Journal of Information Systems Engineering and Management
2025, 10(57s)
e-ISSN: 2468-4376
https://jisem-journal.com/

Research Article

Customer Churn Prediction in Banking Sector Using Machine Learning

[1]Miriyala Lavanya, [2]Dammati Pavan Kumar, [3]Kistam Gopi, [4]Dr. Jajam Nagarju

[1]Mtech Student, Dept of CSE, Tirumala Engineering College, Narasaraopeta, Email id - lavanyamiriyala0505@gmail.com

[2]Associate Professor, Dept of CSE, Tirumala Engineering College, Narasaraopeta, Email id - dammatipavan@gmail.com

[3]Associate Professor, Dept of CSE, Tirumala Engineering College, Narasaraopeta, Email id- ,gopi2252@gmail.com

[4]Associate Professor, Dept of CSE, Tirumala Engineering College, Narasaraopeta, Email id - Jajamnagaraju@gmail.com

ARTICLE INFO

Received: 15 April 2025

Revised: 28 May 2025

Accepted: 13 Jun 2025

ABSTRACT

Bank time deposits provide consistent returns on investment. However, there are difficulties in locating and luring new clients. By using a combination of XGBoost, ADSYN, and Random Search optimization strategies to handle data imbalance, this study improves the predictive power of deposit categorization models. The study makes use of a Bank Marketing dataset from the UCI Repository that is openly accessible. comprising 45,211 items with a notable class disparity (88.3% of "no" responses and 11.7% of "yes" responses). While Random Search effectively optimizes model parameters, ADASYN integration enhances minority class representation. Our suggested hybrid model outperforms conventional methods by achieving an accuracy of 94.93%, precision of 94.93%, recall of 94.95%, and ROC-AUC score of 0.9919. These results demonstrate the efficacy of our approach in comparison to baseline models. This hybrid model accomplishes our research goals and improves customer data analysis. We go over the difficulties of integration, such as the need for computation and the choice of methods. The findings highlight the significance of assessing statistical significance in model enhancements and mitigating noise caused by synthetic samples. The study highlights how machine learning can be used to solve problems in the financial sector, with a focus on how feature engineering and data pretreatment affect performance. In order to improve model scalability and further minimize complexity, future research may investigate AutoML, which could lead to more creative consumer data analysis

Keywords: Data Imbalance; Machine Learning; Minority Class Representation; Optimization; Predictive Models

1. Introduction

Investors can benefit from the reliability and predictability of bank time deposits as an investing instrument [1].

Compared to a standard savings account, investors might earn a greater interest rate on funds that are locked away for a specific length of time [2]. Both individual budgeting and the liquidity management techniques employed by financial institutions rely on this idea, making banks' operational demands and their clients' financial security work together [3]. Both the bank and the customer make a financial commitment when they sign up for a term deposit [4]. Banks get access to long-term capital that they can use towards lending or investments, and customers get competitive interest rates [5]. Standard term deposits and more complicated schemes with value-added features are only two examples of the many deposit alternatives offered by banks today. These innovations aim to satisfy the varied financial needs of consumers.

But it's not easy for banks to find and entice people to sign up for term deposits. The problem is worsened by the fact that there is a class imbalance in the consumer databases; a minority class of customers are more likely to subscribe to time deposits than a majority class of customers. Conventional machine-learning methods, which skew towards the majority, are severely impaired by this inequality. In order to analyse customer data and predict their propensity to subscribe to term deposits, banks have turned to artificial intelligence technologies, particularly machine learning, [7], [8]. This is because traditional direct marketing strategies are costly and frequently ineffective. Machine learning is a subfield of AI that allows computers to autonomously learn new tasks and recognise patterns in data without any human intervention or programming [9].

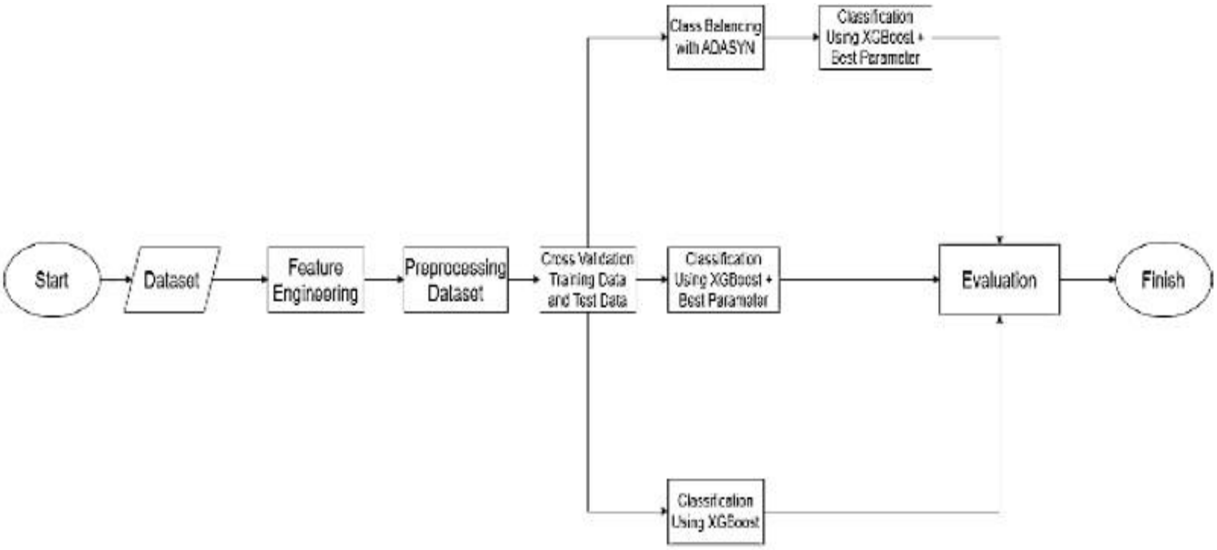
The data imbalance problem is a major obstacle for this technology. When there are more consumers who haven't subscribed than those who have, it makes it hard for AI models to make correct predictions [10]. The current study takes a fresh tack in addressing this issue by combining XGBoost with the Adaptive Synthetic Sampling (ADASYN) technique. Compared to older methods like the SMOTE (Synthetic Minority Over-sampling Technique), ADASYN is better at improving minority class representation and lowering the detrimental impact of data imbalance [11]. A more targeted synthetic sample is employed to do this, which aids in noise reduction and enhances the dataset's quality for training models [12]. Using datasets and comparing ADASYN, SMOTE, and SMOTE-KNN oversampling approaches, research by [13] examined the influence of resolving class imbalance on classification outcomes. In most cases, combining ADASYN with Random Forest is the way to go for optimal results. A different study [14] employed XGBoost and ADASYN, two machine learning models for oversampling approaches, to examine financial data in search of fraud anomalies; the study achieved a 99% accuracy rate. By lowering the bias towards majority classes commonly observed in imbalanced datasets, this method is anticipated to enhance the model's capacity to detect possible depositors with more accuracy.

