



Wavelet kernel least square twin support vector regression for wind speed prediction

Barenya Bikash Hazarika^{1,2} · Deepak Gupta¹ · Narayanan Natarajan³

Received: 14 May 2021 / Accepted: 10 January 2022 / Published online: 24 January 2022
© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2022

Abstract

Wind energy is a powerful yet freely available renewable energy. It is crucial to predict the wind speed (WS) accurately to make a precise prediction of wind power at wind power generating stations. Generally, the WS data is non-stationary and wavelets have the capacity to deal with such non-stationarity in datasets. While several machine learning models have been adopted for prediction of WS, the prediction capability of primal least square support vector regression (PLSTSVR) for the same has never been tested to the best of our knowledge. Therefore, in this work, wavelet kernel-based LSTSVR models are proposed for WS prediction, namely, Morlet wavelet kernel LSTSVR and Mexican hat wavelet kernel LSTSVR. Hourly WS data is gathered from four different stations, namely, Chennai, Madurai, Salem and Tirunelveli in Tamil Nadu, India. The proposed models' performance is assessed using root mean square, mean absolute, symmetric mean absolute percentage, mean absolute scaled error and R^2 . The proposed models' results are compared to those of twin support vector regression (TSVR), PLSTSVR and large-margin distribution machine-based regression (LDMR). The performance of the proposed models is superior to other models based on the results of the performance indicators.

Keywords Wind speed prediction · Wavelet · Kernel · Twin support vector regression · Primal least squares

Introduction

Renewable energy sources are progressively made use of because of the amazingly booming contamination degrees in the air, water and soil. Wind energy has ended up being the prime focus for energy developers because of the

accessibility of megawatt size wind equipment, environmental friendliness, low cost of maintenance, ease of availability, etc. Moreover, wind resource is a tidy, limitless and free resource. This resource has offered mankind for numerous centuries in driving the wind turbines and pumping water (Bakhsh et al. 1985). To utilize this resource efficiently, prediction of wind speed (WS) plays a critical role. This is necessary for site planning, performance analysis and deciding the optimal size of the wind turbine. Rehman and Hala-wani (1994) predicted wind speed using the autoregressive moving average method. Recently, machine learning (ML) models such as artificial neural network (ANN), support vector machine (SVM), Gaussian process regression (GPR), fuzzy logic (FL) and extreme learning machine (ELM) have been extensively used for this purpose. Selection of the most appropriate ML technique is very important for obtaining accurate results. Apart from these ML techniques, researchers are nowadays focused on the development of hybrid ML tools to achieve accurate WS forecasting (Natarajan and Nachimuthu 2019). Furthermore, hybrid ML has the advantage of both algorithms.

Salcedo-Sanz et al. (2011) tried to estimate the short wind speed using an evolutionary SVR model. Wang et al.

Responsible Editor: Philippe Garrigues

✉ Deepak Gupta
deepak@nitap.ac.in

Barenya Bikash Hazarika
barenya1431@gmail.com

Narayanan Natarajan
itsrajan2002@yahoo.co.in

¹ Department of Computer Science and Engineering, National Institute of Technology Arunachal Pradesh, Yupia, Papum Pare 791112, Arunachal Pradesh, India

² Department of Computer Science and Engineering, KL University, Vijayawada 522502, Andhra Pradesh, India

³ Department of Civil Engineering, Dr. Mahalingam College of Engineering and Technology, Pollachi 642003, Tamil Nadu, India

(2015) forecasted the WS using support vector regression (SVR) optimized by the cuckoo search optimization algorithm. Wang et al. (2016) utilized the SVM whose parameter search was optimized using a novel steepest descent Cuckoo search algorithm for the dataset which was pre-processed using the empirical mode decomposition. Khosravi et al. (2018) forecasted the WS and wind direction with the help of multi-layer feed-forward ANN, SVM with radial basis function (RBF) and adaptive neuro-fuzzy inference system (ANFIS) optimized using a swarm optimization (SO) called particle SO algorithm. Mi et al. (2019) developed a new WS multi-step forecasting framework based on the singular spectrum analysis, empirical mode decomposition and convolution SVM. Wu and Lin (2019) forecasted the WS based on LS-SVM optimized using the bat optimization algorithm. They applied variation mode decomposition to disintegrate the original WS series into separate sub-series with different frequencies. Fu et al. (2019) coupled SVM with improved chicken SO algorithm for the prediction of WS, and the results were compared with SVM-CSO. Xiang et al. (2019) forecasted WS based on the hybrid of improved empirical wavelet transform and LS-SVM. Li et al. (2020) performed wind power prediction based on SVM with improved dragonfly algorithm. Vinothkumar and Deeba (2020) used a long short-term memory network model and variants of the SVMs for forecasting the WS in the neighbourhood of windmills. Gupta et al. (2021) adopted convolution long-term short memory (LSTM) for the short-term prediction of wind power density for five stations in Tamil Nadu.

In addition to the above discussed hybrid SVM models, wavelet transform has been used in combination with SVM for the forecasting the WS of a location. There are two types of studies where the wavelet transform (WT) has been adopted. In some of the earlier studies, wavelet has been adopted for the processing of the WS data by considering the same as a signal, while in some other cases, a wavelet-based kernel has been used in the SVM model. Some of the studies which fall under the former category have been performed by Liu et al. (2014), Sangita and Deshmukh (2011), Sun et al. (2013), Sivanagaraja et al. (2014) and Prasetyowati et al. (2019). While most of them have considered the only SVM for the former case, Dhiman et al. (2019) have adopted not only SVM but also its variants. They analysed the efficiency of a hybrid model consisting of WT and several variants of SVR like ϵ -SVR, least square SVR, twin SVR (TSVR) and ϵ -TSVR for the prediction of WS at a farm in Spain. Only a few studies about the latter case have been performed for WS prediction. Zeng and Qiao (2012) performed wind power prediction using a novel wavelet SVM that changes between the RBF kernel and a Mexican hat kernel. They established that the wavelet SVM outperforms the radial basis function kernel and Mexican hat kernels. He

and Xu (2019) predicted the WS of Ningxia using a combination of a wavelet as well as polynomial kernel function and concluded that the combined kernel has better accuracy compared to the single kernel function. A few very recent prominent works are portrayed in Table 1.

In this study, the major contributions are as follows:

1. Motivated by the contribution of Zhang et al. (2004) and Ding et al. (2014), two novel wavelet-based models, namely, Morlet wavelet kernel LSTSVR (MKLSTSVR) and Mexican hat LSTSVR (MHKLSTSVR) models, are proposed for WS prediction.
2. Inspired by the PLSTSVR model, the optimization problem of the proposed MKLSTSVR and MHKLSTSVR is solved in primal, which always shows better performance compared to solving them in duals (Gupta 2017).
3. To expand the kernel selection range of the PLSTSVR model and to improve its generalization ability, the wavelet kernels are embedded in the PLSTSVR model.

The models have been applied to the WS data collected from four locations in the state of Tamil Nadu, India, and the performance of these models has been compared with other ML models such as TSVR, primal least square twin support vector regression (PLSTSVR) and large-margin distribution machine-based regression (LDMR). The rest of this paper is organized as follows: the “[Related works](#)” section briefly describes the related works. The “[Proposed models](#)” section describes the wavelet analysis briefly and also proposes the two novel models. Dataset description and area of study is presented in the “[Dataset description](#)” section. Experimental and numerical analyses are presented in the “[Experimental setup and numerical analysis](#)” section; statistical analyses are presented in the “[Statistical analysis using Friedman test with post hoc analysis](#)” section. The “[Conclusion](#)” section describes the conclusion.

Related works

Consider that m is the total number of samples. The total training samples can be represented as $\{x_i\}_{i=1}^m \in R^n$. $x_i \in R^n$ and $y_i \in R^N$ indicate the input training points and original output, respectively. Also, let us consider that $G \in R^{m \times n}$ are the input samples. Here the i th row vector can be represented as $x_i^t y = (y_1, \dots, y_m)$ denotes the observed data points.

