Analysis of Diabetes Classification Performance Improvement Using Ensemble Bagging and K-Fold

Mawardi Kudin
Informatics Study Program
Universitas Sulawesi Barat
Majene, Indonesia
mawardi.kudin@unsulbar.ac.id

Abd Salam At Taqwa
Informatics and Computer Engineering Education Study Program
Universitas Negeri Makassar
Makassar, Indonesia
abd.salam.attaqwa@unm.ac.id

Angga Kurniawan
Data Science Study Program
Telkom University
Banyumas, Indonesia
anggakurniawan@telkomuniversity.ac.id

Chairi Nur Insani
Informatics Study Program
Universitas Sulawesi Barat
Majene, Indonesia
chairini@unsulbar.ac.id

Abstract— Diabetes mellitus represents a long-term metabolic disorder whose global incidence continues to rise, making precise early identification essential to minimize severe complications. Machine learning techniques have been extensively utilized for diabetes classification; however, single-model approaches often suffer from performance constraints, such as susceptibility to overfitting and high variability in prediction outcomes. To address these challenges, this research introduces a bagging-based ensemble learning strategy integrated with K-Fold Cross Validation to enhance both predictive accuracy and model robustness. The study employs the Pima Indians Diabetes Dataset, which contains 768 patient records described by eight clinical features and one outcome variable. Eight classification methods—Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Decision Tree, Random Forest, Naïve Bayes, Gradient Boosting, and XGBoost—were assessed individually and within the proposed ensemble framework. Model effectiveness was measured using accuracy, precision, recall, and F1-score derived from the confusion matrix. The findings indicate that the ensemble bagging approach generally strengthens model stability and yields improvements in accuracy and precision across most algorithms. Notably, K-Nearest Neighbors and XGBoost demonstrated the most stable gains following ensemble integration. Nevertheless, enhancements in precision were frequently associated with a reduction in recall, reflecting a trade-off in identifying positive cases. In summary, the integration of bagging and K-Fold Cross Validation provides a more resilient and dependable classification model, offering strong potential for supporting clinical decision-making in early diabetes detection.

Keywords—Diabetes Mellitus, Machine Learning, Ensemble Bagging, K-Fold Cross Validation, Classification.

Article info: Date Submitted: 2026-02-16 | Date Revised: 2026-04-14 | Date Accepted: 2026-04-25

I. INTRODUCTION

Diabetes mellitus is a serious metabolic disorder characterized by persistently elevated blood glucose levels (hyperglycemia) resulting from abnormalities in insulin secretion, insulin action, or both. The disease is generally classified into two primary categories: type 1 diabetes, in which the pancreas fails to produce insulin, and type 2 diabetes, where the body becomes resistant to insulin or cannot utilize it effectively. Type 2 diabetes is the predominant form, accounting for approximately 90% of all reported cases [1]. When not detected and managed promptly, the condition can lead to symptoms such as excessive thirst (polydipsia), increased appetite (polyphagia), and frequent urination (polyuria), along with other complications [2].

Over the past decades, diabetes has become a major global health concern, with prevalence increasing rapidly among both adults and children. It is now regarded as one

DOI: https://doi.org/10.37396/jsc.v9i1.630

of the most pressing public health challenges worldwide. In 2015, around 8.8% of the global adult population—equivalent to roughly 415 million individuals—were living with diabetes, and this number is projected to rise to approximately 642 juta by 2040 [3]. Additional reports estimated that 424 million people were affected in 2017, with forecasts reaching 628.6 million by 2045 [4]. The number of cases has escalated dramatically, more than tripling between 1990 and 2010, while the annual incidence of new diagnoses has continued to grow significantly [1].

The burden of diabetes is substantial not only medically but also socioeconomically. The disease affects hundreds of thousands of children and contributes to millions of deaths globally. Mortality associated with diabetes increased from 0.6 million to 1.7 million deaths between 1990 and 2017 [4]. Economically, the worldwide cost of diabetes management reached nearly USD 673 billion in 2015 and is expected to climb to about USD 802 billion by 2040 [3].

These epidemiological trends demonstrate that diabetes is no longer solely an individual health issue but has evolved into a global crisis requiring innovative and effective solutions. Early identification and accurate prediction are crucial to preventing complications, improving disease management, and reducing long-term health and financial burdens.

At the same time, advancements in healthcare technology—particularly in diagnostic methodologies—have progressed rapidly. Emerging approaches increasingly leverage artificial intelligence, especially machine learning, to support early detection of chronic diseases such as diabetes. A variety of classification algorithms have been explored to develop predictive systems capable of analyzing clinical data efficiently and accurately. Techniques including Decision Tree, Support Vector Machine (SVM), Logistic Regression, K-Nearest Neighbors (KNN), Random Forest, Naïve Bayes, Gradient Boosting, and XGBoost have been widely adopted in previous studies to classify and predict diabetes risk [5], [6], [7], [8].

Nevertheless, individual machine learning models often exhibit limitations, such as overfitting, instability, and sensitivity to dataset characteristics. To address these issues, ensemble learning has been introduced as an alternative paradigm. This approach combines multiple base classifiers—either homogeneous or heterogeneous—to produce collective predictions that are typically more accurate and robust than those generated by a single model, thereby improving overall classification performance [9].

One of the most widely used ensemble techniques is Bagging (Bootstrap Aggregating). This method constructs multiple bootstrap samples through random sampling with replacement from the original dataset, trains identical base learners on each sample, and aggregates their predictions—commonly using a voting mechanism in classification tasks. Bagging is particularly effective in reducing variance and mitigating overfitting, resulting in models that are more stable and generalizable [10], [11]. Prior research has demonstrated the capability of Bagging to enhance the performance of various classification algorithms [12],[13].

Despite the widespread application of machine learning techniques for diabetes classification, several limitations remain. Previous studies indicate that individual classification models, such as Logistic Regression, Support Vector Machine, Decision Tree, and K-Nearest Neighbors, are capable of achieving acceptable predictive performance. However, these models often suffer from instability, overfitting, and sensitivity to data distribution, especially when applied to relatively small or imbalanced datasets [14], [15], [16], [17].