To further enhance the model's management of this data imbalance, this study employs the Random Search parameter search technique. Random Search is more efficient in exploring a large parameter space than conventional grid search methods [15]. For complicated models like the one utilised in this work, this method of randomly selecting parameter combinations allows for the efficient and rapid discovery of optimal settings [16]. By optimising parameter values using Random Search, the research conducted by [15] achieved a 95% accuracy rate. Classification was also carried out using the K-nearest-neighbor machine learning algorithm in another work [17], with the hyperparameter value being determined via Random Search. It improved the accuracy to 78% with hyperparameters, up from 77% before tuning. To tackle the issue of class imbalance in bank marketing data, this work seeks to construct a more effective classification model by integrating XGBoost, ADASYN, and Random Search. The goal of this research is to help banks find people who could be interested in term deposits by improving their predictive accuracy and addressing the imbalance difficulty. This will help them save money and improve their marketing strategies for forecasting who might be interested in these deposit products [6]. Boosting marketing effectiveness without cutting down on customer

2. Materials and Methods

This study utilizes machine learning methodologies, notably the XG-Boosts and ADASYN algorithms, to address the issue of class imbalance in the dataset. Figure 1 explains the steps of this research.

Figure 1 : Flow of Work process


Start
Dataset
Feature Engineering
Preprocessing Dataset
Cross Validation Training Data and Test Data
Class Balancing with ADASYN
Classification Using XGBoost + Best Parameter
Classification Using XGBoost + Best Parameter
Classification Using XGBoost
Evaluation
Finish

2.1. Data Collection

The study relied on the UCI Repository Bank Marketing Dataset, a publicly available dataset on bank marketing. Portuguese financial entities made public this dataset in 2012 [18]. It covers the years 2008–2010. You may access the dataset here; it contains 45,211 records with 16 attributes and 1 class label per record [19]. Here is the URL:

https://archive.ics.uci.edu/dataset/222/bank+marketing. Included in it are input and output variables that have distinct qualities. A notable class discrepancy is observed for the 'y' characteristic, with 'yes' outcomes accounting for 11.7% and 'no' outcomes 88.3%; this poses a substantial barrier to both the accuracy and training of models [20]. The table below provides an overview of the dataset's unique attributes. Table 1 lists all of the dataset's characteristics

Table 1. Dataset's Information

Attributes Descriptions	Attributes Descriptions
Age	Numeric
Job	Job Categoreis
Martial	Martial Status
Education	The data include primary , secondary and Tertiary
Balance	Average yearly balance in Rs
Default	Has credit in default or no
Loan	Has housing loan or not

Housing	Last contact day of the month
Day	Last contact month of year
Month	Last contact month of year
Contact	Contact communication types mobile , telephone
Campaign	Contract between client and customer
Duration	Last contract duration
Previous	Several contacts were performed before this campaign and for this client
Pdays	A number of days passed after the client was last contacted from a previous campaign
Poutcome	Outcomes of the previous marketing campaign include unknown, other, failure, and success
Class (Y)	Has the client subscribed to a term deposit or no

2.2. Feature Engineering

Feature engineering is frequently referred to as the process of deriving significant features or attributes from data through the use of existing domain knowledge or transformation techniques [21], [22]. This research employed domain expertise and statistical methods to augment the model's predictive capability. The feature engineering technique generates novel features from existing ones. Attributes that can enhance efficacy in forecasting and categorisation [23]. We developed additional characteristics, including "age category" and "status category," derived from existing data. The "age category" was derived from the "age" characteristic, segmenting it into significant groupings that may represent varying financial behaviours. Likewise, the "status category" was developed from the "balance" feature, classifying individuals into tiers of financial soundness.

2.3. Data Preprocessing

During the preprocessing phase, raw data undergoes cleaning and preparation for integration into the algorithmic model. This process entails addressing absent entries and eliminating duplicates to guarantee data cleanliness for model training. The objective is to create a clean dataset for analysis utilising machine learning algorithms [24]. Encoding techniques were applied for conversion.

