# Comparison of Naïve Bayes and K-Nearest Neighbor (K-NN) Methods in Classifying Stunting in Toddlers in Takalar Regency

**Wahidah Sanusi[1], Irwan[1*], and Aliyah Arianti Halim[1]**
[1]Mathematics Study Program: Faculty of Mathematics and Natural Sciences, Makassar State University, Makassar City, South Sulawesi, Indonesia
*Corresponding author: irwanthaha@unm.ac.id

**Received:** 20 January 2026
**Revised:** 18 May 2026
**Accepted:** 20 May 2026

**ABSTRACT** – **Stunting is a chronic nutritional problem that has long-term effects on children's physical growth and cognitive development. Therefore, classifying the nutritional status of toddlers is an important step in early detection and determining appropriate interventions. This study aims to compare the performance of two classification methods, namely Gaussian Naïve Bayes and K-Nearest Neighbor (K-NN), in identifying the nutritional status of toddlers. The data used consisted of 14,620 toddler data obtained from the Takalar District Health Office covering 11 sub-districts. Gaussian Naïve Bayes is a probabilistic classification method with the assumption of independence between variables, while K-NN is a nonparametric method that classifies data based on the proximity of the distance between observations. The results showed that Gaussian Naïve Bayes produced an accuracy of 91.76%, but was unable to accurately classify stunting classes due to class imbalance and low posterior probability values in minority classes. In contrast, the K- NN method with an optimal parameter value of k=3 produced an accuracy of 97.00% and showed better performance in identifying toddlers with stunting status. Based on these results, the K-NN method is considered superior to Gaussian Naïve Bayes in classifying the nutritional status of toddlers in Takalar Regency.**
**Keywords**– **Stunting; Gaussian Naïve Bayes; K-Nearest Neighbor; Classification, Toddlers.**

## 1. INTRODUCTION

Data mining is a systematic process for extracting patterns, relationships, and hidden knowledge from large data sets. This process uses a multidisciplinary approach involving machine learning, artificial intelligence, statistics, and database systems. Data mining includes various analysis techniques, such as cluster analysis, association analysis, anomaly detection, and predictive modeling, which includes classification and regression. These techniques play an important role in supporting data-driven decision making [1].

Classification is one of the most widely used techniques in data mining. This technique maps objects into specific classes based on their characteristics or attributes. Various algorithms have been developed to support the classification process, including Naïve Bayes and K-Nearest Neighbor (K-NN). Both algorithms perform well in handling data with a relatively large number of features and varying data complexity [2]. The differences in the characteristics of these two algorithms make them interesting to study and compare their performance in the context of specific problems [3].

Naïve Bayes is a probabilistic classification algorithm based on Bayes' theorem. This algorithm uses the assumption of independence between features. Although this assumption is not always true in real data, Naïve Bayes is still widely used because of the simplicity of the model and computational efficiency. This algorithm also has good speed in the training and prediction processes [4].

K-Nearest Neighbor (K-NN) is an instance-based learning algorithm. This algorithm classifies data based on proximity to a number of closest neighbors using a specific distance metric [5]. K- NN has the advantage of capturing nonlinear patterns and complex relationships between features. However, this algorithm has limitations in computational efficiency on large datasets and sensitivity to outliers and class imbalance [6]. With these characteristics, Naïve Bayes and K-NN are relevant algorithms to apply and compare in various fields, including health [7].

Stunting is a public health issue that can be studied using a classification approach. Stunting is a condition of growth failure in children due to chronic malnutrition over a long period of time, especially during the first 1,000 days of life [8]. This condition affects children's cognitive and motor development and increases the risk of illness and death in the future [9].

Globally, 149.2 million or around 22% of toddlers experienced stunting in 2020. Asia accounted for more than half of these cases [10]. In Indonesia, the prevalence of stunting shows a downward trend, but the figure is still above the threshold set by the World Health Organization (WHO), which is 20% [11][12].

Previous studies have shown that classification algorithms can be used to identify the nutritional status of toddlers. Several studies have reported that the application of the Naïve Bayes algorithm resulted in an accuracy rate of 88% [4]. Another study showed that the K-NN method with a parameter of k = 5 achieved an accuracy rate of 73.53% [13]. However, most previous studies have focused on the application of a single algorithm separately. These studies have not directly compared the performance of different methods, particularly on stunting data with specific regional characteristics such as Takalar Regency [14].

While the WHO Z-score formula provides a deterministic method for classifying stunting, its application in real-world settings requires access to WHO reference growth tables and precise anthropometric measurement protocols [15]. This study investigates the capability of machine learning algorithms (specifically K-Nearest Neighbor (K-NN) and

Gaussian Naïve Bayes) to learn the mapping from raw anthropometric features to stunting classification status, offering a computationally lightweight and easily deployable screening model that does not require explicit reference-table lookups. The primary contribution is a comparative evaluation of algorithm performance and robustness under imbalanced class conditions, providing a benchmark for algorithm selection in stunting screening systems..

