Customer Attrition Prediction

I. Introduction

Customer Attrition, commonly referred to as customer churn is a situation where a customer leaves a company. It involves identifying customers likely to cancel a service subscription, which presents a good chance to improve customer satisfaction and prevent loss of income [1]. The findings from customer attrition research significantly influence bank policies because these results enable them to create fresh customer strategies or enhance current ones. Given the difficulty of acquiring new customers in the competitive banking industry, the fundamental objective of banks is to ensure the retention of existing customers[2]
This report, therefore, aims to find out the causes of customer attrition and predict clients who are likely to stop being customers or cancel their credit card service subscriptions. This will be achieved by exploring four supervised learning algorithms to classify the customers using the most important attributes. With these predictions, the organisation can strengthen its marketing strategies and create new marketing strategies to ensure that customers are retained.

This study explored four models. Decision Tree, Random Forest, Support Vector Machine (SVM) and AdaBoost algorithms were trained, optimised, validated and tested for this task. The AdaBoost achieved the highest scores for all the performance metrics. It reported a recall value of 0.89 and classified the highest number of attrited customers correctly. Thus, it is recommended as the best model for predicting credit card customer attrition rate.

II. Data and Preliminary Analysis

A. *The Dataset*

This dataset contains 10127 instances or observations and 21 features or attributes of unique credit card customers. The categorical feature Attrition flag is the class label, and it has two states (Attrited customer and Existing customer). This makes it a binary classification task.
The dataset has no missing or duplicate values. However, some of the attributes contain unknown values. These unknown instances will be assumed to be missing values and they will be dealt with.
Some outliers were observed in the dataset, however, because this data is financial data that deals with money. Some extremely low and high values are expected and for this reason, no outliers were removed from the dataset.

B. *Exploratory Data analysis*

Furthermore, the dataset is also imbalanced that is the distribution of the observations in the class label is biased. In this dataset as seen in figure 1 below, the class label (Attrition flag) contained 83.9% samples for existing customers which can also be referred to as the negative class and 16.1% for the attrited customers also known as the positive class.

.

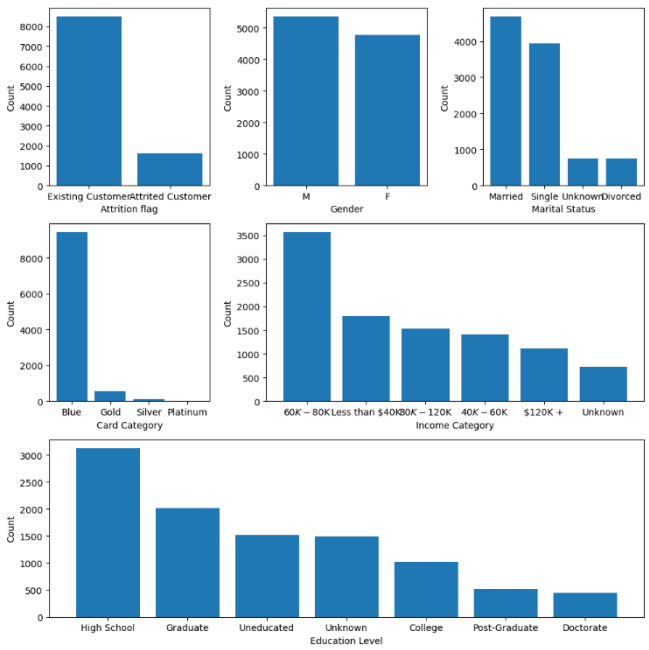

Figure 1: Count plot of the nominal attributes in the dataset

Figure 2 below are histogram plots showing the distribution of some of the continuous attributes in the dataset. Some of the attributes such as Customer age (Customer_Age), Change in transaction amount between Q4 and Q1 (Total _Amt_Chng_Q4_Q1), Average card utilization ratio (Avg_Utilization_Ratio) and period of relationship with the bank (Months_on_book) are normally distributed while the others are not.

