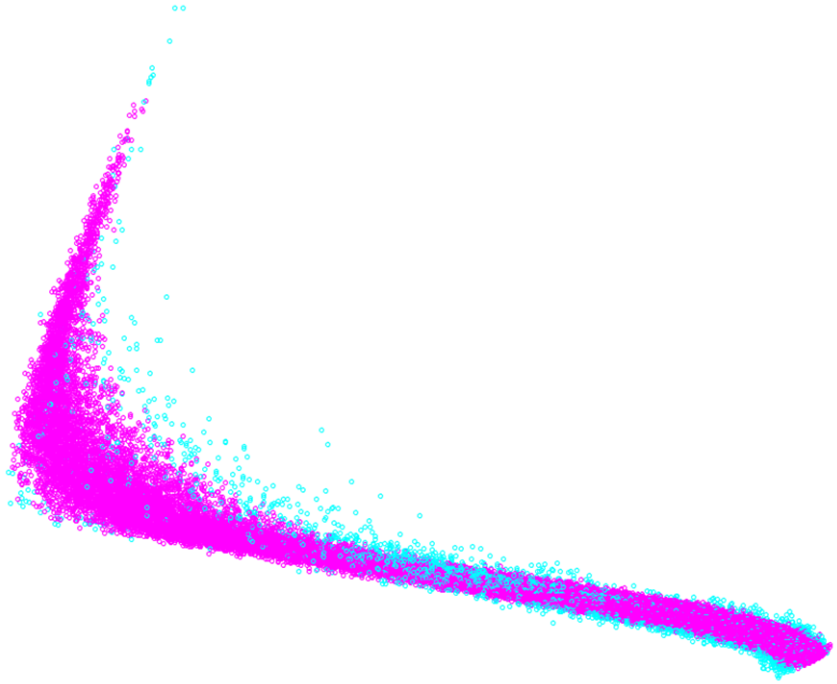
CHALMERS
UNIVERSITY OF TECHNOLOGY



Wind Turbine Performance Monitoring using Artificial Neural Networks

With a Multi-Dimensional Data Filtering Approach

Master's thesis in Sustainable Energy Systems

DANIEL KARLSSON

Department of Energy and Environment
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2014

MASTER'S THESIS

Wind turbine performance monitoring using Artificial Neural Networks

With a Multi-Dimensional Data Filtering Approach

Master's Thesis within the *Sustainable Energy Systems* programme

DANIEL KARLSSON

SUPERVISORS:

Ola Carlsson

Pramod Bangalore

EXAMINER

Ola Carlsson

Department of Energy and Environment

Division of Electric Power Engineering

CHALMERS UNIVERSITY OF TECHNOLOGY

Göteborg, Sweden 2015

Wind turbine performance monitoring using Artificial Neural Networks
With a Multi-Dimensional Data Filtering Approach
Master's Thesis within the *Sustainable Energy Systems* programme
DANIEL KARLSSON

© DANIEL KARLSSON, 2015

Department of Energy and Environment
Division of Electric Power Engineering
Chalmers University of Technology
SE-412 96 Göteborg
Sweden
Telephone: + 46 (0)31-772 1000

Cover:
The figure shows, in 3D, the results from the filtering method described in Section 5.4.5.

Chalmers Reproservice
Göteborg, Sweden 2015

Wind turbine performance monitoring using Artificial Neural Networks
With a Multi-Dimensional Data Filtering Approach
Master's Thesis in the *Sustainable Energy Systems* programme
DANIEL KARLSSON
Department of Energy and Environment
Division of Electric Power Engineering
Chalmers University of Technology

ABSTRACT

The wind power sector has grown rapidly and has become a substantial part of the global sustainable energy production. Performance and condition monitoring systems are gaining ground, but most faults are still detected during planned maintenance. This can lead to long time periods of underperformance, which translates to lost revenues.

In this thesis, Artificial Neural Networks (ANN) are used to model the normal behaviour of a wind turbine, which could be used for real-time monitoring of operations. A number of other studies that use ANN's to predict wind power output were found during the literature study; but this thesis presents a new direction where the standard deviation of the wind speed is used as an input to the model, as well as a multi-dimensional filtering method, meant to exclude outliers in the training set with higher accuracy than conventional filtering techniques.

The study follows the method of (Schlechtingen, et al., 2013a), who made a comparative study of different data-mining approaches, to be able to compare the model results. The proposed model shows an improvement in prediction performance of between 16 % and 22 %, depending on performance parameter. The results from the multi-dimensional filtering shows that unhealthy data situated inside what is conventionally thought of as normal operating range can be excluded with the proposed method.

It is concluded that the model is well suited for performance monitoring, but its applicability to fault prediction could ultimately not be concluded due to a lack of suitable faults during the period. Finally, it is concluded that if the proposed model had been used for performance monitoring in the turbine that was the main subject in this study, earlier maintenance could have resulted in an additional electricity generation of up to 270 MWh during the three years of data used.

Key words: Artificial Neural Networks, performance monitoring, data clustering, supervisory control and data acquisition (SCADA), data mining, fault detection, renewable energy, wind power

Prestandaövervakning för vindkraftverk genom artificiella neurala nätverk
Med en multidimensionell filtreringsteknikstrategi
Examensarbete inom masterprogrammet *Sustainable Energy Systems*
DANIEL KARLSSON
Institutionen för Energi och Miljö
Avdelningen för Elteknik
Chalmers tekniska högskola

SAMMANFATTNING

Allt eftersom vindkraftssektorn växer och är på väg att bli en betydande del av den globala förnybara elproduktionen, ökar behovet av prestandaövervakningssystem. Övervakningssystem för att tidigt upptäcka fel och undvika förestående haverier utvecklas och tar sig sakta in på marknaden, men de flesta fel upptäcks fortfarande vid inspektion eller inte alls, förrän maskinen havererar. Detta kan leda till långa tidsperioder med försämrad prestanda och förlorade intäkter.

I detta examensarbete används artificiella neurala nätverk (ANN) för att modellera normalbeteende hos ett vindkraftverk, vilket skulle kunna användas för att övervaka prestandan i realtid. Ett flertal studier där ANN används för att förutsäga produktionen hos ett vindkraftverk har hittats under litteraturstudien; i denna tes har konceptet utvecklats ytterligare genom att innefatta standardavvikelsen i vindhastigheten som insignal till modellen. Dessutom har ett multidimensionellt datafilter utvecklats, som är tänkt att hitta felaktig träningsdata med större precision än konventionella filter.

Studien är utformad för att kunna jämföra resultaten med en komparativ studie gjord av (Schlechtingen, et al., 2013a), där en rad liknande tekniker utvärderats. I jämförelsen visar den föreslagna modellen en prestandaförbättring på mellan 16 % och 22 %, beroende på vilken prestandaparameter som jämförs. Den framtagna filtertekniken visar sig även kunna utesluta datapunkter som befinner sig i vad som konventionellt brukar ses som det normala arbetsområdet för ett vindkraftverk.

Slutsatsen som dras i studien är att modellen lämpar sig väl för prestandaövervakning, men att det är svårt att säga huruvida den kan användas till att förutspå haverier, då det ej skett större fel som skulle kunna ha förutspåtts med de valda inparametrarna under tiden för undersökningen. Slutligen konstateras det att om den föreslagna modellen hade använts i verket under de tre år som data finns för, hade fel kunnat upptäckas tidigare, vilket skulle kunnat leda till en ökad elproduktion på upp till 270 MWh.

Nyckelord: Artificiella neurala nätverk, prestandaövervakning, klusteranalys, SCADA, feldetektering, förnybar energi, vindkraft

Contents

Preface

This report is the result of a 30 credits Master's Thesis at Chalmers University of Technology for the Division of Electric Power Engineering, at the Department of Energy and Environment. The project has been a collaboration between Chalmers, the non-profit organisation Miljöbron and the wind power company GreenExtreme, and has been carried out under the period of January to June, 2015.

I would like to thank the employees at GreenExtreme for this opportunity to conduct research in an exciting field, a welcoming atmosphere and insight to the wind power industry.

I would also like to thank Ola Carlson for being my examiner in this project, my supervisor Pramod Bangalore for his guidance and encouragement when I needed it the most, Simon Letzgus for the invaluable discussions around the complicated concepts of the study and a big thank you to my friend Joel who showed me the ropes in MSQL. Also the rest of the students down in the basement deserves a mentioning, who have made this last intensive period bearable.

And lastly, I thank my family, friends and especially my partner Emma for the support and understanding during these last intensive weeks.

Daniel Karlsson

Gothenburg, 2015

Acronyms

AC	Alternating Current
ANFIS	Adaptive Neuro-Fuzzy Interference System
ANN	Artificial Neural Networks
CBM	Condition-Based Maintenance
CCFL	Cluster Center Fuzzy Logic
CMS	Condition Monitoring System
DC	Direct Current
DFIG	Doubly-Fed Induction Generator
FF	Feed-Forward
GF	Generalisation Factor
GMM	Gaussian Mixture Model
k-NN	k-Nearest Neighbour
LMedS	Least Median of Squares
MAE	Mean Absolute Error
MAPE	Mean Absolute Percentage Error
MD	Mahalanobis Distance
MSE	Mean Squared Error
MQE	Minimum Quantization Error
NARX	Nonlinear Autoregressive Network with Exogenous Inputs
PDF	Probability Density Function
RMSE	Root Mean Squared Error
SCADA	Supervisory Control and Data Acquisition
SD/STD	Standard Deviation
SOM	Self-Organising Map
SEMS	Self-Evolving Maintenance Scheduler
VPN	Virtual Private Network
WT	Wind Turbine

1 Introduction and background

The initial project description for this thesis was rather brief and mainly stated that the company held historical data from the wind turbines they manage, and that they wished to increase the efficiency of these. This left the direction of the project open for interpretation and at the start, a number of different approaches were investigated. It became obvious that optimisations involving pitch angles required in-depth knowledge of machine coding, instruments for high frequency measurements and that changing controller settings could void service agreements.

Another direction that was considered was to investigate the wake effect from nearby wind turbines, since the company have a few sites where multiple turbines are located. But the site most suitable for this kind of project is newly established, so there would not be enough historical data to make this project viable. The other sites at most have two turbines, which makes the wake effect close to negligible.

In addition to project planning, the company also manages the wind turbines for the investors. To show good return of investment for their clients and possible investors, it is important that the turbines have high availability and produce at their maximum capacity. At present, this is done by scheduled maintenance and a yearly summary, where the wind index (a figure based on the location-specific average wind) and production numbers are checked. Down-time and underperformance could have a potentially large negative impact on production.

Going through one of the summaries over the past year, it was discovered that the turbine at hand had produced less electricity than the year before, despite equal availability and high wind index. It seemed there was room for improvement in this manual check-up strategy, hence a literature study of the subject was made.

Traditionally, the power output is estimated by using a power curve that is unique to each turbine model. A power curve only takes average wind speed into account, and the difference between measured and estimated output can be large. The initial plan was to try to construct a power curve that would be dependent on more than just wind speed, namely; outdoor temperature as this changes the energy content of the wind and wind direction since obstacles in the surroundings of on-shore wind turbines can affect the turbulence patterns of the wind (This will be further discussed in Chapter 2). The literature study showed that a lot of work was done in this area and unable to find a new angle of the problem, the study was broadened and the use of Artificial Neural Networks (ANN) in wind power seemed to be an interesting view of the problem.

Most studies which fuse the ANN concept with wind power, focus on fault prediction in certain parts of the turbine; such as gearbox, bearings and other parts with a high fault rate. Developing the initial thought about a more precise power curve, it was thought that using the flexibility off ANNs and the large amount of data that was gathered, a well-performing model of turbine power production behaviour might be constructed.

A model which predicts the output power with high certainty could be used to monitor the power production in real-time and warn if there is a trend of underperformance. This information could indicate an impending failure in a component. Such an indication can be used by the owner to schedule maintenance at an early stage. If faults and underperformance is found at an early stage, the total production will be increased and downtime may be reduced, hence increasing the overall efficiency.

1.1 Sustainable energy systems and the role of wind power

With the mechanisation of industry and increased production output, the demand of a steady flow of energy grew. The heart of the industrialisation, the United Kingdom, had an abundance of coal which came to be the most important piece in the development of production industry. With the discovery of oil, the transportation sector took off and made it possible for people to travel far distances and transport goods across the globe. To fuel this development, a steady supply of energy was needed, and the most important part in this system was fossil fuels and their seemingly endless resources. (Grübler, 1998)

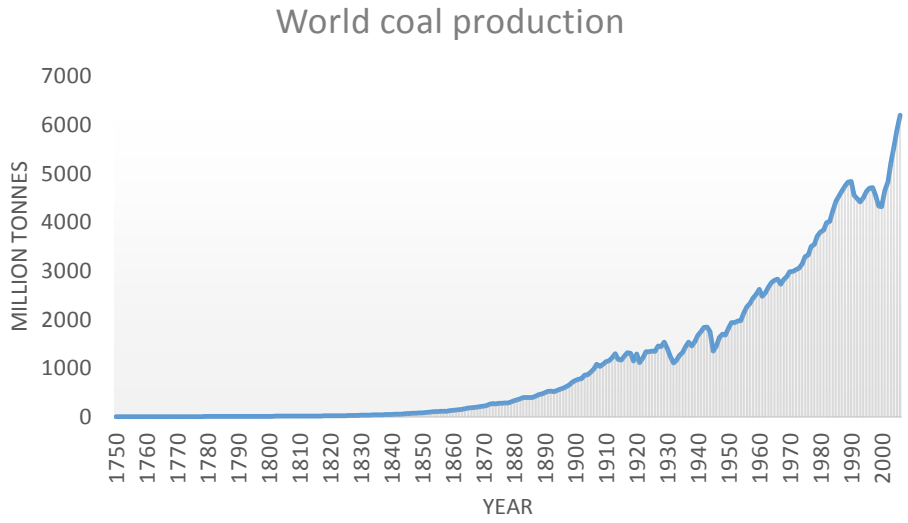


Figure 1.1 The annual coal production from year 1750 to 2000, diagram created with statistics from (Mohr & Evans, 2009).

This paradigm shift that was the industrial revolution, completely changed the foundation of human society. Mass production, industrial innovation and international trade has led to a world that is constantly shrinking, where people have grown accustomed to, and *demand*, affordable energy and transportation.

The greenhouse effect was argued for in the early 19th century by Joseph Fourier, and it was further argued by a number of scientist in the next century, that anthropogenic carbon dioxide could impact the climate. It was not until 1988, when the World Meteorological Organization and The United Nations Environment Program created the Intergovernmental Panel on Climate Change, that world leaders got their eyes open to the negative side effects of the burning of fossil fuels. A few years earlier it had become obvious that anthropogenic emissions of gasses could have massive negative impact on our environment, as scientists discovered that the emissions of refrigerants caused ozone depletion. (Ponting, 2007)

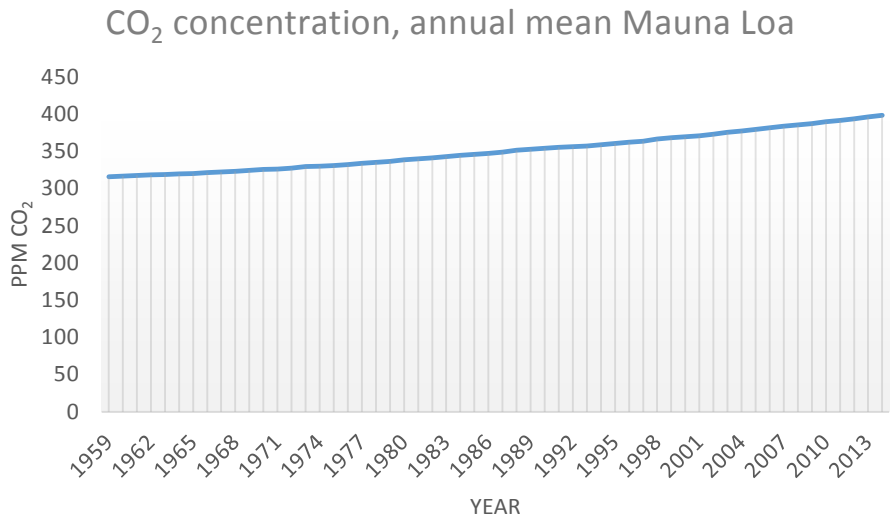


Figure 1.2 Measured concentration of carbon dioxide from 1959 to 2013. Diagram created using data from (National Oceanic & Atmospheric Administration, 2015).

The human society has come to depend on energy in form of electricity and fuel to such an extent, that the agreed upon transition to renewable energy sources will take generations to accomplish. In a positive aspect, there are examples of countries that has succeeded to change their electricity generation towards an energy system built on renewable sources, despite meagre natural resources such as hydropower. In 2014, 39 % of the electricity generated in Denmark came from wind turbines (Energinet.dk, 2015), showing that the transition to a sustainable energy system is possible.

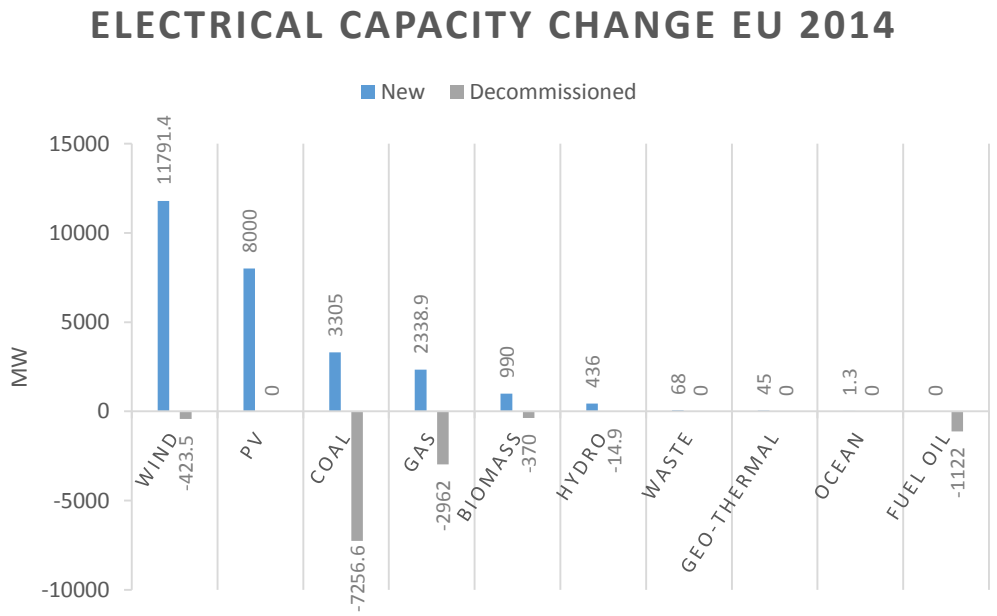


Figure 1.3 The total change of electricity generating capacity in the EU during 2014. Diagram created using statistics from (EWEC, 2015).

The recent global development of wind power follows a clear trend, where wind power in a few years will be a substantial part of the global energy mix. It is forecasted in the scenario with the lowest target scenario that in year 2020, about 6 % of the global electricity production will come from wind energy and in 2030, the figure will be close to 9 % (Global Wind Energy Council, 2014).

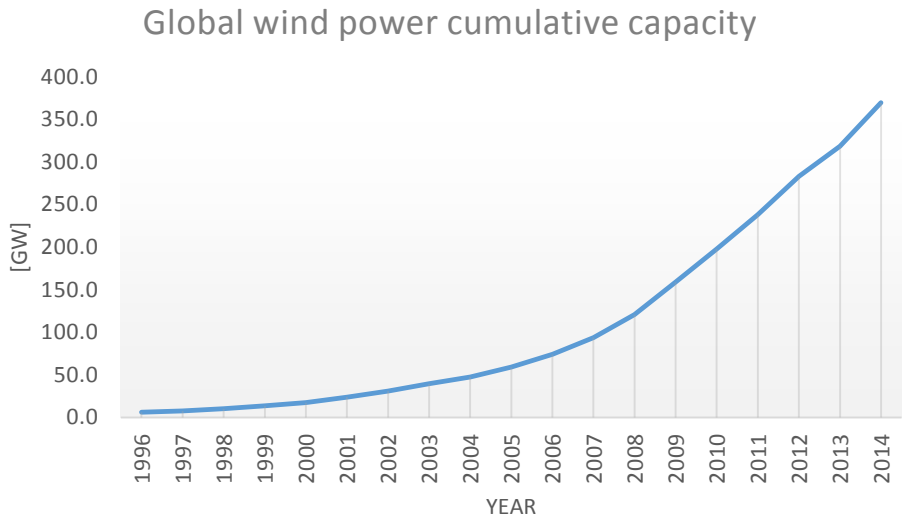


Figure 1.4 The trend of global wind power development is clear, considering only the last five years, the installed capacity has nearly doubled. Diagram created with data from (Global Wind Energy Council, 2014).

In regions such as Sweden, where hydroelectric power plants are used to a large extent, the possibility to complement the intermittent characteristics of the wind. By decreasing the output from the hydro plant when there is high wind, the energy can be stored in the dam until the extra power is needed again. By also having a strong cross-border electrical grid, import and export of power can be utilised when needed and further facilitate a greater penetration of wind power in the electrical system. (Purvins, et al., 2012)

1.2 Aim of thesis

The main aim of this thesis is to create a model that use values from the supervisory control and data acquisition (SCADA) system, to predict the output power of a wind turbine. This model presents a systematic approach to use SCADA data to detect under performance. This is an improvement to the conventional practice of performance monitoring through time-based availability which has limited accuracy, as it does not consider possible productivity during the given period. Moreover this systematic approach is easier to apply compared to the subjective analysis, which becomes tedious with increasing number of wind turbines.

A secondary aim is to investigate whether the model can be used to identify turbine failures at an early stage, thereby providing the owner with time to plan repairs and avoid long costly down times.

1.3 Problem description

There are a number of problems involved in the formulation of the model and its interpretation. The main points are concluded in the following list:

Filter the training set to avoid abnormal behaviour being trained into the model

The intention of the project is to create a model that predicts the power output of a turbine as accurately as possible. To accomplish this, data deemed as representative of normal behaviour need to be fed to the neural network as training set. To create a well performing filter is a very important part of the project.

Decide the best architecture and configuration for the neural network

There are a few different network architectures available, and to find the best one will require some testing. There are some studies made on the subject of network configuration and the main conclusion being that there are no apparent rules as how to configure a network. There are some general rules-of-thumb, but statistical tries are needed to find a good configuration.

Find a network that is both good in performance as well as generalising

The data set is divided into three parts; training, test and validation set. The training algorithm uses the training and test set when it is searching for the best weights for a high performance network. The validation set contains data that the network has not seen before, testing its generalisation. Performance generally increases with an increased number of neurons, but generalisation, the ability to interpret data the network has not "seen" before can degrade as the net is *over-fitted*. Hence a way to find a net that performs well in both aspects is important.

Analysing the residues

The output from the model is a predicted value depending on the inputs at each point in time. The difference between actual target and predicted output, the residue, needs to be interpreted using statistical methods to have real significance.

1.4 Project boundaries

GreenExtreme have a number of turbines, from different manufacturers, spread across the western parts of Sweden. In this thesis, five turbines from one of these manufacturers will be studied. The main reason being that it is these turbines that has the most data logged and that the in-house knowledge is the greatest here.

During interviews with the employees at the company, it became clear that one of the turbines under inspection had produced less than predicted during 2014. This turbine is also the one where most data is available, thus it was chosen as main subject for the analysis. The remaining four turbines have had long downtimes and the data for these

is not as extensive, which means that the training set for these will not be as large as for the study's main turbine.

The methods used in this thesis are at times very complicated and demanding in computational time, some simplifications of the statistical models have to be made in order to efficiently perform this study.

1.5 Outline of thesis

Chapter 1 sets the background of the thesis and describes the aims and problems that will be the main focus of the study.

Chapter 2 presents the theory behind electricity production from wind power. The planning and economical calculations are also briefly described, followed by maintenance strategies and specifications of the wind turbines considered in the thesis.

Chapter 3 explains the concept of Artificial Neural Networks and the most common network structures and the algorithms used to train them. The importance of data filtering is stressed, inherent potential problems are brought up, as well as the procedure of constructing a well-functioning network.

Chapter 4 takes up some common statistical instruments needed for the analysis of the data. Key concepts are explained in a visual way to simplify the understanding of these.

Chapter 5 contains a thorough explanation of all the steps in the study. Starting with assumptions made, followed by the procedure of the data import. The following section shows the architecture and configuration of the net and the method to determine this. Reasoning behind the selection of inputs and outputs are motivated, measures to enhance performance and generalisation are explained. All the steps in the selection of training data is explained in the following section, including manual selection, validation of data, clustering and the multi-dimensional filtering.

Chapter 6 presents the results of the filtering algorithm, the final ANN architecture, and the two main parts of the model result, under-performance detection and fault detection. The results are shown in tables and graphs, they are also compared to another similar study to benchmark the performance.

Chapter 7 contains the final conclusions of the thesis with a discussion and proposed future work.

Chapter 8 lists the references used in the thesis

Chapter 9 holds the appendices to the thesis

2 Wind power

The usage of wind as a power source dates back to at least 5000 B.C, when the first documented sailing boats were used on the Nile River. The Persian Empire used wind power to pump water and grind grain during the first millennium B.C, and in the first century A.D, the Greek engineer Heron constructed the first machine powered by the wind. By the 11th century, the windmills had spread and were used in Europe, Asia and the Middle East. (Paipetis & Ceccarelli, 2010)

The first windmill exclusively used for electricity generation was constructed in Scotland by James Blyth in 1887, and was used to charge accumulators that powered his holiday cottage in Marykirk. The Burgin dynamo was driven by the turbine's flywheel, connected by rope, and could power ten 25 Volt light bulbs in moderate winds. Shortly thereafter, in 1888, Charles F. Brush finished the construction of the world's first fully automated windmill. It had a 12 kW dynamo and supplied Brush's mansion with electricity for 20 years without failures. (Price, 2008)

By the end of the 1900's the mass production of wind turbines with sizes ranging from 5 kW to 25 kW had started in America. At the time of World War I, windmill makers were producing 100,000 units per year, though mostly for pumping water, the wind turbines had started to gain momentum.

In 1931 a 100 kW turbine resembling modern wind turbines was connected to the local distribution system in Yalta, USSR. A decade later the first megawatt-class wind turbine, Grandpas Knob, was connected to the grid in Vermont, USA. It was a two-bladed horizontal axis wind turbine with a 1.25 MW synchronous generator. Due to war-time shortage of materials, the turbine blades were not reinforced according to the schematics, and suffered a massive failure after only 1100 hours of operations. At this time, the scarcity of materials was even greater, and it was finally decommissioned a year later. (Sulzberger, 2009)

Since the electrification in rural areas was very modest in the early 20th century, a market for off-grid power production opened up, which small sized wind turbines could fill. But as electrification initiatives expanded the grid to more remote locations, combustion engine technology developed and the price of oil fell, the market for wind power shrank. (Rhodes & Wheeler, 1996)

Despite the technological advancements in the field and megawatt class turbines, it was not until the 1970's that the technology started its path to what it is today. Like many other technologies that has become commercial despite bad market conditions, the development was driven by enthusiasts willing to overlook the economic disadvantage for a greater goal. It was the anti-nuclear protests in Denmark that triggered the development and production of small turbines in the 22 kW-class. The owners organised themselves into co-operatives and associations, who could use their combined resources to lobby for wind power. This movement pushed the wind turbine market for which later incrementally grew to the multi-megawatt turbines that later has become industry standard. (Meyer, 1995)

2.1 Wind turbine theory

Wind turbine technology has had seen radical development in the recent years, as power output has increased greatly, and a number of different disciplines have been involved in the development. Materials technology has enabled the turbines to reach higher, rotor diameters to grow larger and blades get lighter (Thirumalai, 2014). The development in power electronics has greatly improved the power conversion, enhanced turbine controllability, improve turbine efficiency and improve the connection between the turbine and the main electrical grid (Murthy, 2013). The improvements in computational power and the development of computational fluid dynamics has enabled engineers to design more efficient (Dosaev, et al., 2014) and durable blade types (Yu & Kwon, 2014) as well as optimising planning and control of wind farms (Choi, et al., 2014). Improved computerised automatic maintenance systems has, and have great potential to improve the reliability, production and economics for wind power (Besnard, et al., 2010).

2.1.1 The power of the wind

The energy in the wind comes from the constant movement of air, caused by temperature and pressure differences in the atmosphere, driven by the radiation from the sun. As the wind moves across the terrain, the friction caused by the structure of the terrain (known as the *roughness*), generates turbulence which influence the wind speed, as depicted in Figure 2.1. This gradient in speed is known as *wind shear*, where the resultant speed of the wind at the ground is zero. (Wizelius, 2007)

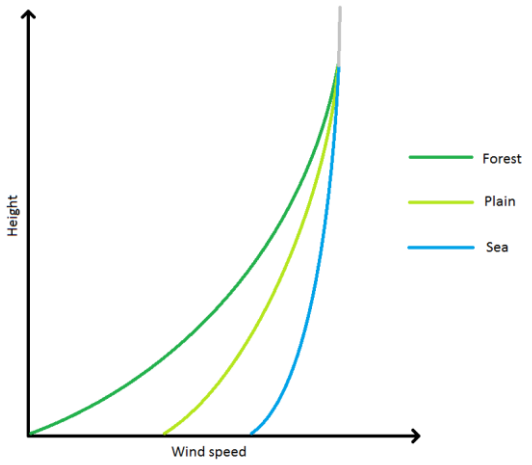


Figure 2.1 The graph shows the roughness of the terrain's impact on wind speed. Based on a graph by (Wizelius, 2007).

In Figure 2.2 below, terminology used when describing the characteristics of a wind turbine is explained.

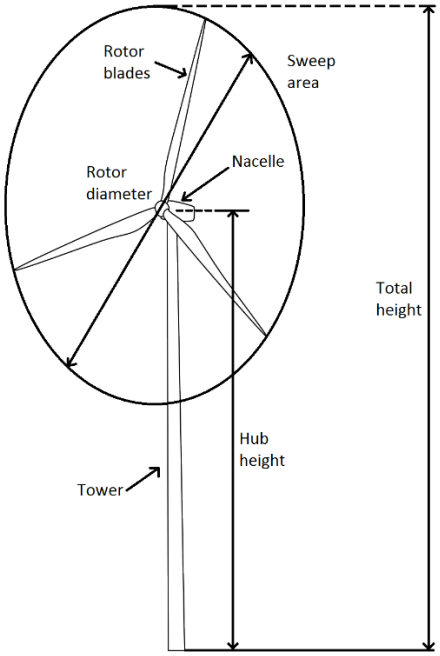


Figure 2.2 Basic wind turbine terminology

If the wind speed v_{w0} is known at a certain height h_0, Eq 2.1 can be used to approximate the speed v_w at height h, where α is a parameter depending on the roughness of the terrain.

$$\frac{v_w}{v_{w0}} = \left(\frac{h}{h_0}\right)^{\alpha} \qquad \textit{Eq 2.1}$$

The kinetic force of the wind P_w is heavily dependent on the wind speed (v_w) as can be seen in Eq 2.2, but also the density of air (ρ) as well as the cross-section A plays an important part (Eq 2.3).

$$P_w = \frac{1}{2}\dot{m}v_w^2 = \frac{1}{2}\rho A v_w^3 \qquad \textit{Eq 2.2}$$

$$\dot{m} = \rho A v_w \qquad \textit{Eq 2.3}$$

The mass flow of air $\dot{m}$, depends on the density of air, which varies with atmospheric pressure, humidity and temperature. The density can be calculated using the ideal gas law (Eq 2.4).

$$\rho = \frac{p}{R_{specific} * T} \qquad \textit{Eq 2.4}$$

Where p is the absolute pressure, $R_{specific}$ is the specific gas constant for dry air and T is the temperature.

Statistically the wind speed measured over time follows a Weibull distribution, but because of the cubic relationship to wind speed, the power density will not follow the wind speed density. Looking at Figure 2.3, we can see that even though there will be a

lot less hours with wind speeds above 9 m/s, half of the energy is concentrated in that interval.

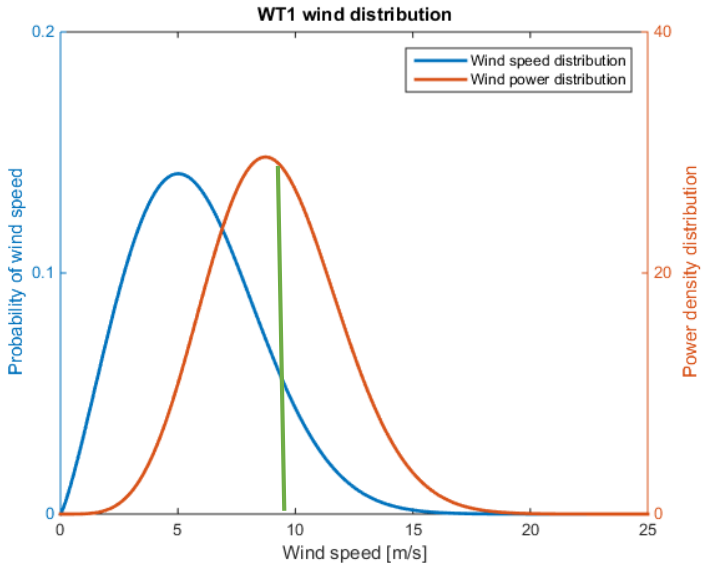


Figure 2.3 Plot showing the wind speed distribution (blue line) and the kinetic power distribution (orange). The green line shows the population mean of the energy distribution.

