Comparison of Discrete Adaptive Boosting Algorithms for Classification and Regression Tree and Naive Bayes in Pistachio Nut Classification

Moch. Anjas Aprihartha[*,a,1], Salwa Paramita Azzahro[a,2], Rahmatul Azizah [a,3], Muhammad Rafly Andrianza[a,4]
[a]Dian Nuswantoro University, Semarang, Indonesia

*Corresponding Author: anjas.aprihartha@dsn.dinus.ac.id

Abstract

Machine learning is an effective tool for identifying and classifying various conditions, such as predicting shoe sales, classifying raisin types, classifying fruit productivity, and so on. This technique is widely used in various sectors. One example is pistachio sorting. In some places, pistachio sorting is still done traditionally by humans. This is disadvantageous because the costs tend to be high, and the sorting process becomes inconsistent and less effective. The use of machine learning algorithms can be a breakthrough in overcoming this problem. Naive Bayes and Classification and Regression Tree (CART) are machine learning algorithms commonly used in the classification process. To improve classification accuracy, these two basic models are integrated with the Discrete Adaptive Boosting (Discrete AdaBoost) algorithm. This study aims to assess the effectiveness of machine learning algorithms in identifying the characteristics of pistachios. Algorithm testing was carried out using the k-fold cross-validation technique. The estimated average performance results of all classification models do not show significant differences. The Discrete AdaBoost CART model has the best accuracy, specificity, and f1-score, at 86.49%, 85.78%, and 88.32%, respectively. Therefore, the Discrete AdaBoost CART model is a suitable model for classifying pistachio types. This shows that ensemble approaches such as Discrete AdaBoost CART can make a significant contribution to improving the performance of classification systems, especially in the context of data with many relevant features. This study was limited to identifying binary classes of pistachios. In further research, it is recommended to explore machine learning algorithms for multiclass of pistachio nuts.

Keywords: CART, Discrete AdaBoost, Pistachio Nuts, Machine Learning, Naive Bayes.

I. INTRODUCTION

Machine learning has emerged as a powerful and versatile tool for identifying, analyzing, and classifying a wide range of conditions and patterns across diverse domains, including but not limited to shoe sales forecasting, raisin type classification, fruit productivity assessment, and numerous other predictive tasks. Its adaptability and efficiency make it particularly valuable in sectors as varied as agriculture, retail, food technology, healthcare, and energy development, where the ability to process and interpret large-scale, high-dimensional datasets is critical for informed decision-making. At its core, machine learning involves the use of sophisticated computer algorithms and mathematical models to detect hidden structures, correlations, and trends within complex datasets, often comprising numerous interdependent variables. These algorithms enable systems to learn from data, adapt over time, and improve predictive accuracy without being explicitly programmed for each specific task. Among the many machine learning methods available, commonly used algorithms include Naive Bayes, which operates on probabilistic principles to make predictions based on prior knowledge, and Decision Tree models, which recursively partition data into subsets to enhance interpretability and classification performance. As machine learning continues to evolve, its integration into data-driven environments is becoming increasingly indispensable for optimizing processes, enhancing productivity, and generating actionable insights across both scientific and industrial applications [1].

The Naive Bayes algorithm was developed by British scientist, Thomas Bayes. This algorithm is carried out through a basic understanding of probability to estimate future events based on previous events [2]. Classification and Regression Tree (CART) is a decision tree-based algorithm that also utilizes probability calculations in building a classification model. Like a tree, the CART model consists of three parts, namely the root node, trunk/internal nodes, and leaf nodes [3]. The AdaBoost algorithm is an algorithm that supports the basic algorithm. AdaBoost is used to improve feature selection and training to address overfitting, poor classification accuracy, and high false positive rates [4]. AdaBoost can achieve the highest classification accuracy by increasing the weights which more effectively improves the classification performance than a single classifier [5].

In some locations, sorting of pistachio nuts is still carried out traditionally by humans. This is not profitable because the costs tend to be large, and the sorting process becomes inconsistent and less effective. As a solution to this problem, the application of machine learning algorithms offers an innovative solution. This technology enables machines to learn the visual characteristics of pistachios based on training data, and it enables automatic, fast, and accurate sorting. By replacing human labor in the sorting process, the use of machine learning can significantly reduce operational costs and increase efficiency and productivity [6]. Furthermore, sorting results become more consistent and reliable over the long term.

Several studies have discussed the use of machine learning in classifying pistachio nuts. Research conducted by Ozcan et al. [7] classifies pistachio nuts with the K-Nearest Neighbor (KNN) algorithm. Whilst experimental results show a high success rate reaching 94.18%. Then, research by Omid et al. [8] identified pistachio nuts into five categories using the Artificial Neural Network and Support Vector Machine (SVM) algorithm. The research results obtained accuracy for Artificial Neural Networks and SVM of 99.4% and 99.8% respectively. Another study by Singh et al. [9] identified two types of pistachio nuts through conventional neural networks such as AlexNet, VGG16, and VGG19. The success rates obtained from the AlexNet, VGG16, and VGG19 models were 94.42%, respectively; 98.84%; and 98.14. This research showed that the results of the VGG16 model could be used successfully in determining the type of pistachio nuts.

