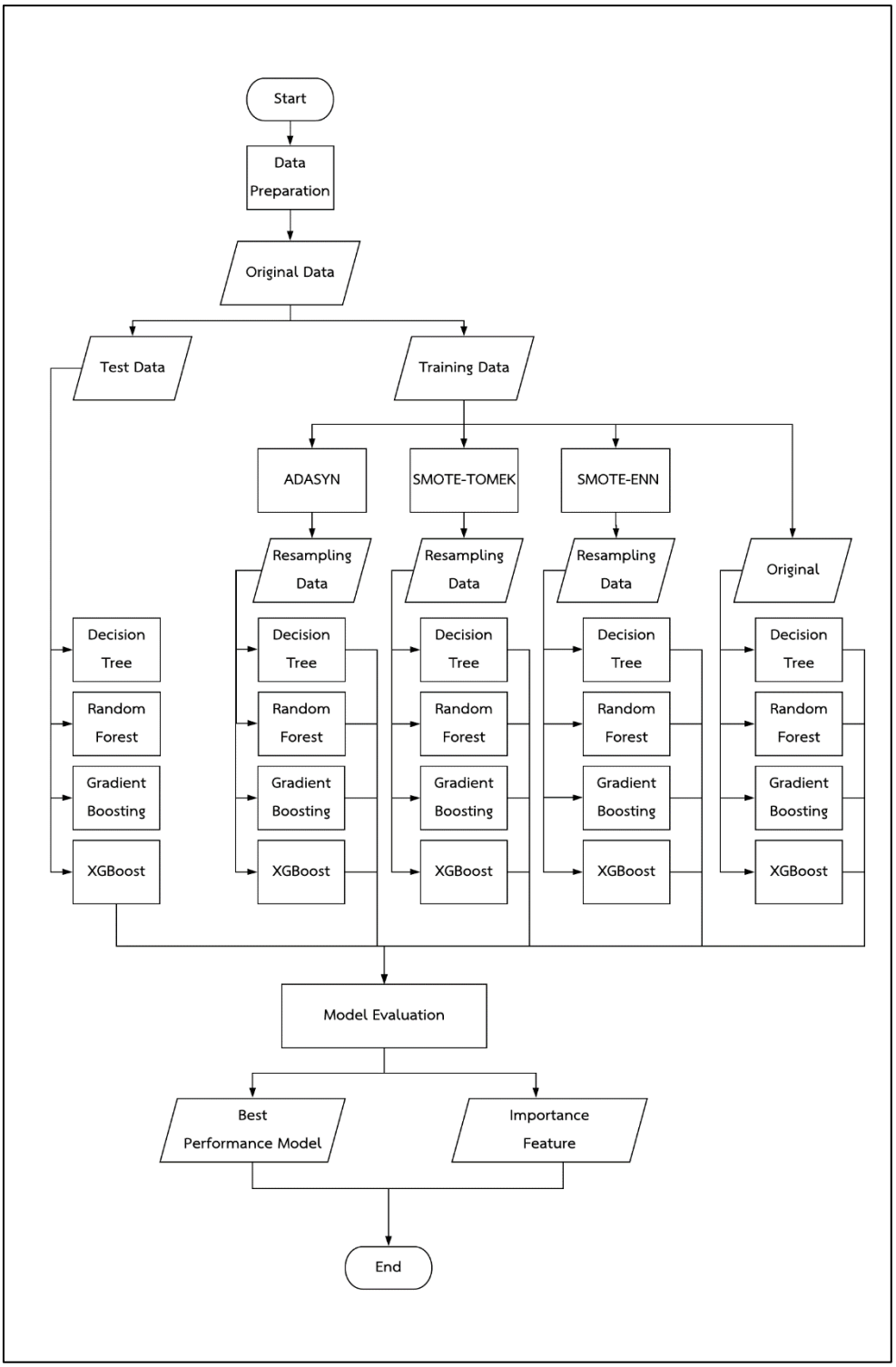



*Figure 8 Experiment processes*

In the data preparation process, we use the package in *scikit-learn* for cleansing data and eliminate irrelevant features. After we drop irrelevant features, we divide features into two categories numerical and categorical and transform categorical

features such as gender by encoding to discrete value. On the other hand, we apply feature scaling to features which have value as numerical to transform all numerical value features have the same scale. In this experiment, we use standardization for this process.

After the data preparation process, we get imbalance data which we call it that the original data. We split this data into two data sets test data and training data. We use the package from imbalanced-learn to resamples on training data, and with this package, we can use the various technique for resampling data include ADASYN, Synthetic Minority Over-sampling Technique (SMOTE), SMOTE-ENN, and SMOTE-Tomek. For this experiment, we use three-techniques: ADASYN, SMOTE-ENN, and SMOTE-Tomek to generate new data sets from training data which have more balance than the original data. These data are called resampling data. So, after this step, we have 4 data sets: training data from original data, training data from resampling with SMOTE-ENN, training data from resampling with SMOTE-Tomek and training data from resampling with ADASYN.

