PREDICTION OF CUSTOMER CHURN IN TELECOMMUNICATION INDUSTRY USING MACHINE LEARNING

GODSON S MATEMU

ASIA PACIFIC UNIVERSITY OF TECHNOLOGY & INNOVATION (APU)

SCHOOL OF COMPUTING AND TECHNOLOGY

SEPTEMBER 2015

Abstract

In recent years, one of the biggest problems in the telecommunications industry has been client loss. Because of the high level of rivalry among businesses and the high cost of customer acquisition, keeping existing consumers is extremely valuable. Thus, preventing churn by establishing effective and accurate prediction models is critical. This study addresses messy data and class imbalance as the problems that influence the accuracy of machine learning models. The aim of this study is to predict customer churn in telecommunication datasets by comparing the accuracy of the machine learning models used to predict. The study analysed the relevant features by using correlation analysis and SMOTE was applied to reduce the imbalance problem. To evaluate the performance of the models' accuracy, kappa and AOC metrics were used and Random Forest outperformed other models

Contents

Introduction

Customer churn is defined as when customers terminate their relationship with a company. In the context of the telecommunications industry, customer churn is when customers cancel their services and switch to competitors (De et al., 2021). The telecommunication industry is plagued by high customer churn as competition within the industry is high. As a way to solve customer churn companies employ customer relationship management practices to strengthen and maintain relationships with current customers (Soltani et al., 2018).

The main cost of customer churn is that it hampers profit optimization initiatives set by telecommunication operators. As customer acquisition costs are higher in comparison to retaining customers. Customer acquisition costs are costs associated with obtaining and attracting new customers (Ganaie and Bhat 2021). These costs are higher because companies have to spend more resources on differentiating and marketing their products in a highly intense competitive environment**.**

High customer retention rates are of utmost importance to telecommunication operators. As the costs associated with retaining customers are lower than acquisition which translates into more cost savings for companies leads to higher profit maximization for telecommunication operators. For telecommunication companies to increase their customer retention rates, it is clear that they should analyze their customer churn rates to develop effective strategies to retain customers and counterbalance the effects of customer churn. Usage of machine classification algorithms can greatly assist companies in developing robust strategies to solve customer churn and in optimizing their customer management techniques (V.Anandi & Ramesh, 2021).

Machine learning has been highly used by researchers for prediction in different fields such as housing prediction, credit card frauds, biostatistics, and customer churn. Researchers developed decision trees, SVM, ensembles techniques, and neural networks that perform better than the statistical analysis techniques that were previously used (Bzdok et al., 2018). With this companies have adopted the use of machine learning in the prediction of customer churn.

Data associated with the predictive models are to be well processed for better accuracy, thus different methods of effective processing have been developed by the researchers in the need to improve the accuracy, one of the main problems with customer churn models is missing values and class imbalance (Picek et al., 2019). As building machine learning models on imbalanced classes and incomplete data leads to misleading and inaccurate predictions (Nguyen & Duong, 2021). In order to overcome these researchers have employed different mechanisms of handling missing data and balancing the datasets such as SMOTE, ROSE, and up-sampling or down-sampling techniques.

Problem statement

Telecommunication companies need accurate forecasts to assist them in developing effective customer relationship strategies to increase customer retention rates and revenue growth although the data available is messy and imbalanced which influences the accuracy of the predictive models.

Aim and objectives

This research report aims to design, tune and implement machine learning models to predict customer churn in telecommunication industries.

Objectives

The objectives of the study are

- To address the class imbalance problem by using sampling techniques to improve the prediction accuracy
- To implement SVM, Random Forest, and decision trees in predicting the customers who are likely to churn
- Evaluating and comparing the performance of the predictive models using different performance metrics.

Scope of the research

Orange Telecom's Churn Dataset is a collection of customer activity data (features) and a churn label indicating whether a client terminated their subscriptions. The dataset contains 3333 instances and 20 factors that machine learning models will utilize to predict churn.

Literature review

Feature selection.

Mohammad et al., (2019) compared the performance of the machine learning models with the recursive feature elimination technique which is a feature selection technique against three machine learning techniques such as logistic regression ANN and random forest, and the models which utilized the feature selection technique had the highest accuracy dataset

Asif Yaseen, (2021) employed PSO as a feature selection technique that would enable to use of the most important variable in each machine learning technique and he further applied 4 machine learning algorithms to compare its performance generally decision tree had the most accuracy.

(Jain et al., 2021) employed different hybrid and standalone machine learning techniques in the prediction of customer churn, the researcher employed feature engineering and selection techniques to the dataset. The resulting dataset was trained with the machine learning models and the Random Forest had the highest performance of 95% with all other models performing well.

Imbalance problem

(Idris et al., 2017) addressed the imbalance problem of the dataset by under-sampling the cell2cell and orange datasets using PSO. The balanced datasets are trained using modified AdaBoost, gradient boosting, and random forest. After evaluation and comparison, the modified AdaBoost outperformed other models. It had an AOC of 0.862, in comparison with random forest and gradient boosting which had 0.75 and 0.737 respectively

Pustokhina et al., (2021) presented a new technique of predicting customer churn in imbalanced datasets using deep learning and SMOTE, the researchers integrated the use of SMOTE and rain optimization algorithms (ROA) in handling imbalanced datasets and train the dataset in deep

learning algorithms (OWELM) the resulting accuracy proved that the proposed technique is competitive in dealing with imbalanced datasets as it had an accuracy of 0.944%

(Jain et al., 2020) predicted customer churn using the orange dataset with 3333 instances, The researchers used logistic regression and logit boost to predict customer churn. After training and testing the dataset, the result showed both models have an accuracy of around 85%, but this dataset had imbalanced distribution between the churns and non-churners It is proved as the kappa statistic was 0.11 and 0.09 on the two machine learning models.

Sergue, (2020) employed two strategies to reduce the effect of the imbalanced dataset. Random oversampling, as well as oversampling and under sampling utilizing SMOTE, are among these strategies. Then, to forecast and explain churn, random forest and logistic regression models are applied. The non-linear technique outperformed logistic regression, indicating that linear models have limitations in our application. Furthermore, in terms of the area under the curve measure, random oversampling performs better.

Nguyen & Duong, (2021) analyzed deep Belief Network (DBN) and SMOTE, two data resampling systems, and compared them to two cost-sensitive learning methods weighted loss and, focused loss. In terms of overall prediction accuracy in customer churn, the empirical data show that focused loss and weighted loss techniques outperform the other techniques.

tuning and model improvement.

Dalli, (2022) experimented with the impact of various hyperparameter setups in Neural networks. The analysis was done by using different monotonic activation functions pairings in the hidden layer, changing the batch size, and finally using different optimizers. From the analysis, all the parameters had an impact on the performance of the customer churn prediction

Lalwani et al., 2021 used K-fold cross-validation over the training set for hyperparameter tuning to avoid overfitting of models, which increased the predictive models' performance. Finally, the AUC curve and the confusion matrix were used to evaluate the test set findings. XG boost and

Adaboost classifier were found to have a maximum accuracy of 81.71 percent and 80.8 percent, respectively.

(Sabbeh, 2018) compared the performance of several machine learning techniques predicting churn using an orange dataset. Cross-validation and hyperparameter tuning are used in increasing the accuracy of the model. Finally, the results show ADA and random forest outperforms the other models with an accuracy of 96%

(Mishra & Reddy, 2017) used the orange dataset, and employed the ensembles techniques to the dataset to predict customer churn, the results showed random forest performs better than the other models with an accuracy of 91.66%

Summary

citations	**dataset**	**Methodology**	**Best performing model**	**Performance Accuracy**
(Dalli, A. 2022).	Orange dataset	Model improvement using hyperparameters	Neural networks	Activation functions 86.8% Batch size 85.75% Optimizer 86.45%
(Mohammad et al., 2019)	7043 21	Implemented recursive feature elimination	Random forest	98.4%
(Sabbeh, 2018)	Orange dataset	Cross-validation and parameters tuning	Random forest SVM	96% 94%
Lalwani et al., 2021	7000 21	Feature selection using gravitational algorithms and K fold validation to tune hyperparameters	Support vector machine Decision trees Random forest	79.14 % 80.14% 78.04%
(Mishra & Reddy, 2017)	Orange dataset		Random forest Decision tree SVM	91.66% 90.97% 90.12%

(Nguyen & Duong, 2021)	USI dataset 3333 21	Imbalance handling using SMOTE and DBN	Logistic regression XG boost	SMOTE 75.68% 86.66% DBN 64.31% 87.14%
(Pustokhina et al., 2021	Orange dataset	Handling imbalance using ISMOTE OWELM model	Deep learning	94%
(Afendi et al., 2019)		Feature selection using multicollinearity checking and recursive feature elimination	Logistic regression Extreme random forest	90.55% 95.05%
(Sergue, 2020)	B2B dataset	Compared resampling techniques to handle imbalance.	Logistic regression Random forest	0.65 0.72 0.69 0.75
(Jain et al., 2020)	Orange dataset	No balancing	Logistic	85%
(Jain et al., 2021)	Orange dataset	Implemented feature selection techniques	Random forest SVM	95% 89%
(Asif Yaseen, 2021)	Orange dataset	Feature selection using PSO	Decision tree	94.56%

Summary

The studies have utilized different methods in feature selection, handling imbalanced data using SMOTE, improving accuracy using hyperparameter tuning and they have effectively increased the accuracy of the models. However, some studies have not implemented both techniques, this study implements feature selection, balancing the data using SMOTE, and finally hyperparameters tuning.

dataset.

