2017

Condition Monitoring of Wind Turbines Using Intelligent Machine Learning Techniques

Majid Morshedizadeh
University of Windsor

Condition Monitoring of Wind Turbines Using Intelligent Machine Learning Techniques

By

Majid Morshedizadeh

A Thesis
Submitted to the Faculty of Graduate Studies
through the Department of Mechanical, Automotive and Materials Engineering
in Partial Fulfillment of the Requirements for
the Degree of Master of Science
at the University of Windsor

Windsor, Ontario, Canada

2017

Condition Monitoring of Wind Turbines Using Intelligent Machine Learning Techniques

by

Majid Morshedizadeh

APPROVED BY:

Dr. Lisa Porter
Department of Biological Sciences

Dr. Jalal Ahamed
Department of Mechanical, Automotive and Materials Engineering

---

Dr. Rupp Carriveau, Co-Advisor
Department of Civil and Environmental Engineering

Dr. David S-K. Ting, Co-Advisor
Department of Mechanical, Automotive and Materials Engineering

April 28, 2017

DECLARATION OF CO-AUTHORSHIP/PREVIOUS PUBLICATIONS

I. Co-Authorship Declaration

I hereby declare that this thesis incorporates material that is result of joint research, as follows:

Thesis Chapters	**Details**
Chapter 2 Chapter 3 Chapter 4	This thesis incorporates the outcome of a joint research project undertaken in collaboration with Dr. Mehrdad Saif and Dr. Mojtaba Kordestani under the supervision of Dr. Rupp Carriveau and Dr. David S-K. Ting. In all cases, the author performed the key ideas, primary contributions and data analysis and interpretation, and the contribution of the co-authors was primarily through the provision of monitoring and manuscript structure checking.

I am aware of the University of Windsor Senate Policy on Authorship and I certify that I have properly acknowledged the contribution of other researchers to my thesis, and have obtained written permission from each of the co-author(s) to include the above material(s) in my thesis.

I certify that, with the above qualification, this thesis, and the research to which it refers, is the product of my own work.

II. Declaration of Previous Publication

This thesis includes three original papers that have been previously published/submitted for publication in peer reviewed journals, as follows:

Thesis Chapters	**Publication title/full citation**	**Publication status**
Chapter 2	M. Morshedizadeh, M. Kordestani, R. Carriveau, D. S-K. Ting, M. Saif, ”Improved power curve monitoring of wind turbines,” Wind Engineering	Accepted for publication
Chapter 3	M. Morshedizadeh, M. Kordestani, R. Carriveau, D. S-K. Ting, M. Saif, ”Application of Imputation Techniques and Adaptive Neuro-Fuzzy Inference System to Predict Wind Turbine Power Production,” Energy	Under review

I certify that I have obtained a written permission from the copyright owner(s) to include the above published material(s) in my thesis. I certify that the above material describes work completed during my registration as graduate student at the University of Windsor.

I declare that, to the best of my knowledge, my thesis does not infringe upon anyones copyright nor violate any proprietary rights and that any ideas, techniques, quotations, or any other material from the work of other people included in my thesis, published or otherwise, are fully acknowledged in accordance with the standard referencing practices. Furthermore, to the extent that I have included copyrighted material that surpasses the bounds of fair dealing within the meaning of the Canada Copyright Act, I certify that I have obtained a written permission from the copyright owner(s) to include such material(s) in my thesis.

I declare that this is a true copy of my thesis, including any final revisions, as approved by my thesis committee and the Graduate Studies office, and that this thesis has not been submitted for a higher degree to any other University or Institution.

ABSTRACT

Wind Turbine condition monitoring can detect anomalies in turbine performance which have the potential to result in unexpected failure and financial loss. This study examines common Supervisory Control And Data Acquisition (SCADA) data over a period of 20 months for 21 pitch regulated 2.3 MW turbines and is presented in three manuscripts.

First, power curve monitoring is targeted applying various types of Artificial Neural Networks to increase modeling accuracy. It is shown how the proposed method can significantly improve network reliability compared with existing models.

Then, an advance technique is utilized to create a smoother dataset for network training followed by establishing dynamic ANFIS network. At this stage, designed network aims to predict power generation in future hours.

Finally, a recursive principal component analysis is performed to extract significant features to be used as input parameters of the network. A novel fusion technique is then employed to build an advanced model to make predictions of turbines performance with favorably low errors.

AKNOWLEDGEMENTS

I would like to express my deepest appreciation to my supervisors Dr. Rupp Carriveau, and Dr. David S-K. Ting for their perpetual supports, patience, and inspiration during my Master's program in the University of Windsor. It was such an honor for me to know you, and be part of your research team. Thank you very much for providing me such an opportunity to learn from you.

In addition, I would like to express my gratitude to my committee members Dr. Jalal Ahamed, and Dr. Lisa Porter for their helpful advice and suggestions to my thesis.

I would also like to express my special thanks to my friends and colleagues for helping me during the past two years.

DEDICATION

I would like to dedicate my thesis to my lovely parents, Jafar and Mali, for their unconditional love and support without which I could never succeed. To my brother and hero, Ali, who has always inspired me to pursue my dreams, and finally to my beautiful sister, Nafiseh, whom I can never get by without her guidance and encouragement.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

Wind energy, which is clean, renewable, and without greenhouse eect, has become an alternative energy to fossil fuels [1]. In recent years, wind power, due to its lowest adverse impacts on environment and its sustainability, has become one of the most promising renewable energy sources with the most extensive utilization worldwide [2, 3].

In addition, the growth in wind energy capacity makes the optimization of turbine performance absolutely essential. Also, It has become apparent that sometimes wind turbine do not perform as well as they are expected and under performance and unexpected failures are observed. Thus, operators might not be able to deliver their traded amount of energy and may even have to pay fines and bear the high maintenance costs. Nowadays, the development of maintenance strategy is supported by computer technology both in hardware and software. Recent developed methods are using artificial intelligent (AI) techniques as tools for maintenance routines [4]. Accurate modeling of wind turbines performance as targeted by ongoing research studies can increase wind energy production capabilities, and reliability and give rise to general reduction of financial risk in wind farm investment.

A variety of approaches can be taken to maximize the accuracy and efficiency of wind turbines condition monitoring. For the presented study, Supervisory Control And Data Acquisition (SCADA) data was available. It was a large amount of historical performance data related to 2.3 MW wind turbines located in Ontario, Canada. This work, in which power curves of wind turbines are investigated, focuses on monitoring and prediction of wind turbine power production based on this machine historical performance data. The general process in establishment of any model following such data-driven approaches, is presented in Figure 1.0.1.

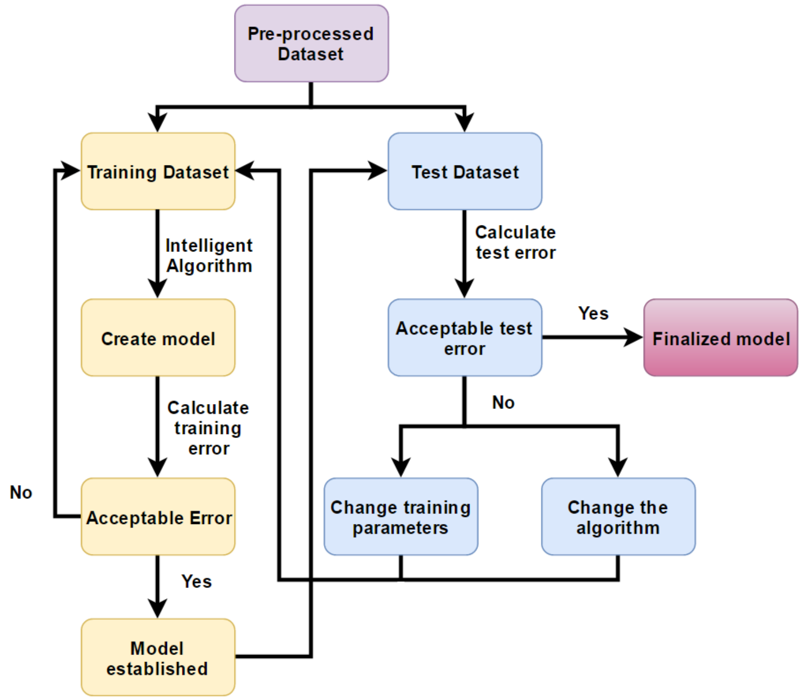

FIGURE 1.0.1: Data-driven modeling steps

The entire thesis will be comprised of three manuscripts, each presented as a separate chapter. In chapter 2, the most influential features on power production are selected based on the correlation coefficient, and two types of artificial neural networks in various structures are established to estimate the turbines power production. In chapter 3, imputation techniques are applied to create smoother data set to achieve better network training, leading to more accurate models. In addition, Adaptive-Neuro Fuzzy Inference System (ANFIS) is applied for the purpose of predicting the future performance of the turbines. In chapter 4, in addition to imputation technique, Principal Component Analysis (PCA) is employed to extract the features with higher level of significance. Also, a novel and practical method for wind turbines performance prediction is presented. Moreover, in this chapter, a new data fusion technique, combining MLP and ANFIS networks, is established to increase the prediction accuracy. Finally, chapter 5 concludes the achievements and present the options to extend this work further.

References

[1] V. Fthenakis and H. C. Kim, “Land use and electricity generation: A life-cycle analysis,” *Renewable and Sustainable Energy Reviews*, vol. 13, no. 6, pp. 1465–1474, 2009.

[2] D. Y. Leung and Y. Yang, “Wind energy development and its environmental impact: A review,” *Renewable and Sustainable Energy Reviews*, vol. 16, no. 1, pp. 1031–1039, 2012.

[3] M. Premalatha, T. Abbasi, S. Abbasi *et al.*, “Wind energy: Increasing deployment, rising environmental concerns,” *Renewable and Sustainable Energy Reviews*, vol. 31, pp. 270–288, 2014.

[4] A. Widodo and B.-S. Yang, “Support vector machine in machine condition monitoring and fault diagnosis,” *Mechanical systems and signal processing*, vol. 21, no. 6, pp. 2560–2574, 2007.

CHAPTER 2

Improved Power Curve Monitoring of Wind Turbines

2.1 Introduction

Continuous reductions in costs have helped to grow the size of the wind energy sector worldwide. The optimization of performance, and the management of operations and maintenance represent some of today's largest challenges. Subsequently, numerous studies have targeted the optimization of power production, improving reliability and the general reduction of financial risk in wind farm investment [1, 2, 3].

Generally, there are two principal approaches to any system performance analysis [4, 5]; physics-based models can be developed [6], or data-driven approaches can be applied [7, 8]. The latter, which analyzes historical performance data has a number of advantages over physics-based models. In most situations, the complexity of the machines and their sub-systems make it challenging to build accurate models. There are complex non-linear relationships between several sub-systems of a wind turbine that make the physical modeling of the system a very complicated task to perform. The share of influence of any sub-system on the performance of other parts can be challenging to characterize. It may even differ from one turbine to another. For instance, wind farm topology, wind pattern, environmental aspects and siting can affect machine performance in various manners [9]. In addition, the normal systems simplifications required in most physical and numerical approaches may negatively affect the results, and can thus reduces the applicability of the models. Experimental methods which, can at times, be considered more valuable than numerical analysis,

TABLE 2.1.1: Wind Turbines Modeling Techniques Comparison

Wind Turbine analysis		Advantages	Disadvantages
Physics-based	Experimental	more reliable results than numerical analysis	having to scale down, expensive and timely
	Numerical	Easier to perform, cheaper than experiment	simplification required, validation required
Data-driven	Parametric	simple model, easy to obtain	not suitable for complex systems, high prediction error
	Non-parametric	very reliable, low prediction error	susceptible to sensor performance

also suffer from major limitations. Two common challenges include the inability to identically mimic field environmental conditions in a laboratory, and the non-trivial obstacle of dynamic scaling [10].

Data-driven methods are well positioned for wind turbine performance analysis owing to a couple of major factors: 1. A large amount of Supervisory Control And Data Acquisition (SCADA) data is available to all wind farm operators. 2. They can often be executed for less cost than comparable numerical or experimental approaches [11, 12].

Data-driven studies can be categorized in two major groups. As explained by Lydia et al. [13], the first type, a parametric model, applies a finite number of parameters to describe a distribution. For wind turbines, in particular, this type of model is established based on fitting mathematical expressions to a power curve. The other approach is known as non-parametric, in which, in contrast, the quality and quantity of parameters are not fixed in advance and are subject to change. Also in this method, the power output is a function of wind speed [14].

Table 2.1.1 summarizes popular wind turbine modeling techniques, their advantages and disadvantages. Several studies have been conducted to develop accurate data-driven models for condition monitoring of wind turbines. A comparative study using neural networks and regression based models was conducted by Schlechtinen and Santos [15] and Li et al. [16], where bearing temperature was monitored [15]; while authors in [16] performed power curve monitoring. In both studies, neural networks produced less test errors although they were more complex in comparison with regression methods. This suggests a need for more versatile and robust models that balance comprehensiveness and simplicity of application.

Kim et al. [17] established two multilayer neural networks using different input signals to design a power curve based fault detection system by way of a practical application of neural networks. A comprehensive study was conducted in [13] comparing a variety of parametric and non-parametric methods, concluding that neural networks were a reliable machine learning technique to monitor wind turbine performance.

Pelletier et al. [18] compared multilayer neural networks with other parametric and non-parametric models and showed that their proposed neural networks resulted in less prediction error in power output. An adaptive neuro-fuzzy approach was employed by Petkovic et al. [19] to estimate the power coefficient. To do so, the authors applied data obtained from the suggested equation of Heier [20]. If experimental data had been used, the results would have been more valuable. Kusiak et al. [21] derived non-linear parametric models in addition to the k-NN model and concluded that although the k-NN model prediction had acceptable accuracy, the parametric approach could also be used as a performance monitoring tool. Given the popularity and proven abilities of neural network modeling for turbine power analytics, they were considered for this study as well.

The work introduced in this paper focuses on monitoring of wind turbine power. For this purpose, an input selection method based on the physical and statistical parameter influence on output power is introduced. Then, a dynamic neural network using historical information from inputs and outputs is constructed to estimate the power curve of the turbines. Two of the most common and powerful NN techniques (RBF and MLP) are built, and their static and dynamic performance is discussed. It is shown how the feature selection method and dynamic network can generate a much more accurate model. The results are compared with existing models in the literature and it is shown that the proposed method returns the lowest prediction error. Comparisons are made in terms of mean absolute error (MAE) which is an appropriate indication of model accuracy.

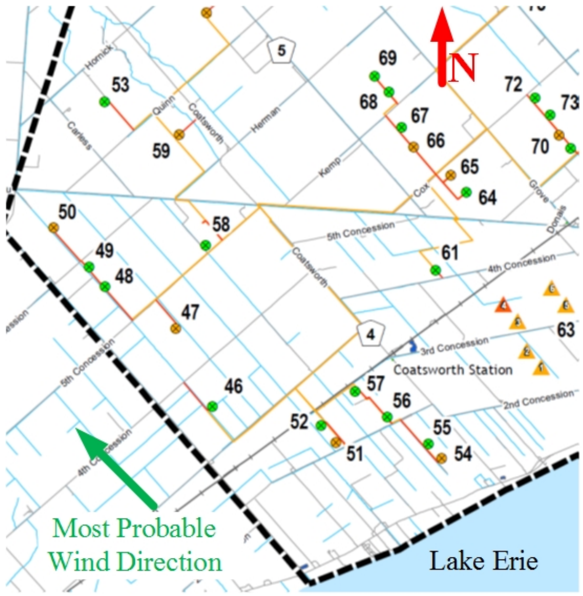

FIGURE 2.2.1: Turbines Layout

The principal novelty of this paper is a powerful feature selection method based on correlation analysis that enables the determination of the optimal number of inputs and outputs for comprehensive system monitoring. Furthermore, for the first time a dynamic neural network is utilized to benefit from historical system information and improve the accuracy of the estimation.

The rest of paper is organized as follows. Wind farm, turbine characteristics, and data pre-processing steps are explained in Section 2.2. Section 2.3 introduces the proposed design methods consisting of the feature selection and the artificial neural networks. Then, a comprehensive simulation study and test results are provided in Section 2.4. A detailed models comparison and analysis of the impacts of feature selection on model accuracy are also presented in this section. Finally, Section 4.7 concludes with summary of the results.

2.2 Turbine Characteristics and Data Pre-Processing

In this research, twenty-one, 2.3 MW, pitch-regulated wind turbines have been investigated. The study takes place in a wind farm in Ontario, Canada and the data cover a range of 20 months from February 2014 to September 2015. Figure 3.3.1 illustrates the farm map and

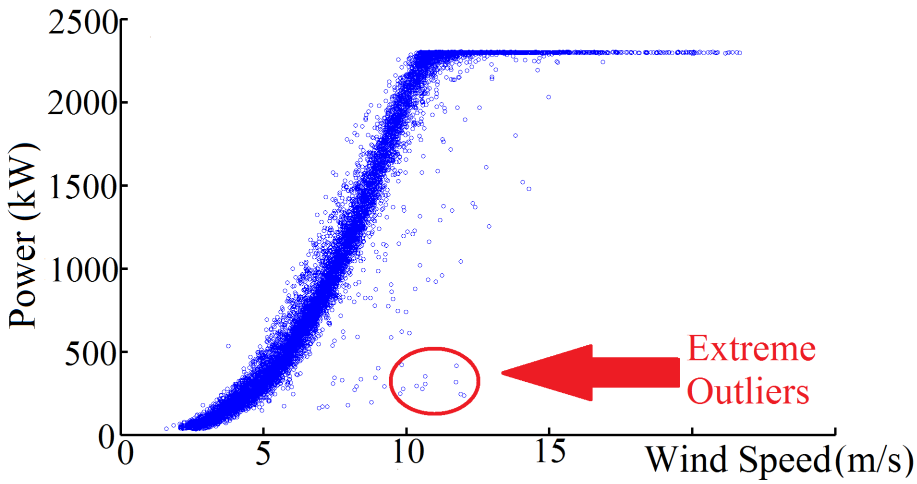

FIGURE 2.2.2: Indication of Outliers in Turbine Power Curve

turbine layout.

It is essential to pre-process the SCADA data before building the networks. As proposed in [15] the first step is to check the validity of data. There are two principal reasons for this. First, by omitting the extreme outliers from the data set, smoother network generalization will be easier to obtain [22]. Secondly, it is possible that due to sensor malfunction or system processing errors, the value of a parameter recorded is well out of the anticipated range as also mentioned by Caselitz and Giebhardt [23]. Thus, it is also important to carefully determine a data range to identify out-of-range values which are not results of a sensor error or misreading.

On the other hand, careful action should be taken not to eliminate anomalies that could be indicative of potential problems in machine performance. Figure 2.2.2 shows the power curve of one of the investigated turbines and extreme outliers are indicated in the Figure. Since the number of these points is negligible in comparison with the whole data set and their occurrence is not regular, we assess them as anomalous and thus remove them for smoother network training.

Eventually, to be able to apply the data as input parameters, the following criteria should be met [23]; a) Data points fall within the anticipated range. b) Components of the data set are mutually consistent. c) Output data is consistent with the input signals.

In order to have the ability to apply multiple inputs and properly train the network, input parameters with different ranges need to be scaled to a similar range. Otherwise, the variable with wider range will dominate over others in the network training phase. Further, it is normal in large data sets to have some data missing. These missing values harm the network and must be either imputed or removed depending on the application [15, 24, 25]. Here, due to availability of large data, the missing values are neglected and since sampling was done at a relatively high frequency, like in [18], 10-min average data was created and applied.

Active power translates most directly to wind farm revenue, subsequently, it was chosen as our network output signal. To monitor turbine performance, power curves that illustrate the relationship between wind speed and output power, are typically applied [18, 26, 27]. This curve, as can be observed in Figure 4.2.2, has three distinct regions. In the first region where wind speed is lower than the required minimum speed for power production, known as cut-in speed, there is no power production. In the second region, as the wind speed increases, the output power also grows rapidly until it reaches the rated power. Finally in the third region, the power output remains constant. This region ends when the wind speed exceeds the maximum cut-out speed beyond which turbines blades are regulated to not rotate due to mechanical limitations.

2.3 design procedure of wind turbine monitoring

This section introduces the design methodology of the proposed monitoring method. The basic premise is to develop an intelligent estimator to fully monitor the system and track the power curve of the wind turbine using dynamic neural networks. First, the feature selection method is explained. Then, the neural network is presented.

2.3.1 Feature Selection

A basic but crucial step in condition monitoring of any system, is the appropriate selection of input and output parameters for which a number of methods have been proposed [28,

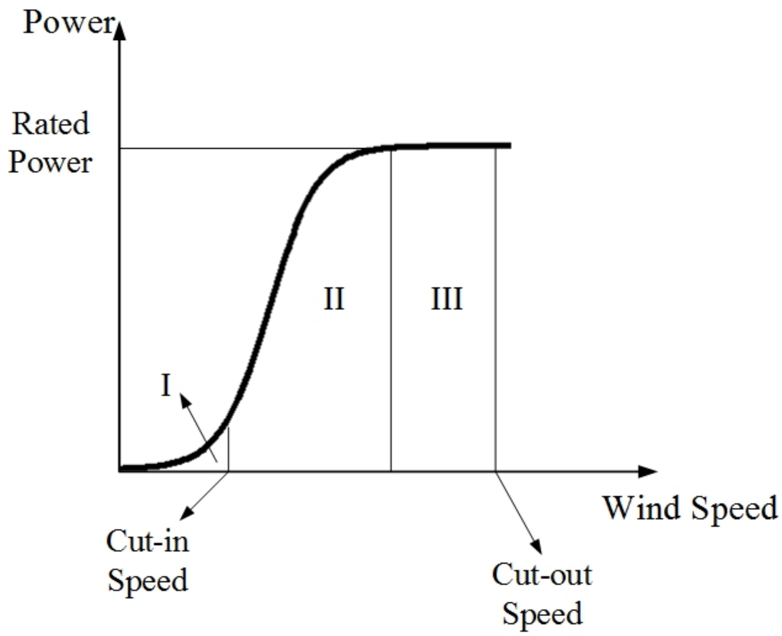

FIGURE 2.2.3: Schematic Power Curve of a Wind Turbine

29, 30, 31]. This is particularly the case for wind turbine performance monitoring, where there are many parameters to choose from.

For instance, authors in [14] and [16] used only wind speed as the input to monitor power curve. A large number of different models were developed by Schlechtingen et al. [11] with a variety of input-output configurations for which little specific reason was given for the selections. Sun et al [32] applied a GPLS method to select effective parameters to influence generator bearing temperature. It was also argued by Schlechtingen et al. [33] that considering environmental effects such as wind direction and ambient temperature would result in more accurate models and they considered those two parameters plus wind speed as the inputs to monitor power curve.

