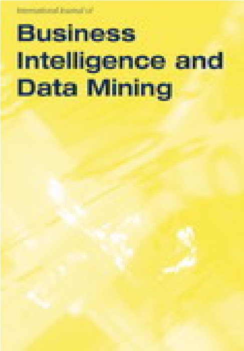


International Journal of Business Intelligence and Data Mining

ISSN online: 1743-8195 - ISSN print: 1743-8187
https://www.inderscience.com/ijbidm

Machine learning models for predicting customer churn: a case study in a software-as-a-service inventory management company

Naragain Phumchusri, Phongsatorn Amornvetchayakul

DOI: 10.1504/IJBIDM.2024.10051203

Article History:

Received:	27 April 2021
Accepted:	30 August 2022
Published online:	01 December 2023

Machine learning models for predicting customer churn: a case study in a software-as-a-service inventory management company

Naragain Phumchusri* and
Phongsatorn Amornvetchayakul

Department of Industrial Engineering,
Faculty of Engineering,
Chulalongkorn University,
Bangkok, Thailand
Email: Naragain.P@chula.ac.th
Email: phongsatorn.amornvetchayakul@sasin.edu
*Corresponding author

Abstract: Software-as-a-service (SaaS) is a software-licensing model, which allows access to software on a subscription basis using external servers. This article proposes customer churn prediction models for a SaaS inventory management company in Thailand. The main focus of this work is seeking the most suitable customer churn prediction model for this case-study SaaS inventory management company which is currently having a high churn rate issue. This paper explores four machine learning algorithms, which are logistic regression, support vector machine, decision tree (DT) and random forest. The results show that the optimised DT model is capable of outperforming other classification models toward recall scorer with validated testing scores of 94.4% of recall and 88.2% of F1-score. Moreover, feature importance scores are investigated for practical insights to identify features that are significantly related to churn behaviour. Therefore, the findings can help the case-study company indicate customers who are going to churn more precisely and enhance the effectiveness of managerial decisions and effective marketing movement.

Keywords: churn prediction; machine learning; software-as-a-service; SaaS.

Reference to this paper should be made as follows: Phumchusri, N. and Amornvetchayakul, P. (2024) 'Machine learning models for predicting customer churn: a case study in a software-as-a-service inventory management company', *Int. J. Business Intelligence and Data Mining*, Vol. 24, No. 1, pp.74–106.

Biographical notes: Naragain Phumchusri is an Associate Professor at the Department of Industrial Engineering, Chulalongkorn University, Thailand. She received her PhD in Industrial Engineering from the Georgia Institute of Technology, Atlanta, GA, USA in 2010.

Phongsatorn Amornvetchayakul received her Bachelor's in Mechanical Engineering and Master's in Industrial Engineering, Chulalongkorn University, Thailand in 2020.

1 Introduction

Software-as-a-service (SaaS) is a software licensing model, which allows access to software a subscription basis using external servers. According to Gartner Forecasts Worldwide Public Cloud Revenue to Grow 17% in 2020 (2020), SaaS is the biggest market segment and is projected to stay in this 1st rank for many years to come. As can be seen in Table 1, it is predicted that by 2022, this SaaS sector will reach $150 billion (public cloud service revenue). SaaS's leading benefit is that users do not need to install or update any software. SaaS revenue model is associated with regular, ongoing payments over a defined time period, in exchange for the use of a software application or other tool. Thus, for this business, customer retention is very important. One of the key performance indicators (KPIs) of the case-study company is percentage of customer churn.

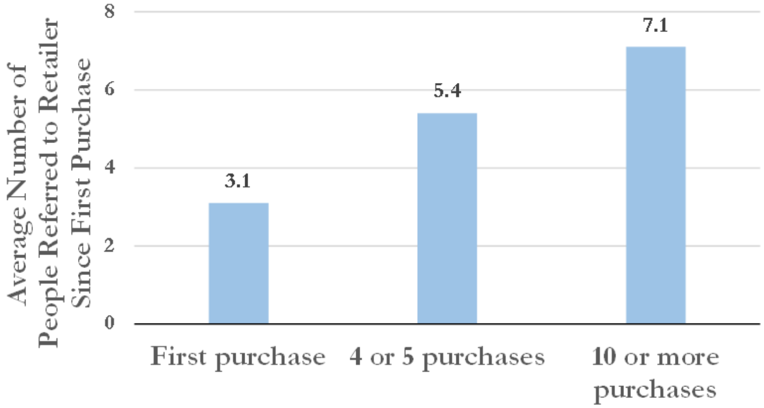
Table 1 Worldwide public cloud service revenue forecast (billions of U.S. Dollars)

Types of cloud service	*2018*	*2019*	*2020*	*2021*	*2022*
Cloud business process as a services (BPaaS)	41.7	43.7	46.9	50.2	53.8
Cloud application infrastructure services (PaaS)	26.4	32.2	39.7	48.3	58.0
Cloud application services (SaaS)	85.7	99.5	116	133	151.1
Cloud management and security services	10.5	12.0	13.8	15.7	17.6
Cloud system infrastructure services (IaaS)	32.4	40.3	50.0	61.3	74.1
Total market	196.7	227.8	266.4	308.5	354.6

Inventory management software is software designed to track and manage items through various stages along the supply chain. It grants visibility into the entirety of your stock, helping company maintain optimal amounts to continue fulfilling orders without holding too much of a given item. It is one of important tools enhancing the business competitiveness in terms of scalability, versatility, and advanced security in affordable prices (MarketWatch, 2019). With increasing market, this industry become more and more competitive, to compete in the market, the inventory management software company must capture the trends and keep scaling up. Nevertheless, attaining a new customer is knowingly costlier than retaining an existing one (Gallo, 2014). Moreover, Bain & Company's study found that the number of people referred to retailer since first purchase increases as the number of purchases (refer to Figure 1) (Reichheld, 2001). Considering that, the repeat customers also potentially provide a massive marketing advertisement as a part of loyalty which is another benefit of repeat customers. With the current economic situation moving towards a highly disruptive and rapidly saturating nature, low customer churn is now becoming a crucial metric towards success. The prediction of customer churn plays a significant role in evaluating not only customers, but also business and industries.

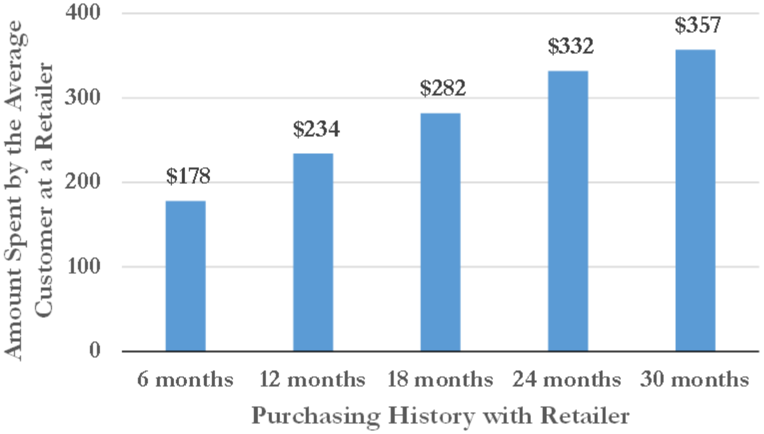
The accurate prediction of customer churn is one of the most valuable information businesses are currently seeking. The ability to identify customers who are high-risk or very likely to terminate their relationship with a service or discontinue their use of a product is a key in strategic decision making (Bain & Company, Inc., 2000). Consequently, customer churn rate, the % of customers who have cancelled their subscription or use of a product or service, should be focused for SaaS businesses. According to a research in retail sector by Bain & Company (Reichheld, 2001), the repeat customers further spend 23 % more per orders after purchasing with the same company for 30 months (refer to Figure 2.) that the power of repeat customer significantly impacts on the spending growth.

Figure 1 Referral impact (see online version for colours)



The case-study company is an inventory software on cloud using SaaS business model. The company offers an integrated inventory management platform and inventory software on cloud tools for SMEs in many types of packages provided the various different options and functional services in the different price tiers. Since the company is a funded startup in which its organisation is still under 100 employees, this company needs to quickly capture the market share by acquiring new customers and create the reliability for its customers to keep the company stays in a positive customer retention rate or lower customer churn rate.

Figure 2 Spending growth impact (see online version for colours)



Currently, the company is facing a high customer churn rate with over 50 % according to the provided raw data. Under this circumstance, the case-study company needed to focus on customer relationship management through customer churn rate prediction. Besides the company was also willing to find a model to solve the issue as soon as possible. Since the number of customers has been increased significantly as a result of that the company has already spend a large amount of investment in marketing for a new customer acquisition. On the contrary, the percentage of customer churn has not reduced causing the problem in the case-study company's cash flow owning to the unpredictable revenue stream. Accordingly, customer churn prediction become an urgent issue for the company to obtain the first step of resolution as the company aims to reduce the high customer churn rate. Therefore, this issue leads to the objective of this paper that is to search for suitable customer churn prediction model for the case-study in Thailand.

2 Literature review

Customer churn prediction is one of analytic topics considered in many businesses. Telecommunication industry is one of the pioneers in this topic (Hung and Wang, 2004; Guo-en and Wei-dong, 2008; Umayaparvathi and Iyakutti, 2012; Lu et al., 2014; Wanchai, 2017; Amin et al., 2019) (shown in Table 2). It applies machine learning models to predict churn on an individual customer basis and take counter measures such as discounts, special offers or other gratifications to keep their customers. Banking is another industry well known for churn prediction research (Glady et al., 2009; Ali and Arıtürk, 2014; Hemalatha and Amalanathan, 2019) (shown in Table 3) where prediction of churning customers predicts which customer is near to leave the services of the specific bank. It is very useful for big organisations who are very conscious about retaining their customers.

Customer churning is also generally studied in retailing industry (Buckinx and Van den Poel, 2005; Yu et al., 2011; Tamaddoni Jahromi et al., 2014; Vadakattu et al., 2015; Chen, 2016; Berger and Kompan, 2019) (refer to Table 4). For example, a retail may be interested in customers who decide not to use the company's services anymore. This can be specified based on a goal of the churn analysis, for instance one may label customers as 'churners' when they do not make any new purchases from the same shop for six months. For example, Buckinx and Van den Poel (2005) studied churn prediction in fast-moving consumer goods (FMCG) retails, while many recent papers concentrated in online retailing due to expanding market shares (Yu et al., 2011; Vadakattu et al., 2015; Berger and Kompan, 2019). With limited data access, fewer research about churn prediction were studied for SaaS industry (Frank and Pittges, 2009; Ge et al., 2017; Rautio, 2019) (shown in Table 5). The majority of papers in this business analysed open data sources which has limitation of recent updated data.

