

Classification of Stunting Prevalence Levels in Indonesia

Dewi Fadriani^{1*}, Dewi Retno Sari Saputro², Nughthoh Arfawi Kurdhi³

^{1,2,3} Department of Mathematics, Faculty of Mathematics and Natural Sciences, Universitas Sebelas Maret
Jl. Ir. Sutami No. 36 Kentingan, Surakarta, 57126, Indonesia

Corresponding Author.

*Email: dewifadriani1107@student.uns.ac.id

Abstract: Stunting is a problem of chronic malnutrition in children that affects their productivity in adults, cognitive development, and learning capacity. The high prevalence of stunting in Indonesia is caused by various health, social, and environmental factors that are different in each region, making it necessary to classify regions to identify areas with a high risk of stunting. Data with many variables present high-dimensionality issues that can reduce the accuracy of distance-based classification due to the curse of dimensionality. This study applies Autoencoder and K-Nearest Neighbors (KNN) on stunting data in Indonesia. The Autoencoder is used to reduce data dimensions through a process of reduction and reconstruction, thereby generating latent features for KNN input. The results show that the Autoencoder reduces 11 input variables to 4 latent features. KNN classification with $K=7$ categorizes provinces into 12 provinces with medium risk, 21 provinces with high risk, and 3 provinces with very high risk, with an accuracy of 75%. This study showed that dimensionality reduction using an Autoencoder improves data representation, enabling KNN to classify stunting more effectively.

Keywords: Dimension Reduction, Classification, Autoencoder, K-Nearest Neighbors, Stunting

© 2026 International Conference on Multidisciplinary Engagement. All rights reserved

1. INTRODUCTION

Child health and growth are important indicators of human resource development. A chronic nutritional problem that remains a global concern is stunting, a condition of growth failure in children caused by long-term undernutrition, a lack of sufficient stimulation, and repeated infections from early childhood [1]. According to the 2024 Indonesian Nutrition Status Survey (SSGI) [2], the prevalence of stunting in Indonesia is high at 19.8%. Stunting impacts cognitive development, learning capacity, and productivity in adults. Stunting is influenced by various interacting factors, such as pregnancy history, birth weight, socioeconomic status, and environmental sanitation [3]. These factors vary between regions, requiring classification to identify patterns and categorize regions based on stunting prevalence.

Classification is an important issue in the field of machine learning, that groups data into specific classes based on the characteristics of each type of data [4]. K-Nearest Neighbors (KNN) is a classification algorithm with a simple concept that is frequently applied to various classification problems involving diverse data characteristics [5]. KNN classifies data based on similarity to its nearest neighbors without requiring any specific distribution assumptions [6]. KNN is based on lazy learning [7] so its performance depends on the quality of data representation and the size of the dataset.

Research by Ukey et al. [8], shows that in high-dimensional data, KNN has a problem of curse of dimensionality, where accuracy decreases and computational complexity increases as the number of dimensions increases. This causes the measurement of distance between data points to not optimally represent the degree of similarity [9]. Therefore, dimension reduction techniques are needed to reduce the data to a lower-dimensional space. An artificial neural network-based dimension reduction method is the autoencoder, which learns latent representations through a process of compression and reconstruction, thereby producing more compact and informative features [10]. Reconstruction is the process of reforming the reduced data to resemble the input data.

Autoencoders are applied to both linear and nonlinear data for dimension reduction and feature extraction. This method is also used for data denoising and various applications in data analysis [11].

Previous studies have applied the KNN algorithm to classify stunting status. Syahril et al. [12] compared several classification algorithms and concluded that KNN resulted in the highest accuracy rate. Syahrani [13] combined KNN with Backward Elimination to improve model efficiency through the selection of more relevant features. However, these studies still focused on individual-level data and did not consider the issue of high feature dimensionality, which can affect the performance of the KNN algorithm. Thus, this study applies autoencoder-based dimensionality reduction and KNN classification to stunting data in Indonesia.