This study aims to evaluate the effectiveness of various machine learning algorithms in accurately and efficiently identifying the distinguishing characteristics of pistachios, thereby contributing to advancements in precision agriculture and intelligent food processing systems. Specifically, the research employs three prominent classification algorithms, Naive Bayes, Classification and Regression Tree (CART), and Discrete Adaptive Boosting (Discrete AdaBoost), each recognized for its distinct advantages in managing complex classification tasks. By conducting a comparative performance analysis of these algorithms, the study seeks to determine their relative accuracy, computational efficiency, and robustness in classifying pistachio varieties based on key morphological and physical attributes. The outcomes of this analysis are anticipated to provide empirical evidence that supports the integration of artificial intelligence in the development of automated pistachio sorting systems. Such systems are expected to enhance the consistency, speed, and precision of the sorting process, ultimately leading to increased productivity and reduced labor dependency. Furthermore, the findings hold broader implications for the innovation of smart agricultural technologies, offering a scalable model for the automation of quality control processes in other crop types. In doing so, the study contributes to the sustainable modernization of the food industry and agricultural sector by fostering data-driven solutions that address challenges in crop handling, quality assurance, and value chain optimization.

II. METHOD

A. Data Type and Variables

The data utilized in this study is secondary data obtained from the research conducted by Ozcan et al. [7], which originally consisted of image-based representations of pistachio nuts. In the referenced study, the image data were preprocessed and transformed into structured numerical data through image analysis techniques, allowing for quantitative evaluation. As a result of this transformation, a total of 2,148 observations were generated, each capturing detailed morphological attributes of individual pistachio samples. The dataset comprises one dependent variable and sixteen independent variables. The dependent variable represents the classification of pistachio types, divided into two distinct categories: Scarlet Pistachio and Siit Pistachio. The independent variables reflect a comprehensive set of 16 morphological characteristics that describe the physical structure and geometry of the pistachio nuts. These variables include: Area, Perimeter, Major Axis, Minor Axis, Eccentricity, Eqdiasq (Equivalent Diameter Squared), Solidity, Convex Area, Extent, Aspect Ratio, Roundness, Compactness, Shape Factor 1, Shape Factor 2, Shape Factor 3, and Shape Factor 4. Each of these features contributes valuable information regarding the size, shape, and structural integrity of the nuts, making them highly relevant for accurate classification using machine learning algorithms. This well-structured dataset serves as a reliable foundation for evaluating and comparing the performance of various classification models in identifying pistachio types based on their physical characteristics.

B. Classification and Regression Tree (CART)

CART is a non-parametric classification algorithm that produces a model in the form of a decision tree [3]. The tree structure consists of root nodes, internal nodes, and leaf nodes. The root node consists of several internal nodes. Each internal node consists of several samples separated into several categories. The allocation of samples to specific nodes depends on the threshold value adjusted for the variables selected by the algorithm. Separation continues until all nodes reach purity (all samples within the node belong to the same class) [10]. In the process of forming the model, the concept of probability is applied. The Gini index is a criterion based on impurity in measuring the differences between the probability distributions of the dependent variable [11]. If the

dependent variable consists of two classes, the probability that the sample falls into class k, with $k = 1,2$ is expressed as p_k. The Gini index can be defined as follows [12].

$$GI(p) = \sum_{k\in\{1,2\}} p_k\,(1 - p_k) = 1 - \sum_{k\in\{1,2\}} {p_k}^2 \quad (1)$$

where $p_k = {C_k}/{S}$, C_k is the number of samples in the k-*th* class from the sample set S.

C. Naive Bayes

Naive Bayes is a parametric classification algorithm derived from a certain distribution. The model produced by this algorithm is a set of probabilities that can be used for classification. In solving classification problems, the Naive Bayes algorithm applies the Bayes approach. Bayesian decision theory targets to minimize inappropriate decisions or desired risks [13]. Bayes' theorem is defined as follows:

$$P(Y = C_k|X = x) = \frac{P(Y=C_k)P(X=x|Y=C_k)}{P(X=x)} \quad (2)$$

Because the independent variables (X_i) for $i = 1,2,\cdots,n$ are assumed to be conditionally independent, equation (2) can be derived as follows:

$$P(Y = C_k|X = x) = \frac{P(Y=C_k)\prod_{i=1}^{n} P(X_i=x_i|Y=C_k)}{P(X_1=x_1,X_2=x_2,\cdots,X_n=x_n)} \quad (3)$$

One type of popular Naive Bayes algorithm is Gaussian Naive Bayes [14]. This algorithm is executed following a Gaussian distribution. If the independent variable (X_i) is continuous, the decision function is derived from the Gaussian distribution is defined as follows:

$$P(X_i = v|Y = C_k) = \frac{1}{\sqrt{2\pi{\sigma_{ik}}^2}} \exp\left(-\frac{(v-\mu_{ik})^2}{2{\sigma_{ik}}^2}\right) \quad (4)$$

where μ_{ik} and σ_{ik} are estimates of the mean and standard deviation.