It is clear that the number of inputs must go beyond what is indicated by expression 4.2.1.

$$P = \frac{1}{2}\rho A C_p V^3 \tag{2.3.1}$$

where ρ is the air density, A is the rotor sweep area, V is the wind speed, and C_p is the power coefficient, which is a function of blades pitch angle and tip-speed ratio.

Based on Equation 4.2.1, wind speed is the most influential factor on power curve mod-

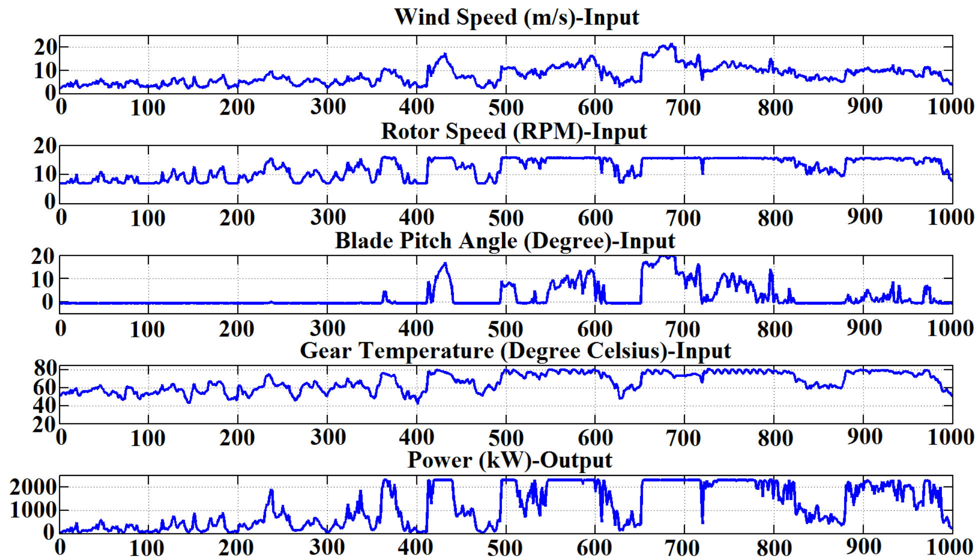

FIGURE 2.3.1: input and output parameters

elling of wind turbines. This parameter used to be the only factor considered in many previous studies in which the power curve was modeled. Established models following this method have shown insufficient accuracy and made it clear that other parameters have to be acknowledged as well. It is proven that consideration of other relevant factors leads to more advanced models and less prediction errors. That said,indiscriminately increasing the number of inputs will decrease the neural network train-ability and result in even more errors and unreliability of the model. Finding this optimum number of values with the most influence is a challenging task.

To address this issue, we apply a statistical correlation method enhanced by knowledge of physical system. To this end, Pearson product-moment rank correlation coefficient is utilized as follows [34, 35]:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.3.2}$$

where x_i and y_i are the two parameters between which the strength of relationship is being investigated, n is the number of data points and $\bar{x}$ and $\bar{y}$ are the mean values of each parameter. The correlation coefficients outline the strength of the relationship between two variables by representing a number between -1 and 1, where 1 represents perfect correlation and 0 means no correlation. Negative values indicate the reverse correlations, i.e. when one variable increases, the other one decreases and vice versa [36]. Finally to evaluate the results of the correlation method and choose the optimal number for the feature selection, the Akande critrion is considered. Akande et al. [36] suggested a rule of thumb for the interpretation of absolute values of the correlation coefficients. For the values greater than 0.68, the correlation is considered strong. There is a moderate relationship between two variables if the coefficient is between 0.36 and 0.68 and finally for values below 0.36, the relationship is considered negligible.

2.3.2 The artificial neural network

The MLP network is a feed-forward neural network that provides a mapping from system inputs to outputs. It is well-suited for function approximation, pattern recognition, etc [37]. The MLP consists of multiple layers of nodes in a forward direction in which each node is fully connected to the nodes in the next layer. The MLP structure are made of three types of layers including an input layer, hidden layer, and output layer. Each node in the hidden layer is a neuron with a non-linear activation function like sigmoids, hyperbolic tangent, etc. The input layer acts as a buffer and the output layer usually has nodes with linear functions. The MLP network applies a supervised learning algorithm known as error back propagation for training the network.
The RBF neural network is a feed forward network that has radial basis functions [38]. The RBF Network also consists of input, hidden and output layers. In the hidden layer all neurons perform a Gaussian function as expressed:

$$F(x) = exp(-\frac{||x-c||}{2\sigma^2}) \tag{2.3.3}$$

where c is the center of basis function and σ is called function radius.
The output layer is a linear function represented as

$$\Phi(x) = \sum_{i=1}^{N} w_i F(x) \tag{2.3.4}$$

where N is the number of hidden neurons and w_i is the weight of output neurons [39]. The RBF network is known as global approximation, meaning that if enough nodes are supplied by the hidden layers, the RBF network can estimate a non-linear function.

Neural networks can be represented in static and dynamic structures. In the static type, the network is simply trained using the selected input parameters as described in Equation 2.3.5:

$$y(t) = f(u_1(t), u_2(t), ..., u_i(t)) \tag{2.3.5}$$

where $y(t)$ is the network output at the time t and u_i is the ith input of the network.
In the proposed dynamic networks, in addition to the selected parameters as the network inputs ($u_i(t)$), the inputs and the output of the previous iteration time step ($u_i(t-1)$ and $y(t-1)$) are also considered as the network inputs. In other words, the output resulted from each set of inputs plus those inputs are also given to the network for better training. In this case the output function would be described as follows:

$$\begin{aligned} y(t) = g(&u_1(t), u_1(t-1), u_2(t), u_2(t-1), ..., \\ &u_i(t), u_i(t-1), y(t-1)) \end{aligned} \tag{2.3.6}$$

2.4 Simulation modelling and test scenario

This section presents several test scenarios to validate the effectiveness of the proposed methods. In the following, the suggested feature selection method is illustrated. Then, the structure of the proposed neural network is introduced. Finally, a comprehensive comparison between proposed and existing methods and an analysis of feature selection will be provided.

TABLE 2.4.1: Correlation Coefficients of the selected signals

Parameter	Pearson product-moment
Wind Speed	0.939
Rotor Speed	0.926
Gear Temperature	0.893
Blade Pitch Angle	0.663
Blade Pitch Angle SD	0.582
Rotor Speed SD	-0.294
Gear Temperature SD	-0.273
Turbulence Intensity	0.185
Active Power SD	0.114
Wind Direction SD	-0.091
Wind Direction	0.003

2.4.1 The proposed feature selection for the inputs of the network

Based on the available data, physical understanding of wind turbines and the most common parameters selected in the literature, the features to be considered as potential input signals are outlined in Table 4.6.1. Furthermore, the result of the correlation analysis given by equation 2.3.2 is also summarized in the Table 4.6.1. It is indicated from this table that wind speed which is expected to have the strongest correlation with output power in comparison with other signals, has the closest number to 1 with the coefficient value of 0.939. While other environmental factors related to the wind, including turbulence intensity and wind direction have been proven to have impacts on general wind turbine performance [40], they do not appear influential for this study based on our correlations. Thus we will not consider them to avoid increasing network complexity and training time. They may offer a marginal prediction improvement, or worse, a potentially wider range of prediction error. Based on the results summarized in Table 4.6.1, wind speed, rotor speed, gear temperature and blade pitch angle are selected as input parameters. Figure 2.3.1 shows a part of the input and output parameters in a high resolution time series. The correlation between parameters is visible in this figure based on their strength. Wind speed, for instance, indicates the most similar trend to power output as anticipated.

2.4.2 The structure of the proposed neural networks

Static and dynamic variations of both Multilayer Perception (MLP) and Radial Basis Function (RBF) networks are established. RBF and MLP networks are established according to

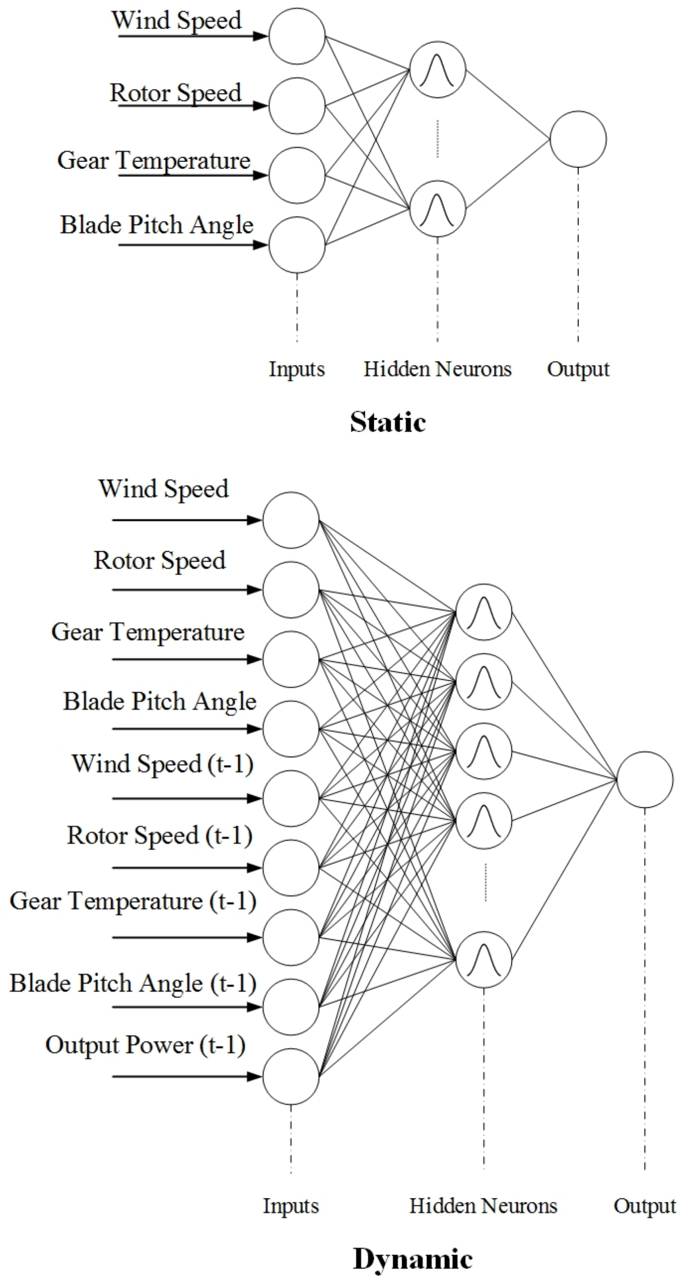

FIGURE 2.4.1: RBF Networks Structures

the structures shown in Figure 2.4.1 and Figure 4.6.1 respectively.

Generally, there is no concrete rule about the size of the data complement required to obtain the best possible training, but in general, the training data should contain the data range boundaries and must be sufficient to represent the entire period [22]. In this study, 50 percent of 10-min average data was used for the training and the other half for testing. Since the data available covers all seasons, this approach makes it possible to consider seasonal changes in the network and thus, the ambient temperature effects will be considered automatically (without it being a distinct input parameter).

To train the networks, the gradient descent with momentum method is applied. In this method, in addition to error calculation, the general error trend will also be determined. This reduces the risk of local minima and results in enhanced generalization [15].

The other important factor in the structure of the network is the number of neurons in the hidden layer. To be able to find the optimum number of neurons, at least 10 runs should be performed while only varying the number of neurons to seek the configuration with the best generalization [23, 22]. This helps to avoid over-fitting.

Test results are shown in Figure 2.4.3. As shown in the figure, the resulting test errors for the networks are quite acceptable. For the majority of the data points, calculated errors are in the very low range. Further, it is also visible in the figures that for both MLP and RBF networks, the dynamic model outperforms the static one, based on the fact that there are more points near zero error in dynamic networks than the respective static ones.

2.4.3 Models Comparison

Mean absolute error (MAE) indicated by Equation 3.4.10 has been applied to analyze the performance of the four established neural networks.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{2.4.1}$$

where x_i is the predicted value by the neural network and y_i is the actual value.

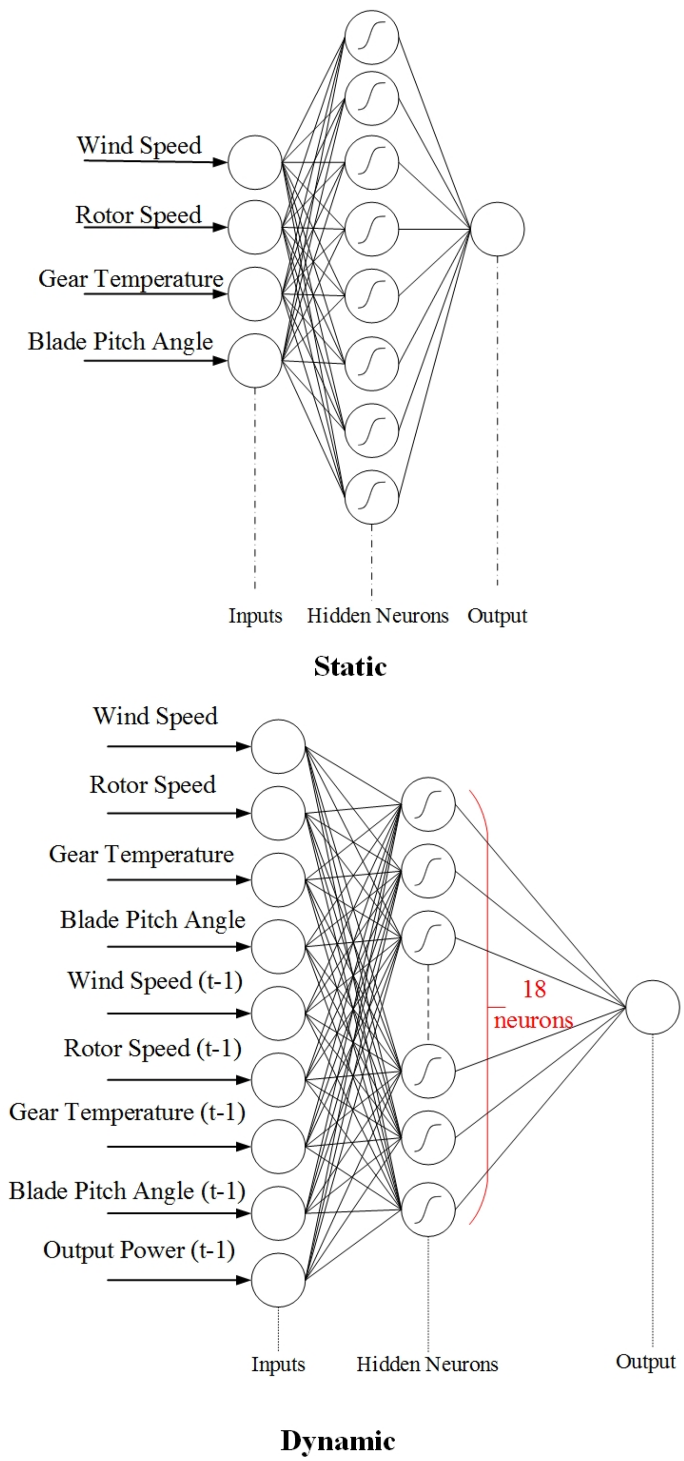

FIGURE 2.4.2: MLP Networks Structures

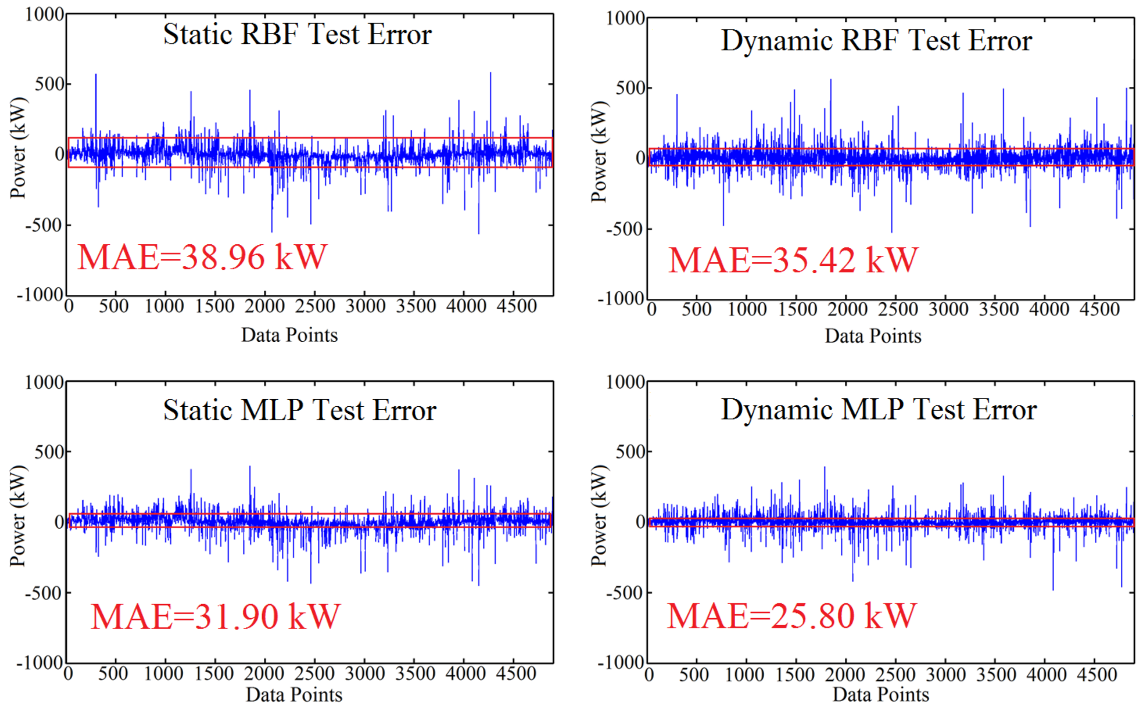

FIGURE 2.4.3: Established Networks Test Errors

TABLE 2.4.2: MAE Results for All Models

Network	MAE(kW)
Dynamic MLP	25.80
Static MLP	31.90
Dynamic RBF	35.42
Static RBF	38.96

TABLE 2.4.3: MAE Comparison

Network Types	Scaled MAE(kW)
Dynamic MLP	1.12
Static MLP	1.39
Dynamic RBF	1.54
Static RBF	1.69
Static MLP [16]	5.30
Static MLP [21]	3.56
Static MLP [14]	5.50
Static MLP [33]	1.62
ANFIS [33]	1.60

MAE indicates the closeness of predicted results with true values. MAE is the most common type of error assessment applied in similar studies, which helps to facilitate comparison with existing models. The MAE results are summarized in Table 2.4.2. As evident in the table, the dynamic MLP network has performed the best with an MAE value of 25.80 kW. This appears acceptable considering its magnitude relative to the 2.3 MW turbines capacity studied here. As configured, this network should reduce false alarms, increase reliability and excel at detecting abnormal performance.

To appropriately compare the results of established models with the ones in the literature, the results should be scaled. Scaled values are presented in Table 2.4.3 along with existing models in the literature. Only NN models are considered for comparison except in [33], where models with the application of adaptive neuro-fuzzy inference system (ANFIS) has been claimed to give acceptable results.

The scaled MAE values in [16, 21, 14] and the two models proposed in [33], are chosen for the comparison. The authors in [18] did not mention their investigated turbines power rating which prevented scaling. Based on the acquired results except for static RBF network, all other networks considered here outperformed those in the literature. It is also worth mentioning that the RBF network training was notably faster than MLP, an advantage in some cases.

2.4.4 Feature Selection Analysis

In order to investigate the effects of each input parameter on the mean absolute error, in this part, MLP networks which outperformed the RBF networks, are established again. First,

TABLE 2.4.4: impact of each input parameter on prediction error

Inputs	Dynamic MLP Prediction Error (kW)	Static MLP Prediction Error (kW)
Wind Speed	42.55	66.68
Wind Speed Rotor Speed	35.33	48.98
Wind Speed Rotor Speed Gear Temperature	32.23	45.11
Wind Speed Rotor Speed Gear Temperature Blade Pitch Angle	25.80	31.90

the dynamic and static networks are trained by only wind speed, then, prediction error is calculated. In the next step, the other parameters, rotor speed, gear temperature and blade pitch angle are added one-by-one to the networks. Results are summarized in Table 2.4.4. It is shown that by adding the appropriate input parameters, the resulting error considerably decreases, leading to better prediction performance. The dynamic MLP network, which gives the highest accuracy, has a prediction error of 42.55 kW with only wind speed as the input, and this error gradually decreases when other inputs are also included. The same trend can be observed in static model as well, confirming proper feature selection. This result seems sensible based on the role of each parameter in turbine output power. Increases in rotor speed, generally suggest more power production. Gear temperature increases can be indicative of faster moving gears associated with greater power production. Finally, blade pitch angles are vital to optimizing the aerodynamics of energy capture.

2.5 Conclusion

In this paper, 2.3 MW, pitch regulated wind turbines were investigated over a period of 20 months. The output power curves were modeled using artificial neural networks. Two types of MLP and RBF networks were established in both static and dynamic states. In the dynamic configuration, the input and the output of the previous time interval were also applied to train the network. To select the most influential parameters from the data, a statistical correlation coefficient was employed. This helped enable the determination of their share of influence on the reduction of network prediction error. It was shown that by applying the rotor speed, gear temperature, and blade pitch angle, in addition to wind speed, the performance of the networks improved significantly. A 40 percent and 53 percent reduction in prediction error was observed for dynamic and static MLP networks, respectively, compared to, the state where only wind speed was considered. Furthermore, a comparison between similar models in literature and the models proposed in this study, revealed that the dynamic MLP network outperforms other models and was 30 percent more accurate than the best model proposed in the literature [33]. Such an outcome offers value to a growing wind energy industry that values accurate performance prediction.

References

[1] N. Gatzert and T. Kosub, "Risks and risk management of renewable energy projects: The case of onshore and offshore wind parks," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 982–998, 2016.

[2] X. Jia, C. Jin, M. Buzza, W. Wang, and J. Lee, "Wind turbine performance degradation assessment based on a novel similarity metric for machine performance curves," *Renewable Energy*, vol. 99, pp. 1191–1201, 2016.

[3] W.-H. Chen, C.-Y. Chen, C.-Y. Huang, and C.-J. Hwang, "Power output analysis and optimization of two straight-bladed vertical-axis wind turbines," *Applied Energy*, vol. 185, pp. 223–232, 2017.

[4] A. Heng, S. Zhang, A. C. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges and opportunities," *Mechanical Systems and Signal Processing*, vol. 23, no. 3, pp. 724–739, 2009.

[5] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *Instrumentation and Measurement, IEEE Transactions on*, vol. 60, no. 3, pp. 703–711, 2011.

[6] J. Montesano, H. Chu, and C. V. Singh, "Development of a physics-based multi-scale progressive damage model for assessing the durability of wind turbine blades," *Composite Structures*, vol. 141, pp. 50–62, 2016.

