

Explainable Machine Learning for Current Stunting Status Classification in Toddlers: A Comparative Evaluation of SMOTEENN and SHAP-Based Interpretation

Siti Nurhasanah Nugraha^{1*}, Muji Ernawati², Evita Fitri³, and Tri Wahyu Lestari⁴

^{1,2,3,4} Universitas Nusa Mandiri, Indonesia

MEDINFTEch is licensed under a Creative Commons 4.0 International License.



ARTICLE HISTORY

Received: 16 August 26
 Final Revision: 16 September 26
 Accepted: 20 September 26
 Online Publication: 30 September 26

KEYWORDS

Explainable Machine Learning, Machine Learning, SHAP, SMOTEENN, Stunting Classification

CORRESPONDING AUTHOR

siti.nhg@nusamandiri.ac.id

DOI

10.37034/medinftech.v4i3.170

ABSTRACT

Stunting remains a major nutritional problem that requires accurate and reliable assessment of children’s current growth status. Machine learning can support stunting classification; however, class imbalance and limited model interpretability remain important challenges. This study aims to evaluate the effect of SMOTEENN across classification algorithms with different learning characteristics and interpret the best-performing model using SHAP. Two secondary datasets were integrated, resulting in 41,176 records, of which 41,173 were retained after data cleaning and numerical conversion. Logistic Regression, Decision Tree, Random Forest, and XGBoost were evaluated using original and SMOTEENN-balanced training data. Hyperparameter optimization was performed using RandomizedSearchCV with stratified five-fold cross-validation and ROC-AUC as the optimization metric. The optimized configurations were fixed and applied to both experimental scenarios. Model performance was evaluated using Accuracy, Precision, Recall, F1-Score, and ROC-AUC on an independent test set. Random Forest trained on the original data achieved the best overall performance, with an Accuracy of 0.9600, Precision of 0.9584, F1-Score of 0.9505, and ROC-AUC of 0.9927. SMOTEENN consistently improved Recall across all models but generally reduced the overall performance of tree-based models. The largest improvement occurred in Logistic Regression, with Recall increasing from 0.7058 to 0.8657. SHAP analysis identified Height and Age (Month) as the most influential features. These findings show that the effect of SMOTEENN depends on dataset and algorithm characteristics, while SHAP provides transparent interpretation of current stunting-status classification within the analyzed datasets.

1. Introduction

Stunting remains one of the major chronic nutritional problems affecting child growth and development. It is characterized by impaired linear growth, in which a child's height is not appropriate for their age as a consequence of prolonged nutritional deficiencies and other health-related factors. The consequences of stunting extend beyond physical growth and may affect cognitive development, learning ability, productivity, and quality of life later in life [1], [2]. Therefore, stunting prevention and accurate assessment of

children’s current growth status are important components of efforts to improve children’s health and human capital.

According to the 2024 Indonesian Nutritional Status Survey (Survei Status Gizi Indonesia/SSGI), the national prevalence of stunting among children under five reached 19.8%. Although this figure represents an improvement compared with previous years, stunting remains a public health concern and requires sustained intervention to achieve national development targets [3]. Addressing stunting requires not only nutritional

and health interventions but also effective monitoring systems capable of identifying children's growth conditions in a timely and consistent manner. In primary healthcare services, particularly at integrated health posts (posyandu), children's growth is commonly monitored through measurements such as age, weight, and height. These measurements are subsequently recorded and compared with child growth standards to determine nutritional or growth status. However, conventional monitoring and assessment processes may require considerable time and may result in inconsistencies when a large number of children are assessed [5]. Therefore, computational approaches may provide additional support for processing growth data more efficiently and consistently without replacing clinical assessment by healthcare professionals.

The development of machine learning provides opportunities to analyze anthropometric and demographic data for stunting classification. Machine learning algorithms can learn patterns and relationships among variables from historical data and subsequently classify new observations. Previous studies have demonstrated the potential of algorithms such as Support Vector Machine, Random Forest, XGBoost, and other classification approaches for stunting prediction and classification [6], [7].

Several studies have reported promising results from machine learning-based approaches to stunting classification. A previous study using Random Forest optimized with a genetic algorithm achieved an accuracy of 89.58%; however, class imbalance was not specifically addressed [8]. Another study compared Random Forest, AdaBoost, XGBoost, and CatBoost for stunting prediction and applied the Synthetic Minority Over-sampling Technique (SMOTE), with Random Forest achieving an accuracy of 97.98% [9]. Similarly, a comparative study involving XGBoost, Random Forest, Support Vector Machine, and k-Nearest Neighbours reported that the application of SMOTE improved classification performance, with XGBoost achieving an accuracy of 85.74% [10]. Other studies have also applied machine learning, ensemble learning, and deep learning approaches to stunting and wasting detection while addressing class distribution issues and achieving promising classification performance [11], [12], [13].

Despite these promising results, class imbalance may affect the performance of classification models when one class is represented more frequently than another. Under such conditions, a model may become more inclined toward the majority class, potentially reducing its ability to identify observations from the less represented class. This issue is particularly important in health-related classification, where the ability to correctly identify positive cases may be as important as overall classification accuracy. SMOTE is commonly

used to increase minority-class representation by generating synthetic observations. However, the application of SMOTE alone may introduce overlapping observations or amplify noisy patterns in certain datasets [14]. To address these limitations, the Synthetic Minority Over-sampling Technique combined with Edited Nearest Neighbours (SMOTEENN) integrates minority-class oversampling with a data-cleaning process based on Edited Nearest Neighbours (ENN). This approach is intended to modify the class distribution while removing observations that may contribute to local class ambiguity.

However, the application of a resampling technique does not necessarily guarantee improved performance across all classification algorithms and evaluation metrics. The effect of data balancing may vary according to the characteristics of the dataset and the learning mechanism of the classifier. A resampling strategy that improves the ability to identify minority-class observations may also influence Precision, Accuracy, F1-Score, or the overall discriminative ability of a model. Therefore, evaluating the effectiveness of a data-balancing technique requires not only determining whether overall performance improves, but also examining potential trade-offs across different classification algorithms and evaluation metrics.

Another challenge in applying machine learning to health-related classification is the limited interpretability of complex predictive models. Models with high predictive performance may operate as black boxes, making it difficult to understand how individual features contribute to their decisions. In healthcare-related applications, predictive performance alone may be insufficient because the basis of a model's decision should also be understandable and transparent. Explainable Machine Learning (XML) provides an approach for improving model interpretability by identifying and explaining the contribution of individual features to model predictions [15]. One widely used explainability method is SHapley Additive exPlanations (SHAP), which can provide both global and local explanations. Global explanations identify features that consistently influence model predictions across the dataset, whereas local explanations describe how individual feature values contribute to a specific prediction [16].

Previous studies have incorporated data balancing, ensemble learning, and explainability into stunting-related classification. A previous study combined SMOTEENN, ensemble learning, and SHAP for stunting risk prediction and reported an accuracy of 91.95% [12]. However, the study focused on a hybrid soft-voting model and did not comprehensively compare algorithms with different learning characteristics. Another study compared Logistic

Regression, k-Nearest Neighbours, Support Vector Classifier, and Decision Tree for stunting risk prediction, although the interpretation of the resulting classification decisions remained limited [13].