2. METHOD

This study is an applied research that applies autoencoder dimensionality reduction and KNN classification to stunting data in Indonesia. The methods used include research data, research methods, autoencoders, Mean Squared Error (MSE), K-Nearest Neighbors (KNN), and evaluation of results.

2.1 Research Data

This study used secondary data obtained from the official 2024 SSGI publication [2], with the 36 provinces of Indonesia as the unit of analysis. Based on WHO standards, the classification labels consist of stunting prevalence categories by region, classified into five categories: very low, low, medium, high, and very high. Research variables include social indicators, mother and child health, and environmental factors. All independent variables are represented in percentages that describe the health and environmental conditions in each province. The research variables used in this study are shown in Table 1.

Table 1. Research data variables

Variable	Description
y	Five categories of stunting prevalence
x_1	Percentage of mothers with an inter-pregnancy interval > 2 years
x_2	Percentage of pregnant mother attending prenatal classes
x_3	Percentage of pregnant mother taking blood supplement tablets
x_4	Percentage of newborns with low birth weight (<2500 grams)
x_5	Percentage of newborns with intrauterine growth restriction
x_6	Percentage of newborns exclusively breastfed
x_7	Percentage of children whose growth is monitored at community health posts
x_8	Percentage of children with a complete primary immunization schedule
x_9	Percentage of the population with National Health Insurance
x_{10}	Percentage of mothers using contraception after childbirth
x_{11}	Percentage of houses with access to safe drinking water

2.2 Research Methodology

The steps required to achieve the objectives of the study are as

1. Data preprocessing, consisting of cleaning, labeling, and scaling.
2. Splitting the data, 80% training data and 20% testing data.
3. Defining the encoder, latent layer, decoder, and activation functions for training the autoencoder model.
4. Calculating the loss value to measure the reconstruction error between the input and output data with Mean Squared Error (MSE).
5. Extracting latent features as data with lower dimensions.
6. Determining the optimal value of K using cross-validation.
7. Classification using KNN based on latent features and the optimal K .
8. Evaluating the classification results using a confusion matrix.

2.3 Autoencoder

An autoencoder is an artificial neural network consisting of two main components an encoder and a decoder [14]. The encoder processes data through the input layer and the first half of the hidden layer to generate a lower-dimensional representation. The decoder uses the second half of the hidden layer and the output layer to

reconstruct the input data [10]. The hidden layer is for learning important data patterns from the input [15]. The representation generated in this layer is called a latent representation, that is more concise and informative for the input during the classification phase. Based on the research by Park et al. [16], the autoencoder structure is shown in Figure 1.

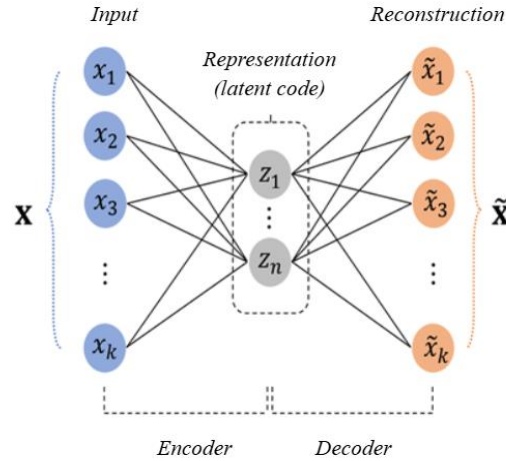


Figure 1. Autoencoder structure with a hidden layer

Generally, the encoder (ω) and decoder (β) are defined as

$$\omega: X \rightarrow F, \quad \beta: F \rightarrow X$$

where $x \in X \subseteq \mathbb{R}^d$ is the input data to the autoencoder and $z \in F \subseteq \mathbb{R}^p$ is the mapping result at the hidden layer. If the autoencoder has L hidden layers, the encoder is written as

