J. Ris. & Ap. Mat. Vol. 10 No. 1 (2026) pp. 84-96
Jurnal Riset dan Aplikasi Matematika (JRAM)
e-ISSN: 2581-0154
DOI: doi.org/10.26740/jram.v10n1.p84-96

ANALYSIS OF DECISION TREE-BASED PREDICTIVE MODELS FOR STUNTING PREVALENCE IN JAVA USING SOCIO-ENVIRONMENTAL PARAMETERS

KARTIKA CHANDRA DEWI[1,*], DINDA GALUH GUMINTA[1], YUNI ROSITA DEWI[1], AND IKE FITRIYANINGSIH[2]

[1]Data Science Study Program, Faculty of Mathematics and Natural Sciences, Universitas Negeri Surabaya, Surabaya, Indonesia
[2]Artificial Intelligence Study Program, Faculty of Mathematics and Natural Sciences, Universitas Negeri Surabaya, Surabaya, Indonesia

*Corresponding Author: kartikadewi@unesa.ac.id

ABSTRACT

Stunting prevalence indicates the percentage of children under five whose height falls below the WHO standard. As the most densely populated region in Indonesia, Java had a critical role in policy formulation related to stunting reduction. This study aimed to develop a predictive model of stunting prevalence in Java influenced by social and environmental factors using tree-based machine learning approaches, Classification and Regression Tree (CART), Random Forest, and Extreme Gradient Boosting (XGBoost). Simulations were conducted using these three models, and model performance was evaluated using Root Mean Square Error (RMSE). Feature selection based on feature importance values was applied with proportions of 25%, 50%, 75%, and 100% of the features. The results indicated that XGBoost achieved the best performance with a mean RMSE of 5.3820 using 50% of the features and demonstrated the highest prediction stability. Across the best-performing configurations of each model, four features were consistently selected: Posyandu Activities, Toddler Mothers Class, Caregiving Class, and Utilization of Family and Village Yard Land. These findings indicated that strengthening interventions in caregiving practices, community-based health education services, and environmental resource utilization had the potential to become priority programs in efforts to reduce stunting prevalence in Java.

Keywords: *Feature Selection, Socio-environmental, Stunting Prevalence, Tree-based Machine Learning, XGBoost,*

1. Introduction

According to the World Health Organization (WHO), stunting is a condition characterized by impaired growth and development in children resulting from malnutrition, repeated infections, and insufficient psychosocial stimulation [1]. Stunting, often referred to as dwarfism or short stature, occurs in children under five years of age due to chronic nutritional deficiencies and repeated infections, particularly during the first 1,000 days of life, from conception to 23 months of age. A child is considered stunted if their height falls below minus two standard deviations of the height of children of the same age, often appearing younger than their chronological age [2, 3].

2020 *Mathematics Subject Classification*: 62R07
Received: 19-04-2026; revised: 27-04-2026; accepted: 28-04-2026

In 2021, UNICEF reported that during 2018–2019 in Southeast Asia, Indonesia had the highest stunting prevalence at 31.8% [4]. Stunting prevalence indicates the percentage of children under five whose height is below WHO standards. In addition to increasing the risk of impaired cognitive development, affecting intelligence and future productivity, stunting and other nutritional problems are estimated to reduce the gross domestic product (GDP) by approximately 3% per year [5]. This highlights that stunting remains a serious public health issue in Indonesia, including on Java Island, the most densely populated region in the country. The causes of stunting include direct and indirect factors. Direct factors involve insufficient nutritional intake and infectious diseases, whereas indirect factors include food security, social environment (infant and child feeding, hygiene, education, workplace conditions), health environment, and living conditions (access to clean water, drinking water, and sanitation facilities) [3]. Understanding the relationships among these factors and predicting stunting prevalence is essential for designing effective government policies and intervention strategies.

Predicting stunting prevalence and analyzing the relationships among causal factors can be achieved by modeling historical stunting data. Machine learning is a suitable approach for this purpose, as it develops models that learn from data to extract knowledge in a manner analogous to human learning. Machine learning can handle large, complex datasets and provide rapid responses, making it suitable for stunting prediction, which is framed as a regression problem under supervised learning.

Previous studies on stunting using machine learning have applied ensemble models (Random Forest, XGBoost), deep learning (Deep Neural Networks), and other approaches such as Support Vector Machines (SVM), Logistic Regression, K-Nearest Neighbors (KNN), and Gradient Boosting. These studies focus on predicting child malnutrition as early detection of stunting and wasting. This research also applies techniques such as Synthetic Minority Over-sampling (SMOTE) to address data imbalance. Ensemble models generally outperform individual models, with Random Forest achieving up to 100% accuracy and XGBoost 99.49% in certain supervised learning classification tasks [6].

