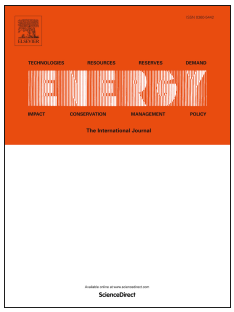
Journal Pre-proof



Using atmospheric inputs for Artificial Neural Networks to improve wind turbine power prediction

Jordan Nielson, Kiran Bhaganagar, Rajitha Meka, Adel Alaeddini

PII: S0360-5442(19)31968-1

DOI: https://doi.org/10.1016/j.energy.2019.116273

Reference: EGY 116273

To appear in: *Energy*

Received Date: 12 March 2019

Revised Date: 24 September 2019

Accepted Date: 1 October 2019

Please cite this article as: Nielson J, Bhaganagar K, Meka R, Alaeddini A, Using atmospheric inputs for Artificial Neural Networks to improve wind turbine power prediction, *Energy* (2019), doi: https://doi.org/10.1016/j.energy.2019.116273.

LA REVUE GESTION ET ORGANISATION 00 (2014) 000–000

HOSTED BY

ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Journal homepage: www.elsevier.com/locate/rgo

LA REVUE GESTION & ORGANISATION

USEK HOLY SPIRIT UNIVERSITY OF KASLIK

Using Atmospheric Inputs for Artificial Neural Networks to Improve Wind Turbine Power Prediction

Jordan Nielson[a], Kiran Bhaganagar[b*], [c]Rajitha Meka, and [d]Adel Alaeddini

[a] *Ph.D. Candidate, Department of Mechanical Engineering, University of Texas at San Antonio, Laboratory of Turbulence, Sensing and Intelligence Systems, San Antonio, Texas, USA*

[b] Associate Professor Department of Mechanical Engineering, University of Texas at San Antonio, Laboratory of Turbulence, Sensing and Intelligence Systems, San Antonio, Texas, USA **Corresponding Author*

[c] *Ph.D. Student, Department of Mechanical Engineering, University of Texas at San Antonio*

[d] Associate Professor Department of Mechanical Engineering, University of Texas at San Antonio

ARTICLE INFO

Article history:

Received 00 December 00

Received in revised form 00 January 00

Accepted 00 February 00

Keywords:

Each keyword to start on a new line

ABSTRACT

A robust machine learning methodology is used to generate a site-specific power-curve of a full-scale isolated wind turbine operating in an atmospheric boundary layer to drastically improve the power predictions, and, thus, the forecasting of the monthly energy production estimates. The study has important implication in measuring the financial feasibility of wind farms by improving the accuracy of monthly energy estimates. The significance of the study is that atmospheric stability and air-density are accounted in the power predictions of the wind turbine. Artificial Neural Networks (ANN) machine learning approach is used to generate multi-parameter input models to estimate the power produced by the wind turbine. The ANN is a Feed Forward Back Propagation (FFBP). The power- and wind-data is obtained from a 2.5 MW wind turbine that has a Meteorological tower located 900m Southwest of the wind turbine in Kirkwood, Iowa, USA. The study investigates the role of atmospheric boundary-layer metrics – Wind Speed, Density (a measure of stratification), Richardson Number, turbulence intensity, and wind shear as input parameters into the ANN model. The study investigates the influence of FFBP ANN hyper-parameters on the power prediction accuracy. Comparison of the FFBP ANN model to other power curve correction techniques demonstrated an improvement in the Mean Absolute Error (MAE) of 40% when compared to the density correction (the next closest). The five-parameter 4-layer FFBP ANN has an average energy production error of 0.4% for the nine months while the IEC is -3.7% and the air density correction is -1.9%. Finally, the study determines the performance of the FFBP ANN model for different atmospheric stability regimes (Unstable, Stable, Strongly Stable, Strongly Unstable and Neutral) classified using two criterions - Richardson number and Turbulence intensity. The largest MAE occurs during the strongly stable regime of the atmospheric boundary layer for both criteria.

1. Introduction

* *Kiran Bhaganagar*. Kiran.bhaganagar@utsa.edu

With exploding populations in the world and the need to meet their ever-increasing energy demands, it is becoming critical for alternative energy sources to be a reliable form of energy. Wind energy is an efficient resource, and the United States wind supply is abundant, as per the recent 2017 Department of Energy report [1]. In the United States, wind has the largest renewable generation capacity of all renewable energies. However, for wind power to compete with the conventional generation sources, and to be a sustainable energy resource, it is crucial for it to be cost-effective. Wind energy becomes more valuable with increased accuracy of prediction of wind-power because more if it can be transmitted to the grid. The uncertainty in the prediction of power due to the stochastic nature of the wind is the ongoing challenge for the wind power generation industry. Within a wind farm, the output power of an individual wind turbine varies with the wind speed, air density, wind turbulence. The inflow wind conditions for each wind turbine is different, so each wind turbine has a unique power performance curve.

The International Standard IEC 61400-12-1 has provided a standard methodology for measuring the power performance of a specific wind turbine [2]. At the test site, the hub height wind-speed and the power generated by the wind turbine are measured for a long duration of time to generate a significant database under varying atmospheric conditions. The power measurements implicitly include the turbulence levels at that site. A method of bins is applied to determine the power curve for that wind turbine, and this is referred to as site-specific power curve for that specific wind turbine. Figure 1 shows an example of a site-specific power curve developed. The data were collected every 10 minutes as averaged wind speed measurements. The figure shows the significant variations of power that can occur at each wind speed. Variations in air density and atmospheric stability are leading causes for the significant variations in power at a given wind speed [3–5]. Therefore, there has been a focus to improve site-specific power curve predictions.

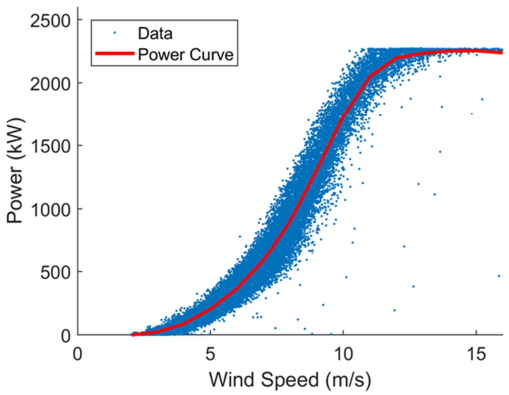


Figure 1. Site specific Power Curve

Recently, deep learning data-analytic methods, such as machine learning are becoming popular to improve power curves. Machine learning allows for a power prediction model to have multiple inputs (an even greater number than described above by power curve corrections). Therefore, the power prediction can incorporate stability effects of the atmospheric boundary layer. Clifton et al. [6] used stochastic simulations, which generated realistic wind speed and wind turbine power data, to develop a regression tree model of power prediction using turbulence intensity at hub height and vertical wind shear. Their results showed a three-fold improvement, with a reduction in prediction scatter from 4% to 1.3%, in power estimation with the machine-learned model. Similarly, Pelletier et al. [7] used a six parameter artificial neural network to predict power production to identify faulty wind turbines. The results showed a higher reduction in MAE compared to the other power curve models. In some studies, machine learning techniques have been used with multiple variables such as wind-speed, wind-direction, and hub-height temperature [12–15].

The literature shows that including additional inputs into power curve models can improve accuracy. Models that use corrections do not fully capture the complex nonlinearities that occur due to the atmospheric boundary layer. Machine learning power curves on the other hand; have been based on data available rather than utilizing metrics that quantify the atmospheric stability. This study is unique because it builds upon the vast studies that have shown that variations in atmospheric stability cause errors in the wind turbine power curves [3,11–15]. Therefore, the machine learning algorithms used in the study are built using metrics that quantify atmospheric stability.

Atmospheric Boundary Layer stratification is generally categorized into three regimes: stable-, neutral-, and unstable- stratification [16]. The stable regime occurs, generally, during night-time conditions where surface cools the air. Therefore, there is a large amount of wind shear with very little turbulence [17,18]. The unstable stratification conditions occur, generally, during the day-time when the surface heats the surrounding air close to the ground. This causes vertical fluctuations and creates large amounts of mixing in the atmosphere. Therefore, the unstable regime has uniform wind speeds with significant turbulent fluctuations [19–21]. Neutral conditions occur when there are negligible buoyancy effects [22,23]. The different stabilities create a wide range of performance in wind turbines [3–5,24–27]. Nielson and Bhaganagar [28,29] showed that the day to day variations of stability can lead to a wide range of energy production forecasts. Their study showed that one sigma standard deviation of surface heat flux resulted in a difference in energy production of 10%. The current study combines this knowledge of atmospheric stability to build machine learning algorithms to improve site-specific power prediction.

Hence, in the current study, the focus is on improving the input variables to build the power curve. The

study includes variations in the atmosphere as inputs into an Artificial Neural Network (ANN), which is a type of machine learning. An ANN is a tool that mimics the neurons in a brain. The study uses different methods to quantify the atmospheric stability (also known as atmospheric metrics) determined from previous research to characterize the atmospheric stability as inputs into the ANN.

The current study uses Feed Forward Back Propagation (FFBP) ANN models to predict the power output of a single wind turbine. The benefit of this approach is improved accuracy prediction of power production by including atmospheric effects.