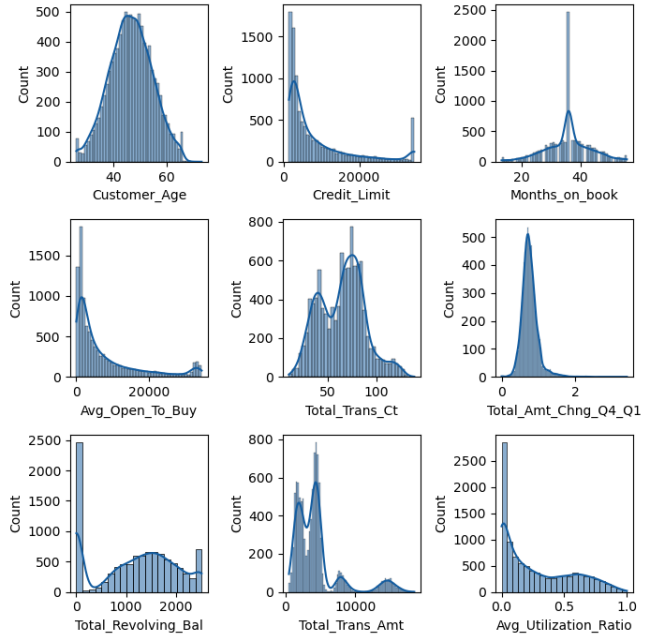

Figure 2 Distribution of the continuous attributes in the dataset

Figure 3 below shows the relationship between some of the attributes in the dataset. A strong positive relationship can be observed between credit limit and average open-to-buy credit line, customer age and period of relationship with the bank. Average utilisation ratio and Total Revolving balance, Total transaction amount and Total transaction credit. In addition, it can also be observed from the plot that the classes are not linearly separable.

Figure 3: Scatterplot showing the relationship between the continuous attributes

III. DATA PRE-PROCESSING

In dealing with the unknown instances which are assumed to be missing. They were mostly ordinal and nominal attributes a simple method of imputation method involving the mode was used to replace the missing values.

Categorical variables were converted to numeric attributes because machine learning algorithms cannot understand them. The class attribute containing 2 classes: Attrited customers and Existing customers were encoded into 0 and 1 respectively. Nominal attributes such as Gender, Marital status and Card category were also encoded using the pandas get dummies function. Ordinal attributes such as education level and Income category were ordinally encoded. In addition to this, the dataset is also imbalanced that is there are very few examples of the Attrited customer class when compared with the Existing customer class. To address this, the random oversampling strategy will be used. The Synthetic Minority Oversampling Technique (SMOTE) will be applied to improve the algorithm's learning of the minority class.

IV. METHODOLOGY

This task is a supervised machine-learning task because it has a labelled class output attribute. Supervised learning is a machine learning approach that learns by mapping input attributes to output attributes and then utilises this mapping to make predictions or to classify (Greene et al.,). In other words, it learns from examples by having a training data set from which the model/algorithm learns and the testing data set which is used to evaluate how much learning the model has done.

The four learning algorithms used in this study include Decision Tree Classifier, Random Forest Classifier, Support Vector Machines Classifier and AdaBoost Classifier. The four algorithms followed the same pipeline.

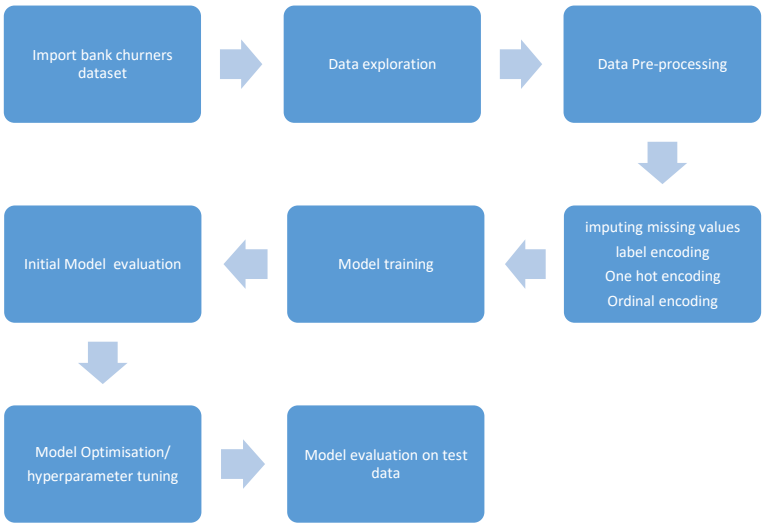

Figure 4: Machine learning architecture for the report

A. *Training, validation and testing data*

After the data pre-processing stage, the dataset was split into training and testing datasets using the 70:30 ratio. Thereafter the training set was further split into training and validation datasets following the 70:30 ratio as well.