D. Discrete Adaptive Boosting (Discrete AdaBoost/ DAB)

Discrete Adaptive Boosting (Discrete AdaBoost or DAB) is a sophisticated ensemble learning technique introduced by Freund and Schapire [15], designed to enhance the performance of weak classifiers by iteratively combining them into a single, highly accurate predictive model. Unlike traditional classification methods that rely on a single algorithm, Discrete AdaBoost employs multiple base classifiers—typically simple or "weak" learners such as decision stumps—which are trained sequentially, with each subsequent model focusing more on the instances misclassified by its predecessors [16]. The core mechanism of this approach lies in adaptively adjusting the weights of training samples after each iteration based on the accuracy of the previous classifier. Misclassified instances are assigned higher weights, thereby forcing the next classifier to pay more attention to these harder-to-classify examples. This weight adjustment process continues iteratively, allowing the ensemble to progressively correct its errors and improve classification performance over time. As a result, the final model is a weighted combination of all the base classifiers, which often outperforms any individual classifier due to the synergy among them. This boosting strategy enhances both the robustness and generalization capability of the model, making it particularly effective for complex classification tasks in domains with high-dimensional or noisy data [17]. The algorithm follows a structured sequence of steps that systematically build and refine the ensemble model, ultimately producing a strong learner capable of delivering high predictive accuracy:

1. Set initial weights on the training data $w_i = \frac{1}{N}, i = 1, 2, \cdots, N$.
2. Repeat for $m = 1, 2, 3, \cdots, M$.
   a) Create a classification model k_m with weights w_i on the training data.
   b) Identify $I(k_m(x), y) = \begin{cases} 0\ jika\ k_m(x) = y \\ 1\ jika\ k_m(x) \neq y \end{cases}$
   c) Calculate $\varepsilon_m = \frac{\sum_{i=1}^{N} w_i I(k_m(x),y)}{\sum_{i=1}^{N} w_i}$
   d) Calculate $a_m = \ln\frac{1-\varepsilon_m}{\varepsilon_m}$
   e) Update weights $w_i \leftarrow w_i \exp a_m\, I(k_m(x), y)$
3. Test the classification model with testing data.
4. Calculate model performance.

E. Majority Vote

Majority voting is a widely adopted decision-making technique in ensemble learning methods, which aims to improve classification accuracy by combining the predictions of multiple individual classifiers [18]. This method operates by selecting the class label that receives the highest number of votes from the ensemble of classifiers, essentially identifying the most frequently predicted class across all models. The effectiveness of majority voting relies on the assumption that each classifier contributes independently to the final decision, meaning that the classifiers should ideally be trained on different subsets of the training data or constructed using different learning algorithms to ensure diversity. This independence reduces the likelihood of correlated errors and enhances the ensemble's ability to generalize to unseen data. By leveraging the collective knowledge of multiple models, majority voting helps mitigate the weaknesses of individual classifiers, leading to more robust and reliable predictions. It is particularly effective when the individual classifiers have complementary strengths, allowing the ensemble to outperform any single model in terms of accuracy and consistency. Assume F is a set of classifiers that has members $k_1(x), k_2(x), \cdots, k_M(x)$. The majority vote formula can be stated as follows [19]:

$$K(x_i) = modus(k_1(x_i), k_2(x_i), \cdots, k_M(x_i)) \quad (5)$$

F. K-Fold Cross Validation

K-fold cross validation is a tool used to measure classification models. This technique applies training to several classification models from various subsets of training data. This allows the resulting accuracy values to vary, depending on the number of folds used to divide the data in the process [20]. The aim of using this technique is to identify overfitting problems in the model, namely the model has difficulty understanding patterns in new, unseen data [21]. Basically k-fold cross validation can be done by following these steps [22][23]:

1. Divide the dataset based on a specified size k, so that it has k data subsets.
2. In the first round, train k-1 subset of data for training data, and another subset for validation. Calculate the performance of the model.
3. Perform additional rounds up to k times by repeating step 2, but then change the data subset for training the model and the data subset for validation.
4. Compute the average model performance over each k rounds.

G. Model Performance Evaluation

Each classification model has different advantages in processing data. Whether a model is good or bad at analyzing data depends on the dataset used. Several measures that can be applied to assess the quality of a classification model are accuracy, precision, recall, specificity, and f1-score [24]. These measurements can be expressed in a formula as in Table 1[25]:

Table 1. Model Performance Formula

Measurement	Formula
Accuracy	$\frac{TP + TN}{P + N}$
Precision	$\frac{TP}{TP + FP}$
Recall	$\frac{TP}{P}$
Spesificity	$\frac{TN}{N}$
F1-Score	$\frac{2 \times presisi \times recall}{presisi + recall}$

where P=TP+TN and N=FP+FN. True Positive (TP) represents the total positive observations labelled correctly, True Negative (TN) represents the total negative observations labelled incorrectly, False Positive (FP) represents the total positive observations labelled incorrectly, and False Negative (FN) represents the total negative observations labelled incorrectly, P represents the number of observations labelled correctly, and N represents the number of observations labelled incorrectly.

