Matrik: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer
Vol. 24, No. 1, November 2024, pp. 117∼128
ISSN: 2476-9843, accredited by Kemenristekdikti, Decree No: 200/M/KPT/2020
DOI: 10.30812/matrik.v24i1.4031

Implementation of The Extreme Gradient Boosting Algorithm with Hyperparameter Tuning in Celiac Disease Classification

Roudlotul Jannah Alfirdausy , Nurissaidah Ulinnuha , Wika Dianita Utami
Universitas Islam Negeri Sunan Ampel, Surabaya, Indonesia

Article Info

Article history:

Received May 06, 2024
Revised August 13, 2024
Accepted October 25, 2024

Keywords:

Celiac disease
Extreme Gradient Boosting
Hyperparameter tuning

ABSTRACT

Celiac Disease (CeD) is an autoimmune disorder triggered by gluten consumption and involves the immune system and HLA in the intestine. The global incidence ranges from 0.5%-1%, with only 30% correctly diagnosed. Diagnosis remains challenging, requiring complex tests like blood tests, small bowel biopsy, and elimination of gluten from the diet. Therefore, a faster and more efficient alternative is needed. Extreme Gradient Boosting (XGBoost), an ensemble machine learning technique that utilizes decision trees to aid in the classification of Celiac disease, was used. The aim of this study was to classify patients into six classes, namely potential, atypical, silent, typical, latent and none disease, based on attributes such as blood test results, clinical symptoms and medical history. This research method employs 5-fold cross-validation to optimize parameters that are max depth, n estimator, gamma, and learning rate. Experiments were conducted 96 times to get the best combination of parameters. The results of this research are highlighted by an improvement of 0.45% above the accuracy value with the default XGBoost parameter of 98.19%. The best model was obtained in the trial with parameters max depth of 3, n estimator of 100, gamma of 0, and learning rate of 0.3 and 0.5 after modifying the parameters, yielding an accuracy rate of 98.64%, a sensitivity rate of 98.43%, and a specificity rate of 99.72%. This research shows that tuning the XGBoost parameters for Celiac disease classification can improve the model's accuracy compared to using the default parameters.

Corresponding Author:

Nurissaidah Ulinnuha, +6282132233280
Faculty of Science and Technology
Universitas Islam Negeri Sunan Ampel, Surabaya, Indonesia
Email: nuris.ulinnuha@uinsa.ac.id

How to Cite:
R. Alfirdausy, N. Ulinnuha, and W. Utami, "Implementation of The Extreme Gradient Boosting Algorithm with Hyperparameter Tuning in Celiac Disease Classification", *MATRIK: Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer*, Vol. 24, No. 1, pp. 117-128, November, 2024.

1. INTRODUCTION

Celiac disease (CeD) in the Greek term ”Koiliakos” meaning ”stomach” is an immune system disorder that primarily impacts the small intestine [1]. CeD is an autoimmune condition triggered by the consumption of gluten, which affects the immune system in the gut through Human Leukocyte Antigen (HLA). It is an autoimmune enteropathy (intestinal disease) that occurs in patients with genetic susceptibility (genetic predisposition) [2]. Gluten sensitivity, a protein present in grains such as wheat, barley, barley, rye, and kamut, characterizes the illness. This indicates that persons with CeD are not only gluten intolerant, but also foods made from these grains [3]. Celiac disease can harm the small intestine, limit nutrition absorption, and reduce people's quality of life at any age [4]. Damage to the small intestine impairs nutrition absorption, possibly leading to stunted development, vitamin shortages, and catastrophic effects such as anaemia, osteoporosis, increased risk of diabetes, autoimmune thyroid disease, and bowel cancer. CeD affects one in every 100 persons globally. However, only around 30% of cases are appropriately identified [5]. Common prevalence of CeD is recorded in populations ranging from 0.5% to 1% worldwide. This is mostly owing to improved diagnostic and screening methods for at-risk populations. Celiac disease is prevalent in varying degrees in different nations. Celiac disease prevalence varies by country, with the greatest rate documented in Western Sahara, reaching 5.6% of the population, while the prevalence in Indonesia is believed to be less than 1% [6].

Diagnosis of CeD remains difficult for medical practitioners. To avoid more serious problems, a precise and quick diagnosis is required. However, diagnosing CeD frequently necessitates a battery of sophisticated and costly testing, such as blood tests, endoscopic biopsies of the small intestine, and gluten elimination [7]. Celiac disease frequently necessitates a battery of medical tests, including blood testing to detect the presence of certain antibodies and, if necessary, a small intestinal endoscopy for confirmation. This difficulty in identification highlights the necessity of a better knowledge of celiac disease's symptom range and enhanced medical awareness of its various clinical manifestations. Machine learning has shown considerable promise in the medical area in recent years, including illness detection. Ensemble learning is a way of mixing machine learning algorithmsto improve prediction accuracy over individual machine learning methods. Gradient Boosting Machine (GBM) is a model of boosting algorithm. Gradient boosting is basically an ensemble learning technique that is built using the decision tree algorithm as its main component [8]. Extreme Gradient Boosting (XGBoost) has been acknowledged as a very sophisticated algorithm that performs exceptionally well in classification and regression problems. XGBoost is capable of avoiding overfitting issues and producing excellent predictions, particularly in the presence of outliers and data with imbalances [9]. XGBoost not only allows users to create reliable classification and regression models on tabular data, but it is also a useful and efficient tool for performing time series forecasting with high accuracy [10]. XGBoost uses a decision tree as the base learner and builds an additive expansion of the objective function to minimize the loss function. However, XGBoost has better scalability which is able to perform optimization faster than GBM [11].