The validation dataset was used to validate the training done by the model and the hyperparameter tuning of the models. The testing data was used only once on the optimised model. This model selection method was used because reusing the same test dataset over and over during the process of model selection will make it become part of the training dataset and it could lead to overfitting

B. *Data Upsampling*

This task applied a random oversampling strategy by using SMOTE technique. This was applied to the training data set to prevent data leakage [3]. Up-sampling of the minority class was done so that the model has enough examples to learn from to increase its predictive ability of the attrited customer class. The models without any form of tuning were used on the datasets before SMOTE and after SMOTE and the results were reported.

C. Model training

The SMOTE training set was then used to train the model and to tune the hyperparameters and then validated with the validation dataset. For the final model evaluation and model selection, all optimised models were applied to the testing data once and the results were reported. The testing data was used only once for this purpose.

D. Model Evaluation

These are metrics that are used to quantify the performance of the model. These include accuracy, precision, recall, F1-score and AUC score.

However, for this task, the precision, recall, F1 score and AUC score will be reported. The accuracy scores will not be reported due to the imbalance in the dataset as it will not give a good understanding of the performance of the models.

Precision measures how accurate the positive predictions are. It is the number of true positives divided by the number of true positives plus the number of false positives.

$$Precision = \frac{True\ Positives\ \ (TP)}{True\ positives + False\ Positives\ \ (FP)}$$

Recall is the ability of the model to detect all the relevant cases or positive samples within the dataset. The higher the recall the higher the number of positive samples detected. It is the number of true positives divided by the number of true positives plus the number of false negatives.

$$Recall = \frac{True\ Positives}{True\ Positives + False\ negatives}$$

F1-Score is the harmonic mean of precision and recall. It is calculated as follows:

$$F1\ score = 2 * (\frac{Precision * Recall}{Precision + Recall})$$

The AUC score measures the ability of a binary classifier to differentiate between classes. It ranges from 0 to 1 and it is used as a summary of the ROC curve. The higher the AUC the better the model's performance at differentiating between positive and negative classes. An AUC of 0.5 indicates that the model is not better than a random guess.

The priority of this task is to predict the positive class (Attrited customer). Therefore, to evaluate and compare the models the precision, recall, F1-score and AUC score of the negative class will be used.

Overall the precision score and AUC score will be preferred. The precision score because it indicates that the learning model is capable of identifying the positive cases which is the attrited customer and the AUC score because it indicates that the model is capable of distinguishing the 2 classes.

The classification algorithms that are relevant to this task are discussed below:

I Decision Tree Algorithm

A decision tree classifier is a learning algorithm that classifies test data by learning decision rules from the features or attributes in the training data set. It is made up of the Parent node, the internal mode and the leaf node. It usually starts at the tree root and then it splits the data based on the feature that gives the highest information gain. It is an iterative process in which the splitting at the node can be repeated until all the leaves are pure. This can however lead to extremely deep trees with many nodes and can easily lead to overfitting which is one of the disadvantages of this algorithm because it uses a greedy algorithm to learn from the training dataset.

Decision trees are flexible models that are useful in fitting both categorical and continuous data. This flexibility makes it, one of the most widely used models in churn prediction [4], [5].

The first stage involved fitting the model on the training dataset without any form of hyperparameter tuning. Then the hyperparameters were tuned to obtain an optimised model. a Criterion used was Gini, a max depth of 20 to prevent overfitting of the data and a random state value of 42 to ensure the reproducibility of results. Other parameters like minimum sample split and minimum sample split leaf which account for several points in a leaf and internal node to was set to 30 and default respectively because this dataset had over ten thousand data points.

II Random Forest Classifier

This is a powerful and popular supervised machine-learning approach that is frequently applied to solve classification tasks [6]. As an extension of the bagging method, it uses both bagging and feature randomisation to build many uncorrelated decision trees on different subsets of a given dataset and averages the results to increase the predicted accuracy of that dataset [7].

Feature randomisation commonly referred to as feature bagging or the random subspace method creates a random selection of features while ensuring minimal correlation among the decision trees. The random forest then takes the results from each decision tree and bases the final output on the majority votes of forecasts rather than depending on a single tree [7], [8].

The higher the number of trees the more precise the outcome is.

This model was used because it reduces the risk of overfitting and is also able to perform classification problems with a high degree of accuracy. It also makes it easy to determine the importance of a feature and its contribution to the model.

A key distinction between decision trees and the random forest algorithm is that decision trees evaluate all the possible feature splits while random forest chooses only some of the features[7].

