Implementation of machine learning to model losses from icing of wind turbines

Johan Ihlis
Abstract

This thesis investigates the possibility to use machine learning algorithms to predict the losses due to icing in the Stor-Rotliten wind farm that is situated in the north of Sweden and operated by Vattenfall. The inputs for the machine learning are historical mesoscale modelled variables that are derived from a Weather Research and Forecasting Model code that is tuned for icing (WRF-model). An ice model has been updated and improved so that it would achieve a better indication of icing, based on the equations from Lasse Makkonen.

A more accurate model of a wind turbine considers the turbine blade as a rotating cylinder at 85% of the length of the blade and not as vertical cylinder that stands still. Besides this, the variables from the mesoscale data are used as inputs for the machine learning algorithm.

The targets are energy production losses due to icing that is computed from historical SCADA data that covers the same time frame as the WRF data. To reduce the complexity and the computational time of the system a statistical variable selection algorithm, called mutual information, is used with the MILCA toolbox for Matlab. The target for the variable selection and the machine learning is the average loss of power per wind turbine per hour. This is extracted from the production data from Vattenfall. The goal with the thesis is to relate the modelled mesoscale data with the production data (SCADA).

The overall result of the study is that the neural network method offers a suitable and more accurate way to predict the losses from icing on wind turbines, but there is some work still to be done to reduce the errors in the input variables.
I would like to thank Vattenfall for all the help with this project, especially Benjamin Martinez who has been my supervisor during this project and Annika Billstein Andersson for the opportunity to look into this very interesting research area. I would also like to thank Carlos Restrepo for his priceless help with evaluating different machine learning algorithms and getting me on the right track for moving the thesis work forward. Carlos also helped with the contact between me and Cristian Guarnizo who helped with the MILCA algorithm for the variable selection.

Johan Ihlis

Solna, October 2014
LIST OF ABBREVIATIONS

- CFD – Computational Fluid Dynamics
- JMI – Joint Mutual Information
- LAM – Limit Area Model
- LWC – Liquid Water Content
- MERRA – Modern Era Retrospective analysis for Research and Applications
- MI – Mutual Information
- MILCA – Mutual Information Least dependent Component Analysis
- MSE – Mean Squared Error
- MVD – Median Volumetric Diameter
- NaN – Not a Number
- NARX – Nonlinear Autoregressive network with External inputs
- NVP – Numerical Weather Prediction
- RMSE – Root Mean Squared Error
- SCADA – Supervisory Control And Data Acquisition
- WETIP – Wind Power Evaluation Tool for Ice related Production losses
- WRF – Weather Research and forecasting model
LIST OF FIGURES

Figure 1. Map of Stor-Rotliden wind farm. (Hellström 2013) .................................10
Figure 2. The method ..........................................................................................11
Figure 3. Entropies and mutual information reproduced from (Brown, Pocock, et al. 2012). ...17
Figure 4. Schematics for the training of a neural network. (Matlab 2014) ..................18
Figure 5. The new and old model, a still standing cylinder of 1 meter in height in the old model and at 85% of the blade length in the new model. The blue area is the investigated area ....19
Figure 6. Modelling of RPM for the turbine blades ..............................................21
Figure 7. Scatter of power production during the warm period. (Hellström 2013) ..........22
Figure 8. Scatter of power production during the cold period. (Hellström 2013) ............22
Figure 9. Neural network with 2 input variables, 1 target 4 delays and 10 nodes ............25
Figure 10. Closed loop NARX network ....................................................................26
Figure 11. Comparison between old and new ice load model ..................................27
Figure 12. Zoom in of January .................................................................................27
Figure 13. Losses plotted over the ice load ..............................................................28
Figure 14. Root mean squared error for the validation ..........................................29
Figure 15. Correlation between outputs and target values ......................................30
Figure 16. Difference between the targets and the outputs for 4 nodes and 2 delays .........30
Figure 17. Output and target values with 4 nodes and 2 delays ..................................31
Figure 18. Zoom in on the beginning of October in Figure 17 ....................................31
Figure 19. Difference between the target values and the outputs for 10 nodes with 2 delays....32
Figure 20. Output and target values with 10 nodes and 2 delays ...............................32
Figure 21. Zoom in on Figure 20. Output and target values with 10 nodes and 2 delays at the end of October .................................................................33
LIST OF TABLES

Table 1. Input variables.......................................................................................................23
Table 2. Years used for training and validation.................................................................25
Table 3. Variables selected through mutual information...................................................28
<table>
<thead>
<tr>
<th>Section</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>FOREWORD</td>
<td>3</td>
</tr>
<tr>
<td>LIST OF ABBREVIATIONS</td>
<td>4</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>5</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>6</td>
</tr>
<tr>
<td>TABLE OF CONTENTS</td>
<td>7</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>9</td>
</tr>
<tr>
<td>1.1 Background</td>
<td>9</td>
</tr>
<tr>
<td>1.1.1 Wind Power in cold climates</td>
<td>9</td>
</tr>
<tr>
<td>1.1.2 Wind farm</td>
<td>9</td>
</tr>
<tr>
<td>1.1.3 Difficulties with physical model</td>
<td>10</td>
</tr>
<tr>
<td>1.2 Objective</td>
<td>10</td>
</tr>
<tr>
<td>1.3 Methodology</td>
<td>11</td>
</tr>
<tr>
<td>2 FRAME OF REFERENCE</td>
<td>12</td>
</tr>
<tr>
<td>2.1 Icing</td>
<td>12</td>
</tr>
<tr>
<td>2.2 Model of ice</td>
<td>12</td>
</tr>
<tr>
<td>2.6 Machine learning</td>
<td>13</td>
</tr>
<tr>
<td>2.7 Variable selection</td>
<td>14</td>
</tr>
<tr>
<td>2.8 Entropy and Mutual Information</td>
<td>15</td>
</tr>
<tr>
<td>2.9 Neural network theory</td>
<td>18</td>
</tr>
<tr>
<td>3 THE PROCESS</td>
<td>19</td>
</tr>
<tr>
<td>3.1 Improving the ice model</td>
<td>19</td>
</tr>
<tr>
<td>3.2 Energy losses from icing</td>
<td>21</td>
</tr>
<tr>
<td>3.4 Obtaining the relevant icing variables with the usage of mutual information</td>
<td>23</td>
</tr>
<tr>
<td>3.5 Machine learning</td>
<td>24</td>
</tr>
<tr>
<td>3.6 The usage of a neural network</td>
<td>24</td>
</tr>
<tr>
<td>4 RESULTS</td>
<td>27</td>
</tr>
<tr>
<td>4.1 Results from the new ice model</td>
<td>27</td>
</tr>
<tr>
<td>4.2 Energy losses from icing</td>
<td>28</td>
</tr>
</tbody>
</table>
1 INTRODUCTION

1.1 Background

1.1.1 Wind Power in cold climates

Wind power is a source of renewable energy that has low life-cycle emissions related to it and it is cost efficient compared to many other alternatives. This means that the wind power is very likely to expand during the coming years and improve its economy even more. Since a lot of the potential sites for wind power have been exploited already, power companies have to expand to more difficult sites to build large wind turbine parks. (Davis, Hahmann, & Clausen, 2013)

In Sweden, new wind power farms are planned especially in the northern parts of the country, mostly on mountainous terrain (WSP, 2011). This is done since the average wind speed generally increases with 0.1 m/s for every 100 m rise in altitude above sea level (Olivier & Ilinca, 2011). Also the cold climate conditions have a positive effect as the higher density of air at low temperatures increases the amount of power available in it. In cold regions the air contains up to 10% more energy due to the higher density of the cold air (Lamraoui, Fortin, Benoit, Perrona, & Masson, 2014).