The Weibull shape can be seen in the figure above and is described by Eq 2.5,

$$f(v) = \frac{k}{\lambda}\left(\frac{v}{\lambda}\right)^{k-1} e^{-\left(v/\lambda\right)^{k}} \qquad \textit{Eq 2.5}$$

Where the shape of the Weibull distribution $f(v)$ is described by the parameters λ and k.

2.1.2 From kinetic to electric power

To convert the energy in the wind into electric power, aerodynamic principles are first applied to convert the energy in the wind flow into rotational energy in the rotor. The rotational energy is then converted into electricity in a generator. There are a number of different technologies used in the wind turbine industry, and in this section the technology used in the turbines in the study will be explained.

The turbines are all of the same model, with some small deviations in measures as can be seen in detail in Section 2.4. The model is a pitch-controlled variable-speed turbine, using a doubly-fed induction generator (DFIG) and a three-stage gearbox.

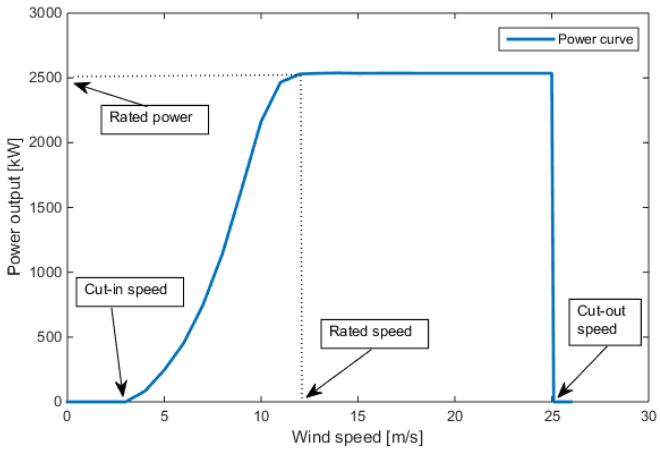


Figure 2.4 Terminology of the pitch controlled turbine specifications explained. The blue curve is a replication of the power curve for the model used in this study.

In Figure 2.4 common terms used when describing the specifications of a wind turbine are explained. The cut in speed is the wind speed where the turbine starts to produce electricity, the rated speed is where the rated power output is reached and the cut-out speed is where the turbine stops to avoid damages to the machinery. The blue curve is called the power curve, which represents the normal power output of the turbine at a given wind speed. This curve is constructed using the standard set by IEC 61400-12-1, but will in fact vary depending on location and the individual turbine (Wan, et al., 2010).

At low to medium wind speeds, the pitch control will vary the angle of the blades to maximise the power output from the turbine. The pitch control is also used to control the output of the turbine when the turbine has reached rated speed, the pitch blade angle of the turbine is varied to lower the coefficient of power at these speeds.

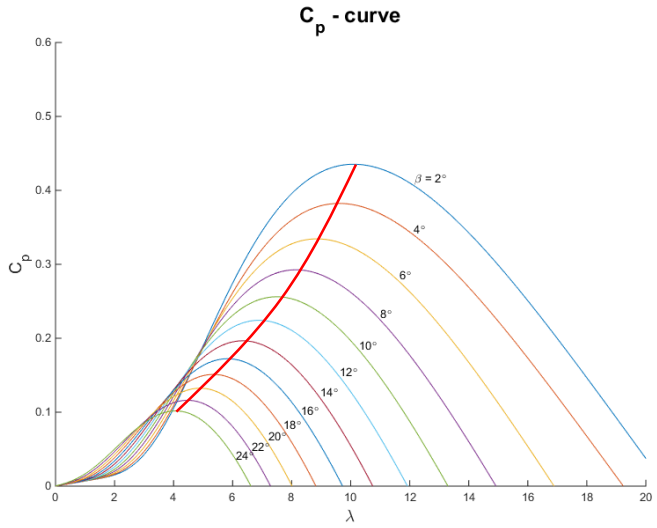


Figure 2.5 Cp dependence on beta and lambda. The plot is created using values and equation from (Belghazi & Cherkaoui, 2012). The red line follows the locus of maxima's and shows the optimal λ for the different pitch angles.

The mechanical power transferred to the turbine from the wind is given by Eq 2.6, where C_p is a function of λ and β, as can be seen in Figure 2.5.

$$P_{mec} = P_w * C_P = \frac{1}{2}\rho A v_w^3 C_P \qquad \textit{Eq 2.6}$$

$$C_P = f(\lambda, \beta) \qquad \textit{Eq 2.7}$$

$$\lambda = \frac{v_{tip}}{v_w} \qquad \textit{Eq 2.8}$$

$$v_{tip} = r\omega \qquad \textit{Eq 2.9}$$

The β in Eq 2.7 is the blade pitch angle. λ is the tip-speed ratio, which depends on the speed of the tip (v_{tip}) in relation to the wind speed. The tip speed is given by Eq 2.9, where r is the radius of the swept area and ω is the rotational speed given in radians per second.

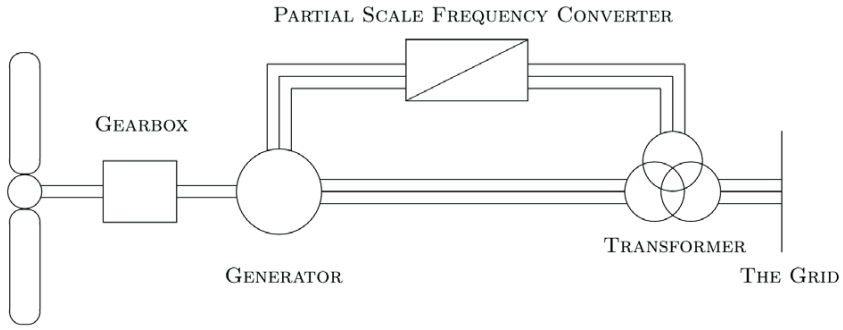


Figure 2.6 Principle of the doubly-fed asynchronous generator.

Figure 2.6 above shows the working principle of the Doubly-Fed asynchronous Induction Generator (DFIG). The frequency converter allows the generator to output a constant voltage and frequency independent of the generator rotor speed. By adjusting the frequency and amplitude of the AC currents from the converter being sent through the windings in the rotor, the voltage frequency and amplitude is controlled to be constant in the generator stator.

There are many advantages to using a DFIG compared to a classic synchronous generator, e.g. less mechanical stress on moving parts, being able to adjust reactive power and generating power at lower wind speeds. (Festo Didactic, 2011)

2.1.3 Wind speed standard deviation

One of the hypotheses in this thesis is that a lot of information is overseen when not taking the standard deviation value into account. In an effort to visualise this importance, a number of plots are presented and ways to take advantage of the behaviours are discussed.

The concept of standard deviation is explained in Section 4.1. Standard deviation represents the intensity of the change in the parameters. In the case of wind speed, it can be seen as an indication of the turbulence during the five minutes over which it is calculated. The turbulence intensity can be calculated using Eq 2.10 (Honrubia, et al., 2012).

$$I_v = \frac{\sigma}{\overline{v}_{mean}} \qquad \textit{Eq 2.10}$$

Where I_v is the turbulence intensity, σ is the standard deviation and $\overline{v}_{mean}$ is the mean average speed.

In Figure 2.7 below, a distinct connection between the standard deviation of wind speed and power output can be seen. At lower wind speeds, the behaviour of the two follow each other closely. As the wind speed increases the pitch angle of the blades are changed, curtailing the power output to constant levels, decreasing the standard deviation of power output as it becomes constant at rated speed.

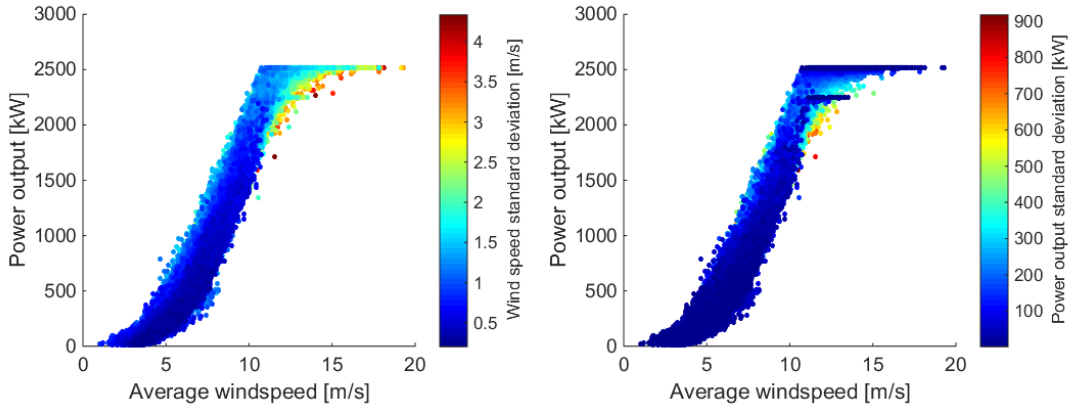


Figure 2.7 The relationship between standard deviation of the wind and the power output is shown in these two figures.

To the left in Figure 2.7 one can also conclude that the standard deviation increases at high wind speeds, which is natural due to more intensive turbulence at these speeds.

The relationship between a high standard deviation and the difference between minimum and maximum wind speed can also be seen in Figure 2.8.

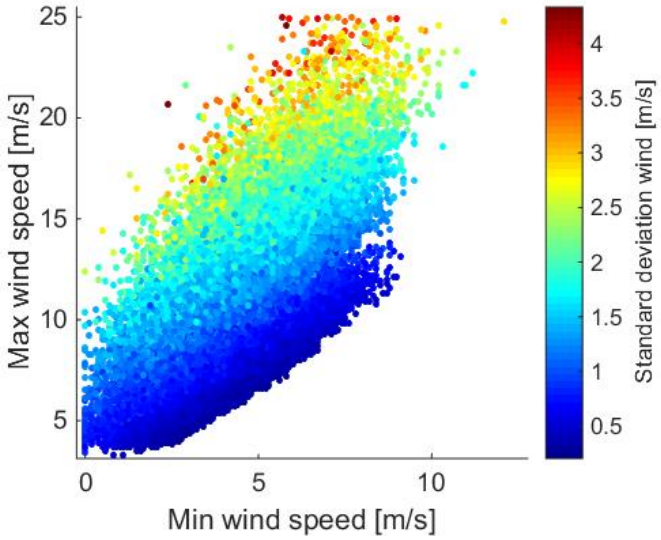


Figure 2.8 The relationship between min, max and standard deviation of wind is shown in this figure. When there is small difference between min and max speed, the standard deviation is low.

In Figure 2.9 below, four examples of different wind situations, with equal min, max and average values, are discussed in order to problematize the usage of only average values. The actual wind profile looks completely different, but the graphs are meant to show a general behaviour.

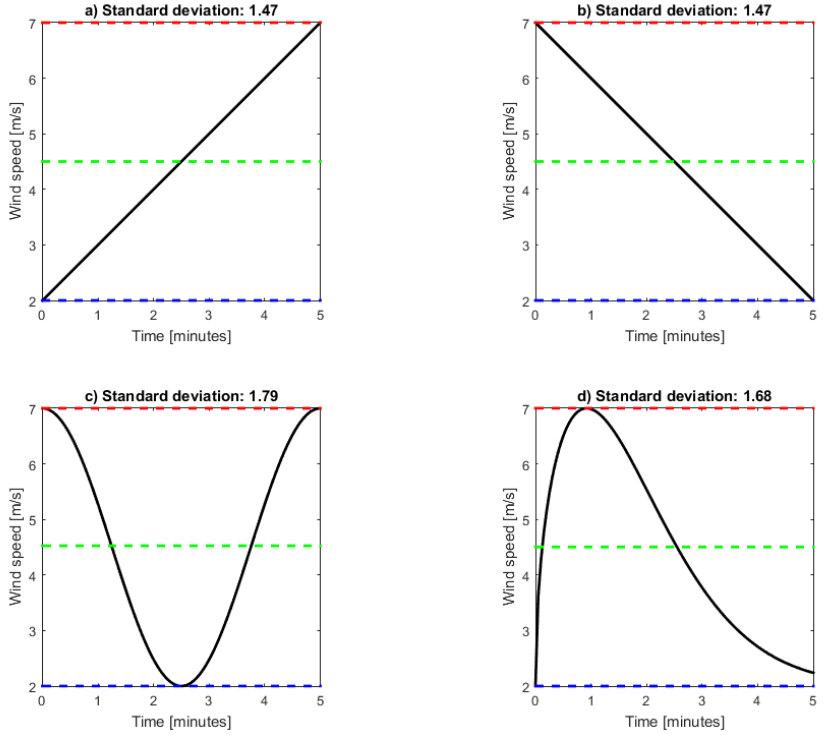


Figure 2.9 The graphs in this figure is meant to visualise some possible general trends during one sampling of the wind speed. The same min, max and mean values are assumed for all four cases.

Graph a) in Figure 2.9 shows a case where the wind speed is gradually increasing, causing the turbine to accelerate. This will cause some of the energy to be stored in the rotor as inertia, and will lead to less generation of electricity until the acceleration has stopped. By the same logic, inertia stored in the rotor will generate electricity while decelerating in the case shown in graph b), thus producing more than should be possible at the current average speed. Graph c) shows a case where the total inertia change would be zero, since the acceleration and retardation profiles are inverted. In graph d) a case where the acceleration curve is much steeper than the deceleration curve is shown. If the acceleration is too fast, the rotor will not have reached its rated speed at the inflexion point, and will have produced less electricity seen to the average speed over the sampling time.

2.2 Planning and maintaining a wind turbine

Before erecting a wind turbine, wind speed measurements are done over a time, normally two years. If the site is deemed feasible, model specifics are determined from the wind statistics. Eq 2.1 can be used to determine a good height, but normally there are restrictions on the total height, so depending on wind speed at the site, the blade length and hub height are adjusted to maximise production for the specific site.

Using the reasoning in Figure 2.3 the total electricity produced can be determined. Furthermore, using an estimated long time price for electricity, the investment can be deemed feasible or not.

The environmental impact of a wind turbine is considered low compared to conventional energy sources, since it does not emit any pollutants during its operations and the impact is concentrated to the manufacturing and construction phase in the lifetime of the turbine. A wind turbine placed in a location with good conditions will normally have produced as much energy as was needed for the construction in seven months' time (Guezuraga, et al., 2012).

The maintenance is typically performed by a contracting firm appointed by the supplier as a part of the warranty period of the turbine. After the warranty period, the responsibility lies on the owner of the wind turbine, who normally hires a contracting firm unless they own enough turbines for it to be feasible to have an in-house maintenance group. (Hockley, 2013)

Maintenance is normally performed according to a schedule according to the service agreement, and the possibility for the owner to monitor their turbine is limited to the SCADA system. There are a number of new studies made to better take advantage of all the data that is stored in the SCADA system. To get an understanding of the possibilities of these new methods, an overview of a selection of studies are presented below.

(Jenab, et al., 2013) Developed a Condition-Based Maintenance (CBM) system based on Artificial Neural Networks, where they by analysing a number of features can determine the type of maintenance and time needed for repairs. By doing this, reducing the downtime and overall cost for maintenance.

(Hameed, et al., 2009) Explore different parameters and techniques for Condition Monitoring Systems (CMS) e.g. vibrational and oil analysis. A large variety of different faults are looked at, with the main focus on frequency analysis and suggestions of new

sensors. Neural networks are examined to some extent to characterize frequencies in vibrational patterns.

(Bangalore, 2014) Developed a maintenance management framework called Self Evolving Maintenance Scheduler (SEMS). The author shows that by using ANN's to monitor the health of the gearbox, it is possible to detect damages at an early stage and avoid complete failures.

(Tchakoua, et al., 2014) Presents an extensive review of trends and future research in wind power CMS. The authors suggests that much more sensors are needed to make these monitoring systems semi-automatic and minimise human control.

(Schlechtingen & Ferreira Santos, 2010) Talk about the fact that set temperature levels for example bearings only will show when the turbine is producing close to maximum output. Using a more advanced model means that deviations can be seen even when the turbine is producing less.

2.3 Data collection

Statistics from a large number of sensors (SCADA data) are normally stored in a site main computer, this can be reached via VPN from outside the site and data can be collected (Figure 2.10).

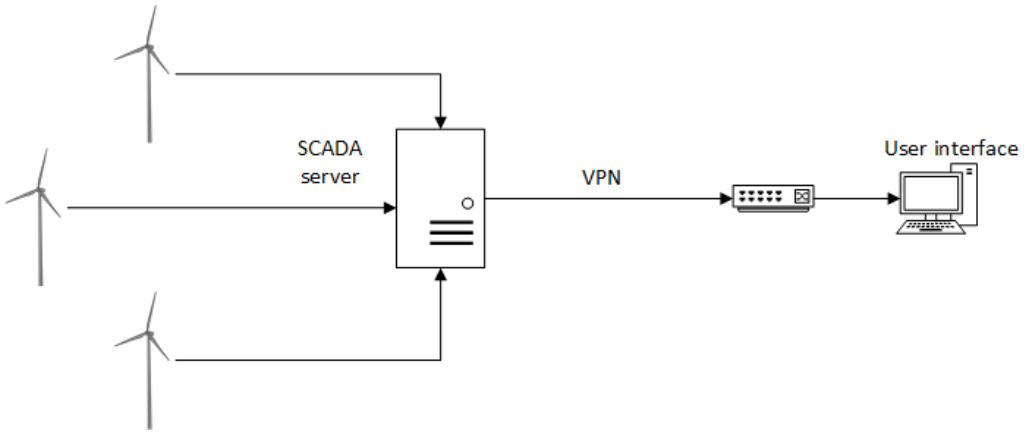


Figure 2.10 A simple schematic of the SCADA network.

The SCADA system itself is somewhat standardised, but the storage of the data differs between manufacturers. Simple statistics is normally rather easy to access, long time data is stored in large databases that might be unavailable from the outside.

2.4 Specifications of the turbines in the study

All the turbines in this study is of the same make and model, but the combination of height and diameter vary depending on the site specific needs.

Table 2.1 Specifications of the turbine types. These numbers are identical for all turbines in this study

Technical information	
Blades	3
Cut-in wind	3.5-4 m/s
Cut-out wind	25 m/s
Rated speed	9.4 - 17.1 rpm
Gearbox	3-stage combination spur/planet gear
Generator	Doubly-fed asynchronous with slip-ring rotors
Converter	Static inverter with intermediate DC circuit
Power	2500 kW at 12 m/s
Speed regulation	Electronic pitch system

Table 2.2 Specific details about each turbine in the study. The specifications differ depending on the site where the turbine is placed.

Turbine	Diameter	Sweep area	Hub height
WT1	100	7854	100
WT2	100	7854	85
WT3	100	7854	85
WT4	104	8595	98
WT5	104	8595	98

3 Artificial Neural Networks

The development of artificial neural networks was inspired by the studies of the human's central nervous system, where the nodes and the interaction within themselves are to mimic the brains neurons and their synaptic connections. By introducing a *training data set* to the network, the *synaptic weights* are iteratively strengthened until the response of the network follows the output data, similar to the learning process in the biological brain. The definitions and theories in this chapter comes from (Haykin, 2009).

3.1 Building blocks of the ANN

The fundamental part of an ANN is the neuron, which acts on the input signal (x_j) and sends it altered forward. The neurons are interconnected via synapses which holds the weights (w_{jk}), deciding the strength of the connection to the next step, being the summing junction where the weighted input signals of the neuron are added. The activation function (Φ) limits the output of the neuron to the set finite range, normally [0,1] or [-1,1], and returns an output (y_k).

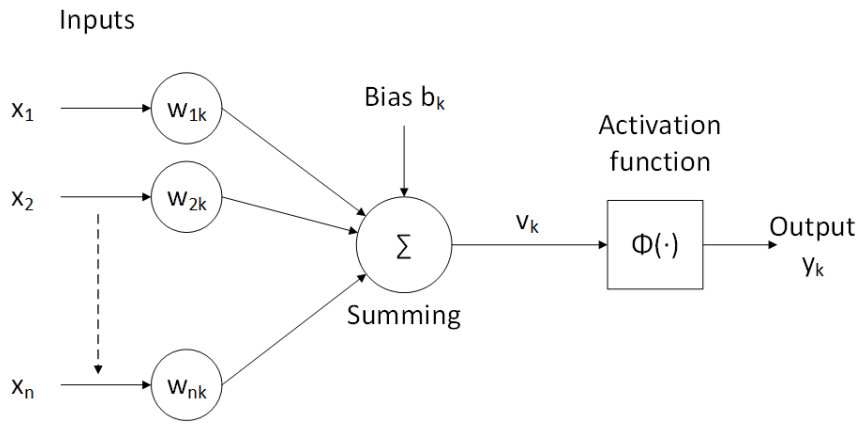


Figure 3.1 The neuron model explained.

The mathematical expression can be denoted as:

$$v_k = \sum_{j=1}^{n} w_{jk} x_j \qquad \textit{Eq 3.1}$$

$$y_k = \Phi(u_k + b_k) \qquad \textit{Eq 3.2}$$

Where b_k is the bias and u_k is the output from the summing function.

The activation function most commonly used is the sigmoid function, which is characterised by the shaping parameter a in Eq 3.3:

$$\Phi(v) = \frac{1}{1 + \exp(-a * v)} \qquad \textit{Eq 3.3}$$

The shaping parameter is used to vary the response of the activation function and deciding the neuron output. When the shaping parameter goes towards infinity, the sigmoid function acts like a Heaviside function. The sigmoid function with a varying shaping parameter is shown in Figure 3.2 below.

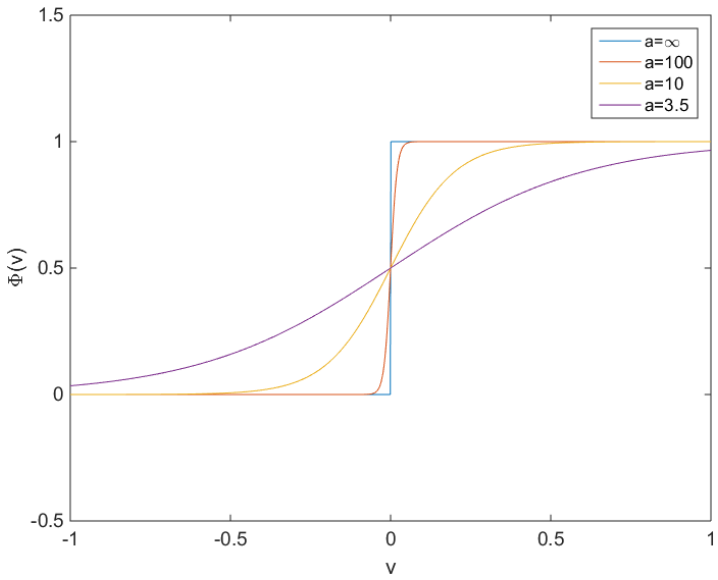


Figure 3.2 The sigmoid function plotted with a varying shaping parameter.

3.2 Structure of a Network

An ANN is characterised by its structure and the two different architectures used in this study will be explained in this section.

Feed-forward network

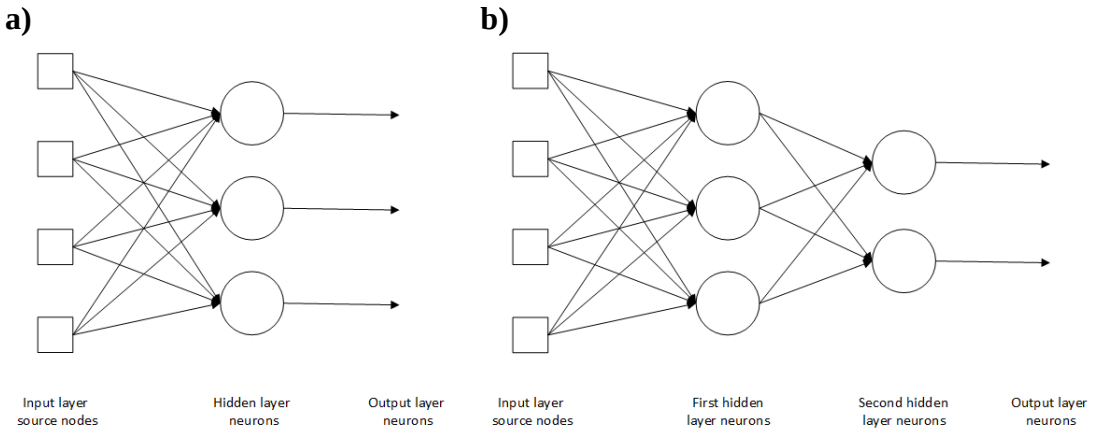


Figure 3.3 The structure of the single- (a) and dual- (b) hidden layer feed-forward network.

The feed forward network is built up of an input layer, one or more hidden layers depending on the complexity of the problem, and an output layer. Generally one hidden layer is sufficient, since the extra layers might induce over-fitting where the network loses its ability to generalise.

Multilayer recurrent network

In the recurrent network, the outputs are fed back as inputs with a delay operator, which will improve the results especially for time series problems.

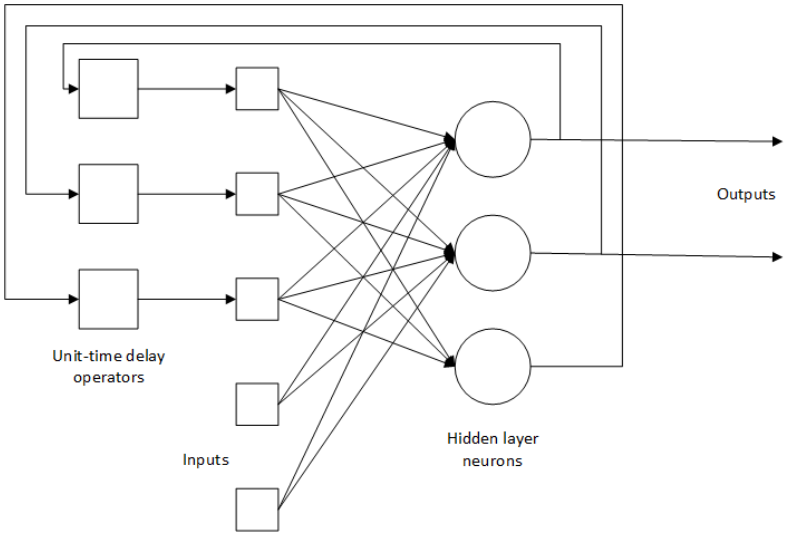


Figure 3.4 The structure of the recurrent network.

3.3 Learning algorithms

The learning algorithm used in this thesis is the Levenberg-Marquardt algorithm, because of its superior speed and precision. The speed of the algorithm comes from its ability to take advantage of the fast second-order training speed, using an approximation of the Hessian matrix (***H***). The elements in the Jacobian (***J***) is the derivatives of the network errors related to the weights and biases of the network. The Hessian is approximated using the Jacobian (Eq 3.4), which takes much less computational power than that of computing the Hessian (Demuth & Beale, 2002).

$$\boldsymbol{H} = \boldsymbol{J}^T\boldsymbol{J} \qquad \textit{Eq 3.4}$$

$$\boldsymbol{g} = \boldsymbol{J}^T\boldsymbol{e} \qquad \textit{Eq 3.5}$$

The gradient $\boldsymbol{g}$ can then be calculated with Eq 3.5, where $\boldsymbol{e}$ is the vector of network errors.

$$\boldsymbol{J} = \frac{d\boldsymbol{f}}{d\boldsymbol{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_m}{\partial x_1} & \cdots & \frac{\partial f_m}{\partial x_n} \end{bmatrix} \qquad \textit{Eq 3.6}$$

$$\boldsymbol{H} = \begin{bmatrix} \frac{\partial^2 f}{\partial x_1^2} & \frac{\partial^2 f}{\partial x_1 \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_1 \partial x_n} \\ \frac{\partial^2 f}{\partial x_2 \partial x_1} & \frac{\partial^2 f}{\partial x_2^2} & \cdots & \frac{\partial^2 f}{\partial x_2 \partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial^2 f}{\partial x_n \partial x_1} & \frac{\partial^2 f}{\partial x_n \partial x_2} & \cdots & \frac{\partial^2 f}{\partial x_n^2} \end{bmatrix} \qquad \textit{Eq 3.7}$$

Comparing the definition of the Hessian and the Jacobian, it is obvious that the computational time needed for the Jacobian is much lower than that of the Hessian.

The algorithm incorporates the fast-converging behaviour of Newton's method and the step-size parameter selection of the Gradient descent method. By adjusting the regularising parameter (λ) each iteration, the calculation can be shifted towards either Newton or Gradient descent, by assigning a small or large value, respectively (Haykin, 2009). This greatly speeds up the detection of minima's, since the algorithm first "sees" the general gradient to find a region where the global minima is located, then change λ to find the local minima, which has potential to be the absolute minima.

$$\Delta \boldsymbol{w} = [\boldsymbol{H} + \lambda \boldsymbol{I}]^{-1} \boldsymbol{g} \qquad \textit{Eq 3.8}$$

The optimal adjustment to the synaptic weight ($\Delta \boldsymbol{w}$) per iteration is stated in Eq 3.8.

3.4 Constructing the ANN model

There is no established method to follow when it comes to deciding the best number of neurons to use in an ANN, the number of hidden neurons and layers depends on the number of inputs, outputs, sample points, complexity of data. The most common way to decide the configuration is by running a series of tests repeatedly, where the number of neurons is varied until the best configuration is found.

At the start of each training the synaptic weights are given a randomly set starting value, which means that unless the starting values are saved, the chance that the exact network is repeated twice is extremely small. There is also a risk that the starting values are far from the minima and the learning algorithm gets stuck in a local minima, the result being a network with poor performance. It is therefore important to train the network iteratively with the established configuration to make sure that the network is not the product of a poor learning cycle.

The training set used for the learning process is divided into a testing, validation and training set. During the learning, the training set is used to train the model, which is then tested against the test set to evaluate the performance of the network. One of the best characteristics of the ANN is its ability to generalise a non-linear behaviour, and to test the generalisation of the network, the model is then tested against a validation

set which has not been used in the training. The combined performance is evaluated each epoch until the training reaches set criterions for ending the iteration.

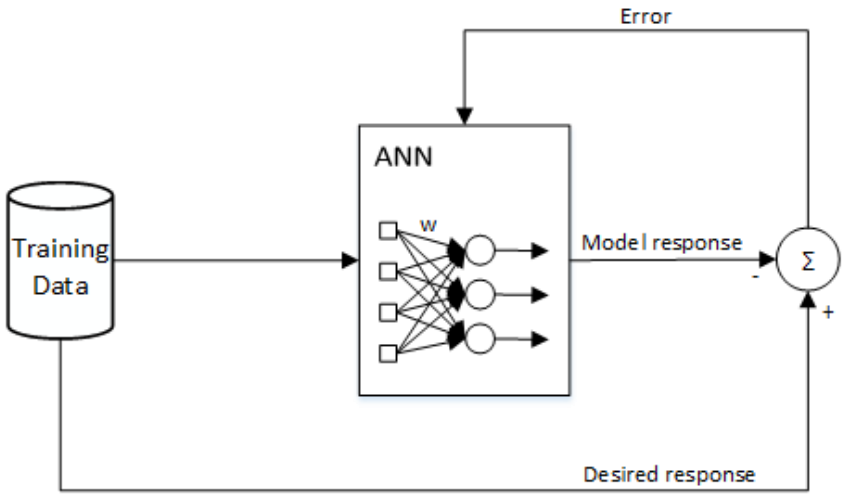


Figure 3.5 An overview of the training process of an ANN

The algorithm used to train the network is visualised in Figure 3.5, where training data is fed into the model who gives its response and compares it to the actual value supplied by the training data. This kind of learning is called supervised learning, and it is the method that will be used in this thesis.

When training the ANN, noisy data can easily be overlooked but faulty data is very harmful for the training, as abnormal trends can be seen as a behaviour which the network should mimic. It is therefore very important to filter the training set from erroneous data prior to the training.

3.5 Applications

ANN is a powerful tool that has a wide variety of uses; from hand writing-, facial- and image-recognition, to basically any signal analysis. It is the network's ability to recognise patterns and generalise where needed that makes it so diverse. But there are some downsides associated with the method, e.g. if the network is taught with faulty data the performance will be bad. The network demands a certain amount of data to create a good model and it is sensitive to changed ratios in the inputs, for example a sensor change. The ANN is also what is called a black box model, where the outputs cannot be back tracked, meaning that it is not possible to calculate backwards to find the faulty input.

ANN theory has been applied to wind turbines in research to a rather large extent, mostly in the CMS field. During the literature study, some papers with interesting findings oriented towards the focus of this thesis have been found, which can be implemented into the methodology.

(Lydia, et al., 2013) Tests a number of different models to try to improve the power curve construction of a wind turbine. The group found that the parametric model performed best, ANN was the best non-parametric model.

(Paiva, et al., 2014) Constructed a more precise power curve by observing a wind turbine for a year, 10%-15% RMSE uncertainty came from power curve. Nacelle anemometer showed more stable measurements than a mast-mounted anemometer. Using Brent's method, creating nacelle power curve by using a scaling parameter.