Transform categorical information into a format suitable for machine processing. Label encoding was employed for binary categories, while one-hot encoding was utilised for attributes with multiple categories. Raw data is often unsuitable for machine learning applications because it may contain missing entries, class imbalances, and inconsistencies. The dataset utilised in this research is of high quality, having undergone rigorous verification to ensure it is devoid of missing entries and duplications. The preprocessing in this study entailed the conversion of categorical data into binary dummy variables via label encoding for the categories "yes" and "no," while one-hot encoding was employed for attributes with multiple categories. One-hot encoding is a commonly employed method for representing categorical variables as vectors essential for statistical models [19], [26].

2.4. Cross Validation

Cross-validation techniques are employed to enhance the work process and improve the accuracy of predictions. Decreasing the magnitude of extensive data sets results in this occurrence. One component validates the model, while the other is responsible for training the classifier. This approach was iterated multiple times using different validation subsets [27]. By employing a K-fold cross-validation methodology, with K set to 10, we could effectively leverage the dataset for training and validation purposes. This strategy mitigated the possibility of model overfitting and yielded a more precise evaluation of the model's performance on unknown data. Another role of Cross-Validation is to convert the original data into training and testing data. If K = 10, then K is defined as being tenfold. For this study, a value of K equal to 10 was utilized. Adjusting the K value to 10 segments the data into ten separate datasets, where one subset is designated for testing, and the others are used for training. This process will be repeated for each dataset based on the set K value [28]. Cross-validation techniques are utilized to evaluate tuning parameters that are unknown beforehand and to measure the prediction error level in the ultimate model [29].

2.5. Adaptive Synthetic Sampling Approach (ADASYN)

ADASYN is a technique for oversampling minority data in machine learning, aimed at mitigating class imbalance issues [30]. This strategy produces new instances of the minority class by taking into account their complexity in the learning process. The primary aim is to create more intricate samples for classification, resulting in a more varied and challenging dataset. Dataset intended for training purposes. This technique specifically focuses on minority data, posing greater analytical challenges compared to more comprehensible data. The process entails the generation of additional data samples for the under-represented class, taking into account the level of learning complexity [12]. ADASYN generates synthetic data for minority classes that are more difficult to understand by assigning distribution weights according to the level of difficulty. This method guarantees the generation of synthetic data from minority classes that are more challenging to understand, rather than those that are easier to interpret [31]. ADASYN represents an advanced version of the SMOTE (Synthetic Minority Over-Sampling Technique), aimed at mitigating over fitting by excluding exact duplicates of minority instances from the primary dataset [11]. The ADASYN method, as indicated in [32], calculates the ratio (d) of minority cases to majority cases using the specified formula.

$$d = ms/mi \quad .... \qquad (1)$$

In this formula, d represents the ratio of minority instances to majority instances, with ms denoting majority instances and mi indicating minority instances. The formula is subsequently employed to calculate G, the total number of synthetic observations to be generated.

$$\mathrm{G} = (\mathrm{mi} - \mathrm{ms})\ \beta \quad .... \qquad (2)$$

In this context, G denotes the quantity of synthetic data to be generated, ms represents majority instances, mi indicates minority instances, and β signifies randomly selected minority examples from randomly chosen minority examples within the same vicinity. Subsequently, ascertain the K-NN value for each minority instance and calculate the proportion of neighbours that belong to the majority class (ri) using the specified formula.

$$ri = \#Majorit/\ \mathrm{k} \qquad .... (3)$$

In this context, ri denotes the neighbours from the majority class, while k signifies the specified number of nearest neighbours (KNN). Normalization is subsequently applied to ri to ensure it equals 1, utilising the specified formula.

$$ri^{\hat{}} = ri \ / \sum ri \qquad .... \quad (4)$$

Where denotes neighbours from the majority, and represents the normalized value. Subsequently, determine the quantity of synthetic observations produced for each environment (Gi) utilizing the formula. a. $Gi =$

$$Gi{=}Gri^{\hat{}} \quad \text{..........................} \qquad (5)$$

The formula, ri, denotes neighbours from the majority class, while Gi signifies the quantity of synthetic observations generated for each local area. Determine the count of Gi in each locality and subsequently generate a new synthetic data point (Si) utilizing the formula.

$$si = (xi + xzi - xi) \ \text{..............................} \ (6)$$

Where si represents synthesized new data points, xi represents samples from minority groups within a locality, xzi represents random minority samples chosen from the same locality, and λ is a randomly selected number ranging from zero to one

2.6. Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) is an advanced machine learning approach that enhances the Decision Tree methodology [19]. The XGBoost model is an ensemble of decision trees. XGBoost demonstrates superior accuracy compared to the Decision Tree algorithm. Additionally, it has the potential to enhance and optimize GBDT (Gradient Boosting Decision Trees).

Decision Tree algorithm. The XGBoost technique, recognized for its efficiency and effectiveness in managing large datasets and complex models, was employed [33]. The XGBoost method accelerates modeling through the incorporation of regularisation terms [34]. XGBoost is a commonly utilized machine learning method [35]. The process of XGBoost, as outlined in the research [36], employs the following formula.

$$F(x) = argmin\_y \sum L(Y, y)\ n\ i{=}1 \quad \text{.........} \qquad (7)$$

L(Y, y) denotes the differential loss function, while n signifies the sample count. Next, calculate the pseudo residual using the following equation.

$$rim = -\ [\ [\delta L(Y{,}F(Xi))]\ /\ \ [\delta(Xi)] \qquad \text{....................} \qquad 8$$

Let i range from 1 to n. Subsequently, fit the base tree to the training data using the specified equation.

$$(xi, rim) \qquad \text{...............} \qquad 9$$

Subsequently, calculate the multiplier utilising the equation provided below.

$$ym = argminy \sum (Yi, Fm - 1(xi) + yh(xi)) \ \text{................} \qquad 10$$

Subsequently, revise the model utilizing the equation presented below.

$$Fm(x) = Fm{-}1(x) + ymhm(x) \qquad \text{.....................} \qquad 11$$

Execute the aforementioned procedures m times, where m represents the number of iterations.

