Iranian Journal of Science and Technology, Transactions of Civil Engineering (2022) 46:3453–3472
https://doi.org/10.1007/s40996-021-00809-2

RESEARCH PAPER

The Performance Analysis of Robust Local Mean Mode Decomposition Method for Forecasting of Hydrological Time Series

Levent Latifoğlu[1]

Received: 12 March 2021 / Accepted: 24 December 2021 / Published online: 31 January 2022

Abstract

Accurate forecasting of streamflow data over daily timescales is a critical problem for the long-term management of water resources, agricultural uses, and many more purposes. This study proposes a new hybrid approach that combines the Robust Local Mean Decomposition (RLMD) method and the Artificial Neural Network (ANN) method for the prediction of streamflow data. Monthly streamflow data were split into the training and the testing part firstly and threefold cross-validation was performed to obtain a more reliable model. After building of the model using the training data, the proposed model was tested on the testing data. Also to compare the performance of the RLMD–ANN model and the Support Vector Regression (SVR) model, Long Short-Term Memory Networks (LSTM) were used for forecasting of subband signal and streamflow data. Also, the hybrid Empirical Mode Decomposition (EMD) model and hybrid Autoregressive Integrated Moving Average (ARIMA) model were used for comparison of the proposed model. Therefore, the RLMD–ANN model was compared with RLMD–SVR, RLMD–LSTM, EMD–ANN, Additive–ARIMA–ANN, ANN, SVR, and LSTM models. The numerical results of the study were assessed concerning the Mean Square Error (MSE), Mean Absolute Error (MAE), Determination Coefficient (R^2), Correlation Coefficient (R), and Kruskal–Wallis test was used to indicate whether the results are statically significant. One- to three-ahead forecast and one–two inputs were applied to the models. The mean one-ahead forecasting performance of the three folds was calculated for two inputs with MSE, MAE, R^2, and R parameters as 0.0060, 0.0522, 0.7342, and 0.8532 respectively. The obtained results show that the novel RLMD–ANN model is a reliable, efficient, and high-performant model for forecasting streamflow data.

Keywords Forecasting · Streamflow data · Robust Local Mean Decomposition (RLMD) · Artificial Neural Networks (ANN) · Support Vector Regression (SVR) · Long Short-Term Memory Networks (LSTM) · Empirical Mode Decomposition (EMD)

1 Introduction

Accurate and precise estimation of river flows has a critical role in reservoir management, risk assessment, drought prediction, flood displacement, disaster management, and water planning. Therefore, many researchers have been studying this topic for the last past decades (Yaseen and El-Shafie 2015; Yaseen et al. 2016, 2017; Wang et al. 2017; Sahoo et al. 2019; Alobaidi et al. 2020; Kasiviswanathan et al. 2016; Tongal and Booij 2018; Humphrey et al. 2016; Ni et al. 2019; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018; Zuo et al. 2020; Solomatine et al. 2008; Siddiqi et al. 2021; Niu and 2021). However, it is difficult to forecast river flow data because of having complex, nonlinear, dynamical, and chaotic disturbances, and the randomness behavior. Forecasting models are defined in two categories as process based and data driven based (Box and Jenkins 1970). Process-based techniques require information about the physical properties of the process. In these techniques, processes are analyzed in two stages. The first stage is the determination of mathematical models; the second stage is the obtaining of numerical solutions.

In the mathematical modeling phase, the process is defined by mathematical equations. Then, an accurate and efficient numerical solution of these equations is realized. It is important to understand the process theoretically in these models with assumptions involving approaches with more

✉ Levent Latifoğlu
latifoglu@erciyes.edu.tr

[1] Department of Civil Engineering, Faculty of Engineering, Erciyes University, 38039 Kayseri, Turkey

data requirements. In case of a lack of information about the process to be modeled, it is not possible to realize process-based models. Data-based techniques are defined as black-box models. These models are empirical, based on observations, simple, and easier to implement. Also, it does not require physical information about the process. Data-driven techniques based on statistical data and linear approaches have been applied for time series modeling and forecasting including Box Jenkins, Autoregressive (AR), Moving Average (MA), and Autoregressive Moving Average (ARMA). Because river flow data are nonlinear and non-stationary in nature, these methods are not suitable for forecasting river flow data. For this reason, Machine Learning methods (ML), Random Forest, Fuzzy Rule-Based systems, Bayesian Regression, Artificial Neural Networks, Genetic Algorithms, Adaptive neural fuzzy inference systems (ANFISs), Complex Networks and Deep learning algorithms, and Long short-term memory network (LSTM) with their combination with good generalization capability and adaptability have been widely used in recent years (Yaseen and El-Shafie 2015; Yaseen et al. 2016, 2017; Wang et al. 2017; Sahoo et al. 2019; Alobaidi et al. 2020; Kasiviswanathan et al. 2016; Tongal and Booij 2018; Humphrey et al. 2016; Ni et al. 2019; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018; Zuo et al. 2020; Solomatine et al. 2008; Siddiqi et al. 2021; Niu and Feng 2021; Ghorbani et al. 2020, 2021).

Forecasting of subband signals obtained from decomposition of the original signal according to multiscale features can be simpler than the original signal, because, hydrological time series including streamflow data have highly nonstationary character and have periodic oscillations terms with noise components. Also preprocessing of streamflow data can reveal the hidden pattern in the data giving important information. Therefore, the application of suitable data preprocessing methods to the original signal combining ML models can improve the forecasting performance of data-driven models using advantages of feature extraction detecting hidden structure in raw (not processed) time-series data.

In the literature, there are studies based on hybrid models to forecast streamflow data by using Fourier Transform (FT) (Yu et al. 2018), Wavelet Transform (WT) (Kasiviswanathan et al. 2016; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018), Empirical Mode Decomposition (EMD) (Kisi et al. 2014), Ensemble Empirical Mode Decomposition (EEMD) (Ali et al. 2020) and Singular Spectrum Analysis (SSA) (Marques et al. 2006; Yu et al. 2017), Variational Mode Decomposition (VMD) (Zuo et al. 2020; Purohit et al. 2021) methods at the pre-processing stage of the data. These hybrid or ensemble methods seem to largely increase the forecasting performance.

Also, hybrid models were proposed using ARIMA and exponential smoothing models (ETS) with machine learning methods for forecasting studies, and statistically promising results were obtained for the used data sets (Panigrahi and Behera 2017; Purohit et al. 2021).

Some studies in the literature show that the decomposition of the raw data directly into subbands without dividing it firstly into training and testing data has led to a problem. This problem is the use of future information in establishing hybrid models (Zhang et al. 2015). In this case, since the training data contain information from the future in the data set, it becomes a hindcasting problem rather than a real forecasting problem. Therefore, the correct preprocessing is important to obtain reliable results for the determination of the hybrid model. In this study, the data set has been first divided into training and testing data and then decomposed into subbands and threefold cross-validation is used.

EMD method is based on local time scale during decomposition stage using the envelopes of extremely determined by a spline. This leads to a problem called the end effect. Similar to EMD, the DWT method suffers from boundary effect due to the necessity of computing the convolution, requiring the non-existent values beyond the boundary. Also, the DWT method needs to define the mother wavelet and the number of decomposition levels. Recently, although a robust VMD method is applied for the decomposition of time series data to eliminate the boundary effect as an advantage, the disadvantage of VMD is that its number of decomposition levels is specified empirically (Kasiviswanathan et al. 2016; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018; Zuo et al. 2020; Niu and Feng 2021; Box and Jenkins 1970; Ghorbani et al. 2020, 2021; Yu et al. 2018; Kisi et al. 2014; Ali et al. 2020; Marques et al. 2006; Yu et al. 2017; Panigrahi and Behera 2017; Zhang et al. 2015; Dragomiretskiy and Zosso 2014; Smith 2005).

According to literature studies (Yaseen and El-Shafie 2015; Yaseen et al. 2016, 2017; Wang et al. 2017; Sahoo et al. 2019; Alobaidi et al. 2020; Kasiviswanathan et al. 2016; Tongal and Booij 2018; Humphrey et al. 2016; Ni et al. 2019; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018; Zuo et al. 2020; Solomatine et al. 2008; Siddiqi et al. 2021; Niu and Feng 2021; Box and Jenkins 1970; Ghorbani et al. 2020, 2021; Yu et al. 2018; Kisi et al. 2014; Ali et al. 2020; Marques et al. 2006; Yu et al. 2017; Panigrahi and Behera 2017; Purohit et al. 2021; Zhang et al. 2015; Dragomiretskiy and Zosso 2014); to eliminate the boundary effect, robust decomposition models are proposed. The robust Local Mean Decomposition (RLMD) method is used for extraction of mixed component signals to mono-component signals called product functions and their associated demodulation signals from a time series signal (Liu et al. 2017; Smith 2005). This approach provides the most attractive feature according to other adaptive signal processing methods, such as the EMD, VMD, and DWT.

In this study, a novel hybrid model with a skilled, reliable, and efficient decomposition approach based on the RLMD method is developed to evaluate the capability of streamflow forecasting performance as well as to broaden the model's use in hydrologic time series forecasts.

The main contributions of the proposed approach are as follows:

1. The forecasting performance of the RLMD preprocessing technique was investigated. Firstly, the streamflow data have been divided into training and testing data. Then, the training and the testing data have been decomposed into subbands using the RLMD method.
2. To train the subband data, the ANN, SVR, LSTM models have been used for the realization of the forecasting model.
3. The EMD preprocessing method widely used literature has been performed for the comparison of streamflow forecasting performance with the RLMD method.
4. Also the ANN, SVR, LSTM, and hybrid Additive-ARIMA-ANN models have been applied for forecasting streamflow data and compared the performance of all models.
5. In the literature forecasting studies, time-series data are divided into training and testing parts. The training part is obtained from the first part of the data (in many studies it is taken as 70% of the data), and the testing part of the data is obtained from the remainder of the data. But when the training part of the data has a complex pattern, the testing performance occurs higher or when the training part of the data is a non-complex pattern, the testing performance occurs lower. This situation doesn't show the true performance of the model. To overcome this problem, threefold cross-validation has been performed.
6. Also, the Kruskal–Wallis test was used to determine whether the forecasted time series data met the statistical significance criteria for each model.

Application of proposed novel model for 1–3 months ahead forecasting streamflow data using RLMD method is seen in Fig. 1.

