Alexandria Engineering Journal 115 (2025) 423–433

Contents lists available at ScienceDirect

Alexandria Engineering Journal

journal homepage: www.elsevier.com/locate/aej

Original article

Hybrid double ensemble empirical mode decomposition and K-Nearest Neighbors model with improved particle swarm optimization for water level forecasting

Vikneswari Someetheram [a], Muhammad Fadhil Marsani *,[a], Mohd Shareduwan Mohd Kasihmuddin [a], Siti Zulaikha Mohd Jamaludin [a], Mohd. Asyraf Mansor [b], Nur Ezlin Zamri [c]

[a] School of Mathematical Sciences, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia
[b] School of Distance Education, Universiti Sains Malaysia, 11800 USM, Penang, Malaysia
[c] Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

ARTICLE INFO

Keywords:
Artificial Intelligence
Machine Learning
Flood Forecasting
Predictive Analytics
Water Level

ABSTRACT

Water level forecasting plays a vital role in environmental protection and flood management because reliable predictions allow for the deployment of early warning systems to alert the public to minimize the impacts of flooding. This study presents an enhanced approach for weekly water level and flood prediction by integrating data decomposition techniques with machine learning models. Specifically, Ensemble Empirical Mode Decomposition (EEMD) was applied to disaggregate the original water level data into distinct Intrinsic Mode Functions (IMFs) to simplify complexity and enhance periodicity detection. A secondary decomposition was performed on the high-frequency IMF 1, derived from EEMD, to further refine the data features. The K-Nearest Neighbor (KNN) and Support Vector Machine (SVM) models, optimized using Improved Particle Swarm Optimization (PSO), were employed for forecasting. The effectiveness of these hybrid models was evaluated using various performance metrics, revealing that the DEEMD-KNN-PSO and DEEMD-SVM-PSO models significantly outperformed other single decomposition and standalone models. Among these, the DEEMD-KNN-PSO model demonstrated superior accuracy in predicting water levels, showcasing its potential for reliable flood prediction in the Klang River region of Sri Muda, Malaysia. This approach highlights the value of data decomposition and machine learning optimization for improving water level prediction accuracy.

1. Introduction

Water level forecasting plays a crucial role in environmental protection and flood management. Accurate predictions are essential during flood events, as they facilitate early warning systems to alert the public and enable real-time control of hydraulic structures, such as diversion channels and floodgates, to minimize flood damage. Effective flood management requires timely and reliable information about flood

Abbreviations: EMD, Empirical Mode Decomposition; EEMD, Ensemble Empirical Mode Decomposition; KNN, K-Nearest Neighbors; SVM, Support Vector Machine; PSO, Particle Swarm Optimization; DEEMD-KNN-PSO, Double Ensemble Empirical Mode Decomposition with K-Nearest Neighbors optimized by Particle Swarm Optimization; DEEMD-SVM-PSO, Double Ensemble Empirical Mode Decomposition with Support Vector Machine optimized by Particle Swarm Optimization; KNN-RF, K-Nearest Neighbor-Random Forest; LSTM, Long Short-Term Memory; LSTM-KNN, Long Short-Term Memory- K-Nearest Neighbor; SVM-ALO, Support Vector Machine optimized by Ant Lion Optimizer; IMF, Intrinsic Mode Function; EEMD-ARIMA, Ensemble Empirical Mode Decomposition with Autoregressive Integrated Moving Average; ELM, Extreme Learning Machine; DEEMD, Double Ensemble Empirical Mode Decomposition; SRM, Structural Risk Minimization; ERM, Empirical Risk Minimization; MAPE, Mean Absolute Percentage Error; SVM-PSO, Support Vector Machine optimized by Particle Swarm Optimization; KNN-PSO, K-Nearest Neighbor optimized by Particle Swarm Optimization; EEMD-SVM-PSO, Ensemble Empirical Mode Decomposition with Support Vector Machine optimized by Particle Swarm Optimization; EEMD-KNN-PSO, Ensemble Empirical Mode Decomposition with K-Nearest Neighbor optimized by Particle Swarm Optimization; RMSE, Root Mean Squared Error; MSE, Mean Squared Error; MAPE, Maen Absolute Percentage Error; DID, Department of Irrigation and Drainage Malaysia.

* Corresponding author.
E-mail addresses: viknessomeetheram@student.usm.my (V. Someetheram), fadhilmarsani@usm.my (M.F. Marsani), shareduwan@usm.my (M.S.M. Kasihmuddin), szulaikha.szmj@usm.my (S.Z.M. Jamaludin), asyrafman@usm.my (Mohd.A. Mansor), ezlinzamri@upm.edu.my (N.E. Zamri).

https://doi.org/10.1016/j.aej.2024.12.035
Received 2 September 2024; Received in revised form 18 November 2024; Accepted 9 December 2024
Available online 23 December 2024

progression, which can be challenging to provide, especially when only historical rainfall data up to the forecasting point is available, without any assumptions about future rainfall patterns [1–4]. This is particularly relevant in Malaysia, where understanding and predicting water levels is critical for safeguarding communities and infrastructure against frequent and often severe flooding events.

The modelling and prediction of water levels have seen extensive use of machine learning techniques over the past 20 years on a global scale. Nearest neighbor can also be applied with noisy samples [5]. K-Nearest Neighbor (KNN) has been widely applied across various fields, including weather forecasting, financial predictions, and flood management, for its effective and versatile forecasting capabilities [6–8]. The study by [9] demonstrates that the hybrid KNN with Random Forest model effectively predicts groundwater level fluctuations. The combination of these approaches allows the KNN-RF model to deliver reliable and interpretable results, making it a powerful tool for seasonal groundwater level forecasting in regions with data constraints. Recent researcher presented the integration of the KNN algorithm with the Long Short-Term Memory (LSTM) neural network significantly enhances the real-time forecasting capabilities of the model, particularly for flood prediction [10]. By using KNN to update LSTM model outputs based on recent observations, the LSTM-KNN hybrid provides more accurate and timely flood forecasts, making it a valuable tool for decision support systems. The study by [11] indicates that the KNN method performs comparably to RF in predicting seawater levels, with both models achieving high R^2 values for short-term forecasts. The results suggest that KNN is a reliable and straightforward approach for seawater level prediction, offering valuable accuracy for future planning and coastal management.

The foundation of Support Vector Machines (SVMs) was laid by [12]. SVMs guarantee solutions that are both distinct and global because they are derived from solving an optimization problem involving a convex quadratic function with constraints, as noted [13]. Researcher conducted a study that evaluates groundwater level predictions across six different sites within the Vizianagaram district of Andhra Pradesh [14]. The results highlight that Support Vector Machine (SVM) shows strong performance, especially when enhanced with hybrid approaches. A study by [15] demonstrates that SVM based machine learning models provide a viable solution for real-time water level prediction in urban rainwater pipe networks, addressing key issues of model accuracy and running speed. This approach not only enhances the predictive accuracy compared to traditional hydrodynamic models but also significantly improves computational speed, making it suitable for practical flood warning applications in complex urban environments. The study by [16] highlights that the hybrid SVM-ALO model, incorporating the ant lion optimizer, provides a highly accurate and reliable approach for predicting groundwater level fluctuations. Compared to other hybrid models, SVM-ALO consistently outperformed in terms of lower root mean-squared error (RMSE) demonstrating its robustness and suitability for groundwater level forecasting.

The Ensemble Empirical Mode Decomposition (EEMD) represents a significant advancement over the conventional Empirical Mode Decomposition (EMD), primarily addressing the challenge of mode mixing [17]. EEMD, introduced by Wu et al. (2009), has emerged as a highly effective approach for processing intricate time series data. Additionally, EEMD facilitates the decomposition of data into Intrinsic Mode Functions (IMFs), effectively reducing stochastic volatility and elevating prediction quality. Hybrid models of EEMD have demonstrated their effectiveness in predicting a wide range of nonlinear phenomena, including runoff [18], wind speed [19] wave height [20] and streamflow [21]. For instance, [18] introduced an EEMD-ARIMA model for the prediction of annual runoff time series. The study established the superiority of EEMD-ARIMA over ARIMA for annual runoff forecasting. More recently, a study by [22] proposed a model that incorporates Ensemble Empirical Mode Decomposition (EEMD) for forecasting tidal river water levels. This water level forecasting model exhibited remarkable accuracy and reliability in predicting water levels at the case study.

