JURNAL INFORMATIKA, Vol.12 No.1 April 2025, Page 15-26
ISSN (Print) 2355-6579 | ISSN (Online) 2528-2247
DOI: https://doi.org/10.31294/inf.v12i1.22299

Optimization of Human Development Index in Indonesia Using Decision Tree C4.5, Support Vector Machine Algorithm, K-Nearest Neighbors, Naïve Bayes, and Extreme Gradient Boosting

Ilham Ramadhan[1], Budiman[2*], Nur Alamsyah[3]

[1,2,3] Universitas Informatika dan Bisnis Indonesia
Jl. Soekarno Hatta No.643 Bandung, Jawa Barat 40285, Indonesia

Correspondence e-mail: budiman@unibi.ac.id

Submission:	Revision:	Acceptance:	Available Online:
04-01-2024	17-12-2024	06-02-2025	04-03-2025

Abstract

The Human Development Index (HDI) is a measure of human development achievement based on quality of life indicators such as Life Expectancy (LE), Mean Years of Schooling (MYS), Expected Years of Schooling (EYS), and Adjusted Per Capita Expenditure (AECE). HDI describes how people access development outcomes through income, health, and education. The determination of development programs implemented by local governments must be based on district/city priorities based on their HDI categories and must be right on target. Therefore, a decision system is needed that can accurately determine the HDI category in each district/city in Indonesia, using machine learning models such as Decision Tree C4.5, Support Vector Machine (SVM), K-Nearest Neighbors (KNN), Naïve Bayes, and Extreme Gradient Boosting (XGBoost). Machine learning models will be used to classify the HDI in Indonesia in 2022 and determine the performance of the most optimal model in classification. This research uses the CRISP-DM method with secondary data from the Central Statistics Agency (BPS) as much as 548 data. The analysis results show that the Decision Tree C4.5 models have an accuracy of 0.86, KNN of 0.95, Naïve Bayes of 0.90, XGBoost of 0.93, and SVM provides the most optimal results with an accuracy of 0.97. UHH, RLS, and HLS variables significantly influence changes in HDI values in Indonesian regions based on the Chi-square, Pearson Correlation, Spearman, and Kendal test results.

Keywords: Human Development Index, Machine Learning, SVM Algorithm

1. Introduction

The Human Development Index (HDI) describes how people access development outcomes through income, health, education, etc. The HDI has three essential components: longevity and healthy living, knowledge, and a decent standard of living. (Badan Pusat Statistik, 2022). Indonesia continues to adjust the methodology with advances made by the United Nations Development Programme (UNDP). Since 2015 until now, Indonesia has used the latest method developed by UNDP in 2014 (UNDP, 2022). HDI can also be one indicator that can determine a region's rank or level of development. For Indonesia, HDI is strategic data because it measures government performance and is also used as one of the allocators for determining the General Allocation Fund (DAU) (Badan Pusat Statistik, 2022).

According to the standards established by UNDP, Indonesia has adapted the dimension of longevity and healthy living, represented by the Life Expectancy (UHH) indicator, which reflects the average number of years a person is expected to live. This indicator is based on the understanding that longevity is influenced by nutrition and health (Nurhalizah & Sitompul, 2022). The indicator method uses an indirect estimation approach and is standardised to UNDP calculation standards (Badan Pusat Statistik, 2022). The knowledge dimension is represented by the indicators of Expected Years of Schooling (HLS) and Average Years of Schooling (RLS) (Nurhalizah & Sitompul, 2022). These two indicators represent the community's ability to access formal education, which describes the community's opportunity to pursue formal education, while RLS describes.

In contrast, the level of education of a region's stock of human capital. The calculation of these two indicators uses data from the National Socio-Economic Survey (Susenas), data on residential education students from the Ministry of Religious Affairs, and the results of sectoral data inventories in districts/cities (Badan Pusat Statistik, 2022). The third dimmer Capita Expenditure (OECD) indicator. PVRD represents the third dimension of decent living standards, an indicator that positively impacts the HDI calculation results, reflecting the income and welfare of the community. Adjusted actual per capita expenditure is calculated using data from the National Socio-Economic Survey (Susenas) consumption module, consumer price index, and non-food commodity prices from the consumer price survey (Badan Pusat Statistik, 2022).

According to the Indonesian Central Bureau of Statistics (BPS), based on an international scale, human development achievements are divided into four categories, namely: 1). Very high HDI ≥ 80; 2). High HDI 70 ≤ HDI < 80; 3). Medium HDI 60≤ HDI < 70; and 4). Low HDI < 60. HDI is calculated as the geometric mean of the health, education, and expenditure indices (Badan Pusat Statistik, 2022). Indonesia is a vast country with social, economic and cultural diversity. The circumstances or situation influences the high and low value of HDI in a region in that region. Therefore, a decision system is needed to accurately determine the classification of HDI categories in each district/city in Indonesia.

Data mining involves extracting useful information and patterns from large data sets and databases (Bardab et al., 2021). One of the main tasks in data mining is classification, which involves grouping data into predefined classes or categories based on certain features or attributes (Verdhan, 2020). Classification focuses on constructing models that automatically categorise data into classes or groups based on input features (Kumar & Jain, 2020). Generally, data mining and classification work together to extract knowledge from large data sets and make them more accessible and useful for decision-making (Verdhan, 2020).

Classification analysis is an essential statistical methodology for understanding and predicting the HDI level of a region. Using this method, it is possible to group different regions into categories based on the HDI level of the area, such as very high, high, medium, or low. In this context, classification analysis helps identify patterns and factors that influence HDI. The performance comparison of classification models helps determine the most optimal model in classifying the Human Development Index dataset. The models used for comparison are Decision Tree C4.5, Extreme Gradient Boosting (XGBoost), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naïve Bayes Nurafidah et al., 2023).