Machine learning models such as Random Forest, SVM, KNN, and regularized linear regression have also been used to perform spatial analysis and predict household-level stunting prevalence in India. Data combined demographic survey data (NFHS-5) and environmental-spatial variables from satellite imagery, with Random Forest providing the highest predictive accuracy for regression tasks [7]. In Indonesia, studies have applied explainable machine learning frameworks to predict stunting in children under five using anthropometric and simple demographic data. Models used include Logistic Regression, SVM, Multi-Layer Perceptron (MLP), KNN, Decision Tree, Random Forest, XGBoost, and Convolutional Neural Networks (CNN). SHapley Additive Explanations (SHAP) were employed for global and local feature interpretability, with XGBoost performing best (97.57% accuracy), followed by Random Forest (97.28%) and Decision Tree (96.62%) using data from Jeneponto Regency, South Sulawesi (2021–2024) in a binary classification setting [8].

Based on prior research, decision tree-based methods consistently achieve the best performance compared to other machine learning approaches for both classification and regression problems in stunting prediction. Therefore, this study applies decision tree-based machine learning methods, Classification and Regression Tree (CART), Random Forest, and XGBoost, to predict stunting prevalence on Java Island, Indonesia's most densely populated region, as a regression problem. The analysis also examines the relationships among stunting determinants, including social and environmental factors. The main focus of this study is to build predictive models for predicting stunting prevalence and identify the most influential variables affecting stunting on Java Island using decision tree-based machine learning approaches.

2. Literature Review

2.1. Research Methodology

This study used secondary data on stunting prevalence at the district/city level in Java, covering social and environmental variables. The research procedures were as follows:

- Data preprocessing, which included selecting numerical features to be used in the modeling process, as well as performing descriptive and statistical analysis of the data.
- Model fitting, employing CART, Random Forest, and XGBoost. Model validation was conducted using 5-fold cross-validation (CV = 5).
- Hyperparameter optimization using GridSearchCV to identify the best combination of hyperparameter for each model.
- Feature importance analysis for each model based on the optimal hyperparameters using all available features. The output of this process was a ranked list of features according to their contribution to predicting stunting prevalence.
- Feature selection based on predetermined proportions, conducted to determine the optimal number of features required to achieve the best model performance according to feature importance.
- Model retraining for each algorithm using the selected feature subsets.
- Model evaluation using Root Mean Square Error (RMSE) to assess predictive performance. The Root Mean Square Error (RMSE) is a commonly used metric for evaluating model performance. For a dataset consisting of n observations $y\,(y_i, i = 1,2, \dots, n)$, with n corresponding model predictions $\hat{y}_i$, RMSE is formulated as follows:.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \quad (1)$$

The RMSE value ranges from 0 to $+\infty$, with the best value = 0 and the worst = $+\infty$ [9] [10]. In machine learning, RMSE is used to measure the prediction error of a model.

2.2. Dataset Description

The data used in this study are secondary data from the Village Potential Statistics published for each province in Java in 2024. The research units consist of regencies/cities in Java with 119 samples. Table 1 presents the research variables, indicating the number of villages/urban villages according to the availability of service packages related to stunting in 2023 [11]. All independent variables are expressed as proportions (0–1) to represent the coverage of intervention indicators at the regency/city level. The independent variables are further classified into social factors and environmental factors. Social factors represent dimensions of health services, education, social protection, caregiving behavior, and household food security. Environmental factors represent basic sanitation conditions that may affect child health and growth. Table 1 below provides a description and classification of the research variables used in this study.

Table 1. Description and Classification of Research Variables

No.	Variable	Variable Type	Factor Group
1	Stunting Prevalence	Dependent	-
2	Posyandu Activities	Independent	Social Factor
3	Pregnant Women Class	Independent	Social Factor
4	Toddler Mothers Class	Independent	Social Factor

5	Supplementary Feeding for Pregnant Women with Chronic Energy Deficiency / High-Risk from Poor Families	Independent	Social Factor
6	Access to Safe Drinking Water	Independent	Environmental Factor
7	Access to Sanitary Latrines	Independent	Environmental Factor
8	Health Insurance for Pregnant Women from Poor Families	Independent	Social Factor
9	Health Insurance for Children under Two from Poor Families	Independent	Social Factor
10	Birth Certificate for Infants from Poor Families	Independent	Social Factor
11	Caregiving Class	Independent	Social Factor
12	Utilization of Family and Village Yard Land	Independent	Environmental Factor

2.3. Classification and Regression Tree (CART)

CART is a data analysis approach that employs decision tree algorithms for exploration and modeling. This method was first introduced by Leo Breiman in 1984 [12]. CART generates a regression tree when the response (target) variable is numerical. A CART tree is a binary decision tree formed by recursively partitioning a node into two child nodes. The process begins with a root node containing all training samples. Nodes that can no longer be split are called terminal nodes. These terminal nodes form the final partitions of the predictor space. The fundamental concept in constructing a CART tree is to choose, at each node, the split from all possible candidates that generates child nodes with parameters that minimize impurity.

