A Comparative Study of Short-Term Wind Speed Forecasting Models

Archita Vijayvargia[1], Kailash Chand Sharma[2] and Rohit Bhakar[1]
[1]Centre for Energy and Environment, Malaviya National Institute of Technology, Jaipur, India
[2]School of Automation, Banasthali Vidyapith, Rajasthan, India
2017PCV5168@mnit.ac.in, kailashsharma3889@gmail.com, rbhakar.ee@mnit.ac.in

***Abstract*—The increasing trend to integrate intermittent wind generation in restructured power systems introduces several operational issues including reserve management and market performance. Therefore, Very Short-Term Forecasting (VSTF) of wind speeds is highly emphasized to address these issues by accurate prediction of Wind Farms' (WFs) power outputs. State-of-the-art of VSTF typically provides a trade-off between machine learning and time series techniques. This paper provides an exhaustive study of statistical and machine learning VSTF models *i.e.* Linear Regression (LR), Autoregressive Integrated Moving Average (ARIMA) k-Nearest Neighbors (k-NN) and Artificial Neural Network (ANN). The performance of these models is evaluated by Root Mean Square Error (RMSE) and Mean Absolute Error (MAE). The study is carried out on the dataset of WF located at Jaisalmer, Rajasthan, India. The results obtained concludes that ANN outperforms for VSTF of wind speed because of its ability to form complex non-linear systems for forecasting based on simple wind speed data. The ANN forecast accuracy is followed by LR, ARIMA and k-NN models.**

***Index Terms*—ARIMA, Artificial Neural Network, Forecasting, k-Nearest Neighbors, Linear regression, Machine learning, Time series, Wind speed forecasting.**

I. INTRODUCTION

The world is continuously inclining towards the grid integration of Renewable Energy (RE) sources to attain sustainability. In this regard, government of India has set the very ambitious target to install 175GW of RE generation capacity by 2022. Wind power is one of the most appealing RE sources due to its mature technology, high efficiency, and modern infrastructure [1-3]. However, due to the stochastic and intermittent nature of wind power, its large scale grid integration introduces several challenges in the power system operations and planning. Major challenges include power and voltage imbalance, scheduling and dispatch abilities, harmonics, *etc*. [2, 4]. All these issues create a gap between demand and supply on the power grid.

Maintaining the demand-supply balance is of utmost necessity for efficient power system operation [5]. As the power output by a WF depends mainly on the variation of wind, so any unanticipated change in the WF output may enhance the operating cost, reserve requirement and other market operation issues [4-7]. Wind Speed Forecasting (WSF) is considered as one of the possible solutions to mitigate these issues related to better scheduling of wind energy [8].

Present power system works on day-ahead, intraday, 30 minutes ahead, 15 minutes ahead timelines. So, time horizon classifies WSF into VSTF, Short-term Forecasting (STF), Medium-term Forecasting (MTF), and Long-term Forecasting (LTF). The time-scale of VSTF varies from few minutes to 6 hours; for STF the time horizon is up to a day; for MTF, the time horizon ranges from a day to a week; and LTF time scale can extend to more than a week [2]. Out of these, the time scale that suits the context of grid operational issues is VSTF [9]. Moreover, adequate and versatile forecast methodologies are available for each of these prediction horizons.

Three main classes of WSF techniques have been identified from the literature, namely, physical methods, statistical methods, and machine learning methods [10-14]. Physical or Numerical Weather Prediction (NWP) approach uses detailed locale meteorological conditions of the WF. Since these methods take a bit longer computation time, therefore are not suitable for STF and VSTF scenarios [15]. Another forecasting strategy utilizes statistical approach to overcome these drawbacks [10].

Statistical methods provide relatively inexpensive forecasting methods that require only wind resource data. These statistical approaches are based on Time Series such as white noise, random walk, Autoregressive (AR), Moving Average (MA), Autoregressive Moving Average (ARMA), and ARIMA models [9]. Time series ARIMA model is a bit classical method and is discussed in several literatures for different applications [12-13]. ARIMA is a better method than the persistence or naive forecasting method that forecasts with the concept of 'the upcoming interval will be similar to existing conditions'. Hence, the later models do not guarantee for accurate forecasting of power produced by wind generators [4]. ARIMA model is suitable for VSTF and STF prediction scenarios. It can capture the temporal correlations of wind speed data with a lower number of parameters and thus the complexity in computation is less. But as the forecasting horizon increases, the prediction error also increases [3, 14].

Randomness and variations in wind speeds are analyzed in a better way by using the Machine Learning approaches such as ANN, random forest, k-NN, LR, decision tree, support vector machines [16]. These are based on Artificial Intelligence (AI), deals in a better way with non-linear and complex problems of forecasting. Although some of these AI methods may face the data acquisition issues and are sensitive to errors concerning learning rates [17], yet these models can fit the WSF problem in a better manner as compared to time series models [17-18].

