Jurnal Teknik Informatika (JUTIF)
Vol. 5, No. 5, October 2024, pp. 1267-1276

DOI: https://doi.org/10.52436/1.jutif.2024.5.5.2066
p-ISSN: 2723-3863
e-ISSN: 2723-3871

CLASSIFICATION MODELS FOR ACADEMIC PERFORMANCE: A COMPARATIVE STUDY OF NAÏVE BAYES AND RANDOM FOREST ALGORITHMS IN ANALYZING UNIVERSITY OF LAMPUNG STUDENT GRADES

Dian Kurniasari[*1], Rekti Nurul Hidayah[2], Notiragayu[3], Warsono[4], Rizki Khoirun Nisa[5]

[1,2,3,4,5]Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Lampung, Indonesia
Email: [1]dian.kurniasari@fmipa.unila.ac.id, [2]rektinurul@gmail.com, [3]notiragayu@fmipa.unila.ac.id, [4]warsono.1963@fmipa.unila.ac.id, [5]rizkikhoirunnisa5@gmail.com

(Article received: May 08, 2024; Revision: May 24, 2024; published: October 20, 2024)

Abstract

At the university, students are provided with a comprehensive assessment of their academic achievements for each course completed at the end of every semester. This study aimed to compare the effectiveness of two classification methods, the Naïve Bayes and the Random Forest methods, in classifying student learning outcomes. The research process is segmented into various stages: data selection, data preparation, model building and testing, and model evaluation. The findings indicated that the Naïve Bayes and Random Forest approaches exhibited superior accuracy levels when employing data splitting strategies, in contrast to k-fold cross-validation. Based on the examination, the Random Forest approach demonstrated superiority in identifying the scores of University of Lampung students, achieving an accuracy percentage of 99.38%. Notably, both techniques showed a substantial performance improvement using Gradient Boosting. The Naïve Bayes method attained an accuracy rate of 99.89%, while the Random Forest method reached 99.45%. The results demonstrate that employing the Random Forest classification method consistently leads to superior performance in identifying and classifying student grades. Furthermore, using Gradient Boosting in the boosting process has demonstrated its efficacy in enhancing the classification methods' accuracy. These findings significantly contribute to the comprehension and advancement of evaluation systems for assessing student learning outcomes in the university environment.

Keywords: *Classification, Final grade, Naïve bayes, Random forest, Student*.

1. INTRODUCTION

Within the college setting, every Student receives an evaluation of their achieved learning objectives after each semester for every course they enrol in. This assessment aims to examine the efficacy of the lecture process and track students' academic success. The approach is additionally employed to identify students who fail to attain sufficient academic progress in a course. Hence, the ability to forecast and comprehend student performance is a crucial measure to attain academic objectives and enhance the overall standard of education. That enables educational institutions to offer prompt assistance to individuals and students in danger or need support early, leading to an enhanced learning experience and increased effectiveness of education as a whole [1, 2].

Prediction student performance using historical academic data is widely used in educational data mining. Educational data mining is a study topic that uses statistical approaches, data mining, and machine learning to analyze information found in educational settings, such as universities, learning management systems, and intelligent guidance systems [3]. Nevertheless, forecasting student performance is a formidable undertaking due to the diverse range of educational data that must be assessed, including tests, assignments, and special projects. Due to the intricate nature of the problem, there is a growing demand for efficient techniques or strategies to forecast student performance [4].

Classification is a data mining approach that uses machine learning to predict the membership of a sample of data into pre-defined classes and groups [5]. The Naïve Bayes Classifier is a commonly employed technique in data classification. This approach utilizes Bayes' theorem to combine preexisting and newly acquired information. The primary benefits of this approach include its simplicity in execution and its exceptional level of precision [6].

Haviluddin et al. [7] comprehensively analyze students' academic performance using the Naïve Bayes Classifier. The analysis used particular criteria, including age, place of birth, gender, school enrollment status, major, organization, and Grade Point Average (GPA). The research revealed that the categorization model achieved an accuracy rate of 76.79%.

Wibawa et al. [6] categorize the quality of a journal by its Quartile rating. The data is categorized

into distinct groups, specifically Q1, Q2, Q3, Q4, and Q5. The variables included in this analysis comprise the H-index, SJR, total documents (3 years), and total citations (3 years). Based on this investigation, it was determined that the Naïve Bayes Classifier algorithm successfully categorized the quality of journals. However, the accuracy achieved was suboptimal.

Random Forest is a machine learning technique that offers superior predictive accuracy compared to other models. The primary benefit of utilizing Random Forest for classification is its capacity to handle datasets with several predictor variables and its aptitude for comprehending the relationship between predictors and the obtained outcomes [8–10].

Ghosh and Janan [11] assess student performance using qualitative and quantitative elements such as psychology, college facilities, learning environment, and teaching influence. They employ the Random Forest algorithm for this purpose. The findings indicated that Random Forest achieved a classification accuracy of 96.88% across many classes.

Yagci [12] investigated the ability to forecast student academic performance using different machine learning algorithms, including Random Forest, Logistic Regression, Naïve Bayes, and k-Nearest Neighbors. The researchers utilized a dataset comprising the academic performance records of 1854 students who enrolled in Turkish Language courses at public universities in Turkey during the autumn semester of 2019-2020. The findings demonstrated that the suggested model attained an accuracy ranging from 70% to 75%, with Random Forest exhibiting a maximum accuracy of 74.6%.