Previous research has evaluated the performance of the XGBoost algorithm in various contexts. First, in the context of virus attack classification on the internet with unbalanced data, XGBoost achieved 100% precision, 99.8% recall, and 99.9% F1-score [12]. Then, in the sentiment analysis of Trip Advisor Hotel Reviews, it compared the performance of three machine learning algorithms XGBoost Classifier, Decision Tree Classifier, and Support Vector Machine (SVM). XGBoost led with 99% accuracy, outperforming the Decision Tree Classifier (97%) and Support Vector Machine (98%) [13]. In addition, another study showed that XGBoost is the best algorithm for disease detection compared to Random Forest, K-Nearest Neighbor, Multi-Layer Perceptron, SVM, Logistic Regression, and GBM on heart disease classification with 95.08% accuracy, high sensitivity (93.80%), and best precision (96.80%) [14]. Finally, in the context of diabetes classification, after hyperparameter tuning, XGBoost achieved an accuracy of 90.10% which was previously only 87.50%, precision 87.87%, recall 84.05%, F1-score 85.92%, and Area Under The Curve (AUC) score 0.960. In a recent study, XGBoost has also been successfully applied in the prediction of electroencephalogram (EEG) results for epilepsy diagnosis, showing significant accuracy in identifying complex epileptic patterns [15]. Another study confirmed the effectiveness of XGBoost in medical image analysis for breast cancer detection with accuracy reaching 96%, which beat several other methods in a comparative study [16]. Finally, in the context of anomaly detection on computer networks, XGBoost has demonstrated its ability to identify attacks with a high detection rate and low error rate [17].

Other studies have shown that the XGBoost model performs very well in health classification cases, especially in disease detection. In Parkinson's disease classification, XGBoost achieved an accuracy of 84% [18]. XGBoost as a Covid-19 pre-screening tool through a person's breathing recordings, this model shows significant potential with 86.2% accuracy [19]. In addition, in the detection of the risk of death of patients with sepsis, XGBoost achieved the best accuracy of 89.5% [20] and XGBoost provides an accuracy of 94.74% for the detection and analysis of breast cancer [21]. This shows that XGBoost is a reliable and effective model in healthcare classification applications. Thus, XGBoost has proven to be effective in various applications and can be improved through hyperparameter tuning. Seeing an accurate and timely diagnosis process of celiac disease is very important to avoid more serious complications with a series of expensive and complicated tests. **There is a gap** that has not been resolved by previous research, namely the lacking in the optimization of XGBoost parameters specifically in the context of celiac disease classification. Previous

studies generally consider the performance of XGBoost in various applications, but rarely specialize in celiac disease with specifically optimized parameters to improve the accuracy and precision of diagnosis. **The difference between** this research and previous research is the focus on optimizing specific XGBoost parameters for celiac disease classification. Thus, **the main contribution** of this research is the adjustment of the XGBoost model parameters through a series of trials and analysis to determine the optimal combination of parameters because **the purpose** of this research is to improve the accuracy and efficiency of the XGBoost model. **The purpose** of this research is to improve the accuracy of the XGBoost model in Celiac disease classification through optimization of key parameters such as learning rate, gamma, max depth, and n_estimator. **This research contributes** to finding the optimal combination of parameters that can improve the performance of the model, thus supporting more accurate diagnosis of Celiac disease.

2. RESEARCH METHOD

2.1. Dataset Description and Flowchart

The data used in this study is a collection of medical data on laboratory checks of patients indicated to have celiac disease. This data was obtained from the Wageningen University & Research Biotechnology Department published by research. This research data has 13 independent variables as attributes and 1 dependent variable as the target variable. The dataset consists of 6 diagnoses, namely 545 potential class data, 301 atypical class data, 400 silent class data. 380 typical class data, 301 latent class data and 350 none disease class data. This study was conducted to show the best classification model based on accuracy, sensitivity, and precision values. Results The results of this study were processed using the Extreme Gradient Boosting method by testing the learning rate, max depth, gamma and n estimator parameters. Figure 1 shows the flowchart of this study.

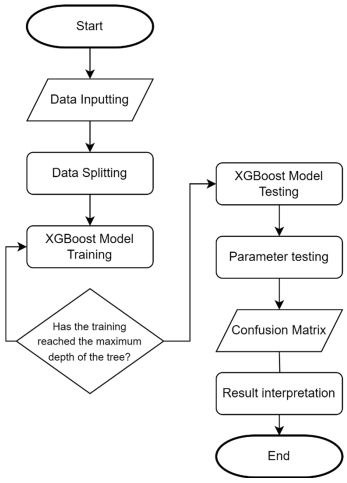


Figure 1. XGboost classification research flowchart

The process starts with entering the data to be used in the XGBoost analysis. After the data is inputted, the data is divided into training and testing data using the k-fold cross-validation method with the number of folds of 5 and 10, which ensures the model is well validated to avoid overfitting. Next, the XGBoost model is trained using the training data, and each training iteration is evaluated whether it has reached the maximum depth of the decision tree (max_depth). If it has not, training is continued until the limit is met. After training is complete, the model is tested using the testing data to evaluate its ability to predict new data. In the parameter testing stage, various combinations of key parameters such as max_depth, n_estimator, gamma, and learning rate are tested to find the configuration that produces the best performance. The test results are then evaluated using Confusion Matrix to calculate performance metrics such as accuracy, sensitivity, and specificity. The final stage is the interpretation of the results, which includes an in-depth analysis of how each parameter affects the performance of the XGBoost model, including its impact on accuracy.