This paper presents a comprehensive study comprising of traditional time series method and machine learning methods for univariate onsite WSF, *i.e.,* forecasting is done only for one WF for which the dataset is available. A detailed conceptualization of ARIMA based time series forecasting along with some machine learning forecast models such as LR, k-NN, and ANN models is presented. These modes are implemented on the wind speed dataset of Jaisalmer, Rajasthan WF available for the month of September 2018. The comparison of these forecast models illustrates that the ANN model forecasts the wind speed with the best precision and quality out of the chosen models.

The rest of the paper is organized as follows. Section II gives detail of the statistical forecasting models with special emphasize on ARIMA model. Section III describes in brief the concept of regression-based machine learning with the details of LR model, k-NN, and ANN models. Section IV shows the case study of WSF for the Jaisalmer WF. Finally, the conclusions are drawn in the section V.

II. Statistical Forecasting Models

Time series models are major statistical models that can be used to predict the wind resource in power system. The primary purpose of time series modelling is data acquisition and to analyse the historical events to propose a model with suitable coefficients. This model can potentially explain the deep-rooted structure of the series and can be used for forecasting the upcoming elements of time series [18]. A time series can be stationary and non-stationary, depending on the mean and variance of the series.

A stochastic time series process, *i.e.,* wind speeds, is generally a non-stationary process. Its mean and variance changes with time. Further, the covariance value depends only on the number of lags between the two values. It has no relation with the exact time gap between the concerned time series elements [14]. By taking an adequate number of differences or logarithmic transformations, these stochastic wind speeds become stationary. Differencing is one way of creating a time series stationary by computing the differences between the two consecutive observations. To make a constant variance of the time series logarithmic transformations are done [14]. The equation for first order differencing and logarithmic transformations are:

$$y_t^{'} = y_t - y_{t-1} \tag{1}$$

$$y_t^{ln} = ln(y_t) \tag{2}$$

Where, y_t is the predicted output of time series, y_{t-1} is the first lag value of the variable of interest.

Based on the dependency of forecast on historical values and forecast errors, time series models are grouped into white noise, random walk model, autoregressive (AR) model, moving average (MA) model, autoregressive moving average (ARMA) model, and autoregressive integrated moving average model [18]. This paper discusses ARIMA model in detail.

A. ARIMA model

Time series ARIMA model uses historical values and past forecast errors of the variable for predicting the future values. AR terms of the time series model uses the concept of forecasting the wind speed using a linear comination of the historical value of the variable up to p lags. MA terms of the time series model uses the forecast errors in a regression-like model up to q lags. ARMA model understands the significance of using both AR and MA terms for accurate forecasting. But the time series for the wind speed is not stationary. For this purpose, the ARIMA model comes into forecasting scenario [13, 18].

The time series equation goes through some mathematical transformations, like differencing, to analyze any non-stationary time series by making it stationary. Thus, the degree of differencing d is included along with the AR and MA lags to conclude the order of ARIMA model as (p,d,q) [13-14]. The final equation of ARIMA (p,d,q) is given by (3):

$$y_t^d = c + e_t + \sum_{k=1}^{p} \phi_k y_{t-k}^d + \sum_{j=1}^{q} \theta_j e_{t-j} \tag{3}$$

Where, c is the constant drift or the average of the changes between the consecutive observations, and e_t is the white noise or error of time series forecasting, j and k are the indices of MA and AR time lags, $\phi_k, k = 1 \ldots p$ are the coefficients of historical values of time series, $\theta_j, j = 1 \ldots q$ are the coefficients of forecast errors of time series. By varying the coefficients $\phi_1 \ldots \phi_p$ and $\theta_1 \ldots \theta_q$, multiple time series are obtained. Also, the error term e_t effects only the scale of the time series. The pattern of the series remains unaltered with any variation in e_t.

The ARIMA (p,d,q) equation can also be written as:

$$\begin{aligned} &\left(1-\phi_1 B-\phi_2 B^2-\ldots-\phi_p B^p\right)\left(1-B\right)^d y_t \\ &= c+\left(1+\theta_1 B+\theta_2 B^2+\ldots+\theta_q B^q\right)e_t \end{aligned} \tag{4}$$

Where, B is called the Backshift operator such that $B^k(y_t) = y_{t-k}$.

B. ARIMA based WSF Methodology

The ARIMA model based wind speed forecasting methodology is shown in Fig. 1.

1. **Data Refining:** The temporally correlated wind speed data for the given location is transformed into time series format for performing ARIMA model simulations. The obtained wind speeds are then plotted and analyzed for any non-stationarity, *i.e.*, seasonality and trends, *etc*.
2. **Mathematical Transformation:** To make the data stationary, logarithmic or differencing transformations

are done to stabilize the variance as in (1) and (2).

