Mathematical Biosciences and Engineering

MBE, 22(9): 2269–2299.
DOI: 10.3934/mbe.2025083
Received: 28 May 2025
Revised: 02 July 2025
Accepted: 09 July 2025
Published: 16 July 2025

https://www.press.com/journal/MBE

Research article

Probabilistic machine learning-based forecasting of wind speed uncertainty using adaptive kernel density estimation

Rami Al-Hajj*

College of Engineering and Technology, American University of the Middle East, Kuwait

* **Correspondence:** Email: rami.alhajj@aum.edu.kw.

Abstract: Short-term wind speed forecasting is essential for enhancing the efficiency and dependability of wind renewable energy installations. Although often used, conventional point predictions generated by machine learning techniques frequently fail to accurately capture the natural uncertainty associated with wind speed variation. Modeling this type of uncertainty is crucial for providing credible information as the level of uncertainty increases. Prediction intervals (PIs) offer a probabilistic framework for quantifying forecast uncertainty. This paper presents a hybrid forecasting methodology that combines support vector regression (SVR) with adaptive kernel density estimation (AKDE) to estimate wind speed prediction intervals over various short-term horizons (10, 30, 60, and 120 minutes). In contrast to standard kernel density estimation (KDE), which employs a uniform bandwidth and may overlook local data attributes, the adaptive KDE approach adjusts the bandwidth in accordance with the local distribution of forecast errors, thereby facilitating more precise and locally tuned uncertainty quantification. The efficacy of the proposed SVR-AKDE model is evaluated against conventional KDE-based interval estimation. Outcomes are assessed by recognized PI quality indicators, including prediction interval coverage probability (PICP), prediction interval normalized average width (PINAW), and coverage width-based criterion (CWC). Simulation findings confirm the efficacy of our approach and demonstrate that the SVR-AKDE-based PI forecasting consistently provides enhanced coverage and narrower widths compared to traditional KDE. This approach provides a comprehensive solution for short-term wind speed forecasting with quantifiable uncertainty, therefore enhancing its application in operational wind energy control.

Keywords: wind speed forecasting; renewable energy; support vector regressors; prediction intervals; probabilistic energy forecasting; adaptive kernel density estimator; AKDE; SVR

1. Introduction

Nowadays, wind energy is sustainable and holds particular promise for enhancing smart grid stability during seasonal periods of the year [1–3]. Wind speed (WS) is a key factor in the wind energy sector; therefore, understanding and modeling WS variation is crucial for optimizing wind power (WP) production. WS changes randomly and fluctuates over time depending on the geographical environment and climate change [4]. This variability is due to non-linear and non-stationary features of the WS time series data [5–8]. A slight change in WS can cause a significant shift in energy production since the power produced by a wind turbine is proportional to the cube of the WS [9]. Therefore, reliable forecasting of WS provides adequate knowledge to make informed choices about the strength of WP in a given area.

Numerous WS prediction techniques have been developed to offer point forecasts, also known as deterministic forecasts, which do not adequately reflect the uncertainty and volatility of WS [10–12].

Consequently, it was considered necessary to identify a method for quantifying the uncertainty factor, such as PIs. Based on a given confidence level reflecting the model's reliability, PIs offer a range within which the measured WS is expected to fall [13].

Because the WS fluctuation is arbitrary, it can be statistically examined. Among the several techniques available to simulate WS is the probability density function (PDF). The PDFs reflect the characteristics of random variables for statistical analysis, including distribution shape, scaling, extreme values, and pertinent factors. In this study, PDFs are applied to analyze wind speed prediction errors, which is a significant foundation for calculating PIs of wind speed.

Numerous research suggest employing non-parametric techniques, such as KDE, to estimate the PDF of WS prediction errors. Typically, this technique relies on a constant bandwidth around error points in a dataset of estimates and may not accurately reflect local density changes in the data. Actually, in a probability density distribution, there is a continuous fluctuation in data density, so using a single bandwidth for the entire distribution is not always satisfactory. In this study, we propose a new approach to address these challenges: an adaptive kernel density estimation (AKDE) method based on nearest neighbor estimation (NNE) that accurately characterizes the error probability density distribution of WS estimations. The AKDE offers a more precise and adaptable analysis of WS errors by adjusting the local bandwidth based on local data density. This approach offers a significant improvement over conventional distribution methods. These improvements will enable a more accurate capture of local density fluctuations in WS prediction errors.

1.1. Related works

PI approximation estimates the range of potential future changes in a value when the original data shows irregular fluctuations. Practically, interval prediction provides upper and lower boundaries of predictions at given confidence levels. That provides decision-makers with sufficient information on uncertainty for more accurate decision-making [14–16]. In WP applications, PI estimation is particularly beneficial for wind farm operation and maintenance engineers, enabling them to plan their tasks and formulate reasonable scheduling procedures [17].

In recent years, machine learning (ML) models have become prominent techniques for both point and interval WS predictions [11,12,18]. WS PI estimation systems have recently employed several ML models and architectures. Most of the recently proposed models are either ensemble models [19–22], combined models [23–25], hybrid models [26–29], or deep learning (DL) models [30–32]. Ensemble

models combine parallel predictors of statistical regressors and/or ML models. Hybrid models are composed of a combination of optimization techniques and machine learning (ML) models.

The authors in [14] combined least support vector machines with the multi-objective ant lion optimization algorithm to estimate hourly prediction intervals of WS. The authors in [18] introduced a framework comprising five single ML predictors of the following types: support vector machine (SVM), extreme learning machine (ELM), deep belief network (DBN), long-short term memory (LSTM), and convolutional neural network (CNN) for both WP deterministic and probabilistic interval prediction. The deterministic point forecasting outputs of the individual predictors are combined using the critical weight method. Then, the nonparametric KDE method is applied under different confidence levels to estimate the PIs for each point prediction. A new probabilistic framework based on explainable neural networks (NNs) was introduced by Huang et al. [16]. The PIs around deterministic forecasts are provided by statistical synthesis of WS forecasting uncertainties via the KDE method [16]. The system presented in [19] introduced ensemble learning for deterministic and interval WS forecasting. The introduced approach consisted of mixed-frequency modeling and applied a multi-objective optimizer to enhance the performance of independent ELM forecasting models. An ensemble learning approach is introduced in [20]. The model applied a multi-objective optimization approach to combine NNs and SVRs. The authors in [21] applied the multi-objective dragonfly algorithm to combine NNs and statistical models. In a preprocessing phase, the authors applied a fuzzy information granulation method to reduce the dimension of WS records. The authors in [22] examined several statistical and ML prediction methods according to modal characteristics. The authors applied several optimization algorithms to improve the overall prediction capabilities. A regression bi-directional LSTM ensemble was proposed in [23] for estimating uncertainty in WS probabilistic forecasting. The authors in [24] introduced multi-objective ensemble learning with variational mode decomposition for short-term WS PI estimation. This hybrid approach combined a multi-objective artificial hummingbird algorithm with statistical and gated recurrent forecasting methods specifically for WS PI forecasting. The authors in [26] applied gated recurrent units (GRUs) to estimate WS PIs based on prediction errors. In a preprocessing phase, a variational mode decomposition is used to decompose the complex WS time series records into simplified modes. The prediction errors are aggregated to compute the final PIs' widths. The particle swarm optimization algorithm was used to determine the optimal weights for the prediction.

In [27], a hybrid model that applies an auto-encoder-based feature extractor was proposed. The suggested approach used bidirectional LSTM (bi-LSTM) models for short-term WS PI forecasting. The simulation results indicated that feature extraction through an auto-encoder helps produce narrow PIs with quality PI coverage. In [28], a hybrid framework that combines a radial basis function (RBF) model, the fast correlation-based filter (FCBF) algorithm, and the Fourier distribution to construct a WS PI model. The FCBF algorithm was applied to filter the influences that affect changes in wind speed and direction. Then, an enhanced version of particle swarm optimization (PSO) was applied to optimize the RBF model. The Fourier distribution was applied to fit the probability distribution of estimation errors used for estimating the WS PIs. The work described in [29] proposed a hybrid model that combines a set of statistical and ML models, a modified multi-objective tunicate algorithm (MMOTA), and a quantile regression (QR) tool for deterministic and probabilistic interval forecasts of WS. The authors in [30] applied empirical mode decomposition to extract the linear component of the WS data series. Next, they used an autoregressive integrated moving average (ARIMA) model and an MLP trained by back-propagation supervised learning to find the deterministic prediction points. Finally, an enhanced Markov chain model was employed to achieve the uncertainty analysis of WS errors and estimate the PIs. The work in [31] introduced a PI forecasting model of type temporal

convolutional networks (TCN) to estimate WS PIs. The structure of the model consists of a TCN layer as the input layer, multiple fully connected hidden layers using the $tanh$ activation function, and an end-to-end sorting layer at the output.

Recent improvements in probabilistic forecasting have integrated DL models with quantile-based loss functions to improve uncertainty estimation. Recurrent models, such as LSTM and GRU, have been employed with the pinball loss function to provide PIs that exhibit enhanced sharpness and calibration, as shown in crude oil price forecasting [32]. Similarly, transformer-based models such as the multi-granularity auto-former have demonstrated efficacy in capturing temporal relationships for long-term power load point and probabilistic forecasting tasks [33]. Moreover, hybrid architectures that integrate Fourier features and quantile regression have been effectively utilized for probabilistic environmental forecasting tasks [34].

Recently, several studies have introduced conformal prediction methods for probabilistic forecasting of wind speed [35,36]. These approaches utilize DL models (e.g., LSTMs and DNNs) that may be combined through a stacking ensemble and fine-tuned using conformal prediction methods [35]. Conformal prediction methods generally produce distribution-free, accurate prediction intervals based on the concept of exchangeability. Additionally, they leverage data-driven calibration and theoretical coverage assurances, making them ideal for complex, high-dimensional forecasting problems where deep structures can extract nonlinear temporal patterns.

Conformal approaches provide robust theoretical guarantees and adaptable integration with complex models; yet, they can be computationally intensive and may exhibit broad intervals in areas with significant data variability. Conversely, nonparametric KDE-based methods incorporating ML point predictors are simpler, faster to train, and more appropriate for smaller or moderate-sized datasets, where probabilistic calibration can be achieved through adaptive density estimation.

1.2. Objectives and contribution

Most of the solutions mentioned above employ nonparametric distribution estimation approaches, such as KDE, to estimate the PDF of the prediction errors. The KDE smooths data points using kernel functions, such as the Gaussian function, and a bandwidth, which represents the degree of smoothing. These two factors are critical in estimating the distribution provided by the KDE method, as they determine the smoothness of the PDF estimate by affecting the kernel's width [37]. The approaches mentioned above often rely on a fixed bandwidth and may not completely capture local density variations of the data. Actually, in a probability density distribution, there is a continuous variation in data density, so adopting a single fixed bandwidth for the entire distribution may not be satisfactory.

The work in [38] has shown that kernel functions are less critical than bandwidth selection. A small bandwidth leads to a PDF estimate that is noisy because each data point significantly influences the forecast, resulting in clear high and low points. Consequently, the PDFs may not represent the data's general empirical density well (under-smoothing). On the other hand, a bandwidth that is too large results in a PDF estimate that is too smooth, losing significant details and structural characteristics of the data distribution. The PDF estimate can also be too flat compared to the empirical density, failing to distinguish outliers in the data (over-smoothing) [39,40].

In this study, we propose a new approach to address these challenges: an AKDE method based on NNE to characterize the error probability density function of WS prediction. The AKDE provides more accurate and flexible WS prediction error distribution by adjusting the local bandwidth based on data density. This method offers a notable advancement over standard distribution models by allowing varying bandwidths according to data density. That will help in better capturing the local density

variations of WS prediction errors. The AKDE is viewed as a bandwidth optimization method that replaces the fixed-bandwidth (global-bandwidth) KDE distribution. This approach allows for different bandwidths among different data points, making the KDE distribution better adapted to the empirical data's distribution structure.

The proposed method in this work enhances the descriptive power of the distribution of prediction errors. Statistical measures are used to compare the performance of the proposed model in predicting short-term WS forecasting PIs with that of the conventional KDE approach. The simulations demonstrate that the AKDE coupled with known SVR models provides a better fit, thereby highlighting its effectiveness in capturing the complex features of the WS prediction error distribution.

The main contributions of this work are summarized as follows:

1. Propose a novel probabilistic approach to estimate PIs for short-term forecasting of WS using SVR models and AKDE on a new time series dataset.
2. The AKDE method based on NNE is applied to describe the WS prediction error probability distribution accurately. The obtained PDFs of point prediction errors enhance the estimation of PIs for WS forecasting compared to those obtained using the standard KDE method on the examined dataset.
3. The AKDE-based probabilistic model is validated on several different resolutions of WS time series data (10, 30, 60, and 120 minutes) without setting a hypothesis about the distribution of the WS prediction errors in advance.
4. The overall performance of the obtained SVR-AKDE-NNE hybrid model is compared to that of others based on the KDE method, using similar deterministic prediction models. Additionally, the simulations demonstrated an overall consistent improvement in the performance of the proposed model as the resolution of data records decreased.

The obtained simulation results demonstrated the effectiveness of the introduced framework. Three evaluation metrics, specifically, the prediction interval coverage probability (PICP), the prediction interval normalized average width (PINAW), and the coverage width-based criterion (CWC), have been chosen to evaluate the WS PI forecasting framework.

The subsequent sections of this work are structured as follows: Section 2 presents the methodology and the techniques employed. Section 3 details the structure of the proposed approach. Section 4 presents and discusses the simulation setup and results. Lastly, in Section 5, we provide the conclusion and implications of this study.

2. Related methodology

2.1. Support vector regressors

SVR models are ML methods to address regression problems. Currently, they are considered competing methods for approximating continuous functions [41–43]. SVRs are a type of SVM intended for modeling regression tasks.