2.2. Classification

Classification methods are used in supervised learning, where output labels are grouped. It is possible to find ways to distinguish one label from another by understanding patterns from pre-existing class or attribute data [22]. The purpose of classification

is to categorize or allocate new objects into pre-labeled groups. The classification process consists of two steps. The first step is to create a data training model and perform training on the existing dataset. The second step is to check the accuracy of the data training model, and if the accuracy of the model is satisfactory, then the model is used to classify data with inappropriate class labels [23]. Classification is a process that involves evaluating data objects and assigning them to a particular class from a predefined set of classes. There are two main tasks involved in classification. The first task is to create a model or prototype that serves as a reference and is stored in memory. This model is built based on the characteristics and features of the existing data objects. The second task is to utilize the built model to analyze, classify, or predict other data objects and determine the most suitable class for each object based on the model's criteria. This helps in effectively organizing and categorizing data objects into appropriate classes [16].

2.3. Ensemble learning

Ensemble learning is a machine learning method that integrates the predictions of multiple models to produce a more robust prediction model. The idea behind ensemble learning is to combine the predictions of multiple models so that the overall performance can be stronger and more accurate than individual models [17]. Boosting is one of the ensemble learning techniques. Boosting is a technique that focuses on improving the performance of weak learners (models that perform slightly better than random chance) by repeatedly overweighting misclassified examples and creating new models that focus on those misclassifications [24]. The main goal of boosting is to combine several weak models or learners that can give slightly better predictions than random guesses, into a strong and accurate model. The architecture of Boosting Algorithm is shown in Figure 2.

Figure 2. Architecture of Boosting Algorithm [25]

2.4. K-Fold Cross Validation

K-fold cross-validation is a model validation method where the dataset is divided into K subsets of uniform size. Each subset in turn serves as the testing dataset, while the other subset becomes the training dataset. This process is repeated until all subsets have served as test data. By using K-Fold Cross Validation, classification can be done more accurately and can avoid bias that may occur. Training and testing are performed K times, the flow of the cross-validation process is depicted in Figure 3.

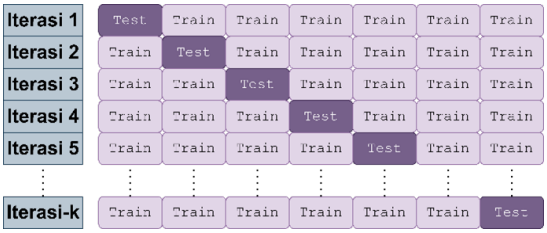


Figure 3. K-fold cross-validation

2.5. Extreme Gradient Boosting

XGBoost is an ensemble learning method of gradient boosting using a decision tree as the base learner [26]. XGBoost is a powerful ensemble learning that combines multiple individual decision trees to improve prediction accuracy [9]. Through ensemble

learning, each subsequent tree in the model aims to reduce the error or residual value of the previous tree's prediction. In XGBoost, the first tree predicts the target variable (Y) based on the input features. Subsequent trees focus on predicting the difference ($\mid Y - Y1 \mid$) between the actual target value and the cumulative prediction of the previous tree, which is shown in Figure 4. By iteratively adding trees and updating predictions, XGBoost progressively minimizes the overall error and improves the prediction performance of the model [27]. The predicted value at step t is denoted $\hat{y}_i^{(t)}$ by Equation 1. $f_k(x_i)$ describes the tree model. For y_i is obtained from the calculation in Equation 2.

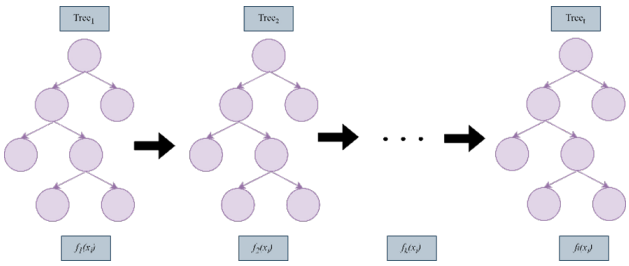


Figure 4. XGBoost forming decision tree

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) \tag{1}$$

$$\begin{aligned} \hat{y}_i^{(0)} &= 0 \\ \hat{y}_i^{(1)} &= f_1(x_1) = \hat{y}_i^{(0)} + f_1(x_1) \\ \hat{y}_i^{(2)} &= f_1(x_1) + f_2(x_2) = \hat{y}_i^{(1)} + f_2(x_2) \\ &\vdots \\ \hat{y}_i^{(t)} &= \hat{y}_i^{(t-1)} + f_t(x_i) \\ \hat{y}_i^{(t)} &= \textstyle\sum_{k=1}^{t} f_k(x_i) \end{aligned} \tag{2}$$

Where, $\hat{y}_i^{(t)}$ represents the final tree model, $\hat{y}_i^{(t-1)}$denotes the tree model generated previously, $f_t(x_i)$is the new model built and t refers to the total number of base tree models, indicating the iterative process of building and refining models Determining the optimal number of trees and depth in the XGBoos) algorithm has important significance. The problem of finding the optimal parameters can be addressed by introducing a new classification approach that is able to reduce the loss function, with the loss function described in Equation 3.

$$\text{Obj}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y}_i^{(t)}) + \sum_{i=1}^{t} \Omega(f_i) \tag{3}$$

Where, $\hat{y}_i^{(t)}$ refers to predicted value, while y_i is the actual value. The function $l(\hat{y}_i^{(t)}, y_i)$ is the loss function, used to measure the difference between the predicted and actual values. Additionally, $\omega(f_i)$ is the regularization parameter, which helps to prevent overfitting by controlling the complexity of the model. Since the ensemble tree model in equation 3 serves as a parameter and cannot be optimized via traditional optimization methods in Euclidean space, the approach is replaced with an additively trained model. This replacement involves using $\hat{y}_i^{(t)}$ at the i-th prediction and t-th iteration, creating a more flexible and effective approach to improving model performance.

