Energy Conversion and Management 198 (2019) 111823

Contents lists available at ScienceDirect

Energy Conversion and Management

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

Wind power forecasting based on daily wind speed data using machine learning algorithms

Halil Demolli[a], Ahmet Sakir Dokuz[b,*], Alper Ecemis[b], Murat Gokcek[c]

[a] University of Prishtina, Faculty of Mechanical Engineering, Bregu i Diellit p.n., 10 000 Prishtina, Kosovo
[b] Department of Computer Engineering, Faculty of Engineering, Nigde Omer Halisdemir University, Main Campus, 51240 Nigde, Turkey
[c] Department of Mechanical Engineering, Faculty of Engineering, Nigde Omer Halisdemir University, Main Campus, 51240 Nigde, Turkey

ARTICLE INFO

Keywords:
Wind energy
Wind power forecasting
Machine Learning
Regression

ABSTRACT

Wind energy is a significant and eligible source that has the potential for producing energy in a continuous and sustainable manner among renewable energy sources. However, wind energy has several challenges, such as initial investment costs, the stationary property of wind plants, and the difficulty in finding wind-efficient energy areas. In this study, long-term wind power forecasting was performed based on daily wind speed data using five machine learning algorithms. We proposed a method based on machine learning algorithms to forecast wind power values efficiently. We conducted several case studies to reveal performances of machine learning algorithms. The results showed that machine learning algorithms could be used for forecasting long-term wind power values with respect to historical wind speed data. Furthermore, the results showed that machine learning-based models could be applied to a location different from model-trained locations. This study demonstrated that machine learning algorithms could be successfully used before the establishment of wind plants in an unknown geographical location whether it is logical by using the model of a base location.

1. Introduction

Renewable energy sources are getting more attention from the energy production authorities, countries, and energy companies because of their natural, free of costs, and environmentally clean nature. Among renewable energy sources, wind energy is one of the most significant and potentially useful energy source [1–3]. Wind energy has the potential to produce power for each hour of the day and is suitable for systems that require continuous energy, seasonal fluctuations of wind can be predicted, wind energy is a sustainable energy source, and wind turbines can be built on existing farms without losing agricultural areas.

However, the use of wind energy has several challenges. Firstly, initial investment costs are higher in comparison with conventional power plants. Secondly, since wind turbines are not easily portable, the wind energy potential of promising locations should be well-analyzed. Thirdly, wind-efficient areas are in remote locations, and connecting these areas to the national grid requires transmission lines. Finally, wind turbines can cause damage to local wildlife and can generate noise and lead to aesthetic pollution.

On the other hand, machine learning is a field of computer science domain that tries to provide learning capability to computers or other devices without being explicitly operated. It aims to develop methods and algorithms to learn from data and forecast based on data [4,5]. Machine learning algorithms are successfully utilized to describe the behavior of the dataset, model input features with respect to the expected output and forecast output features with respect to its historical records. Machine learning algorithms are one of the alternatives to forecasting wind power based on wind speed data.

In the literature, there are many studies on wind power forecasting using several analysis methods and on several horizons. Preliminary studies about wind power forecasting take into account using persistence and statistical methods, however, recent studies prefer to use machine learning algorithms, especially the classification algorithms of Random Forest, support vector machines, and deep learning architectures of long-short-term memory networks. The main reasons for using machine learning algorithms is that these algorithms can adapt themselves to changing trends inside datasets and produce models based on input data rather than using a generalized model. Many literature studies focus on short-term wind power forecasting due to its simplicity and high accuracy. However, there is a limited number of studies on long-term wind power forecasting, especially year-ahead wind power forecasting because of the year-ahead forecast is harder. The main contribution of this study is year-ahead wind power forecast based on wind power values of past years.

* Corresponding author.
E-mail address: adokuz@ohu.edu.tr (A.S. Dokuz).

https://doi.org/10.1016/j.enconman.2019.111823
Received 25 March 2019; Received in revised form 12 July 2019; Accepted 13 July 2019
Available online 25 July 2019

In this study, wind power forecasting was performed based on daily wind speed data using machine learning algorithms. We preferred to use daily wind power forecasting, because wind speed has an unstable and unpredictable nature. Especially for long-term forecasting scenarios, the methods and algorithms cannot provide good and satisfying results with regard to wind speed forecasting. Regression analysis algorithms are used due to the forecasting problem of continuous wind power values. The regression algorithms of Least Absolute Shrinkage Selector Operator (LASSO), k Nearest Neighbor (kNN), eXtreme Gradient Boost (XGBoost), Random Forest (RF), and Support Vector Regression (SVR) algorithms were used in this study. Daily mean wind speed data were calculated using the hourly wind speed dataset, and the daily total wind power was modeled using daily wind speed and also the standard deviation. The proposed method was applied to several candidate locations to see whether the algorithms could produce acceptable results with respect to each location. Finally, to better demonstrate the efficiency of our proposed machine learning based wind power forecasting models, we tested our models against the wind speed values of four different locations. This study showed that machine learning algorithms could be successfully used before the establishment of wind plants in an unknown geographical location whether it is logical by using the model of a base location.

The main contributions of this study can be listed as follows:

- Long-term wind power forecasting was performed using different machine learning algorithms.
- Daily wind speed, the daily standard deviation of hourly wind speed and generated daily wind power data were used as input to machine learning algorithms.
- Five machine learning algorithms were used for the long-term modeling of wind power.
- The results are beneficial for the establishment of a new wind plant in an unknown location.

The remaining part of this study is organized as follows. Section 2 presents the literature review. Section 3 presents preliminaries of wind power calculations and machine learning algorithms and gives problem definition. Section 4 presents the proposed wind power forecasting method. Section 5 presents the experimental evaluation of the study and Section 6 presents conclusions.

2. Literature review

The studies on wind power forecasting could be divided into two categories, such as, short-term wind power forecasting and long-term wind power forecasting. Short-term wind power forecasting methods try to forecast wind power with respect to short time periods, i.e. 1 h to several days ahead, while long-term wind power forecasting methods try to forecast wind power with respect to longer time periods, i.e. several days to 1 year ahead. Each category is divided into two separate sub-categories, such as statistical and machine learning methods.

In short-term wind power forecasting using statistical methods, Rajagopalan and Santoso [6] applied the ARMA model to wind speed time series data and observed that the model was successful in one-hour forecasting and for the data of a group of wind plants. Abdelaziz et al. [7] suggested a method based on the ARIMA model and compared the results with persistence methods. Cadenas et al. [8] used two statistical models, a univariate ARIMA model and a multivariate NARX model, for wind power forecasting and concluded that the NARX model performed one-step ahead forecasting better than the ARIMA model. Dowell and Pinson [9] proposed a method that uses the sVAR model for very short term wind power forecasting of many wind plants in different spatial locations. Lima et al. [10] proposed a meteorological-statistic model to ensure the accurate forecasting of wind power with the horizon of 72 h ahead. Wang et al. [11] used the ARMA model for forecasting short-term wind power and achieved good performance. Robles-Rodriguez and Dochain [12] proposed an ARMAX-based statistical method that employs a two-folded mechanism: time series decomposition and non-linearity handling for the accurate forecasting of wind power 48 h ahead. Pearre and Swan [13] proposed statistical methods to catalog and correct forecast errors and achieved good performance in 1–6 h ahead forecasts.

In short-term wind power forecasting using machine learning methods, Sideratos and Hatziargyriou [14] proposed a combination of neural networks and fuzzy logic for the accurate estimation of a wind plant power output with the horizon of 48 h by taking the input of the data based on the magnitude of wind speed of prediction and of the next hour. Rahmani et al. [15] proposed a hybrid of ant colony and particle swarm optimization algorithms for wind energy forecasting. Najeebullah et al. [16] proposed a Machine Learning-based Short Term Wind Power Prediction system that uses a combination of machine learning techniques for feature selection and regression. Chi et al. [17] used linear regression, multi-layer perceptron, and support vector machine algorithms for wind speed forecasting. Peng et al. [18] proposed a hybrid model based on decomposition and AdaBoost-extreme learning machine for wind power forecasting. Lahouar and Slama [19] used the Random Forest algorithm by using several weather factors other than using only wind speed to forecast wind power. Li et al. [20] and Sun et al. [21] proposed data mining-based methods for wind power forecasting. Wang et al. [22] proposed a deep belief network deep learning architecture and k-means clustering algorithm to better handle numerical weather prediction and wind data. Zheng et al. [23] proposed a hybrid approach integrating the genetic algorithm, particle swarm optimization algorithm, and adaptive neuro-fuzzy inference systems for wind power forecasting in microgrids. Yu et al. [24], Qin et al. [25], and Shi et al. [26] used Long-Short-Term-Memory based deep learning architectures for forecasting wind power.

In long-term wind power forecasting using statistical methods, Eldali et al. [27] used the ARIMA model on hourly wind power – forecast and actual – from ERCOT data to forecast wind power better. Barbosa de Alencar et al. [28] proposed different models for ultra-short, short, medium and long-term wind power forecasting using artificial neural network models, ARIMA, and hybrid models. Ekstrm et al. [29] proposed a method for statistical modeling of the wind power generation of multiple wind power plants without measurement data using a vector autoregressive based methodology. Dokuz et al. [30] proposed an ARIMA and clustering based method to better forecast one-year wind speed for the accurate calculation of wind power.