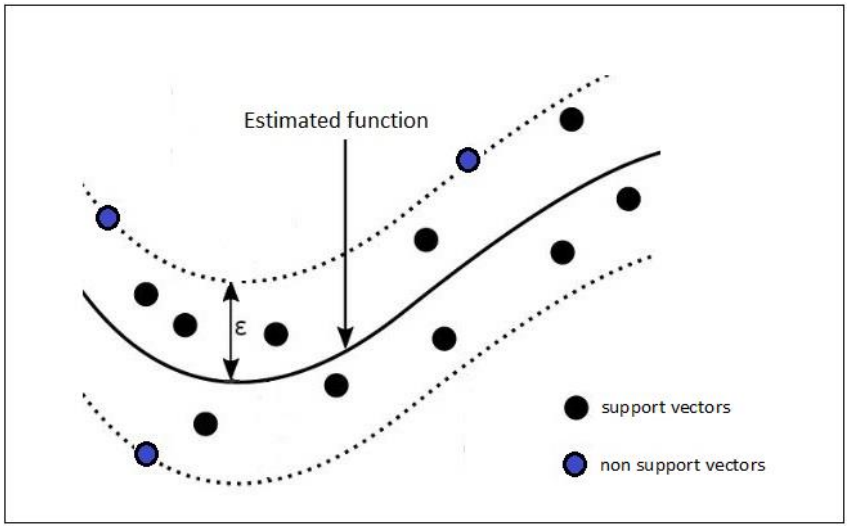


Figure 1. SVRs with support vector points and the epsilon parameter.

SVRs aim to identify a flexible boundary around the target function to be approximated, encompassing as many data points as feasible within a set margin of tolerance provided by the parameter epsilon (ε) [41]. In SVR models, support vectors are the critical data points that lie on or outside the boundaries of the epsilon tube defined by the parameter epsilon ε around the predicted function, as shown in Figure 1. These points only decide and influence the ultimate position and orientation of the regression function. This approach renders the model less susceptible to outliers [42].

SVR models utilize kernel functions to manage non-linear interactions among variables. Frequently utilized kernels comprise the polynomial kernel, radial basis function (RBF) kernel, and sigmoid kernel [41]. These kernels transform the input feature space into a higher-dimensional space, allowing the SVR to identify complex patterns and associations among datasets.

SVRs are helpful because they are resilient to outliers and can represent non-linear correlations between input variables and intended outputs. At the same time, their computing complexity remains independent of the dimensionality of input data vectors.

2.2. Kernel density estimation (KDE)

KDE is a non-parametric estimating technique commonly employed for data fitting when the underlying probability density function (PDF) is unavailable [44,45]. The KDE uses a kernel function and a bandwidth to find an elevation curve for each observation point in the data. In other words, it consists of positioning a smooth kernel function at every data point. It subsequently aggregates these kernels to formulate a continuous approximation of the underlying probability density function of the data points' distribution. The PDF of a random variable quantitatively represents the probability of that variable taking a particular value. This statistical method does not pre-suppose any specific distribution of the data. Our research employs the KDE method to develop a stochastic distribution model of the WS predicting errors. We assessed the performance indicators of the wind speed by examining the distributions of anticipated WS estimation errors generated by the ML prediction models.

The PDF is generally computed as follows:

Let $W = \{w_1, w_2, w_3, \dots, w_n\}$ denotes a set of n sample points representing WS prediction errors. Then, the PDF of the WS errors is implemented as follows:

$$\hat{p}(w,H) = \frac{1}{n*H}\sum_{i=1}^{n} k\left(\frac{w-w_i}{H}\right) \quad (1)$$

where $K(w,H)$ denotes the kernel function, H indicates the bandwidth parameter that regulates the smoothness of the estimation, and w_i is the i^{th} sample of the WS prediction error.

Common kernel functions utilized in the KDE approach include the Gamma kernel, uniform kernel, and Gaussian kernel, among others [44,45]. In this work, the Gaussian kernel function k was chosen for estimating the probability density function of the error distribution using the KDE method. The function $k(\,)$ is defined as:

$$k\left(\frac{w-w_i}{H}\right) = \frac{1}{\sqrt{2\pi}} e^{\left[-\frac{(w-w_i)^2}{2H^2}\right]} \quad (2)$$

Hence, the estimated PDF is represented as:

$$\hat{p}(w,H) = \frac{1}{\sqrt{2\pi}\,nH}\sum_{i=1}^{n} e^{-\frac{1}{2}\left(\frac{w-w_i}{H}\right)^2} \quad (3)$$

where H represents the bandwidth parameter that defines the extent of the distribution interval of the prediction error, while w_i is the i^{th} sample of the WS prediction error.

The performance of KDE is primarily contingent upon the selection of the bandwidth. Sometimes, bandwidth is determined through experiential comparison of PDFs [46]. Nevertheless, these comparison-based methods are laborious, time-consuming, and do not necessarily yield the most accurate estimations.

2.3. Adaptive bandwidth for kernel density estimation (AKDE)

Due to the significant impact of bandwidth on the performance of KDE distributions, numerous studies have been conducted to determine the optimal bandwidth for KDE-based distribution, as reported in the literature [47,48].

The AKDE is a bandwidth optimization technique that substitutes the global bandwidth KDE distribution (fixed-bandwidth) with a variable one. This method allows for varying bandwidths among distinct data points, thereby enhancing the KDE distribution's alignment with the underlying structure of the empirical data. Numerous studies have examined adaptive bandwidth density estimation in KDE, which is regarded as superior to the global bandwidth [49,50]. The primary objective is to ensure that the bandwidth for the KDE distribution is adaptable based on the local density of the observed data. The AKDE technique emphasizes modifying the bandwidth to suit local data. That means the bandwidth is assessed locally. In practice, the NNE approach is commonly employed to evaluate the local density surrounding each data point by measuring the distance to the nearest neighboring points [51]. This local density is used to determine the bandwidth size of the AKDE distribution. Regions with short neighboring distances represent high density; hence, the bandwidth for KDE will be sufficiently small to yield precise estimates and more accurate information. Conversely, in regions with greater nearest-neighbor distances, indicating reduced density, the bandwidth will be increased to ensure smoother, less noisy estimates.

2.3.1. NNE technique

The NNE method utilizes a specific number of nearest data points to approximate the PDF, rather than using the entire dataset. The examined WS prediction error data has a dimensionality $d = 1$. Each particular observation point w has a Euclidean distance to the k^{th} nearest neighbor in the dataset

$\{w_i\}_{i=1}^{n}$, which we name here $r_k(w)$. Let $L_k(w)$ denote the length surrounding $r_k(w)$ on both sides of w, so $L_k(w)$ can be represented as follows:

$$L_k(w) = 2\, r_k(w) \tag{4}$$

Let the density at w be denoted by $p(w)$, and then for a sample of n data points, there are $n\, p(w)\, L_k(w)$ observations in the interval around w. Then, $\hat{p}(w)$ can be estimated as:

$$\hat{p}(w) = \frac{k}{n\, L_k(w)} = \frac{k}{2\, n\, r_k(w)} \tag{5}$$

The parameter k operates as a "frame" for local data density; higher values of k yield smoother estimates, while lower values of k result in less smooth estimates. Around each observation point *w*, every data point within distance $r_k(w)$ from w contributes an amount of $\frac{k}{2\, n\, r_k(w)}$ to the density estimate. Therefore, the density estimation at w looks like a kernel estimation employing the indicator function of a unit interval, scaled by the length of the unit interval to guarantee the kernel integrates to 1, with a bandwidth of $r_k(w)$. This bandwidth directly affects the NNE, making it analogous to a KDE with spatially varying kernel width. The NNE approach can be expanded to incorporate complex kernels beyond basic indicator functions. The application of a kernel K facilitates a generalized form of the NNE [52]. Therefore, the estimated PDF is represented as:

$$\hat{p}_k(w) = \frac{1}{n r_k(w)} \sum_{i=1}^{n} k\, [\frac{w - w_i}{r_k(w)}] \tag{6}$$

This technique determines the closest neighbor of a specified data point and typically employs this neighbor's value for estimation.

2.3.2. Implementing AKDE using NNE

To estimate the adaptive bandwidth for the WS prediction error distribution using NNE, we included the estimation of local bandwidth $b(w_i)$ for each data point w_i. Such bandwidth $b(w_i)$ is proportional to the distance from w_i to its k^{th} nearest neighbor. A smoothed pilot density estimate $\hat{p}(w)$ is the baseline for selecting these bandwidths.
The scaling factor L_i for each data point's bandwidth is defined by:

$$L_i = [\frac{\hat{p}(w_i)}{g}]^{-\alpha} \tag{7}$$

where g, the geometric mean of the pilot densities, is given by:

$$g = exp\left[\frac{1}{n} \sum_{i=1}^{n} log\, (\hat{p}(w_i))\right] \tag{8}$$

The sensitivity parameter α controls the degree of bandwidth adaptation to changes in the pilot density estimate. The adaptive bandwidth for each data point w_i is considered as follows:

$$b(w_i) = h\, L_i \tag{9}$$

where h is the initial bandwidth. Thus, the estimate PDF using AKDE is calculated as follows:

$$\hat{p}(w) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{b(w_i)}\, k\left[\frac{w - w_i}{b(w_i)}\right] = \frac{1}{nh}\sum_{i=1}^{n} \frac{1}{L_i}\, k\left[\frac{w - w_i}{n\, L_i}\right] \tag{10}$$

The sensitivity parameter α in Eq (7) helps in adjusting the bandwidth in response to local density variations. In general, $\alpha = 0.5$ represents a balanced setup [53].

KDE utilizing the NNE technique necessitates the identification of the number of neighbors and their proximity to compute density. In this context, L_i quantifies the impact of the data point for implementing the KDE method, where tuning constants such as h and α are necessary to establish the initial estimate. Figure 2 illustrates the AKDE approach, which includes the recommended adaptive bandwidth selection technique.

After importing the WS prediction error dataset, the local density is computed by measuring the distance to the nearest neighbor points to find the neighborhood length and the density of each data point. Then, the adaptive bandwidth is established according to local density by computing geometric mean values and scaling factors to modify the bandwidth for each point. Subsequently, the PDF is calculated using the adaptively modified bandwidth for each data point. The model is then evaluated and refined to get optimal accuracy.

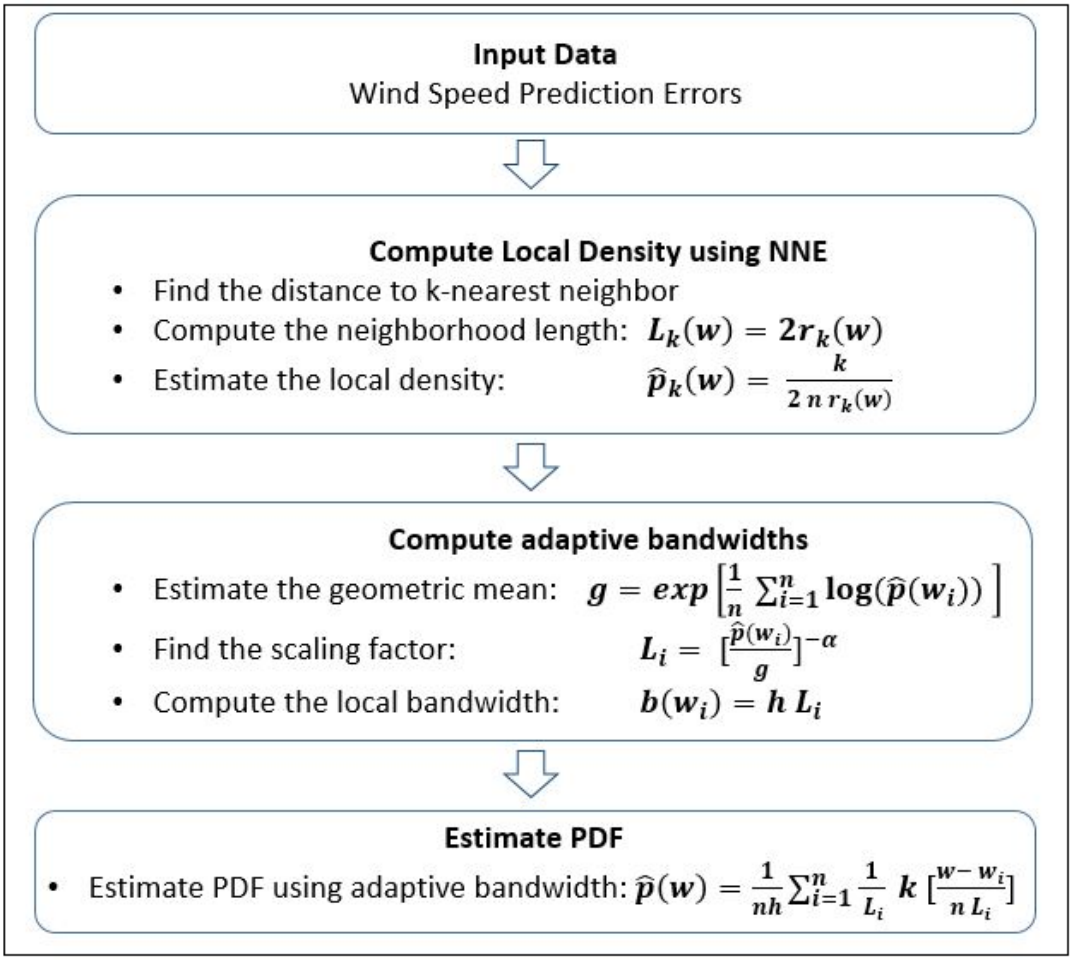


Figure 2. The general steps of the AKDE method which includes the adaptive bandwidth selection technique.

3. Proposed approach

This section outlines the general structure of the hybrid probabilistic approach for estimating the WS PIs. Figure 3 illustrates the overall structure of the proposed approach. The suggested model utilizes the weather parameters in the WS data records at time t as inputs. The output results include the lower bound $\boldsymbol{L_{t+1}^{\alpha}}$ and upper bound $\boldsymbol{U_{t+1}^{\alpha}}$ of the PIs at time $\boldsymbol{t+1}$ with the associated confidence level $100(1-\alpha)\%$, where α does not designate the sensitivity parameter mentioned in Eq (7).