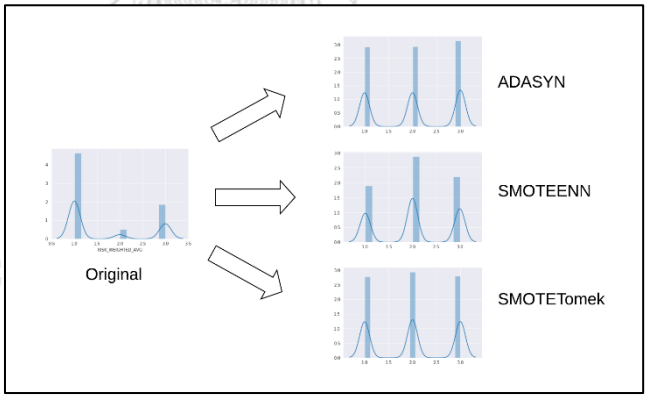



*Figure 9 Distribution and kernel density estimation of data before and after resampling.*

We use four data sets from the resampling process as the data source for training and testing a decision tree classifier and ensemble methods : a random forest classifier, Gradient Boosting, and XGBoost. Ensemble methods have a particular characteristic and it can provide specific features importance which will be used to identify business strategy such as product designing or determine marketing promotion.

This is the reason we use ensemble methods in place of neural network or deep learning which may give more accuracy than ensemble methods but difficult to specify features importance.

To find the best parameter for each model, we use a grid search process for hyperparameter tuning. We try to find parameters which make the model has the best F1 score. For evaluating models' performance, we use stratified 10-fold cross-validation which use stratified sampling to get a training set and test set in each iteration of evaluation. Stratified sampling is a method of sampling which divides data (population) into strata according to the characteristic of data, in this case, are risk attitude and make sample data by random choose data from each beginning data' strata to sample data with the ratio of each stratum. So data in training set and test set from this method have ratios of the number of data between class same as beginning data and ensure that the sample in the evaluation process is representative of the whole population also ensure that every class is used as a training set and test set in this evaluation.

Since our data is imbalanced, using accuracy only is not appropriate for the evaluation model’s performance. From the confusion matrix in figure X, true negative (TN) is the number of correctly prediction negative sample. In the other hand, true positive (TP) is the number of correctly prediction positive, false positive (FP) is the number of incorrectly prediction positive sample and false negative is the number of incorrectly prediction negative sample. In the case of imbalance data, accuracy is not an appropriate evaluation performance measure as a model which learn from imbalance data will classify that most of the data have class as majority class for example in case of a churn model we found that most of the customer is not churn about 99% of all customer and the customer is churn have only 1%. If the model learns from this data, this model may classify all customer is not churned that make accuracy as 99% but this model has a problem to classify minority class which in this case is the customer who will churn from firm's service. Two performance measures, precision and recall, take an important role for case imbalance data. The goal is to try to improve the model’s recall without impact on precision, but when trying to improve recall by increasing the true positive of minority class that may be a cause to increase

false positive which effect to decrease precision. So, we must trade-offs of precision and recall. F-score is a measure which combines the trade-offs of precision and recall. F-score is harmonic mean of precision and recall, so it is the number which shows the balance between precision and recall.

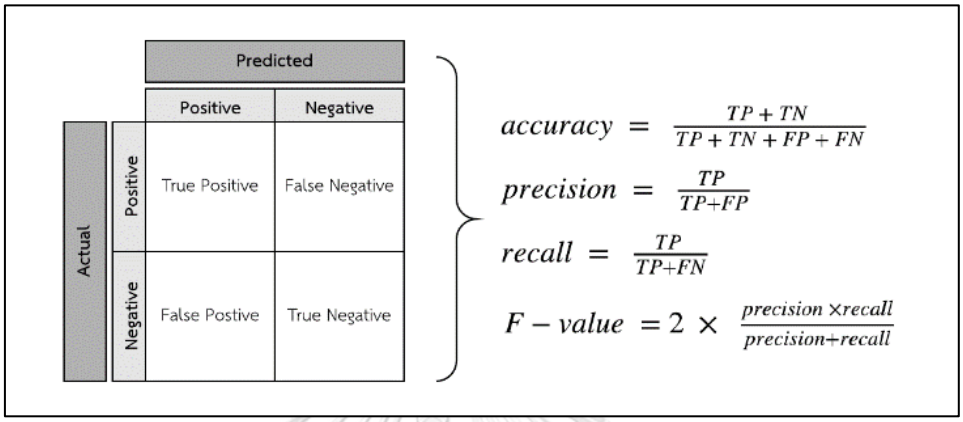



*Figure 10 Confusion matrix*

In addition to accuracy, precision, recall, and F-score, we would like to know the performance of the model to classify each class. So, we use the ROC curve as another measure of model's performance. ROC curve shows the relation of the probability of a true positive rate on Y-axis and a false positive rate of each class. The ideal point on the ROC curve is (0,1) which mean the model can classify correctly 100% and the line x = y represented classify of the model has a performance like randomly guessing.



$$\text{True positive rate} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

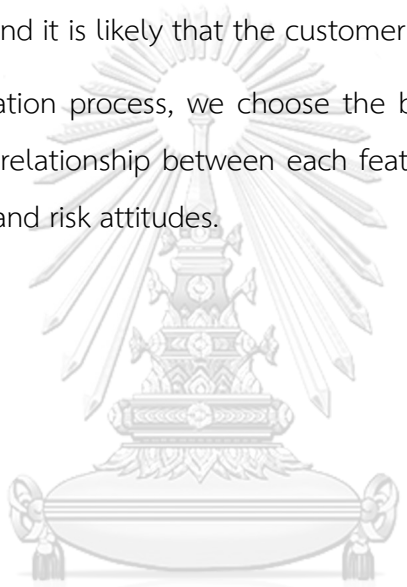
$$\text{False positive rate} = \frac{\text{FP}}{\text{TN} + \text{FP}}$$

ROC shows balance of true positive rate and false positive rate that give information to tradeoff between them to find the best true positive rate on an acceptable false positive rate. On itself is not a single number to define the performance of the model and sometimes it is difficult to compare between models. So area under curve (AUC) is the total area under the ROC curve which is a single

number to define and compare the model's performance. A larger AUC means more effective model.