This study compared the Naïve Bayes and Random Forest approaches for classifying the scores of University of Lampung students. The University of Lampung (UNILA) is a public institution in Bandar Lampung, in the province of Lampung. The University of Lampung's policies stipulate that the evaluation of learning processes and outcomes may take the form of quizzes, structured assignments, practical exams, Midterm Exams (UTS), Final Semester Exams (UAS), and class observations. The evaluation of academic achievements for students in diploma, bachelor, professional, master, and doctorate programs is indicated by quality letters and corresponding quality scores: A (4), B + (3.5), B (3), C + (2.5), C (2), D (1), and E (0). The objective of this study is to ascertain the superiority of one of these strategies by evaluating the ensuing accuracy.

2. RESEARCH METHODS

2.1. Data Pre-processing

Data pre-processing is the first step in machine learning, during which machines modify or transform data to be examined more efficiently [13]. The pre-processing stage accounts for approximately 50% to 80% of the time spent in the categorization process. That validates the significance of the pre-processing phase in constructing the model [14].

2.1.1. Missing Value

During the data pre-processing stage, the primary issue that frequently occurs is the presence of missing values. A missing value happens when a data point is absent for a specific variable or characteristic. This absence can disrupt the overall portrayal of the data by impacting its distribution and potentially introducing bias into the analysis results. Hence, it is imperative to handle missing variables to enhance the classification model's efficacy [15].

2.1.2. Data Normalization

Data normalization refers to altering or adjusting data to ensure each characteristic possesses an equivalent range of values. The objective is to mitigate the bias due to variations and numerical prevalence of features in discerning patterns. Normalization ensures that all characteristics are given equal importance when predicting the output class of unknown data. That is particularly valuable in statistical learning as it allows all features to contribute equally during the learning process [16].

Z-score normalization is a frequently used approach in the process of normalizing. This method is a normalization technique employed when the exact range of data is uncertain [17]. The process involves computing the average and standard deviation of the data and then utilizing this information to adjust the data so that it conforms to a z-score distribution, as described by Equation (1).

$$x'_{i,n} = \frac{x_{i,n} - \mu_i}{\sigma_i} \quad (1)$$

2.2. Data Splitting

Data splitting is a process that involves dividing a dataset into two distinct pieces known as 'training' and 'testing'. This step occurs after the pre-processing stage, once the samples in the dataset have been rectified to diminish noise or undesired fluctuations [18].

Typically, data splitting methods employ an 80:20 ratio to divide data into training and testing sets. Additional ratios, such as 90:10, 70:30, 60:40, and even 50:50, are also employed. Nevertheless, no conclusive manual specifies the optimal ratio for each dataset [19].

2.3. K-Fold Cross Validation

When data is partitioned into training data and test data using data splitting algorithms, there is a potential for overlooking crucial data points that may impact study objectives. This phenomenon arises due to the underutilization of a significant portion of the

data during the training phase, leading to the model's inability to detect crucial patterns [20].

Considering this, alongside the partitioning of the data into training and testing sets, this study will also implement k-fold cross-validation. K-fold cross-validation is a method that partitions a dataset into folds, enabling each data point to be used interchangeably as training and testing data. Consequently, the complete dataset will be utilized for training and assessing the model, thereby minimizing the possibility of losing crucial information and enhancing the accuracy of study findings.

2.4. Naïve Bayes Classifier

The Naive Bayes Classifier is a classification approach that utilizes Bayes' Theorem. This approach employs the principles of probability and statistics to forecast the probability of a future event based on prior experiences. The Naive Bayes Classifier utilizes previous data to create predictions about potential future events [21]. The generic formula for Naïve Bayes in a mathematical setting can be represented by Equation (2) as follows:

$$P(H|X) = \frac{P(X|H).P(H)}{P(X)} \quad (2)$$

2.5. Random Forest

Random Forest is a supervised learning technique that uses Homogeneous Ensemble Learning to aggregate the outcomes of decision trees in order to obtain the ultimate result. This algorithm uses bagging techniques or random feature selection to construct decision trees that are uncorrelated with one another [22, 23].

The Random Forest algorithm differs from the Decision Tree algorithm by randomly selecting a subset of features for each iteration instead of considering all potential feature separations. Random Forest is widely regarded as a highly adaptable and user-friendly method for addressing classification and regression problems [24]. In addition, the training process of Random Forest is often faster when compared to Decision Tree [25, 26].

2.6. Boosting

Boosting is an iterative method that adjusts the weighting of training data in each iteration. It increases the weights on misclassified samples and decreases the weights on correctly classified examples. This approach efficiently alters the allocation of training data [27]. The primary objective of boosting is to enhance the performance of the classification algorithm to its maximum potential. The various forms of boosting in machine learning encompass:

a. Adaptive Boosting (AdaBoost) is a widely utilized and extensively researched boosting technique routinely employed in numerous fields [28].
b. Gradient Boosting (GB) is a supervised learning method based on decision trees commonly used for classification issues. This method operates sequentially by accumulating past forecasts that do not align with the actual data and progressively rectifying faults in those earlier predictions [29].
c. Extreme Gradient Boosting (XGBoost) is a machine learning technique that utilizes the Gradient Boosting Decision Tree (GBDT) algorithm to address regression and classification issues. XGBoost is an ensemble technique that uses decision-making trees based on gradient boosting [30].

