COMMUNICATIONS IN
SCIENCE AND TECHNOLOGY
*Homepage:* cst.kipmi.or.id

# Leveraging machine learning and open accessed remote sensing data for precise rainfall forecasting

Bambang Kun Cahyono[a,*], Muhammad Hidayatul Ummah[a], Ruli Andaru[a], Neil Andika[b], Adjie Pamungkas[c], Hepi Hapsari Handayani[d], Paramita Atmodiwirjo[e] , Rory Nathan[f]

[a]*Geodetic Engineering Department, Universitas Gadjah Mada, Yogyakarta 55281, Indonesia*
[b]*Civil and Environmental Engineering Department, Universitas Gadjah Mada, Yogyakarta 55281, Indonesia*
[c]*Urban and Regional Planning Department, ITS, Surabaya 60117, Indonesia*
[d]*Geomatics Engineering Department, ITS, Surabaya 60117, Indonesia*
[e]*Department of Architecture, Universitas Indonesia, Depok 16425, Indonesia*
[f]*Department of Infrastructure Engineering, University of Melbourne, Parkville 3010, Australia*

*Article history:*
Received: 11 January 2025 / Received in revised form: 10 June 2025 / Accepted: 13 June 2025

## Abstract

Rainfall forecasts are essential for human activities enabling communities to anticipate any impacts. Rainfall events correlate with other natural and hydro-meteorological phenomena, which can be used in modeling and prediction. This study used daily CHIRPS for the Gajahwong watershed in Yogyakarta, Indonesia as the precipitation data. It also used Sea Surface Temperature, Land Surface Temperature (Day and Night), Minimum and Maximum Temperatures, Solar Radiation, Wind Speed (U and V components), Cloud Pressure (Top and Base), and Cloud Height (Top and Base) as the parameters. Further, data processing was performed by means of the Google Earth Engine (GEE) platform. Machine learning methods, including Support Vector Regression, Gradient Boosting Regression, Random Forest, and Deep Neural Networks, were applied. The correlation analysis revealed that only the Wind Speed V-component showed significant correlation with rainfall, other seven parameters showed moderate and four showed weak ones. Meanwhile, accuracy assessments indicated that Support Vector Regression had the most accurate predictions accompanied by Root Mean Square Error (RMSE), Mean Absolute Error (MAE), Mean Squared Error (MSE), $R^2$, and Coefficient Correlation (CC) at 1.366, 0.947, 1.866, 0.948 and 0.982 respectively. This study demonstrated that utilizing openly accessible atmospheric datasets processed through the GEE could yield reliable rainfall predictions, facilitating informed decisions on a wide scale. The methodology is adaptable and can be reproduced for any comparable research or operational purposes.

*Keywords:* Precipitation prediction; rainfall forecast; machine learning; Gajahwong River; hydro-meteorological big data

## 1. Introduction

Rainfall is a primary factor of the hydrological cycle, essential for sustaining ecosystems and maintaining environmental balance on Earth [1]. The process begins with evaporation, followed by condensation, cloud formation, and precipitation [2]. Water, once on the ground, undergoes interception, infiltration, transpiration, and evaporation, completing a hydrological cycle [3,4].

Rainfall distribution in a specific area is determined by various atmospheric factors, including surface temperature, humidity, air pressure, wind speed, cloud cover, and solar radiation [5,6]. In such condition, satellite imagery serves as a reliable tool for detecting and documenting atmospheric phenomena and conditions [7]. As a tropical country, Indonesia experiences substantial rainfall with some equatorial regions receiving it year-round, while others have distinct seasonal patterns [8,9].

Weather conditions significantly impact human activities such as in agriculture (e.g. planting, seedling, maintenance, and harvesting) [10], transportation (land and air), tourism (outdoor and natural attractions) [6], flood mitigation [11], and water resource management [12]. For this, accurate short-term and long-term weather forecasts become vital across multiple sectors [13].

Accurate weather forecasting is deemed vital for anticipating any events that can disrupt daily activities [10]. Climate change has further altered rainfall patterns, making precise prediction increasingly important [13]. Beyond mere delays in human activities, extreme weather events such as prolonged droughts, flash floods, infrastructure damage, and landslides pose serious threats to many regions [14].

To mitigate these negative impacts, reliable rainfall

* Corresponding author. Tel.: +62-274-520226 ; fax: +62-274-520226
Email: bambangkun@ugm.ac.id

https://doi.org/10.21924/cst.10.1.2025.1638

prediction and modeling are crucial. Advanced forecasting models provide early warnings [5], helping to minimize any risks to life, property, infrastructure, and land [6]. While communities have historically relied on local knowledge and wisdom to predict rainfall [15], the growing impact of climate change on weather conditions [9] necessitates more precise and technologically advanced approaches.

Accurate rainfall predictions benefit various sectors by enabling efficient water resource management and effective disaster mitigation. Leveraging big data from diverse weather parameters and applying machine learning algorithms can significantly enhance prediction models [16,17,18]. Techniques such as classification and regression allow for the analysis of historical data to forecast future rainfall patterns [16].

Various methods for rainfall prediction have been developed, including both statistical and machine learning-based approaches [17]. Statistical modeling derives equation models based on existing data (data-driven), utilizing techniques such as Simple Regression Analysis (SRA), Decomposition, Exponential Smoothing (ES), Autoregressive Integrated Moving Average (ARIMA) [17], and Least Squares Adjustment (LA) [19]. Meanwhile, machine learning-based rainfall models such as Neural Networks (NN), Random Forest (RF), Gradient Boosting (GB), Support Vector Machines (SVM) [17], K-Nearest Neighbors (KNN), and Genetic Programming (GP) [18] are considered highly accurate in identifying patterns in existing data.

Machine learning algorithms are able to effectively identify rainfall patterns from historical event data, enabling prediction and regression. These approaches are categorized as univariate forecasting, which relies on patterns in historical data involving a single variable [20,21]. Univariate forecasting is advantageous in consideration to its simplicity in interpretation and computational efficiency [22,23]. In contrast, multivariate time series forecasting incorporates multiple variables as the inputs for predicting future rainfall [20]. This approach is believed to enhance accuracy, capture complex interrelationships among variables, address external factors, and improve robustness [22,23].

Machine learning in this context analyzes correlations between various hydrological and hydro-meteorological phenomena and rainfall occurrences. These correlations can then be used as parameters that influence modeling and prediction [5]. Atmospheric hydro-meteorological conditions correlated with rainfall include evaporation, sunshine, wind speed, humidity, cloud properties, temperature [18], dew point, wind direction, visibility [5], Southern Oscillation Index, NINO 3.4 Index [24], Madden-Julian Oscillation (MJO), Northern Oscillation Index (NOI), and Quasi-Biennial Oscillation (QBO) [13].

Hydro-meteorological data used as covariates are recorded by various sensors attached on satellite platforms. This is an effective alternative to address the limitations of ground station observation data in terms of quantity and accessibility [25]. The quality of satellite data, however, is significantly determined by the resolution of the recording sensor [26], including spatial resolution (detail of coverage) [16,5], temporal resolution (frequency of observations) [16,5], and spectral resolution (sensor capabilities) [27]. With the decades of recorded observations, historical events can be analyzed to uncover patterns and trends [28,25,29]. This wealth of satellite data can be considered "big remote sensing data," ready for processing and analysis to address various challenges [25].

Relevant institutions provide a wide range of hydroclimatic data freely accessible to public. These datasets are collected through continuous observations by satellites, ground stations, or a combination of both [25]. Moreover, these data can be accessed and processed through free platforms such as Google Earth Engine (GEE) [29,30,31]. Many researchers are widely using GEE, which offers multivariate data with strong correlations to rainfall patterns [5,18].

Based on the current conditions and existing literature, the integrated use of complex atmospheric variables and multivariate approaches in rainfall modeling, so far, remains significantly underexplored. Similarly, the application of the Google Earth Engine platform for both univariate and multivariate rainfall prediction using machine learning techniques seems still limited. These gaps highlight a valuable opportunity to investigate the importance of accurate rainfall forecasting and the potential of leveraging publicly available hydro-meteorological data. This study, in turn, aims to forecast rainfall over the next five epochs using machine learning-based univariate and multivariate models.

The novelty of this study lies in the use of complex atmospheric parameters such as Sea Surface Temperature, Land Surface Temperature (Day and Night), Minimum and Maximum Temperatures, Solar Radiation, Wind Speed (U and V components), Cloud Pressure (Top and Base), and Cloud Height (Top and Base) in rainfall modeling and prediction via the GEE platform. Each parameter underwent a correlation analysis to determine its influence on rainfall and alignment with rainfall patterns. Additionally, multiple machine learning methods were evaluated and compared to identify which approach delivered the most accurate predictions.

## 2. Materials and Methods

### *2.1. Area study*

This research was conducted in the Gajahwong River watershed, which spans from the peak of Mount Merapi to Bantul Regency in the Special Region of Yogyakarta, Indonesia. The area is located between coordinates (7°32'26.72"S, 110°26'45.52"E) at the upper part of Mount Merapi and (7°50'21.41"S, 110°23'47.35"E) in the downstream region. The watershed features a steep river channel with significant topographic changes in the northern area, transitioning to low-lying, nearly flat land in the southern area that gradually slopes downward. The low-lying areas have a slope of 10%-15% and are the most vulnerable to flooding [32]. The elevation ranges from 2,905 meters sea level height (SLH) at the peak to 118 meters SLH in the southern part, covering a total area of approximately 44.40 km². Fig. 1 illustrates the study area.

This study area was selected with a consideration that the river serves as a vital source of irrigation for rice fields and traverses several key landmarks, including state and private universities, as well as a zoo. Additionally, the area along the river is prone to frequent flooding, particularly during the period of heavy and high-intensity rainfall [32]. These flooding events are attributed to the river's proximity to a fault zone (geomorphological factor), rapid land use changes, and inadequate water management practices. The study area lies within a tropical climate zone, characterized by unpredictable rainfall with high annual variability and an average air temperature ranging from 23°C to 26°C.