We put attention to model's performance for classifying each class because there is a cost from type I errors and type II errors. Type I errors, or false positive, means that the model classifies that customers have risk attitude in this class but it is not true. In the other way, Type II errors, or false negative means that the model classifies that customers do not have a risk attitude in a specified class but actually they do. Both errors are the cause of opportunity lost from offering the inappropriate product to customers and it is likely that the customer will reject that offering.

After the evaluation process, we choose the best model , get its important feature and found the relationship between each feature and the impact of feature effect on classification and risk attitudes.



# Chapter IV

# Results

After using grid search to find the best parameters for each model we get the list of the parameters in the table 4. We implement models with these parameters to each data set from resampling techniques and get the results.

*Table 4 Hyper parameters of each classifier ("Tuning the hyper-parameters of an estimator ", 2018)*

| Classifier | Parameter |
|---|---|
| Decision Tree | class_weight=None,criterion='gini',max_depth=10, max_features=None, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None,min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, presort=False, random_state=None, splitter='best' |
| Random Forest | bootstrap=True, class_weight=None, criterion='entropy', max_depth=None, max_features=170, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, n_estimators=400, n_jobs=None, oob_score=False, random_state=None, verbose=0, warm_start=False |
| Gradient Boosting | criterion='friedman_mse', init=None, learning_rate=0.5, loss='deviance', max_depth=200, max_features=150, max_leaf_nodes=None, min_impurity_decrease=0.0, min_impurity_split=None, min_samples_leaf=1, min_samples_split=2, min_weight_fraction_leaf=0.0, n_estimators=150, n_iter_no_change=None, presort='auto', random_state=None, subsample=1.0, tol=0.0001, validation_fraction=0.1, verbose=0, warm_start=False |
| XGBoost | base_score=0.5, booster='gbtree', colsample_bylevel=1, colsample_bytree=0.7, eta=0.05, eval_metric='auc', gamma=0, learning_rate=0.1, max_delta_step=0, max_depth=15, min_child_weight=1, missing=nan, n_estimators=1000, n_jobs=1, nthread=5, objective='multi:softprob', random_state=0, reg_alpha=0, reg_lambda=0.1, scale_pos_weight=1, seed=None, silent=True, subsample=1 |

We use F-score and AUC to compare all models. F-score represents the value of precision and recall and AUC represents the capability to distinguish between classes. The result from experiment shows that XGBoost has the best F-score and AUC,

according to the table below. We found that the best performance with F-score as a measure is XGBoost when working with data from SMOTE-Tomek resampling techniques (F1 score: 54.19%) but in case of AUC, XGBoost gives the highest value when uses with data from ADASYN.

*Table 5 Experiment result.*

| | Decision tree | | | | |
|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F1 | AUC |
| Original | 55.42% | 49.32% | 47.64% | 47.85% | 60.41% |
| ADASYN | 53.02% | 47.82% | 49.11% | 48.22% | 60.74% |
| SMOTE-ENN | 40.35% | 45.58% | 50.17% | 40.20% | 61.62% |
| SMOTE-TOMEK | 53.07% | 48.10% | 49.25% | 48.52% | 61.39% |
| | Random Forest | | | | |
| | Accuracy | Precision | Recall | F1 | AUC |
| Original | 63.05% | 62.11% | 50.53% | 49.17% | 61.88% |
| ADASYN | 62.35% | 58.58% | 52.72% | 51.93% | 62.93% |
| SMOTE-ENN | 40.48% | 50.80% | 54.25% | 39.08% | 63.64% |
| SMOTE-TOMEK | 62.30% | 58.27% | 53.17% | 52.27% | 63.23% |
| | Gradient Boosting | | | | |
| | Accuracy | Precision | Recall | F1 | AUC |
| Original | 61.60% | 57.25% | 51.94% | 52.19% | 72.88% |
| ADASYN | 60.50% | 55.06% | 51.98% | 52.12% | 73.09% |
| SMOTE-ENN | 43.15% | 50.26% | 54.43% | 42.64% | 71.53% |
| SMOTE-TOMEK | 61.15% | 55.56% | 52.99% | 52.83% | 72.33% |
| | XGBoost | | | | |
| | Accuracy | Precision | Recall | F1 | AUC |
| Original | 64.32% | 61.53% | 54.10% | 54.16% | 76.71% |
| ADASYN | 63.70% | 60.11% | 53.58% | 53.53% | 76.97% |
| SMOTE-ENN | 47.17% | 52.11% | 57.02% | 47.02% | 74.64% |
| SMOTE-TOMEK | 64.03% | 60.38% | 54.33% | 54.19% | 76.95% |

