MORTALITA

Vol. 2 No.1, Juny 2025, Pages 1-14
Mortalita: Journal of Mathematics and Its Applications
e-ISSN 3088-9812
https://ejournal.darunnajah.ac.id/index.php/mortalita

Implementation of Smote, Random Oversampling and Random Undersampling in Random Forest and Lasso Logistic Regression for Customer Churn Prediction

[1]Muhamad Hilman Rizaldi*, [2]Kenia Anindya Panonsih

[1]*Brawijaya University, Malang, Indonesia*
[2]*Brawijaya University, Malang, Indonesia*
*Email Corresponding author: hilman_rizaldi@student.ub.ac.id

ABSTRACT

Predicting customer churn remains a critical challenge in the banking industry, as retaining existing customers is generally more cost-effective than acquiring new ones. This study addresses the issue of data imbalance in churn prediction by applying resampling techniques, namely Synthetic Minority Oversampling Technique (SMOTE), Random Over-Sampling (ROS), and Random Under-Sampling (RUS). The dataset comprises 10,000 customer records with 14 attributes, analyzed using Random Forest and LASSO Logistic Regression algorithms. Model performance was evaluated using accuracy, precision, recall, and F1-score metrics. The results indicate that Random Forest combined with ROS achieved the highest accuracy (86%), although the churn recall remained low (0.50). SMOTE yielded an accuracy of 82% with a more balanced recall (0.62), while RUS achieved an accuracy of 79% and the highest recall (0.78), albeit at the expense of precision. For LASSO Logistic Regression, SMOTE provided the best results with 73% accuracy and 0.64 recall, whereas both ROS and RUS achieved 71% accuracy with a recall of 0.72. The findings highlight the effectiveness of oversampling techniques in enhancing churn detection, providing practical insights for banking institutions to improve customer retention strategies.

Keywords: *Customer Churn, LASSO Logistic Regression, Machine Learning, Random Forest, Resampling Technique*

1. INTRODUCTION

In the banking industry, professional customer service alone is insufficient; trust is equally essential, as banking is fundamentally a business that sells trust to the public. Marketing practices have recently shifted from a transaction-based orientation to one that emphasizes relationship quality, highlighting the importance of building long-term relationships with customers. Efforts to maintain these relationships are referred to as customer retention. Achieving effective customer retention requires adequate support to ensure satisfactory outcomes, with customer satisfaction serving as the key determinant [13]. Customer churn, defined as the transfer of customers from one financial institution to another, has become a pressing issue due to its potential negative impact on banks. For financial institutions, customer churn represents a significant financial loss that may undermine overall business performance [16]. Predicting the likelihood of customer attrition is therefore crucial, as acquiring new customers generally incurs costs five to ten times higher than retaining existing ones. Accurate churn prediction enables banks to design targeted marketing strategies tailored to specific customer characteristics while addressing their individual needs. Moreover, such predictions allow banks to implement proactive retention campaigns to reduce the risk of losing valuable customers to competitors [8].

Machine learning is a branch of artificial intelligence and computer science that focuses on utilizing data and algorithms to mimic human learning processes, thereby improving accuracy over time [2]. By leveraging machine learning methods, companies can predict the likelihood of customer churn, enabling early preventive measures to be implemented [11]. In the banking sector, one of the primary challenges in churn prediction is data imbalance, as the majority of customers remain loyal while only a small

proportion discontinue their services. This condition can cause prediction models to become biased and less sensitive in identifying customers who are likely to leave the bank.

Previous studies applying machine learning methods, including Random Forest, LASSO Logistic Regression, and Support Vector Machine (SVM) with random undersampling, random oversampling, and SMOTE techniques, demonstrated that Random Forest was the optimal classification method. Meanwhile, SVM showed lower overall performance compared to Random Forest and LASSO Logistic Regression [7]. Another study revealed that a hybrid approach combining K-Means and Support Vector Machine (SVM) with imbalanced data handling through Random Oversampling proved to be an effective and efficient solution for addressing class imbalance problems [19].

This study applies Random Oversampling (ROS), Random Undersampling (RUS), and SMOTE methods to address data imbalance issues in churn prediction, where the number of retained customers (non-churn) significantly exceeds those who discontinue their services. This research utilizes the Confusion Matrix as the primary evaluation method to measure accuracy, precision, recall, and F1-score of the developed models. The use of the Confusion Matrix also aims to enable direct comparison with previous studies that employed similar methods in evaluating model performance. Through this approach, the research is expected to provide a more comprehensive understanding of the advantages and limitations of the applied methods [3]. The implementation of these techniques is anticipated to contribute positively to model effectiveness in detecting churn patterns.

2. METHODS

1. Dataset

This study uses secondary data from the Churn Bank Customers AK dataset, sourced from the public Kaggle repository at https://www.kaggle.com/datasets/akelsayed/churn-bank-customers-ak. The dataset comprises 10,000 records with 14 variables, divided into 13 features and 1 target class. Data collection was conducted with careful consideration to ensure that the obtained data is sufficiently representative to develop an accurate prediction model.

Table 1. Feature Dataset.