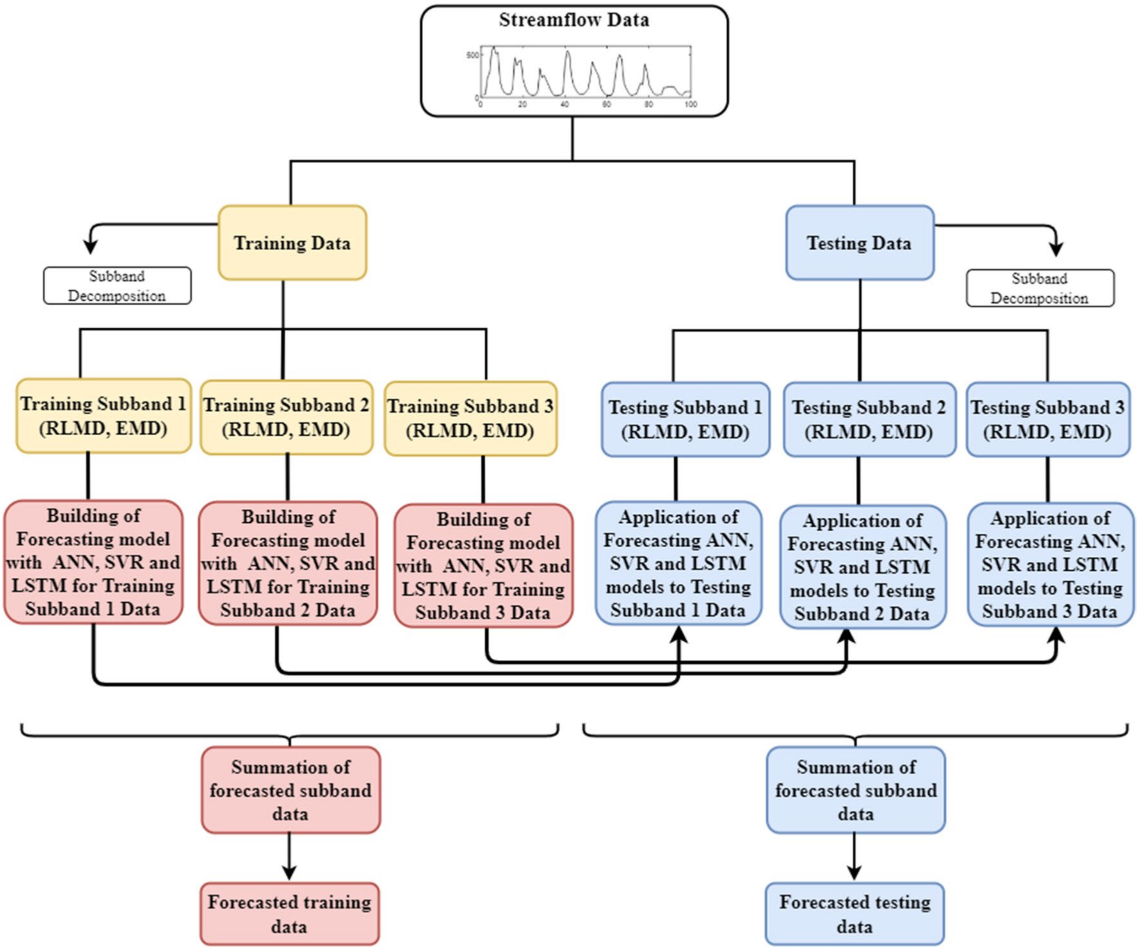

Streamflow Data
500
0
0
20
40
60
80
100
Training Data
Testing Data
Subband Decomposition
Subband Decomposition
Training Subband 1 (RLMD, EMD)
Training Subband 2 (RLMD, EMD)
Training Subband 3 (RLMD, EMD)
Testing Subband 1 (RLMD, EMD)
Testing Subband 2 (RLMD, EMD)
Testing Subband 3 (RLMD, EMD)
Building of Forecasting model with ANN, SVR and LSTM for Training Subband 1 Data
Building of Forecasting model with ANN, SVR and LSTM for Training Subband 2 Data
Building of Forecasting model with ANN, SVR and LSTM for Training Subband 3 Data
Application of Forecasting ANN, SVR and LSTM models to Testing Subband 1 Data
Application of Forecasting ANN, SVR and LSTM models to Testing Subband 2 Data
Application of Forecasting ANN, SVR and LSTM models to Testing Subband 3 Data
Summation of forecasted subband data
Summation of forecasted subband data
Forecasted training data
Forecasted testing data

Fig. 1 The proposed RLMD-based hybrid forecasting model for stream flow data

The rest of the paper is organized as follows. Section 2 gives information about the study area and the data. Section 3 provides a brief review of the RLMD and EMD decomposition methods and ANN, SVR, and LSTM approaches for streamflow estimation. Section 4 describes the estimation results obtained by the RLMD–SVR, RLMD–ANN, RLMD–LSTM, EMD–ANN, ANN, SVR, LSTM, Additive–ARIMA–ANN hybrid models using the proposed approach. Finally, Sect. 5 concludes the paper.

2 Study Area and Data

The mean monthly streamflow in (m^3/s) data have been continuously gauged over the 47 years, between 1965 through 2011, hence consisting of 564 successive numbers are used as the material of this study. The region where data have been gauged has been from the Simav River at Kocadere Station lies between 28° 24′ 21" East and 40° 15′ 51" North, in the Susurluk basin in the South Marmara Region in Turkey.

The drainage basin from which the data have been recorded is shown in Fig. 2. This data set has been obtained from Electrical Power Resources Survey Administration (EIEI), in Turkey (as seen in Fig. 3) (Smith 2005). The drainage area at this site is 21611.2 km^2.

The basin is a hydrologically rich complex due to the number of tributaries joining the river and the presence of lakes. As a result, streamflow data from this study area were employed in this study.

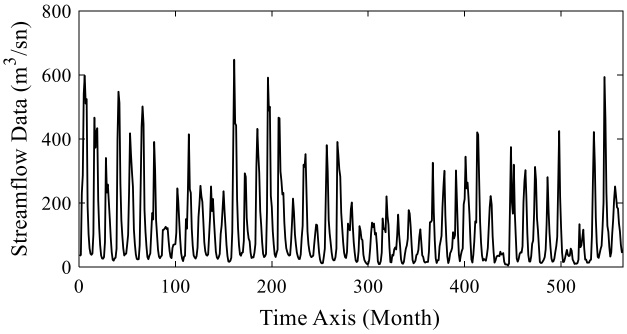


Fig. 3 Streamflow data measured on Simav River

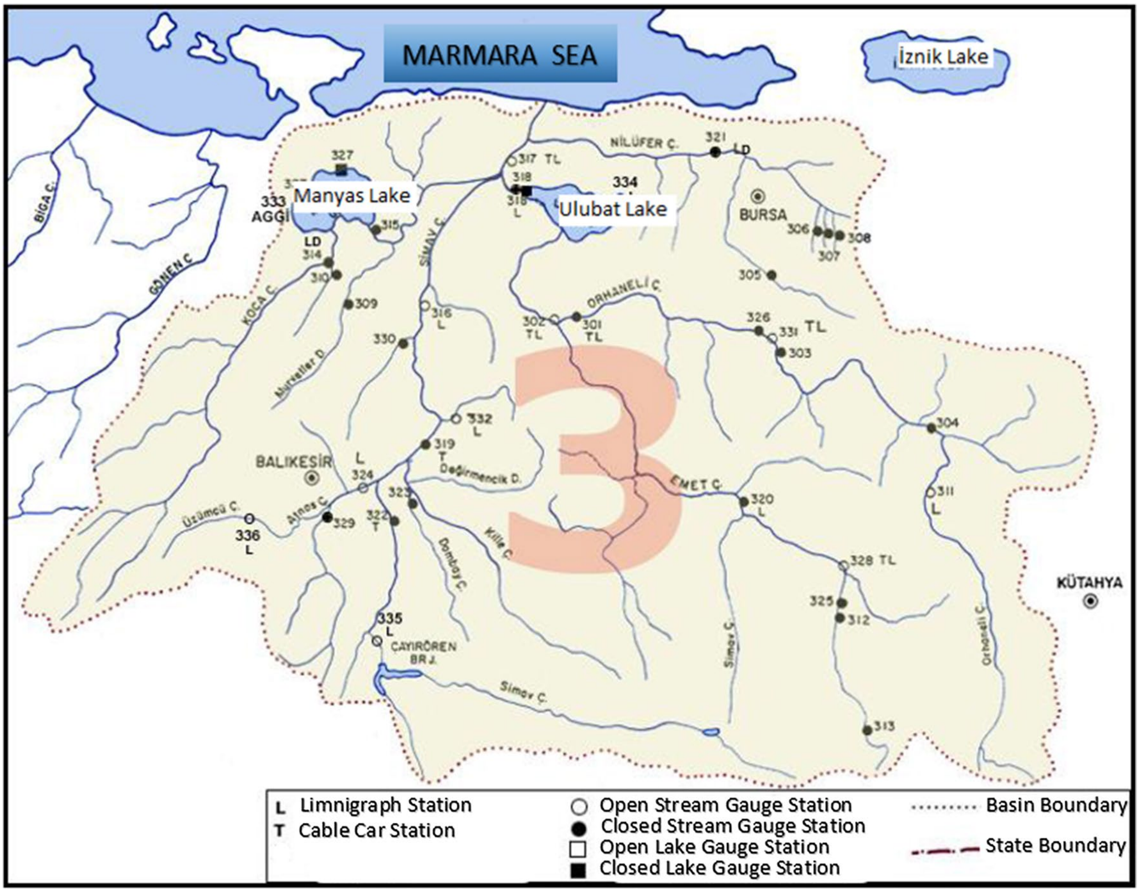


Fig. 2 Basin of Kocadere in Turkey (https://www.dsi.gov.tr/faaliyetler/akim-gozlem-yilliklari, (accession date February, 2020))

The streamflow data have been normalized at first. Nested threefold cross-validation has been performed to get the advantages of this approach for the evaluation of model performance (Hyndman and Athanasopoulos 2018; Bergmeir and Benítez 2012).

In this method, it begins with a small subset of data for training goals and then forecasts further data points. Then the accuracy for the forecasted data points is checked. After this, the same predicted data sets are included in the next training data and subsequent data points are forecasted. In this study, the data have been split into parts as seen in Fig. 4. Then three different models have been built. The first model has been trained on 282 of the data elements and tested on 94 of the following data elements. The second model has been trained on the 376 of the data elements and tested on 94 of the following data elements. The third model has been trained on the 470 of the data elements and tested on 94 of the following data elements. Model performances have computed the average of the performance parameters threefold.

In many literature studies, the number of subbands for forecasting has been defined according to the stopping criteria. But this criterion executes the decomposition (for example for the EMD technique) till the residue signal is so monotonic that has a near to zero frequency. This computation takes time and processing load.

In this study, the streamflow data have been decomposed into three subbands using EMD and RLMD methods. Although these data have been decomposed into more subbands according to the stopping criteria used in EMD and RLMD methods, it has been observed that it is sufficient to decompose these data into three subbands because when the normalized frequency spectrum obtained from the Fourier transform of the streamflow data is examined, as can be seen in the Fig. 5, there are main harmonic peaks in three fundamental frequencies, low, medium, and high frequency in the spectrum.

In addition, the forecasting performance of the models that are decomposed into more subbands during the building model has been also examined. However, according to the obtained results, it has been seen that there has been no more improvement than the forecasting performance of the data decomposed into three subbands. Thus, computer processing load and complexity are reduced.