Based on these papers' conclusions as shown in Table 2 to 5, it can be found that the studies have been conducted in the different angles by each reference, i.e., methodology, type of industries, attributes and customer churn definition. However, the difference in the interesting industry can significantly affect the model outcomes. Therefore, the studies related to SaaS, as described in Table 5, become an important reference to consider. In early-stage of SaaS development, Frank and Pittges (2009) considered churn prediction models using various types of machine learning, e.g., K-means and DT model originally done in telecommunication industry to SaaS. Then Ge et al. (2017) explored variety of model performance, i.e., logistic regression, random forest and ensemble learner of XGBoost for churn prediction where churning is identified as customer suspending payment for at least 2 months. It was found that XGBoost outperform other models. For a recent paper, Rautio (2019) focused on the problem of customer churn prediction in SaaS with different machine learning methods. This paper found that support vector machine (SVM) provided the most accurate results.

Table 2 The studies on customer churn prediction in telecommunication industry

Reference	*Industry*	*Methodology*	*Churn definition*
Hung and Wang (2004)	Telecommunication (Wireless)	Decision tree (C5.0), and artificial neural networks (BPN)	Subscriber switches to a competitor over a period of time
Guo-en and Wei-dong (2008)	Telecommunication	Support vector machine (linear), artificial neural networks, decision tree (C4.5), logistic regression, and Naive Bayes	Customer does not enjoy all services of
Umayaparvathi and Iyakutti (2012)	Telecommunication	Decision tree, and artificial neural networks	Customer shifts to other service provider
Lu et al. (2014)	Telecommunication (Mobile)	Logistic regression, and ensemble learner (boosting)	Subscriber switches to another service during a period of time
Wanchai, P. (2017)	Telecommunication (Mobile)	Decision tree (C4.5), logistic regression, and artificial neural networks	Consumer switch or choose to cancel their subscription
Amin et al. (2018)	Telecommunication	Naive Bayes	Customer leave the service or company

Table 3 The studies on customer churn prediction in banking industry

Reference	*Industry*	*Methodology*	*Churn definition*
Glady et al. (2009)	Banking	Decision tree (cost-sensitive), artificial neural networks (MLP), logistic regression, and ensemble learner (AdaCost)	Customer lifetime value (CLV) decreases over a period of time
Ali and Arıtürk (2014)	Banking	Independently trained binary, multinomial and ordinal logistic regression, and survival analysis	Customer's portfolio size is below a specific threshold value during a period of time
Hemalatha and Amalanathan (2019)	Banking	Support vector machine, KSVM and artificial neural networks	Not mentioned

The literature shows that various machine learning techniques have been executed for customer churn prediction in many different industries with unique churn definition in each research. As a result of that different definition, condition or involved industry as aforementioned, the conflicts between the conclusions of these published papers arise. Therefore, this paper focuses on prediction of customer churn for a SaaS inventory management company based on case study in Thailand. We define churn according to the case-study company marketers' requirement as a customer who have been inactive consecutively for more than 14 days which is different from previous papers.

Table 4 The studies on customer churn prediction in retailing industry

Reference	*Industry*	*Methodology*	*Churn definition*
Buckinx and Van den Poel (2005)	Retailing (FMCG)	Artificial neural networks (ANN), logistic regression, and ensemble learner (random forest)	Customer changes purchasing behaviour during a particular period of time
Yu et al. (2011)	Retailing (Online)	Artificial neural networks (BP), decision tree (C4.5), and support vector machine (linear polynomial, RBF)	Customer shifts to a competitor
Tamaddoni Jahromi et al. (2014)	Retailing (FMCG)	Decision tree (simple, cost-sensitive, CART), logistic regression, and ensemble learner (boosting)	Customer has no purchase in prediction period
Vadakattu et al. (2015)	Retailing (Online)	Decision tree (C4.5) and SMOTE	Customer deactivate in next 3 months
Chen (2016)	Retailing	gamma cumulative sum (CUSUM) chart	Customer not login during a particular period of time
Berger and Kompan (2019)	Retailing (Online)	Support vector machine	User considers this session is the last session

Table 5 The studies on customer churn prediction in SaaS industries

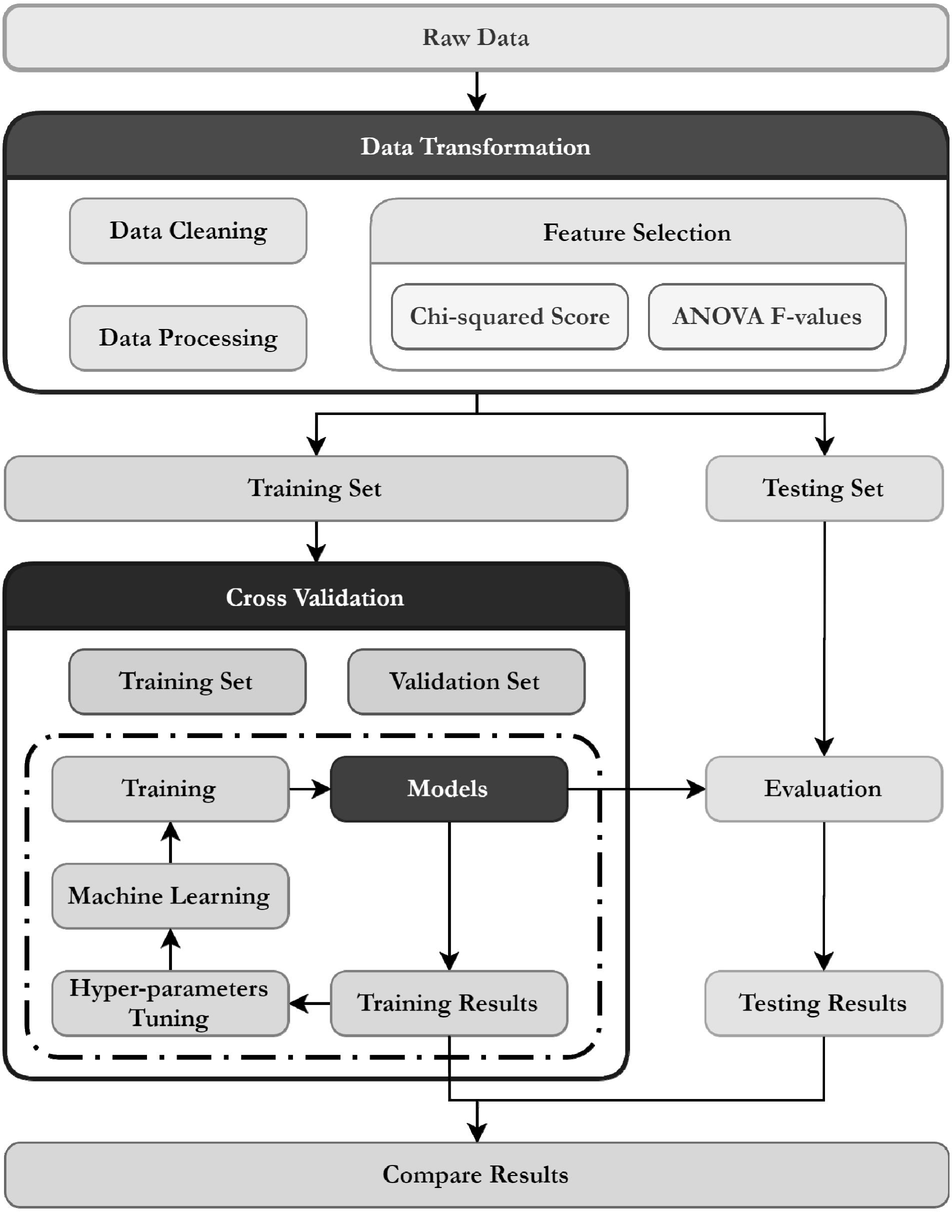
Reference	*Industry*	*Methodology*	*Churn definition*
Frank and Pittges (2009)	Software-as-a-service	K-Means and decision tree	Not mentioned
Ge et al. (2017)	Software-as-a-service	Logistic regression, and ensemble learner (random forest and XGboost)	Customer suspended payment for at least 2 months
Rautio (2019)	Software-as-a-service	Convolutional neural networks (CNN), recurrent neural networks (RNN), support vector machine, and RF	Customer quits using a company's services completely
This paper	Software-as-a-service	Logistic regression, support vector machine, decision tree and random forest	Customer has not been active for more than 14 days

3 Methodology

The machine learning process of logistic regression, support vector machine, decision tree, and random forest is presented in Figure 3.

This process begins with collecting raw data that all features are derived from customer usage and business metric regarding the insight of case-study company marketers.

Figure 3 Machine learning (i.e., logistic regression, support vector machine, decision tree, and random forest) overall process



Then that data is transformed in order to prepare for execution with machine learning algorithms. Data transformation is including with three processes:

1 data cleaning

2 data processing

3 feature selection.

After the data is already prepared by these transformation process, the data is split into training dataset and testing dataset by holdout method. For training dataset, this set is used for training model in each machine learning techniques with hyper-parameters tuning and validating the model with K-fold cross validation in this paper. Moreover, evaluating the model is executed on the split testing dataset in order to confirm the final testing results of each machine learning.

Eventually, these final testing results of every machine learning models (logistic regression, support vector machine, decision tree, and random forest) are compared by evaluation metrics to receive the best-performed model.

3.1 Data collection

This paper is motivated by the case-study company's goal in improving the customer churn issues. Regarding In early stage of data exploration, thanks to the company permission, researcher was granted to use their informative expectation and insights and intelligent raw data. researcher had to consider the possibility of data collection in each dimension. According to experience and competency, business insights provided by the company marketers are derived into several data matric mostly related to customer usage and business metric. Since the company concerns the privacy of their customer, some specific data that can be used to trace back or indicate the customer identification are not allowed to access. And the data are also sorted randomly for the same reason. Therefore, the data collection in the first step must have been checked and confirmed by the company authorities. Nevertheless, the provided data are enough to generate and execute the models.