To overcome these limitations, ensemble learning approaches—especially bagging—have been introduced and have demonstrated the ability to improve classification accuracy and reduce model variance [12], [13]. Nevertheless, most prior studies primarily focus on improving predictive performance without sufficiently addressing the reliability

and robustness of the evaluation process. In many cases, model evaluation is still conducted using simple train-test splits, which may produce biased or inconsistent results depending on how the data is partitioned [18], [19].

Therefore, there remains a research gap in integrating performance enhancement techniques with robust evaluation strategies within a unified framework. To address this gap, this study proposes the combination of ensemble bagging and K-Fold Cross Validation to simultaneously improve model stability and ensure a more reliable performance evaluation. By integrating these two approaches, this research aims to develop a more accurate, stable, and generalizable diabetes classification model, while also providing a more comprehensive and unbiased evaluation of machine learning performance.

In this study, an Ensemble Bagging approach is implemented to enhance both the accuracy and robustness of diabetes classification. Eight widely used algorithms—Decision Tree, Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbors (KNN), Random Forest (RF), Naïve Bayes, Gradient Boosting, and XGBoost—are evaluated individually and within the ensemble framework to enable an empirical comparison of their performance.

Furthermore, to obtain more reliable and generalizable evaluation results, K-Fold Cross Validation is applied to both single models and ensemble models. This technique partitions the dataset into K equal subsets, where the model is iteratively trained on K−1 subsets and validated on the remaining subset. Such an approach reduces bias caused by random data splitting, minimizes evaluation uncertainty, and decreases the risk of overfitting by exposing the model to multiple data variations. Consequently, K-Fold Cross Validation provides a more stable and reliable estimate of predictive performance, enabling more accurate comparisons among algorithms and their ensemble configurations [18], [19].

By integrating the variance-reduction capability of Bagging with the comprehensive evaluation mechanism of K-Fold Cross Validation, this study aims to develop a diabetes prediction model that is both accurate and robust. The findings are expected to contribute practically to the development of machine-learning-based decision support systems for early diabetes detection, as well as to enrich academic understanding of classification optimization through ensemble methodologies.

II. METHOD

This chapter outlines the methodological framework employed to enhance diabetes classification performance through the implementation of a bagging-based ensemble learning approach integrated with K-Fold cross-validation. The research procedure was structured in a systematic sequence, beginning with dataset partitioning, followed by the development of baseline machine learning models, the incorporation of the bagging ensemble technique, and finally the evaluation and comparison of model performance. The complete workflow of the study is illustrated in Figure 1, depicting the stages from the preparation of training and testing data, model training using the K-Fold scheme, to the comparative analysis between standalone models and ensemble bagging models.

In this research, several machine learning algorithms were utilized as base learners, including Logistic Regression, KNN, SVM, Decision Tree, RF, Naïve Bayes, XGBoost, and Gradient Boosting. Each algorithm was initially trained using the K-Fold strategy and subsequently extended through the bagging technique to construct ensemble models. The predictive capability of these ensembles was then systematically evaluated and compared with the corresponding single models to determine how effectively the proposed method enhances diabetes classification performance.

Within this methodology chapter, the Algorithm section presents a detailed description of each machine learning technique employed in the study. The Dataset section outlines

DOI: https://doi.org/10.37396/jsc.v9i1.630

the origin and characteristics of the diabetes dataset utilized, followed by the Data Preprocessing section, which explains the procedures applied to clean, transform, and prepare the data to ensure its quality and suitability for analysis. The Proposed Ensemble Learning and Bagging Method section elaborates on the theoretical foundation and practical implementation of ensemble learning, particularly the bagging strategy used to construct the ensemble model. The K-Fold cross-validation strategy adopted for training and validation is discussed in the Model Evaluation section. Lastly, the Evaluation Metrics section defines the performance indicators used to measure and compare the effectiveness of all developed models.

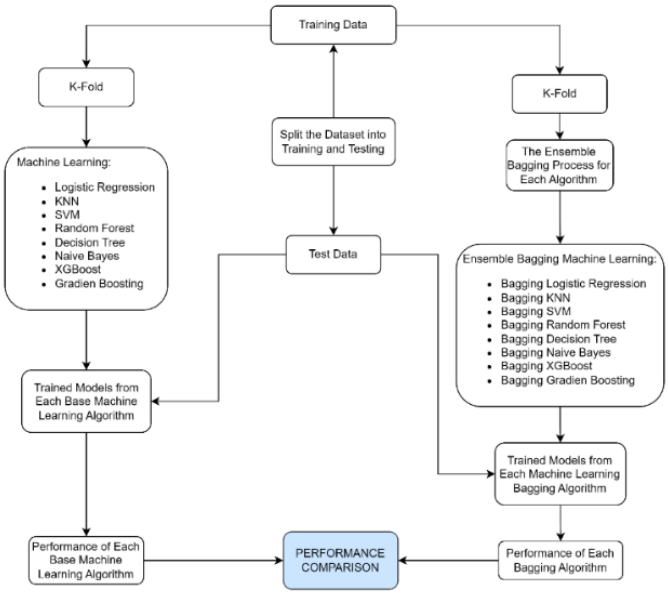

Figure 1. Overview of the research

Through this structured methodological design, the study aims to establish an experimental framework that is objective, consistent, and reproducible, while also delivering a thorough analysis of how the integration of ensemble bagging and K-Fold cross-validation contributes to enhancing diabetes classification performance.

A. Algorithm

Several commonly adopted classification algorithms were implemented as individual models in this research and subsequently combined through a bagging-based ensemble approach, followed by a comparative evaluation to determine the extent of performance enhancement.

1) Logistic Regression

Logistic Regression (LR) is widely applied to model the probability of a categorical outcome by producing values in the range of 0 to 1. These probabilities are then transformed into binary class labels, such as 0 and 1 or yes and no. The method relies on the logistic (sigmoid) function (Equation (1)) to capture the relationship between predictor variables and the likelihood of class membership. The mathematical formulation of the sigmoid function is presented as follows:

$$Sigmoid\ function\ f(x) = \frac{1}{1 + e^{-x}} \qquad (1)$$

The resulting probability value is then compared to a specific threshold value [9], [20].

2) K-Nearest Neighbors (KNN)

The KNN algorithm is commonly utilized for classification tasks, where the main parameter, *k*, determines how many neighboring instances are considered in the decision process. This method identifies the *k* training samples that are closest to a query instance based on a chosen distance measure. The class label is then assigned using a majority

DOI: https://doi.org/10.37396/jsc.v9i1.630

voting mechanism among these neighbors, so the class with the strongest representation becomes the final prediction [21].