III. RESULTS AND DISCUSSION

The initial step that must be taken before the process of extracting data information is to change the data structure of each numerical variable so that it becomes uniform. The goal is that there are no variables that stand out from other variables so that they do not interfere with the results of data analysis. This process involves the Z-Score Normalization transformation.

$$Z = \frac{x_{ij} - \bar{x}_j}{\sigma_j}$$

where x_{ij} is the ith observation on the j-th variable, $\bar{x}_j$ is the average on the j-th variable, and σ_j is the standard deviation on the j-th variable. A summary of data transformation is shown in Table 2. The visualization of each independent variable is expressed in a boxplot as in Figure 1.

Table 2 Data Transformation Results

No	Area	Perimeter	Major Axis	Minor Axis	⋯	Class
1	-1,262	0,37925	-1,72319	-0,05166	⋯	Kirmizi Pistachio
2	-0,88347	1,374501	-1,09075	-0,11743	⋯	Kirmizi Pistachio
3	-0,48482	-0,47777	0,188441	-0,58584	⋯	Kirmizi Pistachio
4	-0,67405	0,051361	-0,51532	-0,73358	⋯	Kirmizi Pistachio
5	0,010391	-0,46449	0,712872	-0,57331	⋯	Kirmizi Pistachio
6	-2,10968	-0,72377	-1,92303	-1,34638	⋯	Kirmizi Pistachio

Table 2 presents the results of the data transformation process applied to the pistachio dataset, in which several numerical features have been standardized or normalized to ensure consistency in scale and distribution. Each row represents an individual observation of a pistachio sample, with transformed values for attributes such as Area, Perimeter, Major Axis, Minor Axis, and others not fully shown in the excerpt. These transformed values appear to be the result of standardization, as indicated by the presence of both negative and positive values centered around zero, which helps improve the performance and convergence of machine learning algorithms. The final column, labeled Class, indicates the corresponding pistachio type classification, with all samples in the provided excerpt belonging to the Kirmizi Pistachio class. This transformation step is crucial for ensuring that no single variable dominates the learning process due to scale differences, thereby enhancing the fairness and effectiveness of the subsequent classification analysis.

The boxplot visualization, as seen in Figure 1, displays the distribution of several standardized features related to pistachio characteristics, including Area, Major Axis, Eccentricity, Solidity, Extent, Roundness, Shapefactor_1, and Shapefactor_4. Each feature has been normalized, as indicated by the median values centered around zero. The boxplots show the interquartile range (IQR), with the middle 50% of the data lying between the first and third quartiles, while the whiskers extend to the minimum and maximum values within 1.5 times the IQR. Numerous outliers are evident across almost all features, particularly in Shapefactor_1, Roundness, and Major Axis, as shown by the many individual points outside the whiskers. Despite the presence of outliers, the symmetry and similar spread of most boxes suggest relatively consistent scaling across features. This visualization confirms the success of the standardization process and provides an overview of feature distributions, which is important for informing subsequent modeling steps in the machine learning pipeline.

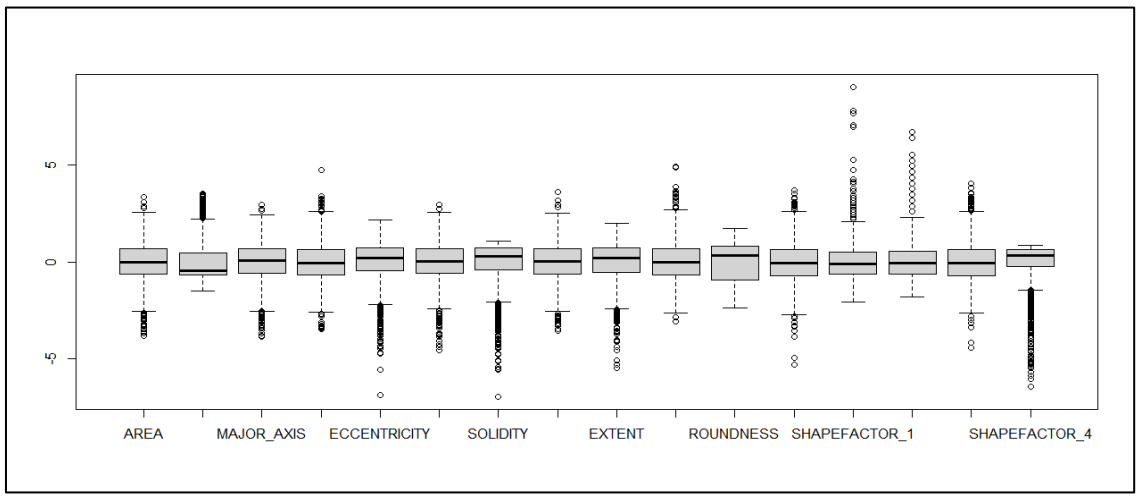

Figure 1 Boxplot of Independent Variables

After the data is transformed, the next process estimates how important each independent variable is in influencing the classification of pistachio nuts. Measuring the significance of each variable involves the CART and Naive Bayes algorithms. The test results are shown in Figure 2.