Let Q be the set of data at node V. For every potential split candidate $\theta = (l, t_V)$, which includes feature l and threshold value t_V, the dataset Q is partitioned into $Q_{left}(\theta)$ for the left child node (V_{left}) and $Q_{right}(\theta)$ for the right child node (V_{right}). The subset $Q_{left}(\theta)$ can be obtained using the following equation:

$$Q_{left}(\theta) = \{(\boldsymbol{x}, y) | x_c \leq t_V\} \tag{2}$$

where $\boldsymbol{x}$ represents the features of the training data satisfying $x_c \leq t_V$, y is the response (target) variable corresponding to $\boldsymbol{x}$, x_c denotes the c-th feature, and t_V is the threshold value for node V. Subsequently, the subset $Q_{right}(\theta)$ can be determined using the following equation:

$$Q_{right}(\theta) = Q - Q_{left}(\theta) \tag{3}$$

The next step is to calculate the purity of node V. The purity of a node in a regression problem is measured using an impurity function, which calculates the Mean Squared Error (MSE) at the corresponding node. Thus, the impurity function can be formulated using the following equation:

$$H(X_V) = \frac{1}{N_V} \sum_{i=1}^{N_V} (y_i - \bar{y}_V)^2 \tag{4}$$

with

$$\bar{y}_V = \frac{1}{N_V} \sum_{i=1}^{N_V} y_i \tag{5}$$

where y_i is the i-th response (target) variable, X_V denotes the features of the training data at node V, N_V is the sample size at node V, and $\bar{y}_V$ is the mean value of the response (target) variable at node V.

After obtaining the impurity of node V, the next step is to calculate the information gain of node V using the following equation:

$$G(Q,\theta) = \frac{n_{left}}{N_V} H\big(Q_{left}(\theta)\big) + \frac{n_{right}}{N_V} H\left(Q_{right}(\theta)\right) \quad (6)$$

where n_{left} denotes the sample size of the left child node, n_{right} denotes the sample size of the right child node, N_V is the number of observations at node V, $H\big(Q_{left}(\theta)\big)$ and $H\big(Q_{right}(\theta)\big)$ are the impurity functions for the left and right child nodes, respectively. The feature and threshold value selected to split the data at node V are those that minimize the impurity. This can be formulated as an optimization problem and expressed using the following equation:

$$\theta^* = min_\theta G(Q,\theta) \quad (7)$$

The process of selecting features and threshold values is performed recursively for the subsets $H\big(Q_{left}(\theta^*)\big)$ and $H\big(Q_{right}(\theta^*)\big)$. This process continues until the stopping criteria are met. Common stopping criteria include reaching the maximum tree depth or when the sample size at node V, N_V, is reached.

Let the terminal node be represented by r, and let $y_{r_1}, y_{r_2}, \dots, y_{r_Z}$ represent the target values of the training data at terminal node r, where Z is the number of target variables at terminal node r. The predicted value of the response (target) variable in a regression problem $\hat{h}(\boldsymbol{x})$ is then obtained by calculating the mean, using the following formulation [13]:

$$\hat{h}(\boldsymbol{x}) = \bar{y}_r = \frac{1}{Z}\sum_{z=1}^{Z} y_{r_z} \quad (8)$$

2.4. Random Forest for Regression Problems

One of the machine learning models that has been widely used in research is Random Forest. In machine learning, Random Forest is classified as an ensemble model. Ensemble models train multiple models to solve the same problem. Unlike conventional learning approaches that attempt to build a single model from the training data, ensemble methods aim to construct a set of models and combine them. Ensemble methods combine multiple learning methods, known as base learners. Base learners are typically constructed from the training data using basic learning algorithms such as decision trees, neural networks, or other learning methods. In general, ensemble methods have stronger predictive capabilities compared to individual base learners. They can transform weak learners into strong learners, thereby producing more accurate predictions [14].