The objectives of the study are to

1. To determine the effectiveness of different atmospheric inputs, including wind speed, air density, wind shear, turbulence intensity, and Richardson Number, used as inputs in the FFBP ANN models, to investigate the optimal number of parameters used as inputs, and to investigate the influence of the FFBP ANN Hyper-Parameters.
2. To compare the error from the FFBP ANN model to other power curve adjustments to quantify the performance and determine how this affects the energy production estimates.
3. To determine the performance of the FFBP ANN model in different atmospheric stability regimes.

2. Methodology

The details of the machine learning algorithms are discussed in section 2.1. Section 2.2 gives the details of the wind turbine data set used for the analysis.

2.1. Machine Learning Algorithms

In this paper, the aim is to provide comprehensive study on the performance of different models for predicting the power of a wind turbine. The study compares the performance of two Artificial Neural Network (ANN) models - Feed Forward Back Propagation (FFBP) neural networks and Radial Basis Function (RBF) neural networks, Random Forest Regression (RF), Support Vector Regression (SVR) and, Gaussian Process Regression (GPR) models.

Feed Forward Back Propagation (FFBP) neural networks

Feed forward back propagation neural networks also known as multi-layer feed forward neural networks or multi-layer perceptron (MLP) or just back propagation neural networks are well known type of ANNs used for wide variety of tasks like prediction, function approximation or pattern classification [30,31]. The MLP model consists of one input layer, at least one hidden layer and one output layer [32,33]. The model correlates inputs (i.e., wind speed and air density) to the output (wind turbine power). It uses a set of connecting links (neurons or hidden layers) that each has a specific weight. The inputs pass through the neurons, or hidden layers, and are multiplied by the weights. Therefore, each neuron has a function $y = \sum_i^n I_i w_i$ where y is the output, I_i is the i^{th} input, and w_i is the i^{th} weight. The goal is to determine the weights to minimize the error at the output. One of the main advantages of neural networks is that they can "learn" from data to determine the weights. The learning occurs through an optimization process, where the weights of the model hidden layers are updated based on available training patterns [34] and compared with data that has known outputs. There are many numerical optimization techniques used to determine the weights of the neurons. This study uses Keras library with Tensorflow, which is open source and free of cost, backend with the Adam optimization algorithm for implementing the MLP model [35], referenced as FFBP ANN. Adam optimization is selected because it is known to work well in practice compared to other adaptive learning algorithms with fairly robust nature to the choice of hyper parameters [36–38].

Adaptive Moment Estimation, popularly known as the Adam optimization algorithm is an efficient stochastic optimization method that requires only first order gradients with a small memory requirement [39]. This method combines the advantages of the AdaGrad [40] and RMSProp [41] methods. The algorithm updates the estimates of the first and second moments of the gradients which are the exponential moving averages of gradient (m_t) and squared gradient (v_t). The moving averages are calculated as $m_t = \beta_1 m_{t-1} + (1-\beta_1) g_t$ and $v_t = \beta_2 v_{t-1} + (1-\beta_2) g_t^2$ with g_t being the gradient of the objective function with respect to the parameter θ at time step t .The hyper parameters $\beta_1, \beta_2 \in [0,1)$ control the exponential decay rates of the moving averages. The moving averages are initialized using a vector of 0's which leads to a biased moment estimates during the initial time steps. To counteract the bias, the moments are computed as $\widehat{m_t} = m_t/(1-\beta_1^t)$ and $\widehat{v_t} = v_t/(1-\beta_2^t)$. The parameters (θ) are updated using the rule, $\theta_{t+1} = \theta_t + \eta/\sqrt{(\widehat{v_t}+\epsilon)}\widehat{m_t}$ with η and ϵ being the learning rate and step size respectively [6] . Although the algorithm works well with the default hyper parameters, Goodfellow et al. [42] suggests that the learning rate might need to be changed sometimes.

Instead of applying the optimization algorithm for the whole data, the training data is split into mini-batches. Training over all the mini-batches once is called an epoch [43]. This study uses design of experiments [44] to optimize the batch size, number of epochs and the learning rate of the Adam optimization algorithm. A full factorial design is generated with 100,200,500,800 and 1000 as five levels for batch size; 10,20,50,100 and150 as five levels for number of epochs and 0.001, 0.05 and 0.1 as three levels for learning rate. For different network architectures, the study used the trial and error method to determine the values of the parameters.

The neural network architecture plays a major role as different network structures can result in different performances of the model [45]. The problem of finding the best network is considered as extremely difficult [46]. The two main aspects of the ANN architecture are to determine the 1) number of hidden layers and 2) number of neurons at each layer [47]. In addition to these, the type of activation function used in each layer also plays a critical role on the performance of the model [48]. The activation function is used to convert the activation level of neurons into the output signal [49]. Some of the most commonly used activation functions are: Sigmoid, Hyperbolic Tangent and Rectified Linear Units (ReLU).

The study uses the following parameters based on the network architecture:

1) For single hidden layer model: number of epochs = 1000, batch size = 150 and learning rate = 0.1, activation function of hidden layer = ReLU
2) For the models with more than one hidden layer: number of epochs = 100, batch size = 10 and learning rate = 0.001, activation function for all layers except last hidden layer = Sigmoid, last hidden layer activation function = ReLU
3) For all the model the linear activation function is used for the output layer

All the models are retrained 10 times and the best model is selected.

Radial Basis Function (RBF) neural networks

RBF neural networks another popular type of ANNs are known to be the universal approximators [50]. They have a compact topology compared to other neural networks [51,52]. It involves three layers – one input, one hidden and one output layer [20]. The input layer connects the source neurons to the only hidden layer of the network. The hidden layer applies the nonlinear transformation from the input space to hidden space [53] . Using the Gaussian kernel as the radial basis function, the output can be expressed as $y = \sum_i^n G_i w_i = \sum_{i=1}^n w_i \exp(-\|x - m_i\|^2 / 2\sigma_i^2)$ where x is the input, m_i and σ_i are the center and width of G_i [54]. The two hyper parameters that are to be optimized for this network are the number of hidden radial basis functions which the number of centers and the width of the Gaussian kernel (σ). The number of hidden radial basis functions relate to the accuracy of the network approximation [55,56]. After many trial and error tests, the optimal value of number of radial basis functions = 250 and $\sigma = 0.1$ are found.

Random Forest Regression

Random Forest regression proposed by [57] is an ensemble learning technique. It combines the performance of multiple decision trees for predicting the output variable [58,59]. It is based on the concept of bagging method in which the trees are constructed using a subset of random sample drawn with replacement from training data [60,61]. The random forest regression algorithm consists of following steps as explained by [62].

1. Draw a number of trees bootstrap samples from original data
2. For each bootstrap sample, grow unpruned regression tree for which at each node randomly sample the number of predictors and choose the best split among those variables
3. Predict the output for test data by averaging the predictions from all the trees

The study used the random forest algorithm by scikit-learn [63]. The version of scikit-learn is 0.21.1. Random forests require at least two hyper parameters to set, the number of trees ($n_{estimators}$) and the maximum number of features ($max_{features}$) [64]. The value of $n_{estimators}$ is fixed to 110 using the grid search ranging from 10 to 200 with an increment of 10. The default $max_{features}$ value is used as for the regression problems the empirical good default value is known to be number of features ($n_{features}$) [65].

Support Vector Machines for Regression

Support vector machines based on structural risk minimization principle proposed by [66] are learning machines for recognizing the subtle patterns in complex datasets [67,68]. Initially developed for solving the classification problems, they were extended to solve the regression problems for promising empirical performance [69]. The idea of the support vector regression is to use the kernel functions to map the initial data into the higher dimensional space such that the nonlinear patterns can be converted into a linear problem [70] . Support Vector Regression (SVR) performance is highly dependent on the selection of kernel functions [71]. Linear, Polynomial, Sigmoid and Radial Basis Function kernels are some of the most commonly used kernels [72]. For an efficient SVR model, the hyper parameters must be set carefully otherwise can be lead to over-fitting or under-fitting [73]. The ε-insensitive zone and regularization parameter C are the two main hyper parameters of the model [74,75]. The regularization parameter C penalizes any deviation than the ε. For higher the values of C, the penalty becomes more important and the SVR fits the data, whereas for smaller values of C, the penalty gets negligible and the SVR gets flat. In the same way, the higher the value of ε increases the margin and the SVR tends to become flat but with lower ε SVR tries to fit the data [76]. The study used SVR algorithm by scikit-learn. The optimal $C = 3000$ and $\varepsilon = 0.01$ are found using the grid search.

Gaussian Process Regression (GPR)

GPR can be viewed as an extension of standard regression models. It is one of the most popular non-parametric probabilistic models for estimating black-box functions [77]. ANNs with infinite hidden neurons are considered to converge to a Gaussian Process [78]. Gaussian Process is defined as a collection of random variables, any finite number of which have a joint Gaussian distribution. It is completely specified by its mean and covariance function that determine the smoothness and

variability of the function [79]. The GPR model is constructed as $y_i = f(x_i) + \varepsilon_i$ for $i = 1,2,..,n$. ε is assumed to be the additive independent identically distributed Gaussian noise with variance σ_n^2 . The predictive distribution of test data is obtained by conditioning on the training data and test inputs [80]. This study uses a Gaussian Process for Machine Learning (GPML) library in MATLAB for fitting the GPR model [81]. It uses the squared exponential kernel for computing the covariance. It uses maximum likelihood approach from GPML library to optimize all the hyper parameters associated with the covariance kernel and the Gaussian noise.

