JSCDM

JOURNAL OF SOFT COMPUTING AND DATA MINING
e-ISSN: 2716-621X

Vol. 7 No. 1 (2026) 22-36
https://publisher.uthm.edu.my/ojs/index.php/jscdm

Evaluating Rainfall Predictions using Statistical Decomposition and Neural Network–Based Approaches

Nur Afiqah Abdul Rahim[1]*, Shuhaida Ismail[1], Azme Khamis[2], Muhammad Usman Tariq[3]

[1] *Faculty of Computer Science & Information Technology, Universiti Tun Hussein Onn Malaysia, Batu Pahat, Johor, MALAYSIA*

[2] *Faculty of Applied Sciences and Technology, Universiti Tun Hussein Onn Malaysia, Pagoh, Johor, MALAYSIA*

[3] *Abu Dhabi University, UNITED ARABIA EMIRATES*

*Corresponding Author: afiqahrahim16@gmail.com
DOI: https://doi.org/10.30880/jscdm.2026.07.01.003

Article Info

Received: 4 September 2025
Accepted: 8 February 2025
Available online: 10 April 2026

Keywords

Singular spectrum analysis artificial neural network, long short-term memory, forecasting

Abstract

The ability to accurately predict impacts of rainfall is essential for managing our water supply, developing agricultural plans and protecting ourselves from natural disasters. For the purpose of this study, this research compares statistical and neural network-based methods used in the prediction of rainfall, particularly Singular Spectrum Analysis (SSA), Artificial Neural Network (ANN) and Long Short-Term Memory (LSTM). According to those models, time series can be decomposed into trend, seasonal and noise components during SSA approaches. Meanwhile, other deep learning models offer solutions that utilize historical data to model nonlinear rainfall behavior and account for random events. Data collected from Mersing on daily rainfall are analysed for statistical properties throughout this study and used to develop and test forecasting models. In this regard, metrics can be analysed to determine the performance of models such as Mean Absolute Error (MAE), Root Mean Square Error (RMSE), Coefficient of correlation (R) and Nash-Sutcliffe Efficiency (NSE). The results indicate that SSA performed the best reconstruction with MAE=4.1931, RMSE=6.8078, R=0.9362, and NSE=0.8727, however both recurrent and vector SSA had poor forecasting scenarios with negative NSE values. Based on training results, LSTM model showed MAE=4.0391, RMSE=10.5818, R=0.7495 and NSE=0.5307 which indicates that the model has a strong capacity to learn. Although the ANN model outperformed the LSTM model marginally in the test, it achieved a MAE of 9.9117 and NSE of -0.2332 which gives it a marginal edge over the LSTM model. It is suggested that rather than concentrating solely on statistical or neural models, both structured and chaotic rainfall behaviours in Malaysian coastal areas can be better understood by integrating SSA with deep learning techniques for a more effective understanding of the phenomenon. A combined approach can provide valuable insights into the implementation of decomposition-based and data-driven forecasting methods specifically for moist climates. Thus, the findings established the foundation for further studies on developing improved and enhance hybrid rainfall forecasting systems.

1. Introduction

Management of water supply resources, planning and implementing crop plans and disaster preparedness all depend on accurate rainfall predictions to succeed. In hot and humid environments, rainfall occurs differently depending on several triggering factors such as monsoon seasons, the monsoon interval and localised thunderstorms. Annually [1], during Northeast Monsoon season (from November to March), there have been severe issues arising from excessive rain and heavy downpours in Malaysia caused damaging infrastructure and disruptions to regular daily operations [2]. In some scenarios, extreme weather events fuelled by socio-economics activities threaten ecosystems and human safety, making accurate rainfall forecasting increasingly important. Consequently, this type of forecast is useful for long-term planning of water resources while allowing for early preparedness and response to rainfall-induced disasters.

Meteorological variables interact to influence atmospheric dynamics, making rainfall predictions challenging. Under real-world conditions, rainfall trends are irregular, non-stationary and chaotic [3]. Three of those traditional models, Autoregressive Models (AR), Autoregressive Integrated Moving Average (ARIMA) and Vector Autoregression (VAR) are less accurate and not suitable for rainfall under extremely variable conditions based on their assumptions of stationarity, linearity and normally distributed residuals. Thus, these models are incapable of predicting rainfall in volatile or extreme weather conditions. In the light of this limitation, more advanced time-series modelling approaches are needed they distinguish structurally meaningful patterns from random variations.

SSA is a statistical tool widely used to forecast time series data. Recent works have shown that its role is still applicable and beneficial for predicting tourism activity [5], monitoring water usage and management [6], energy conservation [7], and weather forecasting tools [8]. In addition, this method is also known as nonparametric decomposition and is applied in time series analysis to extract patterns, cycle and irregular components regardless of the characteristics of the series [9]. The study identified a technique capable of enhancing predictions and separating meaningful components from noise at the same time.

However, there are some drawbacks associated with SSA as well. For example, unstructured datasets might struggle to handle abrupt changes and peaks when embedded [10], despite its capacity to smooth over time series and incorporate dominant structures. Some of the issues associated with this approach are particularly apparent in climate data such as daily rainfall which tends to have large peaks. This data also has heavy tails and nonlinear dynamics. Due to these features, modelling approaches that rely on decomposition come under suspicion and highlight the necessity for alternative or complementary modelling approaches.

To address some of these shortcomings, several SSA advances are considered. There are two types of predictive techniques, Vector SSA (VSSA) and Recurrent SSA (RSSA). Research conducted previously found that VSSA approaches were superior to RSSA approaches when dealing with time series with unit roots [11]. VSSA focuses on capturing correlations within variables to identify similarities and interacting variables not apparent in univariate statistical analyses. By taking advantage of these developments, the model has expanded its robustness and adapted effectively to real-world scenarios emphasizing the precision of forecasts and at the same time, dealing with a variety of interrelated variables.

Artificial Intelligence (AI) has evolved significantly over the past few years and these advances have led to a breakthrough in time series forecasting techniques by analysing data. As an alternative to traditional methods, the use of machine learning and deep learning techniques have emerged and improved forecasting performance compared to traditional methods [12]. In combination with analytical tools, these models are able to identify complex relationships within different industry sectors [12], [13] including commerce, industrial production, the medical field and logistics. Unlike the conventional statistical methods, ANN and Recurrent Neural Network (RNN) models are not constrained by assumptions on the distribution of data, linearity or stationarity [14]. As with conventional statistical algorithms, the models cannot reveal underlying non-linearities in large and multidimensional datasets [15]. These nonlinear models work efficiently because they learn directly from the data given.

ANN is one of the most significant data-driven models because it is based on similar functionality to human brain. This method is widely used for rainfall prediction as it shows the relationship between the nonlinearity of historical data and predicted rainfall. Several studies have demonstrated the effectiveness of ANN. In [16], using ANNs to model rainfall variability showed that it was robust enough in predicting rainfall changes under tropical conditions and performing better than conventional methods. Meanwhile it is reported [17] that ANN is more reliable than autoregressive methods for future rainfall especially in the presence of noisy information within the long-term forecasting process in high-variable environments. Even so, hyperparameters including hidden layer size, learning rate and activation function have considerable influence on ANN. Therefore, ANNs suffer from overfitting problems such as oversaturation when the dataset is incomplete or too small [18]. While the model learns nonlinear mappings with strong impact, the realization that the model only considers long-term temporal relationships when the data is non-linear has not been achieved and so the development of deep learning architectures to manage this problem that address long-term relationships is being studied.