Based on the existing literature, several research gaps can be identified [9], [10], [11], [12], [13]. First, comparative evaluation of classification models using both the original training data and SMOTEENN-balanced training data remains limited. As a result, the effect of SMOTEENN cannot always be clearly distinguished from other experimental factors. Second, the influence of SMOTEENN on algorithms with different learning characteristics, including linear models, single decision trees, and ensemble-based models, requires further investigation. Third, previous studies have primarily emphasized improvements in predictive performance, whereas potential trade-offs among Accuracy, Precision, Recall, F1-Score, and ROC-AUC after data balancing require further examination. Finally, although explainability has been incorporated into some stunting-related studies, the combination of comparative performance evaluation and both global and local model interpretation remains relatively limited.

To address these gaps, this study applies a comparative explainable machine-learning approach to current stunting-status classification using anthropometric and demographic data. Logistic Regression, Decision Tree, Random Forest, and XGBoost were evaluated under two training scenarios: original and SMOTEENN-resampled data. The novel contribution lies in the controlled comparison of algorithms with different learning characteristics using the same fixed hyperparameter configurations and independent test set. This design enables performance changes associated with SMOTEENN to be assessed without the influence of different model configurations.

The study also examines metric-specific trade-offs across Accuracy, Precision, Recall, F1-Score, and ROC-AUC rather than assuming that resampling uniformly improves model performance. The best-performing model is subsequently interpreted using global and local SHAP analyses. Accordingly, this study provides a comparative empirical evaluation combining controlled performance comparison, metric-specific analysis, and model interpretation rather than proposing a new resampling method, explainability method, or technical framework.

This study aims to evaluate the effect of SMOTEENN across classification algorithms with different learning characteristics, identify the best-performing model under the original and SMOTEENN-resampled training scenarios, and interpret its classifications using SHAP. The findings are limited to the classification of current stunting status within the analyzed datasets. Because the predictors and target status were measured concurrently, the models should not be interpreted as

predicting future stunting risk or providing prospective early screening.

2. Research Method

This study employs an Explainable Machine Learning (XML) approach to evaluate models for classifying current stunting status based on anthropometric and demographic characteristics. The study uses integrated secondary datasets containing demographic and anthropometric information. In addition to developing and evaluating classification models, SHAP (Shapley Additive Explanations) is employed to identify and interpret the contribution of each predictor to the model's classification results. This approach is intended to provide transparent information about how the models classify current stunting status within the analyzed datasets. It is not intended to predict future stunting risk or replace assessments performed by qualified healthcare professionals.

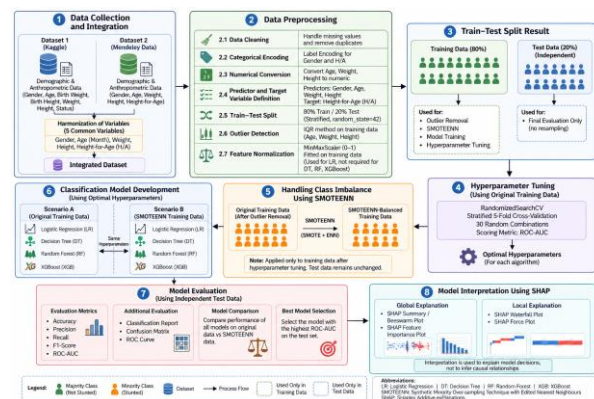


Figure 1. Research Stages

As illustrated in Figure 1, the research process consists of data collection and integration, data preprocessing, train-test splitting, hyperparameter tuning using the original training data, SMOTEENN application, classification model training, model evaluation, and model interpretation using SHAP. The process begins with collecting and integrating two secondary datasets based on common variables. The integrated data are then subjected to data cleaning, numerical conversion, categorical encoding, feature selection, train-test splitting, outlier detection, and feature normalization. Hyperparameter tuning is first performed using the original training data to determine the optimal configuration for each classification algorithm. The obtained configurations are then fixed and applied to both the original and SMOTEENN-balanced training data. The resulting models are evaluated using an independent test set, and the best-performing model is subsequently interpreted using SHAP.

2.1. Data Collection and Integration

This study used two secondary datasets comprising 41,176 observations. The Kaggle dataset (<https://www.kaggle.com/datasets/risants/data-mentah->

[data-mining](#)) 1,105 children aged 1–59 months identified in the dataset as originating from Tasinifu, including 553 Not Stunted and 552 Stunted observations. However, the available Kaggle documentation did not report the original collecting institution, specific study location, collection period, sampling method, measurement and quality-control procedures, or whether the observations were real or synthetic; therefore, the provenance and clinical validity of this dataset could not be independently verified.

The Mendeley dataset contained 40,071 children aged 0–60 months from Jeneponto Regency, South Sulawesi, Indonesia, including 23,867 Not Stunted and 16,204 Stunted observations. The Mendeley data were collected between 2021 and 2024 [17]. Both datasets recorded Age in months, Weight in kilograms, and Height in centimetres. Their sample characteristics and descriptive statistics before integration are presented in Table 1.

Table 1. Characteristics of the Datasets Before Integration

Characteristic	Kaggle	Mendeley
Initial sample, n	1,105	40,071
Not Stunted, n (%)	553 (50.05%)	23,867 (59.56%)
Stunted, n (%)	552 (49.95%)	16,204 (40.44%)
Age range (months)	1–59	0–60
Age, mean $\pm$ SD (months)	31.74 $\pm$ 17.23	33.02 $\pm$ 15.82
Weight, mean $\pm$ SD (kg)	10.64 $\pm$ 2.90	11.66 $\pm$ 29.51
Height, mean $\pm$ SD (cm)	82.89 $\pm$ 13.29	86.05 $\pm$ 11.82
Retained after harmonization, n	1,105	40,071

The two datasets showed broadly comparable Age and Height distributions and used consistent measurement units. Greater variation was observed in Weight in the Mendeley dataset because the source data contained an extreme recorded value. Potential outliers were subsequently assessed using the IQR method on the training data only.

Five common variables were retained: Gender, Age (Month), Weight, Height, and Height-for-Age. In the Kaggle dataset, JK, Umur(X1), BB Saat Ini (X4), TB Saat Ini (X5), and Status were renamed accordingly. Gender codes P and L were standardized as F and M.

Target-label information differed between the datasets. The Mendeley dataset derived Height-for-Age status from the WHO Child Growth Standards, with a Height-for-Age Z-score (HAZ) below -2 SD classified as Stunted [4]. The Kaggle dataset provided binary labels, with 1 indicating Stunted and 0 indicating Not Stunted, but did not document how these labels were established. Therefore, the Kaggle labels were retained as provided and standardized only at the coding level. Although the datasets had comparable variables, measurement units, and age groups, complete equivalence of their target definitions and measurement procedures could not be verified. Following harmonization, the datasets were concatenated and subjected to data cleaning and preprocessing.

2.2. Data Preprocessing

Data preprocessing was conducted to improve data quality and prepare the integrated dataset for machine learning classification. The preprocessing stages consisted of data cleaning, numerical conversion, categorical encoding, feature selection, dataset splitting, outlier detection, and feature normalization.

2.2.1. Data Cleaning

The data cleaning stage involved checking the completeness and consistency of the integrated dataset. Missing values were identified for the variables used in the analysis, and records containing missing values in the selected variables were removed to prevent incomplete observations from affecting the model development process [18]. Duplicate records were also examined and removed to reduce data redundancy and prevent repeated observations from disproportionately influencing the learning process [19].

2.2.2. Categorical Encoding

Categorical variables were transformed into numerical representations using Label Encoding to enable their processing by the machine learning algorithms [21]. In this study, categorical variables were transformed into numerical representations using Label Encoding before the dataset was divided into training and testing subsets. Label Encoding was applied to the Gender variable and the Height-for-Age target variable.