$$\begin{aligned} h^{(1)} &= f(W^{(1)}x + b^{(1)}) \\ h^{(l)} &= f(W^{(l)}h^{(l-1)} + b^{(l)}), \quad l = 2, 3, \dots, L \\ z = h^{(L)} &= f(W^{(L)}h^{(L-1)} + b^{(L)}) \end{aligned} \quad (1)$$

where

- $h^{(l)}$: data representation at layer l ,
- $W^{(l)}$: weights,
- x : input data,
- $b^{(l)}$: bias,
- f : activation function,
- L : number of layers in the encoder, and
- z : latent representation output by the encoder.

The decoder reconstructs the latent representation (z) to obtain the reconstructed output (\tilde{x}) [17], written as

$$\tilde{x} = g(\text{Decoder}(z))$$

In autoencoder applications, the weights and biases are randomly initialized at the start of training using a specific distribution so that each weight element has a different initial value and the training process is stabilized. The autoencoder minimizes the reconstruction error by tuning the weights based on the obtained error [18].

2.4 Mean Squared Error (MSE)

The autencoder training objective is to minimize the loss function between the input (x) and the reconstructed output (\tilde{x}). The loss function used is the Mean Squared Error (MSE). MSE measures the average of the squared differences between the input values (actual) and the reconstructed values (predicted) [19], written as

$$MSE = \frac{1}{n} \sum_{i=1}^n (x_i - \tilde{x}_i)^2 \quad (2)$$

where n is the number of features the data, x_i is the value of the $-i$ feature the input, and \tilde{x}_i is the value of the $-i$ feature the reconstructed output. The smaller value of MSE, the better quality reconstruction produced by model, because the difference between the actual data and the reconstructed result is smaller [10].

2.5 K-Nearest Neighbors (KNN)

KNN is a classification algorithm that classifies new data based on its nearest distance to data points in a dataset [20]. Measurements of proximity or similarity between data points are performed using a distance function. According to research conducted by Pulugan *et al* [21], the distance most commonly used is the Euclidean distance. The Euclidean distance is the straight-line distance between two points in an n -dimensional space, calculated as the square root of the sum squares of the differences in coordinates across each dimension [21], written as

$$d(x, x') = \sqrt{\sum_{i=1}^n (x_i - x'_i)^2} \quad (3)$$

where

- x_i : the coordinates first point in the i dimension,
- x'_i : the coordinates second point in the i dimension, and
- n : the number of dimensions (number of features).

The choice number of neighbors (K) affects classification accuracy and introduces the risk of overfitting and underfitting so requiring an optimal value of K . The optimal K is selected using cross-validation, which involves testing several values of K on the training data and selecting the value that results in the best accuracy [22].

2.6 Evaluation of Results

Evaluation of classification results is important to measure the model's accuracy [23]. One method of evaluating results is the confusion matrix. A confusion matrix is a table that compares the model's predicted classification results with the actual classification labels [24]. The structure of the confusion matrix is shown in Table 2.

Table 2. Confusion matrix

	Actual Positive	Actual Negative
Predicted Positive	TP	FP
Predicted Negative	FN	TN

where

- TP (True Positive) : the number of correct positive predictions,
- TN (True Negative) : the number of correct negative predictions,
- FP (False Positive) : the number of incorrect positive predictions, and
- FN (False Negative) : the number of incorrect negative predictions.

Classification results are evaluated using accuracy calculated based on the confusion matrix. Accuracy measures the model's precision by comparing the number of correct predictions to the total number of test data [25], expressed as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

3. RESULT AND DISCUSSION

The research results consist of four sections data description, dimensionality reduction with autoencoder, classification with K-Nearest Neighbors (KNN), and evaluation of results.

3.1 Data Description

The stunting dataset in this study consists of 36 provinces in Indonesia with 11 independent variables ($x_1 - x_{11}$) and one target variable (y). Summary of each variable is shown in Table 3.