(Schlechtingen, et al., 2013a) Tested a number of models on three pitch-controlled wind turbines to detect abnormal power output, earlier research had shown that k-NN models showed the best performance compared to least square and maximum likelihood parametric models. The group found in literature that Cluster Center Fuzzy Logic (CCFL), ANN and k-nearest neighbour (k-NN) showed the best results for modelling wind turbine power curves, in that order. These models, together with adaptive neuro-fuzzy interference system (ANFIS), were evaluated and compared. It was found that ANFIS models needed less parameters than ANN, meaning faster training. Also, ANN systems are black box models that makes it hard to use prior knowledge of the system. To pre-process the data, the group followed the methodology proposed in their earlier work (Schlechtingen & Ferreira Santos, 2010). When deciding the number of hidden neurons, the authors suggests running the models at least 10 times, changing only the number of neurons, then selecting the network configuration which gives the best generalization. The group concludes that when only using wind speed as input parameters, all the methods show comparable results. When including wind direction and ambient temperature, it was found that CCFL, ANN and ANFIS can detect anomalies in the wind turbines at a much earlier stage than conventional models.

(Schlechtingen, et al., 2013b) In this article, the group has studied data from 18 turbines collected over 30 months. Using 33 different SCADA signals, the group develop 45 normal behaviour models using the ANFIS model as described in their earlier work. ANFIS is compared to ANN to show the advantage in computational time. The normal models are evaluated by studying the prediction error, and it is found that the error should be normalised over a day due to large fluctuations in 10-min errors.

(Lapira, et al., 2012) Tested Gaussian mixture model (GMM) with an L2 distance metric, Self-organizing map (SOM) with Minimum quantization error (MQE) and Neural Network with analysis of residue. The group excludes points where production is less than zero, where wind speed is under rated cut in speed and times where pitch curtailment is active, since this data are not considered representative for normal operations.

4 Mathematical tools

There are a number of mathematical concepts used in this thesis that will be explained in this chapter. An introduction to the concept of distributions will be made and the characteristics of a few of the most common ones will be described. The concepts of the Mahalanobis distance used extensively in the method will be explained, as well as a short introduction to clustering and a few methods of error estimation.

4.1 Statistical instruments

To interpret the results from the model, one needs to be able to discern statistical noise from legitimate deviations from the normal behaviour. By characterising a data set according to a fitting distribution, the normal range of the dataset can be set. A threshold can then be formulated for which points are to be considered a part of the distribution, and which to be considered outliers. The theory and equations in this Section is gathered from (Rice, 2007).

In order to characterise the probability distribution of a data set, there are some different methods, but the most commonly used is by constructing the probability density function (PDF) and comparing the characteristic to known distributions. A PDF is constructed by sorting the density of probabilities of each outcome of the sample, an example showing the PDF of a sample and the best fit is shown in Figure 4.1 below.

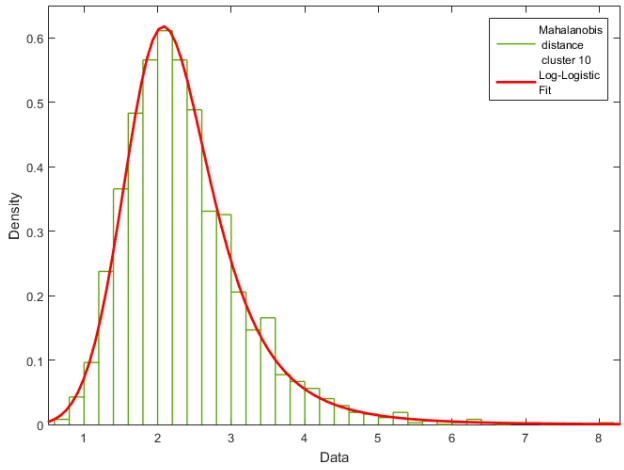


Figure 4.1 The PDF of a sample and the best fitting function.

The PDF of most distributions are described by an equation containing a shape, scale and in some distributions a location parameter. In Eq 4.1 through Eq 4.4, the distributions found in this study are described.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \qquad \textit{Eq 4.1}$$

The PDF of the Normal distribution is shown in Eq 4.1, with the location parameter (also the mean) μ and variance σ^2.

$$f(x) = \begin{cases} \frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1} e^{-\left(\frac{x}{\lambda}\right)^k}, & x \geq 0 \\ 0, & x < 0 \end{cases} \qquad \textit{Eq 4.2}$$

Eq 4.2 shows the PDF of the Weibull distribution, with the scaling parameter λ and shape parameter k.

$$f(x) = \frac{\left(\frac{\beta}{\alpha}\right)\left(\frac{x}{\alpha}\right)^{\beta-1}}{\left(1+\left(\frac{x}{\alpha}\right)^{\beta}\right)^2}, x > 0 \qquad \textit{Eq 4.3}$$

The PDF of the Log-logistic distribution is given by Eq 4.3, with scaling parameter α and shape parameter β.

$$\begin{cases} f(x) = \frac{1}{\sigma}\left[1+\xi\left(\frac{x-\mu}{a}\right)\right]^{\left(-1/\xi\right)-1} \exp\left\{-\left[1+\xi\left(\frac{x-\mu}{\sigma}\right)\right]^{-1/\xi}\right\}, \xi \neq 0 \\ f(x) = \frac{1}{\sigma}\exp\left[-\left(\frac{x-\mu}{\sigma}\right)\right]\exp\left\{-\exp\left[\left(-\frac{x-\mu}{\sigma}\right)\right]\right\}, \xi = 0 \end{cases} \qquad \textit{Eq 4.4}$$

In Eq 4.4, the PDF of the Generalized extreme value distribution is stated, with the location parameter μ, scaling parameter σ and shape parameter ξ.

Standard deviation

The standard deviation σ is an indication of the dispersion of the probability distribution around its centre, and is defined in Eq 4.5.

$$\sigma = \sqrt{\sum_{i=1}^{n} p_i(x_i)(x_i-\mu)^2} \qquad \textit{Eq 4.5}$$

Where μ, or $E[X]$, is the expected value for the discrete random variable X with frequency function p_i and values x_i (Eq 4.6).

$$\mu = E[X] = \sum_{i=1}^{n} p_i(x_i)x_i \qquad \textit{Eq 4.6}$$

The variance, Var is often simply denoted as σ^2, and is defined in Eq 4.7 and Eq 4.8 as the average value of the deviation of X from its mean.

$$Var(X) = E\{[X-E(X)]^2\} \qquad \textit{Eq 4.7}$$

$$Var(X) = \sigma^2 \qquad \textit{Eq 4.8}$$

4.2 Mahalanobis distance

The Mahalanobis distance (MD) is defined as the distance, measured in standard deviations from the mean, from a point to a distribution. The method takes the correlation in the data into account, since it contains the inverse of the covariance matrix of the set, and is widely used for outlier detection and determination of representativity between data sets. (De Maesschalck, et al., 2000)

$$d_{ij} = \sqrt{(\boldsymbol{x}_i - \boldsymbol{\mu}_i)^T \boldsymbol{C}^{-1} (\boldsymbol{x}_j - \boldsymbol{\mu}_j)} \qquad \textit{Eq 4.9}$$

Eq 4.9 shows the Mahalanobis distance between the stochastic vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, given as $\boldsymbol{x}_i = (x_{i1}, x_{i2}, x_{i3}, \dots, x_{in})^T$, where n is the number of variables in observation i. $\boldsymbol{\mu}_i$ and $\boldsymbol{\mu}_j$ are the mean values of the vectors given as $\boldsymbol{\mu}_i = (\mu_{i1}, \mu_{i2}, \mu_{i3}, \dots, \mu_{in})^T$. $\boldsymbol{C}^{-1}$ is the inverse covariance matrix, where the covariance is defined in Eq 4.10.

$$\boldsymbol{C} = E[(\boldsymbol{x}_i - \boldsymbol{\mu}_i)(\boldsymbol{x}_i - \boldsymbol{\mu}_i)^T] = E[(\boldsymbol{x}_j - \boldsymbol{\mu}_j)(\boldsymbol{x}_j - \boldsymbol{\mu}_j)^T] \qquad \textit{Eq 4.10}$$

An advantage of the Mahalanobis distance compared to other distance measures is that the distance is calculated from the distribution, which means it is not from a central point, but rather a line, if seen in two dimensions. Figure 4.2 and Figure 4.3 below shows two randomised sets from different distributions, with the Mahalanobis distance of each point shown by a colour gradient.

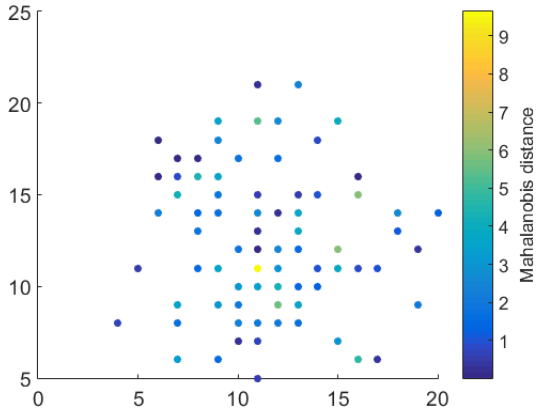


Figure 4.2 A randomised data set based on the Poisson distribution. The Mahalanobis distance shown by colouring.

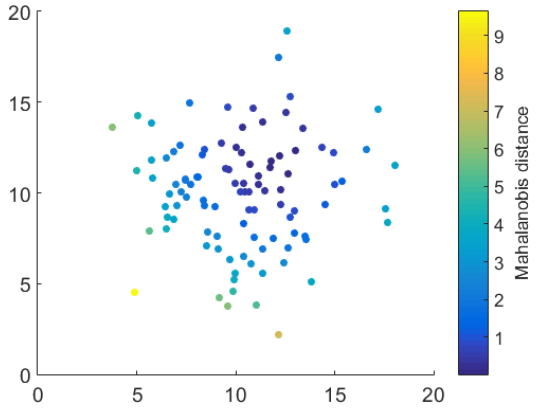


Figure 4.3 A randomised data set based on Normal Distribution. The Mahalanobis distance is shown by colouring.

In Figure 4.2, the Mahalanobis distance was calculated using the Normal distributed data seen in Figure 4.3 as reference. Both sets of data are in the same range, and as can be seen in the figure showing the Normal distribution, the Mahalanobis distance follows the relative distance from the centroid. But in the figure showing the Poisson distribution, the MD does not seem to follow any pattern, which comes from the previously argued case that the distance is decided in relation to a distribution.

4.3 Clustering

To cluster data means to divide data sets into sub sets with similar properties. There are a wide variety of clustering algorithms using different definitions of what is to be considered the linking property between the data, such as distance, distribution, density etc. The clustering method used in this thesis is Ward's method, which uses the minimum variance algorithm and calculates the inner squared distance between the clusters and divides the data according to a specified amount of clusters.

The clustering in this thesis has been done using MATLAB's built-in *clusterdata* function. The mathematical expressions behind the clustering algorithm are very complicated and the actual mechanics behind the clustering was not further investigated as this lies outside the scope of the thesis.

4.4 Error estimation

To evaluate the performance of the models and to be able to compare the results to other studies, four different error estimates with different properties are used.

The mean squared error is used as the performance function in the network training algorithm and is given by Eq 4.11. A disadvantage of this estimator is the inherent property to weight outliers heavily due to the squaring of each term, and might skew the result.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\widehat{\boldsymbol{Y}}_i - \boldsymbol{Y}_i\right)^2 \qquad \textit{Eq 4.11}$$

Where $\widehat{\boldsymbol{Y}}$ is the vector consisting of the predictions of the true values $\boldsymbol{Y}$.

The MSE is not used as a performance indicator in the comparative study, but the following estimators are used; Root mean squared error, which is a commonly used measure in prediction models. It is defined as the square root of the MSE.

$$RMSE = \sqrt{MSE} \qquad \textit{Eq 4.12}$$

The mean absolute percentage error (MAPE), returns the accuracy of the model as a percentage.

$$MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{p_i - a_i}{a_i}\right| \qquad \textit{Eq 4.13}$$

The absolute difference between the actual value a and predicted value p are divided by a and summed up for each point and divided by the total points n.

The mean absolute error (MAE) is calculated by summing the absolute error between the predicted value p and actual value a.

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|p_i - a_i| \qquad \textit{Eq 4.14}$$

5 Method

The purpose of the artificial neural network is to predict the normal response in power output as a function of certain chosen in parameters. In this chapter, the methodology used to reach the results of this thesis is explained.

5.1 Assumptions

A number of assumptions has been made where no information could be found, or where it was deemed that an approximation was good enough to describe a problem. Listed below is a summary of made assumptions.

Data

The standard deviation value reported in the SCADA system is assumed to be calculated as if the data were of a shifted Normal distribution. No information about the actual mathematical expression can be found, but this seems to be the most reasonable assumption. Seen during only a five minute interval, even though the wind is known to follow a Weibull distribution, this could also be true.

Filtering

The filtering technique used in this thesis is the product of a lot of reasoning around the influence of standard deviation and some influences from other studies, mainly from (Kusiak & Verma, 2013). The combination of parameters used for clustering have not been used in any other studies found in the literature study. Since the main focus of this thesis has been the modelling, any deeper mathematical analysis has not been made. The validity could be of interest for further investigation.

The number of clusters chosen was decided by studying plots, finding some general behaviour of the curves, and some trial-and-error. The optimal number could indeed be discussed, but the work needed for a rigid expression was not included in this study due to time limitations.

When deciding the threshold for the Mahalanobis distance considering the clustering, a Log Logistic distribution was found to be the best fit. Setting the limit at the 97.5 % percentile might exclude some valid data points, but it was deemed the best solution to a complicated statistical problem.

ANN

Because of the large data sets, repeated random sub-sampling validation is chosen as the method to find the best number of neurons in the feed forward networks. There are more precise methods available, such as k-fold cross validation, but the computational time needed for these methods were considered unreasonable for the relative small gain. In an effort to limit the computation time in the validation, the number of iterations is set to 30.

5.2 Data import

The data recorded by the SCADA system is stored in the site main server, as described in Section 2.3. Due to limitations in access, the full content of the database is not reachable from the outside.

There are a few different ways to access the data from the outside. The most basic access is by a simple web interface that interacts with the site SQL database, where the most common parameters can be extracted. There is also a possibility to via a software supplied by the manufacturer, reach the controller in the wind turbine via a VPN tunnel, where long time trends and daily statistics can be accessed.

The manufacturer's software also stores a backup of 5 minute data in XML files, one separate for each time step. Deeper analysis of the XML files, which has several hundred outputs, showed that not all needed parameters were recorded in these log files, but merely a selection of them.

It was found that the raw SCADA data stored in an SQL database on site, due to a size limitation, is exported as database dump files (.mdf format) when it has reached this limit. Since the size of these files are large, and the fact that the site computers use a wireless modem with a data limit, the dumps are copied manually by connecting an external hard drive. This led to that not all recent data could be accessed, since it had not yet been collected from site.

To access the data needed for the study, the available dump files were mounted in Microsoft SQL database 2014 Management Studio. The structure of the database is very complex and to find the interesting parameters for extraction, cross-referencing between multiple trees in the database is necessary. This is due to that the tree that stores the actual data does not contain plain text variable names, but merely an ID code for the sensor, which needs to be cross-referenced in two lines to two other trees containing first the ID of the parameter, then the name of the variable.

To add to the complexity, a certain parameter can have multiple ID's, and also the ID's differ between turbines even though it is the same model and make of all turbines in this study. In the database for WT1, which is a single turbine at the site, only data from that turbine is stored in the database. But in the remaining four turbines, data from two turbines is stored in the same database, which means a turbine ID had to be found by linking other trees in.

Data from the SQL database was then exported into a CSV format, which could easily be imported into MATLAB for further analysis. During this import, the data was at the same time cleaned from sets where one or more parameters were missing to avoid corrupt data.

5.3 Configuring the ANN

Creating the model is a process in steps that includes filtering of training data, iterative computing to find the statistically best network configuration and yet an iterative process to find the best possible model.

5.3.1 Input parameter selection

All input parameters to the Neural Network were manually selected using their stated relationship in Section 2.1. To get the appropriate response from the model, the selection of inputs must be well thought through. Using inputs that are too closely correlated to the output might result in a very good fit, but it could result in a model that will not be able to find the deviations that it is meant to find, since they will follow the behaviour of the output. The following parameters were chosen on the basis that they are not effected by, but have a direct effect on the output parameters.

Wind speed

The most important parameter when modelling the outputs is of course the wind speed, due to the cubic relationship between power and speed as can be seen in Eq 2.2. Some studies only use this parameter as input, but with relatively bad results as can be seen in the comparative study by (Schlechtingen, et al., 2013a), so additional inputs needs to be added to improve performance.

Outdoor temperature

The density of the air, and therefore the energy content, is highly dependent on the temperature of the air. Assuming constant pressure and relative humidity, varying the temperature from -15 °C to 25 °C (which can be considered a normal Swedish temperature interval), this can affect the energy content of the wind by close to 20 %, so it is important to include this parameter. In Figure 5.1 below, this effect is shown by plotting the power scatter with a coloured temperature gradient.

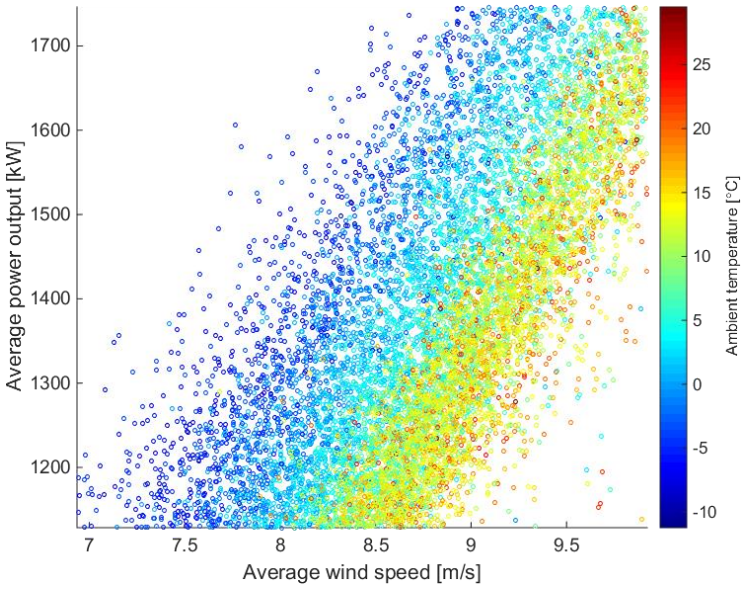


Figure 5.1 The impact of the ambient temperature is clear in this plot. At a constant average wind speed, the power output increases with a decreased temperature.

Nacelle direction

(Friis Pedersen, et al., 2002) Shows that wind direction can have substantial effect on the power curve in complex terrains, mostly due to the effect of turbulence upon the anemometer. The turbulence pattern is a product of the vegetation and obstacles surrounding the wind turbine. In Figure 5.2, the wind speed standard deviation over a year is plotted depending on average wind speed and nacelle direction. According to the figure, there is something causing intense turbulence in the NNE direction, as well as a slight increase in the S/SW direction. From this figure, one can also see the dominating wind direction, being WSW.

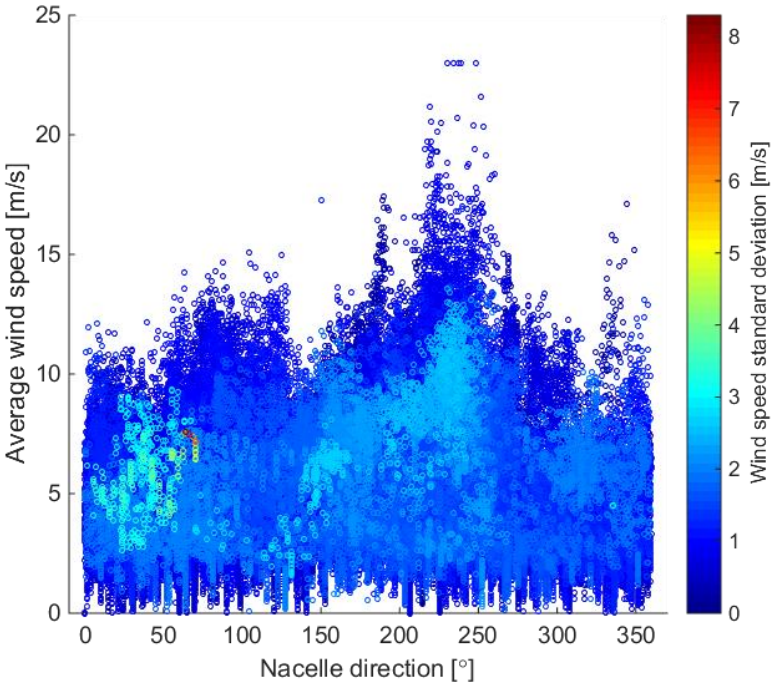


Figure 5.2 The standard deviation for WT1 is shown in different nacelle directions and wind speeds. It seems the standard deviation, and by default, turbulence intensity is strongest in a North-North-East direction, followed by a slightly lower pattern in the South-East to South-Western direction.

This is confirmed by studying the map presented in Figure 5.3, where another wind turbine can be seen in the NNE direction of WT1, causing the increased turbulence observed in Figure 5.2. Also, worth noting is that there is more vegetation to the south, explaining the increased turbulence in that direction as seen in Figure 5.2.

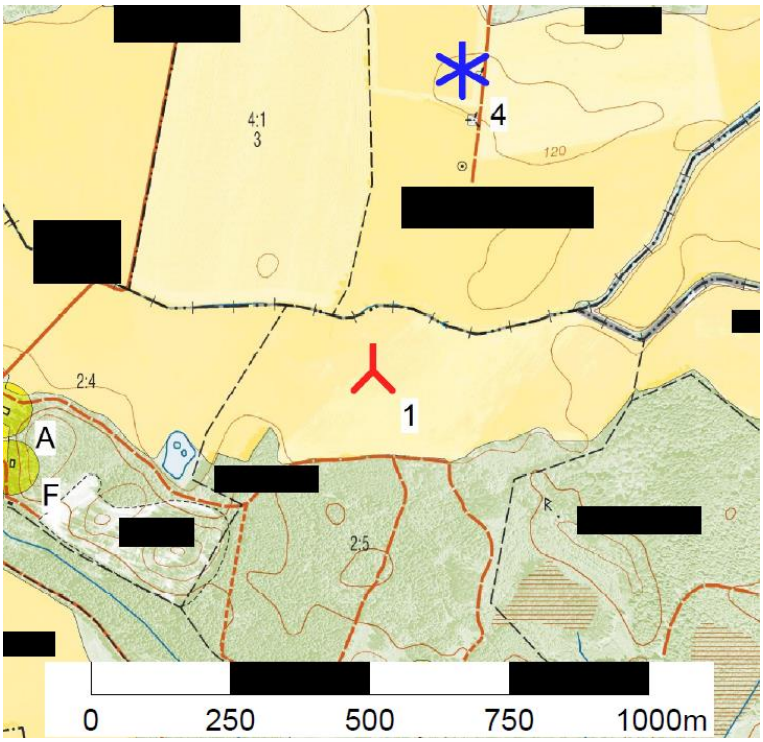


Figure 5.3 The map is showing the surroundings of WT1 (red marker 1), where another wind turbine can be seen to the north (blue marker 4). Nearby location names have been censored to withhold the actual location of the turbine.

System frequency

Trying to make the model even more precise, the system frequency has been included as an input in this study. The contribution might be marginal, but it has an effect on the electromagnetic torque, and have potential to influence the power output behaviour (Kayikçi & Milanovic, 2009).

5.3.2 Output parameter selection

A selection of parameters were selected as outputs in order to be able to monitor their behaviour and detect anomalies. Initially only power output was considered, but since the neural network can be used to predict other values as well, two more parameters were included to determine whether the network is suitable for this task, in addition to power prediction.

Power output

The most interesting parameter, and the main subject of the study is the power output. It is the results from this output that will be the basis for performance validation and it is the fitting of this output that will have most significance when determining the network architecture and configuration.

Pitch angle

Since the power of the turbine is controlled mainly by the pitch angle, any divergence from normal behaviour can be interesting, in order to detect faulty behaviour. If the pitch angle is found to diverge from set control behaviour, this could mean there is adjustments to be done to the pitch control.

At high wind speeds, the pitch angle is controlled by comparing the instant value of the power output to the rated power, adjusting the angle by demand until the right C_p is reached. If the pitch angle does not follow the trained behaviour at high wind speeds, this could mean that the C_p at high winds has been adjusted to be higher than what is normal, indicating that the turbine is not generating as much power as it should at that certain wind speed. Since the power output is limited to the rated power, this would not be visible in the power output of the ANN when the turbine is operating above rated speed.

Rotor speed

The coefficient of power depends on rotor speed and pitch angle, so this parameter could be interesting to monitor. If the rotor speed deviates from the normal behaviour, it could indicate for example faulty controlling or mechanical wear.

5.3.3 Performance

In this study, the performance function used during the training is the mean square error (MSE) function, which sums the squared errors between target and prediction value. The target is to minimise the MSE, and the performance is evaluated each epoch in the algorithm, adjusting the synaptic weights until no further improvement in performance is reached.

5.3.4 Generalisation

To improve generalisation, MATLAB has two functions built in to its neural network functions. Firstly regularisation, where the performance function is modified to enhance the generalisation, but might lead to decreased performance. Secondly, there is early stopping, where the training stops if the validation error increases for a specified number of iterations. (Demuth & Beale, 2002)

To be able to quantify the generalisation between different networks and configurations, the standard deviation of training, test and validation values are used (Eq 5.1). The qualitative significance of this parameter itself is not important, but the minimisation of it during network selection is. The thought is that if the values are close to each other, the standard deviation will be low. This value will be called the generalisation factor (GF).

$$GF = STD(P_{test}, P_{val}, P_{train}) \qquad \textit{Eq 5.1}$$

5.3.5 Architecture and configuration

Three different network architectures have been tried in this study. Two feedforward networks using a single (FF1) or two hidden layers (FF2) and a nonlinear autoregressive network with exogenous inputs (NARX) network. Repeated random sub-sampling validation is used to divide the training set into the three sets; validation, test and train, which are used to evaluate the network during iteration. The division used was 60/20/20 percent for training, test and validation, respectively.

To decide the optimal number of neurons, ten iterations of each number of neurons were performed. Each set of tries was then evaluated by both their performance as well as their generalisation value (Eq 5.2), where n is the index for the number of iterations.

$$N_{best} = \min\left(\frac{GF_n}{\max\left(median(GF_n)\right)} + \frac{MAE_n}{\max\left(median(MAE_n)\right)}\right) \qquad Eq\ 5.2$$

Since the training algorithm is initiated by setting all weights to a random starting value, the process might get stuck in local minima's, and some unreasonable performance values sometimes appear. By using the median instead of the mean, the most extreme values will not have influence over the general behaviour of that particular set of neurons, and since of this inherent randomness, the algorithm will end up with slightly different weights each time it is run. To find the best possible network to use for the model, additional iteration is needed to make sure that the network created is not a product of a bad starting weights. Hence 30 more iterations were run using the best neuron configuration to find the best possible synaptic weights. Below a plot of the described procedure can be seen.

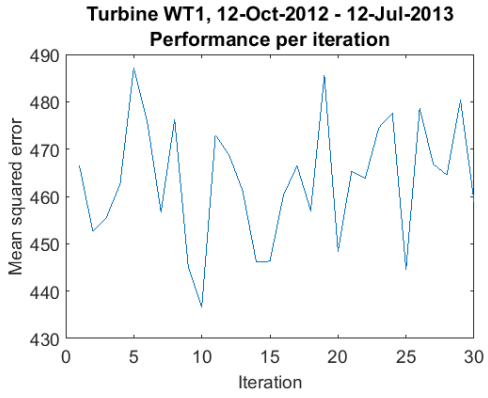


Figure 5.4 The plot shows the total performance in form of MSE during 30 iterations. The difference between the best and worst network is rather large, which proves that iteration is needed.

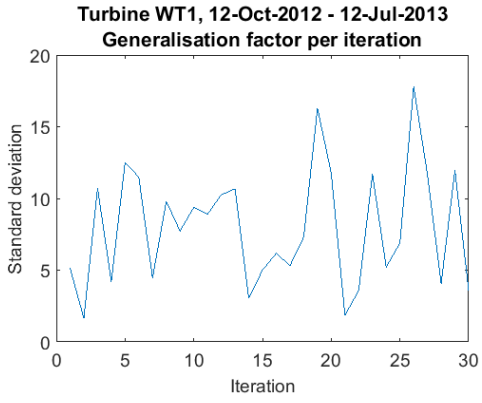


Figure 5.5 In this plot the standard deviation between training, test and validation set is plotted for each iteration.

When the training algorithm finishes, the plots and data is saved into a .mat file and a function using MATLAB's function *genFunction* is generated, containing all the synaptic weights and input parameters for the best net found during the iteration.

5.4 Selection of training data

When training a neural network to model normal behaviour, the importance of selecting healthy data is paramount. A number of measures are taken to exclude points that might lie outside the operating limits of the system, which in worst case might result in an ANN that does not indicate abnormal behaviour.

5.4.1 Manual selection

To find a good and representative data set for training, the first step is a coarse manual selection of data. The criterion are that the set should contain few long term faults, no replacement of vital parts and a wide data range for all parameters. This means that the training set should contain data from all seasons of the year and high as well as low wind speeds.

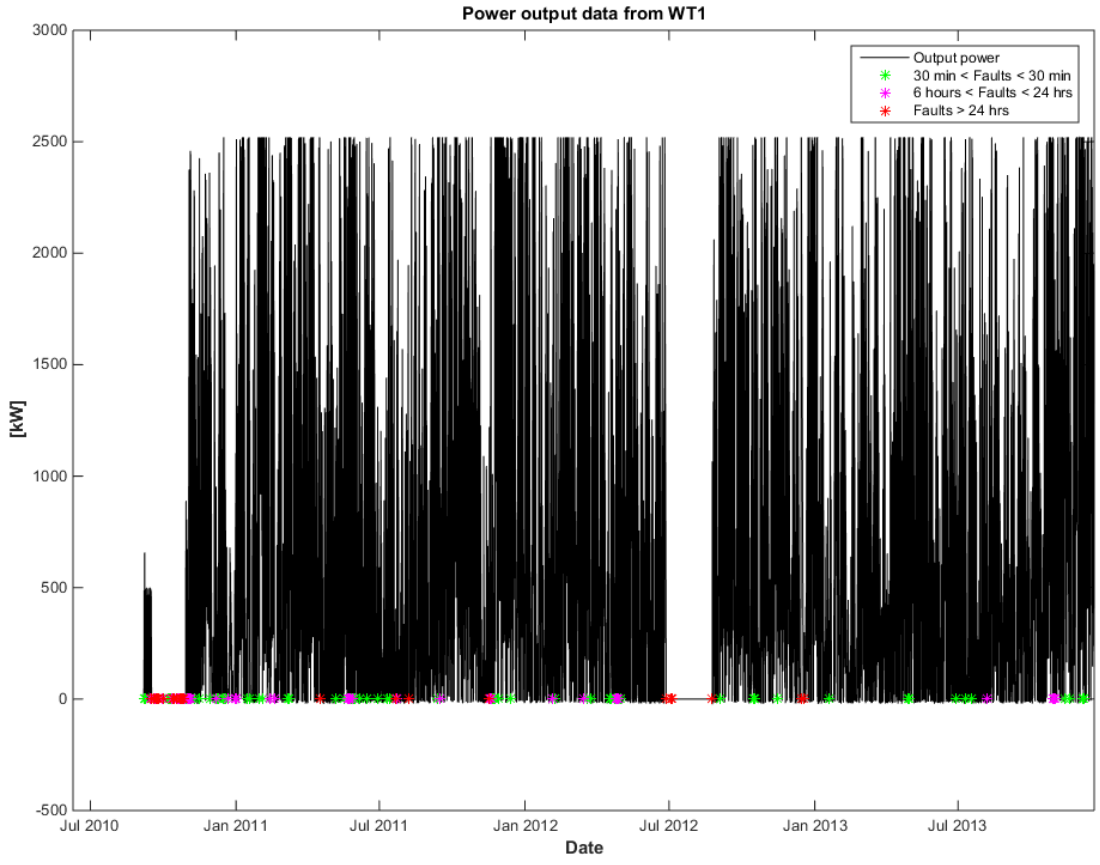


Figure 5.6 The figure show historic power output. Also shown in coloured stars are the points where errors has occurred.

Above the historical power production from WT1 is shown, figures belonging to the remaining turbines in this study can be found in Appendix 9.3.

Curtailment

The model is to be trained with normal behaviour, and in this study curtailment is not considered normal, even though it is set manually. This is because to train this behaviour in to the model, data containing curtailment must be present in the training set. Considering the data gathered in this study, curtailment has only been active a few days. Below in Figure 5.7, the time points of curtailment can be seen. In the figure, one can see that the whole spectra is shifted to the left, meaning that the pitch control has been adjusted to lower the coefficient overall, instead of simply adjusting pitch to a set output.