Data were extracted from the case-study company database system based on records from October 2015 to October 2019. The total raw data consists of 1,788 observations. Regarding data extraction, the related features or attributes are listed in Table 6 which are also described the detail in each attribute. the features contribute of 23 variables in terms of customer usage behaviour and business matric such as transactions. This research defines churn as a customer who have not been active consecutively for more than 14 days while period of churn is based on October 1 to October 14, 2019.channels of the hotel, including walk-in, direct phone call, online booking, and seminar events. Although the seminar events accounted for 63.0% of room occupancies, this information was usually known in advance for more than one month. We therefore excluded the data from seminar events before constructing the forecast model. After retrieving the forecasts, we subsequently combined the forecasts with the known demand of seminar events as shown in the Figure 3.

Table 6 List of attributes

Attribute name	*Description*
daysToAct	Days from registration to first action
daysToExpire	Days until an expected expiration date.
totalTrans	Number of all transactions
currPeriodTrans	Number of transactions in current period of 14 days
prevPeriodTrans	Number of transactions in previous period of 14 days
amountSpend	Amount of customer spending
numAct	Number of actions per customer
Act_Day	Average number of actions per day
numCargo	Number of cargoes
numUser	Number of users per customer
UserAct	Average number of actions per user
lastContactDays	Days since the last contact with salesperson
everContact	Represent if salesperson ever contacted to a customer (ever contact, never contact)
numContact	Number of contacts
amountTrans	Amount of transaction values via platform
hasPhone	Represent if contact is provided (has contact, has no contact)
amountSpend_log	Amount of customer spending per login
numAct_log	Average number of actions per login
numUser_log	Average number of users per login
amountTrans_log	Average amount of transaction values via platform per login
usernumAct_log	Average number of user actions per login
Act_Day_log	Average number of actions per day per login
numCargo_log	Average number of cargoes per login
churn	Represent if a customer has not been active for more than 14 days (churn, not churn)

3.2 Data transformation

3.2.1 Data cleaning

This paper uses the listwise deletion technique to handle with missing data (Allison, 2002). This technique was recommended by literature and was also applied in customer churn prediction research, e.g., (Wanchai, 2017; Khodabandehlou and Zivari Rahman, 2017). In specific dataset, listwise deletion decreases the missing data from 1,788 observations of the whole raw data to 1,718 observations. The data include both churn and non-churn customer records, i.e., 927 churn samples and 791 non-churn samples. The original dataset is quite balanced, having 53.96% of churn samples and 46.04% of non-churn samples.

3.2.2 Data processing

There are two different types of variables: quantitative variables and categorical variables. With the nature of categorical data provided by the company, this research applies one hot encoding method to convert the categorical data. Additionally, dataset standardisation is then applied. Concerning the implementation benefits, function scale in standardisation neglects the distribution shape by scaling to the feature's standard deviation after removing the mean of a certain feature.

3.2.3 Feature selection

Two types of univariate feature selection or statistical filter method for classification (Chi-squared score and ANOVA F-values) are considered in this research. Both methods statistically measure each features' importance separately and rank these features variables according to their importance orders. After that, the top rank of the highest-score features is designated for machine learning input in the next step whereas the number of designated features is a parameter required to be assigned. As a consequence, this research implements all possible numbers of the selected features.

3.3 Machine learning

The machine learning methods for this research are selected based on the functionality, reliability and performance of the algorithms effectively performed in classification technique in SaaS literature. Algorithms are functioned using Scikit-learn in Python (Pedregosa et al., 2011). Moreover, in this paper, grid search technique is used to hyper-parameters tuning for each machine learning classification methods instead of running with only default parameters. The following machine learning techniques is essentially generated by various values or type of hyper-parameters. These models with different constraints or values of hyper-parameters provides unique results. Therefore, choosing relevant hyper-parameters for each model are necessary (Rautio, 2019). And hyper-parameters tuning is important to optimally solve the problem by each machine learning classification methods.

3.3.1 Logistic regression

Logistic regression is a process of modelling the probability of a discrete outcome given an input variable. The most common logistic regression models a binary outcome; something that can take two values such as true/false, yes/no, etc. When applying to churn prediction problem, dependent variable has two possible outcomes: churn and non-churn. This machine learning model was found to perform well in customer churn problem for SaaS business (Ge et al., 2017).

Considering hyper-parameters tuning for logistic regression classifier, the parameters which this paper selected are essential parts to control the model's complexity or detail in application of different techniques in some steps of classification. In this case, the hyper-parameters for logistic regression in which this paper is interested consists of three parameters:

1 penalty

2 C

3 tol. (see Table 7)

Table 7 Hyper-parameters list for logistic regression

Hyper-parameter	*Definition*
penalty	The types of norm used as penalty term
	L1 is defined as squared magnitude of coefficient in penalisation on the loss function in regression model
	L2 is defined as absolute value of magnitude of coefficient in penalisation on the loss function in regression model
C	The values of model simplicity or iregularisation strength
	The higher value of C indicates simpler model. The lower value of C creates more complex model or stronger regularisation
tol	The values of tolerance criteria
	The model stops searching toward the objective or optimisation when the indicated tolerance is reached

3.3.2 Support vector machine

SVM is one of the most popular Supervised Learning algorithms used for classification problems. The objective of the SVM algorithm is to find a hyperplane in an N-dimensional space (in other words, N – the number of features) which distinctly classifies the data points. SVM can handle both the linear and non-linear classification problems. For example, in retail industry, Yu et al. (2011) had found promising results for Support vector machine, similar to Guo-en and Wei-dong (2008) who applied SVM method in telecommunication industry. Additionally, for churn prediction problem in SaaS industry, SVM had been shown to be effective.

Table 8 Hyper-parameters list for support vector machine

Hyper-parameter	*Definition*
kernel	the types of kernel used in algorithms
	Kernel type indicate the mapping model of hyperplane to separate the data
C	the values of model simplicity or irregularisation strength
	The higher value of C indicates simpler model. The lower value of C creates more complex model or stronger regularisation
tol	the values of tolerance criteria
	The model stops searching toward the objective or optimisation when the indicated tolerance is reached

Considering hyper-parameters tuning for SVM classifier, the parameters which this paper selected are essential parts to control the model’s complexity or detail in application of different techniques in some steps of classification. In this case, the hyper-parameters for SVM in which this paper is interested consists of three parameters:

1 kernel

2 C

3 tol. (see Table 8)

3.3.3 Decision tree

DT is a non-parametric supervised learning technique used in classification and regression. The goal is to create a model that predicts the value of a target variable by learning simple decision rules inferred from the data features. This meachine learning method can be easily broken down in smaller portions (Tamaddoni Jahromi et al., 2014). DT was found to yield promising results for churn prediction problem by Hung and Wang (2004) whereas, for the case study in Thailand, Wanchai (2017) had shown accurate performance for churn prediction in telecommunication industry. Later, DT classification was also studied in Frank and Pittges (2009) for predicting customer churn in SaaS industry.

Table 9 Hyper-parameters list for decision tree

Hyper-parameter	*Definition*
criterion	the types of function to measure the impurity quality of split at each node 'gini' is defined as Gini impurity or Gini index 'entropy' is defined as information gain
splitter	the techniques used to split samples at each node 'best' is indicated choosing the best split 'random' is indicated choosing the best random split
max_depth	the maximum depth of decision nodes
min_samples_split	the minimum number of samples required to split an internal or decision node
min_samples_leaf	the minimum number of samples required to generate a leaf node

Considering hyper-parameters tuning for DT classifier, the parameters which this paper selected are essential parts to control the model's complexity or detail in application of different techniques in some steps of classification. In this case, the hyper-parameters for DT on which this paper focused consists of 5 parameters;

1 criterion

2 splitter

3 max_depth

4 min_samples_split

5 min_samples_leaf. (see Table 9)

3.3.4 Random forest

Random forest is a flexible, easy-to-use machine learning technique that produces a great result most of the time. It is also one of the most-used algorithms because of its simplicity and diversity and it can be used for both classification and regression problems. For instance, Rautio (2019) studied churn prediction in SaaS and reported that random forest algorithm offers a great performance, almost equivalent to SVM.

Considering hyper-parameters tuning for random forest classifier, the parameters which this paper selected are essential parts to control the model's complexity or detail in application of different techniques in some steps of classification. In this case, the hyper-parameters for random forest on which this paper focused consists of 5 parameters;

1 n_estimaters

2 criterion

3 max_depth

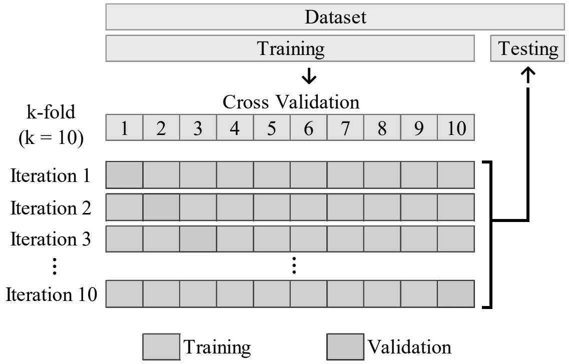
4 min_samples_split

5 min_samples_leaf. (see Table 10)

Table 10 Hyper-parameters list for random forest

Hyper-parameter	*Definition*
n_estimators	the number of decision trees
criterion	the types of function to measure the impurity quality of split at each node 'gini' is defined as Gini impurity or Gini index 'entropy' is defined as information gain
max_depth	the maximum depth of decision nodes
min_samples_split	the minimum number of samples required to split an internal or decision node
min_samples_leaf	the minimum number of samples required to generate a leaf node

Figure 4 K-fold cross validation



3.4 Cross validation

Two methods, i.e., with holdout method and k-fold cross validation (shown in Figure 4.), are applied for this process. The data were separated to 2 groups: 80% of training data and 20% of testing data under holdout method. The evaluation of the final model will be done in the holdout testing set, whereas k-fold cross validation is applied in the training set to negate the effect of bias and overfitting. The k value (in K-fold cross validation) is generally assigned to be 10. In the literature, there is no specific rule for setting k, but the most used value of k is equal to 10 (Kuhn and Johnson, 2013). It is understandable that when k = 10, it is already appropriate for the high variance data. Thus, in this research we randomly split the cross-validation dataset into 10 parts, using 10 iterations for different validation datasets. Then, the selected evaluation metrics from k-fold cross validation are applied to measure the prediction models' results. After that, the top performed model will be tested in the final model using holdout testing dataset to represent the unknown real data to confirm the competence of the model performance.