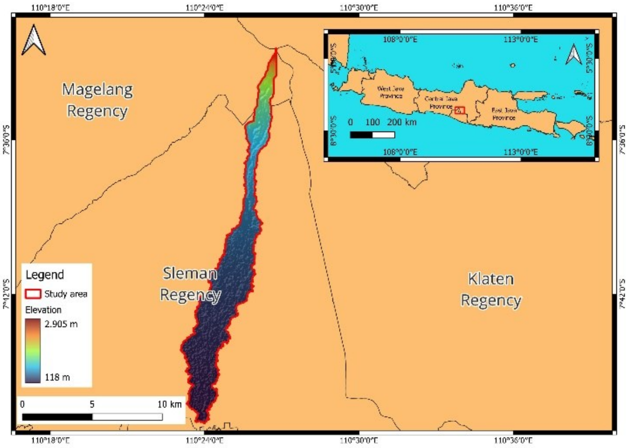

Fig. 1. Overview of the Gajahwong River watershed study area

### 2.2. Data

Table 1. Detailed information about the dataset (sources, spatial resolution, and units of measurement)

| Data | Data sources | Spatial Resolution | Units |
|---|---|---|---|
| Precipitation | CHIRPS https://developers.google.com/earth-engine/datasets/catalog/ UCSB-CHG_CHIRPS_DAILY | 5 KM | mm/day |
| SST | NOAA Optimum Interpolation Sea Surface Temperature (OISST) https://developers.google.com/earth-engine/datasets/catalog/NOAA_CDR_OISST_V2_1 | 0,25 Arc Degree | °C |
| WSU | European Centre for Medium-Range Weather Forecasts (ECMWF) https://developers.google.com/earth-engine/datasets/catalog/ECMWF_ERA5_DAILY | 11 KM | m/s |
| WSV | | 11 KM | m/s |
| STMin | | 11 KM | °C |
| STMax | | 11 KM | °C |
| SSR | | 11 KM | J/m^2 |
| LSTD | MODIS https://developers.google.com/earth-engine/datasets/catalog/modis | 1 KM | °C |
| LSTN | | 1KM | °C |
| CTP | Sentinel 5P https://developers.google.com/earth-engine/datasets/catalog/sentinel-5p | 1 KM | Pa |
| CTH | | 1 KM | m |
| CBP | | 1 KM | Pa |
| CBH | | 1 KM | m |

This study used daily rainfall data from the Climate Hazards Group Infrared Precipitation with Station (CHIRPS) dataset, spanning the period from 2018 to 2023 [33]. For accuracy, the CHIRPS rainfall data have been calibrated against ground station observations within the study area. Twelve additional datasets with daily temporal resolution were incorporated for modeling and prediction, recorded concurrently with the rainfall data. These datasets included sea surface temperature (SST) [34], wind speed component-u (WSU), wind speed component-v (WSV), minimum temperature (STMin), maximum temperature (STMax), surface net solar radiation (SSR) [35], land surface temperature during the day (LSTD), land surface temperature at night (LSTN), cloud top pressure (CTP), cloud top height (CTH), cloud base pressure (CBP), and cloud base height (CBH) [27]. Table 1 depicts the detailed information about the data sources, spatial units, and value units for each dataset.

### 2.3. Method

This study employed four machine learning algorithms across basic learning, ensemble learning, and deep learning categories for multivariate rainfall prediction modeling. The selected algorithms included support vector regression (SVR) representing basic learning, random forest (RF) and gradient boosting regressor (GBR) representing ensemble learning, and deep neural network (DNN) representing deep learning. The research was carried out in three key stages: data preprocessing, multivariate rainfall prediction modeling and evaluation, and future rainfall prediction across upcoming epochs.

#### 2.3.1. Preprocessing data

In this stage, several preprocessing steps were performed, including feature selection, dataset generation, data scaling, and rainfall data filtering. Feature selection aimed to identify variables significantly correlated with rainfall. The Pearson correlation coefficient, as defined in Eq. (1) [36], was utilized to measure the relationship between each variable and rainfall. Variables with an absolute correlation value greater than 0.4, indicating a moderate to strong relationship [37,38], were selected for inclusion in the rainfall prediction process. The parameters description includes $r$ as the correlation; $x_i$ as the value of dataset 1 at point $i$; $\bar{x}$ as the mean of dataset 1; $y_i$ as the value of dataset 2 at point $i$; and $\bar{y}$ as the mean of dataset 2.

$$\boldsymbol{r} = \frac{\sum(x_i-\bar{x})(y_i-\bar{y})}{\sqrt{\sum(x_i-\bar{x})^2}\sqrt{\sum(y_i-\bar{y})^2}} \tag{1}$$

Dataset generation involved defining the model's predictor variables *(x)* and target labels *(y)*. For each variable at a given epoch $x_t$, the predictor variables used in this study included data from the five preceding epochs, denoted as $x_{t-1}, x_{t-2}, x_{t-3}, x_{t-4}, and\ x_{t-5}$. All variables with a correlation greater than 0.4 with precipitation, including precipitation itself, were included as the predicting variables. Thus, the total number of predicting variables in this study was $n \times 5$, where $n$ is the number of variables with a correlation exceeding 0.4. The target label *(y)* is defined as the precipitation at a specific epoch, represented as $y_t$.

Each variable in the dataset had a different range or scale of values, spanning from single units to thousands. To reduce potential biases arising from these differences in value dimensions, this study then applied the Min-Max Scaler normalization technique, rescaling the values of all variables to a uniform range between 0 and 1 [39]. The Min-Max scaling process is represented in Eq. (2), where $x'_i$ is the scaled value of the $i$-th data point, $x_i$ is the original value, $x_{min}$ is the minimum value of the variable, and $x_{max}$ is the maximum value [40].

$$x'_i = \frac{x_i - x_{min}}{x_{max} - x_{min}} \tag{2}$$

In addition to preprocessing the prediction variables, this study applied the label data preprocessing. It involved filtering the data to ensure a clean time-series, and reduce the uncertainty. To achieve these purposes, the Butterworth filter, a widely utilized technique in digital signal processing, was employed. The Butterworth filter is known for its maximally flat passband response and is used to remove any undesired frequencies and noise [41]. The filtering process involved data transformation from the time domain to the frequency domain and the application of the Butterworth transfer function. The transfer function is defined in Eq. (3), where $f$ represents the frequency at the $i$-th data point, $f_c$ is the cut-off frequency, and $N$ is the filter order [42]. For this study, a low-pass Butterworth filter was applied with a cut-off frequency of 0.5 and a filter order of 5. The filtering process was implemented using the NumPy library, specifically utilizing the butter function.

$$\boldsymbol{Butterworth} = \frac{1}{1+(f/f_c)^{2N}} \tag{3}$$

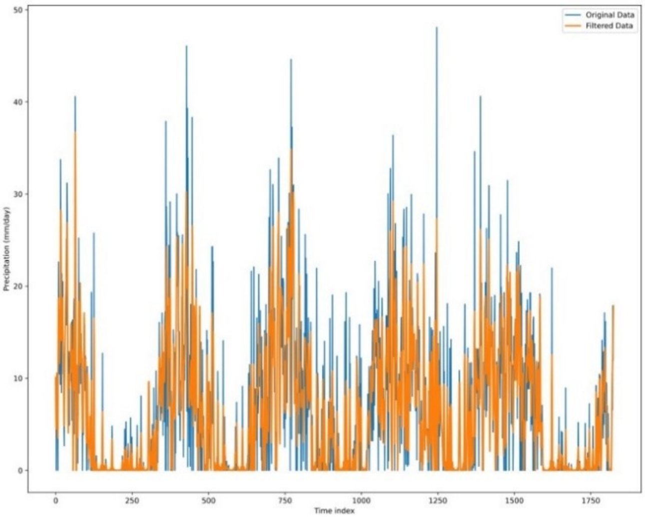

Fig. 2. Filtering results using the Butterworth filter
(data source: CHIRPS accessed and processed using GEE, by the authors)

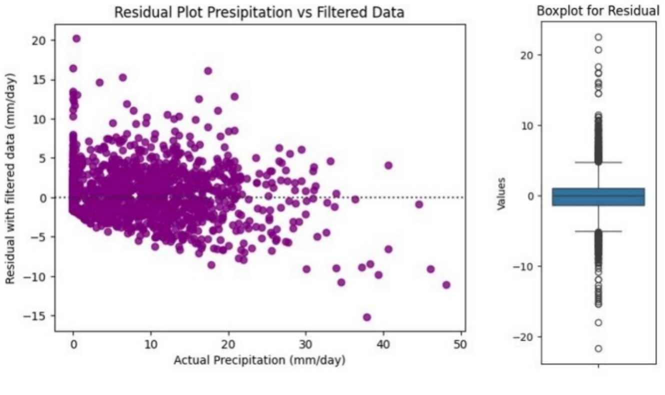

Fig. 3. Residual values from rainfall filtering compared to the original data
(data source: CHIRPS accessed and processed using GEE, by the researchers)

Fig. 2. presents the results of rainfall data filtering using the Butterworth filter. Visually, the differences between the original and filtered data were minimal, with only slight deviations observed. The filtering process successfully reduced a number of outliers, resulting in cleaner data better suited for modeling in subsequent steps.