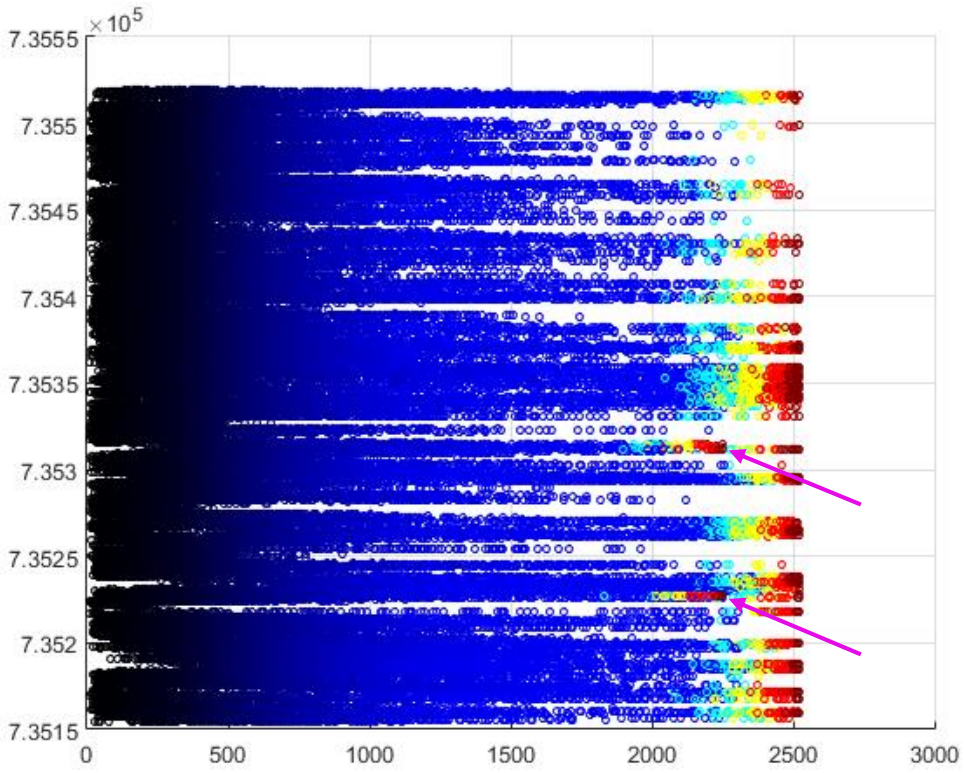


Figure 5.7 The plot is showing the behaviour of curtailment. Y axis is time in numerical format and X axis shows power output at given time point. The colour coding is done by wind speed, ranging from low (black) to high (dark red). The magenta arrows show the instances of active curtailment.

Since the pitch angle control differs greatly during curtailment and normal operations, data points from active curtailment will be excluded not to interfere with the clustering of data for filtering. In this study this is done manually by screening, but could easily be done automatic, since there is a parameter in the SCADA system. However, this effect was overlooked during the database extraction phase and this parameter was not included in the output data.

5.4.2 Validation

To ensure that the model is not fed with training data containing bad sensor readings, a rather wide range per parameter is set to filter out faulty data *(Schlechtingen & Ferreira Santos, 2010)*. To find good intervals, the data from all turbines was analysed and minimum and maximum values of interesting parameters were extracted. With the recorded min and max values, together with a general knowledge of the systems operating range, a valid range is set according to *Table 5.1*.

Table 5.1 Boundaries used when pre-processing the data through validation

Parameter	Min	Max	Set range	Unit	Comments
System frequency	49.24	64.77	49.5 - 50.5	Hz	Should be at 50
Power factor	-1	1	-1 - 1	-	
Blade pitch angle	-9.99	109.98	-3 - 95	°	The extreme min and max values coincide (signal fault), normal min is around -2.75, normal max 92
Rotor RPM	0	17.85	0 - 18	Rpm	Only a few values over 17, which are above rated speed.
Nacelle direction	0	359.9	0 - 359.9	°	
Outdoor temperature	-15.5	30.5	-40 - 40	°C	Set range expanded, still reasonable temperatures for Sweden
Wind speed	0	44.5	0 - 50	m/s	
Power output	-987.23	3726.53	-275 - 2650	kW	A few extreme values shown as min and max

The operating range will also be used in when evaluating the model, since including these faulty data points would render a bad model response due to signalling fault.

5.4.3 Normal behaviour

Normal operating behaviour needs to be defined, in order to continue the filtering of training data. The following limits were imposed on the training set:

- Data points where maximum wind speed has reached more than 25 m/s. Here the turbine will have stopped to prevent damages. These points will be very situational and will not fit any pattern, thus cannot be taught to the model.
- Data points where minimum power is less than 0 kW. In these intervals, the turbine has been switched off during the 5 min sample period, thus a good representation will be hard to achieve.

5.4.4 Clustering

In one of the papers found during the literature study (Kusiak & Verma, 2013), the authors use the power curve, rotor curve and blade pitch curve to monitor turbine performance. Clustering is done with the k-means clustering algorithm with wind speed and power output as distance approximation, outliers are detected using Mahalanobis distance from each cluster centre. It was the methodology used in this paper that

inspired the filtering technique used in this thesis. But instead of using only wind speed and power output as clustering, the relationship between all studied parameters were considered in a multi-dimensional approach.

By dividing the data into clusters, the cluster centres becomes more well-defined and it is easier to detect outliers by comparing the distance between the cluster mass centre and the individual data point. A more intuitive way to see this is that the turbine behaves differently in different parts of the power curve, and by dividing the curve into smaller parts, the behaviour in each part of the curve has a higher probability to follow the same behaviour than if the whole curve was considered.

Choosing too few clusters could lead to that points belonging to different sub sets ends up in the same cluster, and since the distribution is different between these sets, the identification of outliers get much more complicated. Choosing too many clusters could mean that a set of outliers get clustered together, and the majority of these are seen as part of the actual curve.

The number of clusters used during the formulation of the filter was decided by studying the four-dimensional plot in Figure 5.8 and dividing each parameter in to intervals where their behaviour changes, e.g. where the pitch angle changes drastically to follow rated output. The selection of these intervals can be seen below in Table 5.2.

Table 5.2 Chosen intervals used as base for deciding the number of clusters. Figure 5.8 can be used as reference to better understand the thought intervals.

Parameter	Interval				Unit
Wind speed	0 → 5	5 → 8	8 → 11	11 → 25	m/s
Ambient temperature	-30 → -3	-3 → 5	5 →30	-	Celsius
Pitch angle	-5 → 0.5	0.5 → 4	4 → 90	-	Degrees

Multiplying the number of intervals in each set, results in a total number of 36 clusters, which was used during all data clustering.

5.4.4.1 Selection of clustering parameters

Plotting the remaining data points shows that there are certain points that are not supposed to be part of the training set. In order to filter these out, the relationship between several parameters have been studied. Studying the relationship between wind speed, power output, pitch angle and outdoor temperature, it is obvious that these are closely connected and thus suitable for clustering.

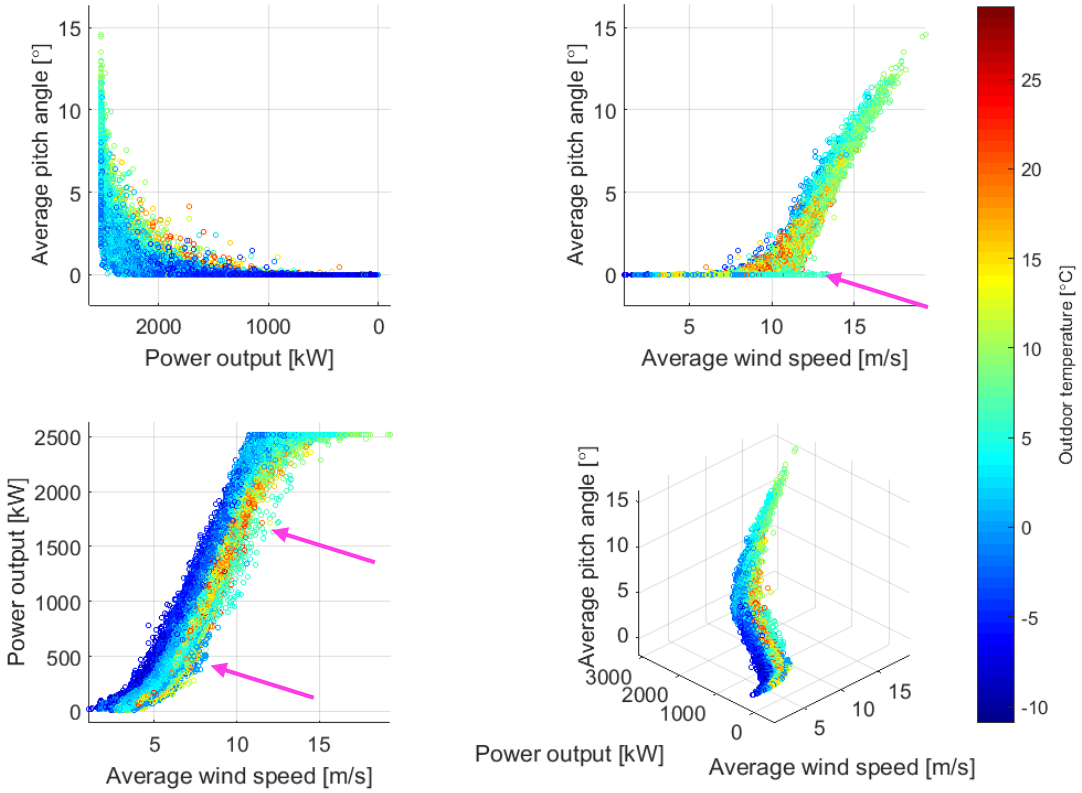


Figure 5.8 Four plots of the same curve, shown from different viewing angles. The bottom right plot shows the curve in a 3D view. The magenta arrows points at interesting areas, where outliers clearly can be seen, as they do not follow the general behaviour of the function.

In Figure 5.8 above, the effect of temperature can clearly be seen, and studying the lower-left figure, the fourth dimension added by the temperature colour map makes it is easy to spot data that clearly does not fit the pattern.

5.4.4.2 Clustering method

The clustering was done using the selected input and MATLAB's *clusterdata* function, with Ward's method. This method was partly chosen because of limitations in hardware to handle such large data sets as used in this study. Ward's method used the squared inner distance between the points and cluster the points into a given number of clusters according to their relative distance in the Euclidean space (Eq 5.3).

$$d_{ij} = \left\| X_i - X_j \right\|^2 \qquad \textit{Eq 5.3}$$

5.4.5 Multi-dimensional filtering using Mahalanobis distance

The output parameters of the model depends on a range of parameter as seen in Section 2.1, and the purpose of filtering data before training is to exclude points where the output in relation to the other parameters does not follow the systems pattern.

Filtering the data is quite hard to do in an automated way. A hypothesis in this thesis is that a lot of information is hidden in the reported standard deviation values. In conventional studies, where only the average values are used, a rather aggressive filtering is used to exclude points that do not follow the general shape of the power curve. This approach is based on the assumption that outliers are situated outside the thought envelope of the normal power curve (see Figure 5.17 for better understanding of the reasoning).

This leads to that all points inside this range will be considered normal operations, while they in fact may be abnormal. In this thesis, the thought is to include more dimensions to determine whether a point can be considered normal or outlier. This means that the filter proposed should be able to determine points situated inside the considered normal range of the power curve, can be identified as outliers. A simple example in three dimensions is shown below in Figure 5.9 and Figure 5.10.

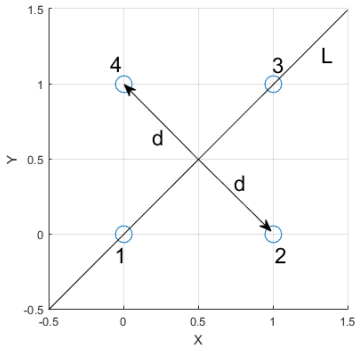


Figure 5.9 Four points in a two-dimensional plot, point 1 and 3 following the line L and point 2 and 4 at the distance D from the line.

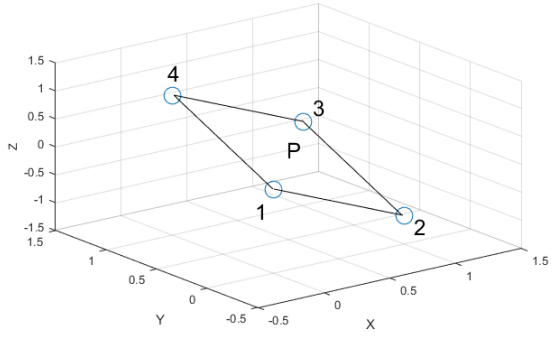


Figure 5.10 The same points seen in a three-dimensional plot, showing that all four points in fact lies in the plane P.

If the function in this example is seen as a two dimensional problem, such as $y = x$ in Figure 5.9, point 1 and 3 would follow the line L and be considered a good fit, while point 3 and 4 would be considered deviating from the function. But by considering a third dimension as seen in Figure 5.10, one can see that the points lie in the plane P and follow the function $y = x + z$.

5.4.5.1 Selection of filtering parameters

The selection of parameters was done by studying four-dimensional plots (3D-scattering and colour maps) from different viewing angles. Distinct patterns between some parameters was found, thus they were chosen for the filtering, since they should

be strongly associated in their respective cluster. Below, a series of plots are listed to give an insight to the reasoning.

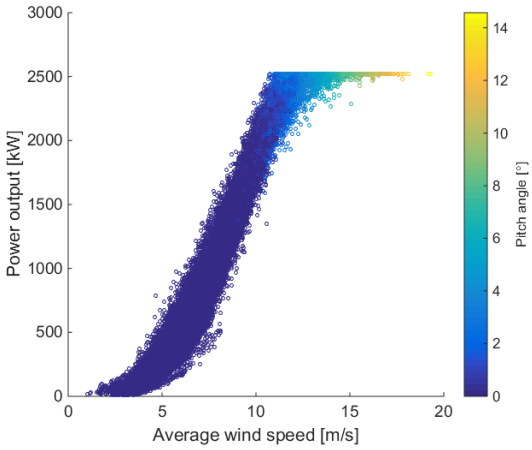


Figure 5.11 The pitch angle visualised by colouring.

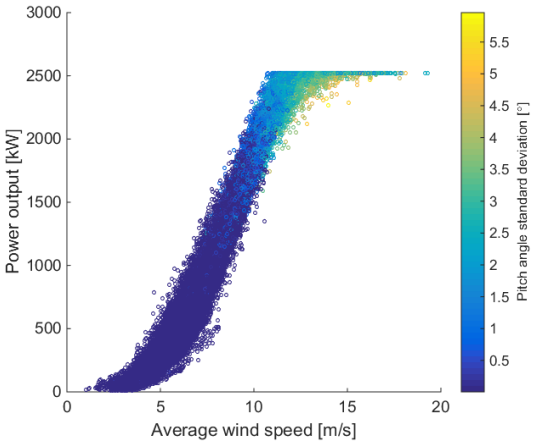


Figure 5.12 The standard deviation of the pitch angle visualised by colouring.

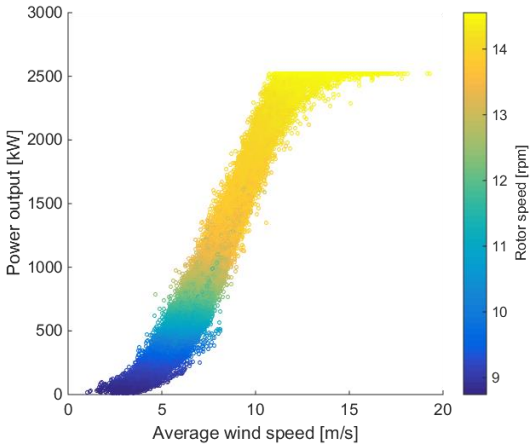


Figure 5.13 The rotor speed visualised by colouring.

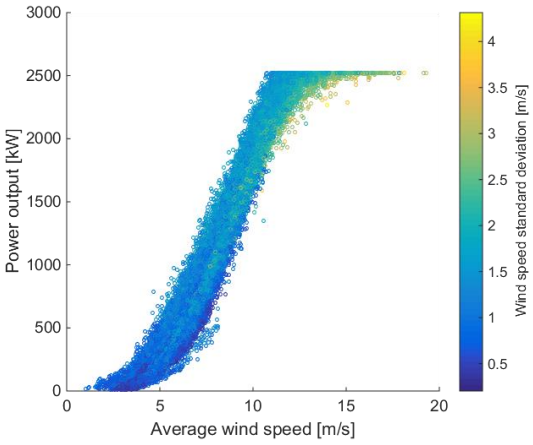


Figure 5.14 The standard deviation of the wind speed visualised by colouring.

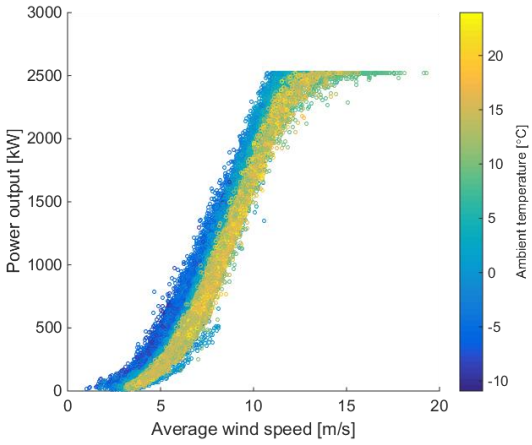


Figure 5.15 The ambient temperature visualised by colouring.

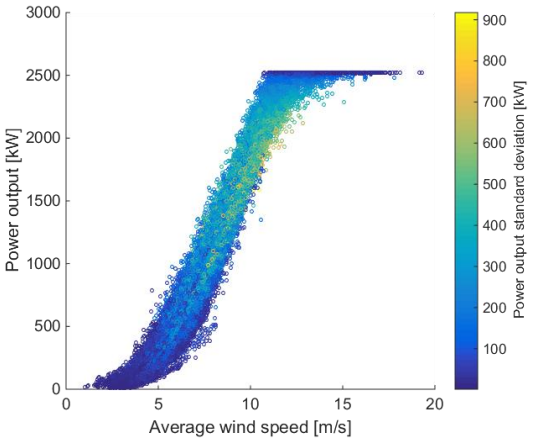


Figure 5.16 The standard deviation of the output power visualised by colouring.

In Figure 5.11 through Figure 5.16, six of the chosen parameters are plotted with a colour map against the two remaining parameters, to show the patterns they generate in the wind speed-power plot.

Table 5.3 A list of the parameters chosen for the filtering

Chosen filtering parameters			
Power output	Wind speed	Pitch angle	Rotor speed
Ambient temperature	Power STD	Wind STD	Pitch angle STD

5.4.5.2 Filtering conditions

Since the data is clustered by their relative distance, each cluster should contain series of samples belonging to the same distribution. By using the Mahalanobis distance, the distance from the distribution is measured by their distance from the cluster mass centre in form of standard deviations from the mean.

The Mahalanobis distance for each point in every cluster was evaluated and the 2.5 % points with the largest Mahalanobis distance were considered outliers. This level was set since the data set at this point, after being pre-filtered using the previously described methods, was considered relatively free from abnormal data.

Another method tried was to use MATLAB's *fitdist* function to find the individual cluster's distribution, and use limits based on the standard deviation of this distribution as a threshold for outliers. But since this would mean each cluster would exclude 2.5 % - 5 % points from the set, a large portion of valid points could end up excluded if the cluster contained mainly normal data.

5.4.6 Other approaches considered

Before formulating the filter used in the study, a somewhat simplified version of the "exponential LMedS" method was used (Sainz, et al., 2009). In this method, a single equation is used to fit the whole wind speed range, where the equation is double exponential to handle the two inflexion points (Eq 5.4).

$$P(v) = \exp[-\exp(a - b * v - c * v^2 - d * v^3)] \qquad Eq\ 5.4$$

Where P is the power depending on v, the normalised wind speed, and a through d are constants to fit the function.

The wind speed and power output were normalised in accordance to the read study, and a fitting function based on this equation was created using MATLAB's function fitting tool, using a Non-Linear Least Squares method and the Levenberg-Marquardt algorithm. The reasoning behind the threshold setting for the filtering process was not very comprehensive, which led to that the 95 % confidence interval was used in the simplified version of this technique. The interval also needed to be shifted somewhat not to exclude points close to minimum and maximum wind speeds, since the function value gets very narrow close to zero and one (Eq 5.5 and Eq 5.6).

$$\lim_{v \to 0} P(v) = 0 \qquad Eq\ 5.5$$

$$\lim_{v \to 1} P(v) = 1 \qquad \textit{Eq 5.6}$$

This filter behaves good and follows the behaviour in the study well when there is a healthy amount of data, but the filtering envelope created by the confidence interval gets very wide when less than 6 months of data is used, which is probably due to the fact that there are a lot of measuring points in the low wind speed region. This led to that the confidence interval had to be changed according to the amount of data, which meant normal behaviour points close to zero and one were being excluded and the process was getting somewhat arbitrary and too manual to be used as the preferred choice. In Figure 5.17, Figure 5.18 and Figure 5.19 the behaviour of the filter at different sizes of data sets is shown.

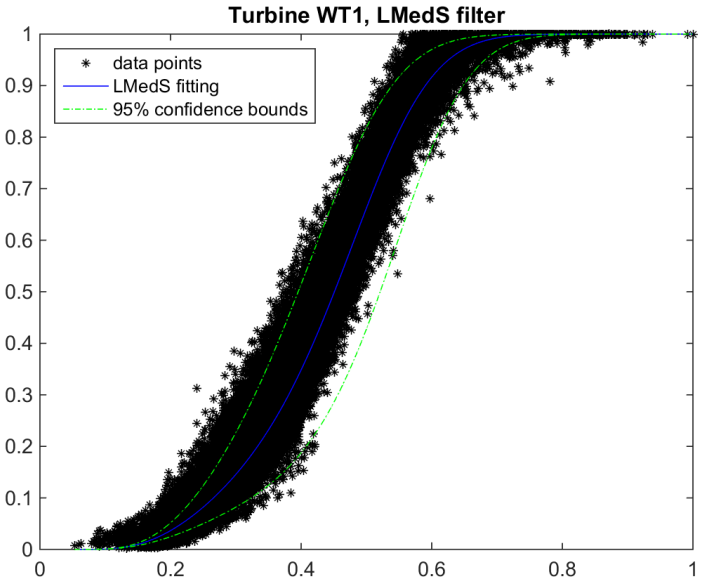


Figure 5.17 The LMedS filter used on one year of turbine data. Here the filter behaves reasonably well and excludes what seems to be outliers quite good.

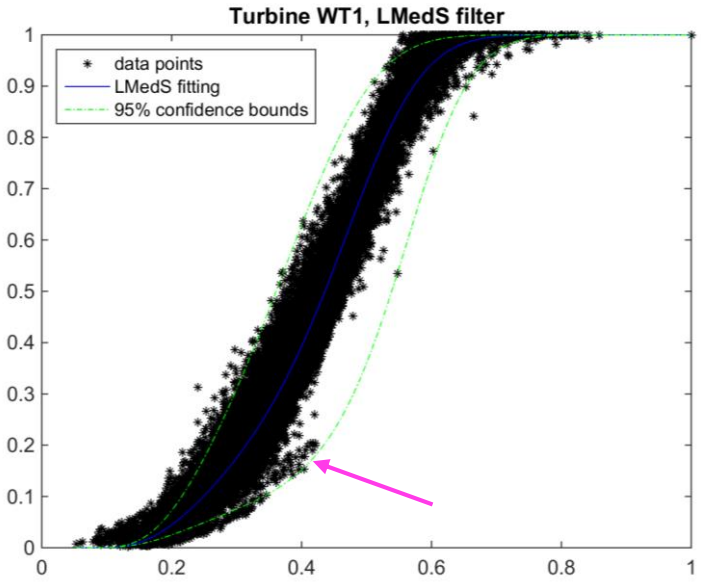


Figure 5.18 With six months data the filter starts to behave badly, by including some points that are most likely faulty, as shown by the magenta arrow.

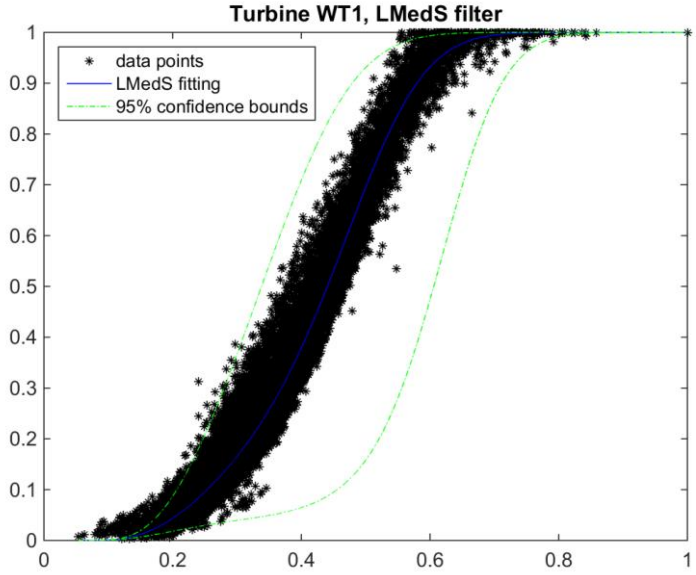


Figure 5.19 With three months data, the confidence interval has expanded by too much, and this filter will not be usable to exclude outliers.

As can be seen in the figures above, this filter method excludes more normal than abnormal points and becomes unusable when the training set is too small.

5.5 Interpretation of the model response

When a set of historical data is tested with the constructed function, the output response is to be considered the normal response to the inputs at the given point in time. The thesis has two goals for the model, performance measure and fault detection. The test data needs to be cleaned up to an extent to be usable in the model, since the ANN is only trained to predict a turbine under operation.

5.5.1 Test data selection

Before feeding the test data to the ANN, the same validity check is performed as in the training data selection (Sub-section 5.4.2), to exclude points where there might be sensor faults. The ANN is only trained to predict a turbine in operation, hence a criterion is set to exclude points where the power output is lower than 25 kW.

The model is furthermore not trained to interpret cases where the maximum wind speed exceeds 25 m/s, since the turbine will shut down to avoid damages at these times. Another special case is when curtailment is active, as discussed in sub-section 5.4.1. To avoid erroneous alarms and misguiding residues, the model response at these periods is replaced with the actual output. But since curtailment is rare, the loss of proper modelling during these few days is considered negligible.

5.5.2 Performance monitoring

The performance error is simply calculated by comparing the real output to the output calculated by the ANN (Eq 5.7).

$$Error_i = Output_i - ANN(input)_i \qquad Eq\ 5.7$$

By simply plotting this error over the time, a rather erratic curve is produced and it is hard to discern any real trends just by studying this plot. To overcome this, a daily average is calculated which shows a more readable curve. This is also where the interesting statistics lies in, since the model have a fault range and might show large faults during certain conditions that might not have been trained into the model. Especially instances where the turbine has shut down during a five minute period without being close to maximum rated wind speed, which the model has a hard time interpreting since there is no apparent reason for underproduction at the moment. The calculation of the daily average per output parameter is shown in Eq 5.8 below, where the sum of errors during the time of averaging is divided by the number of time steps for the time.

$$Output_{av,i} = \frac{\sum_{i=T-1}^{T} Error_i}{N_{steps,T}} \qquad Eq\ 5.8$$

The training set is considered to be relatively free from outliers and the residues roughly fits a Normal distribution, hence a value of plus/minus two times the standard deviation of the residues was chosen as the normal range of operation (Eq 5.9).

$$Normal\ range = \pm 2 * \sigma_{training} \qquad Eq\ 5.9$$

To be able to quantify the meaning of underperformance, the sum of the total loss of generation will be calculated via Eq 5.10, where the length of the time step in minutes is divided by the number of minutes in an hour and multiplied by the error in each time step.

$$E_{loss} = \frac{T_{step}}{T_{hour}} * \sum_{t=1}^{T} Error_t \qquad \textit{Eq 5.10}$$

5.5.3 Error thresholds for fault detection

For fault detection, a normal range for each output needs to be calculated. This is done by following the method in a dissertation by (Bangalore, 2014), where the training set, which is considered to be representative for normal behaviour, is used to calculate the Mahalanobis distance for the training targets and the corresponding errors (Eq 5.11).

$$X_{norm} = [Error_{train}, Output\ target_{train}] \qquad \textit{Eq 5.11}$$

The training set X is constructed by pairing the error of each point in the training data with the output reported by SCADA, this way the normal Mahalanobis centroid can be constructed using Eq 5.12.

$$\left(D_{M,norm}\right)_i = \sqrt{((X_{norm})_i - \mu_{norm})S_{norm}^{-1}((X_{norm})_i - \mu_{norm})^T} \qquad \textit{Eq 5.12}$$

To show the meaning of this in a more intuitive way, the target power output and errors are plotted, with their respective Mahalanobis distance shown by colour coding.

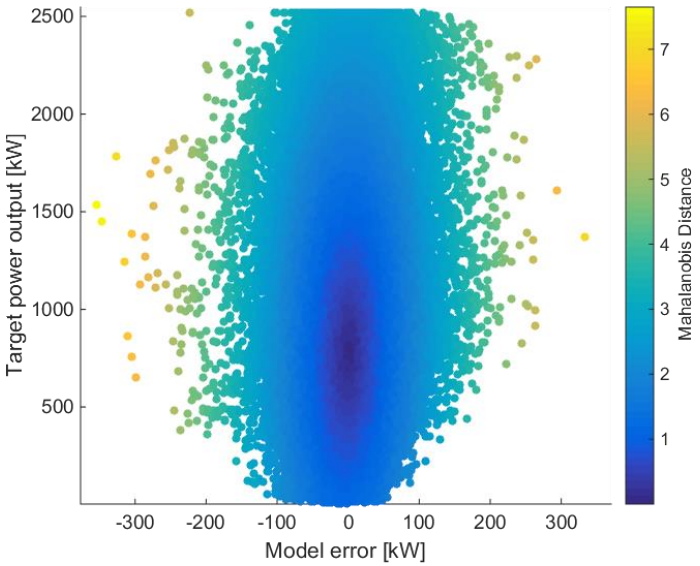


Figure 5.20 The target power output here plotted against the model error, the colour of each point indicates its Mahalanobis distance from the centroid.

Since the model has several outputs with different characteristics, the error set from each output needs to be analysed and the associated distribution determined. When the

normal Mahalanobis distance of the training set is determined, the mean and standard deviation of each set is used to set the threshold for fault indication (Eq 5.13).

$$M_{D,threshold} = mean_{D_{M,norm}} + 2 * \sigma_{D_{M,norm}} \qquad \textit{Eq 5.13}$$

6 Results

In the following chapter, presented using reasoning and plots from the model, are the results from the methodology as presented in Chapter 5. The results will mainly focus on WT1, which is the turbine used for most testing, but results from the remaining turbines can be found in Appendix 9.2.

6.1 Clustering

The clustering is presented by a series of plots, showing the results from a number of different views, since it is hard to see the full picture when the clustering is done in four dimensions.

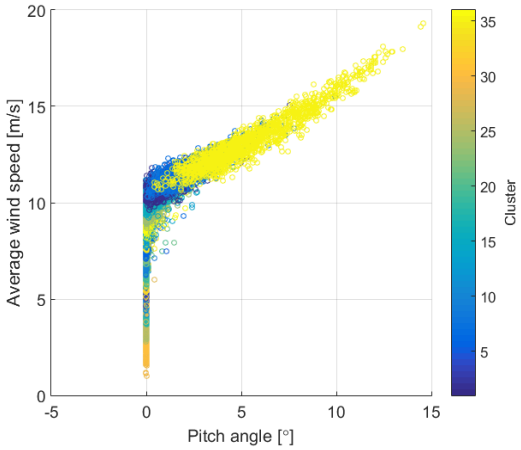


Figure 6.1 Clustering viewed from pitch angle-Wind speed view. Points with the same colour belongs to the same cluster.

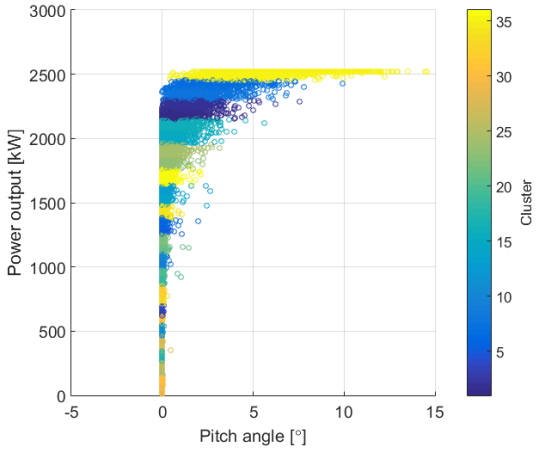


Figure 6.2 The same figure from power-pitch angle.

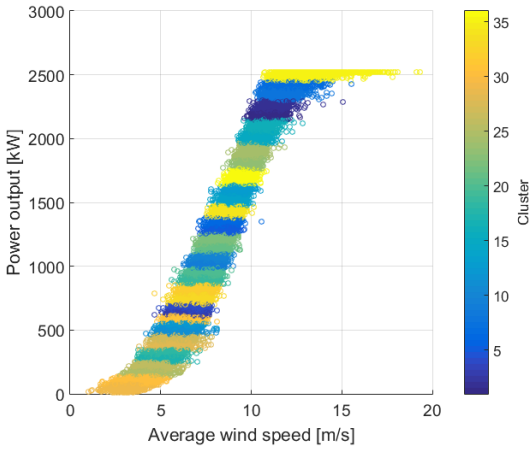


Figure 6.3 Clusters from a power-wind speed view.