The TSVR model

TSVR (Peng 2010) seeks for two non-parallel functions termed as ϵ -insensitive down-bound $f_1(x) = K(x^t, D^t)w_1 + b_1$

Table 1 List of a few recent contributions for wind speed prediction

Sl no	Reference	Model/models proposed	Country/countries	Term
1	Mi et al. (2019)	Singular spectrum analysis, empirical mode decomposition and convolution neural network-SVM	China	Short term
2	Zhang et al. (2019)	LSTM and Gaussian process regression (GPR)	China	Short term
3	Tian et al. (2019)	Improved PSO optimized error minimized extreme learning machine (EM-ELM)	China	Short term
4	Jamil and Zeeshan (2019)	ANN-based chaotic approach	India	Short term and long term
5	Qolipour et al. (2019)	Grey-ELM	Iran	Long term
6	Jha and Bilalovikj (2019)	ANN	North Macedonia	Very short term
7	Harbola and Coors (2019)	1-dimensional-CNN	Germany and Netherlands	Short term
8	Jamil and Zeeshan (2019)	ANN and chaotic approach	India	Short term and long term
9	Vinothkumar and Deeba (2020)	LSTM and SVM	India	Short term
10	Zhang et al. (2020)	GA-ANN	China	Short term
11	Tian (2020)	Local mean decomposition and improved FA optimized least square SVM	China	Short term
12	Blanchard and Samanta (2020)	ANN	USA	Short term
14	Biswas and Sinha (2021)	LSTM, bidirectional LSTM	India	Short term
15	Liu et al. (2020)	Jaya-SVM	China	Short term
16	Ruiz-Aguilar et al. (2021)	EMD-ANN	Spain	Short term and long term
17	Xiao et al. (2021)	Self-adaptive kernel ELM	China	Short term
18	Wang and Yang (2021)	Optimized Elman neural network	China	Very short term

and up-bound function $f_2(x) = K(x^t, D^t)w_2 + b_2$, respectively. The primal problems of TSVR are expressed as follows:

$$\min \frac{1}{2} \|y - \varepsilon_1 e - (K(G, G^t)w_1 + b_1 e)\|^2 + C_2 e^t \psi_1, \quad (1)$$

$$s.t., y(K(G, G^t)w_1 + b_1 e) \geq \varepsilon_1 e - \psi_1, \geq 0$$

and

$$\min \frac{1}{2} \|y - \varepsilon_2 e - (K(G, G^t)w_2 + b_2 e)\|^2 + C_2 e^t \psi_2, \quad (2)$$

$$s.t., y(K(G, G^t)w_2 + b_2 e) \geq \varepsilon_2 e - \psi_2, \geq 0.$$

where regularization parameters are $C_1, C_2 > 0$, and input parameters are $\varepsilon_1, \varepsilon_2 > 0$; the slack variables are indicated by ψ_1 and ψ_2 . e is the ones vector and $G \in R^{m \times n}$ are the input samples.

By introducing the Lagrangian multipliers $\gamma_1, \gamma_2 > 0$ and applying the K.K.T. sufficient conditions on Eq. (2), the duals of Eqs. (1) and (2) are expressed as follows:

$$\max -\frac{1}{2} \gamma_1^t M_1 (M_1^t M_1)^{-1} M_1^t \gamma_1 + R^t M_1 (M_1^t M_1)^{-1} M_1^t \gamma_1 - R^t \gamma_1, \quad (3)$$

$$s.t., 0 \leq \gamma_1 \leq C_1 e$$

and

$$\max -\frac{1}{2} \gamma_2^t M_1 (M_1^t M_1)^{-1} M_1^t \gamma_2 + Q^t M_1 (M_1^t M_1)^{-1} M_1^t \gamma_2 - Q^t \gamma_2, \quad (4)$$

$$s.t., 0 \leq \gamma_2 \leq C_2 e,$$

where $R = (Y - \varepsilon_1 e)$, $Q = (y + \varepsilon_2 e)$, and $M_1 = [K(G, G^t) e]$. The unknowns w_1, w_2, b_1, b_2 can be determined as follows:

$$\begin{bmatrix} w_1 \\ b_1 \end{bmatrix} = (M_1^t M_1 + \partial I)^{-1} M_1^t (R - \gamma_1)$$

and

$$\begin{bmatrix} w_2 \\ b_2 \end{bmatrix} = (M_1^t M_1 + \partial I)^{-1} M_1^t (Q + \gamma_2),$$

where $\partial > 0$ and I is an identity matrix.

The PLSTSVR model

To reduce the computational complexity of the TSVR model, Huang et al. (2013) suggested a novel LSTVSR solved in primal, termed as least PLSTSVR. The formulation of PLSTSVR can be expressed as follows:

$$\min \frac{1}{2} \|R - (K(G, G^t)w_1 + b_1 e)\|^2 + \frac{1}{2} C_1 \psi_1^t \psi_1, \quad (5)$$

$$s.t., y - (K(G, G^t)w_1 + b_1 e) = \varepsilon_1 e - \psi_1$$

and



Fig. 1 Map of Tamil Nadu showing the chosen four sites (marked with boxes)

$$\begin{aligned}
 & \min \frac{1}{2} \|Q - (K(G, G^t)w_2 + b_2e)\|^2 + \frac{1}{2} C_2 \psi_2^t \psi_2, \\
 & \text{s.t.}, (K(G, G^t)w_2 + b_2e) - y = \varepsilon_2 e - \psi_2,
 \end{aligned} \tag{6}$$

where $R = (Y - \varepsilon_1 e)$ and $Q = (y + \varepsilon_2 e)$

Now, by placing the values of Eqs. (5) and (6), the slack vectors in the objective functions, further computation of the gradient with respect to w_i and b_i for $i = 1, 2$, and equating to zero, we get,

$$\begin{bmatrix} w_1 \\ b_1 \end{bmatrix} = (M_1^t M_1 + \partial I)^{-1} M_1^t R \tag{7}$$

and

Table 2 Geographical locations of the cities in Tamil Nadu, India

S. no	Station	LAT (°N)	LON (°E)	ALT (m)
1	Chennai	13.08	80.27	13
2	Madurai	9.92	78.11	137
3	Salem	11.66	78.14	286
4	Tirunelveli	8.71	77.75	41

Table 3 Statistics of the wind speed data

S. no	Station	Mean wind speed (m/s)	Median (m/s)	Maximum wind speed (m/s)	Standard deviation (m/s)
1	Chennai	6.228	6.38	18.1	2.381
2	Madurai	4.522	4.5	13.86	1.916
3	Salem	4.619	4.5	14.43	2.049
4	Tirunelveli	5.468	5.41	22.64	2.377

$$\begin{bmatrix} w_2 \\ b_2 \end{bmatrix} = (M_1^t M_1 + \partial I)^{-1} M_1^t Q, \tag{8}$$

where $M_1 = [K(G, G^t) \ e]$ is the augmented matrix; $\partial > 0$ is a small positive integer and I is an identity matrix.

The final regressor of PLSTSVR for any new instance $x \in R^n$ can be calculated as follows:

$$f(x) = \frac{1}{2}(K(x^t, G)(w_1 + w_2) + (b_1 + b_2)).$$

The LDMR model

Recently, Rastogi et al. (2020) introduced a margin distribution-based LDMR which was on the spirit of the LDM model (Zhang and Zhou 2014). LDMR simultaneously minimizes the ϵ -insensitive loss function and the quadratic loss function. Hence, LDMR shows improved regression performance for noisy data, and it considers the minimization of scattering which is inside the ϵ -tube. The primal problem of LDMR can be expressed as follows:

$$\begin{aligned} \min \frac{1}{2} \|y - (Gw + eb)\|^2 + \frac{1}{2} C \epsilon^t (\psi_1 + \psi_2) + \frac{1}{2} \|w\|^2, \\ \text{s.t., } y - (Gw + eb) \leq \epsilon \epsilon + \psi_1, \psi_1 \geq 0, \\ (Gw + eb) - y \leq \epsilon \epsilon + \psi_2, \psi_2 \geq 0, \end{aligned} \tag{9}$$

where the input parameters are $\epsilon, l_1, l_2, C > 0; \|w\|^2 = u_2^t I_0 u_2$

where $I_0 = \begin{bmatrix} I & 0 \\ & \vdots \\ 0 & \dots 0 \end{bmatrix}; I^{m \times m}$ is an identity matrix.

The dual problem of Eq. (9) may be obtained after adding the Lagrangian multiplier and applying the KKT condition as follows:

$$\begin{aligned} \min \frac{1}{2} (\gamma_1 - \gamma_2)^t M_1 (l_1 I_0 + l_2 M_1^t M_1)^{-1} M_1^t (\gamma_1 - \beta_2) + y^t M_1 (l_1 I_0 + l_2 M_1^t M_1)^{-1} M_1^t (\gamma_1 - \gamma_2), \\ -y^t (\gamma_1 - \gamma_2) + \epsilon \epsilon^t (\gamma_1 + \gamma_2), \\ \text{s.t., } 0 \leq \gamma_1 \leq C \epsilon, 0 \leq \gamma_2 \leq C \epsilon, \end{aligned} \tag{10}$$

where $M_1 = [K(G, G^t) \ e]$.