## 2. LITERATURE REVIEW

### 2.1 Data Mining

Data mining is the process of extracting meaningful patterns and knowledge from large data sets. Mostafa and Mahmoud (2022) explain that data mining is a technology that extracts hidden information from large-scale data as part of the Knowledge Discovery in Databases (KDD) process. The KDD process includes the stages of data cleaning, data integration, data selection, data transformation, application of data mining techniques, pattern evaluation, and knowledge presentation, so that raw data can be processed into valuable information that supports data-driven decision-making [16].

### 2.2 Classification

Classification is a technique in data mining used to group or map data into specific classes based on its characteristics or attributes, where the process involves building models from labeled data to predict new data categories [17]. Srirahayu and Pribadie (2023) state that classification is a method commonly used in data mining for data analysis and grouping with the aim of finding patterns and significant information from large data sets.

### 2.3 Naïve Bayes

Naïve Bayes is a probabilistic-based classification algorithm that uses Bayes' Theorem with the assumption that each attribute is independent of the other attributes. This algorithm determines the class of a data based on the highest posterior probability value obtained from the calculation of the initial probability (prior) and conditional probability (likelihood) of each attribute. Mathematically, Naïve Bayes classification is formulated as follows [18][19]:

$$P(C|X) = \frac{P(X|C).P(C)}{P(X)} \quad (1)$$

$P(X|C).P(C)$
$P(C|X) =$
$P(X)$ (1)
Explanation:
$P(C|X)$ : probability of data $X$ belonging to class $C$
$P(X|C)$ : probability of $X$ data appearing in the class $C$
$P(C)$ : prior probability of the class $C$
$P(X)$ : probability of data occurrence $X$
Assuming independence between features, the combined probability of all data attributes is calculated using the following equation:

$$P(X|C) = \prod_{i=1}^{n} P(x_i|C) \quad (2)$$

$x_i$ : the i-th attribute
$n$ : the number of attributes
For numerical data, the Naïve Bayes algorithm generally uses a normal (Gaussian) distribution approach. The probability of a numerical attribute is calculated using the following equation:

$$P(x_i|C) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x_i-\mu)^2}{2\pi\sigma^2}\right) \quad (3)$$

With:
$\mu$ : attribute mean
$\sigma$ : attribute standard deviation
The final class determination is made by selecting the class with the highest posterior probability or Maximum A Posteriori (MAP), which is formulated as follows:

$$c_{MAP} = \underset{c \epsilon C}{argmax} P(c) \cdot P(X|c) \quad (4)$$

### 2.4 K-Nearest Neighbor

K-NN is an instance-based learning classification algorithm that determines the class of a data point based on its proximity to a number of the closest training data points. The classification process is carried out by calculating the

distance between the test data and all training data, then determining the majority class of the closest neighbors up to a predetermined value of the "$k$" [5][20]. This method does not require explicit model training because all training data is used directly in the classification process.

Distance calculations in the K-NN algorithm generally use Euclidean distance, which is formulated as follows:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{5}$$

With:

$d(x, y)$ : distance between the test data and the training data
$x_i$ : the value of the i-th attribute in the test data
$y_i$ : the value of the i-th attribute in the training data
$n$ : number of attributes

A smaller distance value indicates a higher level of similarity between data. After the distance is calculated, the training data is sorted based on the smallest distance, then the test data class is determined based on the most dominant class among the nearest neighbors [3][21].

### 2.5 *Confusion Matrix*

The Confusion Matrix is an evaluation method used to measure the performance of classification algorithms by comparing prediction results with actual data. The Confusion Matrix consists of four main components, namely True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), which are used as the basis for calculating classification evaluation metrics. Based on the Confusion Matrix, the Classification Measurement and Performance (CMAP) value is calculated using several evaluation metrics in Table 1 [22].

Table 1 Confusion Matrix

| Actual Class | Predicted Class | |
|---|---|---|
| | Positive | Negative |
| True | TP | TN |
| False | FP | FN |

Accuracy describes the proportion of correct predictions in the entire sample, as formulated in Equation (8).

$$accuracy = \frac{TP + TN}{TP + FP + FN + TN} \times 100\% \tag{6}$$

Precision menunjukkan proporsi prediksi positif yang benar dari seluruh prediksi positif, sebagaimana ditunjukkan pada Persamaan (9).

$$precision = \frac{TP}{TP + FP} \times 100\% \tag{7}$$

Recall measures the proportion of positive cases that are correctly predicted from the entire actual positive class, as formulated in Equation (10).

$$recall = \frac{TP}{TP + FN} \times 100\% \tag{8}$$

F1-Score represents the harmonic mean between precision and recall, as shown in Equation (11).

$$F1 - Score = 2.\frac{precision\,.recall}{precision + recall} \tag{9}$$

$$Specifity = \frac{TN}{FP+TN} \times 100\% \tag{10}$$

## 3. METHODOLOGY

### 3.1 Data Sources and Research Variables

This study is an applied quantitative study with a supervised classification approach to compare the performance of the Naïve Bayes and K-NN algorithms in classifying the nutritional status of toddlers in Takalar District [23]. The data used was sourced from the Takalar District Health Office and covers the period January–December 2024. The dataset was obtained through routine nutritional examinations and stunting monitoring at 18 community health

centers in 11 subdistricts. Overall, the dataset consisted of 14620 observations, each representing one toddler, as shown in Table 2.