RNN provides more than disappearing gradients by capturing broad time-dependent relationships with LSTM which is a special type of RNN. In large datasets, LSTMs have the potential to reveal hidden patterns which is particularly useful when modelling complex, inconsistent time series such as rainfall. It is being used for rainfall prediction increasingly because of its capacity to understand the relationship between both long and short-term periods. This is done by cells that store memories with gates connected. LSTM has been proven to be more accurate than ANN in predicting rainfall variability in monsoon zones [19]. Hence, the model achieved reduced forecasting errors compared to prediction based on normal regression and autoregression [20]. The results underline LSTM's superiority when dealing with sequence datasets with temporal fluctuations. However, LSTM models typically require more data and computing resources to reach their maximal capabilities [21], which contributes excessive fitting when sufficient data is not available.

Due to their capabilities and limitations, a comparison between decomposition techniques along with learning-based methods is required to measure the performance of these methods for predicting rainfall. This study focuses on three main topics: the aim to establish rainfall prediction models using three distinct methods, called SSA, ANN and LSTM methods; the challenge of comparing the prediction effectiveness of each method for rainfall prediction; and the pros and cons of predicting rainfall based on statistical analysis and neural networks.

On the east coast of Johor, Mersing is the coastal town that the subject of this study. First, a statistical examination of Mersing's daily rainfall data is conducted. These properties include stationarity, linearity, normality and trend. Based on these insights, models are constructed, trained and tested. Predictability is evaluated by benchmark indicators such as MAE, RMSE, R and NSE.

The comparison between statistics and machine learning for predicting rainfall was lengthy and complex process. If rainfall is statistically significant, then a comparative analysis would offer more insight into the phenomenon. It can be found in regions with a wide range of climates. Moreover, the results of this study are relevant to identifying appropriate methods of modelling for enhancing alert disaster detection, optimizing livestock operations, minimizing flood risks and increasing resilience to global climate change in coastal areas. By bridging the differences between both approaches, we contribute to insights about time series predictions.

2. Methodology

The methodology involves collecting data, applying statistical calculations to determine its characteristics and then preprocessing it for modelling. The following steps are taken, these techniques are developed: SSA, ANN and LSTM and measurements are made regarding MAE, RMSE, R and NSE.

2.1 Data Collection

In this report, the data was gathered from the beginning of January 2018 to the end of December 2021 according to the Malaysian Meteorological Department on the rainfall collected daily in Mersing, Johor. There are several variables contained in the time series that include temperatures at high and low levels, humidity, pressure on the earth's surface and amount of rainfall. It should be noted that only rainfall data used in this study as a single-variate time series. Figure 1 illustrates the coastal location and monsoon expose of a particular area due to its vulnerability due to heavy rains.

Fig. 1 *The location of Mersing, Malaysia is on the east coast*

The following hydrograph shows rainfall per day from 2018 to 2021. Although rainfall was generally low or absent, there were some extreme events that exceeded 100 millimetres per day and reached a maximum of 300 millimetres per day. The monsoon season is marked by irregular rains that fall in clusters between drought periods.

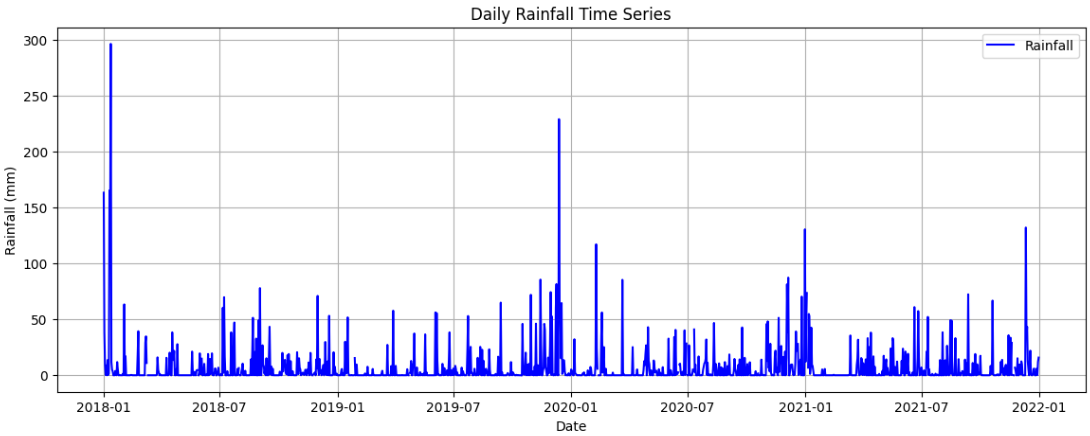

Fig. 2 *Rain amounts (mm) between year 2018-2021*

2.2 Statistics- based Analyses

To assess the adequacy of rainfall data, the study ran a few tests including tests for stationarity, linearity, normality and pattern analysis to verify rainfall data.

2.2.1 Stationary Test

Stationarity defines data series that remain constant over time statistically. For the application of statistical and machine learning forecasting models, this requirement is essential. As an initial assessment of stationarity, the Augmented Dickey-Fuller (ADF) and the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) tests were performed [22]. According to the ADF test, it consists of a unit root and indicates that the time series is non-stationary but for KPSS test, it indicates a stationary test. Consequently, we propose to reject H_0 for the results of KPSS test and accept H_0 the ADF test.

2.2.2 Linearity Test

Linearity testing evaluates simple linear models to decide whether they are appropriate or more complex nonlinear techniques are required. It is determined whether a model has been specified correctly and is purely linear by testing it using the Ramsey Regression Equation Specification (RESET) [23]. The null hypothesis (H_0) is rejected if the *p-value* is less than or equal to 0.05 indicating that the model may have been miss specified or fails to capture nonlinear relationships. This if *p-value* is greater than or equal to 0.05, we fails to reject the null hypothesis (H_0) based on the linear functional form.

2.2.3 Normality Test

It is necessary to conduct a normality test to verify that the data distribution is appropriate [24], which is a basic requirement for parametric forecasting. The data is not normally distributed if the *p-value* is lower than or equal to significance and the null hypothesis (H_0) is rejected. However if the *p-value* is higher than or equal to the significance level, we fail to reject the null hypothesis (H_0) since there is limited evidence supporting that the data is non-normal. In order to obtain these outcomes, we should consent to the Anderson-Darling test.

2.2.4 Trend Test

The present of upward and downward movements of rainfall time series can be evaluated by using Man-Kendall test [25]. Trending can inform systematic changes over time that help predict the forecasting model's accuracy. Hence, the results show that the null hypothesis (H_0) is rejected when *p-value* $\leq$ significance implies a monotone trend in the data. In contrast, we cannot reject the null hypothesis(H_0), if the *p-value* $\geq$ significance level. As a result, it can conclude that there is no trend.

2.3 Pre-processing the Raw Data

Cleaning and screening the datasets were part of the preprocessing step. First, rainfall data was applied, and the omitted values were transformed into numerical values applied through linear interpolation. The dates were also entered as indexes. The datasets were divided into two parts, 80% devoted to training and 20% to testing.

