IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Received 4 March 2023, accepted 3 April 2023, date of publication 6 April 2023, date of current version 13 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3265019

RESEARCH ARTICLE

Boosting Algorithm to Handle Unbalanced Classification of $PM_{2.5}$ Concentration Levels by Observing Meteorological Parameters in Jakarta-Indonesia Using AdaBoost, XGBoost, CatBoost, and LightGBM

TONI TOHARUDIN[1], **REZZY EKO CARAKA**[1,2,3], **(Member, IEEE), INDAH RESKI PRATIWI**[1], **YUNHO KIM**[3], **PRANA UGIANA GIO**[4], **ANJAR DIMARA SAKTI**[5], **MAENGSEOK NOH**[6], **FARID AZHAR LUTFI NUGRAHA**[1], **RESA SEPTIANI PONTOH**[1], **TAFIA HASNA PUTRI**[1], **THALITA SAFA AZZAHRA**[1], **JESSICA JESSLYN CERELIA**[1], **GUMGUM DARMAWAN**[1], **AND BENS PARDAMEAN**[7,8]

[1]Department of Statistics, Faculty of Mathematics and Natural Science, Padjadjaran University, West Java 45361, Indonesia
[2]Research Center for Data and Information Sciences, Research Organization for Electronics and Informatics, National Research and Innovation Agency, Bandung, West Java 40135, Indonesia
[3]Department of Mathematical Sciences, Ulsan National Institute of Science and Technology, Ulsan 44919, Republic of Korea
[4]Department of Mathematics, Universitas Sumatera Utara, Medan 20155, Indonesia
[5]Remote Sensing and Geographic Information Sciences Research Group, Faculty of Earth Sciences and Technology, Bandung Institute of Technology, Bandung 40132, Indonesia
[6]College of Information Technology and Convergence, Pukyong National University, Busan 48513, South Korea
[7]Bioinformatics and Data Science Research Center, Bina Nusantara University, Jakarta 11480, Indonesia
[8]Department of Computer Science, BINUS Graduate Program–Master of Computer Science Program, Bina Nusantara University, Jakarta 11480, Indonesia

Corresponding authors: Toni Toharudin (toni.toharudin@unpad.ac.id) and Yunho Kim (yunhokim@unist.ac.kr)

The work of Toni Toharudin, Rezzy Eko Caraka, Resa Septiani Pontoh, and Gumgum Darmawan was supported by the Directorate of Research and Community Engagement, Universitas Padjadjaran. The work of Rezzy Eko Caraka in part by the National Research Foundation of Korea under Grant NRF-2023R1A2C1006845. The work of Yunho Kim was supported by the National Research Foundation of Korea under Grant NRF-2022R1A5A1033624 and Grant NRF-2023R1A2C1006845. The work of Rezzy Eko Caraka and Maengseok Noh was supported by the ''Leaders in Industry-University Cooperation 3.0'' Research Project by the Ministry of Education and the National Research Foundation of Korea.

ABSTRACT Air quality conditions are now more severe in the Jakarta area that is among the world's top eight worst cities according to the 2022 Air Quality Index (AQI) report. In particular, the data from the Meteorological, Climatological, and Geophysical Agency (BMKG) of the Republic of Indonesia, the latest outcomes in air quality conditions in Jakarta and surrounding areas, says that $PM_{2.5}$ concentrations have increased and peaked at $148\mu g/m^3$ in 2022. While a classification system for this pollution is necessary and critical, the observation of $PM_{2.5}$ concentrations measured through the BMKG Kemayoran station, Jakarta, turns out to be identified as an unbalanced data class. Thus, in this work, we perform boosting algorithm supervised learning to handle such an unbalanced classification toward $PM_{2.5}$ concentration levels by observing meteorological patterns in Jakarta during 1 January 2015 to 7 July 2022. The boosting algorithms considered in this research include Adaptive Boosting (AdaBoost), Extreme Gradient Boosting (XGBoost), Categorical Boosting (CatBoost), and Light Gradient Boosting Machine (LightGBM). Our simulations have proven that boosting classification can significantly reduce bias in combination with variance reduction with unbalanced within-class coefficients, with the classification of $PM_{2.5}$ class values: good 62%, moderate 34%, and unhealthy 59%, respectively.

INDEX TERMS Boosting, unbalanced classification, $PM_{2.5}$, XGBoost, AdaBoost, LightGBM, CatBoost.

I. INTRODUCTION

Data science is an applied science that studies explicitly and analyzes data. In today's digital and big data era, data

The associate editor coordinating the review of this manuscript and approving it for publication was Zhaojun Steven Li.

science is fundamental because there is so much available data that can be utilized in decision and policy-making. The data provides information that can determine important decisions in current government policy-making, especially in the Sustainable Development Goals policy [1], [2], [3], [4], [5], [6]. The application of data science to SDGs policies can directly or indirectly provide data focus so that it becomes accurate information with technology methods that are as automated as possible [7], [8], [9]. However, there are often no ready-to-use formulas, algorithms, or models for specific data processing. So, a data scientist must have knowledge of programming that can give his best contribution in supporting accurate, permanent, professional, effective, accountable, efficient, and economical policy making in the utilization of existing data sources to become information that has added value, especially in policy towards current SDGs [3], [10], [11], [12], [13].

One of the most fundamental aspects is urban pollution which can be assessed with an ambient value of $PM_{2.5}$ [14], [15], [16], [17], [18], [19], [20]. The impact of the pollution produced is that it will undoubtedly make it difficult for the Penta-helix contributor to creating a sustainable city that will positively impact career and business opportunities; a safe, comfortable and affordable place to live will be able to build a resilient society and economy [21], [22], [23], [24], [25], [26]. To anticipate this, the Penta-helix contributor needs to be active in making green public transportation, creating environmentally friendly public spaces, and planning and managing cities in an inclusive and participatory way [27], [28], [29].

More than half of the world's population now lives in urban areas. By 2050, that number will rise to 6.5 billion people, two-thirds of the world's population. Sustainable development will not be achieved without significant changes in building and managing urban areas. The rapid growth of cities in developing countries, coupled with increasing urbanization, has resulted in an explosion in the number of megapolitans. In 1990, there were ten megapolitans with a population of 10 million or more. As of 2014, 28 megapolitans were home to about 453 million people. Several previous studies have shown that population density also impacts pollution levels. In addition, the location factor of an area also has an essential role in the spread of pollution. Previous research involved the variables of Dew Point, Wind Speed [30], Pressure [31], Temperature Relative [32], Humidity [33], [34], Precipitation [35], [36], [37], and Wind Direction [38], [39].

A more sophisticated data-driven method is also called ensemble learning. The basic concept underlying this method is the integration of several basic models with a combination strategy to complete the estimation [40], [41], [42]. The ensemble model is categorized into two, namely heterogeneous and homogeneous ensemble models. The heterogeneous model can build a base model by training different learning algorithms or by training algorithms with different parameter settings but using the same dataset. Meanwhile, homogeneous models use the same base model on different training sets.

Many data science studies, especially machine learning, are related to the environment in urban areas [43]. In addition, the industrial and commercial sectors also play a role in exacerbating the condition. Another environmental aspect of concern is waste management and sanitation. One recent study discussed using internet of things (IoT) technology and machine learning to predict industrial waste production [44]. The system successfully predicts waste production indicators quite well and can provide an early warning system that allows authorized officers to anticipate leaks in the sewage system [45]. These environmental studies align with SDG indicator 11.6 to reduce adverse per capita environmental impacts through air quality and waste management.

The BMKG is considered a Non-ministry Government Institution (LPND), headed by a Head of Agency. BMKG has the mission: to implement government responsibilities in the sector of Meteorology, Climatology, Air Quality and Geophysics in line with the prevailing laws and regulations. Meanwhile, we are interested in analyzing $PM_{2.5}$, especially with unbalanced class data, which has never been performed before in fundamental environmental science topics in Indonesia. The boosting methods such as XGboost, AdaBoost, and LightGBM can prevent overfitting and optimize computational resources. The remainder of the paper is organized as follows. "Recent applications on the Boosting Algorithm" section reviews Adaptive Boosting, Gradient Boosting, XGBoost, CatBoost, and LightGBM. "Materials" section presents our dataset and research location. "Results and Discussion" describes descriptive statistics and analysis using boosting. Finally, conclusions and future research directions are indicated in the "Conclusion and future work" section.

II. RECENT APPLICATION ON THE BOOSTING ALGORITHM

A. ADAPTIVE BOOSTING (AdaBoost)

Adaptive Boosting, abbreviated as AdaBoost, is the first boosting algorithm successfully developed by Freund and Schapire in 1999 [46]. AdaBoost focuses on improving performance in areas where the model's base learner or first iteration fails. in areas where the model's base learner or first iteration fails. AdaBoost uses an iterative approach to learn from the mistakes of weak classifiers, and turn them into strong ones, with a Bayesian classifier approach that minimizes the possibility of misclassification by combining many weak classifiers (weak classifiers) [47], [48], [49]. The AdaBoost algorithm is an iterative procedure with a Bayesian classifier approach that minimizes the possibility of misclassification by combining many weak classifiers. It starts with constructing a classifier from an unweighted training sample, for example, a decision tree [50], [51]. If the sample points from the training data are incorrectly classified, then the weight of the training data is boosted. Then, a second classifier was constructed using a training sample with a new modified weight [52]. New weak learners are added to the model sequentially to learn and identify more complex patterns. The data after each iteration is never the same, and possible misclassifications are pointed out for the algorithm to identify and learn. The misclassification weight is increased so that the next iteration can pick it up. This process is repeated for the number of iterations specified as a parameter. AdaBoost combines a number of these weak learners to form strong learners to achieve better separation between classes [53], [54].

B. GRADIENT BOOSTING (GBoost)

Gradient Boosting is a boosting algorithm that optimizes the appropriate loss function [52], [55]. This idea was further developed by Friedman and called the Gradient Boosting Machine (GBM) [56], [57]. GBM works by trying to find new weak learners according to the residual mistakes made by previous weak learners. GBM has an additive model approach which is an iterative and sequential approach to adding trees (weak learners) step by step. Each iteration must reduce the value of its loss function to become closer to the final model. Gradient Boosting works using a gradient descent framework [58]. Gradient descent is used to change parameters iteratively in minimizing the loss function. In other words, gradient descent measures the local gradient of the loss function for a given set of parameters ($\ominus$). Moreover, take a step towards the downward gradient. After the gradient is zero, it has reached a minimum. An essential parameter in gradient descent is the step size determined by the learning rate [59]. If the learning rate is too low, the algorithm will need many iterations to find the minimum.

The advantage of the Gradient Boosting algorithm is that it has much flexibility to optimize different loss functions and provides several hyperparameter tuning options that make the function very flexible [60]. Also, there is no need for data preprocessing as it often works fine with categorical and numeric values as is and can handle missing data, so imputation is unnecessary. While the weakness is that because the Gradient Boosting model will continue to be improved to minimize all errors, this can overemphasize outliers and cause overfitting. In addition, it is computationally expensive because it often requires many trees ($>$1000), which takes up a lot of time and memory; high flexibility generates many combinations of parameters requiring extensive grid searches during tuning [61], [62], [63].

C. EXTREME GRADIENT BOOSTING (XGBoost)

XGBoost is an advanced implementation of an optimized Gradient Boosting algorithm designed to be highly efficient, flexible, and portable [64], [65], [66]. XGBoost is a tree-based algorithm, which sits under the supervised branch of machine learning. While it can be used for both classification and regression problems, all of the formulas and examples in this story refer to the algorithm's use for classification. XGBoost enhances the basic GBM framework through system optimization and algorithm improvements, following [67], [68], [69]: (1) parallelized tree-building where XGBoost has a sequential tree-building approach using implementations in parallel [70], (2) tree pruning where XGBoost grows the tree to max depth and then prunes backward until the increase in loss function is below a threshold [71], [72], (3) cache awareness and out-of-core computing where XGBoost designed to reduce computation time efficiently and allocate memory resources optimally [73], [74], (4) regularization is a technique used to avoid overfitting linear models and tree-based models that limit, adjust or shrink the estimated coefficients towards zero [68], (5) handling missing values, and (6) built-in cross-validation whereas XGBoost comes with this method at every iteration, eliminating the need to explicitly program this seek and to specify the exact number of boosting iterations required in a single run [75], [76], [77].

However, XGBoost has very high parameter flexibility so it requires finding a large set of parameters in the tuning process [78], [79]. XGBoost is a more regularized form of Gradient Boosting. XGBoost uses advanced regularization (*L1* & *L2*), which improves model generalization capabilities. In addition, XGBoost delivers high performance as compared to Gradient Boosting. Its training is very fast and can be parallelized across clusters [73], [80], [81], [82], [83], [84], [85].