Random Forest is an ensemble learning model that applies the bagging approach, which involves building multiple base models independently, and the final prediction is obtained by taking the average (for regression) or voting (for classification) of the individual base models. Random Forest adopts the Classification and Regression Tree (CART) algorithm as its base learning model. These base models are then combined using the concepts of bootstrap and aggregation. The input dataset is divided into multiple bootstraps, and a CART model is constructed from each bootstrap sample. Suppose the initial training data consist of 4 observations (Observation 1, 2, 3, 4) and 4 features X_1, X_2, X_3, X_4) as well as a response variable Y for each observation. The number of decision trees to be constructed is B. The sampling of observations for constructing bootstrap samples is performed randomly with replacement. Table 2 shows the Random Forest algorithm.

Table 2. Random Forest Algorithm

Random Forest Algorithm for Regression Problems [13]
Given the training data denoted as $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ with $x_i = (x_{i1}, x_{i2}, \dots, x_{ip})^T$ 1. Start with all observations $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$ in a single node (root node). 2. Repeat the following steps recursively for each node that has not been split until the stopping criteria are met: a. Identify the optimal binary split from all possible binary splits across all p predictors b. Divide the node into two child nodes based on the optimal split identified in step 2a. 3. To predict an observation with feature $\boldsymbol{x}$, traverse the tree with $\boldsymbol{x}$ until reaching a terminal node. Let the terminal node be denoted by r, and let the response values of the training data at node by $y_{r_1}, y_{r_2}, \dots, y_{r_Z}$. The predicted value of the response variable in regression is determined by: $$\hat{h}(\boldsymbol{x}) = \bar{y}_r = \frac{1}{Z}\sum_{z=1}^{Z} y_{r_z}, z = 1,2, \dots, Z$$ where Z represents the number of response (target) values at the terminal node r.

The following figure illustrates the process within the Random Forest model.

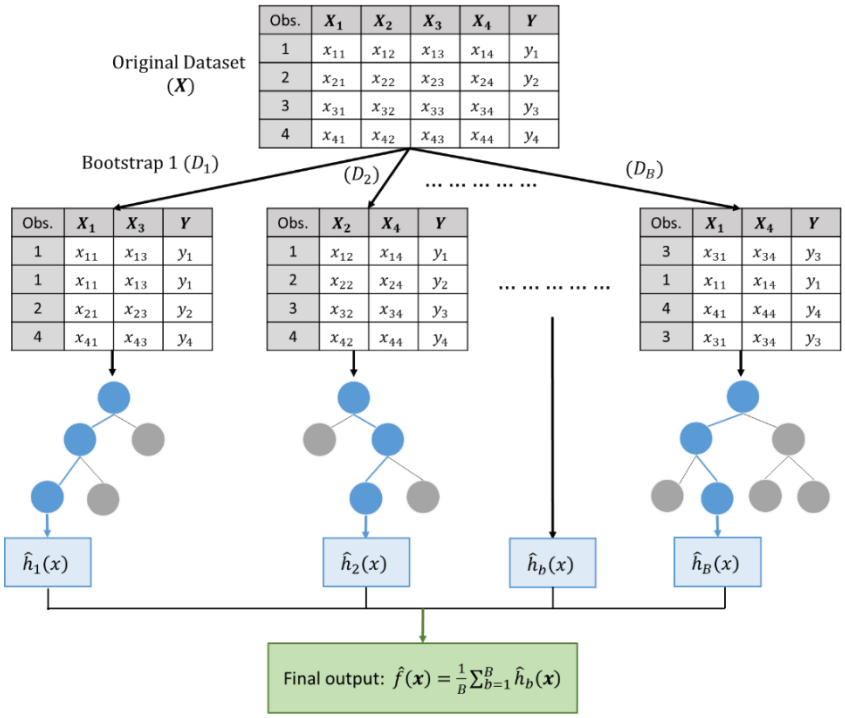

Figure 1. Illustration of the Random Forest Algorithm for Regression with B trees [15], modified

2.5 Extreme Gradient Boosting (XGBoost)

XGBoost is an ensemble model that uses decision trees as its base learners. Unlike Random Forest, which applies a bagging approach, XGBoost employs a boosting strategy in which base models are built sequentially, and each subsequent model depends on the performance of the previous one. XGBoost gained significant popularity in 2016 [16]. In their study, Chen and Guestrin introduced a tree-boosting system designed to be scalable, fast, and computationally efficient for handling large and complex datasets. For a given training dataset $D = \{(x_i, y_i\}_{i=1}^{n}$ consisting of n samples and m features, an XGBoost model can be constructed

from a set of K CART base learners combined using an additive approach. The resulting model can be expressed by the following equation [17, 18].