No.	Atribut	Description
1	RowNumber	Sequential ID of each row.
2	CustomerId	Unique ID for each customer.
3	Surname	Customer's surname.
4	CreditScore	Customer’s credit score.
5	Geography	Country of residence.
6	Gender	Gender of the customer.
7	Age	Age of the customer.
8	Tenure	Number of years the customer has been with the bank.
9	Balance	Account balance.
10	NumOfProducts	The number of bank products the customer uses.
11	HasCrCard	Indicator (1 or 0) if the customer has a credit card.
12	IsActiveMember	Indicate if the customer is an active bank member.
13	EstimatedSalary	Customer’s estimated salary.

Table 2. Dataset class.

No.	Atribut	Description
1	Exited	Target variable indicating if the customer left the bank (1 = exited, 0 = retained).

The "Exited" class serves as the target attribute in the dataset used to predict whether a customer will continue using banking services or discontinue their relationship with the bank. The analysis steps conducted in this study are as follows:

2. Preprocessing Data.

   Preprocessing is a collection of techniques applied to datasets to eliminate noise, missing values, and other errors, ensuring the data is ready for further processing [16].

   a. Missing Value Detection and Handling
      The dataset used in this study underwent verification to ensure no missing values were present in any attribute. The inspection process was conducted using specific functions, and the results indicated that all columns, including key characteristics such as age, gender, balance, and exit status, contained complete data. Therefore, no imputation or data removal processes were required to handle missing values. The dataset was declared clean and ready for subsequent data preprocessing stages.

   b. Encoding Labels

      In this study, categorical data were converted to a numerical format by assigning distinct numerical codes to each category. For example, in the Gender column, the "Female" category was assigned label 0, and "Male" was assigned label 1. Meanwhile, the Geography column contained categories such as "France," "Germany," and "Spain." These categories were transformed using the Label Encoding method by assigning appropriate numerical labels: France = 0, Germany = 1, and Spain = 2. This preprocessing step facilitates efficient data processing by machine learning algorithms.

3. Data Splitting

   At this stage, the dataset was divided into two subsets: 70% for training, used to develop the models, and 30% for testing, reserved for performance evaluation.

4. Data Resampling

Data resampling was applied to address class imbalance, implementing both oversampling and undersampling approaches. The oversampling techniques included Random Over-Sampling (ROS) and the Synthetic Minority Over-Sampling Technique (SMOTE), while undersampling was performed using Random Under-Sampling (RUS).

a. Random Oversampling (ROS) works by randomly duplicating data samples from the minority class and adding them to the training dataset. ROS increases the training dataset size by replicating original samples until the class distribution becomes balanced [9].
b. Random Undersampling (RUS) handles imbalanced data by randomly reducing the number of samples from the majority class while maintaining all samples from the minority class, thereby achieving class balance through sample reduction.
c. Synthetic Minority Oversampling Technique (SMOTE) creates new samples by interpolating minority samples [17]. Like ROS, SMOTE also increases the size and variation of the training dataset by generating synthetic samples within the training dataset through interpolation between existing data points in the minority class that are adjacent to each other [4].

5. Machine Learning Classification Model Development

Classification models were developed using Random Forest and LASSO Logistic Regression algorithms.

a. Random Forest.

   Random Forest is constructed using the bagging method with random attributes, which is an

extension of the Decision Tree method. This method is considered one of the best-performing machine learning algorithms, where decision trees are grown to their maximum size without pruning, using the CART (Classification and Regression Tree) method [15]. Subsequently, a collection of trees is formed, referred to as a forest. Random Forest is a classifier consisting of a collection of tree classifiers $\{h(x, Sb), b = 1, \ldots, B\}$ where $\{Sb\}$ are independent and identically distributed random vectors. The Random Forest algorithm construction process is as follows [6]:

1. Approximately one-third of the sample data is not used when forming bootstrap samples with replacement for each decision tree.
2. This unused data is called Out of Bag (OOB).
3. Each decision tree in the forest has its own OOB data, which is utilized to calculate the error rate of that tree. This estimate is known as the OOB error. Additionally, Random Forest is capable of measuring the importance level of each variable and making predictions. Missing values and outliers can be replaced using predictions generated by the model.

b. LASSO Logistic Regression

Logistic regression is used to analyze the relationship between a binary response variable Y and one or more explanatory variables. In binary logistic regression, the response variable has two possibilities: y=1 representing success and y=0 representing failure [1]. Parameter estimation in logistic regression is obtained by maximizing the log-likelihood function as follows [12]:

$$\begin{aligned} l(\beta) &= \sum_{i=1}^{n} \quad [y_i \, log(\pi_i) + (1 - y_i) log(1 - \pi_i)] \\ &= \sum_{i=1}^{n} \quad \left[y_i \, log\left(\frac{\pi_i}{1-\pi_i}\right) + log(1 - \pi_i)\right] \\ &= \sum_{i=1}^{n} \quad \left[y_i x_i \beta + log\left(1 + e^{x_i \beta}\right)\right] \end{aligned} \tag{1}$$

To transform the logistic regression model above into a LASSO logistic regression model, a L_1 constraint is applied to the parameters (β) by minimizing the negative log-likelihood function as follows (Friedman et al., 2010):

$$\hat{\beta}_{LASSO} = \min_{\beta} \left(\sum_{i=1}^{n} \quad \left[log\left(1 + e^{x_i \beta}\right) - y_i(\beta_0 + x_i \beta)\right] + \lambda \sum_{j=1}^{p} \quad |\beta_j| \right) \tag{2}$$

subject to $\sum_{j=1}^{p} \quad |\beta_j| \leq \lambda$ and $\lambda > 0$. The selection of λ value is performed implicitly through the regularization parameter in the Logistic Regression function in scikit-learn with penalty='l1' and solver='liblinear'. The model is then trained using oversampled or undersampled data to obtain the LASSO logistic regression estimator [10].