D. LIGHT GRADIENT BOOSTING MACHINE (LightGBM)

The LightGBM algorithm uses two new techniques, Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB), to handle a very large number of data samples along with a large number of features. GOSS stores all examples with large gradients and performs random sampling on those with small gradients. The EFB algorithm can combine many exclusive characteristics to a much less characteristic density that can dramatically avoid unnecessary calculations for zero feature value [86]. The LightGBM algorithm is a histogram-based algorithm that inserts continuous feature (attribute) into discrete values [87], giving rise to faster training speed with higher efficiency and reduced memory usage [88], [89].

Unlike most decision tree learning algorithms which grow trees-based and depth-wise, the LightGBM algorithm will grow trees leaf-wise (best-first). Level-wise will maintain the balance of the tree while leaf-wise will reduce more losses by splitting the leaves that experience the most losses. other words, LightGBM will choose the leaves with the maximum delta loss to grow so that they tend to achieve lower losses when compared to the level-wise algorithm [61], [62]. However, although leaf-wise is more flexible, it is also more susceptible to overfitting. Therefore, leaf-wise is preferable when dealing with large datasets.

E. CATEGORICAL BOOSTING (CatBoost)

The term CatBoost is an acronym of 'Category' and 'Boosting' but that doesn't mean that this algorithm can only handle

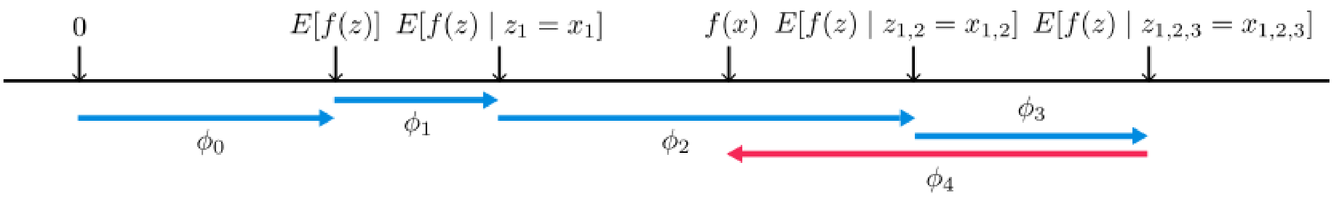


FIGURE 1. SHAP (Shapley additive exPlanation) value.

categorical features but can also support numeric and text features. Nonetheless, CatBoost has good handling techniques for both categorical data and small datasets. The CatBoost algorithm uses a symmetric tree or oblivious tree [90], [91]. Where at each level of the tree, CatBoost uses the same features to divide the training sample into right and left partitions to produce a tree that has a depth of k and exactly $2k$ leaves. During training, a set of decision trees is constructed sequentially. Each successive tree is built at a lower loss when compared to the previous tree. The number of trees is controlled by initial parameters to prevent overfitting [92]. If overfitting occurs then CatBoost may stop training earlier than specified by the training parameters.

F. SHAP VALUES (SHapley ADDITIVE exPlanations)

SHAP (SHapley Additive exPlanations) is a new approach to the complexity of predictive model results and to explore the relationship between individual variables for predicted cases [93]. SHAP is a useful method for sorting effects and breaking down predictions into individual feature impacts [94]. The SHAP value indicates the degree to which a particular feature has changed the prediction, and allows the modeler to decompose any prediction into the sum of the effects of each feature value [95]. The SHAP value is used as a unified measure in measuring feature importance. This Shapley value is the value of the conditional expectation function of the original model [96], [97]. Thus, they are solutions to the equation:

$$\emptyset_i\left(f,x\right)=\sum_{z'\subseteq x'}\frac{\left|z'\right|!\left(M-\left|z'\right|-1\right)!}{M!}\left[f_x\left(z'\right)-f_x(z'\backslash i)\right] \quad (1)$$

where $f_x\left(z'\right) = f(h_x\left(z'\right)) = E\left[f\left(z\right)|z_S|\right]$ and S is the set of non-zero indices in z'. SHAP used to increase the transparency and interpretability of machine learning models.

G. PERFORMANCE EVALUATION METRICS

After implementing a machine learning algorithm, we need tools to evaluate how well the algorithm is performing. This tool is called performance evaluation metrics. In this study, the metrics used for multi-class classification cases are the *F1*-score and the Matthews Correlation Coefficient (MCC) [98]. *F1*-score, also known as f-score or f-measure, takes precision and recall into consideration to calculate the performance of an algorithm. The precision and recall values are obtained from the confusion matrix with the following calculations:

$$Precision=\frac{True\ Positive}{True\ Positive+False\ Positive} \quad (2)$$

$$Recall=\frac{True\ Positive}{True\ Positive+False\ Negative} \quad (3)$$

Then, the F1-score is defined as the harmonic mean of precision and recall which is formulated as follows:

$$F1\ score=2\times\frac{precision\times recall}{precision+recall} \quad (4)$$

The *F1*-score is generally used for unbalanced class cases. The *F1*-score range is [0, 1] where this value will indicate how appropriate the classification is with the algorithm. In other words, this value represents how large and strong the instances are correctly classified. High precision with lower recall will give very accurate results but then miss a large number of instances that are difficult to classify. The bigger the value means the better the model performance.

The Matthews correlation coefficient (MCC) is an alternative metric that is not affected by the problem of unbalanced data. The Matthews correlation coefficient is a contingency matrix method that calculates the Pearson product-moment correlation coefficient between actual and predicted values which is formulated as follows [98], [99], [100].

$$MCC=\frac{TP\cdot TN-FP\cdot FN}{\sqrt{(TP+FP)\cdot(TP+FN)\cdot(TN+FP)\cdot(TN+FN)}} \quad (5)$$

MCC values are in the interval range *[−1, +1]*, with an extreme value of -1 reached when a perfect misclassification occurs and $a + 1$ value for a perfect classification.

III. MATERIALS

The response variable or target class used in the study was the concentration of $PM_{2.5}$ in units of μgram/m3. This variable has 65020 observations per hour with 9141 missing values and 736 irrelevant data. Data labeling will also be carried out on the $PM_{2.5}$ variable by categorizing $PM_{2.5}$ into several categories based on their nature. Labeling is done manually concerning the ISPU parameter concentration value category (Air Pollutant Standard Index) written in Law LHK No.14 Ministry of Environment and Forestry, Republic of Indonesia. This labeling is determined by the author, who aims to produce balanced data classes for multi-class

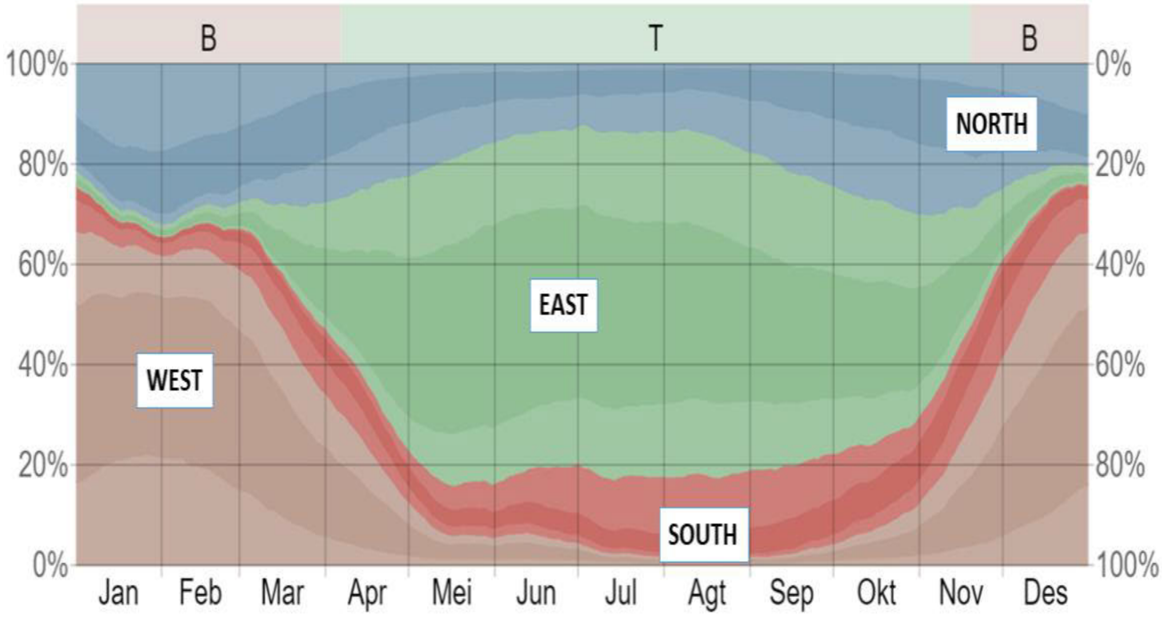


FIGURE 2. Wind direction in Jakarta on 2022 (Source: https://id.weatherspark.com/y/116847.

classification analysis. The PM2.5 labels used include the 'good' category (0 - 28.5 μgram/m^3), the 'moderate' category (28.5 - 40.5 μgram/m^3), and the 'unhealthy' category (>40.5 μgram/m^3).

The dataset in this study consisted of dew point, wind speed, pressure, temperature, relative humidity, precipitation, and wind direction in Jakarta. The most surprising thing is that on June 22, 2022, DKI Jakarta celebrates its 495th anniversary and sadly gets a prize as the city with the worst air quality and pollution in the world. The concentration of PM$_{2.5}$ or air particles smaller than 2.5-micron meters in Jakarta air is 78.5 g/m^3. Jakarta's air quality is 15.7 times above the WHO's annual air quality guideline value. The transportation sector contributed the most to Carbon Monoxide (CO), Nitrogen Oxide (NOx), and PM$_{2.5}$.

Meanwhile, the industrial sector contributed the most to Sulfur Dioxide (SO2), as well as PM$_{2.5}$, in a significant amount. Daily pollution levels are noticeably higher in the dry season than in the rainy season. The variation in the pollution level in various urban areas is more significant in the rain than in the dry season. Jakarta's leading sources of air pollution are vehicle exhaust fumes, coal burning, open burning, construction, road dust, and suspended soil particles. However, gasoline and diesel-fueled vehicles accounted for 32%–57% of PM$_{2.5}$ levels, although the proportion of on-road vehicles and off-road emissions has yet to be determined (e.g., logistics vehicles). Also, the Primary non-vehicle sources accounted for 17%–46% of PM$_{2.5}$ ambient air across sampling sites in both seasons. This portion includes contributions from anthropogenic sources such as coal burning, open burning, construction activities (non-combustion), road dust, and natural resources such as soil and sea salt. Third, Secondary inorganic aerosols account for 1%–16% of the concentration.

The main source of outdoor PM$_{2.5}$ concentrations varies by season and location. Due to variations in local activities or regional sources of pollution, they depend on weather conditions (e.g., upwind emissions from neighboring cities). Figure 2 shows the wind direction in Jakarta during 2022. The percentage of hours during which the average wind direction was from each of the four major cardinal directions, excluding hours with an average wind speed of less than 1.6 kmph. The light colored areas on the border are the percentage of hours spent in the implied center directions (northeast, southeast, southwest, and northwest). It is necessary to divide the data into training and testing sets to find the optimal model parameter set, which has the right balance between these two aspects. The training set is used to build a model with some model parameter settings, and then each model is trained with a testing set. The testing set contains samples of known origin, but this classification is unknown to the model. Therefore, predictions on the testing set allow the operator to judge the model's accuracy. The best separation ratio to use is 80:20. That is, 80% of the dataset goes to the training set, and 20% goes to the testing set which represents in Figure 3.

IV. RESULTS AND ANALYSIS

A. STATISTICS DESCRIPTIVE AND PARAMTER SELECTION

Geographically, Jakarta is bordered to the west by Banten Province and the east and south by West Java Province. To the north, it is bordered by the Java Sea. So, in certain conditions, Jakarta has fog intensity. Fog occurs when water vapor undergoes a process of melting or condensing. During condensation, water vapor molecules combine to create tiny water droplets in the air. The eye can see mist because thick water droplets gather to form clouds. Fog is visible because there is too much moisture in the air, and a very humid area.