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), f_k \in F \tag{9}$$

where $f_k(x_i)$ represents the predicted value obtained when the i-th sample is passed through the k-th tree. The value $\hat{y}_i$ denotes the final prediction, and Frepresents the space of all possible regression trees. The objective function is defined as follows:

$$L = \sum_i l(\hat{y}_i, y_i) + \sum_i \Omega(f_k) \tag{10}$$

where l denotes the loss function, and Ω represent the regularization function used to prevent overfitting. The formulation of the regularization term Ω is expressed as follows:

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda \|w\|^2 \tag{11}$$

where T represents the number of leaves in each CART tree, w denotes the weight assigned to each leaf, γ and λ are parameters that control model complexity. Furthermore, the information gain derived from the objective function after each split is calculated using the following formulation:

$$Gain = \frac{1}{2}\left[\frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L}(h_i + \lambda)} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_L}(h_i + \lambda)} + \frac{(\sum_{i\in I} g_i)^2}{\sum_{i\in I}(h_i + \lambda)}\right] - \gamma \tag{12}$$

Figure 2 illustrates the process of the XGBoost model. Suppose an initial training dataset $D = \{(x_i, y_i\}_{i=1}^n$ is given, consisting of n samples, m features, and a response variable Y for each observation. A total of K decision trees are constructed in the model.

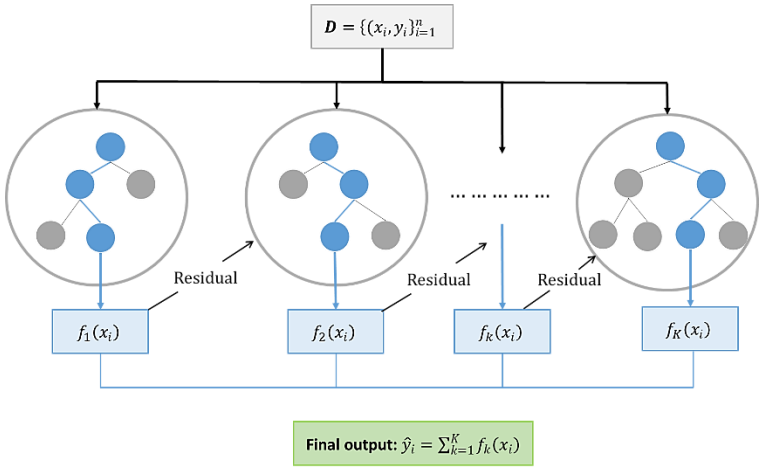

Figure 2. Illustration of the XGBoost Algorithm with K trees [19], modified

3. Results and Discussion

Before running the machine learning simulations, it is important to examined the correlation between the indicators (features) and the target variable (stunting prevalence). Based on the Table 3 below, the correlation values between the features and the target were relatively low. This finding indicated that the linear relationships between the features and stunting prevalence were generally weak. The highest correlation was observed for the caregiving class feature. However, since the correlation coefficient was negative, the relationship between the caregiving class variable and stunting prevalence was inverse. This implies that higher coverage of caregiving classes was associated with lower stunting prevalence. Therefore, strengthening

interventions related to caregiving practices had the greatest potential contribution to reducing stunting prevalence.

Table 3. Correlation Between Features and Stunting Prevalence

Feature	Correlation
Caregiving Class	-0.298608
Health Insurance for Children Under Two from Poor Families	-0.123382
Health Insurance for Pregnant Women from Poor Families	-0.081180
Birth Certificate for Infants from Poor Families	-0.075692
Toddler Mothers Class	-0.052829
Posyandu Activities	0.001935
Pregnant Women Class	0.045736
Access to Safe Drinking Water	0.055174
Supplementary Feeding for Pregnant Women with Chronic Energy Deficiency/High Risk from Poor Families	0.063084
Access to Sanitary Latrines	0.079028
Utilization of Family and Village Yard Land	0.080664

The simulation process began with data preprocessing to determine the numerical features used in the model learning stage. Subsequently, three tree-based machine learning models were implemented: Classification and Regression Tree (CART), Random Forest (RF), and Extreme Gradient Boosting (XGBoost). For each model, optimal hyperparameters were identified using GridSearchCV to obtain the best parameter combination within the predefined search space. In addition to hyperparameter tuning and model fitting, performance evaluation was validated using cross-validation to obtain more stable performance estimates and to reduce potential bias arising from the initial data-splitting scheme. Analysis of the importance of features was conducted to rank features based on their contribution to predicting the target variable. Feature selection was then applied using the top-ranked features with proportions of 25%, 50%, 75%, and 100%. For each feature subset, the models were retrained with hyperparameter optimization as previously performed. Model performance was evaluated using the mean and standard deviation (Std) of the Root Mean Square Error (RMSE). The mean RMSE measured overall predictive accuracy, while the standard deviation assessed model stability.