SVM relies on two key hyperparameters, and their proper configuration is critical to the model's performance. However, there is potential to improve the effectiveness of SVM in predicting time series data. Additionally, KNN is a straightforward and computationally efficient algorithm for forcasting. However, its performance can be hindered by the initial selection of the parameter K. Building on previous research, optimizing SVM and KNN parameters through swarm intelligence optimization techniques has proven beneficial for enhancing model performance in complex problems [23–25]. Particle Swarm Optimization (PSO) is proposed by [26]. PSO is a population-based intelligent optimization algorithm known for its strong global search capability, high efficiency, and rapid convergence. Applying such advanced optimization techniques to water level forecasting could significantly enhance prediction accuracy and reliability, aiding in effective flood management and environmental protection. The application of PSO to fine-tune Long Short-Term Memory (LSTM) networks significantly enhances their ability to forecast water levels by optimizing hyperparameters and capturing complex data sequences [27]. In recent study, PSO with Extreme Learning Machine (ELM) models significantly enhances the accuracy of flood forecasting by optimizing the number of units in the ELM to improve prediction performance [28]. This approach highlights the effectiveness of PSO in fine-tuning model parameters, leading to more reliable and precise flood predictions, which can greatly aid in timely flood response and mitigation efforts. Recent study reveals that PSO significantly enhances the performance of SVM models for short-term rainfall forecasting, particularly for 5-min forecasts [29]. This approach demonstrates PSO's effectiveness in refining forecasting models and underscores its role in achieving superior rainfall prediction accuracy compared to other methods. Despite its effectiveness in numerous applications, PSO has not been applied to decomposed components specifically related to water level forecasting.

The main objectives of the proposed work are given below.

i. To develop an improved Particle Swarm Optimization (PSO) framework for automatically tuning the parameters of the KNN algorithm and SVM algorithm to enhance its accuracy in forecasting water levels. The proposed approach aims to minimize the forecasting error by efficiently searching the parameter space, leveraging the global search capabilities of improved PSO.
ii. To investigate the effectiveness of an improved PSO with hybrid KNN model in addressing the nonlinearity and variability in water level data. By incorporating improved PSO for parameter optimization, the KNN model can dynamically adjust to the complex patterns in the data, leading to more accurate and reliable water level forecasts.
iii. To implement two-stage decomposition approach for improving the accuracy of hydrological time series forecasting. Specifically, the study aims to implement a two-step Ensemble Empirical Mode Decomposition (EEMD) process, where the first Intrinsic Mode Function (IMF 1), representing high-frequency components, undergoes an additional decomposition using EEMD that also known as Double EEMD (DEEMD). This approach seeks to capture and isolate high-frequency variations more effectively, thereby enhancing the predictive capabilities of forecasting models for complex hydrological phenomena.
iv. To conduct a comprehensive comparative analysis of the models with traditional water level forecasting models. This study aims to evaluate the accuracy, robustness, and computational efficiency of the DEEMD-KNN-PSO model, providing insights into its practical applicability and advantages over conventional methods.

2. Methodology

This section outlines the methodology used for developing the proposed models in this paper. It details the step-by-step processes implemented to predict the water levels of the Klang River. The methodology includes the application of specific decomposition methods like EEMD and DEEMD, the use of machine learning models such as Support Vector Machines (SVM) and K-Nearest Neighbor. Additionally, this section presents the parameter settings.

2.1. Double ensemble empirical mode decomposition

Empirical Mode Decomposition (EMD) is a robust method often utilized to examine complex, nonlinear data [30]. This approach works by breaking down the initial dataset into a series of distinct components, which generates multiple Intrinsic Mode Functions (IMFs). Unlike using pre-set kernels, these IMFs are derived directly from the data itself, acting as adaptive basis functions. For a function to be considered an IMF, it must meet two specific conditions [31]. First, the count of local maxima and minima should be the same or differ by just one. Second, the average of the upper and lower envelope values should be zero at any location in the data. These IMFs represent a range of frequency components, from the highest to the lowest. Consider $s(t)(t = 1, 2, \dots, l)$ as an example of a time series from the dataset. The EMD method proceeds through the following steps [32]:

a. Identify all the local extrema of the time series, which includes both the local maxima and minima.
b. Use a cubic spline interpolation to connect the identified local maxima and minima, creating an upper envelope ($e_{up}(t)$) from the maxima and a lower envelope ($e_{low}(t)$) from the minima.
c. Calculate the mean envelope by averaging the upper and lower envelopes using the following Eq. (1).

$$m(t) = \left[\frac{e_{up}(t) + e_{low}(t)}{2}\right] \tag{1}$$

d. Calculate the difference between the original time series and the mean envelope obtained in Step 3, using Eq. (2) provided below.

$$h(t) = s(t) - m(t) \tag{2}$$

e. Test if $h(t)$ meets the criteria for being an IMF. If it does, designate $s(t)$ as the first IMF and replace $s(t)$ with the residuals $r(t) = s(t) - h(t)$. If it does not, replace $s(t)$ with $h(t)$ and continue the process.
f. Repeat Steps 1–5 continuously. Stop the iteration only when the termination condition is satisfied. This implies that the EMD process will end when the residual becomes a monotonic function, at which point no further IMF extraction is possible.

Additionally, the shifting process of EMD will be stopped when residual became a monotonic function where IMFs extraction is no longer available. The final product of decomposition by EMD are a set of IMFs and residual from the original data as in Eq. (3).

$$s(t) = \sum_{i=1}^{n} c_i(t) + r_n(t) \tag{3}$$

where n is the number of IMFs, $r_n(t)$ are the final residuals representing a trend and acts as central tendency of the signal $s(t)(t = 1, 2, \dots, l)$ and $c(t)(t = 1, 2, \dots, l)$ represents the Intrinsic Mode Functions (IMFs), which exhibit periodicity and are nearly orthogonal to each other. Each IMF independently characterizes the local properties of the original signal when describing them. The frequency of each IMFs varies high to low.

The versatility of the EMD has been demonstrated across various applications for signal extraction from noisy and non-linear data [17]. However, a significant limitation of the EMD is the frequent occurrence of mode mixing that occurs when a single IMF contains signals of greatly different scales or when a signal of same scale appears in other IMFs [26]. To address this issue, [33] introduced the Ensemble Empirical Mode Decomposition (EEMD). EEMD mitigates mode mixing by incorporating a finite amount of Gaussian white noise into the data series before computing the overall average, effectively reducing mode mixing [33]. In summary, EEMD, an extension of the EMD method, aims to eliminate mode mixing by introducing white noise into the data prior to analysis [34]. The process of EEMD is briefly explained below.

a. Initialize the ensemble number, M, and the noise amplitude and let $m = 1$.
b. Introduce a white noise series $n_m(t)$ into the original dataset $s(t)$ and produce the below equation $s_m(t) = s(t) + n_m(t)$.
c. Perform the data decomposition on $s_m(t)$ and produce Intrinsic Mode Functions (IMFs) while considering the added white noise using EMD.
d. Repeat these two steps iteratively until the residual $r(t)$ either turns into a monotonic function or has no more than one local extremum, signaling that no additional IMFs can be derived. Importantly, for each iteration, utilize $m = m+1$ white noise series if $m < M$ is satisfied. M is the maximum iteration. If not, proceed directly to Step 5.
e. Compute the ensemble mean, y_n of the corresponding IMFs from all decompositions to obtain the final IMFs and residual.

$$y_n = \frac{1}{M} \sum_{m=1}^{M} IMF_{n,m} \tag{4}$$

Double Ensemble Empirical Mode Decomposition (DEEMD) is a new method that is applied specifically to the first Intrinsic Mode Function (IMF) obtained from the initial decomposition process. This approach focuses on the first IMF because it typically contains the highest frequency components of the original signal. By targeting the first IMF, DEEMD aims to effectively capture and analyze these high-frequency features which are often crucial for understanding the underlying characteristics of the signal, such as noise and rapid fluctuations.

