

# Wind Speed Prediction based on Incremental Extreme Learning Machine

Elizabeta Lazarevska

Faculty of Electrical Engineering and Information Technologies – Skopje, University “Ss. Cyril and Methodius” – Skopje, Macedonia, [elizabeta.lazarevska@feit.ukim.edu.mk](mailto:elizabeta.lazarevska@feit.ukim.edu.mk)

## Abstract

There are many research papers dealing with wind speed forecasting, since it is necessary in many applications, such as agriculture, modern transportation, and wind energy production. This paper presents an alternative approach to modeling and prediction of wind speed based on extreme learning machine, which is gaining a considerable interest in the scientific and research community at the present. Since the wind speed depends on the atmospheric weather conditions, the wind speed forecast in this research is based on different meteorological data, such as ambient temperature, relative humidity, light intensity, dew point, and atmospheric pressure. The paper presents two neural models for wind speed prediction, based on classic and incremental extreme learning machine, which exhibit the attributes of extreme simplicity, extremely good approximation performance, and extremely fast computation. The performance of the models is validated through their performance indices and compared to other available fuzzy and neural models for wind speed prediction. The paper also addresses the applied modeling techniques and proposes a modification which gives improved results and better approximation performance than the original techniques.

*Keywords: wind speed prediction, extreme learning machine, incremental extreme learning machine, random nodes*

## 1 Introduction

The world today widely acknowledges the need for alternative sources of energy to fuel fossils, which are still a primary energy source on Earth (Goldemberg, 2012). There are two main reasons for the intensified efforts and increased research in the field of alternative energy – the exhaustion of fossil fuels, due to the ever-growing energy demands on one hand, and the fact that fossil fuels are not renewable on the other hand, and the pollution caused by utilization of fossil fuels (Casper, 2010). Because of this, mankind is forced to turn to alternative sources of energy such as hydro, solar, wind, and biomass, which are known as renewable energy sources, since they are naturally replenished and are a

part of Earth's natural environment (Ehrlich, 2013). Among them the interest, research and investment in solar and wind energy are booming now (Brown et al., 2015). Solar and wind energy are similar in many ways. Both are renewable, both are intermittent, which means that both are not always available like on cloudy (sunless) or calm (windless) days, both are clean energies, which means no environmental pollution. However, the solar energy utilization has excelled at the residential level, which means that the roof-top solar panels can be seen everywhere, while the wind energy technologies have not yet. Nevertheless, wind is much promising, free, renewable, clean, and non-polluting source of energy, which is expected to play much more important role in the future in economics, electricity generation and emission control. As a matter of fact, wind energy is now considered as the fastest growing source and this trend is expected to continue (Walker and Swift, 2015). According to Global Wind Energy Council (GWEC) statistics (<http://www.gwec.net>), the global cumulative installed wind capacity at the end of 2015 was 432,419 MW, which is 25 times greater than the 17,400 MW installed as of 2000. Industry experts predict that with such pace of growth, one third of the world's energy needs will be met through wind power utilization by the end of 2050.

The bottleneck of wind energy utilization is the time-varying, stochastic, intermittent, and complex nature of wind speed. It is well-known that there is a non-linear cubic relationship between wind speed and the power output of wind turbines (Gasch and Tewe, 2012), which means that only a small deviation in wind speed will result in a large deviation in wind power output of the wind turbines. Therefore, it is of utmost importance for wind energy systems to accurately measure and estimate wind speed at a given site (Tamura et al., 2001; Soisuvarn et al., 2013; Yang and McKeogh, 2011). Normally, anemometers are used for measuring wind speed. However, measuring the wind speed is considered the most difficult among various climatological variables. For one, wind farm multiple anemometers must be used since the wind speed varies from one wind turbine to another and for other, the masts for mounting cup anemometers, which are the accepted standard for resource assessment, inevitably

become much taller as wind turbines grow in size, thus making the application of wind anemometers much more expensive. The high cost of wind anemometers discourages their widespread application, which is why engineers replace wind anemometers with digital wind speed estimators for broad applications, such as in wind farms (Kusiak et al., 2010; Mohandes et al., 2011).

Research works on wind speed and/or power prediction have been conducted on different time scale horizons (Soman et al., 2010). Since the time scales used for wind speed predictions are ranging from several minutes to several days, the wind speed forecasting techniques can be grouped into very short, short, medium and long-term methods, as is shown in Table 1. Here in this paper a medium-term wind speed (up to 1 day ahead) prediction has been considered.

Wind speed is a critical feature of wind resources, and since the wind speed values depend on the atmospheric weather conditions, the wind speed forecast in this research is based on different meteorological data, such as ambient temperature, relative humidity, light intensity, dew point, and atmospheric pressure.