By using this percentage, it was verified that seasonal variations in rainfall were apparent in both sets. By doing so, the model will gather insights into recurring seasonal patterns and be tested under unobserved parameters. Then, the analysis of both training and testing results will provide information to identify potential fitting problem behaviours such as overfitting and underfitting. Considering the importance of maintaining temporal and sequential order and avoiding inappropriate data disclosure, the data split in chronological order was maintained. Initially, the values were converted to a standard scale with StandardScaler. This function transforms the value of X according to the formula.

$$X_{scaled} = \frac{x - \mu}{\sigma} \quad (1)$$

Where μ is the mean and σ is the standard deviation for training data. Fourteen-day lag features were designed to predict the following day based on the previous fourteen rainfall values. Since SSA is a straightforward decomposition approach and requires no additional procedure. When the entire dataset is embedded simultaneously, the algorithm able to extract characteristics such as trends, cyclicity and randomness.

2.4 Singular Spectrum Analysis

For time series decomposition and forecasting, SSA was applied as a purely statistical approach. This model consists of both decomposition and reconstruction and known as two-stage development process [26]. In the reconstruction stage, grouping and diagonal averaging are applied while embedding and Singular Value Decomposition (SVD) are used in decomposition stage. Reconstruction is achieved through grouping and diagonal averaging, while embedding and singular value decomposition (SVD) are used to decompose. Initially, the original series X was embedded into trajectory matrix of dimensions $X = (x_1, x_2, \dots, x_N)$ into a a matrix with dimensions $L \times K$ where $K = N - L + 1$. These matrices must be sorted in reversed order. The L-trajectory matrix of the series X is,

$$X = \begin{bmatrix} x_1 & x_2 & \cdots & x_K \\ x_2 & x_3 & \cdots & x_{K+1} \\ \vdots & \vdots & \ddots & \vdots \\ x_L & x_{L+1} & \cdots & x_N \end{bmatrix} \quad (2)$$

The trajectory matrix X was decomposed by using eigen-decomposition of lag-covarience matrix, $S = \boldsymbol{XX}^T$, where $d = rank(\boldsymbol{X}) \leq \min(L, K)$. This implies that the rank of matrix is restricted by the smaller dimension , $\sqrt{\lambda_i}= \sigma_i$ is singular values while λ_i is eigenvalues of $\boldsymbol{XX}^T$, right singular vectors (dimension K) is $V_i = \frac{\boldsymbol{X}^T U_i}{\sqrt{\lambda_i}}$ and U_i referring to left singular vector (dimension L).

$$\boldsymbol{X} = \sum_{i=1}^{d} \sqrt{\lambda_i}\, U_i V_i^T \quad (3)$$

Therefore, it is possible to formulate SVD of trajectory matrix X as equation below where $\boldsymbol{X}_i = \sqrt{\lambda_i} U_i V_i^T$. As an alternative, collection of ($\sqrt{\lambda_i} U_i V_i$) referred to eigentriples.

$$\boldsymbol{X} = \boldsymbol{X}_1 + \cdots + \boldsymbol{X}_d \quad (4)$$

During this stage, we use eigentriple grouping to perform in reconstruction stage. Each $\boldsymbol{X}_i$ corresponds to one component but one $\boldsymbol{X}_i$. There is no direct grouping of data by trend, seasonality, or noise. By using the weight correlation matrix, there is a strong correlation between elements with weak correlation where a strong correlation will indicate components in grouped. So, the components were grouped together and where I is set of indices, as follows,

$$\boldsymbol{X}_I = \sum_{i \in I} \boldsymbol{X}_i \quad (5)$$

Then, the matrices of grouped trajectory need to be transformed into a reconstructed series of length N, named diagonal averaging step. Consequently, it will generates a reconstructed series, , $\widetilde{\mathbb{X}}^{(k)} = (\tilde{x}_1^{(k)}, \dots, \tilde{x}_N^{(k)})$. Let Y with elements y_{ij} and L x K matrix, the position of I must be in between 1 and L while j must be in between 1 and K and $L \leq K$. Through diagonal averaging, we transfer the matrix **Y** into series $y_1, \dots, y_N$ by applying the given formula,

$$\tilde{y}_{ij} = \sum_{(l,k)\in A_S} ylk/|A_S| \tag{6}$$

where $A_s = \{(l,k): l + k = s, 1 \leq l \leq L, 1 \leq k \leq K\}$ and $|A_s|$ denotes the number of elements in the set A_s. In other words, the matrix elements are averaged over the "antidiagonals". After decomposing the initial series $x_1, \dots, x_N$ into m reconstructed series where n= 1,2,..., N as follows,

$$x_n = \sum_{k=1}^{m} \tilde{x}_n^{(k)} \tag{7}$$

As a result of elementary grouping, the reconstructed series will be elementary in nature.

2.5 Artificial Neural Network (ANN)

ANN mimic the essential functions of neuron performs by replicating the basic functions of living neurons. As they are endowed with a large computational power, AI algorithms imitate how these biological neurons function, making them suitable for solving nearly every kind of problem they could dream up. It is clear that the phenomenal computational ability characteristic of ANN algorithms is the foundation for their intelligent performance.

Input layers, hidden layers and output layers are three basic layers of neural networks [27]. In the input layer, the nodes represent independent variables; in the output layer, the nodes represent dependent variables. Each node carries things, and its synaptic weight indicates its role within these variables. Each node makes its input information, while other nodes process information from it. Nonlinear relationships are formed between explanatory variables, so their interrelationships can be described as follows,

$$y(t) = g\left(\sum_{i=1}^{n} v_j \left(f\left(\sum_{k=1}^{L} W_{ji}\, X(t-k) + b_j\right)\right) + B\right) \tag{8}$$

where the predicted output at time step t, $y(t)$, input variables reflect past rainfall lagged over time, $X(t-k)$ and k = 1, 2, . . ., L , L is the number of input lags, W_{ji} is weight parameter that connecting k-th lagged input $X(t-k)$ to the j-th hidden neuron while w_j is the weight parameter that connecting both output neuron and hidden neuron j , bias term for the j-th hidden neuron is b_j while bias term for output neuron is B, $f(\cdot)$ is the activation function of hidden layer while g($\cdot$) is the activation function of output layer and lastly m is the amount of hidden neurons in hidden layer.