The data used in this study is the orange Telcom dataset consists of 2666 instances and 20 variables. The Orange Telcom dataset has been used by many researchers (Khalid et al., 2021), (Pejić Bach et al., 2021), and (Idris et al., 2017). The data is available on Kaggle. The dataset consists of customer information such as State, account length, the total charges, calls, and minutes of each customer all of these are used as dependent variables that are used to determine the churn characteristic of each customer.

https://www.kaggle.com/mnassrib/telecom-churn-datasets?select=churn-bigml-80.csv

```
> str(df)
'data.frame':   2666 obs. of  20 variables:
 $ State                 : Factor w/ 51 levels "AK","AL","AR",..: 17 36 32 36 37 2 20 25 50 40 ...
 $ Account.length        : int  128 107 137 84 75 118 121 147 141 74 ...
 $ Area.code             : int  415 415 415 408 415 510 510 415 415 415 ...
 $ International.plan    : Factor w/ 2 levels "No","Yes": 1 1 1 2 2 2 1 2 2 1 ...
 $ Voice.mail.plan       : Factor w/ 2 levels "No","Yes": 2 2 1 1 1 1 2 1 2 1 ...
 $ Number.vmail.messages : int  25 26 0 0 0 0 24 0 37 0 ...
 $ Total.day.minutes     : num  265 162 243 299 167 ...
 $ Total.day.calls       : int  110 123 114 71 113 98 88 79 84 127 ...
 $ Total.day.charge      : num  45.1 27.5 41.4 50.9 28.3 ...
 $ Total.eve.minutes     : num  197.4 195.5 121.2 61.9 148.3 ...
 $ Total.eve.calls       : int  99 103 110 88 122 101 108 94 111 148 ...
 $ Total.eve.charge      : num  16.78 16.62 10.3 5.26 12.61 ...
 $ Total.night.minutes   : num  245 254 163 197 187 ...
 $ Total.night.calls     : int  91 103 104 89 121 118 118 96 97 94 ...
 $ Total.night.charge    : num  11.01 11.45 7.32 8.86 8.41 ...
 $ Total.intl.minutes    : num  10 13.7 12.2 6.6 10.1 6.3 7.5 7.1 11.2 9.1 ...
 $ Total.intl.calls      : int  3 3 5 7 3 6 7 6 5 5 ...
 $ Total.intl.charge     : num  2.7 3.7 3.29 1.78 2.73 1.7 2.03 1.92 3.02 2.46 ...
 $ Customer.service.calls: int  1 1 0 2 3 0 3 0 0 0 ...
 $ Churn                 : Factor w/ 2 levels "False","True": 1 1 1 1 1 1 1 1 1 1 ...
```

From the dataset, three supervised machine learning models will be used to predict customer churn. The dataset will be processed then trained and validated, finally a comparative study will be done to analyze the best model for customer churn prediction

Support vector machines.

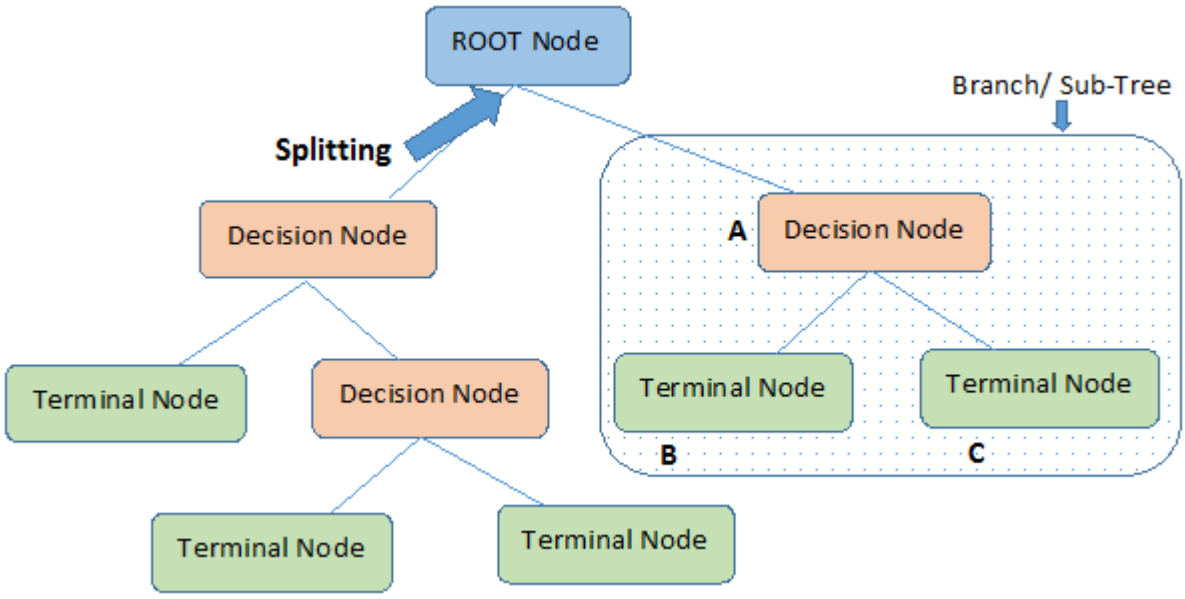
Is a model which involves a surface that draws a line between data points displayed in a multidimensional space that represents feature values. SVM creates a hyperplane that effectively divides the datapoint plotted into homogenous groups. SVM is a powerful model as it integrates the learning knowledge from the KNN and regression to build a hyperplane(Chauhan et al., 2018).

SVM uses a kernel, which is a mathematical function that transforms the data to a format that can be processed. This research will compare the three most common kernels Linear, Sigmoid and polynomial, and evaluate the performance of the best kernel.

This research uses SVM as it handles high dimensionality datasets and also it uses kernel functions to predict well the nonlinear parameters (Xiahou & Harada, 2022). Moreover, SVM can reduce the overfitting problem.

Decision trees.

A decision tree is a supervised machine learning model that describes the relationships between features and potential outcomes using a tree structure. Decision trees support both continuous and categorical variables, and they can predict classification and regression problems (Izza et al., 2020)



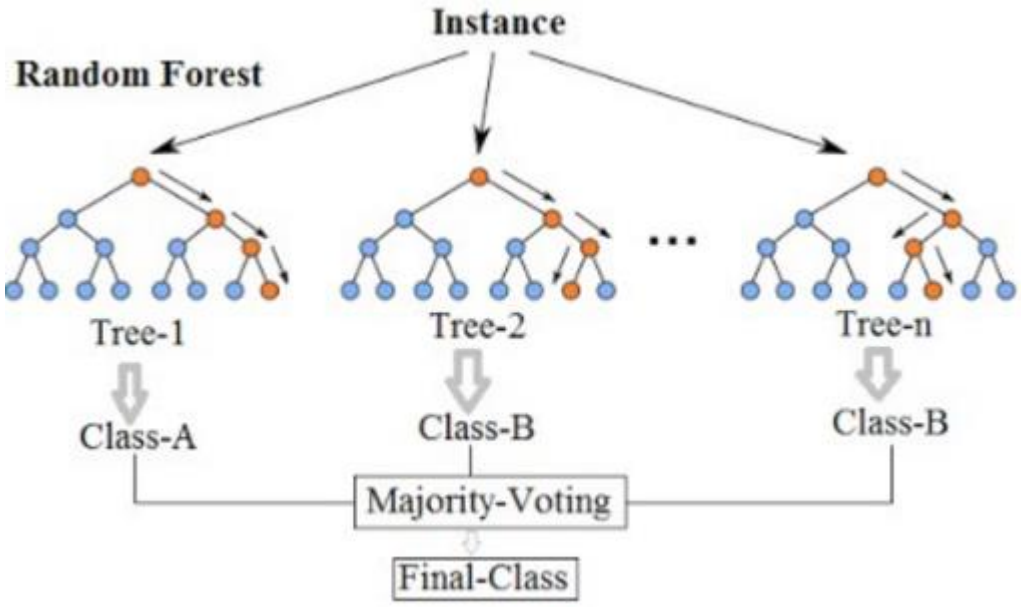
The root nodes are the nodes that represent the complete sample. It is then divided into two or more homogenous groups using the decision criterion. The decision nodes are the nodes that can split further, while the terminal nodes cannot.