After determining the solution from Eq. (10) for β_1 and β_2 , the unknowns w and b can be obtained as follows (Hazarika et al. 2020a):

$$\begin{bmatrix} w \\ b \end{bmatrix} = (l_1 I_0 + l_2 M_1^t M_1)^{-1} M_1^t (\gamma_1 - \gamma_2 + y).$$

The LDMR regressor $f(\cdot)$ can be obtained for any new sample $x \in R^n$ as follows:

$$f(x) = [K(x^t, G^t) \ 1] \begin{bmatrix} w \\ b \end{bmatrix}.$$

Proposed models

The wavelet transform (WT) is an enhanced version of the conventional Fourier transform (FT) that was introduced by Jean Morlet in 1982 (Morlet et al. 1982a, b). Recently, the WT analysis has gained a lot of interest among the researchers as it overcomes the two key disadvantages of FT (Sifuzzaman et al. 2009; Hazarika et al. 2020a, b):

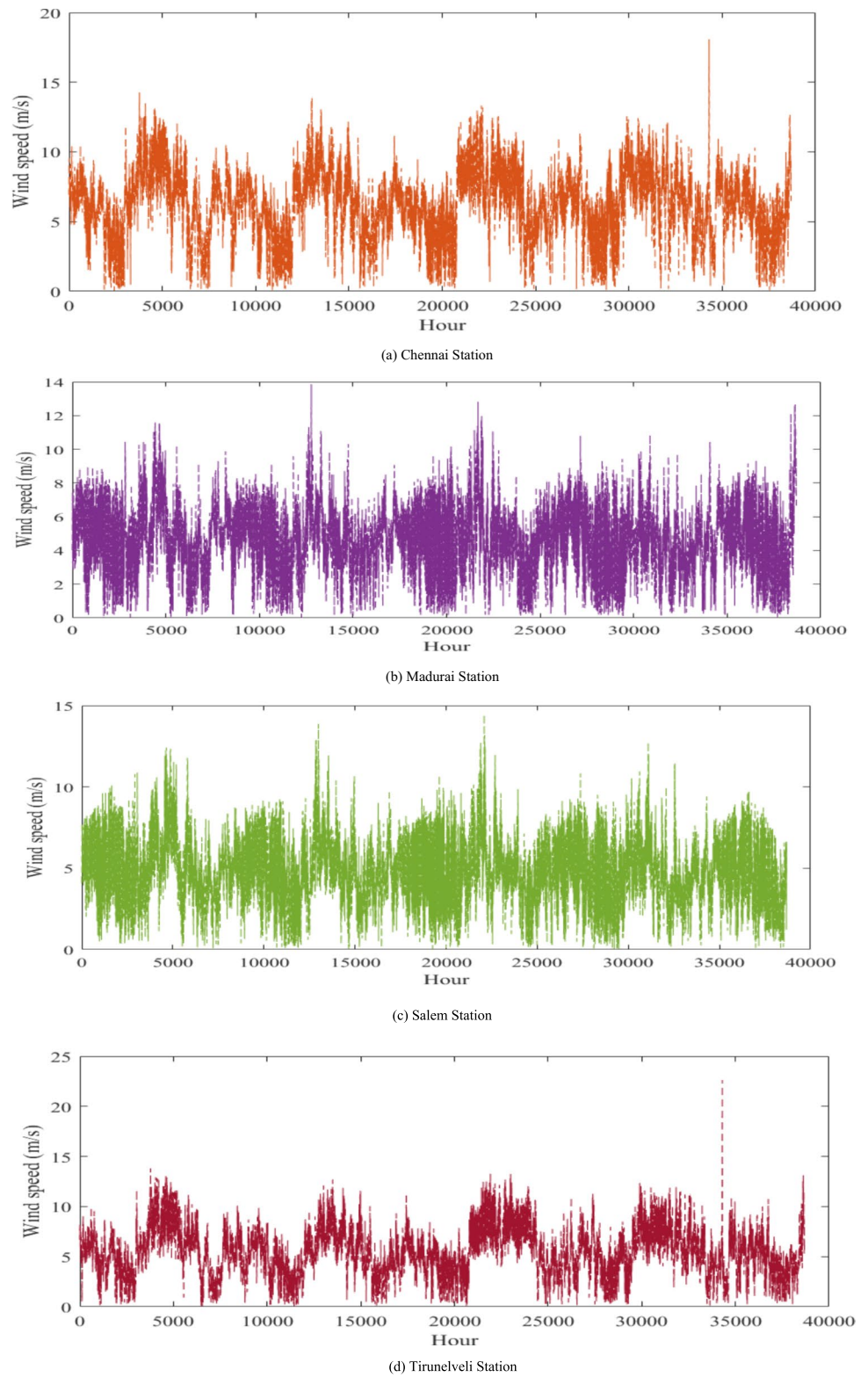
- 1) FT is suitable only for stationary signals, whereas WT is suitable for both stationary as well as non-stationary datasets.
- 2) In FT, the time information is lost while transforming the time domain to the frequency domain, whereas in WT, the time information is not lost.

Wavelet analysis was initially suggested to enhance seismic signal analysis by switching from short-term analysis of Fourier to improved algorithms to identify and analyse abrupt signal changes (Daubechies 1990, 1992; Mallat 1999). The wavelet has the multi-resolution and localization capacity in both time-domain (TD) and frequency domain (FD). Wavelet transform's TD properties can be elaborated through the wavelet functions which are translated from a wavelet base function (Holland 1992).

The wavelet functions can be derived from the mother wavelet (MW). Let $\lambda(z)$ be the MW function (Hazarika and Gupta 2020). The wavelet function $\lambda(z)$ can be obtained by the temporal translation t and with dilation δ as follows:

$$f_t(\delta, t) = \frac{1}{\sqrt{\delta}} \int_{-\infty}^{\infty} t(z) \lambda\left(\frac{z-t}{\delta}\right) \delta z \quad \delta > 0. \tag{11}$$

Fig. 2 Hourly wind speed data measured during January 2014 to May 2018. **a** Chennai Station. **b** Madurai Station. **c** Salem Station. **d** Tirunelveli Station



Equation (11) can be represented in the time domain as follows:

$$f_t(\delta, t) = \frac{1}{\sqrt{\delta}} \int_{-\infty}^{\infty} t(\omega) \lambda(a\omega) e^{i\omega z} \delta z \quad \delta > 0. \tag{12}$$

Table 4 Comparison of TSVR, PLSTSVR, LDMR, MKLSTSVR and MHKLSTSVR based on the performance evaluators, optimum parameters and the computational times for 1-3 lag time period. (Best results are bolded.)

Dataset (train, test)	Performance evaluators	TSVR	PLSTSVR	LDMR	MKLSTSVR	MHKLSTSVR
		(C, μ, ϵ) time (sec)	(C, μ, ϵ) time (sec)	(C_1, μ, C_2, ϵ) time (sec)	(C, ϵ) time (sec)	(C, ϵ) time (sec)
Chennai	E1	0.040317	0.032152	0.031793	0.031951	0.031890
	E2	0.030705	0.023787	0.023889	0.023762	0.024037
	E3	0.110742	0.090936	0.091312	0.091740	0.095099
	E4	0.943912	0.731252	0.734401	0.730474	0.738916
	E5	0.91958	0.94218	0.9434	0.942841	0.942999
	Parameters	($10^{-3}, 8, 1.5$)	($10^{-5}, 4, 1$)	($10^{-1}, 8, 0.001, 0.05$)	($10^{-5}, 0.5$)	($10^{-5}, 1.5$)
	Time	9.77998	1.18909	1.09127	0.992544	0.861059
Madurai	E1	0.065534	0.05581	0.053725	0.054514	0.055597
	E2	0.049161	0.041094	0.040596	0.040875	0.041337
	E3	0.183424	0.154061	0.152974	0.153802	0.156680
	E4	0.959875	0.802353	0.79264	0.798084	0.807099
	E5	0.79137	0.84623	0.85563	0.851104	0.845275
	Parameters	($10^{-1}, 8, 0.05$)	($10^{-5}, 8, 1$)	($10^{-3}, 8, 0.001, 1$)	($10^{-5}, 2$)	($10^{-5}, 2$)
	Time	5.44994	1.52731	1.15297	0.96967	0.887909
Salem	E1	0.066159	0.056481	0.058038	0.055535	0.055013
	E2	0.053527	0.041053	0.041513	0.040799	0.040412
	E3	0.196464	0.154028	0.156257	0.153872	0.155207
	E4	1.04695	0.802962	0.811969	0.798012	0.790428
	E5	0.82664	0.84468	0.83438	0.848699	0.85122
	Parameters	($10^{-1}, 2, 0.05$)	($10^{-5}, 2, 1.5$)	($10^{-3}, 8, 0.001, 0.1$)	($10^{-5}, 2$)	($10^{-5}, 2$)
	Time	9.36352	1.64833	1.15626	1.01518	0.879462
Tirunelveli	E1	0.037187	0.029061	0.029063	0.029685	0.290056
	E2	0.027843	0.0215	0.021501	0.021537	0.021516
	E3	0.131682	0.112339	0.11234	0.112068	0.111240
	E4	1.05238	0.812639	0.812653	0.814044	0.813238
	E5	0.90645	0.93187	0.93186	0.928904	0.932099
	Parameters	($10^{-1}, 2, 2$)	($10^{-5}, 2, 2$)	($10^0, 2, 0.00001, 0.5$)	($10^{-5}, 0.1$)	($10^{-5}, 2$)
	Time	9.31972	1.22065	1.11234	0.927693	0.893671

From Eqs. (11) and (12), one can observe that the local characteristics of a signal can be reflected through the wavelet transform. Hence, the wavelet transforms as an analytical technique has shown great potential (Zhang et al. 2004; Zhou and Ye 2006; Ding et al. 2016).

Translation-invariant kernel

The kernel of translation-invariant (TI) is acceptable if and only if the FT is always positive.

Lemma 1: Let $\lambda(z)$ be an MW, and let δ and t denote the dilation and translation. If $z, z' \in R^N$, then the dot-product wavelet kernel may be expressed as follows:

$$K(z, z') = \prod_{i=1}^N h\left(\frac{z_i - t_i}{\delta}\right) h\left(\frac{z'_i - t'_i}{\delta}\right). \quad (13)$$

The TI kernel of Eq. (13) may be expressed as

$$K(z, z') = \prod_{i=1}^N h\left(\frac{z_i - z'_i}{\delta}\right). \quad (14)$$

In Eq. (14), N represents the total quantity of the sample (Ding et al. 2014).