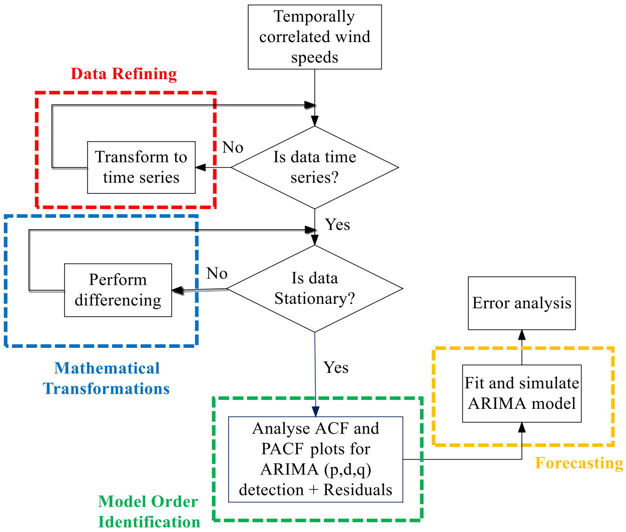

Fig. 1. ARIMA model based WSF methodology.

3. **Model Order Identification:** Next, the Auto-Correlation Function (ACF) and Partial Auto-Correlation Function (PACF) of the stationary data are observed to identify the model order (p,d,q).

   The mathematical formulations for ACF and PACF are provided in (5) and (6) respectively. If y_t is the original series and $\bar{y}$ is the mean of the data, then the autocorrelations of order $k = 0,1,2,\ldots,n$, where n is total number of lags possible, is computed as:

$$ACF = \frac{\frac{1}{(n-k)}\sum_{t=k+1}^{n}(y_t-\bar{y})(y_{t-k}-\bar{y})}{\sqrt{\frac{1}{n}\left(\sum_{t=1}^{n}(y_t-\bar{y})\right)\frac{1}{(n-k)}\sum_{t=k+1}^{n}(y_{t-k}-\bar{y})}} \quad (5)$$

$$PACF = \frac{\operatorname{cov}\left(y_t, y_{t-k} \mid y_{t-1},\ldots,y_{t-k+1}\right)}{\sqrt{\operatorname{var}\left(y_t \mid y_{t-1},\ldots,y_{t-k+1}\right).\operatorname{var}\left(y_{t-k} \mid y_{t-1},\ldots,y_{t-k+1}\right)}} \quad (6)$$

   The model order is selected for which the Akaike Information Criteria (AIC) value is least or minimum. The formula for AIC determination is stated in (7).

$$AIC = 2P - \ln(L) \quad (7)$$

   Where P is the number of parameters estimated for the model and L is the maximum value of the likelihood function for the model. The model order chosen is correct if the ACF and PACF plots of the residuals is similar to that of the white noise pattern.
4. **Forecasting:** Next, the ARIMA model is fitted and simulated for the model order (p,d,q) selected and then the forecasts are obtained for the ahead values.
5. **Error Analysis:** Finally, the error analysis using RMSE and MAE is done. The mathematical formulations for RMSE and MAE are provided in (8) and (9).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - y_t)^2} \quad (8)$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - y_t| \quad (9)$$

Where, y_i = predicted or observed value of variable, and y_t = actual value of variable.

III. Machine Learning Forecasting Models

Machine learning is an implementation of AI that provides the system to learn and improve automatically from experience without being programmed explicitly. Machine learning focuses on the design and development of computer-based algorithms that can access the data, use it, and learn on itself. The primary motto is to grant the computers to acquire the knowledge automatically without human assistance or intervention and tune actions accordingly [17]. The learning process begins with observing the historical data provided to look for trends and patterns in it, thereby making better decisions in the future. Machine learning is further grouped into unsupervised and supervised learning. The below subsections provide a detailed understanding of some machine learning algorithms, *i.e.*, LR, k-NN and ANN forecasting models.

A. Linear Regression Model

LR is a linear forecasting technique to map the correlations among independent variable(s) X and dependent variable Y. The number of independent variable may vary from one to many. LR technique is used to fit a forecast model to a dataset of dependent and independent variable. Such a model can predict the future value of dependent variable Y (wind speeds) based on the new X's values. LR models are generally fitted using the maximum likelihood estimation or least square or approach [16]. This mathematical equation for LR can be generalized as (10):

$$Y = \beta_1 + \beta_2 X + \in \quad (10)$$

Where, β_1 is the intercept and β_2 is the slope. Collectively β_1 and β_2 are called regression coefficients. $\in$ is the error, *i.e.*, the part of Y that this forecasting model is unable to explain.

B. k-Nearest Neighbors Model

k-Nearest Neighbors or k-NN is a classification or regression based forecasting technique that accumulates all possible forecast outputs. Based on the most similar or resembling inputs, the statistical output is predicted using the distance formula. k specifies the number of neighbours that should be considered while making the classification or regression. The value of k effects the prediction results. In general, larger k value raise the precision level by reducing the overall noise. But bigger k is not always the winner. It is advised to keep k as small as possible (but not much smaller) because a smaller k may utilize subtler patterns. The value k relies on the complexity of the trend or pattern to be learned as well as the impact of noisy data [16].