Previous research have discussed the classification of the Human Development Index. In Isnaini Arumnisaa & Arie Wahyu Wijayanto's research, they compare Ensemble Learning methods: Random Forest, Support Vector Machine and Adaptive Boosting by applying 10-fold cross-validation to divide the dataset and SMOTE to resample data if there is data imbalance in performing Human Development Index classification accuracy. The results obtained from the analysis show that the Random Forest model has the best performance, with an accuracy of 85.23% (Arumnisaa & Wijayanto, 2023). Research by Nurafidah, Kris Suryowati & Maria Titah Jatipaningrum compared the K-Nearest Neighbors and Random Forest methods in classifying the Human Development Index. The results obtained in the comparison showed that classification using the K-Nearest Neighbors model obtained an accuracy value of 96.12% (Nurafidah et al., 2023). The research of Intan Kemalaa & Arie Wahyu Wijayantoa compared the performance of Bagging and Non-Ensemble Machine Learning methods by applying 10-fold Cross Validation and SMOTE methods to handle imbalance data on the Human Development Index in classification. The results obtained in this comparison show that predictions using the Random Forest model are the best, with an accuracy value of 95.14% (Kemala & Wijayanto, 2021).

This research objectives to develop a computational model that can classify the Human Development Index (HDI) in provinces, districts, and cities in Indonesia in 2022 into four classes, namely very high, high, medium, and low, with data mining techniques using the C4.5 Decision Tree model, Extreme Gradient Boosting (XGBoost), Support Vector Machine (SVM), K-Nearest Neighbors (KNN), and Naïve Bayes. This research will also compare the performance of each classification model used. The data mining framework used in this research is CRISP-DM.

2. Research Methods

This research uses secondary data from the Indonesian Central Bureau of Statistics published in 2022, which can be accessed through the official website www.bps.go.id. The data used is the Human Development Index in provinces / districts / cities throughout Indonesia in 2022 of 553 records.

Research stages are needed to achieve the goals that have been set. The research method is based on the Cross-Industry Standard Process for Data Mining (CRISP-DM) model, which provides an overview of the life cycle of data mining projects. This model contains the project stages, related tasks, and their results (Wirth & Hipp, 2000), (Huber et al., 2019), (Budiman & Niqotaini, 2021), (Budiman & Parama Yoga, 2023). The life cycle of a data mining project is divided into six stages, shown in Figure 1.

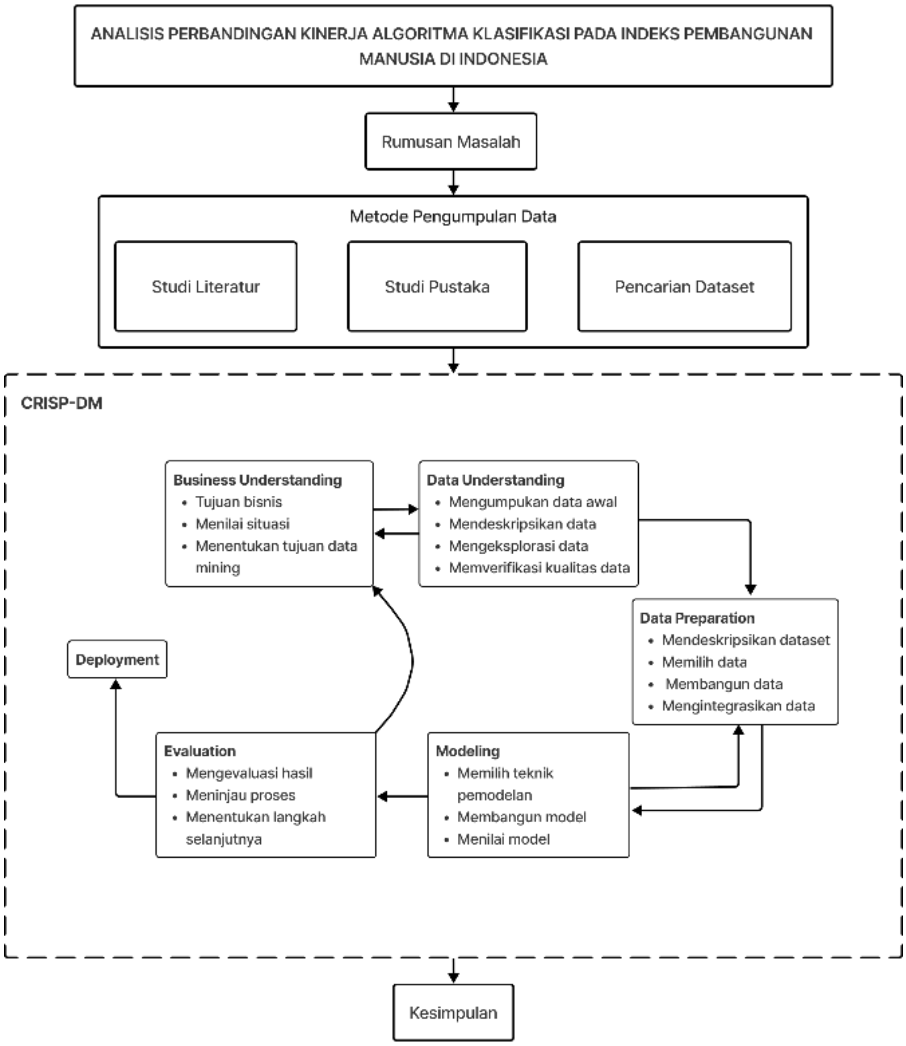

Figure 1. Research Flow

a. Determining the Research Topic

The first step in any research is to set a clear title and describe the main focus of the study. The title must be clear and able to tell the scope and purpose of the research.

b. Problem identification

The problem identification stage is carried out to find existing problems. The process of determining the issues in this study, conducting an initial analysis and being able to find research objectives.

c. Data collection method

In this phase, the researcher's data collection method is a literature study to deepen understanding of the topic, explore previous research, and provide a solid theoretical basis for this research. A literature study was conducted to identify information relevant to the problem under study. Relevant literature sources have been reviewed to support the theoretical basis and research approach. Data was selected in the dataset search stage based on specific criteria and relevance to the research objectives. The data obtained was found on the official website of the Indonesian Central Bureau of Statistics for 2022.

d. Cross-Industry Standard Process for Data Mining (CRISP-DM)

CRISP-DM is the de-facto standard and an industry-independent process model for applying data mining projects (Schröer et al., 2021). The business understanding phase involves understanding the business objectives, assessing the situation, and setting the data mining objectives. Based on this understanding, knowledge is implemented to define data mining problems. Next, strategies are set to achieve those goals.