In long-term wind power forecasting using machine learning methods, Barbounis et al. [31] used local recurrent neural networks to forecast the wind power of a wind plant three days ahead with respect to meteorological data of 4 nearby locations. Khan et al. [32] proposed a technique that is based on Cartesian Genetic Programming and artificial neural networks to forecast wind power 1 h to 1 year ahead. Wang et al. [33] proposed a hybrid of Support Vector Regression, seasonal index adjustment, Elman recurrent neural network to forecast medium-term wind power for three different sites. Dumitru and Gligor [34] investigated the use of feedforward artificial neural networks for daily average wind energy forecasting. Yan and Ouyang [35] proposed a two-step hybrid wind power forecasting model that uses both physical and data mining based techniques for forecasting 3-month ahead wind power for a wind plant. Maroufpoor et al. [36] investigated the use of six different machine heuristic artificial intelligence algorithms for wind speed forecasting using meteorological variables.

When the literature studies are analyzed, although there are several recent studies, statistical methods are not the first preferred methods for wind power forecasting because statistical methods cannot adapt themselves to nonlinear wind data, cannot handle large amounts of data easily, and cannot forecast long horizons [18,22,25]. Moreover, short-term wind power forecasting is studied more than long-term wind power forecasting, due to the fact that long-term forecasts do not have high precision rates as short-term forecasts. Lastly, among machine learning methods, random forests, support vector machine, and deep learning methods are getting more attention compared to other methods.

In this study, we investigated the use of several machine learning algorithms for long-term wind power forecasting with the horizon of 1-year ahead using 4-year hourly wind speed data. For this purpose, the regression algorithms of LASSO, kNN, XGBoost, Random Forest, and SVR algorithms were used. The results showed that using machine learning algorithms provides good performance in wind power forecasting. Furthermore, we investigated the use of standard deviation on the model performance. Finally, to better demonstrate the efficiency of our proposed machine learning based wind power forecasting models, we tested our models against the wind speed values of four different locations. The results showed that our models could be applied to a location different from the model-trained locations.

3. Preliminaries and problem definition

In this section, the wind power forecasting problem is defined, wind energy calculations based on hourly wind speed data are presented, and the machine learning algorithms that are used in this study are introduced.

3.1. Problem definition

The main objective of this study is to forecast the generated wind power with respect to daily wind speed data. Another problem of this study is the analysis of eligibility of the generated wind power forecasting model for one location to be used in other locations. To solve these problems, machine learning algorithms were utilized. Hourly wind power values were calculated using the method which is presented in Section 3.2. The algorithms were trained with the observed wind speed data and generated wind power values and tested using only daily mean wind speed values whether they could provide sufficient wind power values with respect to the given wind speed data. Furthermore, the effect of standard deviation was analyzed.

3.2. Wind energy output calculations based on hourly wind speed

Energy output in wind turbines can be calculated via the wind speed data and power curves prepared by wind turbine companies [37,38]. The calculation methodology is based on combining the power curve of the turbine considered with the wind speed data prepared in the form of time series. The power curve of the wind turbine considered in the present research is given in Fig. 1, and its technical specifications are also listed in Table 1. In this study, a 1 MW wind turbine is selected due to their widely usage in the wind power plants.

An algebraic equation of degree n according to the power curve of the wind turbine between the cut-in and rated speed or the cut-in speed and cut-out speed can be formed as shown in Eq. (1), to predict the wind energy output from the wind turbine.

$$P_i(v) = \begin{cases} 0, & v < v_{ci} \\ (a_n v^n + a_{n-1} v^{n-1} + \ldots + a_1 v + a_0), & v_{ci} \leqslant v < v_R \\ P_R, & v_R \leqslant v < v_{co} \\ 0, & v \geqslant v_{co} \end{cases} \tag{1}$$

where a_n, a_{n-1}, a_1 and a_0 are regression constants, v_{ci} is the cut-in speed, v_R is the rated speed, v_{co} is the cut-out speed, P_R is the rated power, and $P_i(v)$ is the power generated in the related wind speed.

The energy output for considered duration from the turbine can be calculated by Eq. (2)

$$E_c = \sum_{i=1}^{N} P(v_i)\Delta t \tag{2}$$

where N is the number of hours in the period of the considered time, such as year, season or month, and Δt is time interval [39].

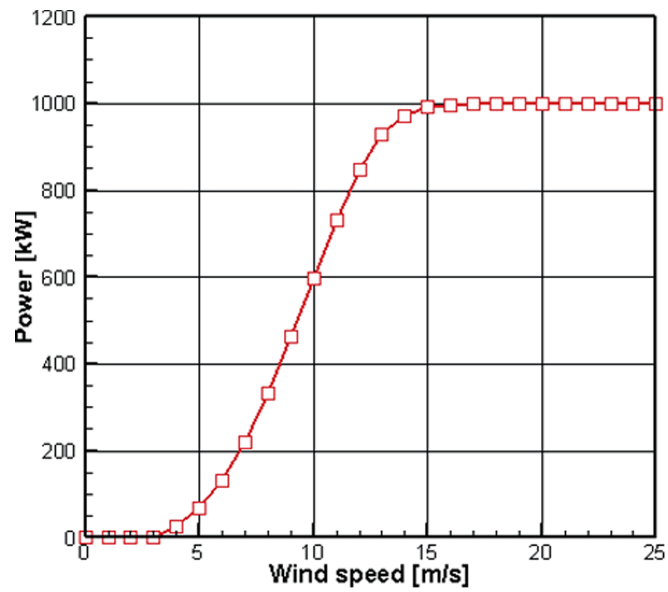

Fig. 1. Power curve for a 1 MW wind turbine.

Table 1
Technical specifications of the considered wind turbine.

Characteristics	Wind Machine
Rated power (kW)	1,000.0
Hub height (m)	50
Rotor diameter (m)	54.2
Swept area (m^2)	2,300.0
Number of blades	3
Cut-in wind speed (V_{ci}) (m/s)	3.0
Rated wind speed (V_R) (m/s)	15.0
Cut-off wind speed (V_{co})	25.0

The energy corresponding to a specific wind speed is computed by the product of the power delivered by the turbine at the wind speed v and the time for which the wind speed v prevails at the studied site. The total energy generated by the turbine over a period of time can be calculated by adding up the energy corresponding to all possible wind speeds under the related conditions, at which the system is operational. In the present research, the hourly mean wind speed is utilized for obtaining the energy output from a turbine.

3.3. Machine learning

Machine learning is a field of computer science domain that tries to provide learning capability to computers or other devices without being explicitly operated. It aims to develop methods and algorithms to learn from data and forecast based on data [4,5]. Machine learning algorithms are successfully utilized to describe the behavior of the dataset, model input features with respect to the expected output, and forecast output features with respect to its historical records.

Machine learning is important for several reasons. First of all, machine learning algorithms perform well when the relationship between inputs and output is not clear or there is no mathematical model on the input–output tuple. Machine learning algorithms can detect environmental changes and adapt to the new environment. Machine learning algorithms can handle complex systems that have many parts and have data flow among these parts.

There are many machine learning methods, each of which is specialized in particular problems. For forecasting purposes, classification and regression algorithms are widely used. Many of these algorithms are originally classification algorithms and modified to produce real values from the input dataset. This subset of classification is called regression analysis.

In this section, we introduced the regression analysis algorithms used in this study, namely LASSO regression, kNN regression, xGBoost regression, Random Forest regression, and Support Vector Regression (SVR). These algorithms were selected because of their widespread usage and high performance in the literature for regression problems. The theoretical background of these algorithms are different from each other, and this would provide information about which background and algorithm is more successful on wind power forecasting problem.

There are several algorithm parameters of each algorithm that affect performance and runtime of the algorithms. To select best parameters for each algorithm based on our problem, we used trial-and-error approach. We run algorithms with different settings of parameters and used best observed results and reported the parameter value at the last paragraph of each algorithm section.

3.3.1. LASSO regression

LASSO (Least Absolute Shrinkage Selector Operator) regression is a specialized version of linear regression [40]. It is also called as shrinkage model because it balances the estimations. The aim of LASSO regression is to acquire the subset of predictors that minimizes the prediction error for a quantitative response variable. LASSO regression is different from linear and ridge regression with the difference of setting the coefficient of some features to zero by parameter. In this way, LASSO can increase or decrease the effect of a feature. LASSO regression is efficient because it performs both variable selection and regularization for the purpose of enhancing prediction accuracy and interpretability of the model.

LASSO performs regression based on Eq. (3), where N is the number of samples, α and β_j are the coefficients of the parameters, and $\hat{\alpha}$ is the prediction.

$$\left(\hat{\alpha}, \hat{\beta}\right) = arg\ min\left\{\frac{1}{N}\sum_{i=1}^{N}\left(y_i - \alpha - \sum_{j=1}^{p} x_{i,j}\cdot\beta_j\right)^2\right\} \tag{3}$$

This formula can be simplified and re-written in Lagrangian form as presented in Eq. (4). As can be seen in Eq. (4), *L1* regularization is preferred in LASSO. *L1* regularization adds the absolute value of coefficients of the features as a penalty term to regularize the effect of the features.

$$\left(\hat{\alpha}, \hat{\beta}\right) = arg\ min\left\{\frac{1}{N}\ \|y - X\beta\|_2^2 + \lambda\ \|\beta\|_1\right\} \tag{4}$$

In this study, LASSO regression was used with a λ parameter of 0.1.

3.3.2. kNN regression

k Nearest Neighbor (kNN) is a well-known and instance-based lazy learning classification algorithm that tries to classify test instances based on their closeness to the number of k class centers in terms of their features [41,42]. The closeness is calculated based on distance measures, i.e. Euclidean, Manhattan or Minkowski distance. It starts with k random points and classifies the training instances based on their closeness to these k centers. An iterative process is performed to model the locations of class centers of k in the best way. Afterward, test instances are classified based on the closeness of their features to these k class centers. kNN regression is the regression version of the kNN classification algorithm that maps test outputs with respect to the given training inputs and their corresponding output values.