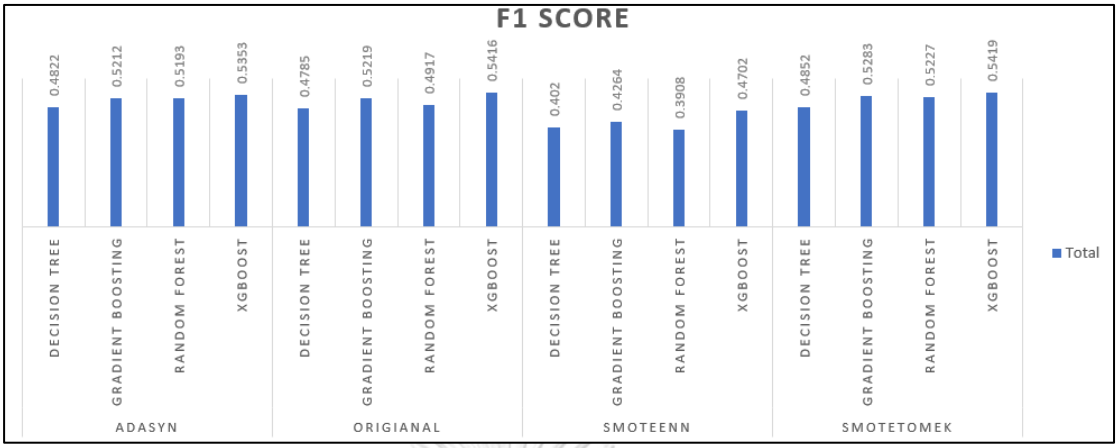



*Figure 11 F1 score.*

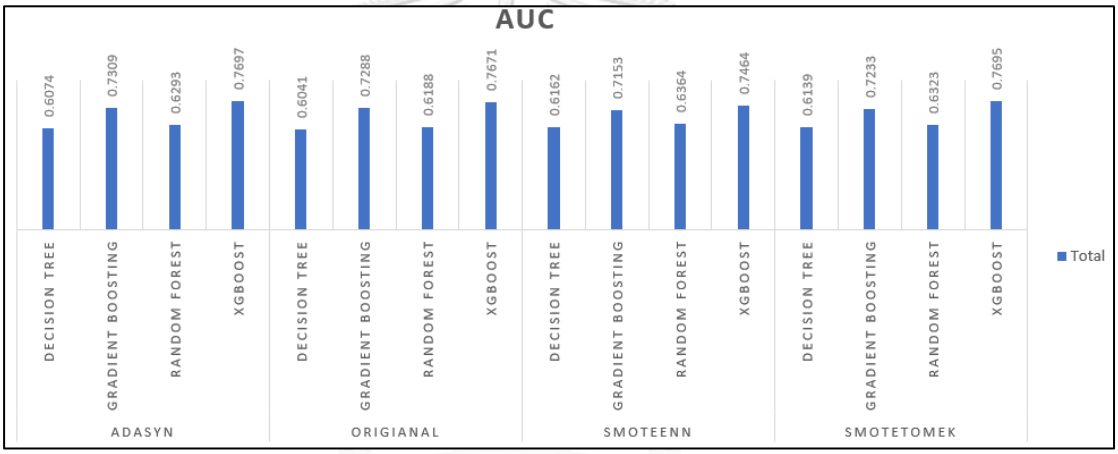



*Figure 12 AUC score.*



When using F-score to evaluate classifier output quality, we use ROC which is a curve of probability occurring true positive rate (Y-axis) and false positive rate (X-axis). From ROC we use AUC to shows the performance of the model to classify each label. According to figures of ROC of XGBoost that use with various data from different resampling technique. We found that all XGBoost models can classify class 3 (risk seeker) most correctly when it is used with data from ADASYN. The result has AUC 0.88. These means that both models have a 88% chance to distinguish between class 3 (positive class) and not class 3 (negative class). The model has the same highest value of AUC in class 1 (risk-averse) and class 2 (risk-neutral) is 0.72 when use data from ADASYN. From this result, we can conclude that XGBoost which is trained with

data from ADASYN have better performance less than using data from SMOTEEN and SMOTE-Tomek in classifying every classes.