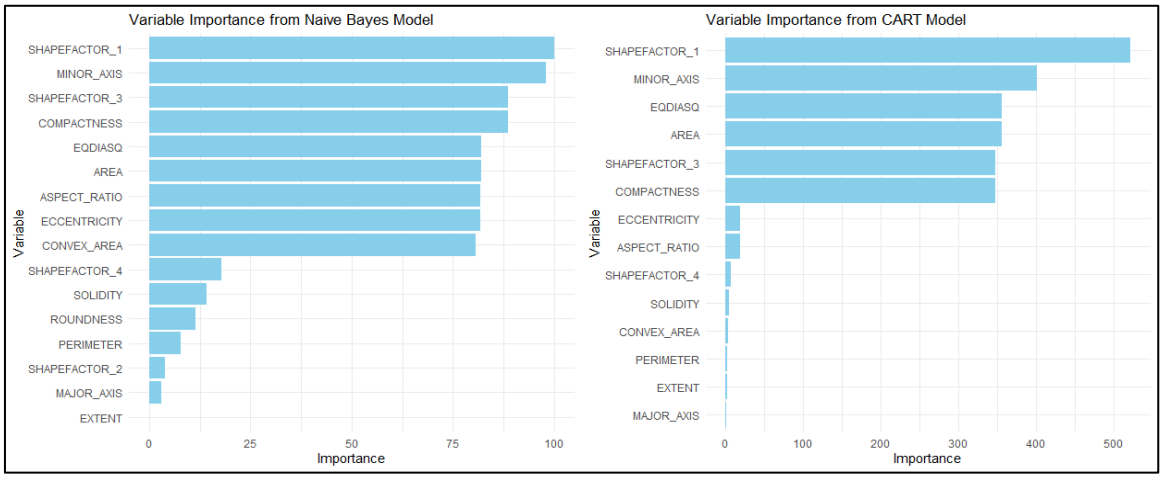

Figure 2 Variable Importance of a Single Classification Models

The CART and Naive Bayes analyses show that the shapefactor_1 and minor axis variables both have a significant influence on model formation. Both provide the largest contribution to the pistachio classification process. The shapefactor_3 variable in the Naive Bayes model and eqdiasq in the CART model both occupy the third position as variables that have an important contribution to the classification of pistachios. The shapefactor_4 variable from both models produces a lower influence than the previous variable, so it tends to form a relatively small model. Meanwhile, in CART, the shapefactor_2 and roundness variables are in the lowest position; neither has a strong influence on the classification of pistachios. Meanwhile, in Naive Bayes, the area and major axis variables do not play an important role in the prediction process and are considered irrelevant in model formation. The classification algorithms applied in this research are CART, Naive Bayes, Discrete Adaptive Boosting CART (DAB-CART), and Discrete Adaptive Boosting Naive Bayes (DAB-NB). In the analysis process, the k-fold cross-validation technique is used to see how well the classification model performs. The dataset of 2148 observations were partitioned into 10 equal parts. Nine sections were used to train the model, while one section was used to test the model. The process of training and testing the model was repeated 10 times, each time the researchers used different training and testing datasets. Then the accuracy, recall, precision, specificity, and f1-score for each iteration were calculated. Then, the results of measuring model performance are shown in Figure 3.