2.7. Evaluation

We employed a confusion matrix to assess the model's effectiveness, evaluating its performance across different classes and ensuring precise representation of positive and negative predictions. The confusion matrix is essential for assessing classification models and comprises True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN). Elements (TN) [9]. True positives (TP) and true negatives (TN) represent accurate predictions for the positive and negative classes, respectively, whereas false positives (FP) and false negatives (FN) denote errors in these predictions. The confusion matrix is particularly effective in situations characterized by an imbalance between positive and negative classes [9]. In these instances, metrics such as Accuracy, Recall, F1-Score, ROC Area, and AUCPR are utilised to comprehensively evaluate minority classes [37], [38]. We will now examine the formulas for these evaluation metrics, starting with accuracy

Algorithm Steps

Algorithm Steps

Steps for implementing an ADASYN (Adaptive Synthetic Sampling) and XGBoost model in Python:

•Import Libraries: Import necessary libraries such as pandas, scikit-learn, and XGBoost.

•Load Data: Read your dataset into a pandas DataFrame.

•Handle Missing Values: Address any missing values in the dataset through imputation or removal.

•Feature Encoding: Encode categorical features into numerical representations if needed.

•Split Data: Divide the data into training and testing sets.

•Apply ADASYN: Use the ADASYN technique to handle imbalanced datasets by generating synthetic samples for the minority class.

•Initialize XGBoost Model: Create an XGBoost classifier or regressor object, specifying hyperparameters as needed.

•Train the Model: Fit the XGBoost model to the balanced training data obtained after applying ADASYN.

•Make Predictions: Use the trained model to make predictions on the test set.

•Evaluate Performance: Assess the model's performance using appropriate metrics like accuracy, precision, recall, F1-score, or AUC-ROC.

3. Results and Discussion

3.1. Model Performances

This research involved conducting several tests, including the application of XGBoost with hyper parameter optimization for deposit classification. After preprocessing the dataset, comprehensive testing was performed using K-fold cross-validation with a K value of 10 to mitigate over fitting. The dataset was partitioned into ten equal segments, followed by iterative testing to verify , Each component functioned as both test and training data. This process facilitated the derivation of a more reliable average performance measure for the model. This research utilized Random SearchCV for hyper parameter tuning to determine the optimal parameter combination for the XGBoost algorithm, thus improving model performance. The hyper parameters utilized in this study consist of co sample by tree, subsample, learning rate, n estimator, and max depth.

Tables 2 and 3 demonstrate that the second round achieves the same training and testing accuracy as the first round, with the optimal hyper parameter settings being colsample_bytree 0.8, subsample 0.6, learning rate 0.1, n_estimators 300, and max_depth 5. The hyper parameter value is constant. This research will employ the optimal hyperparameter value identified through RandomSearchCV. Classification testing will first be conducted using XGBoost, followed by an evaluation with the optimal hyperparameter. The consistent results achieved through multiple rounds of tuning indicate the reliability of the chosen hyperparameters. Subsequently, we will perform XGBoost classification utilising the optimal hyper parameter. Table 4 presents the results of the XGBoost test, comparing outcomes with and without hyper parameter optimization

Table 2. Round 1 RandomSearch CV

Parameters	Values
N_estimator	[100,200,300]
Max_dept	[3,5,7]
Colsample_byte	[0.6,0.6,1.0]
Learning_rate	[0.1,0.001]
Subsample	[0.6,0.8,1.0]

Parameters	Values
N_estimator	[300]
Max_dept	[5]
Colsample_byte	[0.8]
Learning_rate	[0.1]
Subsample	[0.6]
Training Accuracy	**0.9446**
Testing Accuracy	**0.9084**

Table 3. Round 2 Random Search CV

Parameters	Values
N_estimator	[300,500,700,1000]
Max_dept	[3,5,7,9]
Colsample_byte	[0.8,1.0]
Learning_rate	[0.1,0.5,0.7]
Subsample	[0.6]

Parameters	Values
N_estimator	[100]
Max_dept	[5]
Colsample_byte	[0.8]
Learning_rate	[0.1]