To generate the TI kernel function, this work uses two wavelet kernel functions. They are:

- Morlet wavelet kernel: $\left(\lambda(z) = \cos(1.75z) \exp\left(-\frac{z^2}{2}\right)\right)$
and
- Mexican hat wavelet kernel: $\left(\lambda(z) = (1 - z^2) \exp\left(-\frac{z^2}{2}\right)\right)$.

Proposed MKLSTSVR

MKLSTSVR indicates the PLSTSVR embedded with the Morlet wavelet kernel instead of using the traditional kernels. To be an acceptable kernel, the Morlet

Table 5 Average ranks based on evaluators for 1-3. (Best average rank is bolded.)

Dataset	Evaluators	TSVR	LSTSVR	LDMR	MKLSTSVR	MHKLSTSVR
Chennai	E1	5	4	1	3	2
	E2	5	2	3	1	4
	E3	5	1	2	3	4
	E4	5	2	3	1	4
	E5	5	4	1	3	2
	Time	5	4	3	2	1
Madurai	E1	5	4	1	2	3
	E2	5	3	1	2	4
	E3	5	3	1	2	4
	E4	5	3	1	2	4
	E5	5	3	1	2	4
	Time	5	4	3	2	1
Salem	E1	5	3	4	2	1
	E2	5	3	4	2	1
	E3	5	2	4	1	3
	E4	5	3	4	2	1
	E5	5	3	4	2	1
	Time	5	4	3	2	1
Tirunelveli	E1	4	1	2	3	5
	E2	5	1	2	4	3
	E3	5	3	4	2	1
	E4	5	1	2	4	3
	E5	5	2	3	4	1
	Time	5	4	3	2	1
Average rank		4.95833	2.79167	2.5	2.29167	2.45833

kernel should satisfy the TI kernel theorem as shown in Eq. (14).

Lemma 2: The Morlet wavelet kernel function (MK) satisfies the TI kernel theorem as follows:

$$K(z, z') = \prod_{i=1}^N \left(\cos \left(1.75 \times \frac{z_i - z'_i}{\delta} \right) \exp \left(-\frac{\|z_i - z'_i\|^2}{2\delta^2} \right) \right),$$

which is an admissible kernel (Zhang et al. 2004). Hence, it is a kernel by which the wavelet kernel trick can be used to construct MKLSTSVR. The basic goal of MKLSTSVR is to search for the optimum wavelet coefficients in feature space spanned by multidimensional wavelet basis (MWB). Thus, it obtains the optimal prediction function. Thereby, the prediction function for MKLSTSVR can be obtained as follows:

$$f(z) = \frac{1}{2} \prod_{i=1}^N \left(\cos \left(1.75 \times \frac{z_i - z'_i}{\delta} \right) \exp \left(-\frac{\|z_i - z'_i\|^2}{2\delta^2} \right) \right)^{(w_1 + w_2)} + \frac{1}{2}(b_1 + b_2).$$

Proposed MHKLSTSVR

MHKLSTSVR indicates the PLSTSVR combined with the Mexican hat wavelet kernel rather than the traditional kernels. To be an admissible kernel function, the Mexican hat kernel should also satisfy the TI kernel theorem as shown in Eq. (14).

Lemma 3: The Mexican hat wavelet kernel function (MHK) follows the TI kernel theorem as the following:

$$K(z, z') = \prod_{i=1}^N \left(1 - \left(\frac{z_i - z'_i}{\delta} \right)^2 \right) \exp \left(-\frac{1}{2} \left(\frac{z_i - z'_i}{\delta} \right)^2 \right),$$

which also can be considered as an admissible kernel (Zhang et al. 2004). Therefore, the Mexican hat wavelet kernel trick can be used to construct MHKLSTSVR. Like MKLSTSVR, the basic goal of MHKLSTSVR is to find the optimum wavelet coefficients in feature space spanned by MWB. Thereby, it obtains the optimum predictor. Thereby, the predictor for MHKLSTSVR can be obtained as follows:

Table 6 Comparison of TSVR, PLSTSVR, LDMR, MKLSTSVR and MHKLSTSVR based on the performance evaluators, optimum parameters and the computational times for 1-5 lag time period. (Best results are bolded.)

Datasets	Performance measures	TSVR	PLSTSVR	LDMR	MKLSTSVR	MHKLSTSVR
		(C, μ, ε) Time (sec)	(C, μ, ε) Time (sec)	(C ₁ , μ, C ₂ , ε) Time (sec)	(C, ε) Time (sec)	(C, ε) Time (sec)
Chennai	E1	0.033796	0.032002	0.032002	0.302911	0.031758
	E2	0.025916	0.024031	0.024031	0.023576	0.023781
	E3	0.101085	0.096189	0.096194	0.091211	0.094595
	E4	0.7968	0.738843	0.73885	0.724851	0.731174
	E5	0.93735	0.94259	0.94259	0.943063	0.943473
	Parameters	(10 ⁰ , 0.125, 2)	(10 ⁻⁵ , 0.03125, 1.5)	(10 ³ , 0.03125, 0.00001, 1)	(10 ⁻⁵ , 2)	(10 ⁻⁵ , 2)
	Time	4.36071	0.96549	0.96191	0.90619	0.89235
Madurai	E1	0.060527	0.055502	0.055412	0.054339	0.054904
	E2	0.047553	0.041597	0.041871	0.041230	0.041442
	E3	0.180651	0.157376	0.160724	0.156715	0.158961
	E4	0.928585	0.812287	0.817626	0.80513	0.809240
	E5	0.82487	0.84747	0.84682	0.852824	0.849298
	Parameters	(10 ⁻¹ , 2, 0.05)	(10 ⁻⁵ , 2, 2)	(10 ⁻¹ , 0.5, 0.00001, 0.05)	(10 ⁻⁵ , 1)	(10 ⁻⁵ , 0.05)
	Time	5.66962	1.38711	1.20347	1.02159	0.89073
Salem	E1	0.063572	0.05765	0.05618	0.061270	0.053512
	E2	0.050841	0.040804	0.04118	0.041425	0.039551
	E3	0.188865	0.154535	0.156842	0.144427	0.155378
	E4	0.994576	0.798223	0.805592	0.810834	0.773722
	E5	0.836	0.84186	0.84954	0.824340	0.860144
	Parameters	(10 ⁻¹ , 2, 0.05)	(10 ⁻⁵ , 2, 2)	(10 ⁻¹ , 8, 0.001, 0.05)	(10 ⁻⁵ , 0.05)	(10 ⁻⁵ , 1)
	Time	6.48069	1.61223	1.34246	1.01786	0.03955
Tirunelveli	E1	0.033316	0.029369	0.029087	0.032374	0.028655
	E2	0.026206	0.02196	0.02177	0.021404	0.021260
	E3	0.139118	0.117655	0.116576	0.110642	0.110273
	E4	0.986401	0.826578	0.81941	0.809213	0.803793
	E5	0.92185	0.9317	0.93306	0.915509	0.933839
	Parameters	(10 ⁻³ , 0.125, 1)	(10 ⁻⁵ , 0.03125, 0.5)	(10 ¹ , 0.5, 0.00001, 0.1)	(10 ⁻⁵ , 0.5)	(10 ⁻⁵ , 2)
	Time	5.92681	1.43613	1.12457	0.99901	0.88512

$$f(z) = \frac{1}{2} \prod_{i=1}^N \left(1 - \left(\frac{z_i - z'_i}{\delta} \right)^2 \right) \exp \left(-\frac{1}{2} \left(\frac{z_i - z'_i}{\delta} \right)^2 \right) (w_1 + w_2) + \frac{1}{2} (b_1 + b_2).$$

Dataset description

The state of Tamil Nadu is situated in the southern part of India. Tamil Nadu is bordered by the Eastern Ghats in the north, Anaimalai hills and Kerala in the west, Bay of Bengal in the east, Gulf of Mannar and Palk Strait in the south-east and the Indian Ocean in the south. The climate of Tamil Nadu ranges from sub-humid to semi-arid. The sites selected for this study has been provided in Fig. 1. The geographical information such as latitude (LAT), longitude (LON) and altitude (ALT) of the chosen sites are provided in Table 2. Hourly average WS data recorded at a height of 50 m above ground level was collected from MERRA-2 analysis database (NASA) for the period of Jan 1980 to May 2018. However, in

our experiments, we have considered the data from Jan 2014 to May 2018.

The statistics of the WS data of the above four stations have been provided in Table 3 below.

The maximum wind speed of 22.64 m/s is noticed in Tirunelveli, and the lowest wind speed of 13.86 m/s is noticed in Madurai. The observation is similar for the mean WS too. Figure 2 shows the hourly wind speed data from January 2014 to May 2018.

Experimental setup and numerical analysis

To illustrate the prediction capability of the proposed MKLSTSVR and MHKLSTSVR models, their performance has been compared with TSVR, PLSTSVR and LDMR on a portion of the WS dataset. The performances of these implemented models are evaluated using RMSE

Table 7 Average ranks based on evaluators for 1-5. (Best average rank is bolded.)