The k-NN algorithm is sensitive to outliers. So, a more smooth and stable decision can be made by appropriately choosing the k value. The value of k can be optimized using data inspection. Cross-validation is an alternative to retroactively determine an optimum k-value by using an independent data set. Voronoi diagrams are a way to describe the k-NN algorithm solution. The Voronoi diagram is formed from lines that bisect and are perpendicular to the lines that connect two neighboring vertices [16, 19]. Fig. 2 shows the Voronoi Tessellation for k-NN algorithm. There is one Voronoi cell around every training example.

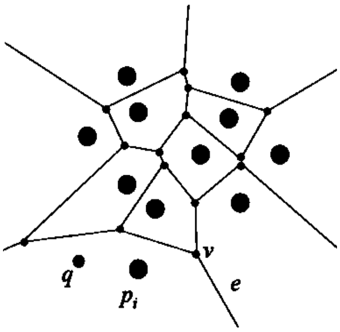

Fig. 2. Voronoi tessellation for k-NN [19].

Here p_i are the site or training points, q is the free or test point, e is the Voronoi edges around each site point, and v is the vertex of the Voronoi tessellation.

The dataset of interest is partitioned into different possible outputs as different symbols in Fig. 3. For any new prediction, the distances between the new input and its possible outputs are calculated. The Euclidian distance function is used for this purpose as shown in (11).

$$\text{Euclidian:} \sqrt{\sum_{i=1}^{k}\left(x_i - y_i\right)^2} \tag{11}$$

Also, the optimal value of k is computed as per the complexity of dataset (for example, k = 5 for Fig. 3).

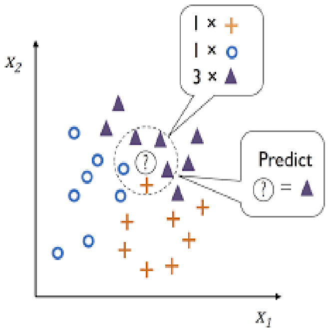

Fig. 3. k-NN decision-making criteria [19].

The nearest neighbors are found by ranking these distances in increasing order. In case, multiple possible outcomes are equidistant to the new input, the final output is predicted based on the probability of occurrence of a particular outcome.

C. Artificial Neural Network Model

The concept of ANN is analogous to the theory of biological neurons which processes and pass on the information to its adjacent neuron after processing. These neurons have a natural tendency to store experimental knowledge and use it when required [17, 20].

ANN is a set of interconnected input-hidden-output layers, each having perceptron units (a single neuron is called as perceptron). Perceptron of the adjacent layers are connected with the weighted values, computed using the learning process.

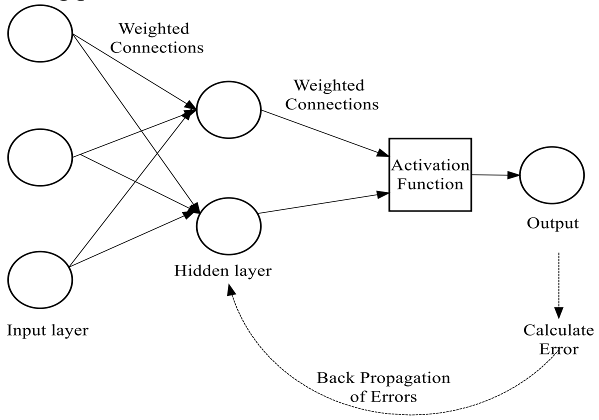

Fig. 4. ANN learning algorithm with back-propagation technique.

ANN is a complex adaptive system, *i.e.,* it has the ability to change its internal structure by adjusting weights of input layers. Based on the analytical procedure adopted for the evaluation process, neural networks are classified as Multilayer Perceptron (MLP) neural network, Elman Recurrent (ER) neural network and Simultaneous Cascade-Correlation (CC) neural network [20, 21]. This paper uses the MLP neural network approach for performing WSF.

1) Multilayer Perceptron Neural Network

MLP is one of the most compatible forms of neural network that provides a powerful solution to several classification and regression problems, including power systems. MLP captures the information from the learning or training dataset for Feed Forward Neural Network and assigns guess weights to the interconnections. Then Back Propagation technique is used to fit the input parameter(s) precisely to the MLP model. The performance metrics are computed for each iteration. The error is fed back for the re-computation of improved synaptic weights so that finally the prediction errors are minimized for a given set of input parameters. In case of MLP neural network, the number of hidden layer can be adjusted according to the needs. A large number of hidden layers may result in defining the input-output correlation with greater accuracy. However, it is often misinterpreted that increased count of hidden layers always leads to good results. In many cases, it may turn out in slowing down the evaluation procedure as well as lead to problems such as over-fitting, negative correlation, *etc.* [20]. Fig. 4 shows the MLP neural network architecture specifying ANN learning algorithm with Back-Propagation technique.

2) Mathematical Formulation

The objective function for the ANN based onsite WSF problem is the minimization of RMSE function given in (12). To minimize the RMSE, it is desired to have optimal values of synaptic weights. Synaptic weights refer to the strength or amplitude of a connection between the two neurons. A weight decides how much influence the input will have on the output.