The following describes the methodology for selecting a training dataset and testing any of the ANN models' performance. A random sample is taken from the complete dataset for training. The sample size was determined using a statistical confidence interval of 99% and a margin of error of 2% as

$$S = \frac{z^2 p(1-p)/e^2}{1 + \frac{e^2 p(1-p)}{e^2 N}} \quad \text{Eq. 4}$$

where z is the z-score based on a 99% interval, e is the margin of error, N is the total sample size of the data set, and p is set to 50% because it is the most conservative case. The sample size was approximately 15% of the complete dataset. The same random sample is used to train each model. Once the optimization is complete, the trained model can be used to predict power for the remaining data. The power predictions are compared to the actual power output to evaluate any ANN model performance.

The study uses the following methods to evaluate all the ANN functions. Mean Absolute Error (MAE), calculated as

$$MAE = \frac{\sum_{i=1}^{n} |P_{act} - P_{est}|}{n} \quad \text{Eq. 5}$$

where P_{act} is the actual power, P_{est} is the estimated power using a model, and n is the number of samples used (the complete dataset).

The normalized MAE is given by Eq.6.

$$MAE_n = \frac{\sum_{i=1}^{n} |P_{act}/P_{rated} - P_{est}/P_{rated}|}{n} \quad \text{Eq. 6}$$

where P_{rated} is the rated power of the wind turbine.

The study uses the power prediction data to compute the energy production estimates. The energy production is the integration of power for a given time. From the power, Energy Production is

$$EP = \sum_{i=1}^{n} P(n) * t(n) \quad \text{Eq. 7}$$

where n is the number of samples (over one hour), $P(n)$ is the power determined from any ANN model or the actual power for a given sample, and $t(n)$ is the measured time of the averaged sample in hours (1/6 for the 10-minute data). Therefore, the energy production is determined for each hour for both the ANN models' prediction and the actual energy produced. The study compares the energy production estimates of the ANN models with other correction models for each of the eight months of wind turbine data collected. The error is determined as

$$EP_{error} = \frac{\sum_{i=1}^{n} \frac{EP_{act} - EP_{est}}{EP_{act}}}{n} \quad \text{Eq. 8}$$

where n is the number of samples in a month, EP_{act} is the actual energy production, and EP_{est} is the estimated energy production from the model.

2.2. Kirkwood Iowa Dataset

The dataset comes from the Kirkwood/University of Iowa Wind Data Project [82]. The data is from January to August of 2015. The project includes one Clipper 2.5 MW wind turbine with Supervisory control and data acquisition (SCADA) from the wind turbine and a meteorological tower 900 m SW of the wind turbine as shown in Figure 2a. The SCADA data includes the power production of the wind turbine, air density, the hub height wind speed, and the standard deviation of the hub height wind speed. The meteorological tower contains measurements described in Table 1. These include wind speed, temperature, barometric pressure, and air temperature at multiple heights. Figure 2b shows the distribution of the Kirkwood Iowa Data wind speeds measured at hub height. 63% of the data occurs between 5 and 10 m/s. Only 13% occurs above 10 m/s and 24% occurs below 5 m/s.

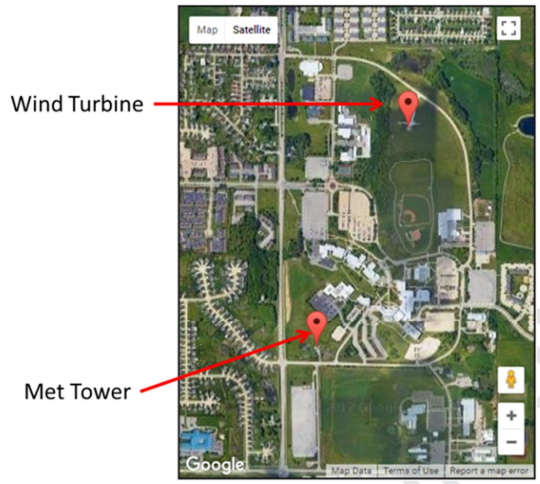
The goal of implementing the ANN models is to incorporate atmospheric stability into the power predictions. Therefore, derived variables that quantify atmospheric stability are used as inputs. The study uses turbulence intensity, Richardson Number, and wind shear as variables that describe atmospheric stability and described each below. Turbulence intensity, shown in Eq.8, describes the amount of turbulence in the atmosphere and relates directly to the atmospheric regime [4,14]. The turbulence intensity is determined from the standard deviation of the horizontal wind speed as a ratio of the total horizontal wind speed. The meteorological data are used to determine the Richardson Number calculated in Eq. 9. The Richardson Number is a dimensionless ratio of the buoyancy and shear in a flow [83]. This ratio aids in classifying a flow as buoyancy dominated or shear dominated flow corresponding to a specific stability regime [12]. For the Richardson Number, the velocity and temperature measurements at 6.6 m and 106.6 m are used. Finally, wind shear can also be used to determine stability

[6,12]. The met tower data determines the wind-shear exponent α, known as the wind shear, by fitting the velocity profile of all six wind speeds to Eq. 10. Table 4 defines how the different parameters calculated in Eq. 8-10 are used to determine atmospheric stability.

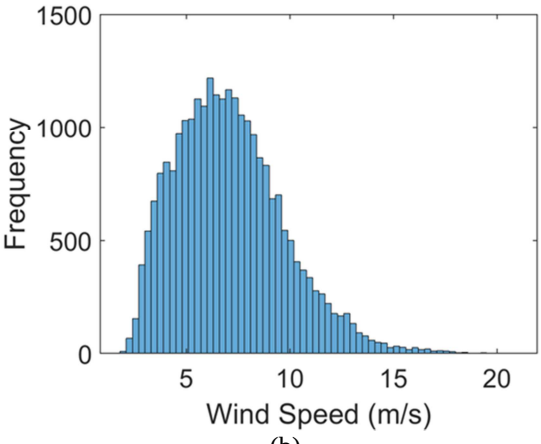
$$U = \beta z^{\alpha} \quad \text{Eq.8}$$

$$Ri_s = \frac{g}{\overline{\theta_v}} \frac{d\theta_v/dz}{(d\bar{U}/dz)^2} \quad \text{Eq.9}$$

$$TI = \frac{\sigma_u}{U} \quad \text{Eq.10}$$



(a)



(b)

Figure 2. (a) Wind Turbine and Meteorological Tower layout (b) histogram of wind speeds for the Kirkwood Iowa data.

Table 1 Meteorological Tower sensor description.

Parameter	Device	Height (m)
Wind Speed	NRG Cup Anemometers	6.6, 10.15, 33.5, 55.8, 82.2, 106.7
Wind Direction	NRG Cup Vane	6.6, 10.15, 33.5, 55.8, 82.2, 106.7
Barometric Pressure	Not Available	6.6, 106.7
Air Temperature	Not Available	6.6, 106.7

3. Results

3.1. Role of input parameters in model performance

The next step was to focus on determining the number of input parameters to be included in the FFBP ANN model to obtain the best performing power curves. Table 2 describes the FFBP ANN models used for the comparisons. The study investigated two-parameter, three-parameter, and five-parameter models (the number of inputs) as shown in Table 2.

Table 2. Description of the FFBP ANN Models used for comparison.

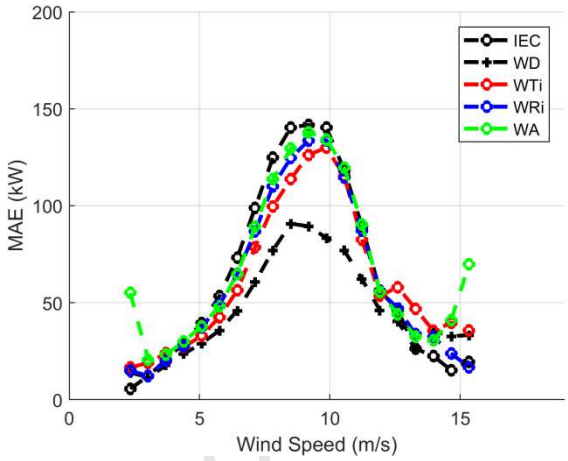
Number of Parameters	Parameters	ID
2	I. Wind Speed II. Density	WD
2	I. Wind Speed II. Turbulence Intensity	WTi
2	I. Wind Speed II. Richardson Number	WRi
2	I. Wind Speed II. α	WA
3	I. Wind Speed II. Density III. Turbulence Intensity	WDTi
3	I. Wind Speed II. Density III. Richardson Number	WDRi
3	I. Wind Speed II. Density III. α	WDA
5	I. Wind Speed II. Density III. Turbulence Intensity IV. Richardson Number V. α	WDTiRiA

The IEC standard is used as the baseline of performance. The study investigates the atmospheric inputs by using a single layer, 10 neuron, FFBP ANN. Figure 3a shows the MAE for the two-parameter cases WD, WTi, WRi and WA as well as the MAE for the IEC. The figure shows the MAE is less for wind speeds below 3 m/s (the cut in speed), it increases to a peak value between 8-10 m/s, and then it decreases to a minimum at 15 m/s (rated wind speed). The peak MAE for the IEC, WD, WTi, WRi, and WA is 141.7, 85.6, 130.0, 133.8, and 138.0 kW, respectively. The WA case had a larger MAE at 2 m/s and 15 m/s than the other models, which was 55.1 and 75.0 , respectively. No other case showed similar behavior. The total error (in addition to the peak MAE) can be quantified