6. Model Evaluation

In this study, the trained models were evaluated through assessment stages that included accuracy score calculations and F1-score measurements [14]. Accuracy was determined by comparing the number of correct predictions for both positive and negative classes against the total number of data points. Meanwhile, the F1-score indicates that the model has good levels of precision and recall.

Table 3. Confusion Matrix.

	Actual Positive	Actual Negative
Predicted Positive	True Positive	False Positive

Predicted Negative	False Negative	True Negative

a. Accuracy (Akurasi)

Accuracy represents the level of correctness of the classification model used.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (3)$$

b. Precision (Presisi)

Precision measures the proportion of correct exited predictions from all exited predictions.

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

c. Recall

Recall indicates the classification model's ability to retrieve correct information related to “exited” cases.

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

d. F1 Score

F1-score is a measurement that combines precision and recall values into a single evaluation metric.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (6)$$

e. Model Evaluation Metrics Comparison

The process of comparing predictive model performance using various evaluation metrics. This step is essential for understanding how the model processes data and assessing whether the model has met analytical objectives, particularly in predicting customers who exited.

3. RESULTS AND DISCUSSION

This section discusses the experimental results obtained using the Google Colab application by developing models using oversampling and undersampling techniques. The modeling was conducted using various classification algorithms for customer churn prediction in the banking sector.

1. Dataset

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
1	15634602	Hargrave	619	France	Female	42	2	0.0	1	1	1	101348.88	1
2	15647311	Hill	608	Spain	Female	41	1	83807.86	1	0	1	112542.58	0
3	15619304	Onio	502	France	Female	42	8	159660.8	3	1	0	113931.57	1
4	15701354	Boni	699	France	Female	39	1	0.0	2	0	0	93826.63	0
5	15737888	Mitchell	850	Spain	Female	43	2	125510.82	1	1	1	79084.1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---
9996	15606229	Obijiaku	771	France	Male	39	5	0.0	2	1	0	96270.64	0
9997	15569892	Johnstone	516	France	Male	35	10	57369.61	1	1	1	101699.77	0
9998	15584532	Liu	709	France	Female	36	7	0.0	1	0	1	42085.58	1
9999	15682355	Sabbatini	772	Germany	Male	42	3	75075.31	2	1	0	92888.52	1
10000	15628319	Walker	792	France	Female	28	4	130142.79	1	1	0	38190.78	0

Figure 1. Dataset.

2. Preprocessing Data

a. Missing Value Inspection and Handling

RowNumber	0
CustomerId	0
Surname	0
CreditScore	0
Geography	0
Gender	0
Age	0
Tenure	0
Balance	0
NumOfProducts	0
HasCrCard	0
IsActiveMember	0
EstimatedSalary	0
Exited	0

Figure 2. Missing value.

b. Encoding Labels

RowNumber	CustomerId	Surname	CreditScore	Geography	Gender	Age	Tenure	Balance	NumOfProducts	HasCrCard	IsActiveMember	EstimatedSalary	Exited
1	15634602	1115	619	0	0	42	2	0.0	1	1	1	101348.88	1
2	15647311	1177	608	2	0	41	1	83807.86	1	0	1	112542.58	0
3	15619304	2040	502	0	0	42	8	159660.8	3	1	0	113931.57	1
4	15701354	289	699	0	0	39	1	0.0	2	0	0	93826.63	0
5	15737888	1822	850	2	0	43	2	125510.82	1	1	1	79084.1	0
---	---	---	---	---	---	---	---	---	---	---	---	---	---
9996	15606229	1999	771	0	1	39	5	0.0	2	1	0	96270.64	0
9997	15569892	1336	516	0	1	35	10	57369.61	1	1	1	101699.77	0
9998	15584532	1570	709	0	0	36	7	0.0	1	0	1	42085.58	1
9999	15682355	2345	772	1	1	42	3	75075.31	2	1	0	92888.52	1
10000	15628319	2751	792	0	0	28	4	130142.79	1	1	0	38190.78	0

Figure 3. Encoding Labels.

3. Data Splitting

In the data splitting stage, the dataset was divided into two main subsets using stratified sampling: the training set and the testing set. The data splitting proportion followed a 70:30 ratio, with 70% of the data allocated for model training and the remaining 30% reserved for testing to evaluate generalization performance on unseen data. This division aims to ensure the validity of evaluation results and prevent overfitting in the model. Detailed characteristics, including the number of features, target variables, and class distribution, for each data subset are presented in Table 4.

