Jouínal of Modeín Applied Statistical Methods

Volume 24 | Issue 1 Aíticle 11

COMPARISON OF RACOG AND RACOG-RUS FOR CLASSIFYING IMBALANCED DATA ON GRADIENT BOOSTING AND NAÏVE BAYES PERFORMANCE

Rahmi Fadhilah
Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia,

Heri Kuswanto*
Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia,
heri.kuswanto@its.ac.id

Dedy Dwi Prastyo
Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia,

Dinda Ayu Safira
Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia

M. Y. Matdoan
Department of Statistics, Institut Teknologi Sepuluh Nopember, Indonesia
Department of Statistics, Pattimura University, Indonesia

Received: 20-08-2024; Revised: 15-09-2024 Accepted: 20-10-2024; Published: 23-10-2024

Recommended Citation

Rahmi Fadhilah, Heri Kuswanto,Dedy Dwi Prastyo, Dinda Ayu Safira, M. Y. Matdoan (2024). Efficient COMPARISON OF RACOG AND RACOG-RUS FOR CLASSIFYING IMBALANCED DATA ON GRADIENT BOOSTING AND NAÏVE BAYES PERFORMANCE. Jouínal of ModeínApplied Statistical Methods, 24(1), https://doi.oíg/10.56801/Jma,sm.V24.i1.11

Journal of Modern Applied Statistical Methods
2025, Vol. 24, No. 1,
Doi: 10.56801/Jmasm.V24.i1.11

ISSN 1538 – 9472

COMPARISON OF RACOG AND RACOG-RUS FOR CLASSIFYING IMBALANCED DATA ON GRADIENT BOOSTING AND NAÏVE BAYES PERFORMANCE

Rahmi Fadhilah
Department of Statistics,
Institut Teknologi Sepuluh
Nopember, Indonesia

Heri Kuswanto
Department of Statistics,
Institut Teknologi Sepuluh
Nopember, Indonesia

Dedy Dwi Prastyo
Department of Statistics,
Institut Teknologi Sepuluh
Nopember, Indonesia

Dinda Ayu Safira
Department of Statistics,
Institut Teknologi Sepuluh
Nopember, Indonesia

M. Y. Matdoan
Department of Statistics,
Institut Teknologi Sepuluh
Nopember, Indonesia
Department of Statistics,
Pattimura University,
Indonesia

This study aims to determine the effect of resampling RACOG and RACOG-RUS data on Gradient Boosting and Naïve Bayes classification in predicting water quality with unbalanced data. The data used in this study were 720 data from January 2022 to December 2023. It was found that Gradient Boosting performed best when using RACOG-RUS resampling data and feature selection with a number of numIntances of 200. While Naïve Bayes has the best performance when using RACOG-RUS resampling data without feature selection with a number of numIntances of 300. It can be seen that resampling RACOG data does not outperform RACOG-RUS in both classification models because it is known that the data generated in RACOG does not make the dataset more balanced than RACOG-RUS. Hybrid sampling is necessary if RACOG samples are used as the training dataset.

Keywords: Gradient Boosting, Naive Bayes, RACOG, RACOG-RUS..

1. Introduction

River water is a source of life that plays an important role in human life and the surrounding ecosystem. In addition, rivers have an important function as a transportation system that carries large amounts of materials in dissolved and particulate form from natural and artificial sources in one direction. However, river water often becomes a reservoir for harmful industrial waste, causing pollution that endangers human health and the surrounding environment. Human activities related to agricultural pesticide use and changes in land use are major factors affecting surface water quality. The impacts are devastating, with millions of deaths of living beings

every year and huge economic losses. Therefore, effective protection and management of river water are critical to maintaining global sustainability in both life and the economy (Azhar et al., 2015; Khan et al., 2022).

Class imbalance in water quality data is a common problem, where one outcome is less common than the other. Although standard classifiers such as logistic regression, SVM, and decision tree are effective with balanced data, they are suboptimal in the case of imbalance (Abo Zahhad, M. et al., 2023). Imbalanced data occurs when there is an imbalance of classes during classification, resulting in majority and minority classes. This results in misclassification in the majority class more often than the minority class. Thus, the focus should be on reducing misclassification in the majority class and sacrificing accuracy in the minority class (Ramyachitra & Manikandan, 2014). The solution to the problem of data imbalance can be divided into three main approaches. The first approach, data-level methods, is an external approach that focuses on balancing the data by reducing samples from the majority class (undersampling) or adding samples to the minority class (oversampling). The second algorithm-level method is an internal approach aimed at addressing bias due to data imbalance through improving existing algorithms or developing new classification algorithms. The third method, the hybrid method, is a combination of data-level and algorithm-level methods with the aim of improving classification accuracy in the face of data imbalance problems (Spelmen & Porkodi, 2018).

Various approaches have been proposed to handle the imbalanced data problem, including data-level techniques such as oversampling and undersampling. Oversampling techniques used such as Rapidly Converging Gibbs Sampler (RACOG) (Das et al., 2014) and undersampling techniques used such as Random Under Sampling (RUS) (Tyagi and Mittal, 2020). Recent developments combine these two techniques in hybrid sampling to create balanced datasets. However, inadequate representation of minority classes and overlap issues between classes can compromise model performance. Therefore, an effective hybrid method is needed to address these issues by considering the probability distribution of minority classes and avoiding over-sampling. The *hybrid sampling* technique performed by Malek (2023) used RACOG-RUS.

