Azerbaijan Journal of High Performance Computing, Vol 5, Issue 1, 2022, pp. 57-71
https://doi.org/10.32010/26166127.2022.5.1.57.71

A Comparative Assessment of Machine Learning Models For Predicting Wind Speed

Navid Atashfaraz[1], Faezeh Gholamrezaie[2], Arash Hosseini[3] and Nigar Ismayilova[4]

[1] North Tehran Branch Azad University, Tehran, Iran. n.atashfaraz@iau-tnb.ac.ir
[2] Shahed University, Tehran, Iran. faeze.gholamrezaie@shahed.ac.ir
[3] North Tehran Branch Azad University, Tehran, Iran. a.hosseini@iau-tnb.ac.ir
[4] Azerbaijan State Oil and Industry University Baku, Azerbaijan, nigar.ismailova@asoiu.edu.az

*Correspondence: Faezeh Gholamrezaie, Department of Computer Science, Shahed University, Tehran, Iran. faeze.gholamrezaie@shahed.ac.ir

Abstract

Renewable energy is one of the most critical issues of continuously increasing electricity consumption which is becoming a desirable alternative to traditional methods of electricity generation such as coal or fossil fuels. This study aimed to develop, evaluate, and compare the performance of Linear multiple regression (MLR), support vector regression (SVR), Bagging and random forest (R.F.), and decision tree (CART) models in predicting wind speed in Southeastern Iran. The data used in this research is related to the statistics of 10 minutes of wind speed in 10-meter, 30-meter, and 40-meter wind turbines, the standard deviation of wind speed, air temperature, humidity, and amount of the Sun's radiation. The bagging and random forest model with an RMSE error of 0.0086 perform better than others in this dataset, while the MLR model with an RMSE error of 0.0407 has the worst.

Keyword: Machine Learning, MLR, SVR, R.F., CART, Wind Speed Forecasting.

1. Introduction

Predicting wind speed is essential for harnessing wind energy; for this reason, researchers have always focused on developing wind speed forecasting models. Also, given the declining conventional energy resources in the coming years, it is necessary to expedite the discovery and exploitation of renewable energy in solving this energy problem (Ma, X., Jin, Y., & Dong, Q., 2017). In addition, wind speed is essential in generating wind power (Santamaría-Bonfil, G., Reyes-Ballesteros, A., & Gershenson, C. J. R. E., 2016). However, wind speeds and strengths are unstable and easily affected by atmospheric, climatic, and geographical factors, increasing wind speed fluctuations (Liu, D., Wang, J., & Wang, H., 2015). In general, forecast periods for wind speeds are divided into highly short-term, short-term, and long-term, each with its advantages. Wind speed forecasting methods are divided into four categories: 1) Stability method: In this approach, the future wind speed is equal to the wind speed at the forecast time (Zhao, X., Wang, S., & Li, T., 2011). The performance of the sustainability method decreases rapidly as the forecast time horizon increases, so this

1. Introduction

Predicting wind speed is essential for harnessing wind energy; for this reason, researchers have always focused on developing wind speed forecasting models. Also, given the declining conventional energy resources in the coming years, it is necessary to expedite the discovery and exploitation of renewable energy in solving this energy problem (Ma, X., Jin, Y., & Dong, Q., 2017). In addition, wind speed is essential in generating wind power (Santamaría-Bonfil, G., Reyes-Ballesteros, A., & Gershenson, C. J. R. E., 2016). However, wind speeds and strengths are unstable and easily affected by atmospheric, climatic, and geographical factors, increasing wind speed fluctuations (Liu, D., Wang, J., & Wang, H., 2015). In general, forecast periods for wind speeds are divided into highly short-term, short-term, and long-term, each with its advantages. Wind speed forecasting methods are divided into four categories: 1) Stability method: In this approach, the future wind speed is equal to the wind speed at the forecast time (Zhao, X., Wang, S., & Li, T., 2011). The performance of the sustainability method decreases rapidly as the forecast time horizon increases, so this model is only reliable for extraordinary short-term goals. 2) Physical methods: This approach requires temperature, pressure, and humidity information. To predict wind speed. One of these methods is NWP (Lorenc, A. C., 1986). Physical methods provide accurate estimates for long-term predictions. The main drawbacks of numerical weather forecasting models are the time and complexity of memory to produce results. Therefore, these methods are not reliable for short-term forecasting horizons. 3) Statistical methods find a mathematical relationship between wind's time series data. 4) Artificial intelligence methods: Includes artificial neural networks (Liu, H., Tian, H. Q., Pan, D. F., & Li, Y. F., 2013), Support vector regression (Chen, K., & Yu, J., 2014), and other machine learning methods such as decision trees and random forest, which have been used to predict wind speed in recent years.