## 2 A Short Overview of the Existing Models for Wind Speed Prediction

Many wind speed estimation methods are presented in the literature as of the present moment (Soman et al., 2010; Stensrud, 2007; Sideratos and Hatziargyriou, 2007; Arjun et al., 2014; Toires et al., 2005; Kavasseri and Seetharaman, 2009; El-Fouly and El-Saadany, 2006; Damousis et al., 2004; Lorenzo et al., 2011; Damousis and Dokopoulos, 2012; Li et al., 2001; Ramasamy et al., 2015; Barbounis et al., 2006; Mohandes et al., 2004; Haque et al., 2012). They can be classified according to different criteria. One such classification according to the adopted prediction period is shown in Table 1. Another classification of wind speed prediction models can be performed based on the applied method, as is shown in Figure 1.

The persistence is the simplest method for wind speed prediction that assumes of a strong correlation between present and future values of wind speed. In other words, it assumes that the future values of wind speed equal the present value. Despite its simplicity, the model is as good as any for short term predictions. Its accuracy decreases rapidly with increasing the prediction time scale.

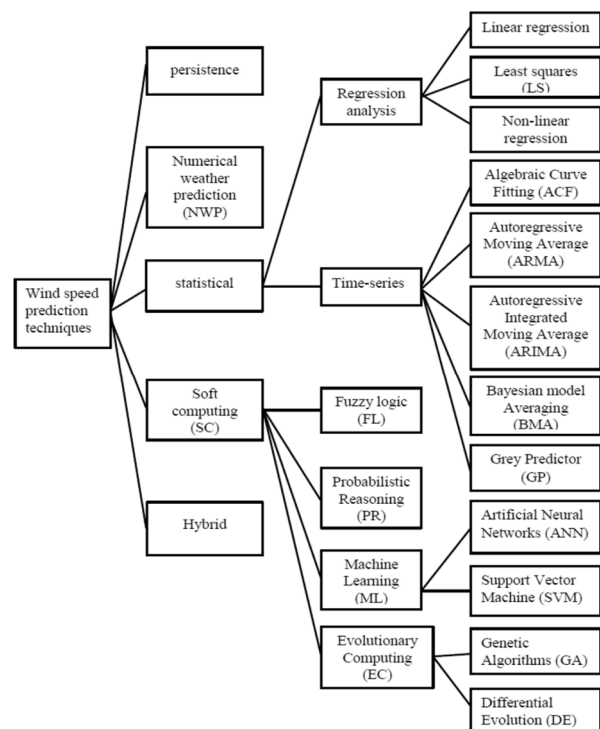
**Table 1.** Wind Speed Prediction Time Scales (Soman et al., 2010).

Time Horizon	Scale
Very short-term	few seconds up to 30 min ahead
Short-term	30 min up to 6 hours ahead
Medium-term	6 hours up to 1 day ahead
Long-term	1 day up to 5 days ahead

The numerical weather prediction (NWP) models (Stensrud, 2014) use mathematical models of the atmosphere and oceans to predict the weather based on current weather conditions. The complex mathematical calculations involved in modern weather prediction require super powerful computers, and yet, the forecasting ability of NWP does not extend past several days, due to the errors caused by the chaotic nature of the partial differential equations governing the atmosphere.

Statistical prediction methods include regression and time-series models (Sideratos and Hatziargyriou, 2007; Arjun et al., 2014; Toires et al., 2005; Kavasseri and Seetharaman, 2009; El-Fouly and El-Saadany, 2006), which can be linear and non-linear models, structural, grey-box, and black box models. Regression analysis is widely used for prediction and forecasting, and some of the most familiar methods are linear regression, ordinary least squares regression, nonlinear regression etc. The structural models for wind speed and power forecasting include an explicit function of meteorological variable predictions.

Along with the conventional forecasting methods, soft computing methods can also be used for wind speed prediction (Damousis et al., 2004; Lorenzo et al., 2011; Damousis and Dokopoulos, 2012; Li et al., 2001; Ramasamy et al., 2015; Barbounis et al., 2006; Mohandes et al., 2004; Haque et al., 2012). Recent research work has focused on artificial neural networks (ANN) and support vector machines (SVM), which generally produce superior approximation performance compared to other forecasting techniques. The wind speed models based on artificial intelligence techniques,



**Figure 1.** Wind speed prediction techniques.

such as ANN and SVM, belong to the class of black-box models. On the other hand, the expert systems based on fuzzy logic techniques, belong to the class of grey-box models. Hybrid models for short-term wind speed prediction have been presented in (Haque et al., 2012; Shi et al., 2013). They unite different modeling techniques and approaches, such as NN and genetic algorithms (GA), or NN and fuzzy inference systems (FIS).