Table 3. Summary of variable statistics

Variabel	Mean	Min	Max
y	22.35	10.70	37.00
x_1	87.28	77.69	93.99
x_2	31.90	9.80	61.10
x_3	48.01	20.00	78.30
x_4	6.60	3.30	9.30
x_5	58.06	27.70	73.10
x_6	91.82	80.60	98.00
x_7	56.09	5.50	81.30
x_8	46.70	5.40	88.20
x_9	54.65	20.50	89.90
x_{10}	23.55	9.60	41.90
x_{11}	87.48	67.20	99.40

The variables have different ranges of values, indicating variations in health and environmental characteristics across provinces. The prevalence of stunting ranges from a minimum of 10.7% to a maximum of 37%, with an average of 22.35%. The variable percentage of pregnant women attending prenatal classes (x_2) has a relatively wide range of values, from 9.8% to 61.1%. Additionally, the variable percentage of households with access to safe drinking water (x_{11}) has a relatively high average value of 87.48%, although there are regional differences. The variation in variable values across provinces indicates that health conditions and environmental factors in Indonesia are diverse.

3.2 Dimensionality Reduction with Autoencoder

The encoder reduction data from an input with 11 variables to a 4-dimensional latent space with two hidden layers, as shown in Equation (1). The complete autoencoder architecture follows a symmetric structure of 11–32–16–4–16–32–11, where the encoder compresses the input from 11 to 4 dimensions through layers of 32 and 16 neurons, while the decoder reconstructs the data back to the original dimension through layers of 16 and 32 neurons. The selection of this architecture is intended to capture complex nonlinear relationships while maintaining a balance between model capacity and generalization.

The choice of a 4-dimensional latent representation is based on experimental evaluation, in which several latent dimension configurations were tested. The results indicate that four latent dimensions provide the best trade-off between reconstruction accuracy and dimensionality reduction. Lower-dimensional representations led to higher reconstruction loss, indicating the loss of important information, whereas higher-dimensional representations did not yield significant performance improvements but increased model complexity. Therefore, four latent features were selected as the optimal representation.

The ReLU (Rectified Linear Unit) activation function was used in each hidden layer so the model could learn nonlinear relationships between variables. The model is trained for 500 epochs with a batch size of 8. During the training process, an early stopping technique is applied to automatically halt training if there is no improvement in the loss value for a certain number of epochs to prevent overfitting. Based on the training process, training loss and validation loss values are obtained at each epoch and used to evaluate the model's performance during training. The loss values were calculated using MSE in Equation (2). A visualization of the changes in training loss and validation loss during the training process is shown in Figure 2.

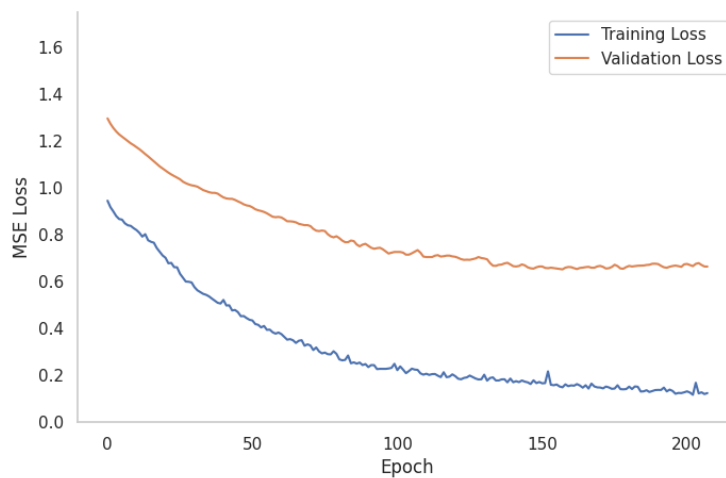


Figure 2. Comparison graph of training loss and validation loss

The training loss value decreased progressively until it reached a stabilized condition at the 208 epoch with a value of 0.11. The validation loss value also showed a similar trend and reached a stabilized condition at the same epoch with a value of 0.65. The similarity between the training loss and validation loss curves indicates that the model is not overfitting.