The ensemble approach and cost-sensitive learning are two common methods used in handling data imbalance in machine learning studies (Spelmen & Porkodi, 2018). The ensemble method combines the decisions of multiple base classifiers to produce more accurate predictions, where the diversity and accuracy of each base classifier become key factors in good performance. Three popular ensemble methods are boosting, bagging, and stacking (Breiman, 1996; Spelmen & Porkodi, 2018). In addition, cost-sensitive learning allocates costs to different classes to improve model performance. However, classification results can be unstable with this approach due to the difficulty in determining the appropriate error cost (Spelmen & Porkodi, 2018). Although several models have been developed to predict water pollution, the development of unbalanced learning systems with established methods is still insufficient in water quality studies. In order to improve the performance of the model in the case of high-dimensional models, feature selection is used.

Gradient boosting is an extension of the concept of ensemble learning, where several weak prediction models are combined to form a stronger model. This method extends the idea of boosting by taking into account the gradient of the loss function when adding new prediction models to the ensemble. As a machine learning method, gradient boosting is used for supervised learning applications, such as classification and regression. It uses an ensemble of weak prediction models, usually a decision tree, and has three main components: loss function, weak learner, and additive (Sahin, 2020). Gradient boosting has advantages in handling high-order relationships in data and various data challenges (Klug et al., 2019). In addition, Naive Bayes is a machine learning algorithm that uses Bayes' Theorem for classification. Known for its high training speed, it works under the assumption of independence between features. It is used for high-dimensional datasets and generates probability predictions for each class by utilizing the joint posterior probability distribution between classes and attributes. Despite relying on fairly simple assumptions, Naive Bayes can compete well with other algorithms in many cases (Mitchell, 1997; Muller & Guido, 2016). A performance comparison between Gradient Boosting and Naive Bayes in water quality research can provide important insights into the best approach for specific situations.

Based on previous research that has been described, this study combines data imbalance handling with RACOG and RACOG-RUS applied to the gradient boosting and naive bayes methods with and without feature selection on water quality data in the Bengawan Solo River using various evaluations, including accuracy, balanced accuracy, sensitivity, specificity, precision, F-messure, and AUC.

2. Feature Selection

The dimensionality of data used in machine learning tasks can lead to significant issues, such as the curse of dimensionality in existing learning methods. Feature selection is one widely used solution to reduce dimensionality. Its objective is to remove specific subsets of irrelevant features from the original dataset based on evaluation criteria of importance and to select a few features that are most representative of the original set. Feature selection typically results in improved learning performance, reduced computational costs, and enhanced model interpretation. Methods for feature selection can be classified into filter, embedded, and wrapper methods. Filter methods use general data characteristics to assess features without requiring a classifier in the process. Meanwhile, wrapper methods rely on the accuracy of specific classifiers to select features, and embedded methods integrate feature subsets as internal mechanisms in the classifier training process. This study employs two methods for feature selection: wrapper and embedded methods. In the wrapper method, the Boruta feature selection algorithm is chosen. The Boruta algorithm is a wrapper method based on random forest algorithms that capture important and interesting features in the dataset while considering the output variable. To support this decision, an embedded method is also used for feature selection. This method utilizes the important features of the classifier as an internal mechanism to measure the predictive strength of each feature and selects the highest predictive strength. This method is chosen because it can provide a simple approach to feature selection by using the average accuracy and an average decrease in node impurity.

3. Rapidly Converging Gibbs Sampler (RACOG)

RACOG is a data resampling method using Gibbs sampling to generate new minority class samples from the minority class probability distribution approximated using the Chow-Liu algorithm with the Markov Chain Monte Carlo (MCMC) method (Das, B. et al., 2014). RACOG offers an alternative mechanism for selecting the initial value of the random variable X (denoted as X(0)) to improve the randomly selected Standard Gibbs sampling. In addition, it uses the minority class data points as the initial sample set and runs the Gibbs Sampler for each minority class sample. The total number of iterations for the Gibbs Sampler is limited according to the distribution of the minority and majority classes. RACOG generates multiple Markov chains, each starting with a different minority class sample, unlike the conventional Gibbs sampler, which starts with a minority class sample in one very long chain. Huan Y. et al. (2020) said that this approach is different from other approaches because it considers the distribution of minority classes and is proven to provide the best performance when compared to other oversampling approaches. MCMC is an increasingly popular method for obtaining information about distributions, especially for estimating posterior distributions in Bayesian inference (Van Ravenzwaaij et al., 2018). The following is the RACOG algorithm with time complexity where *n* = data dimension, *D* = minority class cardinality, and *T* = predetermined number of iterations.

Algorithm RACOG

```
function RACOG (minority, D, n, β, α, T)
Input: minority = minority class data points; D = size of minority ; n = minority dimensions ; β = burn-in period; α = lag; T = total number of iterations
Output: new_samples = new minority class samples
Construct Dependence tree DT using Chow-Liu algorithm.
for d = 1 to D do
    X (0) = minority (d)
    for t = 1 to T do
        for i = 1 to n do
            Simplify $P(X_i|x_1^{(t+1)},...,x_{i-1}^{(t+1)},x_{i+1}^{(t)},...,x_n^{(t)})$ using DT
            xi(t+1) ~ P(Si) where Si is the state space of attribute xi
        if t > betha AND t mod (alpha) = 0
            new_samples = new_samples + X(t)
return new_samples
```

4. RACOG-RUS

The RACOG method is known to simply improve the traditional Gibbs Sampler to address class imbalance without considering the usefulness of the resulting samples. As a result, there is a risk that RACOG may add unnecessary minority class samples, potentially causing overfitting issues and providing little help in building good hypotheses (Das et al., 2014) (Rahmi et al., 2024) . Therefore, (Abdul Malek, N. H & Wan Yaacob, W. F (2023) proposed a new approach that combines the RACOG model with undersampling techniques (RUS) in RACOG-RUS to eliminate redundant minority class samples. In this method, RACOG uses Gibbs Sampler to synthesize the minority classes estimated using the Markov Chain Monte Carlo (MCMC) method. In this algorithm, each sampling step considers the univariate conditional distribution of each dimension. The value of each dimension depends on the values of other dimensions and the previous values of the same dimension. Such conditional distributions are easier to model compared to the complete joint distribution. The standard Gibbs Sampler algorithm is shown below.