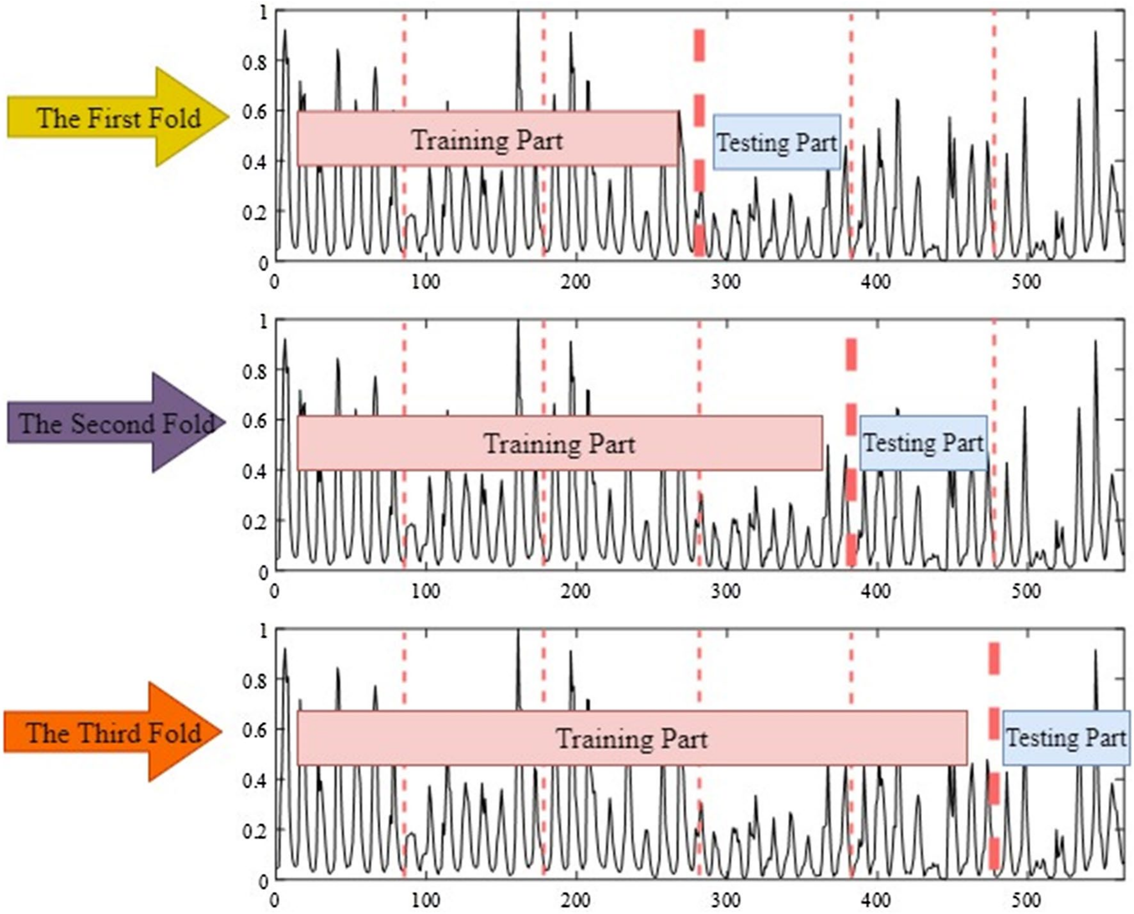


Fig. 4 Training and testing data for the proposed forecasting model for threefold

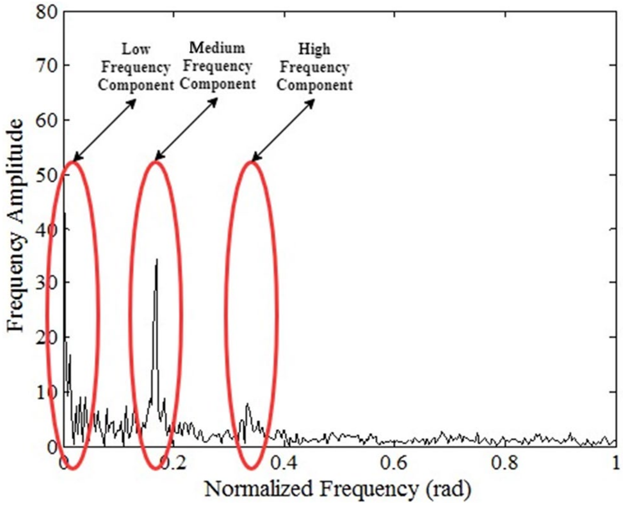


Fig. 5 Normalized Frequency spectrum obtained from Fourier Transform of streamflow data

2.1 Empirical Mode Decomposition Method

The Empirical Mode Decomposition (EMD) method is an adaptive signal decomposition technique and decomposes a signal into different simple components called Intrinsic Mode Functions (IMF). This method doesn't require any condition about stationarity and linearity of the processed data. The signal is defined as the sum of the IMF components and the residual signal, as depicted by Eq. 1 (Huang et al. 1999).

$$x(t) = \sum_{i=1}^{k} \mathrm{IMF}_i + r(t) \tag{1}$$

where k is the number of IMFs and $r(t)$ is the residue term.

The IMF of a signal must satisfy the following conditions.

- The numbers of the maximum or minimum points and zero-crossing points in the whole data set have to be equal to each other or must differ by one element only.
- The envelopes defined by local maxima at any point and envelopes defined by local minima should have an average of zero.

In this method, the signals in the time axis are treated as a combination of repetitive and original oscillations with a local mean of zero and scattered symmetrically around it.

The first step is to extract the maxima and minima or zero crossing points of *x(t)*.

In the second step, by handling both the maxima and the minima by interpolations, the upper and lower envelopes of the signals are obtained.

In the third step, the average of the upper and lower envelopes ($x_{\max}(t)$, $x_{\min}(t)$) is computed for the entire time period by Eq. 2.

$$m(t) = \left[x_{\max}(t) + x_{\min}(t)\right]/2 \tag{2}$$

In the fourth step, the average signal is subtracted from the actual signal as given by Eq. 3 below to obtain the detail signal, *d(t).*

$$d(t) = x(t) - m(t) \tag{3}$$

In the fifth step, Check whether *d(t)* is an IMF or not. The detail signal is treated as if it has been a new original signal, and the process steps are repeated until the termination criterion is reached.

In this study, for an objective achievement as high estimation accuracy as possible, on both the training and testing parts of the streamflow data, a two-level decomposition process has been performed and three subsets including two subsets of IMF and R have been obtained (it is seen in Fig. 6).

2.2 Robust Local Mode Decomposition

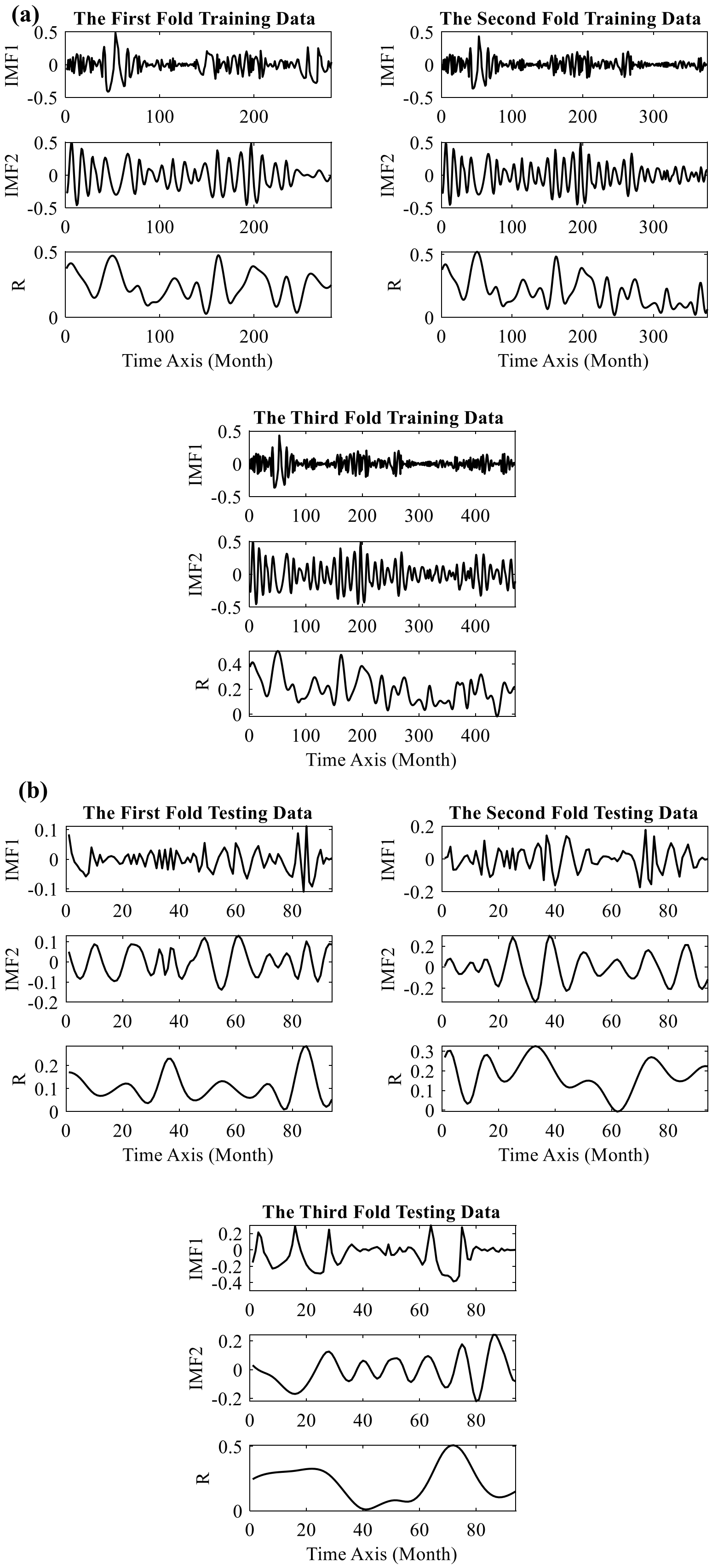
Smith pioneered local mean decomposition (LMD), which is an adaptive time–frequency signal decomposition approach for estimating the instantaneous frequency and amplitude of signals (Smith 2005). LMD is also an effective and promising method for processing multicomponent AM–FM signals. LMD converts amplitude modulated (AM) and frequency modulated (FM) signals into a series of product functions (PFs). A product of an envelope signal and an FM signal yields each of the PFs. LMD possesses a number of potential characteristics; in particular, this method doesn't require any condition about stationarity and linearity of the processed data like EMD, but according to these techniques LMD has an advantage that avoids the limitation of uncertain negative instance frequency occurring in Hilbert Transform (Liu et al. 2017; Smith 2005) because LMD calculates the instantaneous amplitude and frequency data without using HT.