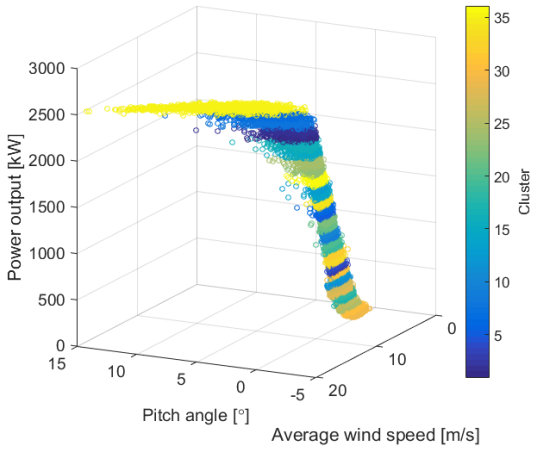


Figure 6.4 3D view of the clustering.

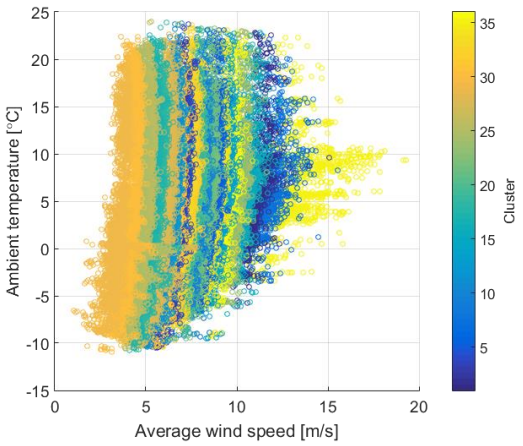


Figure 6.5 In this figure, the clusters are seen from a temperature-wind speed view.

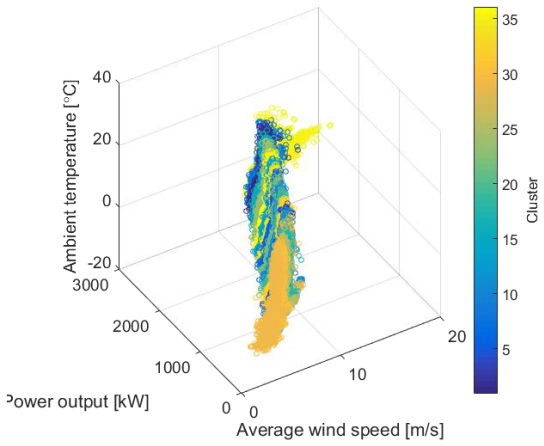


Figure 6.6 3D plot of the same curve as in the previous figure.

As can be seen in Figure 6.1 through Figure 6.6 above, the main influence in the clustering is the power output, which was expected since it is the dominant parameter. Tries were also made where the parameters were normalised ahead of clustering, but the result was a lot of local clustering of obvious faults. This would have led to inclusion of erroneous data in the training set, so the above shown clustering was deemed the best way to cluster the data.

The clusters have a good density in their normal range, which means the filtering should work as planned, since outliers will be far from the cluster centre.

6.2 Filtering

The results from the filtering will be visualised first through a series of plots showing a cluster from different perspectives,

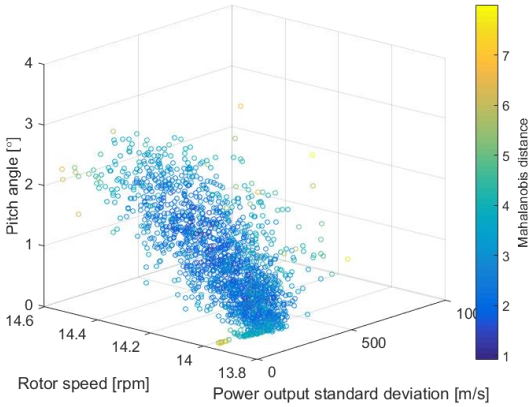


Figure 6.7 Cluster 4 seen in 3D with the Mahalanobis distance for each point shown by colour coding.

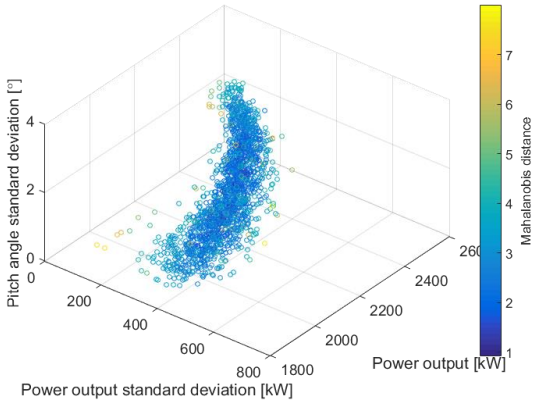


Figure 6.8 Cluster 4 again, showing the Mahalanobis distance in other dimensions.

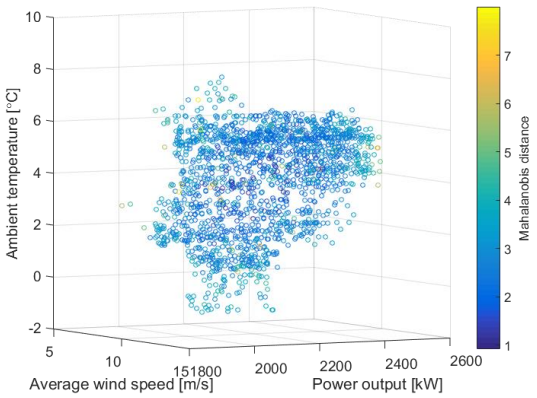


Figure 6.9 Cluster 4, seen in another perspective with other plotting parameters.

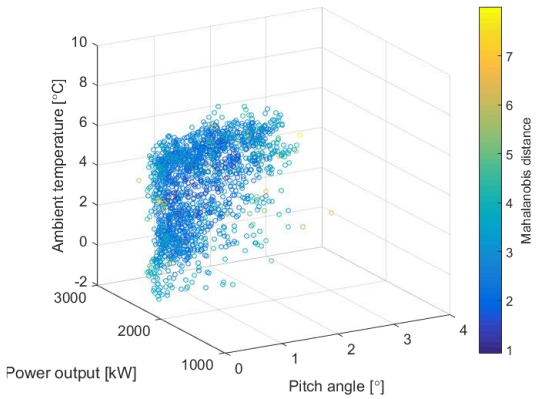


Figure 6.10 Cluster 4 shown through yet another perspective.

Figure 6.7 through Figure 6.10 shows one cluster plotted against different plotting parameters, in an effort to try to show how the multi-dimensional Mahalanobis distance works in a more comprehensive way. In the figures, the cluster centre can clearly be seen, but the Mahalanobis distance for a point will not simply be larger the further away from the centroid it gets, since it is determined in eight dimensions.

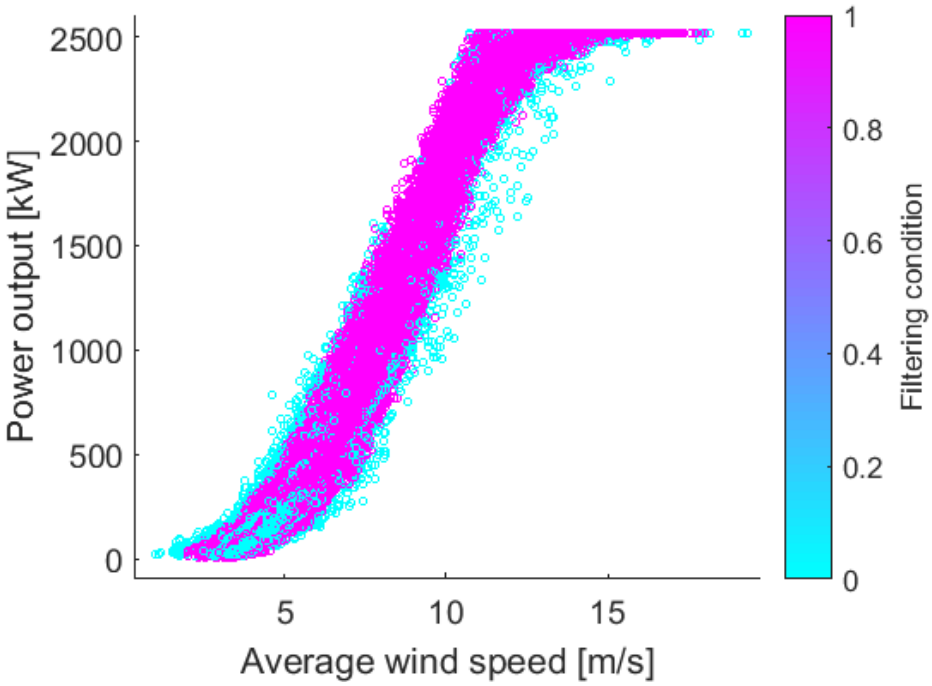


Figure 6.11 The results from the filtering shown in the normal Wind-Power plot. Cyan points represents the filtered points.

In Figure 6.11 the results from the filtering of the normal nine month model is presented and in Figure 6.12 the result can be viewed from another aspect, here the reason why there are points being filtered in the middle of the Wind speed-Power curve can more clearly be seen; they do not follow the set behaviour of the pitch curve at the given conditions.

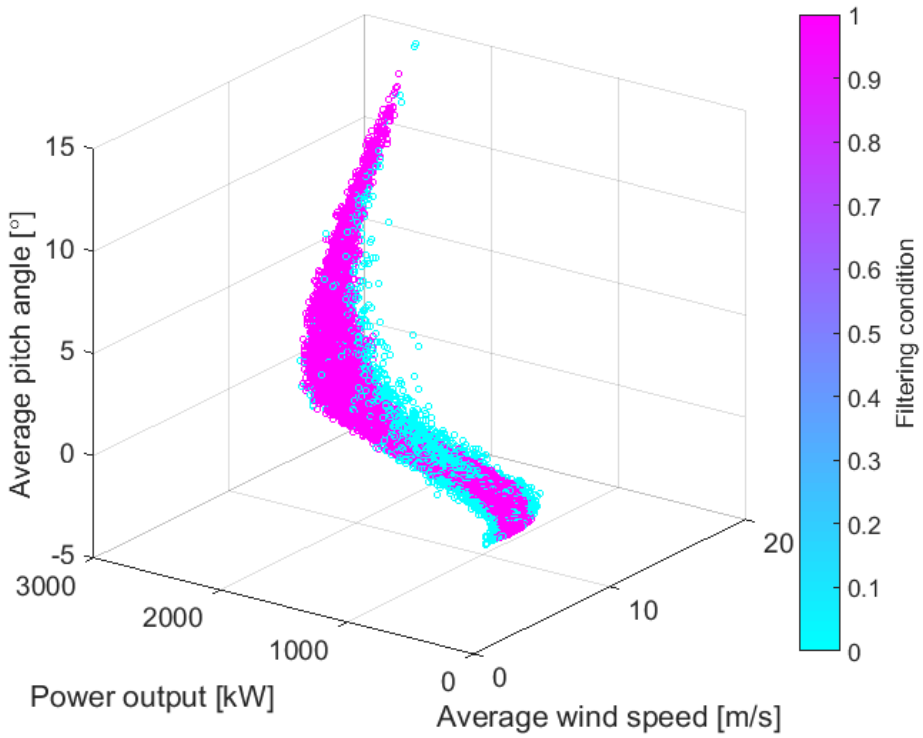


Figure 6.12 Results from the filtering of data from WT1 is shown in an expanded view, with a third dimension added in form of pitch angle.

By using this approach when filtering, points that are situated inside the thought normal range can be detected and excluded from the training set.

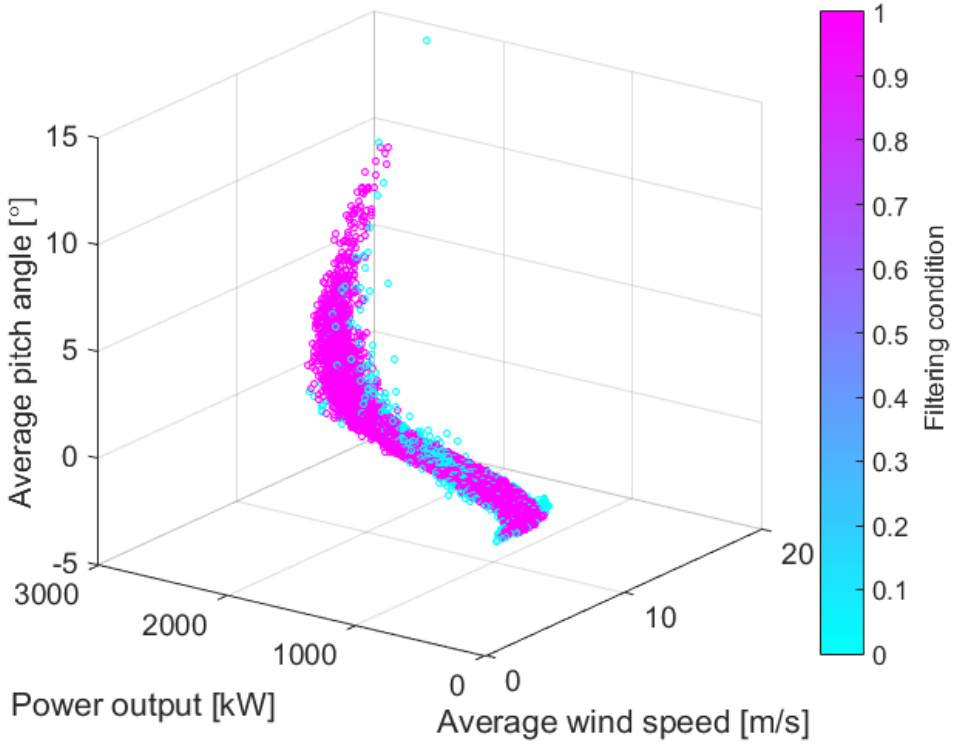


Figure 6.13 The results of filtering for a three month model.

Figure 6.13 shows that the filtering works well even for a model with only three months of data as training set. The LMedS filtering that was considered in early stages of the project did not perform well with a limited data set as can be seen in Section 5.4.6.

6.3 ANN

In this section, the results from the configuration of the three networks are presented. The best number of neurons seemed to be around 20 in total, since the results from both FF1 and the NARX net always ended up between 17 and 23 neurons; and the results from the tests for the FF2 net was near 13 neurons in the first layer and eight in the second layer through all the tests. After a large number of runs the best network configuration was decided to be 20 neurons for FF1 and NARX, and 13-8 for FF2, since the computing time for determining neuron configuration each run is large.

The performance for the NARX network was found to be much better than that of the two feed forward networks, and this network was therefore chosen for a more extensive study where inputs were varied and filtering compared. This is described in detail in Model performance, Section 6.4.

6.3.1 Feed forward net with single layer

Following the method stated in Section 5.3, each number of neurons was tested ten times, in order to find the best configuration for the modelling. In Figure 6.14 and Figure 6.15 the results from this iteration can be seen. The trend for the mean absolute error is very clear, where an increased number of neurons improves the performance. Iterations with even higher number of neurons were also performed, but the trend of improving performance ends at approximately 30 neurons, so this was set as the maximum number in following iterations. In the case of generalisation, the trend is not as visible, but the range is much wider between the maximum and minimum value, meaning this value will have higher impact when determining the optimal number of neurons for this model.

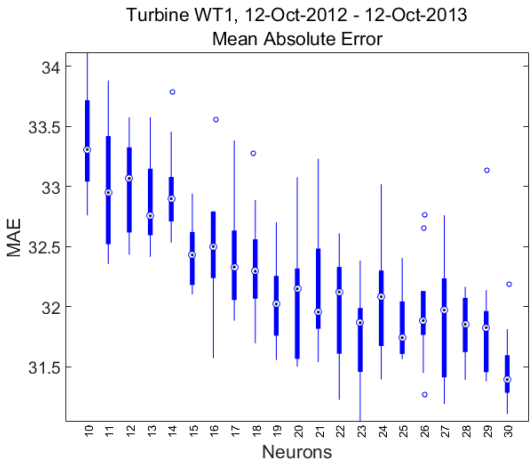


Figure 6.14 The iterative process tests each neuron ten times to statistically determine the performance. The rings represents outliers, which lies too far from the normal range, thin lines represents 1.5 times the interquartile range and the thick lines shows the interquartile range. Rings with dots shows the median of the set.

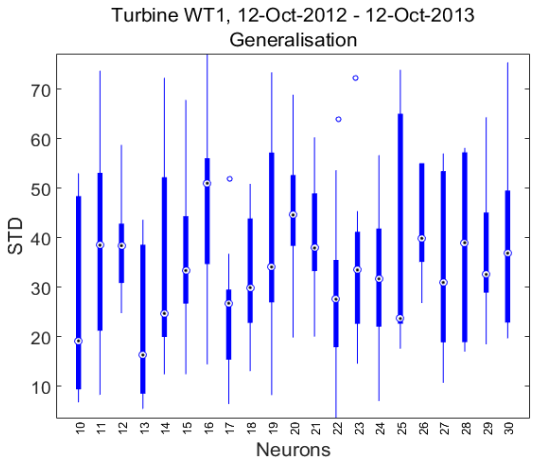


Figure 6.15 Here the generalisation is plotted for each neuron. The trend is not as clear in this plot as in the previous, but the difference is larger between the lowest and highest value.

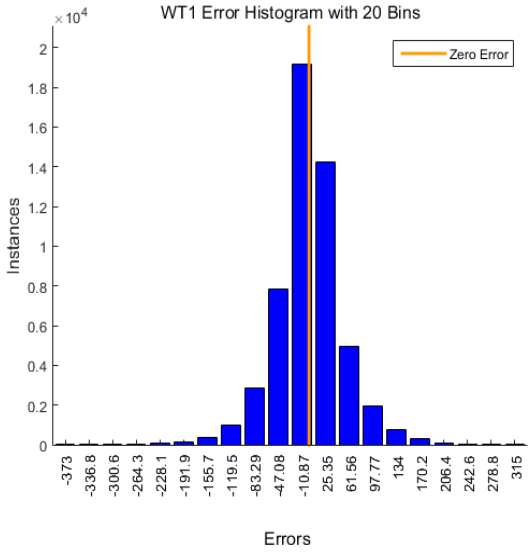


Figure 6.16 The error histogram shows the distribution of errors during the training of the best single layer feed forward network.

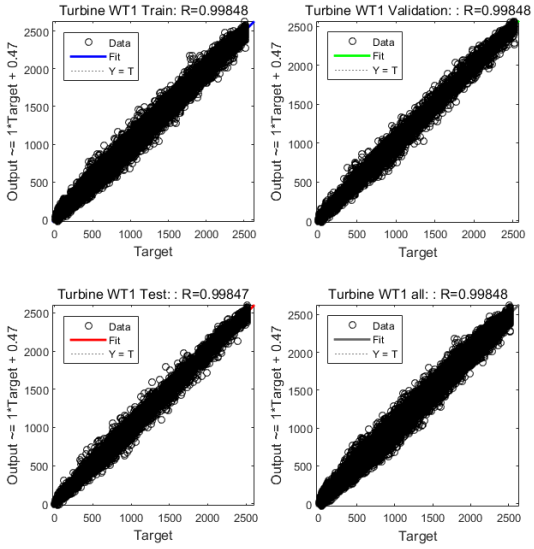


Figure 6.17 The regression plot shows the model response in reference to the output target for all parts of the training.

In Figure 6.16 the distribution of errors is shown in a histogram, the error spectra is quite tight with a low standard deviation. In Figure 6.17 the model response is plotted against the output target, it can be seen that the network performs best at high and low outputs, with a slightly worse performance in the region in between.

6.3.2 Feed forward net with two layers

In the plots below, the results from two different iterations are presented by showing the change in performance parameters, which are used to decide the best configuration.

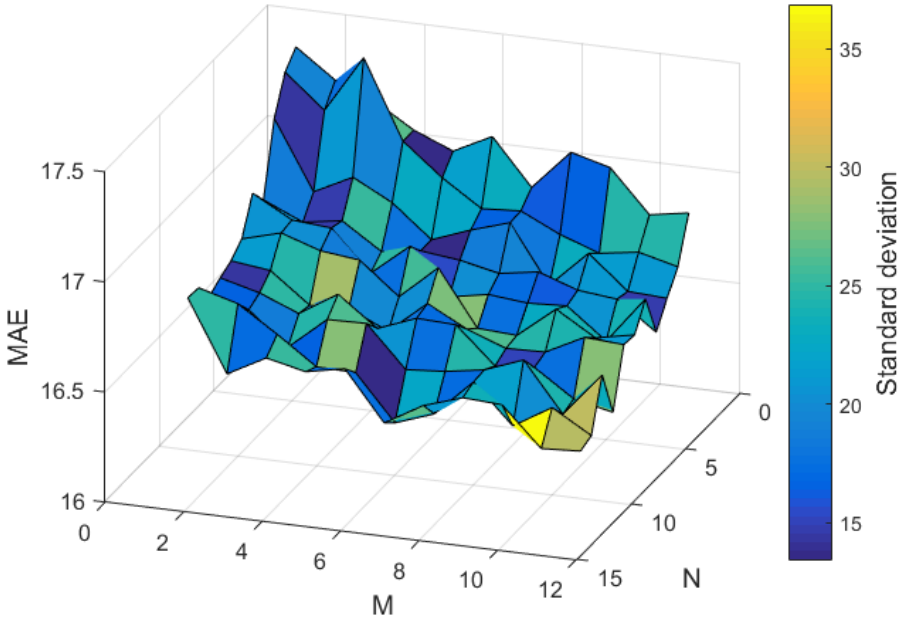


Figure 6.18 The plot shows the mean average error for each combination of neurons (M and N), together with the standard deviation for validation, training and test set at the current combination, which is used as an indicator for the generalisation of the network.

Figure 6.18 clearly shows the trend of decreased error by increased number of neurons. One can also see that the generalisation factor gets worse with increased neurons as expected, though this trend is not as clear as the first.

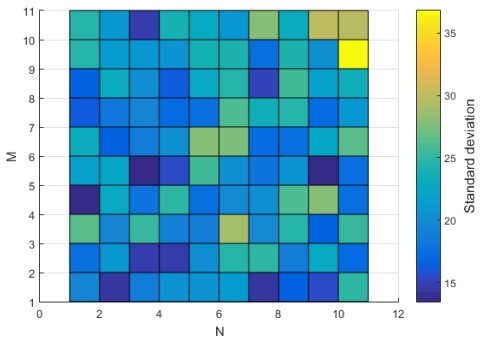


Figure 6.19 This plot shows a different run with fewer neurons than Figure 6.18, the trend of generalisation is clearer here, and it can also be argued that a network with the same amount of neurons in the first and second layers seems to perform poorly.

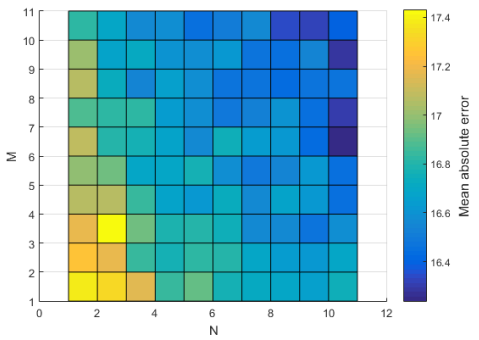


Figure 6.20 This plot is from the same iteration as the previous figure, showing the error instead of generalisation factor. Here the trend is clear that more neurons results in a better performing network.

Figure 6.20 shows the stated trend of increased performance with more neurons even clearer, but Figure 6.19 also shows that the concentration of poor generalisation behaviour is higher in the case of more neurons.

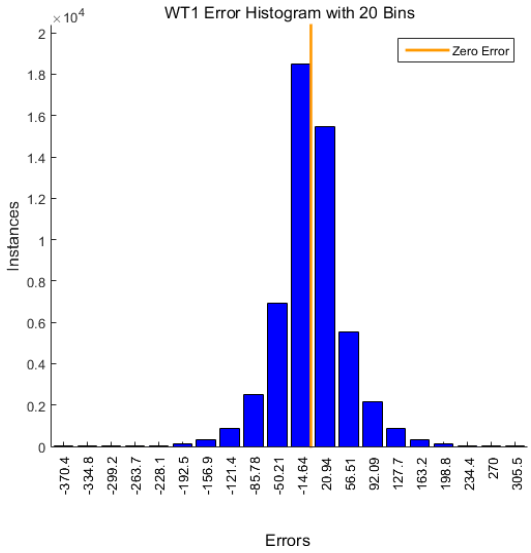


Figure 6.21 The error histogram for the two layer feed forward net looks very similar to the one of the single layer. The range is marginally smaller though and the errors are a little more concentrated towards zero.

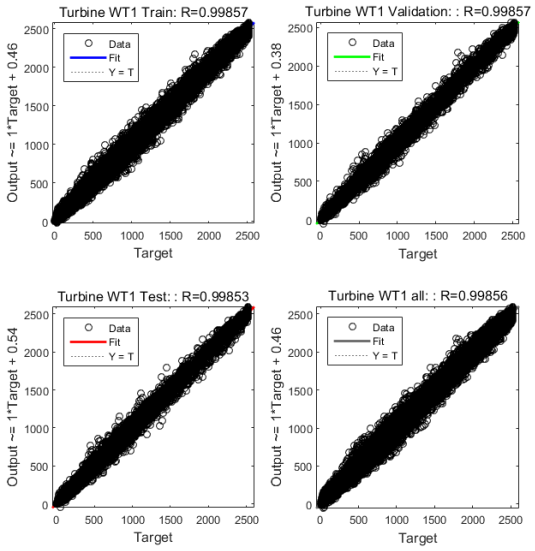


Figure 6.22 In the regression plot it can be seen that the errors are a little more concentrated to the regression line, reflected by a marginally higher R-value.

Looking at Figure 6.21 and Figure 6.22, it can be concluded that the performance is very similar to that of a single layer network. There is a marginal improvement, which is reflected later in the model performance results, Section 6.4.

6.3.3 Non-linear autoregressive network with exogenous inputs

The NARX net shows a great improvement compared to the two previously trained networks, which can be seen extra clearly in Figure 6.23, where the distribution of errors is very narrow. The improvement is not as easily seen in Figure 6.24, but it can be seen that the network can handle the middle range much better than the other networks.

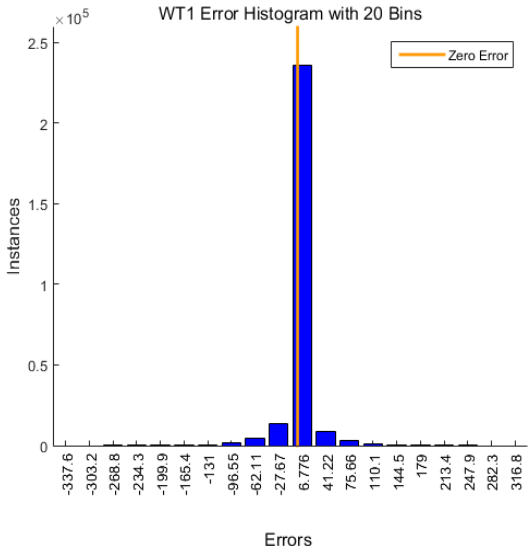


Figure 6.23 In the error histogram for the NARX net, a great improvement can be seen compared to the two previous nets. Here the errors are concentrated very close to zero.

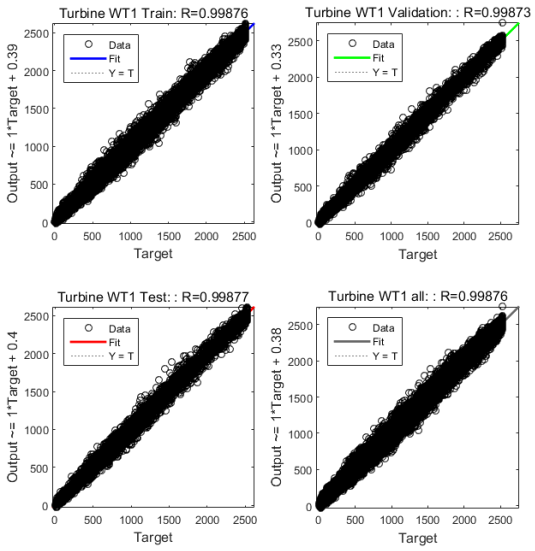


Figure 6.24 The improvement is not as prominent in this plot, but especially the regression plot for training is tighter in the middle range.

The outstanding performance of the NARX network compared to the two other models led to that this network was considered to be the best choice for the model and was tested more extensively than the remaining two.

6.4 Model performance

In this section the numerical values from the different test are presented in bar charts. In section 6.4.6 the results of this study is compared to the results of (Schlechtingen, et al., 2013b). To be able to compare the results, the method in that study was followed and the results are presented in values scaled to 100 kW.

Below in Figure 6.25 the model power output response and the actual power output is plotted in the same graph, showing the precision of the model in a wide power range.

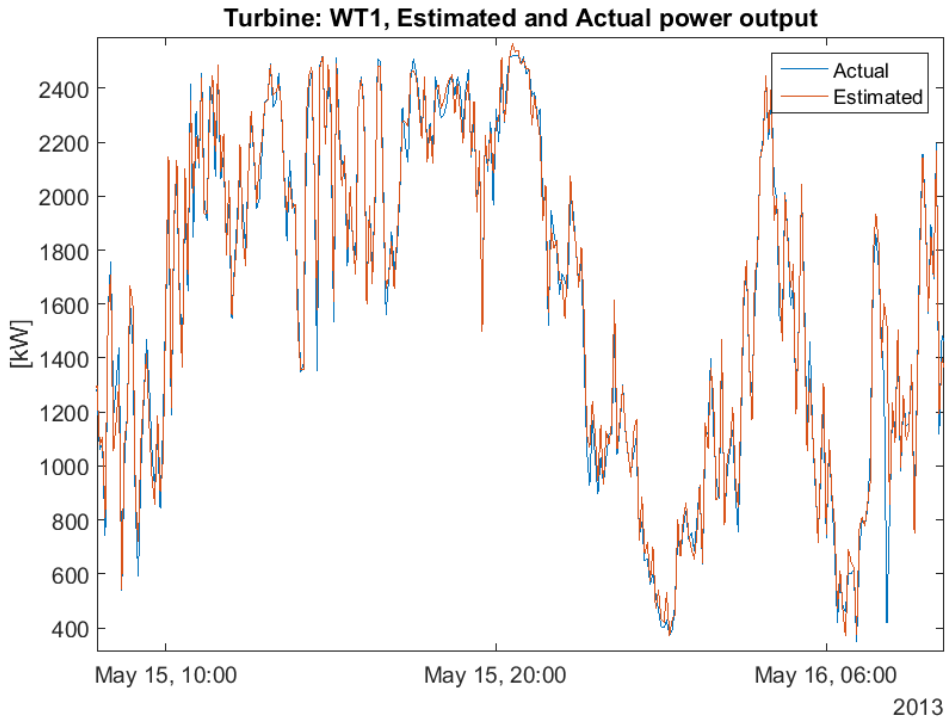


Figure 6.25 This plot shows an example of the model response in comparison to the actual output of the turbine.

In Section 6.4.1, the size of the training data is varied to study how the performance of the model changes as the network has less points for learning. In Sections 6.4.2 through 6.4.4, the results from the three different networks and the different turbines are presented, all using nine months of data for training, per the stated method. In Section 6.4.5, are the results from some additional test, done to evaluate different characteristics and the influence of a selection of parameters.

Section 6.4.6 shows the comparison of the mean value of the developed models for all turbines to those in the comparative study and a summary of all the tested models. Section 6.4.8 shows the results from the fault detection and Section 6.4.7 the output deviation results.

The table below shows the time spans used for the different models. For all the turbine models based on nine months data, temperatures from a wide range is used. All the data used for the following bar charts can be found in Appendix 9.1, including the runtime for the network in question.

Table 6.1 A list of the dates used for the different turbines and models.

Turbine / data size	Start date	End date	Turbine / data size	Start date	End date
WT1			WT2		
12m	2012-10-12	2013-10-12	9m	2013-01-01	2013-10-01
9m	2012-10-12	2013-07-12	WT3		
6m (1)	2012-10-12	2013-04-12	9m	2012-03-18	2012-12-18
6m (2)	2013-04-12	2013-10-12	WT4		
3m (1)	2012-10-12	2013-01-12	9m	2013-01-01	2013-10-01
3m (2)	2013-01-12	2013-04-12	WT5		
3m (3)	2013-04-12	2013-07-12	9m	2013-01-01	2013-10-01
3m (4)	2013-07-12	2013-10-12			

6.4.1 WT1 NARX network results for different data lengths

Below are the results from a number of different models based on a varying amount of data. The purpose of this comparison is to show the importance of size and quality of the training data. In all following bar charts, a lower value is the better.