[7] A. Zaher, S. McArthur, D. Infield, and Y. Patel, "Online wind turbine fault detection through automated scada data analysis," *Wind Energy*, vol. 12, no. 6, pp. 574–593, 2009.

[8] T. Ouyang, A. Kusiak, and Y. He, "Modeling wind-turbine power curve: A data partitioning and mining approach," *Renewable Energy*, vol. 102, pp. 1–8, 2017.

[9] A. Steiner, C. Köhler, I. Metzinger, A. Braun, M. Zirkelbach, D. Ernst, P. Tran, and

B. Ritter, "Critical weather situations for renewable energies–part a: Cyclone detection for wind power," *Renewable Energy*, vol. 101, pp. 41–50, 2017.

[10] M. H. Giahi and A. J. Dehkordi, "Investigating the influence of dimensional scaling on aerodynamic characteristics of wind turbine using cfd simulation," *Renewable Energy*, vol. 97, pp. 162–168, 2016.

[11] M. Schlechtingen, I. F. Santos, and S. Achiche, "Wind turbine condition monitoring based on scada data using normal behavior models. part 1: System description," *Applied Soft Computing*, vol. 13, no. 1, pp. 259–270, 2013.

[12] W. Yang and J. Jiang, "Wind turbine condition monitoring and reliability analysis by scada information," in *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*. IEEE, 2011, pp. 1872–1875.

[13] M. Lydia, A. I. Selvakumar, S. S. Kumar, and G. E. P. Kumar, "Advanced algorithms for wind turbine power curve modeling," *Sustainable Energy, IEEE Transactions on*, vol. 4, no. 3, pp. 827–835, 2013.

[14] A. Kusiak, H. Zheng, and Z. Song, "On-line monitoring of power curves," *Renewable Energy*, vol. 34, no. 6, pp. 1487–1493, 2009.

[15] M. Schlechtingen and I. F. Santos, "Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection," *Mechanical systems and signal processing*, vol. 25, no. 5, pp. 1849–1875, 2011.

[16] S. Li, D. C. Wunsch, E. OHair, and M. G. Giesselmann, "Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation," *Journal of Solar Energy Engineering*, vol. 123, no. 4, pp. 327–332, 2001.

[17] S.-Y. Kim, I.-H. Ra, and S.-H. Kim, "Design of wind turbine fault detection system based on performance curve," in *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*. IEEE, 2012, pp. 2033–2036.

[18] F. Pelletier, C. Masson, and A. Tahan, "Wind turbine power curve modelling using artificial neural network," *Renewable Energy*, vol. 89, pp. 207–214, 2016.

[19] D. Petković, Ž. Ćojbašič, and V. Nikolić, "Adaptive neuro-fuzzy approach for wind turbine power coefficient estimation," *Renewable and Sustainable Energy Reviews*, vol. 28, pp. 191–195, 2013.

[20] S. Heier, *Grid integration of wind energy conversion systems*. Wiley, 1998.

[21] A. Kusiak, H. Zheng, and Z. Song, "Models for monitoring wind farm power," *Renewable Energy*, vol. 34, no. 3, pp. 583–590, 2009.

[22] K. Swingler, *Applying neural networks: a practical guide*. Morgan Kaufmann, 1996.

[23] P. Caselitz and J. Giebhardt, "Advanced maintenance and repair for offshore wind farms using fault prediction techniques," in *World Wind Energy Conference*, 2002.

[24] J. Tang, G. Zhang, Y. Wang, H. Wang, and F. Liu, "A hybrid approach to integrate fuzzy c-means based imputation method with genetic algorithm for missing traffic volume data estimation," *Transportation Research Part C: Emerging Technologies*, vol. 51, pp. 29–40, 2015.

[25] W.-K. Ching, L. Li, N.-K. Tsing, C.-W. Tai, T.-W. Ng, A. Wong, and K.-W. Cheng, "A weighted local least squares imputation method for missing value estimation in microarray gene expression data," *International journal of data mining and bioinformatics*, vol. 4, no. 3, pp. 331–347, 2010.

[26] S. Wang, Y. Huang, L. Li, and C. Liu, "Wind turbines abnormality detection through analysis of wind farm power curves," *Measurement*, vol. 93, pp. 178–188, 2016.

[27] E. Taslimi-Renani, M. Modiri-Delshad, M. F. M. Elias, and N. A. Rahim, "Development of an enhanced parametric model for wind turbine power curve," *Applied Energy*, vol. 177, pp. 544–552, 2016.

[28] M. Ghaemi and M.-R. Feizi-Derakhshi, "Feature selection using forest optimization algorithm," *Pattern Recognition*, vol. 60, pp. 121–129, 2016.

[29] Y. Wan, M. Wang, Z. Ye, and X. Lai, "A feature selection method based on modified binary coded ant colony optimization algorithm," *Applied Soft Computing*, vol. 49, pp. 248–258, 2016.

[30] S. Khokhar, A. A. M. Zin, A. P. Memon, and A. S. Mokhtar, "A new optimal feature selection algorithm for classification of power quality disturbances using discrete wavelet transform and probabilistic neural network," *Measurement*, vol. 95, pp. 246–259, 2017.

[31] Y. Wang, J. Wang, H. Liao, and H. Chen, "An efficient semi-supervised representatives feature selection algorithm based on information theory," *Pattern Recognition*, vol. 61, pp. 511–523, 2017.

[32] P. Sun, J. Li, Y. Yan, X. Lei, and X. Zhang, "Wind turbine anomaly detection using normal behavior models based on scada data," in *High Voltage Engineering and Application (ICHVE), 2014 International Conference on*. IEEE, 2014, pp. 1–4.

[33] M. Schlechtingen, I. F. Santos, and S. Achiche, "Using data-mining approaches for wind turbine power curve monitoring: a comparative study," *Sustainable Energy, IEEE Transactions on*, vol. 4, no. 3, pp. 671–679, 2013.

[34] H. Zhou, Z. Deng, Y. Xia, and M. Fu, "A new sampling method in particle filter based on pearson correlation coefficient," *Neurocomputing*, vol. 216, pp. 208–215, 2016.

[35] W. Xu, R. Ma, Y. Zhou, S. Peng, and Y. Hou, "Asymptotic properties of pearson's rank-variate correlation coefficient in bivariate normal model," *Signal Processing*, vol. 119, pp. 190–202, 2016.

[36] K. O. Akande, T. O. Owolabi, and S. O. Olatunji, "Investigating the effect of correlation-based feature selection on the performance of support vector machines in reservoir characterization," *Journal of Natural Gas Science and Engineering*, vol. 22, pp. 515–522, 2015.

[37] K. Salahshoor, M. Kordestani, and M. S. Khoshro, "Design of online soft sensors

based on combined adaptive pca and dmlp neural networks," in *2009 Chinese Control and Decision Conference*. IEEE, 2009, pp. 3481–3486.

[38] ——, "Design of online soft sensors based on combined adaptive pca and rbf neural networks," in *2009 IEEE Symposium on Computational Intelligence in Control and Automation*. IEEE, 2009, pp. 89–95.

[39] H. Jafarnejadsani, J. Pieper, and J. Ehlers, "Adaptive control of a variable-speed variable-pitch wind turbine using radial-basis function neural network," *Control Systems Technology, IEEE Transactions on*, vol. 21, no. 6, pp. 2264–2272, 2013.

[40] R. Wagner, M. Courtney, T. J. Larsen, and U. Schmidt Paulsen, *Simulation of shear and turbulence impact on wind turbine performance*. Danmarks Tekniske Universitet, Risø Nationallaboratoriet for Bæredygtig Energi, 2010.

CHAPTER 3

Applying ANFIS to Predict Wind Turbine Power Production

3.1 Introduction

Power production by wind energy is rapidly growing, thanks to continuous reduction in costs. Remaining challenges include the enhancement of turbine performance, management of operations and reducing maintenance costs. Ongoing research is focused on increasing wind energy production capabilities, reliability and general reduction of financial risk in wind farm investment. Providing reasonably accurate prediction of the future performance of turbines will also help operators prognosticate the performance of machines and deliver their contractual obligations.

Various types of analysis have been utilized to further advance knowledge of wind turbine performance and optimization [1, 2]. They can be categorized into two general approaches; physics-based models and data driven approaches. The existing complex, nonlinear relationship between subsystems of a wind turbine highlights the challenges ahead for wind turbine physics-based modeling; which include numerical methods [3, 4, 5, 6] and laboratory experimental approaches [7, 8, 9]. Since data driven techniques analyze available machine historical performance data, and therefore bypass some of the issues associated with physics-based models, they have garnered interest as an alternative.

Consequently, data driven methods are well-positioned for analyzing wind turbines, owing to two major advantages. First, most wind farms have access to comprehensive Supervisory Control And Data Acquisition (SCADA) data. Secondly, the execution of these methods is often less costly than comparable numerical or experimental approaches [10, 11].

It must be noted, however, that data driven models generally give less accurate results compared to situations where a mathematical model of a system is available. But this is generally a very rare situation, and considering the complexity of wind turbines and the fact that they include large amount of uncertainty, accurate mathematical modeling of these systems is challenging and impractical.

There is currently a broad spectrum of popular data driven methods. The most common intelligent methods include Neural Networks (NNs) [12, 2, 13, 14, 15], Adaptive Neuro-Fuzzy Inference System (ANFIS) [16, 11, 17], Support Vector Machines (SVM) [18, 19], regression methods [20, 21, 22, 23] and Probability Density Functions (PDF) [24, 25, 26, 27, 28]. Each has their own advantages and limitations, depending on the application. There are other approaches that require expert knowledge of the system and are difficult to employ due to challenges associated with translating the expert knowledge to the model. The most applied methods in this category include Fuzzy Logic (FL) [29, 30], Bayesian analysis [31, 32], Markov model [33, 34] and signal processing methods [35, 36, 37].

Two very common techniques that have been extensively utilized in the area of performance prediction and fault diagnosis are neural networks and fuzzy logic [12, 2, 29, 30]. NNs benefit from computational tools that can be applied for estimation of non-linear systems. Their training is based on input and output data; which provide non-linear mappings from the system input to output. These networks are typically considered black boxes, given that most users have little interaction with the inner workings of the model. On the other hand, fuzzy logic techniques follow a rule structure that is easier to understand or alter. They have proven to be effective in the field of performance estimation and pattern recognition. However, they require expert knowledge of the system [38]. ANFIS models which are considered grey box models, possess acceptable numerical accuracy while providing meaningful interpretation by combining neural networks and fuzzy logic.

To initialize any data driven approach, the data must be prepared to be utilized by the selected technique. One necessary step is to solve the issues related to missing values. In many studies conducted on SCADA data, the missing values were removed [39, 40, 15, 13] while many algorithms have been developed to otherwise address this issue and generate a smoother dataset. These algorithms perform differently based on the application, dataset, and missing values pattern. Some of the more frequently used algorithms include multi objective genetic algorithm [41], fuzzy c-means clustering [42], decision tree algorithm [43], expectation maximization [43], multilayer perceptron [44], support vector regression [45], self-organizing map [46], and local least square imputation [47].

Numerous research efforts have been made to estimate the power output of turbines applying various techniques. Kusiac et al. [15] compared non-linear parametric models with k-NN method as a performance monitoring tool. Many studies [13, 40, 15, 48] applied neural networks for estimating power production concluding that this method is powerful and suitable to fulfill its purpose. ANFIS was also applied in [40] and [14] for power and power coefficient estimation, respectively.

The work introduced in this paper mainly focuses on the prediction of wind turbine future power production. For this purpose, input parameters are selected based on both physical and statistical evidence of influence of each signal in the SCADA data, on power production. Then a dynamic ANFIS model is constructed to predict future performance of turbines. In addition, to more appropriately train the network, an imputation algorithm is applied to replace missing values and outliers with imputed values.

The main originality of this chapter is in the way that the network is designed. The novel application of decision tree concept to substitute missing values improves the dataset quality. Moreover, the aggregation of the proposed feature selection method and dynamic ANFIS model creates an effective tool to predict future performance of the turbines. Simulation results based on the field measurements show that the combination of all these methods gives rise to an acceptable error margin. For the purpose of network evaluation, Mean Absolute Error (MAE) is employed.

A brief system description is presented in Section 3.2. Turbines description and dataset preparation are explained in Section 3.3. The applied imputation algorithm is detailed in this section. Section 3.4 introduces the modeling design procedure consisting of feature extraction, ANFIS networks and performance estimation and prediction. Simulation and test results are presented in Section 4.6. Finally, Section 4.7 concludes with a summary of

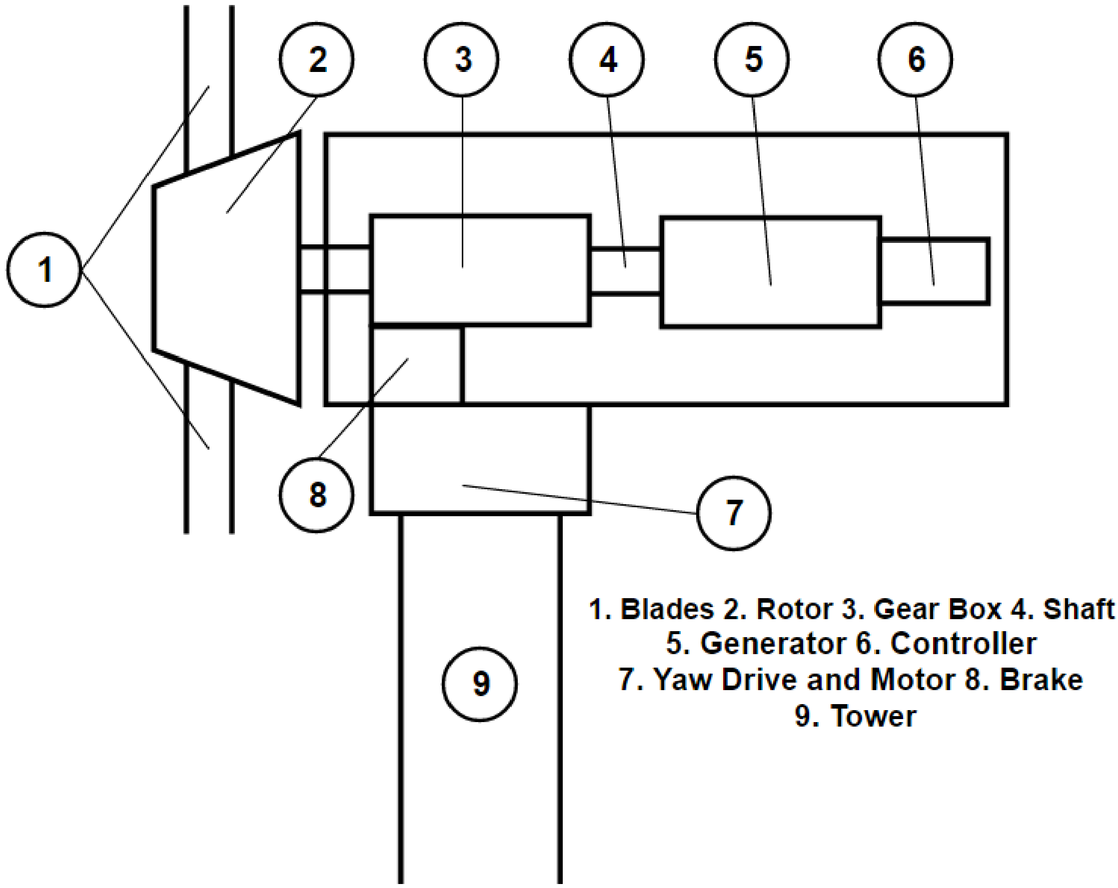

FIGURE 3.2.1: Different Components of a Horizontal Axis Wind Turbine

the results.

3.2 System Description

Wind turbines are mechanical devices designed to harvest kinetic energy from wind and convert it into electrical power. They are comprised of many parts that create a complex system in terms of mathematical modeling. A schematic of location of various parts of a horizontal axis, pitch regulated wind turbine, is illustrated in Figure 4.2.1. Theoretically, the power generated by the wind turbine is calculated using the following equation

$$P = \frac{1}{2}\rho A C_p V^3 \tag{3.2.1}$$

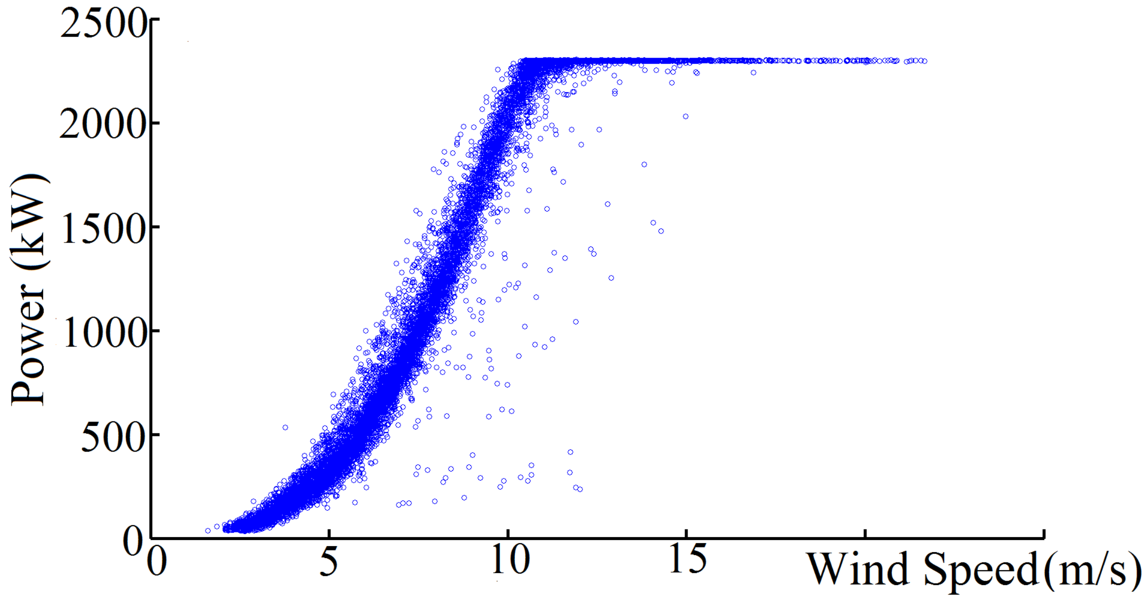

FIGURE 3.2.2: Power Curve of an Investigated Turbine

where ρ is the air density, A is the rotor sweep area, V is the wind speed, and C_p is the power coefficient. This coefficient is the ratio of actual electric power produced by a wind turbine divided by the total wind power flowing into the turbine blades at specific wind speed. It can be obtained by expression proposed by Heier [23].

However, the power curve of a commercialized wind turbine, which illustrates active power in different wind speeds, does not follow the same trend as equation 4.2.1. A sample power curve from one of the turbines investigated in this study is shown in Figure 4.2.2. There are three distinct regions in the power curve. Wind turbines do not start operating until the wind speed reaches a minimum value called cut-in speed. For lower speeds, it is not economical to start the turbines. In the second region, as the wind speed increases, the output power also grows rapidly until it reaches the rated power. Finally in the third region, the power output remains constant. This region ends when the wind speed exceeds the maximum cut-out speed beyond which turbines blades are regulated to not rotate and brakes are activated to prevent mechanical failures. The values for cut-in and cut-out speeds vary depending on turbine size and rating, but it is normally between $2 \sim 4$ m/s and $22 \sim 25$ m/s for cut-in and cut-out speeds, respectively.

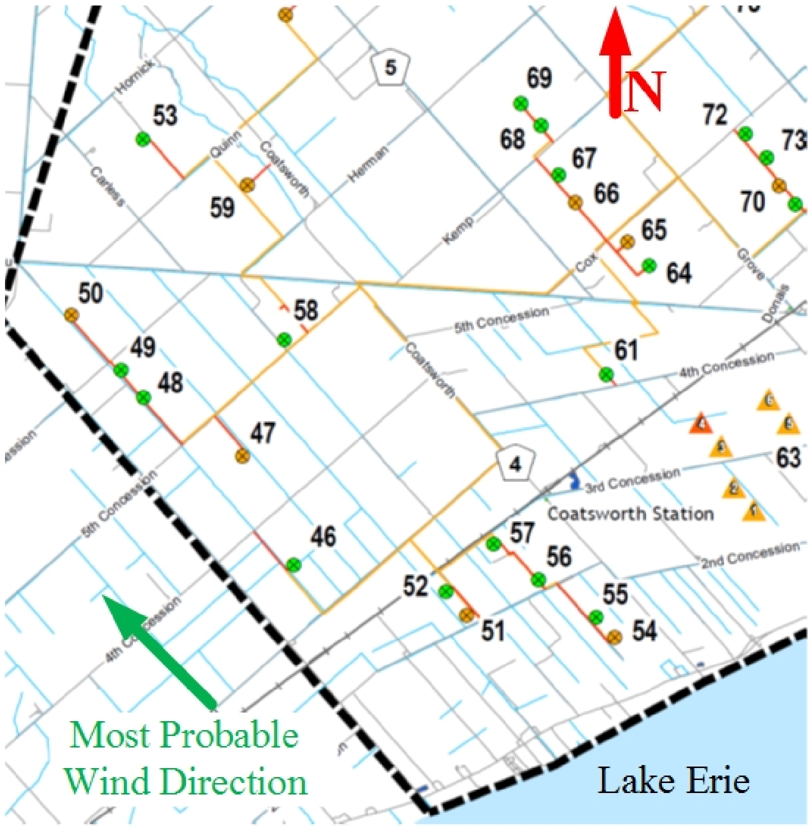

FIGURE 3.3.1: Turbines Layout

3.3 Dataset Preparation

In this study, twenty-one, 2.3 MW, pitch-regulated wind turbines have been investigated. These turbines are located in a wind farm in Ontario, Canada, and the dataset obtained from SCADA system, cover a range of 20 months. The layout of the turbines is shown in Figure 3.3.1. Data collection systems are not perfect and the obtained data nearly always contain missing, incorrect, and noisy values often the result of sensor malfunction. The outcome of data mining algorithms can be significantly affected by such values thus making data pre-processing absolutely essential [43, 49, 50, 51].

To carry out the pre-processing in the dataset under investigation, the following steps are taken:

1. Accuracy check, 2. Missing value labeling, 3. Data range scaling

The sequential execution of these steps are explained in the following sections:

3.3.1 Accuracy Check

This first step is an investigation of the sensors performance; which is done by checking the data range. At this stage, outliers are removed and a smoother data set is generated. It is important to carefully choose the range to avoid classification of real measured values as outliers. This can be the reason for not detecting the abnormal behavior of turbines and is particularly challenging for temperature values of components or lubricants. This stage of pre-processing ensures that the data range is as expected, mutual consistency of components is met, and finally, input and output vectors are consistent. This is especially critical for supervised learning applications [48, 52, 53].