Much more insight into different techniques for wind speed prediction can be found in (Bhaskar et al., 2010; Lawan et al., 2014; Lei et al., 2009).

### 3 A Brief Overview of Extreme Learning Machine

The extreme learning machine (ELM) is a new concept in the field of machine learning, which was introduced in (Huang et al., 2004). It was proposed as a new learning algorithm for training feedforward neural networks with single hidden layer (SLFFNN) which is different from the traditional gradient-descent based learning algorithms in the way of adjusting the network parameters. Unlike traditional learning algorithms, that iteratively adjust and tune all the network parameters during the learning process, ELM adjusts only the weights of the connections between the hidden and the output network layer, while the weights of the connections between the input and the hidden layer are assigned randomly at the beginning of the learning process and never updated. As a result, the training of a SLFFNN with ELM is much faster than the training of these networks with traditional gradient-descent based learning methods such as the error back-propagation method (BP).

A classic ELM is a SLFFNN, which has an input layer, one hidden layer and an output layer. For convenience, the weights of the connections between the input neurons and the hidden neurons can be called as input weights, and the weights of the connections between the hidden neurons and the output neurons can be called as output weights. The main features of ELM can be summarized as follows:

- The input parameters of the hidden layer, i.e. the input weights and the biases of the hidden nodes, are randomly assigned according to any continuous probability distribution and fixed during the complete learning process.
- The output parameters of the hidden layer, i.e. the output weights of the hidden nodes, are the only parameters that are learned during the training process.
- The output nodes of ELM do not possess biases.
- ELM is a model that is linear in the parameters.
- SLFFNN with ELM is a universal approximator.
- The hidden nodes activation functions can be almost any non-linear piece-wise continuous functions.

- The hidden nodes may have different output functions.

For a given input-output data set  $D = \{(\mathbf{x}_k, \mathbf{y}_k)\}; k = 1, 2, \dots, N$  with  $N$  distinct data points, where  $\mathbf{x}_k = [x_{k1}, x_{k2}, \dots, x_{kM}]^T$  is the  $k$ -th input vector of dimension  $M$ , and  $\mathbf{y}_k = [y_{k1}, y_{k2}, \dots, y_{kL}]^T$  is the  $k$ -th output vector of dimension  $L$ , a standard SLFFNN with ELM and  $n$  hidden neurons can be modeled as:

$$\tilde{\mathbf{y}}_k = \sum_{i=1}^n \mathbf{v}_i g_i(\mathbf{x}_k) = \sum_{i=1}^n \mathbf{v}_i g(\mathbf{w}_i \mathbf{x}_k + b_i); \quad (1)$$

$$k = 1, 2, \dots, N$$

The vector  $\mathbf{w}_i = [w_{i1} \ \dots \ w_{iM}]^T$  in (1) represents the  $i$ -th hidden neuron input weights, i.e. the weights defining the connections between the  $M$  input neurons and the  $i$ -th hidden neuron, the vector  $\mathbf{v}_i = [v_{i1} \ \dots \ v_{iL}]^T$  represents the  $i$ -th hidden neuron output weights, i.e. the weights defining the connections between the  $i$ -th hidden neuron and the  $L$  output neurons,  $b_i$  is the threshold, i.e. the bias of the  $i$ -th hidden neuron, and  $g_i(\mathbf{x}_k)$  is some appropriate activation function of the  $i$ -th hidden neuron; the term  $\mathbf{w}_i \mathbf{x}_k$  in (1) denotes the inner product between the vector of hidden layer input weights  $\mathbf{w}_i$  and the input vector  $\mathbf{x}_k$ .

The Equation (1) can be written in a short matrix form as:

$$\tilde{\mathbf{Y}} = \mathbf{W} \cdot \mathbf{V} \quad (2)$$

where  $\tilde{\mathbf{Y}}$  is the model output matrix:

$$\tilde{\mathbf{Y}} = \begin{bmatrix} \tilde{\mathbf{y}}_1 \\ \vdots \\ \tilde{\mathbf{y}}_N \end{bmatrix} = \begin{bmatrix} \tilde{y}_{11} & \dots & \tilde{y}_{1L} \\ \vdots & \ddots & \vdots \\ \tilde{y}_{N1} & \dots & \tilde{y}_{NL} \end{bmatrix} \quad (3)$$

the matrix  $\mathbf{W}$  is called a hidden layer output matrix (Huang and Babri, 1998; Huang, 2003):

$$\mathbf{W} = \begin{bmatrix} g(\mathbf{w}_1 \mathbf{x}_1 + b_1) & \dots & g(\mathbf{w}_n \mathbf{x}_1 + b_n) \\ \vdots & \ddots & \vdots \\ g(\mathbf{w}_1 \mathbf{x}_N + b_1) & \dots & g(\mathbf{w}_n \mathbf{x}_N + b_n) \end{bmatrix} \quad (4)$$

and  $\mathbf{V}$  is the model adjustable parameter matrix:

$$\mathbf{V} = \begin{bmatrix} \mathbf{v}_1 \\ \vdots \\ \mathbf{v}_N \end{bmatrix} = \begin{bmatrix} v_{11} & \dots & v_{1L} \\ \vdots & \ddots & \vdots \\ v_{n1} & \dots & v_{nL} \end{bmatrix} \quad (5)$$