A problem that occurs when a wind turbines is being placed in cold climate is the large probability for icing on blades and structures. When ice starts to form on the blades it affects the aerodynamic properties of the turbine rotor, especially if the icing occurs on the leading edge of the blade. When the aerodynamic properties are changed so is also the power output from the turbine (Lamraoui et al. 2014). Icing of wind turbine blades can lead to energy losses between 5% and 10% for typical locations in the north of Sweden.

For future investments it is critical to find sites where it is profitable to build and operate wind turbines. To do this the wind power business has to be able to model the icing and the losses from it in an accurate way.

When there is a risk for icing there is also a risk for power loss from the turbines. The power loss can originate from both the necessary stopping of the turbine when heavy ice accumulates on blades, and also from the gradually lowered aerodynamic performance at lower degrees of ice accretion. When ice accumulates on the turbine blades their shape and mass is changed and therefore both the aerodynamic and mechanical forces on the blades are changed and may reach undesired levels (Hau, 2013).

1.1.2 Wind farm

The wind farm that is studied in this thesis is Stor-Rotliden (Vattenfall, 2013) situated in Åsele municipality in the north of Sweden. The wind farm is known to have troubles with icing. It consists of 40 Vestas V90 wind turbines that have a tower height of 95 meters and a rotor diameter of 90 meters. The total installed power is 78 MW and most of the turbines are rated at 2 MW, one is at 1.8 MW. The wind farm started to produce at full capacity in the end of 2010 (Vattenfall, 2013). Figure 1 presents a generalized map and a siting scheme of the Stor-Rotliden wind park.
1.1.3 Difficulties with physical model

Icing and shedding from wind turbines is a complex physical process this makes it hard to create a physical model properly covering all the different possible cases. This can be done for a small part of a wind turbine using CFD analysis but for a whole wind farm it is very computationally expensive. One of the difficulties with creating a physical model is that the meteorological data available have a grid size of 1 km which means that only 1 data point is used for the whole wind farm. Since the icing problem is at a small scale and the losses from icing can be found by using the production data for each wind turbine unit within the farm, there is a need to find a relation between the large-scale meteorological data and the small-scale unit-wise production data.

There is also the problem that the variables do not change linearly and that is also the case for the relation between the variables and the target values. This leads to that small changes in one of the variables can lead to large changes in the output. With a physical model based on the meteorological data with one single data point for the whole wind farm there is no possibility to describe or predict the actual variations among the different wind turbines in the wind farm. It is believed that a machine learning algorithm can help finding these variations and to properly take them into account.

1.2 Objective

The purpose of this study is to investigate whether machine learning could be used to provide accurate estimations of the power output losses from wind turbines due to icing of blades, and if it can be applied as a better alternative to the empirical physical models used today at Vattenfall that have been shown to be inaccurate (Martinez, 2014).
1.3 Methodology

To get the best possible output from a mathematical model it is necessary to supply the most accurate input to the model. In a first step, this study aimed at finding the most accurate inputs for the icing model that simulates the amount of icing on the turbine blades and that is used by Vattenfall today, created by Benjamin Martinez. Secondly, the abovementioned model is updated so that it correlates with the IceBlade model found in the peer reviewed article after Davis et al., 2013. This program is described in detail in section 3.1 Improving the ice model. The logic of the method is presented in Figure 2.

The output from the new icing model will together with the available meteorological data from the Weather Research and forecasting model (WRF) model be used as input variables to a newly proposed the variable selection program. To get the target values an internal program called Wind Power Evaluation Tool for Ice-related Production Losses (WETIP) is used, this program analyses production data from the turbines together with meteorological data from the same time spectrum, in order to decide what losses can be attributed to icing and what losses occur due to other reasons. The variable selection program is described in section 2.4.

In order to decide which input data contains the most information about the target values the statistical method of mutual information will be used, this will be done by using the MILCA toolbox for Matlab. This method and program is described in section 3.4. Then different methods of machine learning will be tested in Matlab with the help of the postdoc Carlos Restrepo and discussions will be held with other companies of what methods that they consider applicable. Using the collected information, the most promising method will be selected, implemented in a machine learning algorithm and investigated further for its accuracy in real conditions.

**Figure 2: The method**

- WRF simulated data
- Production data
- MERRA data
- ice model
- updated ice model
- WETIP
- Losses from icing
- Target values
- 8 input variables
- Mutual information
- 4 input variables
- machine learning training
- machine learning validation
- prognosis of losses
2.1 Icing

Icing can occur during different conditions: precipitation, clouds at low level and fog; the temperature also has to be below zero centigrade (Lamraoui et al. 2014). This leads to different types of icing, hoar-precipitation- and in-cloud-icing. When investigating wind turbines the interesting types of icing are the precipitation and in-cloud ones, especially in Sweden where in-cloud icing gives the largest losses (Ronsten, 2008). The moisture from in-cloud icing comes from low stratus clouds and fog (Lamraoui et al. 2014).

Precipitation icing comes from either wet snow or freezing rain. Freezing rain gives a high ice density which is often made up of glaze icing and comes from droplets that hit a surface that has a temperature below 0ºC (Olivier & Ilinca, 2011). Wet snow is mostly a problem for wind turbines when they are not operating (WSP, 2011).

In-cloud icing, which is the focus of the physical model in this paper, comes from small supercooled water droplets that hit a surface with a temperature below 0ºC and down to -30º C (Olivier & Ilinca, 2011). The amount and shape of in-cloud icing depends on some atmospheric variables, the size of the water droplets in the air which is called the Median Volumetric Diameter (MVD), the numbers of water droplets in the air called the Liquid Water Content (LWC), the wind speed, the temperature and the time that the turbine is in the cloud, the length of the blades chord and the collection efficiency of the blade (Olivier & Ilinca, 2011).