3) Super Vector Machine (SVM)

SVM is well known for its strong performance in classification tasks. The technique maps input data into a higher-dimensional feature space and determines an optimal hyperplane to distinguish between classes. This hyperplane is obtained by maximizing the margin, which represents the distance between the decision boundary and the nearest data points from each class. These influential points, called support vectors, define the position of the boundary. By focusing on margin maximization, SVM forms a decision model that can separate classes effectively while maintaining strong generalization capability [9], [22].

4) Decision Tree

A Decision Tree is a supervised machine learning method that can be used for both classification and regression, where decisions are represented in a hierarchical tree structure. The algorithm functions by repeatedly partitioning the dataset according to the most significant attributes, resulting in branches that represent a sequence of logical rules from the root node to the leaf nodes. Besides offering efficient predictive performance, decision trees are inherently interpretable and transparent, thereby supporting responsible artificial intelligence practices, especially regarding explainability, fairness, and accountability [23].

5) Random Forest

Random Forest is an ensemble-based approach within supervised learning that can be applied to both classification and regression problems. It constructs a large number of decision trees using different bootstrap samples of the dataset, while also selecting random subsets of features during the training phase. The final prediction is obtained through majority voting for classification or by averaging the outputs for regression. This strategy effectively mitigates overfitting, enhances predictive accuracy, and demonstrates strong capability in handling high-dimensional datasets, class imbalance, and missing values [24].

6) Naïve Bayes

Naïve Bayes is a classification technique rooted in statistical theory that utilizes Bayes' theorem to determine class membership. The method assigns class labels by estimating conditional probabilities while assuming that the features are mutually independent. The classification mechanism is expressed mathematically in Equation (2):

$$P(A|B) = \frac{P(B|A) * P(A)}{P(B)} \qquad (2)$$

where P(A|B) denotes the posterior probability, P(B|A) represents the likelihood, P(A) indicates the prior probability, and P(B) corresponds to the evidence. Naïve Bayes can function in both descriptive and predictive contexts, is straightforward to implement, requires relatively small amounts of training data, and has demonstrated strong performance in various complex applications such as text classification and sentiment analysis [5], [9].

7) XGBoost

XGBoost is a supervised learning method widely used in predictive analytics, including applications such as time-series forecasting, and represents an advanced development of the Gradient Boosting Tree framework. The algorithm operates by iteratively combining multiple weak learners to construct a more powerful and accurate boosted model. Its objective function is designed not only to minimize prediction error (loss) but also to include a regularization term that controls model complexity. Furthermore, XGBoost employs a second-order Taylor expansion of the loss function together with L1 and L2 regularization to improve both predictive accuracy and generalization capability [25].

DOI: https://doi.org/10.37396/jsc.v9i1.630

8) Gradient Boosting (GB)

GB is an ensemble method that builds models sequentially by leveraging the dependence among base learners. At each stage, a new model is trained to correct the residual errors generated by the preceding model, allowing the learning process to progressively refine predictions. This strategy is referred to as boosting because it incrementally integrates multiple weak learners to achieve a stronger predictive model. To obtain optimal performance, the Gradient Boosting algorithm relies on three essential elements: a defined loss function, weak learners as base models, and an additive modeling framework that combines these learners to minimize the overall loss value [26].

B. Dataset

This study utilizes the Pima Indians Diabetes Dataset, obtained from the National Institute of Diabetes and Digestive and Kidney Diseases, as the primary data source. The dataset was originally created for diagnostic purposes to determine the likelihood of diabetes based on various clinical measurements. It includes data from 768 female subjects aged 21 years and above, all from the Pima Indian community in Phoenix, Arizona, United States. Of these cases, 268 are identified as diabetic, while the remaining 500 are classified as non-diabetic. Accordingly, the dataset consists of one dependent (target) variable and eight independent variables used as predictive features [1], [2], [5]. Due to its structured format and well-defined sampling criteria, the dataset is considered suitable and representative for developing and evaluating diabetes prediction models in this research. A detailed description of the dataset is provided in Table 1.

Tabel 1. Summary of Dataset Variables

Variables	Descriptions	Type Data	Min Value	Max Value
Pregnancies	Number of times pregnant	int64	0	17
Glucose	Plasma glucose concentration (mg/dL)	int64	0	199
Blood Pressure	Diastolic blood pressure (mm Hg)	int64	0	122
Skin Thickness	Triceps skinfold thickness (mm)	int64	0	99
Insulin	2-hour serum insulin (μU/ml)	int64	0	846
BMI	Body mass index (kg/m²)	float64	0	67.1
Diabetes Pedigree Function	Diabetes pedigree function	float64	0.08	2.42
Age	Age of the patient (years)	int64	21	81
Outcome	Class variable: 1 = Diabetes, 0 = Non-diabetes	int64	0	1

Table 1 presents a summary of the variables included in the Pima Indians Diabetes Dataset along with their respective data characteristics. Each entry describes the variable name, a concise explanation, the data type, and the observed minimum and maximum values. The dataset contains eight independent variables (predictors)—Pregnancies, Glucose, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetes Pedigree Function, and Age—which reflect patients' clinical measurements and demographic profiles. In addition, a single dependent variable (target), Outcome, is provided to denote diabetes status, where a value of 1 indicates a diabetic case and 0 represents a non-diabetic case. The reported minimum and maximum values for each variable offer an overview of the data distribution and the measurement ranges applied in this research.

DOI: https://doi.org/10.37396/jsc.v9i1.630

C. Data Preprocessing

The data preprocessing phase is conducted to ensure that the dataset is suitable for training and evaluating both machine learning and deep learning models. Since the dataset comprises several numerical attributes that may include invalid entries and exist on different measurement scales, data cleaning and normalization are required prior to model development.

1) Data Cleaning

The dataset contains a number of medical variables that, from a clinical perspective, cannot logically take a value of zero. These variables include BloodPressure, BMI, Glucose, SkinThickness, and Insulin. Zero entries in these fields are therefore interpreted as missing values. To handle this issue, an imputation process was applied by substituting the missing entries with the median value of each respective attribute. The median was selected because it is robust to outliers and preserves the underlying distribution of the data, thereby maintaining representativeness and stability.