In the existing studies conducted on the SCADA data of wind turbines, outliers of the entire dataset are removed. In this paper, however, they are treated as missing values as explained in the following section.

3.3.2 Missing Value Labeling

Having some missing values in such a large data collection system is not uncommon. These unknown values can negatively impact the value of the data analysis. Data mining algorithms and techniques are not designed to directly handle these values and they are particularly sensitive to them during the training phase. For instance, if a supervised learning tool tries to fit a curve to these values, the generalization error will drastically increase [54]. Subsequently, it is imperative to highlight them and clarify how they should be treated [55].

In the studies conducted on SCADA data, it was common to neglect the whole record of data when one value was missing. Due to large amount of data available, it was stated that there was no need for estimation. This may not be the best approach, since useful information can be lost [54, 56]. It should be highlighted that there can be more than 150 signals in a turbine SCADA dataset where each signal is designated as a column. So, even if only one signal is missing, the others would also be deleted by this approach. In addition, obtaining consecutive historical data of system performance is critical in data-driven analyses. Thus, ignoring the entire instance because one sensor is not functioning properly, damages the reliability of network outcomes.

Observation of sudden changes in the wind pattern as a major factor in power production is customary. So, it is necessary to have access to continuous historical data to thoroughly investigate a turbine performance in all conditions. This is even more important for performance prediction. Thus, the replacement of missing values (in both situations of actual missing values and outlier removal) with plausible data is necessary. This process is called data imputation [57]. This process can be carried out by applying both statistical learning theory and machine learning tools [58]. Selection of the most appropriate method depends on the application, type and size of dataset, number and pattern of missing values, maximum acceptable imputation error, and computation capacity.

The simplest manner of imputation is to replace the missing value with the mean or mode of each attribute. This works best with small datasets when data range is not very wide. For SCADA data, however, it may be more suitable to employ more advanced techniques, since it includes a very large number of instances.

The other common method is K-Nearest Neighbor (K-NN) technique [59]. In this technique, to impute the missing value from the ith instance and jth attribute (r_{ij}) from a dataset D_f, k number of the most similar instances in D_f are found. This similarity measurement is done by utilization of various distance functions such as Euclidean, Manhattan or Minkowski according to Equations 3.3.1, 3.3.2, and 3.3.3

$$\sqrt{\sum_{i=1}^{k} (x_i - y_i)^2} \tag{3.3.1}$$

$$\sum_{i=1}^{k} |x_i - y_i| \tag{3.3.2}$$

$$\left[\sum_{i=1}^{k} (|x_i - y_i|)^q\right]^{\frac{1}{q}} \tag{3.3.3}$$

where x_i and y_i are the points in the space and k represents the number of points.

When the neighbors are found, the algorithm replaces the missing value of each attribute with the mean value of that same attribute from the recognized neighbors. The main advantage of this technique is its simplicity. The major drawback for this method is that it needs to search through the entire dataset to find the neighbors. This can be computationally expensive for large datasets such as SCADA data.

The other powerful algorithm for imputation is Expectation Maximization (EM) algorithm; which has been applied in several studies [43, 60]. In this method, regression parameters in incomplete attributes are calculated for complete datasets (where their missing values are removed) with available values from estimation of mean and covariance matrix. Afterwards a conditional expectation value replaces the missing ones and finally covariance matrix will be re-estimated to optimize the imputed value. After several iterations when there is none, or a negligible update in imputed values, the process is finished and the completed dataset is generated . Although this method usually returns acceptable imputation errors, the iterative pattern of this method consumes non-trivial quantities of time and memory for large datasets.

There must always be a reasonable balance between the complexity of applied algorithm, which determines the outcome error, and its simplicity of implementation. Many aspects affect the path to reach this goal. For analysis of SCADA data, in this paper, a combination of decision trees and mean value has been applied to efficiently create this balance.

Decision trees split data sets into smaller parts and place them in leaf nodes. It is performed in a top-down manner from the root node. In the beginning, all the training instances are at the root. They are then partitioned into subsets according to the selected attributes. This selection can be based on a variety of statistical measures. One of the most popular ones, is information gain, which is itself based on the concept of Gibbs entropy as expressed in Equations 4.3.1 and 4.3.2.

$$E(T) = \sum_{i=1}^{c} -p_i log_2 p_i \tag{3.3.4}$$

$$E(T, X) = \sum_{c \in X} p(c)E(c) \tag{3.3.5}$$

where E is the entropy, c is the number of possible outcomes, and p_i is the possibility of each outcome. T and X represent the features in the dataset, where T is the target feature.

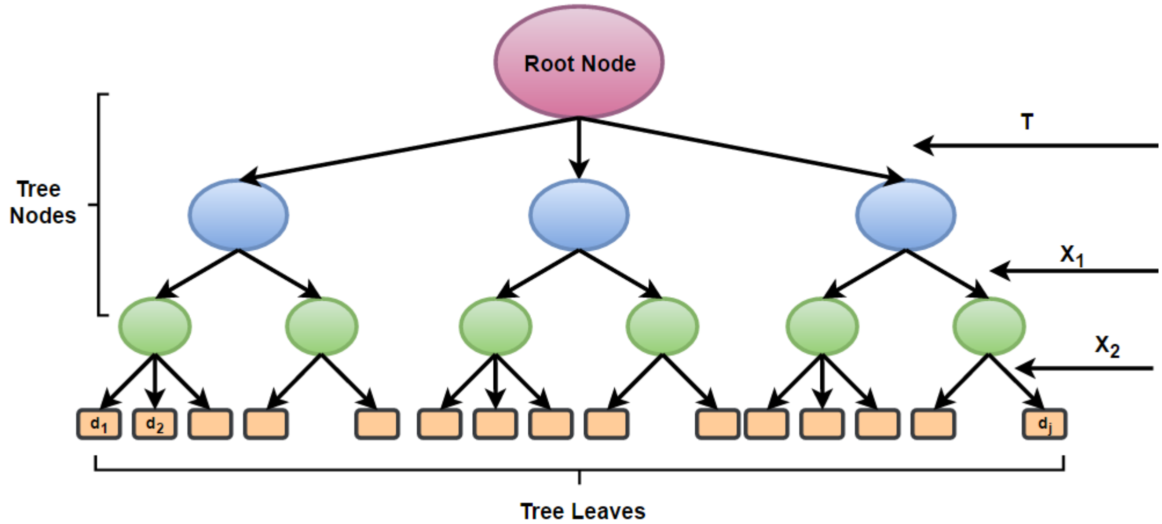

FIGURE 3.3.2: Schematic of a Decision Tree

Figure 3.3.2 illustrates a schematic of a decision tree when the data is divided into different groups by the target feature of T, and then, divided further into smaller sets, applying different ranges in two features of X_1 and X_2. The last feature applied to split the dataset, in this figure, X_2, is called the class attribute. Sub datasets of d_1, d_2,... ,d_j include the data points placed in each leaf. The imputation procedure of this approach is divided in the following steps [43] as also summarized in Figure 3.3.3.

Step 1: The missing values in the dataset are identified. Then the data is separated into two datasets of complete (D_c) and incomplete (D_i) in a way that all instances with missing values are in D_i.

Step 2: The target is to create sub datasets of d_j consisting of the data in each leaf in each decision tree. To do so, the numerical values are generalized for building the trees. The number of groups to divide the numerical values is the root square of the domain size of that feature. Next, the features in which they include missing values are identified and then a set of decision trees (based on D_c) is made considering each of those features with missing values as the class attributes.

Step 3: Sub datasets of d_j are created where only complete records are included. It is again emphasized that these data are those in each leaf, so the number of sub datasets of d_j is equal to the number of leaves. At this stage, the aim is to add the instances with missing

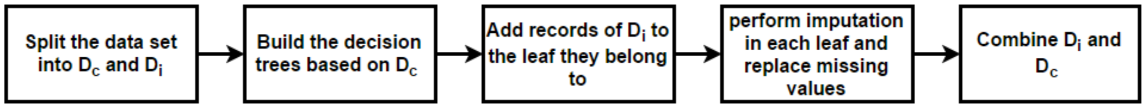

FIGURE 3.3.3: Imputation Procedure

values to the leaf they belong to. For this purpose, after finding each missing value, its respective feature is identified. Then, the decision tree; which has that particular feature of the class attribute, is selected, and the target leaf for that record is predicted. That record is then added to that leaf sub dataset (d_j). If one record has more than one value missing, that record will be added to more than one leaf, since different trees have been used to predict the target leaf.

Step 4: The imputation is performed. First, each feature with a missing value in each instance in D_i is identified and then the sub dataset, where that record was added to, is recognized. The imputation is executed using mean value of each feature in the leaf. Then, the missing value is replaced by the imputed one in D_i. Finally, D_c and D_i are combined together to form the completed dataset.

3.3.3 Data range scaling

When applying multiple inputs to train a model, it is essential to have all variables in the same range. Otherwise, the one with the wider range will dominate during training, over-shadowing possibly more influential parameters. The most common method for scaling is presented in Equation 3.3.6.

$$N = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{3.3.6}$$

where N is the normalized variable and X is the variable.

3.4 Modeling Design Procedure

This section introduces the design methodology of the proposed prediction method. The basic idea is to develop an intelligent predictor to fully track the power curve of the wind turbines using dynamic ANFIS networks. For this, at first, the feature extraction method is explained. Then, the ANFIS is presented, followed by clarification of estimation and prediction concepts, and how networks are defined in each situation.

3.4.1 Feature Extraction

There are substantial number of tracked variables in wind turbine SCADA systems. Taking all of them into consideration for performance monitoring or prediction is not practical and may not be helpful. Selection of appropriate features initially depends on the desired modeling type and investigated target. It is very common to choose parameters solely based on physical understanding of systems [61]. In this approach, features are selected on the basis of conceptual relevance and influence on the wind turbines power production according to Equation 4.2.1

Based on common sense and Equation 4.2.1, which demonstrates that wind speed is the most influential factor, many researchers considered it as the only input parameter [39, 13]. However, it has become apparent that other factors also need to be utilized to create more reliable models. Those factors may include environmental aspects such as wind direction or ambient temperature as argued by Schlechtingen et al. [40]. Other factors; which may initially seem irrelevant, can potentially play a positive role in the establishment of reliable models. In addition, it should be noted that machine learning algorithms will not perform well when the number of input parameters is irrationally high. Therefore, to find the optimum number of input parameters, statistical correlation method is applied here in addition to system physical understanding. Correlation coefficients of variables (r) are calculated by means of the following equation:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{3.4.1}$$

where x_i and y_i are the two variables that are investigated for correlation which in this study, can be any of the input parameters and the output power, n is the number of each variable and $\bar{x}$ and $\bar{y}$ are the mean values. This coefficient, which has a value from -1 to 1, shows the strength of relationship between two variables. Absolute value of 1 indicates perfect correlation while zero means no correlation. Negative values show reverse relationship [61]. Greater value of r for each signal means that parameter has more influence on power and is a more suitable input candidate.

3.4.2 Adaptive Neuro-Fuzzy Inference System

Application of ANFIS models in wind turbines power curve monitoring was first introduced by Schlechtingen et al. [40] in 2013. The ANFIS method was developed to be able to benefit from the best of both fuzzy logic and neural networks. These models, which were first introduced by Takagi and Hayashi [62], are a combination of fuzzy logic and neural networks.

In fuzzy logic, membership functions are employed to provide a mechanism of inference. Two very common types of fuzzy inference have been proposed by Mamdani [63] and Sugeno [64]. In the Mamdani type, the consequent part is a fuzzy linguistic value while in Sugeno, it is a non-fuzzy equation. For the purpose of wind turbines power prediction, the Sugeno type is an acceptable choice since it is proven to be computationally efficient, generates continuity of output surface, and a non-fuzzy equation can represent the consequent part.

According to Sugeno fuzzy model, a sample set of rules can be constructed as follows:

Rule 1: If x_1 is A_1 and x_2 is B_1, then $f_1 = p_1x_1 + q_1x_2 + r_1$

Rule 2: If x_1 is A_2 and x_2 is B_2, then $f_1 = p_2x_1 + q_2x_2 + r_2$

where p_i, q_i and r_i are adaptive parameters and A_i and B_i define the membership functions. This precise mathematical modeling in fuzzy logic requires an expert knowledge of the system, it is normally very time consuming and computationally challenging task to perform. On the other hand, neural networks can handle a variety of non-linear performance estimation, prediction, and even pattern recognition with fast computational abilities, but no inference capability. ANFIS models create a computationally efficient environment while providing an inference mechanism. As a result, ANFIS networks can process the data faster without the need of expert knowledge.

Figure 3.4.1 represents a typical ANFIS structure with two inputs and one output. An

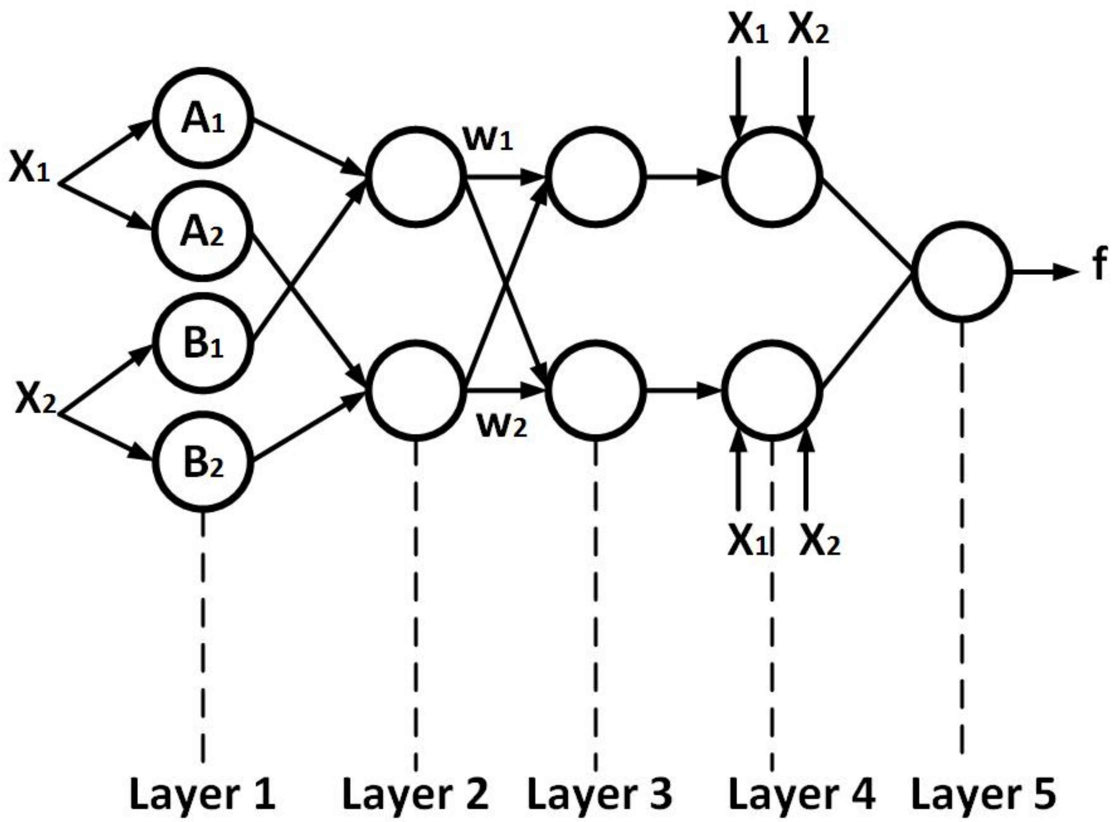

FIGURE 3.4.1: A Typical ANFIS Network Structure

ANFIS network typically consists of five layers with a feed forward structure. Each layer is formulated by mathematical functions as below [65]:

1. In this layer, the adaptive nodes have linguistic labels and the output is the membership function of that label:

$$OL1_i = \mu_{A_i}(x_1) \tag{3.4.2}$$

$$OL1_i = \mu_{B_i}(x_2) \tag{3.4.3}$$

2. In this layer, nodes are fixed. The firing strength of each rule (w_i) is calculated by each node by means of multiplication of incoming signals:

$$OL2_i = w_i = \mu_{A_i}(x_1)\mu_{B_i}(x_2) \tag{3.4.4}$$

3. Again in this layer, nodes are fixed. The result of the previous layer is normalized by each node in this layer. The outcomes which are called relative firing strength ($\bar{w_i}$) are calculated as below:

$$OL3_i = \bar{w_i} = \frac{w_i}{\sum_{j=1}^{i} w_i} \tag{3.4.5}$$

4. Similar to layer 1, nodes are adaptive in this layer. Multiplication of previous layer output (relative firing strength) and the adaptive parameters, are outputs of this layers nodes:

$$OL4_i = \bar{w_i} f_i = \bar{w_i}(p_i x_1 + q_i x_2 + r_i) \tag{3.4.6}$$

5. When having only one output, this layer has one node. The output of this layer is the summation of all signals resulted from the previous layer:

$$OL5_i = \sum_{i=1}^{j} \bar{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{3.4.7}$$

As noted in [65], various ANFIS structures can be obtained by combining those layers.

To train the networks, hybrid or back propagation method is applied. This includes gradient descent and least square methods which determine the optimum values of adaptive parameters in layers 1 and 4. This is done to minimize the network output error.

The dataset is divided into two sets; the training and test data. In the process of training, the training algorithm keeps operating until the desired performance is achieved in which the calculated weights are saved. Otherwise it continues to reach the previously set value for maximum number of epochs. Finally, the trained network is validated by the test data and the network test error is calculated.

3.4.3 Estimation and Prediction

In any data driven modeling, networks can be built for two different purposes. Either estimation or monitoring of the output is targeted, or the model is constructed to predict the future output value based on the current history of the machine. Defining the datasets for network training depends on whether it is developed for monitoring or prediction. In monitoring, the target is to estimate the output value in the time when the input signals are available. In other words, at any time during a machine performance, the network is trained to return the output value at time t when it is trained by input parameters at the same time t. The mathematical representation of the monitoring is described as below:

$$Y(t) = f(u_1(t), u_2(t), ..., u_i(t)) \tag{3.4.8}$$

where Y and u_i represent the output and input parameters, respectively.

In contrast to monitoring, when a network is designed to make a prediction of future performance of a machine, input and output signals are not acquired at the same time. In this case, based on the available input signals at time t, network is designed to predict the output at the time $t + M$. The value of M depends on the frequency in which the data have been acquired and also how far into the future the network is supposed to make the prediction. Similarly, the mathematical expression for prediction is presented below:

$$Y(t + M) = f(u_1(t), u_2(t), ..., u_i(t)) \tag{3.4.9}$$

It should be noted that the term prediction has been used in many studies conducted on wind turbines power curve where the intention was to provide an estimation of output power at the same time when the values of input signals were gathered. Although in these situations, the established networks were predicting a value for the power, the system was just being monitored and not predicted. The actual concept of prediction is fulfilled only when the power at the time $t+M$ is given by the network while the inputs at the time t were provided for training.

To determine networks performance Mean absolute Error (MAE), the following formula is employed:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \tag{3.4.10}$$

where x_i is the predicted value by the ANFIS and y_i is the actual value. MAE indicates the closeness of predicted results with true values and thus creates an appropriate understanding of the model accuracy.

3.5 Simulations and Results

In this section, results of the applied imputation technique, feature extraction analysis and performance of various established ANFIS models will be presented.

3.5.1 Imputation Validation

Before training the networks, imputation must be carried out as part of the data preprocessing stage. In order to apply the described imputation algorithm including the combination of decision tree and average value on the dataset, this method needs to be validated to ensure its suitability for SCADA data. For this purpose, a sub dataset of SCADA system which contains no missing values is selected. Then some random values from a variety of features and records are removed to create a dataset containing missing values, where in reality the true values are known. The dataset was created in such a way that it contained 5 percent of missing values, roughly the same proportion as observed amount of missing values, outliers and out-of-range signals in SCADA data. The imputation is then performed on this sub dataset to compare the imputed values with the true values from the complete dataset. Normalized Root Mean Square (NRMS) error was applied to calculate the im-

TABLE 3.5.1: Correlation Coefficients of the selected signals

Parameter	Pearson product-moment
Wind Speed	0.939
Rotor Speed	0.926
Gear Temperature	0.893
Blade Pitch Angle	0.663
Blade Pitch Angle SD	0.582
Rotor Speed SD	-0.294
Gear Temperature SD	-0.273
Turbulence Intensity	0.185
Active Power SD	0.114
Wind Direction SD	-0.091
Wind Direction	0.003

putation error. This is the most applied error type to investigate the imputation results in numerical datasets. NRMS is calculated according to the following formula:

$$NRMS = \frac{\left\|X^{estimate} - X^{original}\right\|}{\left\|X^{original}\right\|} \tag{3.5.1}$$

$$\|A\| = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}|a_{ij}|^2} \tag{3.5.2}$$

Where $X^{estimate}$ is the imputed value by the algorithm and $X^{original}$ is the true value. Norm of a matrix is also calculated based on Equation 4.6.2. After imputation, the $NRMS$ is calculated to be 0.0306. This is a very low error which clearly validates the implemented procedure for imputation, and confirms that it can also be trusted in real application when actual values are unknown.

The networks established based on the pre-processed dataset are described in the following section.

3.5.2 Feature Extraction Analysis

Features to be considered as potential input parameters and their correlation coefficient with active power are summarized in Table 4.6.1 . This selection is made based on available data, physical understanding of wind turbines and the most common parameters selected in the literature. It is confirmed in Table 4.6.1 that wind speed, which is expected to have the strongest correlation with output power in comparison with other signals, has the closest number to 1 with the coefficient value of 0.939.

In this study, dynamic networks are created which means that the latest available output power, which is at the time $t-1$, is also defined as an input parameter. The other input signals are selected based on the results shown in Table 4.6.1. These parameters are wind speed, rotor speed and gear temperature.

3.5.3 ANFIS Networks Structure

In order to establish the ANFIS networks, the following items must be determined:

1. Type of Membership Functions (MFs):

Various functions can be applied for input and output parameters to serve as membership functions. MFs are basically arbitrary curves that depend on the kind of application for which they are utilized. For this study, the Gaussian function is employed for input parameters as it is proven to work best for most applications which also generates flexibility, and for output parameter, the linear function is applied.

2. Number of Membership Functions (MFs):

Unlike modeling by fuzzy inference system in which, the number of MFs is determined by a system expert [66], in ANFIS structure, this number can be found by investigating the networks error. Theoretically, increasing the number of MFs will decrease the error until it reaches its minimum value and beyond that, further increase in the number of MFs will result in higher errors. On the other hand, the larger the number of MFs, the longer the computational time. Thus, to determine the optimum number, the error of each network is calculated using different number of MFs and when the error stops to considerably decrease, that number is set to be the number of MFs [40]. It is also worth noting that for each input, a different number of MFs can be defined [10]. In this study, three MFs for each input is applied.