The formulation of kNN algorithm is shown in Eq. (5).

$$y' = \underset{v}{argmax} \sum_{(x_i, y_i)\in D_{neighbors}} I\left(v = y_i\right) \tag{5}$$

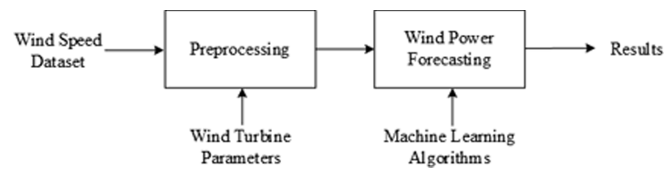

Fig. 2. Experimental setup.

where v is a class label, y_i is the class label of ith nearest neighbors, $I(.)$ is the function that returns the value of 1 if the argument is true and 0 otherwise.

In this study, k was selected as 4 and distance measure was selected as Minkowski distance.

3.3.3. xGBoost regression

The eXtreme Gradient Boost (xGBoost) algorithm is an improved version of the gradient boosting decision tree algorithm that constructs boosted trees in an efficient and parallel manner [43]. In this way, xGBoost algorithm is a fast and scalable algorithm. The idea behind the algorithm is to minimize the objective function by constructing better decision trees. The xGBoost algorithm can also be used for regression purposes. Since it is more efficient and faster than other boosting alternatives, it can find optimal solutions faster.

The formulation of xGBoost regression algorithm is shown in Eq. (6).

$$F_{obj}(\theta) = L(\theta) + \Omega(\theta) where\ L(\theta) = l(\hat{y}_i, y_i) and\ \Omega(\theta) = \gamma T + \frac{1}{2}\lambda\ \|w\|^2 \tag{6}$$

where $F_{obj}(\theta)$ is the objective function, $L(\theta)$ is the loss function between prediction $\hat{y}_i$ and real value y_i, $\Omega(\theta)$ is the regularization term, γ is the learning rate, T is the number of leaves in the tree, λ is the regularization parameter, and w is the weights of the leaves.

In this study, the learning rate was selected as 0.1 and number of trees to fit was selected as 500.

3.3.4. Random forest regression

The Random Forest (RF) algorithm is a popular decision tree algorithm that constructs multiple decision trees from the input dataset [44]. The RF divides input parameters of the dataset into several parts, and constructs decision trees for each part of features, and then the results of each decision tree are used for a final decision. With this methodology, the complex problem of many feature spaces is divided into simpler and more interpretable parts.

In random forest, random vector of θ_k is produced that is a subset of feature space of the dataset, and each tree is constructed using θ_k and training dataset. The generalization error and margin function in random forest are given in Eq. (7).

$$\begin{gathered} PE^* = P_{X,Y}(mg(X, Y) < 0) \\ where\ mg(X, Y) = av_k I(h_k(X) = Y) - max_{j\neq Y} av_k I(h_k(X) = j) \end{gathered} \tag{7}$$

where X and Y are random vectors, mg is the margin function that controls average votes at random vectors for the right output in comparison with any other output, $I(.)$ is the indicator function, and h_k are the classifiers.

In this study, the number of trees was selected as 10 and the random state was selected as 50.

3.3.5. Support vector regression

The Support Vector Regression (SVR) algorithm is regression version of Support Vector Machines (SVM) algorithm [45]. The SVM algorithm constructs a line, plane or hyperplane for one, two or multi-dimensional input space to classify input datasets. Nonlinear SVR tries to find a regression function from the input hyperplanes. SVR is the most common application of the SVM algorithm.

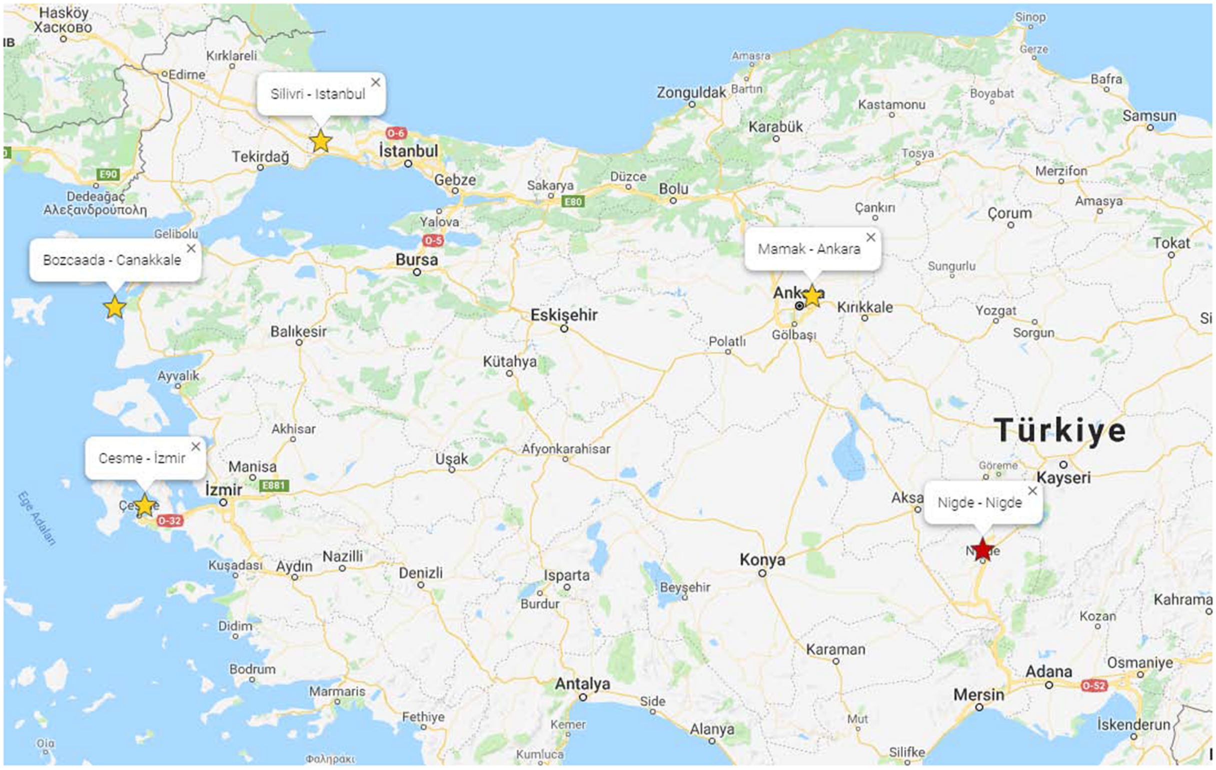

Fig. 3. The locations of the meteorological stations shown on the map.

The SVR algorithm uses training data instances and tries to fit a plane from the input variables within - distance. The basic form of the SVR algorithm is provided in Eq. (8), where w is the space of input patterns, b is the bias, ϵ is the precision, and xi_i and ξ_i^* are the error measures.

$$min\frac{1}{2}\ ||w||^2 + c\sum_{i=1}^{l} \xi_i + \xi_i^*\ subject\ to \begin{cases} y_i - <w, x_i> - b \leqslant \epsilon + \xi_i \\ <w, x_i> + b - y_i \leqslant \epsilon + \xi_i^* \\ \xi_i, \xi_i^* \geqslant 0 \end{cases} \tag{8}$$

There are several parameters that affect the success of SVR, such as the function type, *C*, *gamma* and *epsilon*. The function type is the most effective parameter that should be selected based on input data characteristics. Furthermore, *C* and *gamma* values prevent the model from the over- and under-fitting problem.

In this study, a radial basis function was selected. *C* was selected as 3000 and *gamma* was selected as 0.1.

4. Wind power forecasting method using machine learning algorithms

In the present research, machine learning algorithms were utilized for forecasting the daily total wind power based on the daily mean wind speed and standard deviation. The original dataset contained hourly wind speed values. Hourly wind speed values were converted to daily mean wind speed values and standard deviations. Moreover, hourly wind power values were calculated using hourly wind speed values, and daily total wind power values were calculated using hourly wind power values. Afterward, the machine learning algorithms were trained on 4 years of daily mean wind speed, standard deviation, and the daily total wind power and final one year were forecasted. The methodology of this study is presented in Algorithm 1.

Algorithm 1 Proposed Wind Power Forecasting Method
Input:
D: 5 years of hourly wind speed observations dataset
turbineSpecs: The specifications of the wind turbine
trainingRatio: The ratio of training data instances
Output:
Algorithm:

```
1 hourlyPower = calculate-hourly-power(D, turbineSpecs)
2 [dailyWS, dailySD] = preprocess-dataset(D)
3 dailyPower = calculate-daily-total-power(hourlyPower)
4 [dailyWSTrain, dailySDTrain, dailyWSTest, dailySDTest, dailyPowerTrain,
  dailyPowerTest] = split-train-test(trainingRatio)
5 model = fit-algorithm(dailyWSTrain, dailySDTrain, dailyPowerTrain)
6 forecastedPower = forecast-power(model, dailyWSTest, dailySDTest)
7 metrics = calculate-algorithm-performance(forecastedPower, dailyPowe-
  rTest)
8 return forecastedPower
```

In Algorithm 1, in step 1, hourly wind power calculations are performed, and in step 2, 5 years of hourly wind speed observation dataset *D* is converted to the daily mean and standard deviation format using the preprocess-dataset function. Furthermore, in step 3, hourly power values are converted to daily total power values using the calculate-daily-total-power function. In step 4, daily wind speed, standard deviation, and daily wind power values are split into training and test sets using the provided *trainingRatio* parameter by utilizing the split-train-test function. In step 5, the candidate machine learning algorithms are trained using the training set by utilizing the fit-algorithm function. In step 6, the algorithms return the forecasted power values with respect to the produced model and the daily mean and standard deviation values of the test dataset. In step 7, the performances of the algorithms are evaluated based on their forecasted values using several metrics by utilizing the calculate-algorithm-performance function. Finally, in step 8, the algorithm returns the forecasted values.