2.2.3. Numerical Conversion

The numerical attributes Age (Month), Weight, and Height were converted into numerical data types. Values that could not be converted were treated as missing values and subsequently removed. This step was performed to ensure that the anthropometric variables were represented in a numerical format suitable for subsequent data processing and machine learning modeling.

2.2.4. Predictor and Target Variable Definition

Feature selection is the process of selecting the most relevant attributes for use in modeling, thereby reducing data complexity while preserving the information required by the model [22]. Feature selection in this study was conducted using a compatibility-based criterion rather than a formal algorithmic feature-selection method. Variables were retained when they were available in both datasets, had comparable definitions and measurement units, and could be consistently harmonized. Based on these criteria, Gender, Age (Month), Weight, and Height were selected as predictors. The target variable was the current Height-for-Age status, consisting of Stunted and Not Stunted. No statistical or algorithmic feature-ranking procedure was applied.

2.2.5. Train-Test Split

The dataset was split into 80% training data and 20% testing data using the Train-Test Split method, with `random\_state = 42` and stratification based on the target variable. This split was performed to separate the data used for training from the data used to evaluate the model's generalization capability on data not encountered during the training process [23].

The training data was used for outlier removal, handling class imbalance, model training, and hyperparameter tuning. Meanwhile, the test data is retained without any resampling process and is used only for the final evaluation of the model. The test set was kept unchanged throughout the subsequent preprocessing and modeling stages.

2.2.6. Outlier Detection

Outliers in the training data were detected using the Interquartile Range (IQR) method. IQR identifies observations that fall outside the lower and upper bounds of the distribution based on the first and third quartiles [20]. Outlier detection was performed only on the training data after the train-test split to prevent information from the test set from influencing model development. For each numerical predictor, the interquartile range was calculated as $IQR = Q3 - Q1$. Observations below $Q1 - 1.5 \times IQR$ or above $Q3 + 1.5 \times IQR$ were identified as outliers. An observation was removed when at least one of its Age, Weight, or Height values exceeded the corresponding IQR bounds. The calculated thresholds and numbers of outliers are presented in Table 2.

Table 2. IQR Thresholds and Outliers in the Training Data

Variable	Q1	Q3	Lower bound	Upper bound	Outliers
Age (Month)	20.00	47.00	-20.50	87.50	0
Weight (kg)	9.50	13.40	3.65	19.25	166
Height (cm)	78.00	95.00	52.50	120.50	196

2.2.7. Feature Normalization

Feature normalization was performed using MinMaxScaler to transform numerical feature values into a common range between 0 and 1 [24]. Normalization was primarily used for Logistic Regression because the model can be sensitive to differences in feature scales, whereas Decision Tree, Random Forest, and XGBoost do not require feature scaling. MinMaxScaler was fitted exclusively to the original training data after outlier removal. The fitted scaler was used to transform the original training data and the independent test data for Logistic Regression. After SMOTEENN was applied to the unscaled training data, the resulting resampled data were transformed using the same fitted scaler. Decision Tree, Random Forest, and XGBoost were trained using unscaled features because these tree-based models do not require feature normalization.

2.3. Handling Class Imbalance Using SMOTEENN

The distribution of the target classes was examined to identify potential class imbalance between Stunted and Not Stunted observations. Class imbalance may cause a classification model to become biased toward the majority class and consequently reduce its ability to correctly identify the minority class. Therefore, SMOTEENN, which combines the Synthetic Minority Over-sampling Technique (SMOTE) and Edited Nearest Neighbours (ENN), was employed to address class imbalance. SMOTE generates synthetic minority-class observations based on nearest-neighbour relationships, whereas ENN removes observations that are inconsistent with their local neighbourhood. Thus, SMOTEENN combines minority-class oversampling with neighbourhood-based data cleaning [25].

In this study, SMOTEENN was implemented using SMOTEENN(random_state=42). Accordingly, sampling_strategy='auto' was used. The internal SMOTE component used the default k_neighbors=5, while the ENN component used the default n_neighbors=3 and kind_sel='all'. The processing sequence consisted of train-test splitting, outlier removal from the training data, fitting MinMaxScaler to the original training data, applying SMOTEENN to the unscaled training data, and subsequently transforming the SMOTEENN-resampled data using the same fitted scaler for Logistic Regression. SMOTEENN was applied only to the training data, producing 35,457 observations, comprising 17,555 Not Stunted and 17,902 Stunted observations. The independent test set was not resampled and was retained for final evaluation.

Because SMOTEENN was applied to the unscaled training data, its distance-based nearest-neighbour calculations may have been influenced by differences in feature scales.

2.4. Classification Model Development

Four supervised machine learning algorithms were employed to classify stunting status: Logistic Regression (LR), Decision Tree (DT), Random Forest (RF), and Extreme Gradient Boosting (XGBoost).

Logistic Regression was used as a baseline classification model because it is suitable for binary classification and provides a relatively interpretable relationship between predictor variables and the predicted class [26]. Decision Tree was selected because it can model nonlinear relationships and represent classification decisions through a tree-based structure [27]. Random Forest was employed because it combines multiple decision trees and can improve predictive robustness while reducing the risk of overfitting compared with a single decision tree [28]. XGBoost was selected as a gradient boosting algorithm capable of modeling complex nonlinear relationships and providing strong predictive performance in

classification tasks [28]. The classification models were trained under two experimental scenarios: the original training data and the SMOTEENN-balanced training data. For both scenarios, the same hyperparameter configuration selected using the original training data was applied to each algorithm. Therefore, the comparison was conducted under a fixed model configuration to assess performance changes associated with SMOTEENN. This design does not represent the maximum performance achievable by models tuned specifically on the SMOTEENN-resampled data.

2.5. Hyperparameter Tuning

Hyperparameter optimization was performed on the original training data using RandomizedSearchCV with stratified five-fold cross-validation. The folds were shuffled using `random_state=42`, and ROC-AUC was used as the optimization metric. RandomizedSearchCV was configured with `n_iter=30`, `random_state=42`, and `n_jobs=-1`. Up to 30 randomly selected configurations were evaluated for each model to provide broad exploration of the search space while maintaining computational feasibility. Because the Logistic Regression search space contained only 12 possible combinations, all 12 configurations were evaluated, whereas 30 randomly selected configurations were evaluated for Decision Tree, Random Forest, and XGBoost. The configuration with the highest mean cross-validation ROC-AUC was selected, and the corresponding standard deviation across the five folds was also recorded.

The hyperparameter configuration selected using the original training data was subsequently fixed and applied to both experimental scenarios. This design enabled a controlled comparison between the original and SMOTEENN-resampled training data under the same model configuration. However, no separate hyperparameter tuning was performed for the SMOTEENN scenario. Therefore, its results should not be interpreted as the best performance achievable after resampling.

2.6. Model Evaluation

The models trained under the fixed-hyperparameter design were evaluated using an independent test set that was not involved in model training, resampling, or hyperparameter optimization. Model performance was assessed using Accuracy, Precision, Recall, F1-Score, and Area Under the Receiver Operating Characteristic Curve (ROC-AUC) [29]. Accuracy measures the proportion of correctly classified observations, while Precision measures the proportion of predicted positive cases that are actually positive. Recall measures the ability of the model to correctly identify positive cases, and F1-Score represents the harmonic mean of Precision and Recall. ROC-AUC measures the model's ability to discriminate between the Stunted and Not

Stunted classes across different classification thresholds.

In addition to the numerical evaluation metrics, Classification Report, Confusion Matrix, and ROC Curve were used to provide a more comprehensive assessment of classification performance. The models were compared based on their performance on the independent test set. The model achieving the highest ROC-AUC was selected as the final best-performing model for subsequent explainability analysis.