The Data Understanding Phase includes initial data collection, description, exploration, and quality checks. It defines exciting parts of the data that can be used to reveal hidden information.

The Data Preparation Phase includes dataset description, selection, integration, and quality checks. This process also provides for the selection of tables, records, and data attributes, including data cleaning and transformation.

The Modeling Phase includes selecting modelling methods, model building, and model evaluation. This research uses the test percentage split method to create a model from the dataset, which divides training and testing data with the developed model.

The evaluation phase is conducted when the model is created with the expectation of being of good quality from a data analysis point of view. The steps to be taken are to evaluate the results, review the process, and decide on the next steps. It involves assessing the effectiveness and quality of the data model before use, deciding whether the data model can be used, and determining whether it achieves the goals set based on the initial business stages. The goal is to identify potential business problems that may not have been detected.

The deployment phase involves compiling and presenting the insights and information gained in a format that users can use.

e. Conclusion

This stage is a summary of the results of the overall data analysis, which includes evaluating the performance of each model and determining the most optimal classification model of the five models as compared.

2.1. Decision Tree C4.5

The C4.5 Decision Tree model is one method of making a Decision Tree based on existing training data. Decision Trees result from the entropy and information gain calculation process, after repeated calculations, until all tree attributes have a class and the calculation process can no longer be done (Singh et al., 2019). C4.5 is an extension of ID3 and is used for classification problems. It is also a strong choice because it can handle categorical and continuous variables (Kumar & Jain, 2020). To calculate the gain using the formula shown in equation 1 (Budiman et al., 2020):

$$Gian(S, A) = Entropy(s) - \sum_{i=1}^{n} -\frac{|S_i|}{|S|} \times Entropy(S_i) \quad (1)$$

Where, S set of cases, A attributes, n is the number of partitions of attribute A, $|S_i|$ the number of instances of the to-i, and$|S|$ Is the number of cases of S. At the same time, the calculation of entropy value is shown in Equation 2.

$$Entropy(S) = \sum_{i=1}^{n} -p_i \times \log_2 p_i \quad (2)$$

Where S set of cases, A is the feature, n the sum of S partitions, and p_i Is the proportion of S_i To S.

2.2. Extreme Gradient Boosting (XGBoost)

The ensemble learning method starts by forming a learning set where the machine is trained on a subset of training objects at each learning. If the learning model underperforms or has errors in data classification, more emphasis can be given to the model by increasing the weights of the misclassified classes. XGBoost is a software model and system focusing on residual gradient boosting. It improves the performance of standard gradient boosting by adding parameters, thus providing highly accurate predictions, and can be implemented quickly thanks to its parallel processing capabilities. XGBoost also effectively overcomes problems related to overfitting by applying regularisation techniques (Kumar & Jain, 2020).

With the application of parallel and distributed computing, the speed of the learning and modelling process becomes faster; the process is reflected in Equation 3 (Mo et al., 2019).

$$\hat{y}_t^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \quad (3)$$

Where $\hat{y}$ the final tree model, $y_i^{(t-1)}$ the previously generated tree model, $f_t(x_i)$ the newly generated tree model, and t the number of base tree models. The XGBoost model can accept input as continuous or discrete variables, but the output variable must be discrete, including binary variables (Alamsyah et al., 2024).

2.3. Support Vector Machine (SVM)

Support Vector Machine is a classification model that utilises a dual classifier hyperplane with maximum margins in the feature space. This hyperplane serves as the optimal boundary between classes in the classification problem. SVM was initially derived from linear classifiers and applied to linear separation problems where all training examples can be separated by a hyperplane in their respective space. Using kernel functions in SVM allows the calculation of the similarity or inner product between two vectors in the feature space without having to map the vectors back to the original space. The kernel function is built based on the inner product of two vectors in the feature space. Two vectorsx_1 Andx_2 are vectors in the original space, and $\Phi(x_1)$, $\Phi(x_2)$ are vectors in the feature space. So, SVM can handle nonlinear classification problems using liner separation in the feature space. The inner product of two vectors in the feature space becomes the basis for constructing the kernel function in SVM. The general equation of SVM, which is the kernel function, is shown in equation 4 (Jo, 2021).

$$y = sign\left(b + \sum_{i=1}^{N} \alpha_i \Phi(x_1)\Phi(x)\right) = sign\left(b + \sum_{i=1}^{N} \alpha_i K(x_i, x)\right) \quad (4)$$

Where y Output or prediction of the function, $sign()$ signum function, b bias or constant, N Sum of total samples or several support vectors, α_i Coefficient of each sample, $K(x_i, x)$ Kernel function measures the similarity between data pointsx_i and x in the feature space, $\alpha_i \Phi(x_1)$ and $\Phi(x)$ Representation of data pointsx_i and x in the higher feature space.

2.4. K-Nearest Neighbors (KNN)

K-Nearest Neighbors is a nonparametric model that does not build a model and performs classification based on the majority vote of the nearest neighbors. It is suitable for datasets where the relationship between the attribute and the target class is complex, but the elements within the class tend to be homogeneous. However, it is less suitable for dirty datasets or when the target class is unclear and confusing when determining the majority of votes. KNN can also be used for regression, predicting the value of a continuous variable by averaging the nearest neighbors (Kumar & Jain, 2020). The KNN model is the simplest classification model, assuming similarity between adjacent objects, with the computational process using the Euclidean distance method (Sakarkar et al., 2021), (Majumdar, 2023). The formula of Euclidean distance is shown in equation 5 (Id, 2021).

$$d(p,q) = \sqrt{(p_1 - q_1)^2 + (p_n - q_n)^2} = \sqrt{\sum_{i=1}^{n} (p_i - q_i)^2} \quad (5)$$

Where $d(p,q)$ distance between objectsp_1 and p_2, p_1 testing data, q_1 Training data, and n Number of independent variables. The steps of the KNN model are as follows (Anam et al., 2021):

1. Determine the number of K (nearest neighbors).
2. Calculate the Euclidean distance of objects against testing data to training data with equation 5.
3. Sort the results in number 2 in order from the smallest to the most significant value.
4. Collect categories of the dependent variable (Y) and classify Nearest Neighbors based on the k value.
5. We are using the majority nearest neighbor category to predict the new object.