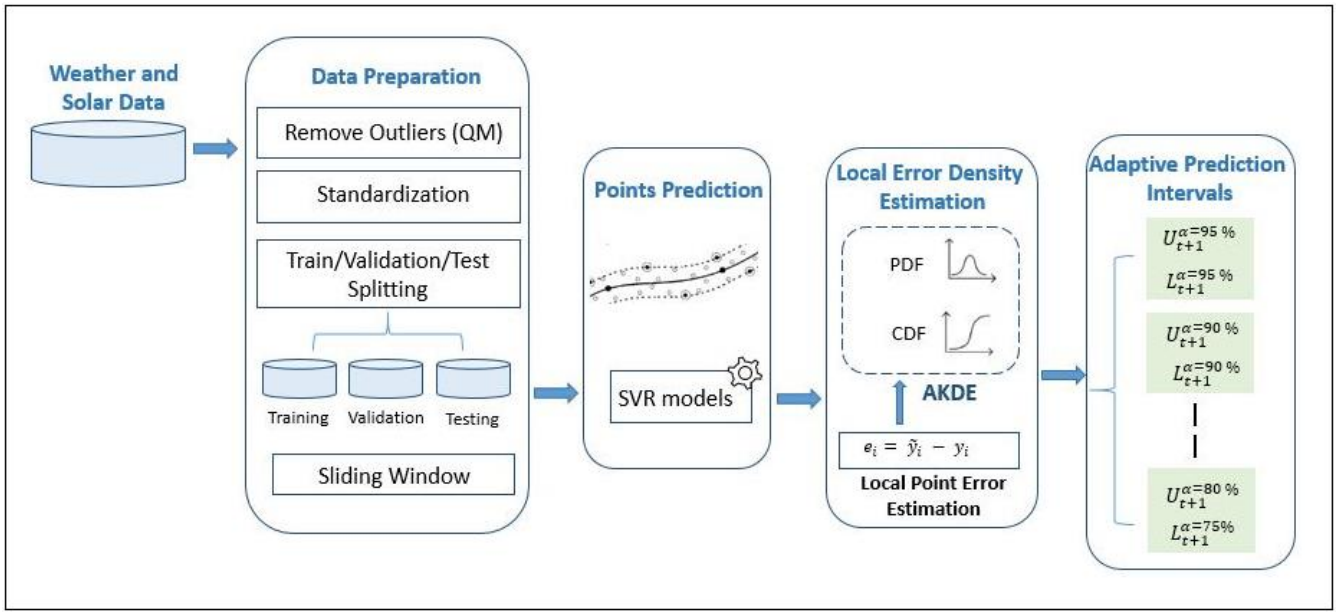


Figure 3. The general structure of the WS adaptive PI forecasting model.

As illustrated in Figure 3, the general structure of the WS prediction approach consists of four steps. The first step involves processing and preparing the data records for deterministic prediction. That includes removing outliers using the QM method [54] and then data normalization. The second step consists of the SVR-based point prediction module. In the third step, the AKDE method is applied to estimate the local distribution of prediction errors at each point provided by the point prediction module. The fourth step estimates the adaptive PIs around each point prediction. The following subsections detail each module of the proposed approach.

3.1. Data processing module

The data pre-processing and preparation module includes three primary steps:

(1) Outlier Removal: Outliers are eliminated using the QM method.

(2) Data Scaling and Normalization: Data normalization through *MinMax* scaling aims to rescale features and variables to a uniform range, typically between 0 and 1. This is achieved by subtracting the minimum value from each data point and subsequently dividing by the range $(maximum - minimum)$, as illustrated in Eq (11).

$$p_{scaled} = \frac{p - min(P)}{max(P) - min\,(P)} \tag{11}$$

where p denotes the value of parameter P, and $min(P)$ and $max(P)$ designate the minimum and

maximum values of the parameter P, respectively.

(3) Sliding Window: This technique enables the transformation of a prediction task into a regression task. The sliding window reorganizes the time series data by utilizing the weather parameters and the WS of the k^{th} record as the input vector, while the wind speed of the $(k+1)^{th}$ record serves as the corresponding goal output.

3.2. Point prediction module

The point prediction module consists of an SVR-based model for deterministic wind speed forecasting. In this study, we examined the performance of the SVR predictor on several resolutions of data records, namely the 10-, 30-, 60-, and 120-minutes resolutions. Therefore, we developed four SVR-based models for the short term deterministic prediction of WS. All analyzed SVR models utilize RBF kernels, which are efficient at simulating non-linear connections between target and input variables [44,45]. The primary hyperparameters that we used for all the SVR models are:

- Kernel: rbf
- C (Regularization parameter): 1.0
- Epsilon (ε-insensitive loss function margin): 0.3
- Gamma (kernel coefficient): 'scale', which is a default setting in the employed framework that we describe in Section 4.3

We chose these hyperparameters after applying a grid search using 5-fold cross-validation on the training set. We explored a range of values for C (like [0.1, 1, 10]), epsilon (like [0.1, 0.2, 0.3]), and gamma (like ['scale', 'auto', 0.01, 0.1, 1]) to find a good compromise between accuracy and generalization. The best-performing combination based on validation performance was selected.
The four acquired deterministic prediction models are enumerated as follows:

Table 1. The SVR-based predictors for point predictions of WS.

Model	Description
SVR-RBF-10mins	SVR model with rbf kernel trained on 10 minutes resolution data
SVR-RBF-30mins	SVR model with rbf kernel trained on 30 minutes resolution data
SVR-RBF-60mins	SVR model with rbf kernel trained on 60 minutes resolution data
SVR-RBF-120mins	SVR model with rbf kernel trained on 120 minutes resolution data

3.3. Interval prediction module

The fourth step of the proposed methodology employs the KDE technique to estimate the PDF that aligns with the regression errors of WS point predictions produced by the SVR models. This helps in estimating the PIs of WS. A significant limitation of conventional KDE is its reliance on a global bandwidth parameter, which assumes a uniform smoothing over the entire data domain. Particularly in areas where the prediction errors show non-uniform density, this assumption could result in underestimating or overestimating local changes. The AKDE technique has been proposed to address this issue by locally adjusting the bandwidth based on the data density. The NNE technique facilitates the computation of local bandwidth at each data point by assessing the distance to the k-nearest neighbors of that point. This capability enables the AKDE method to accurately reflect local variations in the probability distribution of prediction errors, hence improving the estimate of prediction intervals

with highly fluctuating WS data. In this study, we compare the performance of the two methods: the conventional KDE and the adaptive KDE. Both methods were evaluated for each of the four examined data records' resolutions.

3.3.1. KDE models

The following modeling was conducted for each of the SVR models specified in Section 3.2; in other words, each of the following KDE applications was repeated for each of the four examined data resolutions described above.

After training the SVR using a training dataset with a particular resolution r, the errors of the deterministic predictions of the WS were evaluated using the validation dataset. Consequently, a sequence of distinct prediction errors E_r was generated as follows:

$$E_r = \{e_r^1,\ e_r^2, \ldots., e_r^R\} \tag{12}$$

where R represents the number of validation dataset records related to the resolution r.

The KDE method is then applied on E_r to estimate the related PDF. The calculation is applied as follows:

$$\hat{f}(s, h_r) = \frac{1}{\sqrt{2\pi}\, n h_r} \sum_{i=1}^{R} e^{-\frac{1}{2}\left(\frac{s - e_r^i}{h_r}\right)^2} \tag{13}$$

Equation (13) estimates the PDF of the prediction errors provided by the SVR model trained and examined on the datasets with resolution r.

The h_r parameter in Eq (13) represents the bandwidth that determines the width of the distribution interval of the prediction error. In general, selecting an appropriate value of the bandwidth parameter h_r controls the smoothness of the estimation. This, in turn, helps minimize the estimation error [44]. In this study, we employed the trial-and-error method to select an optimal bandwidth value for each error dataset based on the best performance on the validation set in terms of PICP and PINAW. This is a common method in nonparametric density estimation when a closed-form solution in predictive error modeling cannot determine an ideal bandwidth. The same validation-driven selection process was applied uniformly to all KDE-based models, ensuring a fair comparison. In contrast, the AKDE method adapts bandwidths locally and does not rely on a global trial-and-error search, which is precisely what we need to enhance. Consequently, the comparison is unbiased, but it instead underscores the advantages of data-driven adaptability over global manual tuning.

The obtained PDF is used to compute the cumulative distribution function (CDF). The CDF represents the probability that a random variable, in this case, the point prediction error, will take a value less than or equal to a given value [44,45]. In practical terms, integrating the PDF of the point prediction error distribution enables the CDF to accumulate probabilities up to a specific value, facilitating the estimation of the PIs.

The developed model computes the CDF to determine the fluctuation range of the prediction error, given a PI nominal confidence (PINC) set at 100(1-α)%. The range of fluctuation can be articulated in the following manner:

$$I_\alpha = [F(\alpha), F(1-\alpha/2)] \text{ , where } \quad 0 < \alpha < 1 \tag{14}$$

Consequently, the estimated PI associated with the predicted target of WS at a 100(1-α)% confidence level is defined as follows:

$$\left[\hat{L}_i^{(\alpha)}(x_i), \hat{U}_i^{(\alpha)}(x_i)\right] = [\hat{y}_i^{(\alpha)} - F(\alpha)\,, \hat{y}_i^{(\alpha)} + F(1 - \alpha/2)] \tag{15}$$

where $\hat{y}_i^{(\alpha)}$ is the deterministic prediction of WS related to the input x_i and the confidence level $100(1-\alpha)\%$. $F(\alpha)$ designates the value of the related CDF.

3.3.2. Adaptive KDE models

Similar to the KDE process described in 3.3.1, we applied the AKDE method to the prediction errors provided by each of the SVR models introduced in Section 3.2. In other words, each of the AKDE applications is repeated for each of the four examined data resolutions described above.

The same sets of prediction errors E_r described in Eq (12) are used to estimate the PDFs of error distribution using the AKDE method. For each error dataset E_r, we proceed as follows: First, we evaluate the adaptive bandwidth *b(wi)* for each data point w_i using Eqs (7)–(9) with the sensitivity parameter $\alpha = 0.5$.

Thus, the estimated probability density function $\hat{p}_r(w)$ for each resolution r is estimated based on Eq (10) as follows:

$$\hat{p}_r(w) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{b_r(w_i)}\, k\left[\frac{w - w_i}{b_r(w_i)}\right] \tag{16}$$

where r represents the resolutions of data records, namely 10, 30, 60, and 120 minutes. $b_r(w_i)$ is the adaptive bandwidth computed in Eq (9), which is related to resolution r.

In our simulations, the AKDE bandwidth is computed adaptively for each test point using the local error distribution obtained from its *k* nearest neighbors in the validation set. Specifically; we set $k = 10$ for local bandwidth estimation during density scoring. This value was selected based on preliminary validation experiments to balance locality and statistical robustness. As per the initial bandwidth (h), we do not use a fixed initial bandwidth. Instead, we estimate a local bandwidth dynamically as follows:

$$h = 0.5 \times mean(|\ local\ errors\ |) \tag{17}$$

This acts as a smooth scaling factor and was found to be effective in capturing local density variation.

The proposed probabilistic prediction approach consists of three main steps, as depicted in Figure 4. During the initial phase, the training data subset is used to construct the corresponding SVR-based model for point prediction. The optimal version of the trained SVR model for each data resolution is preserved for the next phase. During the second phase, the PDF of the error distribution is computed using the AKDE method. The dataset of prediction errors for each resolution is obtained by examining the trained models using the validation data subset. Subsequently, the acquired error datasets and the estimated PDF functions are used to compute the CDF of the errors associated with the same resolution. The WS prediction models for estimating PIs are statistically built next. In the final phase, the probabilistic framework is evaluated using the testing data subset to determine its efficacy in delivering WS forecasting uncertainties, which are probabilistically represented as a collection of PIs, as demonstrated in Eq (15).

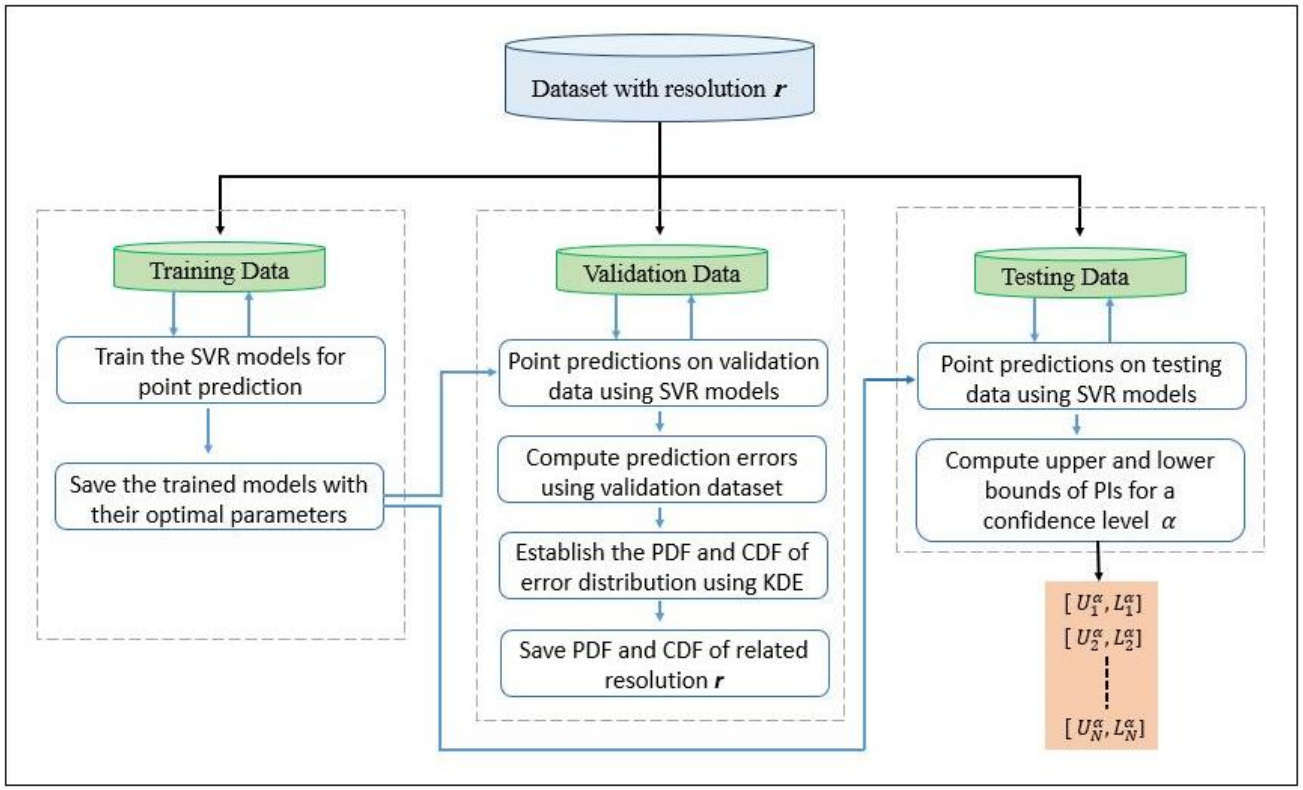


Figure 4. Flowchart illustrating the usage of data subsets in the AKDE probabilistic framework.