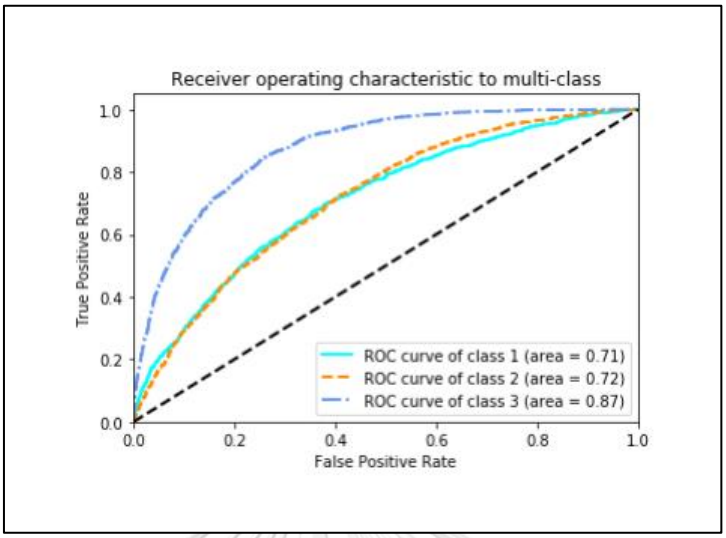



*Figure 13 ROC of XGBoost with data from original data.*

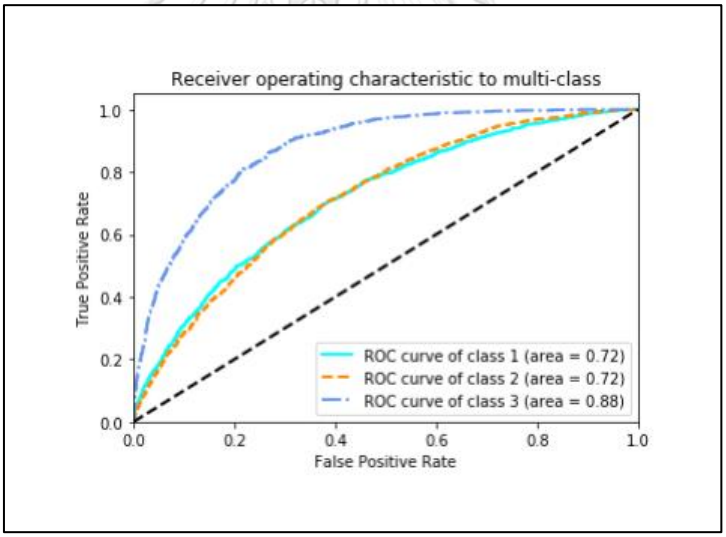



*Figure 14 ROC of XGBoost with data from ADASYN.*

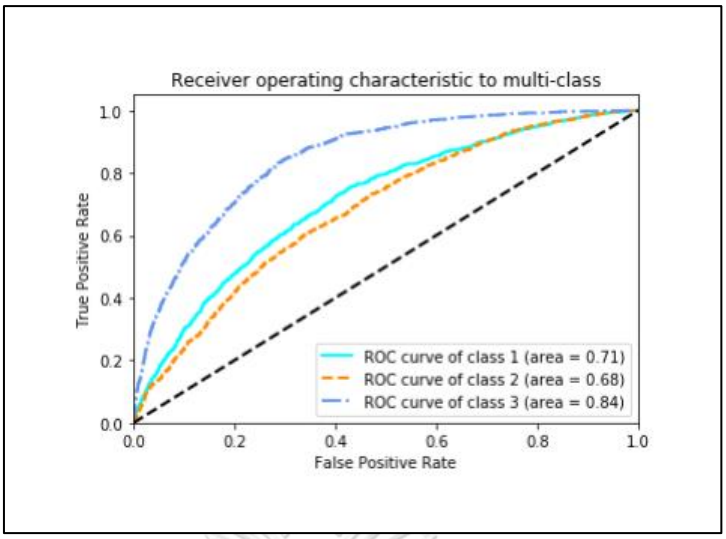



*Figure 15 ROC of XGBoost with data from SMOTE-ENN.*

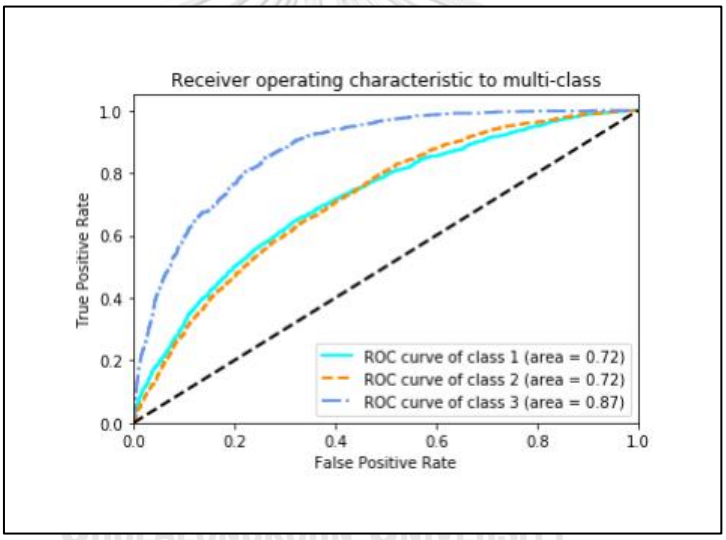



*Figure 16 ROC of XGBoost with data from SMOTE-TOMEK.*

From AUC value in the figures above, we observe that the model which uses data from ADASYN resampling technique can distinguish class 1 (risk aversion), 2 (risk neutral) better than the model which uses data from SMOTE-ENN and equal to the model which uses data from SMOTE-TOMEK but for class 3 (risk seeker) it is a little bit better than the model which uses data from SMOTE-TOMEK and SMOTE-ENN. So, we can conclude that in case the firm wants to classify customer's risk attitudes with imbalance data ADASYN is the better choice.

Next, we use the best model, XGBoost with the data from ADASYN technique to find essential features. From the Pareto principle or 80/20 rule, we prove that only 20 percents of features affect 80 percents of classification accuracy. So we choose the best 95 features or 20 percents of all features in each model and run the experiments and get the results in Table 6.

*Table 6 Result of experiment which use twenty percent number of features.*

| | XGBoost | | | | |
|---|---|---|---|---|---|
| | Accuracy | Precision | Recall | F1 | AUC |
| ADASYN | 56.65% | 50.19% | 48.43% | 48.21% | 69.28% |

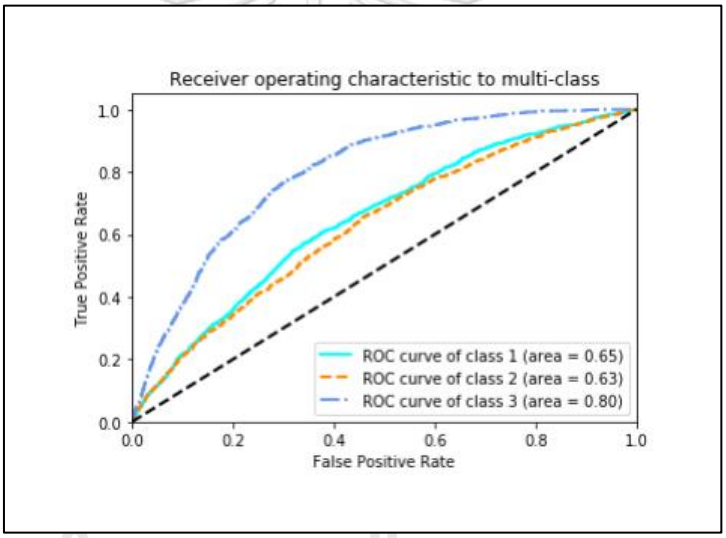





*Figure 17 ROC of XGBoost with data from ADASYN (use twenty percent number of feature).*