Decision trees are used in this research because their output is in human-readable form as a result, it's well-suited to applications where the classification method needs to be examined for

business activities like churn control. It can further aid to explain the influence of each variable on customer churn

Random forest.

A random forest is generally a group of decision trees. It is a supervised machine model that involves bagging by randomly selecting variables and then generating different decision trees. The random forest is more complex than the decision trees because it generates a large number of decision trees. The accuracy of the random forest depends on the number of trees generated,



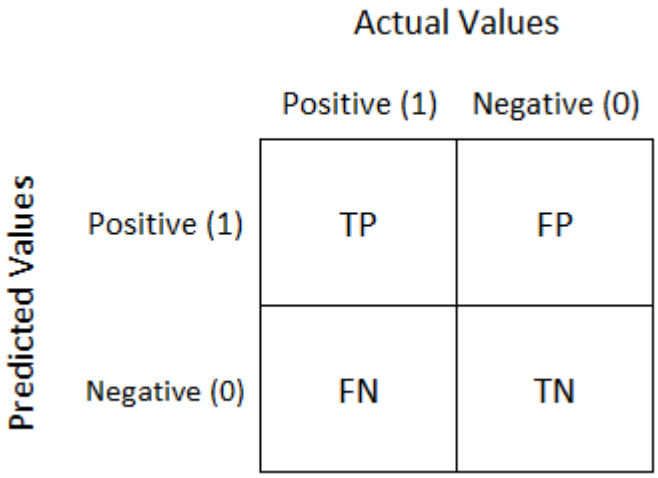
Decision trees, on the other hand, use all of the predictor factors. A random forest is a collection of decision trees that selects a set of parameters at random and generates a decision tree for each set.

Random forest is used in this research because it provides better accuracy than decision trees and fast, and also it works well with unbalanced and data with missing values (Rigatti, 2017).

Performance evaluation.

This research will use several performance metrics to evaluate and compare the machine models created. The models implemented will create a confusion matrix which is the distribution of the

actual and predicted values of the target variables. Further, the confusion matrix will be used to calculate more performance metrics.



Whereas

True positive (TP) means the people who churn and are accurately predicted to churn

True negative (TN) means the people who are non-churners and are accurately predicted as non-churners

False Negative (FN) means those who churn but are wrongly predicted as non-churners

False-positive (FP) means the people who are non-churners but are wrongly predicted as churners

Accuracy is the proportion the of the correct predictions to the total number of predictions

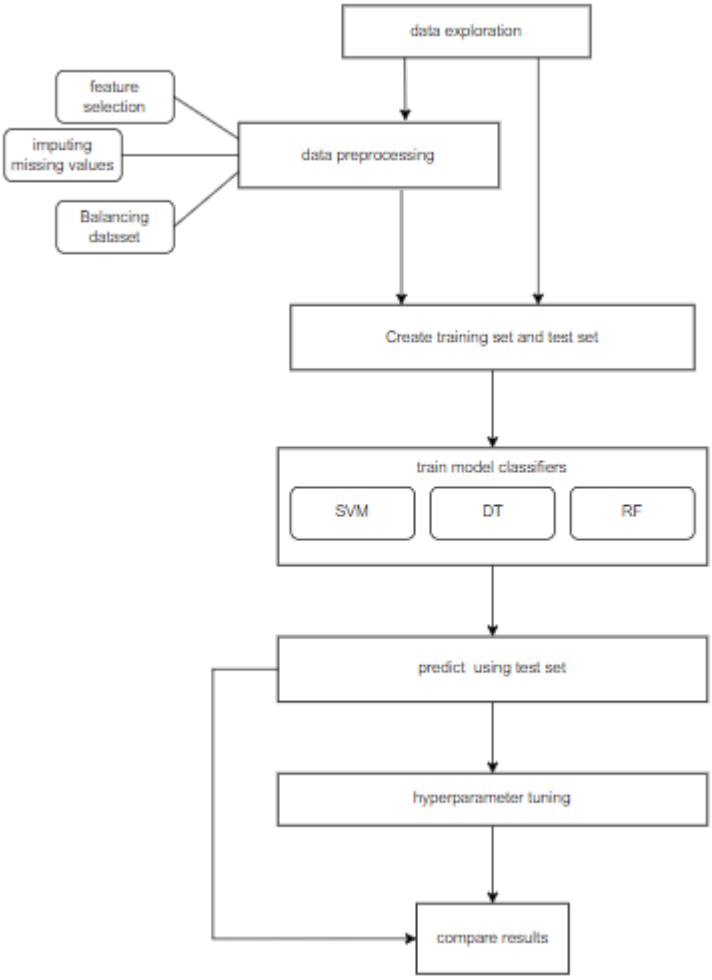
Sensitivity is the total number of clients accurately forecasted as a true positive.

Specificity is the total number of clients accurately forecasted as a true negative.

Kappa is a measure of accuracy that takes into consideration the distribution of the target variables.

ROC is a plot of the true positive rate against the false-positive rate.

The study will follow this flow chart, in which 1 dataset will be preprocessed using different features selection techniques and the dataset will be balanced with SMOTE and then trained on the model, while the other dataset will not be processed and trained directly to the model. Finally, the models with processed and unprocessed data will be compared.



Data exploration

Data exploration is the first phase in predictive modeling, and it involves exploring and visualizing data to find insights right away or to indicate regions or patterns to investigate further. The summary of the dataset shows the numeric variables are normally distributed as the mean and median are almost the same across all the continuous variables. Furthermore, in the categorical variables, State is noted to have more than 5 levels, this variable will be removed as machine learning models cannot learn from those variables also there is an imbalance in the target variables as there is a high number of non-churners than churners

```
     State         Account.length    Area.code     International.plan Voice.mail.plan Number.vmail.messages
 WV     :  88   Min.   :  1.0   Min.   :408.0   No :2396           No :1933        Min.   : 0.000
 MN     :  70   1st Qu.: 73.0   1st Qu.:408.0   Yes: 270           Yes: 733        1st Qu.: 0.000
 NY     :  68   Median :100.0   Median :415.0                                      Median : 0.000
 VA     :  67   Mean   :100.6   Mean   :437.4                                      Mean   : 8.022
 AL     :  66   3rd Qu.:127.0   3rd Qu.:510.0                                      3rd Qu.:19.000
 OH     :  66   Max.   :243.0   Max.   :510.0                                      Max.   :50.000
 (Other):2241
 Total.day.minutes Total.day.calls Total.day.charge Total.eve.minutes Total.eve.calls Total.eve.charge
 Min.   :  0.0     Min.   :  0.0   Min.   : 0.00    Min.   :  0.0     Min.   :  0     Min.   : 0.00
 1st Qu.:143.4     1st Qu.: 87.0   1st Qu.:24.38    1st Qu.:165.3     1st Qu.: 87     1st Qu.:14.05
 Median :179.9     Median :101.0   Median :30.59    Median :200.9     Median :100     Median :17.08
 Mean   :179.5     Mean   :100.3   Mean   :30.51    Mean   :200.4     Mean   :100     Mean   :17.03
 3rd Qu.:215.9     3rd Qu.:114.0   3rd Qu.:36.70    3rd Qu.:235.1     3rd Qu.:114     3rd Qu.:19.98
 Max.   :350.8     Max.   :160.0   Max.   :59.64    Max.   :363.7     Max.   :170     Max.   :30.91

 Total.night.minutes Total.night.calls Total.night.charge Total.intl.minutes Total.intl.calls
 Min.   : 43.7       Min.   : 33.0     Min.   : 1.970     Min.   : 0.00      Min.   : 0.000
 1st Qu.:166.9       1st Qu.: 87.0     1st Qu.: 7.513     1st Qu.: 8.50      1st Qu.: 3.000
 Median :201.2       Median :100.0     Median : 9.050     Median :10.20      Median : 4.000
 Mean   :201.2       Mean   :100.1     Mean   : 9.053     Mean   :10.24      Mean   : 4.467
 3rd Qu.:236.5       3rd Qu.:113.0     3rd Qu.:10.640     3rd Qu.:12.10      3rd Qu.: 6.000
 Max.   :395.0       Max.   :166.0     Max.   :17.770     Max.   :20.00      Max.   :20.000

 Total.intl.charge Customer.service.calls   Churn
 Min.   :0.000     Min.   :0.000          False:2278
 1st Qu.:2.300     1st Qu.:1.000          True : 388
 Median :2.750     Median :1.000
 Mean   :2.764     Mean   :1.563
 3rd Qu.:3.270     3rd Qu.:2.000
 Max.   :5.400     Max.   :9.000

>
```