2) Feature and Label Separation

After data cleaning, the dataset is split into feature and target variables. All independent attributes represent the patient's physiological condition, while the Outcome variable indicates diabetes status. This separation aims to ensure that the target variable does not affect the model learning process during the training stage and to maintain objectivity in the evaluation process.

3) Training and Test Data Distribution

Subsequently, the dataset is partitioned into training and testing subsets using the hold-out validation approach. In this study, 80% of the data is allocated for training and 20% for testing. The split is performed randomly with a fixed random-state configuration to guarantee consistency and reproducibility of the experimental results. The training set is utilized to build the classification models, whereas the test set is employed to assess the model's ability to generalize to previously unseen data. Table 2 presents the outcome of this dataset partitioning.

Tabel 2. Description of Dataset Distribution

Data Type	Percentage	Amount of Data	Description of Use
Training Set	80%	614 data	Used for model training processes
Testing Set	20%	154 data	Used for model performance evaluation
Total	100%	768 data	All data in the dataset

The dataset comprises a total of 768 observations, which are partitioned into two subsets: 80% (614 instances) allocated for training and 20% (154 instances) reserved for testing. This split is intended to allow the model to learn effectively from sufficient data while being evaluated on samples that were not involved in the training phase.

4) Feature Normalization

Because the dataset includes numerical variables with different scales and value ranges, normalization is applied to prevent certain features from disproportionately influencing the learning process. The Standard Scaler method is used to standardize each feature so that it has a mean of zero and a standard deviation of one. Importantly, the normalization parameters are derived exclusively from the training data and subsequently applied to the test data. This procedure is implemented to prevent data leakage, which could otherwise introduce bias into the evaluation results.

After completing all preprocessing steps, the refined dataset is used as input for the various classification models examined in this study, including conventional machine learning techniques, ensemble learning approaches, and deep learning models. All models

are trained and evaluated using the same preprocessed data to maintain consistency in data handling and ensure a fair comparison of performance. Consequently, any observed differences in results can be objectively attributed to the inherent characteristics of each algorithm.

D. Proposed Ensemble Learning and Bootstrap Aggregating (Bagging) Methods

Ensemble learning represents an effective strategy for enhancing classification performance by integrating multiple base machine learning models to generate predictions that are generally more stable and accurate than those obtained from a single model. This approach exploits the diversity among models, allowing errors produced by one model to be compensated for by others [9]. Among the various ensemble methods, bagging (Bootstrap Aggregating) is one of the most widely adopted techniques, primarily aimed at reducing variance and improving the model's ability to generalize to unseen data [10], [11].

Referring to the system design illustrated in Figure 2, the research workflow begins with partitioning the diabetes dataset into training and testing subsets. This separation ensures that model evaluation is conducted objectively using data that is not involved in the training phase. To increase result reliability and minimize bias caused by random data splitting, K-Fold Cross Validation is applied to the training data. In this study, the training set is divided into 10 folds. During each iteration, one fold serves as validation data while the remaining folds are used for training, ensuring that every data instance has an equal opportunity to function as either training or validation data.

Furthermore, in each K-Fold Cross Validation fold, a random sampling process is applied using the Bagging technique to form multiple training data subsets. This process produces a number of randomly sampled datasets called bags (Bag 1; Bag 2, up to Bag n). Each bag has a slightly different data distribution because sampling is done with replacement, allowing for data repetition and differences in composition between bags. This stage is the core of the bagging method, which aims to create diversity between the models to be trained.

DOI: https://doi.org/10.37396/jsc.v9i1.630

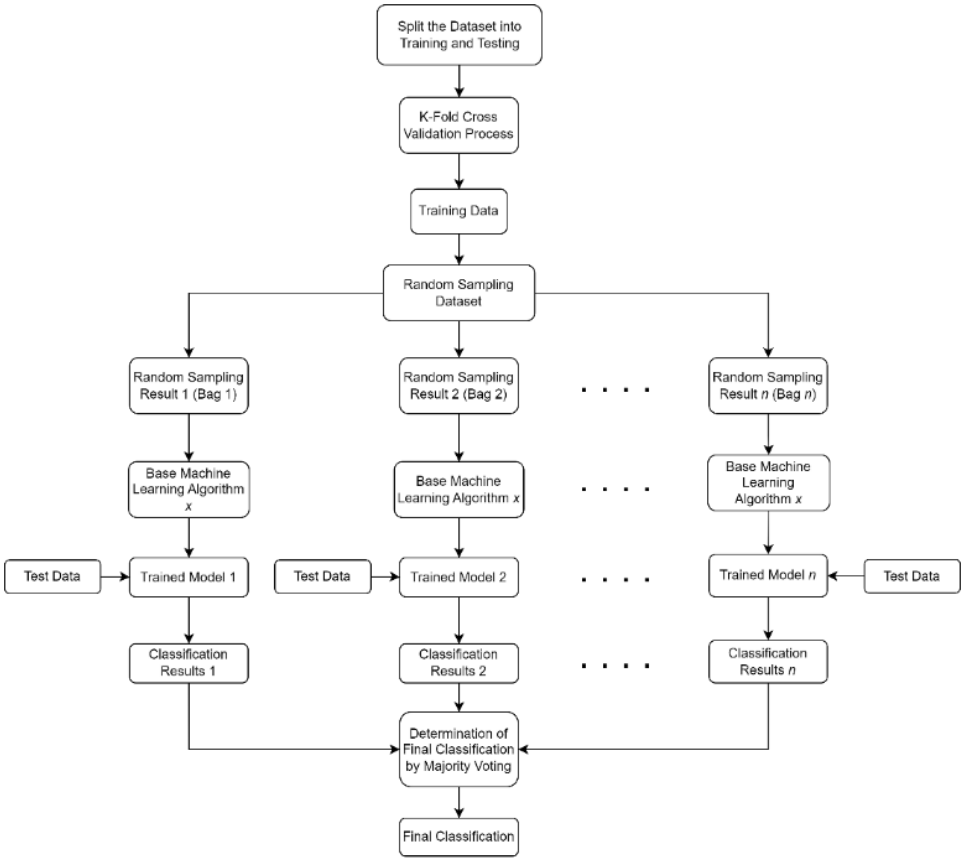

Figure 2. Classification Stage Design from Ensemble Bagging

Each random sampling dataset is then used to train the same base machine learning algorithm (base machine learning algorithm x). Thus, even though the algorithm used is identical, differences in the training data in each bag will result in several trained models with different decision characteristics. This process produces a set of trained models, namely Trained Model 1, Trained Model 2, and so on up to Trained Model *n*.