Table 2 Dataset on the nutritional status of toddlers in Takalar Regency

| Toddler Label | Gender | Age | Weight | Height | Upper Arm Circumference (LiLA) | Z-Score Height/Age (Z/S TB/U) |
|---|---|---|---|---|---|---|
| ID1 | L | 32 | 11.5 | 87.4 | 16.3 | -1.71 |
| ID2 | L | 40 | 14.7 | 92.5 | 13 | -1.63 |
| ID3 | L | 56 | 17.5 | 102.6 | 16 | -1.18 |
| ID4 | L | 39 | 13.6 | 94.7 | 14.2 | -0.83 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| ID14620 | L | 6 | 6.8 | 62 | 13 | -2.29 |

Based on Table 2, the dataset contains six main attributes, namely gender, where the TB/U Z-score value is used as the basis for determining nutritional status. The role and data type of each attribute are presented in Table 3.

Table 3 Input and Output Attributes in the Dataset

| No | Attribute Name | Data Type | Role |
|---|---|---|---|
| 1 | JK | *Integer* | *Input* |
| 2 | Age at Measurement | *Floating* | *Input* |
| 3 | Weight | *Floating* | *Input* |
| 4 | Height | *Floating* | *Input* |
| 5 | LiLA | *Floating* | *Input* |
| 6 | Nutritional Status | *Floating* | *Output* |

### 3.2 Research Stages

The stages of data analysis conducted in this study are described as follows:

1) Collecting data on the nutritional status of toddlers in Takalar Regency from the Health Office.
2) Performing data cleaning and tidying up.
3) Entering the dataset into Google Colab as a programming environment.
4) Performing data preprocessing, including:
   a Data Cleaning: Checking for missing values, removing duplicate data, validating unusual values (zero-values and outliers)
   b Data encoding for gender and toddler nutritional status
   c Data normalization using Z-score standardization
5) Determining input and output attributes in the dataset.
6) Splitting the data into training data and testing data.
7) Handling Class Imbalance with SMOTE [20]
8) Performing the classification process using the Naïve Bayes method, with the following steps:
   a Calculating prior values
   b Calculating the mean and variance parameters
   c Calculating the likelihood value
   d Calculating the posterior value and determining the prediction class
9) Evaluating the performance of the Naïve Bayes model using a confusion matrix.
10) Performing the classification process using the K-Nearest Neighbor (K-NN) method, with the following steps:
   a Calculating the distance between data using Euclidean Distance
   b Determining the optimal k value using 10-fold cross-validation
   c Determining the prediction class based on the k nearest neighbors
11) Evaluate the performance of the K-NN model using a confusion matrix.
12) Comparing the performance of the Naïve Bayes and K-NN methods based on accuracy, precision, recall, and F1-score values.

## 4. RESULTS AND DISCUSSIONS

It is acknowledged that the features used in this study (age, sex, weight, and height) are the direct components of the WHO Z-score calculation. The model therefore learns a data-driven approximation of a known formula. The practical value of this approach lies in its deployment simplicity: the trained model can be integrated directly into mobile or web-based screening tools without requiring WHO reference table access, which is particularly relevant in resource-constrained primary healthcare settings [24].

### 4.1 Data Preprocessing

The initial stage of data processing was carried out through preprocessing, which included data cleaning, coding of gender attributes (L = 0 and P = 1) and nutritional status attributes (non-stunting = 0 and stunting = 1), and data normalization using Z-score standardization. The results of data normalization are presented in Table 4.

Table 4 Dataset of Toddler Nutritional Status After Data Standardization

| No | JK | Age | LiLA | Weight | Height | Nutritional Status |
|---|---|---|---|---|---|---|
| 0 | 0.931255 | -0.021606 | -0.126220 | -0.111769 | 0.028132 | 0 |
| 1 | 0.931255 | -0.332180 | 0.100849 | -0.217877 | -0.228763 | 0 |
| 2 | 0.931255 | 0.351083 | -0.126220 | 0.913933 | 0.327842 | 0. |
| 3 | 0.931255 | -1.760819 | -0.201090 | -1.455793 | -1.872889 | 0. |
| 4 | 0.931255 | -0.518524 | -0.126220 | -0.748412 | -0.879563 | 1.0 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 14640 | 0.931255 | -1.698075 | -0.277599 | -1.703377 | -2.198289 | 1 |

Based on Table 4, the standardized data was used in the training and testing stages with a division of 70% training data (10248 toddlers) and 30% testing data (4068 toddlers), using the Naïve Bayes and K-NN methods.