2.2. Support vector machine (SVM)

Vapnik originally proposed Support Vector Machines (SVM) in 1995 as a method for addressing both regression and classification tasks [12]. This increased interest in SVM can be attributed to its robust mathematical foundation, rooted in the principles of Structural Risk Minimization (SRM) and Empirical Risk Minimization (ERM). In this SVM model, let the training sets to be $S = \{(x_i, y_i) | i = 1, 2, 3, ., N\}$, $x_i = R^n$, $y_i = R$. The optimal decision function will then be located within the high-dimensional feature space. The decision function utilized in this study is represented by Eq. (5).

$$f(x) = \langle \omega, \varphi(x) \rangle + b \tag{5}$$

Where $\varphi(x)$ signifies the high-dimensional feature space that derived a nonlinear mapping from input space, ω is weight, b is bias. The parameters ω and b in the Eq. (5) is derived from solving the constrained minimization problem originally introduced by Vapnik (1995) shown in Eq. (6) and Eq. (7).

$$minimize \frac{1}{2}\|\omega\|^2 + C \sum_{i=1}^{N} \left(\xi_i^- + \xi_i^+\right) \tag{6}$$

$$\begin{cases} y_i - \langle \omega, \varphi(x) \rangle - b \le \varepsilon + \xi_i^- \\ \langle \omega, \varphi(x) \rangle + b - y_i \le \varepsilon + \xi_i^- \\ \xi_i^-, \xi_i^+ \ge 0, \end{cases} \tag{7}$$

In the above expressions, the constant $C > 0$serves as a parameter regulating the penalty level for instances surpassing the error threshold and ε represents the error tolerance. Furthermore, ξ_i^+ and ξ_i^- are positive variables, where ξ_i^+ denotes the upper excess deviation and ξ_i^- signifies the lower excess deviation. Incorporating Lagrange multipliers, the problem expressed in Eq. (6) is subsequently converted into a dual space in Eq. (8) and Eq. (9) below:

$$W(\alpha_i - \alpha_i^*) = -\frac{1}{2}\sum_{i=1}^{n}\sum_{i=1}^{n}\left[(\alpha_i - \alpha_i^*)\left(\alpha_j - \alpha_j^*\right)\langle\varphi(x_i), \varphi(x_j)\rangle\right] - \varepsilon\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) + \sum_{i=1}^{n}(\alpha_i - \alpha_i^*) \tag{8}$$

such that

$$\varepsilon\sum_{i=1}^{n}(\alpha_i - \alpha_i^*) = 0; \quad \alpha_i - \alpha_i^* \in [0, C] \tag{9}$$

Where α_i *and* α_i^* are Lagrange multipliers. Lagrange multipliers, subject to the imposed constraints, must adhere to the conditions. The resultant solution is presented in Eq. (10):

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)\langle\varphi(x_i), \varphi(x_j)\rangle + b \tag{10}$$

The inner product $\langle\varphi(x_i), \varphi(x_j)\rangle$ can be defined through is the kernel function $K_{svm}(x_i,x_j)$. Therefore, the equation can be define as in Eq. (11).

$$f(x) = \sum_{i=1}^{n}(\alpha_i - \alpha_i^*)K(x_i, x_j) + b \tag{11}$$

Eq. (12) presented the kernel function chosen in this paper for SVM model which is the radial basis function (RBF).

$$K_{svm}(x_i, x_j) = \exp\left[\frac{-||x_i - x_j||^2}{2\sigma_{svm}^2}\right] \tag{12}$$

where σ_{svm} is the width of the kernel function [35,36]. Hence, the cost parameter $C$and kernel parameter σ_{svm} of SVM need to be optimized.

2.3. K-Nearest Neighbors

The K-Nearest Neighbors (KNN) algorithm is a straightforward but powerful technique applied to both classification and regression problems that introduced by [37]. KNN operates on the principle that data points that are close to each other in a feature space tend to share similar properties. It predicts the output for an unknown data point by considering the majority class among its k-nearest neighbors from the training set, determined using a specific distance metric such as Euclidean, Minkowski, Chebyshev, or Manhattan. As a nonparametric method, KNN does not assume any specific distribution for the data, which enhances its robustness in dealing with noisy or incomplete observations. KNN is particularly effective for machine learning-based forecasting as it can identify influential patterns within noisy datasets. For continuous data, KNN matches data points based on calculated distances to determine similarity, which directly influences its accuracy and performance. The algorithm involves two main steps: first, it calculates the distance between the target data point and all points in the training set to find the closest neighbors. The Minkowski distance serves as the most comprehensive form of distance measurement; second, it classifies the target data point based on the majority label of these neighbors or predicts a value based on their averaged output in the case of regression [38].

$$d = \sqrt{(x_1 - x_2)^2 - (y_1 - y_2)^2} \tag{13}$$

In this metric, the distance d between two points (x_1, y_1) and (x_2, y_2) is defined in Euclidean space. The number of nearest samples used for classification or estimation is represented by K, a positive integer. Determining the optimal number of neighbors is essential, as it significantly impacts the accuracy of the estimation and the model's performance. In this study, Particle Swarm Optimization is utilized to optimize the value of K.

2.4. Improved particle swarm optimization

The Particle Swarm Optimization (PSO) algorithm, initially enhanced by [26]. PSO is inspired by the coordinated movement observed in natural phenomena like bird flocking, bee swarming, and fish schooling. Renowned for its simplicity in coding, cost-effectiveness, and consistent performance, PSO has established itself as a powerful algorithm for tackling variable optimization problems [39]. In PSO, individuals within the population are referred to as particles, collectively forming a swarm. These particles commence their optimization journey with random initial positions and velocities. Throughout the optimization process, particles adapt their positions and velocities as they navigate the search space. Additionally, each particle retains memory of the best position it has encountered in the search space. The parameters of SVM namely C and σ_{svm} and KNN namely k parameter will be optimized by using PSO algorithm. Initially, upper and lower bounds are defined for the SVM parameters and KNN parameters. Subsequently, random values within these bounds are generated for each particle, which are then employed as inputs for the SVM and KNN model. Following this, the fitness function is applied, with this study utilizing the (MAPE) as the fitness criterion to determine suitable SVM and KNN model parameters. The MAPE value for each particle is calculated using the fitness function in Eq. (14). This MAPE equation serves as the fitness function to evaluate the accuracy of each particle's prediction performance. By using MAPE as the fitness metric, the Improved PSO algorithm selects the optimal parameters for both the SVM and KNN models corresponding to each input variable.

$$\eta_{MAPE} = \frac{1}{\omega}\sum_{i=1}^{\omega}\frac{|y_i - \hat{y}_i|}{y_i} \tag{14}$$

where ω represents the number of subsets, $\hat{y}_i$ denotes the predicted value, y_i denoted the actual value and. D is the dimension of this function defines the length of each particle. Each member of the swarm, referred to as a particle, is represented as a vector X_i encompassing the parameters targeted for optimization within the objective function. In the multidimensional search space, denoted as m, the position $X_i = (x_{i1}, x_{i2}, x_{i3}, \ldots, x_{iD})$and velocity $V_i = (v_{1i}, v_{2i}, v_{3i}, \ldots, v_{Di})$ of the $i-th$ particle are initialized randomly within the range of possible solutions, $i = 1,2,\ldots,m$. To enhance the optimization process, the algorithm computes the objective function value for each particle, subsequently updating both their velocities and positions in accordance with specific equations as in Eq. (15) and Eq. (16)

$$v_{id}^{t+1} = \omega \cdot v_{id}^{t+1} + c_1 \cdot r_1 \cdot (p_{id} - x_{id}^t) + c_2 \cdot r_2 \cdot \left(p_{gd} - x_{id}^t\right) \tag{15}$$

$$x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1} \tag{16}$$

The ideal location of the particle represents as $P_i = (P_{1i}, P_{2i}, P_{3i}, \ldots, P_{Di})$. Optimal swarm location is $P_g = (P_{1g}, P_{2g}, P_{3g}, \ldots, P_{Dg})$. Under $i-th$ particle condition at $t-th$ iteration, x_{id}^t and v_{id}^t are $d-th$position and constituent of speed. $\omega$is inertial weight controlling velocity direction. PSO algorithms often converge to local optima due to the value of inertia weight. To address this issue, the inertia weight was enhanced by incorporating the concept of adaptive adjustment. Additionally, the current iteration count and the size of the population during each update were considered. The modified inertia weight ω is calculated using the following formula.

$$\omega = (\omega_{\max} - \omega_{\min}) \times \frac{(I \times P - q \times p)}{I \times P} + \omega_{\min} \tag{17}$$

Positive coefficient $c_1, c_2, r_1 and\, r_2$ where r_1 and r_2 are distributed evenly within the range of 0–1, and c_1 and c_2 represent constants. $I$is maximum iteration,$P$is size of population, $p$is the current iteration and $q$is the current population. The procedure of PSO optimizing the SVM and LSSVM parameters described as follows.

a. Initialise the parameters of PSO.
b. The collective of particles begins its journey with randomly assigned individual velocities and positions.
c. Fitness evaluation: The different initialized parameters are input into the LSSVM, and the fitness of each particle is assessed using the fitness function of PSO, as defined by Eq. (14).
d. Calculate the inertia weight.
e. Update both the global and individual best values based on the outcomes of the fitness value.
f. Velocity computation: The particle moves towards a fresh position by determining the velocity of its positional change. The velocity for each particle is derived using equation (15).
g. Position Update: Each particle transitions to its subsequent position following the guidelines outlined in Eq. (16).
h. Termination: Continue iterating through Steps 3–7 until the specified termination criteria are met.