The curve stabilized after the 208 epoch, indicates that the training process had reached an optimal point, such that additional epochs do not reduce the loss values. This suggests that the four latent dimensions generated efficiently represent the main information from the 11 input variables. The results of the dimension reduction process are shown in Table 3 in the form of four latent features z_1 , z_2 , z_3 , and z_4 , representing the main characteristics of each province.

Table 3. Latent features resulting from dimension reduction

Province	z_1	z_2	z_3	z_4
Aceh	-3.234336	0.867539	3.347355	-0.568142
Sumatra Utara	-0.051170	-0.934592	0.593853	1.164470
Sumatra Barat	-2.302355	-0.274886	0.447041	-0.288662
Riau	-0.743851	0.917771	0.606296	0.730470
Jambi	-1.055197	3.685398	1.981185	-1.221900

Every province in Table 3 has different latent feature values, indicating that the autoencoder maps the characteristics of each province into a lower-dimensional latent space. The z_3 value for Aceh province is higher than that of other provinces, while Jambi province has a higher z_2 value than other provinces. These differences in values indicate variations in the characteristics of each province and show that the dimension reduction process retains the main characteristics of the data. The lower-dimensional latent representation reduces the impact of the curse of dimensionality, making distance calculations between data points more representative.

3.3 Classification with KNN

Classification was performed using the KNN algorithm with latent features z_1 , z_2 , z_3 , and z_4 as input. The optimal value of K was determined using a 10-fold cross-validation technique to avoid bias resulting from subjective parameter selection. This process consisted of testing several values of K within a specific range, with each value of K tested on 10 different data subsets. The average accuracy for each value of K is shown in Figure 3.

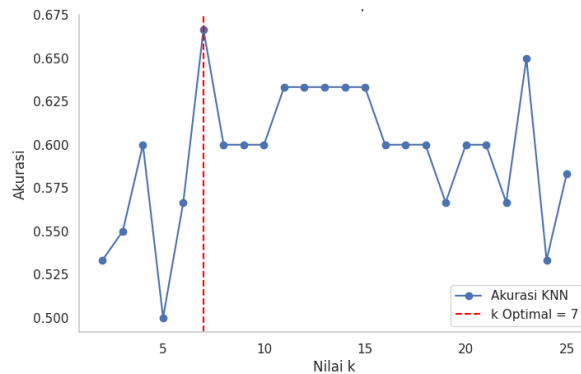


Figure 3. Testing the accuracy of various K values

The test results show that a K value of 7 results in the highest average accuracy compared to other K values, giving a score of 0.667 or 66.7%. This indicates that using the seven nearest neighbors results in a better balance between model complexity and prediction stability. Classification using $K = 7$ and distance calculations based on Equation (3) resulted in predictions of stunting prevalence categories for each province. The classification results show that provinces in Indonesia are divided into three categories, medium, high, and very high. The medium category consists of 12 provinces (33.3%), the high category consists of 21 provinces (58.35%), and the very high category consists of 3 provinces (8.3%), as shown in Figure 4.