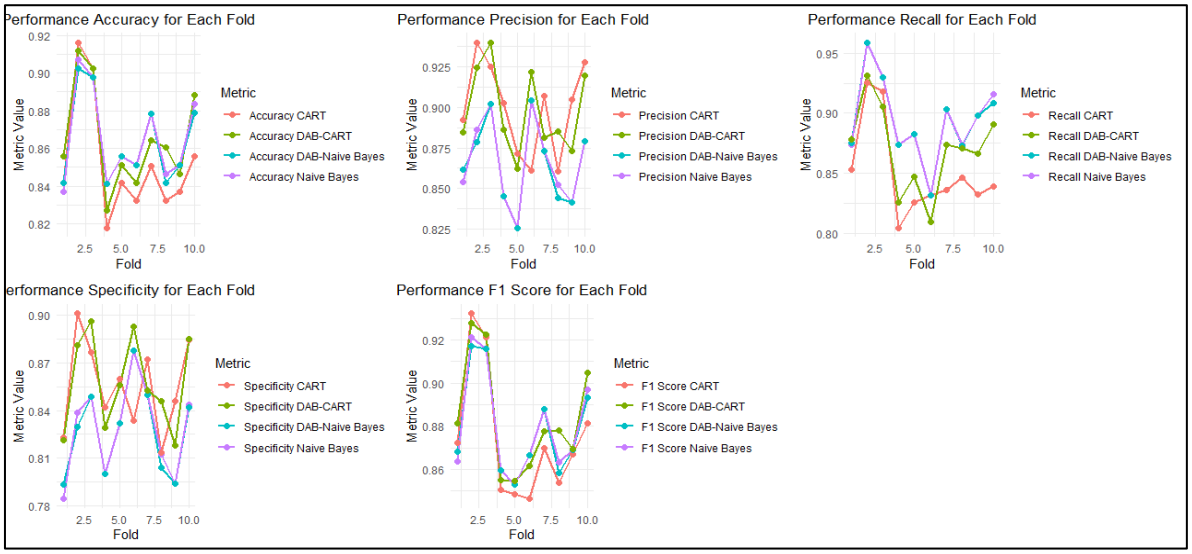

Figure 3. k-Fold Cross-Validation Plot of Classification Model

The first plot displays the accuracy of the classification models on each fold. The highest accuracy in the Naive Bayes model is 0.906 (90.6%), which means the Naive Bayes model succeeds in classifying the two classes correctly 90.6% and the remaining 9.4% is misclassification. The highest accuracy of the CART, DAB-Naive Bayes, and DAB-CART models is 0.916 (91.6%) each; 0.903 (90.3%); and 0.911 (91.1%). The second plot shows a measure of precision. The highest precision in the Naive Bayes model is 0.904 (90.4%), which means that of all the Scarlet Pistachio class predictions made by the model, 90.4% of them are truly the Scarlet Pistachio class, while the remaining 9.6% are misclassified. The highest precision of the CART, DAB-Naive Bayes, and DAB-CART models are 0.939 (93.3%) respectively; 0.904 (90.4%); and 0.939 (93.9%). The third plot depicts the recall measure. The highest recall in the Naive Bayes model is 0.959 (95.9%), which shows that the model succeeds in classifying the Scarlet Pistachio class correctly 95.9%, the remaining 4.1% is a misclassification. The highest recall of the CART, DAB-Naive Bayes, and DAB-CART models respectively is 0.925 (92.5%); 0.958 (95.8%); and 0.931 (93.1%). The fourth plot shows the specificity measure. The highest specificity of the Naive Bayes model is 0.877 (87.7%) which shows that the model succeeds in correctly classifying the Siit Pistachio class 87.7% of the time, the remaining 12.3% was misclassification. The highest specificity of the CART, DAB-Naive Bayes, and DAB-CART models is 0.901 (90.1%) each; 0.848 (84.8%); and 0.896 (89.6%). The fifth plot shows the f1-score measure. The highest F1-score of the Naive Bayes model is 0.921 (92.1%), which means the model has an ability of 92.1% in detecting the Scarlet Pistachio class and ensuring that the predictions for the Scarlet Pistachio class made by the model are indeed correct in the Scarlet Pistachio class. The highest F1-score of the CART, DAB-Naive Bayes, and DAB-CART models is 0.932 (93.2%) each; 0.916 (91.6%); and 0.927 (92.7%).

To provide a more accurate and stable estimate of model performance, it is necessary to take the average value of the k-fold cross validation results for each model performance measure. This can help in selecting the best model. The averages of each model's performance measure are presented in Table 3.

Table 3 Average Model Performance

Measurement Models	Mean			
	CART	**DAB-CART**	**NB**	**DAB-NB**
Accuracy	0,8528	**0,8649**	0,8649	0,8640
Precision	**0,8991**	0,8977	0,8663	0,8655
Recall	0,8512	0,8698	**0,8943**	0,8935
Specificity	0,8552	**0,8578**	0,8281	0,8271
F1-Score	0,8742	**0,8832**	0,8797	0,8789

Table 3 presents the average performance measures for each model do not have a significant difference. Each model has good prediction quality, indicated by a model percentage above 0.85 (85%). The greatest average accuracy was produced by the DAB-CART and Naive Bayes models at 0.8649 (86.49%). The largest average precision comes from the CART model, namely 0.8991 (89.11%). The highest average recall comes from the Naive Bayes model. The highest average specificity produced by the DAB-CART model was 0.8578 (85.78%). Finally, the highest average f1-score came from the DAB-CART model at 0.8832 (88.32%). The DAB-CART model has the advantage of the best accuracy, specificity and F1-score performance compared to other models. Meanwhile, the CART and Naive Bayes models are superior in terms of precision and recall, respectively. Thus, the DAB-CART model is a suitable model to use in classifying types of pistachio nuts.

The results of similar research were carried out by Aprihartha et al. [11], which compares the CART, DAB-CART, and Naive Bayes methods in predicting tire product sales. The DAB-CART model outperformed the single CART and Naive Bayes models with accuracy, sensitivity, and specificity of 79.17%, 89.47%, and 69.84%. Research by Liu et al. [17] predicted rock masses using the Adaboost CART and CART algorithms. The test results show that the accuracy and f1-score size of AdaBoost-CART are 86.5% and 77% respectively, better than the standard CART results of 75.3% and 62.9%. Another research from Chen et al. [26] compared the CART-Adaboost algorithm with Naive Bayes Adaboost. The research results show that the accuracy, precision, recall, and f1-score of the CART-AdaBoost model are 83.06%, 84.31%, 77.48%, and 80.75% higher compared to Naive Bayes AdaBoost, which was 65.29% respectively 63.48%, 56.76%, and 60%. Meanwhile, research from Nugroho [27], shows that the best combination for hypertension diagnosis classification is the CART decision tree model optimized with AdaBoost, with an increase in accuracy from the original 95.91% to 96.78%. Thus, the use of Discrete AdaBoost in the feature selection process proves more effective than a single model, as it can improve classification accuracy and reduce model complexity. This indicates that ensemble approaches such as Discrete AdaBoost can make a significant contribution to improving the performance of classification systems, especially in the context of data with many relevant features.