2.5. Naïve Bayes

The Naïve Bayes model is a machine learning model based on the principle of Bayes' Theorem. The term "Naïve" refers to the assumption that the occurrence of a feature does not depend on the occurrence of other features. Meanwhile, "Bayes" refers to the use of the principle of Bayes' Theorem in calculating the probability of a hypothesis with prior knowledge. Bayes' Theorem, also known as Bayes' Rule, is based on conditional probability. The formula for Bayes' Theorem is shown in equation 6 (Sakarkar et al., 2021).

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (6)$$

The Naïve Bayes model has variations that can be used for binary and multi-class classification, with implementations available in the Scikit-Learn library. One of these is Gaussian Naïve Bayes, which is similar to Naïve Bayes but can handle continuous data using a Gaussian distribution to calculate probabilities, assuming the data is usually distributed. This makes it suitable for classification problems with constant data. The mathematical calculation can be shown (in) equation 7 (Id, 2021).

$$P(x_i \mid y) = \frac{1}{\sqrt{2\pi\sigma_y^2}} \exp\left(-\frac{\left(x_i - \mu_y\right)^2}{2\sigma_y^2}\right) \quad (7)$$

Where μ_y The mean of the target class, and σ_y Standard Deviation of the target class.

2.6. Confusion Matrix

Confusion Matrix is a matrix of dimensions N x N, where N is the total number of classes predicted by comparing the expected results (Majumdar, 2023), (Id, 2021). From this matrix, True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) can be extracted for each class and then calculate statistics such as precision, Recall, and F1-Score for each class. In addition, the extraction results can calculate micro averages, d-weighted averages of precision, Recall, and F1-Score. Table 1 shows how TP, TN, FP, and FN can be extracted for class X (Heydarian et al., 2022).

Table 1. *Confusion Matrix Multiclass*

Classes		*Predicted Values*			
		W	X	Y	Z
Actual Value	W	TN	FP	TN	TN
	X	FN	TP	FN	FN
	Y	TN	FP	TN	TN
	Z	TN	FP	TN	TN

Source: (Grandini et al., 2020)

Table 1 shows the multi-class Confusion Matrix, which is a development of the binary Confusion Matrix. There are four prediction classes: W, X, Y, and Z. In the multi-class Confusion Matrix, there are possible cases; namely, If the actual value and the predicted value are true, it can be said that the case is TP. If the exact and expected values are False, then these cases can be said to be TN. If the actual value is True and the predicted value is False, then it can be said that these are FN. If the exact value is false and the expected value is True, then it can be said that these are False Negatives FP (Majumdar, 2023).

Based on the data in 1, a statistical analysis was performed to evaluate the classification performance. Some of the calculations performed include (Quinto, 2020), (Majumdar, 2023):

a. Accuracy is a measure used to evaluate the performance of a classification model, giving an idea of how accurate the model is in making correct predictions from its total predictions.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FP} \quad (8)$$

b. Precision is a measure that shows how accurate the model is in identifying positive outcomes from all outcomes identified as positive and how consistent the model is in providing the correct results when receiving the same input.

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

c. Recall indicates the ability of the model to identify relevant cases in the dataset.

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

d. F1-Score is an evaluation metric that combines precision and Recall in one number. It is used to evaluate the performance of multi-class classification, especially when the distribution of classes is uneven. The F1-Score calculation formula is the harmonic mean of Precision and Recall to reduce False Negatives and False Positives in classification.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (11)$$

3. Results and Discussion

3.1. Business Understanding Phase

The business objective aims to provide governments and stakeholders with detailed insights about several factors that can affect the Human Development Index (HDI), which focuses on improving people's quality of life through education, healthcare access, and poverty alleviation. The initial phase involved identifying HDI-related issues such as inequality in access to education, healthcare, and income disparities through descriptive analysis of relevant data and understanding historical HDI trends. The data mining process aims to identify essential variables that significantly improve HDI, with the development of models to classify these factors and provide insights for improving HDI in the future, based on detailed analysis of historical data and the usage of appropriate data mining techniques.

3.2. Data Understanding Phase

Initial data collection was done by retrieving 2022 Human Development Index data from the Indonesian Central Statistics Agency website, which contained 548 data records.

Table 2. Data Structure

No	Attributes	Notation	Description
1	EYS	X1	numeric
2	AECE	X2	numeric
3	MYS	X3	numeric
4	LE	X4	numeric
5	HDI	Y	categorical

Source: Badan Pusat Statistik (2022)

The five attributes in Table 2 are based on the researcher's analysis concerning the United Nations Development Programme (UNDP), which has three main components: a long and healthy life, knowledge, and a decent standard of living.

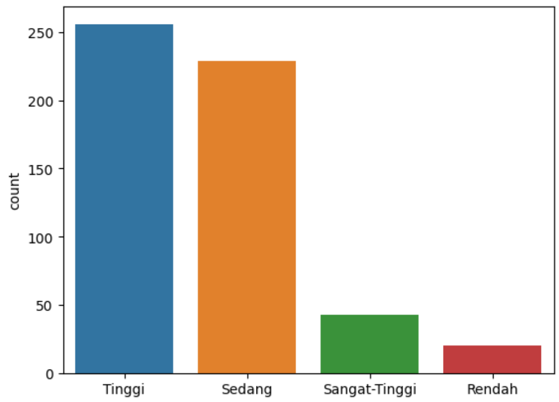

Figure 2. Proportion of Data Distribution for Each Class

The visualisation results in Figure 2 provide an overview of the distribution of datasets based on HDI categories. This information provides an initial overview of the data distribution based on HDI categories in the dataset. According to this data visualisation, it was found that there are 548 total records grouped based on the Human Development Index (HDI) category, with

256 data having a high HDI, 229 data with a medium HDI, 43 data with a very high HDI, and 20 data with a low HDI.