2.6. Evaluation Method

The confusion matrix records the result of analyzing the performance of the classification model. Confusion matrix is used to evaluate and record the classification results by comparing the predicted and actual values of the data objects [28]. Confusion matrix tables are used to measure the performance of the model, including accuracy, sensitivity and specificity, using Figure 5. Values for overall accuracy, sensitivity, and specificity are calculated. This may be accomplished by taking the average of the overall sensitivity

and specificity for each class. This is done to gain a sense of the model's overall performance. The following is an evaluation based on the confusion matrix. The True Positive (TP), False Positive (FP), False Negative (FN), and True Negative (TN) values for each class can be calculated using Equation 4, 5, 6 and 7.

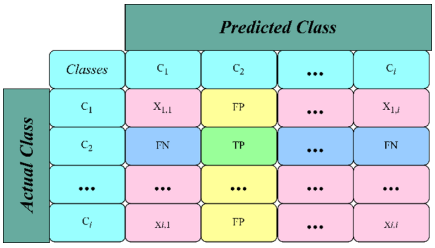


Figure 5. Multiclass confusion matrix

$$TP_{c(i)} = X_{i,i} \tag{4}$$

$$FN_i = \sum_{n}^{j=1} a_{i,j}; a_{i,j} = \begin{cases} x_{i,j}, & \text{if } i \neq j \\ 0, & \text{other} \end{cases} \tag{5}$$

$$FP_i = \sum_{n}^{j=1} a_{i,j}; a_{i,j} = \begin{cases} x_{i,j}, & \text{if } i \neq i \\ 0, & \text{other} \end{cases} \tag{6}$$

$$TN_i = \sum_{j=1}^{n} \sum_{k=1}^{n} c_{j,k}; c_{j,k} = \begin{cases} x_{j,k}, & \text{if } j \neq i \text{ and } k \neq i \\ 0, & \text{other} \end{cases} \tag{7}$$

Next, the calculation of the overall accuracy, sensitivity and specificity values. This calculation can be done by calculating the average of the total sensitivity and specificity for each class. This is done to get an overview of the overall model performance. Evaluation based on Confusion Matrix as follows: (a) Accuracy shows how accurately the model classifies each data set correctly with the formula as in Equation 8; (b) Sensitivity is the model's ability to extract score information if the model properly predicts the data using the method shown in equation 9; (c) Specificity describes the accuracy obtained based on the performance of a method with the formula as in equation 10.

$$Accuracy = \frac{\sum_{i=1}^{n} TP_i}{N} \times 100\% \tag{8}$$

$$Sensitivity = \frac{\sum_{i=1}^{n} \frac{TP_i}{TP_i + FN_i}}{n} \times 100\% \tag{9}$$

$$Specificity = \frac{\sum_{i=1}^{n} \frac{TN_i}{TN_i + FP_i}}{n} \times 100\% \tag{10}$$

3. RESULT AND ANALYSIS

3.1. Data Split Testing

The first experiment focuses on the XGBoost model's k-fold comparison that is showed by Table 1. The k-fold method splits data into k sections for cross-validation. Each experiment in k-fold used the same data and criteria, where the parameters were filled with the default values shown in Table 2. This experiment was created in the context of XGBoost to test the performance of the model with varied values of k in the k-fold, with the goal of discovering the optimal k value that produces the best results. Data split testing experiment was separated from the parameter testing to ensure that the sensitivity analysis of the model to the k-fold value selection

could be performed without interference from other variables. The findings of this experiment can give insight into the sensitivity of the XGBoost model to k-fold value selection.

Table 1. Result of Experiment 1

K Fold	Accuracy (%)	Sensitivity (%)	Specificity (%)
5	98.19	98.01	99.62
10	97.73	97.45	99.53

The finding of this study is that the performance of the model is dependent on the number of folds used. At k=5, the XGBoost model achieved 98.19% accuracy, 98.01% sensitivity, and 99.62% specificity. However, when k is increased to 10, the performance degrades somewhat, with accuracy reaching 97.73%, sensitivity 97.45%, and specificity 99.53%. From this, it can be concluded that choosing the right k value plays an important role in affecting the performance of the XGBoost model, where too low or too high k can result in a decrease in performance. When the k-fold value is higher, the variation of model performance measurements may decrease as each cross-validation measurement is performed with a larger portion of the data. However, this may also increase bias as fewer models are evaluated on different data. Conversely, with a lower k-fold value, the variation may be higher as the measurements are performed on a smaller sample, but this can also help reduce bias as the model is evaluated on a wide range of test data. The results of this study are **in line** with previous research [29] which shows that the number of folds in k-fold cross-validation can significantly affect model performance.

3.2. Parameter Testing

The second experiment employed the 5-fold cross-validation approach to tune parameters. The parameter tuning procedure is a vital stage in model development that involves changing the values of model parameters to improve performance. In the context of XGBoost, parameter tuning involves adjusting parameters such as learning rate, gamma, max depth, and n_estimator. Using 5-fold cross validation, this experiment aims to evaluate the effectiveness and consistency of parameter tuning results on XGBoost models. The results of this experiment can provide insight into the optimal way of configuring the model parameters to achieve maximum performance. This aim of this experiment is to identify the parameters that are effective in improving model performance without being affected by data variations that can occur in the training and testing sets. This is the reason why data split testing and parameter testing are separated. In this case, the XGBoost results obtained through parameter tuning are expected to be more **in line** with the purpose of classification, which is to improve the performance model in diagnosing Celiac disease. XGBoost parameters tested by this study are shown in Table 2.