The dataset exhibits class imbalance, with the non-stunted class significantly outnumbering the stunted class. Training an unmodified classifier on imbalanced data risks producing a model biased toward the majority class, as evidenced by the Gaussian Naïve Bayes baseline results. To address this, Synthetic Minority Over-sampling Technique (SMOTE) [25] is applied to the training set prior to model fitting. SMOTE generates synthetic samples for the minority class by interpolating between existing minority instances in feature space, balancing class distribution without duplicating existing samples. Oversampling is applied exclusively to the training set to prevent data leakage into the test set.

### 4.2 Naïve Bayes Classification

The first stage involved classification using Gaussian Naïve Bayes through the Python software . This model studied the probability distribution of each attribute against nutritional status classes based on prior and likelihood values distributed according to a Gaussian distribution. The prediction results for the test data are presented in Table 5, while the comparison between the actual labels and predictions is shown in Figure 1.

Table 5 Test Data Prediction Class Using Gaussian Naïve Bayes

| Toddler Label | $P(Normal \mid x)$ | $P(Stunting \mid x)$ | $C_{MAP}$ | Label Actual |
|---|---|---|---|---|
| D1 | 0.968249 | 0.031751 | 0 | 0 |
| D2 | 0.964773 | 0.035227 | 0 | 0 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| D24 | 0.948758 | 0.051242 | 0 | 1 |
| D25 | 0.691064 | 0.308936 | 0 | 0 |
| D26 | 0.689902 | 0.310098 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| D156 | 0.667453 | 0.332546 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| D4067 | 0.691002 | 0.308998 | 0 | 0 |
| D4068 | 0.987927 | 0.012073 | 0 | 0 |

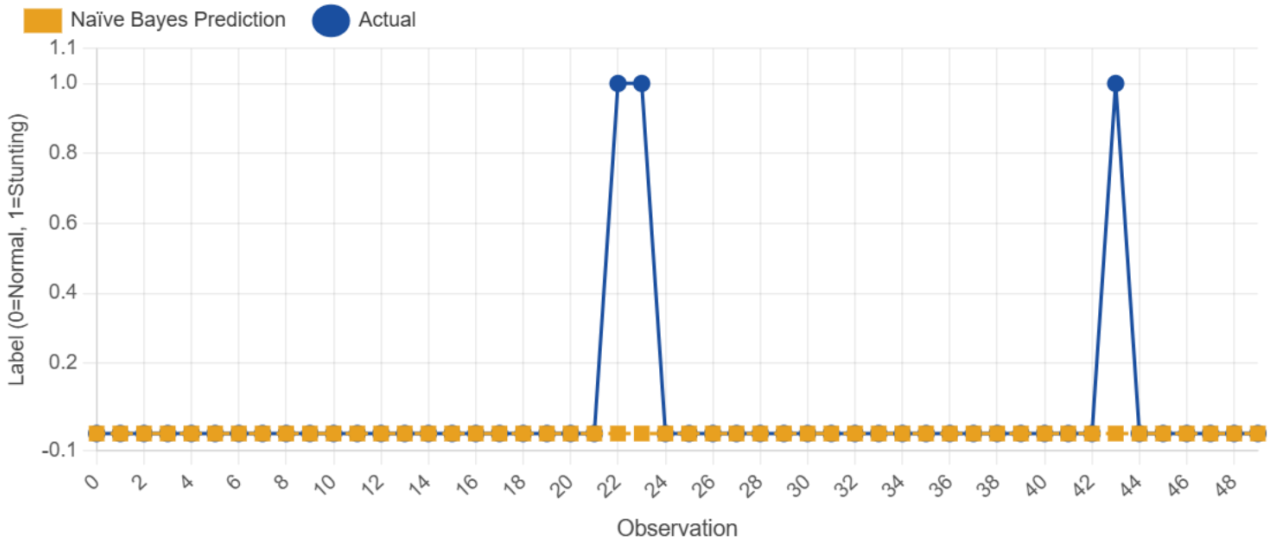


Figure 1 Plot of Predicted Labels and Actual Labels Using the *Naïve Bayes* Method

Based on the test data prediction results presented in Table 5 and Figure 1, the Naïve Bayes model predicts all data on label 0 along the observation axis. The model only produces predictions in the non-stunting class, while some actual data with label 1 appear in observations 24, 26, and 44 without being followed by prediction results in the same class. The discrepancy between the actual labels and the predicted labels indicates that the Naïve Bayes model is unable to identify stunting cases in the test data. This finding indicates the model's low ability to detect the stunting class, especially in conditions of class distribution imbalance.

### 4.3 Evaluation of Naïve Bayes Method Performance

Model performance was evaluated using a confusion matrix, with the results presented in Table 6 and Figure 2.