2.7. Confusion Matrix

A confusion matrix is a data analysis tool that takes the shape of a prediction matrix. It compares the predictions generated by a model with the actual state of the observed data. The objective is to furnish a lucid and precise depiction of the model's performance in data mining. Based on the comparison results, it is possible to generate other evaluation metrics, including accuracy, precision, and recall [25]. The following calculations can be used to quantify accuracy, precision, and recall by utilizing a confusion matrix:

a. Accuracy refers to the ratio of correct predictions to the total number of predictions made by the model. Mathematically, accuracy can be determined by utilizing Equation (3):

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

b. Precision is the level of exactness in making correct forecasts. Precision is a metric that quantifies the proportion of projected positive instances that are positive. The calculation of precision can be determined using Equation (4):

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

c. Recall is sometimes referred to as sensitivity or True Positive Rate. Recall quantifies the extent to which the model accurately identifies positive class instances. The calculation of recall can be determined using Equation (5):

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

3. RESULTS

3.1. Data Input and Selection

The first step in data processing entails the selection of pertinent datasets for research purposes.

The dataset comprises the grades of students enrolled at the University of Lampung during the 2021/2022 academic year's odd semester, including a total of 197,868 data entries. This data will serve as the foundation for the analysis conducted in this study. Table 1 represents a dataset consisting of the scores of University of Lampung students, which is utilized in the research.

Table 1. Research Data Sample

Courses	Number Value	Letter Value
Rekayasa Sistem	66.2	B
Rekayasa Sistem	76.5	A
Rekayasa Sistem	71.1	B+
Rekayasa Sistem	80.15	A
Rekayasa Sistem	66.1	B
…	…	…
Pendidikan Agama Islam	78.5	A
Kewirausahaan	71	B+
Perpetaan	77	A
Geofisika Teknik dan Lingkungan	75	B+
Metodologi Penelitian	62	C+

3.2. Data Preprocessing

The data pre-processing procedure encompasses multiple processes, which include:

1. Identification and Handling of Missing Value

Within this research dataset, two variables have incomplete data, specifically Number Value and Letter Value. The issue can be remedied by eliminating or substituting the missing value with an alternative value. Nevertheless, the methodology employed in this study involves the removal of data items that include missing values to mitigate potential bias resulting from the substitution of values. The data was decreased from 197,868 to 186,097.

2. Categorical Encoding

In this stage, categorical encoding is used to identify Course characteristics and Letter Grades as categorical data. In order to apply statistical algorithms or machine learning techniques to the data, it is necessary to translate these variables into a numerical representation. Consequently, the process of assigning labels to the data is accomplished by using the label encoder technique.

Table 2. Categorical Encoding Result

Courses	Number Value	Letter Value
2180	66.20	1
2180	76.50	0
2180	71.10	2
2180	80.15	0
2180	66.10	1
...	...	...
1899	78.50	0
954	71.00	2
1733	77.00	0
629	75.00	2
1304	62.00	4

The data in Table 2 has been annotated with encoder labels. Once the labelling process is complete, the categorical data is transformed into numerical representation, enabling more in-depth statistical or machine learning analysis.

3. Data Normalization

Table 2 reveals that the Number Value variable exhibits a significantly wide range of values. Hence, it is necessary to implement a data normalization procedure. The z-score normalization method was employed in this investigation. This stage aims to mitigate the disparity in size between variables, ensuring that the model training process is not biased towards variables with significantly larger value ranges. Implementing this approach can provide uniformity in outcomes analysis and enhance the model's efficiency. The findings of the data normalization process that has been conducted are shown in Table 3.

Table 3. Data Normalisation Results

Index	Number Value
0	-0.711493
1	0.218157
2	-0.269232
3	0.547596
4	-0.720519
…	…
186092	0.398672
186093	-0.278258
186094	0.263286
186095	0.082771
186096	-1.090573

3.3. Data Splitting

Once the data pre-processing phase is finished, the subsequent stage involves splitting the data into two distinct parts: training and testing data. Data splitting is crucial for evaluating the performance of pre-trained algorithms using previously unseen data.

The data will be divided into different ratios for training and testing purposes: 60% for training and 40% for testing, 70% for training and 30% for testing, 80% for training and 20% for testing, 90% for training, and 10% for testing. Table 4 displays the outcomes obtained by partitioning the data into training and test data using the specified data-splitting ratio. By employing appropriate proportions, it is anticipated that the constructed model will be able to provide precise and optimal forecasts and effective generalizations to novel data.