IV. CONCLUSION

Based on the results and discussion, several conclusions can be drawn. This study applies classification analysis using four types of algorithms, namely CART, Naive Bayes, Discrete AdaBoost CART, and Discrete AdaBoost Naive Bayes. CART and Naive Bayes analysis show that the shapefactor_1 and minor axis variables both have a significant influence on the process of forming a pistachio nut classification model. The estimated results of the average performance of the classification model do not have a significant difference. The DAB-CART model has the advantage of the best accuracy, specificity, and F1-score performance compared to other models. Meanwhile, the CART and Naive Bayes models are superior in terms of precision and recall, respectively. Therefore, the DAB-CART model is a suitable model to use in classifying types of pistachio nuts. In optimizing this research, further research can be explored with variations of other machine learning algorithms, such as K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Random Forest, etc.

REFERENCES

[1] M. N. Raihen and S. Akter, "Prediction modeling using deep learning for the classification of grape-type dried fruits," *International Journal of Mathematics and Computer in Engineering*, 2024. http://dx.doi.org/10.2478/ijmce-2024-0001

[2] A. Z. M. S. Widodo, A. P. Kusuma, and W. D. Puspitasari, "Analisis algoritma naive bayes classifier (NBC) pada klasifikasi tingkat minat barang di toko violet cell," *Jati (Jurnal Mahasiswa Teknik Informatika)*, vol. 7, no. 1, pp. 87-94, 2023. https://doi.org/10.36040/jati.v7i1.5692

[3] A. Aprihartha, Z. Putrawan, D. Zulhan, and F. A. Nurfaizal, "Klasifikasi Produktivitas Buah Nanas Menggunakan Algoritma Classification and Regression Tree (CART)," *Diophantine Journal of Mathematics and Its Applications*, pp. 64-70, 2024. https://doi.org/10.33369/diophantine.v3i1.34193

[4] Attou, H., Guezzaz, A., Benkirane, S. *et al.* A New Secure Model for Cloud Environments Using RBFNN and AdaBoost. *SN COMPUT. SCI.* **6**, 188 (2025). https://doi.org/10.1007/s42979-025-03691-1

[5] F. Ramadhani, Al-Khowarizmi, and I. P. Sari, "Improving the Performance of Naïve Bayes Algorithm by Reducing the Attributes of Dataset Using Gain Ratio and Adaboost," in *2021 International Conference on Computer Science and Engineering (IC2SE)*, IEEE, Nov. 2021, pp. 1–5. doi: 10.1109/IC2SE52832.2021.9792027.

[6] B. O. U. M. A. R. A. F. Ibtissam, *Automatic date fruit sorting system based on machine learning and visual features*, Doctoral dissertation, University of Biskra, 2024. [Online]. Available: http://archives.univ-biskra.dz/handle/123456789/29324

[7] I. A. Ozkan, M. Koklu, and R. Saraçoğlu, "Classification of pistachio species using improved k-NN classifier," *Health*, vol. 23, p. e2021044, 2021. https://www.mattioli1885journals.com/index.php/progressinnutrition/article/view/9686

[8] M. Omid, M. S. Firouz, H. Nouri-Ahmadabadi, and S. S. Mohtasebi, "Classification of peeled pistachio kernels using computer vision and color features," *Eng. Agric. Environ. Food*, vol. 10, pp. 259–265, 2017. https://doi.org/10.1016/j.eaef.2017.04.002

[9] D. Singh, Y. S. Taspinar, R. Kursun, I. Cinar, M. Koklu, I. A. Ozkan, and H. N. Lee, "Classification and analysis of pistachio species with pre-trained deep learning models," *Electronics*, vol. 11, no. 7, p. 981, 2022. https://doi.org/10.3390/electronics11070981

[10] M. G. bin Md Ghazi, L. C. Lee, A. S. B. Samsudin, and H. Sino, "Evaluation of ensemble data preprocessing strategy on forensic gasoline classification using untargeted GC–MS data and classification and regression tree (CART) algorithm," *Microchemical Journal*, vol. 182, p. 107911, Nov. 2022, https://doi.org/10.1016/j.microc.2022.107911

[11] M. Anjas Aprihartha, F. Astutik, and N. Sulistianingsih, "Comparison of Naïve Bayes, CART, dan CART Adaboost Methods in Predicting Tire Product Sales," *Jurnal Matematika, Statistika dan Komputasi*, vol. 20, no. 3, pp. 596–605, May 2024, https://doi.org/10.20956/j.v20i3.33187

[12] M. A. Aprihartha, J. Prasetya, and S. I. Fallo, "Implementasi CART-Real Adaboost dalam Memprediksi Minat Pelanggan Membeli Sepatu," *Jurnal EurekaMatika*, vol. 12, no. 1, pp. 35-46, 2024., https://doi.org/10.17509/jem.v12i1.67808.