Subsample	[0.6]
Training Accuracy	**0.9446**
Testing Accuracy	**0.9084**

Table 4. XGBoost Test Results with Hyperparameter and without Hyperparameter

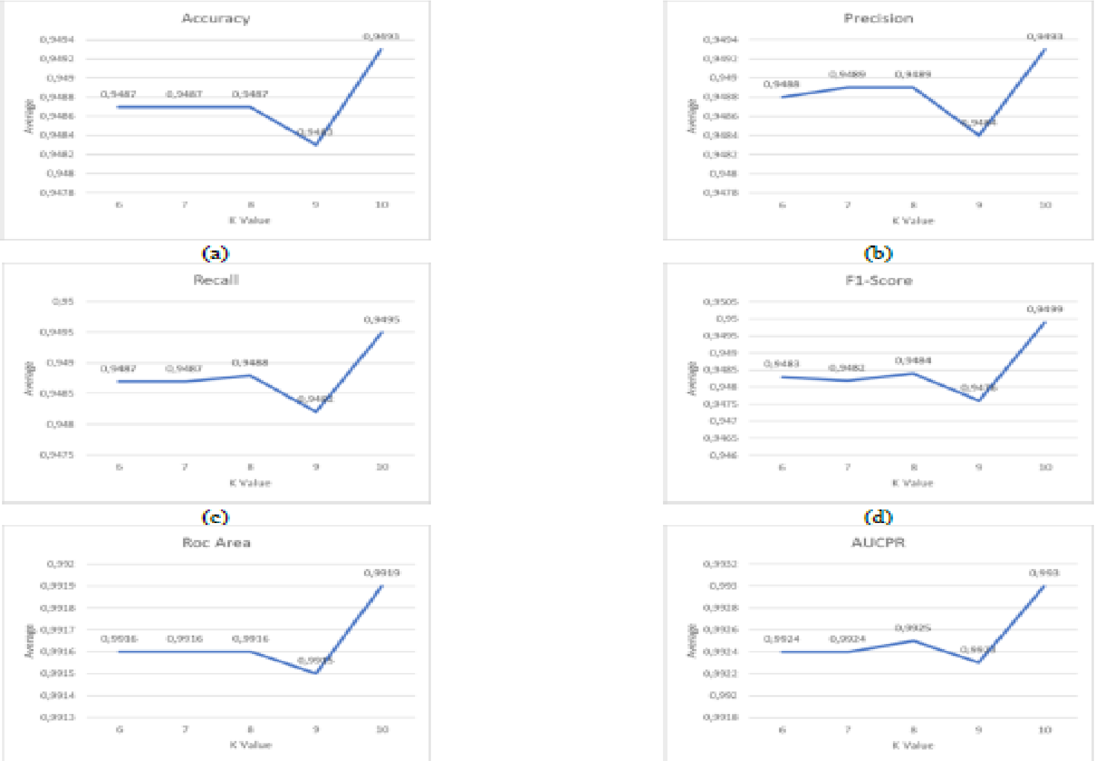
XGBoost (without Hyperparameters) **XGBoost (with Hyperparameters**

TEST NO	ACC	PREC	RECALL	FI-SCORE	ROC AREA	AUC PR	ACC	PREC	RECALL	F1-SCORE	ROC AREA	AUC PR
1	0.9018	0.7677	0.7132	0.5267	0.9288	0.5973	0.9099	0.7846	0.7202	0.5449	0.9398	0.5976
2	0.9098	0.7857	0.7432	0.5764	0.9356	0.6318	0.9095	0.8021	0.7335	0.5654	0.9463	0.6318
3	0.9133	0.8017	0.7445	0.5842	0.366	0.6475	0.9113	0.7901	0.7418	0.5831	0.9488	0.6419
4	0.9133	0.8039	0.7337	0.5298	0.9394	0.6675	0.9182	0.8205	0.7426	0.5987	0.9547	0.6417
5	0.9011	0.7635	0.7133	0.5509	0.9301	0.5994	0.969	0.7837	0.7924	0.5498	0.9326	0.6111
6	0.9044	0.7737	0.7294	0.5476	0.9295	0.5927	0.9099	0.7886	0.7379	0.5702	0.9635	0.6223
7	0.9053	0.7777	0.7250	0.5476	0.9356	0.6038	0.9098	0.7894	0.7295	0.5598	0.9326	0.6189
8	0.9115	0.8052	0.7117	0.5506	0.9308	0.6154	0.9094	0.7947	0.7095	0.5348	0.9301	0.6198
9	0.9122	0.7899	0.745	0.5506	0.944	0.6489	0.9098	0.7989	0.7390	0.5723	0.9461	0.6621
10	0.9122	0.7977	0.7428	0.5812	0.9441	0.6448	0.9122	0.7980	0.7420	0.5808	0.9434	0.6617
AVG	0.9088	0.7868	0.7310	0.566	0.9347	0.6266	0.9103	0.7941	0.7323	0.5658	0.9530	0.6320

Table 4 displayed indicates that the use of hyper parameters in XGBoost leads to enhanced performance across various evaluation metrics. The application of hyperparameters consistently boosts the results in each test, and they will continue to be utilized as XGBoost integrates ADASYN in subsequent analyses. Before conducting tests with ADASYN, it is necessary to establish the appropriate K value. The comparison of the two models determined that a K value ranging from 6 to 10 is optimal, with the best outcomes observed at a K value of 10. Here are the results from testing the optimal model.

Table 5. The XGBoost test results with ADASYN use the value K=10
XGBoost + ADASYN K=10

Test NO	ACC	Precision	Recall	F1-SCORE	ROC AREA	AUCPR
1	0.9513	0.9513	0.9514	0.9515	0.9930	0.9937
2	0.9530	0.9529	0.9531	0.9537	0.9934	0.9942
3	0.9480	0.9479	0.9481	0.9486	0.9920	0.9928
4	0.9497	0.9497	0.9499	0.9502	0.9921	0.9932
5	0.9465	0.9465	0.9466	0.9471	0.9917	0.9928
6	0.9494	0.9494	0.9495	0.9502	0.9915	0.9927
7	0.9507	0.9506	0.9508	0.9513	0.9914	0.9922
8	0.9481	0.9482	0.9484	0.9483	0.9920	0.9932
9	0.9498	0.9498	0.9499	0.9505	0.9914	0.9926
10	0.9471	0.9472	0.9474	0.9475	0.9905	0.9919
Avg	0.9493	0.9493	0.9495	0.9499	0.9919	0.9930



(a)

(b)

(c)

(d)

Figure 2. Results of Accuracy (a), Precision (b), Recall (c), F1-Score (d), ROC Area (e), and AUCPR from Best Model

Table 5 presents the outcomes from testing XGBoost with ADASYN using a K value of 10, showing metrics such as accuracy at 0.9493, precision at 0.9493, recall at 0.9495, F1-Score at 0.9499, ROC Area at 0.9919, and AUCPR at 0.9930. Applying XGBoost alongside ADASYN in this research data has proven highly effective. The data is visualized according to each evaluation metric to interpret the K value test results better.