Table 4. Data Splitting

Ratio	Training	Testing
60% and 40%	111658	74439
70% and 30%	130267	55830
80% and 20%	148877	37220
90% and 10%	167487	18610

3.4. Process of Building and Evaluating Classification Model with Data Splitting Method

Once the data pre-processing is finished and the data is divided into training and test sets, the subsequent stage involves constructing a model using the Naïve Bayes and Random Forest methods, as outlined below:

3.4.1. Naive Bayes

The three types of modelling in Naïve Bayes are Gaussian Naïve Bayes, Multinomial Naïve Bayes, and Bernoulli Naïve Bayes. The modelling technique employed in this study was Gaussian Naïve Bayes. Gaussian Naïve Bayes is a simplified version of the Naïve Bayes classification method. It assumes that the data for each label follows a primary Gaussian distribution.

After constructing the model, the subsequent phase involves assessing its performance by utilizing a confusion matrix to understand its effectiveness comprehensively. The confusion matrix provides information regarding the number of accurate predictions (True Positives), inaccurate forecasts (False Positives), and errors such as False Negatives and True Negatives. By examining the confusion matrix, one can better understand the Naïve Bayes model's performance in data classification. The confusion matrix for each data splitting ratio is depicted in Figure 1-4 as shown below:

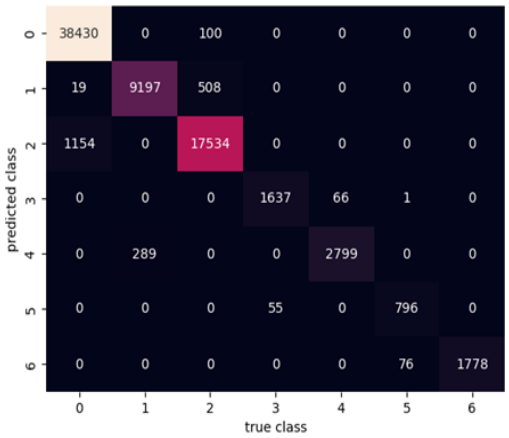

Figure 1. Confusion Matrix Naïve Bayes with a Ratio of 60% and 40%

$$\text{Accuracy} = \frac{38430 + 9197 + \ldots + 796 + 1778}{74439} = \frac{72171}{74439} = 0{,}9695$$

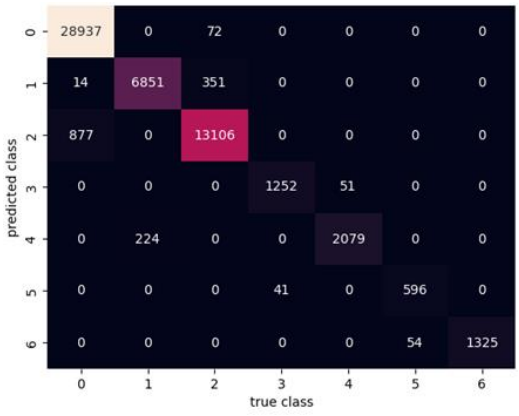

Figure 2. Confusion Matrix Naïve Bayes with a Ratio of 70% and 30%

$$\text{Accuracy} = \frac{28937 + 6851 + \ldots + 596 + 1325}{55809} = \frac{54146}{55809} = 0{,}9698$$

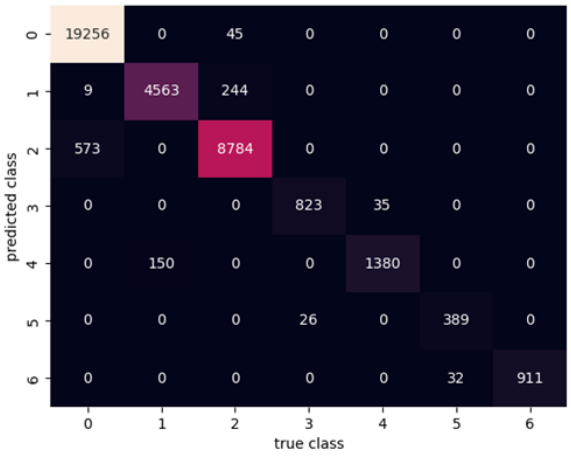

Figure 3. Confusion Matrix Naïve Bayes with a Ratio of 80% and 20%

$$\text{Accuracy} = \frac{19256 + 4563 + \ldots + 389 + 911}{37220} = \frac{36106}{37220} = 0{,}9700$$

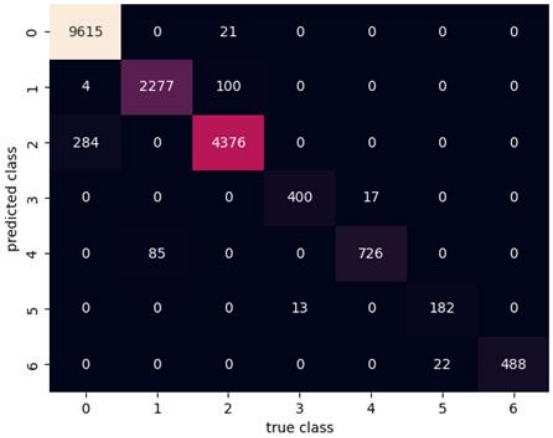

Figure 4. Confusion Matrix Naïve Bayes with a Ratio of 90% and 10%

$$\text{Accuracy} = \frac{9615 + 2277 + \ldots + 182 + 488}{18610} = \frac{18604}{18610} = 0{,}9707$$

The accuracy values derived from the calculations mentioned above can offer insights into the model's ability to categorize data using Gaussian Naïve Bayes modelling, as outlined in Table 5 below:

Table 5. Naïve Bayes Model Accuracy Results

Ratio	Accuracy
60% training and 40% testing	96,95%
70% training and 30% testing	96,98%
80% training and 20% testing	97,00%
90% training and 10% testing	**97,07%**

According to the statistics presented in Table 5, the most accurate results were achieved in data splitting, with a ratio of 90% for training data and 10% for test data, resulting in a 97.07% accuracy rate. The observed increase in accuracy demonstrates the effectiveness of the built Naïve Bayes model in accurately predicting data classes.