2.7. Model Interpretation Using SHAP (Shapley Additive Explanations)

The final best-performing model was subsequently interpreted using SHAP (Shapley Additive Explanations). SHAP is an explainable artificial intelligence method based on Shapley values that quantifies the contribution of individual features to model predictions [30]. TreeExplainer was used to calculate feature contributions for the positive output class, defined as class 1 (Stunted). SHAP values were calculated using the independent test set, while no separate background dataset was explicitly supplied to the explainer. Accordingly, the explainer used the tree-path-dependent information contained in the fitted model. The resulting SHAP values were interpreted on the model-output scale for the Stunted class.

SHAP analysis was performed at both global and local levels. Global SHAP analysis was conducted on all observations in the independent test set. Feature importance was calculated using the mean absolute SHAP value of each predictor, while a summary plot was used to examine the magnitude and direction of feature contributions.

For the local explanation, a true-positive Stunted observation was selected using a predefined criterion. Specifically, the first observation in the test-set order whose actual and predicted classes were both Stunted was selected. The observation was not selected based on its SHAP magnitude or prediction probability. Its feature contributions were visualized using SHAP Waterfall and Force Plots.

The SHAP analysis was interpreted as an explanation of the model's decision-making process rather than as evidence of causal relationships. Therefore, features identified as highly influential by SHAP were considered features contributing to the model's classification of stunting status, rather than definitive causal risk factors for stunting.

3. Result and Discussion

3.1. Dataset and Preprocessing Results

Before integration, the Kaggle dataset contained 1,105 observations, comprising 553 Not Stunted and 552 Stunted observations, while the Mendeley dataset contained 40,071 observations, comprising 23,867 Not

Stunted and 16,204 Stunted observations. All observations were retained after harmonization, resulting in 41,176 integrated records. The initial inspection found no missing or duplicate records in the selected variables. Therefore, no records were removed during the missing-value handling stage. Duplicate checking also indicated that no duplicate records were identified, resulting in the same 41,176 records after duplicate checking.

During numerical conversion, three Not Stunted observations containing nonconvertible values were removed. Therefore, the final dataset consisted of 41,173 observations, comprising 24,417 Not Stunted and 16,756 Stunted observations.

The cleaned dataset was divided into training and testing sets using an 80:20 stratified split. The resulting dataset consisted of 32,938 training records and 8,235 testing records. The IQR procedure identified no outliers in Age, 166 in Weight, and 196 in Height. Because some observations were identified as outliers in more than one variable, 236 unique observations were removed, reducing the training set from 32,938 to 32,702 observations. The removed observations comprised 140 Not Stunted (59.32%) and 96 Stunted (40.68%) cases. This distribution was comparable to the class distribution before outlier removal, indicating that the procedure did not substantially alter the relative class proportions. The class distribution before and after SMOTEENN is presented in Table 3.

Table 3. Class Distribution Before and After SMOTEENN

Class	Before SMOTEENN	After SMOTEENN
Not Stunted	19,393	17,555
Stunted	13,309	17,902
Total	32,702	35,457

Before SMOTEENN, the training data contained 19,393 Not Stunted and 13,309 Stunted observations, indicating an imbalance between the two classes. After applying SMOTEENN, the training dataset increased to 35,457 observations. The number of Stunted observations increased from 13,309 to 17,902, while the number of Not Stunted observations decreased from 19,393 to 17,555. This result indicates that SMOTEENN modified the class distribution through minority-class oversampling combined with the removal of observations identified by the ENN procedure. Importantly, SMOTEENN was applied only to the training data, while the 8,235 testing observations remained unchanged and were reserved for the final model evaluation.

3.2. Hyperparameter Tuning Results

Hyperparameter tuning was performed using the original training data with stratified five-fold cross-validation and ROC-AUC as the optimization metric. All 12 possible configurations were evaluated for Logistic Regression, while 30 randomly selected configurations were evaluated for Decision Tree, Random Forest, and XGBoost. The mean and standard deviation of the cross-validation ROC-AUC for the selected configurations are presented in Table 4.

Table 4. Hyperparameter Tuning Results

Model	Configurations evaluated	Best CV ROC-AUC	Optimal Hyperparameters
Logistic Regression	12	0.9194	C = 0.01; penalty = L1; solver = liblinear
Decision Tree	30	0.9843	criterion = entropy; max_depth = None; min_samples_split = 10; min_samples_leaf = 8; max_features = None
Random Forest	30	0.9927	n_estimators = 500; max_depth = None; min_samples_split = 5; min_samples_leaf = 4; max_features = log2; bootstrap = True
XGBoost	30	0.9931	n_estimators = 700; learning_rate = 0.1; max_depth = 5; min_child_weight = 5; gamma = 0.3; subsample = 1.0; colsample_bytree = 1.0

XGBoost achieved the highest mean cross-validation ROC-AUC of 0.9931, followed by Random Forest with 0.9927. Decision Tree achieved 0.9843, whereas Logistic Regression achieved 0.9194. The standard deviations indicate the variation in model performance across the five validation folds. These results indicate that the tree-based ensemble algorithms, particularly XGBoost and Random Forest, demonstrated stronger discriminative performance during cross-validation than the linear Logistic Regression model.

The hyperparameters selected using the original training data were fixed and applied to both experimental scenarios. Therefore, the reported differences represent performance changes associated with SMOTEENN under the same model

configuration. Because the SMOTEENN scenario was not tuned separately, these results do not demonstrate the best performance that each algorithm could achieve after resampling.

3.3. Classification Performance Results

The classification models were evaluated on the independent test set under the original and SMOTEENN-resampled training scenarios. The same hyperparameter configuration selected using the original training data was applied to both scenarios. Therefore, the comparison represents the effect associated with SMOTEENN under a fixed model configuration rather than a comparison between separately optimized models. The comparison results are presented in Table 5.

Table 5. Classification Performance Results

Model	Dataset	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Decision Tree	Original	0.9563	0.9566	0.9349	0.9457	0.9861
	SMOTEENN	0.9502	0.9380	0.9397	0.9389	0.9653
Logistic Regression	Original	0.8222	0.8319	0.7058	0.7636	0.9140
	SMOTEENN	0.8461	0.7803	0.8657	0.8208	0.9141
Random Forest	Original	0.9600	0.9584	0.9427	0.9505	0.9927
	SMOTEENN	0.9512	0.9346	0.9463	0.9404	0.9906
XGBoost	Original	0.9597	0.9567	0.9436	0.9501	0.9919
	SMOTEENN	0.9540	0.9404	0.9469	0.9436	0.9905

Note: Bold values indicate the higher performance value between the Original and SMOTEENN training scenarios for each model and evaluation metric.

The results indicate that Random Forest using the original training data achieved the highest overall performance, obtaining an Accuracy of 0.9600, Precision of 0.9584, F1-Score of 0.9505, and ROC-AUC of 0.9927. XGBoost using the original data produced a comparable performance, with an Accuracy of 0.9597, F1-Score of 0.9501, and ROC-AUC of 0.9919.

After applying SMOTEENN, the performance of the tree-based models generally decreased in terms of Accuracy, Precision, F1-Score, and ROC-AUC. For Random Forest, Accuracy decreased from 0.9600 to 0.9512, while ROC-AUC decreased slightly from 0.9927 to 0.9906. Similarly, XGBoost experienced a decrease in Accuracy from 0.9597 to 0.9540 and ROC-AUC from 0.9919 to 0.9905. Decision Tree also showed a decrease in ROC-AUC from 0.9861 to 0.9653.