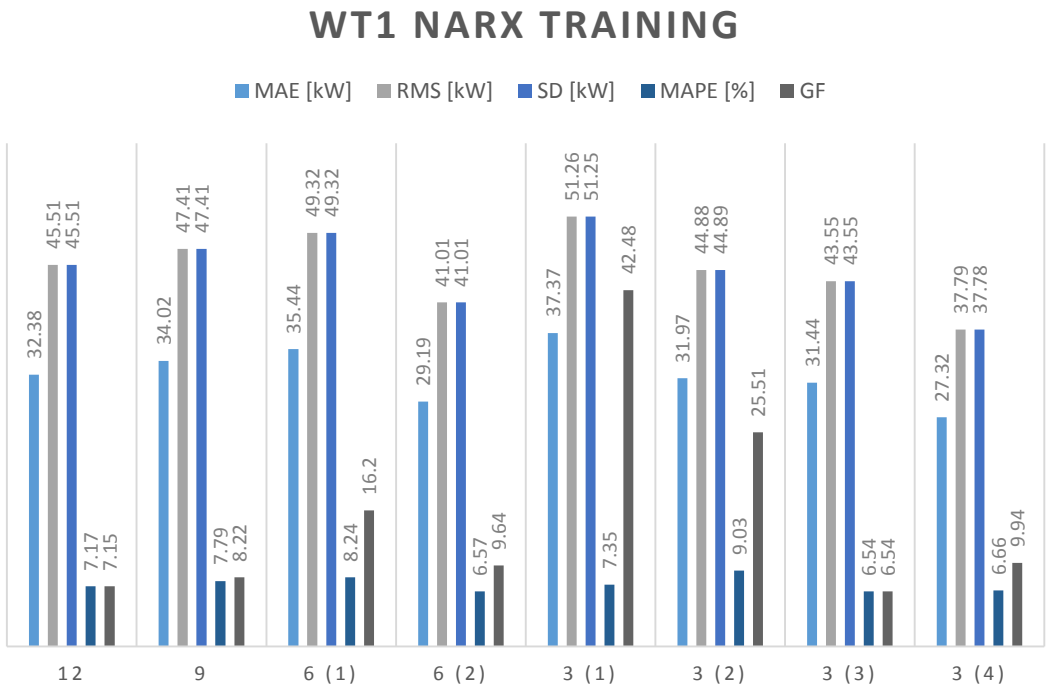


Figure 6.26 Bar chart showing the results from the training of the different models based on a NARX network.

In Figure 6.26 the results from the different models are presented, the most remarkable observation in these results is that model 3 (3) and 3 (4) shows very good performance, despite the fact that the training set is very small. This is because the validation is done with parts of the training set, which in that small period is very similar to the test set, which is very obvious when looking in Figure 6.27.

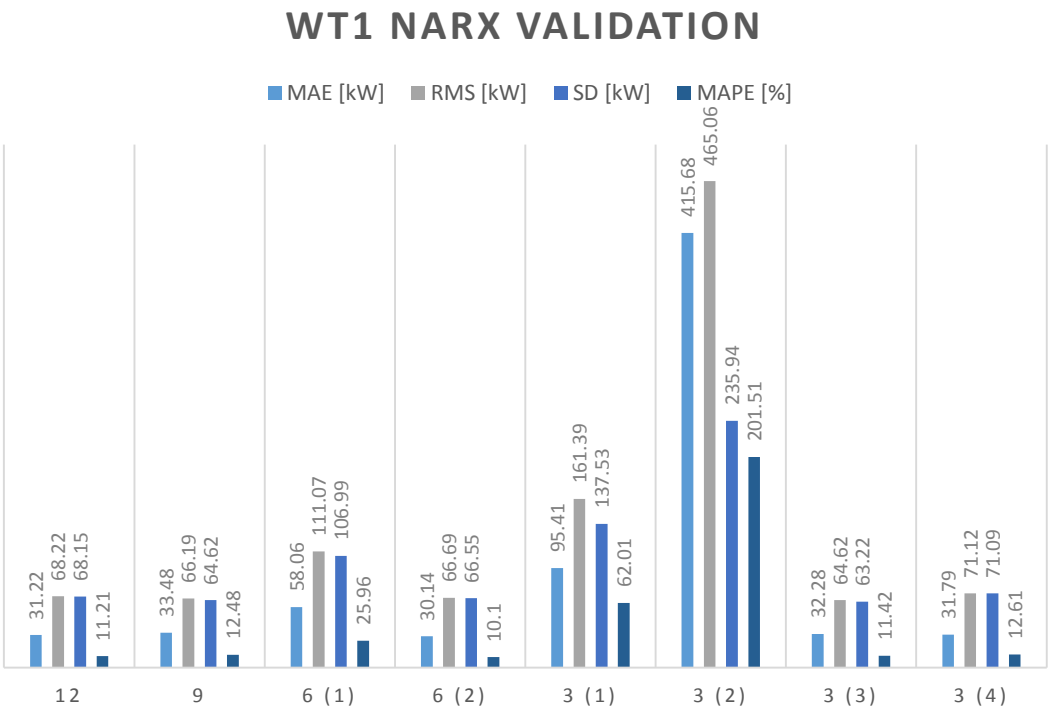


Figure 6.27 Bar chart showing the results from the different NARX models based on unseen data from WT1.

In Figure 6.27 the models are tested on an unseen validation set ranging from 2013-07-13 to 2013-09-13. Here we can see that the performance is very poor for model 3 (1) and 3 (2), which do not have data which is representative for the validation period. But it is also important to notice that model 3 (3) and 3 (4) performs quite well during the period, indicating that even a model based on a short data set could be adequate, given it is based on the same external conditions.

6.4.2 Feed-forward network with one hidden layer

Three different kinds of networks were tested in this study, the first being the single hidden layer feed forward network.

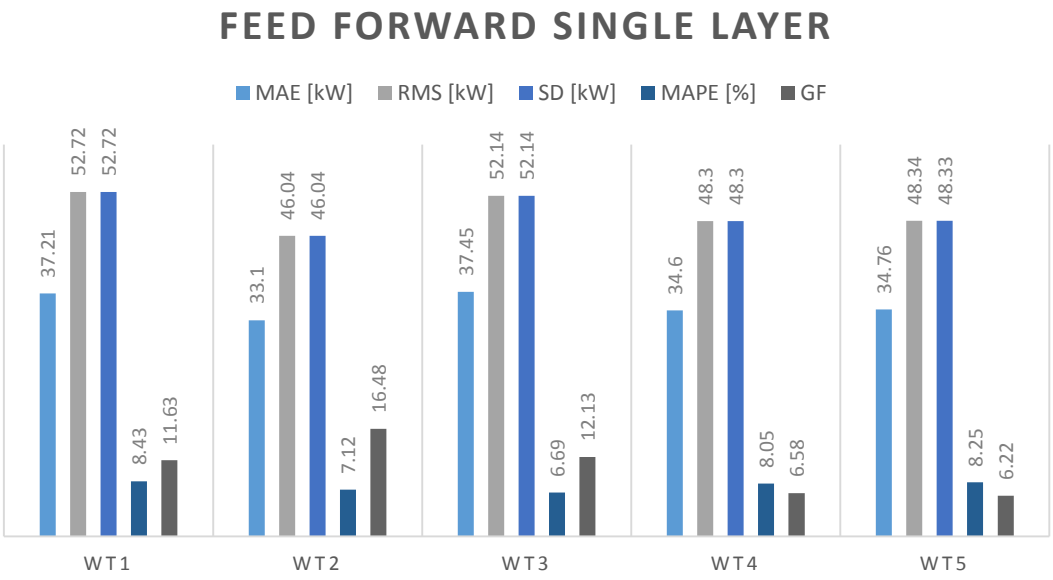


Figure 6.28 Model results from all turbines using a single layer feed forward network.

In Figure 6.28 the results from the modelling of all turbines are shown. The results are fairly similar over all turbines, with generalisation factor for WT2 being the only real deviating result. The order of the MAE from highest to lowest is WT3, WT1, WT5, WT4 and lastly WT2.

6.4.3 Feed-forward network with two hidden layers

The second network studied is the dual hidden layer feed forward network.

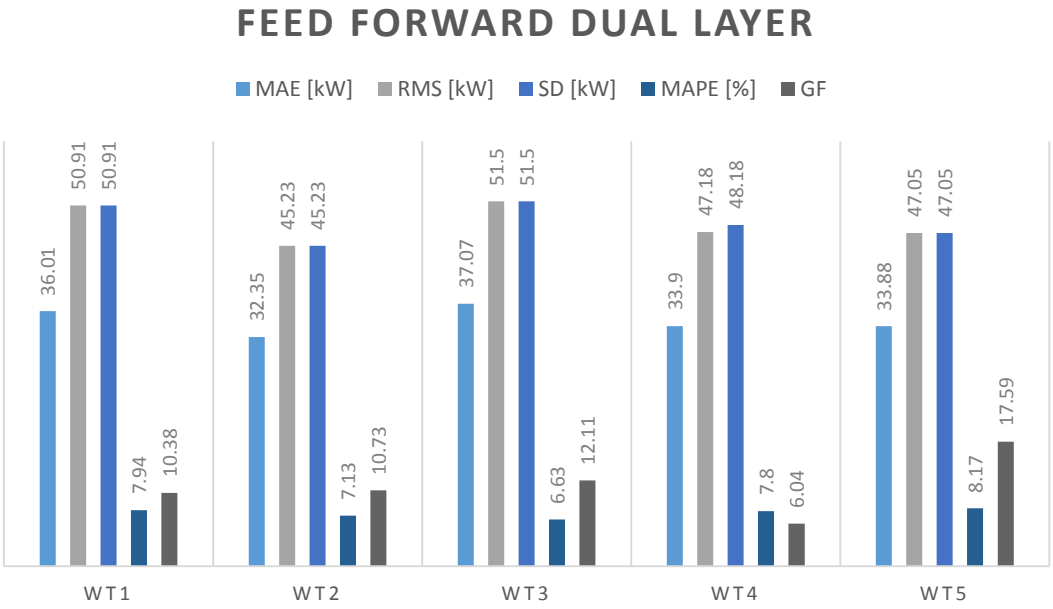


Figure 6.29 Model results from the dual layer feed forward network.

Figure 6.29 shows the same general behaviour as the single layer network, but with a lower GF for WT2 and a higher for WT5.

6.4.4 NARX network with one layer

The third kind of network studied is the NARX, which has proved to be the absolute best network for this application.

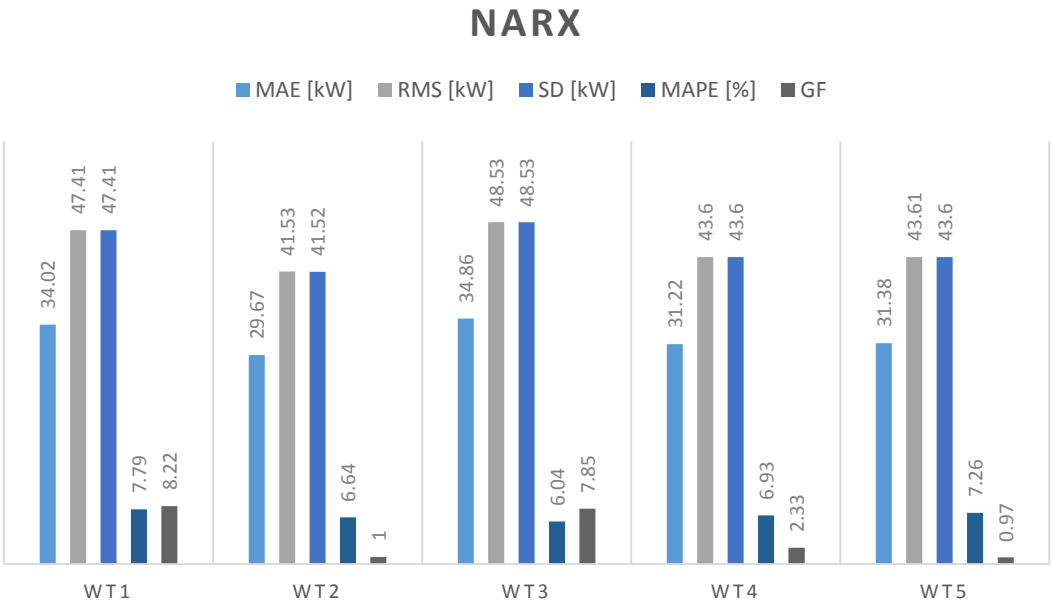


Figure 6.30 Model results from all wind turbines using the NARX network.

In Figure 6.30 the same MAE behaviour is observed as in the previous networks, but the GF is considerably lower especially for WT2, WT4 and WT5. The improvement in average MAE for all turbines from the worst performing network (FF1) to the best performing network (NARX), is 9%.

6.4.5 Additional tests

All the following tests are using nine months data from WT1 and are performed to investigate the importance of the extra parameters included in this study, and evaluate the impact of the filtering method developed.

6.4.5.1 Parameters used in comparative study

In order to evaluate if it is reasonable to compare the models in this study to the one made in the comparative study, a performance test using the same inputs as in that study was made. The single hidden layer neural network model used in the reference study is reported to have a MAE and MAPE of 1.62 kW and 8.38 %, respectively. The results from the single layer in this study, using the same inputs, are 1.72 kW and 9.63 %. These results are deemed to be close enough to make a comparison.

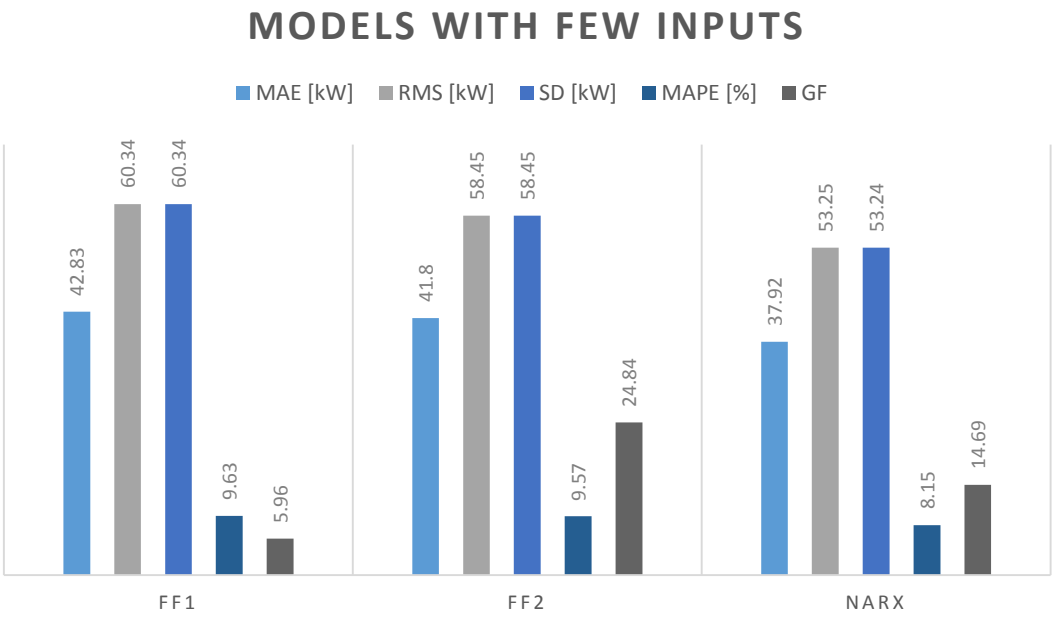


Figure 6.31 Model results using the same input parameters as in the comparative study.

Figure 6.31 shows the clear trend of increasing performance going from FF1 to NARX, the relatively high GF in FF2 is still not alarmingly high and a good generalisation should still be achieved.

6.4.5.2 Excluding system frequency as input

To assess whether the inclusion of system frequency is warranted, tests for all models were done.

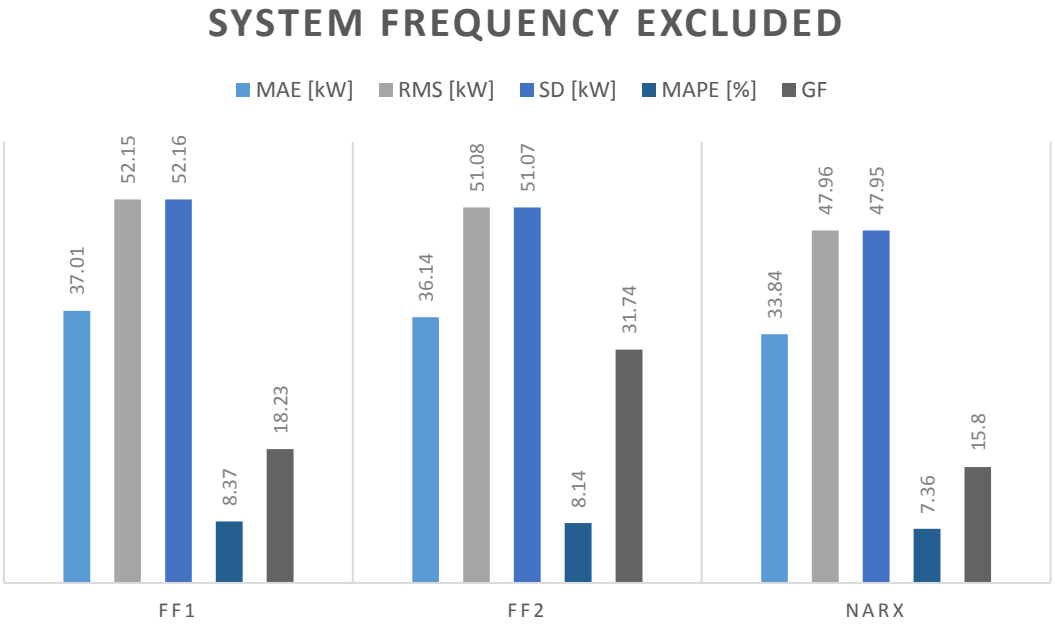


Figure 6.32 Model results from a test with the system frequency excluded as input.

Figure 6.32 shows the results from the models, excluding system frequency as input. Comparing the results to the main models, there seems to be no real improvement in MAE, where the results only differ with 0.01 kW, and could be considered the result of the inherent randomness in training an ANN. The MAPE are all in the same range, but there is a considerable increase in GF. This does not have to mean that it is worse in generalisation though, since a standard deviation of 20 – 30 kW still is low compared to the performance indicators which are the square of the error, normally in the range of 2000 – 3000 kW.

6.4.5.3 Excluding standard deviation as input

One of the main theories in the thesis is that the inclusion of wind speed standard deviation would increase the prediction performance of the ANN. To confirm this theory, a test excluding this parameter as an input was performed and the results are presented below in Figure 6.33.

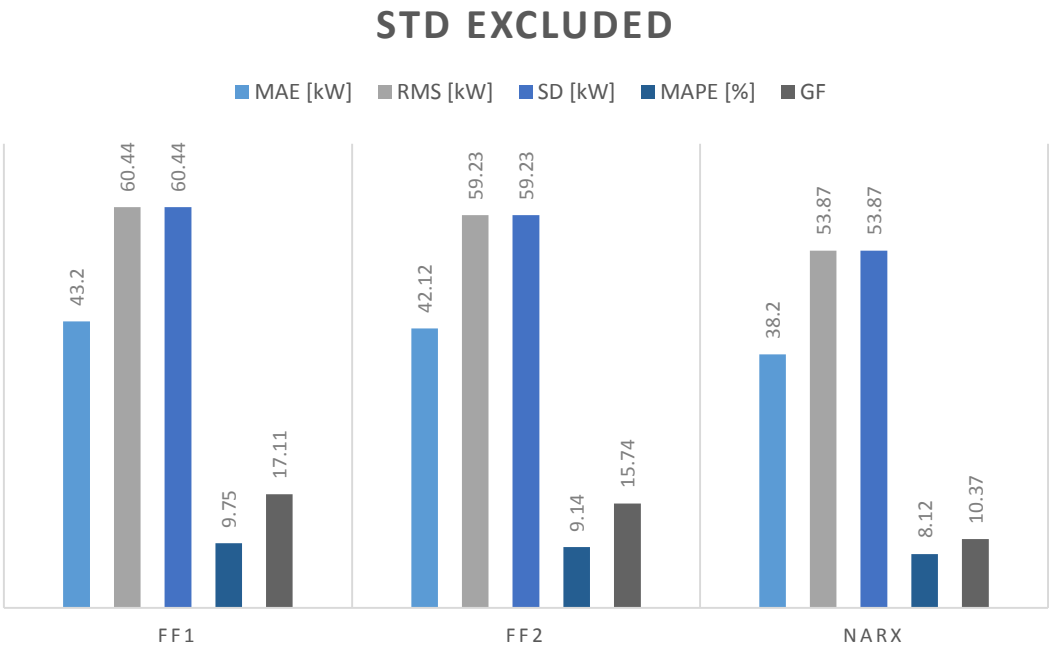


Figure 6.33 Model results excluding standard deviation as input

The results from the test shows a great decrease in performance compared to the original configuration. The NARX MAE has increased from 34.02 kW to 38.20 kW, which is a very significant performance degradation. Thus the selection of this parameter as an input has a considerable effect to the end result.

6.4.5.4 All parameters using LMedS filtering

To evaluate the impact of the developed filtering method, a test using a simplified method described in Section 5.4.6 is performed.

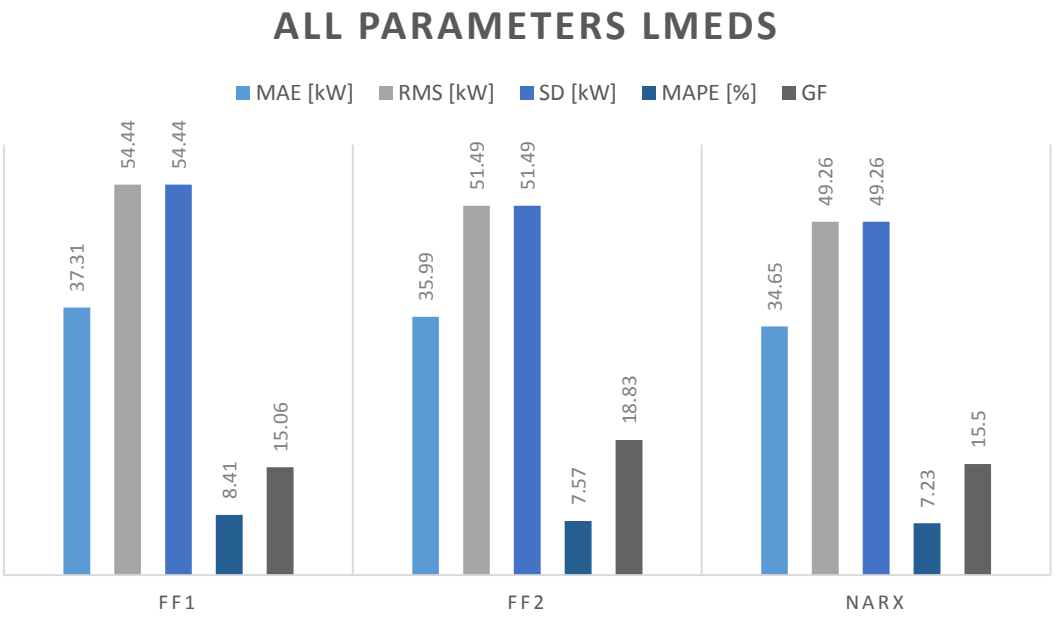


Figure 6.34 Results from a test using LMedS filtering instead of clustering with the chosen input parameters.

The MAE results are persistently worse for all trained networks using the simplified filter compared to the developed filter, though not by a large margin.

6.4.5.5 Parameters used in comparative study, LMedS filtering

A test was also performed with the alternative filter using the basic input parameters, to see if the results were persistent.

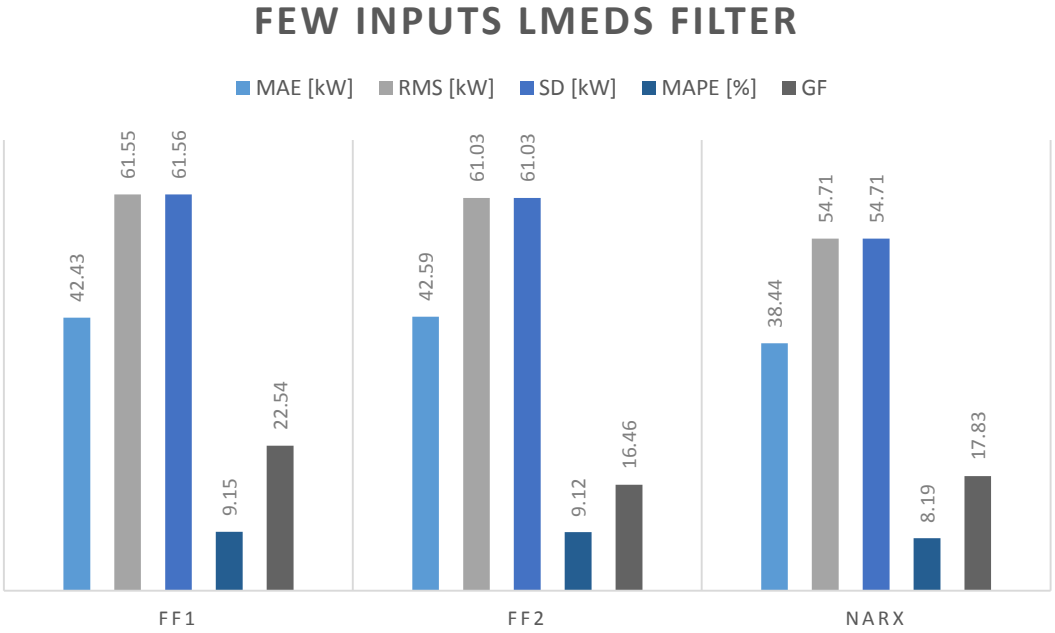


Figure 6.35 Results from a test using the input parameters from the comparative study, using LMedS filtering.

The results shown in Figure 6.35 compared to Figure 6.31, shows a small overall decrease in performance by using the simplified filter. This leads to the conclusion that the performance is better by using the developed filter, but a very marginal improvement.

6.4.6 Comparison to reference study

To be able to compare the results to the best performing model reported in the comparative study performed by (Schlechtingen, et al., 2013b), the model results from all turbines are averaged and scaled to 100 kW.

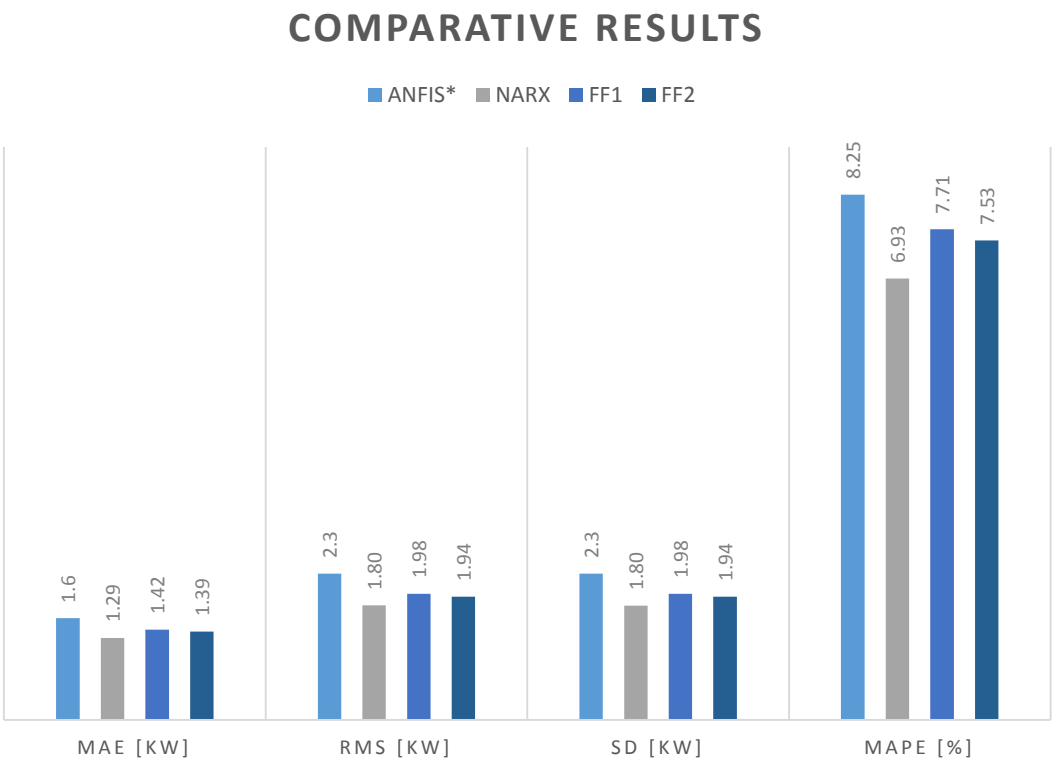


*Figure 6.36 The results from the comparison between the models developed in this study and the best constructed model in the comparative model. *Values reproduced from table in comparative study.*

The results of the assembled models shows very good results in comparison to the reference study. The best net found in this study, the NARX net, shows an improvement in MAE of 19.4 %, 21.7 % in RMS and SD, and 16 % in MAPE.

In Figure 6.37 below, a full summary of the MAE's and MAPE's from the different tests can be seen.

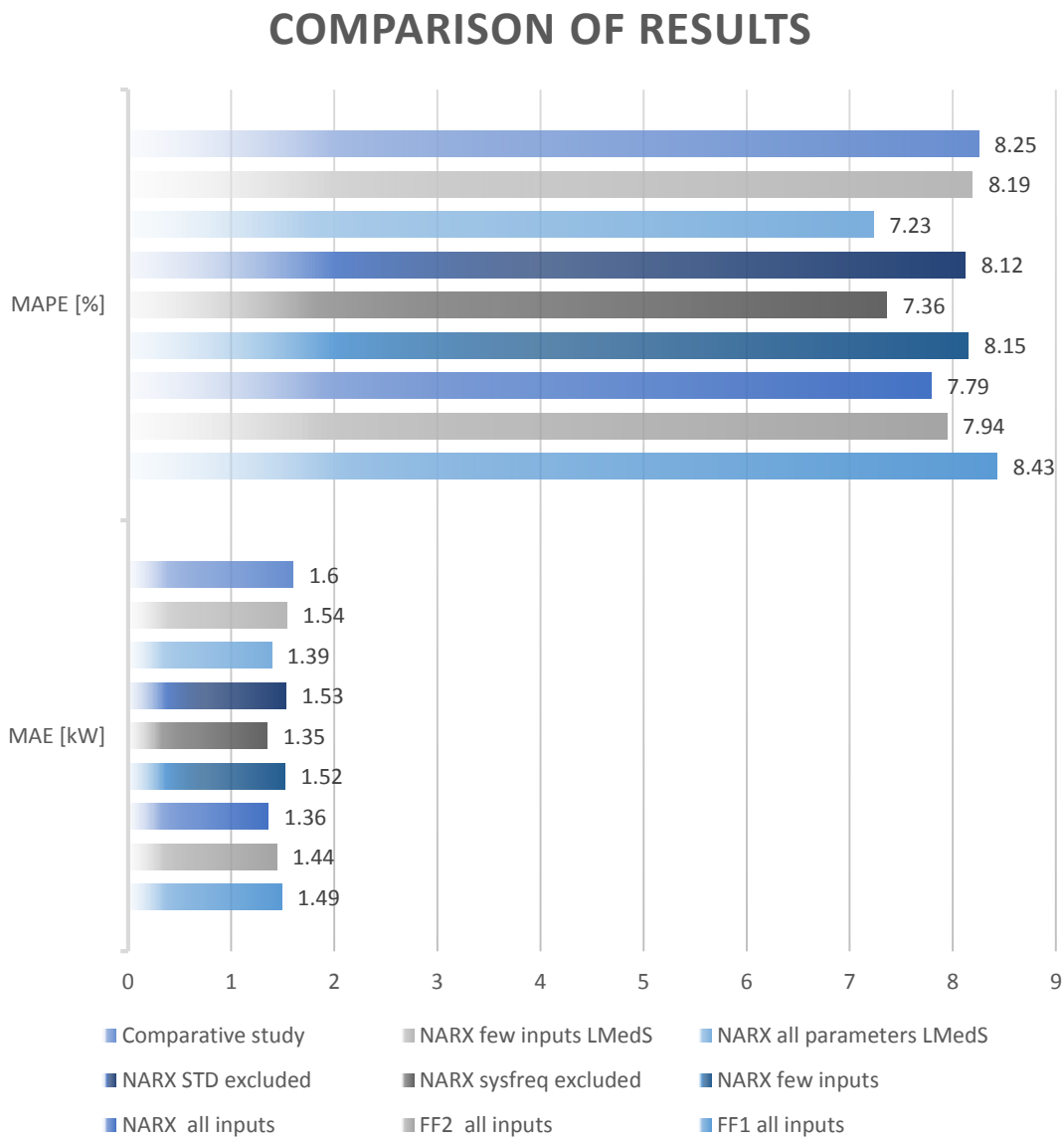
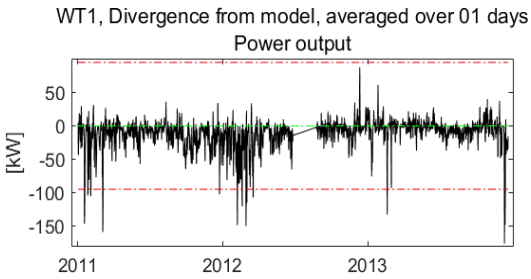


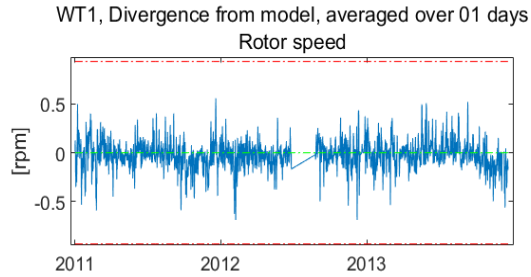
Figure 6.37 Full summary of the scaled Mean Average Error values from the different tests.