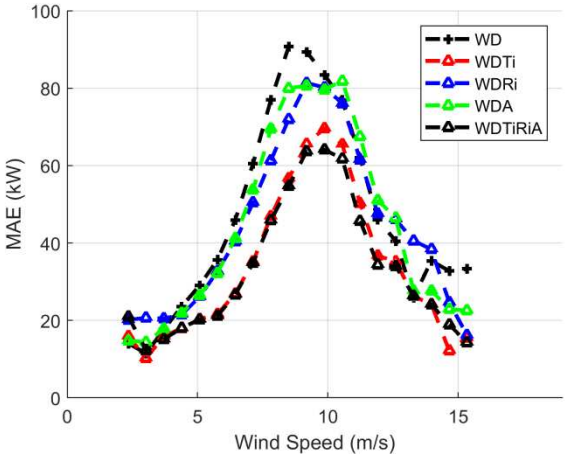
with a weighted average MAE, where the weights are determined based on the number of samples in each bin as shown in Figure 2b. All two-parameter FFBP ANNs showed an improvement over the IEC standard, which has an average MAE of 78.5 kW. The two-parameter FFBP ANNs WTi, WRi, WA had a total MAE of 67.3, 72.2, and 74.9 kW respectively. The improvement, compared to the IEC, was 13.8%, 8.1%, and 4.6% respectively. The two-parameter FFBP ANN WD showed the most significant reduction in MAE with an average MAE of 48.0 kW, or 39% improved over the IEC. The two-parameter FFBP ANN WD reduced the MAE by 50% at a wind speed of 9 m/s when compared to the IEC. The reduction below 5 m/s and above 12 m/s is not as significant and it is 20% and 6% respectively. The results indicate that the WD FFBP ANN case has the highest reduction in MAE compared to all other two-parameter FFBP ANNs.

Figure 3b compares the WD FFBP ANN to the three-parameter and five-parameter FFBP ANNs. The peak MAE for the WDTi, WDRi, WDA were 69.6, 81.3 and 83.4 kW respectively. The WDRi case has a larger MAE below 3 m/s and between 12-14 m/s. The total MAE (determined by a weighted average based on the number of samples in each bin as shown in Figure 2b for cases WDTi, WDRi, WDA were 35.3, 46.4, and 47.9 kW respectively. The three-parameter FFBP ANN WDTi showed the most significant improvement over the two-parameter FFBP ANN WD. The WDTi case reduced the peak MAE by 50% when compared to the IEC standard. The WDTi reduced the average MAE by 55% when compared to the IEC standard. The three-parameter FFBP ANN WDTi also reduced the MAE below 3 m/s and above 12 m/s by 38% and 15% respectively when compared to the IEC.

Also shown in Figure 3b is the five-parameter case WDTiRiA. WDTiRiA had a peak MAE of 64.1 kW at 9.8 m/s. The MAE at 14 m/s was 18.9 kW, compared to 12.2 kW of the WDTi case. The average MAE (determined by a weighted average based on the number of samples in each bin as shown in Figure 2b) for the five-parameter FFBP ANN was 34.3 kW, a reduction of 2% from the WDTi case. Overall, the WDTi had the lowest MAE of all the three-parameter FFBP ANNs. The five-parameter FFBP ANN WDTiRiA showed the lowest average MAE with a 2.75% improvement over the WDTi case. The results show how including additional atmospheric parameters as inputs can improve FFBP ANN.



(a)



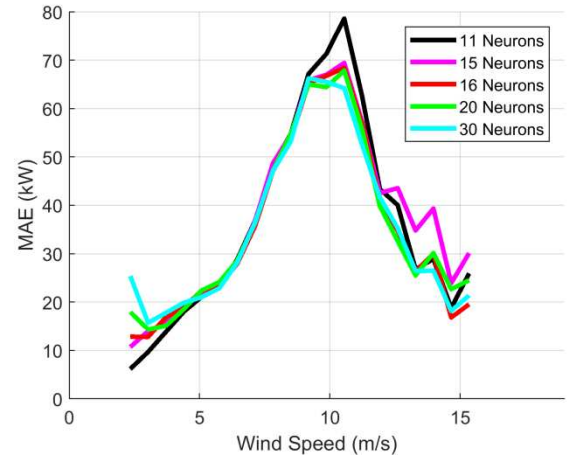
(b)

Figure 3. Mean absolute error for the (a) IEC, two-parameter: Wind Speed and Density (WD), Wind Speed and Turbulence Intensity (WTi), Wind Speed and Richardson Number (WRi), Wind Speed and α (WA); and (b) two-parameter: Wind Speed and Density (WD); three-parameter: Wind Speed, Density and Turbulence Intensity (WDTi), Wind Speed, Density and Richardson Number (WDRi), Wind Speed, Density and α (WDA); Five parameter (WDTiRiA).

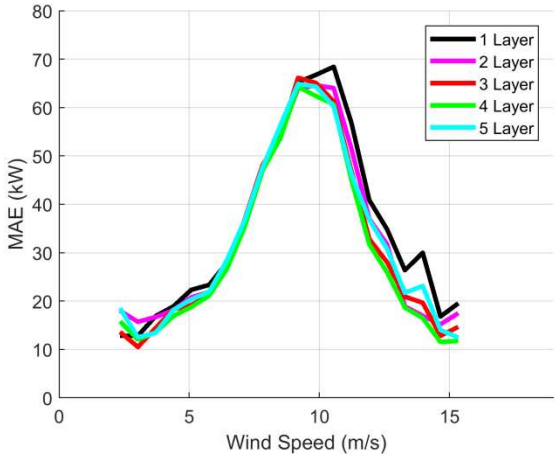
The influence of the FFBP ANN hyper-parameters were studied to optimize the FFBP ANN model using a five-parameter input (WDTiRiA) because it performed the best. Figure 4 a shows the influence of the number neurons in a single layer FFBP ANN. The major differences occur at the peak MAE, and at wind speeds below 5 *m/s.* The peak MAE is 78.6, 69.4, 68.4, 67.8, and 66.2 kW for the 11, 15, 16, 20, and 30 Neuron models respectively. The peak MAE begins to get larger after 50 neurons (not shown in the figure). The additional neurons reduce the peak MAE, but not the average MAE. The average MAE is 36.2, 37.3, 34.9, 35.4, and 35.3 kW respectively. The average MAE reaches a minimum at 16

neurons. The average MAE increases with additional neurons beyond 60 because the MAE increase below 5 *m/s.* The increase in MAE comes from overfitting the data at lower wind speeds.

Next, the study investigates the influence of the number of hidden layers on the MAE. The number of neurons in each layer was optimized, as described in the single layer example shown in the paragraph above. The largest differences occur for wind speeds above 10 *m/s.* The peak MAE for each layer is 68.4, 64.2, 66.2, 64.2, and 64.8 kW for the 1 layer (16 neurons), 2 layer (50, 20 neurons), 3 (100, 50, 20 neurons) layer, 4 layer (100,50,20,10 neurons), and 5 layer FFBP ANN (200,100,50,30,11) respectively. The average MAE is the 34.8, 32.8, 31.5, 30.5, and 32.4 kW respectively. Beyond 4 layers, additional layers tend to over fit the data between 12 and 15 *m/s*. The optimal FFBP ANN model for the five-parameter data is the 4 layer FFBP ANN with 100, 50, 20, 10 neurons in first, second, third and fourth hidden layers respectively. The model consists of 5 input neurons in the input layer and one neuron in the output layer. The network topology for the 4 layer model is 5-100-50-20-10-1.



(a)



(b)

Figure 4. Mean absolute error for the (a) Single FFBP ANN with layer 11 Neurons, 15 Neurons, 16, Neurons, 20 Neurons, and 30 Neurons (b) Optimized FFBP ANN with 1 layer (16 neurons), 2 layer (50, 20 neurons), 3 (100, 50, 20 neurons) layer, 4 layer (100,50,20,10 neurons), and 5 layer FFBP ANN (200,100,50,30,11).

The study compared the 4 Layer FFBP ANN with other Machine learning models to quantify the performance of predicting the wind turbine power. Figure 5 shows the MAE for each 4 layer FFBP ANN, RBFN, RF, SVR, and GP. The peak MAE is 64.3, 77.7, 80.7, 66.1 and 66.2 kW respectively. The average MAE is 30.5, 45.8, 35.8, 38.8, and 41.4 kW respectively. The RBFN and RF models perform the best at the lowest wind speeds, but also have the highest peak MAE. The GP and SVR models have a peak MAE that is only moderately higher than the 4 layer FFBP ANN but also have a large MAE above 12 *m/s* where there is significantly less data (as shown in Figure 2b). The 4 layer FFBP ANN outperforms the other models and has a smaller average MAE, which shows the robustness of the FFBP ANN model for wind turbine power prediction.

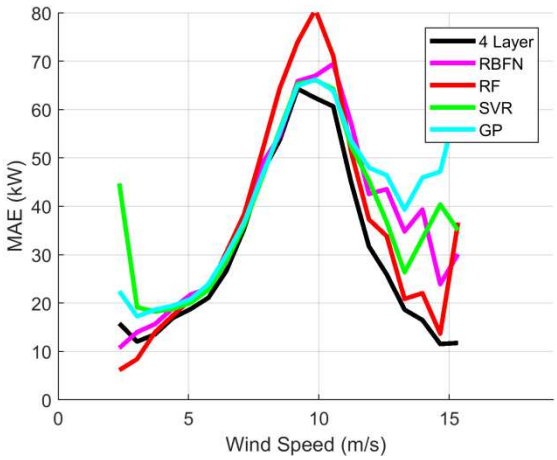


Figure 5. Mean absolute error for the 4 Layer FFBP ANN, RBFN, RF, SVR, and GP.