The learning algorithm is iteratively applied until the optimized weights, wt and bias or threshold correlation coefficient, β are assigned such that any deviation from these weights and bias for the validation dataset keep on raising the final RMSE output of (12). For the given set of input-output parameters, x_i and y_i respectively, the RMSE is minimized iteratively as follows:

$$E(wt,\beta)=\sqrt{\frac{1}{h}\sum_{i=1}^{h}\left[y_i - f\left(x_i, wt, \beta\right)\right]^2} \qquad (12)$$

Where h is the number of neurons in each layer, E is the error signal. The procedure starts with an initial optimal guess values of wt and, β. In every new iteration, the values of wt and, β are modified as (13) and (14).

$$wt \to wt + \Delta wt \qquad (13)$$

$$\beta \to \beta + \Delta\beta \qquad (14)$$

The values of Δwt and $\Delta\beta$ are computed as (15):

$$f\left(x_i, wt+\Delta wt, \beta+\Delta\beta\right) \approx f\left(x_i, wt, \beta\right) + G\left(\Delta wt + \Delta\beta\right) \quad (15)$$

Where, G is the gradient of function f concerning wt and β. The process is repeated until RMSE of (12) becomes minimum and the gradient of $E(wt,\beta)$ is approximately zero.

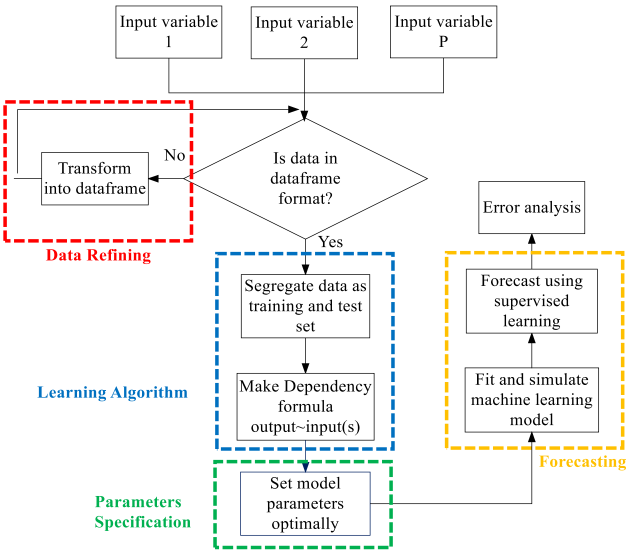

Fig. 5. Machine learning based WSF methodology.

D. Machine Learning based WSF Methodology

Detailed methodology for simulating machine learning models for univariate and multivariate parameters (wind speeds, directions, temperature, *etc.*) is depicted in Fig. 5.

1. **Data Refining:** The wind speed data for the given location is transformed into data frame format for performing machine learning simulations.
2. **Learning Algorithm:** The obtained dataset is then segregated into two parts- training dataset and test set. The training dataset is then used for the model learning process and the test set is used for model performance analysis. Then, a dependency formula is prepared, stating the output depends on the input, to execute the learning process.
3. **Parameter Specification:** Weights or parameters (linear coefficients for LR, k value for k-NN, and weights, hidden layers, hidden neurons and bias for ANN) are assigned based on the knowledge acquired from the training dataset. By adopting the closed loop path, the parameters are optimized for least possible errors.
4. **Forecasting:** The machine learning models are then fitted and simulated on the learning algorithm and then forecasting is performed.
5. **Error Analysis:** Finally, performance evaluation is done using RMSE and MAE performance metrics as per (8) and (9).

IV. Case Study

A. Dataset

The onsite forecast approaches are illustrated by simulating on the wind speed data of Jaisalmer, Rajasthan, India recorded at an interval of 10 minutes. The data is taken from the database of National Institute of Wind Energy, Chennai for ten days in September 2018. The wind speed is recorded at a hub height of 120 meters, having an average of 8.1166 m/sec. Along with wind speeds, prevailing wind direction, temperature and pressure at hub height are also used for forecasting. Whole data is arranged in a data frame format by creating a 2-D grid pattern having variables (wind speeds, prevailing wind direction, temperature and pressure) as the columns and the corresponding values arranged row-wise. Initial 121 observations are given as training set or input. Its affirmation on the learning dataset optimizes the implementation of the model fitting strategy. The next ten observations are used as the test set to evaluate the performance of the forecast models. Finally, the performance evaluation of the proposed model is assessed by comparing its forecast values and errors with benchmark models. Fig. 6 shows the actual wind speed data of class ‘time series’ which is used as an input for the statistical univariate forecasting models.