Based on the simulation results, the mean RMSE values for each model were obtained using 25% (3 features), 50% (6 features), 75% (9 features), and 100% (11 features) of the ranked features. Table 4 shows that the performance of the three models varies according to the proportion of selected features used in model training. Based on the mean RMSE values, it can be observed that XGBoost achieved the best performance in predicting stunting prevalence, as indicated by the lowest RMSE value among the three models. This finding suggests that XGBoost produces the smallest prediction error compared to CART and Random Forest. The best performance of XGBoost was obtained when 50% of the top-ranked features were used, resulting in a mean RMSE of 5.3820. This indicates that the feature selection process contributed to improving model performance compared to using 100% of the features. Feature selection reduces irrelevant or less informative predictors, thereby decreasing variance and minimizing the risk of overfitting.

In contrast, the CART model achieved its lowest mean RMSE of 6.1008 when using 50% of the selected features. However, this value remains higher than the lowest RMSE values obtained by Random Forest and XGBoost. This indicates that CART has the lowest predictive performance among the three models. The relatively higher RMSE value reflects a greater level of prediction error. Meanwhile, the Random Forest model demonstrated improved performance compared to CART, with its lowest mean RMSE of 5.6774 achieved when 75% of the selected features were used. The reduction in RMSE indicates improved predictive accuracy. This

improvement can be attributed to the ensemble mechanism of Random Forest, which aggregates predictions from multiple decision trees to produce a more robust and stable model compared to a single CART model. Overall, XGBoost outperformed both CART and Random Forest, achieving the lowest mean RMSE of 5.3820 with 50% of the selected features. This result is consistent with the boosting principle underlying XGBoost, where sequential error correction enables the model to better capture complex data patterns.

Table 4. Model Performance Based on Mean RMSE

Model	Mean RMSE with p Features			
	$p = 25\%$	$p = 50\%$	$p = 75\%$	$p = 100\%$
CART	6.2434	6.1008	6.1499	6.6191
Random Forest	5.7860	5.6847	5.6774	5.7275
XGBoost	5.5826	5.3820	5.5319	5.6341

Figure 3 shows the comparison of mean RMSE values for each model across different feature proportions.

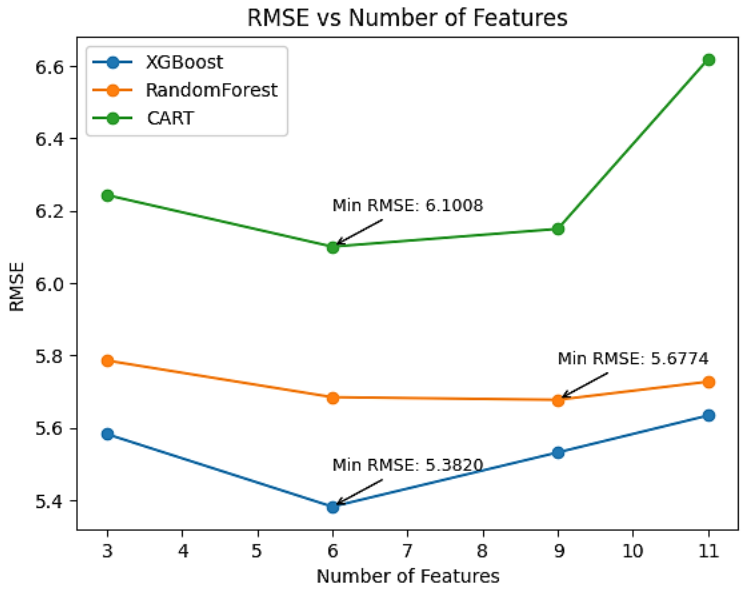

Figure 3. Mean RMSE of Models with p Selected Features.

Based on the standard deviation values of RMSE presented in Table 5, XGBoost achieved the lowest standard deviation of 0.0560 when using 100% of the selected features. Random Forest obtained its lowest standard deviation of 0.1554 with 50% of the features, while CART achieved its lowest value of 0.1456 with 75% of the features. Overall, XGBoost demonstrates the lowest standard deviation among the three models. Furthermore, it exhibits a clear pattern in which increasing the proportion of selected features results in a lower standard deviation, indicating improved model stability. These findings suggest that XGBoost is the most stable model in this study, followed by Random Forest, whereas CART shows the lowest stability.

Table 5. Standard Deviation of RMSE with p Selected Features

Model	Standard Deviation of RMSE with p Features			
	$p = 25\%$	$p = 50\%$	$p = 75\%$	$p = 100\%$
CART	0.7367	0.9900	0.1456	0.6732
Random Forest	0.2025	0.1554	0.2633	0.1750
XGBoost	0.2665	0.1251	0.0810	0.0560

This result also supports the notion that ensemble-based models tend to provide more stable predictive performance compared to a single decision tree model such as CART. Figure 4 shows the comparison of the RMSE standard deviation for each model across different feature proportions.