Fig. 3 depicts the residual values, representing the differences between the original and filtered data. These residuals ranged from -21.71 mm/day to 22.51 mm/day. Most residuals clustered close to zero with an average value of -0.08 ± 3.62 mm/day. When expressed in absolute terms, the mean residual became 2.24 ± 2.85 mm/day, indicating that the filtering process effectively preserved the primary data trends while smoothing out noise. The pattern of daily rainfall fluctuations was still observed, but the seasonal pattern was refined by applying Butterworth filtering. Under these conditions, the prediction algorithm was able to learn the time-series pattern well [43].

### *2.3.2. Multivariate rainfall prediction modeling and evaluation*

The prediction dataset was divided using a sequential splitting method with an 80:20 ratio. This meant that 80% of the data were allocated for training, while the remaining 20% was reserved for testing. The training dataset was used to develop the prediction model, and the testing dataset was utilized to evaluate the model's performance. In this paper, the multivariate prediction model was constructed using four machine learning algorithms: SVR, RF, GBR, and DNN.

Support Vector Regression (SVR) refers to a regression variant of the Support Vector Machine (SVM) algorithm. SVR was developed by Vapnik (2000) [44], and is based on statistical and mathematical principles establishing relationships between a set of independent variables and a dependent variable [44]. In SVR, non-linear relationships are addressed by projecting the data into a higher-dimensional feature space where a linear function is used to approximate the relationships. This function, represented as a vector, includes an epsilon value to account for uncertainty within the vector space. SVR employs a deterministic optimization approach to minimize errors. The general formulation of SVR is shown in Eq. (4), where w is the weight vector in the feature space, φ is the transformation function that linearizes the input data in the new feature space, and b is the bias term [45]. SVR offers several advantages, including its effective handling of multi-dimensional data and relatively low computational requirements [46].

$$\boldsymbol{y = W^T \cdot \psi(x) + b, x \in R^d, \psi(x) \in R^d, b \in R} \tag{4}$$

Random Forest (RF), introduced by Breiman in 2001 [47], is an ensemble machine learning model designed for both classification and regression tasks [47,48]. It operates by constructing multiple decision trees (DTs) and utilizing a bagging technique, or bootstrap aggregation, to generate diverse datasets through randomization strategies [49]. Since RF consists of numerous DTs, the final prediction is obtained by averaging the outputs of all individual trees, as described in Eq. (5). Here, $y$ represents the final output, $y_{DT_i}$ is the output of the $i$-th decision tree, and $N$ is the total number of decision trees generated [50]. The primary strengths of RF include its non-parametric nature, high predictive accuracy, and robustness against noisy or overfitting data [51,52].

$$\boldsymbol{y = \frac{1}{N}\sum_{i=1}^{N} y_{DT_i}} \tag{5}$$

Gradient Boosting Regression (GBR), introduced by Friedman in 2001 [53], is an ensemble learning model that builds multiple decision trees (DTs) iteratively and sequentially [53]. Each subsequent DT is trained to minimize the residual errors of the previous ones. The predicted value of GBR using n decision trees is expressed in Eq. (6), where $DT_m$ represents a weak learner, typically a single decision tree with low individual performance, and $\gamma_m$ is a scaling factor applied to the tree to minimize residuals. To achieve this, GBR employs gradient descent to adjust the model by updating the predictions based on the residuals of prior estimates [54]. The final model combines the initial estimate with appropriately weighted corrections from subsequent trees. GBR offers notable advantages, including the ability to model complex, non-linear relationships and a relatively robust resistance to overfitting [55].

$$\boldsymbol{f(x_i) = \sum_{i=1}^{n} \gamma_m DT_m(x_i)} \tag{6}$$

Furthermore, a Deep Neural Network (DNN) structure consists of an input layer, hidden layers, and an output layer, each of which comprises a set of neurons. Unlike Artificial Neural Networks (ANN), which typically have one or two hidden layers, DNNs feature a more significant number of hidden layers. This increased depth then enhances the network ability to generalize complex non-linear relationships between inputs and outputs [56]. The output of a DNN is expressed in Eq. (7), where $x_n$ represents the input, $w_i$ denotes the weights, $b_i$ is the bias, and $f(\blacksquare)$ is the activation function applied to the neurons. A key advantage of the DNN architecture lies in its capacity to capture and model intricate relationships between inputs and outputs [57].

$$\boldsymbol{y_i = \sum_{n=1}^{N} f(w_i \times x_n + b_i), n \in [1, N]} \tag{7}$$

This research employed the tree-parzen structured estimator (TPE) technique for hyperparameter tuning. Optimal hyperparameter determination is able to provide a better prediction model compared to default settings [58]. The TPE technique is an enhancement of conventional Bayesian methods. In conventional Bayesian methods, the surrogate function for determining hyperparameters uses a Gaussian function, whereas TPE employs probability density function (PDF) modelling to separate good and bad hyperparameter sets using kernel density estimation (KDE) [59]. The process begins with the random initialization of hyperparameter combinations, which are then evaluated by an objective function—in this study, the root means square error (RMSE). Subsequently, the set of good and bad hyperparameters is determined using threshold estimation with KDE. A new configuration of the set of good hyperparameters is then found, and re-evaluated using the objective function. This process is performed iteratively n times [60].

This research involved three to four hyperparameters optimized. In the SVR algorithm, three hyperparameters were optimized: regularization parameter (C), kernel, and kernel coefficient (gamma). For the kernel, a single option was used: the radial basis function (RBF). Meanwhile, for C and gamma, a normal distribution was used with a range of 0.1 to 1000 for C, and 0.0001 to 1 for gamma. In the DNN algorithm, there were four hyperparameters optimized: number of layers, number of neurons each layer, activation function, and optimizer. The search space for the activation function was SoftMax and rectified linear unit (relu), as seen in Eq. (8) and Eq. (9), where x is the input vector.

$$softmax(x_i) = \max\left(\frac{e^{x_i}}{\sum_{j=1}^{n} e^{x_j}}\right) \tag{8}$$

$$relu\,(x) = max\begin{pmatrix} 0, & x < 0 \\ x, & x \geq 0 \end{pmatrix} \tag{9}$$

The search spaces for the DNN hyperparameter optimizer are adaptive moment estimation (Adam) and Adamax (a variation of Adam for large parameters). There were four hyperparameters optimized in the GBR algorithm: the maximum number of trees for iteration (max_estimators), the maximum depth of each decision tree (max_depth), learning rate, and subsample. In the RF algorithm, four hyperparameters were optimized: the number of decision trees (max_estimators), the maximum depth of each decision tree (max_depth), the minimum number of samples for a data set to be split again (min_samples_split), and the maximum number of parameters used to build each DT (max_features). The max_features hyperparameter had three options: “sqrt”, meaning that the number is the square root of the number of features; “log2”, meaning that the maximum number of features is log2(x); and None, meaning that all features are used. Table 2 presents the search spaces for each hyperparameter and algorithm.

The prediction model was evaluated by comparing the predicted values with the actual ones and quantifying the results using five evaluation metrics, including correlation coefficient (CC), coefficient of determination ($R^2$), root mean square error (RMSE), mean square error (MSE), and mean absolute error (MAE). The equations, value ranges, and optimal values for each metric are presented in Table 2, where $y_{ai}$ represents the $i$-th actual data point, $y_{pi}$ is the $i$-th predicted data point, $\overline{y_p}$ is the mean of the predicted data, and $\overline{y_a}$ is the mean of the actual data [61].

Table 2. Model evaluation metrics

| Metric | Formula | Range | Best value |
|---|---|---|---|
| CC | $\frac{\sum(y_{ai} - \overline{y_a})(y_{pi} - \overline{y_p})}{\sqrt{\sum(y_{ai} - \overline{y_a})^2}\sqrt{\sum(y_{pi} - \overline{y_p})^2}}$ | -1,1 | 1 |
| $R^2$ | $1 - \frac{\sum_{i=1}^{N}(y_{pi} - y_{ai})^2}{\sum_{i=1}^{N}(y_{ai} - \overline{y_{ai}})^2}$ | 0,1 | 1 |
| MSE | $\frac{1}{N}\sum_{i=1}^{N}(y_{pi} - y_{ai})^2$ | 0, ∞ | 0 |
| RMSE | $\sqrt{MSE}$ | 0, ∞ | 0 |
| MAE | $\frac{1}{N}\sum_{i=1}^{N}\lvert y_{pi} - y_{ai}\rvert$ | 0, ∞ | 0 |

### *2.3.3. Future epoch prediction*

The evaluated prediction model was then utilized to predict future epochs. Since the dataset generation process used data from the previous five epochs to predict a specific epoch, the future epoch prediction process also followed a stepwise approach using five epochs beyond the last observed data. The future prediction operated iteratively: the previous five epochs of data were used to predict one epoch ahead ($y_{t_{last+1}}$). Subsequently, to predict $y_{t_{last+2}}$, the predicted $y_{t_{last+1}}$ was incorporated as $x_{t-1}$. This process continued iteratively until $y_{t_{last+5}}$ was obtained.

## 3. Results and Discussion

### *3.1. Predicted data characteristics*

Precipitation prediction in this study involved thirteen parameters obtained from time series observation data. The prediction was naturally based on previous time series rainfall observations, allowing for the identification of its characteristics and behavioral patterns. The characteristics and behavior of rainfall can be analyzed in great detail based on historical daily precipitation data. In contrast, the analysis of global rainfall characteristics can be identified based on monthly and annual rainfall data. Fig. 4 shows the monthly precipitation characteristics from 2018 to 2023. The visualization shows that precipitation in the study area tended to be highest from December to March with the peak average occurred in February at 13.75 ± 8.68 mm/day. Conversely, the lowest average precipitation was observed in July at 0.79 ± 2.80 mm/day.