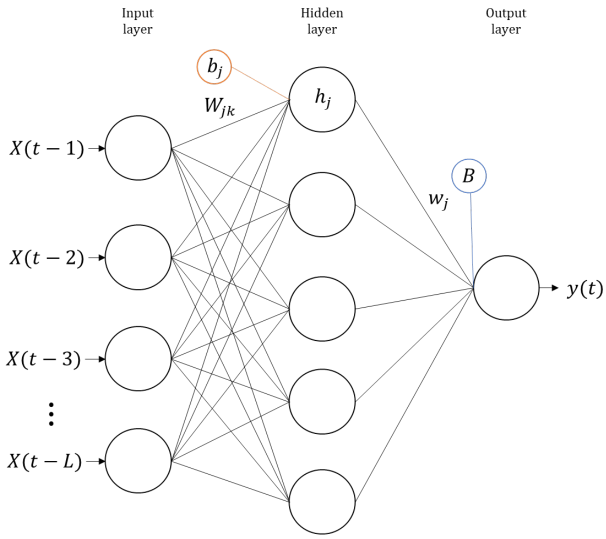

Fig. 3 *Overview of ANN model*

J. of Soft Computing and Data Mining Vol. 7 No. 1 (2026) p. 22-36

Based on the calculation below, these are backward rainfall inputs calculated by hidden neurons, h_j. The result is passed through an activation function, f after multiplying with the weights, W_{ji} add the bias, b_j which generates the output of the hidden neurons.

$$h_j = f\sum_{k=1}^{L} W_{jk}X(t-k) + b_j \tag{9}$$

2.6 Long Short-Term Memory (LSTM)

LSTM is an improved version of Recurrent Neural Network (RNN) developed to overcome its limitations such as diminishing gradients during training or gradient explosions. LSTM is composed of gate mechanisms and memory cells that manage data flow over time. Input gates, forget gates and output gates determine which information need to be stored, updated or discarded within the system in LSTM. Hence, it is possible to predict complex temporal patterns using LSTM network in time series forecasting. Furthermore, it is important to emphasize that the accuracy of prediction relies on the quality and availability of the input data. Using this method to analyse data with single variable can lead to poor results.

Hence, the LSTM model incorporates several levels and gates mechanisms to deal with long-term dependence on rainfall data [28]. Figure 4 illustrates the structure which includes a forget gate (f_t), input gate (i_t) and output gate (o_t) as well as a candidate cell state $(\tilde{C})$. In order to optimize the efficiency of the model, these gates work together to selectively retain, update the relevant information at subsequent time steps.

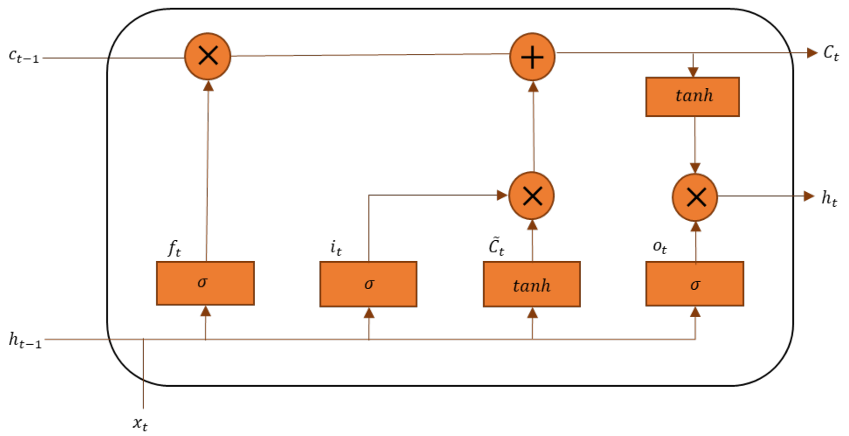

Fig. 4 *An Overview of LSTM model*

Each time step represents the amount of rainfall on a given day using the LSTM model to analyse sequential rainfall data. The hidden state and cell state are updated with gates at regular intervals. This is where x_t is the input at time t, h_t is the hidden state, C_t is the cell state, W_f, W_i, W_C, W_o are weights matrices, b_f, b_i, b_C, b_o are biases, σ is the sigmoid activation function and $\odot$ is the element-wise multiplication. The forget gate, f_t decides which information from the previous cell state, C_{t-1} should be discarded. The calculations are given below.

$$f_t = \sigma\left(W_f \cdot [h_{t-1}, x_t] + b_f\right) \tag{10}$$

The input gate, i_t is responsible for selecting which data changes should incorporated into the current cell state. The following calculation is based on,

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{11}$$

Based on the current input and the previous hidden state, the candidate cell state, $\tilde{C}_t$ returns new candidate values to update the cell state.

$$\tilde{C}_t = tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{12}$$

Next, a new cell state, C_t obtained by summarizing the previous and candidate cell states and multiplying them with the forget and input gates. Accordingly, there is the following calculation,

$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t \tag{13}$$

It is determined by the output gate, o_t which part of the cell state must be in to calculate the hidden state.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{14}$$

To calculate the hidden state, the output gate is applied to the updated cell state. As shown below, the calculation for the result.

$$h_t = o_t \odot \tanh(C_t) \tag{15}$$

2.7 Structure and Methodological Approaches

The methodology is depicted in the following diagram (See Figure 5). During this process, four steps are performed including data preparation process, feature engineering, model design and analysis. The steps in this process are visually arranged in a flowchart.

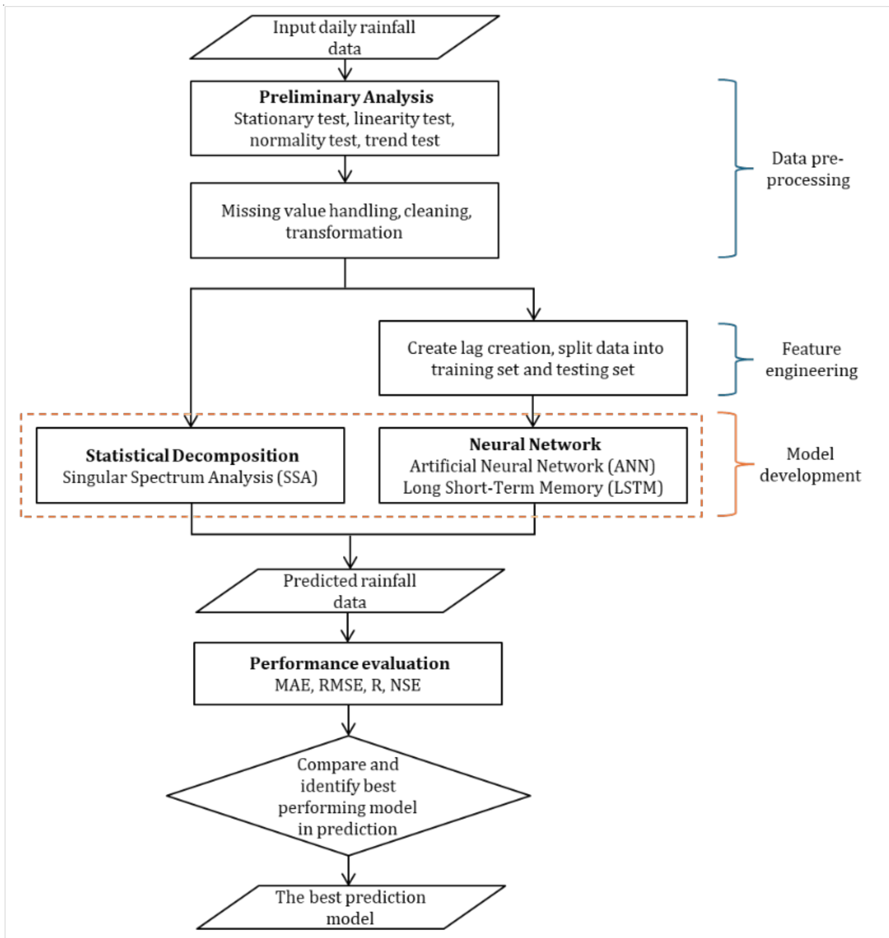

Fig. 5 *Research framework and methodology summary*

The rainfall time series was decomposed into its main components and the signal was reconstructed through SSA. The table summarizes the hyperparameters used for SSA including window length, L and the number of eigen components, r.