Dataset	Evaluators	TSVR	LSTSVR	LDMR	MKLSTSVR	MHKLSTSVR
Chennai	E1	4	2.5	2.5	5	1
	E2	5	3.5	3.5	1	2
	E3	5	3	4	1	2
	E4	5	3	4	1	2
	E5	4	3.5	3.5	2	1
	Time	5	4	3	2	1
Madurai	E1	5	4	3	1	2
	E2	5	3	4	1	2
	E3	5	2	4	1	3
	E4	5	3	4	1	2
	E5	5	3	4	1	2
	Time	5	4	3	2	1
Salem	E1	5	3	2	4	1
	E2	5	2	3	4	1
	E3	5	2	4	1	3
	E4	5	2	3	4	1
	E5	4	3	2	5	1
	Time	5	4	3	2	1
Tirunelveli	E1	5	3	2	4	1
	E2	5	4	3	2	1
	E3	5	4	3	2	1
	E4	5	4	3	2	1
	E5	4	3	2	5	1
	Time	5	4	3	2	1
Average rank		4.83333	3.1875	3.14583	2.33333	1.45833

(root mean squared error), MAE (mean absolute error), SMAPE (symmetric mean absolute percentage error), MASE (mean absolute scaled error) and R^2 (coefficient of correlation). The definitions of these evaluators can be expressed as:

- **RMSE:** RMSE describes how close the data is near the line of best fit. To validate experimental results, RMSE is frequently applied in climatology, forecasting and regression analysis.

$$RMSE = \sqrt{\frac{\sum_{k=1}^N (o - p)^2}{N}}$$

- **MAE:** The amount of error in your measurements is expressed as absolute error (AE). It represents the difference between the measured and original values. The MAE is the average of all absolute errors.

$$MAE = \frac{\sum_{k=1}^N |o - p|}{N}$$

- **SMAPE:** The mean absolute percentage error is a popular metric for assessing forecasting performance. MAPE is asymmetric, penalising negative errors more than positive errors when forecasts are higher than originals. SMAPE is a symmetric version of MAPE that overcomes MAPE’s asymmetry.

$$SMAPE = \frac{\sum_{k=1}^N |o - p|}{N(p + o)}$$

Here, o and p indicate the original and the predicted WS values, respectively.

- **MASE:** Mean absolute scaled error (MASE) is a scale-free error metric that presents each error as a ratio to the average error of a baseline. MASE has the advantage of never returning undefined or infinite values, making it an excellent choice for intermittent-demand series. It can be applied to a single series or used to compare multiple series.

$$MASE = \text{mean } |z|; \text{ where } z = \frac{f}{\frac{\sum_{k=2}^N |o_k - o_{k-1}|}{N-1}}$$

f is the forecast error in the given period.

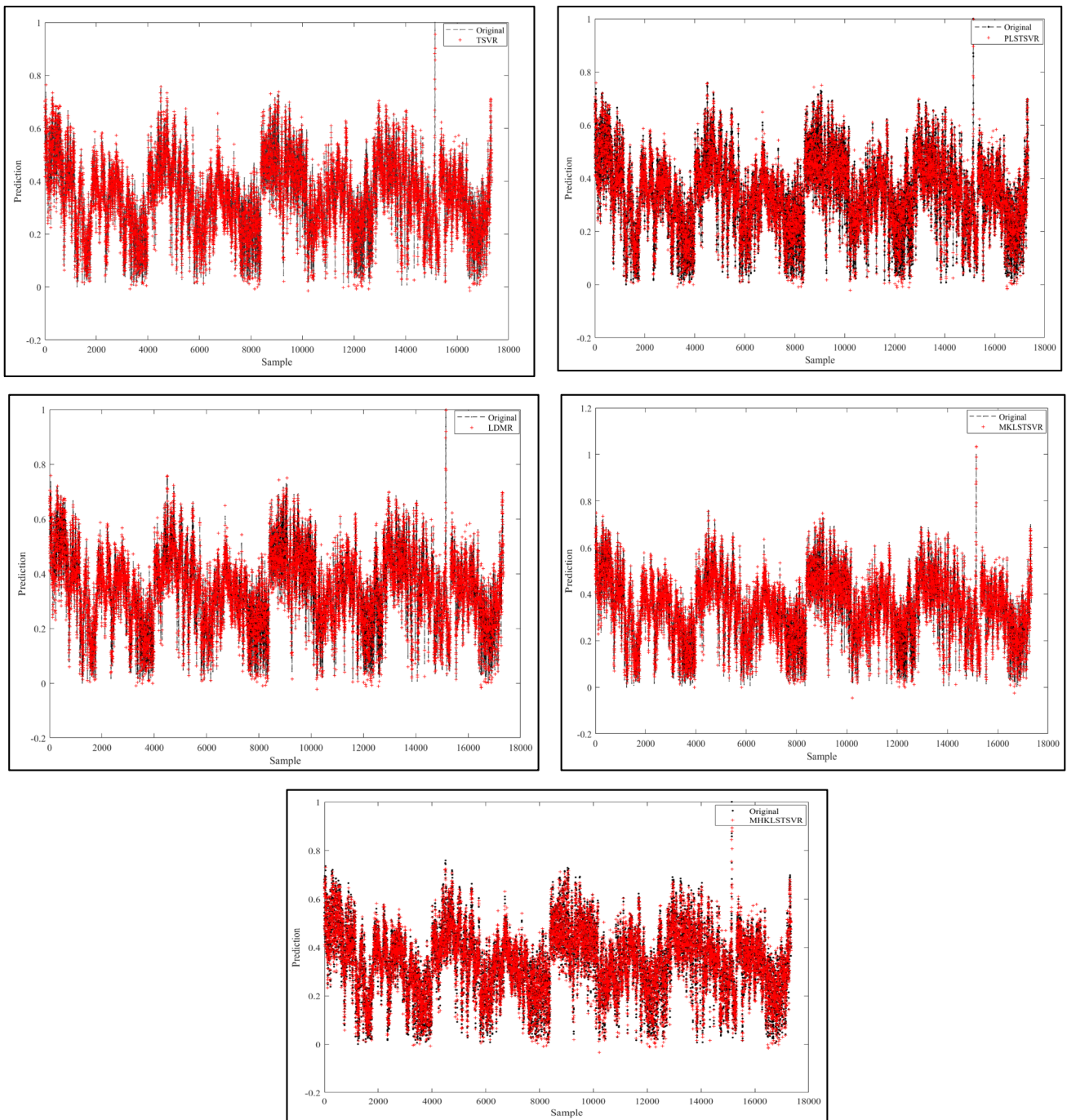


Fig. 3 Comparison of original and predicted WS values for Chennai using 1-5

- R^2 : R^2 is a statistical measure that represents the proportion of the variance described by an independent variable or variables in a regression model for a dependent variable. A model's R^2 value of 0.50 indicates that the model's inputs can describe roughly half of the original variation.

In our experiments, we have considered short names for the evaluators, i.e. RMSE (E1), MAE (E2), SMAPE (E3), MASE (E4) and R^2 (E5).

All the experiments were conducted on Window 8 installed laptop computer with 4 GB RAM. The QPPs of TSVR and LDMR models are solved using the external MOSEK optimization toolbox. The Gaussian kernel is