There are three types of icing that occurs in different temperatures, there is soft rime, which has a low density and consists of flakes and needles. Soft rime is created in temperatures far below 0ºC in air with low LWC and MVD. At warmer temperatures or when the droplet size is larger hard rime is formed, it has higher density and larger adhesion. When there is a part of the water droplet that does not freeze at impact on the surface, for example when there is precipitation or wet in-cloud icing. It floats out and forms glaze ice. The glace ice has a high density, it sticks to the surface very hard and forms a transparent and smooth surface (Olivier & Ilinca, 2011).

Frost can also be created on the turbine blade, it comes from water vapour that freezes on the surface when it has a temperature below 0ºC (Olivier & Ilinca, 2011).

The blades of the turbine are very sensitive to changes both in shape and roughness of the surface on the leading edge. This means that if the surface is changed only by a little it can lose its aero-dynamic properties and if the icing gets really bad, the turbine can be damaged or come to a complete stop (Lamraoui et al. 2014). The ice on the turbine, both nacelle and the blades, can cause damage through vibrations, load asymmetry and icing of instruments (Lamraoui et al. 2014). When the instruments gets iced it can lead to faulty measurements, for example if the wind vanes or anemometers get iced it can give the turbine problems with finding the right position against the wind which can lead to vibrations that wear down the turbine and also it can lead to lower production. If the blades are exposed to glaze icing there is a risk that the turbine overproduces energy, at least if it is passive pitch controlled, since the glace can delay the stall, which also can damage the turbine (Laakso et al. 2003).

2.2 Model of ice

To predict the frequency of icing meteorological mesoscale models, for example MM5 and MC2 can be used by identifying the needed weather conditions. More advanced empirical or statistical models can be used to predict more details about the icing events for example the amount of
icing (Olivier & Ilinca, 2011). The empirical and statistical models use more information such as wind speed, object size, humidity profile of the air. To predict the effects of icing on each individual wind turbine the icing can be modelled using computational fluid dynamics (CFD). (Homola, Virk, Nicklasson, & Sundsbø, 2012)

There are two major software codes that predict icing and its related losses: Lewice (NASA) is an analytical icing model that is mostly used to find the energy needed for anti-icing but can also find collection efficiencies and the ice shape and thickness; the other one is Turbice and it is a numerical program that is based on the method from (Makkonen, Laakso, Marjaniemi, & Finstad, 2001) that can simulate both rime and glaze icing and a heated blade. (Laakso, o.a., 2003) Vattenfall has used a physical model based on empirical studies by Benjamin Martinez and on the theory of Makkonen.

In (Makkonen L., 2000) Lasse Makkonen introduces his equation for the accretion rate of icing, this is shown here as equation (1).

$$\frac{dM}{dt} = \alpha_1 \alpha_2 \alpha_3 w v A$$

In equation (1): A is the cross section area, v is the velocity of the particles relative to the blade, w is the mass concentration of the water particles in the air and the α are correction factors that ranges from 0 to 1 in value. (Makkonen L., 2000) Makkonen defines α₁ as the collision efficiency of the particles; this means how many of the particles in the air that actually hits the turbine blade. Larger particles tend to hit the blade while smaller particles tend to follow the air stream around the blade profile.

The variable α₂ is defined as the collection efficiency which means the amount of the particles that actually sticks to the surface and does not bounce away.

The variable α₃ is defined as the accretion efficiency and represents the amount of the particle that freezes on the surface; thereby it is reduced from 1 if there is runoff from the blade.

To forecast the atmospheric conditions there are different models that can be used, Weather research and forecasting model (WRF) is a numerical model that is developed by National Oceanic and Atmospheric Administration, National Centre for Atmospheric Research and universities together. This is the forecasting model that is used to get the mesoscale data in this thesis. WRF is a type of numerical weather prediction (NWP). The NWP model makes it possible for computers to numerically solve nonlinear differential equations, starting with an initial state of the atmosphere, to find the state of the atmosphere at a later time. This can be done at different levels covering the whole earth or a part of it. For this thesis the interesting part is the regional limited-area models (LAMs) that only cover a small part of a country. (Schelander & Hansson, 2013) This is still computationally expensive, especially if it is needed in high resolution, with a grid size down to 1x1 km. Simulation with the small grid size is only done for shorter times, for example a couple of years. When it is needed for longer time it is done in lower resolution with a grid size of between 4 and 9 km.

### 2.6 Machine learning

Machine learning is a combination of many different fields, it is a branch of artificial intelligence, based on neuroscience, cognitive science and closely connected to statistics. (Jebara, 2004) The neuroscientist McCulloch and the logician Pitts suggested a model of a neuron in 1943, as a computational unit that could perform different Boolean functions.
In 1948 a mathematical model that suggested that every picture and word “could be represented, modelled and transmitted with finite symbols or bits” was presented, this was the start of the field of information theory. (Jebara, 2004) The subfield of machine learning was born in 1956 during the first conference about artificial intelligence. In 1958 the first primitive model of a neural network was proposed in the form of modelled neuron that would get inputs that can be weighted in a certain way to learn a task. The sum of the weighted inputs would then be summed up and compared to a threshold; this would generate a binary output. This model could not handle some nonlinear problems but this was corrected later on when the back-propagation algorithm for a multilayer neural network was proposed. (Jebara, 2004)

2.7 Variable selection

When investigating which variables to use for the program to learn from it is important to use the ones that contribute the most to the result. If too many variables are used to fit the model it is possible that the model suffers from over-fitting. (Brown, Pocock, Zhao, & Luján, 2012) Over-fitting is when the model has too many variables to work with and therefore adapt itself to the data in such a large extent that the model cannot generalize and thereby work with another set of data. (Mitchell, 1997)

It is important to differ between feature selection and feature extraction, where feature extraction is when a new features are created from the original one and feature selection is when the best subset of the original features are selected to represent it. (Jain, Duin, & Mao, 2000)

There are three different methods of feature selection: filters, wrappers and embedded methods. Wrappers work by taking a classifier, training and validating it by trying all combinations of the variables. This method demands very much computation capacity and has the risk of getting the model so specific that it is hard to use it on any other set of features, but can give very good results in generalisations when the same features are being used. (Brown, Pocock, Zhao, & Luján, 2012) (Torkkola, 2003)

Embedded methods incorporate both learning and feature selection and works in the way of a blackbox, it is specific for each of the learning machines. (Torkkola, 2003) Embedded methods needs less computational capacity than wrappers but more than filters. For logical reasons embedded methods are faster than wrappers but slower than filters.