Table 4. Split Data.

Dataset	Target Amount	Target Distribution (0)	Target Distribution (1)
Training Data (70%)	7000	5574	1426
Test Data (30%)	3000	2389	611

4. Resampling Data

a. Oversampling Technique

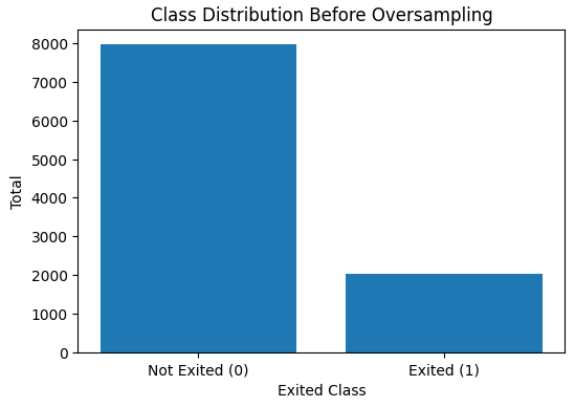

Figure 4. Class Distribution before Oversampling.

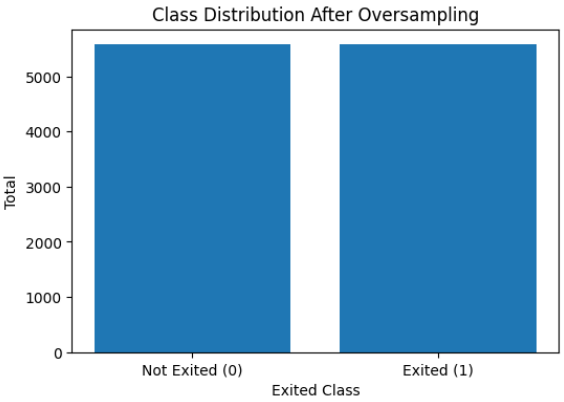

Figure 5. Class Distribution after Oversampling.

b. Undersampling technique

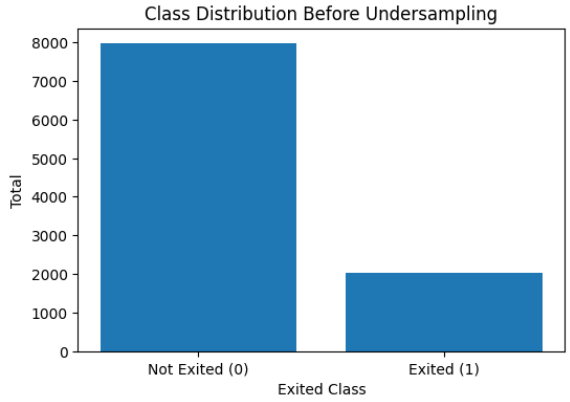

Figure 6. Class Distribution before Undersampling.

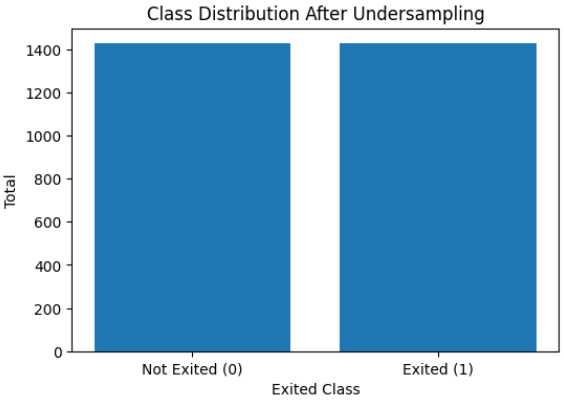

Figure 7. Class Distribution after Undersampling.

5. Machine Learning Classification Algorithms

This study employed two machine learning algorithms to build classification models: Random Forest and Logistic Regression with L1 regularization (LASSO). Random Forest was chosen for its ability to handle non-linear relationships and complex feature interactions, while Logistic Regression with LASSO was selected for its effectiveness in feature selection and interpretability. These complementary characteristics justify the use of both algorithms in this study.

a. Random Forests

Random Forest is an ensemble learning algorithm that integrates multiple decision trees to generate more robust and accurate predictions. In this method, each tree is constructed from bootstrapped samples of the training data and random subsets of features. At the same time, the final prediction is determined through majority voting across the trees. The main advantages of Random Forest include its ability to mitigate overfitting in training data and its effectiveness when dealing with a large number of input variables. In this study, the Random Forest model was implemented using default parameters, with the *random_state* set to 42 to ensure reproducibility of the results.

b. LASSO Logistic Regression

Logistic regression is a statistical method used to model the relationship between predictor variables and a binary outcome. In this study, logistic regression with an L1 penalty (LASSO) was applied. The L1 penalty performs regularization by shrinking regression coefficients toward zero, and in some cases, eliminating irrelevant predictors from the model. Consequently, LASSO logistic regression not only provides a classification model but also facilitates feature selection, leading to a simpler and more interpretable model. For implementation, the *liblinear solver* was utilized since it supports the L1 penalty, with *random_state* = 42 specified to ensure reproducibility of the results.