Table 2. Parameter tuning of XGBoost

Parameter	Description	Value	
		Default	Tuning
Learning_rate	Weight correction value in the training process [30]	0.3	0.1, 0.3, 0.5, 0.7
Gamma	Minimum loss reduction [30]	0	0, 1, 3, 5
Max depth	Maximum depth of tree XGBoost [30]	6	3, 6, 9
N_estimator	Number of trees in the process that XGBoost will try to learn [31]	100	50, 100

Based on the parameters presented in Table 2, 96 trials were conducted for each training data and testing data to evaluate the performance of the XGBoost model. From the various parameter combinations tested, the best model goodness is obtained when XGBoost is set with the max_depth parameter of 3, n_estimator of 100, gamma of 0, and learning rate of 0.3. This parameter selection proved to be optimal in improving model performance. The test results showed that the accuracy, sensitivity, and specificity values reached more than 98%. The application of this parameter tuning not only improves the accuracy of the model, but also strengthens the model's ability to classify data more precisely and efficiently. Overall, these results underscore the importance of parameter optimization in machine learning, especially in algorithms such as XGBoost, to produce superior and reliable classification models under various data conditions.

3.3. Experimental Comparison

Based on the experimental results of parameter tuning on the XGBoost model with 5-fold cross-validation, there is a significant performance improvement compared to the use of default parameters. With optimally tuned parameters, the model achieved an

accuracy of 98.64%, a sensitivity of 98.43%, and a specificity of 99.72%. This comparison was with the model using the default parameters, which had an accuracy of 98.19%, sensitivity of 98.01%, and specificity of 99.62%. Parameter tuning successfully improved the performance of the XGBoost classification model where accuracy increased by 0.45%, Sensitivity increased by 0.42% and specificity increased by 0.10%. **The findings** of this study were that parameter tuning successfully improved all three model performance evaluation metrics. The model comparison can be seen in Figure 6. The results of this study are **in line** with or supported by previous research which shows that parameter tuning can provide a significant improvement in the performance of the XGBoost model. For example, research by [32] which showed that parameter tuning in the XGBoost model provided a significant performance improvement in predicting bank distress in the Eurozone. In addition, research by [33] also supports this finding, where parameter tuning increases the effectiveness of the XGBoost model in detecting network intrusions.

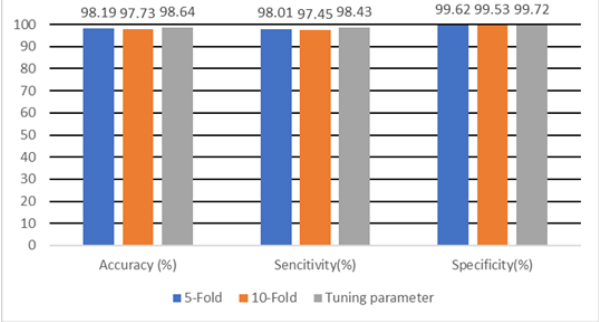


Figure 6. Comparison of XGBoost models

3.4. Confusion Matrix Evaluation

This Confusion Matrix in Figure 7, shows the number of correct predictions for each class, with the main diagonal (from top left to bottom right) reflecting the values correctly identified by the model. For example, the class ”None” was correctly predicted 70 times, the class ”Atypical” 108 times, the class ”Typical” 59 times, the class ”Silent” 44 times, the class ”Latent” 78 times, and the class ”Potential” 76 times. This number indicates how many instances within each class were correctly classified. The overall accuracy can be calculated as the number of correct predictions divided by the total number of predictions (the total number of elements in the Confusion Matrix). Based on the dominance of the values on the main diagonal, it can be concluded that the model has a good performance in classifying the data to the correct class. From this Confusion Matrix, it can be seen that the model performs very well in classifying the classes ”None,” ”Atypical,” ”Latent,” and ”Potential,” with only slight errors in the classes ”Typical” and ”Silent.” This error in prediction is likely due to the similarity in features between these classes. Overall, the model performed strongly, but may require further refinement to reduce errors in the more difficult to distinguish classes. In general, the research results after tuning parameter are shown in Table 3.