The procedure of this algorithm is briefly summarized as below:

The first step is to extract the maxima and minima of *x(t)* signal. The extreme points are denoted as$e(k_i)$, where k_i=$k_1, k_2, \ldots$ is the time index of extrama and the corresponding extreme values are denoted as *x(e(k))* ($x_{\min}$, $x_{\max}$) where $k = 1, 2, 3, \ldots$.
In the second step, by handling both the maxima and the minima local mean and local magnitudes are obtained.

$$m(t) = \frac{1}{2}\left[e\left(k_i\right) + e\left(k_{i+1}\right)\right] \tag{4}$$

Fig. 6 **a.** Training subband signal decomposed by the EMD method. **b.** Testing subband signal decomposed by the EMD method



where $m(t)$ is the local mean of successive maxima and minima, $t \in (k_i, k_{i+1})$

$$a(t) = \frac{1}{2}\left[e(k_i) - e(k_{i+1})\right] \tag{5}$$

where $a(t)$ is the local magnitude of successive maxima and minima, $t \in (k_i, k_{i+1})$

In the third step, the local mean and local magnitude values between successive extrema are interpolated with straight line.

In the fourth step, smoothing the interpolated local mean and local magnitude using moving averaging (MA) approach controlled by the fixed subset size (λ^*) between successive extrama is produced to give the continuous local mean function $\widehat{m}(t)$ and amplitude function $\widehat{a}(t)$.

In the fifth step, prototype PF is obtained as given by Eq. 6

$$h(t) = x(t) - \hat{m}(t) \tag{6}$$

an FM signal is obtained as given by Eq. 7

$$s(t) = \frac{h(t)}{\hat{m}(t)} \tag{7}$$

In the sixth step, if calculated $\widehat{m}(t)$ is close to 1, then $s(t)$ is defined as purely normalized FM. If$\widehat{m}(t) \neq 1$, $h(t)$) may be considered the original signal and the above procedures should be continued until the goal signal $s(t)$ is generated (N is the final number of iterations), where the corresponding envelope function $\widehat{m}(t)$ is equal to one.

In the seventh step, instantaneous amplitude of PF is calculated as given by Eq. 8

$$a(t) = \prod \hat{a}(t) \tag{8}$$

PF is constructed given by Eq. 9

$$c(t) = a(t)s(t) \tag{9}$$

In the eighth step, residual signal is constructed in the equation as given by Eq. 10

$$r(t) = x(t) - c(t) \tag{10}$$

If residual signal $r(t)$ is monotonic (one-trend, has no frequency), then LMD terminates. Otherwise, steps 1–8 repeat on $r(t)$.

LMD has the benefit of not allowing for negative frequency in the decomposed subseries. The edge effect on the entire segment of data is much decreased in LMD due to the lack of a spline fitting procedure, but this technique may still suffer from end effect and mode mixing disruptions. To eliminate the disadvantages of LMD, the RLMD method is proposed (Smith 2005; Liu et al. 2017; Ren et al. 2012). The approach in the RLMD method to eliminate the disadvantages of the LMD method is as follows.

Boundary condition: The proposed robust LMD uses the mirror extension method (Smith 2005) to find symmetry locations for the left and right ends of the signal (Hagan et al. 1996; Liu et al. 2017; Ren et al. 2012).

Envelope estimation: An ideal subset size (λ^*) is determined based on statistical theory and specified in ref detailed (Smith 2005; Liu et al. 2017; Ren et al. 2012).

Sifting stopping criterion: The RLMD approach, which is discussed in (Smith 2005; Liu et al. 2017; Ren et al. 2012) defines the objective function to characterize the zero-baseline envelope signal.

In this study, on both the training and testing parts of the streamflow data, the decomposition process using the RLMD method has been performed and three subsets including PFs have been obtained (it is seen in Fig. 7).

2.3 The Method of Artificial Neural Networks

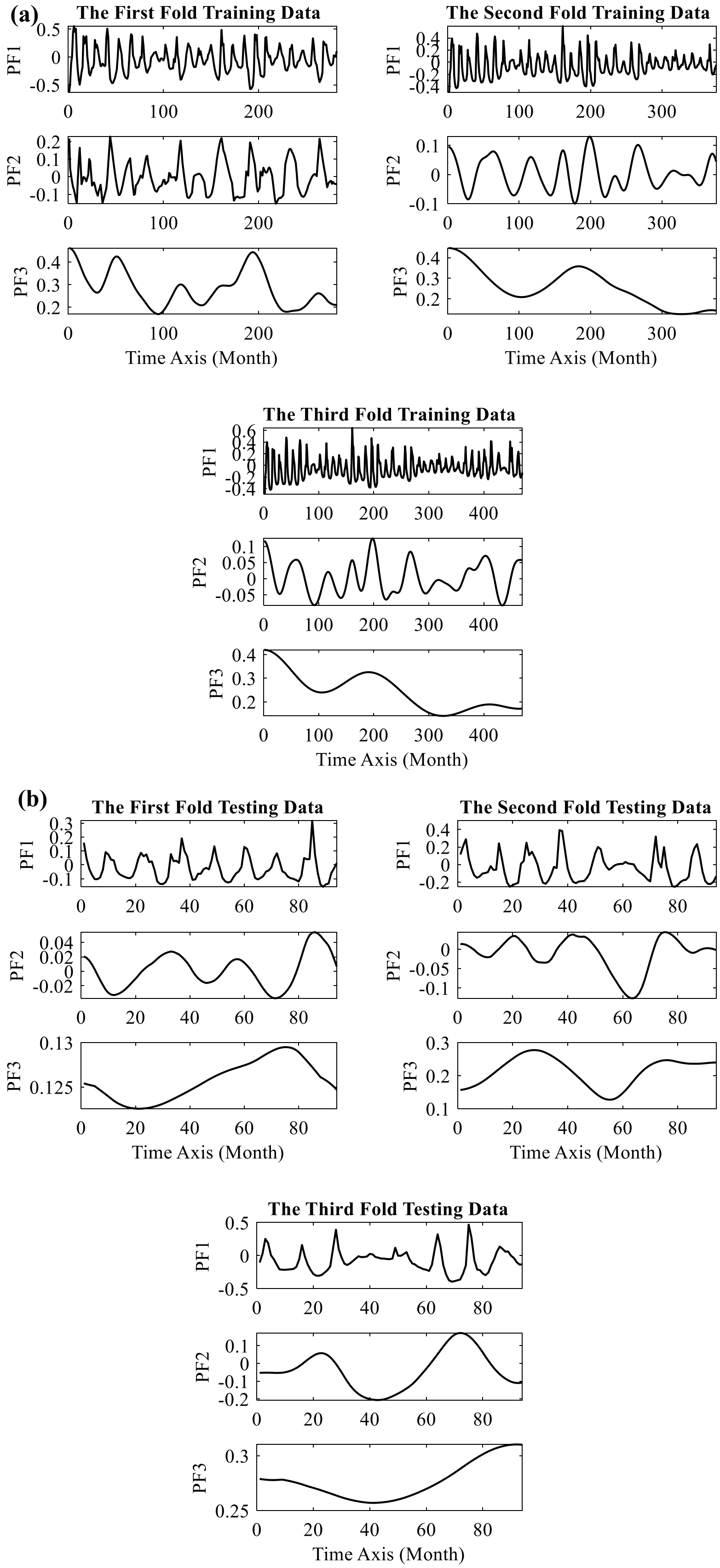
Artificial neural networks are information processing systems that are inspired by biological neural networks and share some of their performance characteristics similar to biological neural networks. Recent ANN research has revealed that it is capable of strong pattern categorization and is widely used in the estimation of data on account of its advantages like the ability to handle nonlinear structures and parallel and serial processing capability. One of the major application areas of ANNs is the forecasting task. Because as opposed to the traditional model-based methods, thanks to these features, ANNs are data-driven self-adaptive approaches and nonlinear models and therefore can produce more realistic solutions to real-life problems.

In an ANN structure, the sum and activation functions and the learning strategy of the processor elements, the learning rule used, and the topology as a result of the connection of process elements determine the model of the network. Artificial neural cells (neurons) come together and form their ANN. Neurons are not gathered randomly. Cells, in general, constitute a three-layer network in which they are arranged in parallel in each layer. These are the input, inner (hidden), and output layers (Hagan et al. 1996; Haykin 1994).

The multilayer perceptron (MLP) model commonly used form of ANN consists of an input layer, one or more hidden layers, and an output layer. The input layer's processor components act as a buffer, distributing input signals to the hidden layer's processing elements.

Each one of these elements takes the input data by multiplying them by the weight coefficients which show the efficiency of the input over the hidden neuron into the sum function. Next, these sum functions are passed through a

Fig. 7 **a** Training subband signal decomposed by the RLMD method. **b** Testing subband signal decomposed by the RLMD method



transfer function and the output value of that neuron is calculated as defined below equation:

$$y_j = f\left(\sum_i w_{ij} a_i + \theta\right) \tag{11}$$

where j is the number of neuron, i is the number of inputs, a_i is the input signal, w_{ij} is the weight coefficient, and θ is the bias term (or threshold).

These operations are repeated for all of the processors in this layer. The processor elements in the output layer also act as interlayer elements, and the network output values are calculated.

The output of neurons in the output layer is calculated in a similar way. The sum of squared differences between the desired and actual values of the output neurons E may be calculated using the equation below.

$$E(w) = \sum \left(y_{dj} - y_i\right)^2 \tag{12}$$

where y_{dj} is the target (actual) output value.

The number of input parameters and hidden neurons in an ANN model have a significant impact on the model's predicting ability. However, only a little number of study has been done on the best architecture for the ANN model. As a result, in this study, from one to twelve numbers of neurons in the hidden layer using 'for' loop in Matlab program have been set in the architecture of the ANN, and the optimum architecture of the model has been determined based on the giving minimum mean square error between forecasted and streamflow data for training performance. Also, determination of the number of the hidden layer is very important not to have too many hidden layers to avoid overfitting. Therefore, to avoid overfitting, the number of hidden layers is defined as one in this study.

The information flow in the forward direction in the MLP model. Because of this feature of the MLP model, it is also known as feed-forward ANN. One of the advantages of this model is to use different learning algorithms to train the network. According to the training algorithm used, the weights of the network are changed until the error between the output of the network and the desired output is minimized. Backpropagation Neural Network is the term given to the MLP model when it is supervised by a learning algorithm (BPNN). The most often used ANN model in time series estimation is the BPNN, which is a feed-forward network. In our study, the BPNN feed-forward network structure has been used.

Because the purpose of the network is to provide an output for each input, MLP–ANN is based on a supervised learning technique. The MLP–ANN learning takes place in two stages. At first, in the forward calculation stage, the output of the network is computed. In the second stage, the backward calculation stage, the weights are calculated according to the difference between the expected output and the output of the network. The learning rule of MLP–ANN is called the backpropagated MLP learning rule because it is carried out in this way by backpropagating. Different learning methods are utilized to train the network in the ANN. For the forecasting of streamflow data, the Levenberg–Marquardt (LM) learning method is utilized, which has a computational speed performance advantage in the ANN. The LM algorithm is similar to the Quasi-Newton method based on the least-squares calculation and the maximum neighborhood approach. LM has second-degree training speed without the use of a Hessian matrix (Marquardt 1963). The weights are updated according to the following formula:

$$\begin{gathered} w_{ij}(t+1) = w_{ij}(t) + w_{ij}(t) \\ w_{ij} = \left[J^T(w)J(w) + \mu I\right]^{-1} J^T(w)E(w) \end{gathered} \tag{13}$$

where J is the Jacobian matrix, I is an identity matrix, μ is a constant, and $E(w)$ is the error function calculated with the equation (Marquardt 1963).