Because ELM assigns randomly the hidden layer input weights  $\mathbf{w}_i$  ( $i = 1, 2, \dots, n$ ) and the hidden neuron biases  $b_i$  ( $i = 1, 2, \dots, n$ ) and does not adjust them further, the hidden neuron output matrix  $\mathbf{W}$  can be calculated only once at the beginning of the ELM learning process and remains fixed throughout the rest of the network training. Reference (Huang et al., 2004) has offered a proof that the input weights and the biases of the hidden neurons in a SLFFNN can be randomly

assigned if the activation function of the hidden layers is infinitely differentiable. Then, the hidden neuron output weights can be determined simply and analytically through the well-known least square methods, since the SLFFNN with fixed hidden neuron input weights and biases becomes a system linear in the parameters and can be solved with adequate techniques for such systems.

The training process of a SLFFNN with fixed hidden layer input weights and fixed hidden neuron biases amounts to finding a least square solution  $\mathbf{V}$  of the linear system:

$$\mathbf{Y} = \mathbf{W} \cdot \mathbf{V} \quad (6)$$

where  $\mathbf{Y} = [y_1 \ \dots \ y_N]^T$  is the vector of measured output data. When the number of hidden layer neurons  $n$  equals the number of input-output data  $N$ , the matrix  $\mathbf{W}$  is a square matrix and (6) can be solved for  $\mathbf{V}$  directly by inversion of  $\mathbf{W}$ , with a zero error. On the other hand, if the number of hidden neurons  $n$  does not equal the number of training data points  $N$ , which is always the case in real applications where  $n \ll N$ , the matrix  $\mathbf{W}$  is not a square matrix and the system (6) is solved for  $\mathbf{V}$  with an adequate least square method yielding the smallest norm least square solution (Huang et al., 2004):

$$\mathbf{V} = \mathbf{W}^+ \cdot \mathbf{Y} \quad (7)$$

In (7) the matrix  $\mathbf{W}^+$  stands for the Moore-Penrose generalized inverse of matrix  $\mathbf{W}$  (Rao and Mitra, 1971; Serre, 2002). The authors in (Huang et al., 2004) have argued that the least square solution (7) is unique, has the smallest norm of weights among all the least square solutions of (6) and secures a minimum training error of ELM.

#### 4 A Brief Overview of Incremental Extreme Learning Machine

The approximation capacity of ELM depends on the number of hidden neurons  $n$ . To solve this problem, it has been proposed to gradually increment the number of hidden neurons in the ELM, and this new learning algorithm, presented in Figure 2, has been called as incremental ELM (IELM) (Huang et al., 2006). The difference between the basic ELM and IELM is in the addition of new neurons to the hidden layer of the later. The new neurons can be added one at a time, or in groups, and the process of learning continues until the preset maximum number of hidden neurons is reached, or the preset acceptable model error is achieved. As with the basic ELM, the input parameters of the hidden layer in IELM are randomly generated and need not to be adjusted at all during the learning process (Huang et al., 2006). IELM is very different from other incremental algorithms proposed for SLFFNN with additive hidden nodes (Meir and Maierov, 2000), since it performs as a

universal approximator without tuning the hidden layer output parameters weights. In other words, when a new hidden neuron is added to the hidden layer, the IELM algorithm does not recalculate the hidden layer output parameters of the existing hidden nodes; once they have been determined, they remain fixed throughout the learning process. IELM just calculates the output weight for the newly added hidden neuron according to the following expression:

$$\mathbf{v}_n = \frac{\mathbf{E} \cdot \mathbf{G}^T}{\mathbf{H} \cdot \mathbf{G}^T} = \frac{\sum_{i=1}^N e(i)g(i)}{\sum_{i=1}^N g^2(i)} \quad (8)$$