[13] J. Prasetya, S. I. Fallo, and M. A. Aprihartha, "Stacking Machine Learning Model for Predict Hotel Booking Cancellations," *Jurnal Matematika, Statistika dan Komputasi*, vol. 20, no. 3, pp. 525–537, May 2024, https://doi.org/10.20956/j.v20i3.32619

[14] T. A. Munshi, L. N. Jahan, M. F. Howladar, and M. Hashan, "Prediction of gross calorific value from coal analysis using decision tree-based bagging and boosting techniques," *Heliyon*, vol. 10, no. 1, p. e23395, Jan. 2024, https://doi.org/10.1016/j.heliyon.2023.e23395

[15] Y. Freund and R. E. Schapire, "Experiments with a new boosting algorithm," in *Proc. ICML*, vol. 96, pp. 148-156, July 1996. https://cseweb.ucsd.edu/~yfreund/papers/boostingexperiments.pdf

[16] G. Hu, C. Yin, M. Wan, Y. Zhang, and Y. Fang, "Recognition of diseased Pinus trees in UAV images using deep learning and AdaBoost classifier," *Biosystems Engineering*, vol. 194, pp. 138–151, Jun. 2020, https://doi.org/10.1016/j.biosystemseng.2020.03.021.

[17] Q. Liu, X. Wang, X. Huang, and X. Yin, "Prediction model of rock mass class using classification and regression tree integrated AdaBoost algorithm based on TBM driving data," *Tunnelling and Underground Space Technology*, vol. 106, p. 103595, Dec. 2020, https://doi.org/10.1016/j.tust.2020.103595.

[18] M. A. Naji, S. el Filali, M. Bouhlal, E. H. Benlahmar, R. A. Abdelouhahid, and O. Debauche, "Breast Cancer Prediction and Diagnosis through a New Approach based on Majority Voting Ensemble Classifier," *Procedia Computer Science*, vol. 191, pp. 481–486, 2021, https://doi.org/10.1016/j.procs.2021.07.061.

[19] G. M. James, "Majority vote classifiers: theory and applications," Stanford University, 1998. https://hastie.su.domains/THESES/gareth_james.pdf

[20] R. Kumalasanti and N. M. Dina Aprilianti, "Sentiment Analysis of Bali Calendar Application Reviews using K-Nearest Neighbour," *International Journal of Engineering Technology and Natural Sciences*, vol. 6, no. 1, pp. 67–74, Jul. 2024, https://dx.doi.org/10.46923/ijets.v6i1.339

[21] T. Ait tchakoucht, B. Elkari, Y. Chaibi, and T. Kousksou, "Random forest with feature selection and K-fold cross validation for predicting the electrical and thermal efficiencies of air based photovoltaic-thermal systems," *Energy Reports*, vol. 12, pp. 988–999, Dec. 2024, https://doi.org/10.1016/j.egyr.2024.07.002.

[22] A. Aprihartha, "Penyelesaian Masalah Ketidakseimbangan Data Melalui Teknik Oversampling dan Undersampling pada Klasifikasi Siswa Tidak Naik Kelas," *Jurnal Teknik Ibnu Sina (JT-IBSI)*, vol. 9, no. 01, pp. 43-52, 2024. https://doi.org/10.36352/jt-ibsi.v9i01.807.

[23] Moch. A. Aprihartha, M. Husniyadi, and T. N. Alam, "Implementasi Metode Random Forest Dalam Memprediksi Sinyal Pergerakan Saham," *E-Jurnal Matematika*, vol. 14, no. 1, p. 43, Jan. 2025, https://doi.org/10.24843/MTK.2025.v14.i01.p477

[24] M. anjas Aprihartha, Z. Putrawan, D. Zulhan, and F. A. Nurfaizal, "Study on Identification of Poisonous and Non-Toxic Mushrooms Using the Cart-Logitboost Algorithm," *Jurnal Matematika, Statistika dan Komputasi*, vol. 21, no. 1, pp. 33–45, Sep. 2024, https://doi.org/10.20956/j.v21i1.35072.

[25] M. A. Aprihartha, T. N. Alam, and M. Husniyadi, "Perbandingan Metrik Euclidean dan Metrik Manhattan untuk K-Nearest Neighbors dalam Klasifikasi Kismis," *Jurnal Ilmu Komputer dan Informatika*, vol. 4, no. 1, pp. 21-30, 2024. https://doi.org/10.54082/jiki.126.

[26] X. Li, X. Chen, and Z. Yuan, "Applicable model of liver disease detection based on the improved CART-AdaBoost algorithm," in *2021 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, IEEE, Jun. 2021, pp. 1177–1181. https://doi.org/10.1109/ICAICA52286.2021.9498046.

[27] Y. A. Nugroho, *Implementasi metode Decision Tree CART dan Adaboost sebagai Ensemble Learning dalam penentuan klasifikasi diagnosis hipertensi*, Doctoral dissertation, Universitas Islam Negeri Maulana Malik Ibrahim, 2025. [Online]. Available: http://etheses.uin-malang.ac.id/id/eprint/74780