3.4.2. Random Forest

This study will utilize the Random Forest approach for modelling, employing the RandomForestClassifier function from the sklearn.ensemble library. The parameter to be utilized is "n_estimators," which governs the quantity of trees to be constructed in the model.

The constructed model is subsequently assessed using a confusion matrix for each implemented data-splitting mechanism. The findings are depicted in Figure 5-8 as shown:

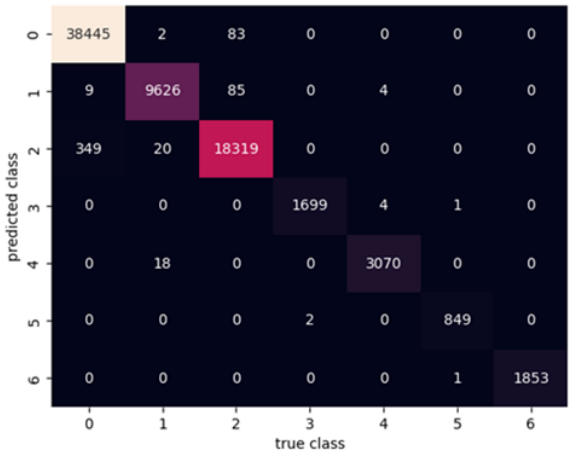

Figure 5. Confusion Matrix Random Forest with a Ratio of 60% and 40%

$$\text{Accuracy} = \frac{38445 + 9626 + \ldots + 849 + 1853}{74366} = \frac{73861}{74366} = 0{,}9932$$

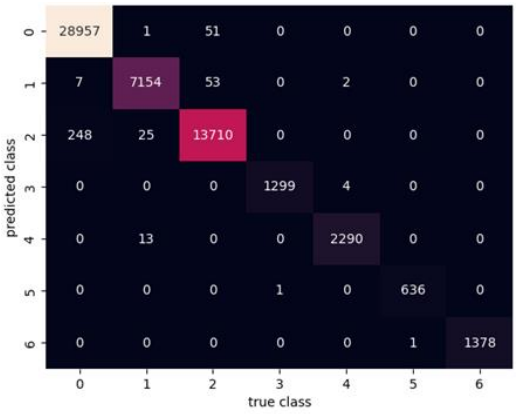

Figure 6. Confusion Matrix Random Forest with a Ratio of 70% and 30%.

$$\text{Accuracy} = \frac{28957 + 7154 + \ldots + 636 + 1378}{55809} = \frac{55424}{55809} = 0{,}9931$$

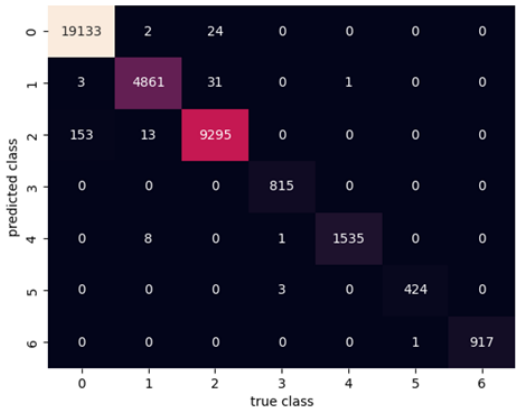

Figure 7. Confusion Matrix Random Forest with a Ratio of 80% and 20%

$$\text{Accuracy} = \frac{19133 + 4861 + \ldots + 424 + 917}{37220} = \frac{36980}{37220} = 0{,}9935$$

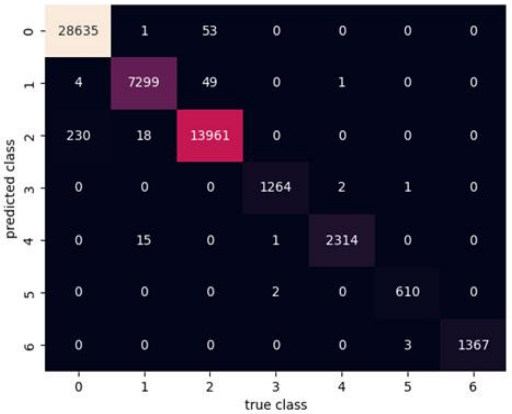

Figure 8. Confusion Matrix Random Forest with a Ratio of 90% and 10%

$$\text{Accuracy} = \frac{28635 + 7299 + \ldots + 610 + 1367}{55795}$$
$$= \frac{55450}{55795}$$
$$= 0{,}9938$$