3.5 Evaluation metrics

To compare the performance of prediction models, we refer to confusion matrix which is known to be a common tool to measure the prediction results as compared to the actual one. Table 11 shows the confusion matrix, having 4 different dimensions: accuracy, precision, recall and F1-score. Equations (1) to (4) present the formula of those measurements, respectively.

Table 11 Confusion matrix

		Predicted	
		Negative (Non-churn)	*Positive (Churn)*
Actual	Negative (Non-churn)	TN	FP
	Positive (Churn)	FN	TP

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

$$F1\ score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{4}$$

Recall as chosen to be our main essential metric because if a real churn customer is predicted as a non-churn customer (i.e., false negative), it will brutally impact the balance sheet of the company. As a result, the scoring parameter in grid search mentioned in machine learning section is evaluated by recall. On the other hand, *precision* values false positive as the most significant is the last important one in this case. In this paper, we consider accuracy and *F1-score* to be the same important indicator. A weakness of *accuracy* is that it depends on data classification balance whereas *F1-score* is able to deal with imbalanced data. For this dataset, there are 53.96% of churn samples and 46.04% of non-churn ones, which can be summarised as a balanced dataset. *Accuracy* is therefore still valid. Yet, *F1-score* is considered as another important metric which still outweighs *accuracy* according to the mentioned reasons. Overall, the results of final customer churn prediction models are required to be at least 0.8 or 80% in every dimension (*recall*, *F1-score, accuracy,* and *precision*) regarding the case-study company's goal.

4 Experimental result

This research applies four machine learning algorithms on the data granted by the case-study company. The raw data was extracted from the case-study company database while the whole raw data consists of 1,788 observations with 23 features in both customer usage behaviour and business matric.

Figure 5 Seasonal decomposition for churn (see online version for colours)

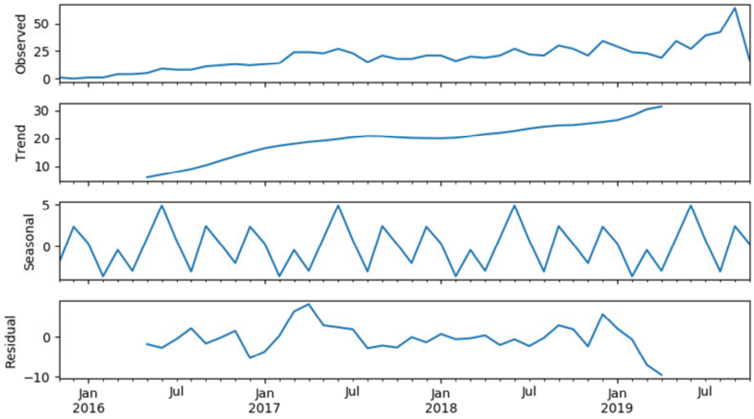
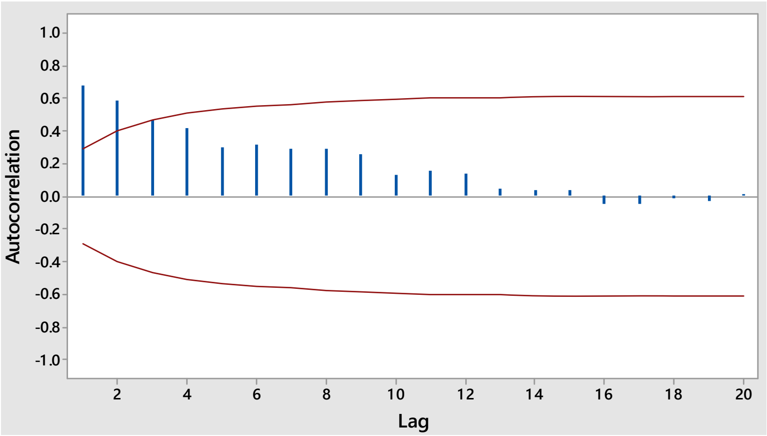


Figure 6 Autocorrelation function for churn (see online version for colours)



Regarding data transformation as explained in the methodology, the dataset had been firstly removed every missing-value observation by listwise method that is data cleaning step. As result, the 1,788 observations of the whole raw data had been cleaned the missing data to become 1,718 observations. The data includes both churn and non-churn customer records; 927 churn samples and 791 non-churn samples. This original dataset tends to have likely balanced distribution with 53.96% of churn samples and 46.04% of non-churn samples.

Then, the data processing had been secondly proceeded in order to prepare data to be readily executed in other steps. the data processing includes one hot encoding method used to convert the categorial data and standardisation used to regularise the dataset without effect of different means.

However, it was possible that these data might be affected by seasonality. Therefore, seasonality test was necessary to set off this concern. These data were decomposed in three components (refer to Figure 5):

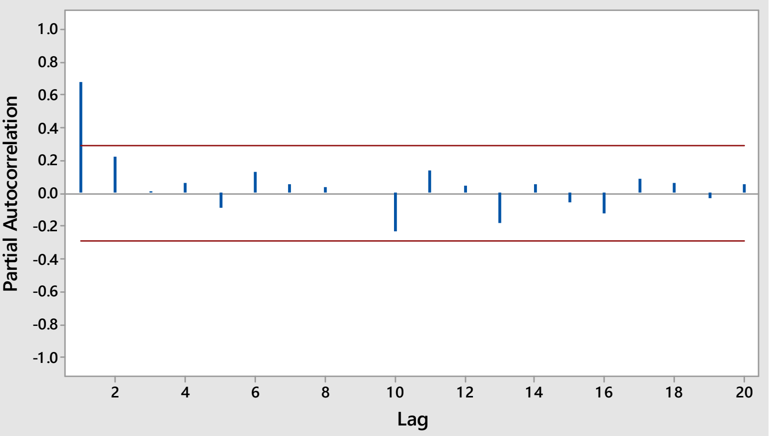
1 trend

2 seasonal

3 residual.

It illustrated that the portion of seasonal was small amount compared others. Moreover, autocorrelation function and partial autocorrelation function were applied in this case in order to confirm whether the data were random without seasonality effect. The autocorrelation function for churn (with 5% significance limits for the autocorrelations) and partial autocorrelation function for Churn (with 5% significance limits for the partial autocorrelations) shown in Figure 6 and Figure 7 respectively can be implied that the churn data were not significantly correlated to the data in lag period in a seasonal pattern.

Figure 7 Partial autocorrelation function for churn (see online version for colours)



And the last step in data transformation is feature selection, the dataset was evaluated the feature importance by two types of univariate feature selection or statistical filter method for classification, i.e., Chi-squared score and ANOVA F-values. The order of feature importance calculated by ANOVA F-values is shown in Figure 8. while the order of feature importance determined by Chi-squared score is shown in Figure 9. the higher score feature means more related to output or churn in this case considering the feature selection methods.

Figure 8 Feature importance by ANOVA (see online version for colours)

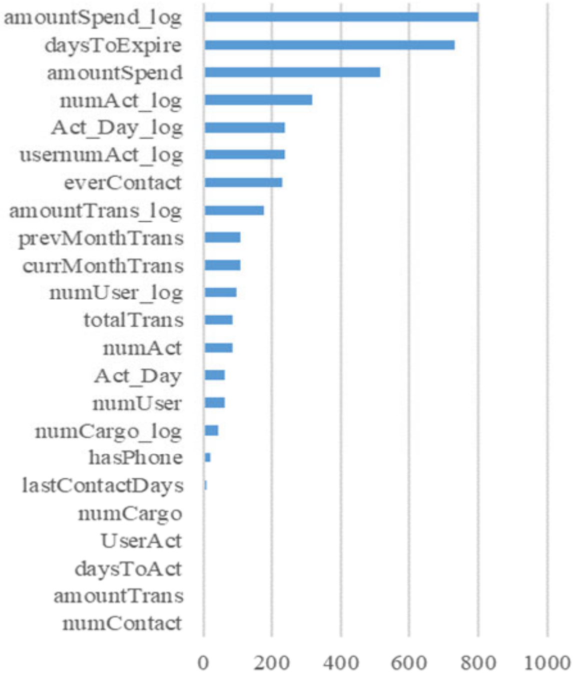
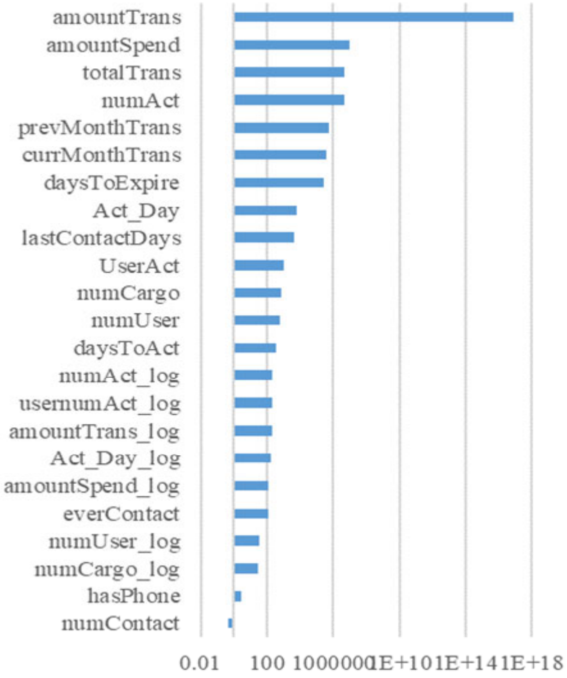


Figure 9 Feature importance by chi-squared (see online version for colours)



After data transformation had been executed, the transformed data which consisted of 1,718 observations of 23 explanatory variables was separated into 2 portions: 80% of training dataset and 20% of testing dataset. It is noticeable that the split data was kept the ratio of churn data to be the same or as close as original dataset which are approximately 54% of churn and 46% of non-churn. It is important to imitate the proportion of churn to training dataset to preserve the condition of distribution.

Next, the split training dataset which selected data varied by the order of feature importance, both ANOVA filter method and Chi-squared filter method, had trained in machine learning classification models with grid search used for hyper-parameters tuning and k-fold cross validation (k = 10) as follows:

4.1 Logistic regression

Logistic regression algorithm is performed as the classifier. The logistic regression was trained with training dataset which had already done the data transformation. The logistic regression model was generated using Scikit-learn as mentioned in methodology section.