6. Model Evaluation

The performance evaluation results of the Random Forest algorithm, both before and after the application of oversampling techniques, are presented in the following table. The oversampling methods employed in this study were Random Oversampling and SMOTE.

Table 5. Random Forest Algorithm Evaluation With SMOTE

Random Forest with Oversampling Method: SMOTE				
Label	Precision	Recall	F1 Score	Accuracy
0	0.90	0.88	0.89	0.82
1	0.56	0.62	0.59	

Table 6. Random Forest Algorithm Evaluation With ROS

Random Forest with Oversampling Method: ROS				
Label	Precision	Recall	F1 Score	Accuracy
0	0.88	0.95	0.91	0.86
1	0.71	0.50	0.58	

Table 7. Evaluation of Random Forest Algorithm With Undersampling

Random Forest with Undersampling Method: Random Undersampling				
Label	Precision	Recall	F1 Score	Accuracy
0	0.93	0.80	0.86	0.79
1	0.50	0.78	0.61	

The evaluation results show that within the Random Forest algorithm, Random Oversampling (ROS) achieved the highest accuracy of 0.86, indicating that balancing the class distribution through duplication of minority-class samples can improve the model's ability to predict churn. In contrast,

SMOTE produced a slightly lower accuracy of 0.82 but yielded higher recall for the churn class, making the model more sensitive to detecting customers likely to leave, though at the cost of increased misclassifications.

For the LASSO Logistic Regression algorithm, SMOTE achieved the best accuracy of 0.73, outperforming ROS, which reached 0.71. Similar to Random Forest, SMOTE also improved recall for the churn class (0.64), although its precision remained relatively low (0.40), indicating a high rate of false positives. Conversely, ROS provided a more balanced trade-off between recall (0.72) and accuracy (0.71), though precision was similarly low (0.38).

Overall, these findings suggest that Random Forest with ROS represents the most effective combination when the primary objective is maximizing accuracy. However, if the focus is on improving churn detection (recall), SMOTE is more suitable for both Random Forest and LASSO Logistic Regression, despite its relatively lower accuracy.

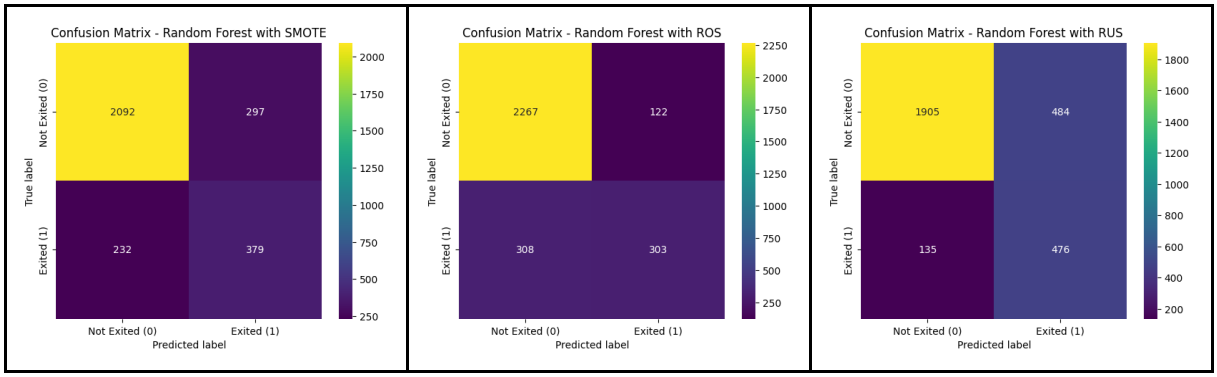

Figure 8. Confusion Matrix Random Forest Algorithm

The performance evaluation results of the LASSO algorithm, both before and after the application of oversampling techniques, are presented in the following table. The oversampling methods employed in this study were Random Oversampling (ROS) and SMOTE.

Table 8. Evaluation of the LASSO Logistic Regression Algorithm with SMOTE

LASSO Logistic Regression with Oversampling Method: SMOTE				
Label	Precision	Recall	F1 Score	Accuracy
0	0.89	0.76	0.82	0.73
1	0.40	0.64	0.49	

Table 9. Evaluation of LASSO Logistic Regression Algorithm With ROS

LASSO Logistic Regression with Oversampling Method: ROS				
Label	Precision	Recall	F1 Score	Accuracy
0	0.91	0.70	0.79	0.71
1	0.38	0.72	0.50	

Table 10. Evaluation of the LASSO Logistic Regression Algorithm with Undersampling

LASSO Logistic Regression with Undersampling Method: Random Undersampling				
Label	Precision	Recall	F1 Score	Accuracy
0	0.91	0.70	0.79	0.71
1	0.38	0.72	0.50	

The LASSO Logistic Regression model with SMOTE achieved the highest accuracy of 73%. This model performed well in predicting the majority class (non-churn), with a precision of 0.89 and recall of 0.76. However, while recall for the minority class (churn) improved to 0.64, the relatively low precision of 0.40 resulted in a higher rate of false positives.