Algorithm Gibbs Sampler

```
$X^{(0)} =< x_1^{(0)}, \dots, x_n^{(0)} >$
$\boldsymbol{for}\ t = 1\ to\ T$
    $\boldsymbol{for}\ i = 1\ to\ n$
        $x_i^{(t+1)} \sim P(X_i | x_1^{(t+1)}, \dots, x_{i-1}^{(t+1)}, x_{i+1}^{(t)}, \dots, x_n^{(t)})$
```

$P\left(X_i \middle| x_1^{(t+1)}, \dots, x_{i-1}^{(t+1)}, x_{i+1}^{(t)}, \dots, x_n^{(t)}\right)$is drawn from the joint probability distribution that represents the univariate conditional distribution of each dimension for selecting attribute values , thus generating a new sample of the minority class $x_i^{(t+1)}$ which is determined by randomly selecting from the state space distribution with all possible values of the attribute x_i. Gibbs Sampling is implemented with two factors, namely burn-in and lag. Burn-in refers to the number of iterations required to generate samples in order to achieve a stable distribution. Lag, on the other hand, pertains to successive samples that are removed from the Markov Chain after each sample is received to prevent correlation between subsequent samples. After oversampling from the minority class, there is a reduction of samples from the majority class to align the data to its original size, thus providing more accurate information. RACOG-RUS combines probabilistic oversampling while considering the minority class distribution and undersampling to produce a more balanced and representative dataset while overcoming the problem of overfitting.

5. Gradient Boosting

Gradient boosting (GBM) is one of the machine learning methods used in supervised machine learning applications, and it includes various classification and regression problems. GBM builds a prediction model in the form of a collection of weak prediction models, such as decision trees. GBM consists of three main components: loss function for optimization, weak learner for prediction, and additive model to combine the weak learner to optimize the loss function (Sahin, E. K, 2020). This algorithm differs from random forests (RF), as it sequentially trains multiple weak learners (tree-based classifiers) to reinforce each other and produce superior results. At each stage, a new decision tree is learned to correct the mistakes made by the existing trees. As a non-linear method, gradient boosting naturally outperforms linear models when higher-order relationships exist in the data and has demonstrated its superiority compared to other machine learning algorithms in various data challenges (Klug, M. 2019).

$$g_t(x) = E_y \left[\frac{\partial \psi(y, f(x))}{\partial f(x)} | x \right]_{f(x)=\hat{f}_{t-1}(x)} \qquad (1)$$

$$(\rho_t, \theta_t) = \arg\min_{\rho,\theta} \sum_{i=1}^{N} [-g_t(x_i) + \rho h(x_i, \theta)]^2 \qquad (2)$$

Algorithm Gradient Boost Algorithm

```
Inputs:
•   input data $(x, y)_{i=1}^{N}$
•   number of iterations $M$
•   choice of the loss-function $\psi(y, f)$
•   choice of the base-learner model $h(x, \theta)$
Algorithm:
initialize $\widehat{f_0}$ with a constant
for $t = 1$ to $M$ do
    compute the negative gradient $g_t(x)$
    fit a new base-learner function $h(x, \theta_t)$
    find the best gradient descent step-size $\rho_t$:
    $\rho_t = \arg min_\rho \sum_{i=1}^{N} \psi\ [y_i, \hat{f}_{t-1}(x_i) + \rho h(x_i, \theta_t)$ for $i = 1$ to $n$ do
    update the function estimate:
    $\hat{f}_t \leftarrow \hat{f}_{t-1} + \rho_t h(x, \theta_t)$
end for
```

6. Naïve Bayes

A naive Bayes classifier is a machine learning algorithm that utilizes Bayes' theorem to solve classification problems by taking probabilities into account. It belongs to the supervised learning category and is known for its high training speed. Naive Bayes operates on the assumption of independence between predictors by evaluating each feature separately and collecting simple statistics per class of each feature. The advantage of this algorithm is its suitability for high-dimensional datasets, as it relies on the assumption of statistical independence among the features. By utilizing the joint posterior probability distribution between classes and attributes, Naive Bayes can effectively derive classification problems by generating probability predictions for each class. Research shows that Naive Bayes is able to compete well with other machine learning algorithms, even outperforming them in some cases (Mitchell, 1997; Muller & Guido, 2016). The algorithm relies on the assumption that conditional attribute values are independent of each other, taking into account the target value of the given instance (Stephen, 2014; Islam et al., 2010).

$$P(a_1, a_2, a_3, \cdots, a_n | v) = \arg\max \prod_{i=1}^{n} P(a_i | v_j) \qquad (3)$$

$$v_{NB} = \arg\max P(v_j) \prod_{i=1}^{n} P(a_i|v_j) \tag{4}$$

In other words, the probability of observing a conjunction of attribute values given a target instance is simply the product of the probabilities for the individual attributes. Although this assumption is sometimes unrealistic in real contexts, Naive Bayes remains effective due to its ability to simplify probability calculations and reduce dimensionality. Despite its simplicity, the algorithm is widely used in data classification and has been shown to produce results comparable to other classification methods in various domains (Islam et al., 2010; Stephen, 2014).