After training data with varying the number of selected features (n) as a certain parameter on logistic regression models, each logistic regression model with n-number selected features had been optimised toward recall with its own optimal value set of hyper-parameters whilst, as previously described, the hyper-parameters for logistic regression in which this paper is interested consists of three parameters:

1 penalty

2 C

3 tol.

The evaluation results applied grid search used for hyper-parameters tuning and k-fold cross validation (k = 10) are shown in Figure 10 and Figure 11. And it is noted that the results of model are required to be at least 0.8 or 80% as a baseline in every dimension (*recall, F1-score, accuracy,* and *precision*) regarding the case-study company.

The results of logistic regression optimised hyper-parameters and varied the number of feature importance (n) by ANOVA filter method in Figure 10 is illustrated that every n-number selected feature can provide above the baseline requirement of 0.8 or 80% in every dimension (*recall, F1-score, accuracy,* and *precision*). However, recall scores of every model which had not passed over 0.9 or 90% is slightly low compared to other dimension despite of that recall is the most preferable metric in this case. Regarding to the experimental results, the top-performance model of logistic regression with ANOVA filter method is the optimised logistic regression with top 21 importance features (n = 21) selected by ANOVA method. This model can provide recall of 86.8%, F1-score of 90.2%, accuracy of 89.6% and precision of 93.9%.

Considering this optimised model of logistic regression with top 21 importance features (n = 21) selected by ANOVA method, grid search values for each hyper-parameter are presented in Table 13. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

The results of logistic regression optimised hyper-parameters and varied the number of feature importance (n) by Chi-squared filter method in Figure 11 are illustrated that although, if the n-number selected features are lower than 7 features, recall score is steeply increased, F1-score, accuracy and precision is reversely plunged down and lower than the baseline requirement of 0.8 or 80%. This can be implied that those logistic regression models of n-number selected feature by Chi-squared method is not suitable to be applied in the case-study. While the top-performance model of logistic regression with Chi-squared filter method is the optimised logistic regression with top 22 importance features (n = 22) selected by Chi-squared method. This model can provide recall of 86.8%, F1-score of 90.2%, accuracy of 89.6% and precision of 93.9%, similar to the logistic regression model with ANOVA filter method. However, it still has the same issue with low recall score compare to other dimensions.

Figure 10 Results of logistic regression with ANOVA filter method (see online version for colours)

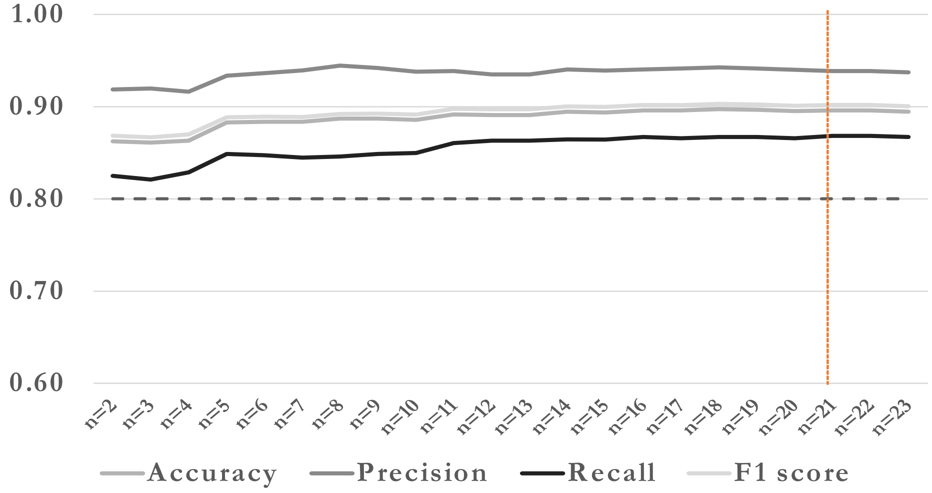


Table 12 Hyper-parameters of logistic regression with ANOVA filter method

Hyper-parameter	*Grid search values*	*Optimal value*
penalty	l1, l2	l2
C	1, 10, 100	10
tol	0.01,0.001,0.0001,0.00001	0.001

Considering this optimised model of SVM with top 18 importance features (n = 18) selected by ANOVA method, grid search values for each hyper-parameter are presented in Table 15. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Focusing on the optimised model of logistic regression with top 22 importance features (n = 22) selected by Chi-squared method, grid search values for each hyper-parameter are presented in Table 14. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Figure 11 Results of logistic regression with chi-squared filter method (see online version for colours)

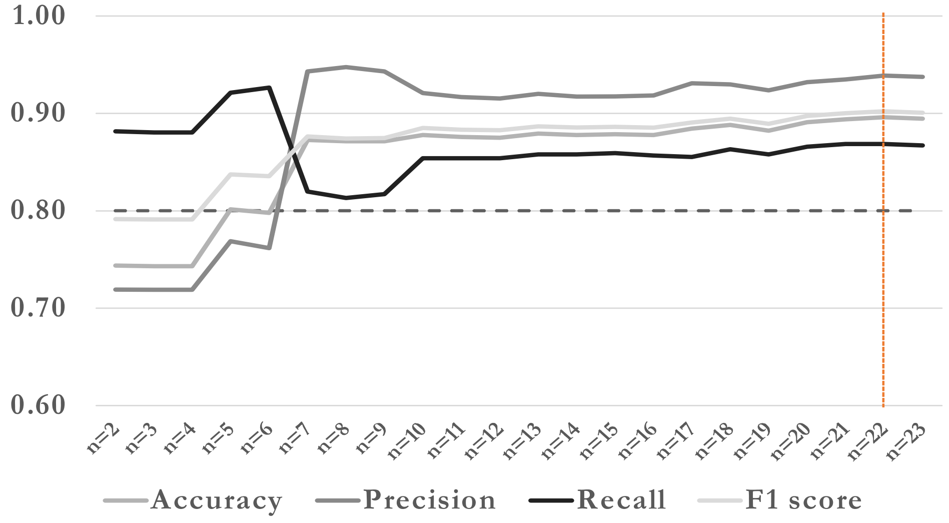


Table 13 Hyper-parameters of logistic regression with chi-squared filter method

Hyper-parameter	*Grid search values*	*Optimal value*
penalty	l1, l2	l2
C	1, 10, 100	10
tol	0.01,0.001,0.0001,0.00001	0.001

4.2 Support vector machine

Using the same training data as for logistic regression, classifier is set to be SVM algorithm. The SVM was trained with transformed training dataset completely cleaned and processed with encoding and standardisation. The SVM model was created using Scikit-learn as described in methodology section.

After training data with varying the number of selected features (n) as a certain parameter on SVM models, each SVM model with n-number selected features had been optimised in accordance with recall score with its own optimal value set of hyper-parameters whilst, as was pointed out earlier, the hyper-parameters for SVM in which this paper is interested consists of three parameters:

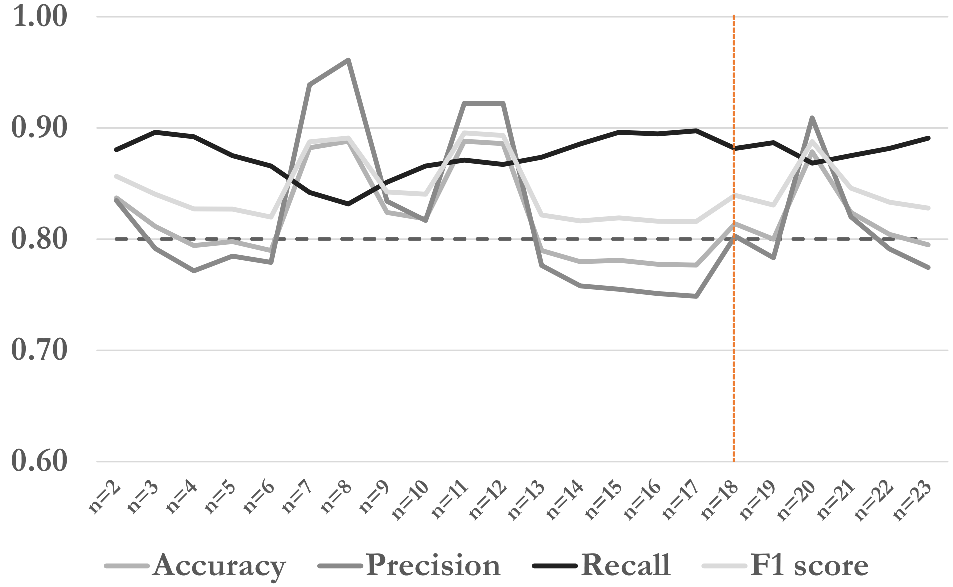
1 kernel

2 C

3 tol.

The evaluation results applied grid search cross validation used for hyper-parameters tuning and k-fold cross validation (k = 10) are shown in Figure 12 and Figure 13. And it is mentioned that the results of model are required to be at least 0.8 or 80% as a baseline in every dimension (*recall, F1-score, accuracy,* and *precision*) regarding the case-study company.

Figure 12 Results of support vector machine with ANOVA filter method (see online version for colours)



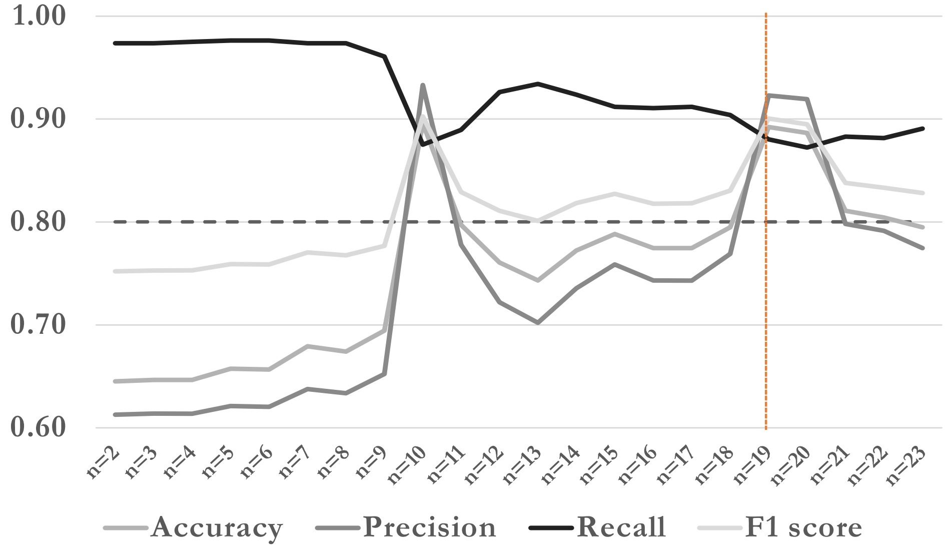
The results of SVM optimised hyper-parameters and varied the number of feature importance (n) by ANOVA filter method in Figure 12 is illustrated that every n-number selected feature can provide recall score above a 0.8 or 80% baseline. In spite of that, other dimensions (*F1-score, accuracy* and *precision)* are dramatically low. In a contrary, when F1-score, accuracy and precision get improved, recall score seems to be declined. Regarding to the experimental results, the top-performance model of SVM with ANOVA filter method is the optimised SVM with top 18 importance features (n = 18) selected by ANOVA method. This model can provide recall of 88.1%, F1-score of 83.9%, accuracy of 81.4% and precision of 80.3% that meets the criteria of 0.8 or 80% in overall scores.