Table 1 *Configuration of SSA model*

Parameter	Value SSA
L	15
r (grouping index)	10

The network learns temporal dependencies by using rainfall records and predicting rainfall amounts. Table 2 shows that the ANN and LSTM parameters include the number of layers, neurons, activation function, dropout, batch size, optimizer, learning rate, loss function and iteration counts.

Table 2 *ANN and LSTM hyperparameters*

Parameter	Value	
	ANN	LSTM
Layer	1	2
Hidden Units / Neurons	32 units	50 units + 30 units
Activation function	tanh	tanh
Recurrent dropout	-	0.2
Batch size	32	16
Optimizer	Adam	
Learning rate	0.001	
Loss function	MSE	
Epochs	500	

2.8 Key Performance Evaluation

Based on the results of the analysis phase, the following four performance metrics were evaluated. These metrics were Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), coefficient of correlation (R) and Nash-Sutcliffe Efficiency (NSE) to determine prediction accuracy and error in this research. Measures of evaluation are described by Eq. (16), (17), (18) and (19) whereas n the number of observations, $\widehat{p_i}$ is the predicted output, $\bar{\hat{p}}$ is the mean of predicted output, a_i is the real output and $\bar{a}$ is the mean of real output.

2.8.1 Mean Absolute Error (MAE)

MAE is applied to estimation and data discernment from the raw signal. MAE is calculates by the following formula,

$$\boldsymbol{MAE} = \frac{1}{n}\sum_{i=1}^{n} |a_i - \hat{p}_i| \tag{16}$$

2.8.2 Root Mean Square Error (RMSE)

RMSE is a measure of residual standard deviation giving a broad view of errors in predicts. Accordingly, a lower RMSE means better performance. RMSE can be calculated from the following formula,

$$\boldsymbol{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (a_i - \hat{p}_i)^2} \tag{17}$$

2.8.3 Pearson Correlation Coefficient (R)

On a linear scale, R expresses how similarly observed the outputs are to expected ones. A higher correlation coefficient appears when projected versus actual outputs appear closer. This coefficient is desirable for the optimal result. R calculation could be done as follows,

$$\boldsymbol{R} = \frac{\sum_{i=1}^{n}(a_i - \bar{a})(\hat{p}_i - \bar{\hat{p}})}{\sqrt{\sum_{i=1}^{n}(a_i - \bar{a})^2} \times \sqrt{\sum_{i=1}^{n}(\hat{p}_i - \bar{\hat{p}})^2}} \tag{18}$$

2.8.4 Nash-Sutcliffe Efficiency (NSE)

Accordingly to its use in hydrology, NSE is frequently applied to assess whether the time series modelled are accordance with the observed ones [22]. Apart from that, performance evaluations range from negative infinity to one. Generally, if the NSE is negative, it indicates that the model performs poorly compared to average predictions, while predicting that observed means is better. As the NSE value approaches zero, it indicates positive predictive potential. If NSE value approaches one, the model is more accurate. The following is a formula for NSE,

$$\boldsymbol{NSE} = \mathbf{1} - \frac{\sum_{i=1}^{n}(\boldsymbol{a_i} - \widehat{\boldsymbol{p}}_i)^2}{\sum_{i=1}^{n}(\boldsymbol{a_i} - \overline{\boldsymbol{a}})^2} \tag{19}$$

3. Findings and Recommendations

The statistical significance of the rainfall time series was calculated through multiple tests such as stationarity, linearity, normality and trend analysis. These calculations were performed to analyse rainfall data.

Table 3 *Statistical test results*

Test	ADF Test	KPSS Test	Ramsey RESET Test	Anderson Darling Test	Kolmogorov–Smirnov Test	Mann-Kendall Test
Statistic value	-20.8507	0.0971	5.8208	285.0271	0.3578	2.1146
p-value	0.01	0.1	0.0035	0.0000	0.0000	0.0345

Based on the tests listed above, stationary rainfall data are supported by both ADF and KPSS tests. ADF analysis resulted in a p-value of 0.01, which is less than the significance level of 0.05, thus rejecting the null hypothesis that the data contained unit roots (non-stationary). This was verified by KPSS analysis, the p-value was greater than 0.05, so the null hypothesis could not be rejected. As a result, this confirms the conclusion that time series data are stable and compatible with the majority of time series forecasting methods. By applying the Ramsey RESET test, it was found that the null hypothesis that the model was linear and appropriate had been rejected with a p-value of 0.0035. It is possible that there is a nonlinear relationship or omitted data variables in this study. This revealed that a simple linear model failed to identify rainfall patterns.

As shown in the diagram below, the SSA findings are illustrated. It works by separating the series into their elements at the Singular Value Decomposition (SVD) stage, which generates singular values, eigenvectors and associated elementary components. Figure 6 presents the amount of variance explained by each component which is significant for identifying the main patterns the in data. Accordingly, plotted eigenvectors are illustrated in Figure 7 which are frequently used to analyse patterns related to seasonality and cycle. Moreover, the weight correlation matrix shown in Figure 8 demonstrates a strong correlation between the reconstructed components.