Figure 2 shows that searching for the K value in ADASYN increases from K=6 to K=8 but decreases at K=9 and increases significantly at K=10. Apart from that, all evaluations show that with parameter K=10, ADASYN has the highest evaluation level and can be said to be the best model

3.2. Discussion

The problem of class imbalance was much alleviated by using XGBoost in conjunction with ADASYN; this is proven by the model's increased sensitivity to the minority class. The model's prediction accuracy and dependability were both enhanced by this integration, which successfully decreased bias towards the majority class. When compared to earlier research, the XGBoost+ADASYN model performs much better, particularly when dealing with datasets that have a large disparity between the classes. A common obstacle to financial data analysis, customer data imbalance is the target of this integration. The findings of the evaluation show that using ADASYN to balance data classes is beneficial. Because minority classes are frequently disregarded in imbalanced datasets, this technique successfully raises the model's sensitivity to them [41].

We compared the XGBoost+ADASYN model's results to those of prior research to determine the method's usefulness. Table 6 displays the results of the comparison between this research and previous studies.

Table 6. Comparative of Studies

Reference	Algorithms	Results %
Krishna et al [41]	Deep Neural Network	Accuracy 91.15%
Fitriani et al[42]	Random Forest + SMOTE	Accuracy 92.61%
Arifah er al [19]	XGBOOST+SMORE	Accuracy 91.07 %
Gupta et al [2]	Light GBM	Accuracy 91%
Proposed Method	XGBOOST+ ADASYa	Accuracy 94.93%

The suggested model is much better at accuracy and other evaluation factors, as shown in the comparison table. This shows that our combined approach can handle minority class problems in customer data better than other methods. The above comparisons show that the XGBoost+ADASYN mix improves the model's performance, being more accurate and better at dealing with class imbalances than other methods. But the fact that ADASYN depends on carefully tuned parameters shows a flaw in the method; it needs a lot of knowledge and careful setup to make the best synthetic data. To set up this system, you need to know a lot about how data is distributed, and it can be hard to do based on the types of customer data you have. Because of these problems, we need to come up with smarter ways to adjust the model's parameters so that it can be used more often and more effectively. The goals of this study were met by creating a very accurate model. This shows that combining XGBoost and ADASYN is a good way to fix the class imbalance in customer data, giving more correct insights and better prediction performance. We plan to get around the current problems by using automatic tuning methods and trying the model on a wider range of datasets to make sure it can be used in different situations.

4. Conclusion

Our studies on customer data using the XGBoost algorithm model in conjunction with the ADASYN approach have yielded substantial results. The problems caused by extremely unbalanced datasets in bank marketing are effectively handled by the suggested hybrid method. The model's performance on the UCI Bank Marketing dataset was commendable; it earned an accuracy of 94.93%, recall scores of 94.95%, and a ROC-AUC of 0.9919. These measures show that it is far better than the old ways, such as SMOTE and Random Forest. This achievement confirms the importance of the hybrid model for analyzing consumer data and highlights the successful completion of our research goals. Though encouraging, there are a number of caveats that need to be ironed out. Though it does a better job of representing minority classes, the ADASYN technique still adds the possibility of noise from manufactured data. When applied to datasets that have never been seen before, this problem might affect how well the model generalizes. Scalability and ease of implementation in varied real-world contexts are further challenged by the model's reliance on precisely tuned hyper parameters.

In the future, we suggest investigating the potential use of AutoML, a state-of-the-art machine-learning tool that streamlines and automates model creation. Nevertheless, to guarantee the hybrid model's resilience and generalisability, it must be validated across diverse datasets. This research paves the way for more flexible and effective machine-learning solutions in financial data processing by tackling these problems. Banks may be able to improve client targeting tactics, cut marketing expenses, and make data-driven decisions with this innovation. Additional research is needed to limit computational costs and prevent over fitting when using AutoML with data balancing approaches such as ADASYN. It is recommended that future research

References

[1] D. A. Anggoro, “The Implementation of Subspace Outlier Detection in K-Nearest Neighbors to Improve Accuracy in Bank Marketing Data,” Int. J. Emerg. Trends Eng. Res., vol. 8, no. 2, hal. 545–550, Feb 2020, doi: 10.30534/ijeter/2020/44822020.

[2] A. Gupta, A. Raghav, dan S. Srivastava, “Comparative Study of Machine Learning Algorithms for Portuguese Bank Data,” in 2021 International Conference on Computing, Communication, and Intelligent Systems (ICCCIS), Feb 2021, hal. 401–406, doi: 10.1109/ICCCIS51004.2021.9397083.

[3] Y. Chen dan K. Du, “The role of information disclosure in financial intermediation with investment risk,” J. Financ. Stab., vol. 46, hal. 100720, Feb 2020, doi: 10.1016/j.jfs.2019.100720.

[4] A. K. Kashyap, R. Rajan, dan J. C. Stein, “Banks as Liquidity Providers: An Explanation for the Coexistence of Lending and Deposit-taking,” J. Finance, vol. 57, no. 1, hal. 33–73, Feb 2002, doi: 10.1111/1540-6261.00415.

[5] D. Broby, “Financial technology and the future of banking,” Financ. Innov., vol. 7, no. 1, hal. 47, Des 2021, doi: 10.1186/s40854-021-00264-y.