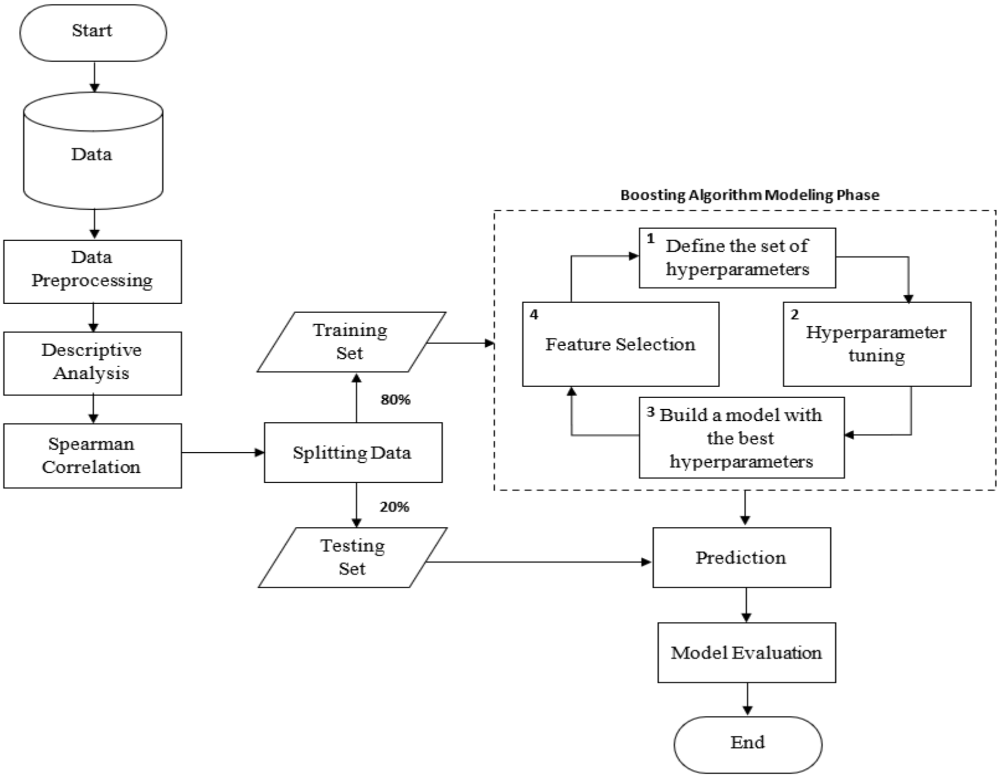


FIGURE 3. Flowchart analysis.

In addition, to make the fog thicker, it must be assisted by components such as pollution or particles in the air. Water vapor condenses around the air pollution particles. Fog is also formed in the sea, often called sea fog. Usually appears around the sea or salt water. They are formed when water condenses around the shores of the ocean. Fog can come suddenly and quickly dissipate depending on the humidity and temperature of the surroundings. The distribution status of $PM_{2.5}$ is also influenced by pressure and temperature. The direction and speed of the wind are influenced by the forces produced by the earth, namely the pressure gradient force, the Coriolis force, the gravity or gravity force, the frictional force, and the centrifugal force. We performed the dataset transformation described in Table 1.

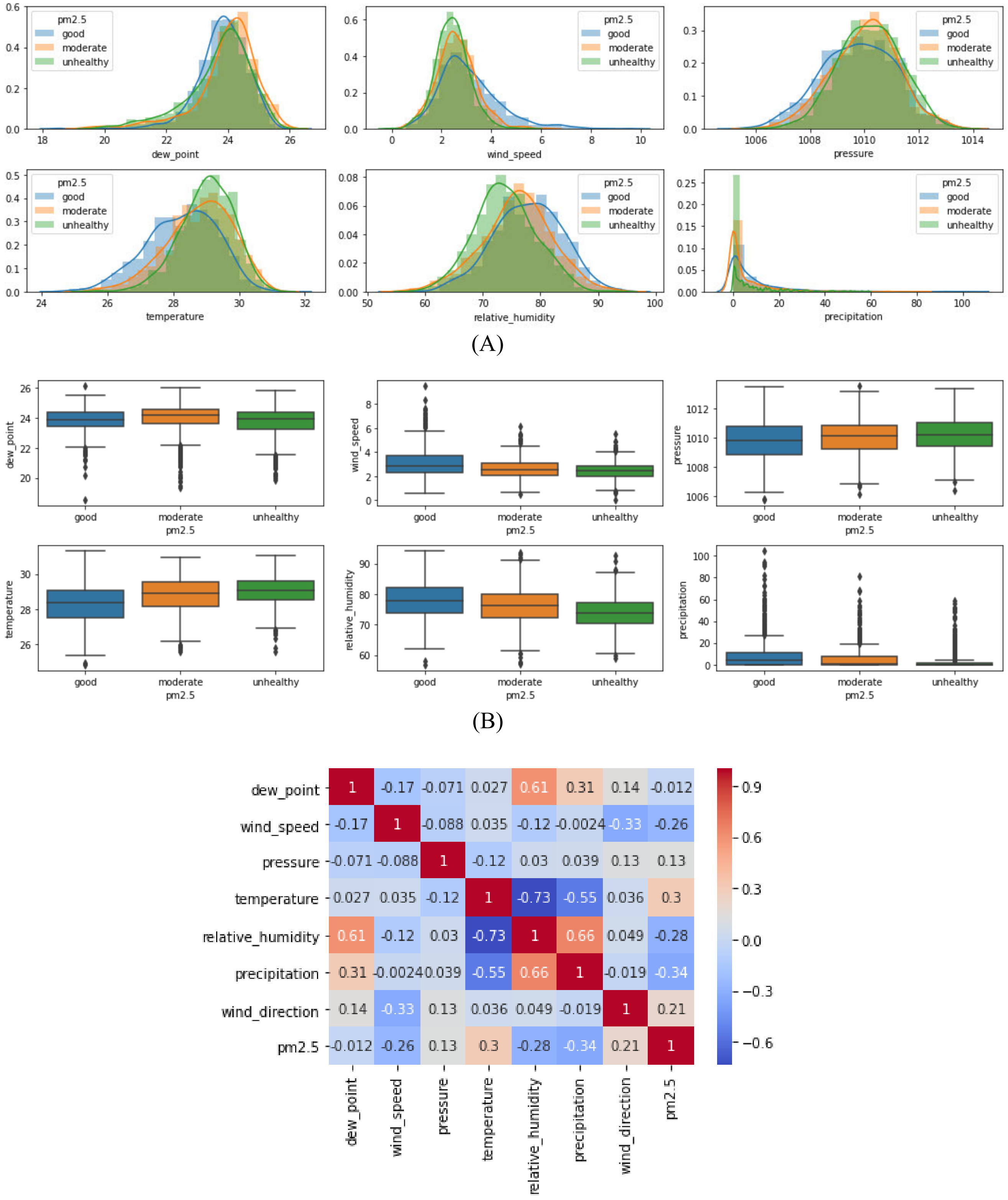
Figure 4a shows the faster the wind speed, the faster the pollutants will move or spread to other locations. Based on Table 2. obtained p-value $< \alpha = 0.05$, which means that there is a statistically significant relationship between features and $PM_{2.5}$. However, based on the obtained r coefficient, the dew point and pressure correlation coefficient are close to zero, meaning there is no relationship between the two features on $PM_{2.5}$.

At the same time, Figure 4b and 4c show that other features have a weak relationship to $PM_{2.5}$, where the correlation coefficient is between 0.25 and 0.5. From these results, it can be concluded that the dew point and pressure variables have no effect, so they should not be included in the modeling process. Figure 4b explains the frequency distribution; it appears that the meteorological parameter variables tend not to follow a specific distribution or distribution asymmetry. When viewed from the distribution slope (skewness), for dew point, air pressure, and temperature variables, they tend to have a negative distribution (negative skewness). Meanwhile, wind speed, relative humidity, and precipitation variables have positive skewness. In addition, by using boxplots, we can detect outliers in the dataset shown in Figure 4c.

Most classification models have one or more model parameters that are used to control for model complexity. The higher the model complexity, the greater the differentiating power the model has, although the risk of overfitting also increases. Overfitting is a phenomenon that is often seen when a training model performs very well on the sample used for training but performs poorly on a new, unknown sample, meaning that the model does not generalize well.

B. USING ENSEMBLE META-ALGORITHM LEARNING

The first stage of data preprocessing is the data cleaning process, where defective, incorrect, incomplete, inaccurate, or irrelevant parts of the data are identified. The



(A)

(B)

(C)

FIGURE 4. Distribution frequency (A), Boxplot of $PM_{2.5}$ concentration, and correlation towards $PM_{2.5}$ (C).

daily hourly observation data of $PM_{2.5}$ concentration consists of 65020 rows of data, identified 9141 missing values and 736 irrelevant data. The missing and irrelevant $PM_{2.5}$ concentration data were deleted to overcome this problem.

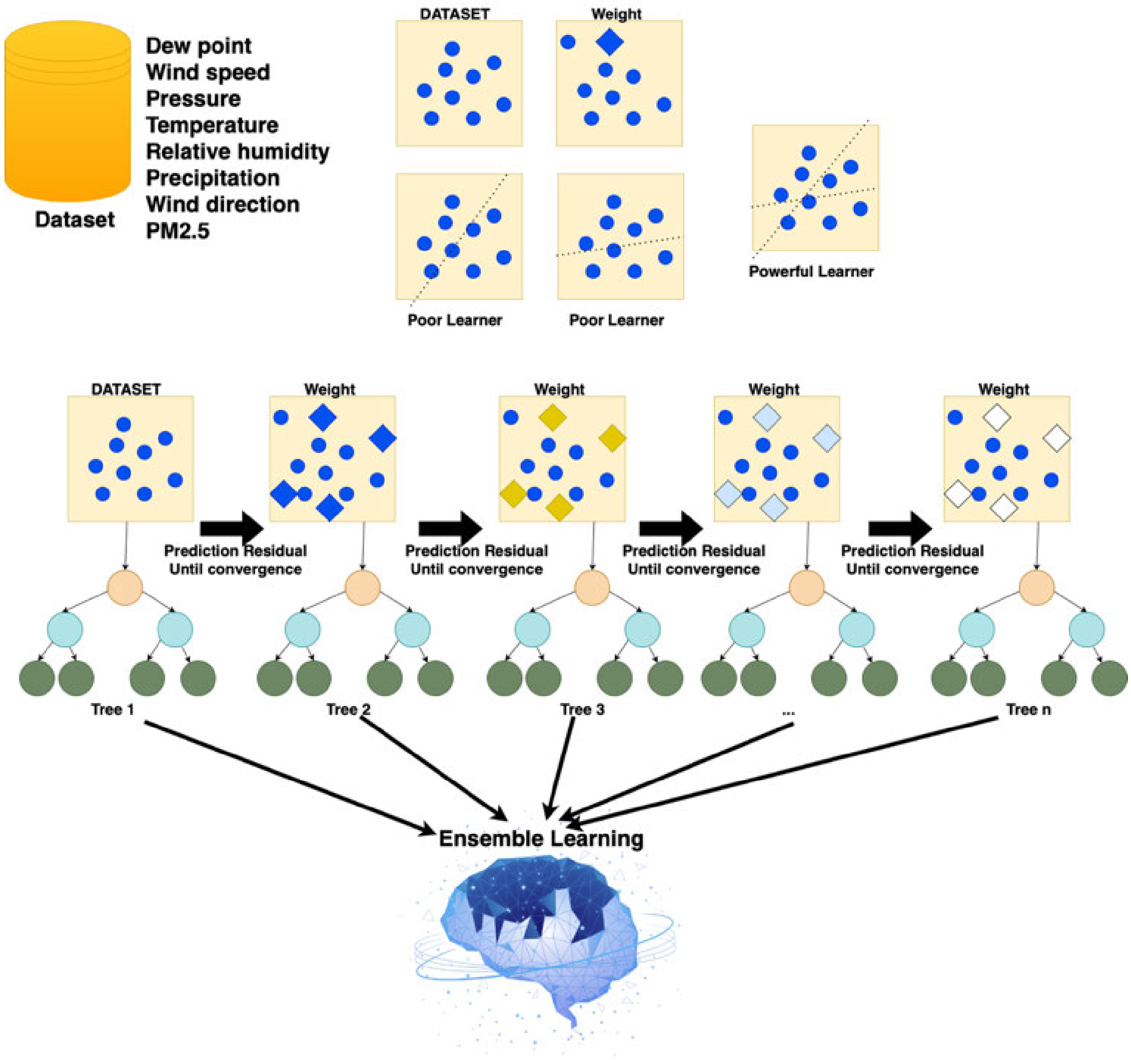


FIGURE 5. Data integration towards ensemble learning.

TABLE 1. Transformation dataset.