Considering this optimised model of SVM with top 18 importance features (n = 18) selected by ANOVA method, grid search values for each hyper-parameter are presented in Table 15. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Table 14 Hyper-parameters of support vector machine with ANOVA filter method

Hyper-parameter	*Grid search values*	*Optimal value*
Kernel	linear, poly, rbf	poly
C	1, 10, 100	1
tol	0.01,0.001,0.0001,0.00001	0.001

Figure 13 Results of support vector machine with chi-squared filter method (see online version for colours)



The results of SVM optimised hyper-parameters and varied the number of feature importance (n) by chi-squared filter method in Figure 13 is illustrated that although many n-number of selected feature by chi-squared method can improve recall of the SVM model over 0.9 or 90% score, other dimensions (F1-score, accuracy and precision) are dropped sharply, especially lower than 10 selected features (n) that accuracy and precision scores dive to nearly 0.6 or 60% score. While the top-performance model of SVM with chi-squared filter method is the optimised SVM with top 19 importance features (n = 19) selected by chi-squared method. This model can provide recall of 88.0%, F1-score of 90.0%, accuracy of 89.2% and precision of 92.3%. This recall score is almost equivalent to the recall of the SVM model with ANOVA filter method. Nevertheless, other dimensions of SVM model with chi-squared filter method are significantly higher than SVM model with ANOVA filter method.

Focusing on the optimised model of SVM with top 19 importance features (n = 19) selected by chi-squared method, grid search values for each hyper-parameter are presented in Table 16. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Table 15 Hyper-parameters of support vector machine with chi-squared filter method

Hyper-parameter	*Grid search values*	*Optimal value*
Kernel	linear, poly, rbf	poly
C	1, 10, 100	100
tol	0.01,0.001,0.0001,0.00001	0.001

4.3 Decision tree

Similar to support vector machine, DT algorithm is performed as the classifier. The DT was trained with training dataset which had already done the data transformation. The DT model was built using Scikit-learn as explained earlier in methodology section.

After training data with varying the number of selected features (n) as a certain parameter on DT models, each DT model with n-number selected features had been optimised considering recall as objective scorer with its own optimal value set of hyper-parameters whilst, as was mentioned previously, the hyper-parameters for DT on which this paper focused consists of 5 parameters;

1 criterion

2 splitter

3 max_depth

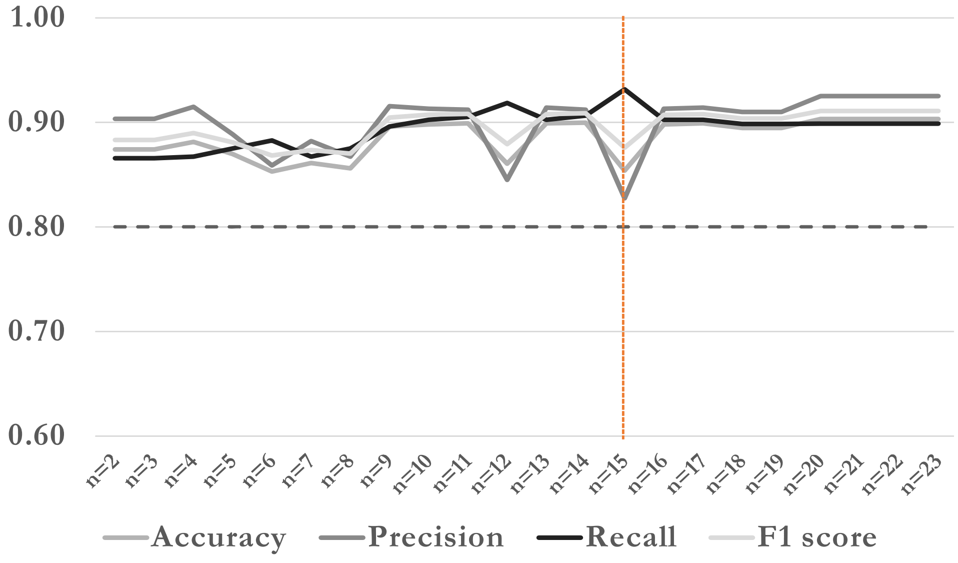
4 min_samples_split

5 min_samples_leaf.

The evaluation results applied grid search cross validation used for hyper-parameters tuning and k-fold cross validation (k = 10) are shown in Figure 14 and Figure 15. And it is noted that the results of model are required to be at least 0.8 or 80% as a baseline in every dimension (*recall, F1-score, accuracy,* and *precision*) regarding the case-study company.

The results of DT optimised hyper-parameters and varied the number of feature importance (n) by ANOVA filter method in Figure 14 is illustrated that

Figure 14 Results of decision tree with ANOVA filter method (see online version for colours)



This DT model with every n-number selected feature by ANOVA filter method can provide evaluation scores above the 0.8 or 80% baseline requirement in every dimension (*recall, F1-score, accuracy,* and *precision*).

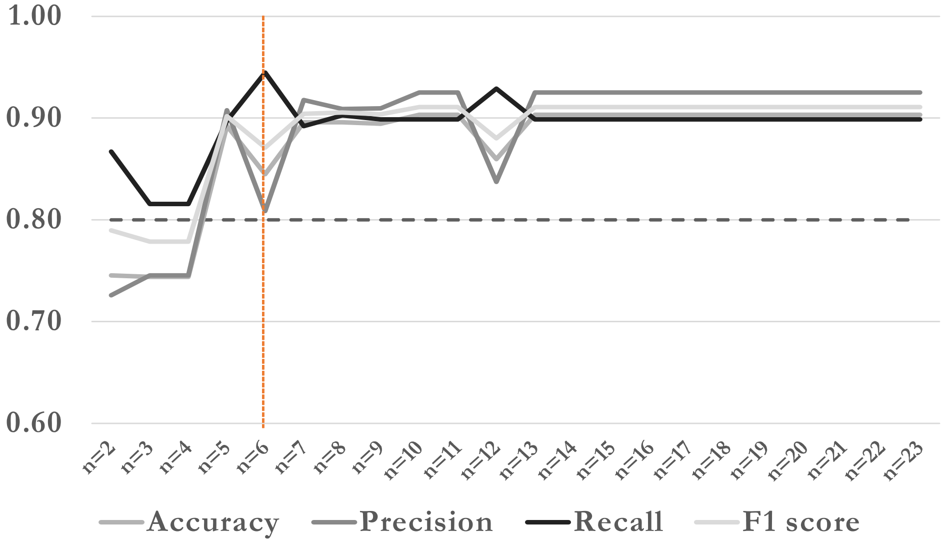
Every n-number selected feature can provide above the baseline requirement of 0.8 or 80% in every dimension (*recall, F1-score, accuracy,* and *precision*). However, recall scores of every model which had not passed over 0.9 or 90% is slightly low compared to other dimension despite of that recall is the most preferable metric in this case. The recall scores seem to be slightly increased when more n-number selected features. However, the DT model with 15 selected features (n = 15) by ANOVA method is outperformed the others in recall score. As a result, the top-performance model of DT with ANOVA filter method is the optimised DT with top 15 importance features (n = 15) selected by ANOVA method. This model can provide recall of 93.2%, F1-score of 87.6%, accuracy of 85.4% and precision of 82.7%.

Considering this optimised model of DT with top 15 importance features (n = 15) selected by ANOVA method, grid search values for each hyper-parameter are presented in Table 16. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Table 16 Hyper-parameters of decision tree with ANOVA filter method

Hyper-parameter	*Grid search values*	*Optimal value*
criterion	gini, entropy	entropy
splitter	best, random	random
max_depth	[1–10], None	3
min_samples_split	2, 4, 6, 8, 10	2
min_samples_leaf	[1–10]	1

Figure 15 Results of decision tree with chi-squared filter method (see online version for colours)



The results of DT optimised hyper-parameters and varied the number of feature importance (n) by Chi-squared filter method in Figure 15 is illustrated that although the DT models with 2 to 4 selected features (n < 5) by Chi-squared method cannot reach the criteria of a baseline requirement of 0.8 or 80% score in *F1-score, accuracy* and *precision*, the models with above 5 n-number selected features (n > 4) by Chi-squared method are able to satisfy the baseline criteria in every evaluation dimension (*recall, F1-score, accuracy,* and *precision*). Regarding recall score, the top-performance model of DT with Chi-squared filter method is the optimised DT with top 6 importance features (n = 6) selected by Chi-squared method. This model can provide recall of 94.5%, *F1-score* of 87.1%, *accuracy* of 84.5% and *precision* of 80.9%.

Focusing on the optimised model of DT with top 6 importance features (n = 6) selected by Chi-squared method, grid search values for each hyper-parameter are presented in Table 17. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

Table 17 Hyper-parameters of decision tree with chi-squared filter method

Hyper-parameter	*Grid search values*	*Optimal value*
criterion	gini, entropy	gini
splitter	best, random	random
max_depth	[1–10], None	3
min_samples_split	2, 4, 6, 8, 10	2
min_samples_leaf	[1–10]	1

4.4 Random forest

Same as decision tree, random forest algorithm is treated as the classifier. The random forest model was trained with the same training dataset which had already done the data transformation (data cleaning and data processing). The random forest model was built using Scikit-learn as was previously pointed out in methodology section.