Next, in Figure 6, the 4 layer FFBP ANN is compared to other improved power curve models used in literature: These include equivalent wind speed which uses the wind speeds at every height of the meteorological tower [11], a multi-power curve model which uses wind speed and turbulence intensity filters [4], and the density correction method [84].

Figure 6 shows the MAE of each of the models. The peak MAE for the IEC, density correction, turbulence intensity filter, equivalent wind speed, and the 4 layer FFBP ANN are 141.7, 98.3, 135, 110.0 and 64.3 kW respectively. The peak MAE occurs at different wind speeds, at 9.2, 8.5, 9.8, 9.2, and 9.2 m/s respectively.

Those that include turbulence intensity shift the peak MAE to higher wind speeds. The average MAE (determined by a weighted average based on the number of samples in each bin as shown in Figure 2b was 78.5, 58.4, 69, 59.3, and 30.5 kW for the IEC, Density correction, Turbulence Intensity, Equivalent Wind Speed, and the 4 layer FFBP ANN respectively. The 4 layer FFBP ANN showed significant improvement over all other models. The FFBP ANN model reduced the peak MAE by 35% when compared to the density correction (the next closest model) and reduced the total MAE by 48%. The 4 layer FFBP ANN reduced the peak MAE by 52% and 42% and the total MAE by 65% and 44% for the turbulence filter and equivalent wind speed respectively. The results show the importance of including both air density and atmospheric effects into power prediction and further validates the methodology of using FFBP ANNs in wind power prediction.

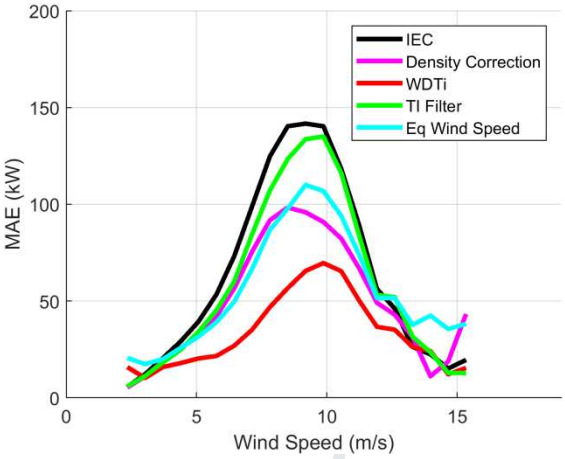


Figure 6. Mean absolute error for the (a) IEC, two-parameter: Wind Speed

The energy production is calculated by integrating the power predictions over time as shown in Eq. 7. Table 3 shows the error, as determined by Eq. 8, of the monthly energy production estimates (compared to the actual energy production) for the IEC model, density correction model, and the 4 layer FFBP ANN model. For the IEC standard, the highest error is 20% and is recorded for August, and the average error over the entire 8 months is 3.7%. The density correction improves the energy predictions with a maximum error of 14.1% in the month of August and an average error of 1.9%. The average errors for the IEC and density correction were -3.7% and -1.9%, which are near the expected 2-5% in the literature [85]. The 4 layer FFBP ANN further enhances the energy production estimates by including atmospheric effects using turbulence intensity. The FFBP ANN model showed a max error of 1.09% during February and an error of 0% in May. The 4 layer FFBP ANN had an average error of 0.4%. The 4 layer FFBP ANN model reduced the averaged energy production error to 0.4%.

The FFBP ANN also performed more consistently from month to month. Both the IEC and air density correction underpredict from January to March but overpredict during the April to August Months. The month to month variation in percent error is much smaller for the 4 layer FFBP ANN, as it remains at or below 1.09%. The month to month variations might come from the variation in atmospheric conditions. For the months Jan-Mar, the temperatures are cooler causing a lower average air density. Therefore, the actual energy production will be higher than the estimated value for the IEC method and the error is positive. For the months Apr-Aug the opposite is true, and the IEC error is positive. The improved predictions show the ability of machine learning to aid in future wind energy estimates and for estimating an Annual Energy Production (AEP) for future wind turbines and further validates the FFBP ANN methodology. To better understand the cause of monthly variations in the errors of energy production, an analysis of the atmospheric stability is conducted next.

Table 3. Energy Production Error, compared to actual energy production in percent, of different power curve models.

Month	IEC	Density Correction	4 Layer FFBP ANN
Jan	6.8%	2.4%	-0.46%
Feb	10.3%	5.1%	1.09%
Mar	4.9%	3.0%	0.67%
Apr	-1.0%	0.2%	-0.1%
May	-6.3%	-2.1%	0.0%
Jun	-11.7%	-4.6%	0.39%
Jul	-12%	-5.3%	0.38%
Aug	-20%	-14.1%	0.96%
Total	-3.7%	-1.9%	0.4%

3.2. Role of atmospheric stability on FFBP ANN model efficacy

To understand the effect of atmospheric stability on the error in the FFBP ANN model, the study performed the following analysis. The results between the convective (negative Richardson Number) and stable (positive Richardson Number) atmospheric stability conditions for the 4 layer FFBP ANN were compared. 37% of the data had a Richardson number above zero. Figure 7 shows the results. The convective conditions had a smaller peak MAE, which was 61 kW compared to 75 kW. The MAE for the convective conditions was smaller between the wind speeds of 7-11 m/s. This might result because the convective atmosphere is more mixed. Therefore, there is less of a variation in wind speed and density throughout the height of the wind turbine. The MAE during the convective conditions is larger for wind speeds above 12 m/s (35 kW

compared to 20 kW), which is the near rated wind speed for the wind turbine. Although, the wind speeds above 12 m/s only account for 5% of the total wind speed samples. This might be due to high turbulence causing the wind speed to fluctuate above and below the rated wind speed.

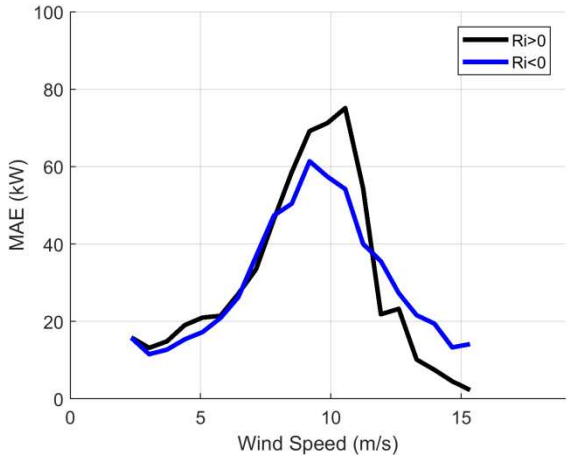


Figure 7. Mean Absolute Error for convective atmospheric stability (negative Richardson Number) and stable atmospheric stability (positive Richardson Number) conditions.

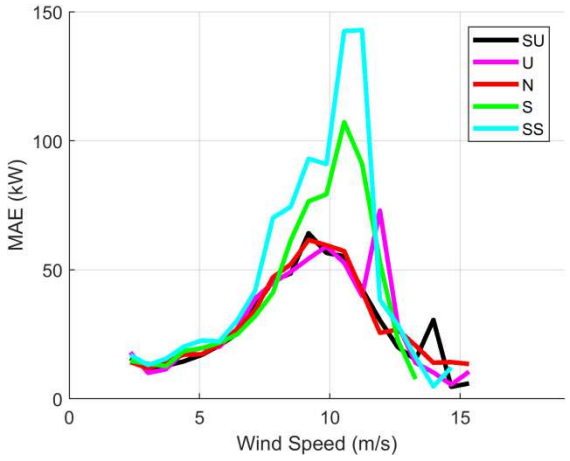
To group data by atmospheric stability conditions, the the classifications shown in Table 4 were used. Table 4 shows the five stability regimes that were obtained using a range of Richardson Numbers and turbulence intensity values [25,26]. The classifications are used to investigate further how the atmospheric stability affects the MAE and are Strongly Unstable (SU), Unstable (U), Neutral (N), Stable (S), and Strongly Stable (SS) conditions.

Table 4 Definition of stability regimes for different parameters.

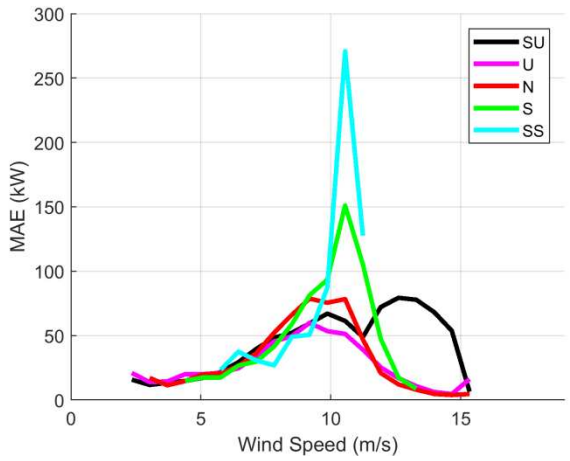
Stability	**Name**	***Ri*** [26]	***TI*** [25]
Strongly Unstable	SU	<-0.2	>0.2
Unstable	U	-0.2 to -0.1	0.13 to 0.2
Netrual	N	-0.1 to 0.1	0.1 to 0.13
Stable	S	0.1 to 0.25	0.08 to 0.1
Strongly Stable	SS	>0.25	<0.08

An analysis is conducted by using one of the two criteria, Richardson Number or turbulence intensity, to classify the atmospheric stability as shown in Figure 8. As seen in Figure 8a, when using the Richardson Number criteria, the strongly stable SS regime shows the largest peak MAE of 142 kW at 10.5 m/s. The SS case begins to deviate to the peak MAE at 7 m/s. The peak MAEs are 107, 61.5, 73.0, and 64.2 kW and occur at 10.5 m/s, 11.9 m/s, 9.2 m/s, and 10.5 m/s for the S, U, N, SU respectively. The peak MAE is two times larger for SS than for N.