Filters statistically rank the features by looking at the connection between the given target value and the variable at hand. The variables with the highest ranking gets chosen, this means that the filter work with different types of data without and classifier. Mutual information can be used as a filter to select the features. (Torkkola, 2003) Filters are also the least likely method to over-fit the model and it is also the fastest method in computation. (Brown, Pocock, Zhao, & Luján, 2012)

The usage of a filter is motivated in a definition from (Brown, Pocock, Zhao, & Luján, 2012), as follows:

“Given an objective function for a classifier, we can address the problems of optimizing the feature set and optimizing the classifier in two stages: first picking good features, then building the classifier to use them.”
2.8 Entropy and Mutual Information

When describing the entropy in variable selection one tries to describe how unpredictable a certain variable is. The entropy is described as the probability of a certain variable taking on a certain value, \( p(x) = Pr\{X = x\}, x \in \mathbb{R} \) where \( x \) is the possible value, \( X \) is the variable and \( \mathbb{R} \) is the feature that \( X \) can take values from. (Brown, Pocock, Zhao, & Luján, 2012) There are several different definitions of entropy, the one given by (Brown, Pocock, Zhao, & Luján, 2012) is called the differential entropy and is shown in Eq. (2).

\[
H(X) = - \sum_{x \in \mathbb{R}} p(x) \log(p(x)) \tag{2}
\]

\( H(X) \) is called the marginal entropy and it can have different bases to the log, if the base is 2 it is counted in bits and if it is in \( e \) then the entropy is in nats. (Cover & Thomas, 1991)

To calculate the entropy of a variable is to know what the probability of that variable coming up being compared to the total amount of instances. This is called the maximum likelihood: (Brown, Pocock, Zhao, & Luján, 2012)

\[
\hat{p}(x) = \frac{\# x}{N} \tag{3}
\]

where \( x \) is one value or span of values of the variable and \( N \) is the total amount of times that any value comes up. (Brown, Pocock, Zhao, & Luján, 2012) This leads to that if one number comes up at most of the instances the entropy will be close to 0 as explained in Eq. (4) and Eq. (5). Since it is not possible to look at every possible value, what is done instead is to look at the k values that are closest to the values of interest and count the number of instances when the variable takes any of these values. This is called the k-nearest neighbour method.

\[
\hat{p}(x) = \frac{N}{N} = 1 \tag{4}
\]

\[
H(X) = - \sum_{x \in \mathbb{R}} 1 \log(1) = 0 \tag{5}
\]

The difference between \( p(x) \) and \( \hat{p}(x) \) is that the first one is the true distribution and second one is the estimated or calculated distribution. The estimated distributions are the ones that are used to calculate the entropy; this is possible through the Strong Law of Large Numbers states that \( \hat{p} \) with great certainty will converge to a value very close to the true distribution value. (Brown, Pocock, Zhao, & Luján, 2012)

The entropy can also be both conditional and joint, given the variables \( X, Y \) and the joint distribution is \( p(x,y) \) the joint entropy is defined as Eq. (6) (Cover & Thomas, 1991)

\[
H(X,Y) = - \sum_{x \in \mathbb{R}} \sum_{y \in Y} p(x,y) \log(p(x,y)) \tag{6}
\]
The joint distribution is defined as (7) according to (Kraskov, Stögbauer, & Grassberger, 2004):

\[ p(x, y) = \frac{p(x) \wedge p(y)}{N} \]  

(7)

Conditional entropy means that there is a part of the uncertainty in a variable \( X \) that is left even when another variable \( Y \) is known. (Brown, Pocock, Zhao, & Luján, 2012) The conditional entropy is defined as (8):

\[ H(X|Y) = -\sum_{x \in Y} p(y) \sum_{x \in \mathbb{R}} p(x|y) \log(p(x|y)) \]  

(8)

The entropy and the conditional entropy can be used to define mutual information (MI) as the amount of information that can be known about \( X \) by knowing \( Y \) and also the other way around, what can be known about \( Y \) by knowing \( X \). (Brown, Pocock, Zhao, & Luján, 2012). The conditional distribution is defined below as Eq. (9).

\[ p(x|y) = \frac{p(x) \wedge p(y)}{p(y)} \]  

(9)

The MI can be defined with help of the conditional entropy as in Eq. (10).

\[ I(X; Y) = H(X) - H(X|Y) = \sum_{x \in Y} \sum_{x \in \mathbb{R}} p(xy) \log \frac{p(xy)}{p(x)p(y)} \]  

(10)

Mutual information means what amount of uncertainty that can be removed from the variables \( X \) when knowing the variable \( Y \). To be able to do the calculations needed to find the mutual information it is important to know that the mutual information can be approximated using the estimation of (3). (Brown, Pocock, Zhao, & Luján, 2012)

\[ I(X; Y) \approx \hat{I}(X; Y) = \frac{1}{N} \sum_{i=1}^{N} \log \frac{\hat{p}(x_i y_i)}{\hat{p}(x^i)\hat{p}(y^i)} \]  

(11)

In Figure 3 the difference between the entropies and the mutual information can be seen. The marginal entropies describe the uncertainty of the variables when no other information is known. The conditional entropies describes the uncertainty that is left of the variable when the other variable is known, the joint entropy is the amount of uncertainty that is removed from both variables when both is known. The mutual information is described by the overlap of the circles, this can easily be seen when looking at equation (10) and in Figure 3.
It is stated in (Studholme, Hill, & Hawkes, 1999) that there is a need for a measure of mutual information that is not affected of the changes in the marginal entropies; otherwise difficulties can occur when there is overlap between two variables. This can be done through the normalized mutual information, this is deduced to equation (12) in (Studholme, Hill, & Hawkes, 1999).

\[
NI(X, Y) = \frac{H(X) + H(Y)}{H(X, Y)}
\]  \hspace{1cm} (12)

When the mutual information is known the next step is to decide what criteria the algorithm should use to choose which of the features to include. Here either a wrapper, embedded or filter method could be used. When the choosing is done by a filter a scoring criterion J that is used to find out how relevant the feature \( \mathbb{R} \) is for the outcome \( Y \). There are several different scoring criteria that are based on different methodology but G. Brown in (Brown, Pocock, Zhao, & Luján, 2012) states that the joint mutual information criterion has the best balance of relevancy and stability and therefore gives the most trustworthy results.

\[
ni(X_k) = I(X_k; Y) - \frac{1}{|S|} \sum_{j \in S} [I(X_k; X_j) - I(X_k; X_j|Y)]
\]  \hspace{1cm} (13)

The difference between joint mutual information (JMI) and the mutual information is that the joint mutual information takes into account the conditional mutual information. The joint algorithm takes into account the complimentary information from the variables \( X_k \) and \( X_j \). The first term of the joint mutual information says how much mutual information there is between the potential feature \( X_k \) and the target \( Y \). The second term sums up the amount of mutual information there is between the feature \( X_k \) and the already selected feature \( X_j \) and subtracts the amount of mutual information that is already known about the target \( Y \) through the already selected variables \( X_j \). The second term is then divided by the absolute value of the already selected features. (Yang & Moody, 1999) This gives the scoring of the joint mutual information approach.
2.9 Neural network theory