According to the study of the confusion matrix above, it is evident that the optimal ratio for the Random Forest model is 90% for training and 10% for testing, resulting in an accuracy of 99.38%. The comprehensive accuracy tabulation results for the Random Forest model are presented in Table 6 as follows:

Table 6. Random Forest Model Accuracy Results

Ratio	Accuracy
60% training and 40% testing	99,32%
70% training and 30% testing	99,31%
80% training and 20% testing	99,35%
90% training and **10% testing**	**99,38%**

3.5. Process of Building and Evaluating Classification Model with K-Fold Cross Validation Method

K-fold is a widely used cross-validation approach that involves dividing the dataset into k groups, or folds, and repeating the process k times. This study included 5-fold, 8-fold, and 10-fold cross-validation folds. In the k-fold cross-validation procedure, each fold will be sequentially utilized as the test data, while the remaining folds will serve as the training data. Maximizing the utilization of available data enhances the accuracy of testing model performance, hence increasing the reliability of model evaluation outcomes. The study can assess model stability and performance in different test scenarios by choosing different variations in k-fold values.

The validation results of the Naive Bayes and Random Forest models using k-fold cross-validation are displayed in Table 7 and Table 8, respectively.

Table 7. Naïve Bayes Model Accuracy Results

K-Fold Cross Validation	Accuracy
5-fold	97,03%
8-fold	97,04%
10-fold	**97,05%**

Table 8. Random Forest Model Accuracy Results

K-Fold Cross Validation	Accuracy
5-fold	97,89%
8-fold	97,82%
10-fold	**97,92%**

Tables 7 and 8 indicate that the ideal k-fold value for the Naive Bayes and the Random Forest models is 10-fold.

3.6. Model Comparison

The optimal outcomes for each examination category are achieved through a sequence of conducted assessments. Hence, the subsequent phase of this invvestigation entails juxtaposing the outcomes derived from the model with the employed data-splitting technique. This comparison aims to enhance our comprehension of the efficacy of both approaches within the particular setting of the study. Therefore, assessing the benefits, drawbacks, and circumstances in which each approach may be more suitable is possible. Figure 9 comprehensively analyses the outcomes obtained by comparing each model and the data-splitting scheme.

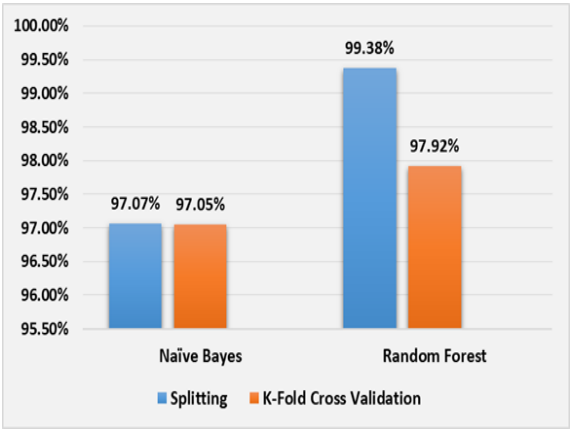

Figure 9. Comparison of Naïve Bayes and Random Forest Models

Based on the examination of the graph provided in Figure 9, it can be inferred that both the Naive Bayes and Random Forest methods exhibit superior accuracy when utilizing the data splitting approach, as opposed to the k-fold cross-validation method. The Naive Bayes approach produced an accuracy rate of 97.07%, while the Random Forest method achieved an accuracy rate of 99.38%, which was the most outstanding result.

3.7. Boosting

This study utilizes the Gradient Boosting method with a data-splitting ratio of 90% for training and 10% for testing. Additionally, at this juncture, the modifications will be examined to the precision of the Naive Bayes and Random Forest models before and after the boosting procedure.

By implementing this strategy, it is anticipated that the predictive capability of the original model will be enhanced for both Naive Bayes and Random Forest. The comparison findings of the models before and after boosting are presented in Table 9 as follows:

Table 9. Boosting Results

Model	Accuracy	
	Before	After
Naïve Bayes	97,07%	99,45%
Random Forest	99,38%	99,89%

Table 9 demonstrates the accuracy comparison between the Naïve Bayes and Random Forest models before and after using the boosting approach. Overall, the precision of both models was enhanced following the implementation of the boosting procedure.

Firstly, let us discuss the Naïve Bayes model. Before boosting, this model had an accuracy rate of 97.07%, indicating a significantly high performance level and demonstrating its classification capability. Nevertheless, following the boosting technique, there was a substantial augmentation of 2.38%, resulting in the model's accuracy reaching 99.45%. That demonstrates that the boosting strategy significantly enhances the performance of the Naïve Bayes model in classification.

Furthermore, it is worth noting that the Random Forest model has a remarkable accuracy rate of 99.38% even before any boosting techniques are applied. Despite already having a high level of accuracy, the boosting method further enhances the performance of this model, resulting in an accuracy of 99.89%. These findings suggest that while the Random Forest model has been highly proficient in prior classifications, boosting strategies can further enhance accuracy.

4. DISCUSSIONS

Based on the description and findings, the data splitting approach is more advantageous in generating a precise model compared to k-fold cross-validation when classifying the final grades of University of Lampung students. Nevertheless, these outcomes are impacted by other factors, including the size of the dataset, the distribution of classes, and the complexity of the model. Hence, the choice of validation techniques should be based on the dataset's distinctive attributes and the research objectives.