Decision trees with decision rules are one of the machine learning methods. Unlike the artificial neural network model, the decision tree model produces law. The decision tree structure explains the anticipated obtained in a series of rules. Trees are decision-tool that can respond to complex and nonlinear issues. Since science, such as electricity and electronics that have nonlinear issues, are well-known for resolving issues, it is a review of this tool to engineering science and studies of natural resources, which is a component of nonlinear and complex phenomena, has also been opened. Reference (Heinermann, J., & Kramer, O., 2016) from the analysis of homogeneous regressions and its comparison with the decision tree and k nearest neighbors and support vector regressions has reached a basic model for predicting wind speed. The reference (Lahouar, A., & Slama, J. B. H., 2017) first found the influential factors for wind speed in the climate, implemented a random forest correlation coefficient on it, and improved its result. Reference (Heinermann, J., & Kramer, O., 2014, September) has brought the regression support vector algorithm to more accurate results using bagging. The present study aimed to evaluate the efficiency of linear regression models as the benchmark, decision tree, random forest, backup vector regression, and bag-making algorithm in predicting wind speed, which has been used to achieve research goals of Sistan pilot station statistics and data.

In Section II, we look at related data mining and machine learning in wind forecasting. In Section III, we have described the proposed machine learning models in their construction. Also, Section IV introduces the data set, the study area's location,

and the models' results and compares them. Section V is the overall conclusion of the work.

2. Related Works

Reference (He, Q., Wang, J., & Lu, H., 2018) provides a powerful three-module hybrid system for predicting wind speed. They used the Ensemble Empirical Mode Decomposition (EEMD) method in the preprocessing data model. They also de- fined Kernel-Based Fuzzy C-Means Clustering (KFCM) as a data clustering module, a ripple neural network to perform the final preview, and used two quarterly data sets from Shandong Province, China. Reference (Yu, C., Li, Y., Xiang, H., & Zhang, M., 2018) proposed a new combined method for predicting short-term wind speeds: wavelet packet analysis, spatial clustering, and the Elman neural network. Also, five series of wind speeds, the first three obtained from Sichuan Province, China, have been used as data sets in their work. Reference (Schyska, B. U., Couto, A., et al., 2017) presented a new approach to estimating wind power generation across Europe based on spatial and temporal clustering. They performed spatial and temporal clustering through the K-Means algorithm.

They also used two sets of COSMOEU analytics data provided by the German Meteorological Service and the MERRA data provided by the U.S. National Aeronautics and Space Administration. Reference (Ghofrani, M., Mulcare, J., et al., 2017, February) proposed a combined framework for predicting wind speed using clustering Game Theoretic Self-Organizing (GTSOM). A Bayesian neural network has also been used as a forecast module to predict wind speed. They used Lowa wind speed data sets in their work. Reference (Ghofrani, M., Mulcare, J., et al., 2017, February) presented a short-term wind speed forecasting method that includes a combination cluster Improved Particle Swarm Optimization- K-Means (IPSO K-Means) component and the wavelet neural network prediction module. The IPSO K-means algorithm is used to search for the best clustering results. They proposed a predictive strategy on two real wind data sets in China. Reference (Wu, W., & Peng, M., 2017) suggested improving training models and predictive accuracy based on data mining, including K-means clustering and forecasting module Bagging Neural Network (BNN) for short-term wind speed forecasting. They also used 10 minutes of data from the Supervisory Control and Data Acquisition (SCADA) as research data.

Reference (Dong, L., Wang, L., et al., 2016) provided a combined model for wind speed forecasts. This model consists of cluster analysis of samples and a set of Generalized Regression Neural networks (GRNN). Data were also used from January 2012 to February 2012 at Yilan Wind Farm in northeastern China for analysis, modeling, and forecasting, which included wind speed and NWP with a time of 15 minutes. Reference (Lydia, M., Kumar, S. S., et al., 2016) provided wind speed prediction models in 10- to 1-hour intervals based on linear and nonlinear autoregression (A.R.) moving models. The Autoregressive moving average (ARMA) model is based on wind direction and annual trend using data from Sotavento Galicia Plc. The second A.R. model is based on wind direction, wind cutting, and temperature based on data from the Wind Energy Technology Center, Chennai, India. Reference (Xu, Q., He, D., et al., 2015) offers a new approach to wind forecasting using insufficient weather data. For the first time, a wrong data analyst was introduced in this case to investigate the relationship between wind speed forecast error and some new features extracted from

raw NWP. Second, they proposed a hierarchical structure that includes a K-means cluster-based detection module and a neural network-based predictor module. In the neural network module, the wind speed prediction is completely adjusted based on the analyzer output of insufficient information. In this work, three types of data sets are used. Reference (Gupta, D., Natarajan, N., & Berlin, M., 2022) worked on hybrid machine learning models deployed for short-term wind speed prediction. The twin support vector regression (TSVR), primal least squares twin support vector regression (PLSTSVR), iterative Lagrangian twin parametric insensitive support vector regression (ILTPISVR), extreme learning machine (ELM), random vector functional link (RVFL), and large-margin distribution machine-based regression (LDMR) models have been adopted in predicting the short-term wind speed collected from five stations named as Chennai, Coimbatore, Madurai, Salem, and Tirunelveli in Tamil Nadu, India.

3. Proposed Wind Speed Forecasting Models

This section explains multiple linear regression, decision tree regression, bagging and random forest tree, support vector regression, and their architecture. We also look at formulas and their relationships and see their architecture in forms.