Neural networks can be either dynamic or static. Static neural networks work by taking in inputs at one end, running them through the network and delivering an output on the other end based on the inputs for that instant. Dynamic neural networks on the other hand take in the inputs of the instant into account but also consider the earlier inputs and the earlier outputs into account when calculating the new output. (Jesus & Hagan, 2001)

A neural network consists of nodes, weights, biases, transfer functions, delays, inputs and outputs. The inputs are connected to the input weights via delays. The delays determine how many of the previous values that should be taken into account when calculating the new output. The weighted sum of the input values are summed together with the biases and sent through a transfer function in each node. All of the nodes work in parallel with each other and all the inputs are connected to all the nodes. The transfer function can be of several different kinds; here a nonlinear sigmoid function is used in the hidden layer. The values from the hidden layer is sent to the output layer where it is weighted and summed up with another bias and then sent through a linear transfer function to get a readable output value. The weights and the biases are random numbers when the network starts its training which gets changed when the output value of the network does not match the target value. This is shown in Figure 4.

Figure 4: Schematics for the training of a neural network. (Matlab, 2014)
This project uses data from different programs. The first step of the process was to modify the ice load program, previously created by Benjamin Martinez (Martinez, Personal communication, Wind engineer at Vattenfall R&D, 2014) and based on the methodology of (Makkonen L., 2000) and empirical studies, to the methodology of (Davis, Hahmann, & Clausen, 2013).

3.1 Improving the ice model

The model that is being used to forecast icing event is created based on both the methodology of Neil Davis IceBlade model (Davis, Hahmann, & Clausen, 2013) and the old ice forecasting program made by Benjamin Martinez at Vattenfall that uses some empirical findings to forecast the shedding. Together this forms the model used to find the ice load on the turbine blade for this study. The ice accretion estimation in both models is based on (Makkonen L., 2000), (Eq. 2). The old model was based on a cylinder that is standing still and the wind is passing by it, the new model is based on a section of 1 meter at 85% of the blade length as shown in Fig. 5.

Figure 5: The old and new model, a still standing cylinder of 1 meter in height in the old model and at 85% of the blade length in the new model. The blue area is the investigated area.

The ice model of this thesis is based on the paper (Davis, Hahmann, & Clausen, 2013) and the model called IceBlade from that paper. The equation for ice accretion (1) is combined with the ablation models from (Davis, Hahmann, & Clausen, 2013). There are three types of ablation that takes place on wind turbine blades; melting, sublimation and shedding (Davis, Hahmann, & Clausen, 2013). To make the model less complicated some simplifications have been introduced into the calculations by Neil Davis, as follows:
• The slightest loss in adhesion during operation is considered to throw the ice off the blade and it can only happen due to melting at the surface of the blade. This will happen when the surrounding temperature is more than 0.5°C for more than 1 hour.
• The fact that the shedding threshold is so low makes melting impossible in this model and it is thereby not considered.
• The turbine is always considered to follow the idealized operating curve (Davis, Hahmann, & Clausen, 2013).

The sublimation is modelled with Eq. (14)

\[
\frac{dm}{dt} = \left( \frac{A \cdot Sh \cdot D \cdot \rho_s(T_\infty)}{L \left(1 + \frac{L_s \cdot D \cdot Sh}{k \cdot Nu \rho_s'}\right)} \right) \left[1 - \alpha(s - s_p) + 2\alpha^2(s - s_p)^2 - 5\alpha^3(s - s_p)^3\right] \tag{14}
\]

which is Eq. 16 in (Srivastava & Coen, 1992) modified according to (Davis, Hahmann, & Clausen, Forecast of Icing Events at a Wind Farm in Sweden, 2013). A is the cross section area, Sh is the Sherwood number, D is the diffusivity of water vapour in air, \(s_p\) is the equilibrium super saturation of the particle and it is set to 0 in this paper due to the fact that it can be neglected for raindrops and is very small for cloud drops. (Srivastava & Coen, 1992) S is the ambient super saturation, \(\rho_s(T_\infty)\) is the saturation vapour density with respect to the ambient air temperature and \(\rho_s'\) is the differential of it. \(L_s\) is the latent heat of sublimation, L is the chord length, Nu is the Nusselt number and \(k\) is the thermal conductivity of the air. Alpha with the modifications from (Davis, Hahmann, & Clausen, 2013) is displayed as Eq. (15)

\[
\alpha = \frac{1}{2} \left(\frac{L_s D Sh}{k Nu} \right)^{\rho_s''} \rho_s \rho_s' \tag{15}
\]

Earlier models did not take evaporation or sublimation into account due to complexity. (Lamraoui, Fortin, Benoit, Perrona, & Masson, 2014)

The chord length is considered to be 1 since the model does not depend on it (Davis, 2014). The length of \(s\), which is the length from the front of the blade to the end of the ice is considered to be 0 since the ice will grow straight out of the front of the blade, according to (Davis, 2014).

The saturation vapour pressure comes from Tetens formula and the dynamic viscosity of the air is taken from Sutherland’s law (White, 2006). The Reynolds and the Prandtl number are calculated with respect to the chord. The calculations of the Prandtl- and Nusselt number come from (Sherif, Pasumarthi, & C.S., 1997) and the Sherwood number comes from (Davis, Hahmann, & Clausen, 2013).

\[
Pr = \frac{c_p \mu}{k} \tag{16}
\]

\[
Nu = Re_L^{0.6} Pr^{-0.4} \left[1.14 \left(\frac{L}{d}\right)^{0.5} - 2.353072 \left(\frac{L}{d}\right)^{3.5} \left(\frac{S}{L}\right)^3\right] \tag{17}
\]

\[
Sh = Re_L^{0.5} Sc^{0.4} \left[1.14 \left(\frac{L}{d}\right)^{0.5} - 2.353072 \left(\frac{L}{d}\right)^{3.5} \left(\frac{S}{L}\right)^3\right] \tag{18}
\]
\[ Sc = \frac{\mu}{\rho D} \] (19)

Where: \( \mu \) is the dynamic viscosity of the air, \( c_p \) is the specific heat at constant pressure of air and \( k \) is the thermal conductivity of the air. \( L \) is the cord length used as the characteristic length, \( D \) is the mass diffusivity, \( s \) is the distance along the airfoil surface measured from the leading edge and \( d \) is the diameter. To verify the programming a discussion with Neil Davis has been held which included the comparison of values from the program. “Sc” in Eq. (19) is the Schmidt number.