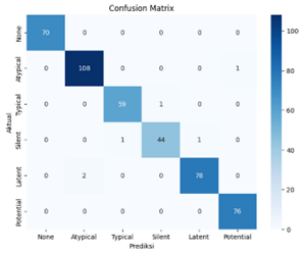


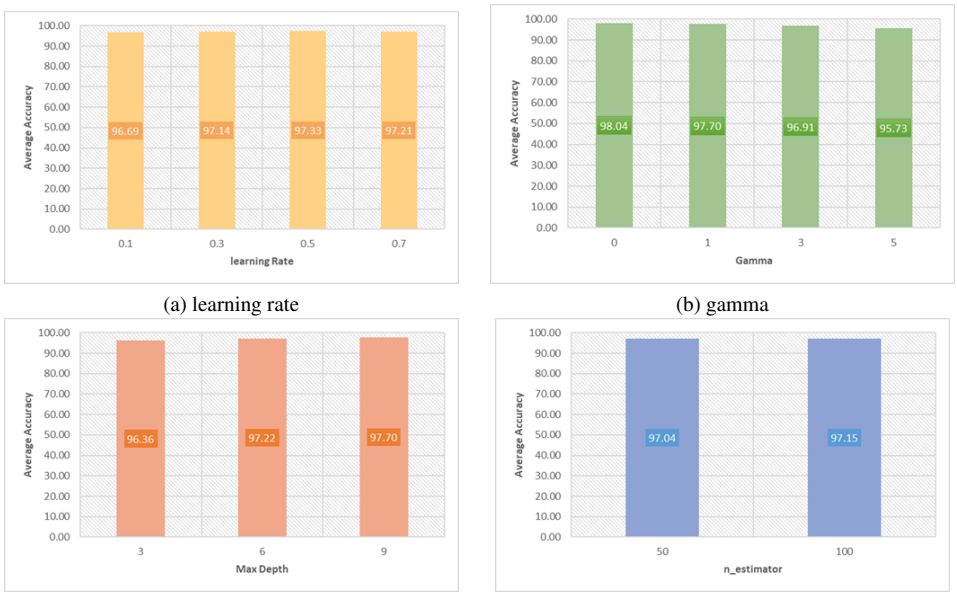
Figure 7. Best Confusion Matrix

Table 3. Pembagian data untuk Training dan Testing

Accuracy (%)	Sensitivity (%)	Specificity (%)
98.64	98.43	99.72

3.5. Parameter Interpretation of XGBoost

The findings of this study are that with this parameter configuration, it is concluded that the default does not produce optimal performance on classification with celiac disease data. However, by reducing the tree depth to 3, optimal results can be achieved. This suggests that in the context of celiac disease, it is better to have a simpler decision tree than a deeper one, as a form of overfitting prevention. The results of testing parameter variations in the XGBoost classification model provide a number of important conclusions (See in Figure 8). First, in the context of learning rate, choosing a high value can improve the accuracy of the model, but must be balanced so as not to cause too slow convergence or overfitting. A low learning rate, on the other hand, can slow down convergence and even stop training before reaching the optimum, that **in line** with [33]. Second, it is discovered that smaller Gamma values can enhance accuracy by making the decision tree deeper and more complicated, which is especially useful in datasets with substantial variance. However, excessive Gamma values must be avoided in order to prevent simplifying the decision tree [30]. Third, it was determined that tree depth influences model accuracy in the Max depth variation. Low depth minimizes overfitting on noisy data, but max_depth captures more complicated patterns. Choosing the best Max depth is critical for striking a compromise between avoiding overfitting and increasing prediction performance, in accordance with [34]. Finally, a higher number of n estimators adds to increased accuracy, although care must be given to avoid potential overfitting, as reported by [32]. As a consequence, the parameter testing results give useful information for improving the XGBoost model in terms of balancing model complexity and data characteristics.



(a) learning rate (b) gamma

(c) max_depth (d) n_estimator

Figure 8. Graph of the Effect of XGboost Parameters on the Average Percentage of Accuracy

It is important to compare the results of this study with previous studies that have also focused on classification. A comparison of the XGBoost model with previous research is shown in Table 4. The Random Forest model, which used the same celiac dataset, achieved the highest accuracy of 97%. While this is an excellent performance, the XGBoost model in this study managed to outperform it with an accuracy of 98.64%. On another celiac dataset, the Linear SVM achieved 94% accuracy, which is lower than XGBoost. Different datasets may have different characteristics, which may affect the model accuracy results.

Table 4. Comparison of the Results of this Research Model with Previous Research

Model	Highest accuracy
Random Forest [35]	0.97
Linear SVM [1]	0.94
XGBoost with tuning parameter	0.98

4. CONCLUSION

This study includes two experiments that use parameter adjustment and k-fold cross-validation to assess the effectiveness of the XGBoost model in data categorization. it can be concluded that the parameter adjustments made, combined with the use of k-fold cross-validation, significantly improved the performance of the XGBoost model. With accuracy, sensitivity, and specificity reaching over 98%, the model showed high effectiveness in classifying the data with consistent accuracy. These results show that XGBoost, with parameters adjusted and validated through k-fold cross-validation, is able to provide optimal performance for data classification in this study. This success also shows the potential of this model to be applied to similar cases in the future, with proper parameter adjustments to achieve the best results. The study's contribution is the validation of the significance of parameter adjusting and k value selection in enhancing the XGBoost model's performance. These results underscore the importance of adjusting parameters appropriately to achieve an optimal model. Optimal parameter configuration not only improves the overall performance of the model but also proves that the right settings can have a significant positive impact on the evaluation results of classification models. However, the range of parameters and the variety of data utilized in this study are limited, thus the conclusions drawn from this research should be carefully reviewed before being broadly generalized beyond the constraints of this current study. To confirm and reinforce the findings, future studies can test this model on a larger range of data and broaden the range of parameters.

5. ACKNOWLEDGEMENTS

The author is grateful to UINSA Mathematics Laboratory, which has helped realize this research.

6. DECLARATIONS

AUTHOR CONTIBUTION

The first author developed the model and wrote the article. The second author supervised the model testing while the third author was responsible for writing the formula.

FUNDING STATEMENT

No grants were received for this study from any governmental, private, or nonprofit organizations.

COMPETING INTEREST

The authors state that none of their personal or financial conflicts might have impacted the research presented in this publication.

REFERENCES

[1] C.-A. Stoleru, E. H. Dulf, and L. Ciobanu, "Automated detection of celiac disease using Machine Learning Algorithms," *Scientific Reports*, vol. 12, no. 1, pp. 4071–4082, Mar. 2022, https://doi.org/10.1038/s41598-022-07199-z.

[2] H. J. Van Der Fels-Klerx, N. G. E. Smits, M. G. E. G. Bremer, J. M. Schultink, M. M. Nijkamp, J. J. M. Castenmiller, and J. H. M. De Vries, "Detection of gluten in duplicate portions to determine gluten intake of coeliac disease patients on a gluten-free diet," *British Journal of Nutrition*, vol. 125, no. 9, pp. 1051–1057, May 2021, https://doi.org/10.1017/S0007114520002974.

[3] L. Ciacchi, H. H. Reid, and J. Rossjohn, "Structural bases of T cell antigen receptor recognition in celiac disease," *Current Opinion in Structural Biology*, vol. 74, pp. 1–12, Jun. 2022, https://doi.org/10.1016/j.sbi.2022.102349.