The turbulence intensity criteria is used to group the data in Figure 8b. Using the turbulence intensity criteria to define the regimes shows different results. The unstable regime showed the lowest peak MAE. The SS group had the largest MAE of 270 kW at a wind speed of 10.5 m/s. The other peak MAEs were 151, 78.7, 59.9, 79.3 kW and occurred at 10.5, 9.8, 9.2, and 12.6 m/s for the S, N, U, and SU cases respectively. The N regime showed higher MAE for wind speeds between 7 m/s and 12 m/s with an increase of 20% compared to the unstable regime. The SU regime showed higher MAE for wind speeds 12-15 m/s that was five times larger than the U regime. This matches the result in Figure 7 for the convective MAE. These high wind speeds only represent 5% of the total samples as shown in Figure 2b. Both the S and SS regimes showed higher MAE for wind speeds between 7 m/s and 12 m/s with the SS being eight times larger and the stable being three times larger than the U regime. It should be noted that using turbulence intensity to determine stability showed S and SS regimes made up 5% of the data whereas the Richardson Number showed 23% of the data in the SS and SS regimes. Therefore, the larger error might come from the small number of statistical values for training the FFBP ANN. The peak MAE for both S and SS using either method is much larger than the other regimes. Overall, the state of the atmosphere has a significant effect on the performance of the 4 layer FFBP ANN, in particular, SS and S conditions produce the largest MAE between 7-12 m/s.



(a)



(b)

Figure 8. Comparison of MAE for different stability regimes using a) Richardson Number, and b) turbulence intensity.

4. Discussion

In summary, the 4 layer FFBP ANN with wind speed, air density, turbulence intensity, Richardson number, and wind shear was selected to compare with other power curve models due to its ability to learn the complex relationship between the inputs and the output. The 4 layer FFBP ANN showed a reduction in peak MAE and average MAE compared to other models that correct for atmospheric stability. The FFBP ANN reduced the peak and average power curve MAE by 35% when compared to the air density correction model, which was the next closest. The reduction in MAE from the 4 layer FFBP ANN power curve led to a reduction in the error of energy production estimates. The 4 layer FFBP ANN had an average energy production error of 0.4% for the nine months while the IEC was -3.7% and the air density correction was -1.9%. Similarly, Manobel et al. [86] used Gaussian Processes to reduce the AEP of a wind turbine from 1.98% to 0.71% although the current study focuses on incorporating atmospheric stability as inputs into the 4 layer FFBP ANN. The study showed how atmospheric stability parameters can be inputs into FFBP ANNs to improve wind power forecasting.

Ata [87] described some limitations of FFBP ANNs which include over training, extrapolation errors, and network optimization. This work did not inspect potential solutions for over-fitting, such as inserting dropout rates in the analysis, since it would go beyond the scope of our research. The hyper-parameters for the artificial neural network were optimized to mitigate these limitations. The study used a full factorial design to test the design space of the hyper parameters. The study found that for a single layer FFBP ANN, 16 neurons produced the optimal MAE. Adding additional layers creates over training (or over fitting) especially at the high and the low wind speeds. This finding suggests that the FFBP ANN could further reduce the power prediction error by using hybrid combinations of machine learning models based on wind speed and stability [88]. Additionally, the study investigated the performance based on the number of hidden layers. The study found the 4 layer FFBP ANN had the lowest average MAE. Again, adding more layers creates over fitting for high wind speeds (above 12 *m/s*) where there are a smaller number of samples.

The study investigated the robustness of the FFBP ANN by comparing other machine learning techniques such as Radial based function (RBF), Random Forest Regression (RF), Support Vector Regression (SVR) and Gaussian Process Regression (GP). The 4 layer FFBP ANN had a smaller average MAE than the other tested models. Specifically, the 4-layer FFBP ANN reduced the MAE above 12 m/s where there are a smaller number of samples to draw into the training set. Li and Shi [45] performed a similar comparison for wind speed estimation and showed that the best performing algorithm depended on the makeup of the data set tested. Implementing this methodology on more data sets will help to improve the robustness of the model. Additionally, the study investigated the upper and lower 95% confidence bounds of MAE for the 4 layer network, shown in Table 5 of Appendix A. The Average MAE for the 4 layer network was 30.45 *kW*. The upper and lower bounds were 30.81 and 30.09 *kW* respectively. The small deviation in the upper and lower bounds also shows the robustness of the 4 layer model.

The current study used FFBP ANNs to develop a five-parameter power curve using wind speed, density, and turbulence intensity, Richardson Number, and Wind shear (measurements of atmospheric stability). The novelty of the approach was to select specific atmospheric inputs shown to affect wind turbines from previous literature. The novelty led to an improved understanding on the atmospheric effects on wind turbines. Specifically, turbulence intensity did not have a substantial impact as an input until after wind density was also included meaning:

- Density should be included as the second parameter in power prediction models
- Not incorporating density in atmospheric studies could hide the true atmospheric effects
- Other atmospheric studies might benefit by implementing a density correction before determining atmospheric effects on wind power curves

The easiest implementation of the above methodology would be for manufacturers to develop the FFBP ANNs and give the resulting model to operators to use as a "black-box". The ANN model could replace power curves that manufacturers already develop. Once the model is

created, there is little computational power required (the current results were performed <1s on a single processor). Since the ANNs were built using open source software, the only additional costs would be for additional sensors used to determine the Richardson number. Although, the study showed that the three-parameter ANN (not including Richardson number as an input), using wind speed, density, and turbulence intensity, also showed improved performance. These three parameters are already used when manufactures follow the IEC standard, and therefore, using a three-parameter FFBP ANN would not increase the cost to above the status quo. The above methodology is computational inexpensive, can easily replace current power curves (already have the measurements), was developed with open-source technologies, and can also be performed with simple toolkits (I.E. Matlab ANN Toolkit) which make it a novel, practical, and cost effective solution to improve accuracy.

The 4-layer FFBP ANN model improved power predictions by 48% (total MAE) over the wind density correction and 59% above the IEC standard used in industry. The three-parameter FFBP ANN directly reduced the energy production estimate's errors. The reduced power prediction errors reduced the energy estimate errors from 3.7% to 0.4%. These results show improvements over other parametric and not parametric models that an energy production error of 0.6% to 2.11% [86]. Reducing energy estimates has potential cost savings by lowering inefficiencies, curtailment, shortfalls, and load shedding. Lew et al. [89] described the economic value of improving wind power predictions. Their study showed that with a 10% improvement in wind power predictions would, assuming 14% wind energy penetration, would incur a savings of $140M. As penetration increases, these savings become even more significant.

The performance of the FFBP ANN was dependent on the stability state of the atmosphere. When the Richardson Number criteria were used for classification of atmospheric stability, the strongly stable data showed the highest peak MAE of 143 kW and the stable conditions showed the second highest peak MAE of 107 kW. The unstable conditions showed a peak of 72.0 kW, but the peak occurred at a higher wind speed of 11.9 m/s. This wind speed was higher than the peak MAE for the other categories, which fell between 9.2-10.5 m/s. When turbulence intensity was used as the classification criteria, the strongly stable conditions also produced the highest MAE, but it was 271 kW. The stable conditions had the second highest MAE of 151 kW. The strongly unstable MAE had a peak of 79.3 kW that occurred at 12.6 m/s, which is a higher wind speed than the other peak MAEs. The results suggest that the FFBP ANN performance is dependent on the atmospheric stability regime, although some of this error might be from the smaller statistical number of occurrences of certain stability regimes. Therefore, the three-parameter FFBP ANN would not be trained with as many samples of these cases.

5. Conclusions

The study successfully implemented a 4 layer Feed Forward Back Propagation (FFBP) Artificial Neural Networks (ANN) to develop an improved power curve for site-specific power curves and energy predictions. The study compared a variety of different inputs for the FFBP ANN power curve. The five-parameter FFBP ANN (with wind speed, air density, turbulence intensity, Richardson Number, and wind shear) showed the lowest MAE. The FFBP ANN improved the performance of the power curve over other methods that use atmospheric stability or density to correct the power curve. The FFBP ANN reduced the total Mean Absolute Error (MAE) by 48% over the wind density correction used and by 59% compared to the IEC method. Using the FFBP ANN also reduced the error in estimated energy production to 0.4% from to 2-4% from the IEC and density correction.