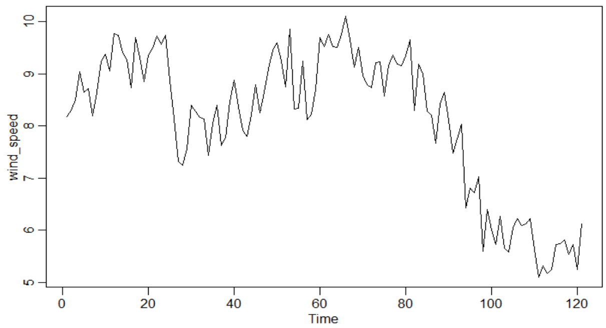

Fig. 6. Actual wind speed data used for simulations.

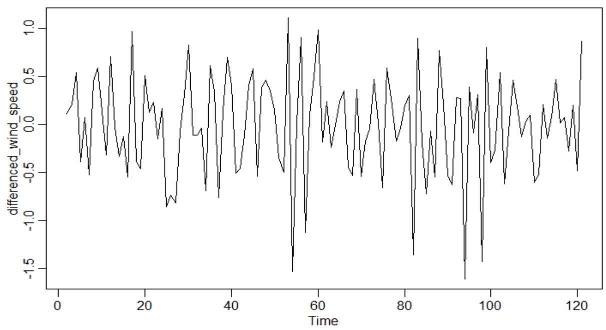

Fig. 7. Stationary differenced wind speed data.

B. ARIMA model

Time series models work on stationary dataset only. The wind speed data in Fig. 6 is non-stationary, *i.e.*, has a non-zero mean and fluctuating variance. Therefore, for using ARIMA forecasting model, the wind speed data should be converted into stationary data by taking the differencing of the original time series data and/or by adopting the differencing technique as shown in Fig. 7.

Next, to estimate and fit ARIMA model on the differenced time series of Fig. 7, the order of ARIMA model is computed. Model order estimation can be done using ACF and PACF plots analysis. Fig. 8 shows the ACF and PACF plots of the differenced wind speed data. Some conclusions are drawn from Fig. 8 that aids to find the order (p, d, q) of the ARIMA model.

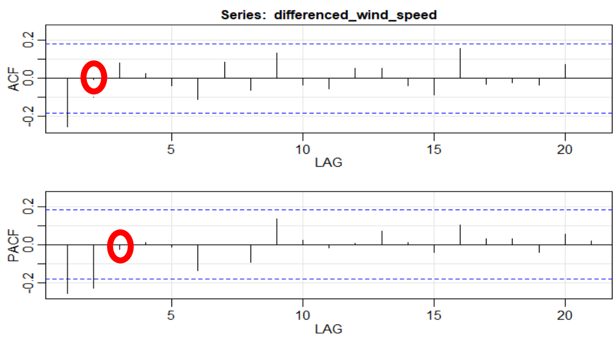

Fig. 8. ACF and PACF plots for differenced wind speed.

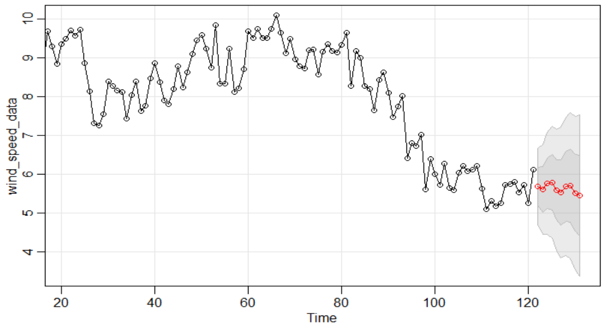

Fig. 9. Forecasts of wind speed obtained using ARIMA.

- This ACF and PACF plot is obtained after once differencing the original time series data, so, the value of $d = 1$.
- Both ACF and PACF plots are cutting off at some time lags instead of tailing off. So, both p and q are non-zero.
- The PACF plot cuts off at lag 3, so order of AR (p) = 3.
- The ACF plot cuts off at lag 2, so order of MA (q) = 2.
- So, ARIMA model order (p, d, q) is (3,1,2).

After obtaining the ARIMA model order, it is fitted and simulated on the differenced wind speed data. The results obtained are the forecasted valued for the next 10 points, each @ 10 minutes' time interval, *i.e.*, up to one-two hours ahead forecast. The 80% and 95% confidence interval are also shown in Fig. 9. The forecast results are shown graphically in Fig. 9 with red colour. Finally, the RMSE analysis is done based on the given test data and the forecast values, which comes out to be 0.834 m/sec or 10.27% deflection from actual wind speed.

C. k -Nearest Neighbors model

The wind speed data of class 'data frame' is given as input vector. k - NN regression is applied to forecast the value of the wind speeds. Twenty lags are used as training variables. The number of neighbors considered to forecast, k, is 3. The forecast error is 0.92 m/sec, *i.e.*, 11.33% deviation from actual wind speed. Fig. 10. shows the graphical form of the forecasted wind speeds using k -NN.