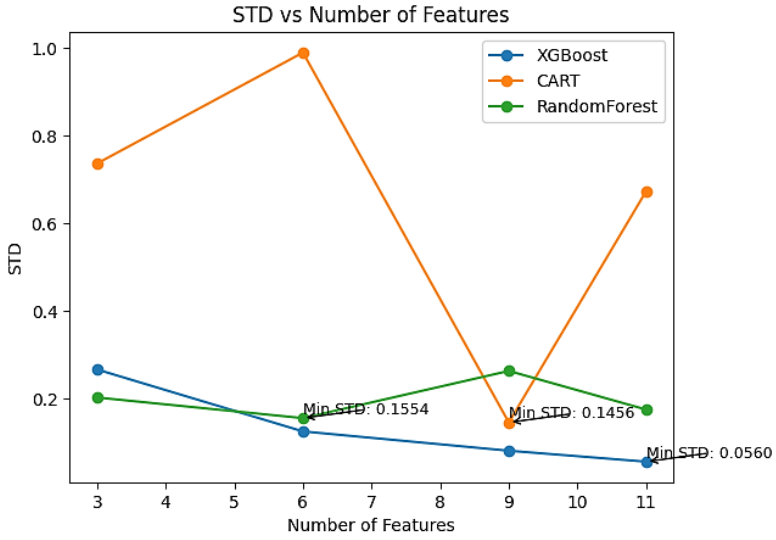

Figure 4. Standard Deviation of RMSE Across Different Feature Proportions.

Furthermore, Table 6 presents the names and proportions of features used by each model to achieve their best performance. Based on Table 6, it can be observed that CART and XGBoost achieved their optimal performance using 50% of the selected features, while Random Forest performed best when using 75% of the features. This finding indicates that CART and XGBoost tend to rely more heavily on the most influential features in predicting stunting prevalence. Notably, four features were consistently selected across all models at their optimal performance levels: Posyandu Activities, Toddler Mothers Class, Caregiving Class, and Utilization of Family and Village Yard Land. The consistent selection of these features suggests that they serve as key indicators in determining stunting prevalence. Specifically, the most influential social factors contributing to stunting prevalence in Java are Posyandu Activities, Toddler Mothers Class, and Caregiving Class. Meanwhile, the most influential environmental factor is Utilization of Family and Village Yard Land. These results highlight the importance of strengthening community-based health services and family empowerment programs, as well as optimizing the utilization of household and village environmental resources, to support stunting reduction efforts.

Table 6. Selected Features Corresponding to the Best Model Performance

Model	Feature Proportion	Selected Features
CART	50%	- Posyandu Activities - Pregnant Women Class - Toddler Mothers Class - Access to Safe Drinking Water - Caregiving Class - Utilization of Family and Village Yard Land
Random Forest	75%	- Posyandu Activities - Toddler Mothers Class - Supplementary Feeding for Pregnant Women with Chronic Energy Deficiency/High Risk from Poor Families - Access to Safe Drinking Water

		- Access to Sanitary Latrines - Health Insurance for Pregnant Women from Poor Families - Birth Certificate for Infants from Poor Families - Caregiving Class - Utilization of Family and Village Yard Land
XGBoost	50%	- Posyandu Activities - Toddler Mothers Class - Access to Sanitary Latrines - Health Insurance for Pregnant Women from Poor Families - Caregiving Class - Utilization of Family and Village Yard Land

4. Conclusion

Based on the simulation and analysis conducted, the correlation between each feature and the target variable was found to be relatively low. The highest correlation was observed for the Caregiving Class variable (−0.298608), indicating a negative relationship with stunting prevalence. This suggests that optimizing Caregiving Class activities has the strongest potential to reduce stunting prevalence among the observed variables. Although the linear correlations within the dataset are weak, tree-based machine learning approaches were able to achieve relatively good predictive performance. Among the three decision tree–based models evaluated, XGBoost demonstrated the best performance, achieving the lowest mean RMSE of 5.3820 when using 50% of the selected features. Random Forest ranked second, with a mean RMSE of 5.6774 using 75% of the features, while CART showed the lowest predictive performance, with a mean RMSE of 6.1008 using 50% of the features. In terms of stability, as reflected by the standard deviation of RMSE, XGBoost also exhibited the lowest variability, indicating that it is the most stable and consistent predictive model for stunting prevalence in this study. Furthermore, feature selection based on feature importance proved effective in improving model performance compared to using all available features. Four features were consistently selected across the best-performing configurations of all three models: Posyandu Activities, Toddler Mothers Class, Caregiving Class, and Utilization of Family and Village Yard Land. Among these, three belong to the social factor group (Posyandu Activities, Toddler Mothers Class, and Caregiving Class), while one represents an environmental factor (Utilization of Family and Village Yard Land). These findings indicate that strengthening parenting programs, community-based health education services, and the effective utilization of family and village environmental resources may serve as strategic priorities in efforts to reduce stunting prevalence in Java.