After training data with varying the number of selected features (n) as a particular parameter on logistic regression models, each logistic regression model with n-number selected features had been optimised toward the purpose of recall scorer with its own optimal value set of hyper-parameters whilst, as stated earlier, the hyper-parameters for random forest on which this paper focused consists of 5 parameters;

1 n_estimaters

2 criterion

3 max_depth

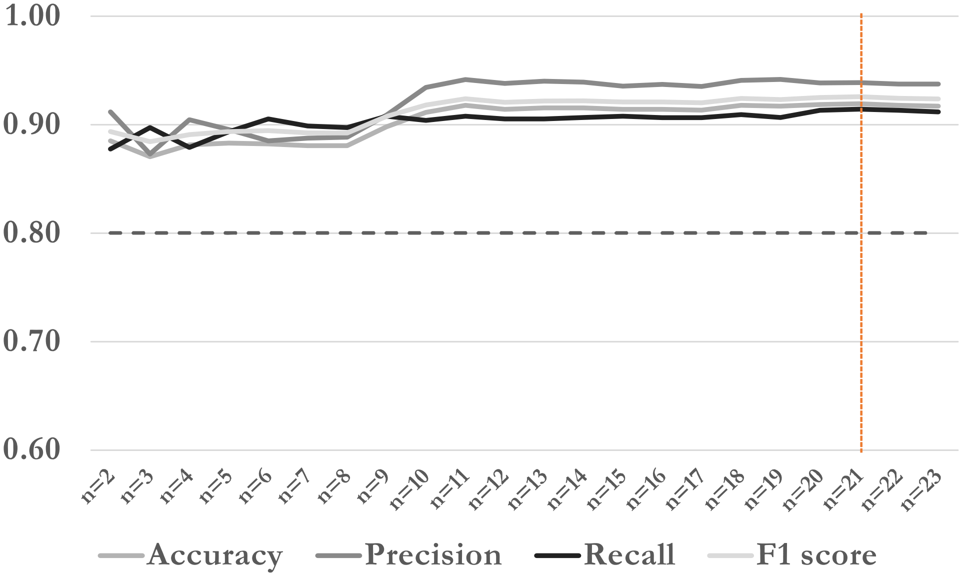
4 min_samples_split

5 min_samples_leaf.

The evaluation results applied grid search cross validation used for hyper-parameters tuning and k-fold cross validation (k = 10) are shown in Figure 16 and Figure 17. And it is noted that the results of model are required to be at least 0.8 or 80% as a baseline in every dimension (*recall, F1-score, accuracy,* and *precision*) regarding the case-study company.

The results of random forest optimised hyper-parameters and varied the number of feature importance (n) by ANOVA filter method in Figure 16 is illustrated that every n-number selected feature can satisfy the requirement of 0.8 or 80% score in every evaluation dimension (Recall, F1-score, Accuracy, and Precision). Even though the random forest model with more than 9 number selected features (n > 9) by ANOVA method tend to have steadily outcomes and the values of scores close to each other, the top recall score model of random forest with ANOVA filter method is the optimised random forest with top 21 importance features (n = 21) selected by ANOVA method. This model can provide recall of 91.4%, F1-score of 92.6%, accuracy of 91.9% and precision of 93.9% that is outstanding with over 0.9 or 90% score in every dimension.

Figure 16 Results of random forest with ANOVA filter method (see online version for colours)



Considering this optimised model of random forest with top 21 importance features (n = 21) selected by ANOVA method, grid search values for each hyper-parameter are presented in Table 18. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

The results of logistic regression optimised hyper-parameters and varied the number of feature importance (n) by Chi-squared filter method in Figure 17 is illustrated that every n-number selected feature can provide above the baseline requirement of 0.8 or 80% in every dimension (*recall, F1-score, accuracy,* and *precision*) except 2, 3 and 4 number selected features (n < 5). The random forest models with more than 4-number selected features (n > 4) by ANOVA method seem to provide steadily outcomes and the values of scores close to each other. Noticeably, the models with more than 4-number selected features (n > 4) by ANOVA method do not only meet the criteria but these models also perform over 0.9 or 90% score in every evaluation dimension.

Table 18 Hyper-parameters of random forest with ANOVA filter method

Hyper-parameter	*Grid search values*	*Optimal value*
n_estimators	10, 20, 40, 60, 80, 100, 200	200
criterion	gini, entropy	gini
max_depth	[1–10], None	None
min_samples_split	2, 4, 6, 8, 10	2
min_samples_leaf	[1–10]	1

Considering the best recall score, the top-performance model of random forest with Chi-squared filter method is the optimised random forest with top 11 importance features (n = 11) selected by Chi-squared method. This model can provide recall of 91.6%, F1-score of 92.6%, accuracy of 92.0% and precision of 93.9%, be slightly better scores comparing to the random forest model with ANOVA filter method.

Figure 17 Results of random forest with chi-squared filter method (see online version for colours)

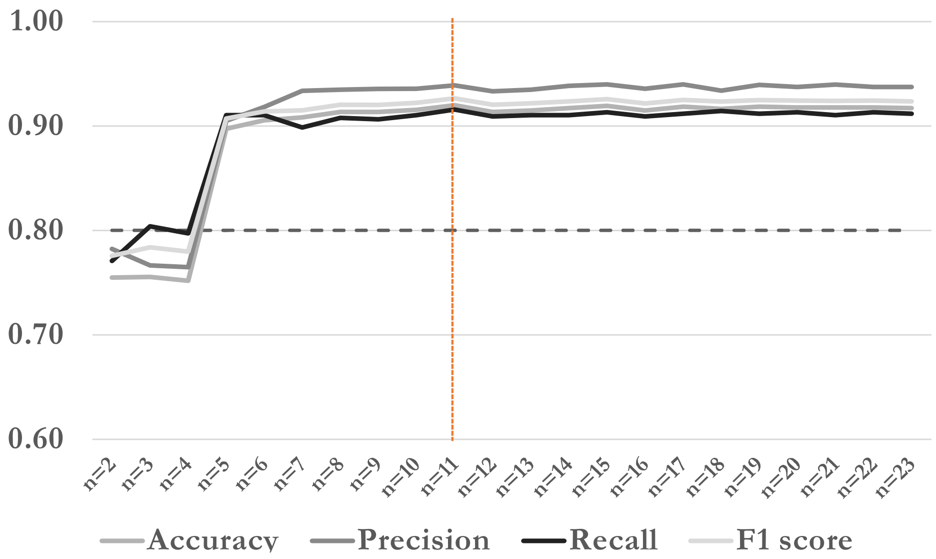


Table 19 Hyper-parameters of random forest with chi-squared filter method

Hyper-parameter	*Grid search values*	*Optimal value*
n_estimators	10, 20, 40, 60, 80, 100, 200	100
criterion	gini, entropy	gini
max_depth	[1–10], None	None
min_samples_split	2, 4, 6, 8, 10	2
min_samples_leaf	[1–10]	1

Focusing on the optimised model of random forest with top 11 importance features (n = 11) selected by Chi-squared method, grid search values for each hyper-parameter are presented in Table 19. And it also shows the results of the optimal values for each hyper-parameter using in the optimised model.

4.5 Results comparison

Considering the top-performance of each model classification with different filter methods, these top- performance models results are presented in Table 20. The top-performance models are listed herein including 8 models as the following:

1 the optimised logistic regression with top 21 importance features selected by ANOVA method (LR ANOVA n = 21)

2 the optimised model of logistic regression with top 22 importance features selected by chi-squared method (LR chi2 n = 22)

3 the optimised model of SVM with top 18 importance features selected by ANOVA method (SVM ANOVA n = 18)

4 the optimised model of SVM with top 19 importance features selected by chi-squared method (SVM Chi2 n = 19)

5 the optimised model of DT with top 15 importance features selected by ANOVA method (DT ANOVA n = 15)

6 the optimised model of DT with top 6 importance features selected by chi-squared method (DT chi2 n = 6)

7 the optimised model of random forest with top 21 importance features selected by ANOVA method (RF ANOVA n = 21)

8 the optimised model of random forest with top 11 importance features selected by chi-squared method (RF chi2 n = 11).

From result comparison, the optimised DT model with top 6 importance features selected by chi-squared method outperform other models regarding the objective towards recall scorer with recall of 94.5%, F1-score of 87.1%, accuracy of 84.5% and precision of 80.9%. Even though the optimised DT model with top 6 importance features selected by chi-squared method is the best models in term of recall score, it is noticeable that in term of overall evaluation results, the optimised random forest models both with top 21 importance features selected by ANOVA method and with top 11 importance features selected by chi-squared method are capable to provide over 0.9 or 90% scores in every dimension instead of focusing on only recall.

As a result of classifiers comparison in previous step, the model that performed outstandingly apart from other algorithms is the optimised DT model with top 6 importance features selected by chi-squared method. Therefore, to confirm the performance of the final model, this model is necessary to be investigated holdout cross validation with firstly separated testing dataset. The testing results are shown in Table 21. In testing result, the optimised DT model with top 6 importance features selected by chi-squared method is capable to perform similarly to the result of cross validation training dataset with 94.4% of recall and 88.2% of F1-score, 85.8% of accuracy and 82.9% of precision that can also satisfy baseline criteria of 0.8 or 80% score in every evaluation metrics (*recall, F1-score, accuracy,* and *precision)*.

Table 20 Result comparison

Models	*Evaluation metrics*			
	Accuracy	*Precision*	*Recall*	*F1-score*
LR ANOVA n = 21	0.896	0.939	0.868	0.902
LR Chi2 n = 22	0.896	0.939	0.868	0.902
SVM ANOVA n = 18	0.814	0.803	0.881	0.839
SVM Chi2 n = 19	0.892	0.923	0.880	0.900
DT ANOVA n = 15	0.854	0.827	0.932	0.876
DT Chi2 n = 6	0.845	0.809	0.945	0.871
RF ANOVA n = 21	0.919	0.939	0.914	0.926
RF Chi2 n = 11	0.920	0.939	0.916	0.926

Table 21 Testing result of the optimised decision tree model with top 6 importance features selected by chi-squared method

DT *Chi2* *n = 6*	*Evaluation metrics*			
	Accuracy	*Precision*	*Recall*	*F1-score*
Training Result	0.845	0.809	0.945	0.871
Testing Result	0.858	0.829	0.944	0.882

Since these evaluation metrics are generally originated from the calculation of confusion matrix which describes the outcome between predications and actuals in a binary classification as was mentioned previously in methodology, Table 22 shows details in confusion matrix of the testing result of final model, the optimised DT model with top 6 importance features selected by chi-squared method.