However, SMOTEENN produced an improvement in Recall for all four algorithms. Recall increased from 0.9349 to 0.9397 for Decision Tree, from 0.7058 to 0.8657 for Logistic Regression, from 0.9427 to 0.9463 for Random Forest, and from 0.9436 to 0.9469 for XGBoost. This indicates that SMOTEENN improved the models' ability to identify the Stunted class, although this improvement was accompanied by reductions in several other performance metrics.

The most substantial change was observed in Logistic Regression. Its Accuracy increased from 0.8222 to 0.8461, while Recall increased considerably from 0.7058 to 0.8657 and F1-Score increased from 0.7636 to 0.8208. Its ROC-AUC, however, remained almost unchanged, increasing only slightly from 0.9140 to 0.9141.

Overall, the results suggest that SMOTEENN did not improve the overall predictive performance of the tree-based models when evaluated using ROC-AUC, Accuracy, Precision, or F1-Score. However, it consistently improved Recall across all algorithms, particularly for Logistic Regression. This finding indicates that the effect of SMOTEENN is not uniformly positive across all evaluation metrics. Instead, its primary contribution in this experiment was improving the detection of the minority Stunted class.

3.4. Effect of SMOTEENN on Classification Performance

To further examine the effect of SMOTEENN, the changes in model performance were calculated by subtracting the performance obtained using the original training data from that obtained using the SMOTEENN-balanced training data. A positive value indicates an improvement after applying SMOTEENN, whereas a negative value indicates a decrease in performance. The results are presented in Table 6.

Table 6. Effect of SMOTEENN

Model	Δ Accuracy	Δ Precision	Δ Recall	Δ F1-Score	Δ ROC-AUC
Random Forest	-0.008865	-0.023863	0.003581	-0.010113	-0.002062
XGBoost	-0.005707	-0.016305	0.003283	-0.006477	-0.001433
Decision Tree	-0.006072	-0.018601	0.004775	-0.006790	-0.020770
Logistic Regression	0.0239220	-0.051610	0.159952	0.0571240	0.0000880

The results demonstrate that the effect of SMOTEENN varied across classification algorithms and evaluation metrics. For the three tree-based models, namely Random Forest, XGBoost, and Decision Tree, SMOTEENN resulted in a decrease in Accuracy, Precision, F1-Score, and ROC-AUC. However, Recall increased for all three models. The increase in Recall was 0.003581 for Random Forest, 0.003283 for XGBoost, and 0.004775 for Decision Tree.

The most substantial effect of SMOTEENN was observed in Logistic Regression. The application of

SMOTEENN increased Accuracy by 0.023922, Recall by 0.159952, F1-Score by 0.057124, and ROC-AUC by 0.000088. However, Precision decreased by 0.051610. The substantial increase in Recall indicates that SMOTEENN considerably improved the ability of Logistic Regression to identify Stunted cases, although this improvement was accompanied by a reduction in Precision.

Overall, the results indicate that SMOTEENN does not consistently improve all classification performance metrics. Its most consistent effect was an improvement

in Recall across all four algorithms. This finding suggests that SMOTEENN primarily contributed to increasing the models' sensitivity toward the Stunted class, while its effect on overall discrimination and other performance measures depended on the classification algorithm.

3.5. Confusion Matrix and ROC Curve Analysis

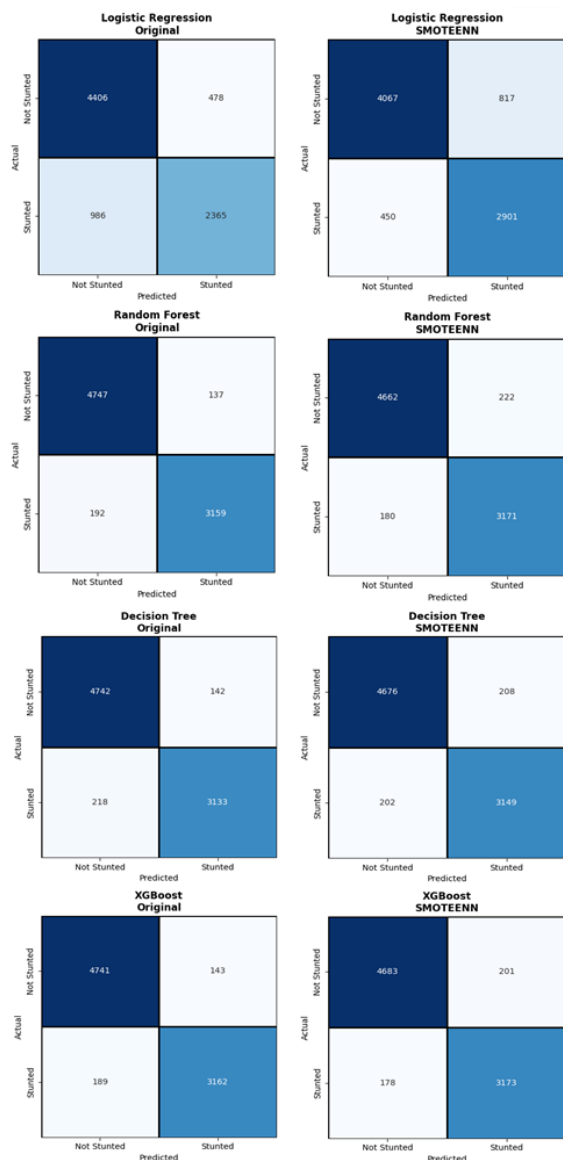


Figure 2. Confusion Matrix Results

Figure 2 shows the confusion matrices of the four classification models using the original and SMOTEENN-balanced training data. Overall, the application of SMOTEENN consistently reduced the number of Stunted instances misclassified as Not Stunted, resulting in improved Recall across all models. The most substantial improvement was observed in Logistic Regression, where the number of correctly classified Stunted instances increased from 2,365 to 2,901.

However, for Random Forest, XGBoost, and Decision Tree, the improvement in detecting Stunted instances was accompanied by an increase in Not Stunted instances incorrectly classified as Stunted. This trade-off contributed to decreases in Precision and other overall performance metrics after applying SMOTEENN. Consistent with the results in Tables 5 and 6, Random Forest using the original data produced the best overall classification performance, while SMOTEENN was particularly beneficial for improving the sensitivity of Logistic Regression toward the Stunted class.

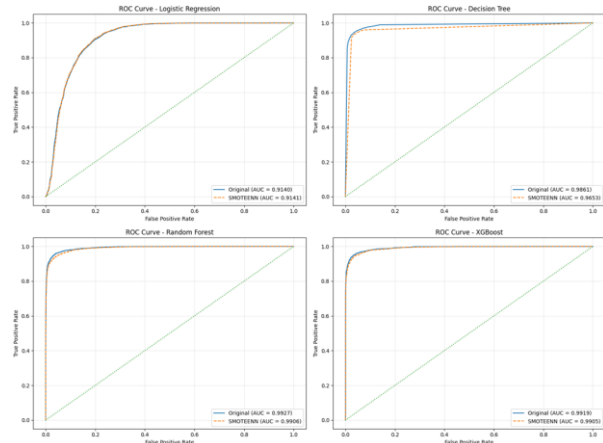


Figure 3. ROC Curve Results

Figure 3 presents the ROC curves of the four classification models using the original and SMOTEENN-balanced training data. Overall, the ROC curves of Random Forest and XGBoost are closer to the upper-left corner, indicating a stronger ability to distinguish between the Stunted and Not Stunted classes. Random Forest using the original data achieved the highest ROC-AUC value of 0.9927, followed closely by XGBoost with an ROC-AUC of 0.9919.