INFORMATION	LABEL	CRITERIA	VALUE
PM2.5	$PM_{2.5}$ GOOD	PM2.5 ($\mu gram/m^3$) 0-28.5	**0**
	$PM_{2.5}$ MODERATE	PM2.5 ($\mu gram/m^3$) **28.5-40.5**	**1**
	$PM_{2.5}$ UNHEALTHY	PM2.5 ($\mu gram/m^3$) **>40.5**	**2**
WIND DIRECTION	WEST	**270**	**0**
	SOUTHWEST	**180-270**	**1**
	NORTHWEST	**270-360**	**2**
	SOUTHEAST	**90-180**	**3**
	EAST	**90**	**4**
	NORTH	**0 AND 360**	**5**

Furthermore, data conversion was carried out from hourly data to daily data using averages. The total data that can be used is 2358 observations. Then to produce a balanced class in the $PM_{2.5}$ concentration category, labeling is carried out consisting of the 'good' category (0 - 28.5 μgram/m3), the 'moderate' category (28.5 - 40.5 μgram/m3), and the 'unhealthy' category (>40.5 μgram/m3). To overcome missing data in the daily hourly observation data of meteorological parameters by deleting the data row, except for rainfall data. In the precipitation data, missing data were resolved by changing it from missing values (NaN) to 0 with the assumption that there was no rain during that hour. In addition, precipitation data containing a value of 8888 indicated that the data was not measured. A value of 9999 indicated that there was no data (no measurements were made), so it was resolved by deleting the data row. Next, data conversion from hourly to daily data was carried out for numerical data using the daily average, while for nominal data such as wind direction using the daily mode. Data integration means combining two or more datasets into a single set for analysis, first integrating numerical data on meteorological parameters with

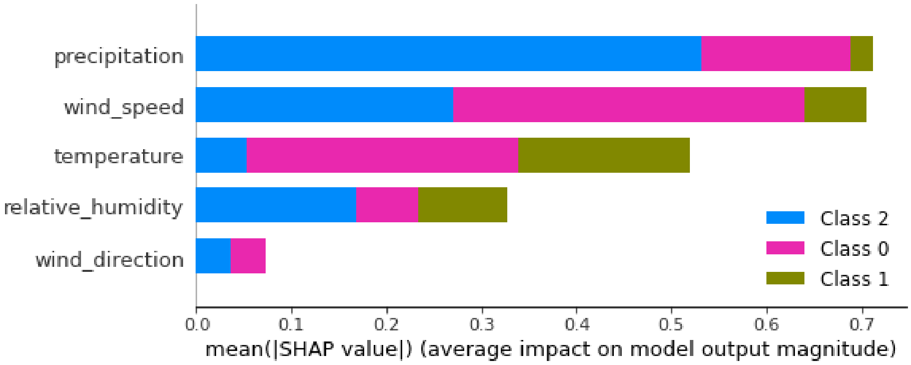


FIGURE 6. Mean SHAP value.

TABLE 2. Variabel information.

Variable	Mean	Std	Min	Max	Spearman (*r*)	p-value
dew point (Celcius °C)	23.84	0.94	18.51	26.10	-0.01239	5.475583e-01
wind speed (m/s)	2.73	0.98	0	9.5	-0.25668	8.623836e-37
Pressure (mb)	1009.98	1.26	1005.71	1013.5	0.13119	1.603591e-10
temperature(Celcius °C)	28.68	1.01	24.84	31.31	0.30354	1.898453e-51
relative humidity (%)	75.95	5.96	56.82	94.16	-0.28087	5.293664e-44
Precipitation(mm)	0.55	1.12	0	11.81	-0.33883	1.964580e-64
wind direction	-	-	-	-	0.21283	1.468484e-25

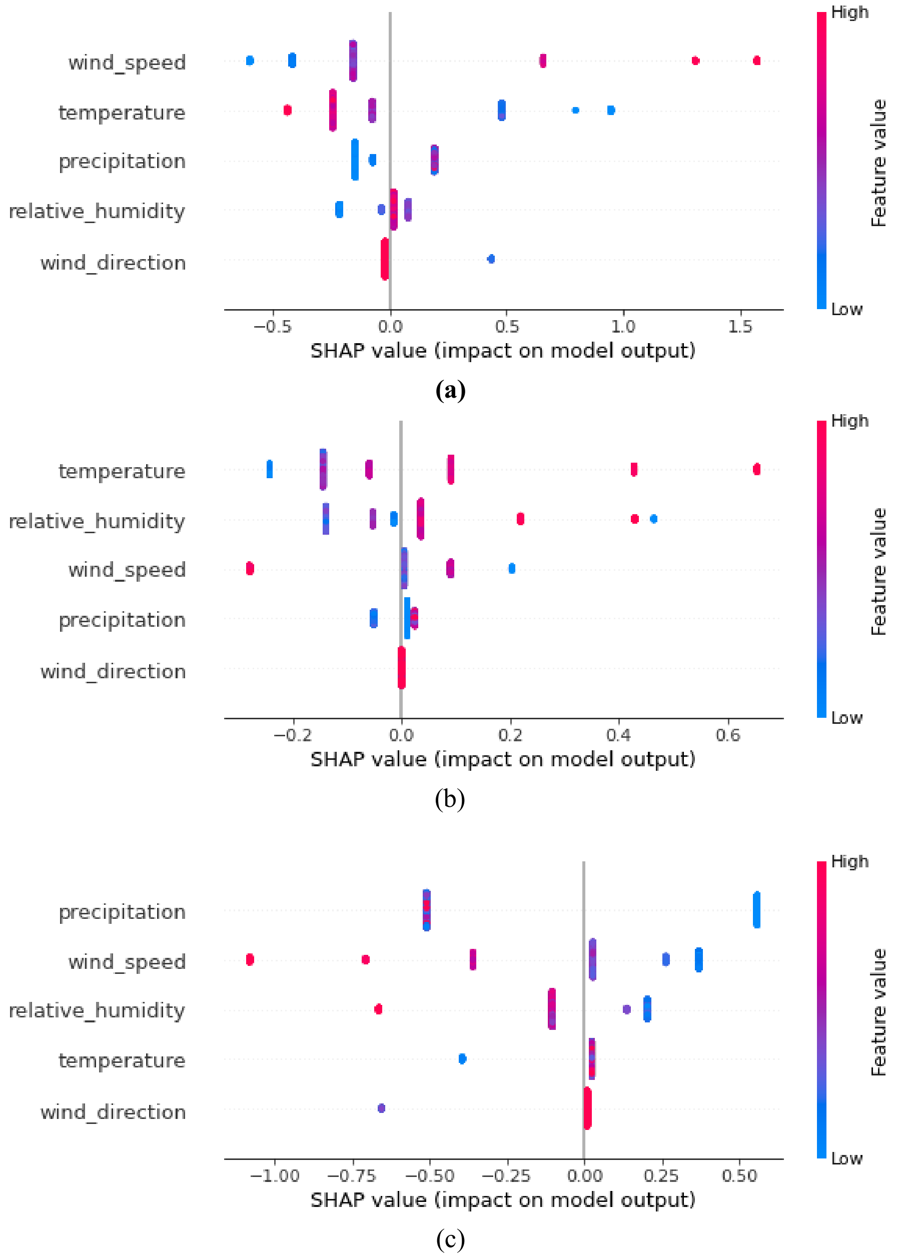
nominal-scale wind direction data. This was conducted due to differences in treatment in converting hourly observation data into daily observation data, where numerical data conversion using the daily average (mean) and wind direction data using the daily mode. The following integrate meteorological parameter data with $PM_{2.5}$ concentration data, which has been converted into daily data. The combined results of the two datasets will be used in the study with a total of 2358 data. Figure 6 explains the conceptual ensemble of learning used in this study.

In the machine learning method, boosting one of the predictive algorithms is very promising and can reduce errors in making predictive models. This study uses several parameter settings for boosting, as in Table 3. The selected boosting techniques include XGBoost, Gradient Boosting, LightGBM, AdaBoost, and CatBoost. The AdaBoost technique initially assigns the same weight to each data set. Then, it automatically adjusts the data point weights after each decision tree. AdaBoost gives more weight to items with incorrect classifications to be corrected in the next round. AdaBoost repeats the process until the remaining error, or the difference between the actual and predicted values, falls below an acceptable threshold. The boosting gradient does not give more weight to items with the wrong classification and can optimize the loss function.

XGBoost is a boosting algorithm that can handle large data sets, making it attractive for big data applications. The main features of XGBoost are parallelization and distributed computing. Light GBM has the tree scaled vertically, while other algorithms have the tree scaled horizontally. Light GBM is leaf-wise, whereas other algorithms are level-wise. In order to expand, we choose a leaf with a max delta loss. When extending the same leaf, leaf-wise algorithms can reduce more losses and losses than level-wise algorithms. Light GBM is literally "Light" light because it is fast. Light GBM can handle large data sizes and takes up little memory when running. Another reason Light GBM is popular is that it focuses on the accuracy of the results. Then, CatBoost can be used to create left and right sections for each tree level and can handle missing values internally.

The parameters used in the Boosting Algorithm are generally diverse and different for each algorithm. Using parameters on the algorithms CatBoost, GradientBoosting, LightGBM, and AdaBoost use the default parameters by setting n_estimators = 100 and tuning the learning rate. In contrast, the XGBoost algorithm is strong enough to handle all kinds of data irregularities. However, building the best model using XGBoost is problematic because this algorithm uses several parameters. In order to improve the model, tuning parameters must be performed. In line with this, Table 4 shows the boosting accuracy as well as runtime training and prediction.

From the Table 4, we can see that XGBoost has the most superior performance in terms of both accuracy and computation in training and prediction.



(a)

(b)

(c)

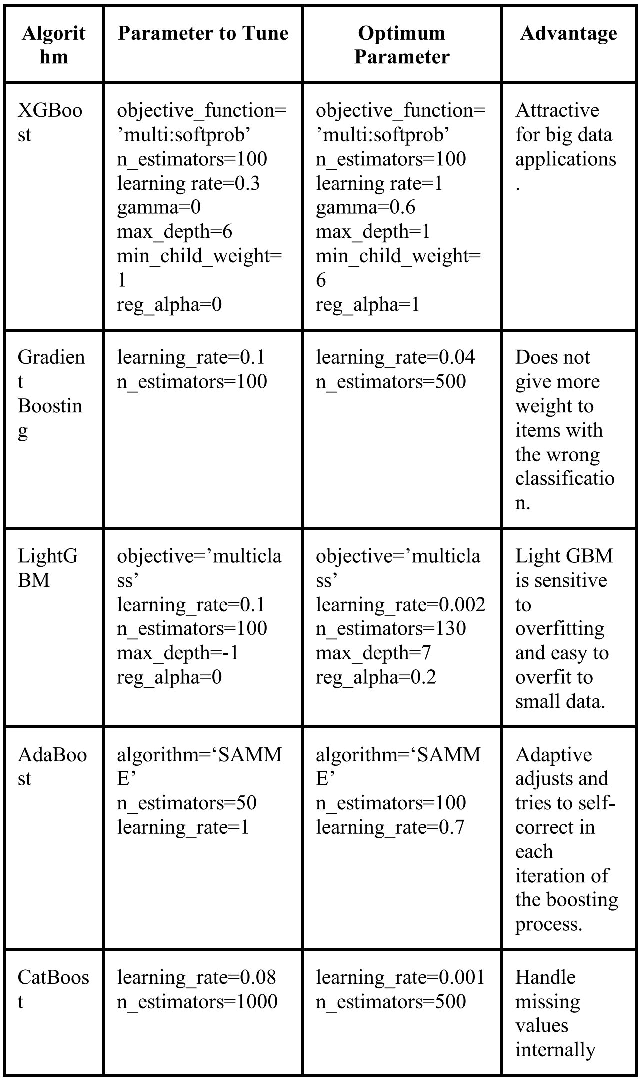
FIGURE 7. Feature impact for each class impact on model output: Class 0 (a), impact on model output: Class 1 (b), and impact on model output: Class 2 (c) 4.3 model performance evaluation.

C. FEATURE SELECTION

Figure 6 shows the average impact of features on the XGBoost model, which is the best model, in predicting $PM_{2.5}$ concentration categories using SHAP Value. The figure shows that the wind direction has a very small contribution to the model. So it can be concluded that this feature is considered not important for the model to make predictions. Figure 7 shows the meteorological parameters that tend to cause the model to predict "good" $PM_{2.5}$ concentrations include higher wind speed, temperature, higher precipitation, and higher relative humidity. Furthermore, meteorological parameters that tend to cause the model to predict "moderate" $PM_{2.5}$ concentrations include higher temperature, higher relative humidity, lower wind speed, and moderate rainfall levels. Meanwhile, meteorological parameters that tend to cause the model to predict "unhealthy" $PM_{2.5}$ concentrations include lower rainfall, lower wind speed, lower relative humidity, and a higher temperature.

Then after modeling using the Boosting algorithm, a performance evaluation of the model is carried out to evaluate how well the algorithm is performing as shown in Table 5. The following table shows the results of the evaluation of the classification model with a model accuracy of 54%. As for the acquisition of *F1*-scores between the "good" class and

TABLE 3. Boosting parameter.