A. Multiple Linear Regression (MLR)

In statistics, linear regression is a linear approach to modeling the relationship between a scalar response (or dependent variable) and one or more explanatory variables (or independent variables). The case of one explanatory variable is called simple linear regression. For more than one explanatory variable, the process is called multiple linear regression (Freedman, D. A., 2009). Here, we develop MLR to predict the hourly wind speed. The following equation represents the regression model for prediction:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \cdots + \varepsilon \quad (1)$$

where y is the dependent variable, $x_i (i = 1, 2, 3, \ldots, k)$ independent explanatory variables, regression coefficients, and ε is the residual error. This study considers different independent variables: direction, temperature, humidity, past wind speed, and pressure. The wind speed is considered a dependent variable.

B. Decision Tree (CART)

A decision tree called Classification and Regression Tree (CART) is a statistical model (Wu, X., & Kumar, V. (Eds.)., 2009). It depicts the different classes or values an output may take regarding input features. Generally speaking, a tree is a set of nodes and branches organized in a hierarchy with no loops. A decision tree is a tree whose nodes store a test function to be applied to incoming data. Terminal nodes are called the tree leaves, and each leaf stores the final test result. The tree is binary if each node has two outgoing branches, the right child and the left child. The decision tree is robust, immune to irrelevant inputs, and provides good interpretability.

The remainder of this section is restricted to regression problems since the prediction is a kind of regression. Let X be an input vector containing m features, Y an output scalar, and S n a training set containing n observations (Xi, Yi).

$$S_n = \{(X_1, Y_1), \ldots (X_n, Y_n)\}, X \in \mathbb{R}^m \; Y \in \mathbb{R} \quad (2)$$

During training, an algorithm drives the inputs split at each node so that the

parameters of split functions become optimized to fit with the S n set. The principle consists of recursively splitting the input space X by searching for optimal sub-partitions. More precisely, the first step of the CART algorithm has to split, at best, the root into two different children according to:

$$\{X^j < d\} \cup \{X^j > d\} \quad (3)$$

To select the best split, the couple (j, d) should minimize a cost function, which is generally the child node variance. The variance of a node is p defined by:

$$var(p) = \sum_{i:X_i \in p} (Y_i - \overline{Y_P})^2 \quad (4)$$

Where $\overline{Y_P}$ is the mean of the scalars $Y_i$present in the node p? Then, children nodes are also divided in the same way. A termination criterion stops the development of the tree. It is common to stop the tree when a maximum number of levels is reached or when a node contains less than a predefined number of observations. A prediction function is constructed at the end of this training process S_n. The testing process determines an estimation $\hat{Y}$ of the output, Y corresponding to any new input vector X.

$$\hat{Y} = \hat{h}(X, S_n) \quad (5)$$

Each node applies its split function to the new input X. According to the result of the binary test, data are sent to the right or left child starting from the root. This process continues until the data reach a leaf (terminal node).

C. Bagging and Random Forest (R.F.)

The random forest is an ensemble method that combines the prediction of several decision trees (Genuer, R., 2010). The basic principle is called bagging (bootstrap aggregation), where a sample of size n taken from the training set S_n is selected randomly and fitted to a regression tree. This sample is called bootstrap and is chosen by replacement, meaning that the same observation (X_i, Y_i) may appear several times. A bootstrap sample is obtained by randomly selecting n observation with replacement from S_n, where each observation has the probability of 1/n to be selected. The independent identically distributed random variables Θ_l represent this random selection. The bagging algorithm selects several bootstrap samples $\left(S_n^{\Theta_1}, \dots, S_n^{\Theta_q}\right)$, applies the previous CART algorithm to these samples to construct a collection of q prediction trees $\left(\hat{h}(X, S_n^{\Theta_1}), \dots, \hat{h}\left(X, S_n^{\Theta_q}\right)\right)$, and then aggregates the output of all these trees.

In addition to bagging, the random forest also selects a predefined number of mtry among the m features for the split in each node. The R.F. algorithm tries to find the best split among only the mtry selected features. The selection at each node is uniform, and each feature has the probability of 1/m to be selected. The number mtry is the same for all prediction trees, and it is recommended to be the square root or the one-third of the features' number m:

$$mtry = \lceil \sqrt{m} \rceil \quad (6)$$

or

$$mtry = \left\lceil \frac{m}{3} \right\rceil \quad (7)$$

The aggregation is performed by averaging the outputs of all trees. Where ⌈x⌉

denotes the ceiling function of x, the remainder of the algorithm is similar to CART, the best-split couple (j, d) is obtained by minimizing a cost function, and the procedure continues until the full development of all trees. Consequently, the estimation $\hat{Y}$ Of the output matching a new input vector, X is as follows:

$$\hat{Y} = \frac{1}{q} \sum_{l=1}^{q} \hat{h}(X, S_n^{\Theta_l}) \tag{8}$$

The main advantage of bootstrap aggregation is immunity to noise since it generates non-correlated trees through different training samples. A weak predictor (a standalone regression tree) may be sensitive to noise, while the average of several decor-related decision trees is not. Selecting a random subset mtry of features has the same aim of decorating trees.