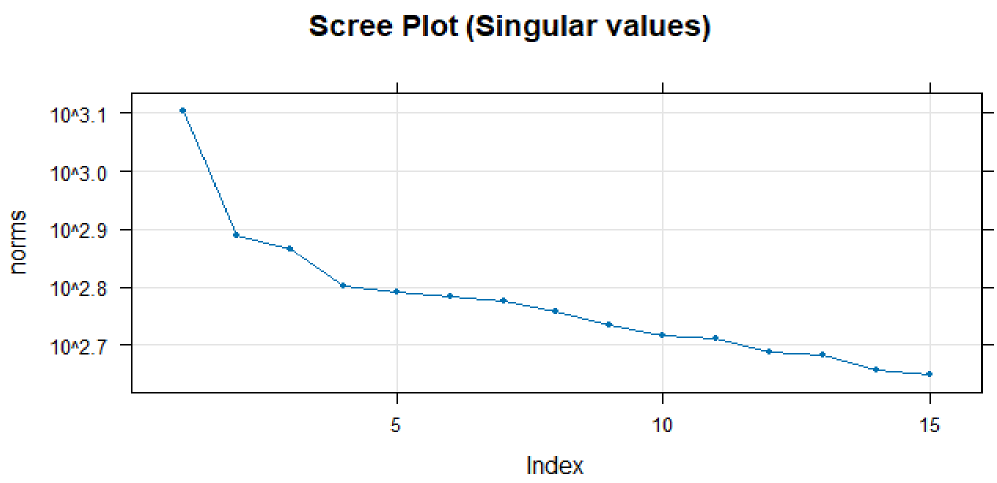

Fig. 6 *Graph of singular values*

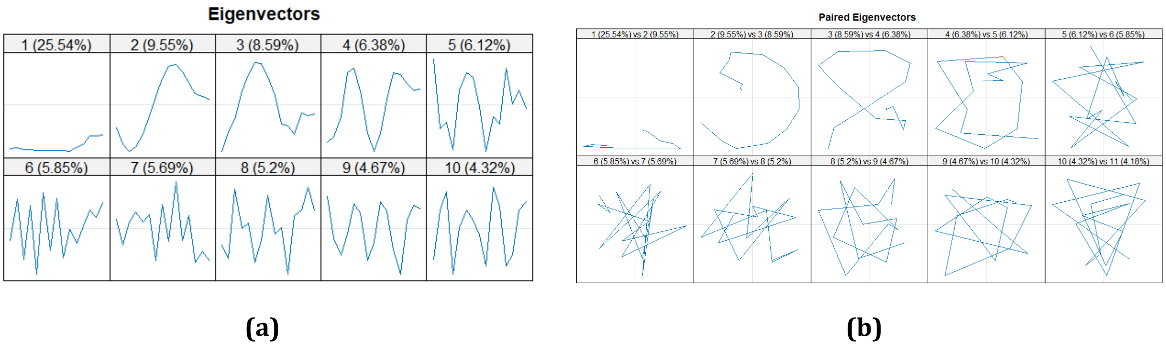

Fig. 7 *Diagram of SSA decomposition (a) Eigenvectors; (b) Pair of eigenvectors plotted*

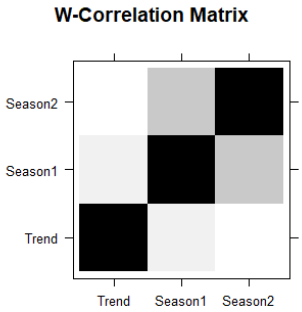

Fig. 8 *Matrix of weighted-correlations*

Figure 9 compares SSA output and actual rainfall for the day. In the training process, the reconstructed rainfall represents by the blue dashed line, indicating good match with the observation data. As part of the testing process, RSSA and VSSA forecasts are shown in red and orange lines respectively. Even though the SSA effectively reconstructs historical rainfall patterns, future rainfall amounts are forecast to be considerably lower than past rainfall amounts.

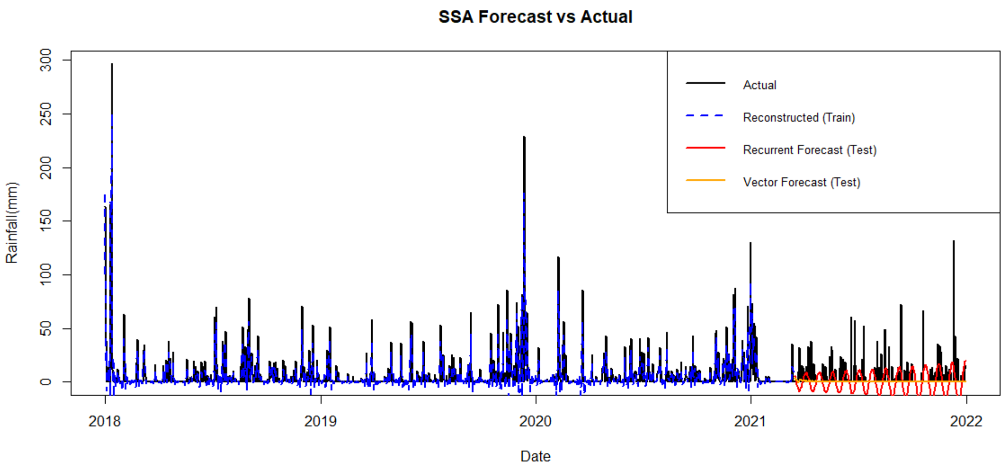

Fig. 9 *Real rainfall versus SSA forecast*

The following table illustrates the performance of the SSA model. SSA works fairly well when reconstructing the training dataset with small MAE (4.1931) and RMSE (6.8078), resulting in an excellent correlation of R=0.9362 between reconstructed and actual rainfall values. In addition, the NSE=0.8727 shows that the model performs well. According to the results, SSA provides excellent representation of the training data structure.

Although the SSA was extremely effective at reconstructing the data, it lacked significant predictive power and failed to perform well. Even though SSA is very effective in reconstructed data, the model performs poorly and has limited predictability. For the RSSA model, MAE rose to 12.2903 and RMSE to 17.42260 respectively, while the correlation declined to 0.07812. The NSE indicates the weak correlation (-0.5675) that shows that the forecast underperformed compared to merely predicting the average rainfall over the testing period. Furthermore, VSSA showed a slight improvement in MAE (6.4506) despite an increased RMSE (15.2406), the lowest negative correlation recorded (-0.0532) and a poor NSE (-0.1990). In the test set, neither the recurrent nor vector SSA approaches correctly captured rainfall patterns.

Table 4 *Testing and training phase of SSA forecast accuracy*

Model	SSA			
	MAE	RMSE	R	NSE
Training (Reconstructed)	4.1931	6.8078	0.9362	0.8727
Testing (RSSA)	12.2903	17.4260	0.07812	-0.5675
Testing (VSSA)	6.4506	15.2406	-0.0532	-0.1990

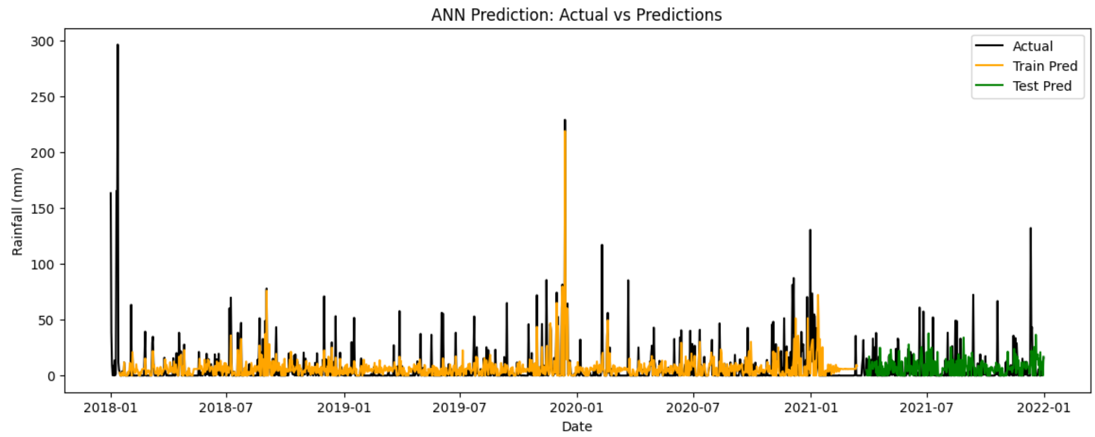

Fig. 10 *Comparison of actual and predicted rainfall using ANN*

Figure 11 below illustrates the actual rainfall values compared with LSTM predictions. The blue line in the training period plot indicates that the model has learned temporal dependencies on rainfall patterns. The model captures the overall trend when tested, as shown by a red line. Even so, the accuracy of the model decreases over time, particularly when it comes to predicting severe weather conditions. In spite of the fact that LSTM model provide performance when used with known data, their generalization data is less effective.