With Random Oversampling (ROS), model accuracy slightly decreased to 71%. Although recall for churn increased to 0.72, precision declined to 0.38, indicating more frequent misclassification of non-churn customers as churn.

A similar pattern was observed with Random Undersampling (RUS), which also yielded an accuracy of 71%. The model maintained churn recall at 0.72, but precision remained low (0.38), reinforcing the trade-off between recall improvement and overall prediction accuracy.

Overall, while SMOTE produced the highest accuracy among the resampling methods, ROS and RUS proved more effective in enhancing churn detection (recall), albeit at the expense of accuracy and precision.

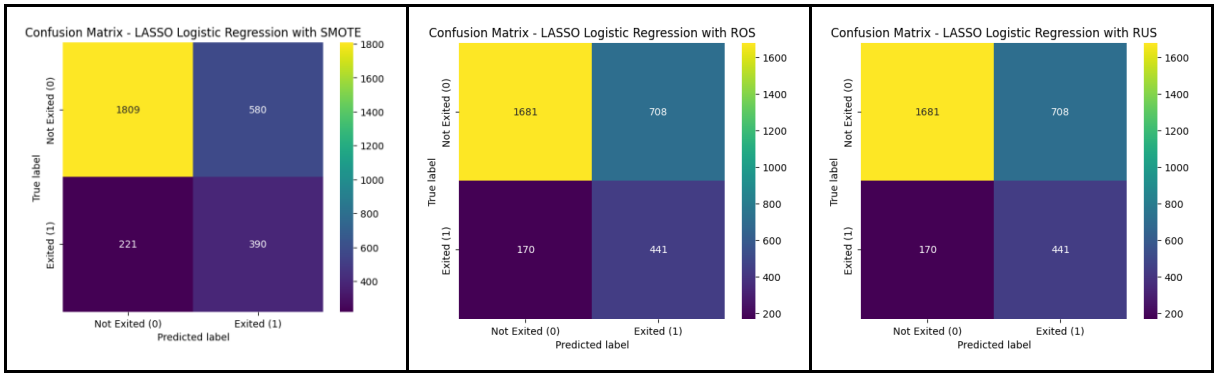

Figure 9. Confusion Matrix LASSO Logistic Regression Algorithm

C. Comparison of Model Evaluation Metrics

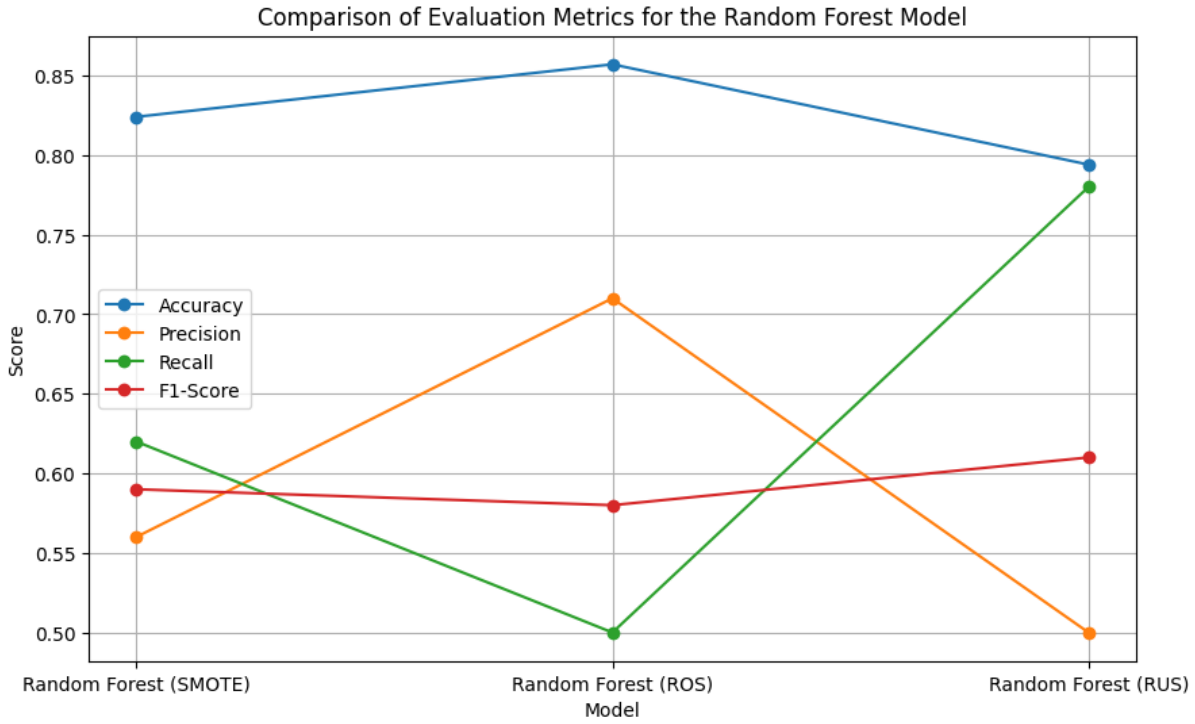

Figure 10. Comparison of Random Forest Model Evaluation Metrics with Resampling

Based on Figure 10, the performance evaluation of the Random Forest model using three resampling methods (Synthetic Minority Oversampling Technique (SMOTE), Random Over-Sampling (ROS), and Random Under-Sampling (RUS)) is summarized as follows.