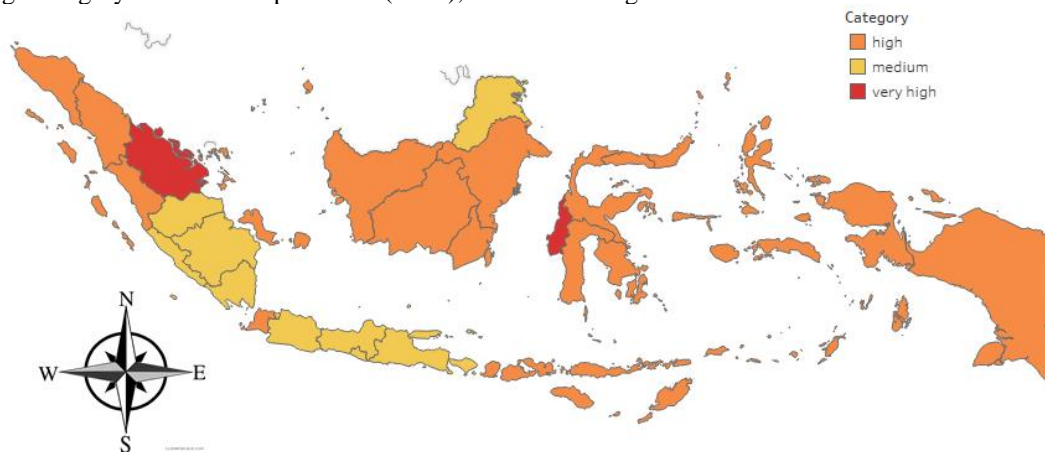


Figure 4. Visualization of classification results

The very high category (red) is observed in three provinces, namely Riau, West Sulawesi, and Southwest Papua. Based on geographic region, the high categories (orange) is predominantly found in central and eastern Indonesia, such as Kalimantan, Sulawesi, Maluku, and Papua. Meanwhile, several provinces on the islands of Java and Sumatra tend to fall into the medium category (yellow).

This spatial pattern suggests that provinces with relatively better access to healthcare services, infrastructure, and economic resources—such as those in Java and parts of Sumatra—tend to have lower stunting prevalence and are classified into the medium category. In contrast, provinces in eastern Indonesia, which are often characterized by limited healthcare access, geographical constraints, and disparities in socio-economic conditions, are more likely to fall into the high and very high categories. The classification results also indicate that provinces grouped within the same category share similar characteristics in the latent feature space generated by the autoencoder, reflecting underlying similarities in key stunting-related indicators.

From a practical perspective, these findings have important implications for public health policy and stunting intervention programs. Provinces in the very high category require priority attention through more intensive and targeted interventions, such as improving maternal and child nutrition, expanding access to healthcare services, and strengthening monitoring programs for child growth. Meanwhile, provinces in the high category may benefit from preventive strategies and program optimization, while those in the medium category can focus on maintaining and improving existing interventions. Therefore, this classification approach can

support policymakers in designing more targeted, efficient, and region-specific strategies to reduce stunting prevalence in Indonesia.

3.4 Evaluation of Results

The objective of the evaluation is to determine the effectiveness of the autoencoder in dimension reduction on classification performance. After applied dimensionality reduction, classification with KNN obtained an accuracy of 0.750 or 75%, calculated based on Equation (4). This value indicates that most provinces can be classified according to their actual stunting prevalence categories. The confusion matrix of the classification results on the testing data using latent features is shown in Figure 5.

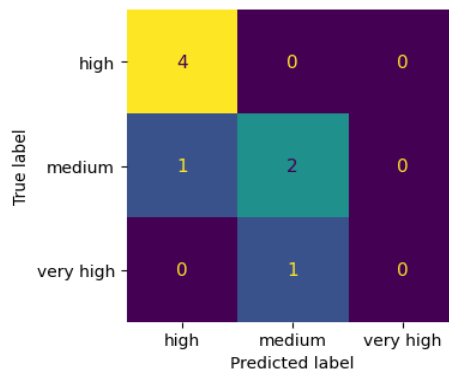


Figure 5. Confusion matrix of classification results

Based on the confusion matrix, the model correctly classified 6 out of 8 testing data points. There were 4 high-class data that were correctly predicted as high and 2 medium-class data that were correctly predicted as medium. However, there were still classification error, where 1 medium-class data was predicted as high and 1 very high-class data was predicted as medium. These results indicate that dimensionality reduction simplifies the data into more concise latent features while maintaining important information that enhances classification performance. However, these errors suggest that the model still has difficulty differentiating classes with similar characteristics.