Two principal characteristics distinguish the random forest: the out-of-bag error OOBE and the measure of variable importance VI. The OOBE helps estimate the generalization capacity of the model. The OOBE, also called generalization error, is a kind of built-in cross-validation. It is the average prediction error of first-seen observations, i.e., using only the trees that did not see these observations while training.

$$OOBE = \frac{1}{n} \sum_{i=1}^{n} (X_i, Y_i)^2 \tag{9}$$

The variable importance measure is obtained by permuting a feature and averaging the difference in OOBE before and after permutation over all trees. Let's define each bootstrap sample $S_n^{\Theta_l}$ It's associated OBB_l , i.e., the set of observations not included in $S_n^{\Theta_l}$. For a fixed j among the m features, the values of the j^{th} variable are permuted randomly over OBB_l to get a disturbed sample called $\overline{OBB_l}$. The new $\overline{OOBB_l}$ of the disturbed sample is then calculated. These operations are repeated for every bootstrap sample. The importance of the j^{th} variable, called $VI(X^j)$, is defined by the difference between average errors of original OBB_l and disturbed $\overline{OBB_l}$ (Genuer, R., 2010).

$$VI(X^j) = \frac{1}{q} \sum_{l=1}^{q} (\overline{OOBB_l} - OBBE_l) \tag{10}$$

This variable is relevant if permutations over the j^{th} variable lead to increasing error. The more the score $VI(X^j)$ increases, the more the j^{th} variable becomes important (Chen, K. Y., & Wang, C. H., 2007). The number of trees q be denoted ntree for the remainder of the paper. The quantile regression forests not be detailed here for concision purposes. Random Forest Regression architecture is shown in Fig. 1.

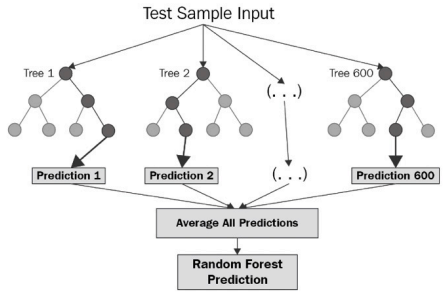

Fig. 1 The architecture Random Forest Regression model

D. Support Vector Regression (SVR)

The support vector machine (SVM) developed by V. Vapnik is gaining popularity due to its attractive features and promising empirical performance. The Structural Risk Minimization (SRM) principle is implemented rather than the Empirical Risk Minimization (ERM) principle employed by most conventional neural network models. SRM minimizes an upper bound on the expected risk, as opposed to ERM, which minimizes the error on the training data. It is this difference that offers SVM a remarkable ability to generalize.

SVM was initially used for classification purposes, but its principles can easily be extended to regression and time series prediction problems. Therefore, the SVM refers to classification and regression methods, which could be named the Support Vector Classification (SVC) and Support Vector Regression (SVR).

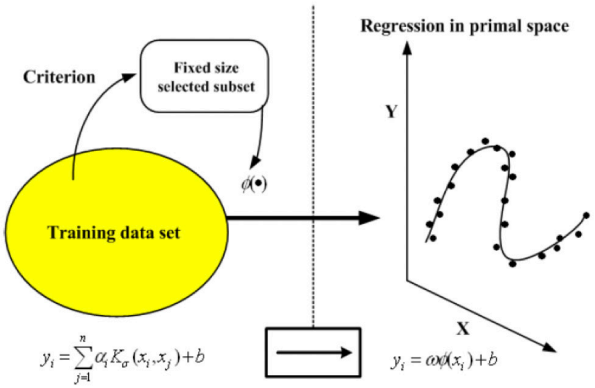

Fig. 2 Mapping input space x into high-dimensional feature space (from (Chen, K. Y., & Wang, C. H., 2007))

The basic concept of the SVR is to map the input data into a high-dimensional feature space by nonlinear mapping, to yield and solve a linear regression problem in this feature space (see Fig.1). Hence, given a set of data set $T = \{(x_i, d_i)\}_{i=1}^{n}$ where x_i is the input vector, d_i denotes the target value, and n is the total number of data patterns). The objective is to get a regression function, $y = f(x)$, which could accurately predict the output corresponding to a new set of input-output examples. The linear regression in the feature space is approximated using the following function:

$$f(x) = \omega\emptyset(x) + b$$
$$\emptyset: R^n \rightarrow F, \omega \in F \qquad (11)$$

where ω and b are coefficients; $\emptyset(X)$ denotes the high dimensional feature space, nonlinearly mapped from the input space x. Therefore, the linear regression in the high-dimensional feature space responds to nonlinear regression in the low-dimensional input space, disregarding the inner product computation between ω and $\emptyset(X)$ in the high-dimensional feature space. This event leads to the original optimization problem involving nonlinear regression, which is transformed into finding the flattest function in the feature space F, not in the input space. The unknown coefficients ω and b in (11)

The basic concept of the SVR is to map the input data into a high-dimensional feature space by nonlinear mapping, to yield and solve a linear regression problem in this feature space (see Fig.1). Hence, given a set of data set $T = \{(x_i, d_i)\}_{i=1}^{n}$ where x_i is the input vector, d_i denotes the target value, and n is the total number of data patterns). The objective is to get a regression function, $y = f(x)$, which could accurately predict the output corresponding to a new set of input-output examples. The linear regression in the feature space is approximated using the following function:

$$f(x) = \omega\emptyset(x) + b$$
$$\emptyset: R^n \rightarrow F, \omega \in F \quad (11)$$

where ω and b are coefficients; $\emptyset(X)$ denotes the high dimensional feature space, nonlinearly mapped from the input space x. Therefore, the linear regression in the high-dimensional feature space responds to nonlinear regression in the low-dimensional input space, disregarding the inner product computation between ω and $\emptyset(X)$ in the high-dimensional feature space. This event leads to the original optimization problem involving nonlinear regression, which is transformed into finding the flattest function in the feature space F, not in the input space. The unknown coefficients ω and b in (11) are estimated by minimizing the following regularized risk function:

$$r(c) = c\frac{1}{n}\sum_{i=1}^{n}\gamma_\varepsilon(\alpha_i, f_i) + \frac{1}{2}\|\omega\|^2 \quad (12)$$