4. Simulations and discussion

4.1. Evaluation metrics of prediction

4.1.1. Evaluation metrics of point prediction

The forecast performance of the point prediction models is measured using three statistical evaluation indices that are defined as follows:

- The root mean square error (RMSE) assesses the errors between real values and predicted ones by estimating the average distance of the estimated values from the real ones. This index is as follows:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(H_{p,i}-H_i)^2}{n}} \tag{18}$$

where $\mathrm{H_{p,i}}$ is an estimated output and H_i is a measured value associated to that output.

- The mean square error (MAE) estimates the average value of prediction errors without counting their directions. The formula that computes this index is as follows:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|\mathrm{H_{p,i}} - H_i| \tag{19}$$

- The mean bias error (MBE) calculates the average bias in a model's predictions and designates whether a prediction model tends to under-predict or over-predict the actual values.

This index is evaluated as follows:

$$MBE = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{H_{p,i}} - H_i) \tag{20}$$

- The mean absolute percentage error (MAPE) calculates the average absolute percentage discrepancy between the expected values and the actual ones. It is calculated as follows:

$$MAPE = \frac{1}{n}\sum_{i=1}^{n} \left| \frac{\mathrm{H_{p,i}} - H_i}{H_i} \right| \times 100 \tag{21}$$

For all the above-cited criteria, the lower values indicate a superior model performance.

4.1.2. Evaluation metrics of PI-based forecasting performance

Three criteria of assessment were chosen to examine the performance of the PI forecasting models. These criteria are defined as follows:

- The PICP measures the proportion of times the real values fit inside the PI. Practically, it evaluates the interval accuracy in terms of coverage. The PICP is calculated as follows:

$$PIPC = \frac{1}{N}\sum_{i=1}^{N} c_i \quad , \quad c_i = \begin{cases} 1\,, & p_i \in [L_i, U_i] \\ 0\,, & p_i \notin [L_i, U_i] \end{cases} \tag{22}$$

where N is the whole number of testing samples, and c_i is a Boolean variable that is equal to 1 if the sampling point $p_i \in [L_i, U_i]$, and 0 if $p_i \notin [L_i, U_i]$.

A perfect PICP should nearly match the confidence level of the intervals, say 90% for 90% confidence intervals. This correspondence implies that the PIs correctly reflect the expected percentage of exact data. On the other hand, a low PICP indicates that the PIs are too tight and may not cover the majority of the data. Moreover, a high PICP suggests that the PIs are excessively broad, including an unwarranted amount of uncertainty.

- The prediction interval normalized average width (PINAW) calculates the mean width of the PIs in relation to the data range. The PINAW is calculated as follows:

$$PINAW = \frac{1}{N}\sum_{i=1}^{N} \frac{U_i - L_i}{Y_{max} - Y_{min}} \tag{23}$$

where U_i and L_i denote the upper and lower boundaries of the i^{th} estimated PI, whereas Y_{max} and Y_{min} are the highest and lowest values of the target output in the testing dataset.

A lower PINAW score designates narrow PIs relative to the data's range, indicating more precise predictions. But, to guarantee that tighter intervals do not cause under-coverage, one needs to find a balance between lower PINAW and PICP.

- The coverage width criterion (CWC) combines both the coverage probability PICP and the normalized width of the PI PINAW metrics. It penalizes both low coverage and broad PIs simultaneously. The CWC is calculated as follows:

$$CWC = PINAW(1 + \gamma\, e^{-\eta(PICP - \mu)}) \tag{24}$$

$$\gamma = \begin{cases} 0, PICP \geq \mu \\ 1, PICP < \mu \end{cases}$$

μ designates the target confidence level, e.g., 85%, and η represents the penalty factor that weighs the importance of achieving the target coverage probability. Usually, η is adjusted empirically. Lower values of CWC are ideal, as they indicate good coverage accuracy and narrow PIs.

4.2. Simulation data set and pre-processing

Our simulations were conducted using time series records from the AUMET dataset, which contains historical weather and solar radiation data records. The AUMET dataset is collected at the AUMET weather station at the American University of the Middle East (AUM) in Kuwait [54]. Each

data record of the AUMET dataset consists of the measures of sixteen weather and solar radiation parameters with a 5-minute resolution. Our experiments considered a selection of nine parameters, which are listed in Table 2.

Table 2. The studied weather and solar data features in the AUMET dataset.

Parameter	Description
Air Density	Density of the air (Kg/m^3)
Air Temperature	Temperature at three meters above surface (C^o)
Corrected Wind Direction	Corrected wind direction (deg^o)
Pressure	Air pressure (hPa)
Relative Wind Direction	Relative wind direction (deg^o)
Relative Wind Speed	Wind speed (m/s)
Relative Humidity	Relative humidity (Pct)
Solar Radiation	Quantity of solar radiation (w/m^2)
Surface Temperature	Air temperature at 10 cm altitude above the surface (C^o)

The AUMET dataset contains 97,176 total records. The resolution was lowered from 5 minutes to 10, 30, 60, and 120 minutes sequentially during the preprocessing stage. As a result, we acquired four datasets at lower resolutions. Each of the obtained datasets was split into three parts: a training subset, a validation subset, and a testing subset, as detailed in Table 3. Every dataset's testing subset consists of hold-out records not displayed during training or validation. The training datasets were used to train the WS predictors, while the validation datasets were used to produce point prediction errors on validation records that had never been seen during the training phase. The prediction errors were collected for analysis using the AKDE probabilistic framework to generate the PDFs, and subsequently, the CDFs of the prediction errors.

The outliers in each dataset are eliminated using the QM method. Then, *MinMax* scaling, described in Eq (11), is applied to reduce the difference in the scales of the parameters' values between the training and testing data. An overview of the resulting subsets is provided in Table 3.

Table 3. The acquired datasets for each examined resolution.

Dataset	Subsets	Size
AUMET-10 mins	AUMET_10 mins_Train	33,999
	AUMET_10 mins_Valid	9699
	AUMET_10 mins_Test	4909
AUMET-30 mins	AUMET_30 mins_Train	11,341
	AUMET_30 mins_Valid	3639
	AUMET_30 mins_Test	1222
AUMET-60 mins	AUMET_60 mins_Train	5668
	AUMET_60 mins_Valid	1620
	AUMET_60 mins_Test	810
AUMET-120 mins	AUMET_120 mins_Train	2834
	AUMET_120 mins_Valid	810
	AUMET_120 mins_Test	405

4.3. Point prediction results

We validated the effectiveness of the point prediction phase in the first stage of our framework. Additionally, we evaluated the point prediction performance of the selected SVR model against other regression machine learning models, including the MLP and RF. Table 4 summarizes the point prediction models variants and their configurations. Table 5 presents the simulation results for the examined data resolutions, including the assessment metrics RMSE, MAE, MBE, and MAPE. In all simulations, we used the Scikit-learn framework [55] to implement, train, and validate all the examined point predictors.

Table 4. Point prediction model variants and their configuration parameters.

Model variant	Type	Key configuration parameters
SVR-RBF	Support Vector Regression	Kernel = RBF, C = 1.0, ε = 0.3
MLP-RLU	Multi-Layer Perceptron	Hidden layers = 1, Neurons = 12, Activation = ReLU
RF	Random Forest Regressor	Max depth = 7, Number of trees = 100

Table 5. Statistical scores of deterministic predictions of WS forecasting for resolutions: 10, 30, 60, and 120 minutes. The bolded scores indicate the best for each predictor, whereas the underlined ones indicate the worst. The best scores among all predictors and for a specific metric are highlighted.

Model	Scores			
	RMSE	MAE	MBE	MAPE
SVR-RBF-10 mins	0.371	0.2957	0.2856	42.0145
SVR-RBF-30 mins	0.2916	02277	0.2151	36.5513
SVR-RBF-60 mins	0.2334	0.1851	0.1665	33.3656
SVR-RBF-120 mins	**0.1511**	**0.1189**	**0.0482**	**28.6989**
MLP-RLU-10 mins	0.3710	0.2957	0.2856	**42.0145**
MLP-RLU-30 mins	0.6771	0.6789	0.6783	61.9751
MLP-RLU-60 mins	0.5105	0.5709	0.5704	59.7794
MLP-RLU-120 mins	**0.2174**	**0.1756**	**-0.0896**	61.5513
RF-10 mins	0.4831	0.3967	0.2746	**43.0246**
RF-30 mins	0.5771	0.4677	0.4771	62.8651
RF-60 mins	0.3206	0.4608	0.4504	59.4794
RF-120 mins	**0.2381**	**0.2564**	**0.1233**	60.6713

The bolded scores in Table 4 indicate the best for each examined predictor, whereas the underlined ones indicate the worst. The best scores among all predictors and for a specific metric are highlighted.

The SVR models overperformed all the other ML predictors for most of the examined resolutions. The SVR model with a resolution of 120 minutes showed the best performance scores for all evaluation metrics. Furthermore, the SVR models demonstrated a consistent improvement in performance as the resolution increased, whereas the other models did not exhibit such consistency.

4.4. *Statistical hypothesis testing of the results*

Statistical hypothesis testing, specifically paired sample t-tests, was performed to compare the actual values with the predicted values obtained through several models: SVR, MLP, and RF at a 30-minute resolution. The analysis results of the statistical hypothesis testing on the paired sample t-tests are summarized in Table 6.

Table 6. Results of statistical hypothesis testing. The absolute large values are bolded, whereas the smallest values are underlined.

		Mean	Std. Dev.	Std. Error Mean	95% Confidence Interval of the Difference		z-value	P-value
					Lower	Upper		
Pair 1	Target values vs. SRV outputs	–.00076	.3327	.00420	–.01058	.00832	–.141	**.873**
Pair 2	Target values vs. MLP	**–.02852**	.33881	.00418	–.03874	–.01838	–4.794	.000
Pair 3	Target values vs. RF	–.00597	**.34645**	**.00487**	–.01652	.00339	–1.297	.187
Pair 4	Pred. RF vs. pred. MLP	–.02344	.07489	.00131	**–.03928**	–.02124	**–16.801**	.000
Pair 5	Pred. RF vs. pred. SVR	.00589	.10494	.00149	.00293	.00889	3.879	.000
Pair 6	Pred. MLP vs. pred. SVR	.02789	.11496	.00171	.02468	**.03189**	16.049	.000

The hypothesis tests above evaluate the similarity between the mean target value (actual real value) and the averages predicted by three distinct predictive models: SVR, MLP, and RF, at the 5% significance level. The null hypothesis states that there is no disparity between the means of the models' outputs, while the alternative hypothesis asserts that there are significant differences.

The collected results demonstrate that predictions using the SVR models are not significantly different from the actual real values. Furthermore, the difference between the real mean value and the predicted mean value generated by the SVR models is minimal, highlighting that the SVR models accurately predict the actual value. Similarly, the MLP and RF models demonstrate no significant discrepancy from the actual value, though they display a greater deviation against the SVR models.

4.5. Interval prediction results

In this section, we evaluate the effectiveness of the suggested probabilistic framework in predicting adaptive PIs of WS using the AKDE approach. In the simulations, we estimated the probability density distribution and the corresponding CDFs of prediction errors provided by the AKDE technique and standard KDE for each predictor described in Section 3.2 with varying confidence levels.

For clarity, we evaluated the performance of the AKDE technique in comparison to the KDE by analyzing the prediction errors generated by the SVR models throughout the four examined resolutions and within the established confidence levels. Table 7 illustrates the evaluation metrics of the PI forecasting provided by the analyzed models within confidence levels of 85%, 90%, and 95%, respectively.

Table 7. The scores of the evaluation metrics of the PI forecasting obtained using various PINC. The bolded scores designate the best ones for each resolution. The highlighted CWC scores designate the cases related to the best improvement in CWC scores for each particular PINC.

Model		PINC = 85% ($\boldsymbol{\alpha = 0.15}$)			PINC = 90% ($\boldsymbol{\alpha = 0.10}$)			PINC = 95% ($\boldsymbol{\alpha = 0.05}$)		
		PICP	PINAW	CWC	PICP	PINAW	CWC	PICP	PINAW	CWC
SVR-RBF-10 mins	KDE	0.8083	0.7831	2.0261	0.8298	0.8720	2.6310	0.8818	**1.0011**	2.9796
	AKDE	**0.8728**	**0.7208**	**0.7608**	**0.8978**	**0.7879**	**1.5925**	**0.9269**	1.0942	**2.4725**
SVR-RBF-30 mins	KDE	0.9049	0.8685	0.8685	**0.9199**	0.9339	**0.9639**	**0.9329**	1.0048	**2.1966**
	AKDE	**0.8268**	**0.4417**	**0.8287**	0.8348	**0.4513**	1.1274	0.8768	**0.6453**	2.2156
SVR-RBF-60 mins	KDE	0.8692	0.6499	0.6499	0.8904	0.7261	1.5253	0.9277	0.8338	1.8753
	AKDE	**0.8468**	**0.3663**	**0.6244**	**0.8929**	**0.4522**	**0.9377**	0.9277	**0.5260**	**1.1829**
SVR-RBF-120 mins	KDE	0.8675	0.3462	0.3462	**0.8850**	0.3971	**0.8586**	0.9200	0.4766	1.1799
	AKDE	**0.8650**	**0.3369**	**0.3359**	0.8945	**0.3682**	0.9483	**0.9300**	**0.4374**	**1.0428**

Table 7 indicates that the proposed probabilistic models utilizing AKDE surpass those employing standard KDE across the majority of evaluation metrics and at most confidence levels. The scores in bold denote the best ones (PICP close to the confidence level, lower PINAW, and lower CWC). The AKDE-based model at 85% confidence level surpassed the KDE-based model across all resolutions of 10, 30, 60, and 120 minutes. At a 90% confidence level, the AKDE-based models for 10-minute and 60-minute resolutions demonstrated significantly superior performance compared to the KDE models at the same resolutions. In contrast, the KDE-based models achieved slightly better PICP scores at 30-minute and 120-minute resolutions, respectively. Finally, at a 95% confidence level, the probabilistic models utilizing the AKDE technique had superior CWC scores at resolutions of 10, 60, and 120 minutes, while the KDE-based model demonstrated a marginally better CWC score at a 30-minute resolution. The obtained results may be justified by the fact that, at lower resolutions, the AKDE was able to manage the local variations in the probability distribution of prediction errors more effectively, and therefore shows enhanced estimation of prediction intervals with highly fluctuating WS data. The AKDE technique demonstrated improvements of 62.45%, 39.47%, and 17.02% in CWC scores compared to the KDE method at resolutions of 85%, 90%, and 95%, respectively, within the 10-minute

resolution, as indicated in Eq (25) for a confidence level of 85%.

$$Decrease\ in\ CWC\ \% = \ 62.45\% \ = \frac{2.0261-0.7608}{2.0261} * 100 \tag{25}$$

Considering that the CWC accounts for both PICP and PINAW, this study demonstrates that the AKDE-based model proposed herein yields tight, narrow PIs of WS while effectively covering point predictions, thereby exhibiting exceptional forecasting performance. Figures 5–7 present the PI estimation outcomes of the best models for a randomly chosen period across varying confidence levels, specifically 85%, 90%, and 95%.