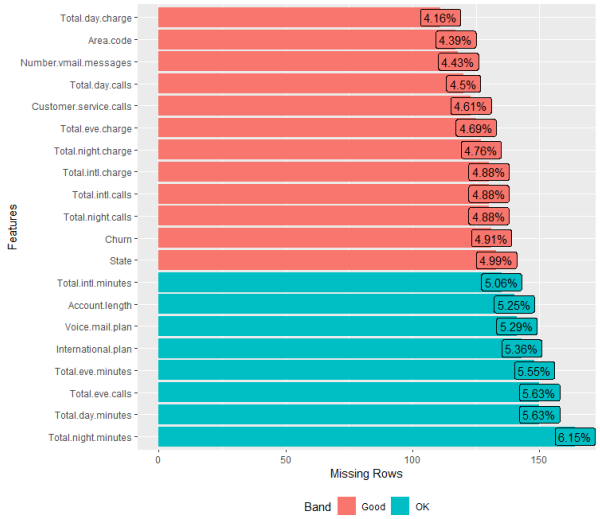
Data preprocessing.

Data pre-processing is one of the most important steps of information discovery operations. It necessitates several stages, including data processing and data reduction. Machine learning efficiency and accuracy will be reduced if the data is turned into low-quality data. Thus, by executing sufficient data preprocessing steps and selecting appropriate machine learning techniques, the acquired data can be accurately examined (Gonzalez Zelaya, 2019).

Orange telecom dataset is pre-processed by addressing missing data and encoding categorical variables using one of two methods: "label encoding" or "one-hot encoding." Furthermore, using multicollinearity checking to remove the features that aren't relevant to the dependent variable. Following that, some EDAs are used to gain more understanding of the data.

Handling missing values.

Because of data pattern inconsistencies and knowledge loss, missing data has an impact on statistical analysis. It is critical to understand the percent of the distribution of missing values in each variable when dealing with missing values. According to (Wadikar, 2020), variables with a missing percentage of more than 30% should be completely deleted.



As a result, missing values are handled using a simple imputation technique. Incomplete continuous and categorical variables with varying median, mean, and mode are imputed using this technique. The mode is used to impute categorical missing values from the datasets, whereas the median is used to impute continuous missing values.

```
                State         Account.length              Area.code     International.plan
                    0                      0                      0                      0
     Voice.mail.plan  Number.vmail.messages       Total.day.minutes         Total.day.calls
                    0                      0                      0                      0
     Total.day.charge       Total.eve.minutes         Total.eve.calls       Total.eve.charge
                    0                      0                      0                      0
 Total.night.minutes       Total.night.calls     Total.night.charge     Total.intl.minutes
                    0                      0                      0                      0
    Total.intl.calls      Total.intl.charge Customer.service.calls                  Churn
                    0                      0                      0                      0
```

Data transformation

According to (Mohammad et al., 2019), Data transformation strategies can significantly improve the overall performance of churn prediction. On datasets, two alternative data transformation techniques are used: one-hot encoding and label encoding.

```
  State Account.length Area.code International.plan Voice.mail.plan Number.vmail.messages Total.day.minutes
1    KS            128       415                 No             Yes                    25             265.1
2    OH            107       415                 No             Yes                    26             161.6
3    NJ            137       415                 No              No                     0             243.4
4    OH             84       408                Yes              No                     0             299.4
5    OK             75       415                Yes              No                     0             166.7
6    AL            118       510                Yes              No                     0             223.4
  Total.day.calls Total.day.charge Total.eve.minutes Total.eve.calls Total.eve.charge Total.night.minutes
1             110            45.07             197.4              99            16.78               244.7
2             123            27.47             195.5             103            16.62               254.4
3             114            41.38             121.2             110            10.30               162.6
4              71            50.90              61.9              88             5.26               196.9
5             113            28.34             148.3             122            12.61               186.9
6              98            37.98             220.6             101            18.75               203.9
  Total.night.calls Total.night.charge Total.intl.minutes Total.intl.calls Total.intl.charge
1                91              11.01                10.0                3              2.70
2               103              11.45                13.7                3              3.70
3               104               7.32                12.2                5              3.29
4                89               8.86                 6.6                7              1.78
5               121               8.41                10.1                3              2.73
6               118               9.18                 6.3                6              1.70
  Customer.service.calls Churn
1                      1 False
2                      1 False
3                      0 False
4                      2 False
5                      3 False
6                      0 False
```

Label encoding.

Is a technique that involves transforming categorical variables to numerical/nominal variables, usually this is done to variables with two levels such as gender, or yes and no outputs. The dataset contains three categorical attributes with two levels which are international plan, voice mail plan, and churn in which yes is replaced by 1 and no is replaced by 0.

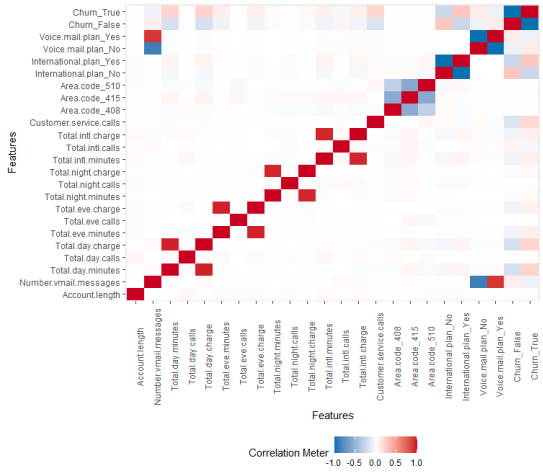
One hot encoding.

is a used binary encoding method that employs a 1 to indicate the presence of a group and a 0 to indicate its absence. In the nominal attribute comprising M categories, it converts each class into more of an M-dimensional vector with an element of (0 or 1). The raw data is not harmed by this procedure. However, several One hot encoding OHE transform difficulties, such as "sparse data" and "high dimensionality," would increase the computational time of machine learning.

```
  Account.length Area.code.1 Area.code.2 Area.code.3 International.plan.0 International.plan.1
1            128           0           1           0                    1                    0
2            107           0           1           0                    1                    0
3            137           0           1           0                    1                    0
4             84           1           0           0                    1                    0
5             75           0           1           0                    0                    1
6            118           0           0           1                    0                    1
  Voice.mail.plan.0 Voice.mail.plan.1 Total.day.calls Total.day.charge Total.eve.charge Total.night.calls
1                 0                 1             110            45.07            16.78                91
2                 0                 1             123            27.47            16.62               103
3                 1                 0             114            41.38            10.30               104
4                 1                 0              71            50.90             5.26                89
5                 1                 0             113            28.34            12.61               121
6                 1                 0              98            37.98            18.75               118
  Total.night.charge Total.intl.minutes Total.intl.calls Total.intl.charge Customer.service.calls Churn
1              11.01                10.0                3              2.70                       1     0
2              11.45                13.7                3              3.70                       1     0
3               7.32                12.2                5              3.29                       0     0
4               8.86                 6.6                7              1.78                       2     0
5               8.41                10.1                3              2.73                       3     0
6               9.18                 6.3                6              2.75                       0     0
> |
```

Feature selection.

It aims to pick a selection of important characteristics to employ in classification or to approximate regression target variables. The Correlation Matrix is utilized in this study to visualize how each variable correlates with the others. And the variables with high multicollinearity with each other are dropped (Senawi et al., 2017), and also variables with low correlation with the target variables are dropped.