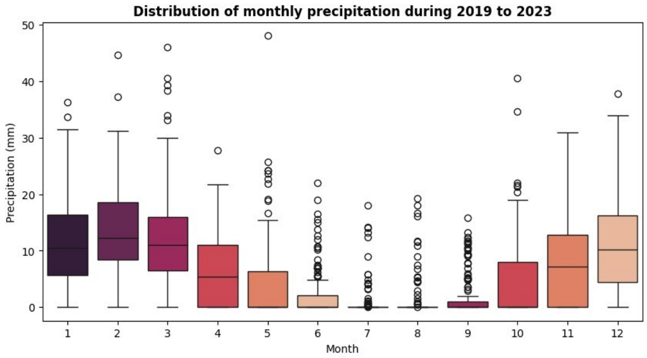

Fig. 4. Monthly Precipitation Characteristics
(data source: CHIRPS accessed and processed using GEE, by the authors)

Fig. 5 shows that the overall distribution of rainfall values remained relatively consistent across the years, with only slight variations in rainfall trends between them. The annual rainfall distribution revealed that 2022 experienced the highest average rainfall at 7.80 ± 7.92 mm/day and the highest maximum yearly rainfall at 48.09 mm/day. Conversely, 2023 recorded the lowest average annual rainfall at 4.79 ± 6.69 mm/day and the lowest maximum yearly rainfall at 31.45 mm/day. These variations were associated to the ENSO phenomenon. In 2022, the ENSO index was predominantly positive, indicating a La Niña event that increased rainfall intensity in the equatorial region, including the study area. In contrast, the ENSO index in 2023 shifted negative, signifying an El Niño event, reducing rainfall intensity in the region. These results are consistent with previous research, which linked local rainfall phenomena with the global phenomena of El Niño and La Niña [13,6].

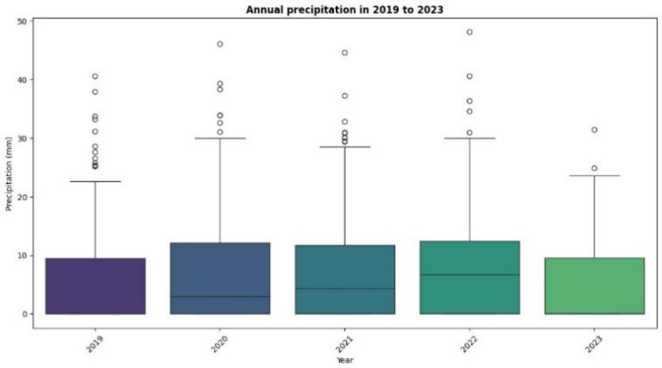

Fig. 5. Characteristics of annual rainfall
(data source: CHIRPS accessed and processed using GEE, by the authors)

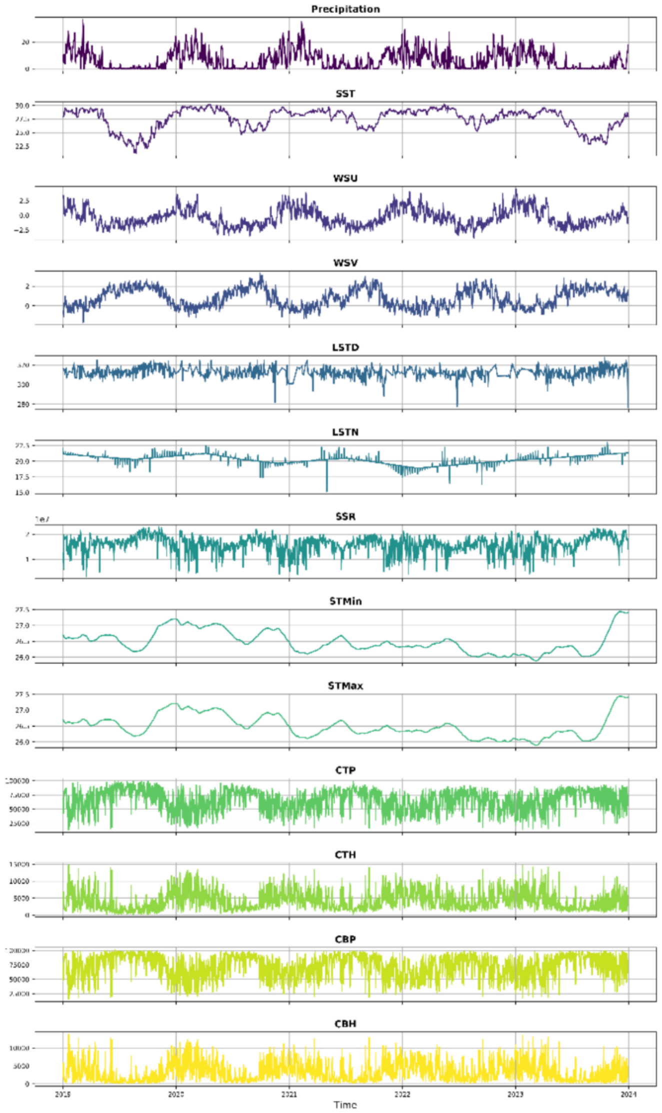

Fig. 6. Prediction data variables

The time-series behavior of each variable is depicted in Fig. 6. To determine the relationship between each variable and rainfall in the study area, correlation analysis was performed. The results are presented as a correlation heatmap in Fig. 7. The first row of the heatmap illustrates the correlation between rainfall values and other variables [62]. Based on this visualization, all variables exhibited low to moderate correlations with rainfall with both positive and negative relationships observed. Of 12 variables, 8 showed a moderate correlation with rainfall including SST, WSU, CTH, and CBH with positive correlations, and WSV, SSR, CTP, and CBP with negative ones. In contrast, both LST (day and night) and STMin/STMax had low correlations with rainfall in the study area with absolute correlation values below 0.1.

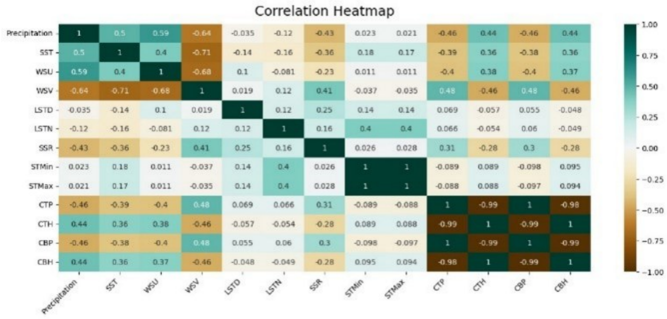

Fig. 7. Correlation heatmap matrix among variables

The characteristics of each data serve as the essential variable in the multivariate rainfall prediction modeling. Ideally, selecting variables with moderate correlations (both positive and negative) to rainfall can enhance the accuracy of the predictions [62]. Based on the correlation analysis, WSU was found as the variable with the strongest positive correlation to rainfall and a correlation coefficient of 0.59. Conversely, WSV became the variable with the strongest negative correlation to rainfall with a correlation coefficient of -0.64.

The results of correlation test indicated no significant relationship between rainfall pattern similarity and the spatial resolution of the data used. Of all components, temperature—including Land Surface Temperature (LSTD & LSTN) and Soil Temperature (STMin & STMax)—showed the weakest correlation with rainfall and most other components. In contrast, cloud properties (CTH, CTP, CBH, and CBP) demonstrated a moderate correlation with wind speed (u and v components), shortwave radiation (SSR), sea surface temperature (SST), and precipitation.

### 3.2. Results of multivariate rainfall prediction modeling

Multivariate rainfall prediction was conducted using variables with an absolute correlation value greater than 0.4 [37,38]. Consequently, only nine variables were included: rainfall, SST, WSU, WSV, SSR, CTH, CTP, CBH, and CBP. For each variable, data from the previous five-time steps (t-1, t-2, t-3, t-4, and t-5) were used as inputs for prediction. This resulted in a total of 45 predictive variables. The algorithms employed in this study were SVR, DNN, GBR, and RF, all of which were implemented with best hyperparameter architecture using TPE results.

Table 3 shows the search spaces and optimal hyperparameters for each algorithm. The optimal hyperparameters were selected based on the combination that produced the minimum objective function value. In this study, there were 50 iterations for performing the optimization process using TPE. Hence, 50 experiments were performed to find an optimal value for each hyperparameter. The TPE technique is able to reduce computational load as it only performs n-experiments to find the optimal hyperparameters, rather than trying each combination of hyperparameters one by one.

Table 3. Hyperparameter tuning result using TPE each algorithm

| Hyperparameter | Search spaces | Optimum value |
|---|---|---|
| | SVR algorithm | |
| Regularization parameter (C) | uniform (0.1, 1000) | 963.969 |
| Kernel | 'rbf' | 'rbf' |
| Kernel coefficient (gamma) | uniform (1e4, 1) | 0.045 |
| | DNN algorithm | |
| n_layers_dnn | 6,7,8,9,10 | 6 |
| n_unit_neurons | 64, 128, 256, 512 | 512 |
| activation_function | 'relu', 'softmax' | 'relu' |
| Optimizer | 'Adam', 'Adamax' | 'Adam' |
| | GBR algorithm | |
| n_estimators | range (100,1000,200) | 400 |
| max_depth | range (3,10,1) | 5 |
| Learning_rate | uniform (0.01, 0.2) | 0.040 |
| subsample | uniform (0.5, 1) | 0.763 |
| | RF algorithm | |
| n_estimators | range (100,1000,200) | 100 |
| max_depth | range (3,10,1) | 9 |
| min_samples_split | range (2,10,1) | 5 |
| max_features | 'sqrt', 'log2', None | None |

The models generated by each algorithm were applied to both training and testing datasets. Fig. 8 illustrates the application of each algorithm to the training data with a focus on rainfall data from January to March 2022 for more precise visualization. The results showed that most algorithms effectively captured the general pattern of the actual data during training. Of four algorithms, GBR provided the best visual fit, showing more minor discrepancies between predicted and actual values compared to others.