Table 6 Naïve Bayes Confusion Matrix

| Actual Class | Predicted Class | |
|---|---|---|
| | Non-stunting | *Stunting* |
| Non-stunting | 3733 | 0 |
| *Stunting* | 335 | 0 |

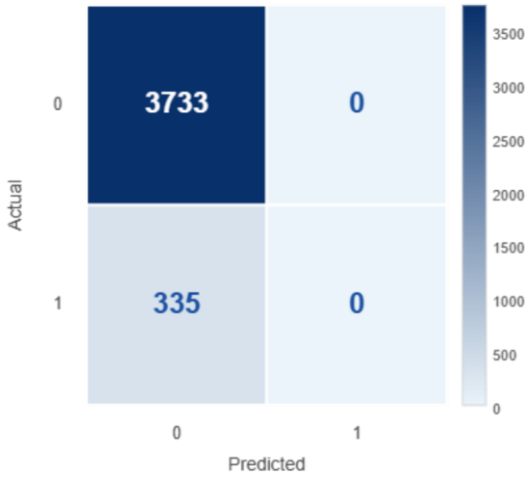


**Figure 2** Plot Confusion Matrix Naïve Bayes

Based on the confusion matrix in Table 6 and its visualization in Figure 2, the Naïve Bayes model classified all test data into the non-stunting class. A total of 3733 data points with the actual non-stunting class were correctly predicted, while all data points with the actual stunting class, namely 335 data points, were predicted as non-stunting and none were successfully classified as stunting. This condition shows that the model has a true positive value of zero and a high false negative value in the stunting class, which indicates a bias towards the majority class. As a result, even though the accuracy produced is relatively high, the ability of the Naïve Bayes model to detect

stunting cases is very low, as reflected in the precision, recall, and F1-score values for the stunting class, which are zero.

### 4.4 K-NN Classification

Next, classification was performed using K-NN. In this method, each test data is classified based on the Euclidean distance to all training data, with the number of nearest neighbors k as the basis for determining the class. An example of the Euclidean distance calculation results sorted from the smallest neighbors is presented in Table 7.

Table 7 Distance Calculation Results on Test Data for Test Data

| No | JK | Age | Weight | Height | LiLA | Nutritional Status | *Distance* |
|---|---|---|---|---|---|---|---|
| 9083 | 0.93126 | -0.0837 | 0.0403 | 0.31266 | 0.33641 | 0 | 0.08069 |
| 6960 | 0.93126 | -0.0216 | 0.02516 | 0.31266 | 0.28503 | 0 | 0.09265 |
| 2375 | 0.93126 | -0.0216 | -0.0203 | 0.3834 | 0.28503 | 0 | 0.10318 |
| 3528 | 0.93126 | 0.04051 | 0.02516 | 0.3834 | 0.45629 | 0 | 0.11154 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| 3752 | 0.93126 | -1.9472 | 8.85055 | -2.2339 | -2.498 | 0 | 9.82298 |
| 1452 | -1.0738 | 0.04051 | 11.2272 | -0.2886 | -0.0061 | 0 | 11.4043 |

The optimal k value was determined through 10-fold cross-validation, with the average accuracy results shown in Figure 3.

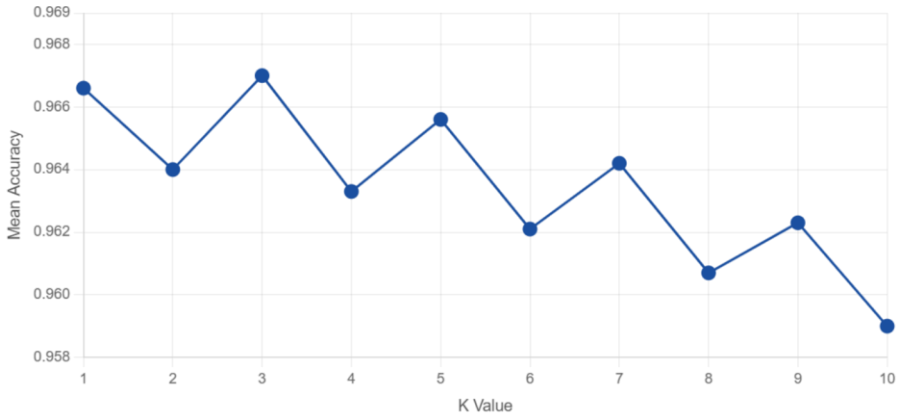


Figure 3 Graph of Average Accuracy Values of 10-fold cross-validation Against k Values

Figure 3 shows that k = 3 is the optimal parameter used in the K-NN classification process on the test data. The prediction results are then compared with the actual labels, as presented in Table 8 and Figure 4.

Table 8 Predicted Classes of Test Data Using the K-NN Method

| Label Toddler | k-value | Predicted Label | Actual Label |
|---|---|---|---|
| ID1 | 3 | 0 | 0 |
| ID2 | 3 | 0 | 0 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| ID24 | 3 | 1 | 1 |
| ID25 | 3 | 0 | 0 |
| **Label Toddler** | **k-value** | **Predicted Label** | **Actual Label** |
| ID26 | 3 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| ID44 | 3 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ |
| ID4068 | 3 | 0 | 0 |