From fig above, some variables had high collinearity between each other and they were being dropped.

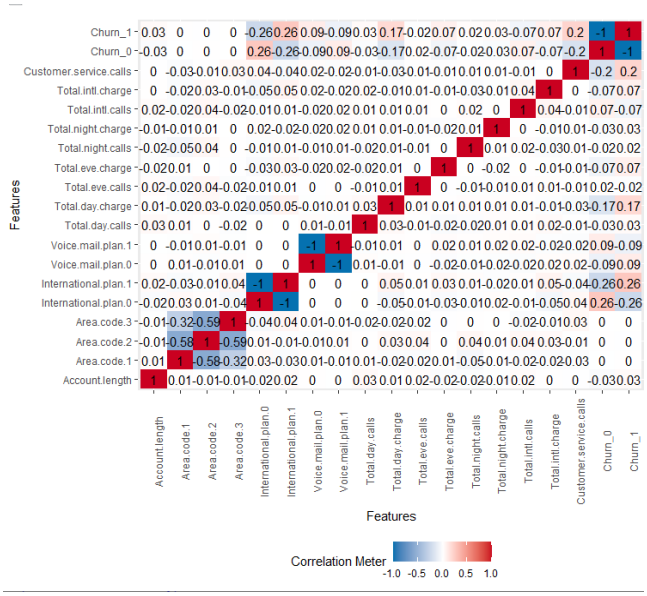
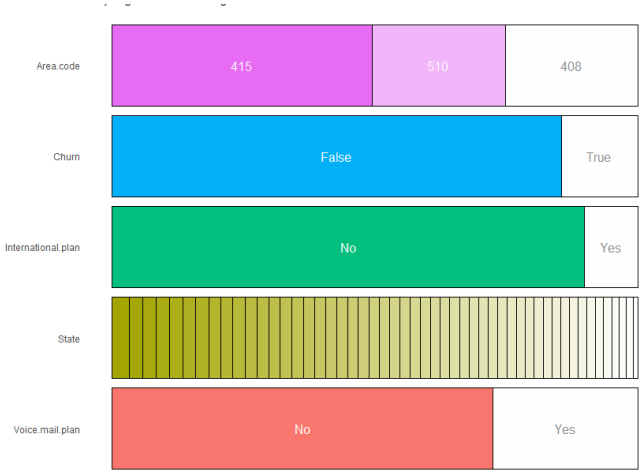


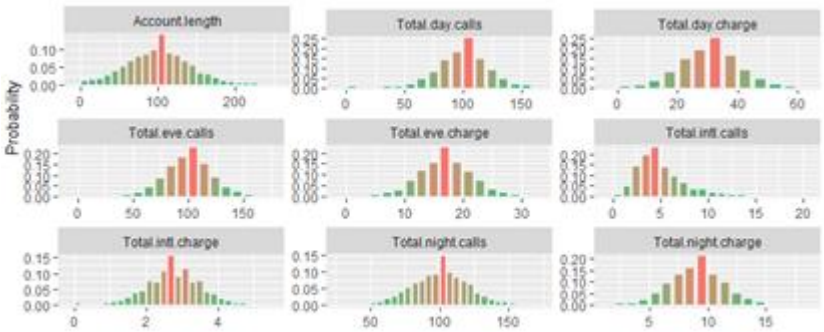
Fig 2 below shows the collinearity between variables and no variables that have high a correlation with each other.

Explanatory data analysis

Finally, the preprocessed data set is visualized to understand the class distribution of the categorical attributes.



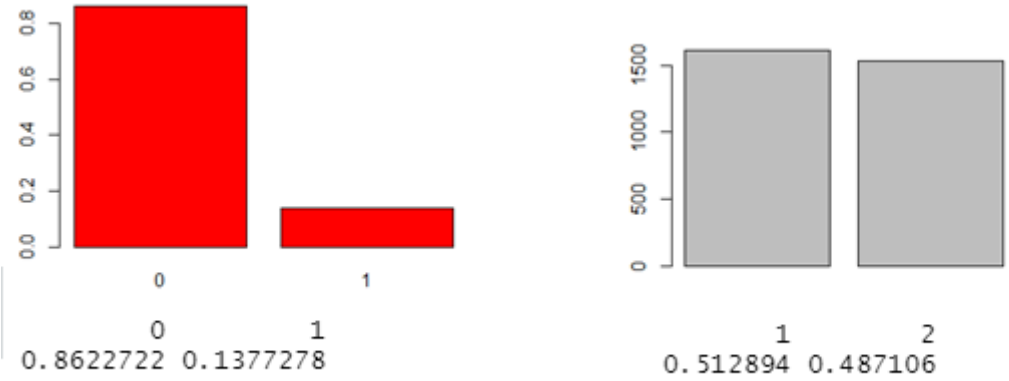
The proportion of the Churn variable is larger in the No category compared to the Yes category, as can be seen in the visualization above. This problem might affect the model accuracy, therefore, relevant techniques should be adapted



From fig ---- above, the numerical distribution of the variables appears to be normally distributed.

Data balancing

Data balancing is quite a problem in this dataset, we will balance the dataset using SMOTE, which the method involves generating a new observation that exists between observations of the minority class it is a better balancing method compared to the random oversampling technique.



From the figures above, fig one presents the proportion of the churned classes on the training set, and fig two shows the balanced proportion of the training set using SMOTE technique.

Data splitting

The dataset obtained from the preprocessing is then split into train and test sets in a ratio of 70% and 30% respectively

Each time the loop runs the data is split randomly and models are created from the training set and are tested its accuracy in the test set.

```
set.seed(123)
split = sample.split(data$Churn, SplitRatio = 0.7)
train_set = subset(data, split == TRUE)
test_set = subset(data, split == FALSE)
```

Model Implementation.

Support vector machine.

In SVM experimentation offered by the e7101 package, three models were generated in which each model used a different kernel, and the hyperplanes generated led to different prediction metrics. The models were trained validated and compared using the same dataset, The best model was then evaluated and further tuned

Sample code.

```
svm_rbf <- svm(class~., data = train_set)
svm_linear = svm (class~., data = train_set, kernel = "linear")
svm_sigmoid = svm (class~., data = train_set, kernel = "sigmoid")
svm_polynomial = svm (class~., data = train_set, kernel = "poly")
```

	Accuracy		sensitivity		sensitivity		Kappa
	Train_set	Test_set	Train set	Test set	Train set	Test set	Test set
Model 1	0.9223	0.896	0.9354	0.9045	0.9084	0.8870	0.8318
Model 2	0.7752	0.7697	0.7666	0.7627	0.7842	0.7771	0.52
Model 3	0.8997	0.867	0.9063	0.8423	0.8927	0.8931	0.799

Table 1, summarizes the prediction measures of the three models

According to (Migut, 2020) the accuracy measure may be used to select the best model (the model with the highest accuracy value), but it does not determine whether the model is of good quality. Thus, the kappa coefficient is used because it takes into account the distribution of the target variables classes in the dataset. From the table above **model 1** outperforms all the other models in terms of accuracy, kappa coefficient, sensitivity, and specificity. The model will be further evaluated.

Model evaluation.

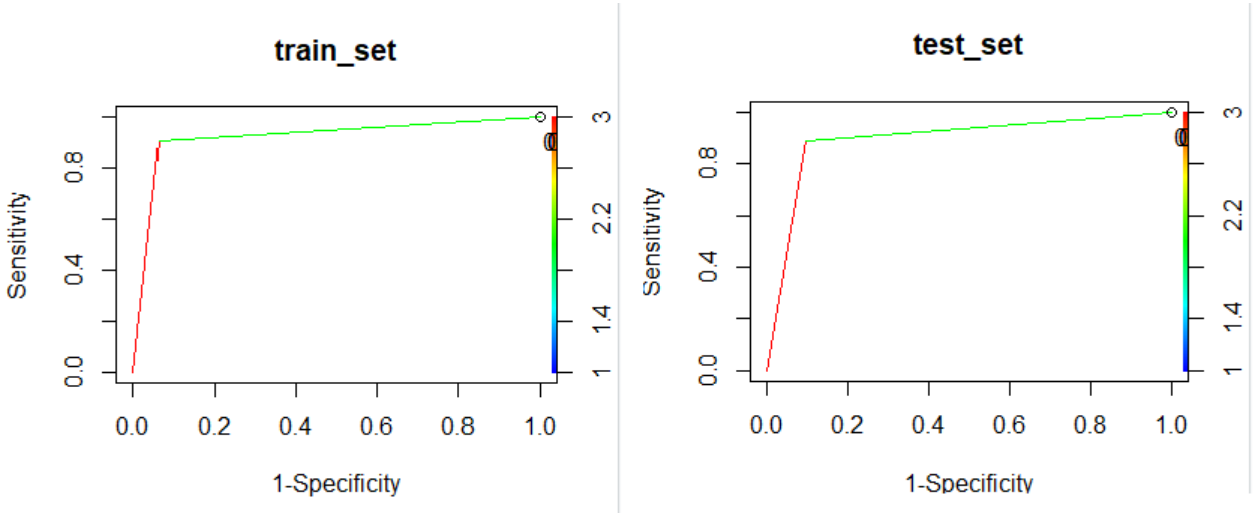
```
predd   1   2
    1 625  73
    2  64 589
```

From the confusion matrix, 625 of the non-churners were correctly predicted, furthermore, 589 instances of churners were correctly classified. However, there was a significant misclassification in which 73 churners were wrongly predicted as non-churners, and 64 of the non-churners were wrongly predicted as churners.