7. Performance

The performance of Gradient Boosting and Naïve Bayes before and after feature selection will be evaluated and validated using the 10-fold cross-validation method and confusion matrix. Cross-validation is used to evaluate the performance of the prediction model by dividing the initial data into two parts, namely training and testing data, which were done 10 times. Although there is no fixed rule, the division of training and testing data is done with a ratio of 70:30 is often used in evaluating prediction models (Khosravi, Y. *et al.*, 2014). As for this study, eight performance metrics will be used, which include accuracy, balanced accuracy, sensitivity, specificity, precision, f-measure, G-means, and Area Under Curve (AUC).

Table 1. Confusion matrix for binary classification

		Predicted	
		Good	**Polluted**
Actual	Good	True Negative (TN)	False Positive (FP)
	Polluted	False Negative (FN)	True Positive (TP)

Based on Table 1, some formulas can be calculated as follows:

1) Accuracy

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

2) Balanced accuracy

$$\text{Balanced accuracy} = \frac{(Sensitivity + Specificity)}{2} \tag{6}$$

3) Sensitivity

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{7}$$

4) Specivicity

$$\text{Specificity} = \frac{TN}{TN + FP} \tag{8}$$

5) Precission

$$\text{Precision} = \frac{TP}{TP + FP} \tag{9}$$

6) F-measure

$$\mathrm{F-measure} \quad = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{10}$$

7) G-means

$$\mathrm{G-means} \quad = \sqrt{Sensificity \times Specificity} \tag{11}$$

8) Area Under Curve (AUC)

$$\mathrm{AUC} \quad = \frac{TP + TN}{TP + FP + FN + TN} \tag{12}$$

A model with an AUC value close to 1 indicates that the model is very good, and if it is closer to 0, then it has the worst measurements (Narkhede, 2018).

8. Methodology

8.1 Data Source

The data in this study are secondary data sourced from the Ministry of PUPR Directorate General of Water Resources of the Bengawan Solo River Basin Center, with a total of 720 data from January 2022 to December 2023.

8.2 Research Variables

The research variables used consist of 8 variables (parameters), namely pH (X1), TDS (X2), TSS (X3), Temperature (X4), BO (X5), BOD (X6), COD (X7), NO3 (X8), and Water Quality Status (Y).

8.3 Research Procedure

The steps in this research are as follows:

1. Identify the problem.
2. Conduct a literature study.
3. Obtained secondary data from the Ministry of PUPR Directorate General of Water Resources Bengawan Solo River Basin Center.
4. Perform data description to describe and describe the data used.
   At this stage, summary results are displayed to show the comparison of the number of polluted and unpolluted river classifications. In addition, data exploration analysis is carried out with simple descriptive statistics.
5. Perform pre-processing of data viz:

a. Cleaning data, making sure there are no *missing values*, and filling in the blanks with the *Expectation Maximization* algorithm.
b. Scaling the data, transforming the data using a standard scaler (Z Score).

6. Identify the imbalance ratio using a *bar graph.*
7. Data partitioning is performed by dividing the data into training and validation sets with a scheme of 70% training data and 30% testing data.
8. Perform feature selection with the Wrapper method using the Boruta algorithm and Embedded method using random forest classifier.
9. Resampling data on training data using individual sampling, namely Rapidly Converging Gibbs Sampler (RACOG), and hybrid sampling, namely RACOG-RUS.
10. Perform classification using Gradient Boosting and Naïve Bayes algorithms.
11. Comparing classification performance with or without feature selection of RACOG and RACOG-RUS samples.

12. Draw conclusions.

Comparison and evaluation of model performance using confusion matrix obtained from the validation process using k-fold cross-validation with 10-fold cross-validation. The quality of the model can be seen based on the value of balanced accuracy, accuracy, specificity, sensitivity, precision, f-measure, recall, and Area Under Curve (AUC) for both training and validation sets.

9. Results and discussion

9.1 Data Description

The description of river water quality parameter data can be seen in Table 2 below.

Table 2. Description of water quality parameters

	Temperature	pH	Nitrate	DO	COD	KOB	TSS	TDS
Min	23.48	3.47	0.01	0.40	1.00	0.20	2.00	0.05
Max	36.26	9.24	18.90	51.00	495.80	59.50	4180.00	1223.00
Mean	27.54	6.61	0.97	5.33	30.58	4.41	91.22	869.44

Based on Table 2 shows descriptive statistics of water quality parameters. The lowest temperature parameter is 23.48, and the highest is 36.26, with an average river water temperature of 27.54. Furthermore, the lowest pH is 3.47, and the highest is 9.24, with an average pH in river water of 6.61. Furthermore, the lowest nitrate is 0.01, and the highest is 18.90, with an average nitrate in river water of 0.97. Furthermore, the lowest DO is 0.40, and the highest is 51.00, with an average DO in river water of 5.33. Furthermore, the lowest COD is 1.00, and the highest is 495.80, with an average COD in river water of 495.80. Furthermore, the lowest KOB is 0.20, and the highest is 59.50, with an average KOB in river water of 4.41. Furthermore, the lowest TSS is 2.00, and the highest is 4180.00, with an average TSS in river water of 91.22. Furthermore, the lowest TDS is 0.05, and the highest is 1223.00, with an average TDS in river water of 869.44.

9.2 Data Pre-processing

Pre-processing of river water quality parameter data is shown in Table 3 below.

Table 3. Data pre-processing results

N	Temperature	pH	Nitrate	DO	COD	KOB	TSS	TDS
1	-0.749	-0.319	-0.880	-1.060	-0.128	1.096	-0.272	-0.137
2	-0.543	-0.381	-0.890	-1.170	-0.420	1.490	-0.268	-0.137
3	-0.247	-1.708	-0.601	-0.866	-0.294	0.122	-0.247	-0.137
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
720	2.952	1.488	0.234	1.348	0.650	0.889	-0.197	6.095

Based on Table 3, pre-processing of river water quality has been carried out, and there are no variables or parameters that have a value of more than 2.5. So, it is concluded that there are no outliers in the research data.