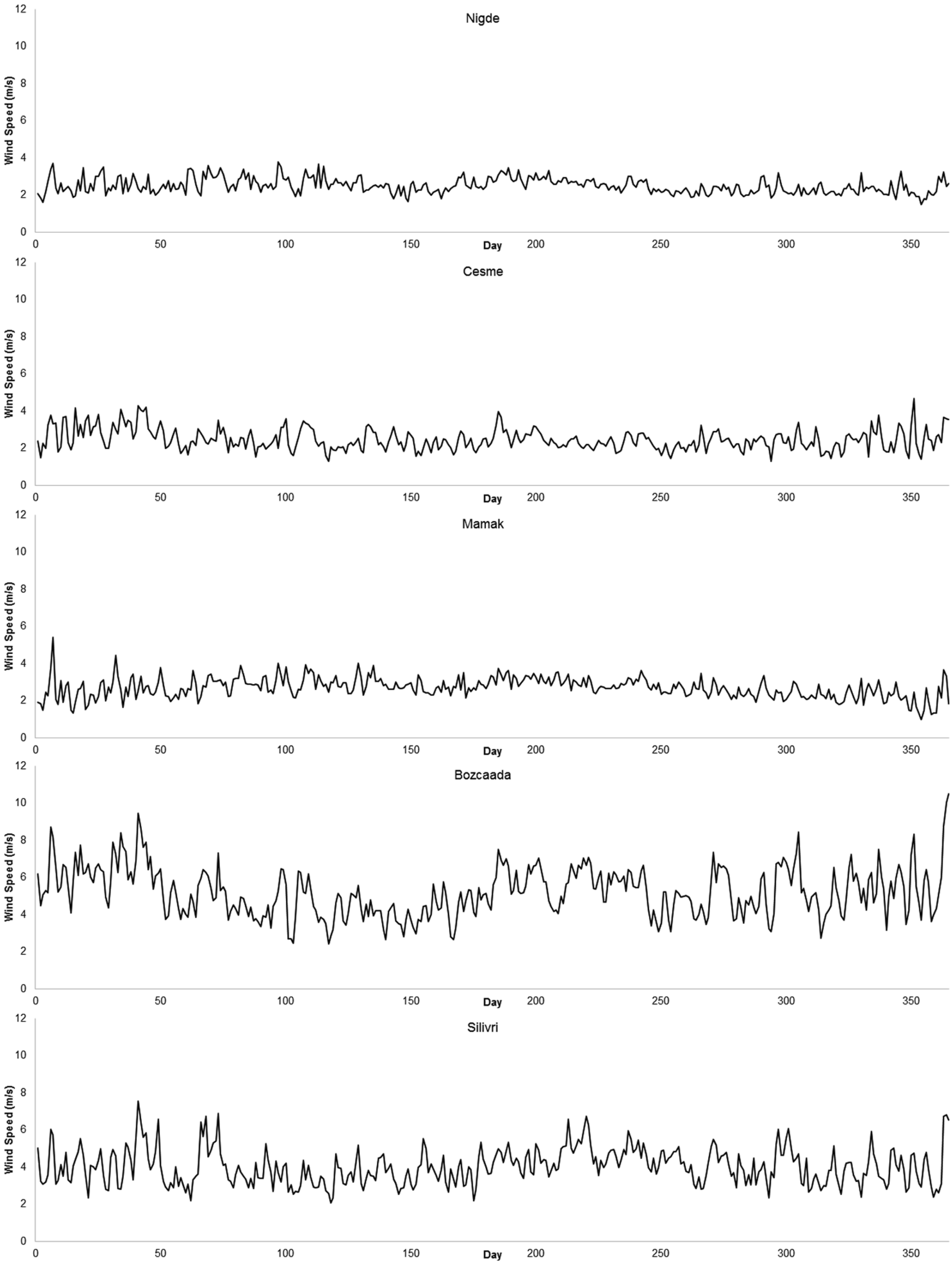

Fig. 4. Wind speed characteristics of selected locations.

5. Results and discussion

In this section, first, the dataset is presented, and then the performances of machine learning algorithms for wind power forecasting are investigated. The algorithms are evaluated based on their R^2 values and the Mean Absolute Error (MAE) and the Root Mean Squared Error (RMSE) metrics. The experimental setup is presented in Fig. 2.

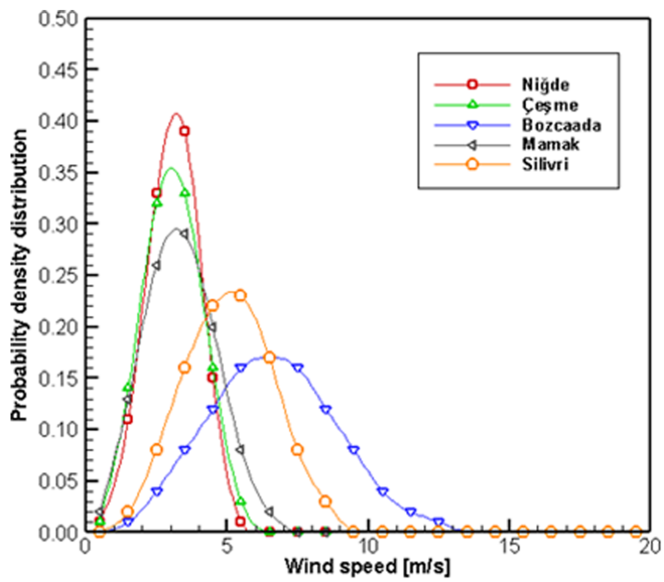

Fig. 5. Wind speed frequency distributions in the selected locations.

Table 2
Calculated Weibull distribution parameters for selected locations and 50 m hub height.

Location	k	c (m/s)
Nigde	3.69	3.45
Cesme	3.17	3.44
Bozcaada	3.20	7.28
Mamak	2.78	3.77
Silivri	3.44	5.60

5.1. The dataset

The original dataset for this study is 5 years of hourly wind speed observation values of Nigde, Turkey. A total of 43.800 observations are present in the dataset for a total of 5 years. The generated wind power for each hour is calculated based on the calculations which are explained in Section 3.2. The resulting dataset contains hourly wind speed values and corresponding power values or each hour of the dataset. Four other datasets were generated for different regions of Turkey, namely Cesme, Mamak, Bozcaada, and Silivri, with the same methodology as Nigde dataset. The locations of the meteorological stations are shown on the map in Fig. 3.

Fig. 4 provides wind speed characteristics of selected 5 locations. Daily average wind speed values are plotted in the figure that are average of 5 years daily wind speed values. As can be seen in the figure, best wind efficient location is Bozcaada since it has highest wind speed values for all year long. Also, Silivri is another location that has high wind speed values. Cesme, Mamak, and Nigde have seasonal fluctuations in wind speed values, they are higher in winter months and getting lower and flatten in summer months.

First of all, the wind speed data of the meteorological stations are observed at a 10 m height. As we will use wind speed values at a wind turbine height to produce the generated wind power accurately, we need to extrapolate 10 m hourly wind speed values to 50 m. We used the power law equation, which is given below in Eq. (9), to obtain hourly wind speed values. We used h as 50 m, h_0 as 10 m, and αas 0.14 as used in [46].

$$v = v_0\left(\frac{h}{h_0}\right)^{\alpha} \tag{9}$$

In the wind energy investigations, it is crucial to have only a few key parameters that can explain the behaviour of a wide range of wind speed data. The simplest and most practical method for the procedure is to use a distribution function. The probability density function shows the frequency of observing different levels of speed. In this study, for showing the wind speed distribution, Weibull distribution was used. The Weibull parameters (k and c) were determined using mean wind speed-standard deviation method [46]. Fig. 5 reveals Weibull distributions with the two parameters derived from observed data for selected locations and 50 m hub height. As can be seen from this figure, the peak probability value for the Nigde is 0.41, while this value is 0.23 for Silivri and 0.17 for Bozcaada. Table 2 lists the calculated Weibull distribution parameters for different locations and 50 m hub height. As can be from this table, the Weibull shape parameter (k) and the scale parameter (c) that is directly proportional to mean wind speed were different for each selected location.

After hourly wind speed observations are extrapolated to a 50 m height, the dataset is preprocessed for obtaining daily mean wind speed values and the standard deviations of daily values. Furthermore, the generated power of 5 years is converted to daily total wind power values by summing each power value of hours of the corresponding day. Finally, the dataset is split into training and test sets, by considering wind characteristics and machine learning requirements. For this

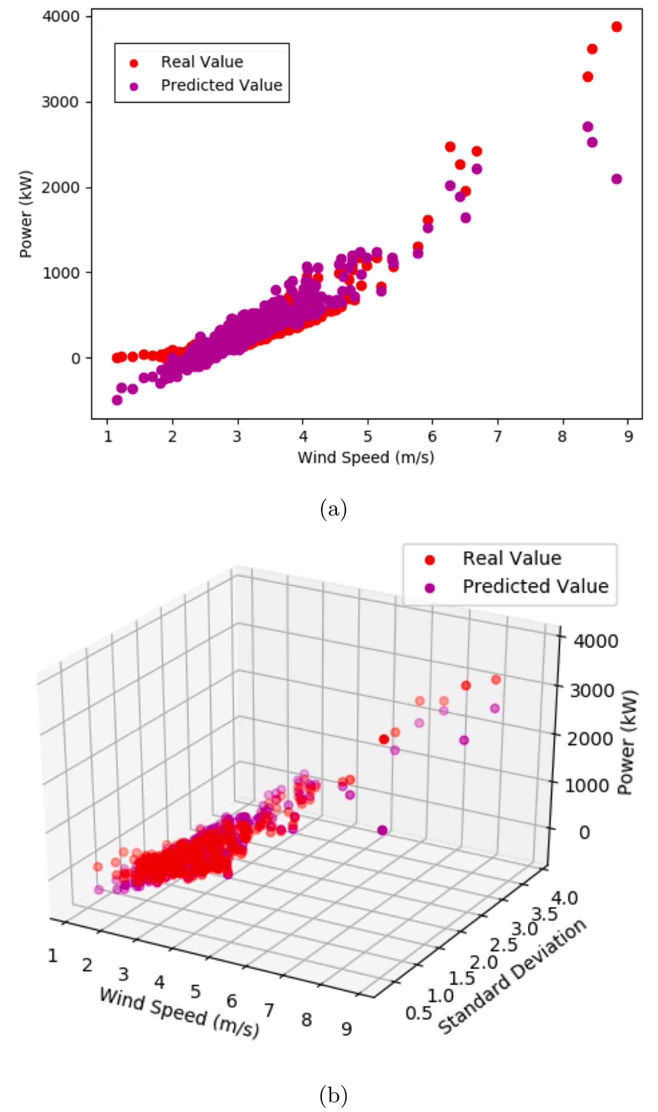

Fig. 6. a) Forecasted power and b) 3D visualization of the effect of wind speed and standard deviation using LASSO regression.