6.4.7 Output deviation detection

By studying the error between the actual output and the model response to the inputs, this thesis' aim is to find degradation in the turbine performance in an earlier stage. The power output is the main interest here, since this can be translated into loss of electrical production. To interpret the other outputs simply by looking at the errors is harder, but it can at least give an indication to why the power production is lower than expected.

The resulting plots seen below shows no seasonal behaviour, which means the nine-month model seems to have a wide enough training set to handle the change in temperature. The missing data in the middle of 2012 comes from a transformer failure.





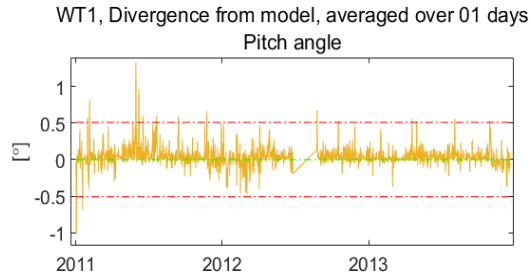
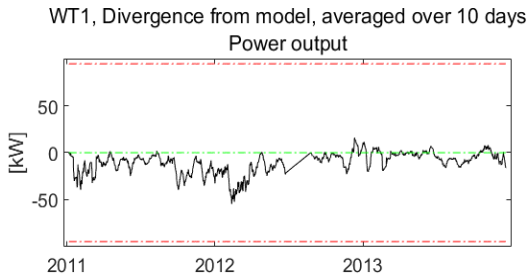
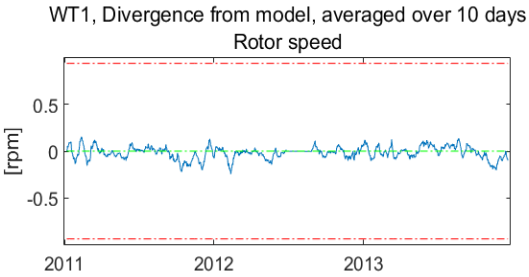


Figure 6.38 The errors between the actual and predicted outputs from the model averaged over a day. If a curve is under the green dashed line, the model predicts the output to be higher. The red line indicates the normal fault range.





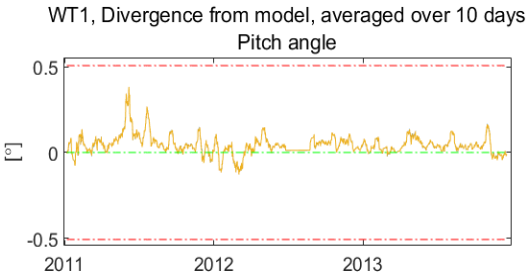


Figure 6.39 In this figure, the 10 day average of the errors are plotted. Using a larger time span means that persistent deviation is easier to detect.

When looking at Figure 6.38, it is clear that even with an averaging of one day, the trends can be hard to discern. There are a few instances where the daily averaged power production is lower than what can be considered a normal deviation from the model. The rotor speed mainly follows the normal behaviour without any larger deviations from the actual output, no real conclusions can be drawn from this output. When it comes to the pitch angle, there are a number of days where the pitch angle has deviated from normal behaviour, especially in the middle of 2011.

In Figure 6.39, it is easier to see the long term trends and the pitch angle peaks in mid-2011 coincides with the dips in power production, indicating the reason for this instance of underproduction might be a faulty pitch control. The constant underproduction from August 2011 to June 2012 does not seem to be clearly correlated to either of the other outputs, but a result of a fault in another part of the turbine.

In Figure 6.40 below, a close up of the area pointed out in Figure 5.8 is presented. The marking shows a group of samples where the turbine seems to have performed badly over a time.

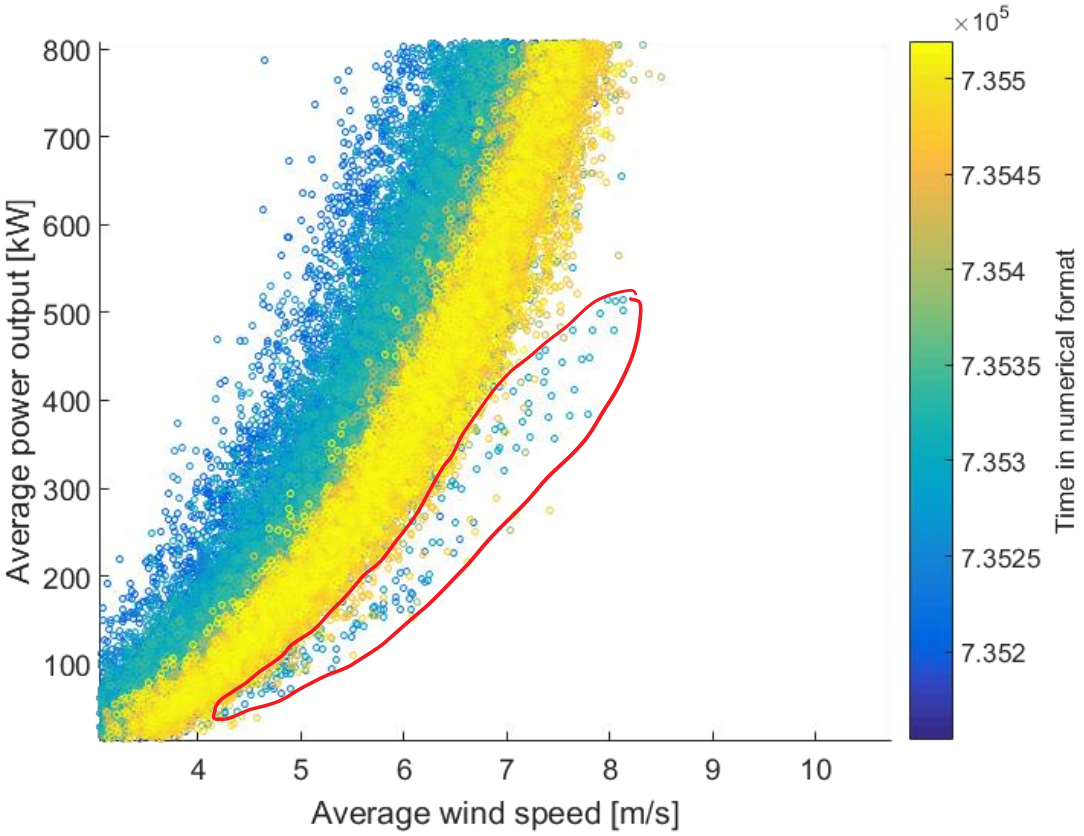


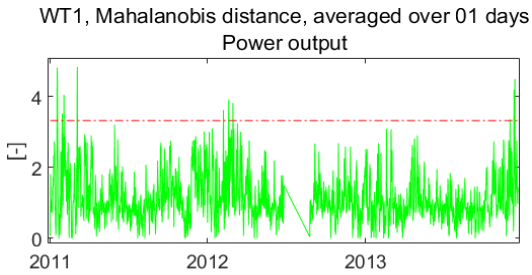
Figure 6.40 A close up of the wind speed - power plot, with red markings showing an interesting area where the turbine seems to perform badly.

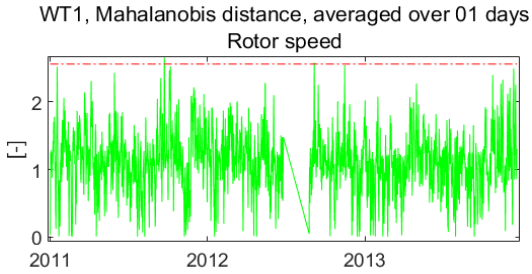
The timestamps of the samples show that they belong to a period around February 2012, which coincides with the decreased performance which is very clear in Figure 6.39.

If the daily average underproduction is summed up over the period August 2011 to June 2012, where there is a clear trend of underperformance, it adds up to a loss of production of 170 MWh, a considerable amount. The sum for the total period investigated is 268 MWh.

6.4.8 Fault detection

An additional aim of the study was to investigate whether the proposed model could be used to predict faults. Other studies where the main aim is to predict faults normally focus on a single subsystem of the wind turbine, mainly the generator, a bearing or the gearbox. A method to set a normal range for the Mahalanobis distance has been used in this thesis to identify possible faults based on the three outputs modelled.





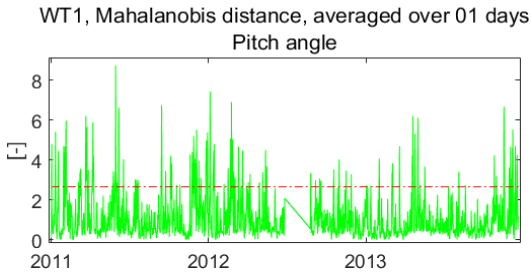
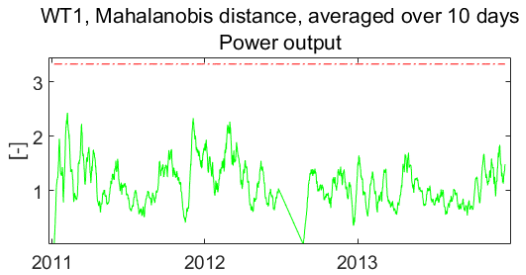
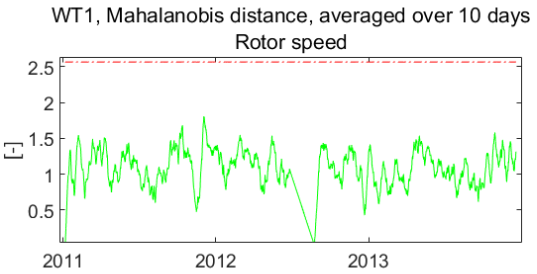


Figure 6.41 The Mahalanobis distance averaged over a day. The advantage of using Mahalanobis distance in the fault prediction is that the error is connected to the current output, which means faults can be recognised even if the divergence model lies within normal range.





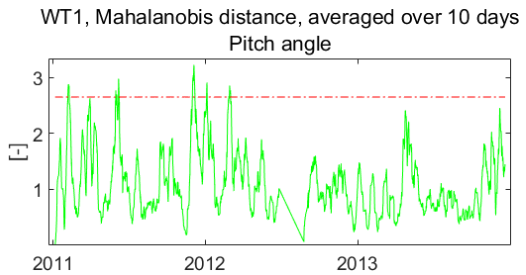


Figure 6.42 In this figure, the 10 day average of the Mahalanobis distance is plotted. Only the pitch angle seems to have a persistent error lasting as long as ten days.

In Figure 6.41, multiple violations of the Mahalanobis threshold can be seen during the three years plotted. Especially the pitch angle shows a very erratic behaviour, which might stem from that the pitch control have a hard time following the highly varying speeds at high wind speeds. Looking at Figure 5.6, it can be seen that the average wind speed at the end of 2011 indeed were high.

Something interesting to notice is when comparing the Mahalanobis distance for rotor speed and power output, is that the curves generally follow the same trend, but the threshold violations do not occur at the same time period. This indicates that the use of two different outputs might be warranted for, to be able to find faulty behaviour that might not be reflected in the power output error plot.

By comparing the pitch angle for one day average in Figure 6.41 to the plot in Figure 6.42 shows that pitch angle might be more interesting to study over a wider time span, because of its erratic behaviour at high speeds. By looking at a time span of ten days, long-term faulty behaviour can be detected, and short term violations discarded.

There has been one big failure each in WT2 and WT3 with long periods of downtime as a result, with reason being the transformer laminations were degraded. To predict this kind of fault is hard, since these kind of failures most of the time are instantaneous.

The reason for a failure like this though, could be vibrations, leading to wear on the insulation of the windings. In WT3, a rather large deviation from model in rotor speed can be seen for about one month prior to the fault (Figure 6.43).

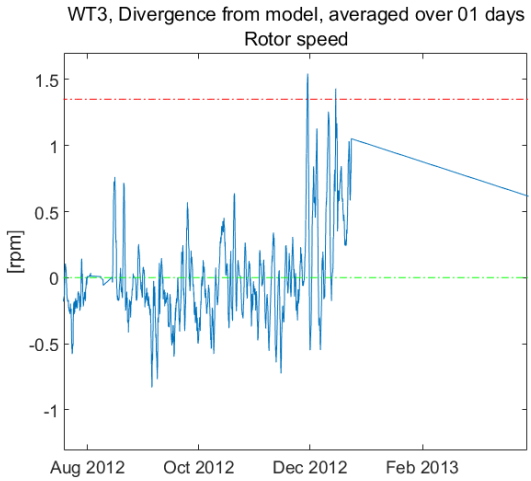


Figure 6.43 Rotor speed divergence from model. The first peak crosses the normal range November 30th.

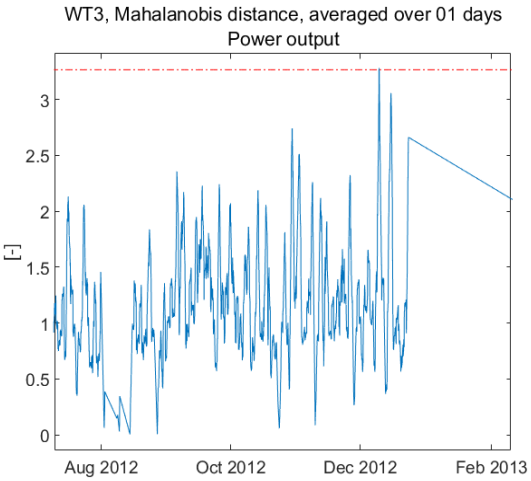


Figure 6.44 Mahalanobis distance rotor speed. The average Mahalanobis distance exceeds the threshold on December 15th and stays above it for almost 16 hours.

In the figures above, the behaviour of the model in terms of rotor speed can be seen for the period before the failure. The rotor speed seems to lie above the expected for almost a month prior to the failure. The average Mahalanobis distance crosses the threshold one week before the failure.

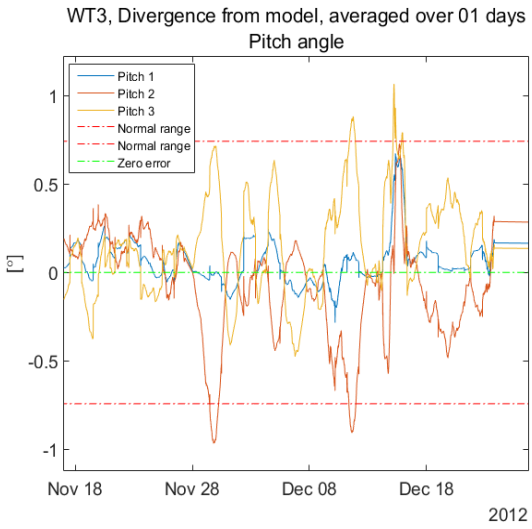


Figure 6.45 Pitch angle divergence from model.

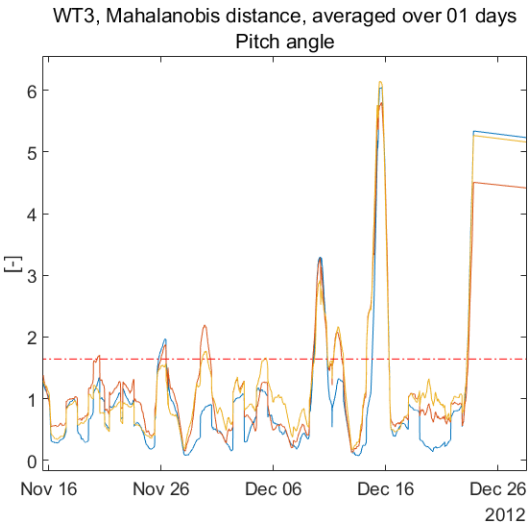


Figure 6.46 Pitch angle Mahalanobis distance.

In Figure 6.45 and Figure 6.46 above, the pitch angle behaviour is shown for a period closer to the failure. The divergence between Pitch 2 and Pitch 3 can be seen over the whole timespan for which data is available. The deviation between pitch angles looks suspicious in Figure 6.45, as the pitch angles diverges in an inverted manner, indicating a possible fault in the controller. In worst case, this could be the reason for the vibrations in the turbine that caused the failure.

One could argue that this is a fault in the model output, which is possible, but the model is trained in the same way as WT2, where the response of the different pitch angles can be seen to follow each other nicely in Figure 6.47.

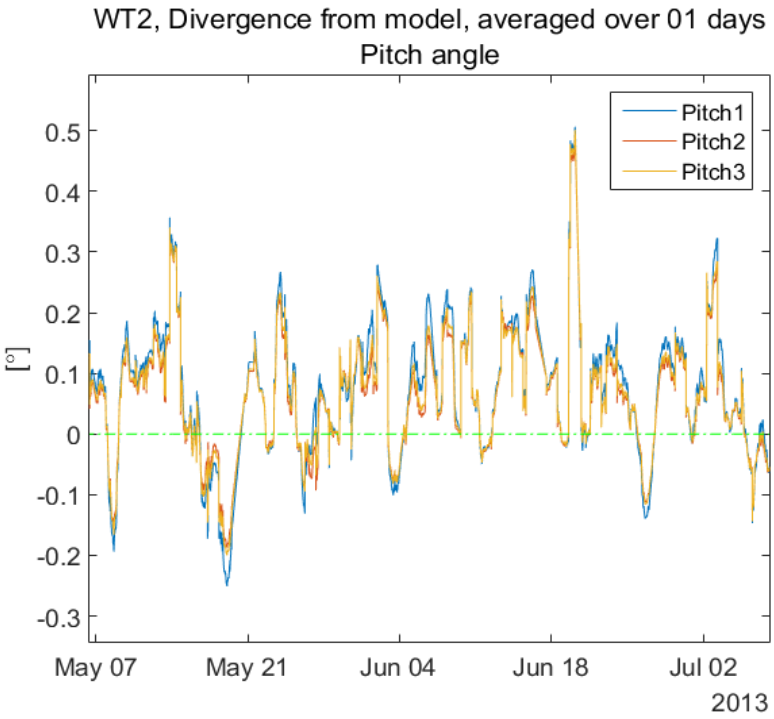


Figure 6.47 Pitch angle divergence for WT2.

7 Conclusions

The following chapter holds the concluding discussion and proposed future work.

7.1 Discussion

In this thesis, an automated cluster-based filtering method has been developed. The results show that it works well even with a small amount of data and is capable of excluding points that diverge from the pattern of the normal data. The method created works in up to eight dimensions, which makes it very hard to grasp, graphically. But the mathematical theory behind the model includes the use of high dimensions, so even though it can be hard to imagine the Mahalanobis distance when going above the normal three dimensions, so the theory should hold.

There are certain things that was too time consuming to do in this project, such as finding the exact right thresholds and number of clusters for the filtering model. So there is indeed room for improvement, and it cannot be said for sure that the best possible results were achieved here.

The actual results if compared to the other considered method, LMedS, were very modest if only comparing the performance parameters used to quantify the results. In order to really estimate the gain of this filtering, more tests would have had to be done, using other parameters than the ones used in this study. But since the main focus of the study was not in this area, there was not time to develop tests better suited for this estimation.

Standard deviation as an additional input proved to be a very large contributor to the good results in this study. On the other hand, the use of system frequency showed no improvement at all, concluding that this parameter is not suitable to use as an input. The additional outputs in form of rotor speed and pitch angle gives some extra information in the interpretation of the power output curve, but no real conclusions based on the outputs alone can be made.

The second large contributor to the good results is the use of a NARX net instead of the feed forward network normally used in other studies. This is probably due to the fact that the conditions in one time set most of the times looks very similar to the previous, and since the trends of the wind normally is much longer than the five minutes used for sampling, the use of the previous result could indeed improve the result.

A downside with the NARX net is that it depends on having the result from the previous sample in order to work well. Since there is limitations to the model meaning that some samples are deleted since they are out of the range for the model, this means that the value following a deleted sample will have a rather large error, worsening the performance of the model. As mentioned earlier, this drawback could be eased by expanding the model to be able to interpret data deemed as out-of-bounds in this study, by including extra parameters found in SCADA which indicate manually set limitations and known faults. There are also algorithms that uses the networks trained response to fill in the missing samples with an approximated value, which makes the model response much less erratic.

One of the main arguments against ANN’s is that they need a wide range of data to be able to generalise properly. This was very obvious when looking at the model response of those models using three months data, when validated against a period in another

season. But at the same time, the models using only three months data based on samples close to the validation set, performed very well. So if there is a need for retraining the model after the replacement of a vital part or an optimisation that has changed the behaviour of the turbine, the model can be trained gradually with the new data accumulated and still retain a good performance for monitoring.

Based on the results in Section 6.4.7, the output deviation model seems to work well and the use of this method could have meant that the performance degradation seen in WT1 could have been found at an early stage. This in turn would mean that maintenance could have been ordered at an early stage, leading to increased electricity production and therefore revenues, by a considerable amount.

It is hard to say anything decisive when it comes to the fault detection performance of the model. The thought was to set a threshold, where the crossing of this would indicate a coming failure. But the service records before 2013 for the turbines were unavailable and there has not been any faults that could be directly detected with the used outputs, which makes it hard to say anything conclusive in this matter. One might however, use it as a supplement to the normal deviation model, since it is more sensitive due to the use of the Mahalanobis distance.

7.2 Continued work

With such an open project description, a lot of time was used to research the field and find a subject which had a new angle, also the data extraction was very time consuming. The loss of time and the relatively wide subject led to that some of the new concepts developed in this thesis were not investigated to the fullest. A few suggestions for future work are listed below:

Model

To enhance the model even further, a thorough investigation of all parameters reported in SCADA should be done. There is for example a manual curtailment indicator and a value of the set value which should work well with the model, if it has this parameter as an input. To incorporate the algorithm to use the model output to fill in missing data in addition to the previous suggestion would make the model output more exact, and valid for all data.

Filter

The filter designed in this study has good potential to be a very good tool for filtering the training data from erroneous patterns. To refine this filter by analysing the parameters used for clustering and the thresholds for the filter could make it even better and enhancing the models prediction.

Model inputs and outputs

The two additional outputs used in this thesis was mainly incorporated to test the ANN's ability to handle multiple outputs and see if they could be used to detect faults in the turbine. There are a large set of sensors in a modern wind turbine, and there could be a lot of other parameters that are closely dependant on the independent inputs used, which

could be of interest to monitor. The system frequency as an additional input proved to be insignificant, but there are other inputs to be considered, such as air pressure and humidity. These parameters were present in the SCADA system, but with no recorded values, so the sensors were probably not installed or not in use.

8 References

Bangalore, P., 2014. *Load and Risk Based Maintenance Management of Wind Turbines,* Gothenburg: Chalmers Bibliotek, Reproservice.

Belghazi, O. & Cherkaoui, M., 2012. PITCH ANGLE CONTROL FOR VARIABLE SPEED WIND TURBINES USING GENETIC ALGORITHM CONTROLLER. *Journal of Theoretical and Applied Information Technology,* 39(1), pp. 6-10.

Besnard, F., Nilsson, J. & Bertling, L., 2010. *On the Economic Benefits of using Condition Monitoring Systems for Maintenance Management of Wind Power Systems.* Singapore, IEEE.

Choi, N. J., Nam, S. H., Jeong, H. J. & Kim, K. C., 2014. CFD Study on Aerodynamic Power Output Changes with Inter-Turbine Spacing Variation for a 6 MW Offshore Wind Farm. *Energies,* 7(11), pp. 7483-7498.

De Maesschalck, R., Jouan-Rimbaud, D. & Massart, D. L., 2000. The Mahalanobis Distance. *Chemometrics and Intelligent Laboratory Systems,* 50(1), pp. 1-18.

Demuth, H. & Beale, M., 2002. *Neural Network Toolbox For Use with MATLAB,* Natick: The MathWorks.

Dosaev, M. Z., Klimina, L. A., Lokshin, B. Y. & Selyutskiy, Y. D., 2014. On Wind Turbine Blade Design Optimization. *Journal of Computer and Systems Sciences International,* 53(3), pp. 402-209.

Energinet.dk, 2015. *Annual report 2014,* Fredericia: Energinet.dk.

EWEC, 2015. *WIND IN POWER: 2014 EUROPEAN STATISTICS,* -: THE EUROPEAN WIND ENERGY ASSOCIATION.

Festo Didactic, 2011. *Principles of Doubly-Fed Induction Generators (DFIG),* Quebec: Festo Didactic.

Friis Pedersen, T. et al., 2002. *Wind Turbine Power Performance Verification in Complex Terrain and Wind Farms,* Roskilde: Risø National Laboratory.

Global Wind Energy Council, 2014. *Global statistics.* [Online] Available at: http://www.gwec.net/global-figures/graphs/ [Accessed 30 05 2015].

Global Wind Energy Council, 2014. *GLOBAL WIND ENERGY OUTLOOK 2014,* Brussels: GLOBAL WIND ENERGY COUNCIL.

Grübler, A., 1998. *Technology and global change.* Cambridge: Cambridge Univ. Press.

Guezuraga, B., Zauner, R. & Pölz, W., 2012. Life cycle assessment of two different 2 MW class wind turbines. *Renewable Energy,* 37(1), pp. 37-44.

Hameed, Z. et al., 2009. Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable and Sustainable Energy Reviews,* 13(1), pp. 1-39.

Haykin, S., 2009. *Neural Networks and Learing Machines.* 3rd ed. New Jersey: Pearson Education.

Hockley, C., 2013. *Wind turbine maintenance and topical research questions.* Shrivenham, Elsevier B.V..

Honrubia, A., Vigueras-Rodríguez, A. & Gómez-Lazaro, E., 2012. *The Influence of Turbulence and Vertical Wind Profile in Wind Turbine Power Curve.* Bertinoro, Springer-Verlag.

Jenab, K., Rashidi, K. & Moslehpour, S., 2013. An Intelligence-Based Model for Condition Monitoring Using Artificial Neural Networks. *International Journal of Enterprise Information Systems,* 9(4), pp. 43-62.

Kayikçi, M. & Milanovic, J. V., 2009. Dynamic Contribution of DFIG-Based Wind Plants to System Frequency Disturbances. *IEEE TRANSACTIONS ON POWER SYSTEMS,* 24(2), pp. 859-.

Kusiak, A. & Verma, A., 2013. Monitoring Wind Farms With Performance Curves. *IEEE TRANSACTIONS ON SUSTAINABLE ENERGY,* 4(1), pp. 192-199.

Lapira, E. et al., 2012. Wind turbine performance assessment using multi-regime modeling approach. *Renewable Energy,* Volume 45, pp. 86-95.

Lydia, M., Immanuel Selvakumar, A., Suresh Kumar, S. & Edwin Prem Kumar, G., 2013. Advanced Algorithms for Wind Turbine Power Curve Modeling. *IEEE TRANSACTIONS ON SUSTAINABLE ENERGY,* 4(3), pp. 827-835.

Meyer, N. I., 1995. Danish wind power development. *Energy for Sustainable Development,* 2(1), pp. 18-25.

Mohr, S. H. & Evans, G. M., 2009. Forecasting coal production until 2100. *Fuel,* 88(11), pp. 2059-2067.

Murthy, Y. N., 2013. A REVIEW ON POWER ELECTRONICS APPLICATION ON WIND. *International Journal of Research in Engineering and Technology,* 2(11), pp. 360-376.

National Oceanic & Atmospheric Administration, 2015. *Trends in Atmospheric Carbon Dioxide.* [Online] Available at: www.esrl.noaa.gov/gmd/ccgg/trends/ [Accessed 30 05 2015].

Paipetis, S. A. & Ceccarelli, M., 2010. *The Genius of Archimedes -- 23 Centuries of Influence on Mathematics, Science and Engineering.* Syracuse, Springer Netherlands.

Paiva, L. T., Veiga Rodrigues, C. & Palma, J. M. L. M., 2014. Determining wind turbine power curves based on operating conditions. *Wind Energy,* 17(10), pp. 1563-1575.

Ponting, C., 2007. *A new green history.* Revised ed. London: Vintage.

Price, T., 2008. James Blyth – Britain's first modern wind power pioneer. *Wind Engineering,* 29(3), pp. 191-200.

Purvins, A., Papaioannou, I. T., Oleinikova, I. & Tzimasa, E., 2012. Effects of variable renewable power on a country-scale electricity system: High penetration of hydro power plants and wind farms in electricity generation. *Energy,* 43(1), pp. 225-236.

Rhodes, S. L. & Wheeler, S. E., 1996. Rural Electrification and Irrigation in the U.S. High Plains. *Journal of Rural Studie,* 12(3), pp. 311-317.

Rice, J. A., 2007. *Mathematical Statistics and Data Analysis.* 3rd ed. Belmont: Thomson.

Sainz, E., Llombart, A. & Guerrero, J. J., 2009. Robust filtering for the characterization of wind turbines: Improving its operation and maintenance. *Energy Conversion and Management,* 50(9), pp. 2136-2147.

Schlechtingen, M. & Ferreira Santos, I., 2010. Comparative analysis of neural network and regression based condition monitoring approaches for wind turbine fault detection. *Mechanical Systems and Signal Processing,* 25(5), pp. 1849-1875.

Schlechtingen, M., Ferreira Santos, I. & Achiche, S., 2013a. Using Data-Mining Approaches for Wind Turbine Power Curve Monitoring: A Comparative Study. *IEEE Transactions on Sustainable Energy,* 4(3), pp. 671-679.

Schlechtingen, M., Ferreira Santos, I. & Achiche, S., 2013b. Wind turbine condition monitoring based on SCADA data using normal behavior models. Part 1: System description. *Applied Soft Computing,* 13(1), pp. 259-270.

Sulzberger, C., 2009. A bold effort in Vermont. *IEEE Power and Energy Magazine,* 7(6), pp. 100-110.

Tchakoua, P. et al., 2014. Wind Turbine Condition Monitoring: State-of-the-Art Review, New Trends, and Future Challenges. *Energies,* 7(4), pp. 2595-2630.

Thirumalai, D. P. R., 2014. A critical review of future materials for wind turbine blades. *International Journal of Materials Engineering Innovation,* 5(2), pp. 81-99.

Wan, Y.-H., Ela, E. & Orwig, K., 2010. *Development of an Equivalent Wind Plant Power-Curve.* Dallas, National Renewable Energy Laboratory.

Wizelius, T., 2007. *Vindkraft i teori och praktik.* 2 red. Lund: Studentlitteratur.

Yu, D. O. & Kwon, O. J., 2014. Predicting wind turbine blade loads and aeroelastic response using a coupled CFD–CSD method. *Renewable Energy,* Volume 70, pp. 184-196.

9 Appendices

9.1 Model performance

9.1.1 WT1 NARX network results for different data lengths

Table 9.1 Results from WT1 based on the training sets validation part are listed for different constructed models.

Data length [months]	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
12	32.38	45.51	45.51	7.17	7.15
9	34.02	47.41	47.41	7.79	8.22
6 (1)	35.44	49.32	49.32	8.24	16.2
6 (2)	29.19	41.01	41.01	6.57	9.64
3 (1)	37.37	51.26	51.25	7.35	42.48
3 (2)	31.97	44.88	44.89	9.03	25.51
3 (3)	31.44	43.55	43.55	6.54	6.54
3 (4)	27.32	37.79	37.78	6.66	9.94

Table 9.2 The same results scaled to 100 kW.

Data length [months]	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]
12	1.3	1.82	1.82	442.9
9	1.36	1.9	1.9	481.1
6 (1)	1.42	1.97	1.97	212
6 (2)	1.17	1.64	1.64	244.3
3 (1)	1.49	2.05	2.05	99.2
3 (2)	1.28	1.8	1.8	45.2
3 (3)	1.26	1.74	1.74	70.1
3 (4)	1.09	1.51	1.51	29.1

Table 9.3 In this table, new values based on a two month period not included in the training set is presented. The results from the two first three month models are extremely bad, but the two last are almost as good as the twelve months model.

Data length [months]	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]
12	31.22	68.22	68.15	11.21
9	33.48	66.19	64.62	12.48
6 (1)	58.06	111.07	106.99	25.96
6 (2)	30.14	66.69	66.55	10.1
3 (1)	95.41	161.39	137.53	62.01
3 (2)	415.68	465.06	235.94	201.51
3 (3)	32.28	64.62	63.22	11.42
3 (4)	31.79	71.12	71.09	12.61

9.1.2 Feed-forward network with one hidden layer

Table 9.4 Results from the nine months models for each turbine, created with a single layer feed forward network.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
WT1	37.21	52.72	52.72	8.43	11.63
WT2	33.1	46.04	46.04	7.12	16.48
WT3	37.45	52.14	52.14	6.69	12.13
WT4	34.6	48.3	48.3	8.05	6.58
WT5	34.76	48.34	48.33	8.25	6.22

Table 9.5 The same results scaled to 100 kW.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]
WT1	1.49	2.11	2.11	235.9
WT2	1.32	1.84	1.84	282.7
WT3	1.5	2.09	2.09	207
WT4	1.38	1.93	1.93	122
WT5	1.39	1.93	1.93	204.9

9.1.3 Feed-forward network with two hidden layers

Table 9.6 The results from the dual layer feed forward network, nine months data.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
WT1	36.01	50.91	50.91	7.94	10.38
WT2	32.35	45.23	45.23	7.13	10.73
WT3	37.07	51.5	51.5	6.63	12.11
WT4	33.9	47.18	48.18	7.8	6.04
WT5	33.88	47.05	47.05	8.17	17.59

Table 9.7 The same results scaled to 100 kW.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]
WT1	1.44	2.04	2.04	135.5
WT2	1.29	1.81	1.81	197.1
WT3	1.48	2.06	2.06	145.9
WT4	1.36	1.89	1.89	272.1
WT5	1.36	1.88	1.88	340.7

9.1.4 NARX network with one layer

Table 9.8 NARX model results for nine months data for all turbines.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
WT1	34.02	47.41	47.41	7.79	8.22
WT2	29.67	41.53	41.52	6.64	1
WT3	34.86	48.53	48.53	6.04	7.85
WT4	31.22	43.6	43.6	6.93	2.33
WT5	31.38	43.61	43.6	7.26	0.97

Table 9.9 The same results scaled to 100 kW.