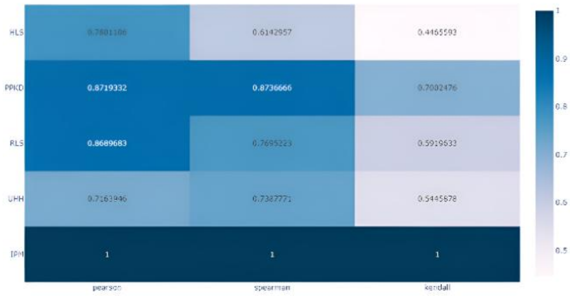

Figure 3. Correlation Pearson, Spearman, and Kendal

Figure 3. is a visualisation using Pearson, Spearman and Kendal methods to display the correlation between features and targets in the dataset. A strong correlation with the target is shown with a dark colour or close to one, while a weak or non-existent correlation is shown with a light colour or up to less than zero. Meanwhile, Figure 4 shows the results of the chi-square test to determine if there is a relationship between features and targets. The assessment is based on the P-value, where if the P-value is less than 0.05, there is enough evidence to reject the null hypothesis and declare a significant relationship.

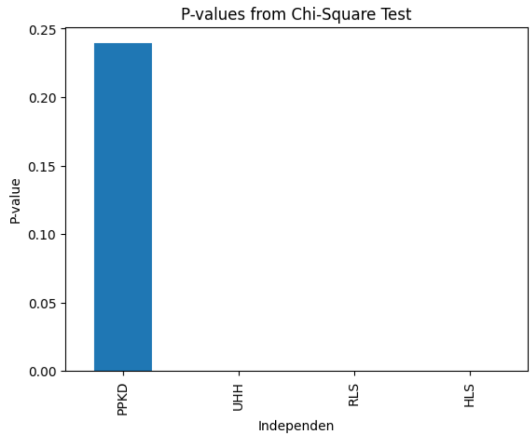

Figure 4. Chi-square Test Results

The analysis in Figure 3 and Figure 4 are the results of the Pearson Correlation, Spearman, Kendal and Chi-square tests, which help identify the relationship between variables with the HDI variable and provide insight into the factors that significantly impact HDI.

3.3. Data Preparation Phase

Based on the Pearson, Spearman and Kendal Correlation results in Figure 3, the four features show a positive correlation with HDI. The chi-square test results in Figure 4 show that the features of the age of life expectancy, average years of schooling and expected years of education have a significant relationship with HDI, while adjusted per capita expenditure does not show a substantial relationship with HDI; so, the features that will be used for data mining analysis can be presented in table 2.

Table 3. Human Development Index Dataset

EYS	AECE	MYS	LE	HDI
14.37	9963	9.44	70.18	High
14.08	7371	9.73	65.48	Medium
14.34	8994	8.69	67.65	Medium
14.69	8353	8.89	64.64	Medium
14.26	8222	9.92	68.48	High
…	…	…	…	…
5.58	5583	2.17	66.13	Low
10.61	5705	4.96	66.13	Low
7.67	5624	3.26	65.93	Low
9.84	4808	3.26	65.66	Low
15.04	15189	11.74	70.76	Very-High

Source: Bada Pusat Statistik (2022)

In Table 2, raw data is required for a dataset integration process. The integration process consists of two main stages: performing label encoding on HDI data, which was initially categorical data into numeric data, represented as 0 for the low category, 1 for the medium category, 2 for the high category, and 3 for the very high category. In addition, normalising the four features in the dataset to equalise the value scale ensures similar value ranges in all variables. It prevents data analysis errors due to significant differences in the value scale.

3.4. Modeling Phase

In this modelling phase, the data mining process is carried out using classification models, such as Decision Tree C4.5, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Naïve Bayes, and Extreme Gradient Boosting (XGBoost). The dataset will be divided into two main parts: 70% training data consisting of 383 data records used to train models and identify patterns, while 30% testing data comprised of 165 data records to test the performance and validity of the trained model.

To implement the C4.5 Decision Tree model, essential parameters, namely Criterion and Max_Depth, must be determined. Criterion determines the split quality measurement method, here using the entropy value. While Max_Depth sets the maximum depth of the decision tree, in this case, it's set at 6.

Extreme Gradient Boosting (XGBoost) incrementally improves the quality of predictive models by optimising a series of weak models and combining them. In classification modelling using XGBoost, various parameters such as Max_Depth=3, learning_rate=0.1, *n_estimators=100, subsample=0.6, and gamma=0.1 are used to achieve optimal performance.*

In the SVM model, an important parameter that needs to be determined is the kernel, which has four value options: linear, polynomial, RBF, and sigmoid. The best kernel that can be used for HDI data classification using SVM is the linear kernel. To improve the accuracy and interpretation of the model, it is essential to enable the probability=True configuration.

For the KNN model, the k value indicates the number of nearest neighbors of an object used as a classification point. In this study, iterations were carried out in the value range of 1 to 20 to determine the optimal k value. The results of the iteration analysis show that the optimal value for k is 18.

Naive Bayes is a model that uses the basic principles of Bayes' Theorem to predict class probabilities from existing attributes. There are several variations of Naïve Bayes, including Gaussian, Multinomial, and Bernoulli. In this process, the Gaussian Naïve Bayes is used due to the continuous character of the dataset.

3.5. Evaluation Phase

In this paper, the evaluation phase provides the results with a comprehensive classification model evaluation. The evaluation is carried out through the Confusion Matrix to identify correctly classified elements located on the main diagonal from top left to bottom right. This provides an overview of the Classification Report for each model.

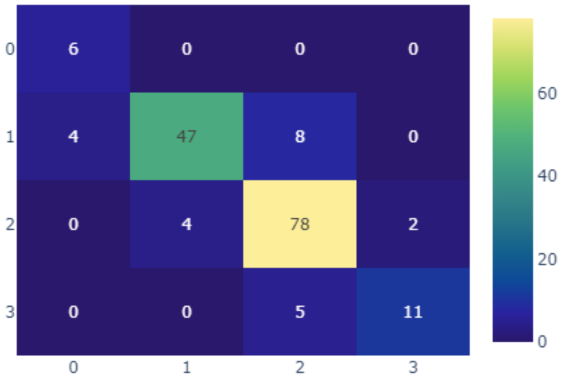

Figure 5. Confusion Matrix Model Decision *Tree* C4.5

Figure 5 shows the testing results of the C4.5 Decision Tree model; it can be seen in the Confusion Matrix that the low class was successfully classified correctly for all data, while for the medium class, there were eight misclassified data from 59 testing data. For the high class, there were six misclassified data from 84 testing data; for the very high class, there were five misclassified data from 16. These results will be used to calculate and get the classification report according to the predefined equation.