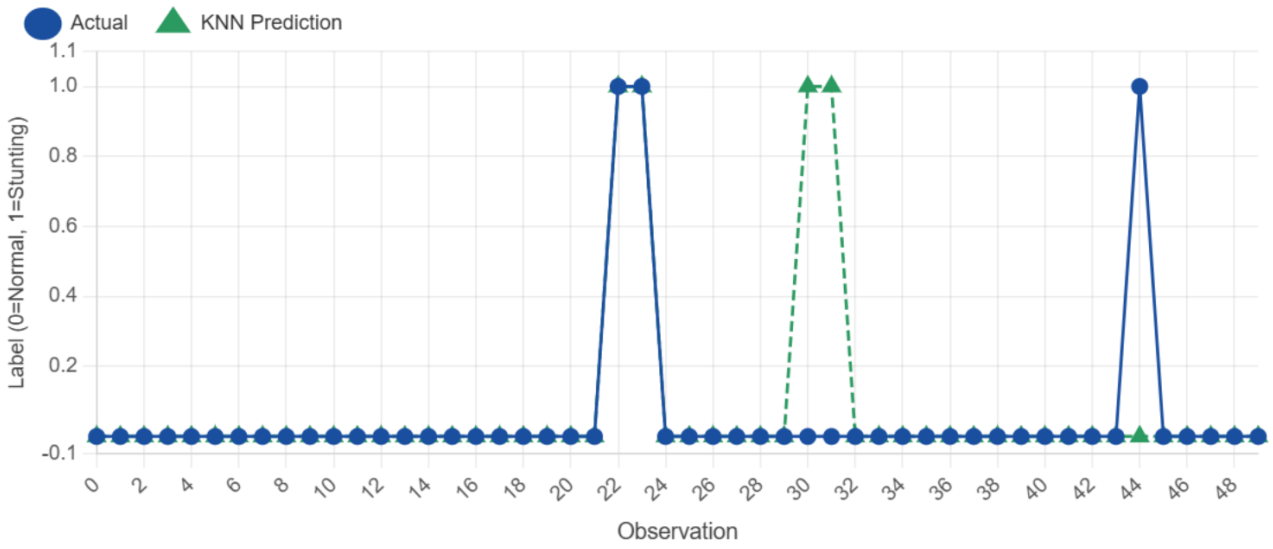


**Figure 4** Comparison Plot of Predicted Labels and Actual Labels Using the K-NN Method

Based on Table 8 and Figure 4, the classification of test data using the K-NN method with parameter k=3 shows a high degree of agreement between the predicted labels and actual labels, including in observations 24, 26, and 44, which are in the stunting class. The model is able to accurately classify non-stunting data and detect most stunting cases, although there are still some classification errors in the form of non-stunting data predicted as stunting. This pattern indicates that the K-NN method has a better ability to distinguish between stunting and non-stunting classes in the test data.

### 4.5 Evaluation of K-NN Method Performance

Next, the model's performance was evaluated using a confusion matrix presented in Table 9 and Figure 5.

**Table 9** K-NN Confusion Matrix

| Actual Class | Predicted Class | |
|---|---|---|
| | Non-stunting | Stunting |
| Non-stunting | 3717 | 16 |
| Stunting | 106 | 229 |


**Figure 5** K-NN Confusion Matrix Plot

Based on the confusion matrix in Table 9 and its visualization in Figure 5, the K-NN method with parameter k=3 shows good classification performance on the test data. The model was able to correctly classify 3717 non-stunted toddler data and 229 stunted toddler data, although there were still classification errors in the form of 16 non-stunted data predicted as stunted and 106 stunted data predicted as non-stunted. Overall, these results indicate that the K-NN method has a better ability to distinguish between stunting and non-stunting classes, particularly in detecting stunting cases.

### 4.6 Comparison of Naïve Bayes and K-NN Performance

Next, a performance comparison between the Naïve Bayes and K-NN methods was conducted by comparing the predicted labels and actual labels to assess the differences in the classification capabilities of the two methods, as presented in Table 10.

Table 10 Comparison of Actual Data and Prediction Results of Naïve Bayes and K-NN

| No | JK | Age | LiLA | Weight | Height | Original Label | NB | K-NN |
|---|---|---|---|---|---|---|---|---|
| ID1 | 0.931255 | -0.02161 | 0.025159 | 0.348028 | 0.370658 | 0 | 0 | 0 |
| ID2 | -1.07382 | 0.413197 | 0.100849 | 0.135814 | 0.302153 | 0 | 0 | 0 |
| ID3 | -1.07382 | -0.33218 | -0.12622 | -0.60694 | -0.22876 | 0 | 0 | 0 |
| ID4 | -1.07382 | -0.64275 | 0.025159 | -0.5362 | -0.57129 | 0 | 0 | 0 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| ID24 | 0.931255 | 0.910116 | -0.12622 | -0.00566 | 0.285027 | 1 | 0 | 1 |
| ID25 | 0.931255 | -1.13967 | 0.025159 | -0.88989 | -1.08508 | 0 | 0 | 0 |
| ID26 | -1.07382 | -1.13967 | -0.03539 | -0.96063 | -1.4276 | 1 | 0 | 1 |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| ID4067 | -1.07382 | -1.57448 | -0.2776 | -1.66801 | -1.6845 | 0 | 0 | 0 |
| ID4068 | -1.07382 | 0.848001 | 0.025159 | 0.312659 | 0.696058 | 0 | 0 | 0 |

Table 10 presents a comparison of actual labels and Gaussian Naïve Bayes and K-NN prediction results on standardized test data, showing the difference in classification performance between the two methods. Furthermore, the performance of the two methods is compared quantitatively using the metrics of accuracy, precision, recall, and f1-score. The comparison of the performance of the two methods is presented in Table 11, while the visualization of the comparison of evaluation metrics is shown in Figure 6.