9.3 Imbalanced Ratio

To find out how big the imbalance of a data set is, one must look at the imbalance ratio. The imbalance ratio is calculated by dividing the number of majority classes by the number of minority classes. When IR = 1, it can be said that the dataset is in a balanced position. In this study, it can be seen that the imbalance ratio is 71, so it can be said that the majority class is 71 times more than the minority class. The unbalanced classes between polluted and unpolluted classes are shown in Figure 1 below.

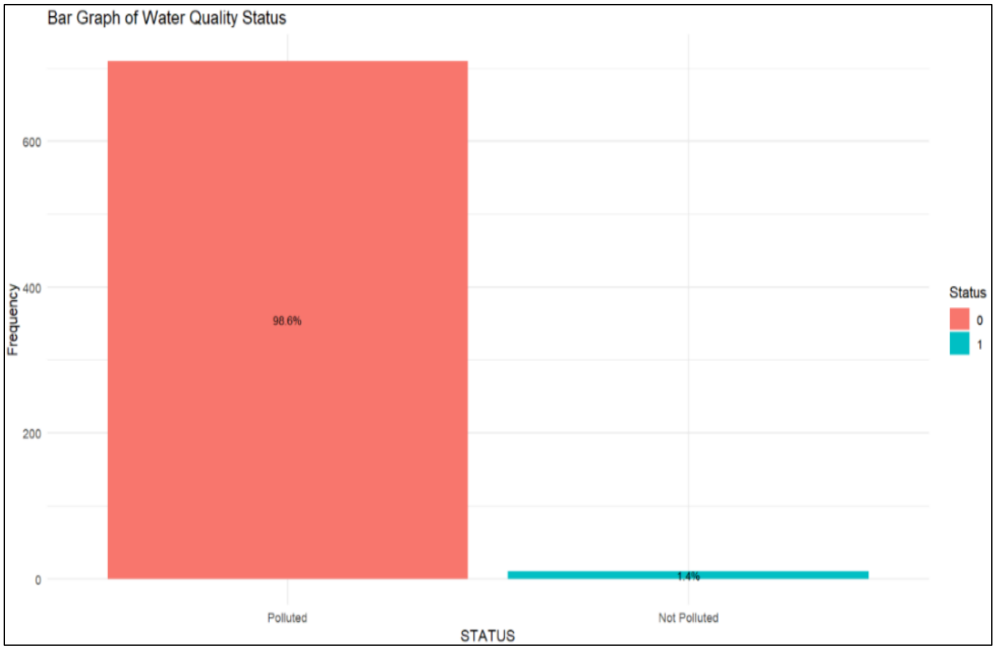

Figure 1. Dataset Imbalance Ratio

9.4 Feature Selection

Feature selection serves to show the importance of features using the Boruta algorithm. Based on Figure 2, the boxplot for each feature shows that 5 influential and important attributes include Chemical Oxygen Demand (COD), Biochemical Oxygen Demand (BOD), Total Suspended Solids (TSS), potential of hydrogen, and Total Dissolved Solids (TDS). The embedded method is used to support the decision and to select features.

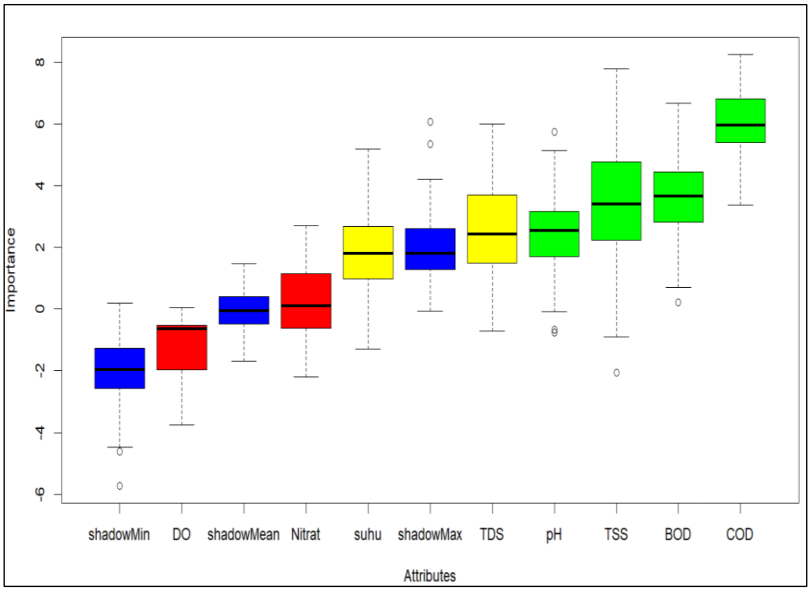

Figure 2. Feature Importance Using Boruta Algorithm

This study uses Random Forest as a classification because it provides a simple method of feature selection using the mean decrease in accuracy and node cleanliness. When the average decrease in accuracy (gini index) is higher, the variable is considered more important in the model. The selected features are used at the top of the tree and contribute to the final decision to predict a larger proportion of the input samples. Furthermore, to find out the results of feature selection using the Variable Importance Plot for Random Forest from the Embedded algorithm is shown in Figure 3 below.