The term  $g(i)$  in (8) denotes the activation of the added new hidden node for the  $i$ -th training input sample, while  $e(i)$  is the corresponding residual error before the addition of the new hidden node in question;  $\mathbf{G} = [g(1) \ \dots \ g(N)]^T$  is the activation vector for the new added node for all the training samples, and  $\mathbf{E} = [e(1) \ \dots \ e(N)]^T$  is the vector of the residual error before the addition of the new hidden neuron. The new value of the residual error after the addition of a new hidden neuron is calculated according to (Huang et al., 2006):

$$\mathbf{E} = \mathbf{E} - \mathbf{v}_n \cdot \mathbf{G}_n \quad (9)$$

Instead of the IELM algorithm described above, we have tested a much simpler algorithm which is a modification of the basic ELM in the sense that it accepts increasing number of hidden nodes and searches for the smallest preset error defined by  $\varepsilon$ . The algorithm recalculates the output parameters of all the hidden nodes in the hidden layer after every new addition to the hidden layer and performs until the preset maximum number of hidden neurons or the preset desired model error is reached. The algorithm has been compared to the other algorithms in Table 2 and has shown the best

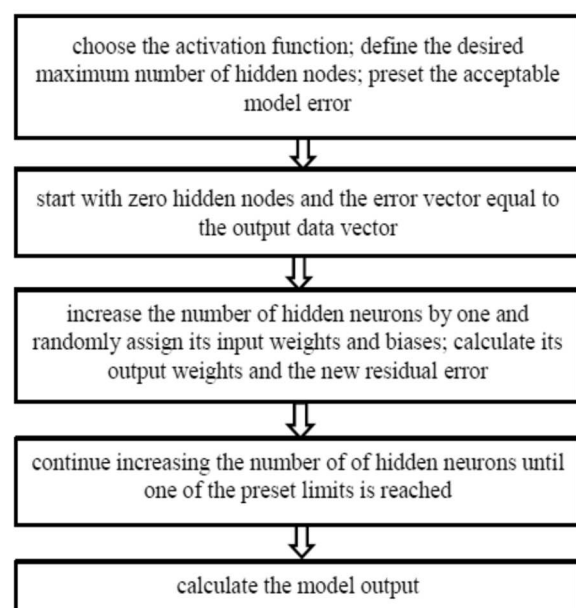


Figure 2. IELM Learning algorithm.

performance in this case. The reason for this lays in the fact that all the output parameters of the hidden layer are recalculated after each new addition of hidden neurons, but this is something that still needs to be examined further.

## 5 Results and Discussion

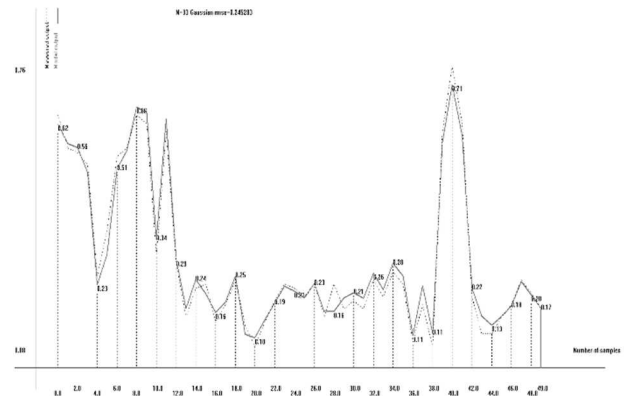
All the models for wind speed prediction in this research were built upon the hourly meteorological data for Mauna Loa (MOA), Hawaii, US for year 2015, available at ([http://www.nrel.gov/gis/data\\_wind.html](http://www.nrel.gov/gis/data_wind.html)).

The research considering the wind speed forecast was conducted in several directions. First the desired model of the wind speed was obtained by classic ELM with several different activation functions for the hidden neurons. To perform the modeling, the available experimental data were divided into two sets: training data and testing data, and fixed number of hidden nodes were assigned for the ELM algorithm. The input parameters of the hidden neurons  $w_i$  ( $i = 1, 2, \dots, n$ ) and  $b_i$  ( $i = 1, 2, \dots, n$ ) were randomly assigned according to the uniform probability distribution. The obtained results were compared according the RMSE criterion and the experimental findings are in complete agreement with the claim that ELM produces good generalization performance with almost any non-zero nonlinear activation function (Huang et al., 2004; Huang and Babri, 1998). However, they contradict the claim in (Liu et al., part I, 2015) that "...the performance of ELM depends heavily on the activation function..."