The prediction results obtained through the DNN algorithm on the training data exhibited a relatively larger gap compared to actual data relative to other models. Other algorithms could consistently learn daily rainfall patterns effectively, particularly the GBR model. Most time steps were able to predict with great precision against the actual data. This condition was not entirely favorable as the prediction results on the testing data must be re-investigated. Since the gap with the actual data was very small, the GBR model could generalize the data well. The time-step segments in Fig. 9 showed that the DNN model tended to produce lower prediction values compared to the actual data with the most significant gap occurred during the first 15 days of March 2023.

Fig. 9 shows the results of applying each algorithm's prediction model to the testing data. Similar to the training data visualization, the comparison between actual and predicted values was focused on the time period from January to March 2023. The visualization indicated that all prediction models produced patterns closely aligned with the actual data, suggesting that they are generally reliable when applied to new data. In the time-step segment, the SVR model was found consistent in predicting actual data with greater precision than other models. While the prediction values GBR model were nearly aligned with actual values in the training data visualization, it showed a significant gap compared to the SVR model in the testing data. Visually, the GBR model was found less consistent in accurately predicting actual values. The DNN model exhibited a similar pattern in the testing data with a relatively larger gap compared to other prediction models. Prediction values in the testing data tended to be lower than actual values.

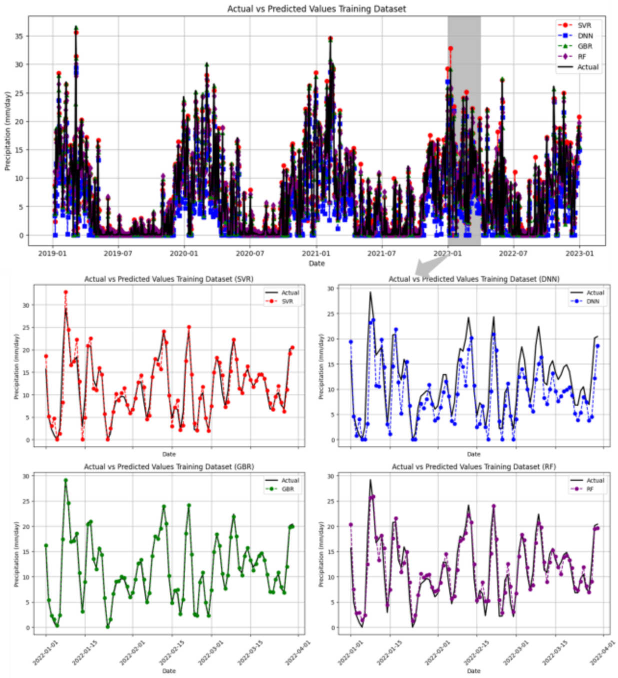

Fig. 8. Prediction results of each algorithm on the training data

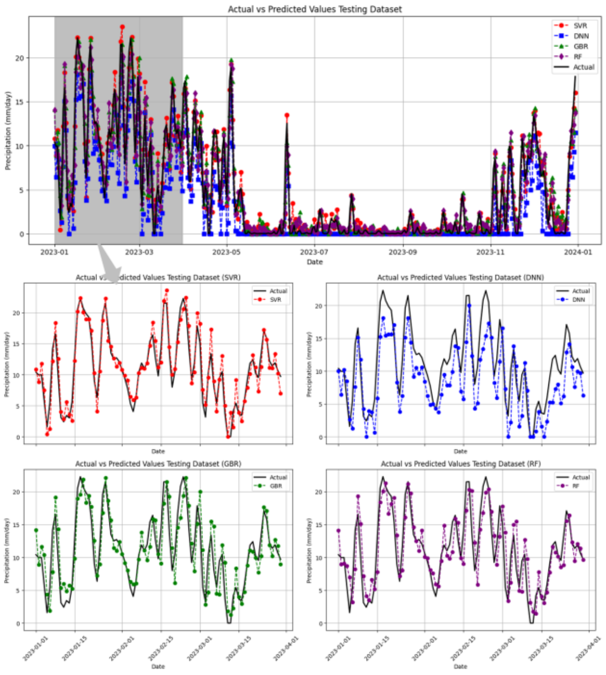

Fig. 9. Prediction results of each algorithm on the testing data

The difference between actual and predicted values was quantified by the residuals, representing a difference between the actual and predicted data. Fig. 10 illustrates the residual distribution for each algorithm on the training data. Of the algorithms, the SVR algorithm had the smallest median value of -3.116×10$^{-3}$, while RF had the highest one at -0.727. Although SVR had the smallest median value, GBR had the smallest standard deviation with a value of ±0.41, meaning it had the most stable relative bias. This can be seen in how the outlier points in the GBR model clustered near the interquartile range (IQR) and formed a shorter boxplot compared to other models. The median value in the GBR model was -0.034. DNN was the model with the highest median and standard deviation values with the values of 1.726 and ±2.492, respectively. This can be visually seen where the DNN model formed a relatively long boxplot compared to other models.

The residual values of each algorithm for the testing data are illustrated using a box plot in Fig. 11. Of the models, SVR remained the model with the lowest median residual value of 0.538. In addition, it was the one with the lowest residual standard deviation of ±0.998. The boxplot generated by the SVR model was visually shorter than those generated by other models. In contrast, the DNN model had the highest median and standard deviation values with the values of 0.950 and ±2.132, respectively. This model visually formed the longest boxplot of other models. Although the GBR model had better residual stability on the training data, when compared to the testing data, SVR outperformed the residual stability.

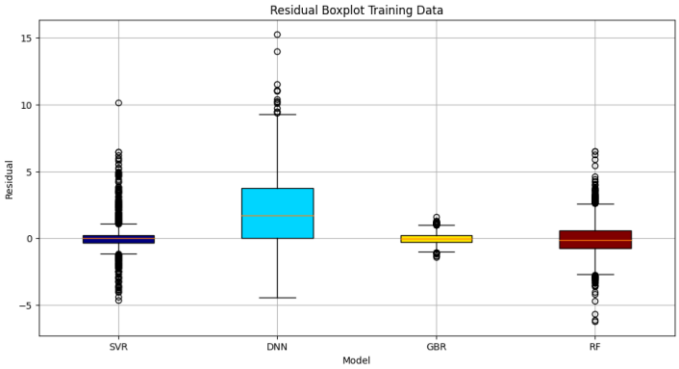

Fig. 10. Residuals of each algorithm on the training data

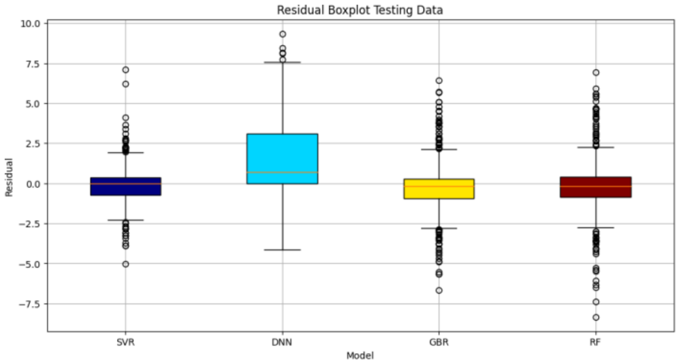

Fig. 11. Residuals of each algorithm on the testing data

### 3.3. Prediction model evaluation

The prediction model was evaluated using both training and testing data, employing five evaluation metrics: CC, R², RMSE, MAE, and MSE. The results for each algorithm and dataset are summarized in **Error! Reference source not found.**. On the t raining data, the SVR algorithm achieved the highest CC and R² values, 0.982 and 0.965, respectively, indicating its strong correlation and goodness-of-fit. Conversely, the DNN algorithm recorded the lowest ones, 0.943 and 0.932, respectively. For error-based metrics (RMSE, MAE, and MSE), GBR demonstrated superior performance with the lowest values at 0.426, 0.332, and 0.182, respectively. In contrast, DNN exhibited the highest ones at 3.363, 2.351, and 11.312, respectively.

Table 4. Evaluation Metrics for Each Algorithm

| Metric | Dataset | SVR | DNN | GBR | RF |
|---|---|---|---|---|---|
| CC | Training | 0.982 | 0.943 | 0.998 | 0.982 |
| | Test | 0.974 | 0.932 | 0.951 | 0.943 |
| $R^2$ | Training | 0.965 | 0.890 | 0.996 | 0.965 |
| | Test | 0.948 | 0.869 | 0.905 | 0.890 |
| RMSE | Training | 1.310 | 3.363 | 0.426 | 1.349 |
| | Test | 1.366 | 2.804 | 1.833 | 1.963 |
| MAE | Training | 0.788 | 2.351 | 0.332 | 0.977 |
| | Test | 0.947 | 1.824 | 1.237 | 1.299 |
| MSE | Training | 1.717 | 11.312 | 0.182 | 1.818 |
| | Test | 1.866 | 7.863 | 3.359 | 3.855 |

The SVR algorithm performed better when evaluated with testing data. It had the highest CC and R² values of 0.974 and 0.948, respectively but showed the lowest error-based metrics with RMSE, MAE, and MSE values of 1.366, 0.947, and 1.866, respectively. While, the DNN algorithm consistently demonstrated the lowest performance with CC and R² values of 0.932 and 0.890, respectively. The RMSE, MAE, and MSE values for the DNN model were 2.804, 2.351, and 7.863, respectively. These results highlighted the variability in algorithm performance between training and testing datasets (see 4).