From the result above we can classify risk attitudes of the customer in this data set with using only 20% of all features or importance features (95 features). In the other words, we can infer that these features are important factors which affect the risk attitudes of the customer. The firm should use these factors for planning marketing strategy and designing the financial product to serve their needs.

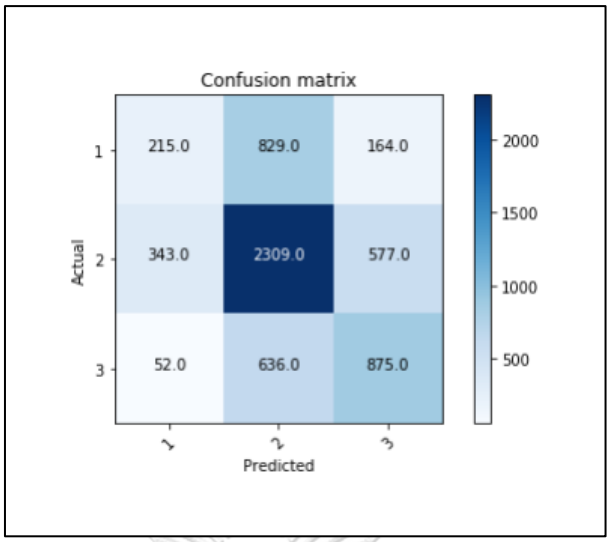



*Figure 18 Confusion matrix from the model.*

We plot the learning curve from the model and the result is shown in the figure below. We found that when using a small training set, we get high training score or low bias, but we get a low cross-validate score or high variance. This situation told us that the model is overfitting. Then, we increase the size of the training set and we found that bias is stable but the variance is decreasing. This proves that adding more the training set eliminate the overfitting problem.

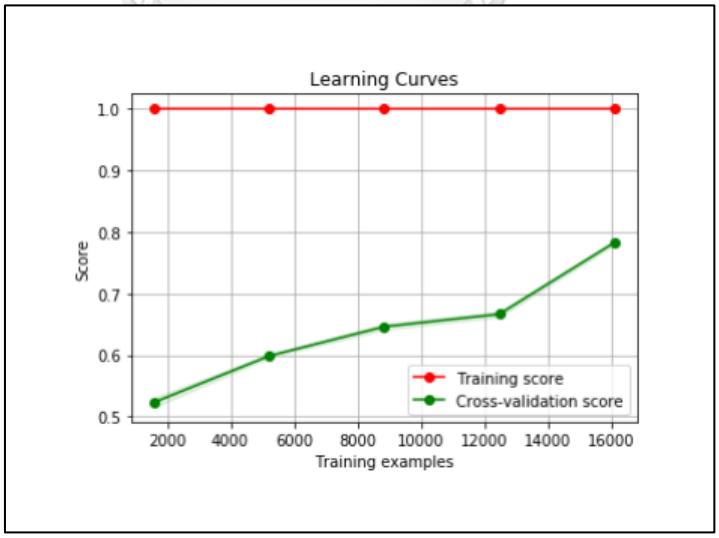



*Figure 19 Learning curve of the model.*

Another result from XGBoost is the important features. We select top ten important features and plot the graph in the figure below.