Where

$$\gamma_\varepsilon(a, f) = \begin{cases} 0, & if\ |a-f| \le \varepsilon \\ |a-f| - \varepsilon, & otherwise \end{cases}$$

and C denotes a cost function measuring the empirical risk, and $\gamma_\varepsilon(a, f)$ in (12) is called the ε -insensitive loss function, and the second term, $\frac{1}{2}\|\omega\|^2$, measures the flatness of the function. Therefore, C is considered to specify the trade-off between the empirical risk and the model flatness. Both C and ε are the parameters selected by users. Two positive slack variables, ξ, and ξ*, representing the distance from actual values to the corresponding boundary values of ε -tube, are introduced. Then (12) is transformed into the following soft margin problem:

$$minimize\ \ \frac{1}{2}\|\omega\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \quad (13)$$
$$subject\ to\ \ d_i - \omega.\phi(x_i) - b \le \varepsilon + \xi_i$$
$$\omega.\phi(x_i) + b - d_i \le \varepsilon + \xi_i^*$$
$$\xi_i, \xi_i^* \ge 0, i = 1, \dots, n, \quad (14)$$

Adding Lagrangian multipliers α and α^*can optimize that problem as a dual problem. Also, SVR can estimate a nonlinear function by employing a Kernel Function, $k(x_i, x_j)$ The regression function estimated by SVR can be written as the following kernel expansion:

$$g(x) = \sum_{i=1}^{ns}(\alpha - \alpha^*)k(x_i, x_j) + b \quad (15)$$

Where ns is the number of support vectors, the kernel function depends on a particular parameter γ.

4. Simulation and Results

In this section, the data sets are presented, and the outputs of the machine learning

models and the benchmark outputs are analyzed and compared.

A. Dataset, Study Area and Simulation's Parameter

This study is focused on the Lutak region from the city of Zabol, one of the cities of Sistan and Baluchestan province in Iran. This region has the most wind blowing in Iran. In Fig. 3, the location of this region within Iran is marked. This data set includes annual wind speeds at 10-minute intervals from 2006-2010. At this station, the power of 660 kW / h and 660 volts is injected into the 20-kV network by trans-terrestrial power. Fig. 4 shows some wind speed values in the data set.

In this process, the data are divided into two categories: training data and testing data; in this study, 80% of the total data belongs to the training, and the remaining 20% as test data for the model has been introduced. There are 40000 wind speed values measured in 10-min intervals; therefore, sufficient data is available for training and testing the proposed approach. We have used different weather features in implementing the models. The 14 independent variables in table I are entered as input to the models to predict the target variable shown in Table II. Also shown in Fig. 5 is a chronology chart or correlation between different variables used as a statistical method.

Fig. 3 Map of Iran, including the case study area

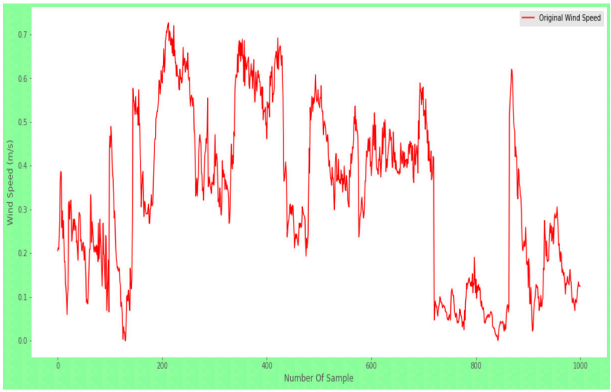

Fig. 4 Original Wind Speed

Table 1: input variables

Input Variable	High	Low	Mean
Temperature	45.60	-9.30	22.94
Rel Humidity	100	4	27.58
Global Radiation	1141	0	265.87
Wind Speed Max 10m	29	0	7.82
Wind Speed Min 10m	12.30	0	3.14
Wind Speed Avg 10m	18	0	5.25
Wind Speed SDev 10m	6.12	0	0.78
Wind Speed Max 30m	31.20	0	9.28
Wind Speed Min 30m	17	0	4.97
Wind Speed Avg 30m	23.40	0	7.08
Wind Speed SDev 30m	7.87	0	0.80
Wind Speed Min 40m	18.70	0	5.51
Wind Speed Avg 40m	24.10	0	7.61
Wind Speed SDev 40m	7.90	0	0.97