The application of SMOTEENN resulted in only minor changes in the ROC curves of Logistic Regression, Random Forest, and XGBoost, while Decision Tree showed a more noticeable decrease in AUC from 0.9861 to 0.9653. These results are consistent with the classification performance results, indicating that SMOTEENN improved Recall but did not improve the overall discriminative ability of the tree-based models. Overall, Random Forest using the original data demonstrated the best classification performance based on the ROC-AUC results.

3.6. SHAP Global Explanation

Figure 4 presents the SHAP summary plot, which illustrates the distribution and direction of each feature's contribution to the model predictions. The results show that Height and Age (Month) have the widest range of SHAP values, indicating that these features have the strongest influence on the model

output. Meanwhile, Weight and Gender show relatively smaller contributions.

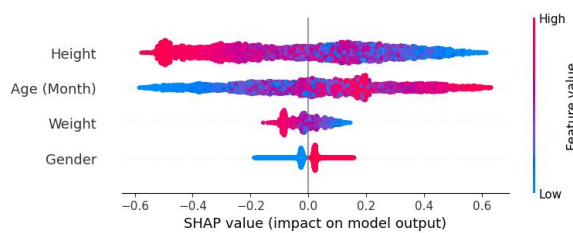


Figure 4. SHAP Summary Plot of Global Feature Contributions

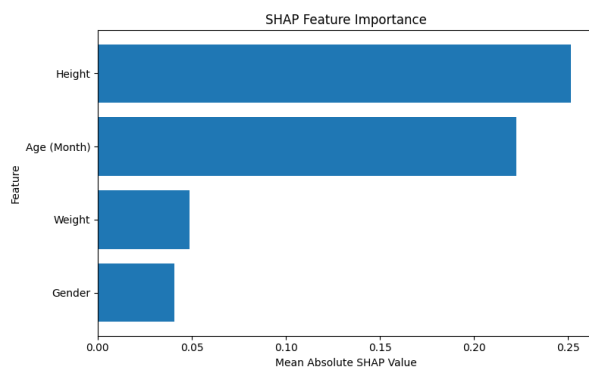


Figure 5. Global Feature Importance Based on Mean Absolute SHAP Values

Figure 5 shows the global feature importance based on the mean absolute SHAP values. Height is identified as the most important feature, followed by Age (Month), whereas Weight and Gender have substantially lower importance. Consistent with the results shown in Figure 4, these findings indicate that Height and Age (Month) made the largest contributions to the model's classification outputs.

3.7. SHAP Local Explanation

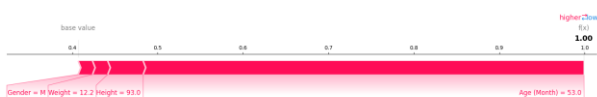


Figure 6. SHAP Force Plot for an Individual Prediction

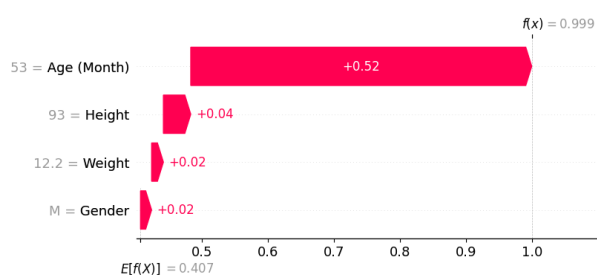


Figure 7. SHAP Waterfall Plot for the Same Individual Prediction

Figures 6 and 7 present the local explanation of the first true-positive Stunted observation identified in the test-set order. The observation was a 53-month-old male child with a Weight of 12.2 kg and a Height of 93 cm. Its actual class was Stunted, and the Random

Forest model also classified it as Stunted, with an approximate predicted probability of 0.999 for the Stunted class. The explanation starts from an expected model output of approximately 0.407 and shows how the predictor values contributed to the final output.

Among all features, Age (Month) = 53 provides the largest positive contribution to the prediction, followed by smaller positive contributions from Height = 93, Weight = 12.2, and Gender = M. The Waterfall Plot shows the magnitude of these contributions more explicitly, with Age (Month) contributing approximately +0.52, while the other features contribute around +0.04 and +0.02. Overall, both visualizations indicate that Age (Month) is the primary factor driving the prediction for this individual instance, while the remaining features provide smaller supporting contributions.

3.8. Discussion

The results demonstrate that the effectiveness of SMOTEENN depended on both the characteristics of the dataset and the classification algorithm. In the present study, the original training data already enabled the tree-based models to achieve strong performance, with Random Forest obtaining the highest overall results. After applying SMOTEENN, Recall increased across all models, indicating an improved ability to identify Stunted cases. However, this improvement was generally accompanied by reductions in Precision and other overall performance measures for the tree-based models.

This trade-off can be explained by the effect of resampling on the class distribution and decision boundary. Although SMOTEENN increases the representation of the minority class, synthetic sampling and neighbourhood-based cleaning may also alter the original data structure. Altalhan et al. noted that oversampling may introduce ambiguous synthetic samples or reduce model generalization, particularly when class overlap or noise is present. Similarly, hybrid methods such as SMOTEENN may unintentionally remove informative observations during the cleaning process. In the present study, this may explain why the improvement in Recall was accompanied by decreases in Precision, Accuracy, F1-Score, and ROC-AUC for Random Forest, XGBoost, and Decision Tree [31].

The findings are consistent with Sugihartono et al. (2026), who compared multiple oversampling strategies for pediatric stunting detection. Their results showed that the effectiveness of sampling strategies was model-dependent: XGBoost achieved its best performance without oversampling, while SMOTEENN consistently produced lower performance across the evaluated classifiers. The authors further concluded that sampling strategies should not be considered universally beneficial because their effects vary

according to the model and the characteristics of the dataset [32]. This pattern is comparable to the present study, where the original data produced the best overall performance for Random Forest, whereas SMOTEENN primarily contributed to improving Recall.

However, the present findings differ from Furqany et al. (2025), who reported substantial improvements after applying SMOTEENN. In their study, all evaluated models showed improvements in Accuracy, Precision, Recall, and F1-Score after resampling, with XGBoost increasing from 74.56% Accuracy and 71.08% F1-Score to 91.82% Accuracy and 91.74% F1-Score [12]. This contrast further demonstrates that the effectiveness of SMOTEENN is highly dependent on the underlying classification problem. The difference may be related to variations in dataset characteristics, feature composition, class distribution, and data complexity between the two studies.

This interpretation is supported by Carvalho et al. (2025), who emphasized that class imbalance alone does not determine the difficulty of a classification problem. Other factors, including class overlap, noise, data structure, and decision boundary complexity, also influence classifier performance [33]. Therefore, the relatively moderate class imbalance in the present dataset should not be considered the sole reason why the original data produced strong results. Instead, the interaction between the data characteristics and the learning mechanisms of the classification algorithms may have enabled Random Forest and XGBoost to effectively distinguish between the Stunted and Not Stunted classes without requiring additional resampling.

Although the class proportions remained relatively stable after outlier removal, IQR identifies statistically extreme observations rather than biologically invalid measurements. Therefore, some of the 96 removed Stunted observations may have represented valid children with severe anthropometric conditions. This possibility should be considered when interpreting the reported model performance.