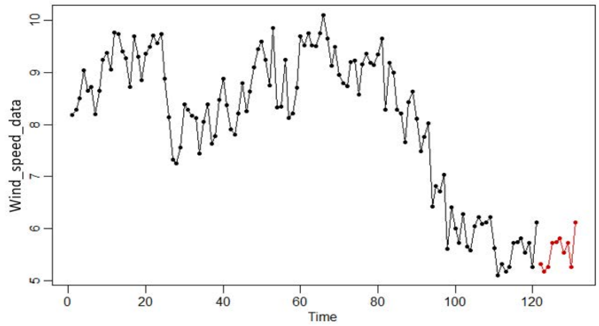

Fig. 10. Forecasts of wind speed obtained using k -NN.

D. Linear Regression model

The above mentioned wind speed dataset is made input as an object of class 'formula' to forecast using the LR technique. The linear model is fitted by giving a symbolic description of the linear predictor, *i.e.*, formula, and Gaussian error distribution. The RMSE forecast error for LR forecasting technique is 0.83 m/sec or the forecasted wind speed has a variation of 10.26% from the test values.

E. ANN Model

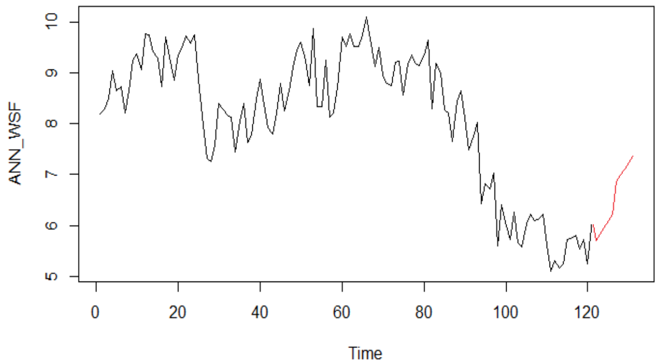

Fig. 11. Forecasts of wind speed obtained using ANN.

ANN forecasting model considers the following formula for performing forecast:

$$speed \sim time + speed \qquad (16)$$

Fig. 11 shows the graphical form of the forecasted wind speeds using ANN. The RMSE forecast error for ANN forecasting technique is 0.472 m/sec.

F. Comparative Performance Evaluation of WSF Models

The results obtained concerning the RMSE and MAE for the Jaisalmer wind speed data after executing various univariate forecasting models are presented in Table I.

The similar results are shown graphically in Figs. 12 and 13 concerning forecast wind speeds and RMSE, respectively.

TABLE I

FORECASTED WIND SPEEDS WITH RMSE AND MAE FOR DIFFERENT MODELS

Actual wind speeds (m/sec)	Predicted wind speeds (m/sec)			
	k -NN	LR	ANN	ARIMA
6.12	5.31	6.51	5.703	5.94
6.43	5.17	6.494	5.831	5.73
5.93	5.25	6.479	5.959	5.84
6.26	5.72	6.463	6.087	5.73
5.44	5.74	6.447	6.215	5.778
5.69	5.81	6.369	6.856	5.712
6.48	5.53	6.354	6.984	5.724
6.47	5.73	6.338	7.112	5.681
7.1	5.25	6.322	7.24	5.676
7.57	6.12	6.307	7.36	5.64
RMSE	0.920	0.830	0.472	0.834
MAE	0.864	0.654	0.415	0.782

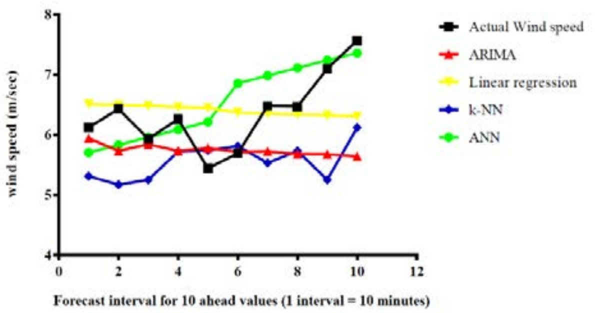

Fig. 12. WSF comparative analysis using ARIMA, LR, k –NN, and ANN.

Hence, the order of forecast accuracy for the models of interest is maximum for ANN followed by LR, ARIMA and then for k-NN. The accuracy for k-NN is least because it is more a classification approach rather than regression.

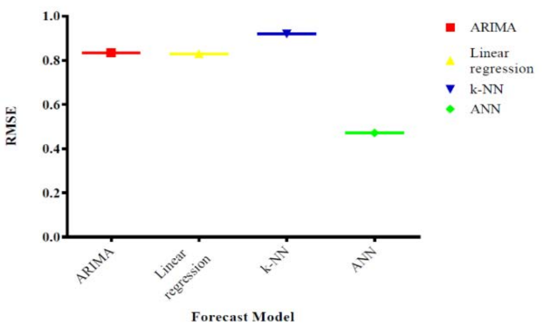

Fig. 13. RMSE comparison for ARIMA, LR, k -NN, and ANN.