The findings of this research indicated that the SVR algorithm was the best algorithm compared to the DNN, GBR, and RF algorithms. It consistently produced prediction values very close to the actual data. A study by Nayak et al. (2025) produced the RMSE values of 0.912 to 1.091 for monthly rainfall predictions using the SVR algorithm [63]. In contrast, Wang et al. (2023) showed that SVR performed worse than RF and Bayesian ridge regression (BRR) for univariate rainfall prediction [64]. The GBR algorithm outperformed the SVR algorithm on the training data, but did not consistently outperform it on the testing data evaluation. Similar to the study by Sumith (2025), GBR had an almost perfect R² of 0.95 on the training data, but when evaluated on the testing data it had an R² value of 0.45 [65]. Nevertheless, in this study the difference in evaluation results between the training and testing data for the GBR model was less significant with an R² value on the testing data greater than 0.9. It was because the GBR algorithm continuously iterated residual values, allowing the model to closely approximate actual values in the testing data; as a consequence, the model performed poorly in generalizing unseen data [66]. The DL algorithm in this study performed less well than the ML algorithm. It did not adequately learn the daily rainfall fluctuations by considering multiple variables. However, when using a very large dataset the DL algorithm can perform well [67]. Therefore, a longer training data series is needed to learn the rainfall fluctuations in the study area.

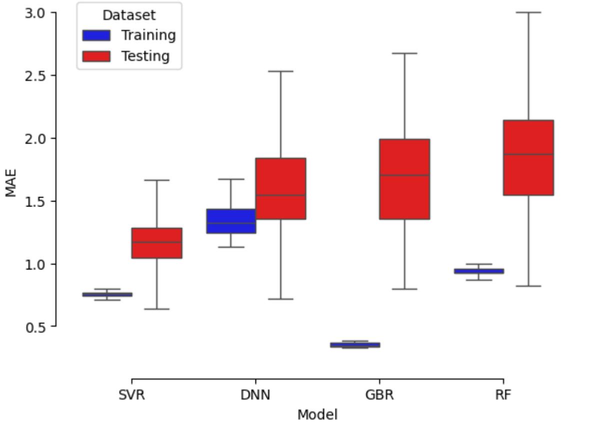

Fig. 12. Box-plot of KFCV results evaluated by MAE

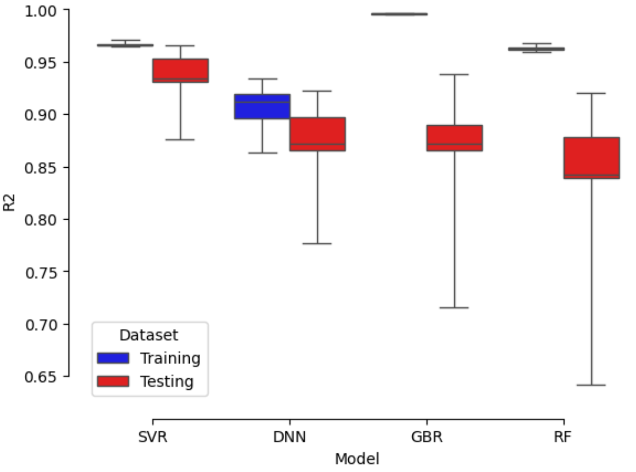

Fig. 13. Box-plot of KFCV results evaluated by $R^2$

This research employed k-fold cross validation (KFCV) to evaluate the stability and robustness of each prediction model. KFCV was conducted by setting the number of folds to 10, thereby dividing the entire dataset into 10 folds of equal size, with 1-fold designated as the testing data and the remaining 9 folds as the training data. The fold used as testing data was used sequentially, and the model was consistently fitted to the training fold in combination with other nine folds. The results of the KFCV process are visualized in the form of box plots, as shown in Fig. 12 and Fig. 13. The metrics used for KFCV were MAE and R². In the box-plot visualizations, the x-axis represents the prediction model, and the y-axis represents the metric value, with each model grouped into training and testing datasets. The training dataset was visualized with a blue box, and the testing data was shown in red. The whiskers represented the range of minimum and maximum metric values. The results of KFCV showed that the MAE value of the GBR model and training data tended to have the smallest MAE value compared to other models and exhibited high stability, as indicated by the short whiskers produced. The MAE value of the GBR model's training data was 0.35±0.02. Meanwhile, when all models were evaluated using the testing data, the SVR model was found more robust, producing the smallest average value and relatively higher stability compared to other models. The short whiskers in the boxplot represented these characteristics. The average MAE value for the SVR model was 1.14 ± 0.32. The KFCV $R^2$ metric results followed the same pattern as the MAE. The GBR model had the highest value with high robustness at 1.00 ± 0.00. Meanwhile, the SVR model had the highest $R^2$ and robust on the testing data with a value of 0.94 ± 0.02. Therefore, in the present study, the SVR model demonstrated both the best performance and strong robustness across various splitting scenarios.

The performance of each prediction model was further evaluated using a Taylor diagram, visually representing the relationship and comparison between prediction and actual data. A Taylor diagram includes three main components: the radial axis, which indicates the standard deviation of the data; isolines, which denote the RMSE values; and the quarter-circle arcs, which represent the CC. The reference point on the diagram corresponds to the actual values, serving as the benchmark for assessing the closeness of the predicted values to the actual data. The Taylor diagrams for the training and testing data are illustrated in Fig. 14 and Fig. 15, respectively.

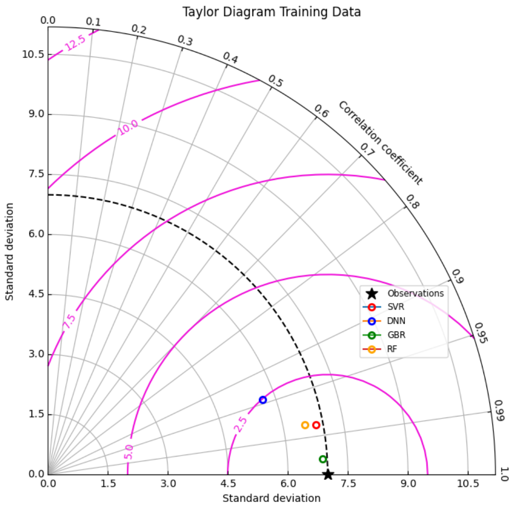

Fig. 14. Taylor diagram for training data

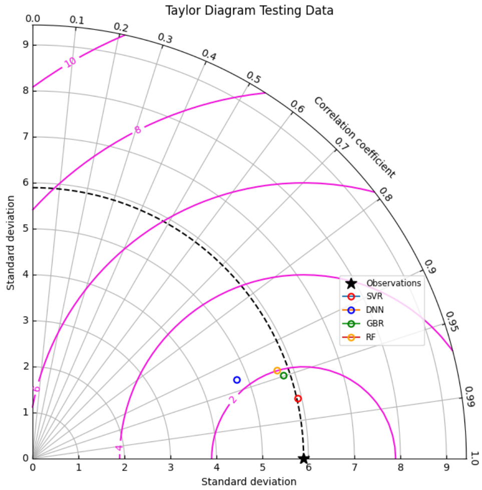

Fig. 15. Taylor diagram for testing data

Fig. 14 shows that all algorithms exhibited a strong correlation with the actual data on the training dataset (CC>0.9). The GBR algorithm had the closest distance to the actual standard deviation value, the lowest RMSE value, and the highest correlation. In contrast, the DNN algorithm had the opposite characteristics, namely the furthest distance from the actual standard deviation value, the highest RMSE, and the lowest correlation.

The Taylor diagram for the testing data (Fig. 15) reveals that SVR not only maintained low RMSE and high CC values but also aligned closely with the standard deviation reference line of the actual data. Based on both quantitative and visual evaluations, the SVR algorithm outperformed other algorithms in this study, showcasing its robust and reliable predictive capabilities.

This research explored the potential for overfitting using learning curves. A learning curve is a graph that illustrates the relationship between the percentage of total training data used and the resulting accuracy, both on new training data (taken from 0-100% of the training data) and testing data. The testing data used to evaluate each scenario was the testing data resulting from an 80:20 split in the initial process. Meanwhile, the fraction of the training data was determined based on the 80:20 split during the initial process. In each scenario, the model was fitted with the resulting training data fraction. Fig. 16 presents the learning curve analysis results for each model that was visualized by different colors. Meanwhile, the difference between the training and testing data results for each algorithm was visualized by the different types of lines, where training data and testing data were represented by a solid line and a dashed line, respectively. The metric used for learning curve analysis was $R^2$. The results obtained then showed that the more training data used (the data fraction approaching 1), the more convergent the graph between training and testing data for each model. This indicated no significant gap between the training and testing data. However, when compared relatively across models, the RF and GBR algorithms showed the highest gap between training and testing data compared to SVR and DNN, indicating the potential of these two models for overfitting. Nevertheless, the $R^2$ value for the testing data of both algorithms was above 0.8, meaning the models are still able to make good predictions on unseen data.

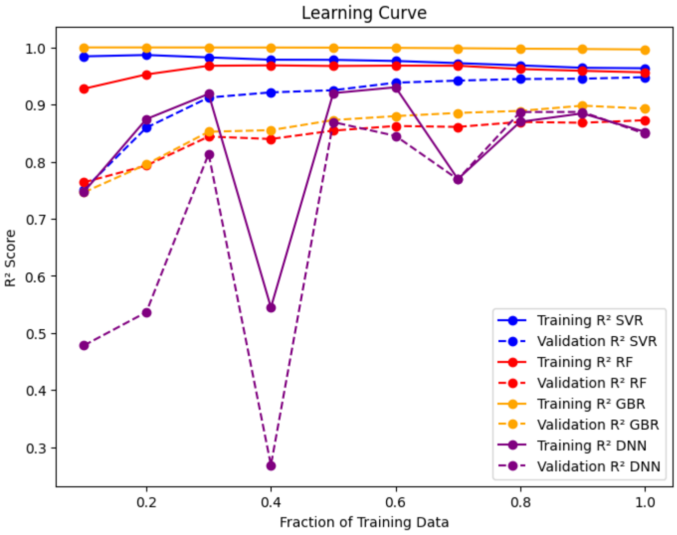

Fig. 16. Learning curve each model

### 3.4. Future epoch prediction

Subsequently, the prediction model was applied to forecast rainfall values for future epochs. Five future epochs were set to align with the number of lags used in the dataset generation process. This corresponds to predicting rainfall for the next five days. Future epoch forecasts were made recursively, where the output at epoch t+1 became the input for the forecast at epoch t+2, and thus onward. Fig. 17 illustrates the forecasting results for these five future epochs, as produced by each algorithm. The data were visualized in a line chart accompanied by a box plot to illustrate the distribution of data predictions across models.