[6] M. A. T. Rony, M. M. Hassan, E. Ahmed, A. Karim, S. Azam, dan D. S. A. A. Reza, “Identifying Long-Term Deposit Customers: A Machine Learning Approach,” in 2021 2nd International Informatics and Software Engineering Conference (IISEC), Des 2021, hal. 1–6, doi: 10.1109/IISEC54230.2021.9672452

[7] I. M. Putra, I. Tahyudin, H. A. A. Rozaq, A. Y. Syafa’At, R. Wahyudi, dan E. Winarto, “Classification analysis of COVID19 patient data at government hospital of banyumas using machine learning,” in 2021 2nd International Conference on Smart Computing and Electronic Enterprise: Ubiquitous, Adaptive, and Sustainable Computing Solutions for New Normal, ICSCEE 2021, Jun 2021, hal. 271–274, doi: 10.1109/ICSCEE50312.2021.9498020.

[8] I. Tahyudin, H. A. A. Rozaq, dan H. Nambo, "Machine Learning Analysis for Temperature Classification using Bioelectric Potential of Plant," in 2022 6th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE), Des 2022, hal. 465–470, doi: 10.1109/ICITISEE57756.2022.10057768.

[9] M. Mahmud et al., "Implementation of C5.0 Algorithm using Chi-Square Feature Selection for Early Detection of Hepatitis C Disease," J. Electron. Electromed. Eng. Med. Informatics, vol. 6, no. 2, hal. 116–124, Mar 2024, doi: 10.35882/jeeemi.v6i2.384.

[10] D. Datta et al., "A Hybrid Classification of Imbalanced Hyperspectral Images Using ADASYN and Enhanced Deep Subsampled Multi-Grained Cascaded Forest," Remote Sens., vol. 14, no. 19, hal. 4853, Sep 2022, doi: 10.3390/rs14194853.

[11] C. Lu, S. Lin, X. Liu, dan H. Shi, "Telecom Fraud Identification Based on ADASYN and Random Forest," in 2020 5th International Conference on Computer and Communication Systems (ICCCS), Mei 2020, hal. 447–452, doi: 10.1109/ICCCS49078.2020.9118521.

[12] X. Gu, P. P. Angelov, dan E. A. Soares, "A self-adaptive synthetic over-sampling technique for imbalanced classification," Int. J. Intell. Syst., vol. 35, no. 6, hal. 923–943, Jun 2020, doi: 10.1002/int.22230.

[13] C. Kaope dan Y. Pristyanto, "The Effect of Class Imbalance Handling on Datasets Toward Classification Algorithm Performance," MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput., vol. 22, no. 2, hal. 227–238, Mar 2023, doi: 10.30812/matrik.v22i2.2515.

[14] I. K. Nti dan A. R. Somanathan, "A Scalable RF-XGBoost Framework for Financial Fraud Mitigation," IEEE Trans. Comput. Soc. Syst., vol. 11, no. 2, hal. 1556–1563, Apr 2024, doi: 10.1109/TCSS.2022.3209827.

[15] D. A. Anggoro dan S. S. Mukti, "Performance Comparison of Grid Search and Random Search Methods for Hyperparameter Tuning in Extreme Gradient Boosting Algorithm to Predict Chronic Kidney Failure," Int. J. Intell. Eng. Syst., vol. 14, no. 6, hal. 198–207, Des 2021, doi: 10.22266/ijies2021.1231.19.

[16] G. Menghani, "Efficient Deep Learning: A Survey on Making Deep Learning Models Smaller, Faster, and Better," ACM Comput. Surv., vol. 55, no. 12, hal. 1–37, Des 2023, doi: 10.1145/3578938.

[17] I. Kawina, K. Amarendra, dan B. Marapelli, "Deep Learning and Machine Learning Approach to Breast Cancer Classification with Random Search Hyperparameter Tuning," Int. J. Intell. Syst. Appl. Eng., vol. 12, no. 16s, hal. 264–275, 2024.

[18] R.-C. Chen, C. Dewi, S.-W. Huang, dan R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," J. Big Data, vol. 7, no. 1, hal. 52, Des 2020, doi: 10.1186/s40537-020-00327-4.

[19] D. Arifah, T. Hamonangan Saragih, D. Kartini, dan M. Itqan Mazdadi, "Application of SMOTE to Handle Imbalance Class in Deposit Classification Using the Extreme Gradient Boosting Algorithm," J. Ilm. Tek. Elektro Komput. dan Inform., vol. 9, no. 2, hal. 396–410, 2023, doi: 10.26555/jiteki.v9i2.26155.

[20] S. E. Saeed, M. Hammad, dan A. Alqaddoumi, "Predicting Customer's Subscription Response to Bank Telemarketing Campaign Based on Machine learning Algorithms," in 2022 International Conference on Decision Aid Sciences and Applications (DASA), Mar 2022, hal. 1474–1478, doi: 10.1109/DASA54658.2022.9765152.

[21] S. Boeschoten, C. Catal, B. Tekinerdogan, A. Lommen, dan M. Blokland, "The automation of the development of classification models and improvement of model quality using feature engineering techniques," Expert Syst. Appl., vol. 213, hal. 118912, Mar 2023, doi: 10.1016/j.eswa.2022.118912.

[22] R. P. Henessa, M. A.-F. Fisabilillah, dan W. R. Azizah, "Detection of Public Sentiment Analysis Model on the Implementation of PPKM in Indonesia," Proc. Int. Conf. Data Sci. Off. Stat., vol. 2021, no. 1, hal. 289–295, Jan 2022, doi: 10.34123/icdsos.v2021i1.237.

[23] C. M. Sitorus, A. Rizal, dan M. Jajuli, "Prediksi Risiko Perjalanan Transportasi Online Dari Data Telematik Menggunakan Algoritma Support Vector Machine," J. Tek. Inform. dan Sist. Inf., vol. 6, no. 2, Agu 2020, doi: 10.28932/jutisi.v6i2.2672.