Once all ensemble members have been constructed, the same test dataset is supplied to each trained model to obtain individual classification outputs. Every model produces predictions according to the patterns it has learned during its respective training phase. These outputs are then gathered as Classification Result 1, Classification Result 2, continuing through Classification Result *n*.

The final stage involves determining the overall prediction using a majority voting mechanism. In this approach, each model contributes one vote for a predicted class, and the class receiving the highest number of votes is selected as the final classification outcome. Majority voting is effective in enhancing predictive accuracy because it reduces the impact of erroneous predictions from individual models, particularly when dealing with complex or non-linear data such as those encountered in diabetes diagnosis.

By combining K-Fold Cross Validation and ensemble bagging, the system proposed in this study is expected to produce a diabetes classification model that is more robust, stable, and better at generalization than a single model. The integration of these two approaches not only improves classification performance but also provides a fairer, more representative evaluation of the actual data conditions.

E. Model Evaluation

As shown in Figure 2, the performance of the proposed models was evaluated using the K-Fold Cross Validation approach. This method is employed to obtain a more objective and dependable measurement of model performance by repeatedly training and

DOI: https://doi.org/10.37396/jsc.v9i1.630

testing the model on different subsets of the data. In this procedure, the dataset is partitioned into *K* segments (folds) of roughly equal size. The training and testing stages are carried out iteratively, where one fold is used as the test set in each cycle, while the remaining *K−1* folds serve as the training data [18], [19].

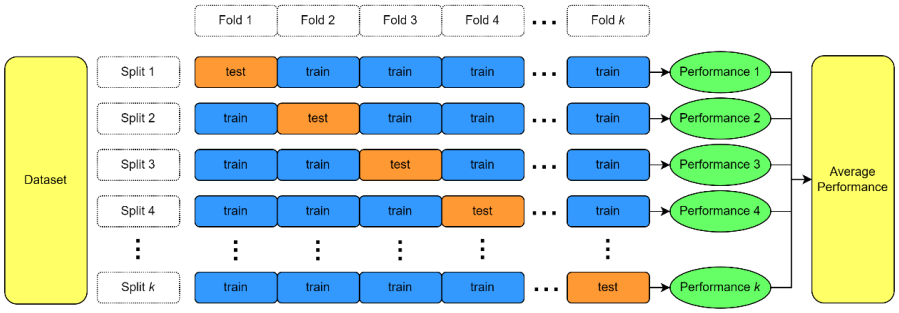

Figure 3. Overview of K-Fold Cross Validation

Referring to the illustration in Figure 3, each fold is alternately assigned as the testing set, ensuring that every data instance in the dataset has an equal opportunity to be evaluated during the testing phase. In every iteration, the model is trained using the designated training data and then assessed on a different testing subset, producing performance outcomes denoted as Performance 1 through Performance *k*. These results are subsequently aggregated and averaged to obtain the model's overall performance score, which serves as the primary indicator of its ability to generalize to previously unseen data.

The application of K-Fold Cross Validation helps reduce the potential bias that may arise from a single split between training and testing data. Moreover, this technique enables more efficient utilization of the available dataset, particularly when the amount of data is limited. For these reasons, K-Fold Cross Validation was selected as the evaluation strategy in this study, as it provides a more dependable and representative assessment of model performance compared to conventional validation approaches.

F. Evaluation Metrics

The performance of each algorithm, whether implemented as an individual model or incorporated into the proposed ensemble bagging framework, is assessed using a Confusion Matrix. From this matrix, the evaluation metrics—Accuracy, Precision, Recall, and F1-Score—are subsequently computed [9].

III. Result and Discussion

This chapter presents the results and discussion of the performance evaluation of diabetes classification models trained and tested using various machine learning algorithms. The evaluation was conducted on algorithms applied individually and in combination using the proposed ensemble bagging approach, utilizing the Pima Indians Diabetes Dataset as the research data source. The dataset underwent a series of preprocessing steps before being used in the model training and testing. Next, each machine learning algorithm was trained and tested, both individually and via ensemble bagging, to assess the effectiveness of the proposed approach in improving classification performance. This analysis aimed to compare the performance of each algorithm when run independently with its performance after integration into the ensemble bagging scheme, thereby determining the extent to which the method can provide significant performance improvements. During the evaluation process, all model training and testing scenarios, using both the individual and ensemble bagging approaches, employed the K-Fold Cross Validation technique. This technique was used to ensure more stable, objective evaluation results and to minimize bias arising from an imbalance in the training and test data. As the final stage of the evaluation, a Confusion Matrix was used to

DOI: https://doi.org/10.37396/jsc.v9i1.630

assess each model's performance using the main evaluation metrics: Accuracy, Precision, Recall, and F1-Score. The performance values obtained from individual algorithm runs were then compared with those of the same algorithms when implemented with ensemble bagging. This allows us to conclude that the proposed approach is effective in improving diabetes classification performance.

Detailed performance evaluation results for each algorithm, both implemented individually and via ensemble bagging, are presented in Table 3.

Tabel 3. Details of the evaluation metrics results of the individual and ensemble bagging

Model	Type	Accuracy	Precision	Recall	F1-score
Decision Tree	*Base Model*	0,76	0,66	0,61	0,64
SVM	*Base Model*	0,76	0,71	0,54	0,61
Logistic Regression	*Base Model*	0,77	0,72	0,56	0,63
K-Nearest Neighbors	*Base Model*	0,74	0,63	0,59	0,61
Random Forest	*Base Model*	0,76	0,67	0,59	0,63
Naive Bayes	*Base Model*	0,75	0,65	0,59	0,62
Gradient Boosting	*Base Model*	0,76	0,67	0,61	0,64
XGBoost	*Base Model*	0,74	0,64	0,60	0,62
Bagging Decision Tree	*Ensemble Bagging*	0,76	0,69	0,56	0,62
Bagging SVM	*Ensemble Bagging*	0,77	0,74	0,52	0,61
Bagging Logistic Regression	*Ensemble Bagging*	0,76	0,71	0,53	0,60
Bagging K-Nearest Neighbors	*Ensemble Bagging*	0,76	0,68	0,57	0,62
Bagging Random Forest	*Ensemble Bagging*	0,77	0,70	0,57	0,63
Bagging Naive Bayes	*Ensemble Bagging*	0,75	0,66	0,57	0,61
Bagging Gradient Boosting	*Ensemble Bagging*	0,76	0,68	0,57	0,62
Bagging XGBoost	*Ensemble Bagging*	0,76	0,67	0,59	0,63