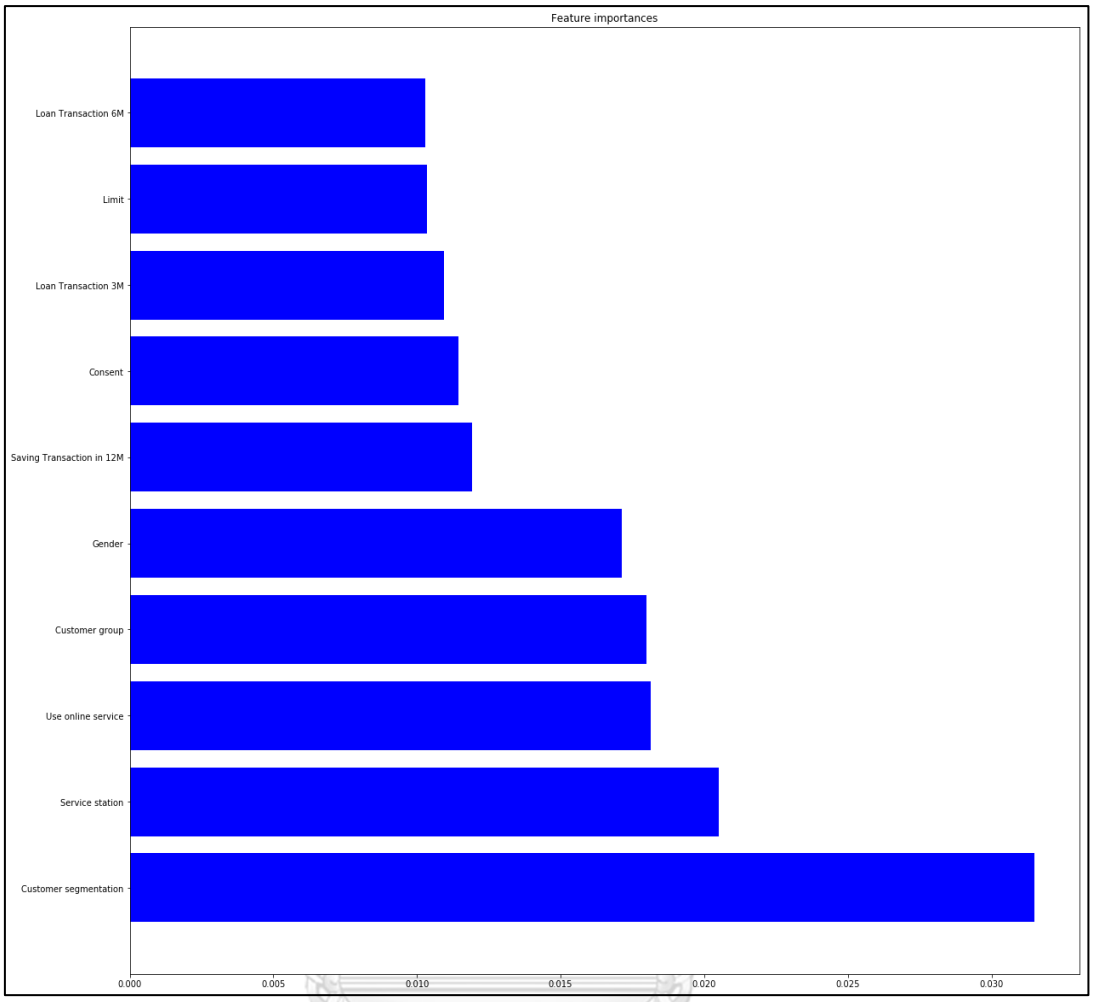



*Figure 20 Top ten important features.*



Now we know what features have an impact on consumers' risk attitudes but we do not know about the strength and direction of the relationship between these features with consumers' risk attitudes. So, we use a Pearson correlation to find the strength and direction of the relationship between each feature to consumers' risk.