[24] Nayan Kumar Sinha, "Developing A Web based System for Breast Cancer Prediction using XGboost Classifier," Int. J. Eng. Res., vol. V9, no. 06, Jun 2020, doi: 10.17577/IJERTV9IS060612.

[25] D. A. Rusdah dan H. Murfi, "XGBoost in handling missing values for life insurance risk prediction," *SN Appl. Sci.*, vol. 2, no. 8, hal. 1336, Agu 2020, doi: 10.1007/s42452-020-3128-y

[26] P. Cerda dan G. Varoquaux, "Encoding High-Cardinality String Categorical Variables," IEEE Trans. Knowl. Data Eng., vol. 34, no. 3, hal. 1164–1176, Mar 2022, doi: 10.1109/TKDE.2020.2992529.

[27] M. Heidari, S. Mirniaharikandehei, A. Z. Khuzani, G. Danala, Y. Qiu, dan B. Zheng, "Improving the performance of CNN to predict the likelihood of COVID-19 using chest X-ray images with preprocessing algorithms," Int. J. Med. Inform., vol. 144, hal. 104284, Des 2020, doi: 10.1016/j.ijmedinf.2020.104284.

[28] H. Wei, C. Hu, S. Chen, Y. Xue, dan Q. Zhang, "Establishing a software defect prediction model via effective dimension reduction," Inf. Sci. (Ny)., vol. 477, hal. 399–409, Mar 2019, doi: 10.1016/j.ins.2018.10.056.

[29] L. A. Yates, Z. Aandahl, S. A. Richards, dan B. W. Brook, "Cross validation for model selection: A review with examples from ecology," Ecol. Monogr., vol. 93, no. 1, Feb 2023, doi: 10.1002/ecm.1557.

[30] A. El Hariri, M. Mouiti, O. Habibi, dan M. Lazaar, "Improving Deep Learning Performance Using Sampling Techniques for IoT Imbalanced Data," Procedia Comput. Sci., vol. 224, hal. 180–187, 2023, doi: 10.1016/j.procs.2023.09.026.

[31] Z. Wang dan H. Wang, "Global Data Distribution Weighted Synthetic Oversampling Technique for Imbalanced Learning," IEEE Access, vol. 9, hal. 44770–44783, 2021, doi: 10.1109/ACCESS.2021.3067060.

[32] A. M. Halim, M. Dwifebri, dan F. Nhita, "Handling Imbalanced Data Sets Using SMOTE and ADASYN to Improve Classification Performance of Ecoli Data Sets," Build. Informatics, Technol. Sci., vol. 5, no. 1, Jun 2023, doi: 10.47065/bits.v5i1.3647.

[33] S. K. Kiangala dan Z. Wang, "An effective adaptive customization framework for small manufacturing plants using extreme gradient boosting-XGBoost and random forest ensemble learning algorithms in an Industry 4.0 environment," Mach. Learn. with Appl., vol. 4, hal. 100024, Jun 2021, doi: 10.1016/j.mlwa.2021.100024.

[34] H. Mo, H. Sun, J. Liu, dan S. Wei, "Developing window behavior models for residential buildings using XGBoost algorithm," Energy Build., vol. 205, hal. 109564, Des 2019, doi: 10.1016/j.enbuild.2019.109564.

[35] A. Paleczek, D. Grochala, dan A. Rydosz, "Artificial Breath Classification Using XGBoost Algorithm for Diabetes Detection," Sensors, vol. 21, no. 12, hal. 4187, Jun 2021, doi: 10.3390/s21124187.

[36] M. K. Hasan, M. A. Alam, D. Das, E. Hossain, dan M. Hasan, “Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers,” IEEE Access, vol. 8, hal. 76516–76531, 2020, doi: 10.1109/ACCESS.2020.2989857.

[37] A. Verma, “Evaluation of Classification Algorithms with Solutions to Class Imbalance Problem on Bank Marketing Dataset using WEKA,” Int. Res. J. Eng. Technol., vol. 6, no. 3, hal. 54–60, 2019..

[38] N. Hasan dan Y. Bao, “Comparing different feature selection algorithms for cardiovascular disease prediction,” Health Technol. (Berl)., vol. 11, no. 1, hal. 49–62, Jan 2021, doi: 10.1007/s12553-020-00499-2.

[39] S. Goel, R. Agrawal, dan R. K. Bharti, “Automated detection of epileptic EEG signals using recurrence plots-based feature extraction with transfer learning,” Soft Comput., vol. 28, no. 3, hal. 2367–2383, Feb 2024, doi: 10.1007/s00500-023-08386-4.

[40] M. Guo, Z. Yuan, B. Janson, Y. Peng, Y. Yang, dan W. Wang, “Older Pedestrian Traffic Crashes Severity Analysis Based on an Emerging Machine Learning XGBoost,” Sustainability, vol. 13, no. 2, hal. 926, Jan 2021, doi: 10.3390/su13020926.

[41] C. L. Krishna dan P. V. S. Reddy, “Deep neural networks for the classification of bank marketing data using data reduction techniques,” Int. J. Recent Technol. Eng., vol. 8, no. 3, hal. 4373–4378, Sep 2019, doi: 10.35940/ijrte.C5522.098319.

[42] M. A. Fitriani dan D. C. Febrianto, “Data Mining for Potential Customer Segmentation in the Marketing Bank Dataset,” JUITA J. Inform., vol. 9, no. 1, hal. 25, Mei 2021, doi: 10.30595/juita.v9i1.7983.