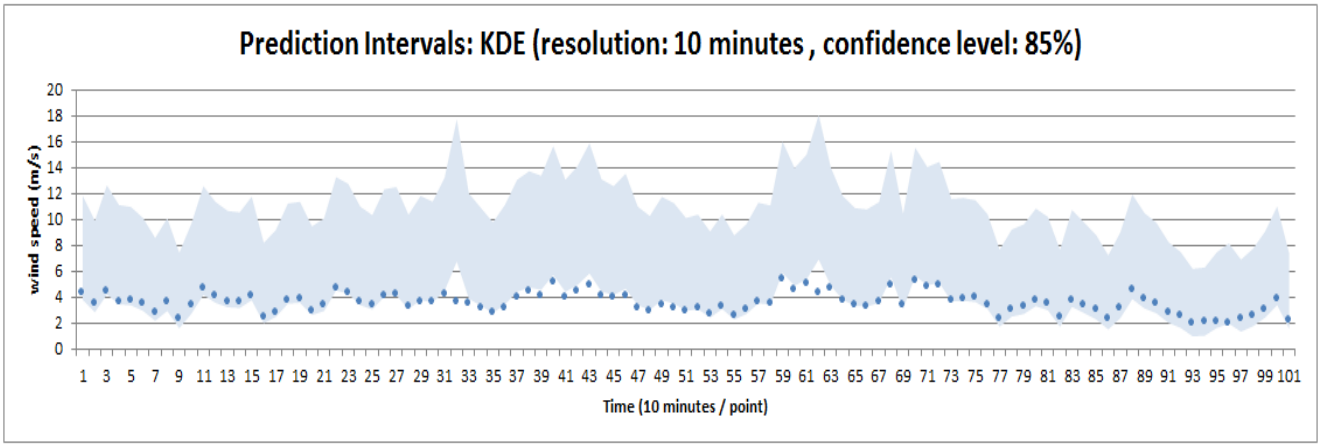
The PIs developed by the AKDE method show a reasonable coverage rate utilizing adaptive interval bandwidths. The total result demonstrates that the proposed prediction framework can generate high-quality PIs for evaluating wind speed information at different confidence levels.

To evaluate the statistical significance of performance enhancements across various temporal resolutions and PINC levels, we performed paired t-tests on the obtained CWC scores. A comparison was conducted between the SVR-AKDE models at 10-minute and 120-minute resolutions across three confidence levels (85%, 90%, and 95%). The simulations yielded a t-statistic of 5.2332 with a p-value of 0.0034, indicating that the performance improvement at the 120-minute resolution is statistically significant ($p < 0.05$). This supports the conclusion that the proposed AKDE adapts well across temporal scales and provides more precise and dependable prediction intervals with increased aggregation. The analysis results of the statistical hypothesis testing are summarized in Table 8.

Table 8. Results of statistical hypothesis testing.

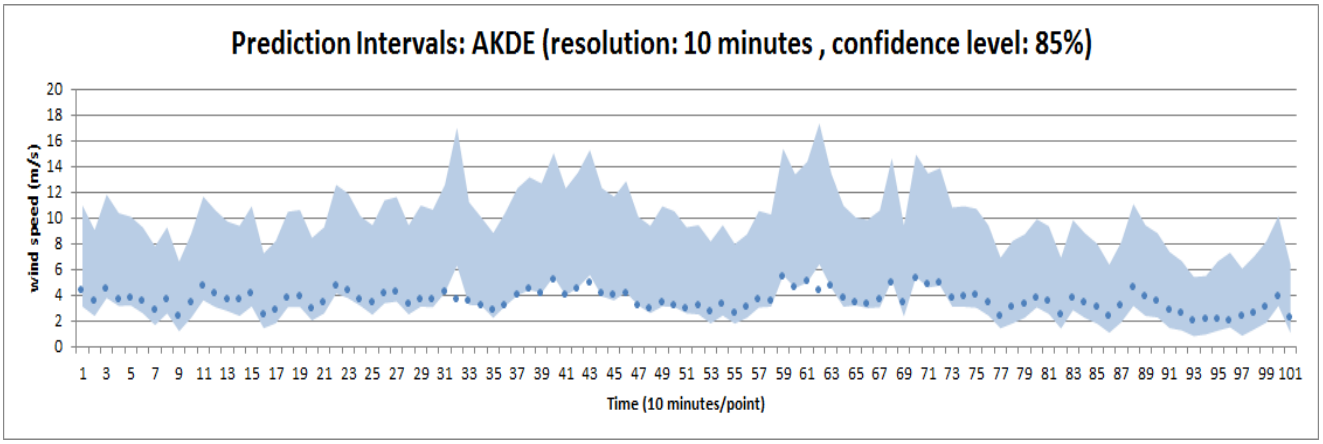
		Mean of Difference	Std. Deviation	Std. Error of Mean	95% Confidence Interval of Difference		t-value	p-value
					CI Lower	CI Upper		
Pair	SVR-RBF-10 mins SVR-RBF-10 mins	1.2913	0.6044	0.2467	0.6570	1.9256	**5.2331**	**0.0034**

The t-value score in Table 8 (5.23) indicates strong evidence that the mean CWC for a 10-minute resolution is significantly higher than for a 120-minute resolution. On the other hand, the p-value's score (0.0034) confirms that the results are statistically significant ($p < 0.05$).

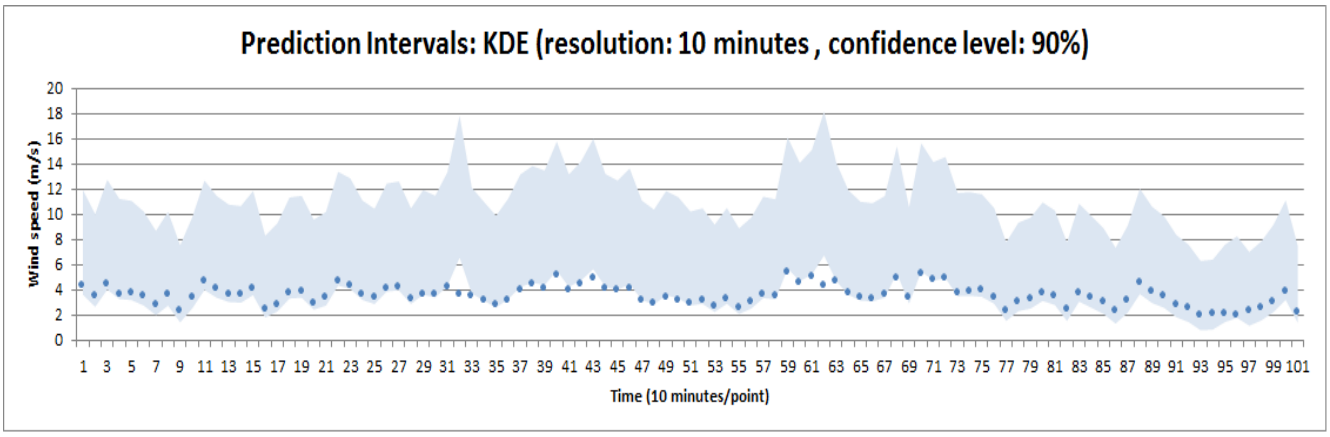


(a) SVR-RBF-10mins with KDE (confidence level 85%)

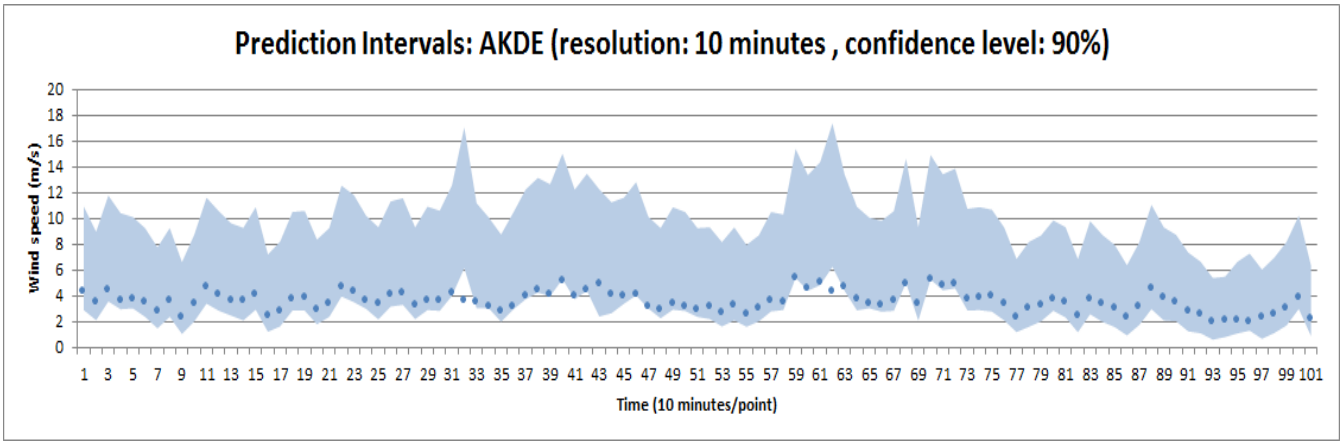
continued on next page



(b) SVR-RBF-10mins with AKDE (confidence level 85%)

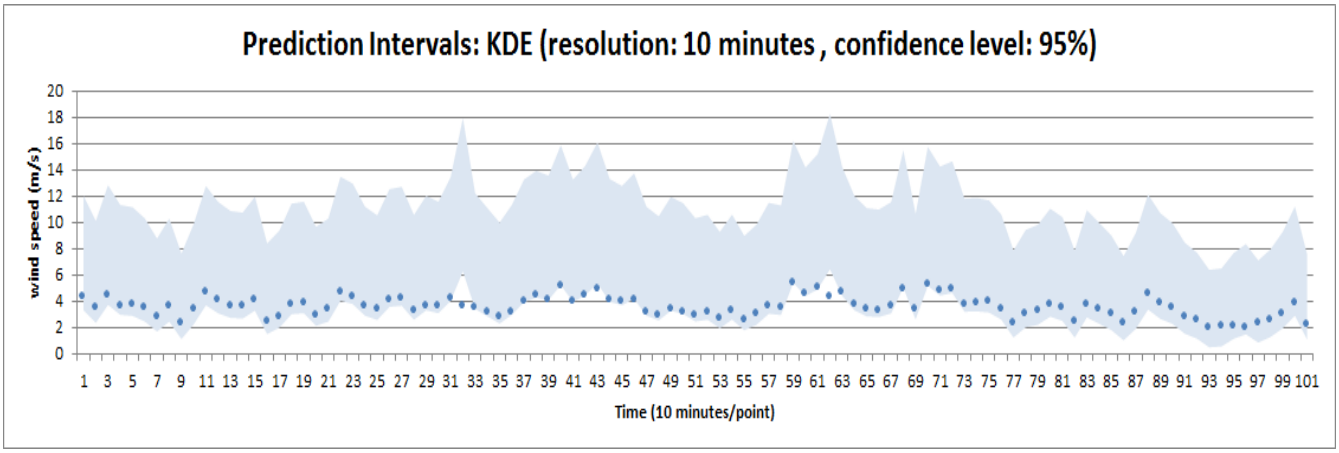


(c) SVR-RBF-10mins with KDE (confidence level 90%)