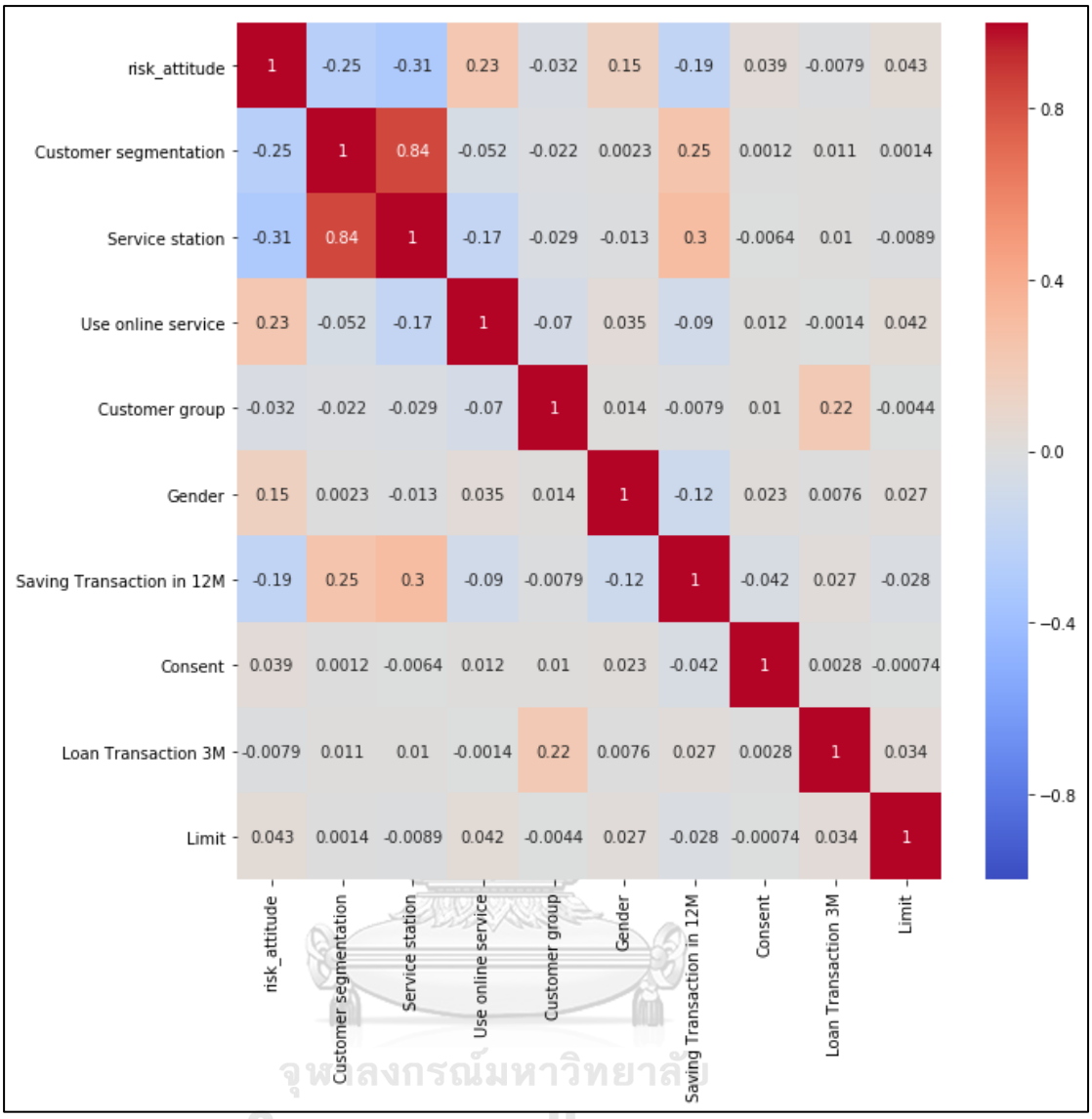





*Figure 21 Correlation matrix.*

# Chapter V

# Conclusion

In this chapter, we mention the research result, the problem and limitation of this research, and suggestions for the future work on the classification of risk attitudes from customer behavior with machine learning.

## 5.1 Research result

This research presents a machine learning model that specifies consumers' risk attitude from their behavior. The model is analysed to find essential features which impact consumers' risk attitude. Knowing the important features is the key to the firms to design their products and services to provide the value appropriates to their customers' attitude.

In this research, we faced unbalance data, and to solve this problem, we use three resampling techniques: ADASYN, SMOTE-Tomek, and SMOTE-ENN. All of them are the hybrid techniques. They use oversampling to generate synthetic instances and then use undersampling to eliminate some instance depend on each algorithm. We use data from resampling and original data as the input of four machine learning methods: Decision Tree, Random Forest, Gradient Boosting, and XGBoost. We use cross-validation to measure their performance and found that using XGBoost gives the best in accuracy, recall, F1, and AUC. We choose XGBoost which use data from ADASYN because from RUC was prove that it is the best approach to classify class 1,2, and 3. So, we choose this method in the next step. In the next step, we choose twenty percents of the number features with the data generated from ADASYN and XGBoost. The result is close to the result from using all features. We conclude that it not necessary to use all features for the model to classify risk attitude. Then we measure the value of importance on these features and the result shows that features about the monetary transaction, customer segmentation, and location of the service station where customer use the service are the top three features which affect the classification of consumers' risk attitude. To specify the direction and strength of the association between features and risk attitude, we use the Pearson correlation

coefficient to measure them. The result exposes that the feature about the location of the service station where customer uses the service has the strongest association with risk attitude in the negative direction and the customer segmentation feature comes in the second. It has a strong association with risk attitude in the negative direction.

For business, we can interpret the result from the model that the risk attitude of consumer depends on location which they come and use the service. The firm must find and analyze what consumers' attitude in each location of the service stations and might find out that some location of the service stations has consumers who have risk-averse or risk seeker more than other location. So the firm can design a marketing plan to promote its product to appropriate to consumers in various location. Another feature which has strong associative with risk attitude is customer segmentation. The result indicates that customer segmentation is important and the firm should design the product for each segmentation with appropriate risk attitude. This information is important for the firms to offer products or promotions which suitable to the customer individually.

## 5.2 Problem and limitation

1. We do not have the data which cover all consumer's behaviors.
2. Some data are missing and incorrect. We use many processes to clean them before they are used in the experiment.
3. Lack of computation power, especially when using grid search for tuning hyper parameters limits the quality of the results.

## 5.3 Suggestion

This research can development more in any aspect as follow.

1. Applying statistical methods to study important features might find more information about data and features.
2. Develop or analyze a resampling technique which most appropriate to the imbalance data.

# REFERENCES


Agresti, F., Klingenberg. (2018). *Statistics. The art and science of learning from data* (4 ed.). Essex: Pearson Education, Inc.

Bing Zhu, B. B., Aimee Backiel, & Seppe K.L.M. vanden Broucke. (2016). Benchmarking sampling techniques for imbalance learning in churn prediction. . *Journal of the operational research society, 69*(1), 17.

Ensemble methods (2018). Retrieved from https://scikit-learn.org/stable/modules/ ensemble.html

Geron, A. (2017). *Hands-On Machine Learning with Scikit-Learn and TensorFlow* (12 ed.). Sebastopol, CA: O'Reilly Media,Inc.

Hussain., Q.-F. W. M. X. A. (2018). Large-scale Ensemble Model for Customer Churn Prediction in Search Ads. *Cognitive Computation, 11*, 9.

Ian H. Witten, F. E., Mark A. Hall (2011). *Data Mining Practicle Machine Learning Tools and Technique.* (3 ed.). Burlington, MA: Morgan Kaufmann Publishers.

Lind, D. A., I Marchal, William G.,I Wathen. (2018). *Statistical Techniques in Business and Economics* (9 ed.). NY: McGraw-Hill Higher Education.

NG, A. (2019). *Machine Learning Yearning: Technical strategy for AI engineer, in the era of deep learning*: Amazon Digital Services LLC.

Philip Kotler, K. L. K. (2012). *A framework for marketing management* (5 ed.). Essex: Pearson Education.

Tuning the hyper-parameters of an estimator (2018).

# VITA

| | |
|---|---|
| **NAME** | Teeranai Sriparkdee |
| **DATE OF BIRTH** | 18 October 1985 |
| **PLACE OF BIRTH** | Bangkok |
| **INSTITUTIONS ATTENDED** | Bachelor degree in science (computer science) at Kasetsart University |
| **HOME ADDRESS** | 73/8 Charoenkrung road Charoenkrung 39 Sri praya Bangrak 10500 |