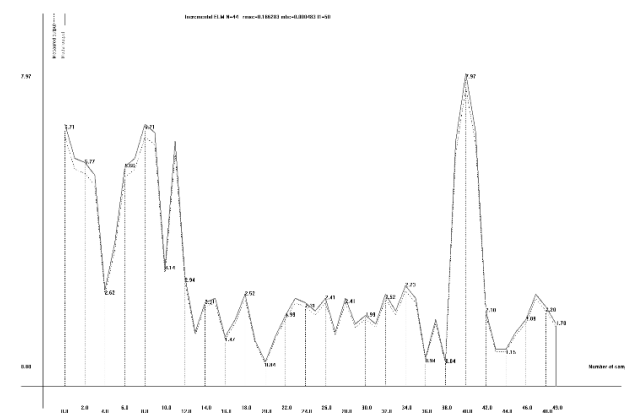
The other group of simulations were performed with the same activation function and a fixed number of hidden neurons, to illustrate the approximation performance of the applied ELM about different random assignments of the hidden layer input parameters. These simulations undoubtedly showed that the approximation performance of classic ELM depends on the randomness mechanism. The random assignment of some ELM parameters causes an uncertainty problem with classic ELM, since there is not a way of knowing whether the obtained model is the best possible solution, i.e. what is the best random assignment of the specific ELM parameters. This dependence was not explicitly and/or clearly mentioned in any of the consulted references at the end of this paper, except in (Liu et al., part I, 2015) to be addressed in (Liu et al., part II, 2015). However, the conducted research in this work showed that the approximation property of ELM suffers from the randomness of the ELM with any type of activation function for the hidden neurons, and not only with Gaussian-type activation functions as concluded in (Liu et al., part II, 2015).

To solve the above problem, in this research the following simple approach is used. After the selection of appropriate activation function and the number of hidden neurons, the desired value of the model error is

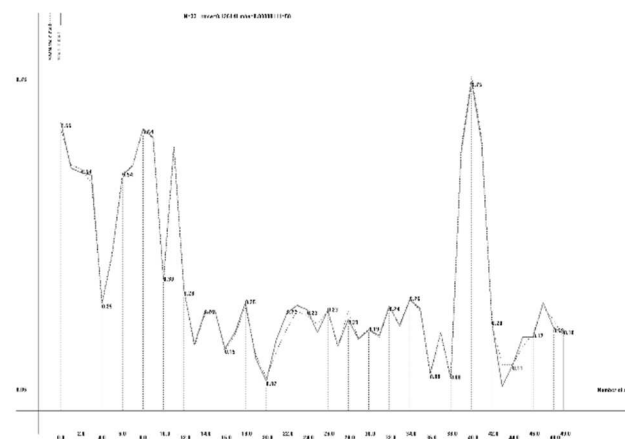
set as  $\varepsilon$ . Then the training of the constructed SFFNN with ELM is conducted and the error of the obtained model is compared to the preset desired value  $\varepsilon$ . If the model error is greater than  $\varepsilon$ , the training process is repeated. Otherwise, it is considered that the obtained model has the desired accuracy. The model produced in this way, and shown in Figure 3, had much smaller error than the models obtained in the other trials.



**Figure 3.** NN model for prediction of wind speed based on our version of ELM;  $n=33$ , Gaussian, RMSE=0.245283.



**Figure 4.** NN model for prediction of wind speed based on IELM;  $n=44$ , Gaussian, RMSE=0.166283.



**Figure 5.** NN model for prediction of wind speed based on our version of IELM;  $n=33$ , Gaussian, RMSE=0.126141.

Finally, the wind speed prediction models obtained with our version of ELM, incremental ELM and with our version of incremental ELM were compared to some other models obtained with different modeling techniques, such as the position type and position-gradient type fuzzy models based on Sugeno and Yasukawa identification method (Sugeno and Yasukawa, 1993), the RVM based NN model (Tipping, 2001), and the neuro-fuzzy model based on extended RVM (Kim et al., 2006), and presented in (Lazarevska, 2016). The results are shown in Table 2. All the models are compared according to the RMSE criterion, defined as follows

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \tilde{y}_i)^2} \quad (10)$$

where  $y_i$  is the measured output,  $\tilde{y}_i$  is the model output, and  $N$  is the number of experimental data. Figure 3 through Figure 5 shows the output (solid line) of the wind speed prediction models from Table 2 obtained with different ELM algorithms compared to the actual measured output (dashed line). The neural model based on our modification of IELM has the best performance index, since it takes into consideration the randomness factor in ELM.

**Table 2.** Comparison of Wind Speed Prediction Models obtained by Different Modeling Techniques.

Model	RMSE
Position type fuzzy model	0.37940
Position-gradient type fuzzy model	0.32747
RVM based neural model	0.342874
Neuro-fuzzy model based on RVM	0.322158
NN model based on our version of ELM	0.245283
NN model based on IELM	0.166283
NN model based on our version of IELM	0.126141

## Acknowledgements

The author gratefully acknowledges that the wind speed prediction models in this research have been based on the wind data available from the National Renewable Energy Laboratory NREL ([http://www.nrel.gov/gis/data\\_wind.html](http://www.nrel.gov/gis/data_wind.html)) which is a national laboratory of the US Department of Energy, Office of Energy Efficiency and Renewable Energy, operated by the Alliance for Sustainable Energy (LLC).