3. Performance metrics

In this study, six different common indices employed to assess the precision of SVM-PSO, KNN-PSO, EEMD-SVM-PSO, EEMD-KNN-PSO, DEEMD-SVM-PSO, DEEMD-KNN-PSO models. The error evaluation methods chosen are Root Mean Square Error (RMSE), Mean Squared Error (MSE), Mean Absolute Percentage Error (MAPE) and Squared Correlation Coefficient (R^2). RMSE is a measure of the differences between values predicted by a model and the values observed. RMSE is one of the most widely used metrics because it gives a clear idea of how far off the model's predictions are from the actual values. The key benefit of RMSE is that it penalizes large errors more heavily due to the squaring of residuals. This is particularly useful in forecasting, where large deviations in predictions can be more problematic than small errors. MSE measures the average squared difference between the estimated and actual values. MAPE provides a measure of prediction accuracy as a percentage. Lower MAPE values indicate more accurate predictions relative to the size of the actual values. R-squared measures the proportion of variance in the actual values that is explained by the model. It is a key measure of how well the model fits the data and indicates the goodness of fit. An R^2 value close to 1 indicates that the model explains most of the variance in the data, while a value closer to 0 means the model does not explain much of the variability. R^2 is important for understanding how well the model captures the underlying patterns in the data. The best prediction model will have the lowest value for RMSE, MSE and MAPE and R^2 value that almost reaches one. R^2 is computed to evaluate the explained variance of models as presented in Eq. (21). The performance of the models can be measure by equation RMSE, MSE, MAPE and R^2as followed [40]:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(z_i - \widehat{z}_i)^2}, \tag{18}$$

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(z_i - \widehat{z}_i)^2, \tag{19}$$

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{\widehat{z}_i - z_i}{z_i}\right| \times 100, \tag{20}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(z_i - \widehat{z}_i)^2}{\sum_{i=1}^{n}(z_i - z_i')^2}, \tag{21}$$

where z_i is the observed value, $\widehat{z}_i$ is the forecasted value and n is number of data.

4. Results and discussion

4.1. Study area and data set

This paper focuses on the Klang River, which flows through Selangor and Kuala Lumpur in Malaysia as illustrated in Fig. 1. Hydrological data for the Klang River, specifically from Sri Muda, has been supplied by the Department of Irrigation and Drainage Malaysia (DID). Taman Sri Muda, a neighborhood situated in Shah Alam, Selangor, is part of this study area. The Klang River, or Sungai Klang, is a prominent waterway in the region. In urban areas like Taman Sri Muda, rivers often play a crucial role in drainage and may be subject to environmental considerations. The length of the river within this community is likely to be a relatively short segment of the entire Klang River, which is approximately 25 kilometers long (Station 3015432).

The daily water level data available covers the period from 2011 to 2022. Extreme weekly water levels are obtained through the utilization of the block maxima-minima approach. The dataset comprises a total of 626 weeks of daily water level readings. To enhance prediction accuracy, the data was segmented into weekly intervals. For model development in this study, 80 % of the data was allocated for training the SVM and LSSVM models, covering the period from January 1, 2011 to August 1, 2020. The remaining 20 % was reserved for the validation or testing phase of these models, spanning from August 2, 2020 to December 31, 2022. Consequently, the forecasting models were built using 500 weeks of training data, while predictions were based on the 126 weeks in testing data. Fig. 2 displays the weekly water level data for Klang River, including training, testing datasets, and the linear regression line. The study area is classified into four alert levels (normal, alert, warning, and danger) with corresponding threshold water levels of 2.8 m, 4.4 m, 4.7 m, and 5.0 m respectively. Flooding occurs when water levels surpass the normal threshold. Therefore, accurate water level forecasting is crucial for effective flood warning systems, as it helps in alerting authorities and informing the affected populations. All the models will run a program by using R Software.

Table 2 shows the descriptive statistics of the river water level dataset that reveals important insights into its distribution and variability. The minimum value recorded is 1.220 while the maximum reaches 10.50, resulting in a range of 9.280 which indicates significant fluctuations over the observation period. The mean water level is 3.090 slightly higher than the median of 2.950 suggesting a right-skewed distribution likely caused by occasional high water level events. The first quartile (Q1) is 2.465 and the third quartile (Q3) is 3.625 with an interquartile range (IQR) of 1.160 indicating that most observations are concentrated within this central range. This study has utilized simple linear regression to identify trends within the time series of water levels. The analysis using simple linear regression reveals a distinct upward trend in extreme weekly water levels. The upward slope of the regression line serves as an early warning signal for potential flooding. When water levels consistently increase, it suggests a heightened risk of reaching critical flood levels. Early detection of this trend allows authorities to implement precautionary measures and issue timely flood warnings to communities in vulnerable areas.

4.2. Data decomposition

In this research, Empirical Ensemble Mode Decomposition (EEMD) is

Fig. 1. Map of Klang River that flows through Taman Sri Muda, Selangor, Malaysia.

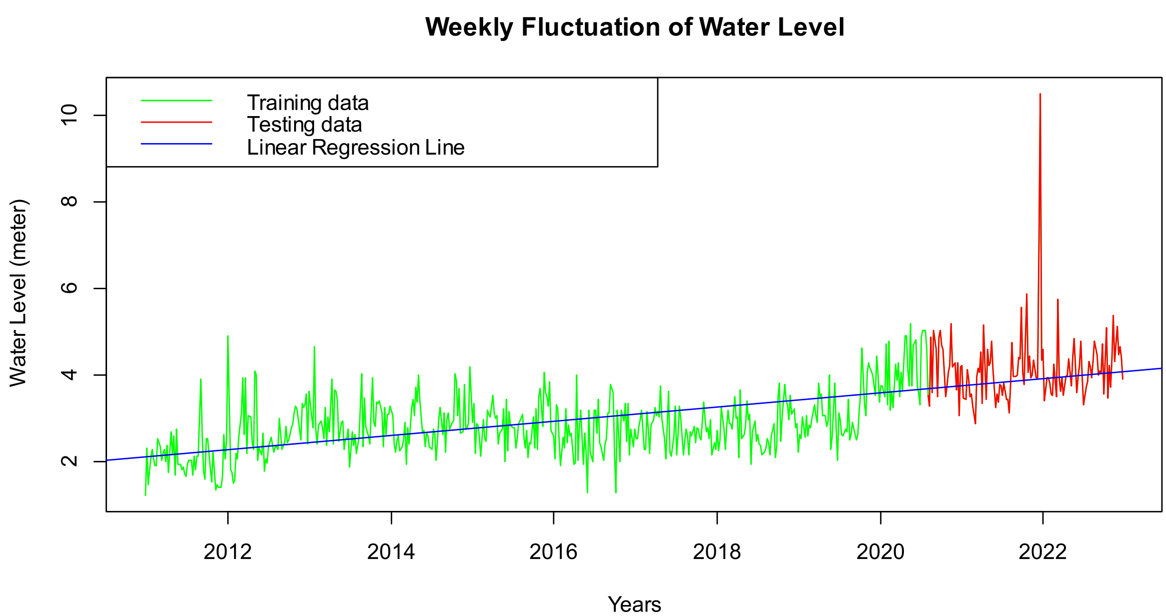

Fig. 2. Weekly series of water level data of Klang River for training, testing and linear regression line.

Table 1
Parameter settings of the methods.

Methods	Parameters	Value
Particle Swarm Optimization	Population Size, P	35
	Maximum Iteration, I	100
	Acceleration constants, (c_1, c_2)	(1,2)
Support Vector Machine	Cost, C	[1100]
	Kernel, σ_{svm}	[0.01, 1]
K-Nearest Neighbor	k	[1,5]

Table 2
Descriptive statistic of weekly water level data.

Minimum	First Quartile	Median	Mean	First Quartile	Maximum	Mode
1.220	2.465	2.950	3.090	3.625	10.500	10.500

used to analyze the time series of water level data. This method was essential in preprocessing the weekly water level data of the Klang River in Sri Muda, Malaysia, from 2011 to 2022. The EEMD technique was specifically applied to decompose the original water level time series into several distinct Intrinsic Mode Functions (IMFs) and a residual component. In this investigation, an ensemble size of 100 was chosen, and each ensemble member was augmented with white noise characterized by a standard deviation of 0.2. It's important to note that these parameter choices align with those used by [11] in prior work, and we won't reiterate these details here. Fig. 3 shows the outcome of the EEMD process, where the Intrinsic Mode Functions (IMFs) and residual components are used as input variables for water level forecasting. The

Table 3
RMSE, MSE, MAPE and R^2 value of proposed models for testing data.

	RMSE(m)	MSE(m^2)	MAPE(m)	R^2
SVM-PSO	0.99425	0.9885	0.16707	0.4125
EEMD-SVM-PSO	0.78265	0.6127	0.12466	0.5123
DEEMD-SVM-PSO	0.64948	0.4228	0.09256	0.7895
KNN-PSO	0.96235	0.9261	0.14045	0.4698
EEMD-KNN-PSO	0.74248	0.5505	0.10980	0.5698
DEEMD-KNN-PSO	0.52235	0.2725	0.08596	0.84596

EEMD technique has decomposed the water level data into 8 IMF components and one residual component. Fig. 4 shows the outcome of the DEEMD process of IMF1 from EEMD.