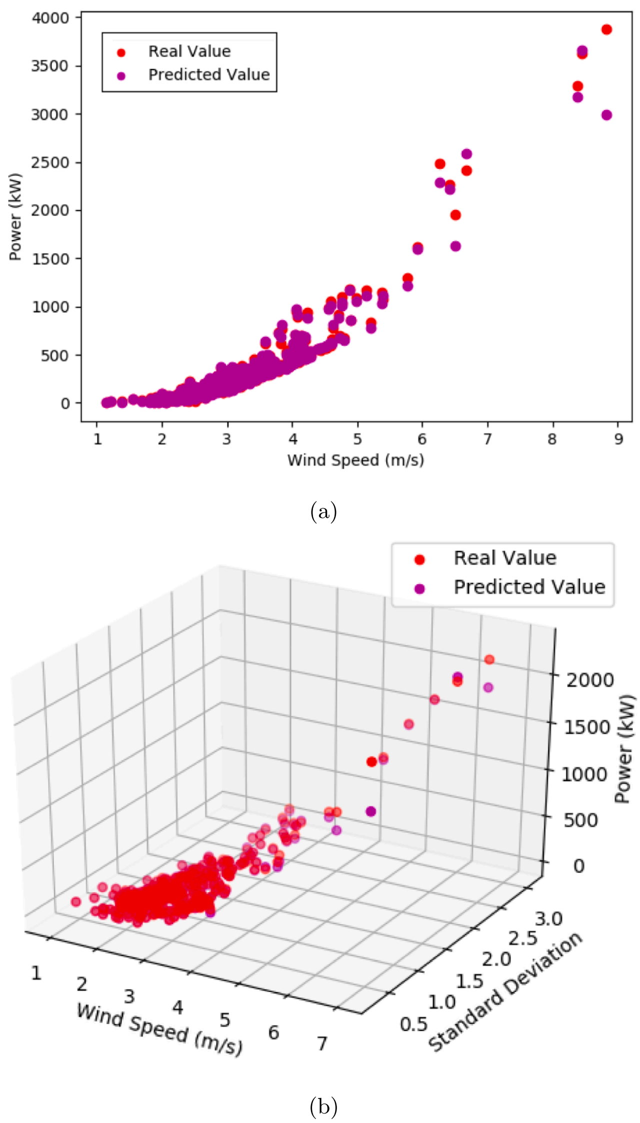

Fig. 7. a) Forecasted power and b) 3D visualization of the effect of wind speed and standard deviation using kNN regression.

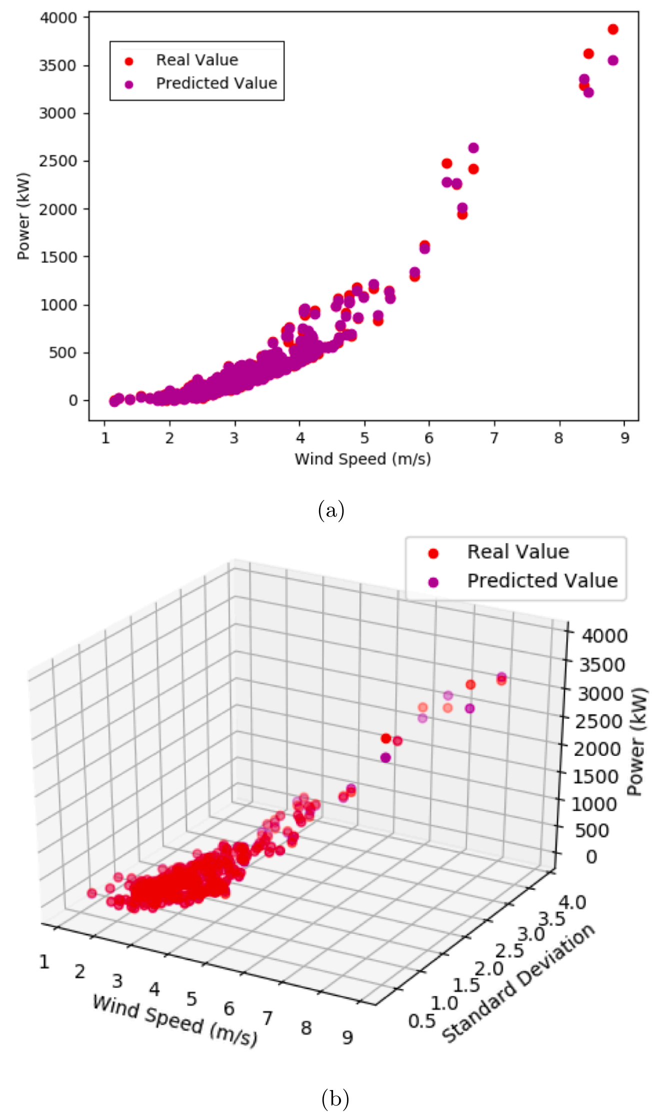

Fig. 8. a) Forecasted power and b) 3D visualization of the effect of wind speed and standard deviation using xGBoost regression.

purpose, the *trainingRatio* of Algorithm 1 is set at 0.8 to select 80% of the dataset as a training set, which corresponds to 4 years, and the remaining 20% of the dataset is selected as a test set, that is the final year. We selected an 80% value for the training set to properly split the dataset based on its seasonality and to train with 4 years of the dataset and test with a one year of dataset.

5.2. Case studies

In this section, the experimental results for machine learning algorithms are presented. The experiments are performed for three cases.

- **Case 1:** The first case is that, given 4 years of daily mean wind speed values and standard deviation and their corresponding daily total generated wind power values, we forecasted 1-year wind power given the daily mean wind speed values and standard deviations of the fifth year.
- **Case 2:** The second case is similar to the first one, but in this case, we used only daily mean wind speed values to forecast 1-year wind power.
- **Case 3:** The third case is that, given 4 years of daily mean wind speed values and standard deviation of Nigde, Turkey as a training set, machine learning algorithms were used to forecast 1-year wind power values of the year 2017 for Cesme, Mamak, Bozcaada, and Silivri, Turkey. The model was built using Nigde data, and the forecasted values were used for four different candidate locations.

Case 1 is performed to test the performance of machine learning algorithms on wind power forecasting using the daily mean wind speed and standard deviation of a location. Case 2 is performed for the purpose of testing the performance of machine learning algorithms when there is no standard deviation. Case 3 is performed to test the performance of machine learning algorithms that build a model on a location and test the model in different locations. The results of each case are presented below.

5.2.1. Case 1: Wind power forecasting based on daily mean wind speed and standard deviation

In this case, the daily total generated power was forecasted using previous daily mean wind speed and standard deviation. The forecasted wind power values were compared with the original generated power values of the fifth year. The results of this case are presented as follows.

Fig. 6 presents the results of LASSO regression. The most important thing to note is that LASSO forecasts negative power for low wind speed

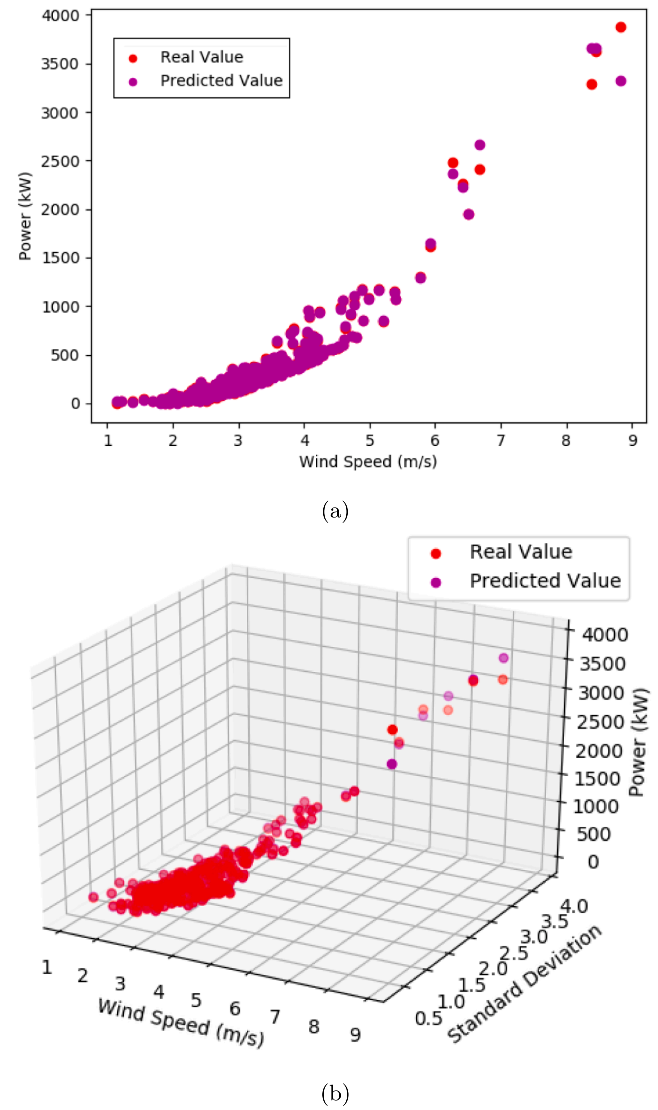

Fig. 9. a) Forecasted power and b) 3D visualization of the effect of wind speed and standard deviation using SVR.