Algorithm	Parameter to Tune	Optimum Parameter	Advantage

XGBoost	objective_function='multi:softprob'
	n_estimators=100
	learning rate=0.3
	gamma=0
	max_depth=6
	min_child_weight=1
	reg_alpha=0
		objective_function='multi:softprob'
		n_estimators=100
		learning rate=1
		gamma=0.6
		max_depth=1
		min_child_weight=6
		reg_alpha=1
			Attractive for big data applications.

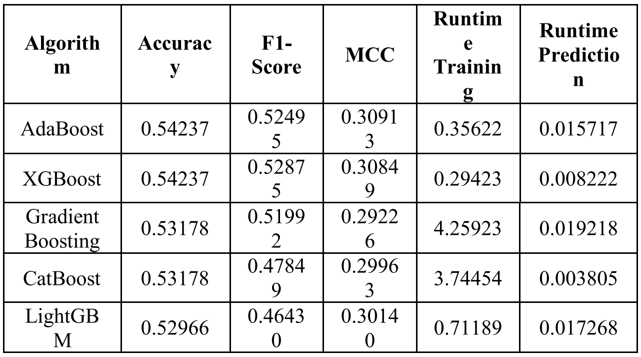
Gradient Boosting	learning_rate=0.1
	n_estimators=100
		learning_rate=0.04
		n_estimators=500
			Does not give more weight to items with the wrong classification.

LightGBM	objective='multiclass'
	learning_rate=0.1
	n_estimators=100
	max_depth=-1
	reg_alpha=0
		objective='multiclass'
		learning_rate=0.002
		n_estimators=130
		max_depth=7
		reg_alpha=0.2
			Light GBM is sensitive to overfitting and easy to overfit to small data.

AdaBoost	algorithm='SAMME'
	n_estimators=50
	learning_rate=1
		algorithm='SAMME'
		n_estimators=100
		learning_rate=0.7
			Adaptive adjusts and tries to self-correct in each iteration of the boosting process.

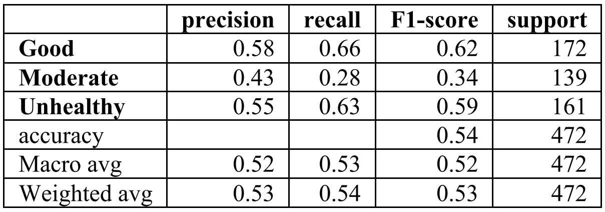
CatBoost	learning_rate=0.08
	n_estimators=1000
		learning_rate=0.001
		n_estimators=500
			Handle missing values internally

TABLE 4. Boosting accuracy.



Algorithm	Accuracy	F1-Score	MCC	Runtime Training	Runtime Prediction
AdaBoost	0.54237	0.52495	0.30913	0.35622	0.015717
XGBoost	0.54237	0.52875	0.30849	0.29423	0.008222
Gradient Boosting	0.53178	0.51992	0.29226	4.25923	0.019218
CatBoost	0.53178	0.47849	0.29963	3.74454	0.003805
LightGBM	0.52966	0.46430	0.30140	0.71189	0.017268

TABLE 5. Classification report.



	precision	recall	F1-score	support
Good	0.58	0.66	0.62	172
Moderate	0.43	0.28	0.34	139
Unhealthy	0.55	0.63	0.59	161
accuracy			0.54	472
Macro avg	0.52	0.53	0.52	472
Weighted avg	0.53	0.54	0.53	472

the "moderate" and "unhealthy" classes have quite a big difference. The *F1*-score value obtained in the "good" class is 62%, while in the "moderate" class is 34% and in the "unhealthy" class is 59%. This shows that the classification model can predict more accurately for "good" $PM_{2.5}$ concentrations than "moderate" and "unhealthy" $PM_{2.5}$ concentrations. When viewed from the precision and recall values, the "good" class obtains a precision value of 0.58 meaning that 58% of the PM2.5 concentration is predicted to be "good" correctly, and a recall value of 0.66 means that there is 66% "good" $PM_{2.5}$ concentration. correctly predicted to have "good" $PM_{2.5}$ In addition, the Matthews Correlation Coefficient (MCC) value of 0.308493 is obtained which is closer to -1. This means that the classification model still produces a high level of misclassification in predicting the $PM_{2.5}$ concentration category.

V. CONCLUSION AND FUTURE RESEARCH

The most powerful factor behind the success of XGBoost is its scalability across all scenarios. While the optimal parameters of the model depend on many scenarios, especially in the XGBoost algorithm, it is imperative to tune them to get a better model. Some notes should be taken into account regarding the tuning parameters in XGBoost, namely: Understanding the Bias-Variance Tradeoff. Most of the parameters in XGBoost are about the bias-variance tradeoff. When we allow the model to become more complicated (i.e., more in-depth), it has a better ability to fit the training data, resulting in a less biased model. However, such complicated models require more data to fit. For that, it is necessary to do parameter tuning to find out whether each parameter will make the model more conservative or not, which is to control overfitting. There are generally two ways to control overfitting in XGBoost: one is to control the model complexity directly, and the other is to add randomness to make the training resistant to noise. It can also reduce the step size.

Solve the problem of imbalanced datasets. This is because a highly imbalanced dataset will affect the training of the XGBoost model. There are two ways to improve it, and if we only look at the overall performance metric of the prediction, then we can balance the positive and negative weights and use AUC for the evaluation metric. However, suppose we are concerned with predicting the exact probability in such a case. In that case, we cannot rebalance the data set but instead can tune the max delta step parameter to a finite number (e.g., 1) to aid convergence. We have provided more information for parameter settings in Table 6 that can be used by those who want to use the same methods.

TABLE 6. Boosting parameter.

AdaBOOST	
Base estimator	The algorithm used as the base learner. If not defined, the value is DecisionTreeClassifier (max_depth=1).
N estimators	The maximum number of estimators at which boosting is stopped. In case of a perfect fit, the learning procedure is stopped early. (Default = 50)
Learning rate	Weights are applied to each classifier at each boosting iteration. A higher learning rate increases the contribution of each classifier. (Default = 1)
Algorithm	The default value is 'SAAME'. Another option for this parameter is the SAMME.R algorithm where the process converges faster by taking fewer incremental iterations and results in lower test errors.
Random state	*The seed used for the random number generator.*
Base estimator	The algorithm used as the base learner. If not defined, the value is DecisionTreeClassifier (max_depth=1).
N estimators	The maximum number of estimators at which boosting is stopped. In case of a perfect fit, the learning procedure is stopped early. (Default = 50)
Learning rate	Weights are applied to each classifier at each boosting iteration. A higher learning rate increases the contribution of each classifier. (Default = 1)
Gradient Boosting	
Max depth	Maximum depth of individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance; the best value depends on the interaction of input variables. (Default = 3)
Learning rate	Controls how fast the algorithm continues to decrease the gradient. A parameter that determines the step size at each iteration while moving towards the minimum loss function. Smaller values reduce the chance of overfitting but also increase the time to find the optimal fit. (Default = 0.1)
N estimators	The number of boosting stages that should be performed. Gradient boosting is strong enough to overfitting so a large number usually results in better performance. (Default = 100)
Random state	*Seeds used for the random number generator.*
Max depth	Maximum depth of individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance; the best value depends on the interaction of input variables. (Default = 3)
XGBOOST	
Learning rate	A parameter that determines the step size at each iteration while moving towards the minimum loss function. Smaller values reduce the chance of overfitting but also increase the time to find the optimal fit. (Default = 0.3).
N estimators	The number of boosting stages that should be performed. (Default = 100)
Gamma	The minimum loss reduction required to create further partitions on the leaf nodes of the tree. The larger the gamma, the more conservative the algorithm. (Default = 0)

TABLE 6. *(Continued.)* Boosting parameter.

Max depth	Maximum depth of the tree. Increasing this value will make the model more complex and more likely to overfit. A value of 0 indicates no depth limit. (Default = 6)
Min child weight	The minimum number of instance weights required in a child. If the tree partitioning step produces leaf nodes with the number of instance weights less than min_child_weight, then the building process will stop further partitioning. The larger the min_child_weight, the more conservative the algorithm. (Default = 1)
Reg alpha	L1 regularization term on the weights. Increasing this value will make the model more conservative. (Default = 0)
Objective	In performing multi-class classification, the objective that should be used is multi:softprob instead of multi:softmax, as the result contains the predicted probability of each data point belonging to each class.
Random state	*The seed used for the random number generator.*
LIGHTGBM	
Max depth	A parameter to explicitly limit the maximum tree depth. This is used to handle overfitting when the data is small. The tree still grows leaf-wise. (Default = -1)
Reg alpha	L1 regularization term on weights. (Default = 0)
Learning rate	A parameter that determines the step size at each iteration while moving towards the minimum loss function. Smaller values reduce the chance of overfitting but also increase the time to find the optimal fit. (Default = 0.1)
N estimators	The number of boosting stages that should be performed. (Default = 100)
Objective	Specify the appropriate learning task and learning objective or the specific objective function to be used. Defaults: 'regression' for LGBMRegressor, 'binary' or 'multiclass' for LGBMClassifier, 'lambdarank' for LGBMRanker.
Random state	*The seed used for the random number generator.*
CatBoost	
Learning rate	A parameter that determines the step size at each iteration while moving towards the minimum loss function. Smaller values reduce the chance of overfitting but also increase the time to find the optimal fit.
N estimators	Number of boosting stages to be performed
Tree depth	The optimal depth ranges from 4 to 10. Values in the range from 6 to 10 are recommended.
L2 regularization	Try different values for the regularizer to find the best one.

Future research is more appropriate to make a comparison between the mathematical models or algorithms that were adopted in the analysis and use a longer range of PM$_{2.5}$ series in minutes, as in this study we used hours. and determine the pros and cons of each one in relation to climatic parameters to make data interpretation and statistical analysis for the data more accurate.

COMPETING INTERESTS

The authors declare no competing interests.

DATA AVAILABILITY

The source code and the material and findings data of this study are openly available in full access by the corresponding author

DECLARATION OF COMPETING INTEREST

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

AUTHOR CONTRIBUTION

Rezzy Eko Caraka and Indah Reski Pratiwi conceived the research and constructed the experimental design. Toni Toharudin, Rezzy Eko Caraka, Yunho Kim, Anjar Dimara Sakti, Maengseok Noh, and Bens Pardamean managed the project. Rezzy Eko Caraka, and Indah Reski Pratiwi analyzed the data. Rezzy Eko Caraka participated in the verification and interpretation of data. Rezzy Eko Caraka and Indah Reski Pratiwi drew the study design, carried out data management, and constructed a database. Rezzy Eko Caraka, and Yunho Kim finalized the instrument. Toni Toharudin, Rezzy Eko Caraka, Indah Reski Pratiwi, Yunho Kim, Prana Ugiana Gio, Maengseok Noh, Anjar Dimara Sakti, Resa Septiani Pontoh, Farid Azhar Lutfi Nugraha, tafia Hasna Putri, Thalita Safa Azzahra, Jessica Jesslyn Cerelia, Gumgum Darmawan, and Bens Pardamean wrote the final manuscript. All the authors read and approved the final manuscript.

ACKNOWLEDGMENT

(Toni Toharudin1 and Rezzy Eko Caraka are co-first authors.)

REFERENCES

[1] L. N. Syahid, A. D. Sakti, R. Virtriana, K. Wikantika, W. Windupranata, S. Tsuyuki, R. E. Caraka, and R. Pribadi, "Determining optimal location for mangrove planting using remote sensing and climate model projection in Southeast Asia," *Remote Sens.*, vol. 12, no. 22, pp. 1–29, 2020.

[2] A. D. Sakti, P. Rohayani, N. A. Izzah, N. A. Toya, P. O. Hadi, T. Octavianti, W. Harjupa, R. E. Caraka, Y. Kim, R. Avtar, N. Puttanapong, C.-H. Lin, and K. Wikantika, "Spatial integration framework of solar, wind, and hydropower energy potential in Southeast Asia," *Sci. Rep.*, vol. 13, no. 1, p. 340, Jan. 2023, doi: 10.1038/s41598-022-25570-y.

[3] F. Provost and T. Fawcett, "Data science and its relationship to big data and data-driven decision making," *Big Data*, vol. 1, no. 1, pp. 51–59, 2013, doi: 10.1089/big.2013.1508.

[4] H. Guo, D. Liang, F. Chen, and Z. Shirazi, "Innovative approaches to the sustainable development goals using big Earth data," *Big Earth Data*, vol. 5, no. 3, pp. 263–276, Jul. 2021, doi: 10.1080/20964471.2021.1939989.

[5] H. Guo, S. Nativi, D. Liang, M. Craglia, L. Wang, S. Schade, C. Corban, G. He, M. Pesaresi, J. Li, Z. Shirazi, J. Liu, and A. Annoni, "Big Earth data science: An information framework for a sustainable planet," *Int. J. Digit. Earth*, vol. 13, no. 7, pp. 743–767, Jul. 2020, doi: 10.1080/17538947.2020.1743785.