3. Input and output parameters

Since our concern here is prediction of power production, active power is chosen. There are two general structures for input parameters. In the first structure, a static network can be established where the chosen signals in the feature extraction section can be utilized. The other way is to create a recurrent network by including previous values of active power as a separate input. It is important to emphasize that in the actual application, when objective is to predict the active power at the time $t + M$, the value of power at the time t is unknown. The latest known value of power to be used as an input is at the previous time $t - 1$. Based on that, the training dataset is defined as below:

$A = [u_i(2 : end - M, j); Y(1 : end - (M + 1), k)]$

$B = Y(M + 2 : end, k)$

where A and B are training datasets for inputs and output, respectively. Also, j and k represent the column number of inputs and outputs in the dataset. This means that for static terms of inputs (u_i), training dataset starts from the second data point and ends with the point that has M distance from the last point. For the recurrent term ($Y4$), training dataset starts from the first point and ends with the point that has $M + 1$ distance from the last point. For the test dataset, however, it starts from the point that has $M + 2$ distance from the first point and ends with the last point. The reason for this type of definition is twofold; a)Matrices dimension must be consistent and b) In order to apply all records in training phase, training dataset for input features must start from the second data point. Here, the first data used for Y is at $t - 1$ when $t = 2$.

3.5.4 Data Obtaining frequency

In order to make a prediction that has pragmatic utility for wind farms, it is important to define values of M in a way that, there is sufficient time for operators to take necessary action when required. This implies that the frequency in which the data points are gathered determines how far in the future, the prediction is made. For this study, a dataset with one hour time intervals was created. In other words, when $M = 1$, the prediction is for the average power production in the next hour, and $M = 2$ shows prediction for the next two hours and so on.

Prediction for five different steps is carried out and MAE is calculated and also scaled according to the method proposed in [40]. This prediction is made in two situations. First, the network is trained using the dataset without performing imputation in which missing values are removed. Then, it is trained again employing the imputed dataset. Comparison

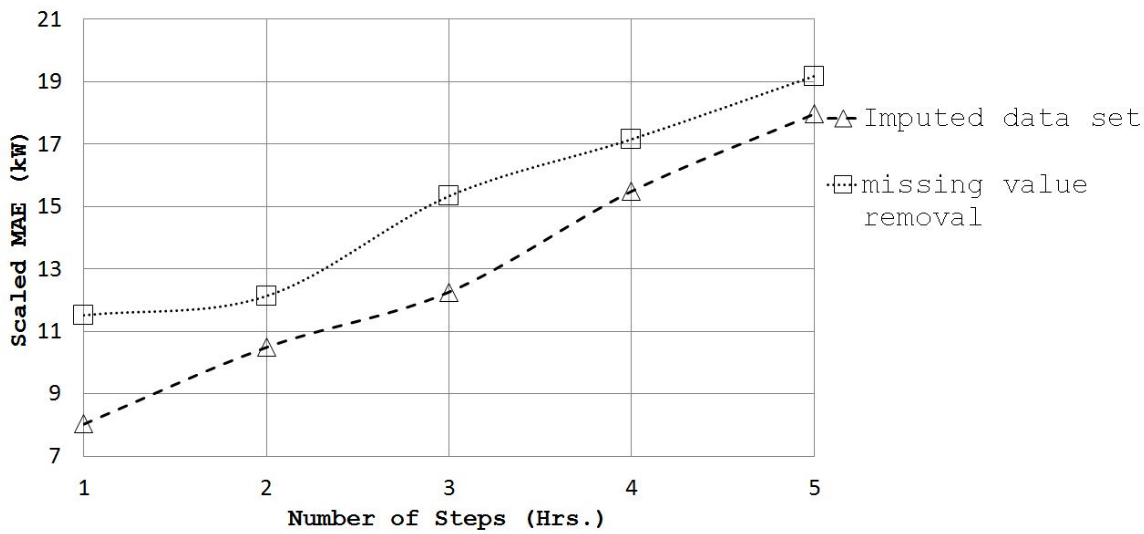

FIGURE 3.5.1: Test Errors of Datasets With and Without Imputation

of the results in these two cases, which is illustrated in Figure 3.5.1, shows the influence of imputation. As expected, by increasing the value of M (number of steps), the networks error increases. Depending on application and importance, prediction can be made for various number of steps in future. It is stressed that, the network reliability will decrease significantly with further increase in M, since the error rises.

In addition, it can be seen that performing imputation decreased the network error which proves that performing imputation on SCADA data generates smoother datasets that leads to better modeling. It should be emphasized that, the imputation algorithm proposed in this study, is not very complex, and thus, not a time consuming process with heavy computations. It is, therefore, a robust to reduce the network error by performing it prior to training. Even slightest improvement in prediction of future performance of wind turbines can be very beneficial to operators, since it can create better images of the farm productivity.

3.6 Conclusion

To further improve the profitability of commercial wind energy, it is important to maximize the accuracy of wind farm performance monitoring systems. To achieve this goal, a new methodology was herein proposed for performance prediction of wind turbines. The introduced technique was based on the combination of a feature selection method, imputation algorithm, and Adaptive Neuro-Fuzzy Inference System (ANFIS). This method was applied to investigate the power curves of 2.3 MW pitch regulated wind turbines.

It was shown how the correlation coefficient could properly identify the most significant features to be considered as the network inputs. It was also demonstrated that the result of this analysis could help improve the previously recognized method; which was based on a physical appreciation of the system. The proposed imputation algorithm was tested using a dataset containing 5 percent of missing values and the obtained results based on the NRMS verified the capabilities of this decision tree based method for replacing missing values in SCADA data.

The proposed structure for the ANFIS network was tested and resulted in favourably low errors for power production prediction. This clearly illustrated the potential of such method to accurately estimate a wind farm power generation in future hours. It was also shown that applying imputed dataset significantly reduced the network error over a dataset with the imputed values omitted. The results also indicated that prediction of further steps into the future would result in greater prediction error, which was anticipated.

The proposed condition monitoring system is capable of presenting realistic pictures of machine performance, reducing false alarms, and help improve the profits of wind farms by detection of underperformance and prevention of unexpected failures.

References

[1] A. Heng, S. Zhang, A. C. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges and opportunities," *Mechanical systems and signal processing*, vol. 23, no. 3, pp. 724–739, 2009.

[2] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703–711, 2011.

[3] Y. G. Heo, N. J. Choi, K. H. Choi, H. S. Ji, and K. C. Kim, "Cfd study on aerodynamic power output of a 110kw building augmented wind turbine," *Energy and Buildings*, vol. 129, pp. 162–173, 2016.

[4] A. M. El-Zahaby, A. Kabeel, S. Elsayed, and M. Obiaa, "Cfd analysis of flow fields for shrouded wind turbines diffuser model with different flange angles," *Alexandria Engineering Journal*, 2016.

[5] A. Kumar, A. Dwivedi, V. Paliwal, and P. P. Patil, "Free vibration analysis of al 2024 wind turbine blade designed for uttarakhand region based on fea," *Procedia Technology*, vol. 14, pp. 336–347, 2014.

[6] V. Raman, M. Drissi-Habti, L. Guillaumat, and A. Khadhour, "Numerical simulation analysis as a tool to identify areas of weakness in a turbine wind-blade and solutions for their reinforcement," *Composites Part B: Engineering*, vol. 103, pp. 23–39, 2016.

[7] C. Yin, Z. Zhang, Z. Wang, and H. Guo, "Numerical simulation and experimental validation of ultrasonic de-icing system for wind turbine blade," *Applied Acoustics*, vol. 114, pp. 19–26, 2016.

[8] J. Chen, Q. Wang, S. Zhang, P. Eecen, and F. Grasso, "A new direct design method of wind turbine airfoils and wind tunnel experiment," *Applied Mathematical Modelling*, vol. 40, no. 3, pp. 2002–2014, 2016.

[9] Q. Li, T. Maeda, Y. Kamada, J. Murata, K. Shimizu, T. Ogasawara, A. Nakai, and T. Kasuya, “Effect of solidity on aerodynamic forces around straight-bladed vertical axis wind turbine by wind tunnel experiments (depending on number of blades),” *Renewable Energy*, vol. 96, pp. 928–939, 2016.

[10] M. Schlechtingen, I. F. Santos, and S. Achiche, “Wind turbine condition monitoring based on scada data using normal behavior models. part 1: System description,” *Applied Soft Computing*, vol. 13, no. 1, pp. 259–270, 2013.

[11] W. Yang and J. Jiang, “Wind turbine condition monitoring and reliability analysis by scada information,” in *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*. IEEE, 2011, pp. 1872–1875.

[12] M. Djeziri, B. Ananou, and M. Ouladsine, “Data driven and model based fault prognosis applied to a mechatronic system,” in *Power Engineering, Energy and Electrical Drives (POWERENG), 2013 Fourth International Conference on*. IEEE, 2013, pp. 534–539.

[13] E. OHair and M. G. Giesselmann, “Comparative analysis of regression and artificial neural network models for wind turbine power curve estimation,” 2001.

[14] S.-Y. Kim, I.-H. Ra, and S.-H. Kim, “Design of wind turbine fault detection system based on performance curve,” in *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*. IEEE, 2012, pp. 2033–2036.

[15] A. Kusiak, H. Zheng, and Z. Song, “Models for monitoring wind farm power,” *Renewable Energy*, vol. 34, no. 3, pp. 583–590, 2009.

[16] G. Vachtsevanos, F. Lewis, M. Roemer, A. Hess, and B. Wu, “Intelligent fault diagnosis and prognosis for engineering systems,” *Usa 454p Isbn*, vol. 13, pp. 978–0, 2006.

[17] M. Mostafaei, H. Javadikia, and L. Naderloo, “Modeling the effects of ultrasound power and reactor dimension on the biodiesel production yield: Comparison of pre-

diction abilities between response surface methodology (rsm) and adaptive neuro-fuzzy inference system (anfis)," *Energy*, vol. 115, pp. 626–636, 2016.

[18] H.-E. Kim, A. C. Tan, J. Mathew, and B.-K. Choi, "Bearing fault prognosis based on health state probability estimation," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5200–5213, 2012.

[19] Z. Yin and J. Hou, "Recent advances on svm based fault diagnosis and process monitoring in complicated industrial processes," *Neurocomputing*, vol. 174, pp. 643–650, 2016.

[20] H. Kim, M. G. Na, and G. Heo, "Application of monitoring, diagnosis, and prognosis in thermal performance analysis for nuclear power plants," *Nuclear Engineering and Technology*, vol. 46, no. 6, pp. 737–752, 2014.

[21] G. Niu and B.-S. Yang, "Dempster–shafer regression for multi-step-ahead time-series prediction towards data-driven machinery prognosis," *Mechanical systems and signal processing*, vol. 23, no. 3, pp. 740–751, 2009.

[22] D. Petković, Ž. Ćojbašič, and V. Nikolić, "Adaptive neuro-fuzzy approach for wind turbine power coefficient estimation," *Renewable and Sustainable Energy Reviews*, vol. 28, pp. 191–195, 2013.

[23] S. Heier, *Grid integration of wind energy conversion systems*. Wiley, 1998.

[24] A. Lorton, M. Fouladirad, and A. Grall, "A methodology for probabilistic model-based prognosis," *European Journal of Operational Research*, vol. 225, no. 3, pp. 443–454, 2013.

[25] W. Wang, "A two-stage prognosis model in condition based maintenance," *European Journal of Operational Research*, vol. 182, no. 3, pp. 1177–1187, 2007.

[26] C. Chen, D. Brown, C. Sconyers, B. Zhang, G. Vachtsevanos, and M. E. Orchard, "An integrated architecture for fault diagnosis and failure prognosis of complex engineering systems," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9031–9040, 2012.

[27] J. Son, Q. Zhou, S. Zhou, X. Mao, and M. Salman, "Evaluation and comparison of mixed effects model based prognosis for hard failure," *IEEE Transactions on Reliability*, vol. 62, no. 2, pp. 379–394, 2013.

[28] S. Liu, Y. Wang, and F. Tian, "Prognosis of underground cable via online data-driven method with field data," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7786–7794, 2015.

[29] Z. Wang, C. Hu, W. Wang, X. Si, and Z. Zhou, "An off-online fuzzy modelling method for fault prognosis with an application," in *Prognostics and System Health Management (PHM), 2012 IEEE Conference on*. IEEE, 2012, pp. 1–7.

[30] M. Yu, D. Wang, M. Luo, and L. Huang, "Prognosis of hybrid systems with multiple incipient faults: Augmented global analytical redundancy relations approach," *IEEE Transactions on systems, man, and cybernetics-part A: systems and humans*, vol. 41, no. 3, pp. 540–551, 2011.

[31] J. W. Sheppard and M. A. Kaufman, "A bayesian approach to diagnosis and prognosis using built-in test," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 3, pp. 1003–1018, 2005.

[32] J. Hu, L. Zhang, L. Ma, and W. Liang, "An integrated safety prognosis model for complex system based on dynamic bayesian network and ant colony algorithm," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1431–1446, 2011.

[33] Q. Liu, M. Dong, W. Lv, X. Geng, and Y. Li, "A novel method using adaptive hidden semi-markov model for multi-sensor monitoring equipment health prognosis," *Mechanical Systems and Signal Processing*, vol. 64, pp. 217–232, 2015.

[34] M. Dong and D. He, "Hidden semi-markov model-based methodology for multi-sensor equipment health diagnosis and prognosis," *European Journal of Operational Research*, vol. 178, no. 3, pp. 858–878, 2007.

[35] R. K. Singleton, E. G. Strangas, and S. Aviyente, "Discovering the hidden health

states in bearing vibration signals for fault prognosis," in *Industrial Electronics Society, IECON 2014-40th Annual Conference of the IEEE*. IEEE, 2014, pp. 3438–3444.

[36] G. Li, S. J. Qin, Y. Ji, and D. Zhou, "Reconstruction based fault prognosis for continuous processes," *Control Engineering Practice*, vol. 18, no. 10, pp. 1211–1219, 2010.

[37] W. G. Zanardelli, E. G. Strangas, H. K. Khalil, and J. M. Miller, "Wavelet-based methods for the prognosis of mechanical and electrical failures in electric motors," *Mechanical Systems and Signal Processing*, vol. 19, no. 2, pp. 411–426, 2005.

[38] A. HABBI and M. ZELMAT, "Design of a fuzzy model-based controller for a drum boiler-turbine system," in *Applied Computational Intelligence*. World Scientific, 2004, pp. 659–664.

[39] A. Kusiak, H. Zheng, and Z. Song, "On-line monitoring of power curves," *Renewable Energy*, vol. 34, no. 6, pp. 1487–1493, 2009.

[40] M. Schlechtingen, I. F. Santos, and S. Achiche, "Using data-mining approaches for wind turbine power curve monitoring: a comparative study," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 3, pp. 671–679, 2013.

[41] F. Lobato, C. Sales, I. Araujo, V. Tadaiesky, L. Dias, L. Ramos, and A. Santana, "Multi-objective genetic algorithm for missing data imputation," *Pattern Recognition Letters*, vol. 68, pp. 126–131, 2015.

[42] I. B. Aydilek and A. Arslan, "A hybrid method for imputation of missing values using optimized fuzzy c-means with support vector regression and a genetic algorithm," *Information Sciences*, vol. 233, pp. 25–35, 2013.

[43] M. G. Rahman and M. Z. Islam, "Missing value imputation using decision trees and decision forests by splitting and merging records: Two novel techniques," *Knowledge-Based Systems*, vol. 53, pp. 51–65, 2013.

[44] P. K. Sharpe and R. Solly, "Dealing with missing values in neural network-based diagnostic systems," *Neural Computing & Applications*, vol. 3, no. 2, pp. 73–77, 1995.

[45] X. Wang, A. Li, Z. Jiang, and H. Feng, "Missing value estimation for dna microarray gene expression data by support vector regression imputation and orthogonal coding scheme," *BMC bioinformatics*, vol. 7, no. 1, p. 32, 2006.

[46] T. Samad and S. A. Harp, "Self–organization with partial data," *Network: Computation in Neural Systems*, vol. 3, no. 2, pp. 205–212, 1992.

[47] H. Kim, G. H. Golub, and H. Park, "Missing value estimation for dna microarray gene expression data: local least squares imputation," *Bioinformatics*, vol. 21, no. 2, pp. 187–198, 2005.

[48] M. Schlechtingen and I. F. Santos, "Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection," *Mechanical systems and signal processing*, vol. 25, no. 5, pp. 1849–1875, 2011.

[49] X. Zhu, S. Zhang, Z. Jin, Z. Zhang, and Z. Xu, "Missing value estimation for mixed-attribute data sets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 1, pp. 110–121, 2011.

[50] A. Farhangfar, L. Kurgan, and J. Dy, "Impact of imputation of missing values on classification error for discrete data," *Pattern Recognition*, vol. 41, no. 12, pp. 3692–3705, 2008.

[51] A. Farhangfar, L. A. Kurgan, and W. Pedrycz, "A novel framework for imputation of missing values in databases," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 37, no. 5, pp. 692–709, 2007.

[52] K. Swingler, *Applying neural networks: a practical guide*. Morgan Kaufmann, 1996.

[53] P. Caselitz and J. Giebhardt, "Advanced maintenance and repair for offshore wind farms using fault prediction techniques," in *Proceedings of the World Wind Energy Conference*, 2002.

[54] A. Frank, "Asuncion," *UCI machine learning repository*, 2010.

[55] J. W. Graham, "Missing data analysis: Making it work in the real world," *Annual review of psychology*, vol. 60, pp. 549–576, 2009.

[56] Z. Meng and Z. Shi, "Extended rough set-based attribute reduction in inconsistent incomplete decision systems," *Information Sciences*, vol. 204, pp. 44–69, 2012.

[57] S. Zhang, "Shell-neighbor method and its application in missing data imputation," *Applied Intelligence*, vol. 35, no. 1, pp. 123–133, 2011.

[58] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing and Applications*, vol. 19, no. 2, pp. 263–282, 2010.

[59] Y. Ding and A. Ross, "A comparison of imputation methods for handling missing scores in biometric fusion," *Pattern Recognition*, vol. 45, no. 3, pp. 919–933, 2012.

[60] X. Yang, B. Huang, Y. Zhao, Y. Lu, W. Xiong, and H. Gao, "Generalized expectation–maximization approach to lpv process identification with randomly missing output data," *Chemometrics and Intelligent Laboratory Systems*, vol. 148, pp. 1–8, 2015.

[61] K. O. Akande, T. O. Owolabi, and S. O. Olatunji, "Investigating the effect of correlation-based feature selection on the performance of support vector machines in reservoir characterization," *Journal of Natural Gas Science and Engineering*, vol. 22, pp. 515–522, 2015.

[62] H. Takagi and I. Hayashi, "Nn-driven fuzzy reasoning," *International Journal of Approximate Reasoning*, vol. 5, no. 3, pp. 191–212, 1991.

[63] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," in *Proceedings of the sixth international symposium on Multiple-valued logic*. IEEE Computer Society Press, 1976, pp. 196–202.

[64] M. Sugeno, *Industrial applications of fuzzy control*. Elsevier Science Inc., 1985.

[65] K. Salahshoor, M. Kordestani, and M. S. Khoshro, "Fault detection and diagnosis of an industrial steam turbine using fusion of svm (support vector machine) and anfis (adaptive neuro-fuzzy inference system) classifiers," *Energy*, vol. 35, no. 12, pp. 5472–5482, 2010.

[66] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE transactions on systems, man, and cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

CHAPTER 4

Power Production Prediction of Wind Turbine Using Fusion of MLP and ANFIS Networks

4.1 Introduction

Large scale wind facilities are approaching the output rating of conventional power plants, enlightening the importance of precise system condition monitoring in managing optimal turbines operation. Active control has an immediate impact on the cost of wind energy. Moreover, high performance and reliable controllers are essential to enhance the competitiveness of wind technology. Accurate modeling of wind turbines as targeted by the current research can enhance wind energy production capabilities and reliability, and give rise to a general reduction in financial risk of wind farm investment.

There are three general methods for any system modeling. Physics-based models, data-driven approaches and hybrid methods, which are the combination of the first two, can be employed [1, 2]. The method selection is based on the systems characteristics, application, modeling purpose, and information availability.

Various methodologies can be considered for condition monitoring and performance optimization of wind turbines. These include methods of determining wind turbine condition through addressing structural and aerodynamic concerns pertaining to wind turbines. Two major categories of such studies include airfoil shape optimization and entire blade performance optimization [3, 4, 5]. These types of studies are conducted using numerical analysis employing concepts like Computational Fluid Dynamics (CFD) or Finite Element Analysis (FEA) [6, 7, 8, 9] and experimental approaches [10, 11, 12]. Each of these methods encompasses certain limitations. Numerical analysis outcome mainly suffers from required simplifications and necessary validation. Similarly, laboratory experimental models are more expensive and the entailed challenges normally put their industrial application at risk. Moreover, they do not address current issues associated with wind farms such as underperformance and unexpected failures. These issues constrain the applicability and practicality of physics-based models in analyzing wind turbines where complex, non-linear relationship between subsystems causes numerous difficulties to effectively model the system. Consequently, physics-based models might not return similar results to actual commercialized machines.

Alternatively, data-driven approaches can be applied that analyze machine historical performance data. These methods are generally less costly than physics-based methods, and since wind farms have access to Supervisory Control And Data Acquisition (SCADA) data, they are suitable choices for performance optimization of wind turbines.

For the systems that expert knowledge is available but still consist of uncertainties and unknown influential factors, hybrid methods can be employed. In such cases like wind turbines, when comprehensive mathematical modeling based on the physical knowledge is practically impossible and also, a large amount of data is available, hybrid approaches are the best choice. This method, which benefits from the advantages of both types is applied in this paper.

A significant number of techniques function based on the historical performance of a system. These techniques include the algorithms that do not require expert knowledge of the system such as Neural Networks (NNs) [2, 13, 14, 15], Adaptive Neuro-Fuzzy Inference System (ANFIS) [16, 17], Support Vector Machines (SVM) [18, 19], regression methods [20, 21, 22] and Probability Density Functions (PDF) [23, 24, 25, 26, 27]. The algorithms that demand expert knowledge are normally more challenging to employ mostly due to difficulties associated with translating the expert knowledge into mathematical expressions. Such methods include Fuzzy Logic (FL) [28, 29], Bayesian analysis [30, 31], Markov model [32, 33] and signal processing methods [34, 35, 36].

To initiate training of the networks for any aforementioned algorithm, the input parameters must be determined. In this research, a combination of practical knowledge related to wind energy and farms and a recursive Principal Component Analysis (PCA) is utilized. PCA is a useful statistical technique practiced in various research areas to find a latent pattern in high dimensional data. It helps express and highlight similarities and differences in the dataset and can be utilized for condition monitoring, pattern recognition, anomaly detection, etc. for various industrial processes. In such applications, PCA, formulated as a multivariate statistical process control task, extracts a few independent components from highly correlated process data and use them to monitor the operation of the process more efficiently [37, 38, 39]. In addition, unlike many studies conducted on SCADA data where missing values were removed [15, 40, 41] as part of the data pre-processing, they are substituted by application of the decision tree concept in this paper.