4. CONCLUSION

Based on the research results, the autoencoder reduced 11 input variables to 4 latent features with a training loss value 0.11. The latent features were used as input for the KNN model with an optimal parameter of $K = 7$. The classification results showed that the provinces in Indonesia fell into three categories, with 12 provinces classified as medium, 21 as high, and 3 as very high. These findings can support policymakers in designing more targeted and region-specific stunting intervention strategies in Indonesia. Evaluation of the model using a confusion matrix resulted in 75% accuracy on the test data, with 6 out of 8 test data correctly classified. However, there are still classification errors in classes with relatively similar characteristics. Further research is recommended to optimize the autoencoder architecture and conduct testing on a dataset with a more extensive coverage.

ACKNOWLEDGMENTS

The Author would like to thank all parties who have contributed to this study, particularly in the development of mathematical formulations, the selection of appropriate statistical models, and the estimation and validation processes. The scientific discussions conducted further strengthened the methodological validity of this study. Additionally, the author appreciates the support in providing data and computational resources that enabled the efficient implementation of algorithms and numerical simulations, which resulted in reliable and reproducible

REFERENCES

[1] W. Yanti Sihotang *et al.*, “Determinants of stunting in children under five: a scoping review,” 2023.

- [2] Kemenkes RI, “SSGI 2024 DALAM ANGKA,” Jakarta, 2025.
- [3] A. N. Widya, B. R. Samudro, and E. Gravitiani, “Economics Development Analysis Journal Stunting In Java Island: Spatial and Risk Factor Analysis,” / *Economics Development Analysis Journal*, vol. 13, no. 2, 2024, [Online]. Available: <http://journal.unnes.ac.id/sju/index.php/edaj>
- [4] A. A. Soofi and A. Awan, “Classification Techniques in Machine Learning: Applications and Issues,” *Journal of Basic & Applied Sciences*, vol. 13, pp. 459–465, 2017.
- [5] J. Salvador-Meneses, Z. Ruiz-Chavez, and J. Garcia-Rodriguez, “Compressed kNN: K-nearest neighbors with data compression,” *Entropy*, vol. 21, no. 3, Mar. 2019, doi: 10.3390/e21030234.
- [6] “Review of Literature on Improving the KNN Algorithm,” *Transactions on Machine Learning and Artificial Intelligence*, vol. 11, no. 3, Jun. 2023, doi: 10.14738/tecs.113.14768.
- [7] P. Cunningham and S. J. Delany, “k-Nearest Neighbour Classifiers: 2nd Edition (with Python examples),” Apr. 2020, doi: 10.1145/3459665.
- [8] N. Ukey, Z. Yang, B. Li, G. Zhang, Y. Hu, and W. Zhang, “Survey on Exact kNN Queries over High-Dimensional Data Space,” Jan. 01, 2023, *MDPI*. doi: 10.3390/s23020629.
- [9] A. A. Amer, S. D. Ravana, and R. A. A. Habeeb, “Effective k-nearest neighbor models for data classification enhancement,” *J. Big Data*, vol. 12, no. 1, Dec. 2025, doi: 10.1186/s40537-025-01137-2.
- [10] S. Chen and W. Guo, “Auto-Encoders in Deep Learning—A Review with New Perspectives,” Apr. 01, 2023, *MDPI*. doi: 10.3390/math11081777.
- [11] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, and Y. Xu, “Autoencoders and their applications in machine learning: a survey,” *Artif. Intell. Rev.*, vol. 57, no. 2, Feb. 2024, doi: 10.1007/s10462-023-10662-6.