2.4 The Method of Support Vector Regression

Support vector machine (SVM) analysis is a common machine learning method and used for classification and regression problems. The statistical learning theory for support vector regression was first established by Vladimir Vapnik et al. (1995). Support Vector Regression is the use of SVMs in regression (SVR). Because it uses kernel functions, SVR is a nonparametric approach that may balance the trade-off between lowering empirical error and the complexity of the resulting fitted function, reducing the possibility of overfitting. SVR has recently been widely used in forecasting studies leading to good predictive results in time series analysis.

In the SVR method, it is tried to minimize the error rate by keeping the error in a regression within a certain threshold value with hyperplanes. Let's assume that the data set meets the following conditions.

$$\begin{gathered} \left(x_1, y_1\right), \left(x_2, y_2\right), \ldots, \left(x_i, y_i\right), x \in R^D, y \in R \\ f(x) = w.x_i + b \end{gathered} \tag{14}$$

In this equation, x_i represents the input vector of D dimensional space, y_i denotes the output vectors that correspond to these input vectors, whereas w denotes the hyperplane's normal, as well as the weight vector and b the deflection.

In linear support vector regression, it is assumed that there is a linear relationship between x_i and y_i. The goal is to create a function *f*(*x*) that can calculate the predicted value

y_i within a predetermined distance $\in$ (error tolerance) using the actual value x_i, which is each training input data. Errors are ignored in the regression procedure as long as they are less than $\in$, but any deviation more than is not accepted.

A convex optimization problem is defined by Eq. 14.

The optimum regression function can be found by minimizing the expression $\frac{1}{2}\|w\|^2$, considering the assumption in Eq. 15

$$\left|y_i - \left(w.x_i + b\right)\right| \leq \in \tag{15}$$

It's impossible to find a function *f(x)* that meets this constraint for all data. The elasticity variables ξ^+ and ξ^- are utilized for each point ($\xi^+ \geq 0$, $\xi^- \geq 0$) to eliminate this problem.

$$\begin{aligned}
&y_i - \left(w.x_i + b\right) \leq \in + \xi^+ \\
&\left(w.x_i + b\right) - y_i \leq \in + \xi^- \\
&f(x) = C\sum_{i=1}^{L}\left(\xi^+ + \xi^-\right) + \text{minimize}\tfrac{1}{2}w^2
\end{aligned} \tag{16}$$

where C is a constant value which has an effect of penalty loss when an error occurs during the training and the value of it is bigger than zero. Using the Lagrange multiplier to minimize the error function subject to constraints, the following equation is obtained.

$$L_p = C\sum_{i=1}^{L}\left(\xi^+ + \xi^-\right) + \frac{1}{2}w^2 - \sum_{i=1}^{L}\left(\mu_i^+\xi^+ + \mu_i^-\xi^-\right) \tag{17}$$

$$-\sum_{i=1}^{L}\alpha_i^+\left(\in + \xi^+ + y_i - f\left(x_i\right)\right) - \sum_{i=1}^{L}\alpha_i^-\left(\in + \xi^- - y_i - f\left(x_i\right)\right)$$

In this equation for $\forall_i, \alpha_i^- \geq 0, \alpha_i^+ \geq 0, \mu_i^- \geq 0, \mu_i^+ \geq 0$

For the optimal solution, the partial derivative of Lp with respect to the variable w, b, ξ^+ and ξ^- is performed.

$$\frac{\partial L_p}{\partial w} = 0, \frac{\partial L_p}{\partial b} = 0, \frac{\partial L_p}{\partial \xi^+} = 0, \frac{\partial L_p}{\partial \xi^-} = 0 \tag{18}$$

According to obtained results from Eq. 18, Lp is maximized with respect to α_i^+ and α_i^-.and the prediction function is obtained as below equation:

$$\begin{aligned}
&f(x) = \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)x_i x + b \\
&b = f\left(x_s\right) - \in - \sum_{m \in S}^{L}\left(\alpha_m^+ - \alpha_m^-\right)x_m x_s
\end{aligned} \tag{19}$$

For indices i matching the criterion $0 \leq \alpha \leq C$ and $\xi^+ = 0$ or $\xi^- = 0$, S support vectors exist.

The same stages are repeated with the classifier that cannot be separated linearly in nonlinear regression, and the solution is achieved similarly to linear regression Vapnik 1995; Fan et al. 2005, 2006; Huang et al. 2006). Moving data to the property space or utilizing the kernel function give the answer. In Eq. 14, the nonlinear kernel function $K(x_i, x_j) = \varphi(x_i)\varphi(x_j)$ is replaced by the dot product $x\ i.x\ j$ to achieve nonlinear regression.

Thus, the forecasting function can be written as follows:

$$f(x) = \sum_{i=1}^{L}\left(\alpha_i^+ - \alpha_i^-\right)K\left(x_i, x\right) + b \tag{20}$$

In the proposed study, training data are used for the building of support vector regression model. During the building of the model, Radial basis kernel function is used. Different kernel functions have been tried for the forecasting model. The best performance has been obtained by the SVR model with a radial basis kernel function.

2.5 The Method of Long Short-Term Memory Networks

Long short-term memory networks (LSTMNs) are a type of Recurrent Neural Network (RNN) that can learn long-term dependencies. This model, first proposed by Hochreiter & Schmidhuber (1997), is widely used today. The LSTM model was created to solve long-term dependence concerns by capturing nonlinear data patterns and recalling knowledge from the past (Hochreiter and Schmidhuber 1997).

Therefore, LSTM has been successfully applied to a large number of time series problems. As shown in Fig. 8, there are three types of gates in the LSTM structure, including input, forget and output. The '*x*' and '+' symbols in the model represent addition and multiplication operations. The arrow points in the direction of the information flow. The memory structure's initial layer determines to delete unnecessary data from the cell state.

The number of forgetting gates determines how much information is lost and how much is carried on to the next level. The sigmoid function is used for this operation, which returns a number between 0 and 1. 0 indicating that no information will be transmitted, whereas 1 indicates that all information must be transmitted. The following step is to decide what data should be saved. The sigmoid function at the entrance gate is used to accomplish this. The "tan*h*" function then creates a C_y valued vector of candidate values. After then, the two procedures are integrated. Following this, the memory cell's new state information is computed. Finally, the system's output is computed. These operations can be mathematically stated as:

$$f_t = \sigma\left(W_f\left[h_{t-1}, x_t\right] + b_f\right) \tag{21}$$

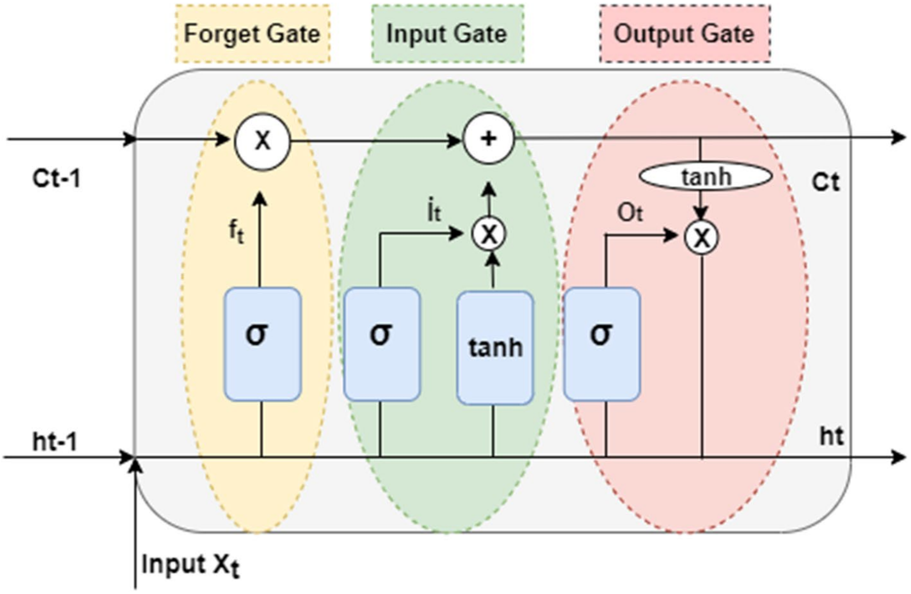


Fig. 8 LSTM Unit

$$i_t = \sigma\left(W_i \cdot \left[h_{t-1}, x_t\right] + b_i\right) \quad (22)$$

As demonstrated in Eq. 21, ft is the forget gate's output.

$$C_y = \tanh\left(W_c \cdot \left[h_{t-1}, x_t\right] + b_c\right) \quad (23)$$

$$C_t = f_t * C_{t-1} + i_t * C_y \quad (24)$$

C_t shows the update status in LSTM networks. Long-term memory is updated in this step by adding new information to sections of long-term memory. The update status is calculated by adding the values of the forget gate layer and the input gate layer.

$$O_t = \sigma\left(W_o \cdot \left[h_{t-1}, x_t\right] + b_o\right) \quad (25)$$

$$h_t = O_t * \tanh\left(C_t\right) \quad (26)$$

As shown in Eq. 25, the last layer of LSTM networks passes through a sigmoid layer that determines how much the cell state affects the output. The cell state is transferred via the tan$h$ activation function and multiplied by the exit gate output, as shown in Eq. 26 (Hochreiter and Schmidhuber 1997).

In this study, early stopping criteria were used to prevent overfitting for the modeling of the ANN, SVR and LSTM approaches.

2.6 Parameters Used for Performance Criteria

The statistical criteria commonly used for comparing the forecasting accuracies of the hydrologic data by various models are the mean absolute error (MAE), the mean square error (MSE), the determination coefficient (R^2,Nash–Sutcliffe (NSEC)), and the correlation coefficient (R). In the forecasting of monthly streamflow study, these parameters have been used to evaluate the performances of the ensemble models. In this study in each-fold MSE, MAE, R^2, and R parameters have been calculated during the training and testing stage. Overall training and testing performances have been calculated from the mean of MSE, MAE, R^2, and R parameters obtained during the training and testing phase of each fold.

The Mean Square Error (MSE): MSE is the arithmetic average of the squares of the differences of the observed time-series data from the forecasted by the model used and is defined by Eq. 27 below.

$$\text{MSE} = \frac{1}{N}\sum_{i=1}^{N}\left(X_{\text{observed},i} - X_{\text{estimated},i}\right)^2 \quad (27)$$

The Mean Absolute Error (MAE): MAE is the arithmetic average of the absolute differences of the observed time series data from the forecasted by the model used and is defined by Eq. 28 below.