To find the relation between the wind speed and the speed of the turbine blade through the air the SCADA production data is used. From the SCADA data the revolutions per minute of the turbine blade is found for different wind speeds, these numbers are then fitted to a curve. The curve is used to relate the modelled wind speeds from the WRF model to a specific speed of the turbine blade, this speed is used to model the icing with the model described above. The method of doing this is shown in Figure 6.

**Figure 6: Modelling of RPM for the turbine blades.**

![Modelling of RPM for the turbine blades.](image)

### 3.2 Energy losses from icing

The WETIP program is used to get the actual losses caused by icing on the wind farm for the years in question. This program works by comparing the actual power produced with optimal power curve. These power curves are created in 8 different directions to take into account the possibility that the wind turbine is placed in a wake from another wind turbine or from the topography. These directional power curves are created both for warm climate. To find the losses from icing the warm power curves are considered as optimal in each of the 8 directions. Those are compared with the actual production and the difference is considered as losses. If the program recognizes the losses as icing the temperature has to be lower than 2°C and the loss has to be more than 150 kW. The program has been changed during this thesis to mark the instances that does not have the status set to either Turbine OK, Operation Status or Run as NaN instead of 0 as it was before. The program also replaces the instances where one of the following is not available, the wind speed, the power or the rotational speed, with NaN. This change is used to filter out the data that cannot be used. The WETIP program uses MERRA and SCADA data to find the losses.
\[ E_{\text{loss}} = (P_{\text{poss}}(V) - P_{\text{curr}}) \cdot t_{\text{timestep}} \]  

The equation above is used in the program to estimate the losses from icing and uses the possible power as a function of the wind speed minus the current power production times the time step. In Figure 7 it is possible to see that the power production during the warm period is centered along one curve compared to where the cold period is shown and the scatter is spread out over a much larger area.

*Figure 7: Scatter of power production during the warm period. (Hellström, 2013)*

![Scatter of power production during the warm period.](image)

*Figure 8: Scatter of power production during the cold period. (Hellström, 2013)*

![Scatter of power production during the cold period.](image)

The losses from icing can be measured indirectly in retrospect by the Wind power Evaluation Tool for Ice related Production losses (WETIP) program made by Erik Hellström for Vattenfall. WETIP uses measured atmospheric data produced by NASA, and the Modern Era Retrospective analysis for Research and Applications (MERRA). The MERRA data consist of the time, the mean wind speed, the direction of the wind and the temperature. These are measured for a point southwest of the wind farm, this is the closest point where there is MERRA data available.
The tool takes in SCADA data from every wind turbine and temperature data that comes from MERRA data, the MERRA data is used to density correct the wind speed (Hellström, 2013) This is done since the lower temperature gives higher wind density and thereby more energy in the wind, and thereby if the wind speed was not density corrected the energy in the wind would be larger in the cold air compared to that in the hot air. This means that the estimated losses from icing found by the program would be smaller compared to the actual losses.

The production data from the turbines comes in 10-minute averages and the losses data from the program comes in hourly averages. The conversion from 10 minutes to hours are made by summing up all the 10-minute average production data for the given hour and then dividing it with the number of instances that were recorded during that hour. This is due to the fact that the meteorological variables are available in hourly averages. The losses are then divided into ice losses and other losses through a temperature gate at 2°C, if the temperature is above this gate the occurring loss is considered to be due to other reasons and not from icing.

The output from the WETIP program is the losses for each wind turbine but to be able to use the output from the machine learning program on another wind farm close to this site the output has to be in average loss per wind turbine instead. This is done by taking the losses from each wind turbine in each instant and adding them together, and then the answer is divided by the number of available wind turbines.

The data for the production losses are filtered by Benjamin Martinez, when none of the wind turbines are sending production data, this is marked with a “not a number” (NaN) notation. The data set has then been corrected for the missing values.

### 3.4 Obtaining the relevant icing variables with the usage of mutual information

As input to the model that is going to be trained there is the metrological data from the WRF modelling in combination with the ice load from the IceBlade model being the most valuable. The target for the training is the actual losses time series from the WETIP.

To decrease the computational time, expense and level of difficulty of the models the number of inputs to the models has to be decreased. This is done by finding the amount of mutual information between the different variables and the output. The algorithm that is used to do this is a version of the MILCA algorithm for Matlab that has been modified by Cristian Guarnizo (Restrepo et al., 2014) in order to work with variable selection for fuel cells. The starting variables are shown in Table 1. Input variables

<table>
<thead>
<tr>
<th>Variable</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wind speed</td>
<td>m/s</td>
</tr>
<tr>
<td>Wind direction</td>
<td>Degrees</td>
</tr>
<tr>
<td>Temperature</td>
<td>ºC</td>
</tr>
<tr>
<td>Potential temperature</td>
<td>ºC</td>
</tr>
<tr>
<td>Atmospheric pressure</td>
<td>hPa</td>
</tr>
<tr>
<td>Relative humidity</td>
<td>%</td>
</tr>
<tr>
<td>Water content</td>
<td>kg/kg</td>
</tr>
<tr>
<td>Ice load</td>
<td>kg/m</td>
</tr>
</tbody>
</table>
The selected variables will be the ones used as input to the machine learning algorithms. The MILCA program goes through the variables one by one and calculates the mutual information between the variable and the target value. Then the program selects the variable that gives the highest value of the mutual information. Then the rest of the variables are tested against the set of the selected variable and the target value to see what variable that could add the most information to the selected set. This continues until the value of the mutual information does not increase any more. Then the program is stopped and the selected set of variables is displayed. To do this the program uses Eq.(11) with the 6 nearest neighbour as the k-value, as described in the 4th paragraph of section 2.8.

3.5 Machine learning

There are several different algorithms that have been tested by Carlos Restrepo (Restrepo, 2014): interpolations, linear and nonlinear autoregressive exogenous models and Box-Jenkins Model. These methods will be compared by Carlos to see which method that gives the best results. Carlos has been working on a Matlab program that tests the different approaches and that shows if any of them can be used in this application.

In the algorithm written by Carlos Restrepo (Restrepo, 2014), the long periods of zero losses in the target series that represent the summers, are filtered out using 4 alternative gates. These are that the ice load has to be above 0, the ambient temperature has to be below 0ºC, the relative humidity is above 99% or the atmospheric pressure is below 910 hPa. This is done in order to reduce the computational time and the complexity. (Restrepo, 2014)

After this the database is split into two sets, one set for training that consist out of two years’ worth of data and one set with the data for the last winter that is used as validation set.

3.6 The usage of a neural network

In the Matlab neural network toolbox there are 4 different types of networks that can be used that all have different applications. There is the clustering, the fitting, the pattern recognition and the time series. The one used in this thesis is the time series since the goal is to predict a future time series based on historical training data. Within the time series application there are 3 types of networks that could be used. They are presented in details below.