Table 2: target variable

Output Variable	High	Low	Mean
Wind Speed Max 40m	32.10	0	9.75

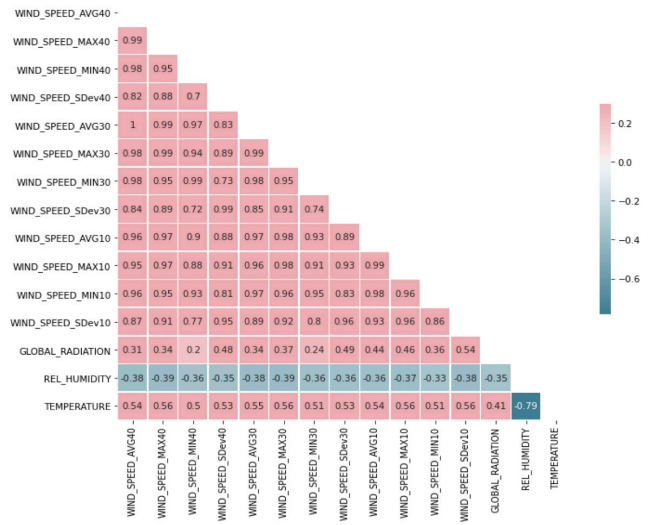

Fig. 5 Correlation of Variable

B. Evaluation Criteria

The root-mean-square error (RMSE) and mean absolute error (MAE) are employed as two evaluation metrics:

$$RMSE = \sqrt{\frac{1}{M}\sum_{n=1}^{M} e(n)^2} \tag{16}$$

And the MAE is expressed as:

$$MAE = \frac{1}{M}\sum_{n=1}^{M} |e(n)| \tag{17}$$

Here, e (n) = t (n) - y (n), and M is the number of samples in the testing set. t(n) and y(n) are the desired output and the actual output of the models for the nth sample, respectively.

C. Numerical Result and Comparison for Prediction

In this paper, 4 predictive models, including multivariate linear regression as benchmarks, support vector regression, decision tree regression, and random forest as machine learning algorithms used in regression tasks, are implemented and compared to predict short wind speeds. We have used the linear regression model as a benchmark because regression focuses on predicting the output variable. However, it is doubtful that the existence of a relationship between independent and dependent variables in the regression model indicates the causality of this relationship. Therefore, this generalization of the regression model results is weaker than other new methods; other statistical and research methods must be used to analyze causality. Table III shows the evaluation criteria for the models, including RMSE and MAE. As can be seen in the table, the highest RMSE error is related to multivariate linear regression with 0.0407, so this model does not perform well on this data set compared to other algorithms. However, the SVR model with RMSE 0.0241 outperformed multivariate linear regression. Among the tree models, the R.F. model with RMSE 0.0086 performs better than the regression decision tree. It can be said that the Bagging methodology impacts R.F. efficiency and has the lowest RMSE on the southeastern wind speed data collection in Iran compared to other models. Also, considering that the lowest error rate is related to R.F. and CART, it is possible to achieve better answers by optimizing the parameters of these two models.

Although SVR has better results than R.F. in some cases, it depends on the area's topography and the data's quality and quantity. Therefore, it can be said that the R.F. model has a more suitable performance than other models for Lutak's wind power station located in southeastern Iran. Fig. 6 and 7 show the output for actual data and

Table 3: RMSE and MAE of short-term forecasting methods

Method	RMSE	MAE
MLR	0.0407	0.0307
SVR	0.0241	0.0164
CART	0.0128	0.0089
RF	0.0086	0.0060