For this model, hyperparameters were tuned to obtain an optimised model. The best number of estimators was obtained by using a cross-validation process. The number of estimators was set to 231, a maximum depth of 52 and a random state of 42 to ensure the reproducibility of results.

This Classifier was selected because it is an ensemble learning model that uses the bagging technique. It is one of the most popular and powerful techniques in the recognition of patterns and machine learning for high-dimensional classification and skewed problems and as mentioned earlier this class attribute of this data is skewed.

III Support Vector Machine

It is a supervised learning method of machine learning used for classification problems. It works by finding out the best plane or hyperplane (depending on the data dimension). By identifying the ideal hyperplane that divides the classes of data points, SVM categorises the data points [9].

It ensures that the margin is maximised for the vectors—the data points nearest to the hyperplane. If the data points are not linear, SVM projects them to n-dimension, identifies the best hyperplane and then uses the kernels to project the data back to the original dimension. SVM works effectively with linear data [9].

The SVM algorithm has received the greatest attention and inspection for its outstanding predictive ability for churn prediction [10]–[12].

Before training the model, standardization was applied to the data. Standardisation helps to put the features on the same scale so that features with a higher magnitude do not control the model. It gives the attributes a mean of 0 and a standard deviation of 1 [13]. For this task, a pipeline with the standard scaler was used.

This study optimised the model several times by adjusting hyperparameters like kernel type, C and gamma. The trade-off between sacrificing margin separation and separating the data points is controlled by the C parameter.

The optimised model used the RBF kernel gamma of 0.01 and C of 15.

It was also observed that any form of hyperparameter tuning in SVM tended to choose between precision and recall. When the precision value increased the recall value reduced

IV AdaBoost Algorithm

This algorithm is a boosting algorithm to enhance prediction performance by combining weak classifiers to produce an improved classifier. It trains by using a randomly selected subset of training data points. AdaBoost gives more weight to incorrectly classified observations after each training step to increase the probability of accurate classification in the following training[14]

Until all training data points are fit without any misclassification, or the maximum number of estimators is reached this process will be repeated [14].

An optimised model was obtained by experimenting with the hyperparameters. The number of estimators which is the number of weak classifiers to be trained was set to 300, the base estimator Decision Trees was used, and other base estimators were tried but the decision trees performed best. A learning rate of 0.9 was set, and the algorithm was set to "SAMME". A random state of 42 to ensure the reproducibility of the results.

V. EXPERIMENT

This report aimed to predict whether a customer is likely to cancel his account. This means that a model that classifies the minority label class (Attrited customer) is the ideal one for this task and the organisation.

Table 1: Performance of the four models without the SMOTE technique

Data type	Algorithm	precision	recall	F1-score	AUC score
Training	Decision Tree	1.0	1.0	1.0	1.0
	Random Forest	1.0	1.0	1.0	1.0
	SVM	0.92	0.74	0.82	0.97
	AdaBoost	0.91	0.87	0.89	0.99
Validation	Decision Tree	0.81	0.77	0.79	0.86
	Random Forest	0.89	0.77	0.83	0.98
	SVM	0.81	0.63	0.71	0.95
	AdaBoost	0.87	0.84	0.86	0.98

Table 2: Comparison of the performance of the four models with the SMOTE technique

Data type	Algorithm	precision	recall	F1 score	AUC score
Training	Decision Tree	1.0	1.0	1.0	1.00
	Random Forest	1.0	1.0	1.0	1.00
	SVM	0.96	0.98	0.97	0.95
	AdaBoost	1.00	1.00	1.00	0.97
Validation	Decision Tree	0.73	0.84	0.78	0.89
	Random Forest	0.84	0.80	0.82	0.98
	SVM	0.68	0.80	0.74	0.95
	AdaBoost	0.81	0.87	0.84	0.98

Table 1 shows the precision, recall, F1 score and AUC score of the models without any hyperparameter tuning and before up-sampling the training dataset (application of SMOTE technique) while table 2 shows the same performance metrics after the application of SMOTE. It is observed that after up sampling the recall and AUC scores for all the models increased. This means that the ability of the models to correctly predict the attrited customers increased. However, the precision scores were reduced for all models. Only the SVM model recorded an increase in the F1 score.