Table 4. Classification Report Model Decision Tree C4.5

	Precision	*Recall*	F1-*Score*
Low	0.86	1.00	0.92
Medium	0.92	0.81	0.86
High	0.86	0.93	0.88
Very High	0.85	0.69	0.76
Accuracy			0.87
Macro AVG	0.87	0.86	0.86
Weighted AVG	0.87	0.87	0.87

The Classification Report shows that the C4.5 Decision Tree model performs well using precision, Recall, and F1-Score metrics. The model's precision differed in each class, with the highest value obtained in the medium class of 0.92, indicating an accurate prediction for the medium class. The highest Recall was obtained in the low class of 1.00 and the high class of 0.93, which showed the ability of the model to identify all relevant cases in that class. The high F1-Score in the low class of 0.92 indicates a good balance between Precision and Recall. However, additional analysis is required to evaluate whether the model is acceptable or needs improvement in some areas. The model's overall accuracy was 0.87, with the mean values for precision, Recall, and F1-Score being 0.87, 0.86, and 0.86, respectively. The average weighted values show balanced and consistent results across all classes.

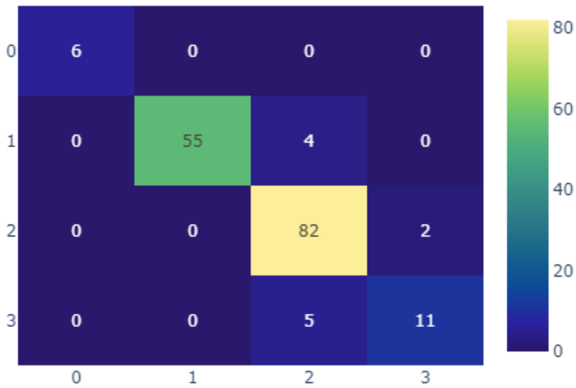

Figure 6. Confusion Matrix Model Extreme Gradient Boosting

Based on Figure 6, Extreme Gradient Boosting model test results successfully classified all low-grade data, 55 out of 59 medium-grade data, 82 out of 84 high-grade data, and 11 out of 16 very high-grade data correctly. However, there some data are classified. Furthermore, mathematical calculations will be carried out based on the specified equations to obtain the Classification Report on the model.

Table 5. Classification Report Model Extreme Gradient Boosting

	Precision	Recall	F1-Score
Low	1.00	1.00	1.00
Medium	1.00	0.93	0.96
High	0.90	0.98	0.94
Very High	0.85	0.69	0.76
Accuracy			0.93
Macro AVG	0.94	0.90	0.92
Weighted AVG	0.93	0.93	0.93

The classification report in Table 5 shows the performance metrics of the Extreme Gradient Boosting model in four different classes. The model shows a very high Precision, especially in the low and medium courses, with a value of 1.00. For Recall, the model identified all relevant cases in the low and high classes, with values of 1.00 and 0.98, respectively. The high F1-Score in the low class indicates a good balance between precision and Recall. However, further analysis is required to evaluate the model's overall performance model. Overall, the Extreme Gradient Boosting model has an accuracy of 0.93, with Macro AVG and Weighted AVG values for precision, Recall, and F1-Score showing good performance consistency across all classes.

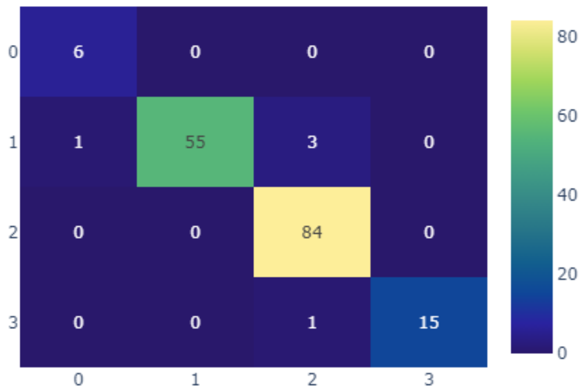

Figure 7. Confusion Matrix Model Support Vector Machine

In Figure 7, the testing results of the Suppothe rt Vector Machine model show that this model successfully classified all low-class data correctly from 6 total testing data, 55 out of 59 medium-class data correctly but misclassified 4 data, and classified all 82 high-class data correctly. For the high class, 15 of 16 data were classified correctly, but one was misclassified. By utilising the Confusion Matrix results, mathematical calculations are performed based on the specified equations to obtain the Classification Report for this model.

Table 6. Classification Report Model Support Vector Machine

	Precision	Recall	F1-Score
Low	0.86	1.00	0.92
Medium	1.00	0.93	0.96
High	0.95	1.00	0.98
Very High	1.00	0.94	0.97
Accuracy			0.97
Macro AVG	0.95	0.97	0.96
Weighted AVG	0.97	0.97	0.97

The Classification Report in Table 6 shows that Support Vector Machine models perform well in precision, especially in the medium and very high classes, with a value of 1.00. High Recall values in the low and high classes indicate the model's ability to identify all relevant cases. The t model also has a high F1-Score, especially in the high class, indicating a good balance between Precision and Recall. Overall, the model accuracy was 0.97, with Macro AVG and Weighted AVG values for precision, Recall, and F1-Score showing consistent performance across all classes.