The work presented in this paper introduces a new data fusion methodology to predict future wind turbine generated power. For this purpose, a classification fusion scheme is proposed to merge two different networks of MLP and ANFIS. Data fusion approach utilizes available informative sources to enhance decision-making processes. The synergistic use of overlapping and complementary data sources enable a rich database which is not otherwise available through individual sources. Therefore, the fused monitoring tool can result in a more precise and reliable outcome by applying the complementarities among different networks.

The principal novelty of this paper includes combining an efficient imputation method with a dynamic PCA approach to extract suitable input parameters; which enables the establishment of a comprehensive prediction tool. Furthermore, a new dynamic fused structure is established to benefit from two diverse classifiers to improve the precision of the prediction.

The paper is organized as follows. A brief Mechanical System Description is presented in Section 4.2. Data pre-processing is explained in Section 4.3. The applied imputation algorithm is thoroughly explained in this section. Section 4.4 introduces the fusion classification scheme consisting of OWA operator, MLP, and ANFIS networks. Off-line and on-line PCAs are mathematically expressed in Section 4.5. Simulation and test results

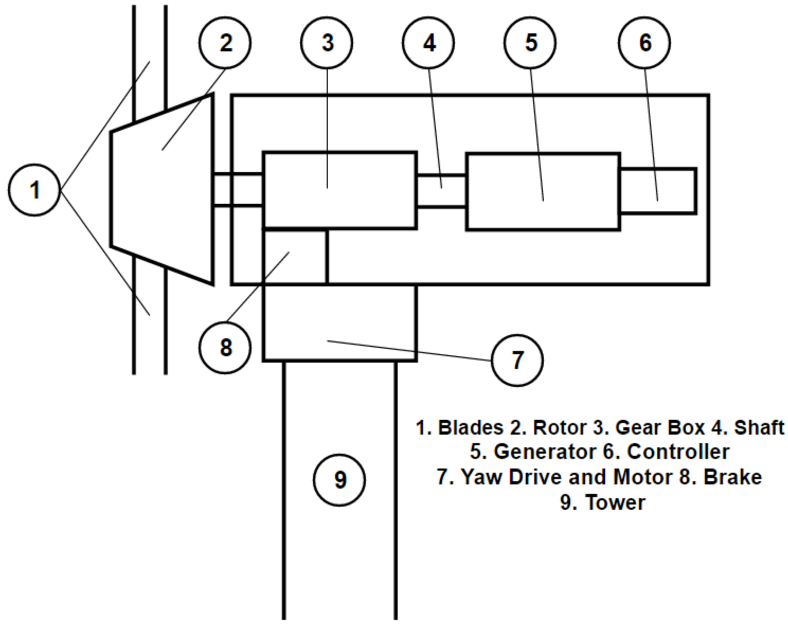

FIGURE 4.2.1: Main Components of a Horizontal Axis Wind Turbine

are presented in Section 4.6. Finally, Section 4.7 concludes with highlights of the major findings.

4.2 Mechanical System Description

Wind turbines are mechanical devices that are used to generate electricity from the kinetic power of the wind. They are complex, nonlinear, dynamic systems forced by gravity, stochastic wind disturbances, and gravitational and gyroscopic loads. The aerodynamic behavior of wind turbines is also nonlinear, unsteady, and complex. Turbine rotors are subjected to a complicated three-dimensional turbulent wind inflow field that drives fatigue loading. Consequently, wind turbine modeling is also complex and challenging. Accurate models must contain many degrees of freedom (DOF) to capture the most important dynamic effects. The rotation of the rotor adds complexity to the dynamics modeling. Designs of control algorithms for wind turbines must account for these complexities. Algorithms must capture the most important turbine dynamics without being too complex or impractical [42]. A schematic of the location of major parts of a horizontal axis, pitch regulated wind turbine, is illustrated in Figure 4.2.1.

The wind turbine control system consists of sensors, actuators and a system that ties these elements together. A hardware or software system processes input signals from the sensors and generates output signals for actuators. The main goal of the controller is to modify the operating states of the turbine to maintain safe turbine operation, maximize power, decrease damaging fatigue loads, and detect possible fault conditions. A supervisory control system starts and stops the machine, yaws the turbine when there is a significant yaw misalignment, detects fault conditions, and performs emergency shut-downs. Other parts of the controller are intended to maximize power and reduce loads during normal turbine operation.

Theoretically, the power generated by a wind turbine is calculated using the following equation

$$P = \frac{1}{2}\rho A C_p V^3 \tag{4.2.1}$$

where ρ is the air density, A is the rotor sweep area, V is the wind speed, and C_p is the power coefficient. However, a similar trend is not observed in the power curve of a commercialized wind turbine. A typical power curve of a wind turbine consisting of three regions is shown in Figure 4.2.2. Wind turbines start operating when the wind speed reaches a minimum value called cut-in speed. In the second region, as the wind speed increases, the output power also grows rapidly until it reaches the rated power. Finally, in the third region, the power output remains constant. This region ends when the wind speed exceeds the maximum cut-out speed. If a turbine is in generating mode when the wind speed exceeds the cut-out speed, the blades, generator, and even the drive between them can be damaged. To avoid that, at higher speeds, the pitch is set to the feathered position so that rotation speed decreases and comes to a quick full stop.

4.3 Data Pre-processing

In this research, twenty-one, 2.3 MW, pitch-regulated wind turbines have been investigated. These turbines are located in a wind farm in Ontario, Canada, and the dataset obtained from SCADA system covers a range of 20 months. The turbine layout can be observed in

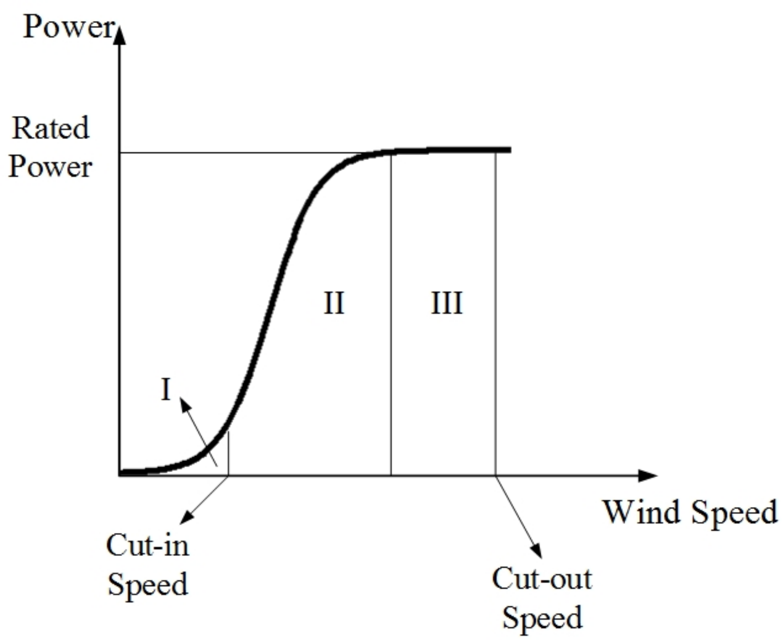

FIGURE 4.2.2: A typical power curve

Figure 4.3.1. Due to a customary deficiency in data collecting systems such as SCADA, it is inevitable to pre-process the data for any type of decision-making purposes [43, 44, 45, 46]. Collected data often include missing, incorrect, and noisy values. This type of inaccuracy and low quality in data cause a significant reduction in the excellence of any network training since these types of networks are highly sensitive to the data quality [47].

To carry out the pre-processing, the following steps are taken in the presented order: Accuracy check, treatment of missing values, and data range scaling. The sequential execution of these steps are explained in the following sections.

4.3.1 Accuracy Check

To control and investigate sensors performance, data range check is carried out. In this stage, outliers are removed to generate a smoother data set. It is extremely important to carefully choose the range to avoid classification of real measured values as outliers. This might prevent detection of abnormal behavior of turbines. This stage of pre-processing ensures that the data range is as expected, mutual consistency of components is met and finally, input and output vectors are consistent. This is especially critical for supervised

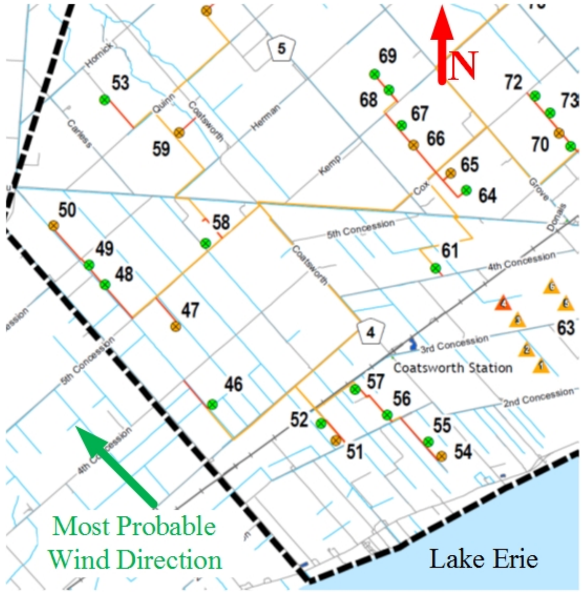

FIGURE 4.3.1: The Turbine Layout

learning applications [48, 49, 50].

Unlike similar studies conducted on the SCADA data of wind turbines in which outliers of the entire dataset are removed, in this paper, they are treated as missing values. The imputation of these values is explained in the following section.

4.3.2 Treatment of Missing Values

Missing values bring harmful consequences to data analysis. Data mining algorithms and techniques are not designed to directly handle these values and they are particularly sensitive to them during the training phase. For instance, if a supervised learning tool tries to fit a curve to these values, the generalization error will drastically increase and suitable training will not be achieved [51]. Subsequently, it is imperative to identify them and clarify the treatment method [52].

Since there is a large amount of information available in SCADA data, it was claimed that there was no need for estimation of missing values in similary conducted studies. It was very common to neglect the entire record of data when one value was missing in such studies. This is not the best approach since useful information can be lost [51, 53]. There are several signals in a dataset collected by SCADA system where each of them is designated as a column. So, even if only one signal is missing, the others would also be deleted by this approach. In addition, obtaining consecutive historical data of system performance is critical in data-driven analyses. Thus, ignoring the entire row of data just because one sensor is not functioning properly, damages the reliability of the network outcomes. This is a matter of even higher importance in wind turbines analysis.

Wind pattern that is the most dominant factor on turbine power production is subject to rapid variations. This strengthens the importance of having access to uninterrupted historical data in order to provide a prediction of future machine performance. Thus, substitution of missing values (in both situations of actual missing values and outlier removal) with reasonable data is necessary. This process is called data imputation [54]. This process can be carried out by applying both statistical learning theory and machine learning tools [55]. Selection of the proper method highly depends on the application, type and size of dataset, missing values amount and pattern, maximum acceptable imputation error and computation capacity.

The simplest method of imputation is to replace the missing values with the mean or mode of each attribute. This works best with small datasets when data range is not very wide. For SCADA data, however, it is reasonable to employ more advanced techniques, since it includes a significant number of instances.

A substantial number of algorithms have been used in studies conducted for the purpose of imputation. K-Nearest Neighbor (K-NN) technique [56] and Expectation Maximization (EM) algorithm [43, 57] are among the most popular. They have been proven to result in a low imputation error in many cases, although their complexity and iterative manner make it computationally inefficient to utilize them for large datasets. There must always be a reasonable balance between the complexity of the applied algorithm, which determines the outcome error and its required computations. Here, a combination of decision trees and mean value has been applied to efficiently create this balance.

Decision trees split data sets into smaller parts and place them in leaf nodes. It is performed in a top-down manner starting from the root node. In the beginning, all the training instances are at the root. They are then partitioned into subsets according to the selected attributes. This selection can be based on a variety of statistical measures. One

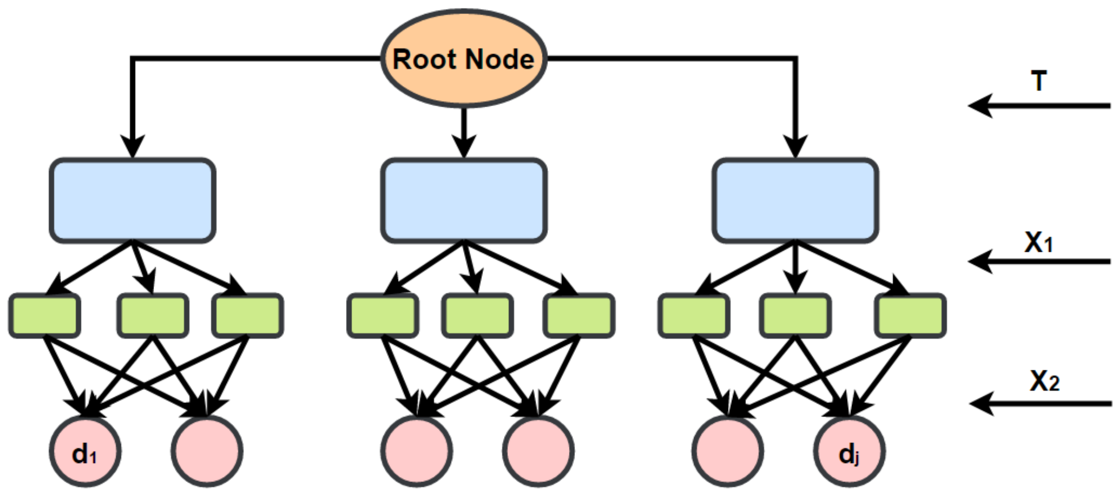

FIGURE 4.3.2: Schematic of a Decision Tree

of the most popular measures is information gain, which is based on the concept of Gibbs entropy as expressed in Equations 4.3.1 and 4.3.2.

$$E(T) = \sum_{i=1}^{c} -p_i log_2 p_i \qquad (4.3.1)$$

$$E(T, X) = \sum_{c \in X} p(c) E(c) \qquad (4.3.2)$$

where E is the entropy, c is the number of possible outcomes, and p_i is the possibility of each outcome. T and X represent the features in the dataset, where T is the target feature. Figure 4.3.2 illustrates a schematic of a decision tree when the data is divided into different groups by the target feature of T, and then, divided further into smaller sets, applying different ranges in two features of X_1 and X_2. The last feature applied to split the dataset, in this figure, X_2, is called the class attribute. Sub datasets of d_1, d_2,...,d_j include the data points placed in each leaf.

The procedure to impute missing values are summarized in the following steps [43] and illustrated in Figure 4.3.3.

Step 1: In this step, after identification of missing values in the dataset (Df), the data is separated into two datasets of complete (D_c) and incomplete (D_i) data sets in a way that all instances with missing values are in D_i.

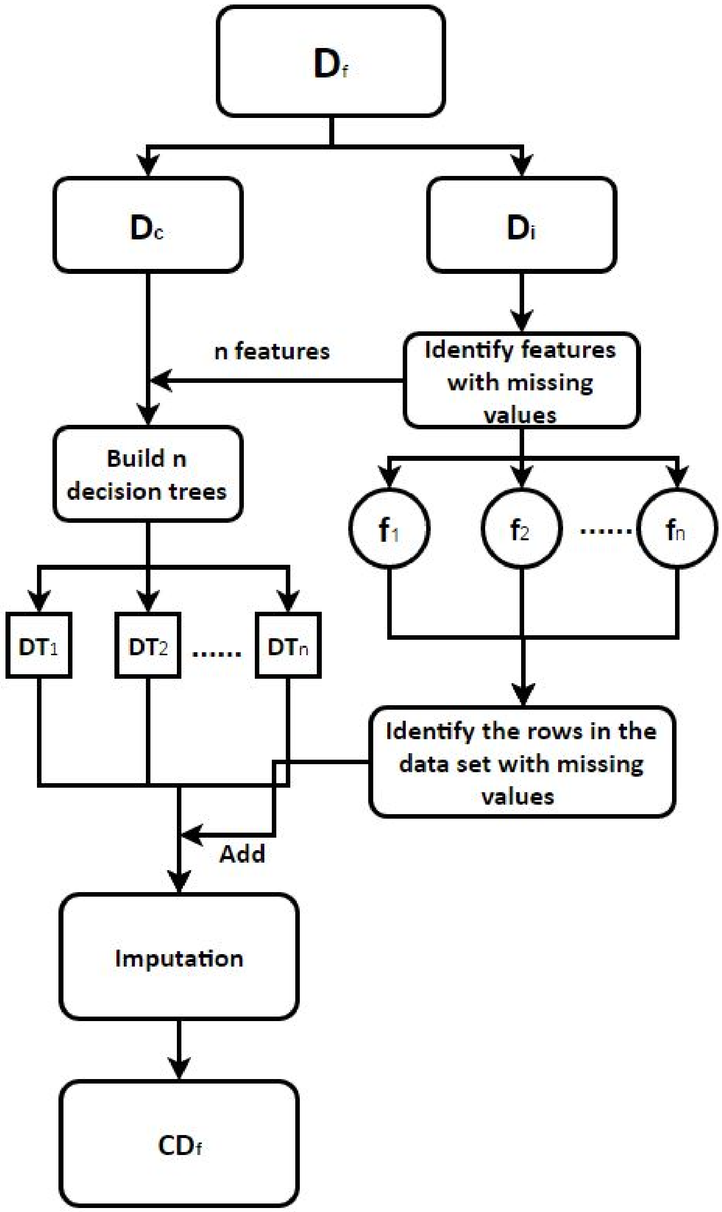

FIGURE 4.3.3: Imputation Procedure

Step 2: In this step, the target is to create sub datasets of d_j consisting of the data in each leaf of each decision tree. At first, in order to create the sub datasets, the numerical values are generalized for building the trees. The number of groups to divide the numerical values is the root square of the domain size of that feature. Next, the features in which they include missing values are identified (f_1, f_2, ..., f_n) and then a set of decision trees (based on D_c) is made considering each of those features with missing values as the class attributes (DT_1, DT_2, ..., DT_n).

Step 3: In this stage, we aim to add the instances with missing values to each leaf they belong to. For this purpose, after finding each missing value, its respective feature is identified. Then, the decision tree which has that particular feature as the class attribute is selected and the target leaf for that record is predicted. That record is then added to that leaf sub dataset (d_j). If one record has more than one value missing, that record will be added to more than one leaf, since different trees have been used to predict the target leaf.

Step 4: In this step, the imputation is performed. First, each feature with a missing value in each instance of D_i is identified and then the sub dataset, to which that record was added, is recognized. The imputation is done using the mean value of each feature in the leaf. Then, the missing value is replaced by the imputed one in D_i. Finally, D_c and D_i are combined together to form the completed dataset (CD_f).

4.3.3 Data Range Scaling

When applying multiple inputs to train a model, it is essential to have all variables in the same range; otherwise, the one with the wider range will dominate during training, possibly overshadowing more influential parameters.

4.4 Wind turbine performance prediction using a fusion classification scheme

This section presents the design procedure of implemented fusion classification technique. The major purpose is to benefit from complementary inference due to each individual MLP and ANFIS classifiers. In this method, each algorithm is trained over the entire training dataset and a final decision is then made based upon an aggregation space [58]. To develop this technique for the purpose of accurately predicting the machine performance, Ordered Weighted Averaging (OWA) is introduced as the fusion method to integrate inferences of MLP and ANFIS techniques. Then, the structures of MLP and ANFIS networks are demonstrated.

4.4.1 OWA Operator

The OWA operator, in general, creates a class of parameterized aggregations operators including minimum, maximum, and the mean. The basis of this operator has been employed in a variety of applications including data mining, decision making, approximate reasoning, expert systems, fuzzy systems and control [59, 60, 61, 62]. The main advantage of this operator is its capability to encompass various operators bounded between a minimum and a maximum value.

An OWA operator creates a mapping of $F : R^n \rightarrow R$ where n is the space dimension and the associated weight vector $w = (w_1, w_2, ..., w_n)^T$ is in such a way that satisfies the conditions expressed in Equations 4.4.1 and 4.4.2.

$$w_1 + w_2 + ... + w_n = 1, 0 \leq w_i \leq 1, i = 1, 2, ..., n \tag{4.4.1}$$

$$F(a_1, a_2, ..., a_n) = \sum_{i=1}^{n} b_i w_i \tag{4.4.2}$$

where b_i represents the ith largest of $a_1, a_2, ..., a_n$.

The most important item in the application of an OWA operator is the employed method to accurately determine the weights. We consider that there are M number of various data, each of which includes values that are provided by N information sources and also the ideal output which is intended to be estimated. Therefore, each example consists of $N+1$ values and for the ith example, it can be expressed as $a_1^i a_2^i ... a_N^i | b^i$. where a_i denotes the value provided by the jth information source and b_i is the ideal output. For aggregation purposes, the goal is to find the weighting vector w in such a way that the following condition is met

$$minimize \sum_{j=1}^{M}(w(a_1^j, ...a_n^j) - b^j)^2 \tag{4.4.3}$$

Several methods can be employed to address Equation 4.4.3. In this paper, to implement the OWA operators, gradient descent approach is applied [63, 64].

4.4.2 Multi-Layer Perceptron (MLP) Neural Network

MLP networks are widely applied for a variety of purposes in data-driven modeling. They are proven to create reliable networks with fast computational capabilities. They are feed forward neural networks that create a mapping between input and output spaces. They have been suitably utilized for pattern recognition, condition monitoring, fault diagnosis, function approximation and many other purposes [65, 66, 67]. The MLP network consists of multiple layers of nodes in the forward direction and all nodes in each layer are fully connected to the nodes in the next layer. There exists three layers of input, output, and hidden layers in its structure. The nodes in the hidden layer are neurons with a non-linear activation function that takes various forms such as hyperbolic tangent, sigmoid, and so on. The output layers, however, mostly use nodes with linear functions while input layer acts as a buffer.

During the training of the network, weights of each input $(w_1, w_2, ..., w_n)$ and also the bias term (b) are determined and then added together to be applied by the activation function. In addition, MLP network employs a supervised learning tool called the error back propagation algorithm for training. A typical structure of an MLP with n number of inputs, $x_1, x_2, ..., x_n$, is illustrated in Figure 4.4.1.