- SMOTE produced relatively high accuracy with balanced precision and recall. However, the F1-Score remained at a moderate level, suggesting that the model still faced challenges in maintaining overall consistency across both classes.
- ROS achieved the highest accuracy among the three methods. Precision improved significantly, but recall decreased, which led to the lowest F1-Score. This result indicates that the model became better at recognizing the majority class but less effective at detecting the minority class.
- RUS showed a different trend. Recall increased sharply, approaching the accuracy value, while precision dropped substantially. Despite the reduction in precision, the F1-Score was higher than that of SMOTE and ROS, indicating a more balanced trade-off between precision and recall.

Overall, ROS excelled in terms of accuracy, while RUS provided a more balanced performance for both classes, as reflected by its better recall and F1-Score.

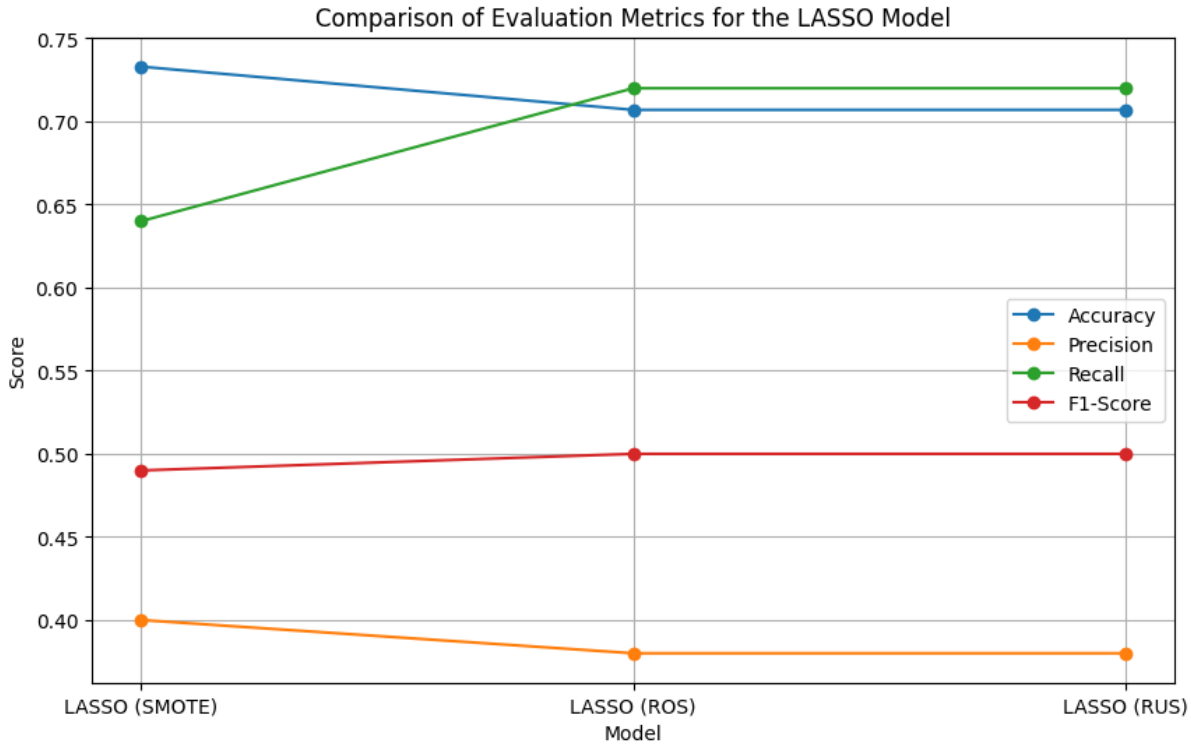

Figure 11. Comparison of Evaluation Metrics of LASSO Logistic Regression Model with Resampling

Based on Figure 11, the evaluation of the LASSO model was conducted using four performance metrics—accuracy, precision, recall, and F1-Score—across three resampling methods: Synthetic Minority Oversampling Technique (SMOTE), Random Over-Sampling (ROS), and Random Under-Sampling (RUS).

- SMOTE yielded the highest accuracy; however, precision and F1-Score remained relatively low, while recall was not yet optimal.
- ROS improved recall considerably and slightly increased the F1-Score, although its accuracy declined marginally compared to SMOTE.
- RUS produced results similar to ROS, maintaining high recall and stable F1-Score, albeit with lower precision.

Overall, both ROS and RUS demonstrated more balanced performance in detecting the minority class compared to SMOTE, though this improvement came at the cost of reduced accuracy.