The SHAP analysis showed that Height and Age (Month) contributed most strongly to the outputs of the best-performing model, whereas Weight and Gender had smaller contributions. These findings should be interpreted cautiously because Height-for-Age status is generally determined using a child's height, age, and gender, while the same variables were included as model predictors. This creates a potentially circular or partially deterministic relationship between the predictors and the target. Such a relationship may partly explain the very high ROC-AUC values and the dominant contributions of Height and Age in the SHAP results. Therefore, Height and Age should not be interpreted as newly identified or causal risk factors. In this study, their SHAP values only describe their

contributions to the model's classification outputs. This result indicates that Height had the most consistent influence on model predictions across the evaluated dataset. In contrast, the local SHAP explanation showed that feature contributions may vary for individual observations. For the analyzed instance, Age (Month) provided the largest contribution to the prediction, followed by smaller contributions from Height, Weight, and Gender. Therefore, the feature with the greatest global importance is not necessarily the feature with the largest contribution to every individual prediction.

This distinction between global and local explanations highlights the complementary role of SHAP in model interpretation. Global SHAP analysis identifies features that consistently influence predictions across the dataset, whereas local SHAP analysis explains how specific feature values contribute to an individual prediction. A similar use of SHAP for both interpretability and feature contribution analysis was also demonstrated by Furqany et al. (2025), although the dominant predictors in their study differed because their dataset focused primarily on household-level demographic, socioeconomic, and environmental variables [12].

Overall, the findings indicate that Random Forest trained on the original data provided the best overall classification performance, achieving an Accuracy of 0.9600, Precision of 0.9584, F1-Score of 0.9505, and ROC-AUC of 0.9927. SMOTEENN consistently improved Recall but did not improve all evaluation metrics, particularly for the tree-based models. These findings reinforce the evidence that data resampling should not be assumed to automatically improve classification performance. Instead, the selection of a resampling strategy should consider the characteristics of the dataset, the classification algorithm, and the intended objective of the model, such as prioritizing overall predictive performance or maximizing the detection of Stunted cases.

The contribution of this study lies in its controlled comparison of four algorithms under original and SMOTEENN-resampled training scenarios using fixed model configurations, combined with an evaluation of metric-specific trade-offs and global and local SHAP explanations. Therefore, the study represents a comparative empirical approach rather than a newly developed technical framework.

Despite the high classification performance, the evaluated models cannot yet be considered ready for clinical use. At this stage, the approach may support the automated classification of current stunting status within the analyzed datasets. Its use for practical screening or healthcare decision-making would require external validation using independent populations and evaluation involving qualified healthcare professionals.

3.9. Study Limitations

This study has several limitations. First, it relied on secondary data, limiting control over sampling, measurement, and data-quality procedures. The provenance and clinical validity of the Kaggle dataset could not be independently verified because information regarding the collecting institution, collection period, sampling method, measurement quality, and whether the observations were real or synthetic was unavailable. Its label-generation procedure was also undocumented, limiting verification of compatibility with the Mendeley dataset. Although variable names, units, and age groups were harmonized, integrating two data sources may have introduced heterogeneity in target definitions and measurement procedures.

Second, Height-for-Age status is generally derived from Height, Age, and Gender, which were also used as predictors. This potentially deterministic relationship may partly explain the high ROC-AUC values and the dominance of Height and Age in the SHAP results. Third, outliers were removed using IQR thresholds rather than WHO biologically plausible limits; consequently, some valid extreme cases may have been excluded. In addition, applying SMOTEENN to numerically encoded Gender may have generated values that do not represent valid categories.

Finally, the evaluation relied on a single stratified train-test split and lacked external validation. Hyperparameters optimized on the original training data were applied to the SMOTEENN data without separate tuning. Thus, the comparison evaluates the effect of SMOTEENN under fixed model configurations rather than its maximum achievable performance. Future studies should use repeated evaluation, fold-specific resampling and tuning, and external validation using well-documented datasets before practical or clinical implementation.

4. Conclusion

This study compared Logistic Regression, Decision Tree, Random Forest, and XGBoost for classifying current stunting status using original and SMOTEENN-resampled training data. Within the analyzed datasets, Random Forest trained on the original data achieved the best overall classification performance. SMOTEENN consistently improved Recall, particularly for Logistic Regression, but did not uniformly improve the other evaluation metrics. SHAP analysis showed that Height and Age made the largest contributions to the model outputs; however, this finding reflects their direct relationship with the Height-for-Age target and should not be interpreted as the identification of new risk factors.

The findings are limited to the analyzed datasets and support only the automated classification of current stunting status. The models should not be interpreted as predicting future stunting risk or as being ready for prospective screening or healthcare decision-making. External validation using independent, well-documented datasets and evaluation involving qualified healthcare professionals are required before practical implementation.

Acknowledgements

This research was funded by the Ministry of Higher Education, Science, and Technology of the Republic of Indonesia through the Penelitian Dosen Pemula (PDP) funding scheme for the 2026 Fiscal Year under Main Grant Contract No. 282/C3/DT.05.00/PL-BARU/2026, Contract No. 0633/LL3/AL.04/2026 between LLDIKTI Region III and Universitas Nusa Mandiri, and Institutional Contract No. 006/SPK/LPPM-UNM/PDP/2026 between Universitas Nusa Mandiri and the Principal Investigator.

References

- [1] C. P. W. Kase and S. Y. J. Prasetyo, "Analisis Faktor Risiko Stunting pada Balita di Desa Kesetnana Menggunakan Metode Random Forest," *J. Indones. Manaj. Inform. dan Komun.*, vol. 6, no. 3, pp. 1556–1566, 2025, doi: 10.63447/jimik.v6i3.1449.
- [2] I. Indriani, M. Mujahadatuljannah, and R. Rabiattunnisa, "Factors Affecting Incidence of Stunting in Infants and Toddlers," *J. Surya Med.*, vol. 9, no. 3, pp. 131–136, 2023, [Online]. Available: <https://akper-sandikarsa.e-journal.id/JIKSH/article/view/1019/584>
- [3] K. K. R. Indonesia, "Survei Status Gizi Indonesia (SSGI) 2024 dalam Angka," Jakarta, 2025. [Online]. Available: <https://stunting.go.id/ssgi-2024-prevalensi-stunting-nasional-turun-jadi-198-capai-angka-di-bawah-proyeksi-bappenas/>
- [4] World Health Organization, *WHO Child Growth Standards: Length/Height-for-Age, Weight-for-Age, Weight-for-Length, Weight-for-Height and Body Mass Index-for-Age: Methods and Development*. Geneva, Switzerland: World Health Organization, 2006. [Online]. Available: <https://www.who.int/publications/i/item/924154693X>
- [5] R. Ratnasari, A. J. Wahidin, and T. H. Andika, "Deteksi Dini Stunting Pada Anak Berdasarkan Indikator Antropometri dengan Menggunakan Algoritma Machine Learning," *J. Algoritma.*, vol. 21, no. 2, pp. 378–387, 2024, doi: 10.33364/algoritma/v.21-2.2122.
- [6] T. Hidayat, I. Sembiring, H. D. Purnomo, and A. Iriani, "Prediksi Prevalensi Stunting Balita dengan Pendekatan Algoritma Support Vector Machine dan Synthetic Minority Oversampling Technique (SMOTE)," *J. Pekommas*, vol. 10, no. 1, pp. 9–16, 2025, doi: 10.56873/jpkm.v9i1.5389.
- [7] N. I. Syahfitri, A. P. Juledi, and R. Muti'ah, "Comparative Analysis of Machine Learning Algorithm Performance in Predicting Stunting in Toddlers," *Sinkron*, vol. 8, no. 3, pp. 1452–1462, 2024, doi: 10.33395/sinkron.v8i3.13698.
- [8] E. N. Candra, I. Cholissodin, and R. C. Wihandika, "Klasifikasi Status Gizi Balita Menggunakan Metode Optimasi Random Forest Dengan Algoritme Genetika (Studi Kasus: Puskesmas Cakru)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 5, pp. 2188–2197, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>