[4] I. K. W. Jayawardhana and I. N. B. A. Kresnapati, "Anemia Megaloblastik: Sebuah Tinjauan Pustaka," *Biocity Journal of Pharmacy Bioscience and Clinical Community*, vol. 1, no. 1, pp. 25–35, Nov. 2022, https://doi.org/10.30812/biocity.v1i1.2422.

[5] V. Caratelli, M. Moccia, F. R. Paggioro, L. Fiore, C. Avitabile, M. Saviano, A. L. Imbriani, P. Dardano, L. De Stefano, D. Moscone, N. A. Colabufo, I. Ghafir El Idrissi, F. Russo, G. Riezzo, G. Giannelli, and F. Arduini, "Liquid Biopsy beyond Cancer: A miRNA Detection in Serum with Electrochemical Chip for Non-Invasive Coeliac Disease Diagnosis," *Advanced NanoBiomed Research*, vol. 2, no. 9, pp. 1–15, Sep. 2022, https://doi.org/10.1002/anbr.202200015.

[6] H. Oktadiana, M. Abdullah, K. Renaldi, and N. Dyah, "Diagnosis dan Tata Laksana Penyakit Celiac," *Jurnal Penyakit Dalam Indonesia*, vol. 4, no. 3, pp. 157–162, Sep. 2017, https://doi.org/10.7454/jpdi.v4i3.131.

[7] O. A. Orjiekwe, "Nutritional Management of Celiac Disease," *Journal of Clinical & Experimental Immunology*, vol. 8, no. 2, pp. 561–572, Apr. 2023, https://doi.org/10.33140/JCEI.08.02.03.

[8] S. Diantika, H. Nalatissifa, R. Supriyadi, N. Maulidah, and A. Fauzi, "Implementasi Multi-Class Gradient Boosting untuk Mengklasifikasikan Jenis Hewan pada Kebun Binatang," *Antivirus : Jurnal Ilmiah Teknik Informatika*, vol. 17, no. 1, pp. 33–40, Jun. 2023, https://doi.org/10.35457/antivirus.v17i1.2812.

[9] J. M. A. S. Dachi and P. Sitompul, "Analisis Perbandingan Algoritma XGBoost dan Algoritma Random Forest Ensemble Learning pada Klasifikasi Keputusan Kredit," *Jurnal Riset Rumpun Matematika dan Ilmu Pengetahuan Alam*, vol. 2, no. 2, pp. 87–103, Jul. 2023, https://doi.org/10.55606/jurrimipa.v2i2.1470.

[10] B. Jange, "Prediksi Harga Saham Bank BCA Menggunakan XGBoost," *ARBITRASE: Journal of Economics and Accounting*, vol. 3, no. 2, pp. 231–237, Nov. 2022, https://doi.org/10.47065/arbitrase.v3i2.495.

[11] Y. Rombe, S. A. Thamrin, and A. Lawi, "Application of Adaptive Synthetic Nominal and Extreme Gradient Boosting Methods in Determining Factors Affecting Obesity: A Case Study of Indonesian Basic Health Research Survey 2013," *Indonesian Journal of Statistics and Its Applications*, vol. 6, no. 2, pp. 309–317, Aug. 2022, https://doi.org/10.29244/ijsa.v6i2p309-317.

[12] T.-T.-H. Le, Y. E. Oktian, and H. Kim, "XGBoost for Imbalanced Multiclass Classification-Based Industrial Internet of Things Intrusion Detection Systems," *Sustainability*, vol. 14, no. 14, pp. 87–96, Jul. 2022, https://doi.org/10.3390/su14148707.

[13] H. Christanto, J. Rahmad, S. H. Sinurat, D. R. Hamonangan Sitompul, A. Sitomorang, D. J. Ziegel, and E. Indra, "Analisis Perbandingan Decision Tree, Support Vector Machine, dan XGBoost dalam Mengklasifikasi Review Hotel Trip Advisor," *Jurnal Teknologi Informatika dan Komputer*, vol. 9, no. 1, pp. 306–319, Mar. 2023, https://doi.org/10.37012/jtik.v9i1.1429.

[14] Y. Amelia, "Perbandingan Metode Machine Learning untuk Mendeteksi Penyakit Jantung," *IDEALIS : Indonesia Journal Information System*, vol. 6, no. 2, pp. 220–225, Jul. 2023, https://doi.org/10.36080/idealis.v6i2.3043.

[15] M. K. Nasution, R. R. Saedudin, and V. P. Widartha, "Perbandingan Akurasi Algoritma Naïve Bayes Dan Algoritma Xgboost Pada Klasifikasi Penyakit Diabetes," *eProceedings of Engineering*, vol. 8, no. 5, pp. 9765–9772, 2021.

[16] R. Kurniawan, "Klasifikasi Tingkat Kematangan Buah Pepaya Berdasarkan Warna Kulit Menggunakan Sensor Warna TCS3200," *Journal ICTEE*, vol. 4, no. 1, pp. 1–10, Mar. 2023, https://doi.org/10.33365/jictee.v4i1.2692.

[17] Y. Hui, L. Shuli, L. Rongxiu, and Z. Jianyong, "Prediction of component content in rare earth extraction process based on ESNs-Adaboost," *IFAC-PapersOnLine*, vol. 51, no. 21, pp. 42–47, 2018, https://doi.org/10.1016/j.ifacol.2018.09.390.

[18] G. Abdurrahman and M. Sintawati, "Implementation of xgboost for classification of parkinson's disease," *Journal of Physics: Conference Series*, vol. 1538, no. 1, p. 012024, May 2020, https://doi.org/10.1088/1742-6596/1538/1/012024.