This study utilized boosting techniques along with the Gradient Boosting algorithm, employing a 90% data allocation for training and a 10% allocation for testing. The boosting strategy demonstrated efficacy in enhancing the performance of classification models. The notable enhancement in precision suggests that this approach should be considered and implemented when building classification models for similar datasets.

The findings of this study align with prior research conducted by Yagci [12], which concluded that the Random Forest model outperformed other machine learning models, including Naïve Bayes. The Random Forest model described in this work outperforms their model, with an accuracy of 97.07% before and 99.89% after boosting. Meanwhile, their proposed model demonstrates a level of accuracy of 74.6%.

5. CONCLUSION

Testing has been conducted on the Naïve Bayes and Random Forest algorithms for classifying student scores at the University of Lampung. According to the investigation, when employing the data splitting technique, both the Naïve Bayes and Random Forest approaches achieved the highest accuracy levels compared to k-fold cross-validation.

After evaluating the model, it has been determined that the Random Forest approach is the optimal selection for classifying student grades at the University of Lampung. This method achieves a remarkable accuracy rate of 99.38%. However, the Naïve Bayes technique exhibits a slightly lower accuracy rate of 97.07%.

Moreover, the efficacy of both approaches can be enhanced by implementing boosting techniques via Gradient Boosting. Post-enhancement, the Naïve Bayes approach exhibited an accuracy rate of 99.45%. However, the Random Forest method outperformed it with a better accuracy rate of 99.89%. The results demonstrate that the implementation of boosting techniques substantially enhances the performance of both methods in categorizing student grades at the University of Lampung.

REFERENCES

[1] M. A. Baig, S. A. Shaikh, K. K. Khatri, M. A. Shaikh, M. Z. Khan, and M. A. Rauf, "Prediction of Students Performance Level Using Integrated Approach of ML Algorithms," *Int. J. Emerg. Technol. Learn.*, vol. 18, no. 1, 2023, doi: 10.3991/ijet.v18i01.35339.

[2] A. A. Nafea, M. Mishlish, A. M. S. Shaban, M. M. AL-Ani, K. M. Ali Alheeti, and H. J. Mohammed, "Enhancing Student's Performance Classification Using Ensemble Modeling," *Iraqi J. Comput. Sci. Math.*, vol. 4, no. 4, 2023, doi: 10.52866/ijcsm.2023.04.04.016.

[3] M. Nachouki, E. A. Mohamed, R. Mehdi, and M. Abou Naaj, "Student course grade prediction using the random forest algorithm: Analysis of predictors' importance," *Trends Neurosci. Educ.*, vol. 33, 2023, doi: 10.1016/j.tine.2023.100214.

[4] F. Ofori, E. Maina, and R. Gitonga, "Using Machine Learning Algorithms to Predict Students' Performance and Improve Learning Outcome: A Literature-Based Review," *J. Inf. Technol.*, vol. 4, no. 1, 2020.

[5] S. Sharma, J. Agrawal, S. Agarwal, and S. Sharma, "Machine learning techniques for data mining: A survey," in *2013 IEEE International Conference on Computational Intelligence and Computing Research, IEEE ICCIC 2013*, 2013. doi: 10.1109/ICCIC.2013.6724149.

[6] A. P. Wibawa *et al.*, "Naïve Bayes Classifier for Journal Quartile Classification," *Int. J. Recent Contrib. from Eng. Sci. IT*, vol. 7, no. 2, 2019, doi: 10.3991/ijes.v7i2.10659.

[7] Haviluddin, N. Dengen, E. Budiman, M. Wati, and U. Hairah, "Student Academic Evaluation using Naïve Bayes Classifier

Algorithm," in *Proceedings - 2nd East Indonesia Conference on Computer and Information Technology: Internet of Things for Industry, EIConCIT 2018*, 2018. doi: 10.1109/EIConCIT.2018.8878626.

[8] M. Fernández-Delgado, E. Cernadas, S. Barro, and D. Amorim, "Do we need hundreds of classifiers to solve real-world classification problems?," *J. Mach. Learn. Res.*, vol. 15, 2014.

[9] J. L. Speiser, V. L. Durkalski, and W. M. Lee, "Random forest classification of etiologies for an orphan disease," *Stat. Med.*, vol. 34, no. 5, 2014, doi: 10.1002/sim.6351.

[10] J. L. Speiser, M. E. Miller, J. Tooze, and E. Ip, "A comparison of random forest variable selection methods for classification prediction modeling," *Expert Systems with Applications*, vol. 134. 2019. doi: 10.1016/j.eswa.2019.05.028.

[11] S. K. Ghosh and F. Janan, "Prediction of student's performance using random forest classifier," in *Proceedings of the International Conference on Industrial Engineering and Operations Management*, 2021. doi: 10.46254/an11.20211238.

[12] M. Yağcı, "Educational data mining: prediction of students' academic performance using machine learning algorithms," *Smart Learn. Environ.*, vol. 9, no. 1, 2022, doi: 10.1186/s40561-022-00192-z.