- [9] M. K. Ayele, G. A. Baye, S. H. Yesuf, A. A. Engda, and E. T. Mitiku, "Predicting stunting status among under five children in ethiopia using ensemble machine learning algorithms," *Sci. Rep.*, vol. 15, no. 1, pp. 1–11, 2025, doi: 10.1038/s41598-025-03206-1.
- [10] T. Sugihartono, B. Wijaya, Marini, A. F. Alkayes, and H. A. Anugrah, "Optimizing Stunting Detection through SMOTE and Machine Learning: a Comparative Study of XGBoost, Random Forest, SVM, and k-NN," *J. Appl. Data Sci.*, vol. 6, no. 1, pp. 667–682, 2025, doi: 10.47738/jads.v6i1.494.
- [11] W. R. Mgonezulu, 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," *Hum. Nutr. Metab.*, vol. 42, no. August, p. 200340, 2025, doi: 10.1016/j.hnm.2025.200340.
- [12] N. El Furqany, M. Subianto, and A. Rusyana, "Hybrid Ensemble Learning with SMOTEENN and Soft Voting for Stunting Risk Prediction: A SHAP-Based Explainable Approach," *J. Appl. Data Sci.*, vol. 6, no. 4, pp. 2989–3004, 2025, doi: 10.47738/jads.v6i4.829.
- [13] N. Novalina, I. A. A. Tarigan, F. K. Kameela, and M. Rizkinia, "Benchmarking machine learning algorithm for stunting risk prediction in Indonesia," *Bull. Electr. Eng. Informatics*, vol. 14, no. 3, pp. 2252–2263, 2025, doi: 10.11591/eei.v14i3.8997.
- [14] Y. Li, Y. Yang, P. Song, L. Duan, and R. Ren, "An improved SMOTE algorithm for enhanced imbalanced data classification by expanding sample generation space," *Sci. Rep.*, vol. 15, no. 1, pp. 1–21, 2025, doi: 10.1038/s41598-025-09506-w.
- [15] H. W. Loh, C. P. Ooi, S. Seoni, P. D. Barua, F. Molinari, and U. R. Acharya, "Application of explainable artificial intelligence for healthcare: A systematic review of the last decade (2011–2022)," *Comput. Methods Programs Biomed.*, vol. 226, p. 107161, 2022, doi: 10.1016/j.cmpb.2022.107161.
- [16] N. Hettikankanamage, N. Shafiabady, F. Chatteur, R. M. X. Wu, F. Ud Din, and J. Zhou, "eXplainable Artificial Intelligence (XAI): A Systematic Review for Unveiling the Black Box Models and Their Relevance to Biomedical Imaging and Sensing," *Sensors*, vol. 25, no. 21, pp. 1–31, 2025, doi: 10.3390/s25216649.
- [17] A. H. RS et al., "Dataset Stunting and Nutritional Status of Toddler from Jeneponto Regency, South Sulawesi, Indonesia," 2026, *Mendeley Data*. doi: 10.17632/WZWPC9J5BX.
- [18] J. Peng, J. Hahn, and K. W. Huang, "Handling Missing Values in Information Systems Research: A Review of Methods and Assumptions," *Inf. Syst. Res.*, vol. 34, no. 1, pp. 5–26, 2023, doi: 10.1287/isre.2022.1104.
- [19] V. Çetin and O. Yıldız, "A comprehensive review on data preprocessing techniques in data analysis," *Pamukkale Univ. J. Eng. Sci.*, vol. 28, no. 2, pp. 299–312, 2022, doi: 10.5505/pajes.2021.62687.
- [20] I. Souiden, M. N. Omri, and Z. Brahmi, "A survey of outlier detection in high dimensional data streams," *Comput. Sci. Rev.*, vol. 44, p. 100463, 2022, doi: 10.1016/j.cosrev.2022.100463.
- [21] F. Pargent, F. Pfisterer, J. Thomas, and B. Bischl, "Regularized target encoding outperforms traditional methods in supervised machine learning with high cardinality features," *Comput. Stat.*, vol. 37, no. 5, pp. 2671–2692, 2022, doi: 10.1007/s00180-022-01207-6.
- [22] M. R. Islam, A. A. Lima, S. C. Das, M. F. Mridha, A. R. Prodeep, and Y. Watanobe, "A Comprehensive Survey on the Process, Methods, Evaluation, and Challenges of Feature Selection," *IEEE Access*, vol. 10, no. July, pp. 99595–99632, 2022, doi: 10.1109/ACCESS.2022.3205618.
- [23] A. Géron, *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. "O'Reilly Media, Inc.," 2022.
- [24] L. B. V De Amorim, G. D. C. Cavalcanti, and R. M. O. Cruz, "The choice of scaling technique matters for classification performance," *Appl. Soft Comput.*, vol. 133, p. 109924, 2023, doi: 10.1016/j.asoc.2022.109924.
- [25] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD Explor. Newsl.*, vol. 6, no. 1, pp. 20–29, 2004, doi: 10.1145/1007730.1007735.
- [26] E. Dumitrescu, S. Hué, C. Hurlin, and S. Tokpavi, "Machine learning for credit scoring: Improving logistic regression with non-linear decision-tree effects," *Eur. J. Oper. Res.*, vol. 297, no. 3, pp. 1178–1192, 2022, doi: 10.1016/j.ejor.2021.06.053.
- [27] C. Kwenda, M. Gwetu, and J. V. F. Dombeu, "Machine Learning Methods for Forest Image Analysis and Classification: A Survey of the State of the Art," *IEEE Access*, vol. 10, pp. 45290–45316, 2022, doi: 10.1109/ACCESS.2022.3170049.
- [28] I. D. Mienye, Y. Sun, and S. Member, "A Survey of Ensemble Learning: Concepts , Algorithms , Applications , and Prospects," *IEEE Access*, vol. 10, no. September, pp. 99129–99149, 2022, doi: 10.1109/ACCESS.2022.3207287.
- [29] G. Naidu, T. Zuva, and E. M. Sibanda, "A Review of Evaluation Metrics in Machine Learning Algorithms BT - Artificial Intelligence Application in Networks and Systems," R. Silhavy and P. Silhavy, Eds., Cham: Springer International Publishing, 2023, pp. 15–25.
- [30] H. Chen, I. C. Covert, S. M. Lundberg, and S.-I. Lee, "Algorithms to estimate Shapley value feature attributions," *Nat. Mach. Intell.*, vol. 5, no. 6, pp. 590–601, 2023, doi: 10.1038/s42256-023-00657-x.
- [31] M. Altalhan, A. Algami, and M. T. Alouane, "Imbalanced Data Problem in Machine Learning: A Review," *IEEE Access*, vol. 13, no. January, pp. 13686–13699, 2025, doi: 10.1109/ACCESS.2025.3531662.
- [32] T. Sugihartono, D. Soetarno, R. Sulaiman, Sarwindah, Marini, and Fitriyani, "Ensemble learning for pediatric stunting detection: A comparative study of XGBoost, Random Forest, and LightGBM with oversampling techniques," *J. Inf. Syst. Inform.*, vol. 8, no. 2, pp. 1672–1692, 2026, doi: 10.63158/journalisi.v8i2.1568.
- [33] M. Carvalho, A. J. Pinho, and S. Brás, "Resampling approaches to handle class imbalance : a review from a data perspective," *J. Big Data*, 2025, doi: 10.1186/s40537-025-01119-4.