## References

N. N. Arjun, V. Prema, D. K. Kumar, P. Prashanth, V. S. Preeksht and K. U. Rao. Multivariate regression models for prediction of wind speed. In *2014 International Conference*

- on Data Science and Engineering (ICDSE)*, Kochi 26-28 Aug. 2014, pp. 171-176.
- T. G. Barbounis, J. B. Theocharis, M. C. Alexadis and P. S. Dokopoulos. Long term wind speed and powercasting using local recurrent neural network models. *IEEE Transactions on Energy Conversion*, 21(1): 273-284, 2006.
- M. Bhaskar, A. Jain and N. V. Srinath. Wind speed forecasting: Present Status. In *2010 International Conference on Power System Technology (POWERCON)*, 24-28 Oct. 2010, Hangzhou, China, 1-6.
- L. R. Brown, E. Adams, J. Larsen and J. M. Roney. *The Great Transition: Shifting from Fossil Fuels to Solar and Wind Energy*, 1st edition. W. W. Norton & Company. 2015.
- J. K. Casper. *Fossil fuels and pollution: The future of air quality (Global warming (Facts on File))*, 1st edition. Facts on File. 2010.
- I. G. Damousis and P. Dokopoulos. A fuzzy expert system for the forecasting of wind speed and power generation in wind farms in *Wind Energy Conversion Systems: Technology and Trends*. S. M. Mayeen ed., London: Springer, 2012, 197-226.
- I. G. Damousis, M. C. Alexiadis, J. B. Theocharis and P. S. Dokopoulos. A fuzzy model for wind speed prediction and power generation in wind parks using spatial correlation. *IEEE Transactions on Energy Conversion*, 19(2): 352-361, 2004.
- R. Ehrlich. *Renewable energy: A first course*, 1st Edition. CRC Press. 2013.
- T. M. H. El-Fouly and E. F. El-Saadany. Grey predictor for hourly wind speed and power forecasting. *IEEE Transactions on Power Systems*, 21(3), 1450-1452, 2006.
- R. Gasch and J. Twele. *Wind Power Plants: Fundamentals, Design, Construction, and Operation*, 2nd edition. Springer. 2012.
- J. Goldemberg. *Energy: What everyone needs to know®*, 1st edition. Oxford University Press. 2012.
- GWEC," *Global wind statistics 2015*," GWEC, 2015. Available at: [http://www.gwec.net/wp-content/uploads/vip/GWEC-PRstats-2015\\_LR\\_corrected.pdf](http://www.gwec.net/wp-content/uploads/vip/GWEC-PRstats-2015_LR_corrected.pdf) (accessed 06.01.2016)
- A. U. Haque, P. Mandal, J. Meng, M. E. Kaye and L. Chang. A new strategy for wind speed forecasting using hybrid intelligent models. In *2012 25th IEEE Canadian Conference on Electrical and Computer Engineering CCECE*, 2012, 1-4.
- G. -B. Huang. Learning capability and storage capacity of two hidden-layer feedforward networks. *IEEE Transactions on Neural Networks*, 14(2): 274-281, 2003.
- G. -B. Huang and H. A. Babri. Upper bounds on the number of hidden neurons in feedforward networks with arbitrarily bounded nonlinear activation functions. *IEEE Transactions on Neural Networks*, 9(1): 224-229, 1998.
- G. -B. Huang, L. Chen and C. -K. Siew. Universal approximation using incremental constructive feedforward networks with random hidden nodes. *IEEE Transactions on Neural Networks*, 17(4): 879-892, 2006.
- G. -B. Huang, Q. -Y. Zhu and C. -K. Siew. Extreme learning machine: A new learning scheme of feedforward neural networks. In *Proceedings of the IEEE International Joint*