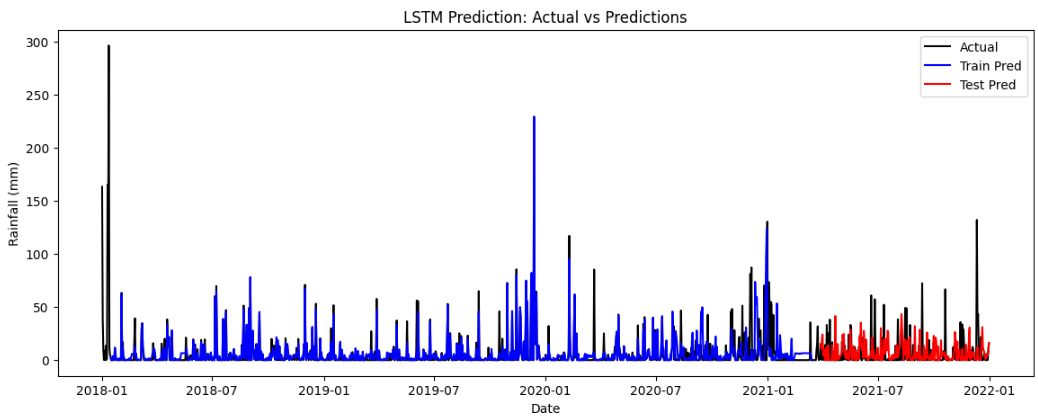

Fig. 11 *LSTM comparison of real and predicted rainfall*

Table 5 illustrates the statistical findings that confirm this conclusion and further demonstrates the trends in such outcomes. In this training phase, the LSTM model outperformed ANN with the minimal MAE (4.0391) and RMSE (10.5818) and the leading correlations (R= 0.7495) and NSE (0.5307). Therefore, LSTM has a better understanding of rainfall than ANN which leads to a clearer picture of allocated rainfall. Despite having very weak model accuracy (lower R and NSE scores), the ANN showed slightly higher MAE (9.9117 versus 9.3575) and NSE (-0.2332 versus -0.3535) during the testing phase.

In general the findings indicate that the LSTM technique performed more effectively on the task. The results can be interpreted as capturing the long-term relations of the rainfall pattern and learning the dependencies better than ANN. However, ANN performed less but had slightly better performance, although both models did not generalize very well under unpredictable weather scenarios. Both ANN and LSTM models can effectively model intricate rain data structures, but their ability to predict unexpected occurrences is restricted.

Table 5 *The performance of ANN and LSTM prediction models*

Model	Training				Testing			
	MAE	RMSE	R	NSE	MAE	RMSE	R	*NSE*
ANN	6.5834	12.1158	0.6241	0.3847	9.9117	15.6829	**0.0417***	**-0.2332***
LSTM	**4.0391***	**10.5818***	**0.7495***	**0.5307***	**9.3575***	**16.4302***	0.0220	-0.3535

* shows the ideal outcomes

From these observations, it is clear that the ANN and LSTM models are particularly efficient at rainfall prediction and can detect long-term patterns and short-term changes. Although the ANN has more accurate results than LSTM, it can handle sequential data and more appropriate method for hydrological conditions prediction. As a whole, the study suggests that deep learning methods (ANN and LSTM models) are more suitable than SSA methods for handling nonlinear and irregular rainfall distributions. Nevertheless, SSA still proves to be effective for filtering, decomposing and extracting trends from data, especially while analysing long-term trends and reducing noise. In this regard, it is an optimal method when preprocessing or postprocessing data. For providing real world forecasts and understanding rainfall behaviour complexity, the ANN and LSTM methods have shown high adaptability and efficiency.

Conclusions

This paper uses SSA, ANN and LSTM networks to predict rainfall forecasting models based on daily historical rainfall data. Statistics evaluation shows rainfall data are ideally suited to time series analysis because they are relatively stationary. In the process of reconstructing the dataset for training, SSA decomposition captures the underlying data with low error values and high efficiency. Unfortunately, both RSSA and VSSA performed poorly during the testing process. Based on this finding, SSA is better suited to decomposing, smoothing and identifying meaningful from complex and noisy rainfall time series.

In contrast, both deep learning models show increased prediction efficiency. The ANN model identified the pattern of rainfall over year changes, although the LSTM model showed better performance in learning temporal dependencies during the training phase. However, both ANN and LSTM approaches were unable to handle irregularity and excessive rainfall, leading to a loss of accuracy when tested. Despite these limitations, these approaches tend to provide better results than traditional models at modelling complex and nonlinear rainfall processes. Ultimately, ANN and LSTM approaches are more suitable for rainfall prediction while SSA methods contribute to the reduction of noise and extraction of trends as a complementary approach to improve the interpretability and accuracy of rainfall forecasting models.

Models based on SSA and deep learning (ANN and LSTM) can work in collaboration in the future to develop hybrid SSA-DL frameworks that exploit SSA's knowledge decomposition and nonlinear learning aspects through neural networks. Additionally, considering meteorological factors such as humidity, peak and minimum temperatures along with wind speed would make such analyses further and more accurate when this is being forecast and extreme rain is predicted particularly in tropical regions.

Acknowledgement

The authors would like to thank Universiti Tun Hussein Onn Malaysia for supporting this research under Ministry of Higher Education Malaysia (MOHE) Fundamental Research Grant Scheme (FRGS/1/2022/ICT06/UTHM/03/1).

The authors used Wordtune for grammar checking and language editing. In addition, ChatGPT was additionally used to support the preparation of the manuscript. All content generated was reviewed and verified by the authors, who take full responsibility for the final submission.

Conflict of Interest

The authors declare that there is no conflict of interest regarding the publication of the paper.

Author Contribution

The authors confirm contribution to the paper as follows: Nur Afiqah Abdul Rahim: ***Conceptualization, methodology, analysis and interpretation of results****; Shuhaida, I.:* ***Supervise, review and edit****; Azme bin Khamis;* ***Review and edit;*** *Usman Thariq:* ***Review and edit****. All authors reviewed the results and approved the final version of the manuscript.*

References

[1] Malaysian Meteorological Department,. (2024). *Official Website of Malaysian Meteorological Department*. Www.met.gov.my. https://www.met.gov.my/en/pendidikan/fenomena-cuaca/

[2] Ahmad, N., Peng Ng, C., Ismai, N., Ridwan Wong, M. M., Abbas, F., Hua Law, T., Zul Bahari, A. N. 'Azam, & Mohd Rofia, M. F. (2025). A Review of Major Flood Events in Malaysia Between 1970-2024. *Jurnal Kejuruteraan*, *37*(2), 909–921. https://doi.org/10.17576/jkukm-2025-37(2)-28

[3] Kabbilawsh, P., Kumar, D. S., & Chithra, N. R. (2022). Performance evaluation of univariate time-series techniques for forecasting monthly rainfall data. *Journal of Water and Climate Change*, *13*(12), 4151–4176. https://doi.org/10.2166/wcc.2022.107

[4] Rajab, K., Qian Long Kweh, & Ch. Aswani Kumar. (2022). *Forecasting COVID-19: Vector Autoregression-Based Model*. *47*(6), 6851–6860. https://doi.org/10.1007/s13369-021-06526-2