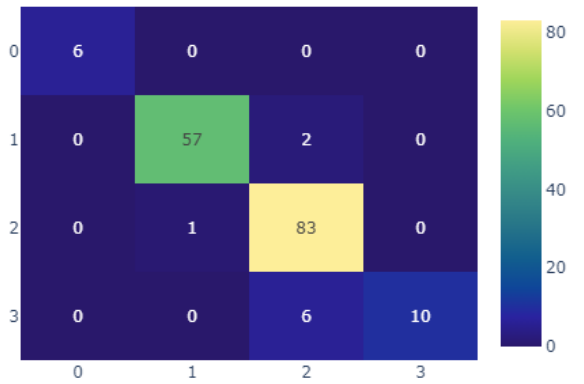

Figure 8. Confusion Matrix Model K-Nearest Neighbor

In Figure 8, the K-Nearest Neighbor model correctly classified all data for the low class out of 6 testing data. For the medium class, 57 out of 59 data were classified correctly, but there were two misclassified data. Meanwhile, 83 out of 84 data were classified correctly for the high class, but there was one misclassified data. For the very high class, 10 out of 16 data were classified correctly, but there were six misclassified data. The Confusion Matrix results are used for mathematical calculations based on the specified equations to obtain the Classification Report on the model.

Table 7. Classification Report Model K-Nearest Neighbor

	Precision	Recall	F1-Score
Low	1.00	1.00	1.00
Medium	0.98	0.97	0.97
High	0.91	0.99	0.95
Very High	1.00	0.62	0.77
Accuracy			0.95
Macro AVG	0.97	0.89	0.92
Weighted AVG	0.95	0.95	0.94

The classification Report in Table 7 shows that the K-Nearest Neighbour model performance metrics have good precision performance, with a score of 1.00 for low and very high class and 0.98 for medium class. This model also achieved high Recall results, especially in the low class, with a value of 1.00, which indicates the model's ability to identify all relevant cases in that class. The high F1-Score, especially in the low class with a value of 1.00, means a good balance between Precision and Recall. Overall, the model has an accuracy of 0.95 and consistent performance across all classes, as indicated by the Macro AVG and Weighted AVG values for precision, Recall, and F1-Score.

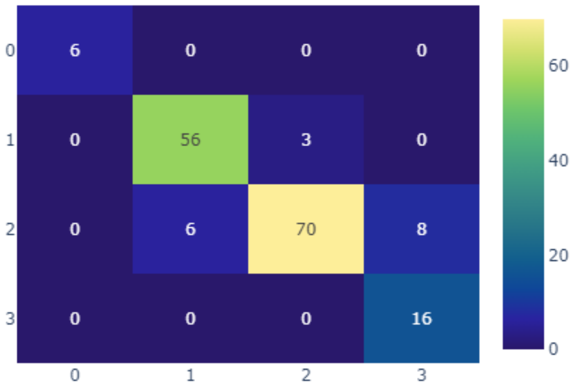

Figure 9. Confusion Matrix Model Naïve Bayes

In Figure 9, the Naïve Bayes model correctly classified low-class testing databases. For the medium class, from 59 data, 56 data were classified correctly, with three misclassified data. While for the high class, from 84 data, 70 were correctly classified, with 15 misclassified data. However, all 16 testing data were correctly classified for the very high class. The Confusion Matrix results are used for mathematical calculations based on the specified equations to obtain the Classification Report on the model.

Table 8. Classification Report model Naïve Bayes

	Precision	Recall	F1-Score
Low	1.00	1.00	1.00
Medium	0.90	0.95	0.93
High	0.96	0.83	0.89
Very High	0.67	1.00	0.80
Accuracy			0.90
Macro AVG	0.88	0.95	0.90
Weighted AVG	0.91	0.90	0.90

The classification Report in Table 8 shows that the Naïve Bayes model performs well in precision, Recall, and F1-Score. The highest precision, 1.00, is in the low class, indicating the accuracy of the model's predictions in that category. The highest Recall, also 1.00, suggests the model's ability to identify all relevant cases in the low class. The high F1-Score in the low class, 1.00, indicates a good balance between Precision and Recall. Overall, Naïve Bayes model accuracy reached 0.90, with Macro AVG and Weighted AVG values for precision, Recall, and F1-Score showing balanced and consistent performance across all classes.

3.6. Deployment Phase

The performance comparison results between classification models, including Decision Tree C4.5, SVM, KNN, Naïve Bayes, and XGBoost using Python, are obtained at the deployment stage. Each data is divided into 70% training data consisting of 165 data and 30% testing data comprised of 383 data, with evaluation based on Precision, Recall, F1-Score, and Accuracy. Table 9 shows the results of the model performance comparison on Human Development Index data.

Table 9. Comparison of Classification Models

Model	*Precision* (%)	*Recall* (%)	F1-Score (%)	Accuracy (%)
Decision Tree C4.5	0.86	0.85	0.85	0.85
XGBoost	0.93	0.93	0.93	0.93
SVM	0.97	0.97	0.97	0.97
KNN	0.95	0.95	0.94	0.95
Naïve Bayes	0.91	0.90	0.90	0.90

Based on the model performance analysis in Table 9, the SVM model shows the most outstanding results with Precision, Recall, and F1-Score of 0.97, which indicates an accurate and consistent prediction. XGBoost also provides good results, with a value of 0.93 for all three metrics. Although Decision Tree C4.5 has an accuracy of 0.85, the lower precision, Recall, and F1-Score values indicate an imbalanced performance compared to SVM and XGBoost. Based on the business understanding phase, which aims to provide detailed insights to governments and stakeholders on the factors affecting HDI, with a focus on education, health, and living standards, integrating SVM into decision-making related to the Human Development Index (HDI), is essential as it provides accurate predictions for various HDI categories.

4. Conclusion

Based on the analytical results, it can be concluded that Decision Tree C4.5, Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Naïve Bayes, and Extreme Gradient Boosting (XGBoost) models successfully classified HDI levels in various regions of Indonesia in 2022. The classification accuracy results show that SVM achieved the highest accuracy of 0.97, indicating an accurate prediction in the context of HDI-level classification. At the same time, the other models also performed well, with accuracies ranging from 0.86 to 0.95. This conclusion shows that machine learning models can provide reasonable predictions for classifying HDI levels in Indonesia in 2022.