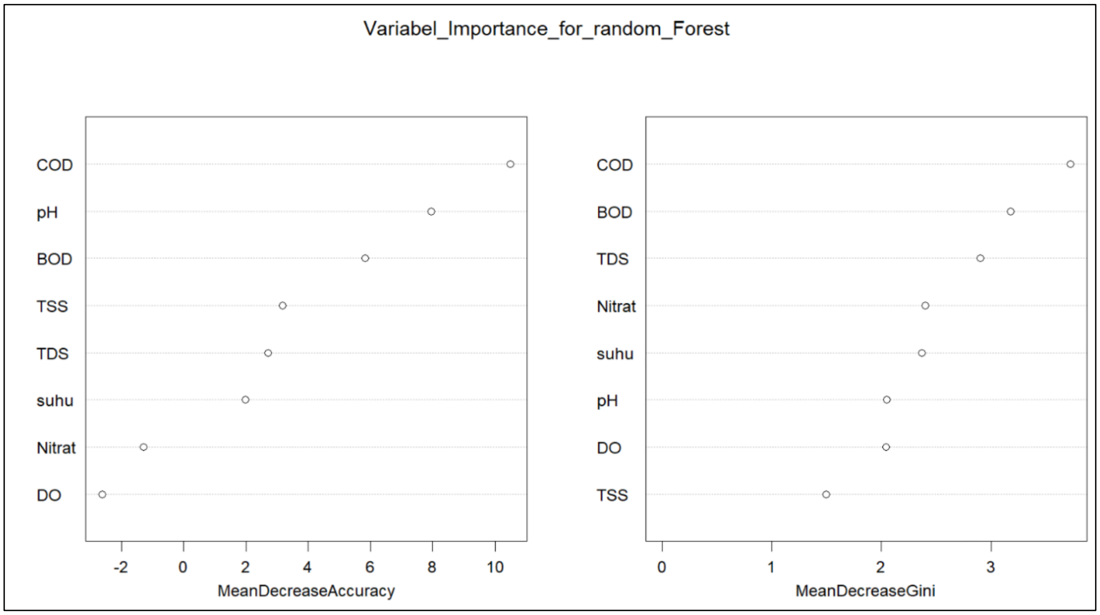

Figure 3. Feature Importance Using Embedded Algorithm

Based on the output of Mean Decrease Accuracy, it is known that there are three features that are not important, including Dissolved Oxygen (DO), Nitrate (NO_3), and Temperature. This is in accordance with the wrapper method that has been used in previous research. Therefore, both Wrapper and Embedded methods show similar results.

9.5 Gradient Boosting

This section presents the performance of the Gradient Boosting ensemble model without and with feature selection using both the Boruta algorithm and the embedded method with a random forest classifier. In the performance comparison, the effect of individual sampling (RACOG) and hybrid sampling (RACOG-RUS) in classifying water quality is examined. Performance is evaluated using eight performance metrics: accuracy, balanced accuracy, sensitivity, specificity, precision, f-measure, and AUC. Furthermore, Table 4 presents the performance of Gradient Boosting with and without feature selection.

Table 4. Performance of Gradient Boosting with and without feature selection

Algorithm	**Feature Selection**	**Sampling**	**numInstances**	**Accuracy**	**Balanced Accuracy**	**Sensitivity**	**Spesificity**	**Precision**	**F-measure**	**AUC**
Gradient Boosting	With	RACOG	100	0.97	0.49	0.99	0.00	0,99	0.99	0.50
			200	**0.98**	**0.50**	**0.99**	**0.00**	**0,99**	**0.99**	**0.67**

			300	0.98	0.50	0.99	0.00	0,99	0.99	0.67
		RACOG-RUS	100	0.95	0.48	0.96	0.00	0,99	0.97	0.64
			200	**0.97**	**0.49**	**0.98**	**0.00**	**0,99**	**0.98**	**0.66**
			300	0.96	0.49	0.98	0.00	0,99	0.98	0.66
	Without	RACOG	**100**	**0.98**	**0.50**	**0.99**	**0.00**	**0,99**	**0.99**	**0.50**
			200	0.95	0.48	0.97	0.00	0,99	0.98	0.50
			300	0.96	0.49	0.98	0.00	0,99	0.98	0.50
		RACOG-RUS	100	0.90	0.46	0.92	0.00	0,98	0.95	0.65
			200	**0.96**	**0.49**	**0.98**	**0.00**	**0,99**	**0.98**	**0.66**
			300	0.93	0.47	0.94	0.00	0,99	0.96	0.50

Table 4 shows that RACOG outperforms RACOG-RUS from all performance metrics, except AUC, by using the number of numIntances of 100. Likewise, when feature selection is carried out, it can also be seen that RACOG also outperforms RACOG-RUS from all performance metrics with a numIntance of 200.

9.6 Naïve Bayes

This section presents the performance of the Naïve Bayes model when not and with feature selection both with the Boruta algorithm and the embedded method. Furthermore, Table 5 presents the performance of Naïve Bayes with and without feature selection.