[6] H. Guo, D. Liang, Z. Sun, F. Chen, X. Wang, J. Li, L. Zhu, J. Bian, Y. Wei, L. Huang, Y. Chen, D. Peng, X. Li, S. Lu, J. Liu, and Z. Shirazi, "Measuring and evaluating SDG indicators with big Earth data," *Sci. Bull.*, vol. 67, no. 17, pp. 1792–1801, Sep. 2022, doi: 10.1016/j.scib.2022.07.015.

[7] M. S. Sherraden, B. Slosar, and M. Sherraden, "Innovation in social policy: Collaborative policy advocacy," *Social Work*, vol. 47, no. 3, pp. 209–221, Jul. 2002, doi: 10.1093/sw/47.3.209.

[8] S. Pramana, B. Yuniarto, R. Kurniawan, R. Yordani, J. Lee, I. Amin, P. P. N. L. P. Satyaning, Y. Riyadi, A. N. Hasyyati, and R. Indriani, "Big data for government policy: Potential implementations of bigdata for official statistics in Indonesia," in *Proc. Int. Workshop Big Data Inf. Secur. (IWBIS)*, Sep. 2017, pp. 17–21.

[9] R. E. Caraka, F. A. Hudaefi, P. Ugiana, T. Toharudin, A. E. Tyasti, N. E. Goldameir, and R. C. Chen, "Indonesian Islamic moral incentives in credit card debt repayment: A feature selection using various data mining," *Int. J. Islamic Middle Eastern Finance Manage.*, vol. 15, no. 1, pp. 100–124, Jan. 2022, doi: 10.1108/IMEFM-08-2020-0408.

[10] Z. Tufekci, "Engineering the public: Big data, surveillance and computational politics," *1st Monday*, vol. 19, no. 7, Jul. 2014, doi: 10.5210/fm.v19i7.4901.

[11] M. Viceconti, P. Hunter, and R. Hose, "Big data, big knowledge: Big data for personalized healthcare," *IEEE J. Biomed. Health Informat.*, vol. 19, no. 4, pp. 1209–1215, Jul. 2015, doi: 10.1109/JBHI.2015.2406883.

[12] K. Zhou, C. Fu, and S. Yang, "Big data driven smart energy management: From big data to big insights," *Renew. Sustain. Energy Rev.*, vol. 56, pp. 215–225, Apr. 2016, doi: 10.1016/j.rser.2015.11.050.

[13] T. Hey, K. Butler, S. Jackson, and J. Thiyagalingam, "Machine learning and big scientific data," *Phil. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 378, no. 2166, Mar. 2020, Art. no. 20190054, doi: 10.1098/rsta.2019.0054.

[14] H. Li, B. Guo, M. Han, M. Tian, and J. Zhang, "Particulate matters pollution characteristic and the correlation between PM ($PM_{2.5}$, PM_{10}) and meteorological factors during the summer in Shijiazhuang," *J. Environ. Protection*, vol. 6, no. 5, pp. 457–463, 2015, doi: 10.4236/jep.2015.65044.

[15] G.-Y. Lin, H.-W. Chen, B.-J. Chen, and Y.-C. Yang, "Characterization of temporal PM2.5, nitrate, and sulfate using deep learning techniques," *Atmos. Pollut. Res.*, vol. 13, no. 1, Jan. 2022, Art. no. 101260, doi: 10.1016/j.apr.2021.101260.

[16] X. Yan, Z. Zang, N. Luo, D. Li, and Y. Guo, "Retrieval of real-time $PM_{2.5}$, temperature and humidity profiles from satellite and ground-based remote sensing data using advanced deep learning models," in *Proc. ACRS 41st Asian Conf. Remote Sens.*, 2020, pp. 1–9.

[17] J. S. Apte, J. D. Marshall, A. J. Cohen, and M. Brauer, "Addressing global mortality from ambient PM2.5," *Environ. Sci. Technol.*, vol. 49, no. 13, pp. 8057–8066, Jul. 2015, doi: 10.1021/acs.est.5b01236.

[18] M.-C. Yang and M. C. Chen, "PM2.5 forecasting using pre-trained components," in *Proc. IEEE Int. Conf. Big Data (Big Data)*, Dec. 2018, pp. 4488–4491, doi: 10.1109/BigData.2018.8622559.

[19] Z. Zhang, X. Ma, and K. Yan, "A deep learning model for PM2.5 concentration prediction," in *Proc. IEEE Intl Conf Dependable, Autonomic Secure Comput., Intl Conf Pervasive Intell. Comput., Intl Conf Cloud Big Data Comput., Intl Conf Cyber Sci. Technol. Congr. (DASC/PiCom/CBDCom/CyberSciTech)*, Oct. 2021, pp. 428–433, doi: 10.1109/DASC-PICom-CBDCom-CyberSciTech52372.2021.00078.

[20] P. Yuan, Y. Mei, Y. Zhong, Y. Xia, and L. Fang, "A hybrid deep learning model for predicting PM2.5," in *Proc. 7th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2022, pp. 274–278, doi: 10.1109/ICSP54964.2022.9778520.

[21] R. E. Caraka, Y. Lee, R. C. Chen, T. Toharudin, P. U. Gio, R. Kurniawan, and B. Pardamean, "Cluster around latent variable for vulnerability towards natural hazards, non-natural hazards, social hazards in west Papua," *IEEE Access*, vol. 9, pp. 1972–1986, 2021, doi: 10.1109/ACCESS.2020.3038883.

[22] N. Masseran and M. A. M. Safari, "Modeling the transition behaviors of PM_{10} pollution index," *Environ. Monitor. Assessment*, vol. 192, no. 7, pp. 1–15, Jul. 2020, doi: 10.1007/s10661-020-08376-1.

[23] N. Masseran and M. A. M. Safari, "Risk assessment of extreme air pollution based on partial duration series: IDF approach," *Stochastic Environ. Res. Risk Assessment*, vol. 34, nos. 3–4, pp. 545–559, Apr. 2020, doi: 10.1007/s00477-020-01784-2.

[24] N. Masseran and M. A. M. Safari, "Intensity–duration–frequency approach for risk assessment of air pollution events," *J. Environ. Manage.*, vol. 264, Jun. 2020, Art. no. 110429, doi: 10.1016/j.jenvman.2020.110429.

[25] N. A. AL-Dhurafi, N. Masseran, and Z. H. Zamzuri, "Hierarchical-generalized Pareto model for estimation of unhealthy air pollution index," *Environ. Model. Assessment*, vol. 25, no. 4, pp. 555–564, Aug. 2020, doi: 10.1007/s10666-020-09696-9.

[26] N. A. AL-Dhurafi, N. Masseran, and Z. H. Zamzuri, "Compositional time series analysis for air pollution index data," *Stochastic Environ. Res. Risk Assessment*, vol. 32, no. 10, pp. 2903–2911, Oct. 2018, doi: 10.1007/s00477-018-1542-0.

[27] E. Kristiani, H. Lin, J.-R. Lin, Y.-H. Chuang, C.-Y. Huang, and C.-T. Yang, "Short-term prediction of $PM_{2.5}$ using LSTM deep learning methods," *Sustainability*, vol. 14, no. 4, p. 2068, Feb. 2022, doi: 10.3390/su14042068.

[28] A. G. Mengara Mengara, E. Park, J. Jang, and Y. Yoo, "Attention-based distributed deep learning model for air quality forecasting," *Sustainability*, vol. 14, no. 6, p. 3269, 2022, doi: 10.3390/su14063269.

[29] T. Xayasouk, H. Lee, and G. Lee, "Air pollution prediction using long short-term memory (LSTM) and deep autoencoder (DAE) models," *Sustainability*, vol. 12, no. 6, p. 2570, Mar. 2020, doi: 10.3390/su12062570.

[30] R. Cichowicz, G. Wielgosiński, and W. Fetter, "Effect of wind speed on the level of particulate matter PM10 concentration in atmospheric air during winter season in vicinity of large combustion plant," *J. Atmos. Chem.*, vol. 77, nos. 1–2, pp. 35–48, Jun. 2020, doi: 10.1007/s10874-020-09401-w.

[31] J. Schwartz, K. Fong, and A. Zanobetti, "A national multicity analysis of the causal effect of local pollution, NO_2, and $PM_{2.5}$ on mortality," *Environ. Health Perspect.*, vol. 126, no. 8, Aug. 2018, Art. no. 087004, doi: 10.1289/EHP2732.

[32] J. Taylor, C. Shrubsole, M. Davies, P. Biddulph, P. Das, I. Hamilton, S. Vardoulakis, A. Mavrogianni, B. Jones, and E. Oikonomou, "The modifying effect of the building envelope on population exposure to $PM_{2.5}$ from outdoor sources," *Indoor Air*, vol. 24, no. 6, pp. 639–651, Dec. 2014, doi: 10.1111/ina.12116.

[33] J. O. Klompmaker, J. E. Hart, P. James, M. B. Sabath, X. Wu, A. Zanobetti, F. Dominici, and F. Laden, "Air pollution and cardiovascular disease hospitalization—Are associations modified by greenness, temperature and humidity?" *Environ. Int.*, vol. 156, Nov. 2021, Art. no. 106715, doi: 10.1016/j.envint.2021.106715.

[34] R. Zalakeviciute, J. López-Villada, and Y. Rybarczyk, "Contrasted effects of relative humidity and precipitation on urban PM2.5 pollution in high elevation urban areas," *Sustainability*, vol. 10, no. 6, p. 2064, Jun. 2018, doi: 10.3390/su10062064.

[35] Supari, F. Tangang, L. Juneng, and E. Aldrian, "Observed changes in extreme temperature and precipitation over Indonesia," *Int. J. Climatol.*, vol. 37, no. 4, pp. 1979–1997, Mar. 2017, doi: 10.1002/joc.4829.

[36] Y. B. Dibike, P. Gachon, A. St-Hilaire, T. B. M. J. Ouarda, and V. T.-V. Nguyen, "Uncertainty analysis of statistically downscaled temperature and precipitation regimes in northern Canada," *Theor. Appl. Climatol.*, vol. 91, nos. 1–4, pp. 149–170, Feb. 2008, doi: 10.1007/s00704-007-0299-z.

[37] V. Telesca, D. Caniani, S. Calace, L. Marotta, and I. M. Mancini, "Daily temperature and precipitation prediction using neuro-fuzzy networks and weather generators," in *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2017, doi: 10.1007/978-3-319-62407-5_31.

[38] N. Masseran, "Markov chain model for the stochastic behaviors of wind-direction data," *Energy Convers. Manage.*, vol. 92, pp. 266–274, Mar. 2015, doi: 10.1016/j.enconman.2014.12.045.

[39] T. Deryugina, G. Heutel, N. H. Miller, D. Molitor, and J. Reif, "The mortality and medical costs of air pollution: Evidence from changes in wind direction," *Amer. Econ. Rev.*, vol. 109, no. 12, pp. 4178–4219, Dec. 2019, doi: 10.1257/aer.20180279.

[40] L. Rokach, "Ensemble-based classifiers," *Artif. Intell. Rev.*, vol. 33, nos. 1–2, pp. 1–39, Feb. 2010, doi: 10.1007/s10462-009-9124-7.

[41] F. S. Hosseini, B. Choubin, A. Mosavi, N. Nabipour, S. Shamshirband, H. Darabi, and A. T. Haghighi, "Flash-flood hazard assessment using ensembles and Bayesian-based machine learning models: Application of the simulated annealing feature selection method," *Sci. Total Environ.*, vol. 711, Apr. 2020, Art. no. 135161, doi: 10.1016/j.scitotenv.2019.135161.

[42] S. Singh, S. Zhang, M. State, W. A. Pruett, and R. Hester, "Ensemble traces: Interactive visualization of ensemble multivariate time series data," in *Proc. Int. Symp. Electron. Imag. Vis. Data Anal.*, 2016, pp. 1–9, doi: 10.2352/ISSN.2470-1173.2016.1.VDA-505.

[43] Y. Casali, N. Y. Aydin, and T. Comes, "Machine learning for spatial analyses in urban areas: A scoping review," *Sustain. Cities Soc.*, vol. 85, Oct. 2022, Art. no. 104050, doi: 10.1016/J.SCS.2022.104050.

[44] H. Cai, L. D. Xu, B. Xu, C. Xie, S. Qin, and L. Jiang, "IoT-based configurable information service platform for product lifecycle management," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1558–1567, May 2014, doi: 10.1109/TII.2014.2306391.

[45] M. Kamruzzaman, N. I. Sarkar, J. Gutierrez, and S. K. Ray, "A study of IoT-based post-disaster management," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, 2017, pp. 406–410, doi: 10.1109/ICOIN.2017.7899468.