Table 11 Classification Performance Comparison

| Classification Method | Correct Predictions | Classification Performance Measurement Values | | | |
|---|---|---|---|---|---|
| | | *Accuracy* | *Precision* | *Recall* | *F1-score* |
| Gaussian Naïve Bayes | 3.733 | ***91.76*** | ***0.00*** | ***0.00*** | ***0.00*** |
| K-NN | 3.946 | ***97.00*** | ***93.47*** | ***68.36*** | ***78.89*** |

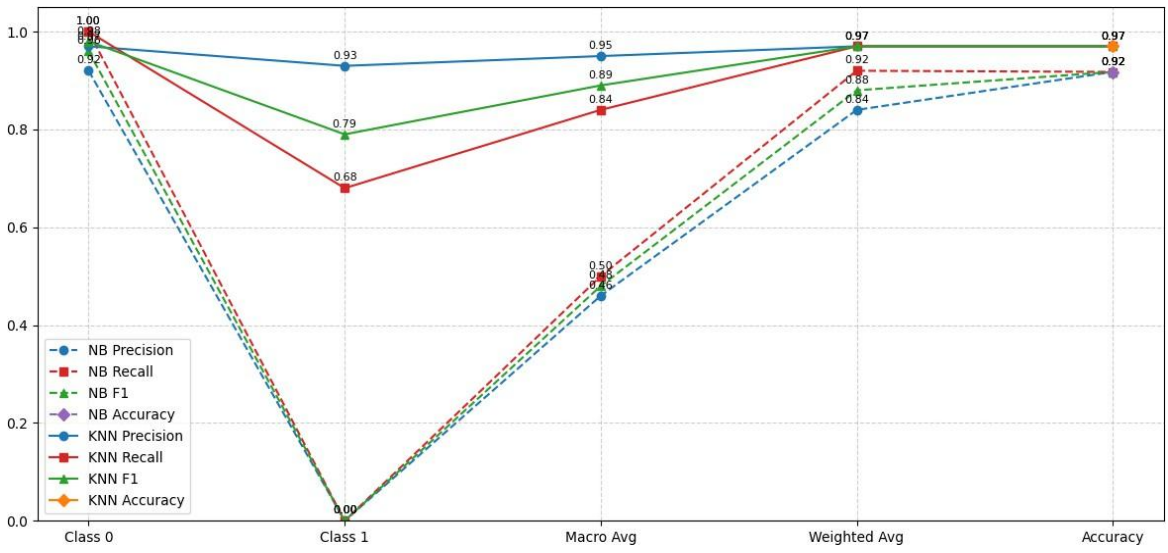


Figure 6 Classification Performance Comparison Chart

Table 11 shows a comparison of classification performance between the Gaussian Naïve Bayes and K-NN methods based on evaluation metrics. Gaussian Naïve Bayes produced an accuracy of 91.76%, but was unable to

detect the Stunting class, as indicated by precision, recall, and f1-score values of 0.00%. In contrast, the K-NN method achieved a higher accuracy of 97.00% and showed more balanced performance in classifying both classes, with precision, recall, and f1-score values of 93.47%, 68.36%, and 78.89%, respectively. A comparison of the performance of the two methods is visualized in Figure 6, which confirms the superiority of K-NN in identifying stunting status in toddlers.

## 5. CONCLUSIONS AND SUGGESTIONS

The results of evaluating the classification model for stunting nutritional status in toddlers in Takalar Regency using the Naïve Bayes and K-NN algorithms show that the Naïve Bayes method, despite achieving an accuracy rate of 91.76%, is not yet able to optimally identify stunting classes due to a tendency to bias towards the majority class. In contrast, the K-NN method with a parameter of k=3 showed more balanced performance with an accuracy rate of 97.00%, making it more effective in distinguishing between stunting and non-stunting classes. These results indicate that the K-NN method performs better than Naïve Bayes in classifying stunting in Takalar Regency.

Future studies are recommended to explore other classification algorithms and apply data balancing techniques such as SMOTE to improve stunting classification performance. In addition, expanding the dataset and incorporating relevant variables are encouraged to obtain more robust results.

## REFERENCES

[1] D. S. Wisdayani, "Perbandingan Algoritma K-Nearest Neighbor Dan Naive Bayes Untuk Klasifikasi Tingkat Keparahan Korban Kecelakaan Lalu Lintas Di Kabupaten Pati Jawa Tengah," Muhammadiyah University, Semarang, 2019.