Table 22 Confusion matrix of the holdout testing result of the optimised decision tree model with top 6 importance features selected by chi-squared method

		Predicted	
		Negative (Non-churn)	*Positive (Churn)*
Actual	Negative (Non-churn)	111	38
	Positive (Churn)	11	184

The final model can be used to evaluate the classification model and describe the important features. Table 23 summarises of the top 6 importance features sorted by feature importance scores deriving from the final model. It can be seen that the important features are business features such as transaction and usage frequency.

Table 23 Confusion matrix of the holdout testing result of the optimised decision tree model with top 6 importance features selected by chi-squared method

Rank	*Attribute name*	*Description*
1	amountTrans	Amount of transaction values via platform
2	amountSpend	Amount of customer spending
3	numAct	Number of actions per customer
4	totalTrans	Number of all transactions
5	currPeriodTrans	Number of transactions in current period of 14 days
6	prevPeriodTrans	Number of transactions in previous period of 14 days

5 Conclusions

The market of SaaS industry has continuously growing because of the e-commerce global trend. The case-study company is an inventory management SaaS in Thailand having a goal of scaling up in this market. Nevertheless, currently the company is having high customer churn. Churn prediction model developed in this paper could help the company indicating the right customers who are likely to churn and assist them in time.

This research presented effective customer churn prediction models focusing on recall performance measures. The best performed model is found to be DT model using top 6 importance features selected by Chi-squared filter method. Its capability in prediction with the highest average training recall score and being above case-study baseline criteria of 0.8 or 80% in every performance metrics. The training result can score up to 94.5% recall, 87.1% F1-score, 84.5% accuracy and 80.9% precision. Additionally, the holdout testing result of this model is also validated its performance in customer churn prediction with 94.4% of recall and 88.2% of F1-score, 85.8% of accuracy and 82.9% of precision. Since this final prediction model of DT classifier was optimised by hyper-parameters tuning, the optimal values for each hyper-parameter using grid search cross validation are including with, 1 gini criterion

2 Random splitter

3 3 of max_dept

4 2 of min_samples_split

5 1 of min_samples_leaf.

Hence this final model can satisfy the case-study objective indicating real churn customer from all churn customer correctly and perform better than other models.

Furthermore, the classification model provides details in attributes or important futures that related to customer churn. The importance scores provide managerial insights to the company to learn that business metric, i.e., transaction and usage frequency, are the top features. In other words, if customer is online more frequent and creates more amount of transactions via the platform, that customer is less likely to churn.

However, it is noticeable that the final model proposed in this paper is under the objective of case-study to receive the best recall score while others are needed to only be above 0.8 or 80% baseline. In case of overall performance, it is recommended that the optimised random forest models both with top 21 importance features selected by ANOVA method and with top 11 importance features selected by chi-squared method are more suitable to be properly used as the churn classification prediction model with over 0.9 or 90% in all dimensions according to the results. Thus, if the case-study company found that the prediction model needs to be improved in other evaluation dimensions in order to meet new criteria in the future, these optimised random forest models would be able to perform properly as well.

In conclusion, this paper provides the final model of customer churn prediction and features importance list that is able to satisfy the objective of this paper and also the case-study company requirements. As result, the company can indicate the right risky churn customers by applying the final customer churn prediction model and offer them with active marketing campaigns. This can enhance the efficiency and effectiveness of managerial decision for the company.

There are some limitations of the studied dataset. Most of available features had been exposed in business-related feature usage area, e.g., transactions and number of various types of usage. However, these data still lack of some important quality metrics such as service rating or customer satisfaction in many dimensions. Moreover, after this research had provided the customer churn prediction model, it is interesting to extend these models to forecast the number of customers, the future transaction and also the revenue of the company (Sukow and Grant, 2013). Due to the nature of SaaS business model as subscription base, customer churn is potentially variated to the projected future revenues. Future research can also apply more concrete information like customers' financial data in the prediction model.

References

Ali, O. and Arıtürk, U. (2014) 'Dynamic churn prediction framework with more effective use of rare event data: the case of private banking', *Expert Systems with Applications*, Vol. 41, No. 17, pp.7889–7903, https://doi.org/10.1016/j.eswa.2014.06.018.

Allison, P. (2002) *Missing Data, Quantitative Applications in the Social Sciences*, SAGE Publications, Inc., Thousand Oaks, California.

Amin, A., Al-Obeidat, F., Shah, B., Adnan, A., Loo, J. and Anwar, S. (2019) 'Customer churn prediction in telecommunication industry using data certainty', *Journal of Business Research*, Vol. 94, pp.290–301, https://doi.org/10.1016/j.jbusres.2018.03.003

Bain & Company, Inc., (2000) *The Value of Online Customer Loyalty and How you Can Capture it*, 1 April 2000 [online] https://www.bain.com/insights/the-value-of-online-customer-loyalty-and-how-you-can-capture-it/. (accessed 14 February 2020).

Berger, P. and Kompan, M. (2019) 'User modeling for churn prediction in e-commerce', *IEEE Intelligent Systems*, Vol. 34, No. 2, pp.44–52, https://doi.org/10.1109/MIS.2019.2895788.

Buckinx, W. and Van den Poel, D. (2005) 'Customer base analysis: partial defection of behaviourally loyal clients in a non-contractual FMCG retail setting', *European Journal of Operational,* Vol. 164, No. 1, pp.252–268, https://doi.org/10.1016/j.ejor.2003.12.010.

Chen, S. (2016) 'The gamma CUSUM chart method for online customer churn prediction', *Electronic Commerce Research and Applications*, Vol. 17, No. 1, pp.99–111, https://doi.org/10.1016/j.elerap.2016.04.003

Frank, B. and Pittges, J. (2009) *Analyzing Customer Churn in the Software as a Service* (SaaS) *Industry*, Radford University, Virginia.

Gallo, A. (2014) *The Value of Keeping the Right Customers*, Harvard Business School Publishing Corporation, 29 October 2014 [online] https://hbr.org/2014/10/the-value-of-keeping-the-right-customers. (accessed 14 February 2020).

Gartner Forecasts Worldwide Public Cloud Revenue to Grow 17% in 2020 (2020) Gartner Inc., 13 November 2019 [online] https://www.gartner.com/en/newsroom/press-releases/2019-11-13-gartner-forecasts-worldwide-public-cloud-revenue-to-grow-17-percent-in-2020 (accessed 14 February 2020).

Ge, Y., He, S., Xiong, J. and Brown, D.E. (2017) 'Customer churn analysis for a software-as-a-service company', in *2017 Systems and Information Engineering Design Symposium* (SIEDS), Charlottesville, VA.

Glady, N., Baesens, B. and Croux, (2009) 'Modeling churn using customer lifetime value', *European Journal of Operational Research*, Vol. 197, No. 1, pp.402–411.

Guo-en, X. and Wei-dong, J. (2008) 'Model of customer churn prediction on support vector machine', *Systems Engineering–Theory and Practice*, Vol. 28, No. 1, pp.71–77, https://doi.org/10.1016/S1874-8651(09)60003-X.

Hemalatha, P. and Amalanathan, G.M. (2019) 'A hybrid classification approach for customer churn prediction using supervised learning methods: banking sector', in *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking* (ViTECoN), Vellore, India.

Hung, S. and Wang, H. (2004) 'Applying data mining to telecom churn management', in *Pacific Asia Conference on Information Systems* (PACIS).

Khodabandehlou, S. and Zivari Rahman, M. (2017) 'Comparison of supervised machine learning techniques for customer churn prediction based on analysis of customer behavior', *Journal of Systems and Information Technology*, Vol. 19, Nos. 1–2, pp.65–93, https://doi.org/10.1108/JSIT-10-2016-0061.

Kuhn, M. and Johnson, K. (2013) 'Over-fitting and model tuning', in *Applied Predictive Modeling*, New York, Springer, pp.61–92, https://doi.org/10.1109/ISBI.2017.7950735.

Lu, N., Lin, H., Lu, J. and Zhang, G. (2014) 'A customer churn prediction model in telecom industry using boosting', *IEEE Transactions on Industrial Informatics*, Vol. 10, No. 1, pp.1659–1665, https://doi.org/10.1109/TII.2012.2224355.

MarketWatch (2019) *Global Inventory Management Software Market to Hit USD 3 billion by 2024,* MarketWatch, Inc., 26 November 2019 [online] https://www.marketwatch.com/press-release/global-inventory-management software-market-to-hit-usd-3-billion-by-2024-2019-11-26 (accessed 14 February 2020).

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M. and Perrot, M. (2011) 'Scikit-learn: machine learning in Python', *The Journal of Machine Learning Research* (JMLR), Vol. 12, pp.2825–2830.

Rautio, A. (2019) *Churn Prediction in SaaS using Machine Learning,* Tampere University, Tampere, Finland.

Reichheld, F. (2001) 'BAIN & COMPANY, INC.', September 2001 [online] http://www2.bain.com/Images/BB_Prescription_cutting_costs.pdf. (accessed 14 February 2020).

Sukow, A. and Grant, R. (2013) 'Forecasting and the role of churn in software-as-a-service business models', *iBusiness*, Vol. 5, No. 1, pp.49–57, https://doi.org/10.4236/ib.2012.51A006.

Tamaddoni Jahromi, A., Stakhovych, S. and Ewing, M. (2014) 'Managing B2B customer churn, retention and profitability', *Industrial Marketing Management*, Vol. 43, No. 7, pp.1258–1268, https://doi.org/10.1016/j.indmarman.2014.06.016.

Umayaparvathi, V. and Iyakutti, K. (2012) 'Applications of data mining techniques in telecom churn prediction', *International Journal of Computer Applications*, Vol. 42, No. 20, pp.5–9, https://doi.org/10.5120/5814-8122.

Vadakattu, R., Panda, B., Narayan, S. and Godhia, H. (2015) 'Enterprise subscription churn prediction', in *2015 IEEE International Conference on Big Data* (Big Data), Santa Clara, California.

Wanchai, P. (2017) 'Customer churn analysis: a case study on the telecommunication industry of Thailand', in *2017 12th International Conference for Internet Technology and Secured Transactions* (ICITST), Cambridge.

Yu, X., Guo, S., Guo, J. and Huang, X. (2011) 'An extended support vector machine forecasting framework for customer churn in e-commerce', *Expert Systems with Applications*, Vol. 38, No. 3, pp.1425–1430, https://doi.org/10.1016/j.eswa.2010.07.049.