[46] Y. Freund and R. E. Schapire. (1999). *A Short Introduction to Boosting*. [Online]. Available: https://www.research.att.com/

[47] V. Karthikeyan and S. Priyadharsini, "Adaptive boosted random forest-support vector machine based classification scheme for speaker identification," *Appl. Soft Comput.*, vol. 131, Dec. 2022, Art. no. 109826, doi: 10.1016/j.asoc.2022.109826.

[48] G. Chen, H. He, L. Zhao, K.-B. Chen, S. Li, and C. Y.-C. Chen, "Adaptive boost approach for possible leads of triple-negative breast cancer," *Chemometric Intell. Lab. Syst.*, vol. 231, Dec. 2022, Art. no. 104690, doi: 10.1016/j.chemolab.2022.104690.

[49] Y. Fang, Y. Xia, P. Chen, J. Zhang, and Y. Zhang, "A dual-stream deep neural network integrated with adaptive boosting for sleep staging," *Biomed. Signal Process. Control*, vol. 79, Jan. 2023, Art. no. 104150, doi: 10.1016/j.bspc.2022.104150.

[50] J. Huang, J. Lu, and C. X. Ling, "Comparing Naive Bayes, decision trees, and SVM with AUC and accuracy," in *Proc. 3rd IEEE Int. Conf. Data Mining*, Dec. 2003, pp. 553–556, doi: 10.1109/icdm.2003.1250975.

[51] A. Y. Rianto and Y. Kuntoro, "Prediction of netizen tweets using random forest, decision tree, Naïve Bayes, and ensemble algorithm," *Sinkron*, early access, pp. 58–71, 2020.

[52] T. Chen and T. He, "XgBoost: eXtreme gradient boosting," *R Package Version 0.4–2*, vol. 1, no. 4, pp. 1–4, 2015.

[53] Y. Zhao, G. Gao, G. Ding, Q. Zhou, Y. Zhang, J. Wang, and J. Zhou, "Improving the performance of an unmixing model in sediment source apportionment using synthetic sediment mixtures and an adaptive boosting algorithm," *CATENA*, vol. 217, Oct. 2022, Art. no. 106491, doi: 10.1016/j.catena.2022.106491.

[54] H. Q. Tran and C. Ha, "Reducing the burden of data collection in a fingerprinting-based VLP system using a hybrid of improved co-training semi-supervised regression and adaptive boosting algorithms," *Opt. Commun.*, vol. 488, Jun. 2021, Art. no. 126857, doi: 10.1016/j.optcom.2021.126857.

[55] D. Nielsen. (2016). *Tree Boosting With XGBoost*. [Online]. Available: http://dx.doi.org/10.1111/j.1758-5899.2011.00096.x

[56] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers Neurorobotics*, vol. 7, p. 21, Dec. 2013, doi: 10.3389/fnbot.2013.00021.

[57] A. V. Konstantinov and L. V. Utkin, "Interpretable machine learning with an ensemble of gradient boosting machines," *Knowledge-Based Syst.*, vol. 222, Jun. 2021, Art. no. 106993, doi: 10.1016/j.knosys.2021.106993.

[58] S. Islam and S. H. Amin, "Prediction of probable backorder scenarios in the supply chain using distributed random forest and gradient boosting machine learning techniques," *J. Big Data*, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00345-2.

[59] H. Chen, Z. Shen, L. Wang, C. Qi, and Y. Tian, "Prediction of undrained failure envelopes of skirted circular foundations using gradient boosting machine algorithm," *Ocean Eng.*, vol. 258, Aug. 2022, Art. no. 111767, doi: 10.1016/J.OCEANENG.2022.111767.

[60] H. Nguyen and N.-D. Hoang, "Computer vision-based classification of concrete spall severity using metaheuristic-optimized extreme gradient boosting machine and deep convolutional neural network," *Autom. Construct.*, vol. 140, Aug. 2022, Art. no. 104371, doi: 10.1016/J.AUTCON.2022.104371.

[61] T. Thongthammachart, S. Araki, H. Shimadera, T. Matsuo, and A. Kondo, "Incorporating light gradient boosting machine to land use regression model for estimating NO_2 and $PM_{2.5}$ levels in Kansai region, Japan," *Environ. Model. Softw.*, vol. 155, Sep. 2022, Art. no. 105447, doi: 10.1016/J.ENVSOFT.2022.105447.

[62] D. Mishra, B. Naik, J. Nayak, A. Souri, P. B. Dash, and S. Vimal, "Light gradient boosting machine with optimized hyperparameters for identification of malicious access in IoT network," *Digit. Commun. Netw.*, vol. 9, no. 1, pp. 125–137, Feb. 2023, doi: 10.1016/J.DCAN.2022.10.004.

[63] S. Islam and S. H. Amin, "Prediction of probable backorder scenarios in the supply chain using distributed random forest and gradient boosting machine learning techniques," *J. Big Data*, vol. 7, no. 1, p. 65, Dec. 2020, doi: 10.1186/s40537-020-00345-2.

[64] M. Liu, C. Guo, and S. Guo, "An explainable knowledge distillation method with XGBoost for ICU mortality prediction," *Comput. Biol. Med.*, vol. 152, Jan. 2023, Art. no. 106466, doi: 10.1016/J.COMPBIOMED.2022.106466.

[65] J. Dong, W. Zeng, L. Wu, J. Huang, T. Gaiser, and A. K. Srivastava, "Enhancing short-term forecasting of daily precipitation using numerical weather prediction bias correcting with XGBoost in different regions of China," *Eng. Appl. Artif. Intell.*, vol. 117, Jan. 2023, Art. no. 105579, doi: 10.1016/j.engappai.2022.105579.

[66] Y. Wu, G. Mei, and K. Shao, "Revealing influence of meteorological conditions and flight factors on delays using XGBoost," *J. Comput. Math. Data Sci.*, vol. 3, Jun. 2022, Art. no. 100030, doi: 10.1016/j.jcmds.2022.100030.

[67] Z. Yan, H. Chen, X. Dong, K. Zhou, and Z. Xu, "Research on prediction of multi-class theft crimes by an optimized decomposition and fusion method based on XGBoost," *Exp. Syst. Appl.*, vol. 207, Nov. 2022, Art. no. 117943, doi: 10.1016/J.ESWA.2022.117943.

[68] X. Zhou, C. Zhao, and X. Bian, "Prediction of maximum ground surface settlement induced by shield tunneling using XGBoost algorithm with golden-sine seagull optimization," *Comput. Geotechnics*, vol. 154, Feb. 2023, Art. no. 105156, doi: 10.1016/J.COMPGEO.2022.105156.

[69] A. Goswamy, M. Abdel-Aty, and Z. Islam, "Factors affecting injury severity at pedestrian crossing locations with rectangular RAPID flashing beacons (RRFB) using XGBoost and random parameters discrete outcome models," *Accident Anal. Prevention*, vol. 181, Mar. 2023, Art. no. 106937, doi: 10.1016/J.AAP.2022.106937.

[70] A. Dezhkam and M. T. Manzuri, "Forecasting stock market for an efficient portfolio by combining XGBoost and Hilbert–Huang transform," *Eng. Appl. Artif. Intell.*, vol. 118, Feb. 2023, Art. no. 105626, doi: 10.1016/J.ENGAPPAI.2022.105626.

[71] Z. Gao, X. Yin, F. Zhao, H. Meng, Y. Hao, and M. Yu, "A two-layer SSA-XGBoost-MLR continuous multi-day peak load forecasting method based on hybrid aggregated two-phase decomposition," *Energy Rep.*, vol. 8, pp. 12426–12441, Nov. 2022, doi: 10.1016/J.EGYR.2022.09.008.

[72] J. Su, Y. Wang, X. Niu, S. Sha, and J. Yu, "Prediction of ground surface settlement by shield tunneling using XGBoost and Bayesian optimization," *Eng. Appl. Artif. Intell.*, vol. 114, Sep. 2022, Art. no. 105020, doi: 10.1016/J.ENGAPPAI.2022.105020.

[73] Y. Dai, Q. Zhou, M. Leng, X. Yang, and Y. Wang, "Improving the bi-LSTM model with XGBoost and attention mechanism: A combined approach for short-term power load prediction," *Appl. Soft Comput.*, vol. 130, Nov. 2022, Art. no. 109632, doi: 10.1016/J.ASOC.2022.109632.

[74] S. Wang, Y. Zhou, X. You, B. Wang, and L. Du, "Quantification of the antagonistic and synergistic effects of Pb^{2+}, Cu^{2+}, and Zn^{2+} bioaccumulation by living *Bacillus subtilis* biomass using XGBoost and SHAP," *J. Hazardous Mater.*, vol. 446, Mar. 2023, Art. no. 130635, doi: 10.1016/J.JHAZMAT.2022.130635.

[75] Y. Wang, F. Su, Y. Guo, H. Yang, Z. Ye, and L. Wang, "Predicting the microbiologically induced concrete corrosion in sewer based on XGBoost algorithm," *Case Stud. Construction Mater.*, vol. 17, Dec. 2022, Art. no. e01649, doi: 10.1016/J.CSCM.2022.E01649.

[76] J. Li, X. An, Q. Li, C. Wang, H. Yu, X. Zhou, and Y.-A. Geng, "Application of XGBoost algorithm in the optimization of pollutant concentration," *Atmos. Res.*, vol. 276, Oct. 2022, Art. no. 106238, doi: 10.1016/J.ATMOSRES.2022.106238.

[77] A. A. Alabdullah, M. Iqbal, M. Zahid, K. Khan, M. N. Amin, and F. E. Jalal, "Prediction of rapid chloride penetration resistance of metakaolin based high strength concrete using light GBM and XGBoost models by incorporating SHAP analysis," *Construction Building Mater.*, vol. 345, Aug. 2022, Art. no. 128296, doi: 10.1016/J.CONBUILDMAT.2022.128296.

[78] M. Ye, L. Zhu, X. Li, Y. Ke, Y. Huang, B. Chen, H. Yu, H. Li, and H. Feng, "Estimation of the soil arsenic concentration using a geographically weighted XGBoost model based on hyperspectral data," *Sci. Total Environ.*, vol. 858, Feb. 2023, Art. no. 159798, doi: 10.1016/J.SCITOTENV.2022.159798.

[79] T. Wang, Y. Bian, Y. Zhang, and X. Hou, "Classification of earthquakes, explosions and mining-induced earthquakes based on XGBoost algorithm," *Comput. Geosci.*, vol. 170, Jan. 2023, Art. no. 105242, doi: 10.1016/J.CAGEO.2022.105242.

[80] R. Li, L. Cui, H. Fu, Y. Meng, J. Li, and J. Guo, "Estimating high-resolution PM1 concentration from Himawari-8 combining extreme gradient boosting-geographically and temporally weighted regression (XGBoost-GTWR)," *Atmos. Environ.*, vol. 229, May 2020, Art. no. 117434, doi: 10.1016/j.atmosenv.2020.117434.

[81] S. Abdikan, A. Sekertekin, O. G. Narin, A. Delen, and F. B. Sanli, "A comparative analysis of SLR, MLR, ANN, XGBoost and CNN for crop height estimation of sunflower using Sentinel-1 and Sentinel-2," *Adv. Space Res.*, vol. 71, no. 7, pp. 3045–3059, Apr. 2023, doi: 10.1016/j.asr.2022.11.046.

[82] J. Zhang, X. Ma, J. Zhang, D. Sun, X. Zhou, C. Mi, and H. Wen, "Insights into geospatial heterogeneity of landslide susceptibility based on the SHAP-XGBoost model," *J. Environ. Manage.*, vol. 332, Apr. 2023, Art. no. 117357, doi: 10.1016/j.jenvman.2023.117357.

[83] A. Petropoulos and V. Siakoulis, "Can central bank speeches predict financial market turbulence? Evidence from an adaptive NLP sentiment index analysis using XGBoost machine learning technique," *Central Bank Rev.*, vol. 21, no. 4, pp. 141–153, Dec. 2021, doi: 10.1016/j.cbrev.2021.12.002.

[84] A. Ibrahem Ahmed Osman, A. Najah Ahmed, M. F. Chow, Y. Feng Huang, and A. El-Shafie, "Extreme gradient boosting (Xgboost) model to predict the groundwater levels in Selangor Malaysia," *Ain Shams Eng. J.*, vol. 12, no. 2, pp. 1545–1556, Jun. 2021, doi: 10.1016/j.asej.2020.11.011.

[85] H. Hu, A. J. van der Westhuysen, P. Chu, and A. Fujisaki-Manome, "Predicting lake Erie wave heights and periods using XGBoost and LSTM," *Ocean Model.*, vol. 164, Aug. 2021, Art. no. 101832, doi: 10.1016/j.ocemod.2021.101832.