4.4.3 Adaptive Neuro-Fuzzy Inference System (ANFIS)

ANFIS models were first introduced by Takagi and Hayeshi [68] and also further developed by Jang [69]. They have been extensively operated in many research works and were first employed by Schlechtingen et al. [41] for wind turbine power monitoring purposes. As its

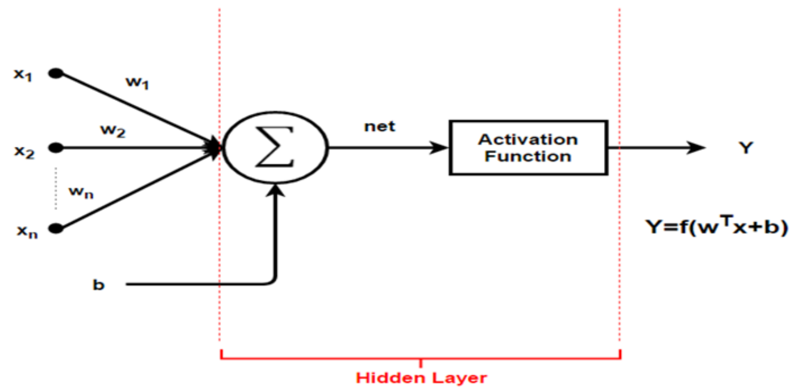

FIGURE 4.4.1: A typical MLP network

name implies, ANFIS networks are a combination of the quality based fuzzy approach and neural networks adaptive characteristics to achieve enhanced network performance.

Fuzzy logic provides an inference system under cognitive uncertainty while neural networks benefit from computational advantages like adaption, fault tolerance, and generalization. Therefore, ANFIS method is established as NNs and FL are engaged for dealing with cognitive uncertainty in a more qualitative manner. In practice, ANFIS is a fuzzy model that facilitates adapting and learning in a flexible system framework. As a result, ANFIS does not require expert knowledge to establish the modeling system.

The mathematical transformation of expert knowledge in fuzzy logic modeling is a time consuming and challenging task to perform. On the other hand, NNs can confidently handle non-linear prediction and estimation with high computational capabilities. Therefore, this integration of methods can create a computationally efficient environment with an inference mechanism.

The inference mechanism in fuzzy logic is created by utilization of membership functions. Mamdani [70] and Sugeno [71] have proposed two common types of inference systems. In the Sugeno type, the consequent part is a non-fuzzy equation, whereas a fuzzy linguistic value is proposed in the Mamdani type. The Sugeno type is a better choice for wind turbine analysis mainly due to its computational efficiency, continuity of the created output surface and also the fact that a non-fuzzy equation represents the consequent part.

To model fuzzy rules into desired outputs, the Sugeno fuzzy model is implemented as follows:

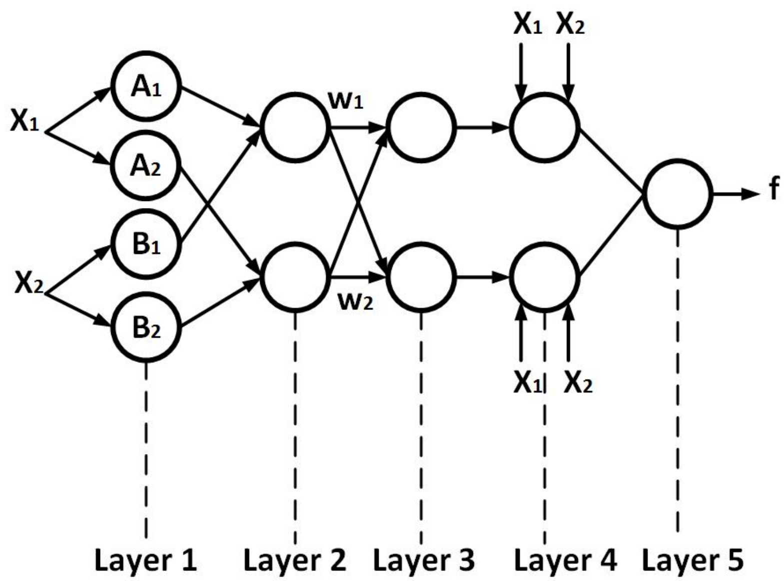

FIGURE 4.4.2: A five-layer ANFIS structure

If $x_1 = A_i$ and $x_n = B_j$ then $f_i = p_i x_1 + q_i x_n + r_i$

where p_i, q_i and r_i are adaptive design parameters that are determined in the training phase. A typical five-layer ANFIS structure is shown in Figure 4.4.2. In this feed-forward structure, parameters in adaptive nodes are constantly adjusted during the training. Mathematical summarization of each layer is presented herein [72, 73].

Layer 1. The first layer consists of adaptive nodes. All the neurons in this layer correspond to a linguistic label and the resulting output is equal to the membership function of this label.

$$OL1_i = \mu_{A_i}(x_1) \tag{4.4.4}$$

Layer 2. Unlike the previous layer, nodes in the Layer 2 are not adaptive (fixed). They estimate the firing strength (w_i) of a rule calculated by multiplication of incoming signals.

$$OL2_i = w_i = \mu_{A_i}(x_1)\mu_{B_i}(x_n) \tag{4.4.5}$$

Layer 3. The nodes in this layer are also fixed. The output of each layer is the ratio of the ith rule firing strength over the summation of firing strengths of all rules, as formulated in Equation 4.4.6. These nodes basically normalize the firing strength of the previous layer.

$$OL3_i = \bar{w}_i = \frac{w_i}{\sum_{j=1}^{i} w_i} \tag{4.4.6}$$

Layer 4. All the nodes in this layer are adaptive. Each node outcome is equal to multiplication of relative firing strength (found in the previous layer) and the consequent parameter as

$$OL4_i = \bar{w}_i f_i = \bar{w}_i (p_i x_1 + q_i x_n + r_i) \tag{4.4.7}$$

Layer 5. The number of nodes is equal to the number of output parameters, which in this case is one. This node operates as a summer and computes the summation of all signals from Layer 4.

$$OL5_i = \sum_{i=1}^{j} \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{4.4.8}$$

There is no unique structure for ANFIS models since a combination of some layers may still produce similar results. According to the structure shown in Figure 4.4.2, there are two adaptive layers. Parameters in the Layer 1 called premise parameters are modified based on the input membership functions. Additionally, in Layer 4, there are three modifiable parameters of p_i, q_i and r_i which are first order polynomials. These are the ones that are considered consequent. During the training phase, all these modifiable parameters are tuned in such a way that the network output matches the training data.

4.5 Feature Extraction Applying Principal Component Analysis (PCA)

A primary but still important step in any data mining approach is a suitable selection of inputs based on the desired output. In this paper, PCA partially satisfies this goal. This section explains the mathematical basis for off-line and recurrent PCA.

The general procedure for the PCA mathematical calculations in an off-line manner is as follows:

Step 1: Subtract the mean from the data point values.

Step 2: Covariance matrix is then calculated.

Step 3: At this stage, the eigenvalues and eigenvectors of the covariance matrix are determined.

Step 4: Components are then selected and features are formed. To achieve this result, eigenvectors are sorted based on the eigenvalues from the highest to the lowest. That basically put the components into an order of significance. At this step, the items of lesser significance can be neglected.

Step 5: At this step, a new dataset is formed that finalizes the PCA transformation. Once the important components are kept and the feature vector is created, the transpose of that vector is considered and multiplied by the original dataset on the left of the transposed.

This method is a simple and yet popular multivariate statistical technique, which suffers from a certain disadvantage. This is an off-line method and is only useful for stationary models. To address this issue, the recursive PCA as explained in the following section is employed to track the dynamic behavior of the system.

4.5.1 Recursive PCA

When PCA models are built from the data according to the aforementioned procedure, they are time-invariant, unlike most actual industrial processes that tend to change over time. The main characteristics of the industrial processes that are time-varying include the changes in the mean, variance, and the correlation structure between variables that can affect the number of significant principal components (PCs). In the case of applying a time-invariant PCA model for the condition monitoring tasks in processes that might characteristically change over time, false alarms will occur that obviously question the reliability of the model. Slow time-varying behaviors are normally observed in industrial processes as a result of equipment ageing, sensors or process drifting, and preventive maintenance and cleaning. To address this issue, adaptive or recursive PCA models have been developed [74, 75].

The intent here is to develop a method to update the PCA whenever a new block of data becomes accessible. Considering X_1^0 as the initial block of data, the mean of each column is given in the vector

$$b_1 = \frac{1}{n_1}(X_0^1)^T 1_{n_1} \tag{4.5.1}$$

where $1_{n_1} = [1, 1, ..., 1]^T \in R^{n_1}$

The data is scaled to zero mean and unit variance is given by

$$\Sigma_1 = diag(\sigma_{1.1}, ..., \sigma_{1.m}) \tag{4.5.2}$$

$$X_1 = (X_1^0 - 1_{n_1} b_1^T)\Sigma_1^{-1} \tag{4.5.3}$$

Whose ith element is the standard deviation of the ith sensor; $i = 1, ...m$. The correlation matrix is

$$R_1 = \frac{1}{n_1 - 1} X_1^T X_1 \tag{4.5.4}$$

The new block of data is expected to augment the data matrix and calculate the correlation matrix recursively. Assume that b_k, X_k and R_k have been calculated when the kth block of data is collected. The task for recursive calculation is to calculate b_{k+1}, X_{k+1} and R_{k+1} when the next block of data $X_{n_{k+1}}^n \in R^{n_{k+1} \times m}$ is available. Denoting

$$X_{k+1}^0 = \begin{bmatrix} X_k^0 \\ X_{n_{k+1}}^0 \end{bmatrix} \tag{4.5.5}$$

for all the $k+1$ block of data, the mean vector b_{k+1} is related to b_k by the following relation:

$$(\sum_{i=1}^{k+1} n_i) b_{k+1} = (\sum_{i=1}^{k} n_i) b_k + (X_{n_{k+1}}^0)^T 1_{n_{k+1}} \tag{4.5.6}$$

Denoting $N_k = \sum_{i=1}^{k} n_i$, above equation yields to the following recursive calculation

$$b_{k+1} = \frac{N_k}{N_{k+1}} b_k + \frac{1}{N_{k+1}} (X_{n_{k+1}}^0)^T 1_{n_{k+1}} \tag{4.5.7}$$

The recursive calculation of X_{k+1} is given by

$$
\begin{aligned}
X_{k+1} = [X^0_{k+1} - 1_{k+1}b^T_{k+1}]\Sigma^{-1}_{k+1} = [\begin{bmatrix} X^0_k \\ X^0_{n_{k+1}} \end{bmatrix} - 1_{k+1}b^T_{k+1}]\Sigma^{-1}_{k+1} \\
= \begin{bmatrix} X^0_k - 1_k\Delta b^T_{k+1} - 1_k b^T_k \\ X^0_{n_{k+1}} - 1_{k+1}b^T_{k+1} \end{bmatrix}\Sigma^{-1}_{k+1} = \begin{bmatrix} X_k\Sigma_k\Sigma^{-1}_{k+1} - 1_k\Delta b^T_{k+1}\Sigma^{-1}_{k+1} \\ X_{n_{k+1}} \end{bmatrix}
\end{aligned}
\tag{4.5.8}
$$

where

$$X_k = (X^0_k - 1_k b^T_k)\Sigma^{-1}_k \tag{4.5.9}$$

$$X_{n_{k+1}} = (X_{n_{k+1}} 1_{n_{k+1}} b^T_{k+1})\Sigma^{-1}_{k+1} \tag{4.5.10}$$

$$\Sigma_j = diag(\sigma_{j.1}, ..., \sigma_{j.m}), j = k, k+1 \tag{4.5.11}$$

$$\Delta b_{k+1} = b_{k+1} - b_k \tag{4.5.12}$$

The recursive computation of the standard deviation, has the following relationship [75]

$$(N_{k+1} - 1)\sigma^2_{k+1.i} = (N_k - 1)\sigma^2_{k.i} + N_k\Delta b^2_{k+1}(i)) + \left\| X^0_{n_{k+1}}(:, i) - 1_{n_{k+1}} b_{k+1}(i) \right\|^2 \tag{4.5.13}$$

where $X^0_{n_{k+1}}(:, i)$ is the ith column of the associated matrix. $b_{k+1}(i)$ and $\Delta b_{k+1}(i)$ are the ith elements of the associated vectors. Similarly, the recursive calculation of the correlation matrix has the following form

$$
\begin{aligned}
R_{k+1} = \frac{1}{N_{k+1} - 1}X^T_{k+1}X_{k+1} - \frac{N_k - 1}{N_{k+1} - 1}\Sigma^{-1}_{k+1}\Sigma_k R_k \Sigma_k \Sigma^{-1}_{k+1} \\
- \frac{N_k}{N_{k+1} - 1}\Sigma^{-1}_{k+1}\Delta b_{k+1}\Delta b^T_{k+1}\Sigma^{-1}_{k+1} + \frac{1}{N_{k+1} - 1}X^T_{n_{k+1}}X_{n_{k+1}}
\end{aligned}
\tag{4.5.14}
$$

Therefore, Equations 4.5.7, 4.5.8, and 4.5.14 create the recursive manner for the PCA model.

4.6 Fusion Technique Simulation Modeling

In this section, the result of the applied imputation technique, feature extraction analysis, and performance of designed data fusion network will be presented.

4.6.1 Imputation Results

To ensure suitability of the proposed imputation technique, the validation of this algorithm is presented in this section to be followed by finalization of data pre-processing and network training. For this purpose, a sub dataset with no missing values from SCADA data is extracted and then 5 percent of the values from various features and instances are randomly deleted. This amount is chosen based on the normal amount of observed missing, outlier, and out-of-range values in SCADA data. The imputation is then performed, and imputed values are compared with the true values originally removed from the sub dataset. To calculate the imputation error, Normalized Root Mean Square (NRMS) error is applied according to Equation 4.6.1 and the norm of a matrix is also determined according to Equation 4.6.2.

$$NRMS = \frac{\left\| X^{estimate} - X^{original} \right\|}{\left\| X^{original} \right\|} \tag{4.6.1}$$

$$\|A\| = \sqrt{\sum_{i=1}^{m} \sum_{j=1}^{n} |a_{ij}|^2} \tag{4.6.2}$$

The calculated NRMS after performing imputation was 0.0306, and this low error clearly validates the proposed imputation procedure and justifies its application when actual values are unknown.

4.6.2 Feature Extraction

In this section, the features to be considered as the inputs and the applied method to define these parameters for the training of the networks are explained.

Input parameters

Based on Equation 4.2.1, wind speed is the most influential factor in wind turbines power curve analysis, and there cannot be any reasonable modeling without considering the wind speed. However, it is also clear that the number of inputs must go beyond the wind speed alone. The other inputs are partially chosen based on the results of PCA analysis. All the features in the dataset other than wind speed and the output parameter (active power) are analyzed to extract two principal components with a higher level of significance. These PCs are the second and third input parameters. Finally, in order to create recurrent networks, latest value of the active power that is available in practice at any data point should also be considered as the fourth and last input. The following section explains how these inputs are applied to train the networks.

Creating the training dataset

As explained before, to create recurrent networks previous values of output parameter (active power) should be defined as an input parameter. It should also be noted that the outcome of the models is to make predictions of the future performance of the machines. We consider M as the duration of time into the future for which an estimation of power production is intended to be calculated. So, if t is the present time, the prediction will be for the power production at the time $t + M$. The latest known input parameters are normally at the time t and consequently, the recurrent term is available at $t - 1$.

For static terms, which are the two PCs, training dataset starts from the second data point of the original dataset and ends with the point that is M distance ahead of the previous point. Also, in practice, it is very normal to have access to the weather forecast for the meteorological inputs which in this situation is wind speed. It is important to note that since the value of M is in the scale of hours, the weather forecast would be precise enough and it is practical to apply that as an input. So, the training dataset is created as following

$A = [PCi(2 : end - M, j); V(M + 2 : end, k); Y(1 : end - (M + 1), l)]$

$B = Y(M + 2 : end, l)$

where A and B represent the training datasets for the input and output parameters, respectively. V is the wind speed data points and Y denotes the output.

This type of training is performed for two specific reasons. At first, all the matrices for data points must have the same dimensions. Also, it is intended to use all the data points for training and consequently, training dataset for input features begin with the second data point of the original dataset.

In order to make a prediction that has practical value for wind farms, it is imperative to define values of M in a way that there is sufficient time for operators to take necessary action when required. This implies that the frequency in which the data points are gathered determines how far into the future, the prediction is made. For this study, a dataset with one hour time intervals was created. In other words, when $M = 1$, the prediction is for the average power production in the following hour, and $M = 2$ shows the prediction for the next two hours and so on.

In the following sections, the designed structure of the MLP and ANFIS networks are presented.

MLP structure

MLP networks are established according to the structure shown in Figure 4.6.1. Generally, there is no universal rule about the size of the data required to obtain the best possible training but the data should contain at least the data range boundaries and must be sufficient to represent the entire period [49]. In this study, 50 percent of data was used for the training, leaving the other half for testing. Since the data available covers all seasons, this approach makes it possible to consider seasonal changes in the network without it being a distinct input parameter. To train the networks, the gradient descent with momentum method is applied. In this method, in addition to error calculation, the general error trend will also be determined. This reduces the risk of local minima and results in enhanced generalization [48]. The other important factor in the structure of the network is the number of neurons in the hidden layer. To find the optimum number of neurons, at least 10 runs should be performed while changing only the number of neurons to seek the configuration with the best generalization [49, 50]. This helps to avoid over-fitting. All design criteria are summarized in Table 4.6.1.

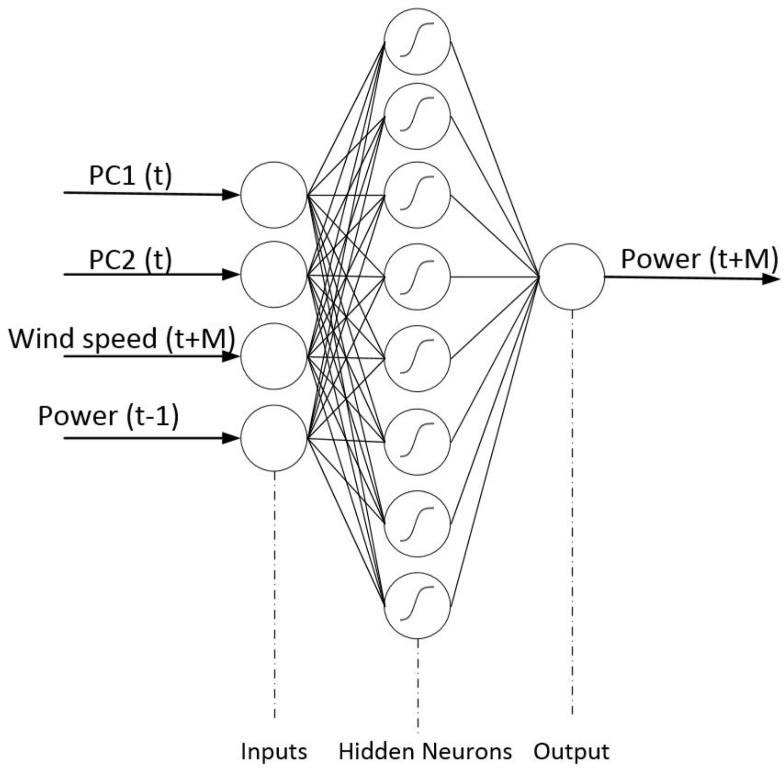

FIGURE 4.6.1: MLP Structure

TABLE 4.6.1: MLP Network Parameters

Type	Feed-forward with back propagation
Error calculation	Gradient descent with momentum
Performance goal	0.0002
Learning rate	0.02
Maximum number of epochs	1000
Activation Function	Sigmoid
Number of neurons in the hidden layer	8

TABLE 4.6.2: ANFIS Network Design Criteria

MF type for inputs	Gaussian
MF type for output	Linear
Number of MFs	3 for each input
Optimization method	Hybrid
Error tolerance	0.0002
Maximum number of epochs	30

ANFIS Network

The major items to be determined for the establishment of an ANFIS network are type and number of membership functions (MFs). MFs are basically arbitrary curves that depend on the kind of application for which they are utilized. Various functions can be applied for input and output parameters to serve as the membership functions. For this study, the Gaussian function is employed for input parameters as it is proven to generate flexibility and work best for the most applications, and for the output parameter, the linear function is applied. The number of MFs is determined by investigating the networks' error while in modeling by fuzzy logic, the number of MFs is selected by a system expert [69]. Theoretically, increasing the number of MFs will decrease the error until it reaches its minimum value and a further increase in the number of MFs will result in higher errors. On the other hand, the larger the number of MFs, the longer the computational time. Thus, to determine the optimum number, the error of each network is calculated using a different number of MFs and when the error stops to considerably decrease, that number is set to be the number of MFs [41]. It is also worth noting that for each input, a different number of MFs can be defined [76]. In this study, three MFs for each input is applied. The ANFIS network structural information is presented in Table 4.6.2.

Prediction Results

Prediction for five different steps is carried out and test errors based on MAE are calculated for MLP and ANFIS networks, and also result of the application of the proposed fusion technique is investigated. The results are illustrated in Figure 4.6.2. As a general trend, increasing the value of M (number of steps) expectedly increases the networks' error for all applied algorithms. Generally, depending on the application and importance, prediction can be made for a various number of steps in the future. It should be noted that the network

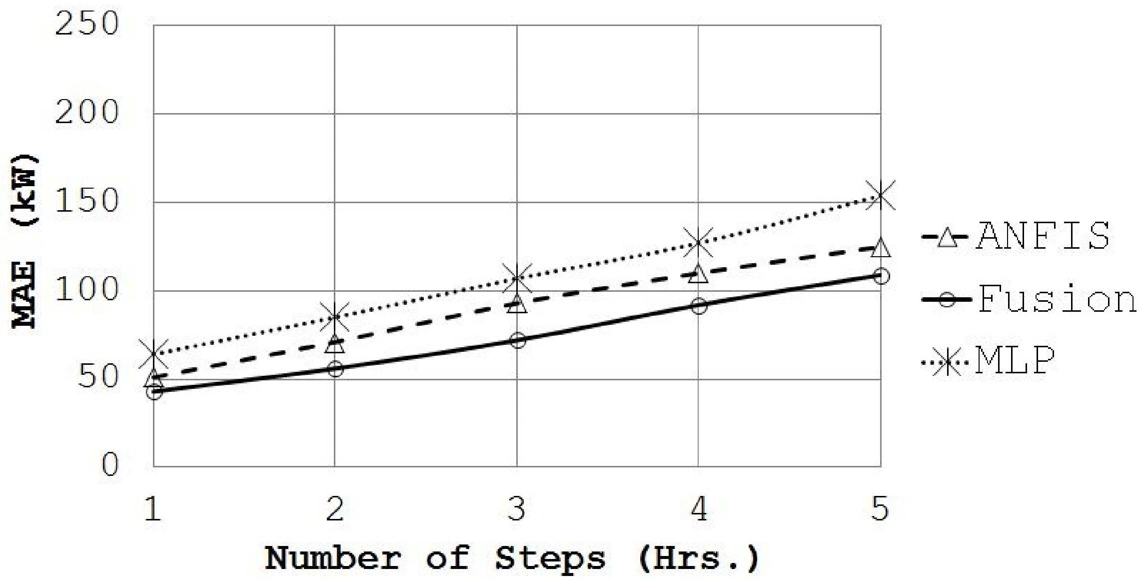

FIGURE 4.6.2: Prediction Results

reliability will decrease significantly with further increase in M since the error rises.