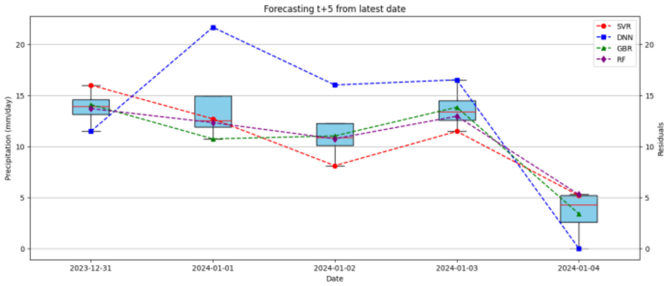

Fig. 17. Future forecasting for each epoch

Predictions were generated for the period from December 31, 2023, to January 5, 2024. During epoch t+1, the four algorithms predicted rainfall closely, as indicated by the small box plots generated. The standard deviation value at epoch t+1 was ±1.862. However, during epochs t+2 to t+5, the DNN algorithm had prediction values relatively far from the SVR, RF, and GBR prediction values. A significant difference was observed at epoch t+2, with a standard deviation of ±4.952. The t+2 epoch had the highest average predicted rainfall at 14.362 mm. While, the lowest rainfall was recorded at t+5 with an average of 3.467 mm.

## 4. Conclusion

This research evaluated the use of open-access remote sensing data and artificial intelligence algorithms for predicting daily rainfall in the Gajahwong watershed area, Yogyakarta, Indonesia. Rainfall prediction was based on the trend of rainfall events (univariate) and was made by considering a number of variables, including SST, WSU, WSV, STMin, STMax, SSR, LSTD, LSTN, CTP, CTH, CBP, and CBH. The results indicated that 7 out of 12 variables showed good correlation coefficients during the feature selection process. These seven variables were then used in the prediction model. The rainfall forecasting in this study considered 5-step epochs before the data at epoch t, leading to a total of 40 variables (5 epochs multiplied by 7 plus 1 variable). The artificial intelligence algorithms employed ranged from basic machine learning (SVR), ensemble machine learning (RF and BGR), to deep learning (DNN). Based on the modeling results, GBR demonstrated a very high performance on the training data, achieving an $R^2$ value of 0.996. However, when evaluated on the testing data, it did not consistently outperform other algorithms with an $R^2$ value of 0.905. Conversely, SVR achieved the highest performance with the testing data, obtaining an $R^2$ of 0.948. Overall, this study highlighted SVR as the best-performing algorithm and the one that was resistant to overfitting. The limitation of this research is that it merely provided predictions in the time domain, rather than spatial domain predictions. Nonetheless, the results are promising for daily rainfall predictions in the study area, utilizing open-access remote sensing big data. Despite differences in the spatial resolution of the data, the accuracy of the predictions is commendable. It is essential that the data have the similar temporal resolution (daily). This research is expected to apply across various sectors, particularly hydro-meteorological disaster management.

## Acknowledgements

We would like to extend our heartfelt acknowledgments to the research team of "Smart Sustainable Urban Flood Management: A Decision Support System Towards a Sensing City – Lessons Learned from the Surabaya Case for IKN Development", for their collaboration on this joint research. Our deepest appreciation also goes to the Faculty of Engineering, Universitas Gadjah Mada (UGM), for providing research funding, facilities, and various forms of support that have facilitated the successful progress of this research.

## References

1. Syahza, A., Suwondo, Bakce, D., Nasrul, B., Mustofa, R. *Utilization of peatlands based on local wisdom and community welfare in Riau Province, Indonesia*. Int. J. Sustain. Dev. Plan. 15 (2020) 1119–1126.
2. Danladi, A., Stephen, M., Aliyu, B.M., Gaya, G.K., Silikwa, N.W., Machael, Y. *Assessing the influence of weather parameters on rainfall to forecast river discharge based on short-term*. Alexandria Eng. J. 57 (2018) 1157–1162.
3. Chen, H., Shao, M., Li, Y. *The characteristics of soil water cycle and water balance on steep grassland under natural and simulated rainfall conditions in the Loess Plateau of China*. J. Hydrol. 360 (2008) 242–251.
4. Gerrits, A.M.J. *The role of biodiversity in the hydrological cycle*, Delft University of Technology, 2016.
5. Latif, S.D., Alyaa Binti Hazrin, N., Hoon Koo, C., Lin Ng, J., Chaplot, B., Feng Huang, Y., El-Shafie, A., Najah Ahmed, A. *Assessing rainfall prediction models: Exploring the advantages of machine learning and remote sensing approaches*. Alexandria Eng. J. 82 (2023) 16–25.
6. Novitasari, D.C.R., Rohayani, H., Suwanto, Arnita, Rico, Junaidi, R., Setyowati, R.D.N., Pramulya, R., Setiawan, F. *Weather Parameters Forecasting as Variables for Rainfall Prediction using Adaptive Neuro Fuzzy Inference System (ANFIS) and Support Vector Regression (SVR). J.* Phys. Conf. Ser. (2020) 1501.
7. Basith, A., Nuha, M.U., Prastyani, R., Winarso, G. *Aerosol optical depth (AOD) retrieval for atmospheric correction in Landsat-8 imagery using second simulation of a satellite signal in the solar spectrum-vector (6SV)*. Commun. Sci. Technol. 4 (2019) 68–73.
8. Thoha, A.S., Saharjo, B.H., Boer, R., Ardiansyah, M. *Characteristics and causes of forest and land fires in Kapuas district, Central Kalimantan Province, Indonesia*. Biodiversitas 20 (2019) 110–117.
9. Nurfaida, W., Ramdhani, H., Shimozono, T., Triawati, I., Sulaiman, M. *Rainfall trend and variability over Opak River basin, Yogyakarta, Indonesia*. J. Civ. Eng. Forum 1000 (2020).