The 4 layer FFBP ANN also reduced the MAE over other machine learning algorithms tested, specifically Radial based function (RBF), Random Forest Regression (RF), Support Vector Regression (SVR) and Gaussian Process Regression (GP). This work successfully employed the FFBP NN to forecast a time series. In future work, it would be interesting to investigate the accuracy between other time-series efficient networks, such as the FFBP NN versus Recurrent Neural Network alternatives such as the long-short term memory (LSTM) and the GRU architectures. The 4 layer FFBP ANN had a smaller average MAE compared to the other methods tested, and also reduced the MAE above 12 *m/s* where there was less data to draw into the training set. The 4 layer model also limited over fitting compared to the 5 layer model.

The 4 layer FFBP ANN performance was dependent on atmospheric stability. At low wind speeds of less than 7 m/s there is not a significant difference in MAE between the different atmospheric stability regimes. The MAE was lower in unstable conditions for wind speeds between 7 -11 m/s using either the Richardson Number or turbulence intensity criterion. The stable and strongly stable regimes had a large MAE between 7-11 m/s for both the criterion. The strongly unstable regime, when using the turbulence intensity criteria, had a higher MAE at high wind-speeds above 11 m/s. When comparing FFBP ANN performance by separating into five stability regimes, strongly stable conditions had the largest peak MAE for both criteria, but it also has the smallest statistical samples to perform the analysis and train the FFBP ANN.

Acknowledgements

The Authors acknowledge the work of those that set up and performed the the Kirkrood/University of Iowa Wind Data Project and for making the data publically accessible, TACC for use of the super computing, and Dr. Max Kilger who provided guidance on the use of the ANNs.

The authors declare no conflict of interest and no outside funding used for this article.

REFERENCES

1. Wiser, R.; Bolinger, M. 2017 Wind Technologies Market Report 2018.
2. Commission, I.E. IEC 61400-1: Wind turbines part 1: Design requirements. *Int. Electrotech. Comm.* **2005**.
3. Wharton, S.; Lundquist, J.K. Atmospheric stability affects wind turbine power collection. *Environ. Res. Lett.* **2012**, *7*, 014005.
4. St Martin, C.M.; Lundquist, J.K.; Clifton, A.; Poulos, G.S.; Schreck, S.J. Wind turbine power production and annual energy production depend on atmospheric stability and turbulence. *Wind Energy Sci.* **2016**, *1*, 221.
5. van den Berg, G.P. Wind turbine power and sound in relation to atmospheric stability. *Wind Energy* **2008**, *11*, 151–169.
6. Clifton, A.; Kilcher, L.; Lundquist, J.K.; Fleming, P. Using machine learning to predict wind turbine power output. *Environ. Res. Lett.* **2013**, *8*, 024009.
7. Pelletier, F.; Masson, C.; Tahan, A. Wind turbine power curve modelling using artificial neural network. *Renew. Energy* **2016**, *89*, 207–214.
8. Li, S.; Wunsch, D.C.; O'Hair, E.; Giesselmann, M.G. Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation. *J. Sol. Energy Eng.* **2001**, *123*, 327–332.
9. Ouyang, T.; Kusiak, A.; He, Y. Modeling wind-turbine power curve: A data partitioning and mining approach. *Renew. Energy* **2017**, *102*, 1–8.
10. Schlechtingen, M.; Santos, I.F.; Achiche, S. Using data-mining approaches for wind turbine power curve monitoring: a comparative study. *IEEE Trans. Sustain. Energy* **2013**, *4*, 671–679.
11. Wagner, R.; Courtney, M.; Gottschall, J.; Lindelöw-Marsden, P. Accounting for the speed shear in wind turbine power performance measurement. *Wind Energy* **2011**, *14*, 993–1004.
12. Clifton, A.; Schreck, S.; Scott, G.; Kelley, N.; Lundquist, J.K. Turbine Inflow Characterization at the National Wind Technology Center. *J. Sol. Energy Eng.* **2013**, *135*, 031017-031017–11.
13. Elliott, D.L. (Pacific N.L.; Cadogan, J.B. (usdoe *Effects of Wind Shear and Turbulence on Wind Turbine Power Curves*; Pacific Northwest Lab., Richland, WA (USA), 1990;
14. Hansen, K.S.; Barthelmie, R.J.; Jensen, L.E.; Sommer, A. The impact of turbulence intensity and atmospheric stability on power deficits due to wind turbine wakes at Horns Rev wind farm. *Wind Energy* **2012**, *15*, 183–196.
15. Oh, H.; Kim, B. Comparison and verification of the deviation between guaranteed and measured wind turbine power performance in complex terrain. *Energy* **2015**, *85*, 23–29.
16. Stull, R.B. *An introduction to boundary layer meteorology*; Springer Science & Business Media, 2012; Vol. 13;.
17. Bhaganagar, K.; Debnath, M. The effects of mean atmospheric forcings of the stable atmospheric boundary layer on wind turbine wake. *J. Renew. Sustain. Energy* **2015**, *7*, 013124.
18. Beare, R.J.; Macvean, M.K.; Holtslag, A.A.; Cuxart, J.; Esau, I.; Golaz, J.-C.; Jimenez, M.A.; Khairoutdinov, M.; Kosovic, B.; Lewellen, D.; et al. An intercomparison of large-eddy simulations of the stable boundary layer. *Bound.-Layer Meteorol.* **2006**, *118*, 247–272.
19. Paulson, C.A. The mathematical representation of wind speed and temperature profiles in the unstable atmospheric surface layer. *J. Appl. Meteorol.* **1970**, *9*, 857–861.
20. Archer, C.L.; Colle, B.A.; Veron, D.L.; Veron, F.; Sienkiewicz, M.J. On the predominance of unstable atmospheric conditions in the marine boundary layer offshore of the US northeastern coast. *J. Geophys. Res. Atmospheres* **2016**, *121*, 8869–8885.
21. Carson, D.J. The development of a dry inversion-capped convectively unstable boundary layer. *Q. J. R. Meteorol. Soc.* **1973**, *99*, 450–467.
22. Porté-Agel, F.; Meneveau, C.; Parlange, M.B. A scale-dependent dynamic model for large-eddy simulation: application to a neutral atmospheric boundary layer. *J. Fluid Mech.* **2000**, *415*, 261–284.
23. Blackadar, A.K.; Tennekes, H. Asymptotic similarity in neutral barotropic planetary boundary layers. *J. Atmospheric Sci.* **1968**, *25*, 1015–1020.
24. Ghaisas, N.S.; Archer, C.L.; Xie, S.; Wu, S.; Maguire, E. Evaluation of layout and atmospheric stability effects in wind farms using large-eddy simulation. *Wind Energy* **2017**.
25. Newman, J.F.; Klein, P.M. The impacts of atmospheric stability on the accuracy of wind speed extrapolation methods. *Resources* **2014**, *3*, 81–105.
26. Wharton, S.; Lundquist, J.K. Atmospheric stability impacts on power curves of Tall wind turbines-an Analysis of a West Coast North American wind farm. *Rep. Lawrence Livermore Natl. Lab. LLNL-TR-424425* **2010**.
27. Xie, S.; Archer, C.L. A Numerical Study of Wind-Turbine Wakes for Three Atmospheric Stability Conditions.