[86] C. Candido, A. C. Blanco, J. Medina, E. Gubatanga, A. Santos, R. S. Ana, and R. B. Reyes, "Improving the consistency of multi-temporal land cover mapping of Laguna lake watershed using light gradient boosting machine (LightGBM) approach, change detection analysis, and Markov chain," *Remote Sens. Appl., Soc. Environ.*, vol. 23, Aug. 2021, Art. no. 100565, doi: 10.1016/J.RSASE.2021.100565.

[87] E. Oram, P. B. Dash, B. Naik, J. Nayak, S. Vimal, and S. K. Nataraj, "Light gradient boosting machine-based phishing webpage detection model using phisher website features of mimic URLs," *Pattern Recognit. Lett.*, vol. 152, pp. 100–106, Dec. 2021, doi: 10.1016/J.PATREC.2021.09.018.

[88] Y. Gu, Y. Yang, Y. Gao, S. Yan, D. Zhang, and C. Zhang, "Data-driven estimation for permeability of simplex pore-throat reservoirs via an improved light gradient boosting machine: A demonstration of sand-mud profile, Ordos Basin, northern China," *J. Petroleum Sci. Eng.*, vol. 217, Oct. 2022, Art. no. 110909, doi: 10.1016/J.PETROL.2022.110909.

[89] J. Sun, J. Li, and H. Fujita, "Multi-class imbalanced enterprise credit evaluation based on asymmetric bagging combined with light gradient boosting machine," *Appl. Soft Comput.*, vol. 130, Nov. 2022, Art. no. 109637, doi: 10.1016/J.ASOC.2022.109637.

[90] L. Wu, G. Huang, J. Fan, F. Zhang, X. Wang, and W. Zeng, "Potential of kernel-based nonlinear extension of Arps decline model and gradient boosting with categorical features support for predicting daily global solar radiation in humid regions," *Energy Convers. Manage.*, vol. 183, pp. 280–295, Mar. 2019, doi: 10.1016/J.ENCONMAN.2018.12.103.

[91] S. Lee, T. P. Vo, H.-T. Thai, J. Lee, and V. Patel, "Strength prediction of concrete-filled steel tubular columns using categorical gradient boosting algorithm," *Eng. Struct.*, vol. 238, Jul. 2021, Art. no. 112109, doi: 10.1016/J.ENGSTRUCT.2021.112109.

[92] N.-H. Nguyen, K. T. Tong, S. Lee, A. Karamanli, and T. P. Vo, "Prediction compressive strength of cement-based mortar containing metakaolin using explainable categorical gradient boosting model," *Eng. Struct.*, vol. 269, Oct. 2022, Art. no. 114768, doi: 10.1016/J.ENGSTRUCT.2022.114768.

[93] F. Ghafarian, R. Wieland, D. Lüttschwager, and C. Nendel, "Application of extreme gradient boosting and Shapley additive explanations to predict temperature regimes inside forests from standard open-field meteorological data," *Environ. Model. Softw.*, vol. 156, Oct. 2022, Art. no. 105466, doi: 10.1016/J.ENVSOFT.2022.105466.

[94] Y. Nohara, K. Matsumoto, H. Soejima, and N. Nakashima, "Explanation of machine learning models using Shapley additive explanation and application for real data in hospital," *Comput. Methods Programs Biomed.*, vol. 214, Feb. 2022, Art. no. 106584, doi: 10.1016/J.CMPB.2021.106584.

[95] N. Nordin, Z. Zainol, M. H. Mohd Noor, and L. F. Chan, "An explainable predictive model for suicide attempt risk using an ensemble learning and Shapley additive explanations (SHAP) approach," *Asian J. Psychiatry*, vol. 79, Jan. 2023, Art. no. 103316, doi: 10.1016/J.AJP.2022.103316.

[96] Y. Zou, Y. Shi, F. Sun, J. Liu, Y. Guo, H. Zhang, X. Lu, Y. Gong, and S. Xia, ''Extreme gradient boosting model to assess risk of central cervical lymph node metastasis in patients with papillary thyroid carcinoma: Individual prediction using Shapley additive exPlanations,'' *Comput. Methods Programs Biomed.*, vol. 225, Oct. 2022, Art. no. 107038, doi: 10.1016/J.CMPB.2022.107038.

[97] P. S. Palar, L. R. Zuhal, and K. Shimoyama, ''Enhancing the explainability of regression-based polynomial chaos expansion by Shapley additive explanations,'' *Rel. Eng. Syst. Saf.*, vol. 232, Apr. 2023, Art. no. 109045, doi: 10.1016/J.RESS.2022.109045.

[98] A. Luque, A. Carrasco, A. Martín, and A. de las Heras, ''The impact of class imbalance in classification performance metrics based on the binary confusion matrix,'' *Pattern Recognit.*, vol. 91, pp. 216–231, Oct. 2019, doi: 10.1016/j.patcog.2019.02.023.

[99] D. Chicco and G. Jurman, ''The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation,'' *BMC Genomics*, vol. 21, no. 1, pp. 1–13, 2020, doi: 10.1186/s12864-019-6413-7.

[100] R. E. Caraka, Y. Lee, R.-C. Chen, and T. Toharudin, ''Using hierarchical likelihood towards support vector machine: Theory and its application,'' *IEEE Access*, vol. 8, pp. 194795–194807, 2020, doi: 10.1109/ACCESS.2020.3033796.

TONI TOHARUDIN received the M.Sc. degree from Katholieke Universiteit Leuven, in 2005, and the Ph.D. degree in spatial sciences from the University of Groningen, in 2010. He is currently a Professor with the Department of Statistics, Padjadjaran University. He acted as the Head of the Research Group, Time Series and Regression. His current research interest includes statistics.

REZZY EKO CARAKA (Member, IEEE) was a Postdoctoral Researcher with the Department of Statistics, Seoul National University, from 2019 to 2010, and the Department of Nuclear Medicine, Seoul National University Hospital, from January 2021 to January 2022. He was also a Research Assistant Professor with the Department of Statistics, Seoul National University, from January to April 2022. He has been an Adjunct Lecturer with the Faculty of Economics and Business, Universitas Indonesia, since 2021; an Adjunct Lecturer with the Graduate School, Department of Statistics, Padjadjaran University, since 2021; and a Senior Research Fellow with the Department of Mathematics, Ulsan National Institute of Science and Technology, South Korea, since 2022. He has also been a Researcher with the Research Center for Data and Information Sciences, Research Organization for Electronics and Informatics, National Research and Innovation Agency (BRIN), Indonesia, since February 2022. His research interests include statistics, large-scale optimization, machine learning, big data analytics, data science, and sustainable development goals.

INDAH RESKI PRATIWI is currently pursuing the degree with the Department of Statistics, Padjadjaran University. She was a Laboratory Assistant with the Department of Statistics, Padjadjaran University, in exploratory data analysis, categorical data analysis, multivariate data analysis, and time-series data analysis subjects using the R and Python programming languages, from August 2020 to December 2022. She was also a Data Analyst Student in the independent study Generasi Gigih 2.0 organized by Yayasan Anak Bangsa Bisa and GoTo Group, from February 2022 to July 2022. She was a Teaching Assistant in a data science class with the Pacmann AI Academy, focused on a deep understanding of SQL and Shell Tooling, from May 2022 to July 2022. Her current research interests include deepening her knowledge in data science, machine learning, and big data.

YUNHO KIM is currently an Associate Professor with the Department of Mathematical Sciences, Ulsan National Institute of Science and Technology, Republic of Korea. The research projects he either had finished or is still pursuing include image denoising/deblurring/segmentation problems and AI research for image processing tasks using reservoir computing networks in neuromorphic computing. The generalized eigenvalue problems and their numerical computations, medical/biomedical image reconstruction problems, and variants of the Allen–Cahn equation in connection with (volume preserving) mean curvature motion. His research interests include the mathematical understanding of image data, especially medical and biomedical data.

PRANA UGIANA GIO is currently the Founder of STATCAL (statistical software; https://statcal.com/) and content creator on the Youtube channel STATKOMAT (programming statistics). He is also a Lecturer with the Department of Mathematics, Universitas Sumatera Utara. He has published dozens of books related to programming and statistics. His research interests include study is building web-based applications using R and Javascript, probability distribution modeling, Monte Carlo simulation, and Bayesian.

ANJAR DIMARA SAKTI is currently a Lecturer with the Remote Sensing and Geographical Information Sciences Research Group, Bandung Institute of Technology, focusing on the application of remote sensing and spatial data science for developing the geospatial product and global–regional–local policy models to achieve the SDGs concerning the water–food energy–ecosystems nexus.

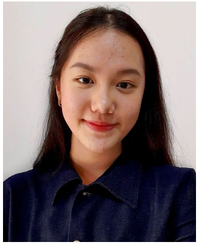
MAENGSEOK NOH was born in Busan, South Korea, in 1973. He received the B.S., M.S., and Ph.D. degrees from the Department of Statistics, Seoul National University, in 1996, 1998, and 2005, respectively. His thesis was on analyzing binary data and robust modeling via hierarchical likelihood. Since 2006, he has been a Professor with the Department of Statistics, Pukyong National University, Busan. His current research interests include application and software development for hierarchical generalized linear models, methodology development for zero-inflated Poisson model with spatial correlation, and hierarchical approach non-Gaussian factor analysis.

FARID AZHAR LUTFI NUGRAHA is currently pursuing the degree with the Department of Statistics, Padjadjaran University. He was an AI and Big Data Research Assistant with the Department of Statistics, Padjadjaran University, conducting research on segmenting 3D images by applying the deep CNN model for segmentation (3DUNet), from August 2022 to December 2022. He was also a Machine Learning Path Student with Bangkit, in 2022, led by Google, and received the TensorFlow Developer Certificate, from February 2022 to August 2022. He was a Laboratory Assistant with the Department of Statistics, Padjadjaran University, in time-series data analysis subject, demonstrating time-series forecasting using various NN models with TensorFlow and Python, from August 2022 to December 2022. His current interests include studying and working in professional data roles, especially machine learning, data science, and big data.

RESA SEPTIANI PONTOH received the bachelor's degree in statistics from Padjadjaran University and the master's degree in business administration and statistical science from the Bandung Institute of Technology and La Trobe University. She is currently a Lecturer with the Department of Statistics, Padjadjaran University. Her research interests include econometrics, behavior statistics, and epidemiology.

TAFIA HASNA PUTRI received the bachelor's degree from the Department of Statistics, Padjadjaran University, where she is currently pursuing the master's degree in statistics (fast-track program). She was a Data Scientist Intern with PT Telkom Indonesia, using R programming languages and SQL tools. Her current research interests include statistics in time-series data analysis, multivariate data analysis, and the big data field to study and work in professional data roles, especially machine learning and data science.

THALITA SAFA AZZAHRA is currently pursuing the degree with the Department of Statistics, Padjadjaran University. She was a Machine Learning Path Student with Bangkit, in 2022, led by Google, Gojek, Tokopedia, and Traveloka, and received the Tensorflow Developer Certificate, from February 2022 to August 2022. She was a Data Science Intern with PT Erajaya Swasembada, Tbk, who made the dashboard for employee internal assessment, from August 2022 to December 2022. Her research interests include learning more about data, especially data analysis, data science, machine learning, and big data.



JESSICA JESSLYN CERELIA received the degree (Hons.) in machine learning from Bangkit Academy, in 2022. She is currently pursuing the degree with the Department of Statistics, Padjadjaran University. She is a Google Certified TensorFlow Developer. She was a Google-led program in collaboration with GoTo and Traveloka, from February 2022 to August 2022. She was an Assistant Lecturer with the Department of Statistics and Mathematics, Padjadjaran University, in a parametric statistics course, from August 2021 to December 2021. Her current research interests include learning more about data analytics, data science, machine learning, and big data.

GUMGUM DARMAWAN received the bachelor's degree in statistics from Padjadjaran University, the master's degree in statistical science from Institut Teknologi Sepuluh November (ITS), Surabaya, and the Ph.D. degree in mathematics from Gadjah Mada University, in 2017. He is currently a Lecturer with the Department of Statistics, Padjadjaran University. His research interests include time-series analysis, statistics computation, and queueing systems.

BENS PARDAMEAN received the bachelor's degree in computer science and the master's degree in computer education from California State University at Los Angeles, Los Angeles, CA, USA, and the Ph.D. degree in informatics research from the University of Southern California (USC). He currently holds a dual appointment as the Director of the Bioinformatics and Data Science Research Center (BDSRC), AI Research and Development Center (AIRDC), and a Professor of computer science with Bina Nusantara (BINUS) University, Jakarta, Indonesia. He has over 30 years of global experience in information technology, bioinformatics, and education. His professional experience includes being a practitioner, researcher, consultant, entrepreneur, and lecturer.