[13] K. Maharana, S. Mondal, and B. Nemade, "A review: Data pre-processing and data augmentation techniques," *Glob. Transitions Proc.*, vol. 3, no. 1, pp. 91–99, Jun. 2022, doi: 10.1016/j.gltp.2022.04.020.

[14] A. I. Kadhim, "An Evaluation of Pre-processing Techniques for Text Classification," *Int. J. Comput. Sci. Inf. Secur.*, vol. 16, no. 6, pp. 22–32, 2018, [Online]. Available: https://sites.google.com/site/ijcsis/

[15] S. Roy, P. Sharma, K. Nath, D. K. Bhattacharyya, and J. K. Kalita, "Pre-processing: A data preparation step," in *Encyclopedia of Bioinformatics and Computational Biology: ABC of Bioinformatics*, vol. 1–3, 2018. doi: 10.1016/B978-0-12-809633-8.20457-3.

[16] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Appl. Soft Comput.*, vol. 97, 2020, doi: 10.1016/j.asoc.2019.105524.

[17] H. Aji Prihanditya and N. Hestu Aji Prihanditya, "The Implementation of Z-Score Normalization and Boosting Techniques to Increase Accuracy of C4.5 Algorithm in Diagnosing Chronic Kidney Disease," *J. Soft Comput. Explore.*, vol. 1, no. 1, pp. 63–69, 2020.

[18] C. L. M. Morais, M. C. D. Santos, K. M. G. Lima, and F. L. Martin, "Improving data splitting for classification applications in spectrochemical analyses employing a random-mutation Kennard-Stone algorithm approach," *Bioinformatics*, vol. 35, no. 24, 2019, doi: 10.1093/bioinformatics/btz421.

[19] V. R. Joseph, "Optimal ratio for data splitting," *Stat. Anal. Data Min.*, vol. 15, no. 4, pp. 531–538, 2022, doi: 10.1002/sam.11583.

[20] N. Alifiah, D. Kurniasari, Amanto, and Warsono, "Prediction of COVID-19 Using the Artificial Neural Network (ANN) with K-Fold Cross-Validation," *J. Inf. Syst. Eng. Bus. Intell.*, vol. 9, no. 1, pp. 16–27, 2023, doi: 10.20473/jisebi.9.1.16-27.

[21] M. Garonga and Rita Tanduk, "Comparison of Naive Bayes, Decision Tree, and Random Forest Algorithms in Classifying Learning Styles of Universitas Kristen Indonesia Toraja Students," *J. Tek. Inform.*, vol. 4, no. 6, 2023, doi: 10.52436/1.jutif.2023.4.6.1020.

[22] I. D. Mienye, Y. Sun, and Z. Wang, "An improved ensemble learning approach for the prediction of heart disease risk," *Informatics Med. Unlocked*, vol. 20, 2020, doi: 10.1016/j.imu.2020.100402.

[23] R. A. Putri and N. S. Fatonah, "Perbandingan Metode Klasifikasi serta Analisis Faktor Akademis Pola Kelulusan Mahasiswa di Perguruan Tinggi," *J. Inform. J. Pengemb. IT*, vol. 7, no. 2, 2022, doi: 10.30591/jpit.v7i2.3082.

[24] D. A. Rachmawati, N. A. Ibadurrahman, J. Zeniarja, and N. Hendriyanto, "Implementation of the Random Forest Algorithm in Classifying the Accuracy of Graduation Time for Computer Engineering Students at Dian Nuswantoro University," *J. Tek. Inform.*, vol. 4, no. 3, 2023, doi: 10.52436/1.jutif.2023.4.3.920.

[25] G. A. Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest," *CogITo Smart J.*, vol. 6, no. 2, 2020, doi: 10.31154/cogito.v6i2.270.167-178.

[26] J. Muliawan and E. Dazki, "Sentiment Analysis of Indonesia's Capital City Relocation using Three Algorithms: Naïve Bayes, KNN, and Random Forest," *J. Tek. Inform.*, vol. 4, no. 5, 2023, doi: 10.52436/1.jutif.2023.4.5.1436.

[27] A. R. Arrahimi, M. K. Ihsan, D. Kartini, M. R. Faisal, and F. Indriani, "Teknik Bagging Dan Boosting Pada Algoritma CART Untuk Klasifikasi Masa Studi Mahasiswa," *J. Sains*

dan Inform., vol. 5, no. 1, 2019, doi: 10.34128/jsi.v5i1.171.

[28] Y. Pristyanto, "Penerapan Metode Ensemble untuk Meningkatkan Kinerja Algoritma Klasifikasi pada Imbalanced Dataset," *J. Teknoinfo*, vol. 13, no. 1, 2019, doi: 10.33365/jti.v13i1.184.

[29] S. E. Suryana, B. Warsito, and S. Suparti, "Penerapan Gradient Boosting dengan Hyperopt untuk Memprediksi Keberhasilan Telemarketing Bank," *J. Gaussian*, vol. 10, no. 4, 2021, doi: 10.14710/j.gauss.v10i4.31335.

[30] Ichwanul Muslim Karo Karo, "Implementasi Metode XGBoost dan Feature Importance untuk Klasifikasi pada Kebakaran Hutan dan Lahan," *J. Softw. Eng. Inf. Commun. Technol.*, vol. 1, no. 1, 2020.