Table 3 summarizes the performance evaluation results of several diabetes classification algorithms implemented as individual (base) models and within the ensemble bagging framework, assessed using accuracy, precision, recall, and F1-score metrics. As shown in Table 3, the accuracy values of the individual models ranged between 0.74 and 0.77, with Logistic Regression attaining the highest accuracy of 0.77, while K-Nearest Neighbors and XGBoost recorded the lowest accuracy at 0.74. In terms of precision, Logistic Regression achieved the highest individual model value at 0.72, while Decision Tree and Gradient Boosting achieved the highest recall values at 0.61 each. The F1-score values for individual models were relatively stable, with the highest score being 0.64.

Ensemble bagging, as shown in Table 3, yielded more stable performance across most algorithms. The accuracy values for the bagging method ranged from 0.75 to 0.77, with

Bagging SVM and Bagging Random Forest achieving the highest accuracy at 0.77. These results show that the bagging approach maintains and even slightly improves classification performance compared to using individual algorithms.

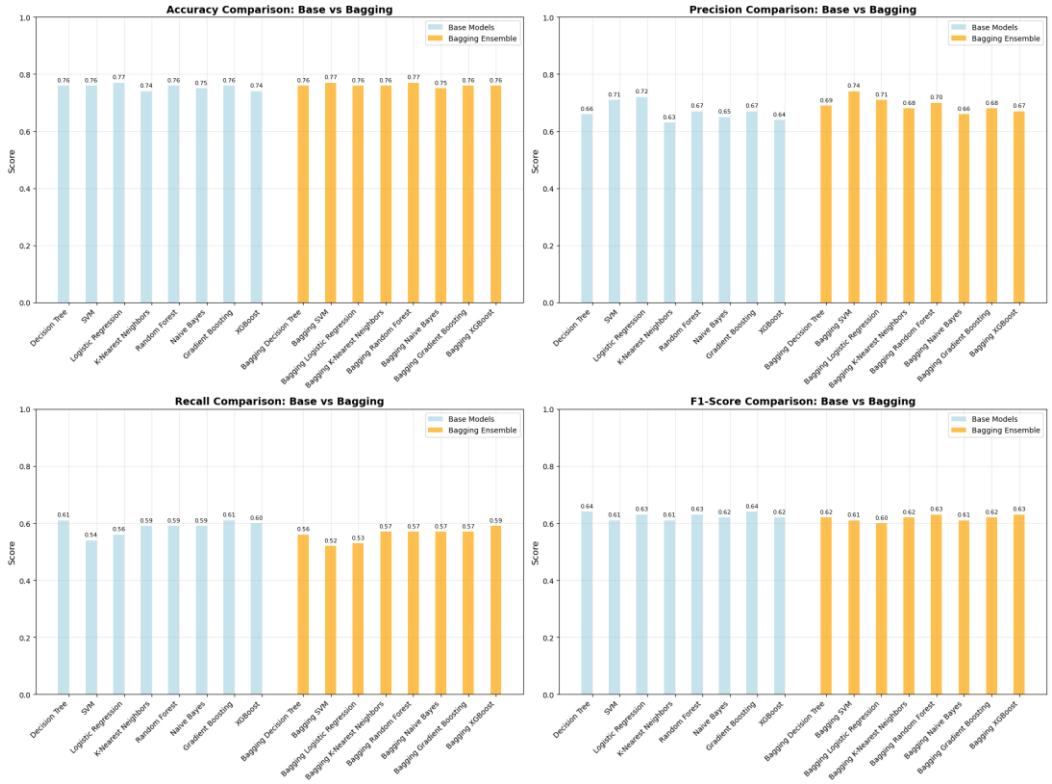

Figure 4. Performance comparison chart between the individual learning and the proposed ensemble bagging method

A visual comparison between the individual models and those developed using ensemble bagging is illustrated in Figure 4, which displays four performance graphs corresponding to accuracy, precision, recall, and F1-score. Figure 4 shows that, for most algorithms, the accuracy and precision of ensemble bagging models are equal to or higher than those of the individual models. This indicates that the bagging technique effectively reduces model variance and improves prediction consistency.

In the precision metrics shown in Figure 4, almost all algorithms improved precision after bagging, particularly for K-Nearest Neighbors, XGBoost, and Decision Tree. This increase in precision indicates that the bagged models are more selective in classifying patients with diabetes, thereby reducing false positives.

However, a different trend is observed in the recall metric. According to Figure 4, most algorithms experienced a decrease in recall after bagging was applied. This decrease indicates that the models tend to be more conservative in providing positive predictions, potentially increasing the number of false negatives. In the context of diabetes classification, this is important to consider because recall is directly related to the model's ability to detect patients who truly have diabetes.

The analysis of the balance between precision and recall is shown through the F1-score value in Figure 4. It can be seen that only a few algorithms were able to maintain or increase the F1-score after applying bagging, while other algorithms experienced a decrease, which indicates that the increase in precision has not been fully realized.