As it can be seen in Figure 4.6.2, ANFIS network slightly outperforms the MLP network and gives rise to more precision. In addition, the test errors of the applied fusion technique are lower than those of MLP and ANFIS networks, which proves the suitability of the technique that can provide more reliability for the established model.

4.7 Conclusion

To effectively reduce the wind farm investment risk and substantially reduce maintenance costs, this study investigated wind turbines power curves. SCADA data obtained from twenty-one, 2.3 MW, pitch regulated wind turbines were used. These data were gathered over a period of 20 months. For the modeling, a combination of Multi-Layer Perceptron (MLP) neural network and Adaptive Neuro-Fuzzy Inference System (ANFIS) as a novel data fusion method was employed to make predictions of the future performance of the machines based on the existing historical information. To pre-process the data for training, missing values and outliers were replaced with imputed values. A variety of imputation algorithms were explained; it was argued that combination of the decision tree and mean value could work efficiently for the significant amount of data points obtained from SCADA system. Also, the steps required to execute the imputation according to this approach were clearly delineated. The imputation method was tested using a dataset containing 5 percent of missing values and imputation error was calculated using NRMS. The small resulting error proved the suitability of the proposed algorithm. This method while not computationally intensive helped generate smoother dataset and consequently, augmented the model reliability.

Principal Component Analysis (PCA) in a recursive manner was employed to extract inputs with the higher level of importance based on the power output. This method enabled the model to update whenever a new block of data became available. This advanced method of feature extraction empowered the network to employ the most significant and influential parameters as the inputs while having the ability to modify itself when the system behavior changed. This was a key element in more accurate prediction results considering the dynamic nature of wind turbines that tend to change over time. In addition to the two extracted principal components, forecasted value of the wind speed was also considered as an input. Finally, in order to create a recurrent network, the last input was designated to the latest available value of machine generated power. The combination of all these methods to collect valuable information for training the networks proved to be well applicable in wind turbine monitoring.

To make a practical prediction of future performance, a dataset with acquiring frequency of one hour was created. It was shown that although both designed MLP and ANFIS networks revealed favorable test errors, the proposed fusion technique resulted in even better accuracy that can be suitably applied for the purpose of wind turbine power curve analysis. This achievement can help operators deliver their contractual obligations by enabling them to monitor the system condition more efficiently. Also, it was observed that as predictions were made further into the future, prediction errors increased. This indicated that performance prediction in a further distance in the future would increase the risk of inaccuracy.

Furthermore, this study illustrated that the application of the fusion technique combined with proposed imputation algorithm, and the employment of PCA to extract significant parameters in the dataset established an advanced monitoring system. Such systems present a more realistic picture of machines performance; thus, helped enhance the benefits of wind energy.

References

[1] A. Heng, S. Zhang, A. C. Tan, and J. Mathew, "Rotating machinery prognostics: State of the art, challenges and opportunities," *Mechanical systems and signal processing*, vol. 23, no. 3, pp. 724–739, 2009.

[2] A. Malhi, R. Yan, and R. X. Gao, "Prognosis of defect propagation based on recurrent neural networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 3, pp. 703–711, 2011.

[3] J. Li, R. Li, Y. Gao, and J. Huang, "Aerodynamic optimization of wind turbine airfoils using response surface techniques," *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, vol. 224, no. 6, pp. 827–838, 2010.

[4] Y. Ju and C. Zhang, "Multi-point robust design optimization of wind turbine airfoil under geometric uncertainty," *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, vol. 226, no. 2, pp. 245–261, 2012.

[5] E. Sagol, M. Reggio, and A. Ilinca, "Issues concerning roughness on wind turbine blades," *Renewable and Sustainable Energy Reviews*, vol. 23, pp. 514–525, 2013.

[6] Y. G. Heo, N. J. Choi, K. H. Choi, H. S. Ji, and K. C. Kim, "Cfd study on aerodynamic power output of a 110kw building augmented wind turbine," *Energy and Buildings*, vol. 129, pp. 162–173, 2016.

[7] A. M. El-Zahaby, A. Kabeel, S. Elsayed, and M. Obiaa, "Cfd analysis of flow fields for shrouded wind turbines diffuser model with different flange angles," *Alexandria Engineering Journal*, 2016.

[8] A. Kumar, A. Dwivedi, V. Paliwal, and P. P. Patil, "Free vibration analysis of al 2024 wind turbine blade designed for uttarakhand region based on fea," *Procedia Technology*, vol. 14, pp. 336–347, 2014.

[9] V. Raman, M. Drissi-Habti, L. Guillaumat, and A. Khadhour, "Numerical simulation

analysis as a tool to identify areas of weakness in a turbine wind-blade and solutions for their reinforcement," *Composites Part B: Engineering*, vol. 103, pp. 23–39, 2016.

[10] C. Yin, Z. Zhang, Z. Wang, and H. Guo, "Numerical simulation and experimental validation of ultrasonic de-icing system for wind turbine blade," *Applied Acoustics*, vol. 114, pp. 19–26, 2016.

[11] J. Chen, Q. Wang, S. Zhang, P. Eecen, and F. Grasso, "A new direct design method of wind turbine airfoils and wind tunnel experiment," *Applied Mathematical Modelling*, vol. 40, no. 3, pp. 2002–2014, 2016.

[12] Q. Li, T. Maeda, Y. Kamada, J. Murata, K. Shimizu, T. Ogasawara, A. Nakai, and T. Kasuya, "Effect of solidity on aerodynamic forces around straight-bladed vertical axis wind turbine by wind tunnel experiments (depending on number of blades)," *Renewable Energy*, vol. 96, pp. 928–939, 2016.

[13] M. Djeziri, B. Ananou, and M. Ouladsine, "Data driven and model based fault prognosis applied to a mechatronic system," in *Power Engineering, Energy and Electrical Drives (POWERENG), 2013 Fourth International Conference on*. IEEE, 2013, pp. 534–539.

[14] S.-Y. Kim, I.-H. Ra, and S.-H. Kim, "Design of wind turbine fault detection system based on performance curve," in *Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012 Joint 6th International Conference on*. IEEE, 2012, pp. 2033–2036.

[15] A. Kusiak, H. Zheng, and Z. Song, "Models for monitoring wind farm power," *Renewable Energy*, vol. 34, no. 3, pp. 583–590, 2009.

[16] W. Yang and J. Jiang, "Wind turbine condition monitoring and reliability analysis by scada information," in *Mechanic Automation and Control Engineering (MACE), 2011 Second International Conference on*. IEEE, 2011, pp. 1872–1875.

[17] V. George, L. Frank, R. Michael, H. Andrew, and W. Biqing, "Intelligent fault diagnosis and prognosis for engineering systems," 2006.

[18] H.-E. Kim, A. C. Tan, J. Mathew, and B.-K. Choi, "Bearing fault prognosis based on health state probability estimation," *Expert Systems with Applications*, vol. 39, no. 5, pp. 5200–5213, 2012.

[19] Z. Yin and J. Hou, "Recent advances on svm based fault diagnosis and process monitoring in complicated industrial processes," *Neurocomputing*, vol. 174, pp. 643–650, 2016.

[20] H. Kim, M. G. Na, and G. Heo, "Application of monitoring, diagnosis, and prognosis in thermal performance analysis for nuclear power plants," *Nuclear Engineering and Technology*, vol. 46, no. 6, pp. 737–752, 2014.

[21] G. Niu and B.-S. Yang, "Dempster–shafer regression for multi-step-ahead time-series prediction towards data-driven machinery prognosis," *Mechanical systems and signal processing*, vol. 23, no. 3, pp. 740–751, 2009.

[22] D. Petković, Ž. Ćojbašič, and V. Nikolić, "Adaptive neuro-fuzzy approach for wind turbine power coefficient estimation," *Renewable and Sustainable Energy Reviews*, vol. 28, pp. 191–195, 2013.

[23] A. Lorton, M. Fouladirad, and A. Grall, "A methodology for probabilistic model-based prognosis," *European Journal of Operational Research*, vol. 225, no. 3, pp. 443–454, 2013.

[24] W. Wang, "A two-stage prognosis model in condition based maintenance," *European Journal of Operational Research*, vol. 182, no. 3, pp. 1177–1187, 2007.

[25] C. Chen, D. Brown, C. Sconyers, B. Zhang, G. Vachtsevanos, and M. E. Orchard, "An integrated architecture for fault diagnosis and failure prognosis of complex engineering systems," *Expert Systems with Applications*, vol. 39, no. 10, pp. 9031–9040, 2012.

[26] J. Son, Q. Zhou, S. Zhou, X. Mao, and M. Salman, "Evaluation and comparison of mixed effects model based prognosis for hard failure," *IEEE Transactions on Reliability*, vol. 62, no. 2, pp. 379–394, 2013.

[27] S. Liu, Y. Wang, and F. Tian, "Prognosis of underground cable via online data-driven method with field data," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7786–7794, 2015.

[28] Z. Wang, C. Hu, W. Wang, X. Si, and Z. Zhou, "An off-online fuzzy modelling method for fault prognosis with an application," in *Prognostics and System Health Management (PHM), 2012 IEEE Conference on*. IEEE, 2012, pp. 1–7.

[29] M. Yu, D. Wang, M. Luo, and L. Huang, "Prognosis of hybrid systems with multiple incipient faults: Augmented global analytical redundancy relations approach," *IEEE Transactions on systems, man, and cybernetics-part A: systems and humans*, vol. 41, no. 3, pp. 540–551, 2011.

[30] J. W. Sheppard and M. A. Kaufman, "A bayesian approach to diagnosis and prognosis using built-in test," *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 3, pp. 1003–1018, 2005.

[31] J. Hu, L. Zhang, L. Ma, and W. Liang, "An integrated safety prognosis model for complex system based on dynamic bayesian network and ant colony algorithm," *Expert Systems with Applications*, vol. 38, no. 3, pp. 1431–1446, 2011.

[32] Q. Liu, M. Dong, W. Lv, X. Geng, and Y. Li, "A novel method using adaptive hidden semi-markov model for multi-sensor monitoring equipment health prognosis," *Mechanical Systems and Signal Processing*, vol. 64, pp. 217–232, 2015.

[33] M. Dong and D. He, "Hidden semi-markov model-based methodology for multi-sensor equipment health diagnosis and prognosis," *European Journal of Operational Research*, vol. 178, no. 3, pp. 858–878, 2007.

[34] R. K. Singleton, E. G. Strangas, and S. Aviyente, "Discovering the hidden health states in bearing vibration signals for fault prognosis," in *Industrial Electronics Society, IECON 2014-40th Annual Conference of the IEEE*. IEEE, 2014, pp. 3438–3444.

[35] G. Li, S. J. Qin, Y. Ji, and D. Zhou, "Reconstruction based fault prognosis for continuous processes," *Control Engineering Practice*, vol. 18, no. 10, pp. 1211–1219, 2010.

[36] W. G. Zanardelli, E. G. Strangas, H. K. Khalil, and J. M. Miller, "Wavelet-based methods for the prognosis of mechanical and electrical failures in electric motors," *Mechanical Systems and Signal Processing*, vol. 19, no. 2, pp. 411–426, 2005.

[37] B. Zhou, H. Ye, H. Zhang, and M. Li, "Process monitoring of iron-making process in a blast furnace with pca-based methods," *Control engineering practice*, vol. 47, pp. 1–14, 2016.

[38] E. Cai, L. Liang, G. Xu *et al.*, "Monitoring of chemical industrial processes using integrated complex network theory with pca," *Chemometrics and Intelligent Laboratory Systems*, vol. 140, pp. 22–35, 2015.

[39] J. Camacho, A. Pérez-Villegas, P. García-Teodoro, and G. Maciá-Fernández, "Pca-based multivariate statistical network monitoring for anomaly detection," *Computers & Security*, vol. 59, pp. 118–137, 2016.

[40] A. Kusiak, H. Zheng, and Z. Song, "On-line monitoring of power curves," *Renewable Energy*, vol. 34, no. 6, pp. 1487–1493, 2009.

[41] M. Schlechtingen, I. F. Santos, and S. Achiche, "Using data-mining approaches for wind turbine power curve monitoring: a comparative study," *IEEE Transactions on Sustainable Energy*, vol. 4, no. 3, pp. 671–679, 2013.

[42] D. Spera, "Wind turbine technology–fundamental concepts of wind turbine technology," 1998.

[43] M. G. Rahman and M. Z. Islam, "Missing value imputation using decision trees and decision forests by splitting and merging records: Two novel techniques," *Knowledge-Based Systems*, vol. 53, pp. 51–65, 2013.

[44] X. Zhu, S. Zhang, Z. Jin, Z. Zhang, and Z. Xu, "Missing value estimation for mixed-attribute data sets," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 1, pp. 110–121, 2011.

[45] A. Farhangfar, L. Kurgan, and J. Dy, "Impact of imputation of missing values on classification error for discrete data," *Pattern Recognition*, vol. 41, no. 12, pp. 3692–3705, 2008.

[46] A. Farhangfar, L. A. Kurgan, and W. Pedrycz, "A novel framework for imputation of missing values in databases," *IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans*, vol. 37, no. 5, pp. 692–709, 2007.

[47] R. Blake and P. Mangiameli, "The effects and interactions of data quality and problem complexity on classification," *Journal of Data and Information Quality (JDIQ)*, vol. 2, no. 2, p. 8, 2011.

[48] M. Schlechtingen and I. F. Santos, "Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection," *Mechanical systems and signal processing*, vol. 25, no. 5, pp. 1849–1875, 2011.

[49] K. Swingler, *Applying neural networks: a practical guide*. Morgan Kaufmann, 1996.

[50] P. Caselitz and J. Giebhardt, "Advanced maintenance and repair for offshore wind farms using fault prediction techniques," in *Proceedings of the World Wind Energy Conference*, 2002.

[51] A. Frank, A. Asuncion *et al.*, "Uci machine learning repository," 2010.

[52] J. W. Graham, "Missing data analysis: Making it work in the real world," *Annual review of psychology*, vol. 60, pp. 549–576, 2009.

[53] Z. Meng and Z. Shi, "Extended rough set-based attribute reduction in inconsistent incomplete decision systems," *Information Sciences*, vol. 204, pp. 44–69, 2012.

[54] S. Zhang, "Shell-neighbor method and its application in missing data imputation," *Applied Intelligence*, vol. 35, no. 1, pp. 123–133, 2011.

[55] P. J. García-Laencina, J.-L. Sancho-Gómez, and A. R. Figueiras-Vidal, "Pattern classification with missing data: a review," *Neural Computing and Applications*, vol. 19, no. 2, pp. 263–282, 2010.

[56] Y. Ding and A. Ross, "A comparison of imputation methods for handling missing scores in biometric fusion," *Pattern Recognition*, vol. 45, no. 3, pp. 919–933, 2012.

[57] X. Yang, B. Huang, Y. Zhao, Y. Lu, W. Xiong, and H. Gao, "Generalized expectation–maximization approach to lpv process identification with randomly missing output data," *Chemometrics and Intelligent Laboratory Systems*, vol. 148, pp. 1–8, 2015.

[58] K.-C. Ng and B. Abramson, "Consensus diagnosis: A simulation study," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 22, no. 5, pp. 916–928, 1992.

[59] E. Herrera-Viedma, O. Cordón, M. Luque, A. Lopez, and A. Muñoz, "A model of fuzzy linguistic irs based on multi-granular linguistic information," *International Journal of Approximate Reasoning*, vol. 34, no. 2-3, pp. 221–239, 2003.

[60] J. Kacprzyk and S. Zadrożny, "Computing with words in intelligent database querying: standalone and internet-based applications," *Information Sciences*, vol. 134, no. 1, pp. 71–109, 2001.

[61] X. Liu, "Some properties of the weighted owa operator," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 36, no. 1, pp. 118–127, 2006.

[62] V. Torra, "Owa operators in data modeling and reidentification," *IEEE Transactions on Fuzzy Systems*, vol. 12, no. 5, pp. 652–660, 2004.

[63] D. Filev and R. R. Yager, "Learning owa operator weights from data," in *Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on*. IEEE, 1994, pp. 468–473.

[64] ——, "On the issue of obtaining owa operator weights," *Fuzzy sets and systems*, vol. 94, no. 2, pp. 157–169, 1998.

[65] K. Salahshoor, M. Kordestani, and M. S. Khoshro, "Design of online soft sensors based on combined adaptive pca and rbf neural networks," in *Computational Intelligence in Control and Automation, 2009. CICA 2009. IEEE Symposium on*. IEEE, 2009, pp. 89–95.

[66] A. E. Shrme, "Hybrid intelligent technique for automatic communication signals recognition using bees algorithm and mlp neural networks based on the efficient features," *Expert Systems with Applications*, vol. 38, no. 5, pp. 6000–6006, 2011.

[67] T. Waqar and M. Demetgul, "Thermal analysis mlp neural network based fault diagnosis on worm gears," *Measurement*, vol. 86, pp. 56–66, 2016.

[68] H. Takagi and I. Hayashi, "Nn-driven fuzzy reasoning," *International Journal of Approximate Reasoning*, vol. 5, no. 3, pp. 191–212, 1991.

[69] J.-S. Jang, "Anfis: adaptive-network-based fuzzy inference system," *IEEE transactions on systems, man, and cybernetics*, vol. 23, no. 3, pp. 665–685, 1993.

[70] E. H. Mamdani, "Application of fuzzy logic to approximate reasoning using linguistic synthesis," in *Proceedings of the sixth international symposium on Multiple-valued logic*. IEEE Computer Society Press, 1976, pp. 196–202.

[71] M. Sugeno, *Industrial applications of fuzzy control*. Elsevier Science Inc., 1985.

[72] A. Hasiloglu, M. Yilmaz, O. Comakli, and İ. Ekmekci, "Adaptive neuro-fuzzy modeling of transient heat transfer in circular duct air flow," *International Journal of Thermal Sciences*, vol. 43, no. 11, pp. 1075–1090, 2004.

[73] K. Salahshoor, M. Kordestani, and M. S. Khoshro, "Fault detection and diagnosis of an industrial steam turbine using fusion of svm (support vector machine) and anfis (adaptive neuro-fuzzy inference system) classifiers," *Energy*, vol. 35, no. 12, pp. 5472–5482, 2010.

[74] S. Wold, "Exponentially weighted moving principal components analysis and projections to latent structures," *Chemometrics and intelligent laboratory systems*, vol. 23, no. 1, pp. 149–161, 1994.

[75] W. Li, H. H. Yue, S. Valle-Cervantes, and S. J. Qin, “Recursive pca for adaptive process monitoring,” *Journal of process control*, vol. 10, no. 5, pp. 471–486, 2000.

[76] M. Schlechtingen, I. F. Santos, and S. Achiche, “Wind turbine condition monitoring based on scada data using normal behavior models. part 1: System description,” *Applied Soft Computing*, vol. 13, no. 1, pp. 259–270, 2013.

CHAPTER 5

Conclusion and Future Work

5.1 Contributions

In this study, wind turbines power curves were analyzed based on the SCADA data obtained from twenty-one, 2.3 MW, pitch regulated wind turbines. The data covered a period of 20 months from February 2014 to September 2015. The intention was to create a reliable condition monitoring system that could furnish a clearer picture of the performance of the turbine for operators. This would minimize false alarms helping wind farms manage the operations more efficiently, lowering the wind energy investment risks.

Unlike numerical or experimental methods, data analysis approach proved to be effective in addressing the current issues related to commercialized wind turbines. For that purpose, numerous data mining tools and algorithms along with intelligent machine learning techniques were studied to apply the most suitable ones for the SCADA data. This study resulted in three manuscripts for which it was attempted to increase the accuracy and reliability of the established models employing advanced data mining and machine learning tools.

In the first paper, a novel method of feature selection and dynamic structures of artificial neural networks created a model with high level of accuracy. Comparisons with the existing models in the literature showed an improvement of about 30 percent in power production estimation.

In the second and third manuscripts, the focus was future performance prediction of wind turbines as opposed to power estimation. For the second one, an imputation method was proposed for the large SCADA data, combining the concept of decision trees and mean value. It was shown that this method, while not very complex nor computationally extensive, resulted in very low imputation error. Additionally, Adaptive Neuro-Fuzzy Inference System (ANFIS) which was a combination of fuzzy logic and neural networks was applied to build a model for the purpose of prediction. Two different conditions were created to examine the effectiveness of the imputation. It was concluded that although ANFIS model gave rise to acceptable prediction results, performing imputation could promote the model accuracy significantly, declining the networks test errors.

The third paper strived to make accurate predictions as much as possible. For that goal, at first, data pre-processing included the proposed imputation technique in the second paper. Then, a recursive Principal Component Analysis (PCA) was employed to extract two main Principal Components to be applied as the inputs to the network, along with forecasted value of wind speed and the dynamic term for the power production. Finally, a novel data fusion technique was introduced that combined MLP and ANFIS classifiers to reduce the error. It was shown that the proposed integration of two methods created a more advanced model that could consistently provide accurate and reliable predictions for wind turbines power generation.

According to European Wind Energy Association (EWEA), an onshore wind turbine with similar capacity to the investigated turbines in this study can produce more than 6 million kWh in a year or roughly about 16440 kWh per day. Since the Province of Ontario pays 11-13.5 cents per kWh for wind power (windontario.ca), the financial loss resulted from a turbine shut down would be around $2000 per day, not including the the repair costs. This fact highlights the importance of installation of a reliable condition monitoring system.

5.2 Future Steps

Based on the available data for this study, the focus was on the condition monitoring of the turbines and for that, many algorithms and tools were studied and applied. Since there are a countless number of such techniques and also, newer and more advanced methods are introduced by the researchers for numerous purposes, other techniques can be investigated for all aspects of data pre-processing, feature extraction, and model establishment.

In this study, only the turbine's power curve was investigated. There are as many as 150 signals in a wind turbine SCADA data and other parameters can also be the subject of interesting studies.

Moreover, the future studies in this area can focus on fault diagnosis, prognosis, and finally Remaining Useful Life estimation (RUL). These methods can be pursued depending on the data availability. For instance, if historical data is available for a system in which a specific fault has occurred, that trend of data can be classified as that specific fault and models can be trained to detect it as opposed to normal behavior trends. Similarly, if data shows a degradation in the system or components performance, it can be used to estimate remaining useful life of various components which may lead to prognostic and predictive maintenance and ultimately cost reduction.

VITA AUCTORIS

NAME:	Majid Morshedizadeh
PLACE OF BIRTH:	Tehran, Iran.
EDUCATION:	Bachelor of Science in Mechanical Engineering, K.N.T University of Technology, Tehran, Iran, 2008.
	Master of Applied Science in Mechanical Engineering, University of Windsor, Windsor, Ontario, Canada, 2017.