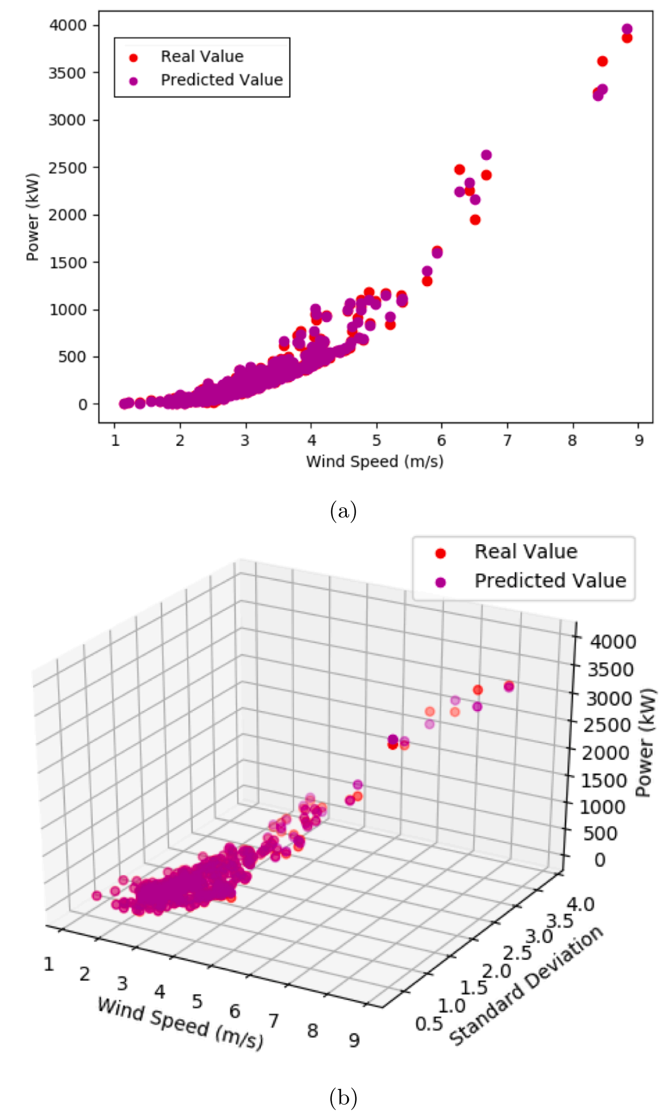

Fig. 10. a) Forecasted power and b) 3D visualization of the effect of wind speed and standard deviation using RF regression.

observations. The main reason for this behavior is that LASSO is a type of linear regression and it tries to fit the model onto a linear plane. As can be seen in Fig. 6 b), standard deviation has an effect on the forecasted power. Lower standard deviations are more accurately forecasted by LASSO, while high standard deviations are not forecasted well.

Fig. 7 presents the results of kNN regression. As can be seen in Fig. 7 a), the kNN algorithm could forecast power values positively. Moreover, its forecasting performance is better in comparison with LASSO regression. Higher wind speed values are not forecasted correctly by the kNN algorithm. Furthermore, as can be seen in Fig. 7 b), kNN could not handle the increase in standard deviation and forecasted wrong values. However, the overall performance of the kNN algorithm is good since it has a high R^2 value.

Fig. 8 presents the results of xGBoost regression. As can be seen in Fig. 8 a), similarly to the kNN algorithm, the xGBoost algorithm could not forecast correct power values for higher wind speed values, but has better accuracy than the kNN algorithm. As can be seen in Fig. 8b), it could handle high standard deviation and could forecast correct values for high standard deviations.

Fig. 9 presents the results of SVR. As can be observed in Fig. 9a), SVR could correctly forecast power values of low wind speed values. However, high wind speed values could not be fit into a correct model for SVR and forecast errors are observed. Furthermore, Fig. 9b) shows that SVR could also handle the variation at standard deviation.

Fig. 10 presents the results of Random Forest regression. As can be seen in Fig. 10a), RF could be observed to be more successful at forecasting both low and high wind speed values and is better at handling high wind speed values in comparison with SVR. Moreover, as can be seen in Fig. 10b), RF could adapt to the change in standard deviation and successfully forecast power based on wind speed and standard deviation.

Fig. 11 presents the forecasting accuracy of the algorithms for the fifth year. As can be seen in the figure, LASSO could not fit the curve to the highest and lowest values. Other algorithms are comparably good. Among these algorithms, kNN has the worst R^2 value. However the value is not lower than that of other algorithms The RF algorithm is observed to have the best performance since it successfully fits the curve based on the actual power value both at peaks and standard values.

Table 3 presents the forecasting accuracy of machine learning algorithms for the wind power forecasting problem. Among the five candidate algorithms, LASSO regression is the worst algorithm since it tries to fit the model to a linear plane. The kNN and xGBoost regression

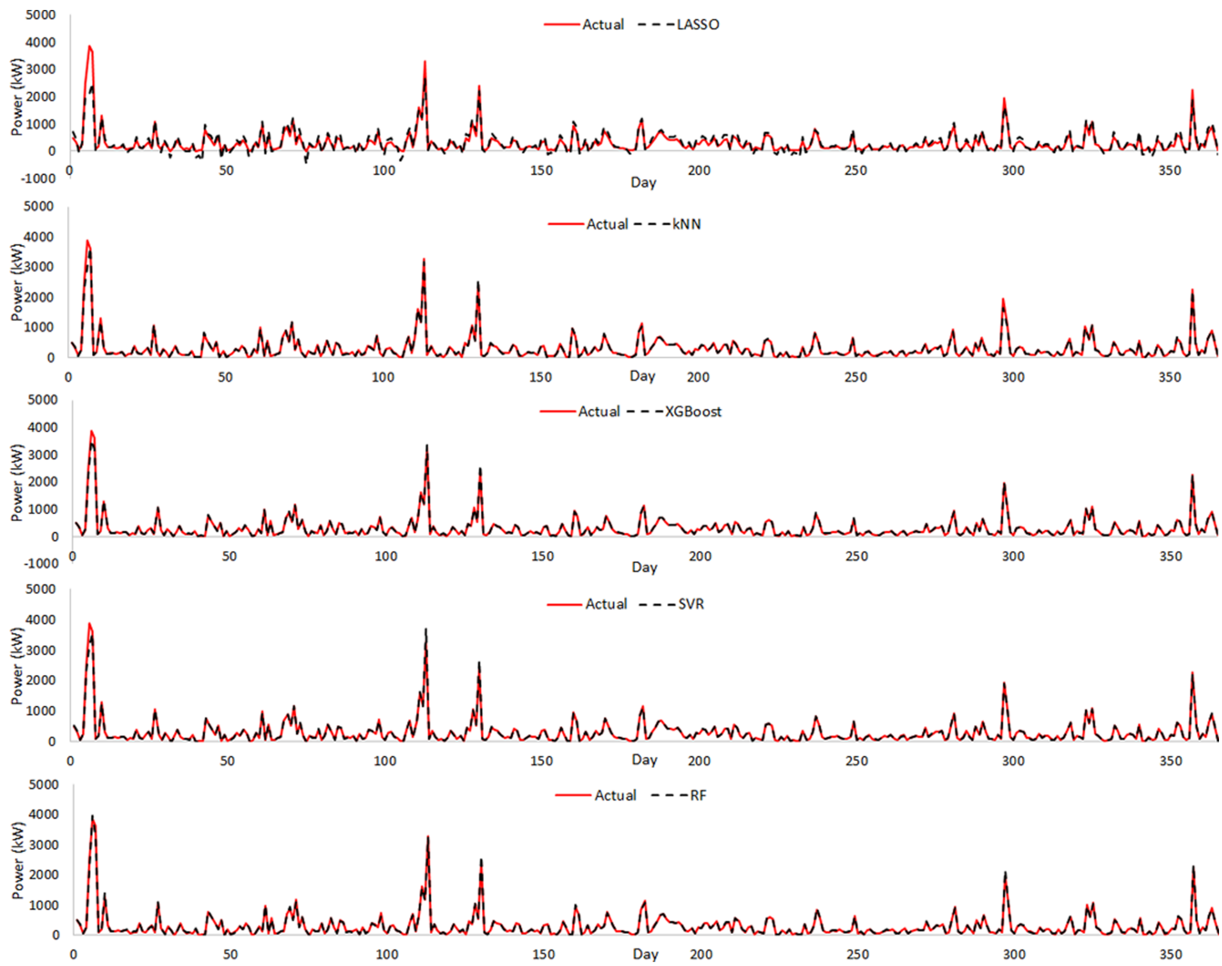

Fig. 11. Daily generated power values and forecasts of the algorithms for a one-year horizon.

Table 3
Forecasting accuracy of the algorithms for daily wind speed and standard deviation.

Algorithm/Metric	R^2	RMSE	MAE
LASSO	0.8619	164.61	88.85
kNN	0.9852	53.82	7.197
xGBoost	0.9939	34.40	6.528
SVR	0.992	38.52	**5.430**
RF	**0.995**	**30.224**	7.048

The bold values are the results that are observed as best performance within the tables.