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|X_{\text{observed},i} - X_{\text{estimated},i}\right| \quad (28)$$

Determination Coefficient (R^2, Nash–Sutcliffe Efficiency Coefficient (NSEC)): R^2 is a measure that quantitatively reflects the accuracy of the forecasted data obtained by the model because it equals 1 minus the sum of squares of differences of the observed time-series data from the forecasted by the model used divided by the sum of squares of differences of the observed data from their mean. R^2 is defined by Eq. 23 below. The second term in Eq. 23 approaches zero for a powerful model, and it tends to 1 for an insignificant model. Therefore, R^2 approaches 1 for a good model, while it tends toward zero for a weak one.

$$R^2 = 1 - \frac{\sum_{i=1}^{N}\left[\, X_{\text{observed},i} - X_{\text{estimated},i}\,\right]^2}{\sum_{i=1}^{N}\left[X_{\text{observed},i} - \text{mean of}_{X_{\text{observed},i}}\right]^2} \tag{29}$$

Correlation Coefficient (*R*): It is an indication of the degree and trend of whether there is a linear relationship between the observed and estimated time series data. R takes values between − 1 and + 1. If R is close to zero, there is no relationship between the two data sets, if it is close to + 1 there is a strong positive relationship, and if it is close to − 1, there is a negative relationship between the two data sets. R is defined by Eq. 30.

$$R = \frac{1}{N-1}\sum_{i=1}^{N}\left(\frac{X_i - \mu_X}{\sigma_X}\right)\left(\frac{Y_i - \mu_Y}{\sigma_Y}\right) \tag{30}$$

Here, μ_X and σ_X denote the mean and standard deviation of the X data set, while μ_Y and σ_Y indicate the mean and standard deviation of the Y data set.

3 Forecasting Results

3.1 Forecasting Results by the RLMD–ANN, RLMD–SVR and RLMD–LSTM model

In the first stage of the study, to evaluate the forecasting performance of streamflow data RLMD–ANN and RLMD–SVR models have been performed with five steps including data decomposition by the RLMD and forecasting by the ANN and SVR.

Step 1. The original normalized (amplitude of data is adjusted as a 0–1 range) streamflow data were split into two groups: training and testing data.

Cross-validation on a rolling basis has been used to test the data set and evaluate the model performance. The data have been split into 6 parts. In the first fold, 282 elements of the data set (1–3th parts) have been used for the training phase and 94 following values (4th parts) have been used for the testing phase. In the second fold, 376 elements of the data set (1–4th part) have been used for the training phase and 94 of the following values (5th parts) have been used for the testing phase. In the third fold, 470 elements of the data set (1–5th part) have been used for the training phase and 94 of the following values (6th parts) have been used for testing phase.

Step 2. The training and the testing data have been decomposed into three subband components defined as product functions using the RLMD and are named PF1–PF3. The effects of various RLMD decomposition levels on model efficiency have also been investigated to optimize to get the best result. The best results have been obtained for three subband decomposition levels as described in the method section.

Step 3. The ANN, SVR and LSTM models have been applied to the decomposed streamflow data for forecasting 1–3 months ahead of the PFs using one to three inputs during the training phase for the building of forecasting model.

During the design of ANN, the MLP network performed with supervised LM (Levenberg–Marquardt) learning algorithm has been designed with a single hidden layer. In the hidden layer, different neurons combination (from one–two twelve) has been tried to get the best forecasting result described in the method section with detailed. The goal of mean square error has been set to 0.0001 for stopping criteria during training of ANN.

During the design of SVR, the model selection of SVR has been concerned with two key issues as the selection of appropriate kernel function and the determination of the optimal parameters of SVR. At first, different kernel functions have been tried to evaluate the forecasting performance of the RLMD–SVR model and the best results have been obtained by the radial basis kernel function. Therefore, the radial basis kernel function has been used for this study. For the latter, the most popular strategy for solving SVM issues, the Sequential Minimal Optimization (SMO) algorithm was employed to find the best parameters (Fan et al. 2005; Platt 1999).

During the design of LSTM, in this study, the different numbers of memory units (between 32 and 256) and Deep Neural Network layers were tried while creating LSTM architecture. Accordingly, the first layer consists of the sequence input layer and the second layer consists of the LSTM layer. There are 128 memory elements in the LSTM layer. The third layer is the fully connected layer. 10–50 units were tested in this layer and 10 units were used in this study. The fourth layer is the Dropout layer. In this study, the dropout value was determined as 50%. The fifth layer is the one unit fully connected layer, and this output is applied as an input to the regression layer.

During the training of the network, the maximum epoch value was determined as 200, the initial learning rate was determined as 0.002, the learning rate drop range was determined as 100, and the drop factor was determined as 0.1 by trial and error, and Adam learning algorithm was used.

MSE, MAE, R^2, and R statistical performance parameters have been calculated during the forecasting of each PFs and forecasted streamflow data that are the sum of PFs. One to three numbers of inputs obtained from subbands of streamflow data have been performed to show forecasting performance. Four and more inputs have been tried for forecasting streamflow data. However, no further improvement in forecasting performance has been seen after three inputs.

For one input; x_t, for two inputs; x_{t-1}, x_t and for three inputs; x_{t-2}, x_{t-1}, x_t have been used to one-ahead forecast the value of x_{t+1} of time series of data. Therefore, the forecasting was performed recursively. The forecasting process is summarized in Table 1.

Step 4. Constructed forecasting model during training phase applied to testing subband PFs data.

Step 5. The forecasted streamflow data have been obtained by the summation of all the predicted PFs (PF1– PF3).

The obtained results are summarized in Tables 2, 3 and 4 for one- to three-input RLMD–ANN, RLMD–SVR, and RLMD–LSTM models for one- to three-ahead forecasting.

As seen in Tables 2 and 3, the best forecasted results have been achieved using three inputs to the RLMD–ANN model according to the performance measures. Also forecasting performance of RLMD–ANN model is better than RLMD–SVR and RLMD–LSTM models. As the number of forward forecasting steps (two- and three-ahead forecasting) increases, the performance of the RLMD–LSTM and RLMD–SVR model decreases significantly compared to the RLMD–ANN model.

The scatter plots of one month ahead forecasted streamflow data obtained from RLMD–ANN model using the one-three inputs are shown in Fig. 9.

It is clear from Fig. 10 and Tables 2 and 3 that the forecasted data closely follow the observed monthly time series obtained from the RLMD–ANN method. Also, the RLMD–ANN model has been outperformed the other RLMD–SVR and RLMD–LSTM model.

3.2 Forecasting Results for Computational Intelligence Methods and Hybrid Models

In this study to evaluate the performance of the proposed approach, ANN, SVM, and LSTM computational intelligence models, an EMD-based hybrid model and one of the statistically based hybrid time series forecasting methods Additive-ARIMA were considered.

During the design of ANN, SVM, and LSTM models, the same procedure as in the RLMD–ANN, RLMD–SVM, and RLMD–LSTM have been performed.

Also, the same structure and design procedures have been carried out in the EMD–ANN model as in the RLMD–ANN model.

In addition, the Additive-ARIMA–ANN model from the hybrid approaches was obtained as stated in References (Panigrahi and Behera 2017; Zhang et al. 2015), and the performance of the method was compared with the RLMD–ANN model. The average of one-input and one-ahead forecasting performance of three folds is shown in Table 4.

The Kruskal–Wallis test was applied to test whether the forecasted data using the proposed model has similar statistical properties to the observed data. Table 5 presents the test results (p values) for the Kruskal–Wallis test. The null hypothesis in which there is no statistically significant difference between observed and forecasted data should be accepted if p values are above 0.05.

As can be seen from the table, all p values calculated from ANN, SVR, LSTM, EMD–ANN, Additive–ARIMA–ANN, RLMD–ANN, RLMD–SVR, and RLMD–LSTM models are above 0.05. It has been observed that there is no statistically significant difference between observed and forecasted data for all models.

4 Discussion and Conclusion

When the literature studies are examined, many studies forecast streamflow data using machine learning algorithms and hybrid approaches (Yaseen and El-Shafie 2015; Yaseen et al. 2016, 2017; Wang et al. 2017; Sahoo et al. 2019; Alobaidi et al. 2020; Kasiviswanathan et al. 2016; Tongal and Booij 2018; Humphrey et al. 2016; Ni et al. 2019; Freire et al. 2019; Nourani et al. 2017; Hadi and Tombul 2018; Zuo et al. 2020; Solomatine et al. 2008; Siddiqi et al. 2021; Niu and Feng. 2021; Ghorbani et al. 2020, 2021; Apaydin and Sibtain 2021; Chu et al. 2021; Cheng et al. 2020).

Table 1 One- to three-ahead forecasting process

One- to three-ahead forecast		
One-ahead forecast	Two-ahead forecast	Three-ahead forecast
One input		
$\hat{x}(t) = f(x(t-1))$	$\hat{x}(t) = f(x(t-2))$	$\hat{x}(t) = f(x(t-3))$
Two inputs		
$\hat{x}(t) = f(x(t-1), x(t-2))$	$\hat{x}(t) = f(x(t-2), x(t-3))$	$\hat{x}(t) = f(x(t-3), x(t-4))$

Table 2 Testing performance of RLMD–ANN, RLMD–SVR and RLMD–LSTM models for one input

	RLMD–ANN Model				RLMD–SVR Model				RLMD–LSTM Model			
	First fold	Second fold	Third fold	Mean of folds	First fold	Second fold	Third fold	Mean of folds	First fold	Second fold	Third fold	Mean of folds
One-ahead forecasting												
MSE	0.0040	0.0114	0.0129	0.0094	0.0135	0.0435	0.0506	0.0359	0.0095	0.0092	0.0144	0.0110
MAE	0.0456	0.0727	0.0765	0.0649	0.0879	0.1559	0.1600	0.1346	0.0701	0.0682	0.0830	0.0737
R^2	0.5031	0.5712	0.5772	0.5505	0.4966	0.5049	0.5568	0.5194	−0.189	0.6545	0.5282	0.3312
R	0.7251	0.7562	0.7627	0.7480	0.7159	0.7538	0.7493	0.7397	0.5820	0.8123	0.8234	0.7392
Two-ahead forecasting												
MSE	0.0066	0.0211	0.0251	0.0176	0.0098	0.0392	0.0426	0.0305	0.1197	0.1706	0.4280	0.2394
MAE	0.0634	0.1136	0.1163	0.0978	0.0753	0.1479	0.1533	0.1255	0.2910	0.3525	0.5119	0.3851
R^2	0.1713	0.1956	0.1821	0.1830	0.1188	0.0682	0.1736	0.1202	−13.962	−5.4971	−12.941	−10.8
R	0.4165	0.4494	0.4591	0.4417	0.3924	0.4317	0.4511	0.4251	0.3474	0.6364	0.7216	0.5684
Three-ahead forecasting												
MSE	0.0080	0.0240	0.0268	0.0196	0.0095	0.0309	0.0430	0.0278	0.1207	0.1754	0.4956	0.2639
MAE	0.0684	0.1192	0.1268	0.1048	0.0715	0.1374	0.1594	0.1228	0.2801	0.3684	0.5439	0.3974
R^2	0.0079	0.0617	0.1023	0.0573	0.0496	0.0873	0.1648	0.1006	−13.927	−5.8653	−15.589	−11.793
R	0.2474	0.3058	0.4203	0.3245	0.2454	0.3122	0.4172	0.3249	0.3706	0.6189	0.7130	0.5675