Reference

Alamsyah, N., Budiman, B., Yoga, T. P., & Alamsyah, R. Y. R. (2024). Xgboost Hyperparameter Optimization Using Randomizedsearchcv For Accurate Forest Fire Drought Condition Prediction. *Jurnal Pilar Nusa Mandiri*, *20*(2), Article 2. https://doi.org/10.33480/pilar.v20i2.5569

Anam, M. K., Pikir, B. N., & Firdaus, M. B. (2021). Penerapan Na ̈ive Bayes Classifier, K-Nearest Neighbor (KNN) dan Decision Tree untuk Menganalisis Sentimen pada Interaksi Netizen danPemeritah. *MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer*, *21*(1), 139–150. https://doi.org/10.30812/matrik.v21i1.1092

Arumnisaa, R. I., & Wijayanto, A. W. (2023). Comparison of Ensemble Learning Method: Random Forest, Support Vector Machine, AdaBoost for Classification Human Development Index (HDI). *SISTEMASI*, *12*(1), 206. https://doi.org/10.32520/stmsi.v12i1.2501

Badan Pusat Statistik. (2022). Indeks Pembangunan Manusia. *Badan Pusat Statistik*, 178.

Bardab, S. N., Ahmed, T. M., & Mohammed, T. A. A. (2021). *Data mining classification algorithms: An overview*. *8*(1), 41–49.

Budiman, B., Nursyanti, R., Alamsyah, R. Y. R., & Akbar, I. (2020). *Data Mining Implementation Using Naïve Bayes Algorithm and Decision Tree J48 In Determining Concentration Selection*. *1*(3).

Budiman, & Niqotaini, Z. (2021). Perbandingan Algoritma Klasifikasi Data Mining untuk Penelusuran Minat Calon Mahasiswa Baru. *NUANSA INFORMATIKA*, *15*(2), 37–52. https://doi.org/10.25134/nuansa.v15i2.4162

Budiman, & Parama Yoga, T. (2023). Optimalisasi K-Means Berbasis Particle Swarm Optimization untuk Hasil Produksi Tanaman Sayuran di Indonesia. *Jurnal Nuansa Informatika*, *17*, 2614–5405. https://doi.org/10.25134/nuansa

Grandini, M., Bagli, E., & Visani, G. (2020). *Metrics for Multi-Class Classification: An Overview*. https://doi.org/doi.org/10.48550/arXiv.2008.05756

Heydarian, M., Doyle, T. E., & Samavi, R. (2022). MLCM: Multi-Label Confusion Matrix. *IEEE Access*, *10*, 19083–19095. https://doi.org/10.1109/ACCESS.2022.3151048

Huber, S., Wiemer, H., Schneider, D., & Ihlenfeldt, S. (2019). DMME: Data mining methodology for engineering applications – a holistic extension to the CRISP-DM model. *Procedia CIRP*, *79*(March), 403–408. https://doi.org/10.1016/j.procir.2019.02.106

Id, I. D. (2021). *MACHINE LEARNING : Teori, Studi Kasus dan Implementasi Menggunakan Python*. UR PRESS.

Jo, T. (2021). Machine Learning Foundations: Supervised, Unsupervised, and Advanced Learning. In *Machine Learning Foundations: Supervised, Unsupervised, and Advanced Learning*. Springer International Publishing. https://doi.org/10.1007/978-3-030-65900-4

Kemala, I., & Wijayanto, A. W. (2021). Perbandingan Kinerja Metode Bagging dan Non-Ensemble Machine Learning pada Klasifikasi Wilayah di Indonesia menurut Indeks Pembangunan Manusia. *Jurnal Sistem Dan Teknologi Informasi (Justin)*, *9*(2), 269. https://doi.org/10.26418/justin.v9i2.44166

Kumar, A., & Jain, M. (2020). Ensemble Learning for AI Developers. In *Ensemble Learning for AI Developers: Learn Bagging, Stacking, and Boosting Methods with Use Cases*. Apress. https://doi.org/10.1007/978-1-4842-5940-5

Majumdar, P. (2023). *Mastering Classification Algorithms for Machine Learning: Learn how to apply Classification algorithms for effective Machine Learning solutions (English Edition)*. BPB Publications.

Mo, H., Sun, H., Liu, J., & Wei, S. (2019). Developing window behavior models for residential buildings using XGBoost algorithm. *Energy and Buildings*, *205*, 109564. https://doi.org/10.1016/j.enbuild.2019.109564

Nurafidah, Suryowati, K., & Jatipaningrum, M. T. (2023). Perbandingan Metode K-Nearest Neighbor Dan Random Forest Pada Klasifikasi Indeks Pembangunan Manusia Di Kabupaten/Kota Seluruh Indonesia. *Jurnal Statistika Industri …*, *08*(1), 58–67.

Nurhalizah, & Sitompul, P. (2022). Analysis of Ordinary Least Square and Geographically Weighted Regression on the Human Development Index of North Sumatra 2021. *Formosa Journal of Applied Sciences*, *1*(6), 981–1000. https://doi.org/10.55927/fjas.v1i6.1718

Quinto, B. (2020). Next-generation machine learning with spark: Covers XGBoost, LightGBM, Spark NLP, distributed deep learning with keras, and more. In *Next-Generation Machine Learning with Spark: Covers XGBoost, LightGBM, Spark NLP, Distributed Deep Learning with Keras, and More*. https://doi.org/10.1007/978-1-4842-5669-5

Sakarkar, G., Patil, G., & Dutta, P. (2021). Machine Learning Algorithms Using Python Programming. In *Machine Learning Algorithms Using Python Programming*. Nova Science Publishers, Inc.

Schröer, C., Kruse, F., & Gómez, J. M. (2021). A Systematic Literature Review on Applying CRISP-DM Process Model. *Procedia Computer Science*, *181*, 526–534. https://doi.org/10.1016/j.procs.2021.01.199

Singh, H., Navaneeth, N. V., & Pillai, G. N. (2019). Multisurface Proximal SVM Based Decision Trees For Heart Disease Classification. *TENCON 2019 - 2019 IEEE Region 10 Conference (TENCON)*, 13–18. https://doi.org/10.1109/TENCON.2019.8929618

UNDP, (United Nations Development Programme). (2022). Human Development Report 2021-22. *UNDP (United Nations Development Programme)*.

Verdhan, V. (2020). Supervised Learning with Python. In *Supervised Learning with Python*. Apress. https://doi.org/10.1007/978-1-4842-6156-9

Wirth, R., & Hipp, J. (2000). CRISP-DM: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*.