4.3. Results and discussion

The figure displays convergence curves for SVM and KNN models applied to various components obtained after a double decomposition process. This includes the IMFs and the residual component. The convergence curves indicate how well each model (SVM and KNN) performs across different components extracted by the decomposition method. Across the subplots, it can be observed that both models generally exhibit good convergence, with performance improving as the iteration progresses. However, there are differences in the convergence speed and final accuracy between SVM and KNN for individual IMFs and the residual. In most subplots, the KNN model (green lines) achieves slightly better convergence and higher accuracy than the SVM model (blue lines), especially for the earlier IMFs (high-frequency components). KNN's faster convergence in the initial iterations suggests that it can quickly learn and fit patterns in the data particularly for high-frequency IMFs. SVM shows a slower convergence rate but tends to reach a high accuracy in the later iterations. This indicates that while SVM requires more iterations to optimize, it eventually captures more complex, nonlinear patterns present in the data.

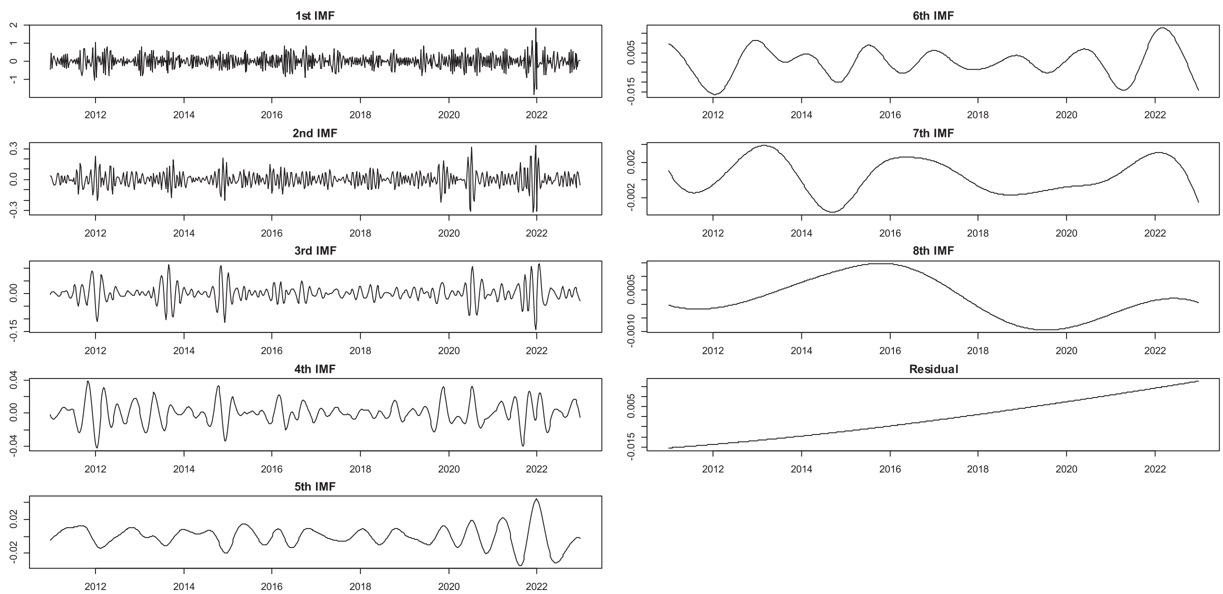

Fig. 3. The outcomes derived from the application of the EEMD method for the decomposition of weekly water level data.

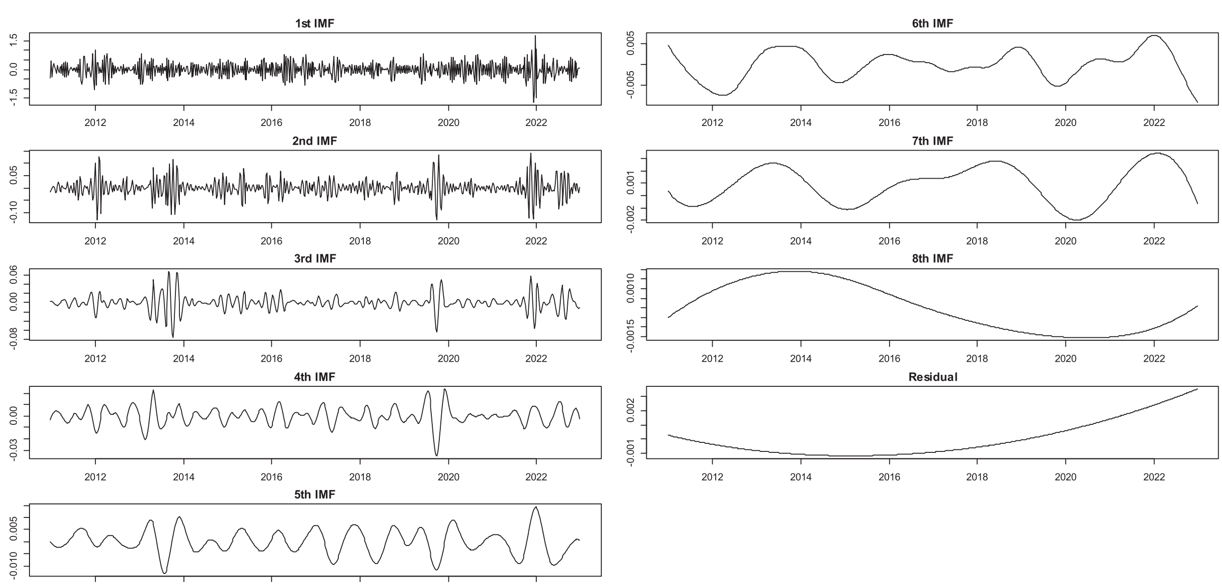

Fig. 4. The outcomes derived from the application of the DEEMD.

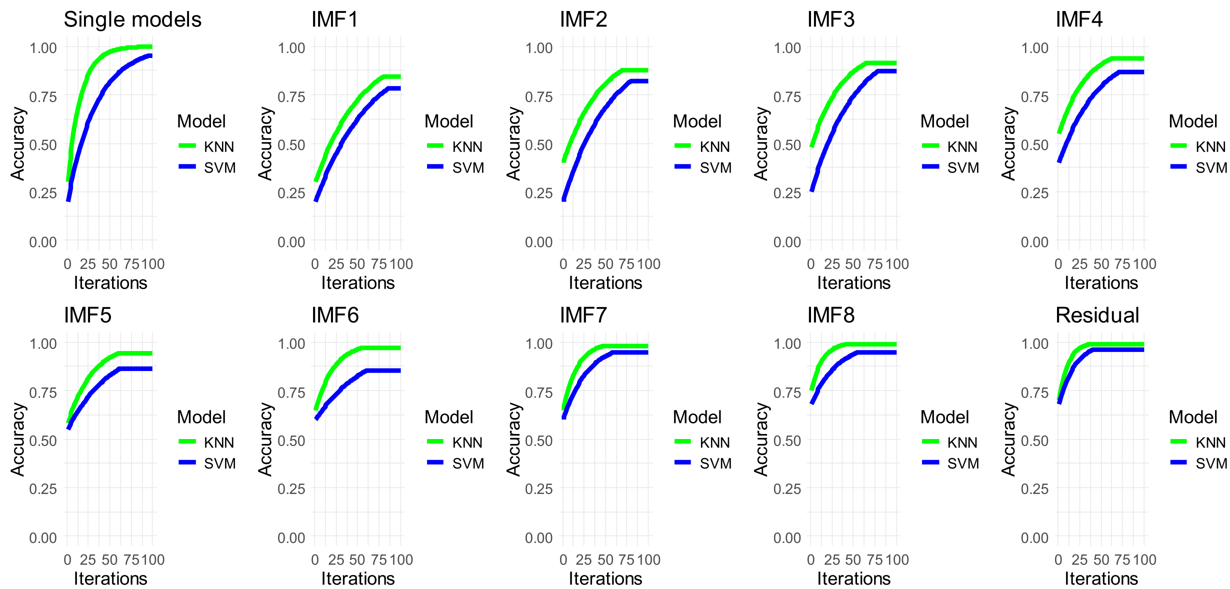

Fig. 5. Convergence Analysis of KNN and SVM models on single models and decomposed water level components.