V. CONCLUSION

The power system operational issues have spiked due to the grid integration of intermittent and uncertain wind power. Therefore, accurate wind resource forecasting is an indivertible necessity for optimal wind penetration to the grid. In this context, this paper first presents a review of the state-of-the-art of various elementary wind forecasting models. WSF models based on machine learning, namely, k-NN, LR, ANN are compared with time series ARIMA model. Numerical results imply that machine learning based ANN has maximum forecast accuracy out of the four mentioned models. ARIMA model is not so accurate because the forecasts converge to the mean of the series after some forecast values. Moreover, ANN fits the dataset best due to its enormous capacity to learn and predict. A possible improvement of the proposed work could be achieved by proposing an advanced WSF technique that considers the spatio-temporal dependency of WFs located near each other. Also, the impact assessment of spatio-temporal correlations in wind forecasting can be done in future.

VI. REFERENCES

[1] M. I. Alizadeh, M. Parsa Moghaddam, N. Amjady, P. Siano, and M. K. Sheikh-El-Eslami, "Flexibility in future power systems with high renewable penetration: A review," Renew. Sustain. Energy Rev., vol. 57, pp. 1186–1193, 2016.

[2] J. Jung and R. P. Broadwater, "Current status and future advances for wind speed and power forecasting," Renew. Sustain. Energy Rev., vol. 31, pp. 762–777, 2014.

[3] W. Chang, "A Literature Review of Wind Forecasting Methods," vol. 2, no. 4, pp. 161–168, April 2014.

[4] G. Sideratos and N. D. Hatziargyriou, "An Advanced Statistical Method for Wind Power Forecasting," vol. 22, no. 1, pp. 258–265, Feb. 2007.

[5] M. H. Albadi and E. F. El-Saadany, "Overview of wind power intermittency impacts on power systems," *Electric Power Syst. Res.*, vol. 80, no. 6, pp. 627–632, June 2010.

[6] L. Ma, S. Luan, C. Jiang, H. Liu, and Y. Zhang, "A review on the forecasting of wind speed and generated power," *Renewable Sustain. Energy Rev.*, vol. 13, no. 4, pp. 915–920, 2009.

[7] S. Soman, H. Zareipour, O. Malik, and P. Mandal, "A review of wind power and wind speed forecasting methods with different time horizons," in*Proc. North Amer. Power Symp.*, 2010, pp. 1–8.

[8] P. S. Georgilakis, "Technical challenges associated with the integration of wind power into power systems," *Renewable Sustain. Energy Rev.*, vol. 12, no. 3, pp. 852–863, Apr. 2008.

[9] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. Mckeogh, "Current methods and advances in forecasting of wind power generation," *Rene. Energy*, vol. 37, no. 1, pp. 1-8, Jan. 2012.

[10] J. Yan, Y. Liu, S. Han, Y. Wang, and S. Feng, "Reviews on uncertainty analysis of wind power forecasting," *Rene. and Sust. Energy Rev.*, vol. 52, pp. 1322-1330, Dec. 2015.

[11] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renew. and Sustain. Energy Rev.*, vol. 32, pp. 255-270, 2014.

[12] D. B. Alencae, R. C. L. Oliveira, and J. L. M. Rodrigaz, "Different models for forecasting wind power generation: Case Study," *Energies*, vol. 10, no. 1976, pp.1-27, 2017.

[13] K. Yunus, T. Thiringer, and P. Chen. "ARIMA-based frequency-decomposed modeling of wind speed time series," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 2546-2556, 2015.

[14] R. J. Hyndman, and G. Athanasopoulos (2018, May 8). *Forecasting: principles and practice*. OTexts, (2nd ed.) [Online]. Available: https://otexts.com/fpp2/

[15] S. Buhan, Y. Ozkazanc, and I. Cadirci, "Wind pattern recognition and reference wind mast data correlations with numerical weather prediction for improved wind-electric power forecasts," *IEEE Trans. Ind. Informat.*, vol. 12, no. 3, pp. 991-1004, Jun. 2016.

[16] T. Mithell, '*Machine learning*,' MIT press and McGrawHill, 1997.

[17] Q. Cao, B. Ewing, and M. Thompson, "Forecasting wind speed with recurrent neural networks," *Eur. J. Oper. Res.*, vol. 221, no. 1, pp. 148– 154, 2012.

[18] G.E.P. Box, and G. Jenkins, '*Time series analysis: forecasting and control*', 5th ed., John Wiley and Sons, 2015.

[19] S. Zhang, X. Li, M. Zhong, X. Zhu, "Effective knn classification with different numbers of nearest neighbors," *IEEE Trans Neural Netw Learn Syst.*, vol. 29, no. 5, pp. 1774-1785, 2017.

[20] Z. O. Olaofe, and K. A. Folly, "Wind power estimation using recurrent neural network technique," *IEEE Power Energy Soc. Conf. Expo. Africa Intell. Grid Integr. Renew. Energy Resour. Power Africa 2012*, pp. 1–7, 2012.

[21] M. Gavrilas and G. Gavrilas, "An enhanced ANN wind power forecast model based on a fuzzy representation of wind direction," *10th Symp. Neural Netw. Appl. Electr. Eng. NEUREL-2010 - Proc.*, pp. 31–36, 2010.