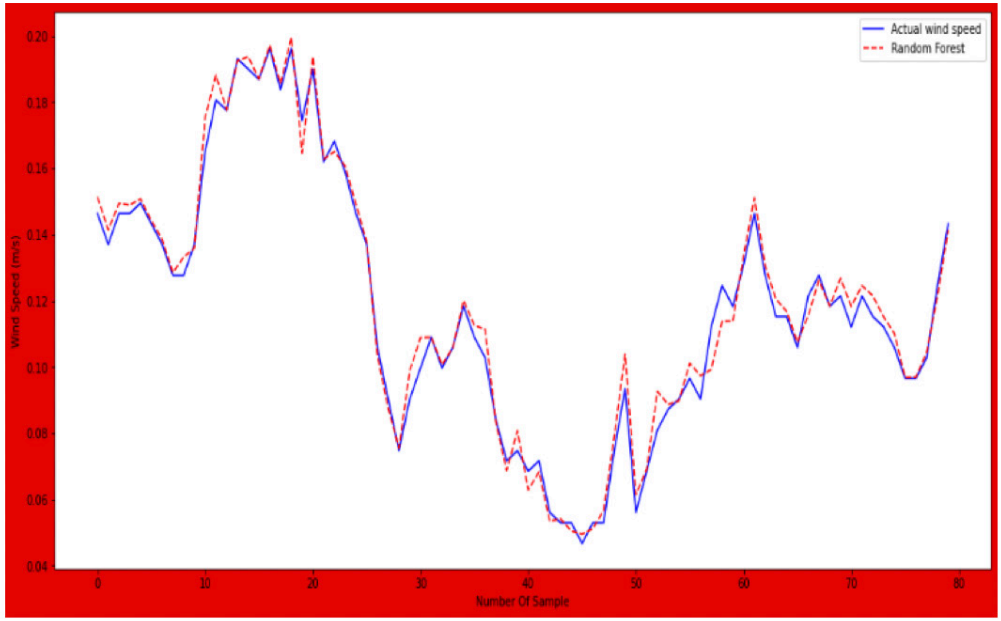

Fig. 6 R.F. Output: 10 min forecasting

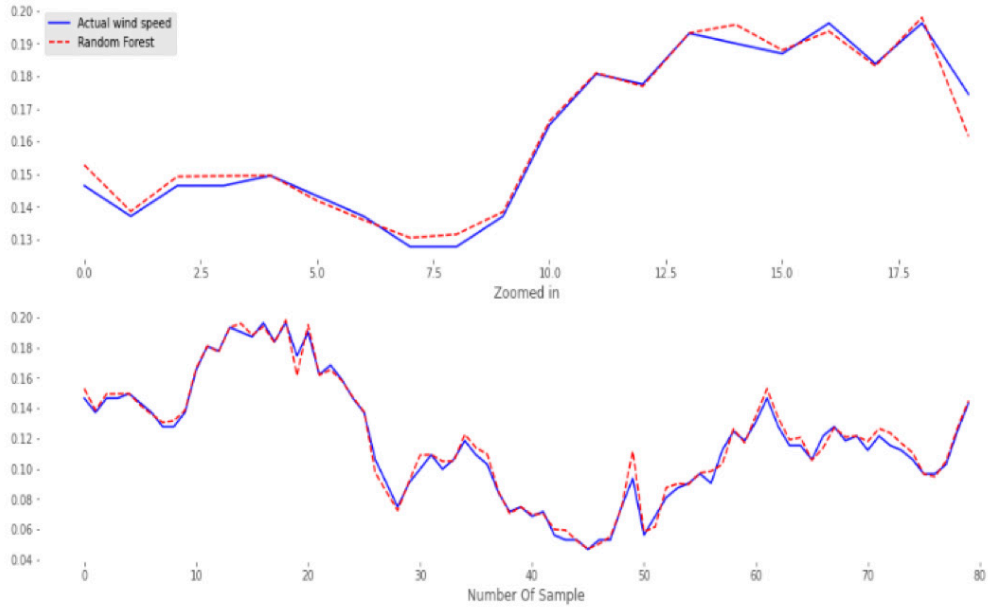

Fig. 7 Zoomed in 20 samples R.F. Output

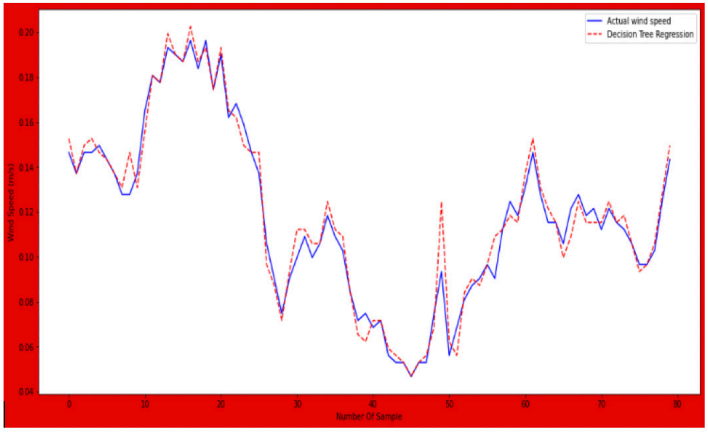

Fig. 8 CART Output: 10 min forecasting

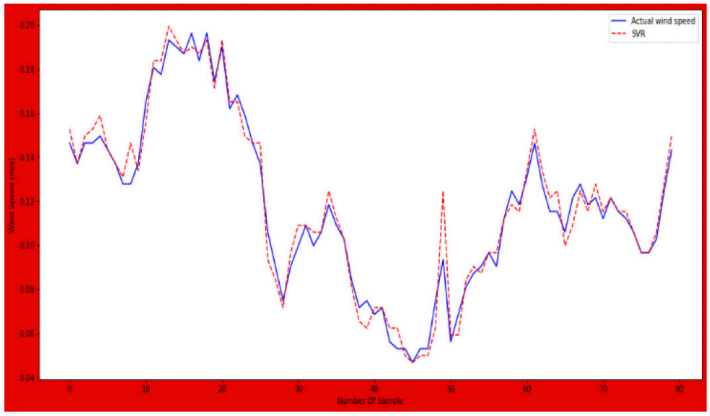

Fig. 9 SVR Output: 10 min forecasting

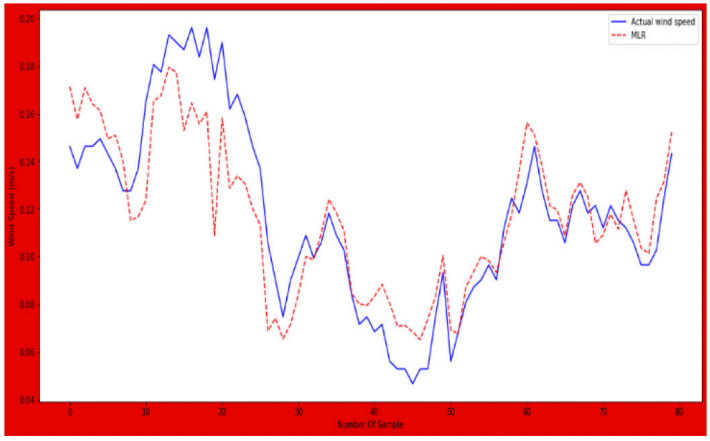

Fig. 10 MLR Output: 10 min forecasting

R.F. predicted data. Also, the output of the actual and predicted CART, SVR, and MLR data is shown in Fig. 8, 9, and 10, respectively.