Table 3 Testing performance of RLMD–ANN, RLMD–SVR and RLMD–LSTM models for two inputs

	RLMD–ANN Model				RLMD–SVR Model				RLMD–LSTM Model			
	First fold	Second fold	Third fold	Mean of folds	First fold	Second fold	Third fold	Mean of folds	First fold	Second fold	Third fold	Mean of folds
One-ahead forecasting												
MSE	0.0032	0.0087	0.0062	0.0060	0.0151	0.0455	0.0548	0.0385	0.0087	0.0093	0.0184	0.0123
MAE	0.0387	0.0615	0.0563	0.0522	0.0928	0.1606	0.1655	0.1396	0.0714	0.0701	0.0903	0.0722
R^2	0.6959	0.7092	0.7974	0.7342	0.6097	0.6536	0.7581	0.6738	−0.0917	0.6466	0.4015	0.3188
R	0.8181	0.8442	0.8973	0.8532	0.7835	0.8108	0.8722	0.8222	0.6667	0.8068	0.8188	0.7641
Two-ahead forecasting												
MSE	0.0046	0.0147	0.0133	0.0109	0.0131	0.0391	0.0520	0.0347	0.0188	0.0117	0.0253	0.0186
MAE	0.0503	0.0844	0.0740	0.0696	0.0844	0.1523	0.1647	0.1338	0.1134	0.0817	0.1156	0.1035
R^2	0.4352	0.4664	0.6401	0.5139	0.2751	0.4500	0.5783	0.4345	−1.3283	0.5403	0.1527	−0.2117
R	0.6644	0.6878	0.8006	0.7176	0.5321	0.6753	0.7635	0.6570	0.0871	0.7495	0.7779	0.5381
Three-ahead forecasting												
MSE	0.0048	0.0153	0.0201	0.0134	0.0365	0.0413	0.0454	0.0411	0.1211	0.0298	0.0384	0.0631
MAE	0.0558	0.0876	0.0952	0.0795	0.1691	0.1539	0.1623	0.1618	0.2481	0.1438	0.1438	0.1785
R^2	0.4390	0.5147	0.5710	0.5082	0.3842	0.4348	0.3622	0.3937	−13.840	−0.1580	−0.3040	−4.7673
R	0.6714	0.6857	0.7566	0.7046	0.6254	0.6642	0.6078	0.6325	0.0770	0.3927	0.7536	0.4077

Table 4 One-input–one-ahead mean of forecasting performance for each fold

Models	MSE	MAE	R^2	R
ANN	0.0363	0.1390	0.5066	0.7405
SVR	0.0107	0.0665	0.5013	0.7230
LSTM	0.0441	0.1581	−0.6282	0.4354
Additive–ARIMA–ANN	0.0130	0.0672	0.4102	0.7427
EMD–ANN	0.0102	0.0726	0.5320	0.7415
RLMD–ANN	0.0094	0.0649	05.505	0.7480

When the hybrid studies in the literature are examined, it has been tried to increase the prediction performance by using preprocessed methods such as Wavelet transform, Empirical Mode Decomposition, Additive ARIMA, etc. (Freire et al. 2019; Nourani et al. 2017; Zuo et al. 2020; Kisi et al. 2014; Yu et al. 2017; Panigrahi and Behera 2017; Ibrahim et al. 2022). According to our review of the literature, it is shown that a variety of forecasting techniques have been utilized. However, none of the forecasting techniques have been demonstrated to be superior to the others in terms of overall performance.

In this study, the effect of the RLMD model on the prediction performance was analyzed and the forecasting performance of the RLMD preprocessed hybrid models was

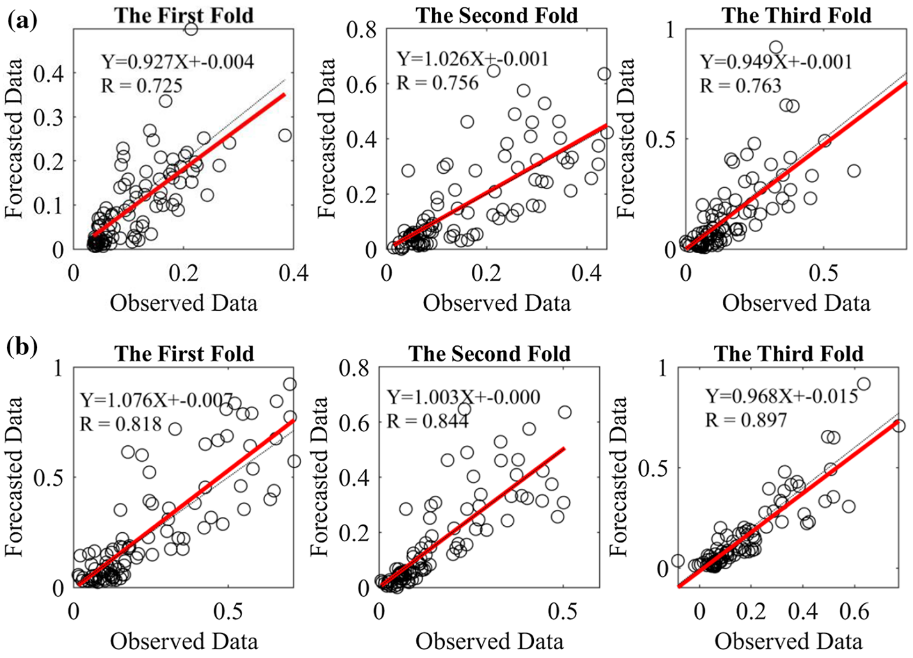


Fig. 9 Scatter plots of one-ahead forecasting with RLMD–ANN model for **a** one input and **b** two inputs

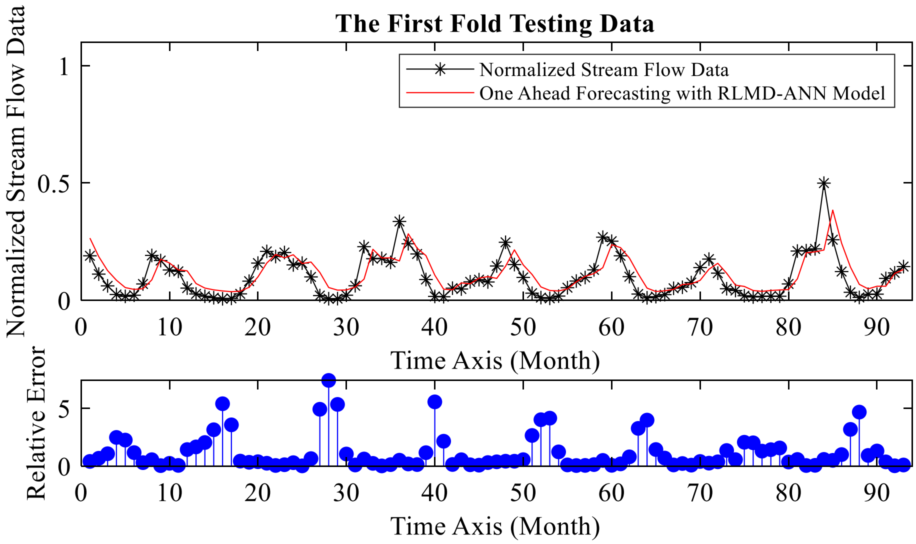


Fig. 10 One-ahead forecasted data obtained from one-input RLMD–ANN model with Relative Error chart calculated with $\mathrm{RelativeError} = \left|\frac{\mathrm{forecasteddata}-\mathrm{realdata}}{\mathrm{realdata}}\right|$ formula

Table 5 *p* values of Kruskal–Wallis test at 95% significance level for one-ahead forecast for all folds

Models	*p* value	*H_0	Models	*p* value	*H_0	Models	*p* value	*H_0
The first fold			The second fold			The third fold		
ANN	0.3514	Reject	ANN	0.5209	Reject	ANN	0.4511	Reject
SVR	0.3872	Reject	SVR	0.4212	Reject	SVR	0.387	Reject
LSTM	0.4222	Reject	LSTM	0.4743	Reject	LSTM	0.4567	Reject
EMD–ANN	0.3706	Reject	EMD–ANN	0.4355	Reject	EMD–ANN	0.4526	Reject
Additive–ARIMA–ANN	0.3387	Reject	Additive–ARIMA–ANN	0.4870	Reject	Additive–ARIMA–ANN	0.4519	Reject
RLMD–ANN	0.3549	Reject	RLMD–ANN	0.5061	Reject	RLMD–ANN	0.4538	Reject
RLMD–SVR	0.3504	Reject	RLMD–SVR	0.5055	Reject	RLMD–SVR	0.4515	Reject
RLMD–LSTM	0.4465	Reject	RLMD–LSTM	0.4511	Reject	RLMD–LSTM	0.4530	Reject

*H_0 represents a hypothesis based on the significant difference between mean predicted and observed values

compared with the ANN, SVR, LSTM, the EMD based hybrid, and the Additive-ARIMA-based hybrid models. Here, MSE, MAE, R^2, and R performance parameters were used to investigate the efficiency of applied models.

The obtained performance parameters shown in Tables 2 and 3 indicate that the forecasting performance of the RLMD-based ANN model is much better than RLMD-based SVR and LSTM models for one to two-ahead forecasting. For example, while the MSE, MAE, R^2, and R values have been obtained as 0.0060, 0.0522, 0.7342, and 0.8532 in the RLMD–ANN method for two-input and one-ahead forecasting, it has been obtained as 0.0385, 0.1396, 0.6738, and 0.8222, respectively, in the RLMD–SVR method and 0.0088, 0.0695, −0.0826 and 0.6118, respectively, in the RLMD–LSTM method. Also, the MSE, MAE, R^2, and R values have been obtained as 0.0094, 0.0649, 0.5505, and 0.7480 in the RLMD–ANN method for one-input–one-ahead forecasting, and it has been obtained as 0.0359, 0.1346, 0.5194, and 0.7397 in the RLMD–SVR method and 0.0110, 0.0737, 0.3312 and 0.7392, respectively, in the RLMD–LSTM method. In three folds and different input–output combinations, the obtained MSE and MAE values are lower in the RLMD–ANN model than in the RLMD–SVR model, and the obtained R^2 and R values are bigger than in RLMD–SVR and RLMD–LSTM models.

The performance of the RLMD-based ANN model has been slightly better than EMD-based ANN model for one-ahead forecasting in one-input model. While the MSE, MAE, R^2, and R values have been 0.0094, 0.0649, 0.5505, and 0.7480 in the RLMD–ANN method for one-input and one-ahead forecasting, it has been obtained as 0.0102, 0.0726, 0.5310, and 0.7415, respectively, in the EMD–ANN method.