Bound.-Layer Meteorol. **2017**, 1–26.
28. Nielson, J.; Bhaganagar, K. Capturing Day-to-Day Diurnal Variations in Stability in the Convective Atmospheric Boundary Layer Using Large Eddy Simulation. *Open Atmospheric Sci. J.* **2018**.
29. Nielson, J.; Bhaganagar, K. Using field data–based large eddy simulation to understand role of atmospheric stability on energy production of wind turbines. *Wind Eng.* **2019**.
30. Lek, S.; Guégan, J.-F. J E. modelling Artificial neural networks as a tool in ecological modelling, an introduction. **1999**, *120*, 65–73.
31. Gardner, M.W.; Dorling, S. J A. environment Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. **1998**, *32*, 2627–2636.
32. Michael, N. Artificial intelligence a guide to intelligent systems. **2005**.
33. Ekonomou, L. J E. Greek long-term energy consumption prediction using artificial neural networks. **2010**, *35*, 512–517.
34. Jain, A.K.; Mao, J.; Mohiuddin, K.M. Artificial neural networks: A tutorial. *Computer* **1996**, *29*, 31–44.
35. Chollet, F. Keras. **2015**.
36. Burkov, A. The Hundred-Page Machine Learning Book. **2019**.
37. Ruder, S. J arXiv preprint arXiv:.04747 An overview of gradient descent optimization algorithms. **2016**.
38. Goodall, N.J. Machine ethics and automated vehicles. In *Road vehicle automation*; Springer, 2014; pp. 93–102.
39. Kingma, D.P.; Ba, J. J arXiv preprint arXiv:. Adam: A method for stochastic optimization. **2014**.
40. Duchi, J.; Hazan, E.; Singer, Y. J J. of M.L.R. Adaptive subgradient methods for online learning and stochastic optimization. **2011**, *12*, 2121–2159.
41. Tieleman, T.; Hinton, G. J C.N. networks for machine learning Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. **2012**, *4*, 26–31.
42. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep learning*; MIT press, 2016; ISBN 0-262-33737-1.
43. Nielsen, M.A. *Neural networks and deep learning*; Determination press San Francisco, CA, USA:, 2015; Vol. 25;.
44. Montgomery, D.C. J N.Y. Design of experiments. **1995**, 225–364.
45. Li, G.; Shi, J. J A.E. On comparing three artificial neural networks for wind speed forecasting. **2010**, *87*, 2313–2320.
46. Crisosto, C. J I.J. of P. Autoregressive Neural Network for Cloud Concentration Forecast from Hemispheric Sky Images. **2019**, *2019*.
47. Walczak, S. Artificial neural networks. In *Advanced Methodologies and Technologies in Artificial Intelligence, Computer Simulation, and Human-Computer Interaction*; IGI Global, 2019; pp. 40–53.
48. Hayou, S.; Doucet, A.; Rousseau, J. On the Selection of Initialization and Activation Function For Deep Neural Networks. *ArXiv Prepr. ArXiv180508266* **2018**.
49. Karlik, B.; Olgac, A.V. Performance analysis of various activation functions in generalized MLP architectures of neural networks. *Int. J. Artif. Intell. Expert Syst.* **2011**, *1*, 111–122.
50. Aljarah, I.; Faris, H.; Mirjalili, S.; Al-Madi, N. J N.C.; Applications Training radial basis function networks using biogeography-based optimizer. **2018**, *29*, 529–553.
51. Er, M.J.; Wu, S.; Lu, J.; Toh, H.L. J I. transactions on neural networks Face recognition with radial basis function (RBF) neural networks. **2002**, *13*, 697–710.
52. Lee, S.; Kil, R.M. J N. networks A Gaussian potential function network with hierarchically self-organizing learning. **1991**, *4*, 207–224.
53. Haykin, S. *Neural networks: a comprehensive foundation*; Prentice Hall PTR, 1994; ISBN 0-02-352761-7.
54. Fan, S.; Liao, J.R.; Yokoyama, R.; Chen, L.; Lee, W.-J. Forecasting the wind generation using a two-stage network based on meteorological information. *IEEE Trans. Energy Convers.* **2009**, *24*, 474–482.
55. Mhaskar, H.N. J N. computation Neural networks for optimal approximation of smooth and analytic functions. **1996**, *8*, 164–177.
56. Niyogi, P.; Girosi, F. J N.C. On the relationship between generalization error, hypothesis complexity, and sample complexity for radial basis functions. **1996**, *8*, 819–842.
57. Breiman, L. J M. learning Random forests. **2001**, *45*, 5–32.
58. Rodriguez-Galiano, V.; Sanchez-Castillo, M.; Chica-Olmo, M.; Chica-Rivas, M. J O.G.R. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. **2015**, *71*, 804–818.
59. Xu, Z.; Lian, J.; Bin, L.; Hua, K.; Xu, K.; Chan, H.Y. J W. Water Price Prediction for Increasing Market Efficiency Using Random Forest Regression: A Case Study in the Western United States. **2019**, *11*, 228.
60. Segal, M.R. Machine learning benchmarks and random forest regression. **2004**.
61. Singhal, Y.; Jain, A.; Batra, S.; Varshney, Y.; Rathi, M. Review of Bagging and Boosting Classification Performance

on Unbalanced Binary Classification.; IEEE, 2019; pp. 338–343.
62. Liaw, A.; Wiener, M. J R. news Classification and regression by randomForest. **2002**, *2*, 18–22.
63. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V. J J. of machine learning research Scikit-learn: Machine learning in Python. **2011**, *12*, 2825–2830.
64. Mercadier, M.; Lardy, J.-P. J E.J. of O.R. Credit spread approximation and improvement using random forest regression. **2019**.
65. Wang, Q.; Li, Z.; Zheng, S.; Gu, S.; Sun, Y.; Wang, K. Detecting Unknown Malware on Android by Machine Learning Using the Feature of Dalvik Operation Code.; Atlantis Press, 2017.
66. Vapnik, V. *The nature of statistical learning theory*; Springer science & business media, 2013; ISBN 1-4757-3264-3.
67. Basak, D.; Pal, S.; Patranabis, D.C. J N.I.P.-L.; Reviews Support vector regression. **2007**, *11*, 203–224.
68. Vapnik, V.; Golowich, S.E.; Smola, A.J. Support vector method for function approximation, regression estimation and signal processing.; 1997; pp. 281–287.
69. Gunn, S.R. J I. technical report Support vector machines for classification and regression. **1998**, *14*, 5–16.
70. Qi, M.; Luo, H.; Wei, P.; Fu, Z. J F. Estimation of low calorific value of blended coals based on support vector regression and sensitivity analysis in coal-fired power plants. **2019**, *236*, 1400–1407.
71. Zhong, H.; Wang, J.; Jia, H.; Mu, Y.; Lv, S. J A.E. Vector field-based support vector regression for building energy consumption prediction. **2019**, *242*, 403–414.
72. Yu, P.-S.; Chen, S.-T.; Chang, I.-F. J J. of H. Support vector regression for real-time flood stage forecasting. **2006**, *328*, 704–716.
73. Chen, K.-Y.; Wang, C.-H. J T.M. Support vector regression with genetic algorithms in forecasting tourism demand. **2007**, *28*, 215–226.
74. Ortiz-García, E.G.; Salcedo-Sanz, S.; Pérez-Bellido, Á.M.; Portilla-Figueras, J.A. J N. Improving the training time of support vector regression algorithms through novel hyper-parameters search space reductions. **2009**, *72*, 3683–3691.
75. Cherkassky, V.; Ma, Y. J N. networks Practical selection of SVM parameters and noise estimation for SVM regression. **2004**, *17*, 113–126.
76. Ito, K.; Nakano, R. Optimizing support vector regression hyperparameters based on cross-validation.; IEEE, 2003; Vol. 3, pp. 2077–2082.
77. Williams, C.K.; Rasmussen, C.E. *Gaussian processes for machine learning*; MIT Press Cambridge, MA, 2006; Vol. 2;.
78. Neal, R.M. *Bayesian learning for neural networks*; Springer Science & Business Media, 2012; Vol. 118; ISBN 1-4612-0745-2.
79. Hu, J.; Wang, J. J E. Short-term wind speed prediction using empirical wavelet transform and Gaussian process regression. **2015**, *93*, 1456–1466.
80. Murphy, K.P. *Machine learning: a probabilistic perspective*; MIT press, 2012; ISBN 0-262-30432-5.
81. Rasmussen, C.E.; Nickisch, H. J J. of machine learning research Gaussian processes for machine learning (GPML) toolbox. **2010**, *11*, 3011–3015.
82. Kirkwood/University of Iowa Wind Data Project Available online: http://epscor2.cgrer.uiowa.edu/html/kirkwood.html (accessed on Jan 24, 2017).
83. Louis, J.-F. A parametric model of vertical eddy fluxes in the atmosphere. *Bound.-Layer Meteorol.* **1979**, *17*, 187–202.
84. Svenningsen, L. *Power Curve Air Density Correction And Other Power Curve Options In WindPRO*; Technical report, EMD International A/S, 2010. URL www. emd. dk/files/windpro/WindPRO_Power_Curve_Options. pdf, 2010;
85. Clifton, A.; Smith, A.; Fields, M. *Wind Plant Preconstruction Energy Estimates. Current Practice and Opportunities*; National Renewable Energy Lab.(NREL), Golden, CO (United States), 2016;
86. Manobel, B.; Sehnke, F.; Lazzús, J.A.; Salfate, I.; Felder, M.; Montecinos, S. Wind turbine power curve modeling based on Gaussian Processes and Artificial Neural Networks. *Renew. Energy* **2018**, *125*, 1015–1020.
87. Ata, R. Artificial neural networks applications in wind energy systems: a review. *Renew. Sustain. Energy Rev.* **2015**, *49*, 534–632.
88. Liu, J.; Wang, X.; Lu, Y. A novel hybrid methodology for short-term wind power forecasting based on adaptive neuro-fuzzy inference system. *Renew. Energy* **2017**, *103*, 620–629.
89. Lew, D.; Milligan, M.; Jordan, G.; Piwko, R. *Value of Wind Power Forecasting*; National Renewable Energy Lab.(NREL), Golden, CO (United States), 2011;

Appendix A

Table 5. Upper and Lower bounds for the 4 Layer MAE.

4 Layer ANN			
Wind Speed (m/s)	**MAE (kW)**	**UB (kW)**	**LB (kW)**
2.3	15.79	15.90	15.69
3.0	12.04	12.06	12.02
3.7	13.53	13.55	13.51
4.4	16.90	16.92	16.87
5.1	18.76	18.78	18.74
5.8	21.10	21.12	21.08
6.4	26.62	26.64	26.59
7.1	35.33	35.37	35.30
7.8	46.94	46.98	46.89
8.5	53.66	53.72	53.60
9.2	64.28	64.37	64.19
9.9	62.37	62.52	62.22
10.6	60.65	60.83	60.47
11.2	45.02	45.16	44.87
11.9	31.72	32.16	31.27
12.6	25.92	26.60	25.24
13.3	18.63	19.43	17.84
14.0	16.49	17.52	15.46
14.7	11.54	12.98	10.10
15.3	11.76	13.63	9.89

Highlights

The major highlights of the manuscript are

1. Atmospheric Inputs improve ANN performance for long term energy forecasting.
2. The optimal number of hidden layers was 4 (100,50,20,10 neurons).
3. The 4 layer ANN outperformed RBF, RF, SVR and GP machine learning algorithms.
4. The model improved the energy production forecasting error to 0.4%.
5. The results were dependent on atmospheric stability.

Conflict of interest Form

The authors notify that there's no financial/personal interest or belief that could affect their objectivity and declare not conflicts of interest.