The model also derives the 90.45% sensitivity value which is the percentage of accurately predicting the churners, In Customer churn prediction the main aim is to predict the disloyal customers so the higher the sensitivity value the better is to predict the model.

88.7% specificity value describes the percentage in which the non-churners are accurately predicted.

Finally, the ROC curve is plotted which is the measure of the prediction power of the used model in perfectly separating churned customers from non-churned customers, from the model The AOC is 0.896 for the test set and 0.922 in the test set, no overfitting problem is noted from the ROC

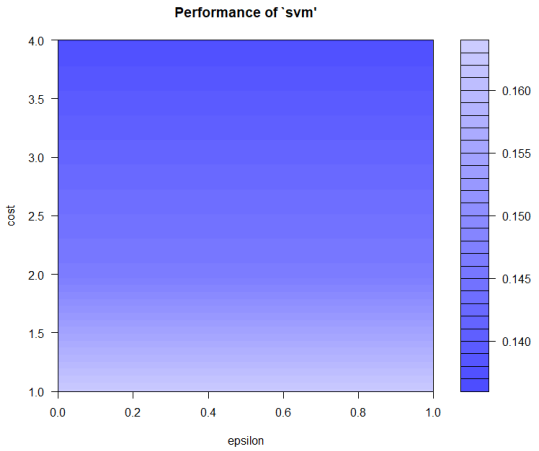


Model Improvement.

In SVM, two hyperparameters are used to increase the efficiency and prediction accuracy of the SVM models the hyperparameters are tuned and the parameters with the lowest error are used to build the best model. These are the hyperparameters in the SVM.

- Cost
- Epsilon

The grid search technique is implemented to search for the best hyperparameters in the SVM model.



The model was then built using the best parameters, the parameters with the lowest error rate and dispersion were used to build the best parameters. From the grid search technique, the parameters with the lowest errors are cost = 4 and epsilon 0, the parameters are further tuned and the minimum error was attained when cost = 8 and epsilon = 0

```
svm_best <- svm (class~., data = train_set, epsilon = 0, cost = 4)
```

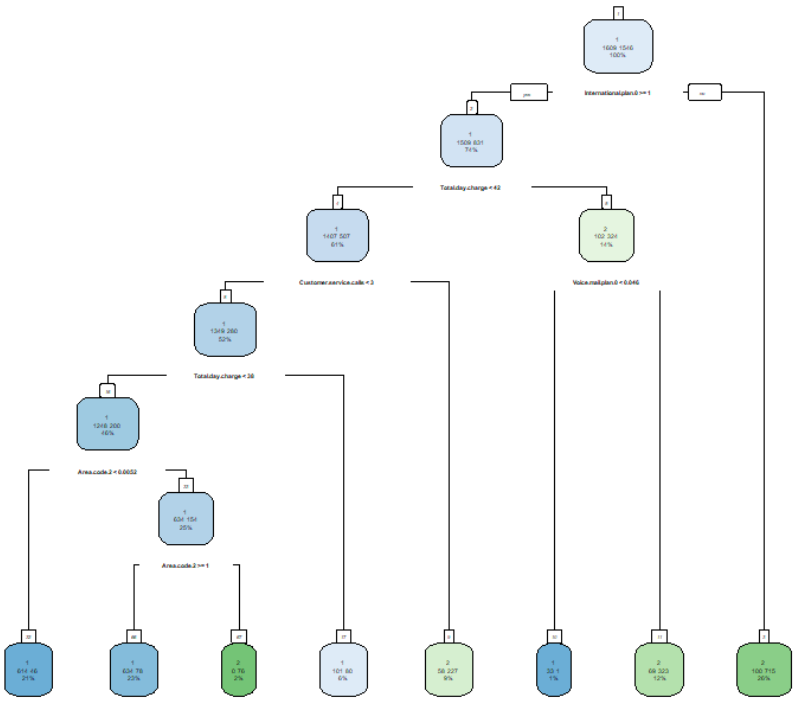
Metrics	Accuracy	Specificity	Sensitivity	Kappa	AOC
Best model	0.9134	0.9215	0.9057	0.868	0.8268
Best model 2	0.9235	0.9160	0.9305	0.8468	0.923

Decision tree.

In a decision tree, two models have been generated in which one model used a split with the entropy index to generate the decision tree and the other used the split with the Gini index, the resulting model both had the same prediction metrics.

Sample code

```
tree = rpart(Churn~ ., data=up_train)
tree
rpart.plot(tree, extra = 101, nn = TRUE)
```



From the tree above, the root node splits first using the international plan variable, the ones without the plan are churned, the decision nodes are subdivided 5 times more, and the total terminal nodes acquired are 8 nodes with different proportions of churners and non-churners

Model evaluation.

```
model5   1   2
     1 595  85
     2  94 577
```

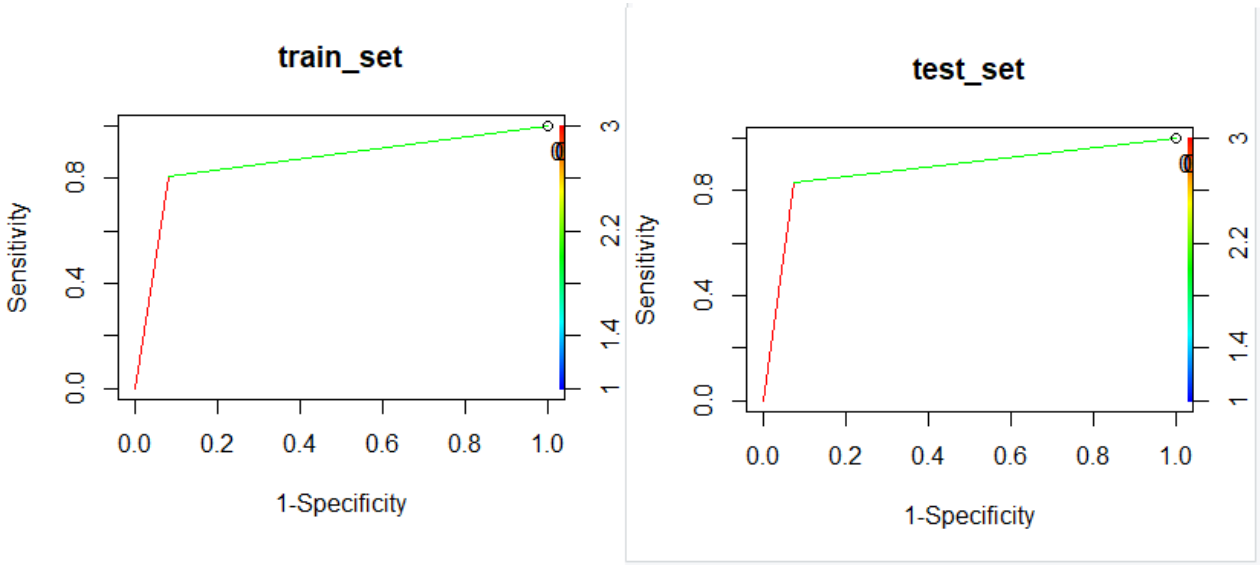
	Accuracy		Sensitivity		Specificity		Kappa
	Train set	Test set	Train set	Test set	Train set	Test set	Test set
Model 1	0.8659	0.8782	0.9193	0.9247	0.8097	0.8290	0.735

From the model, 595 of the non-churners were correctly predicted, furthermore, 577 instances of churners were correctly classified. However, there was a significant misclassification in which 85 churners were wrongly predicted as non-churners, and 94 of the non-churners were wrongly predicted as churners.