5. Conclusions

Accurate wind speed and power forecasting are vital in wind farm performance and risk management. In addition, ten-minute data on wind turbines are easily affected by atmospheric fluctuations. Therefore, to assess the issue's importance, the recorded statistics were used every ten minutes for wind speed, air temperature, humidity, and solar radiation in the statistical period (2010-2006). Based on the results, it can be said that the random forest model in the Sistan pilot air power station is efficient in predicting wind speed using processed data (data quality cannot be ignored in the process). However, due to the better and closer performance of the regression decision tree to the Random forest relative to the linear regression and support regression vector, we suggest optimizing the R.F. and CART parameters.

References

Chen, K., & Yu, J. (2014). Short-term wind speed prediction using an unscented Kalman filter based state-space support vector regression approach. *Applied Energy, 113*, 690-705.

Chen, K. Y., & *Wang, C. H. (2007). Support* vector regression with genetic algorithms in forecasting tourism demand. *Tourism management, 28*(1), 215-226.

Dong, L., Wang, L., et al. (2016). Wind power day-ahead predict*ion with cluster an*alysis of NWP. *Renewable and Sustainable Energy Reviews, 60,* 1206-1212.

Freedman, D. A. (2009). Statistical models: theory and practice. cambridge un*iversity press.*

Genuer, R. (2010). Forêts aléatoires: aspects théoriques, sélection de variables et applications *(Doctoral dissertation, Université Paris Sud-Paris XI).*

Ghofrani, M., Mulcare, J., et al. (2017, February). A modified game theoretic self-organizing map for wind speed forecasting. *In 2017 IEEE Power and Energy Conference at Illinois (PECI) (pp. 1-5)*. IEEE.

Gupta, D., Natarajan, N., & Berlin, M. (2022). Short-term wind speed prediction using hybrid machine learning techniques. *Environmental Science and Pollution Research, 29*(34), 50909-50927.

He, Q., Wang, J., & Lu, H. (2018). A hybrid system for short-term wind speed forecasting. *Applied energy, 226, 756-771.*

*H*einermann, J., & Kramer, O. (2014, September). Precise wind power prediction with SVM ensemble regres*sion. In International* conference on artificial neural networks (pp. 797-804). Springer, Cham.

Heinermann, J., & Kramer, O. (*2016). Machine learning* ensembles for wind power prediction. Renewable Energy, 89, 671-679.

Lahouar, A., & Slama, J. B. H. (2017). Hour-ahead wind power forecast based on random forests. *Renewable en*ergy, 109, 529-541.

Liu, D., Wang, J., & Wang, H. (2015). Short-term wind speed forecasting based on

spectral clustering and optimised echo state networks. *Renewable Energy, 78,* 599-608.

Liu, H., Tian, H. Q., Pan, D. F., & Li, Y. F. (2013). Forecasting models for wind speed using wavelet, wavelet packet, time series and Artificial Neural Networks. *Applied Energy, 107, 19*1-208.

Lorenc, A. C. (1986). Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society, 112*(474), 1177-1194.

Lydia, M., Kumar, S. S., et al. (2016). Linear and non-linear autoregressive models for short-term wind speed forecasting. *Energy conversion and management, 112,* 115-124.

Ma, *X., Jin, Y., & Dong, Q. (2017).* A generalized dynamic fuzzy neural network based on singular spectrum analysis optimized by brain storm optimization for short-term wind speed forecasting. Applied Soft *Computing, 54,* 296-312.

Santamaría-Bonfil, G., Reyes-Ballesteros, A., & Gershenson, C. J. R. E. (2016). Wind speed forecasting for wind farms: A method based on support vector regression. *Renewable Energy, 85, 790*-809.

Schyska, B. U., Couto, A., et al. (2017). Weather dependent estimation of continent-wide wind power generation based on spatio-temporal clustering. *Adva*nces in science and research, 14, 131-138.

Sun, G., Jiang, C., et al. (2018). Short-term wind power forecasts by a synthetical similar time series data mining method. *Renewa*ble energy, 115, 575-584.

Wu, W., & Peng, M. (2017). A data mining approach combining $ K $-Means clustering with bagging neural network for short-term wind power forecasting. *IEEE Internet of Thin*gs Journal, 4(4), 979-986.

Wu, X., & Kumar, V. (Eds.). (2009). The top ten algorithms in data mining. CRC press.

Xu, Q., He, D., et al. (2015). A short-term wind power forecasting approach with adjustment of numerical weather prediction input by data mining. *IEEE Transactions on sustainable energy, 6*(4), 1283-1291.

Yu, C., Li, Y., Xiang, H., & Zhang, M. (2018). Data mining-assisted short-term wind speed forecasting by wavelet packet decomposition and Elman neural network. *Journal of Wind Engineering and Industrial Aerodynamics, 175*, 136-143.

Zhao, X., Wang, S., & Li, T. (2011). Review of evaluation criteria and main methods of wind power forecas*ting. Energy Procedia, 12,* 761-769.

Submitted: 29.12.2021
Accepted: 17.05.2022