Turbine	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]
WT1	1.36	1.9	1.9	481.1
WT2	1.19	1.66	1.66	156.2
WT3	1.39	1.94	1.94	140.3
WT4	1.25	1.74	1.74	209.7
WT5	1.26	1.74	1.74	212.1

9.1.5 Parameters used in comparative study

Table 9.10 Shows the results from the different models using the same parameters as in the reference study.

Network	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
FF1	42.83	60.34	60.34	9.63	5.96
FF2	41.8	58.45	58.45	9.57	24.84
NARX	37.92	53.25	53.24	8.15	14.69

Table 9.11 The same values scaled to 100 kW.

Network	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]	neurons
FF1	1.72	2.41	2.41	272.7	20
FF2	1.67	2.34	2.34	125.2	13 - 8
NARX	1.52	2.13	2.13	160.3	20

9.1.6 Excluding system frequency as input

Table 9.12 The results from an iteration excluding the system frequency as an input.

Network	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GE
FF1	37.01	52.15	52.16	8.37	18.23
FF2	36.14	51.08	51.07	8.14	31.74
NARX	33.84	47.96	47.95	7.36	15.8

Table 9.13 Same results presented in previous table, but scaled to 100 kW.

Network	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]	neurons
FF1	1.48	2.09	2.09	270.3	20
FF2	1.45	2.04	2.04	172.5	13 - 8
NARX	1.35	1.92	1.92	249.7	20

9.1.7 Excluding standard deviation as input

Table 9.14 In this test, the standard deviation was excluded as an input.

Network	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
FF1	43.2	60.44	60.44	9.75	17.11
FF2	42.12	59.23	39.23	9.14	15.74
NARX	38.2	53.87	53.87	8.12	10.37

Table 9.15 The results scaled to 100 kW.

Network	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]	neurons
FF1	1.73	2.42	2.42	159.4	20
FF2	1.68	2.37	2.37	244.3	13 - 8
NARX	1.53	2.15	2.15	189.1	20

9.1.8 Original parameters using LMedS filtering

Table 9.16 In this trial, the LMedS filtering was used for the training data.

Network	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
FF1	37.31	54.44	54.44	8.41	15.06
FF2	35.99	51.49	51.49	7.57	18.83
NARX	34.65	49.26	49.26	7.23	15.5

Table 9.17 The results scaled to 100 kW.

Network	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]	neurons
FF1	1.49	2.18	2.18	110.1	20
FF2	1.44	2.06	2.06	345.1	13 - 8
NARX	1.39	1.97	1.97	256.8	20

9.1.9 Parameters used in comparative study, LMedS filtering

Table 9.18 In this trial, the parameters from the comparative study was used, together with LMedS filtering.

Network	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]	GF
FF1	42.43	61.55	61.56	9.15	22.54
FF2	42.59	61.03	61.03	9.12	16.46
NARX	38.44	54.71	54.71	8.19	17.83

Table 9.19 The results scaled to 100 kW.

Network	MAE [kW]	RMS [kW]	SD [kW]	runtime [s]	neurons
FF1	1.7	2.46	2.46	703.8	20
FF2	1.7	2.44	2.44	152.6	13 - 8
NARX	1.54	2.19	2.19	363.8	20

9.1.10 Results from the comparative study and the developed models

Table 9.20 The results in this table are reproduced from the comparative to be able to better evaluate the results in this study.

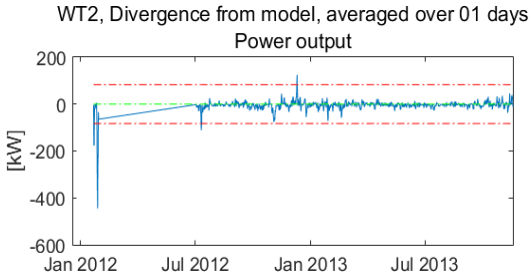
Method	MAE unscaled [kW]	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]
CCFL	33.53	1.68	2.42	2.41	8.91
NN	32.49	1.62	2.34	2.34	8.38
k-NN	85.35	4.27	6.02	5.96	25.48
ANFIS	32.01	1.6	2.3	2.3	8.25

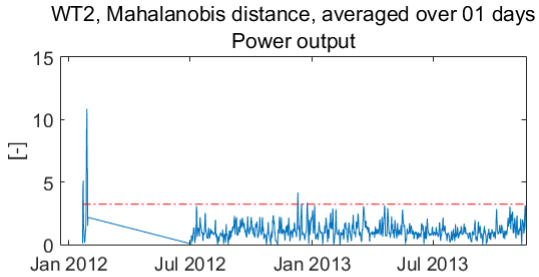
Table 9.21 The results presented in the same format as in the comparative study for easy comparison. Since the unscaled values in the comparative study comes from a 2 MW turbine, they are not directly comparable to the unscaled values in this table.

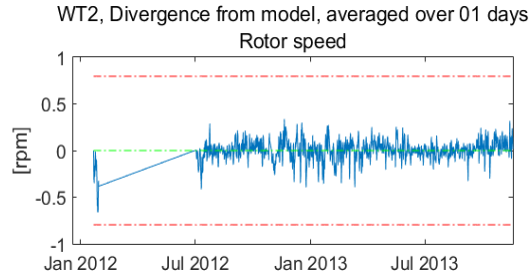
Network	MAE unscaled [kW]	MAE [kW]	RMS [kW]	SD [kW]	MAPE [%]
NARX	32.23	1.29	1.80	1.80	6.93
FF1	35.42	1.42	1.98	1.98	7.71
FF2	34.65	1.39	1.94	1.94	7.53

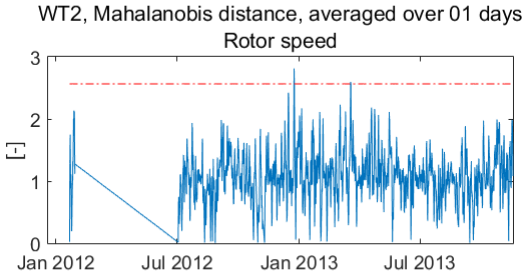
9.2 Results from remaining turbines

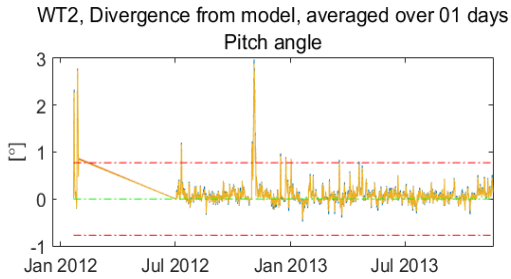
9.2.1 WT2











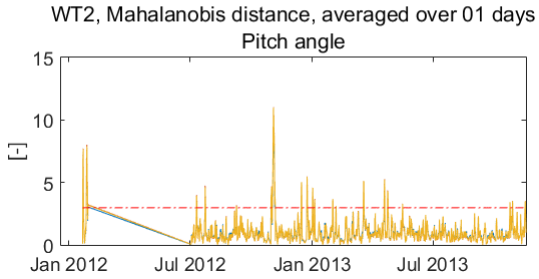


Figure 9.1 Error response WT2

Figure 9.2 Mahalanobis distance WT2

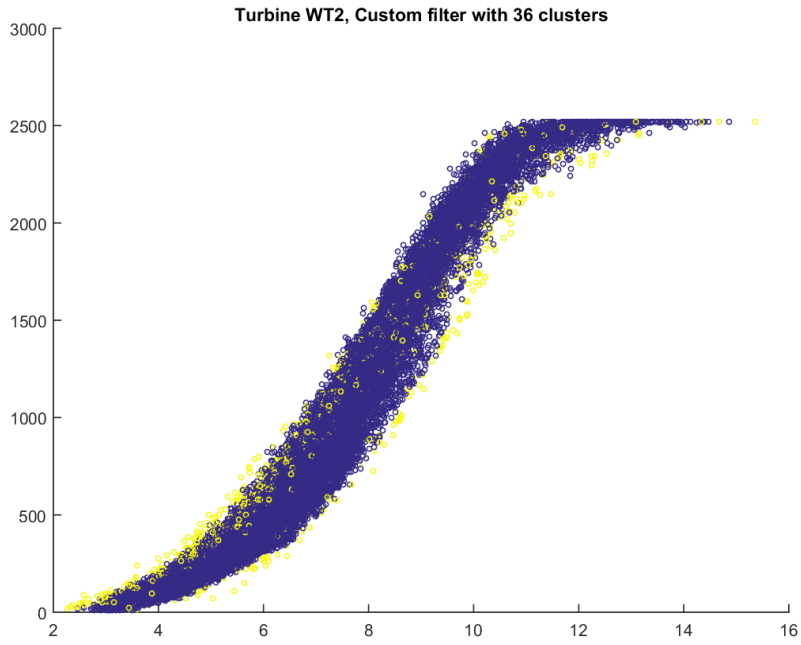


Figure 9.3 Filtering result WT2

9.2.2 WT3

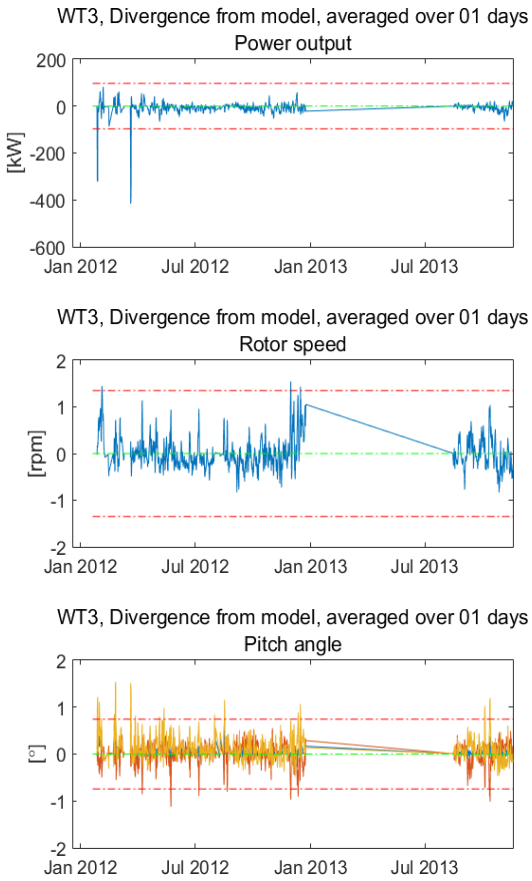


Figure 9.4 Error response WT3

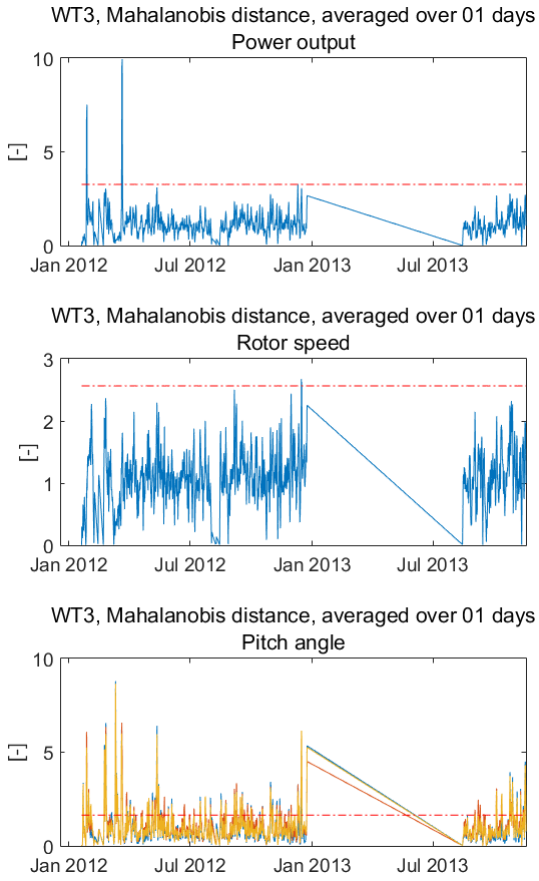


Figure 9.5Mahalanobis distance WT3

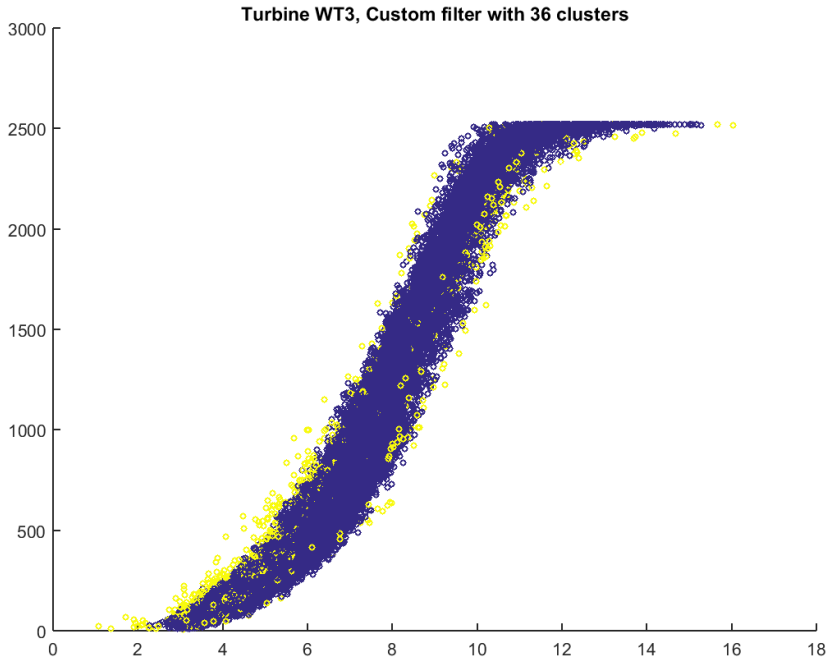


Figure 9.6 Filtering result WT3

9.2.3 WT4

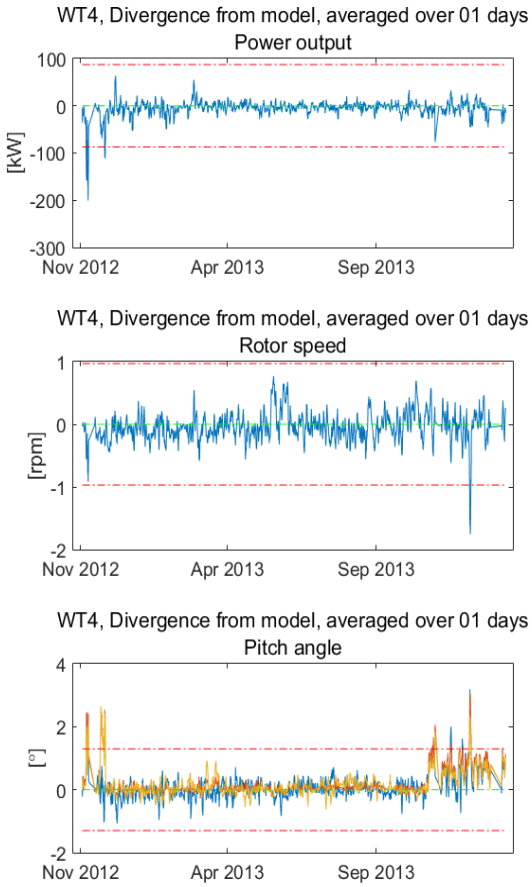


Figure 9.7 Error response WT4

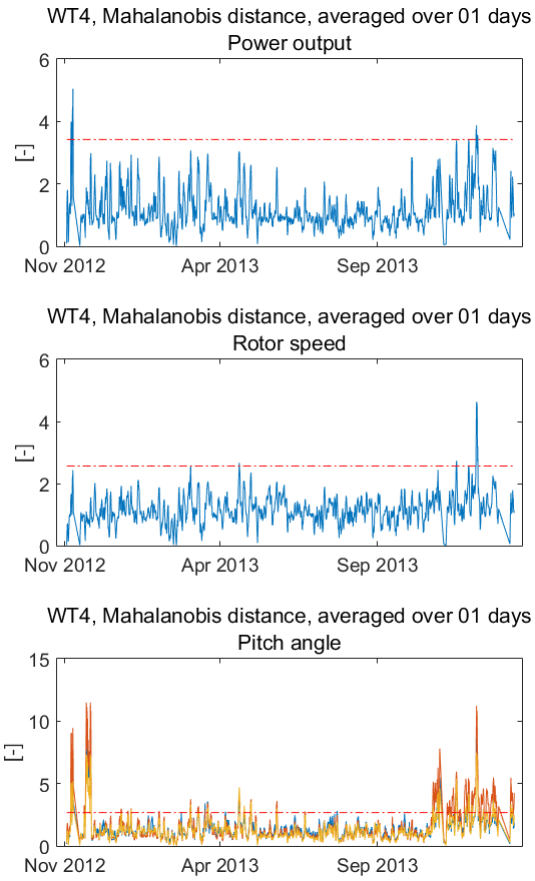


Figure 9.8 Mahalanobis distance WT4

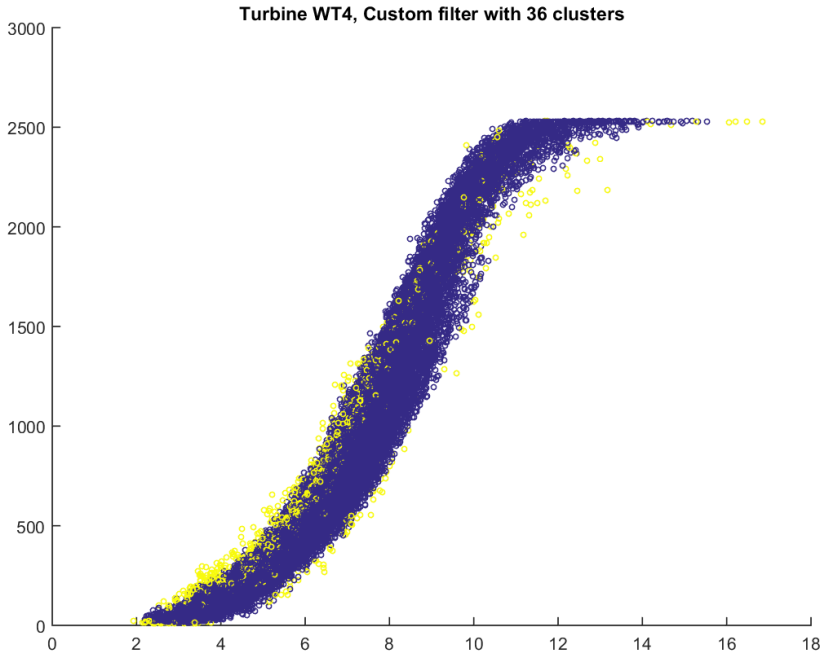


Figure 9.9 Filter result WT4

9.2.4 WT5

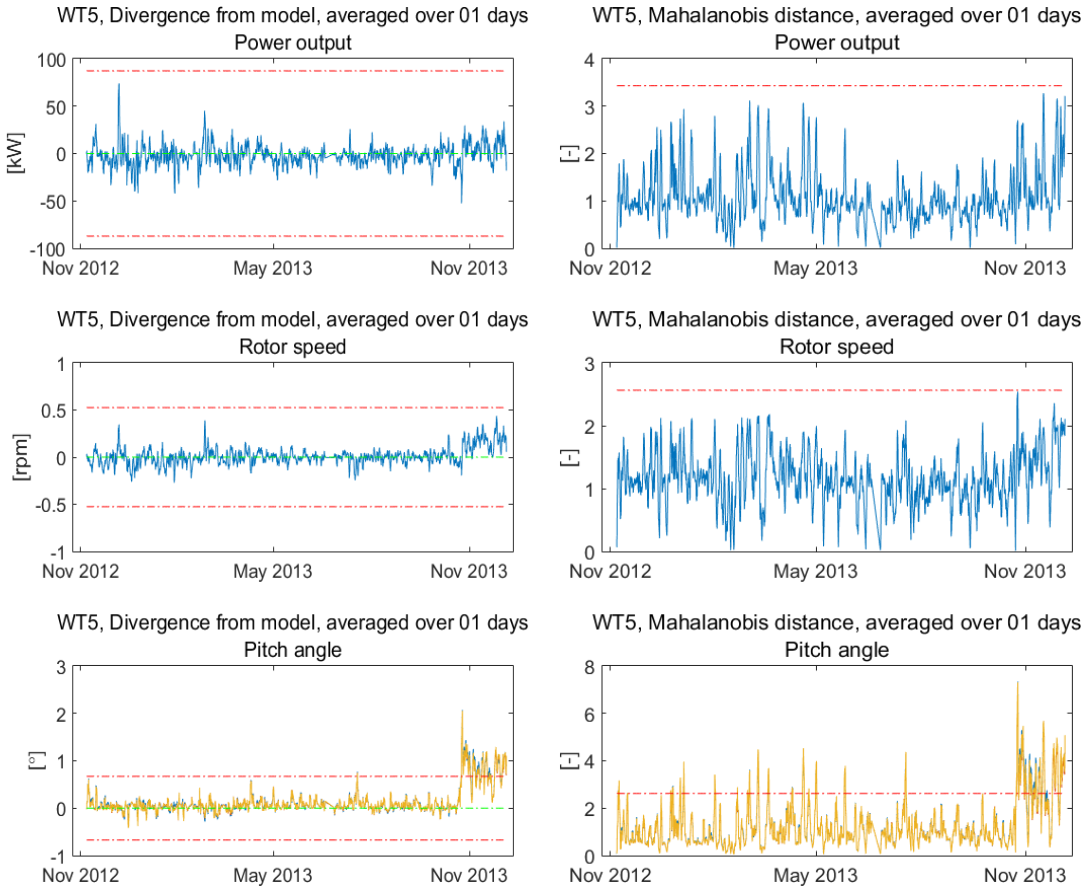


Figure 9.10 Error response WT5

Figure 9.11 Mahalanobis distance WT5

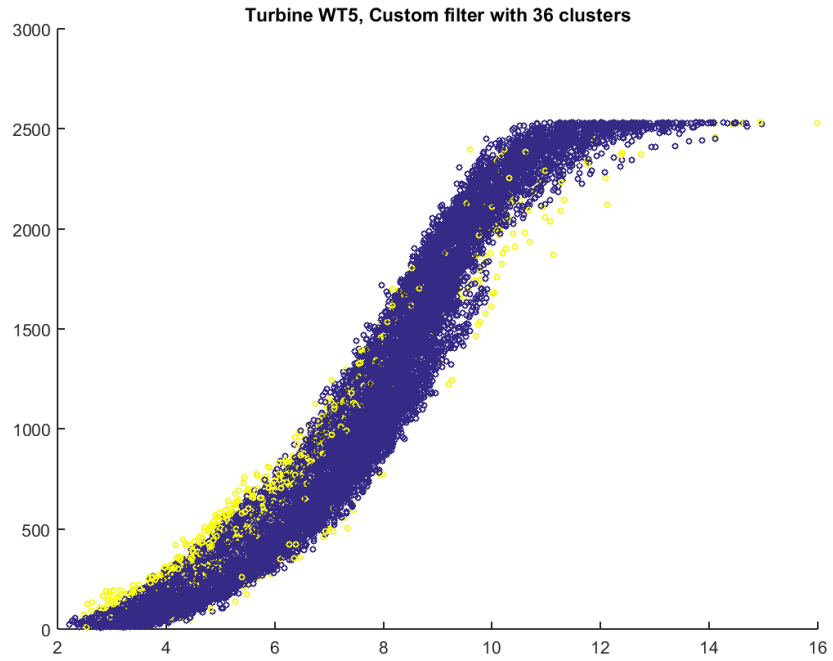


Figure 9.12 Filter result WT5

9.3 Historic power data remaining turbines

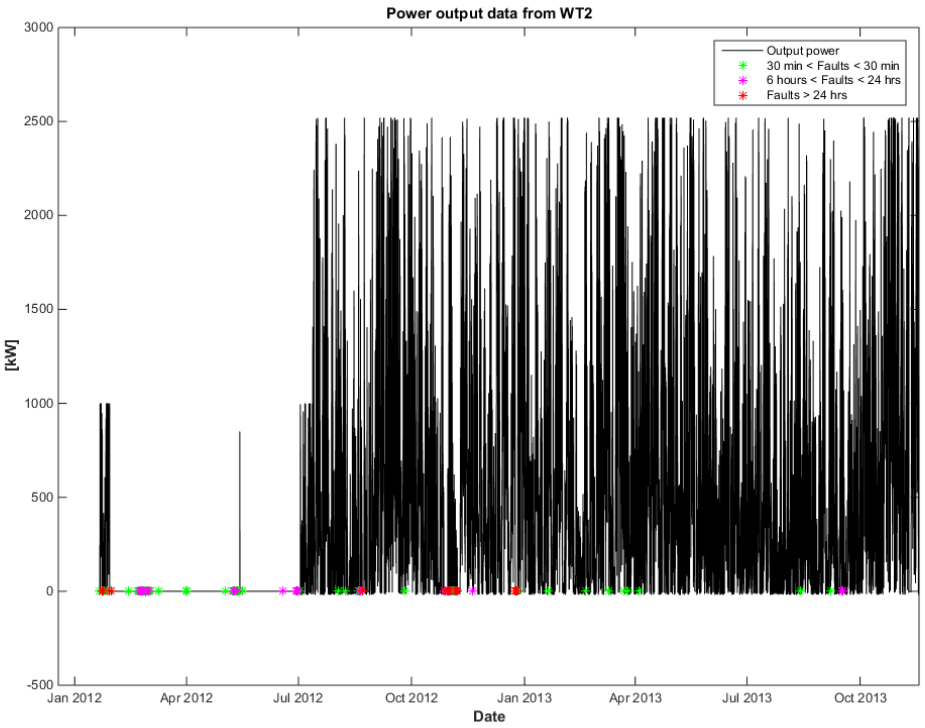


Figure 9.13 Historical power output WT2.

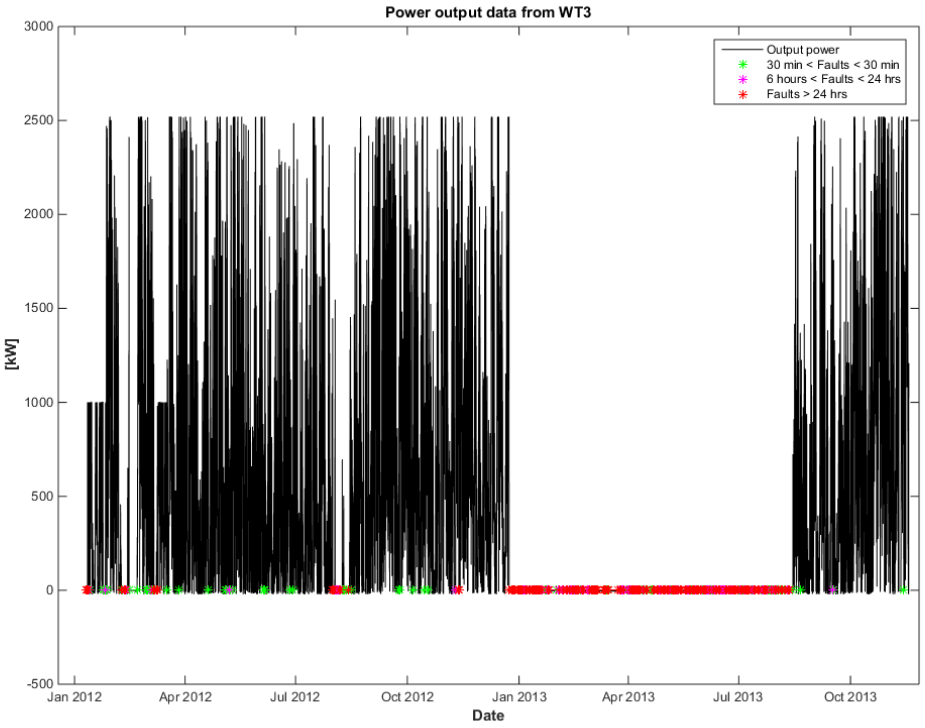


Figure 9.14 Historical power output WT3.

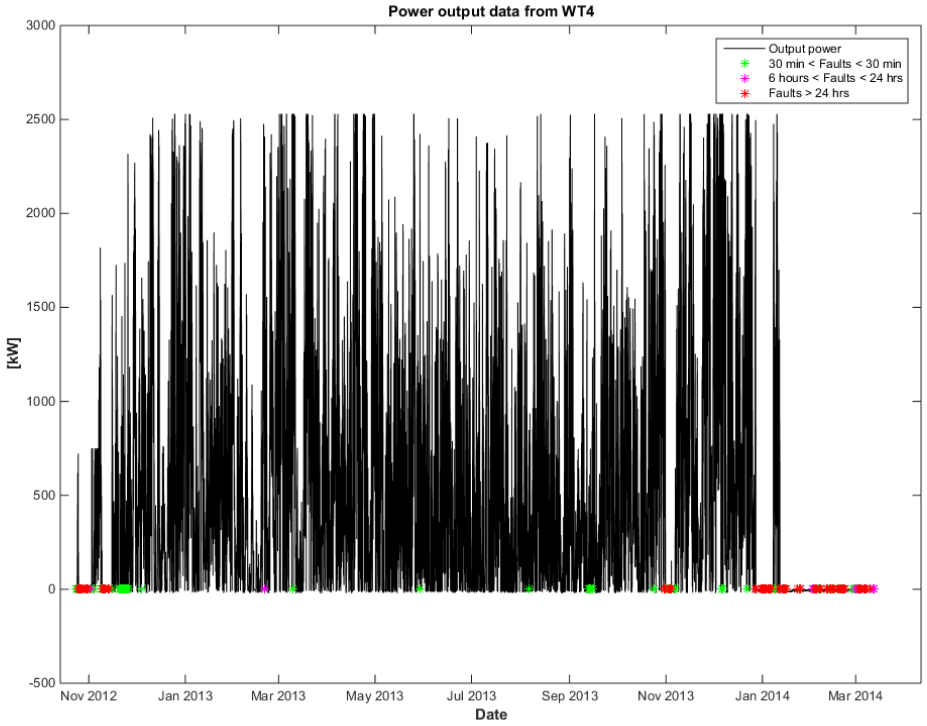


Figure 9.15 Historical power output WT4.

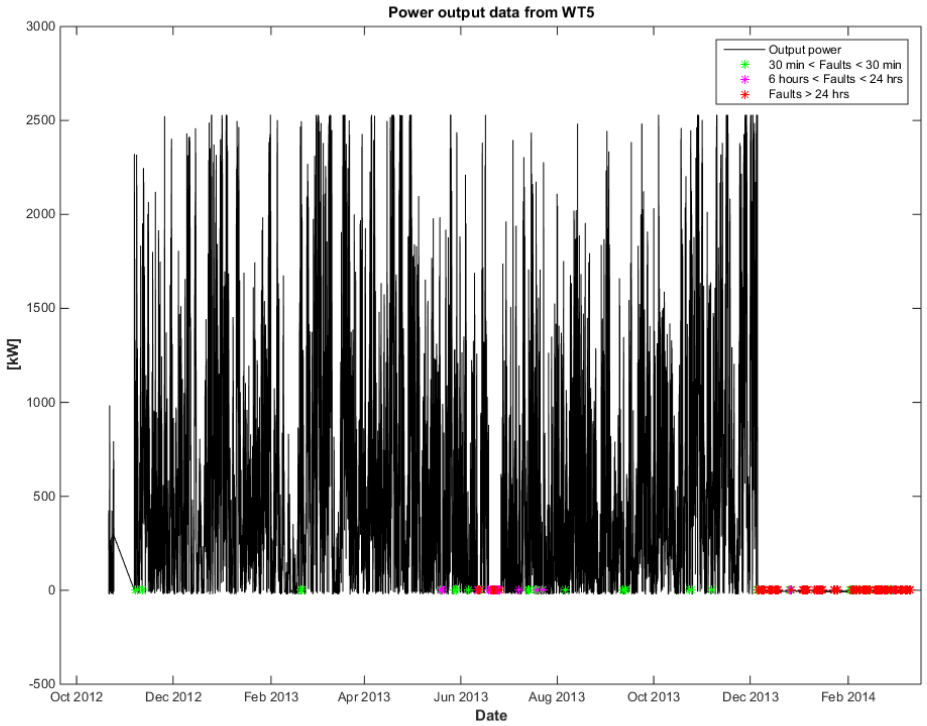


Figure 9.16 Historical power output WT5.