[5] Joanne, G., Disa Rahma Kirana, & Gumgum Darmawan. (2025). Forecasting the Number of ASEAN Tourists to Indonesia with the Hybrid Singular Spectrum Analysis-Support Vector Machine Method. *GeoJournal of Tourism and Geosites*, *59*(2), 769–777. https://doi.org/10.30892/gtg.59223-1455

[6] Nisa, E., None Adriyansyah, & Izma Fahria. (2025). Prediction of Water Availability in Kolong ST 12 Reservoir as a Raw Water Source using Singular Spectrum Analysis. *Bentang Jurnal Teoritis Dan Terapan Bidang Rekayasa Sipil*, *13*(2), 261–275. https://doi.org/10.33558/bentang.v13i2.10605

[7] Li, S., Duan, M., & Zhou, X. (2025). Dynamic Detection Method for Spatiotemporal Data Based on Hybrid Model and Singular Spectrum Analysis. *Informatica*, *49*(12). https://doi.org/10.31449/inf.v49i12.7578

[8] Wang, W., Ye, F., Wang, Y., & Gu, M. (2024). A singular spectrum analysis-enhanced BiTCN-selfattention model for runoff prediction. *Earth Science Informatics*, *18*(1). https://doi.org/10.1007/s12145-024-01524-y

[9] Arnobio Palacios Gutiérrez, Luis, J., & María Villeta López. (2023). Time series clustering using trend, seasonal and autoregressive components to identify maximum temperature patterns in the Iberian Peninsula. *Environmental and Ecological Statistics*, *30*(3), 421–442. https://doi.org/10.1007/s10651-023-00572-9

[10] Movahedifar, M., Hassani, H., Yarmohammadi, M., Kalantari, M., & Gupta, R. (2021). A robust approach for outlier imputation: Singular spectrum decomposition. *Communications in Statistics: Case Studies, Data Analysis and Applications*, *8*(2), 234–250. https://doi.org/10.1080/23737484.2021.2017810

[11] Ghodsi, M., Hassani, H., Rahmani, D., & Silva, E. S. (2017). Vector and recurrent singular spectrum analysis: which is better at forecasting? *Journal of Applied Statistics*, *45*(10), 1872–1899. https://doi.org/10.1080/02664763.2017.1401050

[12] Patil, D., Rane, N. L., Desai, P., & Rane, J. (2024). Machine learning and deep learning: Methods, techniques, applications, challenges, and future research opportunities. In *Trustworthy Artificial Intelligence in Industry and Society* (pp. 28–81). Deep Science Publishing. https://doi.org/10.70593/978-81-981367-4-9_2

[13] Razzaq, K., & Shah, M. (2025). Machine Learning and Deep Learning Paradigms: From Techniques to Practical Applications and Research Frontiers. *Computers*, *14*(3), 93–93. https://doi.org/10.3390/computers14030093

[14] Kim, T. Y., Oh, K. J., Kim, C., & Do, J. D. (2004). Artificial neural networks for non-stationary time series. *Neurocomputing*, *61*, 439–447. https://doi.org/10.1016/j.neucom.2004.04.002

[15] Sham, F. A. F., El-Shafie, A., Jaafar, W. Z. W., S, A., Sherif, M., & Ahmed, A. M. A. N. (2025). Advances in AI-Based Rainfall Forecasting: A Comprehensive Review of Past, Present, and Future Directions with Intelligent Data Fusion and Climate Change Models. *Results in Engineering*, 105774. https://doi.org/10.1016/j.rineng.2025.105774

[16] Chuasuk, P., Bhatrasataponkul, T., & Akkarapongtrakul, A. (2025). Comparative analysis and enhancing rainfall prediction models for monthly rainfall prediction in the Eastern Thailand. *MethodsX*, *14*, 103094. https://doi.org/10.1016/j.mex.2024.103094

[17] Ostojic, D., Lalousis, P. A., Donohoe, G., & Morris, D. W. (2024). The challenges of using machine learning models in psychiatric research and clinical practice. *European Neuropsychopharmacology*, *88*, 53–65. https://doi.org/10.1016/j.euroneuro.2024.08.005

[18] Wani, O. A., Mahdi, S. S., Md Yeasin, Kumar, S. S., Gagnon, A. S., Danish, F., Nadhir Al-Ansari, Salah El-Hendawy, & Mattar, M. A. (2024). Predicting rainfall using machine learning, deep learning, and time series models across an altitudinal gradient in the North-Western Himalayas. *Scientific Reports*, *14*(1). https://doi.org/10.1038/s41598-024-77687-x

[19] Rygh, T., Vaage, C., Westgaard, S., & Petter. (2025). Inflation Forecasting: LSTM Networks vs. Traditional Models for Accurate Predictions. *Journal of Risk and Financial Management*, *18*(7), 365–365. https://doi.org/10.3390/jrfm18070365

[20] Safonova, A., Ghazaryan, G., Stiller, S., Main-Knorn, M., Nendel, C., & Ryo, M. (2023). Ten deep learning techniques to address small data problems with remote sensing. *International Journal of Applied Earth Observation and Geoinformation*, *125*, 103569. https://doi.org/10.1016/j.jag.2023.103569

[21] Shaharudin, S. M., Ismail, S., Hassan, N. A., Tan, M. L., & Sulaiman, N. A. F. (2021). Short-Term Forecasting of Daily Confirmed COVID-19 Cases in Malaysia Using RF-SSA Model. *Frontiers in Public Health*, *9*.

[22] Perktold, J., Seabold, S., & Taylor, J. (2025, July 18). *Stationarity and detrending (ADF/KPSS) — statsmodels*. Www.statsmodels.org. https://www.statsmodels.org/dev/examples/notebooks/generated/stationarity_detrending_adf_kpss.html

[23] *Ramsey Regression Equation Specification Error Test Definition & Examples - Quickonomics*. (2024, September 8). Quickonomics. https://quickonomics.com/terms/ramsey-regression-equation-specification-error-test/

[24] Fabian Corrêa Cardoso, Rafael Alceste Berri, Lucca, G., Eduardo Nunes Borges, & Leite, V. (2023). Normality tests: a study of residuals obtained on time series tendency modeling. *Exacta*. https://doi.org/10.5585/2023.22928

[25] Duy, D. N., & Johannes Himmelbauer. (2025). A Two-Step Approach for Trend Detection in Injection Molding: Mann-Kendall Test and Decision Trees. *Procedia Computer Science*, *253*, 2003–2013. https://doi.org/10.1016/j.procs.2025.01.262

[26] Golyandina, N., & Zhigljavsky, A. (2013). Singular Spectrum Analysis for Time Series. In *SpringerBriefs in Statistics*. Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-34913-3

[27] Uzair, M., & Jamil, N. (2020, November 1). *Effects of Hidden Layers on the Efficiency of Neural networks*. IEEE Xplore. https://doi.org/10.1109/INMIC50486.2020.9318195

[28] Mo, J., Xiong, Q., Chen, Y., Liu, Y., Wu, X., Xiao, N., & Hou, W. (2025). Forecasting motion trajectories of elbow and knee joints during infant crawling based on long–short-term memory (LSTM) networks. *BioMedical Engineering OnLine*, *24*(1). https://doi.org/10.1186/s12938-025-01360-1