[2] T. Prasetiya, I. Ali, C. L. Rohmat, and O. Nurdiawan, "Klasifikasi Status Stunting Balita Di Desa Slangit Menggunakan Metode K-Nearest Neighbor," *Informatics Educ. Prof. J. Informatics*, vol. 5, no. 1, p. 93, 2020, doi: 10.51211/itbi.v5i1.1431.

[3] I. A. A. Amra, "Students Performance Prediction Using KNN and Naïve Bayesian," pp. 909–913, 2017.

[4] M. Y. Titimeidara and W. Hadikurniawati, "Implementasi Metode Naïve Bayes Classifier Untuk Klasifikasi Status Gizi Stunting Pada Balita," *J. Ilm. Inform.*, vol. 9, no. 1, pp. 54–59, 2021, doi: 10.33884/jif.v9i01.3741.

[5] S. Zhang, "Learning k for kNN Classification," vol. 8, no. 3, 2026, doi: 10.1145/2990508.

[6] R. Setiawan and A. Triayudi, "Klasifikasi Status Gizi Balita Menggunakan Naïve Bayes dan K-Nearest Neighbor Berbasis Web," *MIB*, vol. 6, no. 2, pp. 777–785, 2022, doi: 10.30865/mib.v6i2.3566.

[7] B. Y. Pratama, "Personality Classification Based on Twitter Text Using Naive Bayes , KNN and SVM," pp. 170–174, 2015.

[8] B. M. Amsalu Feleke, "Magnitude of Stunting and Associated Factors Among 6-59 Months Old Children in Hossana Town, Southern Ethiopia," *J. Clin. Res. Bioeth.*, vol. 06, no. 01, pp. 4–11, 2015, doi: 10.4172/2155-9627.1000207.

[9] O. S. Bachri and R. M. H. Bhakti, "Penentuan Status Stunting Pada Anak Dengan Menggunakan Algoritma KNN," *J. Ilm. Intech Inf. Technol. J. UMUS*, vol. 3, no. 2, pp. 130–137, 2021.

[10] E. Silvana and J. L. Simbolon, *Determinan Stunting pada Balita*. Yogyakarta: Selat Media Partners, 2025.

[11] Bappenas, *Laporan Pembangunan Manusia dan Pemerataan Sosial 2024*. Jakarta: Kementerian PPN/Bappenas, 2024. [Online]. Available: https://bappenas.go.id

[12] *de Onis, M., Borghi, E., Arimond, M., Webb, P., Croft, T., Saha, K., ... & Flores-Ayala, R. (2019). Prevalence thresholds for wasting, overweight and stunting in children under 5 years.* Public Health Nutrition, 22*(1), 175–179.*

*[13] H. HS, N. Azmi, H. Hazriani, and Y. Yuyun, "Klasifikasi Status Gizi Balita Menggunakan Algoritma K-Nearest Neighbor (KNN)," in Prosiding SISFOTEK, 2023, pp. 313–318.*

*[14] J. Zeniarja, K. Widia, and R. R. Sani, "Penerapan algoritma Naive Bayes dan Forward Selection dalam Pengklasifikasian Status Gizi Stunting pada Puskesmas Pandanaran Semarang," JOINS (Journal Inf. Syst., vol. 5, no. 1, pp. 1–9, 2020.*

*[15] P. H. Kusnanto, STUNTING. Husada Mandiri Poltekkes Kemenkes Yogyakarta, 2018.*

*[16] J. W. & Sons, "Introduction to data mining," pp. 1–26, 2005.*

*[17] A. Srirahayu and L. S. Pribadie, "Review Paper Data Mining Klasifikasi Data Mining," vol. 14, no. April, 2023.*

*[18] M. Ontivero-ortega, A. Lage-castellanos, G. Valente, and R. Goebel, "Fast Gaussian Naïve Bayes for searchlight classification analysis Fast Gaussian Naïve Bayes for searchlight classi fi cation analysis," Neuroimage, vol. 163, no. 2017, pp. 471–479, 2026, doi: 10.1016/j.neuroimage.2017.09.001.*

*[19] Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification.* IEEE Transactions on Information Theory, 13*(1), 21–27.*

*[20] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression.* The American Statistician, 46*(3), 175–185.*

*[21] Rish, I. (2001). An empirical study of the Naive Bayes classifier.* IJCAI Workshop on Empirical Methods in Artificial Intelligence, 3*(22), 41–46.*

*[22] S. Sathyanarayanan and B. R. Tantri, "Confusion Matrix-Based Performance Evaluation Metrics," vol. 27, no. 4, 2024.*

*[23] E. Y. Reza, T. W. Widyaningsih, T. Informatika, F. Teknik, and U. T. Abeng, "Analisis untuk Memprediksi Kualitas Tumbuh Kembang Balita dengan Menerapkan Metode kNN dan Naïve Bayes Analysis to Predict the Quality of Toddler Growth by Implementing the kNN and Naïve Bayes Methods," vol. 13, pp. 1865–1875, 2024.*

*[24] World Health Organization. (2006).* WHO child growth standards: Length/height-for-age, weight-for-age, weight-for-length, weight-for-height and body mass index-for-age. *WHO Press.*

[25] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over-sampling technique. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953