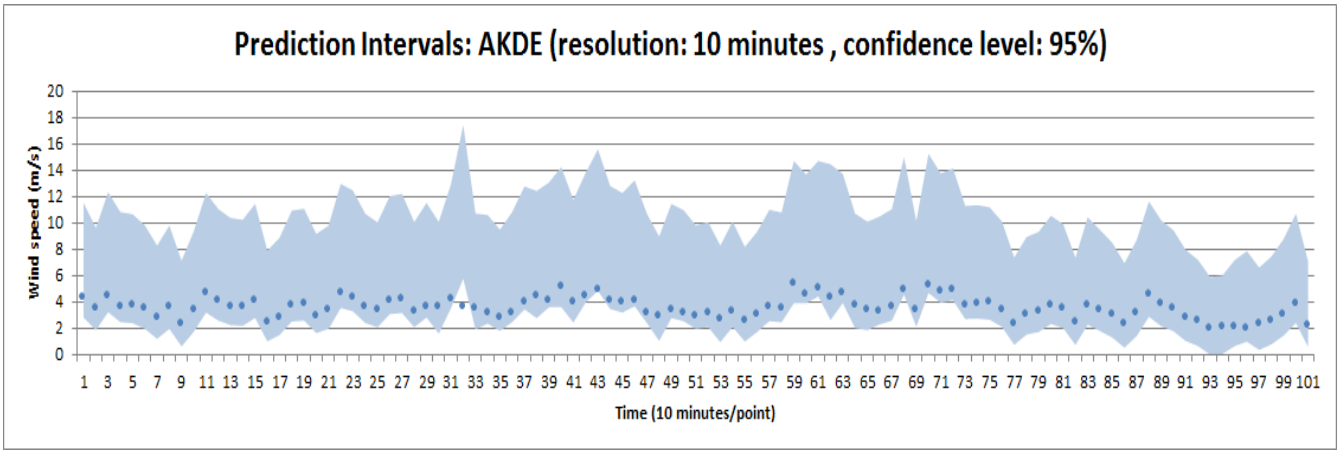


(d) SVR-RBF-10mins with AKDE (confidence level 90%)

continued on next page

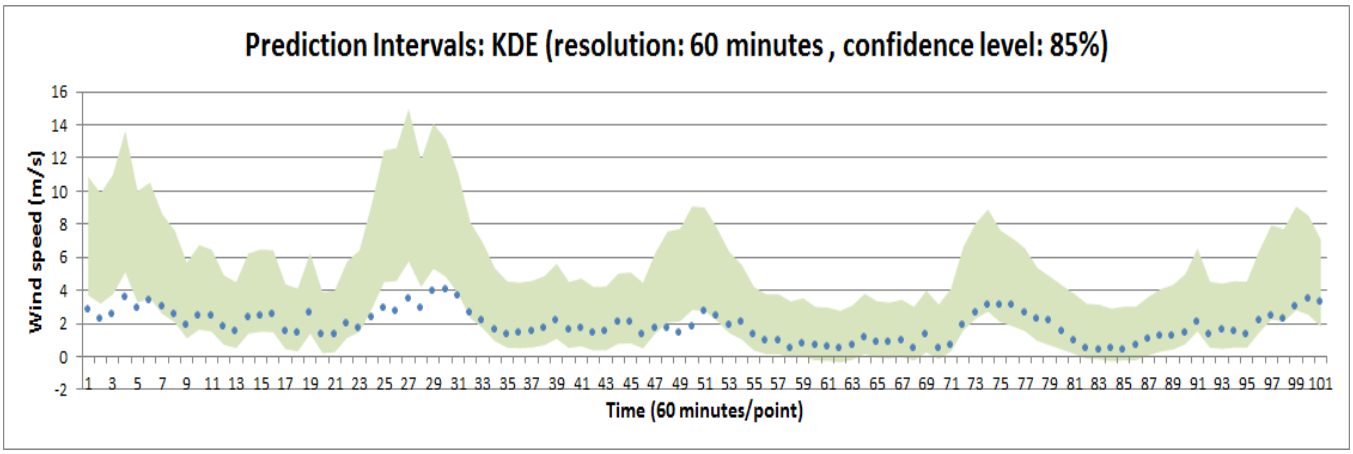


(e) SVR-RBF-10mins with KDE (confidence level 95%)



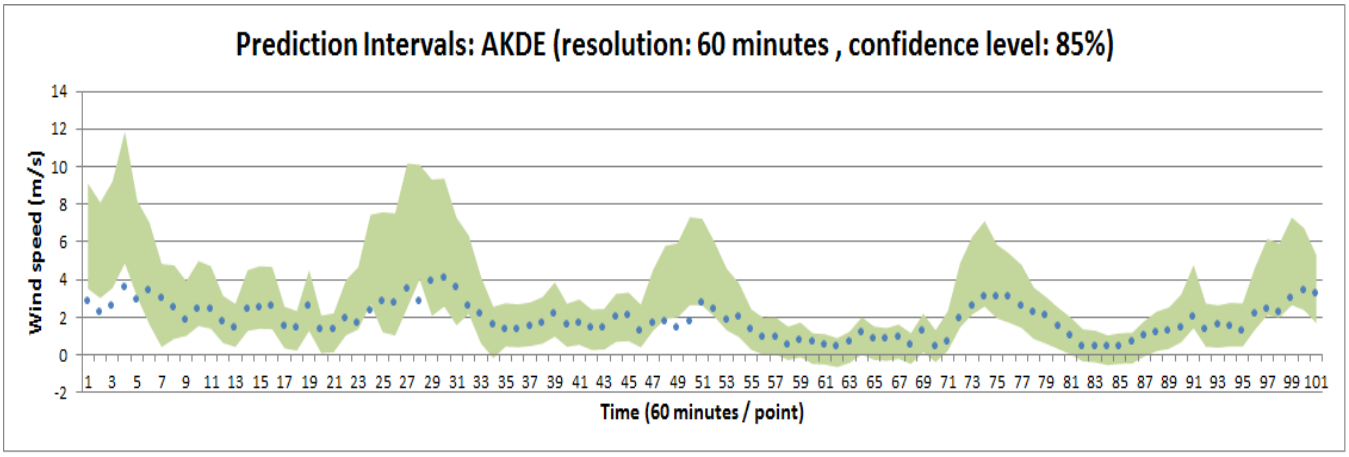
(f) SVR-RBF-10mins with AKDE (confidence level 95%)

Figure 5. Prediction intervals provided by SVR models for 10-minute resolution. The charts related to AKDE approaches in (b), (d), and (f) show better coverage and adaptive intervals against those related to KDE approaches in (a), (c), and (e), respectively.

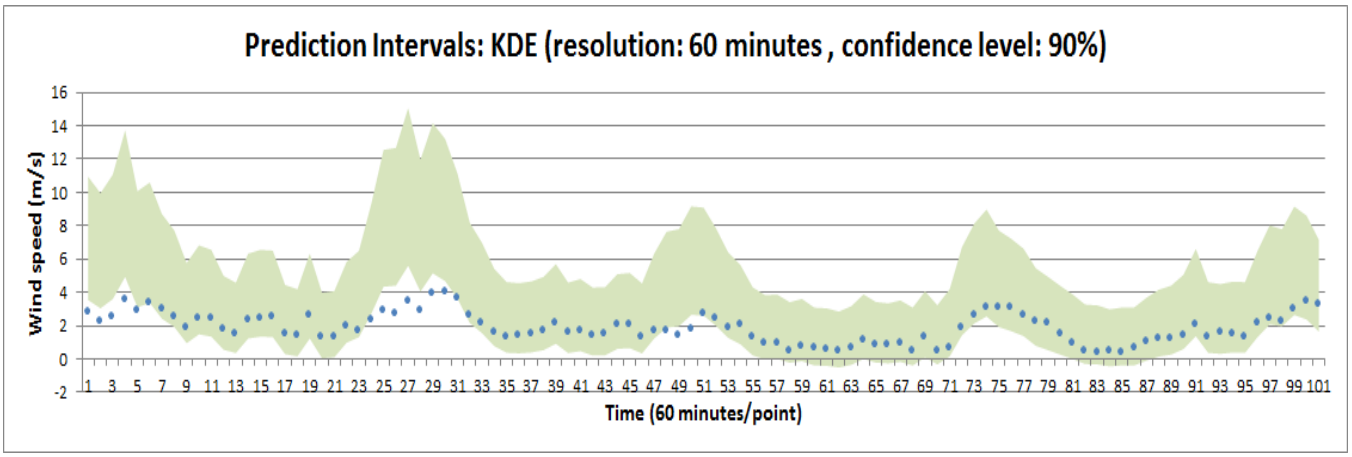


(a) SVR-RBF-60mins with KDE (confidence level 85%)

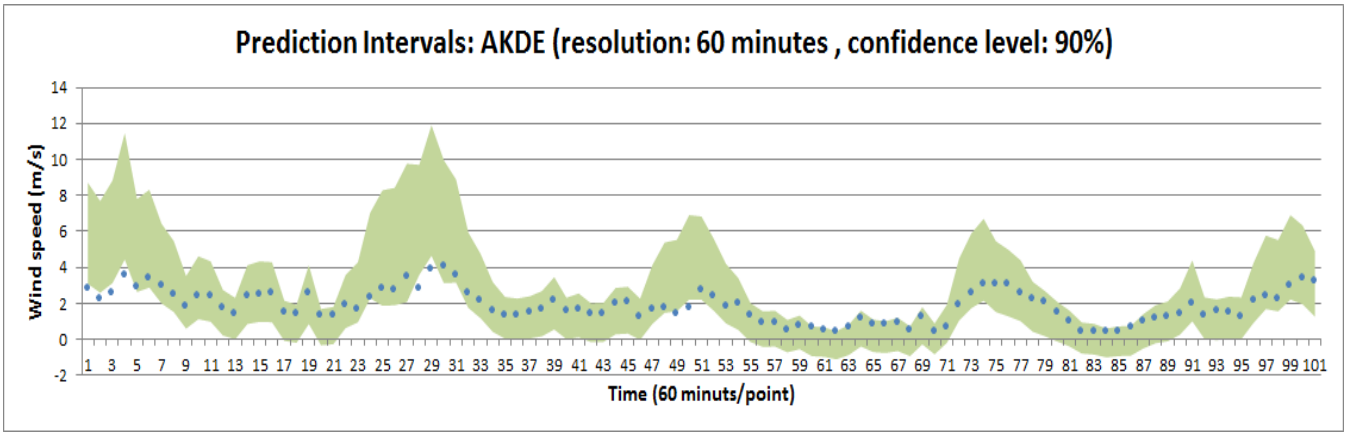
continued on next page



(b) SVR-RBF-60mins with AKDE (confidence level 85%)

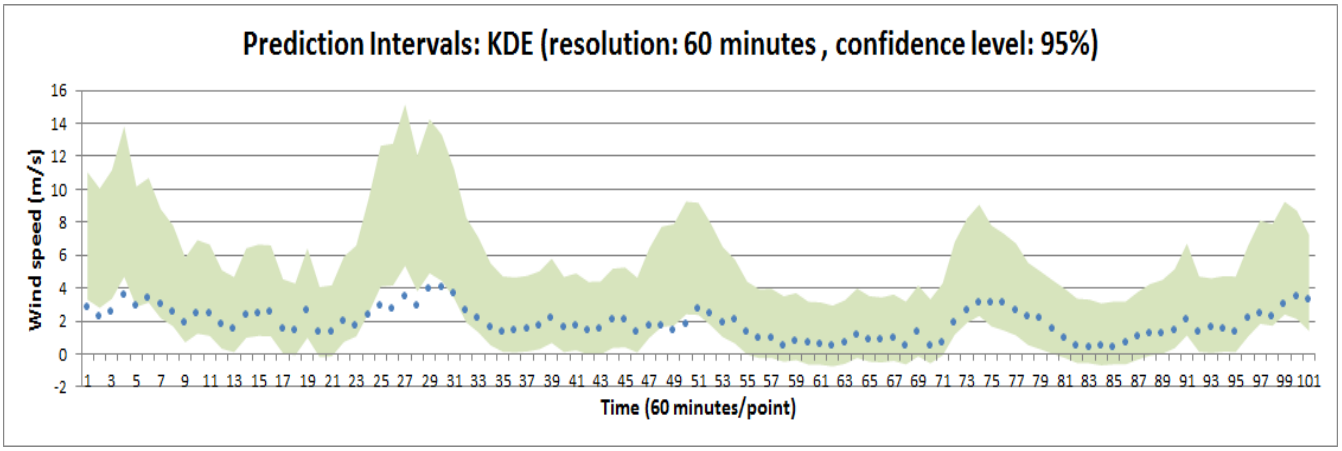


(c) SVR-RBF-60mins with KDE (confidence level 90%)

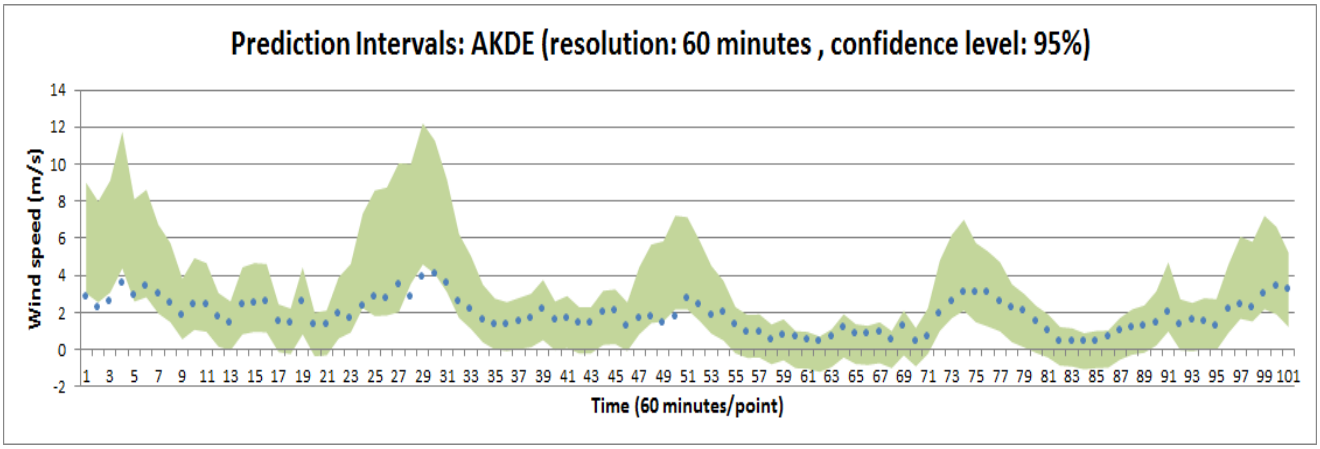


(d) SVR-RBF-60mins with AKDE (confidence level 90%)

continued on next page

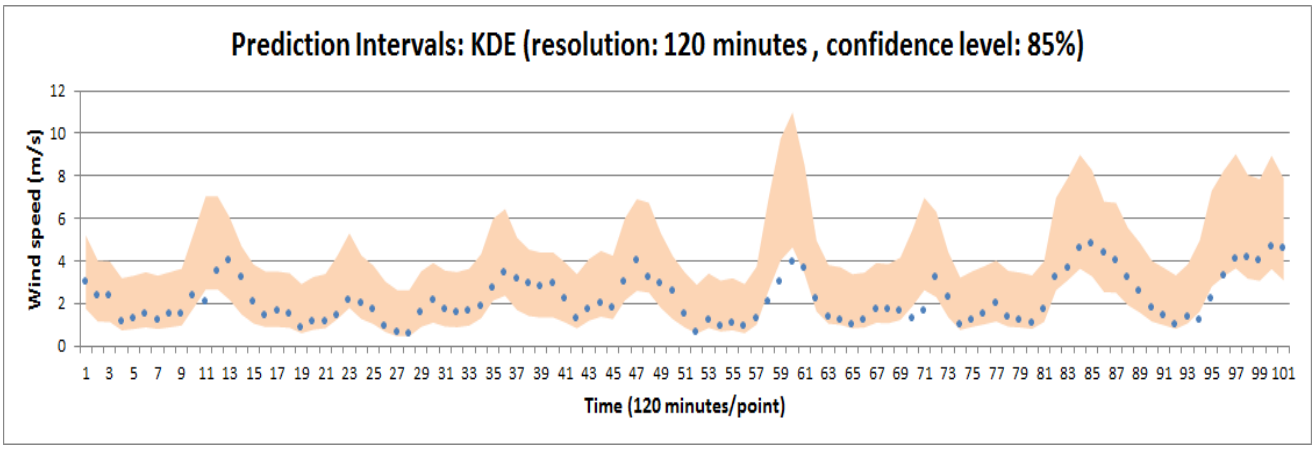


(e) SVR-RBF-60mins with KDE (confidence level 95%)