In addition, the MSE, MAE, R^2, and R performance results for one-input and one-ahead forecasting were obtained as 0.0363, 0.1390, 0.5066, and 0.7405 for ANN, 0.0107, 0.0665, 0.5013, and 0.7230 for SVR, 0.0441, 0.1581, −0.6282, and 0.4354 for LSTM, 0.0130, 0.0672, 0.4102 and 0.7427 for Additive-ARIMA–ANN models, respectively, as seen in Table 4. When these results are examined, it is seen that the performance of the RLMD–ANN model is significantly better than the LSTM model and it is better than other models.

The Kruskal–Wallis method was used for the analysis of whether the forecasted and observed time series had similar distributions, and it was seen that the forecasted data obtained with all the applied models in this study were statistically similar to the observed data as seen in Table 5.

The LSTM approach took a much longer time than the ANN and SVR procedures in terms of computational time for the computational intelligence techniques employed in this study. In addition, the SVR algorithm required longer on average time than the ANN technique. The computational complexity of most machine learning models, such as LSTM, SVR, and ANN models, increased as the value and number of employed model parameters increased. The three machine learning models may be ranked in terms of general modeling performance in this study as follows: ANN, SVR, and LSTM, based on the aforementioned findings.

The findings show that the RLMD–ANN model can predict the streamflow data within acceptable limits. The suggested model can provide a variety of development and application methods with various hydrological data.

A new hybrid decomposition model called RLMD–ANN is proposed in this study, where the streamflow data have been decomposed into subbands by RLMD methods. The main goal is to obtain better forecasting performance using the skilled, robust, efficient, and reliable model. For this aim, firstly, threefold cross-validation has been applied for the training and the testing data. The testing data are completely unused during the training stage and development of the model. Secondly, the optimal number of subband levels has been searched by analyzing the Fourier domain of the data and by using stopping criteria. Also, to eliminate the disadvantages (end effect, boundary effect) of decomposition methods like EMD, DWT, etc., the RLMD method has been proposed for decomposition of the data to obtain the hidden structure of streamflow data.

In summary, the RLMD–ANN model offers the important advantage of avoiding using future streamflow information. By using the optimal number of subbands, the RLMD–ANN model saves time and computation resources. Numerical results demonstrate that the optimal decomposition ensemble model, RLMD–ANN, can be an important tool for forecasting highly nonstationary, nonlinear streamflow data. This model can be used reliably not only for the prediction of streamflow data but also for the prediction of other time series such as rainfall, evaporation, temperature, etc.

In future studies, to increase the streamflow forecasting performance, different hydro-meteorological data can also be used in the prediction study and the prediction performance can be analyzed according to different input numbers.

In this study, estimating each subband signals separately require a computational processing load. However, in future studies, the RLMD method can be used not to obtain subband signals, but to extract features for input to model in forecasting studies.

References

Alobaidi MH, Meguid MH, Chebana F (2020) Varying-parameter modeling within ensemble architecture: application to extended streamflow forecasting. J Hydrol 582:124511.

Ali M, Prasad R, Xiang Y, Yaseen ZM (2020) Complete ensemble empirical mode decomposition hybridized with random forest and kernel ridge regression model for monthly rainfall forecasts. Jf Hydrol 584:124647

Apaydin H, Sibtain M (2021) A multivariate streamflow forecasting model by integrating improved complete ensemble empirical mode decomposition with additive noise, sample entropy, Gini index and sequence-to-sequence approaches. J Hydrol 603:126831.

Bergmeir C, Benítez JM (2012) On the use of cross-validation for time series predictor evaluation. Inf Sci 191:192–213

Box GEP, Jenkins GM (1970) Time series analysis: forecasting and control. Holden-Day, San Francisco

Cheng M, Fang F, Kinouchi T, Navon IM, Pain CC (2020). Long leadtime daily and monthly streamflow forecasting using machine learning methods. J Hydrol 590:125376.

Chu H, Wei J, Wu W, Jiang Y, Chu Q, Meng X (2021) A classification-based deep belief networks model framework for daily streamflow forecasting. J Hydrol 595:125967.

Dragomiretskiy K, Zosso D (2014) Variational mode decomposition. IEEE Trans Signal Process, 62.

Fan RE, Chen PH, Lin CJ (2005) Working set selection using second order information for training support vector machines. J Mach Learn Res 6:1871–1918

Fan RE, Chen PH, Lin CJ (2006) A study on SMO-type decomposition methods for support vector machines. IEEE Trans Neural Netw 17:893–908

Freire PKMM, Santos CAG, Silva GBL (2019) Analysis of the use of discrete wavelet transforms coupled with ANN for short-term streamflow forecasting. Appl Soft Comput 80:494–505

Ghorbani MA, Deo RC, Kim S, Kashani MH, Karimi V, Izadkhah M (2020) Development and evaluation of the cascade correlation neural network and the random forest models for river stage and river flow prediction in Australia. Soft Comput, pp 1–12.

Ghorbani MA, Karimi V, Ruskeepää H, Sivakumar B, Pham QB, Mohammadi F, Yasmin N (2021) Application of complex networks for monthly rainfall dynamics over central Vietnam. Stoch Env Res Risk Assess 35(3):535–548

Hadi SJ, Tombul M (2018) Monthly streamflow forecasting using continuous wavelet and multi-gene genetic programming combination. J Hydrol 561:674–687

Hagan MT, Demuth HB, Beale MH (1996) Neural network design. PWS Publishing, Boston.

Haykin S (1994) Neural networks: a comprehensive foundation. Macmillan College Publishing Company Inc., New

Hochreiter S, Schmidhuber J (1997) Long short-term memory. Neural Comput 9(8):1735–1780

Huang TM, Kecman V, Kopriva I (2006) Kernel Based Algorithms for Mining Huge Data Sets: Supervised, Semi-Supervised, and Unsupervised Learning. Springer, New York

Huang NE, Shen Z, Long SR (1999) A new view of nonlinear water waves: the Hilbert spectrum. Annu Rev Fluid Mech 31:417–457

Humphrey GB, Gibbs MS, Dandy GC, Maier HR (2016) A hybrid approach to monthly streamflow forecasting: Integrating hydrological model outputs into a Bayesian artificial neural network. J Hydrol 540:623–640

Hyndman RJ, Athanasopoulos G (2018) Forecasting: Principles and Practice. Monash University, Australia

Ibrahim KSMH, Huang YF, Ahmed AN, Koo CH, El-Shafie A (2022) A review of the hybrid artificial intelligence and optimization modelling of hydrological streamflow forecasting. Alex Eng J 61(1):279–303

Kasiviswanathan KS, He J, Sudheer KP, Tay JH (2016) Potential application of wavelet neural network ensemble to forecast streamflow for flood management. J Hydrol 536:161–173

Kisi O, Latifoğlu L, Latifoglu F (2014) Investigation of empirical mode decomposition in forecasting of hydrological time series. Water Resour Manage 28:4045–4057

Liu Z, Jina Y, Zuo MJ, Feng Z (2017) Time-frequency representation based on robust local mean decomposition for multicomponent AM-FM signal analysis. Mech Syst Signal Process 95:468–487

Marques CAF, Ferreira JA, Rocha A, Castanheira JM, Melo-Gonçalves P, Vaz N, Dias JM (2006) Singular spectrum analysis and forecasting of hydrological time series. Phys Chem Earth, Parts a/b/c 31(18):1172–1179

Marquardt DW (1963) An algorithm for least-squares estimation of nonlinear parameters. J Soc Ind Appl Math 11:431–441

Ni L, Wang D, Singh VP, Wu J, Wang Y, Tao Y, Zhang J (2019) Streamflow and rainfall forecasting by two long short-term memory-based models. J Hydrol, 124296 (in press).

Niu WJ, Feng ZK (2021) Evaluating the performances of several artificial intelligence methods in forecasting daily streamflow time series for sustainable water resources management. Sustain Cities Soc 64:102562.

Nourani V, Andalib G, Sadikoglu F (2017) Multi-station streamflow forecasting using wavelet denoising and artificial intelligence models. Procedia Computer Sci 120:617–624

Panigrahi S, Behera HS (2017) A hybrid ETS–ANN model for time series forecasting. Eng Appl Artif Intell 66:49–59

Purohit SK, Panigrahi S, Sethy PK, Behera SK (2021) Time series forecasting of price of agricultural products using hybrid methods. Appl Artificial Intell, pp 1–19

Platt J (1999) Sequential minimal optimization: a fast algorithm for training support vector machines. Technical Report MSR-TR-98–14.

Ren D, Yang S, Wu Z, Yan G (2012) Research on end effect of LMD based time-frequency analysis in rotating machinery fault diagnosis. China Mech. Eng 8:951–956

Sahoo A, Samantaray S, Ghose DK (2019) Stream flow forecasting in Mahanadi River Basin using Artificial Neural Networks. Procedia Comp Sci 157:168–174

Siddiqi TA, Ashraf S, Khan SA, Iqbal MJ (2021) Estimation of data-driven streamflow predicting models using machine learning methods. Arab J Geosci 14(11):1–9

Smith JS (2005) The local mean decomposition and its application to EEG perception data. J R Soc Interface 2:443–454

Solomatine D, See LM, Abrahart RJ (2008) Practical hydroinformatics computational intelligence and technological developments in water applications, Water Science and Technology Library.

Tongal H, Booij MJ (2018) Simulation and forecasting of streamflows using machine learning models coupled with base flow separation. J Hydrol 564:266–282

Vapnik VN (1995) The nature of statistical learning theory. Springer, New York

Wang H, Wang C, Wang Y, Gao X, Yua C (2017) Bayesian forecasting and uncertainty quantifying of stream flows using Metropolis-Hastings Markov Chain Monte Carlo algorithm. J Hydrol 549:476–483

https://www.dsi.gov.tr/faaliyetler/akim-gozlem-yillikleri. Accessed Feb 2020

Yaseen ZM, Ebtehaj I, Bonakdari H, Deod RC, Mehr AD, Mohtar WHM, Diop L, El-shafi A, Singhi VP (2017) Novel approach for streamflow forecasting using a hybrid ANFIS-FFA model. J Hydrol 554:263–276

Yaseen ZM, El-Shafie A, Othman Jaafar, Haitham Abdulmohsin Afan, Khamis Naba Sayl (2015) Artificial intelligence based models for stream-flow forecasting: 2000–2015. J Hydrol 530:829–844

Yaseen ZM, Jafar O, Deo RC et al (2016) Stream-flow forecasting using extreme learning machines: A case study in a semi-arid region in Iraq. J Hydrol 542:603–614

Yu C, Li Y, Zhang M (2017) An improved Wavelet Transform using Singular Spectrum Analysis for wind speed forecasting based on Elman Neural Network. Energy Convers Manage 148:895–904

Yu X, Zhang X, Qin H (2018) A data-driven model based on Fourier transform and support vector regression for monthly reservoir inflow forecasting. J Hydro-environment Res 18:12–24

Zhang X, Peng Y, Zhang C, Wang B (2015) Are hybrid models integrated with data preprocessing techniques suitable for monthly streamflow forecasting? Some experiment evidences. J Hydrol 30(2015):137–152

Zuo G, Luo J, Ni W, Lian Y, He X (2020) Decomposition ensemble model based on variational mode decomposition and long short-term memory for streamflow forecasting. J Hydrol 585:124776