Table 5. Naïve Bayes performance with and without feature selection

Algorithm	**Feature Selection**	**Sampling**	**numInstances**	**Accuracy**	**Balanced Accuracy**	**Sensitivity**	**Spesificity**	**Precision**	**F-measure**	**AUC**
Naïve Bayes	With	RACOG	100	0.97	0.49	0.99	0.00	0.99	0.99	0.61
			200	0.98	0.50	0.99	0.00	0.99	0.99	0.60
			300	**0.98**	**0.50**	**0.99**	**0.00**	**0.99**	**0.99**	**0.60**
		RACOG-RUS	100	0.80	0.57	0.80	0.33	0.99	0.89	0.71
			200	0.84	0.58	0.84	0.33	0.99	0.91	0.74
			300	**0.85**	**0.60**	**0.86**	**0.33**	**0.99**	**0.92**	**0.59**
	Without	RACOG	**100**	**0.87**	**0.61**	**0.89**	**0.33**	**0.99**	**0.94**	**0.49**
			200	0.85	0.59	0.85	0.33	0.99	0.92	0.48
			300	0.84	0.59	0.84	0.33	0.99	0.91	0.48
		RACOG-RUS	100	0.73	0.70	0.73	0.67	0.99	0.84	0.47
			200	0.76	0.71	0.76	0.67	0.99	0.86	0.47
			300	**0.82**	**0.58**	**0.83**	**0.33**	**0.99**	**0.90**	**0.48**

Table 5 shows that RACOG outperforms RACOG-RUS in all performance metrics, except AUC, by using the number of numbers of 100. Likewise, when feature selection is carried out, it can also be seen that RACOG also outperforms RACOG-RUS from all performance metrics with a numIntance of 200.

9.7 Receiver Operating Characteristic (ROC) plot of Gradient Boosting and Naïve Bayes algorithms

Theoretically, if the classifier predicts the sample correctly, then the ROC curve will rise to the upper left of the graph. Furthermore, Figure 4 shows the ROC plot of each model from resampling RACOG and RACOG-RUS data on the Gradient Boosting classification algorithm with and without feature selection.

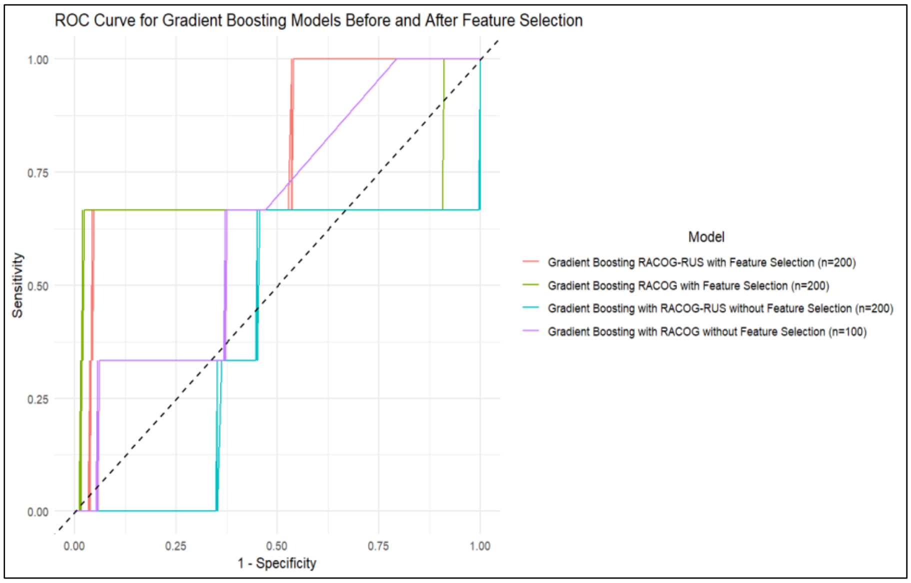

Figure 4. Comparison of RACOG and RACOG-RUS with and without Feature Selection on Gradient Boosting

Figure 4 shows that Gradient Boosting correctly predicts samples with the highest performance when using RACOG-RUS resampling data with feature selection with a numIntances of 200. Furthermore, Figure 5 shows the ROC plot of each model from resampling RACOG and RACOG-RUS data in the Naïve Bayes classification algorithm with and without feature selection. Figure 5 shows that Naïve Bayes predicts samples correctly with the highest performance when using RACOG-RUS resampling data without feature selection with the number of numIntances 300.

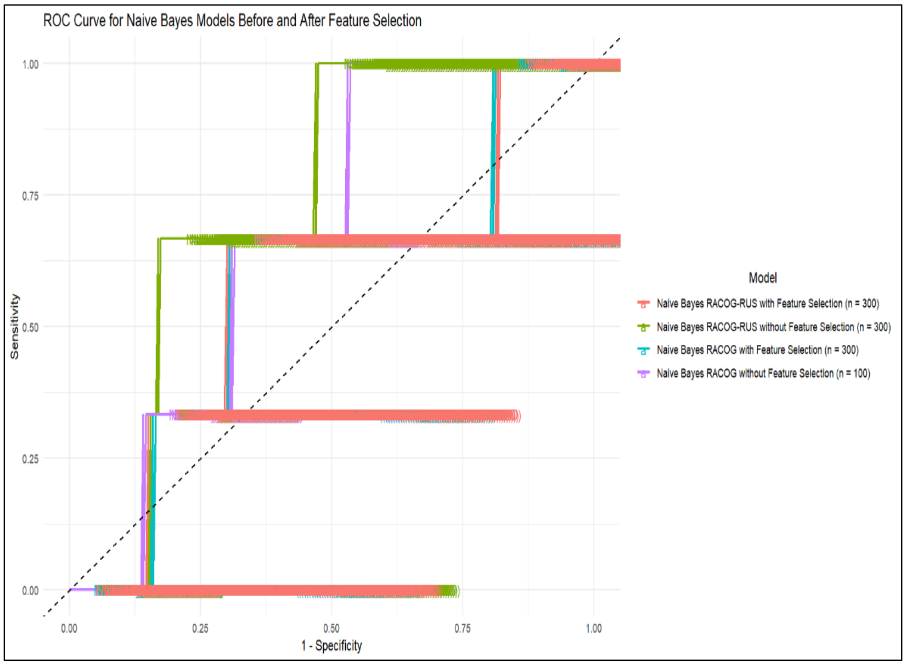

Figure 5. Comparison of RACOG and RACOG-RUS with and without Feature Selection on Naïve Bayes

10 Conclusions

Based on the results and discussion, it is concluded that Gradient Boosting has the best performance when using RACOG-RUS resampling data with feature selection with a numIntances of 200. While Naïve Bayes has the best performance when using RACOG-RUS resampling data without feature selection with a numIntances of 300. It can be seen that resampling RACOG data does not outperform RACOG-RUS in both classification models because it is known that the data generated in RACOG does not make the dataset more balanced than RACOG-RUS, so it is necessary to do hybrid sampling if using RACOG samples as a training dataset.