CONCLUSION

This study successfully addressed the issue of data imbalance in banking customer churn prediction by applying resampling techniques—Random Oversampling (ROS), SMOTE, and Random Undersampling (RUS)—in combination with Random Forest and LASSO Logistic Regression algorithms. The Random Forest model with ROS achieved the highest accuracy (86%), although it showed relatively low recall for churn (0.50). Meanwhile, SMOTE produced a more balanced outcome, with 82% accuracy and 0.62 recall. For LASSO Logistic Regression, SMOTE yielded the best overall performance, achieving 73% accuracy. These results highlight that the choice of resampling technique should be aligned with business priorities: ROS is suitable when maximizing accuracy is the primary goal, SMOTE is preferable for achieving a balance across evaluation metrics, and RUS offers advantages when the focus is on maximizing churn detection sensitivity. Nevertheless, the study has limitations, particularly in the LASSO implementation, where ROS and RUS produced identical results, potentially due to parameter tuning issues related to the penalty term. Future research could expand by

incorporating other machine learning algorithms such as Support Vector Machine, XGBoost, or Neural Networks to provide a broader comparative perspective.

REFERENCES

[1] A. Agresti, Categorical Data Analysis (Hoboken, 2002).

[2] A. Ahmad, Mengenal Artificial Intelligence, Machine Learning, Neural Network, dan Deep Learning, Jurnal Teknologi Indonesia 3 (2017).

[3] A. F. Anjani, D. Anggraeni, and I. M. Tirta, Implementasi Random Forest Menggunakan SMOTE untuk Analisis Sentimen Ulasan Aplikasi Sister for Students UNEJ, Jurnal Nasional Teknologi Dan Sistem Informasi 9 (2023) 163-172.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, SMOTE: Synthetic Minority Over-sampling Technique, Journal of Artificial Intelligence Research 16 (2002) 321-357.

[5] J. H. Friedman, T. Hastie, and R. Tibshirani, Regularization Paths for Generalized Linear Models via Coordinate Descent, Journal of Statistical Software 33 (2010) 1-22.

[6] E. Goel, E. Abhilasha, E. Goel, and E. Abhilasha, Random Forest: A Review, International Journal of Advanced Research in Computer Science and Software Engineering 7 (2017) 251-257.

[7] U. Hasanah, A. M. Soleh, and K. Sadik, Effect of Random Under Sampling, Oversampling, and SMOTE on the Performance of Cardiovascular Disease Prediction Models, Jurnal Matematika, Statistika dan Komputasi 21 (2024) 88-102.

[8] R. Hidayat, M. A. Syawaludin, and N. Nurmalitasari, Prediksi Churn Pelanggan Multinational Bank Menggunakan Algoritma Machine Learning, Simpatik: Jurnal Sistem Informasi dan Informatika 4 (2024) 89-97.

[9] F. Ismail and I. I. Lawanda, Implementasi EDMS dalam Penataan Dokumen di Rail Document System PT. Kereta Api Indonesia (Persero) Daerah Operasi 1 Jakarta, Baca: Jurnal Dokumentasi Dan Informasi 41 (2020) 143-168.

[10] S. M. Kim, Y. Kim, K. Jeong, H. Jeong, and J. Kim, Logistic LASSO Regression for the Diagnosis of Breast Cancer Using Clinical Demographic Data and the BI-RADS Lexicon for Ultrasonography, Ultrasonography 37 (2018) 36-42.

[11] M. Marcellina and A. Mukhlason, Analisis Prediktif Churn untuk Meningkatkan Tingkat Retensi Pelanggan pada Perusahaan SaaS Menggunakan Machine Learning, ILKOMNIKA 6 (2024) 21-32.

[12] J. M. Pereira, M. Basto, and A. F. Da Silva, The Logistic Lasso and Ridge Regression in Predicting Corporate Failure, Procedia Economics and Finance 39 (2016) 634-641.

[13] A. N. R. Putri and Y. S. Rahayu, Customer Retention sebagai Variabel Intervening pada Pengaruh Relationship Quality terhadap Loyalitas Nasabah Tabungan Bank Syariah, Jurnal Ilmu Manajemen (JIM) 11 (2023) 241-251.

[14] F. I. Silfana and M. A. Barata, Using K-NN Algorithm for Evaluating Feature Selection on High Dimensional Datasets, Jurnal Teknik Informatika 17 (2024) 190-202.

[15] J. Han, M. Kamber, and D. Mining, Data Mining: Concepts and Techniques (Morgan Kaufmann, San Mateo, 2006).

[16] M. F. Naufal, S. Subrata, A. F. Susanto, C. N. Kansil, and S. Huda, Analisis Perbandingan Algoritma Machine Learning untuk Prediksi Potensi Hilangnya Nasabah Bank, Techno.com 22 (2023) 1-11.

[17] T. Wongvorachan, S. He, and O. Bulut, A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining, Information 14 (2023) 54.

[18] A. U. Zailani and N. L. Hanun, Penerapan Algoritma Klasifikasi Random Forest untuk Penentuan Kelayakan Pemberian Kredit di Koperasi Mitra Sejahtera, Infotech: Journal of

Technology Information 6 (2020) 7-14.

[19] J. Zhang and L. Chen, Clustering-based Undersampling with Random Over Sampling Examples and Support Vector Machine for Imbalanced Classification of Breast Cancer Diagnosis, Computer Assisted Surgery 24 (2019) 62-72.