DOI: https://doi.org/10.37396/jsc.v9i1.630

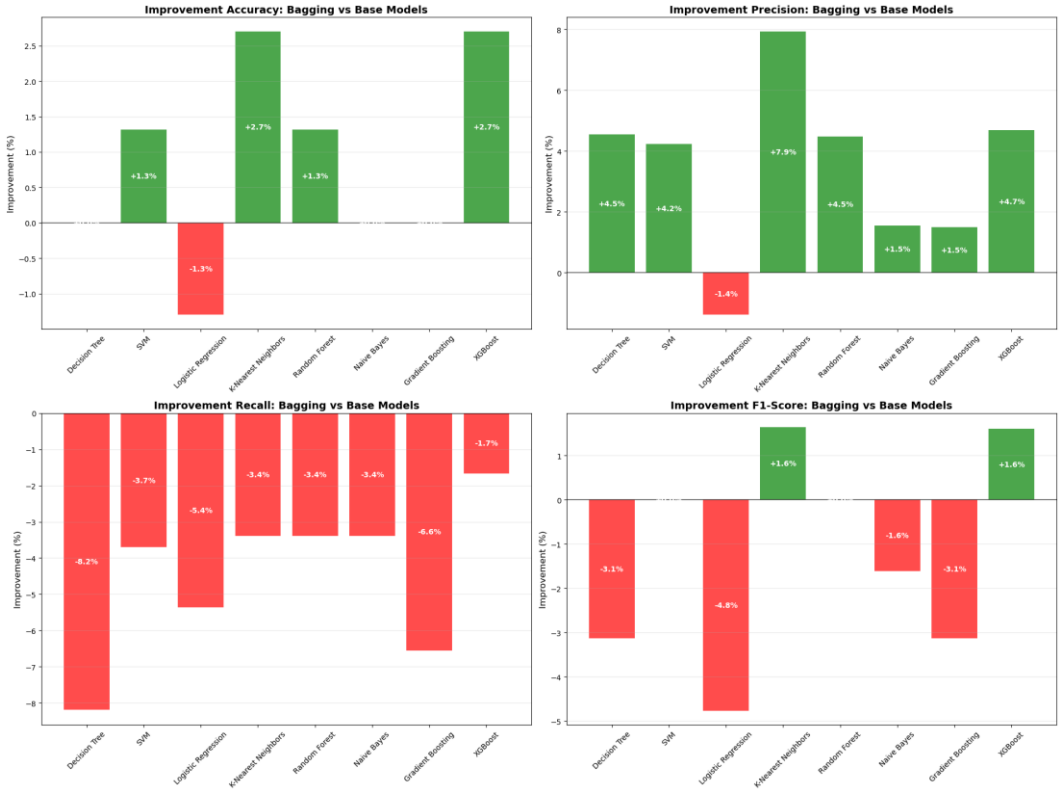

Figure 5. Percentage increase and decrease in performance of each algorithm after using ensemble bagging

To provide a more quantitative overview, Figure 5 displays the percentage improvement in ensemble bagging performance compared to individual models. The visualization likewise presents four evaluation metrics: accuracy, precision, recall, and F1-score. Figure 5 shows that several algorithms improved in accuracy, with K-Nearest Neighbors and XGBoost achieving the largest gains of +2.7% each.

In the precision metric shown in Figure 5, almost all algorithms showed improvements, with the most significant increase occurring with K-Nearest Neighbors (+7.9%), followed by XGBoost (+4.7%) and Random Forest (+4.5%). This confirms that ensemble bagging is highly effective in increasing the accuracy of positive predictions.

Conversely, Figure 5 also shows that all algorithms experienced a decrease in recall after applying bagging, with the largest decreases occurring with Decision Tree (-8.2%) and Gradient Boosting (-6.6%). This finding reinforces the results in Figure 4, which show that bagging methods tend to increase precision at the expense of recall.

In the F1-score metric shown in Figure 5, only K-Nearest Neighbors and XGBoost increased by 1.6%, while the other algorithms decreased. This indicates that these two algorithms benefited most significantly from the application of ensemble bagging overall.

Based on the results presented in Table 3, Figure 4, and Figure 5, applying ensemble bagging can increase model stability and improve performance on accuracy and precision metrics. However, this increase is generally accompanied by a decrease in recall. The algorithms that benefited most from ensemble bagging were K-Nearest Neighbors and XGBoost, which showed consistent improvements in accuracy, precision, and F1-score. These findings indicate that ensemble bagging is effective at reducing variance and noise sensitivity, especially in algorithms heavily influenced by data distribution.

IV. Conclusion

The results of this study demonstrate that combining ensemble bagging with K-Fold Cross Validation plays a substantial role in enhancing both the predictive performance and the stability of diabetes classification models. This approach effectively reduces

DOI: https://doi.org/10.37396/jsc.v9i1.630

model variance and enhances resilience against variations in the data. When applied across eight distinct machine learning algorithms, the ensemble strategy generally yielded improvements in accuracy, precision, and F1-score. Among these, K-Nearest Neighbors and XGBoost exhibited the most consistent enhancements compared to their individual implementations. Nevertheless, the application of this method was also associated with a reduction in recall for the majority of models, revealing a critical trade-off: while precision tends to increase, the sensitivity to correctly identify positive diabetes cases may decline. This phenomenon is particularly significant in clinical contexts, where false negatives can have serious implications. Overall, the results affirm the value of combining bagging and cross-validation in developing more reliable classification systems, while also underscoring the necessity for future investigations—such as hyperparameter optimization, exploration of other ensemble strategies, or adoption of cost-sensitive learning techniques—to achieve a more favorable equilibrium between precision and recall in medical diagnostic applications.

REFERENCES

[1] N. Abdulhadi and A. Al-Mousa, "Diabetes Detection Using Machine Learning Classification Methods," in *2021 International Conference on Information Technology, ICIT 2021 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., Jul. 2021, pp. 350–354. doi: 10.1109/ICIT52682.2021.9491788.

[2] H. El Massari, Z. Sabouri, S. Mhammedi, and N. Gherabi, "Diabetes Prediction Using Machine Learning Algorithms and Ontology," *Journal of ICT Standardization*, vol. 10, no. 2, pp. 319–338, 2022, doi: 10.13052/jicts2245-800X.10212.

[3] F. A. Khan, K. Zeb, M. Al-Rakhami, A. Derhab, and S. A. C. Bukhari, "Detection and Prediction of Diabetes Using Data Mining: A Comprehensive Review," 2021, *Institute of Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2021.3059343.

[4] M. S. Reza, R. Amin, R. Yasmin, W. Kulsum, and S. Ruhi, "Improving diabetes disease patients classification using stacking ensemble method with PIMA and local healthcare data," *Heliyon*, vol. 10, no. 2, Jan. 2024, doi: 10.1016/j.heliyon.2024.e24536.

[5] V. Chang, J. Bailey, Q. A. Xu, and Z. Sun, "Pima Indians diabetes mellitus classification based on machine learning (ML) algorithms," *Neural Comput. Appl.*, vol. 35, no. 22, pp. 16157–16173, Aug. 2023, doi: 10.1007/s00521-022-07049-z.