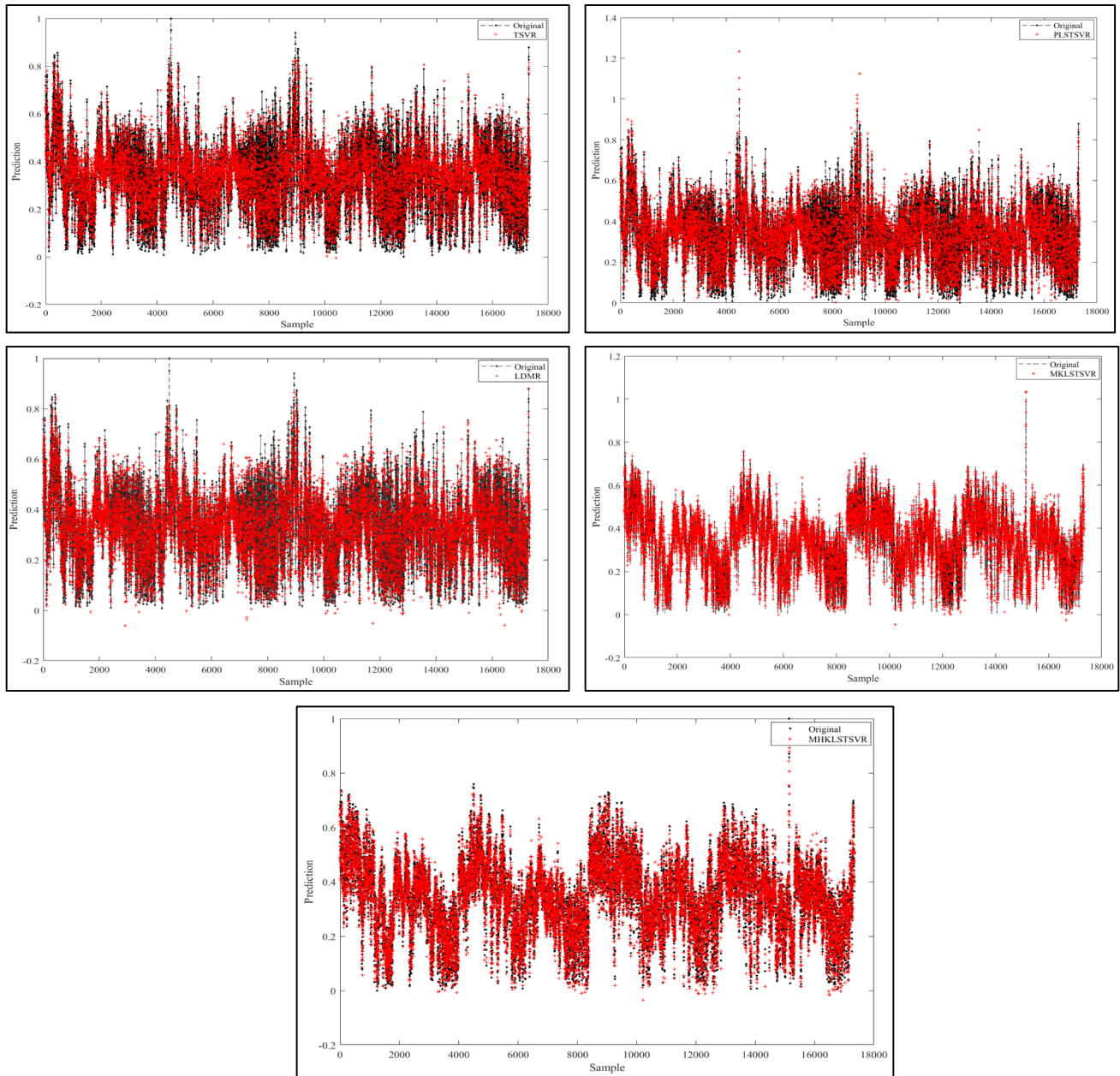


Fig. 4 Comparison of original and predicted WS values for Madurai using 1-5

used for the TSVR, PLSTSVR and LDMR models. The Gaussian kernel may be represented as follows:

$K(d_m, d_n) = \exp(-\mu||d_m - d_n||^2)$, for $m, n = 1, \dots, m$, where d_m and d_n indicate the input data samples. The Gaussian kernel parameter $\mu > 0$ is selected from the range $\{ 2^{-5}, \dots, 2^5 \}$ for TSVR, PLSTSVR and LDMR models. The value of the ϵ parameter is set from $\epsilon = \{0.05, 0.1, 0.5, 1, 1.5, 2\}$ for all models including the proposed MKLSTSVR and MHKLSTSVR.

The model parameters $C_1 = C_2$ are considered from $\{ 10^{-5}, 10^{-3}, \dots, 10^5 \}$ for all models. The Gaussian kernel parameter λ is taken from the set $\{ 2^{-5}, 2^{-3}, \dots, 2^5 \}$ for TSVR, PLSTSVR and LDMR models. The λ values are chosen from $\{ 10^{-5}, 10^{-3}, \dots, 10^5 \}$ for LDMR. For the proposed MKLSTSVR and MHKLSTSVR models, the wavelet kernel parameter d is fixed as 1.

Two types of lag periods are considered: 1-3 and 1-5. Table 4 and Table 5 portray the results based on the

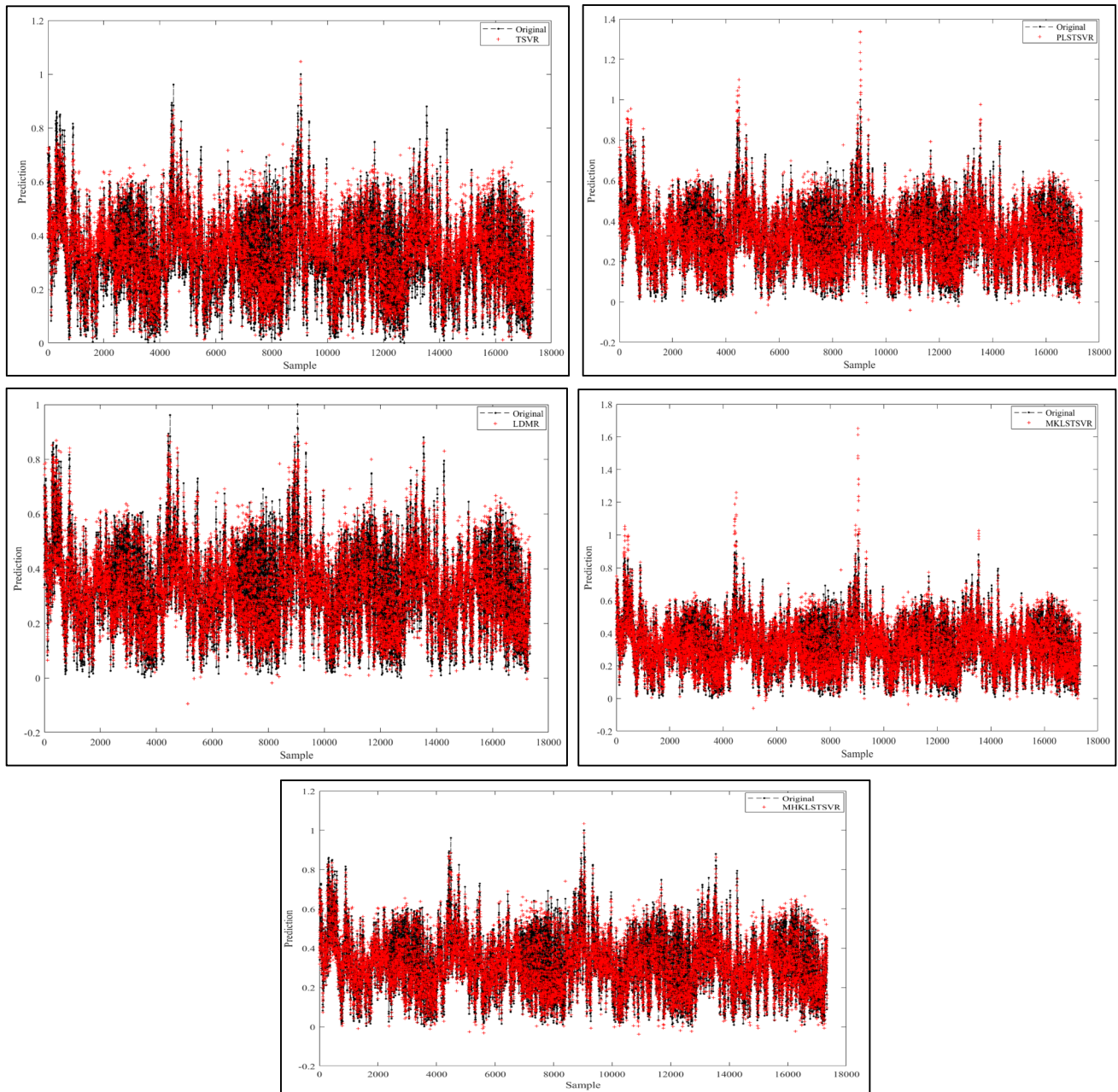


Fig. 5 Comparison of original and predicted WS values for Salem using a 1-5

various performance evaluators utilized in this study for the various cities considered for 1-3 and 1-5, respectively. From Table 4, the following conclusions can be derived:

- a) The proposed MKLSTSVR and MHKLSTSVR show the best performance based on RMSE for 0 and 2 cases, respectively.
- b) The proposed MKLSTSVR and MHKLSTSVR show best results based on MAE in 1 case each.
- c) The proposed MKLSTSVR and MHKLSTSVR show best results based on SMAPE in 2 cases and 1 case, respectively.
- d) The proposed MKLSTSVR and MHKLSTSVR indicate the best performance based on MASE in 1 case each.
- e) The proposed MKLSTSVR and MHKLSTSVR show the best performance based on R^2 in 0 and 2, respectively.

Table 5 shows ranks obtained by the reported models based on various performance evaluators. It can be observed