[19] M. E. Irawati and H. Zakaria, "Classification Model for Covid-19 Detection Through Recording of Cough Using XGboost Classifier Algorithm," in *2021 International Symposium on Electronics and Smart Devices (ISESD)*. Bandung, Indonesia: IEEE, Jun. 2021, pp. 1–5, https://doi.org/10.1109/ISESD53023.2021.9501695.

[20] C. Hu, L. Li, W. Huang, T. Wu, Q. Xu, J. Liu, and B. Hu, "Interpretable Machine Learning for Early Prediction of Prognosis in Sepsis: A Discovery and Validation Study," *Infectious Diseases and Therapy*, vol. 11, no. 3, pp. 1117–1132, Jun. 2022, https://doi.org/10.1007/s40121-022-00628-6.

[21] R. Hoque, S. Das, M. Hoque, and E. Haque, "Breast Cancer Classification using XGBoost," *World Journal of Advanced Research and Reviews*, vol. 21, no. 2, pp. 1985–1994, Feb. 2024, https://doi.org/10.30574/wjarr.2024.21.2.0625.

[22] A. T. R. Dani, V. Ratnasari, L. Ni'Matuzzahroh, I. C. Aviantholib, R. Novidianto, and N. Y. Adrianingsih, "Analisis Klasifikasi Artist Music Menggunakan Model Regresi Logistik Biner dan Analisis Diskriminan," *Jambura Journal of Probability and Statistics*, vol. 3, no. 1, pp. 1–10, May 2022, https://doi.org/10.34312/jjps.v3i1.13708.

[23] H. Nuraliza, O. N. Pratiwi, and F. Hamami, "Analisis Sentimen IMBd Film Review Dataset Menggunakan Support Vector Machine (SVM) dan Seleksi Feature Importance," *Jurnal Mirai Management*, vol. 7, no. 1, pp. 1–17, Aug. 2022, https://doi.org/10.37531/mirai.v7i1.2222.

[24] K. Luxmi, N. K. Tiwari, and S. Ranjan, "Estimation and comparison of gabion weir oxygen mass transfer by ensemble learnings of bagging, boosting, and stacking algorithms," *ISH Journal of Hydraulic Engineering*, vol. 29, no. sup1, pp. 196–211, Dec. 2023, https://doi.org/10.1080/09715010.2023.2203109.

[25] P. Y. Lim, W. Y. Chin, and Y. H. Kong, "Exploring the Effectiveness of Ensemble Learning for Business-to-Business (B2B) Demand Forecasting: An Empirical Analysis," in *2024 3rd International Conference on Digital Transformation and Applications (ICDXA)*. Kuala Lumpur, Malaysia: IEEE, Jan. 2024, pp. 226–230, https://doi.org/10.1109/ICDXA61007.2024.10470870.

[26] R. Sibindi, R. W. Mwangi, and A. G. Waititu, "A boosting ensemble learning based hybrid light gradient boosting machine and extreme gradient boosting model for predicting house prices," *Engineering Reports*, vol. 5, no. 4, pp. 1–19, Apr. 2023, https://doi.org/10.1002/eng2.12599.

[27] R. Siringoringo, J. Jamaluddin, and R. Perangin-angin, "Text Mining dan Klasifikasi Multi Label Menggunakan XGBoost," *METHOMIKA Jurnal Manajemen Informatika dan Komputerisasi Akuntansi*, vol. 6, no. 6, pp. 234–238, Oct. 2022, https://doi.org/10.46880/jmika.Vol6No2.pp234-238.

[28] S. Syihabuddin Azmil Umri, "Analisis Dan Komparasi Algoritma Klasifikasi dalam Indeks Pencemaran Udara di DKI Jakarta," *JIKO (Jurnal Informatika dan Komputer)*, vol. 4, no. 2, pp. 98–104, Aug. 2021, https://doi.org/10.33387/jiko.v4i2.2871.

[29] I. K. Nti, O. Nyarko-Boateng, and J. Aning, "Performance of Machine Learning Algorithms with Different K Values in K-fold CrossValidation," *International Journal of Information Technology and Computer Science*, vol. 13, no. 6, pp. 61–71, Dec. 2021, https://doi.org/10.5815/ijitcs.2021.06.05.

[30] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A comparative analysis of gradient boosting algorithms," *Artificial Intelligence Review*, vol. 54, no. 3, pp. 1937–1967, Mar. 2021, https://doi.org/10.1007/s10462-020-09896-5.

[31] A. Murtadho and D. H. Sulistyawati, "Machine Learning Untuk Perbandingan Tingkat Akurasi Prediksi Penyakit Diabetes Metode Supervised Learning," Undergraduate, Universitas 17 Agustus 1945, Surabaya, 2020.

[32] F. Climent, A. Momparler, and P. Carmona, "Anticipating bank distress in the Eurozone: An Extreme Gradient Boosting approach," *Journal of Business Research*, vol. 101, pp. 885–896, Aug. 2019, https://doi.org/10.1016/j.jbusres.2018.11.015.

[33] P. Devan and N. Khare, "An efficient XGBoost–DNN-based classification model for network intrusion detection system," *Neural Computing and Applications*, vol. 32, no. 16, pp. 12 499–12 514, Aug. 2020, https://doi.org/10.1007/s00521-020-04708-x.

[34] O. Sagi and L. Rokach, "Approximating XGBoost with an interpretable decision tree," *Information Sciences*, vol. 572, pp. 522–542, Sep. 2021, https://doi.org/10.1016/j.ins.2021.05.055.

[35] A. D. Călin, "Machine Learning Models for Predicting Celiac Disease Based on Non-invasive Clinical Symptoms," in *Artificial Intelligence Applications and Innovations*, I. Maglogiannis, L. Iliadis, J. Macintyre, M. Avlonitis, and A. Papaleonidas, Eds. Cham: Springer Nature Switzerland, 2024, vol. 711, pp. 145–159.