- Conference on Neural Networks (IJCNN2004)*, 25-29 July, 2004, Budapest, Hungary, 985-990.
- R. G. Kavasseri and K. Seetharaman. Day-ahead wind speed forecasting using f-ARIMA models. *IEEE Transactions on Renewable Energy*, 34(5): 1388-1393, 2009.
- J. Kim, Y. Suga and S. Won. A new approach to fuzzy modeling of nonlinear dynamic systems with noise: relevance vector learning mechanism. *IEEE Trans. on Fuzzy Systems*, 14(2): 222-231, 2006.
- A. Kusiak and W. Li. Estimation of wind speed: a data-driven approach. *Journal of Wind Engineering and Industrial Aerodynamics*, 98(10-11): 559-567, 2010.
- S. M. Lawan, W. A. W. Z. Abidin, W. Y. Chai, A. Baharun and T. Maasri. Different models of wind speed prediction: A comprehensive review. *International Journal of Scientific and Engineering Research*, 5(1): 1760- 1768, 2014.
- E. Lazarevska. A neuro-fuzzy model for wind speed prediction based on statistical learning theory. *Journal of Electrical Engineering and Information Technologies*, 1(1-2): 45-55, 2016.
- M. Lei, L. Shiyun, J. Chuanwen, L. Hongling and Z. Yan. A review on the forecasting of wind speed and generated power. *Renewable and Sustainable Energy Reviews*, 13(4): 915-920, 2009.
- S. Li, D. C. Wunsch, E. A. O'Hair and M. G. Giesselmann. Using neural networks to estimate wind turbine power generation. *IEEE Trans. Energy Convers.*, 16(3): 276-282, 2001.
- J. Lorenzo, J. Méndez, M. Castrillón and D. Hernández. Short-term wind power forecast based on cluster analysis and artificial neural networks. In *Proceedings of the 11th International Work-Conference on Artificial Neural Networks, IWANN 2011*, Torremolinos-Málaga, Spain, June 8-10, 2011, Part I, 191-198.
- X. Liu, S. Lin, J. Fang and Z. Xu. Is extreme learning machine feasible? A theoretical assessment (Part I). *IEEE Transactions on Neural Networks and Learning Systems*, 26(1): 7-20, 2015.
- X. Liu, S. Lin, J. Fang and Z. Xu. Is extreme learning machine feasible? A theoretical assessment (Part II). *IEEE Transactions on Neural Networks and Learning Systems*, 26(1): 21-34, 2015.
- R. Meir and V. E. Maiorov. On the optimality of neural network approximation using incremental algorithms. *IEEE Transactions on Neural Networks*, 11(2):323-337, 2000.
- M. A. Mohandes, T. O. Halawani, S. Rehman and A. A. Hussain. Support vector machine for wind speed prediction. *Renewable Energy*, 29(6): 939-947, 2004.
- M. Mohandes, S. Rehman and S. M. Rahman. Estimation of wind speed profile using adaptive neuro-fuzzy inference system (ANFIS). *Applied Energy*, 88(11): 4024-4032, 2011.
- [http://www.nrel.gov/gis/data\\_wind.html](http://www.nrel.gov/gis/data_wind.html) (accessed 06.01.2016)
- P. Ramasamy, S. S. Chandel A. K. Yadav. Wind speed prediction in the mountainous region of India using an artificial neural network model. *Renewable Energy*, 80: 338-347, 2015.
- C. R. Rao and S. K. Mitra. *Generalized Inverse of Matrices and its Applications*. New York: Wiley. 1971.
- D. Serre. *Matrices: Theory and Applications*. New York: Springer-Verlag. 2002.
- N. Shi, S. -Q. Zhou, X. -H. Zhu, X. -W. Su and X. -Y. Zhao. Wind speed forecasting based on grey predictor and genetic neural network models. In *2013 International Conference on Measurement, Information and Control (ICMIC)*, 16-18 Aug. 2013, 2:1479 – 1482.
- G. Sideratos and N. Hatzigargyriou. An advanced statistical method for wind power forecasting. *IEEE Transactions on Power Systems*, 22(1): 258-265, 2007.
- S. Soisuvarn, Z. Jelenak, P. S. Cheng, S. O. Alsweiss and Q. Zhu. CMOD5.H. - A high wind geophysical model function for c-band vertically polarized satellite scatterometer measurements. *IEEE Transactions on Geoscience and Remotesensing*, 51(6): 3741-3760, 2013.
- S. S. Soman, H. Zareipour, O. Malik and P. Mandal. A review of wind power and wind speed forecasting methods with different time horizons. *North American Power*, 2(5): 8-16, 2010.
- D. J. Stensrud. *Parametrization Schemes: Key to understanding numerical weather prediction models*. Cambridge University Press. 2007.
- M. Sugeno and T. Yasukawa. A fuzzy-logic-based approach to qualitative modeling. *IEEE Transactions on Fuzzy Systems*, 1: 7-33, 1993.
- Y. Tamura, K. Suda, A. Sasaki, Y. Iwatani, K. Fujii, R. Ishibashi and K. Hibi. Simultaneous measurements of wind speed profiles at two sites using Doppler sodars. *Journal of Wind Engineering and Industrial Aerodynamics*, 89(3-4): 325-335, 2001.
- M. E. Tipping. Sparse Bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1: 211-244, 2001.
- J. L. Torres, A. García, M. De Blas and A. De Francisco. Forecast of hourly average wind speed with ARMA models in Navarre (Spain). *Solar Energy*, 79(1): 65-77, 2005.
- R. P. Walker and A. Swift. *Wind energy essentials: Societal, economic, and environmental Impacts*, 1st edition. Wiley. 2015.
- S. Yang and E. McKeogh. LIDAR and SODAR measurements of wind speed and direction in upland terrain for wind energy purposes. *Remote Sensing*, 3(9):1871-1901, 2011.