10. Wani, O.A., Mahdi, S.S., Yeasin, M., Kumar, S.S., Gagnon, A.S., Danish, F., Al-Ansari, N., El-Hendawy, S., Mattar, M.A. *Predicting rainfall using machine learning, deep learning, and time series models across an altitudinal gradient in the North-Western Himalayas*. Sci. Rep. 14 (2024).
11. Pathan, A.I., Agnihotri, P.G., Patel, D., Prieto, C. *Mesh grid stability and its impact on flood inundation through (2D) hydrodynamic HEC-RAS model with special use of Big Data platform—a study on Purna River of Navsari city.* Arab. J. Geosci. 15 (2022).
12. Granata, F., Gargano, R., de Marinis, G. *Support vector regression for rainfall-runoffmodeling in urban drainage: A comparison with the EPA's storm water management model*. Water 8 (2016).
13. Singh, V., Qin, X. *Study of rainfall variabilities in Southeast Asia using long-term gridded rainfall and its substantiation through global climate indices*. J. Hydrol. 585 (2020).
14. Liyew, C.M., Melese, H.A. *Machine learning techniques to predict daily rainfall amount*. J. Big Data 8 (2021).
15. Pramudia, A., Misnawati, Awanis, Sabur, A., Hidayanto, M., Sri Ratmini, N.P., Dewi, D.O., Agustini, S., Fiana, Y., Bhermana, A. *Strengthening the Agroclimatology Analysis against Local Wisdom Paddy Planting Time at Coastal Area in Indonesia*. IOP Conf. Ser. Earth Environ. Sci. 1095 (2022).
16. Kurniadi, A., Weller, E., Salmond, J., Aldrian, E. *Future projections of extreme rainfall events in Indonesia*. Int. J. Climatol. 44 (2024) 160–182.
17. Mislan, Haviluddin, Hardwinarto, S., Sumaryono, Aipassa, M. *Rainfall Monthly Prediction Based on Artificial Neural Network: A Case Study in Tenggarong Station, East Kalimantan - Indonesia*. Procedia Comput. Sci. 59 (2015) 142–151.
18. Sarasa-Cabezuelo, A. *Prediction of Rainfall in Australia Using Machine Learning*. Inf. 13 (2022).
19. Cahyono, B.K., Aditya, T., Istarno *The Least Square Adjustment for Estimating The Tropical Peat Depth using LiDAR Data*. Remote Sens. 12 (2020) 1–22.
20. Amini, A., Dolatshahi, M., Kerachian, R. *Real-time rainfall and runoff prediction by integrating BC-MODWT and automatically-tuned DNNs: Comparing different deep learning models*. J. Hydrol. 631 (2024) 130804.
21. Tharun, V.P., Ramya, P., Renuga Devi, S. A *Univariate Data Analysis Approach for Rainfall Forecasting*. Lect. Notes Networks Syst.204 (2021) 669–689.
22. Peña, D., Sánchez, I. *Measuring the advantages of multivariate vs. univariate forecasts*. J. Time Ser. Anal. 28 (2007) 886–909.
23. Salehi, S., Kavgic, M., Bonakdari, H., Begnoche, L. *Comparative study of univariate and multivariate strategy for short-term forecasting of heat demand density: Exploring single and hybrid deep learning models*. Energy AI 16 (2024) 100343.
24. Adhani, G., Buono, A., Faqih, A. *Support Vector Regression modelling for rainfall prediction in dry season based on Southern Oscillation Index and NINO3.4*. Int. Conf. Adv. Comput. Sci. Inf. Syst. (2013) 315–320.
25. Kumar, L., Mutanga, O. *Google Earth Engine Applications.* Remote Sens. (2019).
26. Karimi, P., Bastiaanssen, W.G.M. *Spatial evapotranspiration, rainfall and land use data in water accounting - Part 1: Review of the accuracy of the remote sensing data*. Hydrol. Earth Syst. Sci. 11 (2015) 1073–1123.
27. De Graaf, M., De Haan, J.F., Sanders, A.F.J. *TROPOMI ATBD of the Aerosol Layer Height*, Paris, 2019.
28. Li, H., Li, S., Ghorbani, H. *Data-driven novel deep learning applications for the prediction of rainfall using meteorological data*. Front. Environ. Sci. 12 (2024) 1–15.
29. Rincón-Avalos, P., Khouakhi, A., Mendoza-Cano, O., López-De la Cruz, J., Paredes-Bonilla, K.M. *Evaluation of satellite precipitation products over Mexico using Google Earth Engine*. J. Hydroinformatics 24 (2022) 711–729.
30. Kan, J. *Predicting Drought Hazard In Sweden Using Google Earth Engine And Machine Learning Approach*, KTH Royal Institute of Technology, 2022.
31. Cahyono, B.K., Aditya, T., Istarno *The Determination of Priority Areas for the Restoration of Degraded Tropical Peatland Using Hydrological, Topographical, and Remote Sensing Approaches*. Land 11 (2022).
32. Suprayogi, S., Purnama Sari, S., Setiacahyandari, H.K. *Flood Risk Analysis in Gajah Wong River, Yogyakarta City*. J. Ilmu Lingkung. 22 (2024) 1033–1040.
33. Funk, C., Peterson, P., Landsfeld, M., Pedreros, D., Verdin, J., Shukla, S., Husak, G., Rowland, J., Harrison, L., Hoell, A. *The climate hazards infrared precipitation with stations - A new environmental record for monitoring extremes*. Sci. Data 2 (2015) 1–21.
34. Reynolds, R.W., Banzon, V.F., *NOAA Optimum Interpolation 1/4 Degree Daily Sea Surface Temperature (OISST) Analysis*, NOAA. 2008.
35. Johnson, S.J., Stockdale, T.N., Ferranti, L., Balmaseda, M.A., Molteni, F., Magnusson, L., Tietsche, S., Decremer, D., Weisheimer, A., Balsamo, G. *SEAS5: The new ECMWF seasonal forecast system*. Geosci. Model Dev. 12 (2019) 1087–1117.
36. Asuero, A.G., Sayago, A., González, A.G. *The Correlation Coefficient: An Overview*. Crit. Rev. Anal. Chem. - CRIT REV ANAL CHEM. 36 (2006) 41–59.
37. Schober, P., Schwarte, L.A. *Correlation coefficients: Appropriate use and interpretation*. Anesth. Analg. 126 (2018) 1763–1768.
38. Car, F.R.A.M., Sugeng Subagio, B., Rahman, H., Care, F., Subagio, B.S., Rahman, H. *Porous concrete basic property criteria as rigid pavement base layer in indonesia*. MATEC Web Conf. 147 (2018) 2008.
39. Izonin, I., Tkachenko, R., Shakhovska, N., Ilchyshyn, B., Singh, K.K. *A Two-Step Data Normalization Approach for Improving Classification Accuracy in the Medical Diagnosis Domain.* Mathematics 10 (2022) 1–18.
40. de Amorim, L.B. V, Cavalcanti, G.D.C., Cruz, R.M.O. *The choice of scaling technique matters for classification performance*. Appl. Soft Comput. 133 (2023) 1–37.
41. Yadav, G., Yadav, D.K., Chandra Mouli, P.V.S.S.R. *Chapter 4 - Statistical measures for Palmprint image enhancement. In Cognitive Data Science in Sustainable Computing*, Academic Press, (2022) 65–85.
42. Parmann, L.D., Paarmann, L.D. *Design and analysis of analog filters: a signal processing perspective*, Springer Science & Business Media: New York, 617 (2001).
43. Narejo, S., Jawaid, M.M., Talpur, S., Baloch, R., Pasero, E.G.A. *Multi-step rainfall forecasting using deep learning approach*. PeerJ. Comput. Sci. 7 (2021) e514.
44. Heddam, S., Kim, S., Danandeh Mehr, A., Zounemat-Kermani, M., Elbeltagi, A., Malik, A., Kisi, O. *Chapter 11 - A long short-term memory deep learning approach for river water temperature prediction. In Intelligent Data-Centric Systems*, Academic Press, (2022) 243–270.
45. Sui, X., He, S., Vilsen, S.B., Meng, J., Teodorescu, R., Stroe, D.I. *A review of non-probabilistic machine learning-based state of health estimation techniques for Lithium-ion battery*. Appl. Energy 300 (2021) 117346.
46. Mesut, B., Başkor, A., Buket Aksu, N. *Chapter 3 - Role of artificial intelligence in quality profiling and optimization of drug products*. Academic Press, (2023) 35–54.
47. Umoh, U.A., Eyoh, I.J., Murugesan, V.S., Nyoho, E.E. *Chapter 14 - Fuzzy-machine learning models for the prediction of fire outbreaks: A comparative analysis.* Academic Press, (2022) 207–233.
48. Wang, H., Liu, Y., Zhou, B., Li, C., Cao, G., Voropai, N., Barakhtenko, E. *Taxonomy research of artificial intelligence for deterministic solar power forecasting*. Energy Convers. Manag. 214 (2020) 112909.
49. Marimuthu, R., Shivappriya, S.N., Saroja, M.N. *Chapter 14 - A study of*

*machine learning algorithms used for detecting cognitive disorders associated with dyslexia*. Academic Press, (2021) 245–262.

50. Breiman, L. Random forests. *Mach. Learn.* 45 (2001) 5–32.
51. Rodriguez-Galiano, V., Mendes, M.P., Garcia-Soldado, M.J., Chica-Olmo, M., Ribeiro, L. *Predictive modeling of groundwater nitrate pollution using Random Forest and multisource variables related to intrinsic and specific vulnerability: A case study in an agricultural setting (Southern Spain)*. Sci. Total Environ.476 (2014) 189–206.
52. Nurwatik, N., Ummah, M.H., Cahyono, A.B., Darminto, M.R., Hong, J.-H. *A Comparison Study of Landslide Susceptibility Spatial Modeling Using Machine Learning*. ISPRS Int. J. Geo-Information 11 (2022) 602.
53. Friedman, J.H. *Greedy function approximation: a gradient boosting machine*. Ann. Stat. (2001) 1189–1232.
54. Otchere, D.A., Ganat, T.O.A., Ojero, J.O., Tackie-Otoo, B.N., Taki, M.Y. *Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions*. J. Pet. Sci. Eng. 208 (2022) 109244.
55. Khan, A.A., Chaudhari, O., Chandra, R. *A review of ensemble learning and data augmentation models for class imbalanced problems: Combination, implementation and evaluation*. Expert Syst. Appl. 244 (2024) 122778.
56. Hoang, N.D., Tran, V.D. *Deep Neural Network Regression with Advanced Training Algorithms for Estimating the Compressive Strength of Manufactured-Sand Concrete*. J. Soft Comput. Civ. Eng. 7 (2023) 114–134.
57. Gao, B., He, Y., Chen, X., Chen, H., Yang, W., Zhang, L. *A Deep Neural Network Framework for Landslide Susceptibility Mapping by Considering Time-Series Rainfall*. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. 17 (2024) 5946–5969.
58. Khan, H.M., Khan, A., Villar, S.G., Lopez, L.A.D., Almaleh, A., Al-Qahtani, A.M. *A Comparative Study of Optimized-LSTM Models Using Tree-Structured Parzen Estimator for Traffic Flow Forecasting in Intelligent Transportation*. Comput. Mater. Contin. 83 (2025) 3369–3388.
59. Kraemer, R., Duzgol, O., Li, S., Calabretta, N. *Data-Driven SOA Parameter Discovery and Optimization Using Bayesian Machine Learning With a Parzen Estimator Surrogate*. J. Light. Technol. 42 (2024) 721–731.
60. Bergstra, J., Bardenet, R., Bengio, Y., Kégl, B. *Algorithms for hyper-parameter optimization*. Adv. Neural Inf. Process. Syst. 24 (2011).
61. Plevris, V., Solorzano, G., Bakas, N., Ben Seghier, M. *Investigation of performance metrics in regression analysis and machine learning-based prediction models*, The 8th European Congress on Computational Methods in Applied Sciences and Engineering. (2022).
62. Raniprima, S., Cahyadi, N., Monita, V. *Rainfall Prediction Using Random Forest and Decision Tree Algorithms*. J. Informatics Commun. Technol. 6 (2024) 110–119.
63. Nayak, K., Nayak, S.K., Shivarama, S.B. *Rainfall prediction using support vector regression in Udupi region Karnataka, India*. Telkomnika (Telecommunication Comput. Electron. Control. 23 (2025) 166–174.
64. Wang, Y., Pei, L., Wang, J. *Precipitation prediction in several Chinese regions using machine learning methods*. Int. J. Dyn. Control 12 (2024) 1180–1196.
65. Sumith, K. V. *Comprehensive Evaluation of Satellite-Based Rainfall Measurements Through Rain Gauge Validation Using Advanced Statistical Regression and Machine Learning Models by Using Python*. Water Resour. Manag. (2025).
66. Agapitos, A., Brabazon, A., O'Neill, M. *Regularised gradient boosting for financial time-series modelling*. Comput. Manag. Sci. 14 (2017) 367–391.
67. Sarker, I.H. *Machine learning: algorithms, real-world applications and research directions*. SN Comput. Sci. 160 (2021).