The model also derives an accuracy of 87.82 % in the test set which is higher than that of the training set, furthermore in predicting the probability of churners and non-churners the model accuracy was 92.47%, and 82.90% respectively. The model shows the distribution

Also, the kappa coefficient was quite good.

From the results, the test set has higher accuracy than the training set, this shows signs of overfitting of a model, the ROC curve is plotted and the AOC of the training set is 0.864 and the test is 0.877 this shows there is slight overfitting

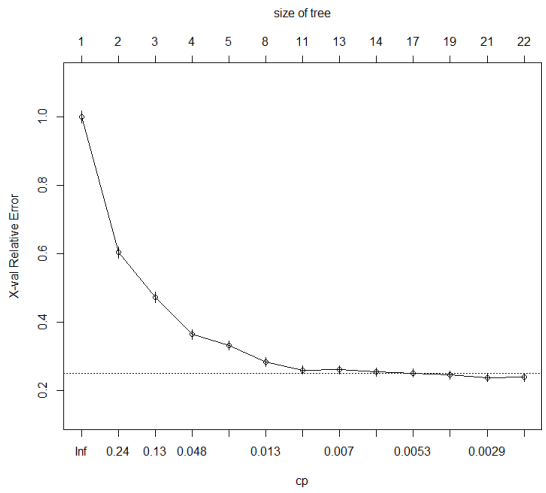


Model improvement.

In comparison to the SVM model, this accuracy number is poor. As a result, we may apply the pruning procedure to the model we just generated.

Pruning is a step in the process of creating a decision tree. Some nodes in a decision tree are outliers or are the outcome of data noise. Pruning a decision tree can help to eliminate data noise and outliers in the initial decision tree, resulting in improved data classification accuracy. The pruning algorithm has its own set of value requirements, which are as follows:

- Min split.
- Min bucket.
- Complexity parameter.



The model accuracy has improved by altering the complexity parameter, although changing the cp values increases the accuracy but also the number of trees increases do it is a trade-off between the accuracy and the complexity of the model.

Therefore cp = 0.0029 min split = 0.2 and min bucket = 20

```
tuned = rpart(class ~ ., data=train_set, method="class", minsplit = 0.2, minbucket = 20, cp = 0.0029)
```

Metrics	Accuracy	Specificity	Sensitivity	Kappa	AOC
Tuned model	0.9012	0.8901	0.9117	0.8021	0.901

The decision tree model has improved the accuracy from 87.82% to 90.21% although the accuracy of predicting the churn of customers reduced. From 92.47% to 91.17%. Generally, the predictive accuracy of the tuned model is better but predicting the customers who are likely to churn the original model should be considered.

Random forest.

Lastly, the dataset is trained by the random forest and the generated model is evaluated and tuned.

Sample code.

```
set.seed(345)
Randomf <- randomForest(class~.,data = train_set )
```

```
Call:
 randomForest(formula = class ~ ., data = train_set)
               Type of random forest: classification
                     Number of trees: 500
No. of variables tried at each split: 4

        OOB estimate of  error rate: 6.21%
Confusion matrix:
     1    2 class.error
1 1522   87  0.05407085
2  109 1437  0.07050453
```

From the model, 4 variables were used in each split, with 500 trees bootstrapped, and finally the OOB error rate of 6.21% which is the average error of the variables that are not in the bootstrapped sample.

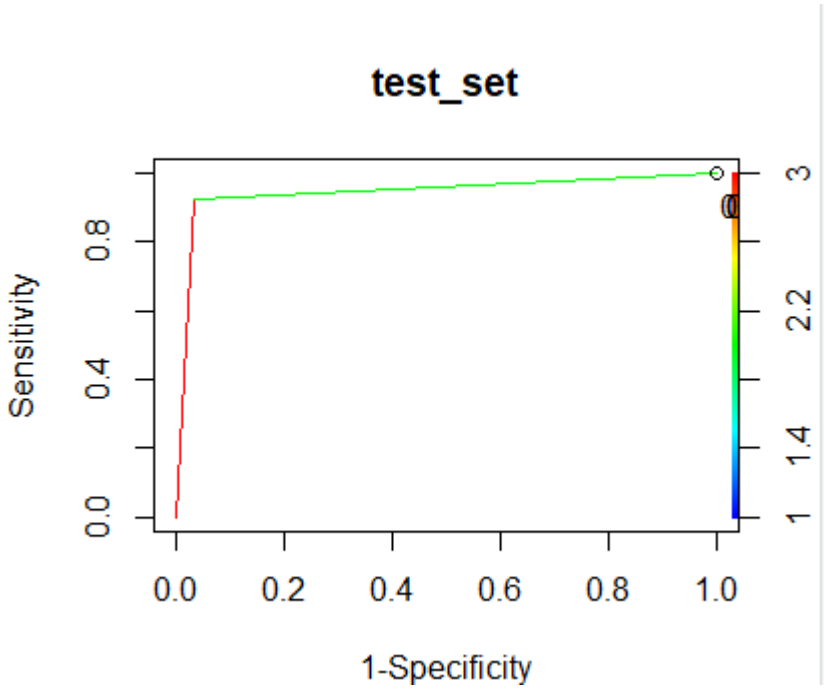
Model evaluation.

```
          Reference
Prediction   1   2
         1 654  46
         2  35 616
```

	Accuracy		Sensitivity		Specificity		Kappa
	Train set	Test set	Train set	Test set	Train set	Test set	Test set
Randomf	1	0.9458	1	0.9653	1	0.9252	0.8913

From the model, 654 of the non-churners were correctly predicted, furthermore, 616 instances of churners were correctly classified. However, there was a significant misclassification in which 46 churners were wrongly predicted as non-churners, and 35 of the non-churners were wrongly predicted as churners.

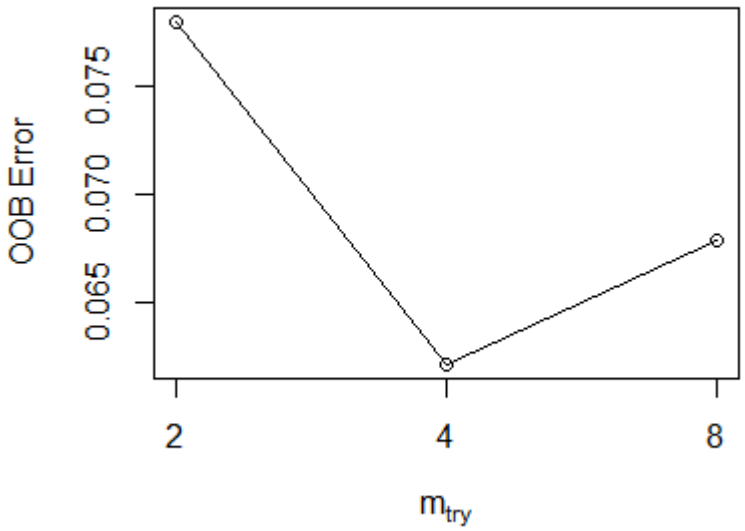
The model has an accuracy of 94.58%. Furthermore, the sensitivity and specificity of the model are 96.53% and 92.52% respectively. Also, the model had an AOC of 94.5% which shows how good random forest is in predicting customer churn.



Model improvement.

In Random Forest the model is improved by tuning the hyperparameters and the ones with the lowest OOB are used to build the best model. From the random forest model, two hyperparameters can be tuned to get the best model

- Ntreetry is the number of decision trees
- Mtry is the number of variables used in the random forest method



From the plot, the OOB error is minimum when the number of decision trees used is 800 and the number of variables is 4 then the best model is plotted from the tuned hyperparameters

```
randomf1 <- randomForest(class~.,data = train_set,
                         ntreeTry = 800,
                         mtry=2,
                         importance = TRUE,
                         proximity = TRUE)
```

	accuracy	specificity	sensitivity	kappa	AOC
Tuned model	0.9473	0.9282	0.9653	0.8943	0.947

The model prediction accuracy increased from 0.94 to 9473, the accuracy in predicting customers who churn did not improve but the specificity improved from 0.9252 to 0.9282, generally the tuned model accuracy is better than the previous model

Discussion of the results.