[6] J. J. Khanam and S. Y. Foo, "A comparison of machine learning algorithms for diabetes prediction," *ICT Express*, vol. 7, no. 4, pp. 432–439, Dec. 2021, doi: 10.1016/j.icte.2021.02.004.

[7] I. Tasin, T. U. Nabil, S. Islam, and R. Khan, "Diabetes prediction using machine learning and explainable AI techniques," *Healthc. Technol. Lett.*, vol. 10, no. 1–2, pp. 1–10, Feb. 2023, doi: 10.1049/htl2.12039.

[8] M. Sinsirimongkhon, S. Arwatchananukul, and P. Temdee, "Multi-Class Classification Method with Feature Engineering for Predicting Hypertension with Diabetes," *Journal of Mobile Multimedia*, vol. 19, no. 3, pp. 799–822, 2023, doi: 10.13052/jmm1550-4646.1937.

[9] M. Kudin, A. A. Ilham, and A. W. Paundu, "Performance Enhancement of Individual Learning Methods for Sentiment Analysis Using Ensemble Learning and Soft Voting Techniques," in *2023 IEEE International Conference on Communication, Networks and Satellite (COMNETSAT)*, IEEE, Nov. 2023, pp. 164–169. doi: 10.1109/COMNETSAT59769.2023.10420690.

[10] S. Alelyani, "Stable bagging feature selection on medical data," *J. Big Data*, vol. 8, no. 1, Dec. 2021, doi: 10.1186/s40537-020-00385-8.
[11] L. Li *et al.*, "Cluster-based bagging of constrained mixed-effects models for high spatiotemporal resolution nitrogen oxides prediction over large regions," *Environ. Int.*, vol. 128, pp. 310–323, Jul. 2019, doi: 10.1016/j.envint.2019.04.057.
[12] N. S. F. Putri, A. P. Wibawa, H. Al Rasyid, A. Nafalski, and U. R. Hasyim, "Boosting and bagging classification for computer science journal," *International Journal of Advances in Intelligent Informatics*, vol. 9, no. 1, pp. 27–38, Mar. 2023, doi: 10.26555/ijain.v9i1.985.
[13] Moch. Syahrir, I. N. Switrayana, and I. M. A. W. Darmawan, "Integrasi Bagging dan Stacking Untuk Memperbaiki Kinerja Algoritma Klasifikasi C4.5 dan K-Nearest Neighbor(KNN)," *JST (Jurnal Sains dan Teknologi)*, vol. 14, no. 2, pp. 218–228, Jul. 2025, doi: 10.23887/jst-undiksha.v14i2.100794.
[14] S. Sutrisno and Jupron, "Analisa Klasifikasi Penyakit Diabetes Dengan Algoritma Neural Network," *bit-Tech*, vol. 6, no. 3, pp. 303–310, Apr. 2024, doi: 10.32877/bt.v6i3.1161.
[15] K. H. Abushahla and M. A. Pala, "Optimizing Diabetes Prediction: Addressing Data Imbalance with Machine Learning Algorithms," *ADBA Computer Science*, vol. 1, no. 1, pp. 26–35, Jul. 2024, doi: 10.69882/adba.cs.2024075.
[16] O. P. Handayani, Purwono, I. A. Ashari, and R. Ardianto, "Systematic Literature Review: Penerapan Machine Learning dalam Diagnosis dan Prediksi Penyakit Diabetes," *Komputa : Jurnal Ilmiah Komputer dan Informatika*, vol. 14, no. 2, pp. 108–118, Nov. 2025, doi: 10.34010/komputa.v14i2.16642.
[17] X. Feng, Y. Cai, and R. Xin, "Optimizing diabetes classification with a machine learning-based framework," *BMC Bioinformatics*, vol. 24, no. 1, p. 428, Nov. 2023, doi: 10.1186/s12859-023-05467-x.
[18] H. L. Vu, K. T. W. Ng, A. Richter, and C. An, "Analysis of input set characteristics and variances on k-fold cross validation for a Recurrent Neural Network model on waste disposal rate estimation," *J. Environ. Manage.*, vol. 311, p. 114869, Jun. 2022, doi: 10.1016/j.jenvman.2022.114869.
[19] X. Zhang and C.-A. Liu, "Model averaging prediction by K-fold cross validation," *J. Econom.*, vol. 235, no. 1, pp. 280–301, Jul. 2023, doi: 10.1016/j.jeconom.2022.04.007.
[20] S. Kaur *et al.*, "High-accuracy lung disease classification via logistic regression and advanced feature extraction techniques," *Egyptian Informatics Journal*, vol. 29, p. 100596, Mar. 2025, doi: 10.1016/j.eij.2024.100596.
[21] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, "Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction," *Sci. Rep.*, vol. 12, no. 1, p. 6256, Apr. 2022, doi: 10.1038/s41598-022-10358-x.
[22] D. A. Anggoro and D. Permatasari, "Performance Comparison of the Kernels of Support Vector Machine Algorithm for Diabetes Mellitus Classification," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 2, 2023, doi: 10.14569/IJACSA.2023.0140226.
[23] H. Blockeel, L. Devos, B. Frénay, G. Nanfack, and S. Nijssen, "Decision trees: from efficient prediction to responsible AI," *Front. Artif. Intell.*, vol. 6, Jul. 2023, doi: 10.3389/frai.2023.1124553.
[24] H. A. Salman, A. Kalakech, and A. Steiti, "Random Forest Algorithm Overview," *Babylonian Journal of Machine Learning*, vol. 2024, pp. 69–79, Jun. 2024, doi: 10.58496/BJML/2024/007.
[25] B. Zhang, X. Dong, Y. Hu, X. Jiang, and G. Li, "Classification and prediction of spinal disease based on the SMOTE-RFE-XGBoost model," *PeerJ Comput. Sci.*, vol. 9, p. e1280, Mar. 2023, doi: 10.7717/peerj-cs.1280.

[26] K. Yongcharoenchaiyasit, S. Arwatchananukul, P. Temdee, and R. Prasad, "Gradient Boosting Based Model for Elderly Heart Failure, Aortic Stenosis, and Dementia Classification," *IEEE Access*, vol. 11, pp. 48677–48696, May 2023, doi: 10.1109/ACCESS.2023.3276468.