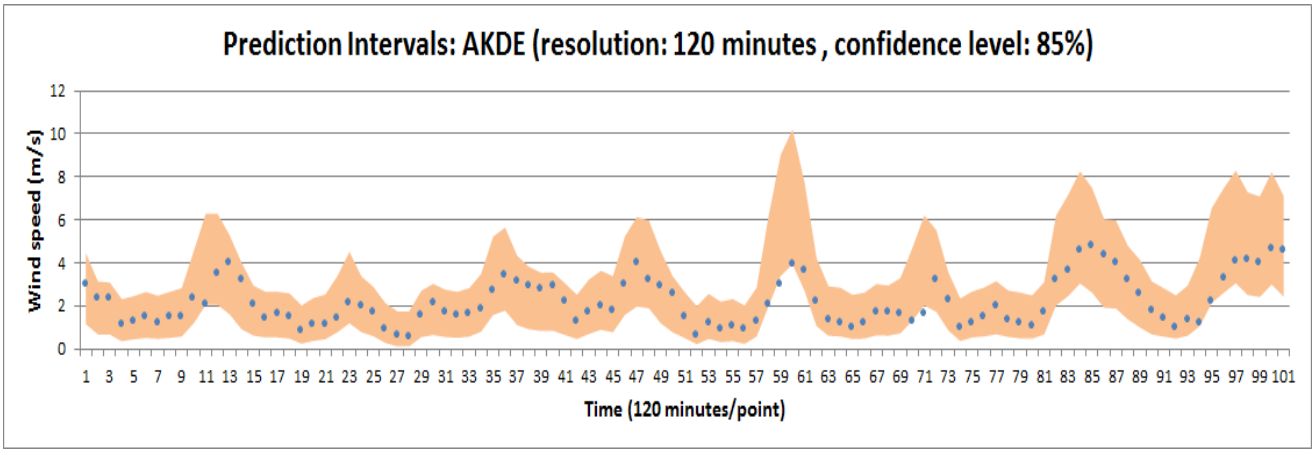
(f) SVR-RBF-60mins with AKDE (confidence level 95%)

Figure 6. Prediction intervals provided by SVR models for 60-minute resolution. The charts related to AKDE approaches in (b), (d), and (f) show better coverage and adaptive intervals against those related to KDE approaches in (a), (c), and (e), respectively.

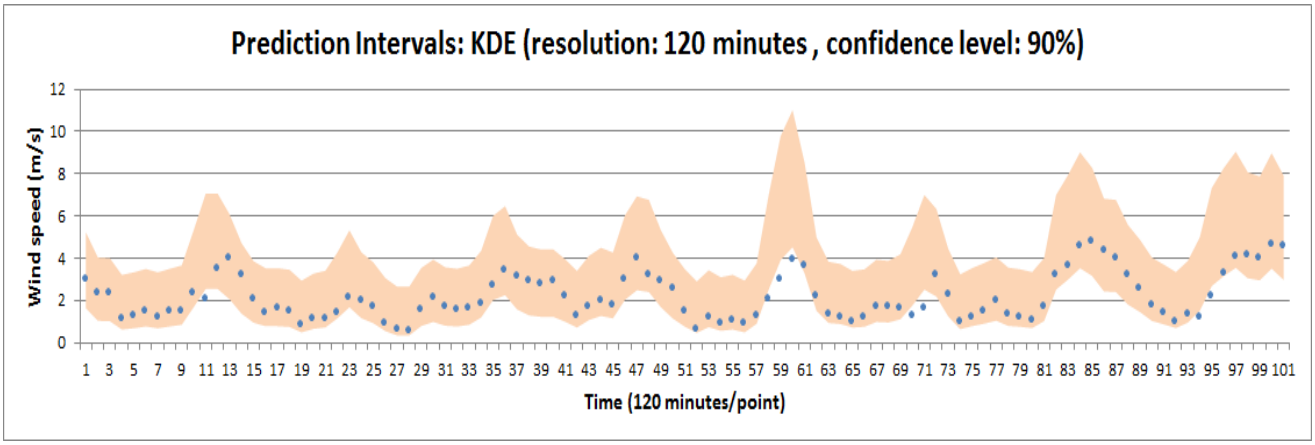


(a) SVR-RBF-120mins with KDE (confidence level 85%)

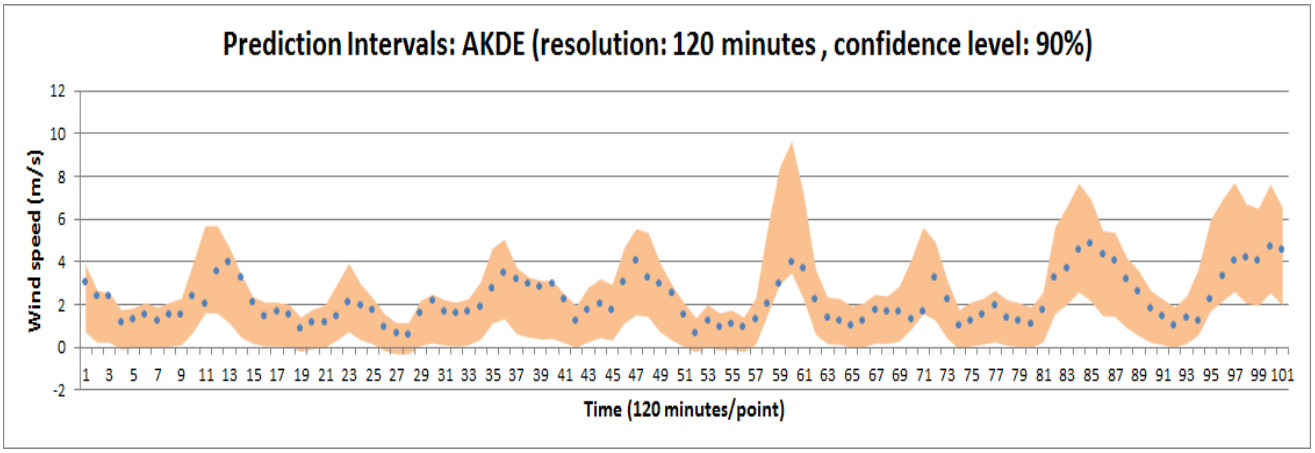
continued on next page



(b) SVR-RBF-120mins with AKDE (confidence level 85%)

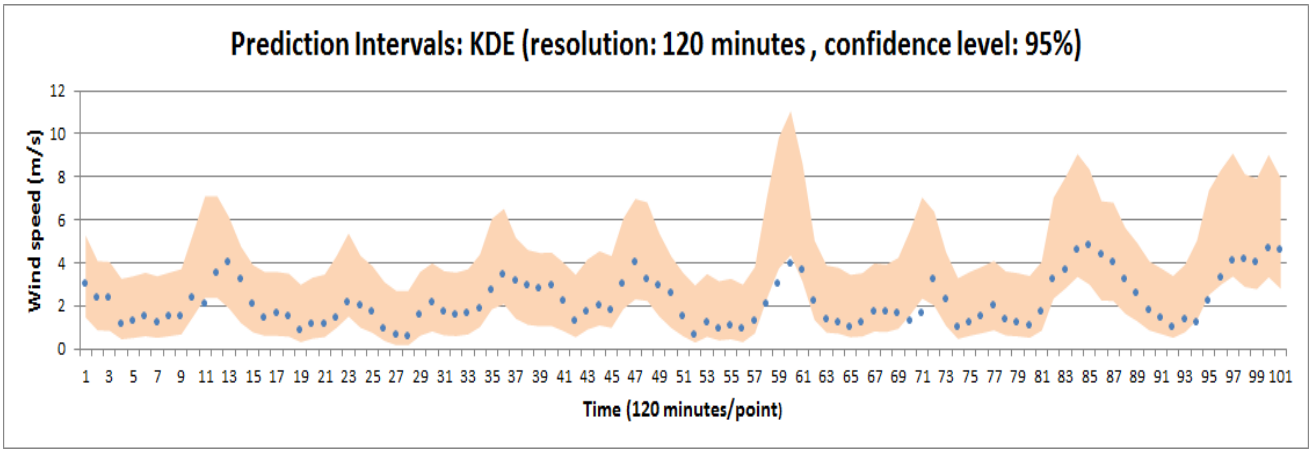


(c) SVR-RBF-120mins with KDE (confidence level 90%)

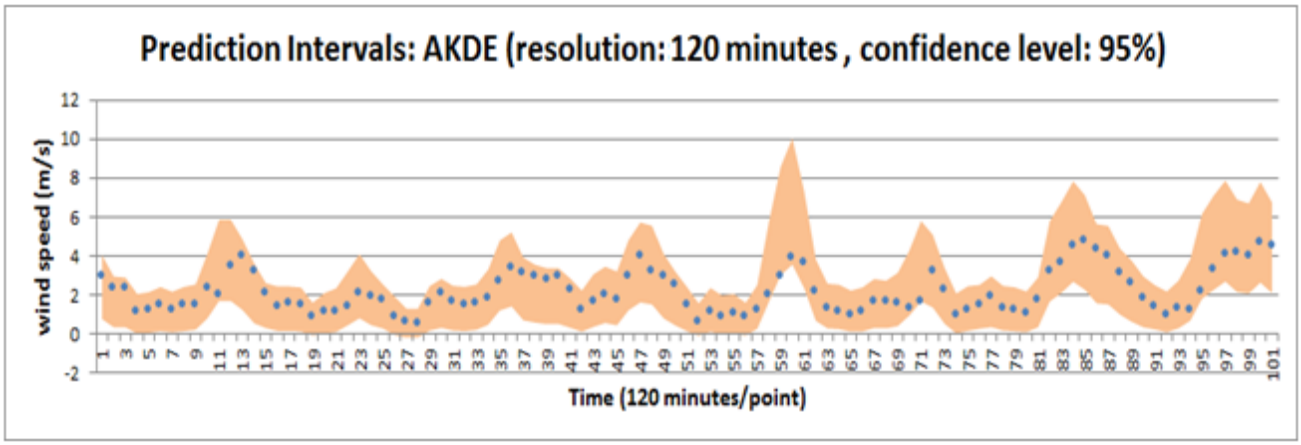


(d) SVR-RBF-120mins with AKDE (confidence level 90%)

continued on next page



(e) SVR-RBF-120mins with KDE (confidence level 95%)



(f) SVR-RBF-120mins with AKDE (confidence level 95%)

Figure 7. Prediction intervals provided by SVR models for 120-minute resolution. . The charts related to AKDE approaches in (b), (d), and (f) show better coverage and adaptive intervals against those related to KDE-based approaches in (a), (c), and (e), respectively.

From the observations in Table 7 and Figures 5–7, we can conclude that the probabilistic framework for WS PI estimation coupled with the AKDE method outperforms the one associated with standard KDE for most of the examined resolutions and within the majority of confidence levels. In other words, the study demonstrates that the AKDE method, based on nearest neighbor estimation (NNE), exhibits superior performance in estimating the PDF of WS forecasting errors.

5. Conclusions

Precise short-term WS prediction presents a challenging task in managing the consumption/production of wind energy within a smart grid. Recent works in the literature underscore the importance of conducting uncertainty modeling of WS, which assesses the probability of accurate predictions. In practice, reliable PI estimators of WS are crucial for evaluating and analyzing the risk associated with WP, enabling informed decision-making for stakeholders.

We introduce a robust prediction framework for short-term WS PI forecasting using ML models and an adaptive KDE method. The framework consists of three modules for data processing, point prediction, and estimation of WS prediction intervals. The data processing module utilizes the QM

approach to detect and remove outliers. Then, distinct SVR-based WS predictors with RBF kernels were trained on datasets corresponding to particular resolutions, namely 10, 30, 60, and 120 minutes.

The PI estimation module utilizes the nonparametric adaptive KDE method to analyze and estimate the PDFs of point prediction errors associated with each resolution. The estimated PDFs and their related CDFs were used to compute the local bandwidth of PIs around point predictions based on the local data density. The NNE technique facilitates the computation of local bandwidth at each data point by assessing the distance to the k-nearest neighbors of that point. This step enables the AKDE method to accurately reflect local variations in the probability distribution of prediction errors. The proposed approach helps in improving the estimate of prediction intervals given a particular confidence level with highly fluctuating WS data. The proposed WS prediction framework is validated for short-term WS forecasting using a weather dataset. Simulation results indicate that considering SVR models for point prediction with the AKDE method provides better forecast accuracy than using the conventional KDE method with this type of data. For PI estimation, the evaluation indices of the proposed PIs forecasting models are consistently improved when we increase the resolution gradually from 10 minutes to 120 minutes. This advantage can be justified by the fact that variations in local data density decrease as the resolution increases. The main contribution of this work includes the introduction and validation of a framework for wind speed variable PI forecasting using SVR models and the AKDE method. The proposed approach is validated over different time resolutions and within distinct confidence levels. Essentially, the AKDE model based on NNE is found to be suitable for describing the distribution of WS prediction errors due to the fluctuating features of wind speed. Some future perspectives of this work include validating the proposed approach across different regions or seasons with different temporal dynamics, as well as enhancing prediction performance while improving operational efficiency.