- The Nonlinear Autoregressive network predicts a time series when trained on past values from the same series. It predicts the losses from past values of the losses.

- Nonlinear Input-Output network uses the atmospheric inputs to train the network against the losses and then the network predicts the losses using the new inputs.

- Nonlinear autoregressive network with external inputs (NARX) trains in the same way as the input output network but it uses both the values from the inputs and the past outputs to predict the new value of the losses. The NARX network is a dynamic network.

There are also 3 different algorithms that can be used for the training of the network, as listed below.
- First is the Levenberg-Marquardt training that stops the training when the mean squared error (MSE) of the validation values of the training set does not improve any more.

- The second type of training is the Bayesian Regularization which is good to use on difficult and noisy datasets.

- The third type of training is the scaled conjugate gradient, this is a method that is supposed to use less memory but that was hard to get to work on the computer used.

The neural network that was used in this study is a NARX network and the training that was used was Bayesian Regularization in the Matlab Neural Network toolbox. Here the time series was also split into two parts, the training and the validation set, with the training as the two first years and the validation as the last year. Of the two years’ data that was used for training 70% was actually used to train on and 15% was used to test on and the last 15% was used as validation in the toolbox.

**Figure 9:** Neural network with 2 input variables, 1 target, 4 delays and 10 nodes.

The network can be constructed in different ways with different numbers of nodes and delays. Different designs of the network are tested with 4, 8, 10, 15, 20 and 25 nodes, all the different configurations of nodes it is trained with 1-7 delays. The networks are trained with the help of the data from the first 2 years and then when the networks have finished training the networks are tested against the validation year to see what design of network that is most promising to use in the future. The years are shown in Table 2.

The training is carried out by the program by comparing the output from the neural network to the target value for that instant. If the values do not match the program changes the weights and does the iteration again.

**Table 2. Years used for training and validation**

<table>
<thead>
<tr>
<th>Training data</th>
<th>2011, 2012</th>
</tr>
</thead>
<tbody>
<tr>
<td>Validation data</td>
<td>2013</td>
</tr>
</tbody>
</table>
When the network is tested against the validation year this is done with a closed loop network which is shown in Figure 10. In the closed loop network the earlier outputs from the network are used as inputs to the network compared to Figure 9, where the earlier target inputs are the correct values from the measured series.

**Figure 10:** Closed loop NARX network.
4 RESULTS

4.1 Results from the new ice model

The output from the updated Ice load model shows a much higher icing rate than the earlier model from Vattenfall as shown in Figure 11.

*Figure 11: Comparison between old and new ice load model*

![Comparison between old and new ice load model](image)

To further elaborate on the differences between the two models a zoom in of January is shown in Figure 12 that presents the difference in both the accretion and the decreasing of the icing. The graph shows the more rapid accretion in the new ice model and also the rapid ablation compared to the old model for the selected time scope.

*Figure 12: Zoom in of January*

![Zoom in of January](image)

The shape of the old and the new icing model are quite different, but the instances of icing are often concurrent. The new icing model shows a significantly more rapid accretion of ice, this is likely due to the fact that the old model simulates the wind turbine as a cylinder that stands still and the new model simulates it as a cylinder rotating at 85% of the blade length. This means that the new model will meet the wind at a significantly higher speed than the old one. Since the wind speed has a strong effect on the amount of icing on the turbine this is a very likely reason for the increased amount of ice on the blade.
4.2 Energy losses from icing

The impact of ice shedding in the physical model was underestimated, most of the icing events are picked up by the model but the icing stays on the turbine too long. This is due to the fact that the only decrease in icing on the turbine when the temperature is below 0°C is sublimation, which is very small compared to the accreted amount of icing. Severe shedding can be seen on pictures for example between 2013-01-06 05:00 and 2013-01-06 10:00, this is at times when both the actual and potential temperature is below 0°C. The losses ends already at 2013-05-01 00:00 which is a lot earlier, but this can be due to the fact that there is data from different numbers of wind turbines available at different times. Even though there are no losses shown from the production data that have been analysed with the WETIP program it is still possible to see icing on the wind turbines that are recorded. During the period 2013-02-02 17:00 to 2013-02-04 02:00 there are 0 losses according to the production data, but according to the visual images from the cameras there is icing during this entire period. The resulting losses from the WETIP program can be seen in Figure 13.

![Figure 13: Losses plotted over the ice load.](image)

4.3 Mutual information

When running the mutual information least dependent component algorithm MILCA, it is found that variables that have the most influence on the target series are the ones shown in Table 3.

Variables selected through mutual information

<table>
<thead>
<tr>
<th>Ice load</th>
<th>Temperature</th>
<th>Atmospheric pressure</th>
<th>Wind direction</th>
</tr>
</thead>
</table>

These variables are the ones used in the method selection process.
4.5 Model validation using neural network

To find the configuration of the neural network that gives the best result and the lowest error different configuration are tested. First the different configurations described in section 3.6 are trained as described in the same section. When the network has been trained with the first two years of data the data of the last year is fed to the program and is used as a validation of the training. The result of this validation is presented in the figures below.

When running the neural networks several different sizes of networks with different delays were tested. This gives different values for the root mean squared error (RMSE), this is shown in Figure 14.

**Figure 14: Root mean squared error for the validation**

Here it can be seen that the best value of the RMSE is found when the network is as simple as possible. This is found in the points 4 nodes with 2 delays where the RMSE is 43.77 kWh and 10 nodes with 2 delays where the RMSE is 44.2 kWh.

When looking at the correlation between the target values and the outputs from the program a similar picture is seen. The largest correlations are at 4 nodes and 1, 2, and 3 delays and also at 10 nodes with 2 delays. This is shown in Figure 15 further below.

The small and simple networks are quicker to train, on average a couple of minutes or less, but the more advanced networks can take more than 1.5 hours to train until reaching acceptable validation results.
These two comparisons showed that the interesting points to look closer at is 4.2 and 10.2 since these are the points where the RMSE is the lowest at the same time as the correlation is the highest.

In Figure 16 it is shown that the error between the targets and the outputs is quite large at several instances. But one interesting thing to note is that when the errors are the largest there is a negative error with a size near the one of the actual error. This can better be understood when looking at Figure 17, as shown below.
Figure 17: Output and target values with 4 nodes and 2 delays.

In Figure 17 is possible to see that the targets and outputs follow each other in the large sense, but there are some instances where the errors are very large, for example in the beginning of September where the outputs are below zero, which is practically not possible. At the same time the errors are also quite large as seen in the same figure. The outputs follow the same trends as the targets when above zero, but with a certain delay.

Figure 18: Zoom in on the beginning of October from Figure 17.