Table 3: Comparison of validation scores with test scores using and non-optimised and optimised models

Model Type	Algorithm	Precision	recall	F1 score	AUC Score
Non-optimised model	Decision Tree	0.73	0.84	0.78	0.89
	Random Forest	0.84	0.80	0.82	0.98
	SVM	0.68	0.80	0.74	0.95
	AdaBoost	0.81	0.87	0.84	0.98
Optimised model	Decision Tree	0.74	0.86	0.79	0.93
	Random Forest	0.85	0.81	0.83	0.98
	SVM	0.69	0.83	0.75	0.95
	AdaBoost	0.84	0.89	0.86	0.98

Table 3 shows the precision, recall, F1-score and AUC score of the optimised models (tuned hyperparameters and SMOTE application) and non optimized (SMOTE technique alone). It compares the scores obtained from the validation dataset before the application of the optimised model and after the application of the optimised model. It can be observed that hyperparameter tuning slightly increased some of the performance metrics (precision, and recall scores for most of the models and left some scores the same. Increases observed were between 0.01 and 0.03. All models except Decision Tree recorded higher precision, recall and F1 scores. All models except Random Forest recorded higher AUC scores. The Random Forest model and SVM model recorded higher precision scores and F1- scores. The AdaBoost model had the same scores for both the validation and test dataset.

Table 4: Comparison of the selected models' performance on test data

Algorithm	Precision	Recall	F1-Score	AUC Score
Decision Tree	0.72	0.82	0.77	0.93
Random Forest	0.88	0.80	0.84	0.98
SVM	0.71	0.83	0.76	0.96
AdaBoost	0.84	0.89	0.86	0.98

Table 4 shows the performance of the selected models on the test data. The AdaBoost model is recommended for credit card churn prediction because it had the highest scores for all the performance metric scores. It also had the highest AUC score which implies that it can distinguish between attrited customers and existing customers.

Figure 5 shows the confusion matrix result of the AdaBoost learning algorithm on the test data. It can be observed that it correctly classified the majority of both the attrited customers and existing customers. This would enable the organization to target their marketing strategies towards the right people and also prevent wastage of resources since the number of misclassifications for the existing customers is not many.

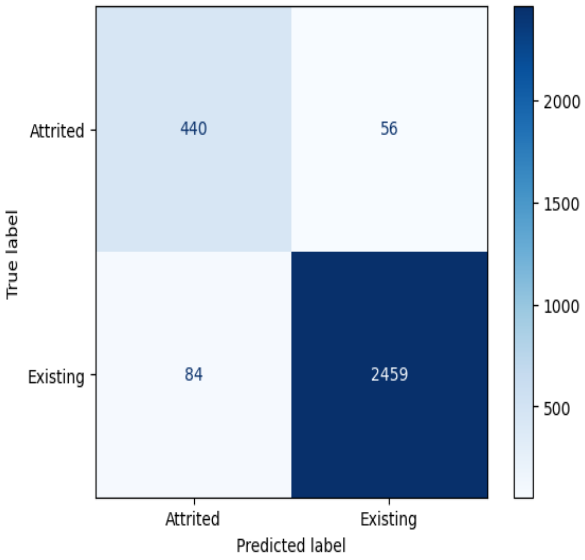

Figure 5: Confusion matrix using AdaBoost algorithm on test data

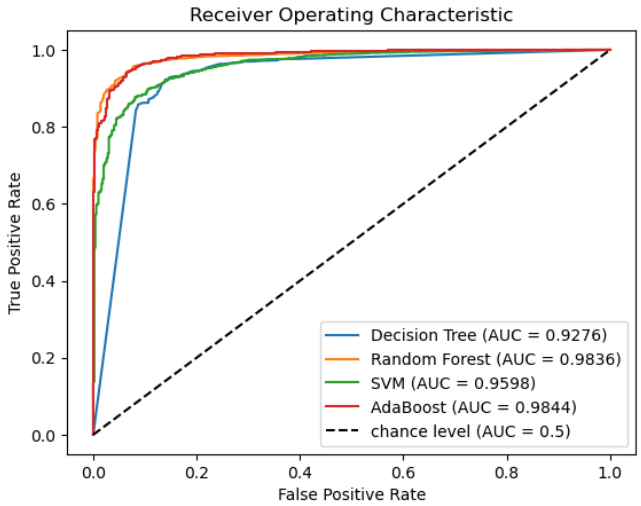

Figure 6: ROC curve of the four models in the study

Figure 6 above shows the trade-off between the true positive rate and the false positive rate for the four models. All four models had high AUC scores and performed better than a random guessing model. However, the AdaBoost model had the highest AUC score.