Fig. 6 indicate that both EEMD-SVM and EEMD-KNN show decreasing error metrics across the IMFs, reflecting the effectiveness of EEMD in breaking down the time series into components that are easier for the models to predict. However, EEMD-KNN consistently shows lower RMSE, MSE, and MAPE values compared to EEMD-SVM across most IMFs and the residual component. This suggests that the EEMD-KNN combination are more robust in handling different components of the data, including both the detailed and residual parts. Based on the error metrics, EEMD-KNN seems to outperform EEMD-SVM, showing lower prediction errors and higher accuracy across all metrics (RMSE,

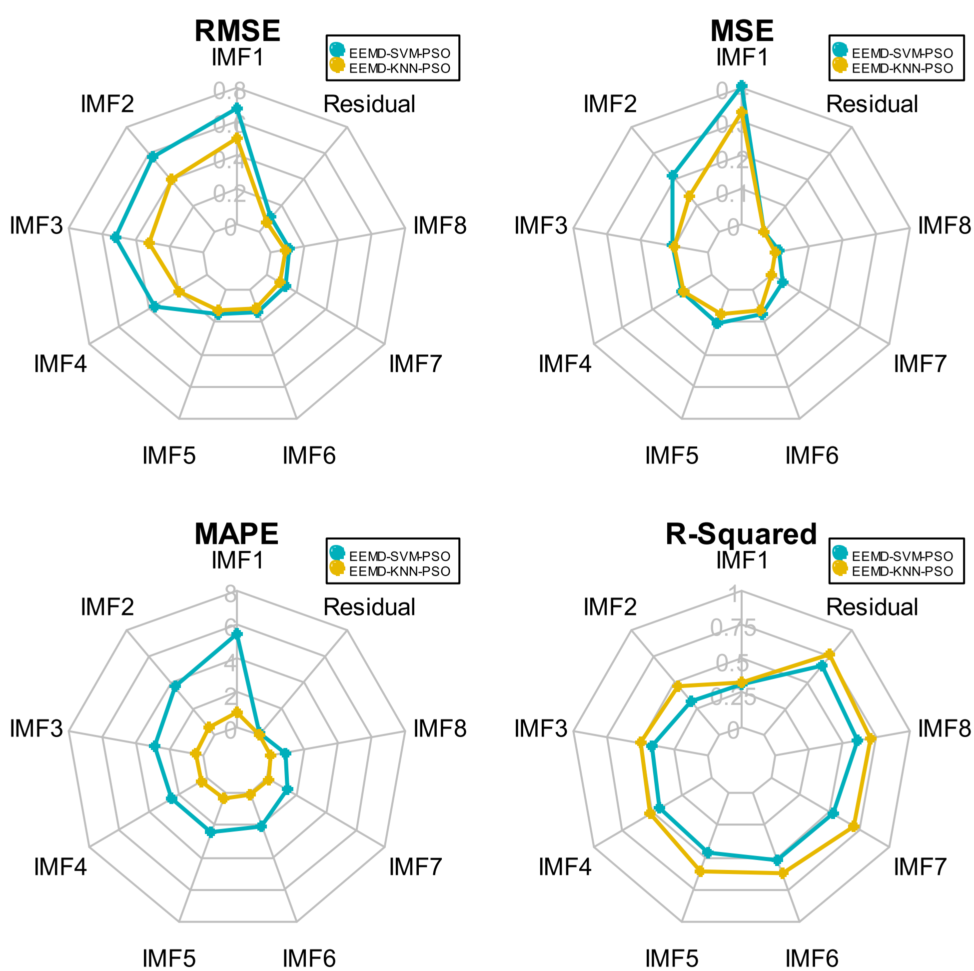

Fig. 6. Radar plot of RMSE, MSE, MAPE, R^2 of IMFs from EEMD on water level model.

MSE, and MAPE). KNN better suited for this particular data set, especially when accuracy and handling finer details (as indicated by the residuals) are crucial. In summary, while both EEMD-SVM and EEMD-KNN are effective in reducing error metrics progressively through the IMFs, EEMD-KNN consistently shows superior performance. This indicates that KNN, when combined with EEMD is more capable of capturing the underlying patterns and nuances of the data, leading to more accurate and reliable predictions. Further analysis and validation could involve testing these models on different datasets or using other metrics to confirm these findings.

The error metrics (RMSE, MSE, and MAPE) consistently show that DEEMD-KNN outperforms DEEMD-SVM in forecasting water levels in Fig. 7. DEEMD-KNN demonstrates lower errors across all metrics and higher R^2 value, especially in the later IMFs and residuals, suggesting it can more effectively capture both major trends and minor fluctuations in water levels. KNN's performance indicates that its non-parametric nature and reliance on historical patterns make it more adept at handling the decomposed, often non-linear features resulting from DEEMD. Based on the observed metrics, DEEMD-KNN is preferable for forecasting water levels due to its superior ability to minimize errors and accurately predict future levels. While DEEMD-SVM is also effective, it appears less suited for capturing the intricate details and high variability that characterize water level data. The SVM's higher error values across initial IMFs indicate potential limitations in handling complex relationships within the data, which KNN manages more effectively. The combination of DEEMD with KNN offers a more accurate and reliable approach for forecasting water levels compared to DEEMD with SVM. DEEMD-KNN consistently shows lower RMSE, MSE, and MAPE values and high R^2 value, demonstrating its capability to handle both broad and detailed aspects of water level data effectively. These findings suggest that using DEEMD-KNN could improve water level forecasting, which is crucial for applications like flood prediction.

Compared to Figs. 5 and 6, the analysis of DEEMD (Double Ensemble Empirical Mode Decomposition) models combined with KNN (K-Nearest Neighbors) and SVM (Support Vector Machine) shows that DEEMD models consistently outperform EEMD models in forecasting water levels. Across all error metrics, DEEMD models demonstrate lower error values and highest R^2 value, indicating better accuracy and reliability in predictions. The enhanced performance of DEEMD models can be attributed to their ability to capture more complex and intricate patterns within the water level data, providing more refined decompositions. This improvement makes DEEMD models a more effective choice for accurate water level forecasting, which is essential for flood prediction. Therefore, implementing DEEMD-based approaches should be considered for more accurate forecasting in scenarios where water level predictions are critical.

Table 1 shows RMSE, MSE, MAPE and R^2 values of proposed models reveals clear distinctions in their predictive performance. Single models like SVM with PSO and KNN with PSO show higher error values, indicating less accurate forecasting abilities. Incorporating PSO into these models helps in optimizing the selection of hyperparameters, which slightly improves their performance. However, hybrid models using EEMD combined with KNN and SVM, and further enhanced by PSO, show a noticeable reduction in error values. This improvement highlights how EEMD effectively decomposes the water level time series into intrinsic mode functions (IMFs), enabling the predictive models to better capture complex data patterns.

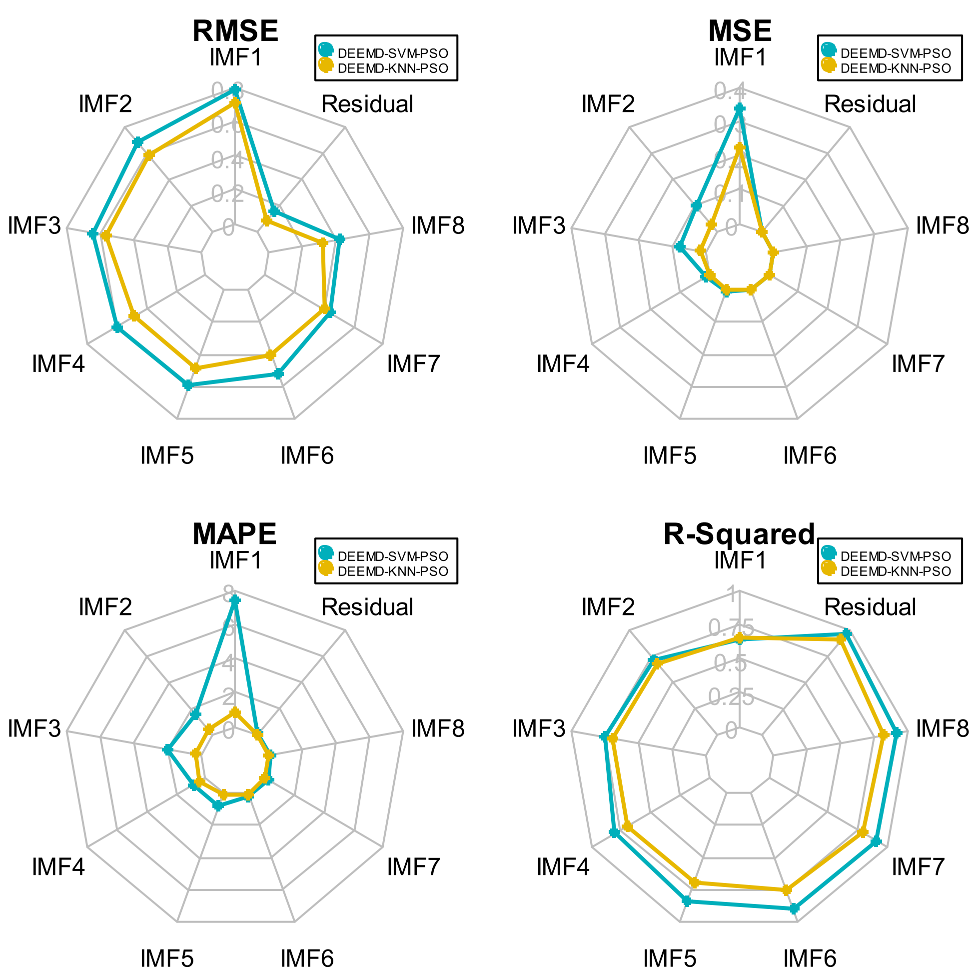

Fig. 7. Radar plot of RMSE, MSE, MAPE and R^2 of IMFs from DEEMD on water level model.