When looking at Figure 18 it is possible to notice that the output values follow the general shape of the target curve with comparatively high accuracy, but there is often a small delay in the outputs compared to the targets. Also the network tends to over- and underestimate the values of the losses when there are sharp turns or abrupt changes of the target values.
**Figure 19:** Difference between the target values and the outputs for 10 nodes with 2 delays.

In Figure 19 the differences in the case of 10 nodes and 2 delays are displayed, here a pattern similar to the one in Figure 16 can be seen.

**Figure 20:** Output and target values with 10 nodes and 2 delays.

In Figure 20 it is possible to see that the target and the output values follow each other in the same way as in the case with 4 nodes 2 delays but the network does not estimate values below zero in the same extent.
In Figure 21 it is noted that the outputs also follow a similar pattern as the target values. This also gives a good indication of that even though there is an error in almost all of the output points, the correlation is still very accurate and should lead to a possibly acceptable result.
5 DISCUSSION AND CONCLUSIONS

5.1 Discussion

The hereby evaluated IceBlade model results in a very large and fast accretion of ice compared to the old ice model from Vattenfall. This is due to the difference in the relative speed between the wind and the different models. Since the old model only simulates the wind turbine as a cylinder at a steady place, it will only be exposed to the air at the actual wind speed. The new model that simulates the turbine blade as a cylinder rotating at 85% of the blade length will meet the air at much higher speeds and be exposed to more of the air mass. The large difference between the two wind speeds explains the large difference in the models.

The IceBlade model does not consider shedding at temperatures below 0.5 degrees. This could be a problem for the actual implementation and needs to be further analysed, since shedding can occur at different times and often also at freezing temperatures because of flexing of the blades or loss of adhesion.

The author of this study was forced to run the model evaluation against only overall and generalised data offered by Vattenfall regarding the performance of their old in-house ice model. Proper comparison and assessment of the economic implications by applying the new IceBlade model in practice could not be accomplished due to the very limited information that Vattenfall handed over to the author, neither could any sensitivity analysis be done, without which it is impossible to predict any proper monetary advantage that the new model can achieve.

To further improve the physical model the physics behind ice shedding would have to be studied, the problem being that shedding is highly dependent on the surface structure and finishing of the turbine blades. This would mean that the shedding would have to be modelled for the specific type of turbine that is of interest. This could be done but at an expense, however, it could be worth it when one knows the particular type of blades that are installed on the wind turbines under consideration.

Since there are instances where the WETIP program claims that there is no icing occurring on the turbine blades while at the same time there is a considerable icing process going on that can be clearly seen by visual inspection, there is a possibility that the target series that are taken from the WETIP program have an inbuilt error inside. This would certainly introduce an error into the simulation result and would negatively affect the overreaching output quality of the program. Unfortunately, the reason for these detrimental effects neither the definition of a plausible solution strategy were not possible to be followed up on within the present study, due to the lack of specific data offered by the project commissioner Vattenfall.

The production data from the turbines that goes into the WETIP program comes in 10 minutes average but then is converted into hourly averages. During this process some of the information is lost and again an error is forced into the target values. This could introduce an error in the neural network training which could be an explanation to a part of the output error.

It is interesting to see that neither the relative humidity nor the water content of the air is selected by the MI algorithm to be used as input to the machine learning, since both variables are strongly affecting the icing of structures. This is explained by the fact that both variables are accounted for in the IceBlade model and thereby the amount of mutual information between the variables and the target does not increase when investigating these two variables.
Since wind direction is chosen as a variable the results of this thesis are very specific to the site of the investigated wind farm. Thus it would be important to make a further comparison with the results made when the wind direction was included. However, the work presented in this thesis did not manage to investigate these aspects due to the limited time available. It could be very interesting to run a sensitivity analysis between different wind parks with and without the wind direction as a variable, in order to closely study how much the wind direction affects the outcome of the program.

As seen in the results the output from the network follows the target curve very well in most cases when a simple network is used, and when a more complex network is used the result gets worse. This could mean that it is possible to make a simple physical model from the selected inputs; especially the output from the IceBlade model would be very interesting to follow up.

The point with 10 nodes and 7 delays looks interesting since the MSE gets lower even though the network gets more complicated compared to 10 nodes with fewer delays, but when looking at the correlation value it can be seen that it goes down and thereby it was not investigated any further. Since the results from the training of the neural network can vary between the different runs, depending on what starting value the nodes have, it could be interesting to see how much the results would vary. This has not been possible in this thesis due to time limitations but it could be interesting for future studies.

The type of network used is based on the discussion with Carlos Restrepo but it would be interesting to investigate how the network could be improved if it was constructed outside of the Neural Network toolbox and coded directly in Matlab instead since it then offers more options available on how to design the network.

It would be very interesting to compare the results from this thesis to prognoses from studies that have larger data sets available for training to see how the larger amount of training data and different design of the network influence the outputs.

5.2 Conclusions

The results from the analysis and comparison of neural networks are interesting and the method looks promising and worth investigating. If it were possible to achieve smaller errors in the input and target data it would probably be possible to get better results. This could also be the case with hand-programmed networks.

If these areas are investigated in depth and if there are more years of training data available for the analysis, the author firmly believes that it would be possible to forecast the losses from icing in a good way using the neural network method. Furthermore, the studied approach would fulfil its promise for a simpler practical implementation offering better prediction and consecutively much better financial results in the long run.
6 FUTURE WORK

6.1 Future work

It would be necessary to run a deeper analysis of the shedding of the ice from the turbine blades and implement this knowledge in the IceBlade model to get a better value of the actual ice load variation on the blades.

For avoiding the error introduced by the averaging of the production data it would be important to get the meteorological data in 10-minute averages as well as to see if this could produce a difference in the target values. It could also be interesting to test different types of averaging algorithm and to do a sensitivity study in order to see how dependent the output is on the type of averaging.

It would be necessary to further investigate the WETIP program with regards to the errors seen in the losses generated in the program that does not match the visual inspection data and photos from the wind farm at certain particular moments.

To force the neural network to generalize between different wind parks it would be interesting to run the test without the wind direction input and thereby make it less dependent on the specific conditions of the Stor-Rotliden wind park that were used for this particular analysis.

On the other hand, to make the result even more specific to the site and probably get better and more detailed results it could be interesting to look at each wind turbine individually and train a network for each turbine. The problem here would be the meteorological data that does not have that good resolution so it is a question if the results would improve at all because of this. However, such an approach is certainly worth of investigation, provided there is reliable input data that can be utilized.

To investigate more in detail about the different types of neural networks that could be coded directly in Matlab without using the toolbox, it would be interesting to analyse what kind of results that could be achieved with the other options that are available when the code is done by hand instead of via the toolbox.


Martinez, Benjamin. “Personal communication, Wind engieneer Vattenfall R&D.” n.d.