Acknowledgments

This research was supported by the Directorate General of Science and Technology Resources and Higher Education, Kemristekdikti, through the scholarship "Master Education Program Towards Doctor for Excellent Scholars (PMDSU)," which funded this research.

References

Abdul Malek, N. H., & Wan Yaacob, W. F. (2023). Performance Evaluation of Classification Methods with Hybrid Sampling for Imbalanced Data: A Comparative Simulation Study. *Performance Evaluation of Classification Methods with Hybrid Sampling for Imbalanced Data: A Comparative Simulation Study*.

Abo-Zahhad, M. M., Elsayed, M., Sayed, M., Abdel Malek, A., Fawaz, A., Sharshar, A., & Abo Zahhad, M. (2023). Design of smart wearable system for sleep tracking using SVM and multi-sensor approach. *JES. Journal of Engineering Sciences*, *51*(4), 1-15.

Azhar, S. C., Aris, A. Z., Yusoff, M. K., Ramli, M. F., & Juahir, H. (2015). Classification of river water quality using multivariate analysis. *Procedia Environmental Sciences*, *30*, 79-84.

Breiman, L. (1996). Bagging predictors. *Machine learning*, *24*, 123-140.

Das, B., Krishnan, N. C., & Cook, D. J. (2014). RACOG and wRACOG: Two probabilistic oversampling techniques. *IEEE transactions on knowledge and data engineering*, *27*(1), 222-234.

Huan, Y., Kong, Q., Mou, H., & Yi, H. (2020). Antimicrobial peptides: classification, design, application and research progress in multiple fields. *Frontiers in microbiology*, *11*, 582779.

Islam, M. M., Hossain, M. A., Jannat, R., Munemasa, S., Nakamura, Y., Mori, I. C., & Murata, Y. (2010). Cytosolic alkalization and cytosolic calcium oscillation in Arabidopsis guard cells in response to ABA and MeJA. *Plant and Cell Physiology*, *51*(10), 1721-1730.

Khan, M. S. I., Islam, N., Uddin, J., Islam, S., & Nasir, M. K. (2022). Water quality prediction and classification based on principal component regression and gradient boosting classifier approach. *Journal of King Saud University-Computer and Information Sciences*, *34*(8), 4773-4781.

Khosravi, Y., Asilian-Mahabadi, H., Hajizadeh, E., Hassanzadeh-Rangi, N., Bastani, H., & Behzadan, A. H. (2014). Factors influencing unsafe behaviors and accidents on construction sites: A review. *International journal of occupational safety and ergonomics*, *20*(1), 111-125.

Klug, M., Barash, Y., Bechler, S., Resheff, Y. S., Tron, T., Ironi, A., & Klang, E. (2020). A gradient boosting machine learning model for predicting early mortality in the emergency department triage: devising a nine-point triage score. *Journal of general internal medicine*, *35*, 220-227.

Malek, N. H. A., Yaacob, W. F. W., Wah, Y. B., Nasir, S. A. M., Shaadan, N., & Indratno, S. W. (2023). Comparison of ensemble hybrid sampling with bagging and boosting machine learning approach for imbalanced data. *Indones. J. Elec. Eng. Comput. Sci*, *29*, 598-608.

Mitchell, T. M. (1997). Does machine learning really work?. *AI magazine*, *18*(3), 11-11.

Müller, A. C., & Guido, S. (2016). *Introduction to machine learning with Python: a guide for data scientists*. "O'Reilly Media, Inc.".

Narkhede, S. (2018). Understanding auc-roc curve. *Towards data science*, *26*(1), 220-227.

R. Fadhilah, H. Kuswanto and D. D. Prastyo, "Performance Analysis of Random Forest with Sampling for River Water Quality Classification," *2024 7th International Conference on Informatics and Computational Sciences (ICICoS)*, Semarang, Indonesia, 2024, pp. 456-461, doi: 10.1109/ICICoS62600.2024.10636858.

Ramyachitra, D., & Manikandan, P. (2014). Imbalanced dataset classification and solutions: a review. *International Journal of Computing and Business Research (IJCBR)*, *5*(4), 1-29.

Sahin, E. K. (2020). Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest. *SN Applied Sciences*, *2*(7), 1308.

Şahin, M. (2020). Impact of weather on COVID-19 pandemic in Turkey. *Science of the Total Environment*, *728*, 138810.

Spelmen, V. S., & Porkodi, R. (2018). A review on handling imbalanced data. In *2018 international conference on current trends towards converging technologies (ICCTCT)* (pp. 1-11). IEEE.

Stephens-Davidowitz, S. (2014). The cost of racial animus on a black candidate: Evidence using Google search data. *Journal of Public Economics*, *118*, 26-40.

Tyagi, S., & Mittal, S. (2020). Sampling approaches for imbalanced data classification problem in machine learning. In *Proceedings of ICRIC 2019: Recent innovations in computing* (pp. 209-221). Springer International Publishing.

Van Ravenzwaaij, D., Cassey, P., & Brown, S. D. (2018). A simple introduction to Markov Chain Monte-Carlo sampling. *Psychonomic bulletin & review*, *25*(1), 143-154.