The most significant improvement is observed in the double decomposition models, DEEMD-KNN-PSO and DEEMD-SVM-PSO, which exhibit the lowest RMSE, MSE, and MAPE values and highest R^2 value. By taking the first IMF from EEMD and decomposing it again through DEEMD, these models are able to capture even finer details and underlying patterns in the data that are not evident in single or hybrid models. The double decomposition approach not only enhances the representation of the data but also reduces noise, resulting in higher accuracy in forecasting. This layered decomposition, combined with the optimization capability of PSO, provides a robust framework for handling the non-linear, non-stationary nature of water level time series, making DEEMD models the most effective for accurate water level predictions.

Fig. 8 illustrates a comparison between observed water levels and water level forecasts generated by different models, each represented by distinct colored lines. The observed water levels serve as a baseline to evaluate the accuracy and efficiency of the forecasting models. The models include single and hybrid approaches, such as KNN (K-Nearest Neighbors) and SVM (Support Vector Machine) optimized with improved PSO (Particle Swarm Optimization), as well as more advanced models like EEMD-KNN-PSO, EEMD-SVM-PSO, and DEEMD-based models DEEMD-KNN-PSO and DEEMD-SVM-PSO.

The DEEMD-KNN-PSO model is indicated to be the most efficient among the models compared, as it closely follows the observed water level trends with minimal deviation. This enhanced performance can be attributed to the combination of double decomposition using DEEMD (Double Ensemble Empirical Mode Decomposition) and the optimization capabilities provided by improved PSO. DEEMD improves the forecasting accuracy by decomposing the first intrinsic mode function (IMF) from EEMD, allowing the model to capture finer details and complex patterns in the data that are often associated with non-linear and non-stationary processes. This further decomposition helps in isolating noise and providing a more refined input for the forecasting model. Improved PSO plays a critical role in enhancing the model's performance by optimizing the parameters of KNN, ensuring that the model is better suited to handle the intricacies of water level forecasting. By using a swarm intelligence-based approach, improved PSO efficiently navigates the parameter space to find optimal solutions that minimize error metrics such as RMSE, MSE, and MAPE and produce high R^2 value. As a result, the DEEMD-KNN-PSO model is not only capable of accurately predicting water levels but also does so with greater stability and less variability compared to other models. Overall, the graph demonstrates that while all models attempt to capture the trends in water levels, the DEEMD-KNN-PSO model achieves the highest accuracy. This conclusion is drawn from the model's ability to closely align with the observed data, exhibiting fewer spikes and fluctuations, indicating its superior capacity for precise water level forecasting.

5. Conclusion

In this study, KNN and SVM models were extensively used for weekly water level forecasting, but prior research often overlooked the importance of integrating data features into model construction. This research proposed an enhanced method for predicting weekly water levels by integrating data decomposition techniques. Specifically, Ensemble Empirical Mode Decomposition (EEMD) was utilized to decompose the original water level dataset into distinct Intrinsic Mode Function (IMF) components with lower complexity and clear periodicity. Additionally, a secondary decomposition was applied to the highest-frequency IMF 1 derived from EEMD. The machine learning models employed in this study for water level prediction were KNN and SVM, both optimized using improved Particle Swarm Optimization (PSO). The efficacy of the EEMD and double EEMD (DEEMD) methods was evaluated using various performance metrics when combined with KNN and SVM models. The experimental results revealed that the DEEMD-KNN-PSO and DEEMD-SVM-PSO models outperformed other single decomposition and standalone models across multiple evaluation criteria. Notably, the DEEMD-KNN-PSO model demonstrated superior accuracy in predicting water levels for the Klang River in Taman Sri Muda, Malaysia, highlighting its effectiveness and reliability for water level forecasting based on the testing data.

Future work could explore the application of this dual decomposition approach with KNN and SVM optimized by PSO to other hydrological datasets and geographical regions to validate its robustness and generalizability. Additionally, incorporating other advanced decomposition methods or integrating deep learning techniques, such as hybrid models with LSTM or GRU, could further enhance prediction accuracy. Moverover, the optimization process could be enhanced by replacing the Particle Swarm Optimization (PSO) technique with Genetic Algorithm (GA), which may improve convergence speed and solution quality [41, 42].

Funding

The research is fully funded and supported by Ministry of Higher Education Malaysia for Fundamental Research Grant Scheme, FRGS/1/2022/STG06/USM/03/1.

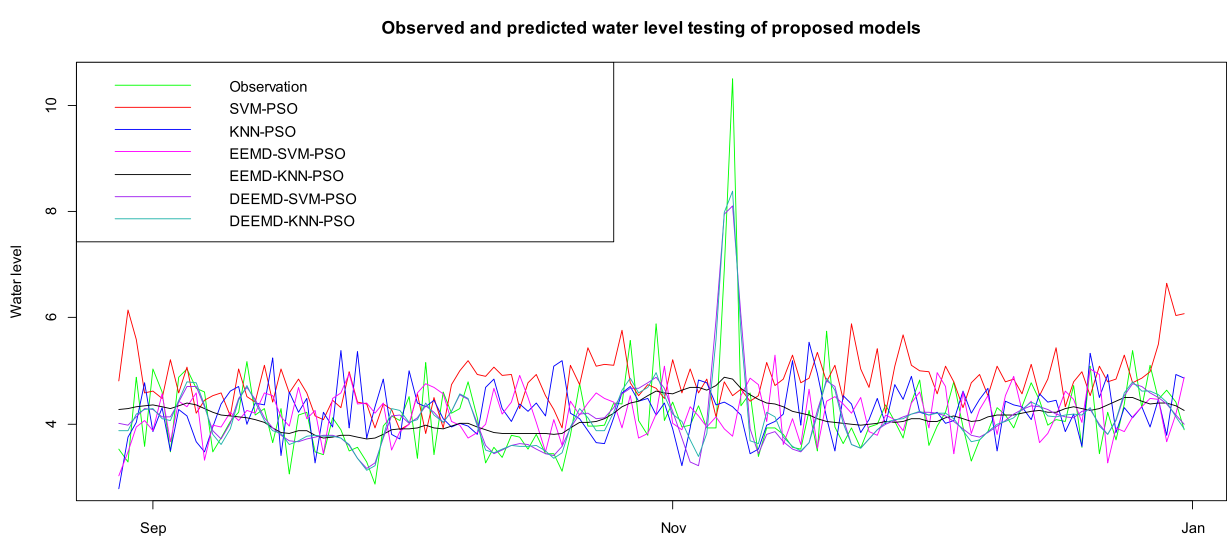

Fig. 8. Forecasted and observed of weekly water level in testing phase.

CRediT authorship contribution statement

Siti Zulaikha Mohd Jamaludin: Validation. **Mohd Shareduwan Mohd Kasihmuddin:** Validation, Supervision. **Mohd. Asyraf Mansor:** Validation, Supervision. **Nur Ezlin Zamri:** Validation, Writing – review & editing. **Muhammad Fadhil Marsani:** Supervision, Funding acquisition. **Vikneswari Someetheram:** Writing – review & editing, Software, Project administration, Formal analysis, Conceptualization.