Acknowledgment

The data used in this study were obtained from publicly available sources provided by Badan Pusat Statistik (BPS) Provinsi Jawa Timur. The authors acknowledge and appreciate the availability of these data.

References

[1] World Health Organization, "*Stunting in a nutshell*", WHO, 2015.

[2] B. A. Takele, L. D. Gezie, and T. S. Alamneh, "Pooled prevalence of stunting and associated factors among children aged 6–59 months in Sub-Saharan Africa countries: A Bayesian multilevel approach," *PLOS ONE*, vol. 17, no. 10, Art. no. e0275889, 2022.

[3] S. W. P. R. Indonesia and K. K. B. P. Manusia, *Strategi Nasional Percepatan Pencegahan Anak Kerdil (Stunting) Periode 2018–2024*, 2018.

[4] UNICEF East Asia and Pacific Region, *Southeast Asia Regional Report on Maternal Nutrition and Complementary Feeding*, 2021.

[5] World Bank, *Water Supply and Sanitation in Indonesia: Turning Finance into Service for the Future*. Washington, DC, USA: World Bank, 2015.

[6] W. R. Mgomezulu, P. Thangata, B. Mkandawire, and N. Amoah, "Advancing predictive analytics in child malnutrition: Machine, ensemble and deep learning models with balanced class distribution for early detection of stunting and wasting," *Human Nutrition & Metabolism*, 2025, Art. no. 200340.

[7] P. K. Arya, K. Sur, T. Kundu, S. Dhote, and S. K. Singh, "Unveiling predictive factors for household-level stunting in India: A machine learning approach using NFHS-5 and satellite-driven data," *Nutrition*, vol. 132, Apr. 2025, Art. no. 112674.

[8] A. Wicaksono, D. Prasetyo, Y. Mar'atullatifah, D. U. Iswavigra, H. Mahmudah, and A. Hapsari, "Data analysis and explainable machine learning for stunting prediction," *Journal of Artificial Intelligence and Legal Technology*, vol. 1, no. 1, pp. 35–44, 2025.

[9] T. O. Hodson, "Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not," *Geoscientific Model Development Discussions*, pp. 1–10, 2022.

[10] D. Chicco, M. J. Warrens, and G. Jurman, "The coefficient of determination R-squared is more informative than SMAPE, MAE, MAPE, MSE and RMSE in regression analysis evaluation," *PeerJ Computer Science*, vol. 7, Art. no. e623, 2021.

[11] Badan Pusat Statistik (BPS) Provinsi Jawa Timur, *Statistik Potensi Desa Provinsi Jawa Timur 2024. Surabaya*, Indonesia, 2024.

[12] G. Biau, "Analysis of a random forests model," *The Journal of Machine Learning Research*, vol. 13, pp. 1063–1095, 2012.

[13] A. Cutler, D. R. Cutler, and J. R. Stevens, "Random Forests," in *Ensemble Machine Learning: Methods and Applications*, C. Zhang and Y. Ma, Eds. Boston, MA, USA: Springer, 2012, pp. 157–175.

[14] Z.-H. Zhou, *Ensemble Methods: Foundations and Algorithms*. Boca Raton, FL, USA: CRC Press, 2012.

[15] M. Usta and A. Tosyali, "Characterization of model-based uncertainties in incompressible turbulent flows by machine learning," in *Proc. ASME Int. Mech. Eng. Congr. Expo. (IMECE)*, vol. 52101, Paper No. V007T09A029, Nov. 2018.

[16] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, pp. 785–794, Aug. 2016.

[17] P. Zhang, Y. Jia, and Y. Shang, "Research and application of XGBoost in imbalanced data," *International Journal of Distributed Sensor Networks*, vol. 18, no. 6, Art. no. 15501329221106935, 2022.

[18] W. Li, Y. Yin, X. Quan, and H. Zhang, "Gene expression value prediction based on XGBoost algorithm," *Frontiers in Genetics*, vol. 10, Art. no. 1077, 2019.

[19] M. Niazkar, A. Menapace, B. Brentan, R. Piraei, D. Jimenez, P. Dhawan, and M. Righetti, "Applications of XGBoost in water resources engineering: A systematic literature review (Dec. 2018–May 2023)," *Environmental Modelling & Software*, vol. 174, Art. no. 105971, 2024.