Table 4
Forecasting accuracy of the algorithms using only daily wind speed.

Algorithm/Metric	R^2	RMSE	MAE
LASSO	0.823	186.10	94.67
kNN	0.943	105.76	42.47
xGBoost	0.932	115.47	41.58
SVR	**0.955**	**93.13**	**32.63**
RF	0.921	123.88	42.76

The bold values are the results that are observed as best performance within the tables.

algorithms are better than LASSO regression, however, they have different limitations. SVR and RF can be seen as more successful because they have a higher R^2 value and lower error values. Among these two algorithms, RF has better accuracy with respect to R^2 and RMSE values.

The results of this case show that, machine learning algorithms could be applied to long-term wind power forecasting problem and produce successful results.

5.2.2. Case 2: Wind power forecasting based on only daily mean wind speed

In this case, we used only daily mean wind speed values for forecasting the daily total wind power. This case aims to reveal the effect of standard deviation on our models and to see whether the models produce reasonable results when standard deviation values are not present. The results of the algorithms are presented as follows.

Table 4 presents the performances of the algorithms based on only daily wind speed values. As can be observed from the table, the accuracy and the performances of the algorithms decrease and error rates increase in comparison with Table 3. However, the results are still acceptable for the algorithms other than LASSO since they produce R^2 values higher than 0.9. In addition, when there is no standard deviation, the models are still acceptable. Especially, the SVR algorithm produces the best results. The greatest decrease is observed with the RF algorithm, since its performance increases when there are more features.

The results of this case show that machine learning algorithms could build reliable wind power models using only wind speed values even there are no standard deviation to forecast long-term wind power values of a location.

5.2.3. Case 3: Wind power forecasting for a different region

In this case, we used our Nigde-trained wind power forecasting model for forecasting four different regions of Turkey. The aim of this case is to reveal the performances of the algorithms when test values

Table 5
Forecasting accuracy of the algorithms for a different test location with and without standard deviation (WS + STD indicates that wind speed and standard deviation are used, and WS indicates that only wind speed is used).

Algorithm/Metric	Cesme			Mamak			Bozcaada			Silivri		
	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE	R^2	RMSE	MAE
LASSO (WS + STD)	0.7696	206.22	136.64	0.8970	200.71	119.84	0.7691	621.61	148.91	0.8135	450.41	124.17
LASSO (WS)	0.7462	216.44	166.24	0.8456	251.26	127.12	0.8415	514.92	177.01	0.8531	442.84	211.16
kNN (WS + STD)	0.9872	48.47	8.187	0.9778	93.145	10.128	0.9106	386.81	39.482	0.9579	213.92	124.17
kNN (WS)	0.9092	129.43	32.26	*0.9334*	*165.04*	64.450	0.9162	374.30	147.45	0.9199	326.90	127.40
xGBoost (WS + STD)	0.9922	37.83	7.942	**0.9851**	**76.148**	10.158	**0.9526**	**281.38**	24.490	**0.9845**	**129.49**	15.498
xGBoost (WS)	0.8993	136.34	26.653	0.9302	168.84	56.04	*0.9324*	*336.32*	138.04	0.9315	302.45	113.07
SVR (WS + STD)	**0.9992**	**11.487**	**6.4941**	0.9652	116.68	**7.7585**	0.8843	439.87	**12.83**	0.9766	159.32	**7.438**
SVR (WS)	*0.9362*	*108.46*	*22.310*	0.9260	173.90	*51.19*	0.9155	376.03	*130.49*	*0.9346*	*295.34*	*105.80*
RF (WS + STD)	0.9924	37.431	7.9680	0.9817	84.452	10.024	0.9509	286.49	28.339	0.9826	137.58	16.020
RF (WS)	0.8823	147.40	34.444	0.9242	176.00	72.41	0.9257	352.50	143.39	0.9230	320.67	128.74

The bold values are the results that are observed as best performance within the tables.

are not consistent with training values. We selected Nigde as training set, because we wonder whether the algorithms could handle different wind characteristics that has high wind speed locations, such as Bozcaada, and Silivri. Four different locations were selected for this case, such as Cesme, Mamak, Bozcaada, and Silivri. The locations are from different geographical regions, and each of them has different wind characteristics. The difference in the wind characteristics of these four locations could be clearly seen in Fig. 4 and Fig. 5. The results of the algorithms are presented below.

Table 5 presents the performances of the algorithms for four different locations with and without standard deviation. As presented in Cases 1 and 2, using standard deviation with wind speed increases algorithm performances. Furthermore, the algorithms exhibit slightly poorer performance compared to Cases 1 and 2. The main reason for this behavior is that the wind characteristics of these locations are not consistent with Nigde wind characteristics. However, the performances of the algorithms are still high, i.e. above 0.9, for all of the locations with the exception of LASSO.

These results show that using a pilot region for the training set and using the generated model with the training data could be used for other locations that have different wind characteristics. Moreover, an interesting result is obtained from the SVR algorithm since its accuracy value and error measures are more satisfying than other algorithms for different locations.

6. Conclusions

Wind energy is one of the main renewable energy sources due to its natural, cheap, and clean nature. It is possible to produce energy from wind turbines at each hour of the day, and it is suitable for systems that require energy continuously. However, using wind energy is challenging due to its initial investment costs, the requirement of careful analyses before establishing a wind plant, the distance of wind-efficient areas to the national grids, and its environmentally disruptive effects.

In this study, wind power forecasting was performed based on daily wind speed data using machine learning algorithms. In particular, classification algorithms were used to forecast values of the given wind speed values. Daily mean wind speed values were generated using the hourly wind speed dataset, and the daily total wind power was modeled using daily wind speed and also the standard deviation. The proposed method was applied to different locations to see whether the algorithms could produce acceptable results with respect to the trained location. The results showed that the xGBoost, SVR, and RF algorithms are powerful in forecasting long-term daily total wind power. Among these algorithms RF is the best algorithm with R^2 value of 0.995, and MAE of 7.048. LASSO is the worst algorithm due to its linear basis. However, 0.862 R^2 value of LASSO is relatively acceptable. When standard deviation is excluded from the dataset, SVR becomes best algorithm with R^2 value of 0.955 and MAE of 32.63. Moreover, these algorithms, xGBoost, SVR and RF, are powerful in forecasting the daily total wind power values of locations other than the model-trained location. R^2 values of these algorithms for forecasting wind power of different locations are above 0.95. Bozcaada has the lowest R^2 value, due to its distinctive and higher wind speed characteristics with respect to other locations.

An important outcome of this study is that machine learning algorithms could be successfully used before the establishment of wind plants in an unknown geographical location whether it is logical by using the wind power model of a base location.

Declaration of Competing Interest

None.

Acknowledgements

We would like to thank the Turkish State Meteorological Service for providing wind speed data.

References

[1] Jung J, Broadwater RP. Current status and future advances for wind speed and power forecasting. Renew. Sustain. Energy Rev. 2014;31:762–77. https://doi.org/10.1016/j.rser.2013.12.054. URL: http://www.sciencedirect.com/science/article/pii/S1364032114000094.
[2] Wang X, Guo P, Huang X. A review of wind power forecasting models. Energy Procedia 2011;12:770–8. https://doi.org/10.1016/j.egypro.2011.10.103. URL: http://www.sciencedirect.com/science/article/pii/S1876610211019291.
[3] Chang W-Y. A literature review of wind forecasting methods. J. Power Energy Eng. 2014;2:161–8. https://doi.org/10.4236/jpee.2014.24023. URL: https://www.scirp.org/journal/PaperInformation.aspx?PaperID=44881.
[4] Samuel AL. Some studies in machine learning using the game of checkers ii – recent progress. IBM J. Res. Dev. 1967;11(6):601–17. https://doi.org/10.1147/rd.116.0601.
[5] Alpaydin E. Introduction to Machine Learning. MIT Press; 2009.
[6] Rajagopalan S, Santoso S. Wind power forecasting and error analysis using the autoregressive moving average modeling. IEEE Power Energy Society General Meeting 2009;2009:1–6. https://doi.org/10.1109/PES.2009.5276019.
[7] Abdelaziz A, Rahman MA, El-Khayat M, Hakim MA. Short term wind power forecasting using autoregressive integrated moving average modeling. Proceedings of the 15th International Middle East Power Systems Conference (MEPCON'12). 2012. pp. 208:1–208: 6.
[8] Cadenas E, Rivera W, Campos-Amezcua R, Heard C. Wind speed prediction using a univariate ARIMA model and a multivariate NARX model. Energies 2016;9(2). https://doi.org/10.3390/en9020109. URL: http://www.mdpi.com/1996-1073/9/2/109.
[9] Dowell J, Pinson P. Very-short-term probabilistic wind power forecasts by sparse vector autoregression. IEEE Trans. Smart Grid 2016;7(2):763–70. https://doi.org/10.1109/TSG.2015.2424078.
[10] Lima JM, Guetter AK, Freitas SR, Panetta J, de Mattos JGZ. A meteorological-statistic model for short-term wind power forecasting. J. Control Autom. Electr. Syst. 2017;28(5):679–91. doi: 10.1007/s40313-017-0329-8.
[11] Wang J, Zhou Q, Zhang X. IOP Conf. Ser.: Earth Environ. Sci. 2018;199:022015https://doi.org/10.1088/1755-1315/199/2/022015.

URL: https://doi.org/10.1088.

[12] Robles-Rodriguez C, Dochain D. Decomposed threshold armax models for short- to medium-term wind power forecasting. IFAC-PapersOnLine 2018;51(13):49–54. https://doi.org/10.1016/j.ifacol.2018.07.253. 2nd IFAC Conference on Modelling, Identification and Control of Nonlinear Systems MICNON 2018.http://www.sciencedirect.com/science/article/pii/S240589631831005X.