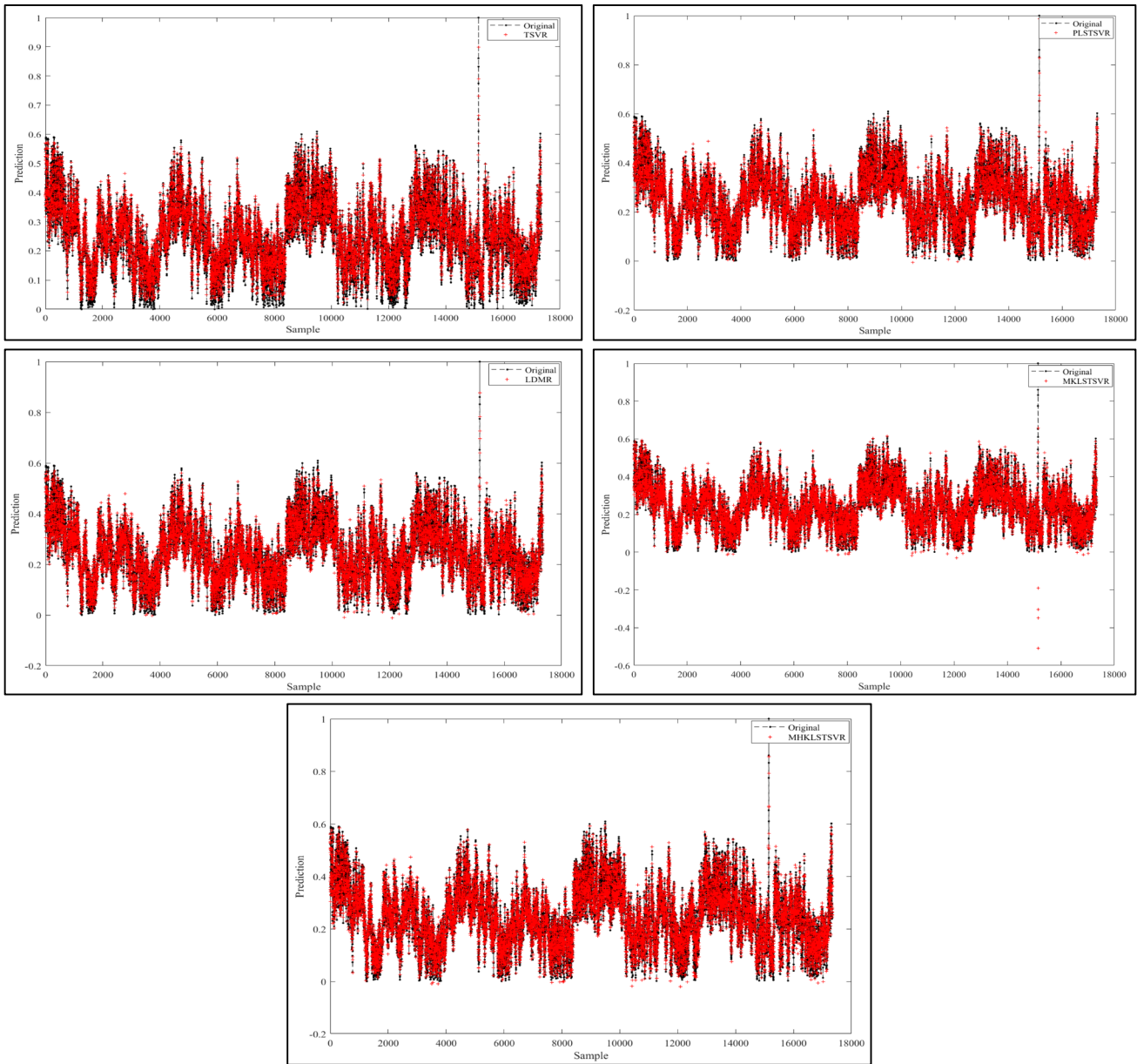


Fig. 6 Comparison of original and predicted WS values for Tirunelveli using a 1-5

that our proposed MKLSTSVR shows the lowest average rank followed by the MHKLSTSVR, which promulgates the efficiency of these models.

The window size is enlarged from 1-3 to 1-5 to further evaluate the performance of these models. In Table 6, the results are shown using a lag window of 5. From Table 6, the following conclusions can be derived:

- a) The proposed MKLSTSVR and MHKLSTSVR reveal the best results based on RMSE in 2 cases each.
- b) The proposed MKLSTSVR and MHKLSTSVR report the best performance based on MAE in 2 cases each.
- c) Based on SMAPE, the proposed MKLSTSVR and MHKLSTSVR show the best performance in 3 and 1 case, respectively.
- d) Based on MASE, the proposed MKLSTSVR and MHKLSTSVR show the best performance in 2 cases each.
- e) Based on R^2 , the proposed MKLSTSVR and MHKLSTSVR show the best performance in 1 and 3 cases, respectively.

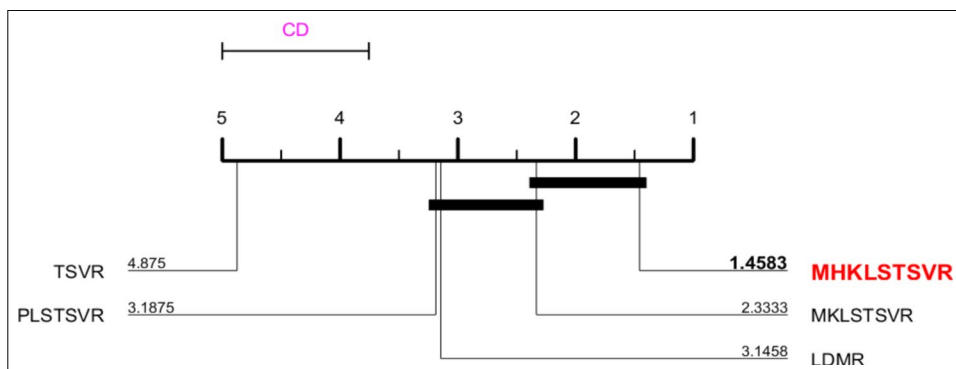


Fig. 7 Statistical comparison based on Friedman test with Nemenyi statistics

Table 7 shows ranks obtained by the reported models based on different evaluators. One can notice that our proposed MHKLSTSVR shows the lowest average rank followed by the MKLSTSVR, which indicates the efficacy of the models.

Overall, it is noticeable that the increase in the time lag period shows improved performance for the proposed MKLSTSVR and MHKLSTSVR. Figures 3, 4, 5, and 6 reveal the comparison between the original and predicted WS values of TSVR, PLSTSVR, LDMR and the proposed MKLSTSVR and MHKLSTSVR for 1-5. Although all the plots seem to be similar, the overall performance of the proposed models is better compared to TSVR, PLSTSVR and LDMR models based on the evaluators.

It can also be observed from Table 4 and Table 5 that the proposed models MKLSTSVR and MHKLSTSVR take less time for computation compared to other models. Thus, it will be interesting to see how the proposed MKLSTSVR and MHKLSTSVR behave in short-term WS prediction at different sites.

Statistical analysis using Friedman test with post hoc analysis

It is noticeable that the proposed models MKLSTSVR and MHKLSTSVR do not show the best results in all the cases. To further justify the efficiency of the best proposed model, the non-parametric Friedman test (Demšar 2006) has been performed. To generate the Friedman statistics, the average ranks of the five models are considered from Table 7.

Let us assume that all the models are not significantly different, from the null hypothesis (NH). Then, we formulate NH as follows:

$$\chi^2_F = \frac{12 \times 24}{5 \times (5+1)} \left[4.8333^2 + 3.1875^2 + 3.1458^2 + 2.3333^2 + 1.4583^2 - \frac{7 \times (7+1)^2}{4} \right] = 57.5909$$

$$F_F = \frac{(24-1) \times 57.5909}{24 \times (5-1) - 57.5909} = 34.3371$$

The critical value C_V for F_F with $(5 - 1)$ and $(5 - 1) \times (24 - 1)$ degrees of freedom is 2.471 for $\alpha = 0.05$. As it can be noticed that $F_F > C_V$; therefore, we can reject the NH. Furthermore, the Nemenyi test is performed with $p = 0.05$ as follows:

$$CD = 2.728 \sqrt{\frac{5 \times (5 + 1)}{6 \times 24}} = 1.2452$$

For significance, the pairwise difference in average ranks should be more than CD. It is worth noting that the proposed MHKLSTSVR is significantly different compared to TSVR, PLSTSVR and LDMR. However, no significant difference can be found between MKLSTSVR and MHKLSTSVR. The Friedman test with Nemenyi Statistics can be visualized from Fig. 7. Those models that are not connected by a line are statistically significant. It can be observed that our proposed MHKLSTSVR is significantly different than TSVR, PLSTSVR and LDMR.

Conclusion

The wavelets are powerful enough to deal with the non-stationary datasets. Moreover, the PLSTSVR model shows excellent prediction performance. Hence, in this work, the wavelet kernels are embedded in the PLSTSVR models and two novel wavelets kernel-based PLSTSVR models, namely, MKLSTSVR and MHKLSTSVR are proposed for WS prediction in four different wind stations in India. The results show the proposed models to be superior when compared to the related models, i.e. TSVR, PLSTSVR and LDMR. Among the proposed MKLSTSVR and MHKLSTSVR models, the latter shows a closer relationship with the original data. Moreover, the proposed models are computationally efficient as they take less time for computation compared to the other reported models. Comprehensively, one can conclude that both MKLSTSVR

and MHKLSTSVR are efficient models and applicable for short-term WS prediction. These models can also be applied in the field of engineering like prediction of river suspended sediment load, rainfall forecasting, runoff prediction, etc.

Author contribution Role of Barenya Bikash Hazarika — conceptualization, methodology, writing original draft and visualization. Role of Deepak Gupta — investigation, validation, writing review and editing. Role of Narayanan Natarajan — writing original draft and formal analysis.

Data availability The datasets that has been used in this study are available from co-author on reasonable request.

Declarations

Ethics approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

Competing interests The authors declare no competing interests.