Declaration of Competing Interest

None

Acknowledgments

All the authors gratefully acknowledged the financial support from the "Ministry of Higher Education Malaysia for Fundamental Research Grant Scheme with Project Code: FRGS/1/2022/STG06/USM/03/1 "

References

[1] S.K. Jain, P. Mani, S.K. Jain, P. Prakash, V.P. Singh, D. Tullos, S. Kumar, S. P. Agarwal, A.P. Dimri, A Brief review of flood forecasting techniques and their applications, Int. J. River Basin Manag. 16 (3) (2018) 329–344.
[2] M.N.A. Zakaria, M.A. Malek, M. Zolkepli, A.N. Ahmed, Application of artificial intelligence algorithms for hourly river level forecast: a case study of Muda River, Malaysia, Alex. Eng. J. 60 (4) (2021) 4015–4028.
[3] F.Y. Dtissibe, A.A.A. Ari, C. Titouna, O. Thiare, A.M. Gueroui, Flood forecasting based on an artificial neural network scheme, Nat. Hazards 104 (2020) 1211–1237.
[4] S. Nevo, E. Morin, A. Gerzi Rosenthal, A. Metzger, C. Barshai, D. Weitzner, D. Voloshin, F. Kratzert, G. Elidan, G. Dror, G. Begelman, Flood forecasting with machine learning models in an operational framework, Hydrol. Earth Syst. Sci. 26 (15) (2022) 4013–4032.
[5] R. Lguensat, P. Tandeo, P. Ailliot, M. Pulido, R. Fablet, The analog data assimilation, Mon. Weather Rev. 145 (2017) 4093–4107.
[6] N. Zhou, J. Hou, H. Chen, G. Chen, B. Liu, A rapid forecast method for the process of flash flood based on hydrodynamic model and KNN algorithm, Water Resour. Manag. 38 (6) (2024) 1903–1919.
[7] Y. Luo, Z. Dong, Y. Liu, X. Wang, Q. Shi, Y. Han, Research on stage-divided water level prediction technology of rivers-connected lake based on machine learning: a case study of Hongze Lake, China, Stoch. Environ. Res. Risk Assess. (2021) 1–17.
[8] A. Altunkaynak, E. Kartal, Transfer sea level learning in the Bosphorus Strait by wavelet based machine learning methods, Ocean Eng. 233 (2021) 109116.
[9] O.H. Kombo, S. Kumaran, Y.H. Sheikh, A. Bovim, K. Jayavel, Long-term groundwater level prediction model based on hybrid KNN-RF technique, Hydrology 7 (3) (2020) 59.
[10] M. Liu, Y. Huang, Z. Li, B. Tong, Z. Liu, M. Sun, F. Jiang, H. Zhang, The applicability of LSTM-KNN model for real-time flood forecasting in different climate zones in China, Water 12 (2) (2020) 440.
[11] Y. Karsavran, Comparison of random forest, SVR and KNN based models in sea level prediction for Erdemli Coast of Mersin, Celal Bayar Univ. J. Sci. 20 (2) (2024) 14–18.
[12] V.N. Vapnik, The Nature of Statistical Learning Theory, Springer, New York, 1995, p. 314.
[13] B. Schölkopf, A.J. Smola, Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, MIT press, 2002.
[14] N. Natarajan, C. Sudheer, Groundwater level forecasting using soft computing techniques, Neural Comput. Appl. 32 (2020) 7691–7708.
[15] H. Wang, L. Song, Water level prediction of rainwater pipe network using an SVM-based machine learning method, Int. J. Pattern Recognit. Artif. Intell. 34 (02) (2020) 2051002.
[16] S. Samantaray, A. Sahoo, D.P. Satapathy, Prediction of groundwater-level using novel SVM-ALO, SVM-FOA, and SVM-FFA algorithms at Purba-Medinipur, India, Arab. J. Geosci. 15 (8) (2022) 723.
[17] N.E. Huang, Z. Shen, S.R. Long, M.C. Wu, H.H. Shih, Q. Zheng, N. Yen, C. Tung, H. H. Liu, The empirical mode decomposition and the Hilbert spectrum for nonlinear and non-stationary time series analysis, Proc. R. Soc. Lond. Ser. A Math. Phys. Eng. Sci. 454 (1998) 903–995.
[18] W.C. Wang, K.W. Chau, D.M. Xu, X.Y. Chen, Improving forecasting accuracy of annual runoff time series using arima based on EEMD decomposition, Water Resour. Manag 29 (2015) 2655–2675.
[19] W.Y. Duan, Y. Han, L.M. Huang, B.B. Zhao, M.H. Wang, A hybrid emd-svr model for the short-term prediction of significant wave height, Ocean Eng. 124 (2016) 54–73.
[20] G. Napolitano, F. Serinaldi, L. See, Impact of emd decomposition and random initialisation of weights in ann hindcasting of daily stream flow series: An empirical examination, J. Hydrol. 406 (2011) 199–214.
[21] L. Karthikeyan, D. Nagesh Kumar, Predictability of nonstationary time series using wavelet and emd based arma models, J. Hydrol. 502 (2023) 103–119.
[22] Chen, Y.C., Yeh, H.C., Kao, S.P., Wei, C., Su, P.Y. Water level forecasting in tidal rivers during typhoon periods through ensemble empirical mode decomposition.
[23] S.J. Mohammed, S.L. Zubaidi, N. Al-Ansari, H.M. Ridha, N.S.S. Al-Bdairi, Hybrid technique to improve the river water level forecasting using artificial neural network-based marine predators algorithm, Adv. Civ. Eng. (1) (2022) 6955271.
[24] B. Du, S. Huang, J. Guo, H. Tang, L. Wang, S. Zhou, Interval forecasting for urban water demand using PSO optimized KDE distribution and LSTM neural networks, Appl. Soft Comput. 122 (2022) 108875.
[25] S. Salari, M. Moghaddasi, M. Mohammadi Ghaleni, M. Akbari, Groundwater level prediction in Golpayegan aquifer using ANFIS and PSO combination, Iran. J. Soil Water Res. 52 (3) (2021) 721–732.
[26] Kennedy, J. and Eberhart, R., 1995, November. Particle swarm optimization. In Proceedings of ICNN'95-International Conference on Neural Networks (Vol. 4, pp. 1942-1948).
[27] J.F. Ruma, M.S.G. Adnan, A. Dewan, R.M. Rahman, Particle swarm optimization based LSTM networks for water level forecasting: a case study on Bangladesh river network, Results Eng. 17 (2023) 100951.
[28] S. Anupam, P. Pani, Flood forecasting using a hybrid extreme learning machine-particle swarm optimization algorithm (ELM-PSO) model, Model. Earth Syst. Environ. 6 (1) (2020) 341–347.
[29] F.R. Aderyani, S.J. Mousavi, F. Jafari, Short-term rainfall forecasting using machine learning-based approaches of PSO-SVR, LSTM and CNN, J. Hydrol. 614 (2022) 128463.
[30] Y. Dehghan, S.M. Amini Zenooz, Z.F. Pour, Analysis of sea level fluctuations around the Australian coast with anomaly time series analysis approach, Mar. Environ. Res. 181 (2022) 105742.
[31] A.R. Chowdhury, K. Guhathakurta, I. Mukherjee, Empirical mode decomposition analysis of two different financial time series and their comparison, Chaos, Solitons Fractals 37 (2008) 1214–1227.
[32] G. Sun, R. Zhang, Z. Liu, L. Wu, Q. Yu, X. Tan, EMD-based noise reduction study of steel cored conveyor belt containing slag signal, Alex. Eng. J. 98 (2024) 56–67.
[33] Z. Wu, N.E. Huang, Ensemble empirical mode decomposition: a noise-assisted data analysis method, Adv. Adapt. Data Anal. 1 (2009) 1–41.
[34] M.A. Colominas, G. Schlotthauer, M.E. Torres, An unconstrained optimization approach to empirical mode decomposition, Digit. Signal Process 40 (2015) 164–175.
[35] O. Abu Arqub, J. Singh, B. Maayah, M. Alhodaly, Reproducing kernel approach for numerical solutions of fuzzy fractional initial value problems under the Mittag–Leffler kernel differential operator. Math. Methods Appl. Sci. 46 (7) (2023) 7965–7986.
[36] O. Abu Arqub, J. Singh, M. Alhodaly, Adaptation of kernel functions-based approach with Atangana–Baleanu–Caputo distributed order derivative for solutions of fuzzy fractional Volterra and Fredholm integrodifferential equations, Math. Methods Appl. Sci. 46 (7) (2023) 7807–7834.
[37] T. Cover, P. Hart, Nearest neighbor pattern classification, IEEE Trans. Inf. Theory 13 (1) (1967) 21–27.
[38] K.S.M.H. Ibrahim, Y.F. Huang, A.N. Ahmed, C.H. Koo, A. El-Shafie, A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting, Alex. Eng. J. 61 (1) (2022) 279–303.
[39] M.I. Dieste-Velasco, Fault detection in analog electronic circuits using fuzzy inference systems and particle swarm optimization, Alex. Eng. J. 95 (2024) 376–393.
[40] W. Zhang, M. Kou, M. Lv, Y. Shao, Improved combined system and application to precipitation forecasting model, Alex. Eng. J. 61 (12) (2022) 12739–12757.
[41] O.A. Arqub, Z. Abo-Hammour, Numerical solution of systems of second-order boundary value problems using continuous genetic algorithm, Inf. Sci. 279 (2014) 396–415.
[42] Z.E. Abo-Hammour, O. Alsmadi, S. Momani, O. Abu Arqub, A genetic algorithm approach for prediction of linear dynamical systems, Math. Probl. Eng. (1) (2013) 831657.