[13] Pearre NS, Swan LG. Statistical approach for improved wind speed forecasting for wind power production. Sustainable Energy Technol. Assess. 2018;27:180–91. https://doi.org/10.1016/j.seta.2018.04.010. URL: http://www.sciencedirect.com/science/article/pii/S221313881730512X.

[14] Sideratos G, Hatziargyriou ND. An advanced statistical method for wind power forecasting. IEEE Trans. Power Syst. 2007;22(1):258–65. https://doi.org/10.1109/TPWRS.2006.889078.

[15] Rahmani R, Yusof R, Seyedmahmoudian M, Mekhilef S. Hybrid technique of ant colony and particle swarm optimization for short term wind energy forecasting. J. Wind Eng. Ind. Aerodyn. 2013;123:163–70. https://doi.org/10.1016/j.jweia.2013.10.004. URL: http://www.sciencedirect.com/science/article/pii/S0167610513002249.

[16] Najeebullah, Zameer A, Khan A, Javed SJ. Machine learning based short term wind power prediction using a hybrid learning model. Comput. Electr. Eng. 2015;45:122–33. https://doi.org/10.1016/j.compeleceng.2014.07.009. URL: http://www.sciencedirect.com/science/article/pii/S0045790614001876.

[17] Chi Z, Haikun W, Tingting Z, Kanjian Z, Tianhong L. Comparison of two multi-step ahead forecasting mechanisms for wind speed based on machine learning models. 2015 34th Chinese Control Conference (CCC) 2015. p. 8183–7. https://doi.org/10.1109/ChiCC.2015.7260941.

[18] Peng T, Zhou J, Zhang C, Zheng Y. Multi-step ahead wind speed forecasting using a hybrid model based on two-stage decomposition technique and adaboost-extreme learning machine. Energy Convers. Manage. 2017;153:589–602. https://doi.org/10.1016/j.enconman.2017.10.021. URL: http://www.sciencedirect.com/science/article/pii/S0196890417309238.

[19] Lahouar A, Slama JBH. Hour-ahead wind power forecast based on random forests. Renewable Energy 2017;109:529–41. https://doi.org/10.1016/j.renene.2017.03.064. URL: http://www.sciencedirect.com/science/article/pii/S0960148117302550.

[20] Li C, Lin S, Xu F, Liu D, Liu J. Short-term wind power prediction based on data mining technology and improved support vector machine method: a case study in northwest china. J. Cleaner Prod. 2018;205:909–22. https://doi.org/10.1016/j.jclepro.2018.09.143. URL: http://www.sciencedirect.com/science/article/pii/S095965261832866X.

[21] Sun G, Jiang C, Cheng P, Liu Y, Wang X, Fu Y, He Y. Short-term wind power forecasts by a synthetical similar time series data mining method. Renewable Energy 2018;115:575–84. https://doi.org/10.1016/j.renene.2017.08.071. URL: http://www.sciencedirect.com/science/article/pii/S0960148117308327.

[22] Wang K, Qi X, Liu H, Song J. Deep belief network based k-means cluster approach for short-term wind power forecasting. Energy 2018;165:840–52. https://doi.org/10.1016/j.energy.2018.09.118. URL: http://www.sciencedirect.com/science/article/pii/S0360544218318826.

[23] Zheng D, Semero YK, Zhang J, Wei D. Short-term wind power prediction in microgrids using a hybrid approach integrating genetic algorithm, particle swarm optimization, and adaptive neuro-fuzzy inference systems. IEEJ Trans. Electr. Electron. Eng. 2018;13(11):1561–7.

[24] Yu R, Gao J, Yu M, Lu W, Xu T, Zhao M, Zhang J, Zhang R, Zhang Z. LSTM-EFG for wind power forecasting based on sequential correlation features. Future Gener. Comput. Syst. 2019;93:33–42. https://doi.org/10.1016/j.future.2018.09.054. URL: http://www.sciencedirect.com/science/article/pii/S0167739X18314420.

[25] Qin Y, Li K, Liang Z, Lee B, Zhang F, Gu Y, Zhang L, Wu F, Rodriguez D. Hybrid forecasting model based on long short term memory network and deep learning neural network for wind signal. Appl. Energy 2019;236:262–72. https://doi.org/10.1016/j.apenergy.2018.11.063. URL: http://www.sciencedirect.com/science/article/pii/S0306261918317690.

[26] Shi X, Lei X, Huang Q, Huang S, Ren K, Hu Y. Hourly day-ahead wind power prediction using the hybrid model of variational model decomposition and long short-term memory. Energies 2018;11:3227:1–3227:20. https://doi.org/10.3390/en11113227. URL: https://www.mdpi.com/1996-1073/11/11/3227.

[27] Eldali FA, Hansen TM, Suryanarayanan S, Chong EKP. Employing ARIMA models to improve wind power forecasts: a case study in ERCOT. North Am. Power Symp. (NAPS) 2016;2016:1–6. https://doi.org/10.1109/NAPS.2016.7747861.

[28] de Alencar DB, deMattos Affonso C, de Oliveira RCL, Rodriguez JLM, Leite JC, Filho JCR. Different models for forecasting wind power generation: case study. Energies 1976;10(2017). https://doi.org/10.3390/en10121976. pp. 1–1976: 27.

[29] Ekstrom J, Koivisto M, Mellin I, Millar RJ, Lehtonen M. A statistical modeling methodology for long-term wind generation and power ramp simulations in new generation locations. Energies 2018;11:2442:1–2442:27. https://doi.org/10.3390/en11092442.

[30] Dokuz AS, Demolli H, Gokcek M, Ecemis A. Year-ahead wind speed forecasting using a clustering-statistical hybrid method. International Conference on Innovative Engineering Applications. 2018. p. 971–5. CIEA'2018.

[31] Barbounis TG, Theocharis JB, Alexiadis MC, Dokopoulos PS. Long-term wind speed and power forecasting using local recurrent neural network models. IEEE Trans. Energy Convers. 2006;21(1):273–84. https://doi.org/10.1109/TEC.2005.847954.

[32] Khan GM, Ali J, Mahmud SA. Wind power forecasting – an application of machine learning in renewable energy. International Joint Conference on Neural Networks (IJCNN) 2014;2014:1130–7. https://doi.org/10.1109/IJCNN.2014.6889771.

[33] Wang J, Qin S, Zhou Q, Jiang H. Medium-term wind speeds forecasting utilizing hybrid models for three different sites in Xinjiang, China. Renewable Energy 2015;76:91–101. https://doi.org/10.1016/j.renene.2014.11.011. URL: http://www.sciencedirect.com/science/article/pii/S0960148114007265.

[34] Dumitru C-D, Gligor A. Daily average wind energy forecasting using artificial neural networks. Proc. Eng. 2017;181:829–36. https://doi.org/10.1016/j.proeng.2017.02.474. 10th International Conference Interdisciplinarity in Engineering, INTER-ENG 2016, 6–7 October 2016, Tirgu Mures, Romania.http://www.sciencedirect.com/science/article/pii/S1877705817310597.

[35] Yan J, Ouyang T. Advanced wind power prediction based on data-driven error correction. Energy Convers. Manage. 2019;180:302–11. https://doi.org/10.1016/j.enconman.2018.10.108. URL: http://www.sciencedirect.com/science/article/pii/S0196890418312366.

[36] Maroufpoor S, Sanikhani H, Kisi O, Deo RC, Yaseen ZM. Long-term modelling of wind speeds using six different heuristic artificial intelligence approaches. Int. J. Climatol. 2019;2019:1–15.

[37] Gokcek M, Genc MS. Evaluation of electricity generation and energy cost of wind energy conversion systems (WECSs) in central turkey. Appl. Energy 2009;86(12):2731–9. https://doi.org/10.1016/j.apenergy.2009.03.025. URL: http://www.sciencedirect.com/science/article/pii/S0306261909001433.

[38] Burton T, Jenkins N, Sharpe D, Bossanyi E. Introduction to Machine Learning. John Wiley & Sons Ltd; 2011.

[39] Manwell JF, McGowan JG, Rogers AL. Wind Energy Explained: Theory, Design and Application. Wiley; 2010.

[40] Tibshirani R. Regression shrinkage and selection via the lasso. J. R. Stat. Soc. Ser. B (Methodological) 1996;58(1):267–88. URL: http://www.jstor.org/stable/2346178.

[41] Yao Z, Ruzzo WL. A regression-based k nearest neighbor algorithm for gene function prediction from heterogeneous data. BMC Bioinf. 2006;7(1):S11. https://doi.org/10.1186/1471-2105-7-S1-S11.

[42] Hu C, Jain G, Zhang P, Schmidt C, Gomadam P, Gorka T. Data-driven method based on particle swarm optimization and k-nearest neighbor regression for estimating capacity of lithium-ion battery. Appl. Energy 2014;129:49–55. https://doi.org/10.1016/j.apenergy.2014.04.077. URL: http://www.sciencedirect.com/science/article/pii/S0306261914004401.

[43] Chen T, Guestrin C. Xgboost: a scalable tree boosting system. Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16 New York, NY, USA: ACM; 2016. p. 785–94. https://doi.org/10.1145/2939672.2939785.

[44] Breiman L. Random forests. Mach. Learn. 2001;45(1):5–32. https://doi.org/10.1023/A:1010933404324.

[45] Drucker H, Burges CJ, Kaufman L, Smola AJ, Vapnik V. Support vector regression machines. Advances in neural information processing systems. 1997. p. 155–61.

[46] Gokcek M, Bayulken A, Bekdemir S. Investigation of wind characteristics and wind energy potential in Kirklareli, Turkey. Renewable Energy 2007;32(10):1739–52. https://doi.org/10.1016/j.renene.2006.11.017. URL: http://www.sciencedirect.com/science/article/pii/S0960148106003405.