Use of AI tools declaration

The author states that he has not applied Artificial Intelligence tools in preparing this article in terms of generating paragraphs, composition, and editing.

Conflict of interest

The author declares there are no conflicts of interest.

References

1. S. Roga, S. Bardhan, Y. Kumar, S. K. Dubey, Recent technology and challenges of wind energy generation: A review, *Sustainable Energy Technol. Assess.*, **52** (2022), 102239. https://doi.org/10.1016/j.seta.2022.102239
2. T. M. Dinku, M. S. Manshahia, K. S. Chahal, Soft computing techniques for maximum power point tracking in wind energy harvesting system: A survey, *Artif. Intell. Renewable Energy Climate Change*, (2022), 137–170. https://doi.org/10.1002/9781119771524.ch6
3. Y. Zhuo, L. Li, J. Tang, W. Meng, Z. Huang, K. Huang, et al., Optimal real-time power dispatch of power grid with wind energy forecasting under extreme weather, *Math. Biosci. Eng.*, **20** (2023), 14353–14376. https://doi.org/10.3934/mbe.2023642

4. E. Zio, R. B. Patil, T. Kara, A. Duransahin, D. Duransahin, Implications of climate change Change on Wind Energy Potential, *Sustainability*, **15** (2023), 14822. https://doi.org/10.3390/SU152014822F
5. S. M. R. H Shawon, M. A. Saaklayen, X. Liang, Wind speed forecasting by conventional statistical methods and machine learning techniques, in *2021 IEEE Electrical Power and Energy Conference (EPEC)*, (2021), 304–309. https://doi.org/10.1109/EPEC52095.2021.9621686
6. L. Peng, S. X. Lv, L. Wang, Explainable machine learning techniques based on attention gate recurrent unit and local interpretable model-agnostic explanations for multivariate wind speed forecasting, *J. Forecast.*, **43** (2024), 064–2087. https://doi.org/10.1002/for.3097
7. Y. Yang, H. Lou, J. Wu, S. Zhang, S. Gao, A survey on wind power forecasting with machine learning approaches, *Neural Comput. Appl.*, (2024), 1–21. https://doi.org/10.1007/s00521-024-09923-4
8. Y. Yang, Y. Gao, Z. Wang, X. Li, H. Zhou, J. Wu, Multiscale-integrated deep learning approaches for short-term load forecasting, *Int. J. Mach. Learn. Cybern.*, **15** (2024), 6061–6076.
9. J. A. Carta, P. Ramrez, S. Velzquez, A review of wind speed probability distributions used in wind energy analysis: Case studies in the Canary Islands, *Renewable Sustainable Energy Rev.*, **13** (2009), 933–955. https://doi.org/10.1016/j.rser.2008.05.005
10. Y. L. Chen, X. Hu, L. X. Zhang, A review of ultra-short-term forecasting of wind power based on data decomposition-forecasting technology combination model, *Energy Rep.*, **8** (2022), 14200–14219. https://doi.org/10.1016/j.egyr.2022.10.342
11. R. A. Hajj, M. M. Fouad, A. Assi, E. Mabrouk, Ultra-short-term forecasting of wind speed using lightweight features and machine learning models, in *2023 12th International Conference on Renewable Energy Research and Applications (ICRERA)*, (2023), 93–97. https://doi.org/10.1109/ICRERA59003.2023.10269374
12. Z. Wang, Y. Ying, L. Kou, W. Ke, J. Wan, Z. Yu, et al., Ultra-short-term offshore wind power prediction based on PCA-SSA-VMD and BiLSTM, *Sensors*, **24** (2024), 444. https://doi.org/10.3390/s24020444
13. J. Naik, R. Bisoi, P. K. Dash, Prediction interval forecasting of wind speed and wind power using modes decomposition based low rank multi-kernel ridge regression, *Renewable Energy*, **129** (2018), 357–383. https://doi.org/10.1016/j.renene.2018.05.031
14. R. Li, Y. Jin, A wind speed interval prediction system based on multi-objective optimization for machine learning method, *Appl. Energy*, **228** (2018), 2207–2220. https://doi.org/10.1016/j.apenergy.2018.07.032
15. W. Ding, F. Meng, Point and interval forecasting for wind speed based on linear component extraction, *Appl. Soft Comput.*, **93** (2020), 106350. https://doi.org/10.1016/j.asoc.2020.106350
16. H. Huang, Y. Hong, H. Wang, Probabilistic prediction intervals of wind speed based on explainable neural network, *Front. Energy Res.*, **10** (2022), 934935. https://doi.org/10.3389/fenrg.2022.934935
17. J. Zhang, C. Draxl, T. Hopson, L. Delle Monache, E. Vanvyve, B. M. Hodge, Comparison of numerical weather prediction based deterministic and probabilistic wind resource assessment methods, *Appl. Energy*, **156** (2015), 528–541. https://doi.org/10.1016/j.apenergy.2015.07.059
18. G. Hou, J. Wang, Y. Fan, J. Zhang, C. Huang, A novel wind power deterministic and interval prediction framework based on the critic weight method, improved northern goshawk optimization, and kernel density estimation, *Renewable Energy*, **226** (2024), 120360. https://doi.org/10.1016/j.renene.2024.120360

19. W. Yang, M. Hao, Y. Hao, Innovative ensemble system based on mixed frequency modeling for wind speed point and interval forecasting, *Inf. Sci.*, **622** (2023), 560–586. https://doi.org/10.1016/j.ins.2022.11.145
20. Z. Tian, J. Wang, A novel wind speed interval prediction system based on neural network and multi‐objective grasshopper optimization, *Int. Trans. Electr. Energy Syst.*, **1** (2022), 5823656. https://doi.org/10.1155/2022/5823656
21. X. Wang, J. Wang, X. Niu, C. Wu, Novel wind-speed prediction system based on dimensionality reduction and nonlinear weighting strategy for point-interval prediction, *Expert Syst. Appl.*, **241** (2024), 122477. https://doi.org/10.1016/j.eswa.2023.122477
22. Y. Zhang, Y. Zhao, X. Shen, J. Zhang, A comprehensive wind speed prediction system based on Monte Carlo and artificial intelligence algorithms, *Appl. Energy*, **305** (2022), 117815. https://doi.org/10.1016/j.apenergy.2021.117815
23. J. Wang, S. Wang, B. Zeng, H. Lu, A novel ensemble probabilistic forecasting system for uncertainty in wind speed, *Appl. Energy*, **313** (2022), 118796. https://doi.org/10.1016/j.apenergy.2022.118796
24. Q. Zhu, Y. Xu, Q. Lin, Z. Ming, K. C. Tan, Clustering-based short-term wind speed interval prediction with multi-objective ensemble learning, *IEEE Trans. Emerging Top. Comput. Intell.*, 2024. https://doi.org/10.1109/TETCI.2024.3400852
25. P. Sun, Z. Liu, J. Wang, W. Zhao, Interval forecasting for wind speed using a combination model based on multi-objective artificial hummingbird algorithm, *Appl. Soft Comput.*, **150** (2024), 111090. https://doi.org/10.1016/j.asoc.2023.111090
26. G. Tang, Y. Wu, C. Li, P. K. Wong, Z. Xiao, X. An, A novel wind speed interval prediction based on error prediction method, *IEEE Trans. Ind. Inf.*, **16** (2020), 6806–6815. https://doi.org/10.1109/TII.2020.2973413
27. A. Saeed, C. Li, M. Danish, S. Rubaiee, G. Tang, Z. Gan, et al., Hybrid bidirectional LSTM model for short-term wind speed interval prediction, *IEEE Access*, **8** (2020), 182283–182294. https://doi.org/10.1109/ACCESS.2020.3027977
28. Y. Zhang, G. Pan, Y. Zhao, Q. Li, F. Wang, Short-term wind speed interval prediction based on artificial intelligence methods and error probability distribution, *Energy Convers. Manage.*, **224** (2020), 113346. https://doi.org/10.1016/j.enconman.2020.113346
29. J. Wang, S. Wang, Z. Li, Wind speed deterministic forecasting and probabilistic interval forecasting approach based on deep learning, modified tunicate swarm algorithm, and quantile regression, *Renewable Energy*, **179** (2021), 1246–1261. https://doi.org/10.1016/j.renene.2021.07.113
30. Z. Gan, C. Li, J. Zhou, G. Tang, Temporal convolutional networks interval prediction model for wind speed forecasting, *Electr. Power Syst. Res.*, **191** (2021), 106865. https://doi.org/10.1016/j.epsr.2020.106865
31. C. Li, G. Tang, X. Xue, X. Chen, R. Wang, C. Zhang, The short-term interval prediction of wind power using the deep learning model with gradient descend optimization, *Renewable Energy*, **155** (2020), 197–211. https://doi.org/10.1016/j.renene.2020.03.098
32. Z. Cui, T. Li, Z. Ding, X. A. Li, J. Wu, Probabilistic oil price forecasting with a variational mode decomposition-gated recurrent unit model incorporating pinball loss, *Data Sci. Manage.*, 2024. https://doi.org/10.1016/j.dsm.2024.10.003
33. Y. Yang, Y. Gao, H. Zhou, J. Wu, S. Gao, Y. G. Wang, Multi-Granularity Autoformer for long-term deterministic and probabilistic power load forecasting, *Neural Networks*, (2025), 107493. https://doi.org/10.1016/j.neunet.2025.107493

34. S. Liu, J. Deng, J. Yuan, W. Li, X. A. Li, J. Xu, et al., Wang, Probabilistic quantile multiple fourrier feature network for lake temperature forecasting: incorporating pinball loss for uncertainty estimation, *Earth Sci. Inf.*, (2024), 1–14. https://doi.org/10.1007/s12145-024-01448-7
35. S. Althoff, J. H. Szabadvary, J. Anderson, L. Carlsson, Evaluation of conformal-based probabilistic forecasting methods for short-term wind speed forecasting, in *Proceedings of the Twelfth Symposium on Conformal and Probabilistic Prediction with Applications*, (2023), 100–115.
36. L. de Azevedo Takara, A. Teixeira, H. Yazdanpanah, V. C. Mariani, L. dos Santos Coelho, Optimizing multi-step wind power forecasting: Integrating advanced deep neural networks with stacking-based probabilistic learning, *Appl. Energy*, **369** (2024), 123487. https://doi.org/10.1016/j.apenergy.2024.123487
37. B. W. Silverman, Density estimation: for statistics and data analysis, in *Density Estimation: For Statistics and Data*, (2018), 1–175. https://doi.org/10.1201/9781315140919
38. J. Zhang, S. Chowdhury, A. Messac, L. Castillo, A multivariate and multimodal wind distribution model, *Renewable Energy*, **51** (2013), 436–447. https://doi.org/10.1016/j.renene.2012.09.026
39. A. Gramacki, *Nonparametric Kernel Density Estimation and Its Computational Aspects*, Cham, Switzerland: Springer International Publishing, **37** (2018). https://doi.org/10.1007/978-3-319-71688-6
40. W. Ding, F. Meng, Point and interval forecasting for wind speed based on linear component extraction, *Appl. Soft Comput.*, **93** (2020), 106350. https://doi.org/10.1016/j.asoc.2020.106350
41. M. Awad, R. Khanna, M. Awad, R. Khanna, Support vector regression, in *Efficient Learning Machines*, Springer, (2015), 67–80. https://doi.org/10.1007/978-1-4302-5990-9_4
42. B. Kumar, O. P. Vyas, R. A. Vyas, Comprehensive review on the variants of support vector machines, *Modern Phys. Lett. B*, **33** (2019), 1950303. https://doi.org/10.1142/S0217984919503032
43. J. Wu, Y. G. Wang, H. Zhang, Augmented support vector regression with an autoregressive process via an iterative procedure, *Appl. Soft Comput.*, **158** (2024), 111549. https://doi.org/10.1016/j.asoc.2024.111549
44. Y. C. Chen, A tutorial on kernel density estimation and recent advances, *Biostat. Epidemiol.*, **1** (2017), 161–187. https://doi.org/10.1080/24709360.2017.1396742
45. S. Węglarczyk, Kernel density estimation and its application, in *ITM Web of Conferences*, **23** (2018), 00037. https://doi.org/10.1051/itmconf/20182300037
46. S. J. Sheather, Density estimation, *Stat. Sci.*, **19** (2004), 588–597.
47. M. C. Jones, J. S. Marron, S. J. Sheather, A brief survey of bandwidth selection for density estimation, *J. Am. Stat. Assoc.*, **91** (1996), 401–407. https://doi.org/10.1080/01621459.1996.10476701
48. M. Rudemo, Empirical choice of histograms and kernel density estimators, *Scand. J. Stat*, **9** (1982), 65–78.
49. Y. Jin, Y. He, D. Huang, An improved variable kernel density estimator based on L2 regularization, *Mathematics*, **9** (2021). https://doi. org/10.3390/math9162004
50. Y. Ziane, S. Adjabi, N. Zougab, Adaptive Bayesian bandwidth selection in asymmetric kernel density estimation for nonnegative heavy-tailed data, *J. Appl. Stat.*, **42** (2015), 1645–1658. http://doi.org/10.1080/02664763.2015.1004626
51. H. Wang, Z. Lei, X. Zhang, B. Zhou, J. Peng, A review of deep learning for renewable energy forecasting, *Energy Convers. Manage.*, **198** (2019), 111799. https://doi.org/10.1016/j.enconman.2019.111799

52. T. T. Chau, T. T. H. Nguyen, L. Nguyen, T. D. Do, Wind speed probability distribution based on adaptive bandwidth kernel density estimation model for wind farm application, *Wind Energy*, **28** (2025), e2970. http://doi.org/10.1002.we.2970
53. I. S. Abramson, On bandwidth variation in kernel estimates-a square root law, *Ann. Stat.*, **10** (1982), 1217–1223.
54. R. Al-Hajj, G. Oskrochi, M. M. Fouad, A. Assi, Probabilistic prediction intervals of short-term wind speed using selected features and time shift dependent machine learning models, *Math. Biosci. Eng.*, **22** (2025), 23–51. http://doi.org/ 10.3934/mbe.2025002
55. F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, et al., Scikit-learn: machine learning in Python, *J. Mach. Learn. Res.*, **12** (2011), 2825–2830.

AIMS Press