References

- Bakhsh H, Srinivasan R, Bahel V (1985) Correlation between hourly diffuse and global radiation for Dhahran. *Saudi Arabia Solar Wind Technol* 2(1):59–61
- Blanchard T, Samanta B (2020) Wind speed forecasting using neural networks. *Wind Eng* 44(1):33–48
- Biswas, S., & Sinha, M. (2021). Performances of deep learning models for Indian Ocean wind speed prediction. *Modeling Earth Systems and Environment*, 7(2), 809–831
- Daubechies I (1990) The wavelet transform, time-frequency localization and signal analysis. *IEEE Trans Info Theory* 36(5):961–1005
- Daubechies I (1992) Ten lectures on wavelets. *S Ind Appl Math*. <https://doi.org/10.1137/1.9781611970104>
- Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7:1–30
- Dhiman HS, Anand P, Deb D (2019) Wavelet Transform and Variants of SVR with Application in Wind Forecasting. In: Deb D, Balas V, Dey R (eds) *Innovations in Infrastructure. Advances in Intelligent Systems and Computing*, vol 757. Springer, Singapore. https://doi.org/10.1007/978-981-13-1966-2_45
- Ding S, Wu F, Shi Z (2014) Wavelet twin support vector machine. *Neural Comput Appl* 25(6):1241–1247
- Ding S, Zhang J, Xu X, Zhang Y (2016) A wavelet extreme learning machine. *Neural Comput Appl* 27(4):1033–1040
- Fu C, Li GQ, Lin KP, Zhang HJ (2019) Short-term wind power prediction based on improved chicken algorithm optimization support vector machine. *Sustainability* 11(2):512
- Gupta D (2017) Training primal K-nearest neighbor based weighted twin support vector regression via unconstrained convex minimization. *Appl Intell* 47(3):962–991
- Gupta D, Kumar V, Ayus I, Vasudevan M, Natarajan N (2021) Short-term prediction of wind power density using convolutional LSTM. *FME Trans* 49:653–663
- Harbola S, Coors V (2019) One dimensional convolutional neural network architectures for wind prediction. *Energy Convers Manage* 195:70–75
- Hazarika BB, Gupta D (2020) Modelling and forecasting of COVID-19 spread using wavelet-coupled random vector functional link networks. *Appl Soft Comput* 96:106626
- Hazarika BB, Gupta D, Berlin M (2020a) A coiflet LDMR and coiflet OB-ELM for river suspended sediment load prediction. *Int J Environ Sci Technol* 18:2675–2692
- Hazarika BB, Gupta D, Berlin M (2020b) Modeling suspended sediment load in a river using extreme learning machine and twin support vector regression with wavelet conjunction. *Environ Earth Sci* 79:234
- He J, Xu J (2019) Ultra-short-term wind speed forecasting based on support vector machine with combined kernel function and similar data. *EURASIP J Wireless Comm Network* 2019(1):248
- Holland JH (1992) *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT Press, pp. 207–211
- Huang HJ, Ding SF, Shi ZZ (2013) Primal least squares twin support vector regression. *J Zhejiang Uni SCI C* 14(9):722–732
- Jamil M, Zeeshan M (2019) A comparative analysis of ANN and chaotic approach-based wind speed prediction in India. *Neural Comput Appl* 31(10):6807–6819
- Jha SK, Bilalovikj J (2019) Short-term wind speed prediction at Bogdanci power plant in FYROM using an artificial neural network. *Int J Sustain Energy* 38(6):526–541
- Khosravi A, Koury RNN, Machado L, Pabon JGG (2018) Prediction of wind speed and wind direction using artificial neural network, support vector regression and adaptive neuro-fuzzy inference system. *Sustain Energy Technol Assess* 25:146–160
- Li LL, Zhao X, Tseng ML, Tan RR (2020) Short-term wind power forecasting based on support vector machine with improved dragonfly algorithm. *J Clean Prod* 242:118447
- Liu D, Niu D, Wang H, Fan L (2014) Short-term wind speed forecasting using wavelet transform and support vector machines optimized by genetic algorithm. *Renew Energy* 62:592–597
- Liu M, Cao Z, Zhang J, Wang L, Huang C, Luo X (2020) Short-term wind speed forecasting based on the Jaya-SVM model. *Int J Elect Power Energy Syst* 121:106056
- Mallat S (1999) *A wavelet tour of signal processing*. Academic Press, San Diego, CA
- Mi X, Liu H, Li Y (2019) Wind speed prediction model using singular spectrum analysis, empirical mode decomposition and convolutional support vector machine. *Energy Convers Manage* 180:196–205
- Morlet J, Arens G, Fourgeau E, Giard D (1982a) Wave propagation and sampling theory—part II: sampling theory and complex waves. *Geophysics* 47(2):222–236
- Morlet J, Arens G, Fourgeau E, Giard D (1982b) Wave propagation and sampling theory—part I: complex signal and scattering in multilayered media. *Geophysics* 47(2):203–221
- Natarajan YJ, Nachimuthu DS (2019) New SVM kernel soft computing models for wind speed prediction in renewable energy applications. *Soft Comput* 24:11441–11458
- Peng X (2010) TSVR: an efficient twin support vector machine for regression. *Neural Netw* 23(3):365–372
- Prasetyowati A, Sudiana D, Sudibyo H (2019) Prediction of wind power model using hybrid method based on WD-SVM algorithm: case study Pandansimo Wind Farm. In: *Journal of Physics: Conference Series*, IOP Publishing, 1338(1): 012048
- Qolipour M, Mostafaeipour A, Saidi-Mehrabad M, Arabnia HR (2019) Prediction of wind speed using a new grey-extreme learning machine hybrid algorithm: a case study. *Energy Environ* 30(1):44–62

- Rastogi R, Anand P, Chandra S (2020) Large-margin distribution machine-based regression. *Neural Comput Appl* 32(8):3633–3648
- Rehman S, Halawani TO (1994) Statistical characteristics of wind in Saudi Arabia. *Renew Energy* 4(8):949–956
- Ruiz-Aguilar JJ, Turias I, González-Enrique J, Urda D, Elizondo D (2021) A permutation entropy-based EMD-ANN forecasting ensemble approach for wind speed prediction. *Neural Comput Appl* 33(7):2369–2391
- Salcedo-Sanz S, Ortiz-Garcı EG, Pérez-Bellido ÁM, Portilla-Figueras A, Prieto L (2011) Short term wind speed prediction based on evolutionary support vector regression algorithms. *Expert Syst Appl* 38(4):4052–4057
- Sangita BP, Deshmukh SR (2011) Use of support vector machine, decision tree and naive Bayesian techniques for wind speed classification. In: 2011 International Conference on Power and Energy Systems, IEEE, 1–8
- Sifuzzaman M, Islam MR, Ali MZ (2009) Application of wavelet transform and its advantages compared to Fourier transform. *J Phy Sci* 13:121–134
- Sivanagaraja T, Tatinati AK, Veluvolu KC (2014) A hybrid method based on discrete wavelets and least squares support vector machines for short-term wind speed forecasting. *Int J Info Comput Tech* 4(14):1473–1480
- Sun Y, Li LL, Huang XS, Duan CY (2013) Short-term wind speed forecasting based on optimized support vector machine. In: *Applied Mechanics and Materials*, 300:189–194, Trans Tech Publications Ltd
- Tian Z (2020) Short-term wind speed prediction based on LMD and improved FA optimized combined kernel function LSSVM. *Eng Appl Artif Intell* 91:103573
- Tian Z, Ren Y, Wang G (2019) Short-term wind speed prediction based on improved PSO algorithm optimized EM-ELM. *Energy Sources A Recov Util Environl Effec* 41(1):26–46
- Vinothkumar T, Deeba K (2020) Hybrid wind speed prediction model based on recurrent long short-term memory neural network and support vector machine models. *Soft Comput* 24(7):5345–5355
- Wang C, Wu J, Wang J, Hu Z (2016) Short-term wind speed forecasting using the data processing approach and the support vector machine model optimized by the improved cuckoo search parameter estimation algorithm. *Math Probl Eng* 2016:1–17
- Wang J, Yang Z (2021) Ultra-short-term wind speed forecasting using an optimized artificial intelligence algorithm. *Renew Energy* 171:1418–1435
- Wang J, Zhou Q, Jiang H, Hou R (2015) Short-term wind speed forecasting using support vector regression optimized by cuckoo optimization algorithm. *Math Probl Eng* 2015:1–13
- Wu Q, Lin H (2019) Short-term wind speed forecasting based on hybrid variational mode decomposition and least squares support vector machine optimized by bat algorithm model. *Sustainability* 11(3):652
- Xiang L, Deng Z, Hu A (2019) Forecasting short-term wind speed based on IEWT-LSSVM model optimized by bird swarm algorithm. *IEEE Access* 7:59333–59345
- Xiao L, Shao W, Jin F, Wu Z (2021) A self-adaptive kernel extreme learning machine for short-term wind speed forecasting. *Appl Soft Comput* 99:106917
- Zeng J, Qiao W (2012) Short-term wind power prediction using a wavelet support vector machine. *IEEE Trans Sustain Energy* 3(2):255–264
- Zhang L, Zhou W, Jiao L (2004) Wavelet support vector machine. *IEEE Trans Syst Man Cyber Part B* 34(1):34–39
- Zhang T, Zhou ZH (2014) Large margin distribution machine. In: *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 313–322
- Zhang Y, Pan G, Chen B, Han J, Zhao Y, Zhang C (2020) Short-term wind speed prediction model based on GA-ANN improved by VMD. *Renew Energy* 156:1373–1388
- Zhang Z, Ye L, Qin H, Liu Y, Wang C, Yu X, Yin X, Li J (2019) Wind speed prediction method using shared weight long short-term memory network and Gaussian process regression. *Appl Energy* 247:270–284
- Zhou XY, Ye YZ (2006) Application of wavelet analysis to fault diagnosis. *Control Eng China* 13(1):70–73

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.