- [12] S. Syahrial, R. Ilham, Z. F. Asikin, and St. S. I. Nurdin, “Stunting Classification in Children’s Measurement Data Using Machine Learning Models,” *Journal La Multiapp*, vol. 3, no. 2, pp. 52–60, Mar. 2022, doi: 10.37899/journalmultiapp.v3i2.614.
- [13] Syahrani Lonanf, “Klasifikasi Status Stunting Pada Balita Menggunakan K-Nearest Neighbor Dengan Feature Selection Backward Elimination,” *Jurnal Media Informatika Budidarma*, vol. 5, pp. 999–999, Mar. 2021, doi: 10.30865/mib.v5i1.2293.
- [14] F. J. Pulgar, F. Charte, A. J. Rivera, and M. J. Del Jesus, “AEkNN: An autoencoder kNN–based classifier with built-in dimensionality reduction,” *International Journal of Computational Intelligence Systems*, vol. 12, no. 1, pp. 436–452, Nov. 2018, doi: 10.2991/ijcis.2018.125905686.
- [15] K. Berahmand, F. Daneshfar, E. S. Salehi, Y. Li, and Y. Xu, “Autoencoders and their applications in machine learning: a survey,” *Artif. Intell. Rev.*, vol. 57, no. 2, Feb. 2024, doi: 10.1007/s10462-023-10662-6.
- [16] C. Park, J. Lee, Y. Kim, J. G. Park, H. Kim, and D. Hong, “An Enhanced AI-Based Network Intrusion Detection System Using Generative Adversarial Networks,” *IEEE Internet Things J.*, vol. 10, no. 3, pp. 2330–2345, Feb. 2023, doi: 10.1109/JIOT.2022.3211346.
- [17] J. P. Roche, O. Niggemann, and J. Friebe, “Using Autoencoders and Automatic Differentiation to Reconstruct Missing Variables in a Set of Time Series,” *SN Comput. Sci.*, vol. 6, no. 4, Apr. 2025, doi: 10.1007/s42979-025-03798-5.
- [18] M. Abu, B. Siddique, S. Sakib, and A. Rahman, “Performance Analysis of Deep Autoencoder and NCA Dimensionality Reduction Techniques with KNN, ENN and SVM Classifiers.”
- [19] M. Wildi, “Sign Accuracy, Mean-Squared Error and the Rate of Zero Crossings: a Generalized Forecast Approach,” Jan. 2026, [Online]. Available: <http://arxiv.org/abs/2601.06547>
- [20] M. Garouani, A. Ahmad, M. Bouneffa, and M. Hamlich, “Autoencoder-kNN meta-model based data characterization approach for an automated selection of AI algorithms,” *J. Big Data*, vol. 10, no. 1, Dec. 2023, doi: 10.1186/s40537-023-00687-7.
- [21] A. F. Pulungan, M. Zarlis, and S. Suwilo, “Analysis of Braycurtis, Canberra and Euclidean Distance in KNN Algorithm,” *Sinkron*, vol. 4, no. 1, p. 74, Sep. 2019, doi: 10.33395/sinkron.v4i1.10207.
- [22] R. Rahim and A. S. Ahmar, “Cross-Validation and Validation Set Methods for Choosing K in KNN Algorithm for Healthcare Case Study,” *JINAV: Journal of Information and Visualization*, vol. 3, no. 1, pp. 57–61, Jul. 2022, doi: 10.35877/454ri.jinav1557.
- [23] S. Uddin, I. Haque, H. Lu, M. A. Moni, and E. Gide, “Comparative performance analysis of K-nearest neighbour (KNN) algorithm and its different variants for disease prediction,” *Sci. Rep.*, vol. 12, no. 1, Dec. 2022, doi: 10.1038/s41598-022-10358-x.
- [24] S. Sathyanarayanan, “Confusion Matrix-Based Performance Evaluation Metrics,” *African Journal of Biomedical Research*, pp. 4023–4031, Nov. 2024, doi: 10.53555/ajbr.v27i4s.4345.
- [25] G. Zeng, “Invariance Properties and Evaluation Metrics Derived from the Confusion Matrix in Multiclass Classification,” *Mathematics*, vol. 13, no. 16, Aug. 2025, doi: 10.3390/math13162609.