After evaluation and tuning of the models, the models with the best results from each machine learning algorithm are chosen and compared to each other. The table below is the summary of all the best models in each machine learning method with a different performance metric.

	accuracy	sensitivity	specificity	AOC
SVM	0.896	0.9045	0.8870	0.896
Tuned SVM	0.9235	0.9305	0.9160	0.923
DT	0.8782	0.9247	0.8290	0.877
Tuned DT	0.9012	0.9117	0.8901	0.901
RF	0.9458	0.9653	0.9252	0.945
Tuned RF	0.947	0.9653	0.9282	0.947

From the table above, the tuned random forest outperforms the other machine learning techniques in each and every performance metric from the accuracy, sensitivity, specificity, and AOC. The accuracy of the tuned random forest is 94.7 which is 6% more than the weakest model. The tuned random forest can accurately predict the people who churn by 96.53, Moreover, the normal random forest (without tuning) performs second best. Generally, the Random Forest is the best predictor in this study. SVM follows next with an accuracy of 92.35% and finally, the decision tree is the weakest with a prediction accuracy of 90%. One peculiar observation is that the standard decision trees outperform the standard SVM in accurately predicting the churners

Researcher	dataset	Best prediction method	Accuracy
(Dalli, A. 2022).	Orange dataset 3333instances 21 variables	Neural networks	86.8
(Sabbeh, 2018)		Random forest	96%
(Jain et al., 2020)		Logistic	85%
(Jain et al., 2021)		Random forest	95%
(Asif Yaseen, 2021)		Decision tree	94.56%
This study		Random forest	94.7
(Mishra & Reddy, 2017)		Random forest	91.66
Lalwani et al., 2021	7000 21	Decision trees	80.14%
(Asif Yaseen, 2021)	4250 19	Decision tree	94.56%
(Pustokhina et al., 2021	3333 21	Deep learning	94%

From the table below, this study is compared with the works of other researchers who predicted customer churn using the same orange dataset, From the comparison, Random Forest which was employed by (Sabbeh, 2018) is the best model compared to the other studies, this researcher used cross-validation and parameters tuning to improve the model to an accuracy of 96%. (Jain et al., 2021) used different feature engineering and selection techniques and the Random Forest model accuracy was 95% finally, this study model is third with an accuracy of 94.7%.

The weakest model is (Lalwani et al., 2021) which used gravitational algorithms in feature selection the resulting model had an accuracy of 80.14%.

Generally, the random forest has proved efficient in accurately predicting customer churn, as shown in the table Random Forest is the best model for many researchers. From the study, this

model from the study can be improved by using cross-validation and appropriate feature selections techniques.

Conclusion

Machine learning models have proved efficient in predicting customer churn with all the models having an accuracy of more than 90% after tuning, The success of machine models lies in preprocessing, This study utilized SMOTE in solving the imbalanced problem of the dataset, in comparison to balancing techniques SMOTE does not randomly replicate the minority sample, but it effectively uses KNN to generate new instances around the minority sample, therefore effective preprocessing leads to and better accuracy.

Finally, the Random Forest is the most efficient classifier of telco customer churn problem as it is proved by (Sabbeh, 2018), (Jain et al., 2021), (Mishra & Reddy, 2017) in which random forest performed best in predicting customer churn

References

Asif Yaseen. (2021). Next-Wave of E-commerce: Mobile Customers Churn Prediction using Machine Learning. *Lahore Garrison University Research Journal of Computer Science and Information Technology*, *5*(2), 62–72. https://doi.org/10.54692/lgurjcsit.2021.0502209

Bzdok, D., Altman, N., & Krzywinski, M. (2018). Statistics versus machine learning. *Nature Methods*, *15*(4), 233–234. https://doi.org/10.1038/nmeth.4642

Chauhan, V. K., Dahiya, K., & Sharma, A. (2018). Problem formulations and solvers in linear SVM: a review. *Artificial Intelligence Review*, *52*(2), 803–855. https://doi.org/10.1007/s10462-018-9614-6

Dalli, A. (2022). Impact of Hyperparameters on Deep Learning Model for Customer Churn Prediction in Telecommunication Sector. *Mathematical Problems in Engineering*, *2022*, 1–11. https://doi.org/10.1155/2022/4720539

De, S., P, P., & Paulose, J. (2021). Effective ML Techniques to Predict Customer Churn. *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*. https://doi.org/10.1109/icirca51532.2021.9544785

Ganaie, T. A., & Bhat, M. A. (2021). Switching Costs and Customer Loyalty: A Review of literature. International Journal of Management and Development Studies, 10(05), 7–14. https://doi.org/10.53983/ijmds.v10i05.369

Huang, X., Izza, Y., Ignatiev, A., & Marques-Silva, J. (2021). On Efficiently Explaining Graph-Based Classifiers. *ArXiv:2106.01350 [Cs]*. https://arxiv.org/abs/2106.01350

Jain, H., Khunteta, A., & Srivastava, S. (2020). Churn Prediction in Telecommunication using Logistic Regression and Logit Boost. *Procedia Computer Science*, *167*, 101–112. https://doi.org/10.1016/j.procs.2020.03.187

Jain, H., Khunteta, A., & Shrivastav, S. P. (2021). *Telecom Churn Prediction Using Seven Machine Learning Experiments integrating Features engineering and Normalization*. https://doi.org/10.21203/rs.3.rs-239201/v1

Lalwani, P., Mishra, M. K., Chadha, J. S., & Sethi, P. (2021). Customer churn prediction system: a machine learning approach. *Computing*. https://doi.org/10.1007/s00607-021-00908-y

Mishra, A., & Reddy, U. S. (2017, November 1). *A comparative study of customer churn prediction in telecom industry using ensemble based classifiers*. IEEE Xplore. https://doi.org/10.1109/ICICI.2017.8365230

Mohammad, N. I., Ismail, S. A., Kama, M. N., Yusop, O. M., & Azmi, A. (2019). Customer Churn Prediction In Telecommunication Industry Using Machine Learning Classifiers. *Proceedings of the 3rd International Conference on Vision, Image and Signal Processing*. https://doi.org/10.1145/3387168.3387219

Nguyen, N. N., & Duong, A. T. (2021). Comparison of Two Main Approaches for Handling Imbalanced Data in Churn Prediction Problem. *Journal of Advances in Information Technology*, *12*(1), 29–35. https://doi.org/10.12720/jait.12.1.29-35

Picek, S., Heuser, A., Jovic, A., Bhasin, S., & Regazzoni, F. (2019). The Curse of Class Imbalance and Conflicting Metrics with Machine Learning for Side-channel Evaluations. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 209–237. https://doi.org/10.13154/tches.v2019.i1.209-237

Pustokhina, I. V., Pustokhin, D. A., Nguyen, P. T., Elhoseny, M., & Shankar, K. (2021). Multi-objective rain optimization algorithm with WELM model for customer churn prediction in telecommunication sector. *Complex & Intelligent Systems*. https://doi.org/10.1007/s40747-021-00353-6

Rigatti, S. J. (2017). Random Forest. *Journal of Insurance Medicine*, *47*(1), 31–39. https://doi.org/10.17849/insm-47-01-31-39.1

Sabbeh, S. (2018). Machine-Learning Techniques for Customer Retention: A Comparative Study. *IJACSA) International Journal of Advanced Computer Science and Applications*, *9*(2). https://pdfs.semanticscholar.org/2a9f/505e1ab148aa3d91810f509ee133272be554.pdf

Senawi, A., Wei, H.-L., & Billings, S. A. (2017). A new maximum relevance-minimum multicollinearity (MRmMC) method for feature selection and ranking. *Pattern Recognition*, *67*, 47–61. https://doi.org/10.1016/j.patcog.2017.01.026

Sergue, M. (2020). *Customer Churn Analysis and Prediction using Machine Learning for a B2B SaaS company*. DIVA.https://www.divaportal.org/smash/record.jsf?pid=diva2%3A1426161&dswid=8990

V.Anandi, D., & Ramesh, D. M. (2021). Analysis of customer relationship management using Machine Learning. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, *12*(14), 5505–5512. https://www.turcomat.org/index.php/turkbilmat/article/view/11670

Xiahou, X., & Harada, Y. (2022). B2C E-Commerce Customer Churn Prediction Based on K-Means and SVM. *Journal of Theoretical and Applied Electronic Commerce Research*, *17*(2), 458–475. https://doi.org/10.3390/jtaer17020024