VI. REFLECTION

This task comprehensively investigated banks' credit card churn prediction problems using machine learning techniques. A prediction system using Decision Trees, Random Forest, SVM and AdaBoost algorithms was explored. The best results were achieved when the skewed data set was up-sampled using SMOTE. Recall scores and AUC scores were observed to increase. When the hyperparameters for each model were tuned the models reported better precision, recall and F1scores. While there is a slight decrease in the performance metric scores of the training, validation and test data, there is no indication of overfitting in all the models.

All the models were better than random guessing and can distinguish the two classes as observed from their AUC scores.

The optimised model was applied once to the test data for model evaluation and selection. The AdaBoost model performed consistently better than all other models and it is recommended for predicting which customers are likely to cancel their accounts.

One of the observations from this task is that the classifiers struggled to learn from the minority class because they had fewer data points. SMOTE was however able to make this better. In addition, the SVM performed the worst on the testing data probably because it is sensitive to outliers.

Future work on this task can down-sample the majority class instead of up-sampling the minority class as was done in this study. However, it should be noted that down-sampling can lead to information loss.

In addition, deep learning models can also be implemented to improve precision and accuracy.

In terms of hyperparameter tuning instead of the manual process that was adopted in this study, hyperparameter tuning algorithms such as GridSearchCV, RandomisedSearchCV can be adopted. It should be noted that many of them are computationally expensive and take a long time to run.

VII. REFERENCES

[1] J. Latheef and S. Vineetha, "Exploring Data Visualization to Analyze and Predict Customer Loyalty in Banking Sector with Ensemble Learning," *International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE*, vol. 4, no. 9, 2021, doi: 10.29027/IJIRASE.v4.i7.2021.891-904,March.

[2] H. Guliyev and F. Yerdelen Tatoğlu, "Customer churn analysis in banking sector: Evidence from explainable machine learning models," *Journal of Applied Microeconometrics*, vol. 1, no. 2, pp. 85–99, Dec. 2021, doi: 10.53753/jame.1.2.03.

[3] M. T. Haque Khan Tusar, M. T. Islam, and F. I. Raju, "Detecting Chronic Kidney Disease(CKD) at the Initial Stage: A Novel Hybrid Feature-selection Method and Robust Data Preparation Pipeline for Different ML Techniques," in *5th International Conference on Computing and Informatics, ICCI 2022*, 2022, pp. 400–407. doi: 10.1109/ICCI54321.2022.9756094.

[4] M. Islam and M. Habib, "A data mining approach to predict prospective business sectors for lending in retail banking using decision tree," *arXiv preprint arXiv:1504.02018*, 2015.

[5] G. R. Kumar, K. Tirupathaiah, and B. Krishna Reddy, "Client Churn prediction of banking and fund industry utilizing machine learning techniques," *International Journal of Computer Sciences and Engineering*, vol. 7, no. 6, pp. 842–846, 2019.

[6] L. Breiman, "Random forests," *Mach Learn*, vol. 45, pp. 5–32, 2001.

[7] "What is Random Forest? | IBM." https://www.ibm.com/topics/random-forest (accessed Mar. 17, 2023).

[8] D. Petkovic, R. Altman, M. Wong, and A. Vigil, "Improving the explainability of Random Forest classifier-user centered approach," 2017. [Online]. Available: www.worldscientific.com

[9] M.-W. Huang, C.-W. Chen, W.-C. Lin, S.-W. Ke, and C.-F. Tsai, "SVM and SVM ensembles in breast cancer prediction," *PLoS One*, vol. 12, no. 1, p. e0161501, 2017.

[10] A. S. Kumar and D. Chandrakala, "An optimal churn prediction model using support vector machine with adaboost," *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol*, vol. 2, no. 1, pp. 225–230, 2017.

[11] D. Mahajan and R. Gangwar, "Improved customer Churn behaviour by using SVM," *International Journal of Engineering and Technology*, pp. 2372–2395, 2017.

[12] S. Kumar, S. Viswanandhne, and S. Balakrishnan, "Optimal customer Churn prediction system using boosted support vector machine," *International Journal of Pure and Applied Mathematics*, vol. 119, no. 12, pp. 1217–1231, 2018.

[13] A. Jin, P. M. S. Basnet, and S. Mahtab, "Microseismicity-based short-term rockburst prediction using non-linear support vector machine," *Acta Geophysica*, vol. 70, no. 4, pp. 1717–1736, Aug. 2022, doi: 10.1007/s11600-022-00817-4.

[14] "What is Boosting? | IBM." https://www.ibm.com/uk-en/topics/boosting (accessed Mar. 17, 2023).